Honours Research Reports

WST795/STK795

Department of Statistics, University of Pretoria

2015

Booklet compiled by IN Fabris-Rotelli

Time series analysis of the South African gross domestic product

Larúchelle de Almeida 10688553

STK795 Research Report

Submitted in partial fulfilment of the degree BCom(Hons) Statistics

Supervisor: Mrs. J van Niekerk

Department of Statistics, University of Pretoria

2 November 2015

Abstract

To measure the performance of a country's economy, it is preferred to use the gross domestic product (GDP) index. The analysis of GDP is carried out by adopting a relevant time series model. However, the stationarity of this model plays an important role in forecasting. For the purpose of identifying an accurate time series model to analyse the Real GDP of South Africa, we will be testing whether the time series model for the Real GDP is stationary for the period of 19 years, i.e from 1995 to 2014. In this respect, we briefly review some methods of measuring the stationarity of a time series model and apply relevant methods to the data set.

Declaration

I, *Larúchelle de Almeida,* declare that this essay, submitted in partial fulfilment of the degree *BCom(Hons) Statistics* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Larúchelle de Almeida

_____
Janet van Niekerk

_____
Date

Contents

List of Figures

List of Tables

1 Introduction

Gross domestic product (GDP) is a measure to describe how well a country's economy is performing. The main purpose of GDP is to summarize different types of data into one single monetary value to represent economic activity for a given period of time [12]. GDP can be calculated by three methods namely: the income method, production method and the expenditure method. The expenditure method is most frequently used, as it is the summation of all spending on final goods and services by households, businesses and government that have occurred in a given year [22].

The GDP will be analysed through time series analysis, primarily focusing on whether GDP is a stationary time series model and the consequence thereof. Modelling the GDP correctly whether stationary or non-stationary will have important implications on forecasting, testing and macroeconomic policy [7]. One such important implication is in correctly determining business cycles, as portfolio managers and investors use business cycles to position their investments correctly [11].

Various statistical and mathematical procedures will be used to determine whether South African GDP exhibits stationary or non-stationary characteristics. The Dickey-Fuller test will be applied to see if the GDP process possesses any unit roots, if the process has a unit root then it is a non-stationary process [8]. Further analysis will depend on the findings of stationarity.

Based on findings the appropriate autoregressive-moving average (ARMA) model will be applied (in case of a stationary process). Should it otherwise be observed that the GDP process is non-stationary, it will be determined if the trend or difference stationary model best fits.

In the following section certain concepts about the GDP and time series will be explained and the application will follow to determine the time series characteristics of the South African GDP. There has not been sufficient research especially on the South African GDP regarding stationarity. However, there has been in-depth research on numerous other countries' GDP, which has found that their GDP is non-stationary. In this research report it will be concluded whether the South African GDP exhibits stationarity.

2 Background Theories

In this section certain key concepts will be explained and reviewed to better understand the background of GDP and time series.

2.1 Gross Domestic Product

The GDP is a very important financial indicator when *analysing* a country's economic output. Below it will be explained what GDP is, how to calculate it and the importance thereof.

Definition 1. The GDP is defined as the total market value of all final goods and services produced annually within the boundaries of a country, whether by the country itself or foreign-supplied resources [22].

The expenditure method will be used in the analysis and is the most widely used method for calculating GDP and can be formulated as follows:

$$GDP(E) = C + I + G + (X - Z)\,, \tag{2.1}$$

where

- C = Final consumption expenditure by households
- I = Gross capital formation
- G =Final consumption expenditure by general government
- X = Exports of goods and services

- Z = Imports of goods and services

There are however two different categories in reporting GDP, namely nominal GDP and real GDP. Nominal GDP is based on the prices that prevailed when the output was produced [14], whereas real GDP reflects changes in price levels. GDP is deflated or inflated to reflect changes in price levels. The relationship between nominal and real GDP is best explained by (2.2).

$$Real\,GDP = \frac{Nominal\,GDP}{GDP\,Deflator} \tag{2.2}$$

	Nominal GDP (R millions)	GDP Deflator (2005=100)	Real GDP (R millions) (constant 2005 prices)
2003	1 272 537	89.15	1 427 332
2004	1 415 273	94.84	1 492 330
2005	1 571 082	100.00	1 571 082
2006	1 767 422	106.53	1 659 121

Table 1: Calculating Real GDP

***Source**: SARB Quarterly Bulletin*, March 2010

When calculating the Real GDP using (2.2), it is necessary to select a base year. In table [1] , 2005 was chosen as the base year and the GDP deflator for the base year will always be 100 [14]. Real GDP provides a better picture of the actual economic activity in a country as prices of goods are adjusted for inflation. Once a series of figures is collected over time, the figures can be compared and economists can determine business cycles. Business cycles play a key part in establishing when an economy is expanding or contracting. Business cycles are usually associated with economic instability and have four stages; peaks, recessions, troughs and expansions which usually proceed in that order [12]. With this information the country's economic status can be determined. To know whether a country is economically stable and has a positive growth rate is especially important for domestic and foreign investors. Any investor would want to know if the country he or she is investing in will deliver positive real returns.

2.2 Time series

A time series is broadly defined as a record of values of any sporadic variable measured at different time points. An important feature of time series is that in a majority of time series there are values recorded at different points in time that are partly influenced by some random mechanism [8]. There are two widely known fundamental assumptions for statistical analysis of time series; namely that (i) a series is *stationary,* if not it can be transformed into a stationary series and (ii) the series converges to a linear model [18].

2.2.1 Stochastic Process

A stochastic process is defined as a collection of random variables$\{W_t : t \epsilon T\}$ where t is the indexing parameter and some element of the set T, the parameter space. Now suppose we have some random variable W_t, of which we have an observed sample of size T, then the sample is given by:

$$\{w_1, w_2, w_3, \ldots, w_T\} \tag{2.3}$$

For example, consider a collection of T independent and identically distributed (i.i.d) variables ε_t,

$$\{\varepsilon_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_T\}, \tag{2.4}$$

with

$$\varepsilon_t \sim N\left(0, \sigma^2\right).$$

Representation (2.4) is referred to as a sample of size T from a *Gaussian white noise* process, which is defined as

$$E(\varepsilon_t) = 0 \tag{2.5}$$

$$E(\varepsilon_t^2) = \sigma^2 \tag{2.6}$$

$$E(\varepsilon_t \varepsilon_\iota) = 0 \quad \text{for} \quad t \neq \iota \tag{2.7}$$

The observed sample (2.3) represents only one possible outcome of the underlying stochastic process that generated data. The observed sample can be observed for an infinite period of time and can be displayed in sequence

$$\{w_t\}_{t=-\infty}^{\infty} = \{..., w_{-t}, w_0, w_1, ..., w_T, w_{T+1}, w_{T+2}, ...\}.$$

The random variable, w_t, also has some density function, denoted $f_{W_t}(w_t)$, which is called the unconditional density function of W_t, is given by

$$f_{W_t}(w_t) = \frac{1}{\sigma\sqrt{2\pi}} exp\left[\frac{-w_t^2}{2\sigma^2}\right].$$

For example, if $\{W_t\}_{t=-\infty}^{\infty}$ represents the sum of a constant μ plus a *Gaussian white noise process* $\{\varepsilon_t\}_{t=-\infty}^{\infty}$ [8], then

$$W_t = \mu + \varepsilon_t, \tag{2.8}$$

with expected value

$$E(W_t) = \mu, \quad E(\varepsilon_t) = \mu. \tag{2.9}$$

If W_t is a *time trend* plus a Gaussian white noise, it follows

$$W_t = \beta t + \varepsilon_t \tag{2.10}$$

and the expected value is

$$E(W_t) = \beta t. \tag{2.11}$$

For emphasis the expectation $E(W_t)$ is called the *unconditional mean* of W_t to allow for the possibility that the mean can be a function of time t. The unconditional mean is denoted by μ_t:

$$E(W_t) = \mu_t \tag{2.12}$$

The constant mean, μ (2.9) , is not a function of time, while the time varying mean, μ_t (2.12), is a function of time.

2.2.2 Stationarity

Many time series in macroeconomics and finance are non-stationary, the precise form of the non-stationarity is a widely debated topic [9].

Definition 2. The j^{th} autocovariance, γ_{jt}, of a stochastic process W_t is the covariance between the value at time t and $t-j$, where $j < t$. The stochastic process W_t is said to be stationary if neither the mean μ_t, nor the autocovariance γ_{jt} depend on time t, then the process W_t is said to be *covariance-stationary* or *weakly stationary* [8] :

$$E(W_t) = \mu \quad \text{for all t}$$

$$E(W_t - \mu)(W_{t-j} - \mu) = \gamma_j \quad \text{for all } t \text{ and any } j$$

Thus (2.8) is stationary whereas the process (2.10) is non-stationary since its mean, βt, is a function of time. Note that if a process is covariance-stationary, the covariance between W_t and W_{t-j} depends only on j, the length of the interval. A process is only weakly stationary if all the W_t's have the same mean values and if the process is covariance stationary. Another concept to take note of is *strict stationarity*.

Definition 3. A process is said to be strictly stationary if, for any values of $j_1, j_2, \ldots, j_n$ the joint distribution of $(W_t, W_{t+j_1}, W_{t+j_2}, \ldots, W_{t+j_n})$ depends only on the intervals separating the dates $(j_1, j_2, \ldots, j_n)$ and not the date itself (t).

In this article when it is said a process is stationary it is taken to mean a covariance-stationary process. The simplest generated time series is a white noise process, denoted by w_t which consists of independent random variables. The Gaussian white noise process (2.8) is said to be covariance-stationary when

$$\begin{aligned} E(W_t) &= \mu \\ E(W_t - \mu)(W_{t-j} - \mu) &= \begin{cases} \sigma^2 & \text{for } j = 0 \\ 0 & \text{for } j \neq 0 \end{cases} \end{aligned}$$

By contrast the process of (2.10) is not covariance stationary, since it is a function of time.

A non-stationary time series arises more than one would expect. To explain non-stationarity, a univariate time series model will be used that can be written in the following form:

$$w_t = \mu + \varepsilon_t + \psi_1 \varepsilon_{t-1} + \psi_2 \varepsilon_{t-2} + \cdots = \mu + \psi(L)\, \varepsilon_t \tag{2.13}$$

where $\sum_{j=0}^{\infty} |\psi_j| < \infty$, the roots of $\psi(L) = 0$ are outside the unit circle, and $\{\varepsilon_t\}$ is a Gaussian white noise sequence with mean zero and variance σ^2. The expected value of w_t is constant and the forecast converges to the unconditional mean, μ. These are unappealing assumptions for many economic time series encountered in the real world [8]. There are two approaches to trends:

The first is the *unit root* process,

$$(1 - L) w_t = \delta + \psi(L) \varepsilon_t \tag{2.14}$$

where $\psi(1) \neq 0$ and $(1 - L)$ is the *first-difference* operator. The first-difference operator will further be indicated by the Greek symbol Δ.

$$\Delta w_t \equiv w_t - w_{t-1}.$$

A stationary representation of the univariate time series (2.13) for a unit root process describes changes in the series. The classical example of a unit root process is achieved when setting $\psi(L)$ equal to 1 in (2.14):

$$w_t = w_{t-1} + \delta + \varepsilon_t \tag{2.15}$$

where the mean of Δw_t is denoted by δ rather than μ. The process (2.15) is known as a *random walk with drift* δ. A unit root process is also widely known as a *difference-stationary* process.

The second type of approach in describing a trend is to include a *deterministic time trend,* in (2.13):

$$w_t = \alpha + \delta t + \psi(L)\, \varepsilon_t. \tag{2.16}$$

The mean μ of the stationary process (2.13) is replaced by a linear function of the time t. This process is also known as *trend stationary*. When the trend δt is subtracted from (2.16) the result is a stationary process [8].

2.2.3 Linear Time Trend vs. Exponential Time Trends

The *deterministic time trend* (2.16) is specified as a linear function of time (δt) rather than a quadratic ($\delta t+\gamma t^2$) or exponential ($e^{\delta t}$) function of time. This is in contrast to time trends seen in economic and financial time series. It is widely noted that economic and financial time series trends will be better characterised by exponential trends than linear trends. A key characteristic of exponential growth is that it exhibits constant proportional growth, if

$$\begin{aligned} w_t &= e^{\delta t} \\ dw/dt &= \delta.w_t\,. \end{aligned} \tag{2.17}$$

Economists simply assume growth as exponential growth, as it is often confirmed by visual inspection of the time series. By taking the natural logarithm of the exponential trend, it reduces to a linear trend (2.17),

$$log(w_t) = \delta t\,.$$

Thus, it is common to take the natural logarithms of the data before attempting to describe them [8].

2.2.4 ARMA(p,q) Process

In this section a more general class of ARMA(p,q) processes is investigated. Note that the ARMA(p,q) process is found by joining two time series processes AR(p) and MA(q). It is complex to detect a pure AR(p)- or MA(q) process by the behaviour of its observed autocorrelation and partial autocorrelation functions, because neither decreases with increasing lag order [17]. First the ARMA(p,q) process will be explained separately by the two time series processes of which it is created.

AR models are based on the idea that the current value , w_t, of the series can be explained as a function of p past values, $w_{t-1}, w_{t-2}, \ldots, w_{t-p}$[21]. The degree to which it might be feasible to forecast a data series can be assessed by analysing the autocorrelation function (ACF) and partial autocorrelation function (PACF). The preceding section motivates the following definition.

Definition 4. An autoregressive model of order p, abbreviated AR(p), is of the form

$$w_t = \Phi_1 w_{t-1} + \Phi_2 w_{t-2} + \cdots + \Phi_p w_{t-p} + \varepsilon_t \tag{2.18}$$

where w_t is stationary, $\Phi_1, \Phi_2, \ldots, \Phi_p$are constants ($\Phi_p \neq 0$) and ε_t is a Gaussian white noise process with a mean of 0 and variance of σ^2. If the mean of $w_t$is not 0, the process can be rewritten as

$$w_t = \alpha + \Phi_1 w_{t-1} + \Phi_2 w_{t-2} + \cdots + \Phi_p w_{t-p} + \varepsilon_t \tag{2.19}$$

where $\alpha = \mu(1 - \Phi_1 - \cdots - \Phi_p)$.

To explain the requirements for the AR(p) process to be stationary, (2.19) with the lag operator L, is rewritten as

$$(1 - \Phi_1 L - \Phi_2 L^2 - \cdots - \Phi_p L^p) w_t = \alpha + \varepsilon_t \tag{2.20}$$

It can then be shown that an AR(p) process as in (2.19) is stationary if all roots $z_0$of the polynomial

$$\Phi_p(z) = 1 - \Phi_1 z - \Phi_2 z^2 - \cdots - \Phi_p z^p$$

have a modulus greater than one. Theoretical partial autocorrelation, Φ_{kk}, can be used to determine whether the AR process is appropriate for the observed series and to select the order of the process, in effect the value of p. This can be done by testing

$$\begin{array}{lll} H_{0:} & \Phi_{kk} = & 0 \\ H_1: & \Phi_{kk} \neq & 0 \end{array} \text{ for } k=1,2,\ldots$$

and that for any AR(p) process it can be shown that $\Phi_{kk} = 0$ for $k > p$ where the lags are indicated by k [4].

As an alternative to the AR representation where w_t, in (2.18), is assumed to be linearly combined, whereas with the MA(q) process, it is assumed that the white noise ε_t is linearly combined to form the following definition.

Definition 5. The moving average model of order q, MA(q) model, is defined to be

$$w_t = \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \cdots + \theta_q\varepsilon_{t-q} \tag{2.21}$$

where there are q lags in the moving average and $\theta_1, \theta_2, \ldots \theta_q$ ($\theta_q \neq 0$) are parameters. The noise ε_t is assumed to be Gaussian white noise.

Again (2.21) can be rewritten to include its mean, giving the general representation of MA(q) as

$$w_t = \mu + \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \cdots + \theta_q\varepsilon_{t-q}. \tag{2.22}$$

With the lag operator L, this process can be rewritten as

$$\begin{array}{rcl} w_t - \mu & = & \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \cdots + \theta_q\varepsilon_{t-q} \\ & = & (1 + \theta_1 L + \theta_2 L^2 + \cdots + \theta_q L^q)\varepsilon_t. \end{array} \tag{2.23}$$

Theoretical autocorrelation, θ_k, can be used to determine whether the MA process is appropriate for the observed series and to select the order of the process, in effect the value of q. This can be done by testing

$$\begin{array}{rcl} H_0: & \theta_k = & 0 \\ H_1: & \theta_k \neq & 0 \end{array} \text{ for } k\text{=}1,2,\ldots$$

and that for any MA(q) process it can be shown that $\theta_k = 0$ for $k > q$ [4].

Combining (2.18) and (2.21), the general form of a ARMA(p, q) process can be illustrated as

$$w_t = \mu + \Phi_1 w_{t-1} + \Phi_2 w_{t-2} + \cdots + \Phi_p w_{t-p} + \varepsilon_t + \theta_1\varepsilon_{t-1} + \theta_2\varepsilon_{t-2} + \cdots + \theta_q\varepsilon_{t-q}. \tag{2.24}$$

The first step in model building is to determine the order of p and q. The sample autocorrelation function and partial sample autocorrelation function are used as indicators of the values p and q.

2.2.5 Implications of external shocks

The one key difference between trend-stationary and difference-stationary processes is in the impact of a shock, both in its persistence and magnitude [16]. In a trend-stationary process, an external shock has a transient effect meaning the series will return to its mean (mean reverting) after a sufficient period of time has passed. In contrast, when the difference-stationary process experiences a shock, the shock is permanently incorporated into the time series, thus it has a permanent effect. The effects can easily be indicated by rewriting (2.14) in the following form,

$$(1 - \rho L)w_t = \varepsilon_t. \tag{2.25}$$

If $|\rho| < 1$, then w_t comprises a linear trend subject to deviations that are transient in their impact; this is a simple form of a trend-stationary model in which deviations about the trend are stationary. The closerρ is to positive 1, the more long-lasting are the shocks [18].

2.2.6 Stationarity tests

As mentioned above and argued by Nelson and Plosser [15], many macroeconomic time series are non-stationary and are more appropriately described by unit root non-stationarity than by deterministic trends. When determining whether the process is stationary or not, the order of integration must also be determined. Once it is determined if the process does indeed have a unit root, indicating it is a non-stationary process, the process can be reduced to stationarity by differencing. There are numerous tests that can be used to determine if a process is stationary or not, however, only the Augmented Dickey-Fuller (ADF) test and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test will be used to test for unit roots and stationarity. The Phillips & Perron (PP) test and other tests can also be used, but The ADF and KPSS tests have been found to be more effective.

The ADF test was proposed by Said & Dickey [20] to determine if any unit roots in ARMA models of unknown orders p and q exist. The Dickey-Fuller test controls for serial correlation by including higher-order autoregressive terms in the regression [5] and the ADF test statistic is a *t-statistic*. The null and alternative hypotheses for ADF are as follows, where the integrated order is denoted by I:

$$H_0 : \textit{The process is stationary}$$

$$H_1 : \textit{The process is non} - \textit{stationary}$$

Hence, the null hypothesis states that the process is non-stationary (since it has a unit root) is tested against the alternative hypothesis that the process is stationary (no unit root). There are three cases which are considered for the Dickey-Fuller tests [8],

- Case 1 - An ARMA process with a zero mean
- Case 2 - An ARMA process with a single mean
- Case 3 - An ARMA process with a trend.

Recall that W_t is a stochastic process and is only stationary if it satisfies the conditions of definition 2. Consider an AR(1) model with the first ordinary least squares (OLS) estimation of Φ,

$$w_t = \Phi w_{t-1} + \varepsilon_t, \tag{2.26}$$

where ε_t is $i.i.d$ with mean 0 and variance σ^2. The OLS estimate is given by

$$\hat{\Phi}_T = \frac{\sum_{t=1}^{T} w_{t-1} w_t}{\sum_{t=1}^{T} w_{t-1}^2}. \tag{2.27}$$

There are two possible test statistics that can be used [17]. The first test statistic is the regression coefficient-based test statistic:

$$\Phi = T(\hat{\Phi}_T - 1) \tag{2.28}$$

where n is the sample size. The second test statistic is the studentised test statistic:

$$\tau_T = \frac{\hat{\Phi}_T - 1}{\hat{\sigma}_{\hat{\Phi}_T}} = \frac{\hat{\Phi}_T - 1}{\left\{ s_T^2 \div \sum_{t=1}^{T} w_{t-1}^2 \right\}^{\frac{1}{2}}}, \tag{2.29}$$

where $\hat{\sigma}_{\hat{\Phi}_T}$ is the OLS standard error for the estimated coefficient,

$$\hat{\sigma}_{\hat{\Phi}_T} = \left\{ s_T^2 \div \sum_{t=1}^{T} w_{t-1}^2 \right\}^{\frac{1}{2}},$$

and s_T^2 denotes the OLS estimate of the residual variance [8]:

$$s_T^2 = \frac{\sum_{t=1}^{T}(w_t - \hat{\Phi}_T w_{t-1})^2}{(T-1)}.$$

The ADF test however fails in rejecting the null hypothesis of difference stationary processes. This could be due to the low power of such a test against the actual (non-stationary) data generating process rather than the acceptability of the presence of the unit root [13].

The KPSS test is used to test the null hypothesis that an observed time series is stationary around the deterministic trend. The KPSS test is intended to harmonise unit root tests, by testing both the unit root hypothesis and the stationarity hypothesis [1]. The null and alternative hypotheses for the KPSS test are as follows,

$$H_0 : \textit{The process is stationary}$$

$$H_1 : \textit{The process is non} - \textit{stationary}$$

which states the null hypothesis is a stationary process around a linear trend and the alternative assumes the process is non-stationary due to the presence of a unit root [10]. The KPSS and the ADF hypotheses are direct opposites, as it can be seen above. Testing both stationarity and the presence of a unit root helps distinguish between a process that appears to be stationary, has unit roots or is insufficient to confirm stationarity or not.

2.2.7 ARCH(q) and GARCH(q,p)

Heteroskedasticity means that the error terms are mutually uncorrelated, while the variance is non-constant. In financial time series volatility clustering is often observed, volatility clustering can be defined as periods of stability and instability tend to cluster together.An example of this would be stock markets which are typically characterised by periods of high and low volatility.

The Autoregressive Conditional Heteroskedasticity (ARCH) model was introduced almost 30 years ago by Engle [6]. ARCH can be simply explained by the variance of the error term at time t, which depends upon the squared error terms from preceding periods [23]. If a random variable y_t is drawn from the conditional density function $f(y_t|y_{t-1})$, the forecast of today's value is based upon the past information [6] and the variance of this one period forecast is given by $\sigma^2(y_t|y_{t-1})$. While the ARCH process allows for variance to fluctuate over time as a function of past errors, the traditional time series and econometric models operate under the assumption of constant variance, meaning the conditional variance does not depend on preceding periods [2].

Engle [6] , however, proved the usefulness of conditional variance dependent on y_{t-1} in economics and also stated that heteroskedasticity corrections are difficult and are rarely used in time series data. The preferred model for explaining ARCH(q) can be expressed in terms of y_{t-1}, the information set available (consisting of all information available) at time $t-1$ with the assumption of normality,

$$y_t|y_{t-1} \sim N(0, h_t), \tag{2.30}$$

$$y_t = \varepsilon_t h_t^{\frac{1}{2}}$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i y_{t-i}^2 \tag{2.31}$$

where ε_t denotes a real-valued discrete-time stochastic process with $\sigma^2(\varepsilon_t) = 1$.

In applications, the ARCH model has been replaced by the generalised ARCH (GARCH) model proposed by Bollerslev [2], allowing for a more adaptable lag structure. The ARCH process was extended to GARCH to also include a prolonged memory. To avoid negative variance parameter estimates, a fixed lag is imposed [2]. The GARCH(p, q) process is then given by

$$\varepsilon_t|\psi_{t-1} \sim N(0, h_t) \tag{2.32}$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{p} \beta_i h_{t-i} \tag{2.33}$$

where

$$p \geq 0, \quad q > 0$$

$$\alpha_0 > 0, \quad \alpha_i \geq 0, \quad i = 1, \ldots, q,$$

$$\beta_i \geq 0, \quad i = 1, \ldots, p.$$

For $p = 0$ the process reduces to the ARCH(q) process which is similar to an AR(p) process, and for $p = q = 0$, ε_t is simply white noise. In the GARCH(p, q) process lagged conditional variances are allowed to enter, whereas the conditional variance of an ARCH(q) process only specifies a linear function of past sample variance [2].

2.2.8 Forecasting

The primary objective of constructing a time series model is to enable future forecasting of the specified series. It is also important to assess the precision of those forecasts. We will assume that w_t, the observed data, is stationary and that the parameters are known [21]. There are numerous forecasting methods that apply to deterministic trends and ARIMA models. To understand forecasting in depth one can refer to [4] or [21]. However the basic concepts of forecasting will be explained in this section by using a basic AR(1) model, rewriting (2.19) with $p = 1$. Forecasting can be illustrated and expanded in many ways with the simple AR(1) process with a non-zero mean that satisfies

$$w_t - \mu = \Phi(w_{t-1} - \mu) + \varepsilon_t \tag{2.34}$$

Consider the basic problem of forecasting one time unit into the future, by replacing t with $t+1$ in (2.34),

$$w_{t+1} - \mu = \Phi(w_t - \mu) + \varepsilon_{t+1}\,. \tag{2.35}$$

The conditional expectations are then taken on both sides of (2.35), given $w_1, w_2, \ldots, w_{t-1}, w_t$, to obtain

$$\hat{w}_t(1) - \mu = \Phi[E(w_t|w_1, w_2, \ldots, w_{t-1}, w_t) - \mu] + E(\varepsilon_{t+1}|w_1, w_2, \ldots, w_{t-1}, w_t)\,. \tag{2.36}$$

From the properties of conditional expectation [4], we have

$$E(w_t|w_1, w_2, \ldots, w_{t-1}, w_t) = w_t$$

and since ε_{t+1} is independent of $w_1, w_2, \ldots, w_{t-1}, w_t$, we obtain

$$E(\varepsilon_{t+1}|w_1, w_2, \ldots, w_{t-1}, w_t) = E(\varepsilon_{t+1}) = 0\,.$$

Thus, (2.36) can be written as

$$\hat{w}_t(1) = \mu + \Phi(w_t - \mu)\,. \tag{2.37}$$

In other words, a proportion Φ of the current deviation from the process mean is added to the process mean to forecast the next process value. This simple forecasting illustration can be expanded for a general lead time l. Replacing t with $t+l$ in (2.34) and repeating the above steps, we have

$$\hat{w}_t(l) = \mu + \Phi^l(w_t - \mu)\,. \tag{2.38}$$

Now consider the forecasting error for AR(1) for one unit into the future,

$$\begin{aligned} e_t(1) &= w_{t+1} - \hat{w}_t(1) && (2.39)\\ &= [\mu + \Phi(w_t - \mu) + \varepsilon_{t+1}] - [\mu + \Phi(w_t - \mu)] \\ &= \varepsilon_{t+1} && (2.40) \end{aligned}$$

where the variance of (2.40) at time t is

$$\begin{aligned} var\,[e_t(1)] &= var\,(\varepsilon_{t+1}) \\ &= \sigma^2_{\varepsilon_t}\,. \end{aligned}$$

The forecast error can also be generalised for a general lead time of l, by replacing the one in (2.39) with l, we then have

$$e_t(l) = w_{t+1} - \hat{w}_t(l)$$

where $l \geq 1, 2, \ldots$. For stationary models, if $l \to \infty$, then $\hat{w}_t(l) \to \mu$. The forecasts converge to the process mean [3]. However for non-stationary models if $\alpha \neq 0$ in (2.16), then $\hat{w}_t(l)$ will not converge to the process mean, for large l.

3 Application

In this section we apply the proposed methods of time series. The coding for this section can be found in the appendix.

3.1 Real GDP

The first step in time series modelling is to plot the data against time and analyse the graph. The second step is to determine if the data exhibits a trend, seasonal or remainder component. These steps are applied in this section.

3.1.1 Time plot of Real GDP

Since South Africa only became a democracy in 1994, only data from 1995 will be analysed to avoid any misreading. As explained in section 2.2.3 the natural logarithm of Real GDP is taken, which gives the following time plot of Real GDP, GDP_t.

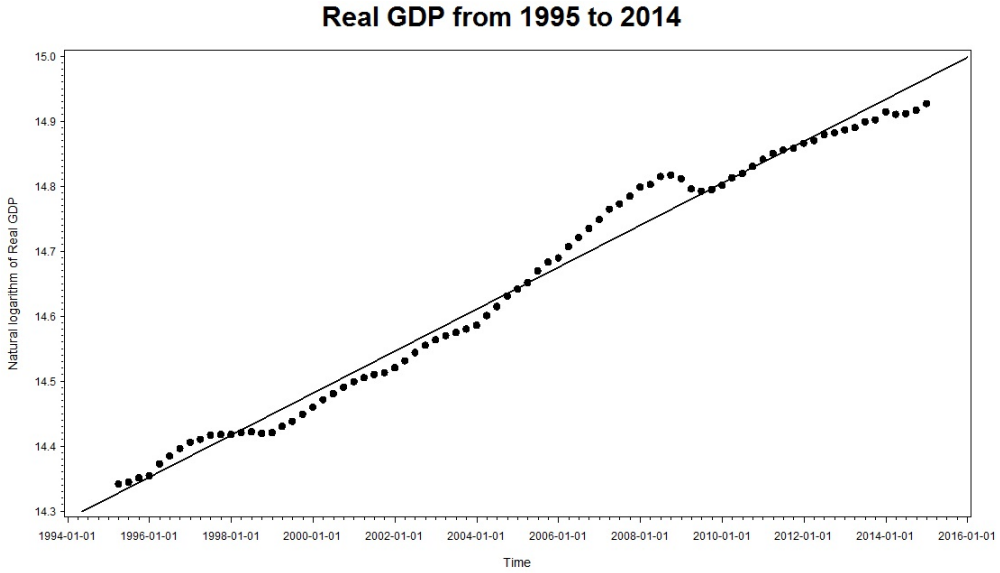


Figure 1: Quarterly South Africa Real GDP from 1995 to 2014

From figure [1], it is clear that Real GDP is upward trending, as indicated with the upward plotted regression line. There are two points on the figure that show a decrease, the first one is in 1999 and this is due to numerous reasons, one being the Asian financial crisis. The second decline in Real GDP is just before the 2010 marker, this was the result of the global financial crisis that occurred in 2008. The Financial Crisis Inquiry Commission concluded that the crisis was caused by failure in financial regulation and supervision, and excessive unregulated borrowing.

3.1.2 Seasonal-trend Decomposition

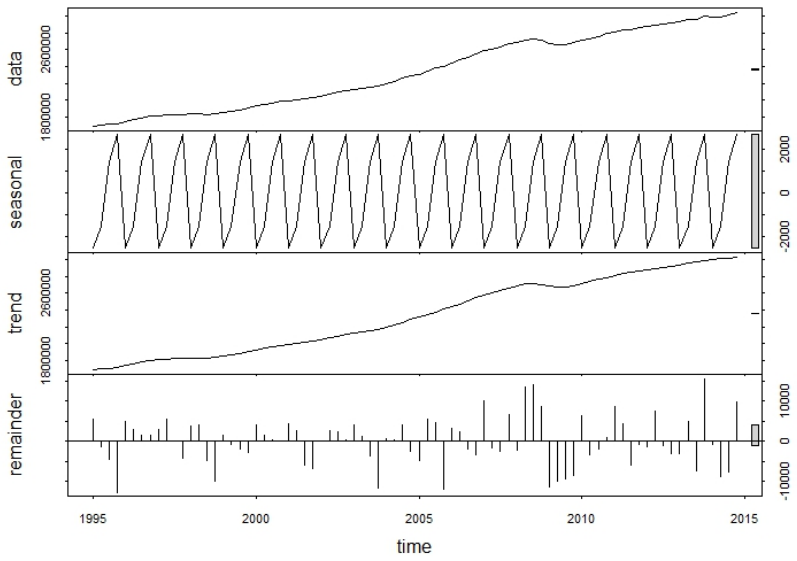


Figure 2: Seasonal-trend decomposition by loess smoothing

Figure [2] was obtained by using the seasonal-trend decomposition approach in *R* [19], which uses loess (local polynomial regression fitting - STL) smoothing, is an algorithm that divides the data set into three components namely: seasonality, trend and remainder. Neither the remainder nor the seasonal components show any significant signs that the model has remainder or seasonal characteristics. Figure [2] clearly indicates that the data has trend characteristics.

3.2 Testing for stationarity

There are numerous tests that will be performed in this section as mentioned in section 2. It has to be noted that a 5% significance level will be used throughout this section. Before it can be determined whether the process possesses stationarity, the tentative order selection tests are used. Three tests of the tentative order selection tests are used, namely:

1. Smallest canonical correlation method (SCAN)
2. Extended sample autocorrelation function method (ESACF)
3. Minimum information criterion method (MINIC)

Only the first two tests can be used for either stationary or non-stationary data. The Minimum information criterion method can only be used for stationary data and since it has not been determined whether the model is stationary, only the first two methods are used to determine the parameters of the ARMA model. The results are achieved using the *Arima* procedure method in *SAS*©.

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	<.0001	<.0001	<.0001	0.0003	0.0017
AR 1	<.0001	<.0001	<.0001	<.0001	0.0001
AR 2	0.0145	0.3277	0.3537	0.9978	0.2970
AR 3	0.7501	0.4751	0.5266	0.6786	0.5194
AR 4	0.4676	0.8268	0.9799	0.4794	0.5355

Table 2: SCAN method

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	<.0001	<.0001	0.0002	0.0019	0.0074
AR 1	<.0001	0.0051	0.3065	0.6403	0.7953
AR 2	0.0042	0.3933	0.3126	0.9308	0.3610
AR 3	0.4334	0.1124	0.3204	0.7207	0.2187
AR 4	0.0065	0.1076	0.6331	0.5369	0.1704

Table 3: ESACF method

From the results displayed in tables 2 and 3, both the SCAN and ESACF methods suggest an ARMA (2,1) or an ARMA (3,0). For effectiveness the first model will be used until tested otherwise.

3.2.1 ADF test

From the results obtained using the *ARIMA* procedure in *SAS* are represented in table 4. The autocorrelations lie outside the bounds and this is the first indication that the process is non-stationary. The ADF test before differencing yields the following:

Lags	τ	*p-value* of τ	Φ	*p-value* of Φ	F	$Pr>F$
0	-0.51	0.9814	-1.0865	0.9860	0.48	0.9900
1	-1.68	0.7497	-7.1729	0.6332	1.59	0.8590
2	-2.10	0.5388	-12.0048	0.2853	2.32	0.7162

Table 4: Results of regression coefficient-based test and the studentised test

Both the regression coefficient-based test and the studentised tests are rejected at a 5% and 10% significance level at lag 0, thus the null hypothesis of the ADF cannot be rejected. Lag 0 is used since the autocorrelations and partial autocorrelations also clearly shows non-stationarity, indicating that the process has a unit root, $I(1)$.

The following step is to take the first difference of the log of the Real GDP once and then analyse the results. As mentioned the ADF test will be used first. An ARMA (2,1) is suggested before differencing the model so the results will be analysed at lag 1. Differencing the Real GDP once yields the following ADF test results in table 5.

Lags	τ	*p-value* of τ	Φ	*p-value* of Φ	F	$Pr>F$
0	-4.70	0.0003	-34.87	0.0008	11.04	0.0010
1	-3.53	0.0096	-25.84	0.0014	6.23	0.0110
2	-3.84	0.0039	-38.75	0.0007	7.37	0.0010

Table 5: Results of first difference of regression coefficient-based test and the studentised test

Analysing the results at lag 1, the null hypothesis is rejected at a 5% and 10% significance level signifying that there is no unit root present after differencing. According to the results, the best suited model to explain Real GDP is an IMA (1,1) model.

3.2.2 KPSS test

In this section we will test whether Real GDP exhibits trend or level stationarity. We first do a KPSS test on the original Real GDP to confirm that it does indeed have a unit root. The KPSS test is preformed in *SAS,* making use of the *SAS Autoreg procedure,* yields results showed in table 6. The output gives two possible trends, for the first test the Type=Trend is used, as the time series has an upward trend as shown in figure [1].

	Critical values		
Test statistic value (t^*)	10% significance level	5% significance level	1% significance level
0.2356	0.1190	0.1460	0.2160

Table 6: KPSS test - Real GDP

The null hypothesis is rejected at a 5% significance level since 0.2356 (t^*) > 0.1460 (*critical value*). Real GDP does in fact exhibit a unit root and is thus non-stationary. Real GDP is then differenced and again tested using the KPSS test, in this case the Type=Single Mean will be looked at, and yields the results in table 7.

	Critical values		
Test statistic value (t^*)	10% significance level	5% significance level	1% significance level
0.1332	0.3470	0.4630	0.7390

Table 7: KPSS test - $\triangle$Real GDP

The null hypothesis is not rejected at a 5% significance level since 0.1332 (t^*) < 0.4630 (*critical value*). Concluding that $\triangle$Real GDP is stationary around a linear trend.

3.2.3 Testing for Normality

The results on whether a data set is well-modelled by the normal distribution can be used for model selection in several ways depending on the interpretations of probability. Normality tests are used to determine whether a normal distribution accurately describes the data set and to determine how likely the random variable is normally distributed. This test is not necessary to compute in determining stationarity, but it is necessary when analysing any random variable in time series. Normality tests provide a better understanding of how the random variable Real GDP is distributed. Normality is first tested using the Kolmogorv-Smirnov normality test. The null hypothesis for normality is rejected at a 5% significance level against a p-value of <0.0100 in table 8 for Real GDP, whereas the null hypothesis for normality is not rejected at a 5% significance level in table8 for $\triangle$Real GDP.

Kolmogorov-Smirnov (D)	*Statistic*	*p-value*
Real GDP	0.142177	<0.0100
$\triangle$Real GDP	0.068315	>0.1500

Table 8: Normality test of Real GDP and $\triangle$Real GDP

There are three main graphical methods for normality testing namely: normal probability plot, quantile-quantile plot (QQ-plot) and fitting a normal curve to a histogram of the data. For simplicity only the QQ-plot will be used. Both the Real GDP and the first difference of Real GDP will be analysed to see if there are any significant changes to the distribution of the Real GDP. For any QQ-plot one would like to see that the data is well aligned on the black line, but as seen in figure [3], this does not occur. This concludes that the distribution of Real GDP is not normal.

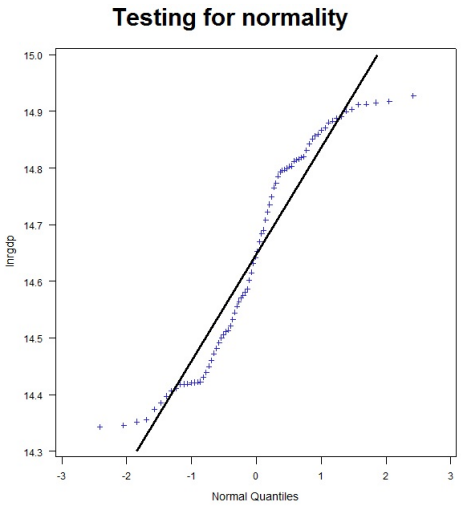


Figure 3: QQ-Plot of Real GDP

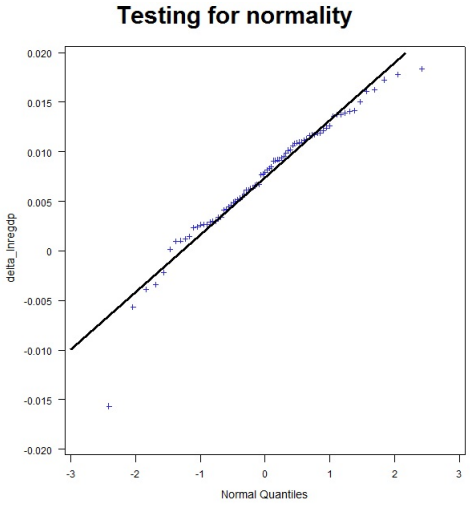


Figure 4: QQ-Plot of ΔReal GDP

When taking the first difference of Real GDP, the data is more aligned to the normality line as seen in figure [4]. However, ΔReal GDP does not perfectly fit on the normality line; this could be due to external factors influencing the Real GDP, for example the financial crisis in 2008.

3.2.4 Tentative Order Selection tests

Tentative order selection tests were used in a previous section; this was only done to roughly establish the parameters of the ARMA process. These tests will be preformed again, but with the MINIC method. It has been determined that Real GDP is in fact non-stationary and has been differenced. Using this newly founded

information and newly calculated variable, ΔReal GDP, the tentative order selection tests SCAN, ESACF and MINIC will be used in determining the ARMA parameters p and q.

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	<.0001	0.0037	0.2994	0.6675	0.7687
AR 1	0.2125	0.1093	0.8938	0.4060	0.7846
AR 2	0.1494	0.3896	0.5142	0.5383	0.6588
AR 3	0.7077	0.8484	0.6280	0.9150	0.5598
AR 4	0.5749	0.6020	0.7491	0.5885	0.5969

Table 9: SCAN method of ΔReal GDP-Model 1

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	<.0001	0.0037	0.2994	0.6675	0.7687
AR 1	0.2125	0.1093	0.8938	0.4060	0.7846
AR 2	0.1494	0.3896	0.5142	0.5383	0.6588
AR 3	0.7077	0.8484	0.6280	0.9150	0.5598
AR 4	0.5749	0.6020	0.7491	0.5885	0.5969

Table 10: SCAN method of ΔReal GDP-Model 2

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	<.0001	0.0047	0.3027	0.6755	0.7735
AR 1	0.0342	0.0270	0.6323	0.6411	0.3654
AR 2	<.0001	0.0294	0.8632	0.3038	0.3588
AR 3	0.0307	0.1496	0.1091	0.8444	0.3519
AR 4	0.0002	0.0376	0.2222	0.8505	0.5900

Table 11: ESACF method - ΔReal GDP

	MA 0	MA 1	MA 2	MA 3	MA 4
AR 0	-10.375	-10.5299	-10.6601	-10.661	-10.6287
AR 1	-10.7467	-10.6995	-10.6601	-10.613	-10.5746
AR 2	-10.7066	-10.6543	-10.6125	-10.5583	-10.5196
AR 3	-10.6722	-10.6199	-10.5677	-10.5145	-10.4781
AR 4	-10.628	-10.5823	-10.5292	-10.4766	-10.4302

Table 12: MINIC method-ΔReal GDP

The SCAN method suggests two models, ARMA(1,0) and an ARMA(0,2) in tables 9 and 10. The MINIC method in table 12 suggests an ARMA(1,0) which coincides with one of the SCAN suggested models. There is however now two equally weighted suggested models, ARMA(1,0) and ARMA(0,2). The SCAN method suggests both an ARMA(1,0) model in table 9, and an ARMA(0,2) model in table 10. There is no I(1), as the unit root has been taken out with differencing. Since it has now occurred that the three models suggest the ARMA(1,0) and ARMA(0,2) model. It should be known that the MINIC method carries more weight when analysing the parameters of the ARMA model. We will then use the ARMA(1,0) model to further analyse the data and plot autocorrelation functions in the next section to confirm the suggested parameters.

3.2.5 ACF and PACF

As mentioned in section 2.2.3, autocorrelation and partial autocorrelation functions can be used to determine the parameters of the ARMA model. The *ARIMA* SAS function was used with a lag of 12 to graphically represent the autocorrelations. Looking at figure [5], the autocorrelation plots suggests an ARMA(1,2) model as the null hypothesis, $H_0 : \Phi_{22} = 0$ and $H_0 : \theta_3 = 0$ is not rejected. Coincidently, this is exactly the opposite model proposed in the introduction of section 3.2, the Real GDP before differencing. Since the MINIC method suggests an ARMA(1,0) model and the partial autocorrelations align with this suggested parameters, it is

concluded that the Real GDP is an AR(1) model, after taking the first difference. We do not use inverse correlations for analysing this data set.

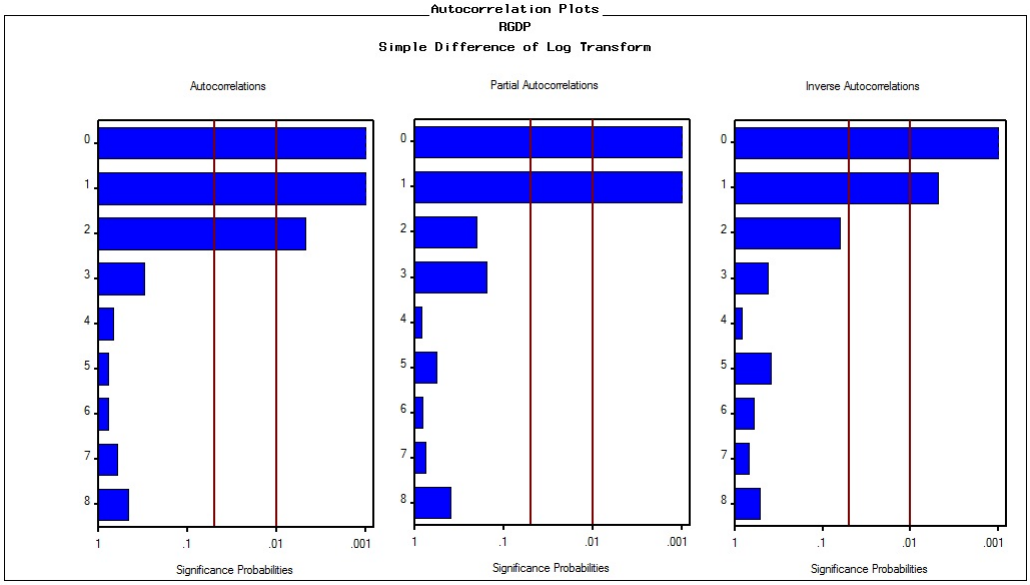


Figure 5: Correlation plots of ΔReal GDP

3.2.6 The model and Residual analysis

As explained in section 2.2.3 we take the log of the GDP to have a linear trend and then have a time plot shown in figure [1]. However, the Real GDP is non-stationary. As mentioned above, we took the first difference of Real GDP and then applied the model in *Proc Autoreg*. The model is then transformed and μ is replaced by a constant c in (2.22):

$$\begin{aligned} GDP_t &= c + \Phi GDP_{t-1} + \varepsilon_t \\ \triangle GDP_t &= c + \Phi GDP_{t-1} + \varepsilon_t - GDP_{t-1} \\ &= c + (\Phi - 1) GDP_{t-1} + \varepsilon_t. \end{aligned} \tag{3.1}$$

Then add the estimated parameters to (3.1),

$$\begin{aligned} \triangle GDP_t &= 0.00738 + (0.55014 - 1) GDP_{t-1} + \varepsilon_t \\ &= 0.00738 - 0.44986 GDP_{t-1} + \varepsilon_t \end{aligned} \tag{3.2}$$

where estimated $var(\varepsilon_t) = 0.0000236$. The residuals of any model should be analysed to determine if volatility is present.

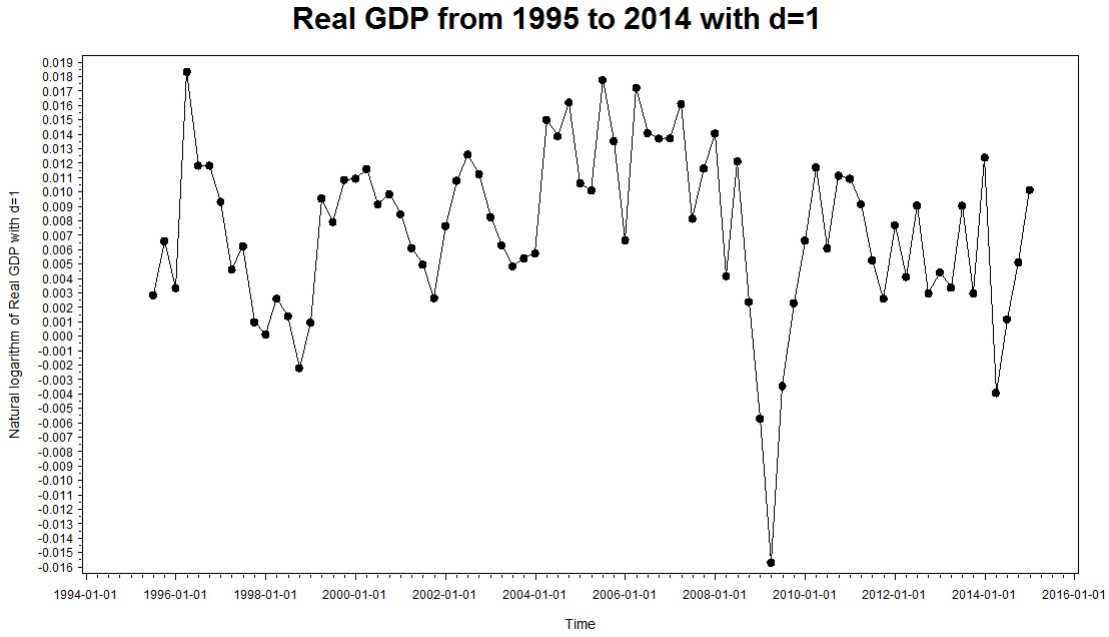


Figure 6: South Africa Quarterly △Real GDP

An AR(1) model is fitted to △Real GDP and we conclude that the residuals resemble a white noise. It has been suggested that some financial time series have ARCH errors, so we will investigate this possibility by analysing the residuals. Note that the residuals of △Real GDP do have a normal distribution in figure [7]. Ideally we would like to see residuals scattered rectangularly around a zero horizontal level with no trends in the first graph in figure [7], this is however not the case. The graph does not support the AR(1) model as there is reduced variation from observation 10 to 40 and then increased variation from observation 40 to 60.

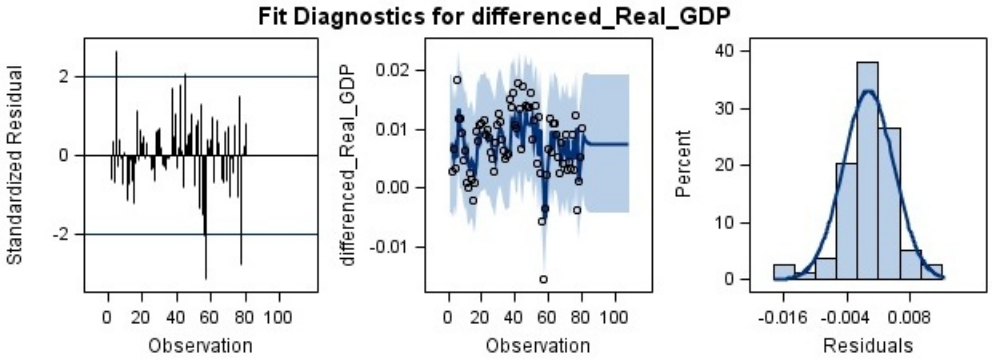


Figure 7: Diagnostics of Residuals for △Real GDP

Our model is thus not accurate enough to forecast yet. Since there seems to be a pattern still present in the data, we will need to incorporate this into the model by making use of ARCH(1). The model is then given by adjusting the parameters of (3.1),

$$\begin{aligned} \triangle GDP_t &= 0.00829 - 0.5185GDP_{t-1} - GDP_{t-1} + \varepsilon_t \\ &= 0.00829 - 1.5185GDP_{t-1} + \varepsilon_t \end{aligned} \tag{3.3}$$

where $\varepsilon_t \sim N(0, h_t)$, which is proven to be normally distributed by the output where the null hypothesis

of normality is not rejected and

$$h_t = 0.0000121 + 0.5930 y_{t-1}^2 \, .$$

The model in (3.3) is a better quality model than the model in (3.2) according to the Akaike information criterion (AIC). The AIC for (3.2) is -588.53 and -615.15 for (3.3), thus (3.3) is the preferred model and will be used to forecast the Real GDP.

3.2.7 Forecasting

Before forecasting the Real GDP model with ARCH(1) (3.3), we first forecast the Real GDP model with AR(1) (3.2), to allow us to compare the accuracy of the models. Consider the Real GDP model (3.2 with the first difference with the simulated data using MLE (maximum likelihood estimation) and with a 95% interval on the prediction, illustrated in figure [8].

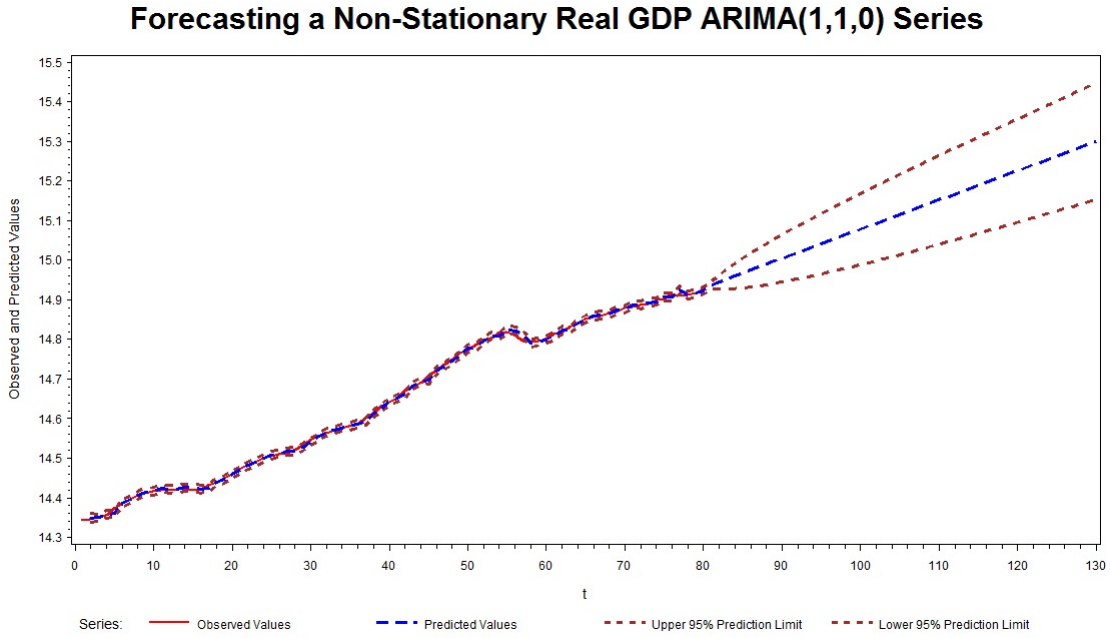


Figure 8: Forecasting ΔReal GDP with AR(1)

Due to the non-stationarity of Real GDP, the forecasts and the prediction intervals do not converge and with an AR(1) forecasting it is a predicted upward trending Real GDP in figure [8]. The forecasting of Real GDP using ARCH(1) in figure [9] does not resemble a perfect upward trending prediction.

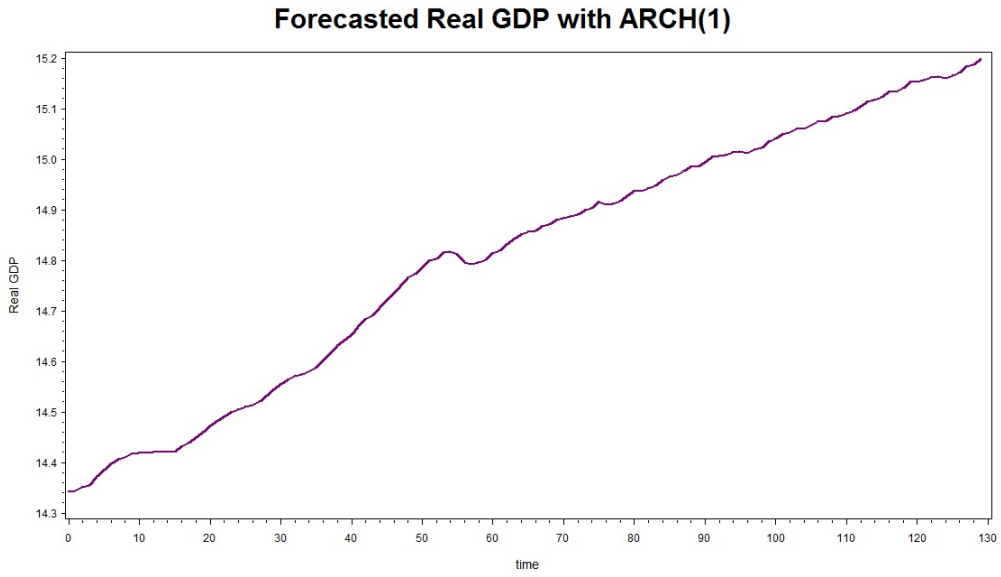


Figure 9: Forecasting ΔReal GDP with ARCH(1)

In figure [10] a clear comparison can be drawn from the AR(1) forecast and the ARCH(1) forecast. Both the AR(1) and ARCH(1) forecasts lie within the 95% prediction limit.

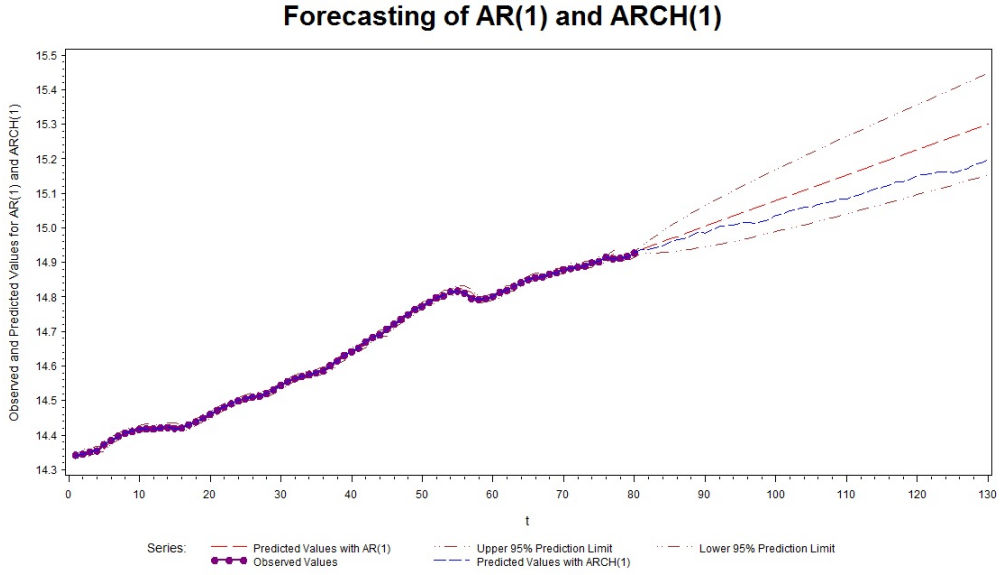


Figure 10: Comparison of AR(1) and ARCH(1) forecasting of Real GDP

Looking at a small window, drawn from figure [10] without the prediction limits, the predicted ARCH(1) model is clearly below the predicted AR(1) model predictions. The ARCH(1) model predictions also resemble a more natural time series trend.

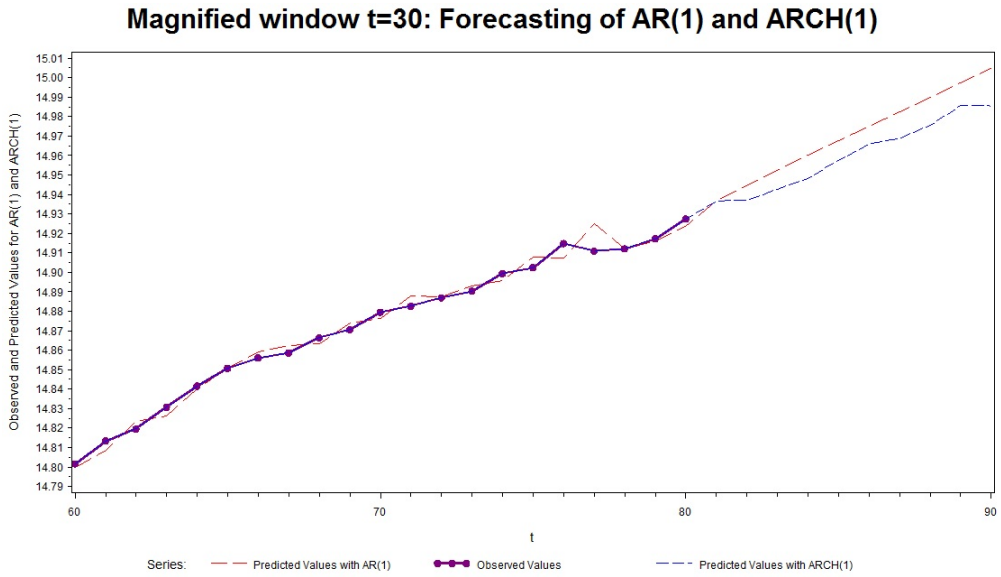


Figure 11: Magnified Window of Forecasted AR(1) and ARCH(1) for t=30

In figure [11] the AR(1) predicted values vary a little from the observed values, whereas the ARCH(1) model better fits the observed model.

4 Conclusion

The central idea of this paper was to determine whether South African Real GDP exhibits stationarity. Although there are many external factors that can influence the Real GDP, it is still a good measure of an economy's performance. The Real GDP is upward trending which is the first indication that Real GDP is not stationary.

The null hypothesis of the ADF test for Real GDP was rejected at a 5% confidence level indicating non-stationarity. However, after taking the first difference of Real GDP the null hypothesis could not be rejected, indicating that by differencing the model, Real GDP became stationary. This assumption was supported by the non-rejection of the null hypothesis of the KPSS test of $\triangle$Real GDP. Concluding that $\triangle$Real GDP is stationary around a linear trend. The model identification of Real GDP had many suggestions, however, through careful analysis it can be concluded that Real GDP can be represented as an ARIMA(1,1,0) model, or better known as an ARI(1,1) model. Furthermore, Real GDP residuals do not have constant variance; hence the ARCH model is applied to compensate for this variation. Through regression of $\triangle$Real GDP, it is found that an ARCH(1) model is the best suited model according to AIC. With forecasting it is discovered that the ARCH model also forecasts better as it acknowledges the variation in the residuals of Real GDP.

To conclude South Africa Real GDP is not stationary. Real GDP can either be modelled by AR(1) or ARCH(1) models, but in this paper it is tested and found that the ARCH(1) model yields more accurate results. South Africa is an emerging economy which implies a higher Real GDP growth rate. However, the forecasting of Real GDP shows us that Real GDP will increase, but not as rapid as expected from an emerging market.

References

[1] A. Bhargava. On the theory of testing for unit roots in observed time series. *The Review of Economic Studies*, 53(3):369–384, 1986.

[2] T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

[3] G.E.P. Box, G.M. Jenkins, and G.C. Reinsel. *Time Series Analysis: Forecasting and Control*, volume 734. John Wiley & Sons, 2011.

[4] J.D. Cryer and K. Chan. *Time Series Analysis: With applications in R*. Springer Science & Business Media, 2008.

[5] D.A. Dickey and W.A. Fuller. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366a):427–431, 1979.

[6] R.F. Engle. Autoregressive Conditional Heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4):987–1006, 1982.

[7] A.R. Fleissig and J. Strauss. Is OECD real per capita GDP trend or difference stationary? evidence from panel unit root tests. *Journal of Macroeconomics*, 21(4):673–690, 1999.

[8] J.D. Hamilton. *Time Series Analysis*, volume 2. Princeton University Press, 1994.

[9] C. Kleiber and A. Zeileis. *Applied Econometrics with R*. Springer Science & Business Media, 2008.

[10] D. Kwiatkowski, P.C.B. Phillips, P. Schmidt, and Y. Shin. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, 54(1):159–178, 1992.

[11] B. Lucke. Is Germany's GDP Trend-Stationary? A Measurement With Theory Approach. *Jahrbücher für Nationalökonomie und Statistik*, pages 60–76, 2005.

[12] N.G. Mankiw. *Macroeconomics*. Charles Linsmeier, 8th edition edition, 2003.

[13] B.P.M. McCabe and A.R. Tremayne. Testing a time series for Difference Stationarity. *The Annals of Statistics*, 23(3):1015–1028, 1995.

[14] P. Mohr. *Economic Indicators*. Unisa Press, 2005.

[15] C.R. Nelson and C.I. Plosser. Trends and random walks in macroeconmic time series: some evidence and implications. *Journal of Monetary Economics*, 10(2):139–162, 1982.

[16] K. Patternson. *Unit Root Tests in Time Series*, volume 1. Palgrave Macmillan, 2011.

[17] B. Pfaff. *Analysis of Integrated and Cointegrated Time Series with R*. Springer Science & Business Media, 2008.

[18] M.B. Priestly. *Non-Linear and Non-stationary Time Series Analysis*. Academic Press, 1988.

[19] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015.

[20] S.E. Said and D.A. Dickey. Testing for unit roots in autoregressive-moving average models of unknown order. *Biometrika*, 71(3):599–607, 1984.

[21] R.H. Shumway and D.S. Stoffer. *Time Series Analysis and its applications*. Springer Science & Business Media, 2013.

[22] V. Van Rensburg, C.R. McConnell, and S. Brue. *Economics for Southern African*. Economics. McGraw-Hill Education, 2011.

[23] M. Verbeek. *A Guide to Modern Econometrics*. John Wiley & Sons, 2008.

5 Appendix

Code

All coding was done using *SAS© Procedures,* except the seasonal trend decomposition was done in *R [19].*

Timeplot

```
data a;
set sasuser.gdp_test;
d1=dif(log(rgdp));
run;
goptions reset=all;
title1 'Real GDP from 1995 to 2014 with d=1';
axis1 label=(angle=90 'Natural logarithm of Real GDP with d=1');
axis2 label=('Time') minor=(number=7) order=('March95'd to 'December14'd by qtr);
symbol1 color=black i=join value=dot width=1;
proc gplot data=a;
plot d1*date / vaxis=axis1 haxis=axis2;
run;
```

Season Trend Decomposition

```
RGDP<-ts(gdp,start=1993,freq=4)
plot(stl(RGDP,s.window="periodic"))
plot(stl(log(RGDP),s.window="periodic"))
```

ADF test

```
proc arima data=use;
identify var=lnrgdp nlag=6;
run;
proc arima data=new;
identify var=delta_lnregdp stationarity=(adf=(0,1,2,3,4));
run;
```

KPSS test

```
data hh;
set sasuser.Gdp_test;
lnrgdp = log(RGDP);
diff_lnrdgp = dif(lnrgdp);
run;
proc autoreg data=hh;
model lnrgdp=date / stationarity = (KPSS);
run;
proc autoreg data=hh;
model diff_lnrdgp=date / stationarity = (KPSS);
run;
```

Testing for normality

```
goptions reset=all;
title1 'Testing for normality';
axis1 label=(angle=90 'ln( Real GDP)') order=14 to 15 by 0.1;
proc univariate data=use normal;
var lnrgdp;
histogram lnrgdp / normal (mu=est sigma=est color=black w=2);
qqplot lnrgdp /normal (mu=est sigma=est color=black w=2) square;
probplot lnrgdp /normal (mu=est sigma=est color=black w=2) square;
run;
proc univariate data=new normal;
var delta_lnregdp;
histogram delta_lnregdp / normal (mu=est sigma=est color=black w=2);
qqplot delta_lnregdp /normal (mu=est sigma=est color=black w=2) square;
probplot delta_lnregdp /normal (mu=est sigma=est color=black w=2) square;
run;
```

Tentative Order Selection tests

```
proc arima data=new;
identify var=delta_lnregdp scan esacf minic p=(0:4) q=(0:4);
run;
```

The model and residual analysis

```
data a;
set sasuser.gdp_test;
d1=log(rgdp);
run;
proc arima data=A out=ima_out;
identify var=d1(1) noprint;
estimate p=1 method=ml;
forecast lead=50;
run;
data graph;
set ima_out;
t=_n_;
run;
proc autoreg data=gdp1;
model differenced_Real_GDP = / nlag=1 method=ml garch=(p=1);
output out=a cev=v r=resid;
run;
```

Forecasting

```
proc autoreg data=gdp1;
model differenced_Real_GDP = / nlag=1 method=ml garch=(q=1) ;
output out=ab cev=v r=resid;
run;
Proc iml;
use ab; read all var{v RGDP time differenced_real_gdp} into matrix;
use SASUSER.forecast3; read all var{Forecast} into forecastmatrix;
n=nrow(matrix);
error = matrix[,1];
realGDP = log(matrix[,2]);
difflog_rgdp=matrix[,4];
time = matrix[,3];
sigma=0.5930;
ht = J(n,1,.);
yt = J(n,1,.);
x=J(n,1,.);
yt[1,]=error[1,]*sqrt(ht[1,]) ;
do i=2 to n;
ht[i,]=0.0000121+0.5930*(yt[i-1,]##2);
yt[i,]=error[i,]*sqrt(ht[i,]) ;
x[i,]=0+ht[i,]*rannor(1);
end;
test = difflog_rgdp;
mu = J(n,1,0.00829);
phi_min_one=J(n,1,-1.5185);
gdp_forgraph = J(n,1,.);
gdp_forgraph = RealGDP[1:80,];
do o=81 to n;
test[o,] = mu[o,] + phi_min_one[o,]*test[o-1,]+x[o,];
test = test//test[o,];
end;
print test;
GDP = RealGDP[1:80,]//forecastmatrix[1:50,];
newmatrix = time||GDP;
varlist = {'time' 'GDP'};
create newset from newmatrix[colname=varlist];
append from newmatrix;
quit;
goptions reset=all i=join;
axis1 label=(angle=90 'Real GDP');
symbol1 color=purple line=1 width=2;
title1 'Forecasted Real GDP with ARCH(1)';
proc gplot data=newset;
plot GDP*time / vaxis=axis1;
run;
proc iml;
use ima_out; read all var{forecast u95 l95 d1} into jj;
use newset; read all var{GDP} into kk;
n=nrow(jj);
nn=nrow(kk);
wholedata = jj[1:130,]||kk[1:130,];
varlist = {'Forecast with AR(1)' 'U95' 'L95' 'Observed data' 'Forecast with GARCH(1)'};
create cmpare from wholedata[colname=varlist];
append from wholedata;
quit;
goptions reset=all i=join;
axis1 label=(angle=90 'Observed and Predicted Values for AR(1) and ARCH(1)');
```

Improving the enrolment strategy in the Faculty of Economic and Management Sciences through an inquiry into the throughput rates of diverse enrolment and transfer streams - 2015 Study

Claudia Di Santolo 12019462

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor: Mr A Swanepoel

Department of Statistics, University of Pretoria

2 November 2015 (final)

Abstract

The objective of this research report is to predict using among others regression analysis, if any of the enrolment streams as experienced by the Economic and Management Sciences (EMS) Faculty at the University of Pretoria produce students that are more successful than others. The main variable of measure will be that of Grade Point Average (GPA), which is computed using a credits system. The use of basic descriptive statistics such as averages will give a broad overview on the trends experienced by the enrolment streams. Due to averages being analysed with caution further investigation will be evaluated to see if there are definite differences in the performance of students at the First-year level in the different streams.

Furthermore, emphasis will be placed on measuring numerous variables to determine performance of students within the Faculty of Economic and Management Sciences. Such variables include examining the performance of students according to their gender irrespective of what stream they fall under. In addition, the Kruskal-Wallis test will be applied to matric authority description to analyse whether students who attended different schooling types lead to a more accurate reflection of First-year GPA scores and level of performance. The population distribution functions of the four leveled independent matric authority description will also be estimated to try and reduce the assumptions that are made.

Classical linear regression analysis will be applied to possibly find which predictor, if any, using the three NBT categories, the Grade 11 and 12 APS scores, nationality and gender to determine which is the most effective to base acceptance on. This will be useful to enlighten the Faculty of EMS as to which predictor to base their decision on in accepting potential students in order to improve the throughput rate. Along with this, the different streams were compared in a between group manner to evaluate if any stream does indeed have higher First-year GPA scores. This will once again be of assistance to the Faculty in the number of places that are reserved for the different streams.

Declaration

I, *Claudia Di Santolo,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Claudia Di Santolo

_____
Mr A Swanepoel

_____
Date

Acknowledgements

The author would like to thank Mr Carel Venter from BIRAP for providing the data that is used in this research report. In addition, many thanks to SoTL for funding this research project as without that this project would not be possible.

Contents

List of Figures

List of Tables

1 Introduction

Faculties at tertiary institutions strive for the best possible level of success, none more so than the Faculty of Economic and Management Sciences (EMS) at the University of Pretoria (UP). Furthermore, in this context the main aim is to analyse which student enrolment group (simply referred to as streams) at the tertiary level performs the best in order to enrol the best students. The results of the analysis can be used to develop an optimal enrolment strategy which can be used to forecast student success in order to improve the throughput rate. Several student streams are employed by the Faculty of EMS including: New students (students enrolled for the first time at UP), Readmitted students, Transferring students and students enrolled for the Extended Programme. Thorough analysis will be conducted amongst these streams to conclude whether a certain stream is indicative of students from that stream being more successful.

One of the main questions asked is: how can one measure success? Measuring success is not as straightforward as computing a calculation as there are other external factors that contribute to a student's level of success.

Firstly, what is considered to be a measure of success? Camara et al. in [4] suggests that by calculating a correlation coefficient between different variables will result in success. These variables may include among others secondary school averages, Grade Point Averages (GPA) computed after each round of examination marks in the case of this study. Hence, a higher level of correlation (where the coefficient value is close to one) will reflect an increased or higher level of achievement i.e. success [4]. Furthermore, Camara et al. in [4] went on to discuss the different methods that can be used to determine student success; these include SAT scores, this correlates to the National Benchmark Test (NBT) scores in a South African context; graduation rates; cumulative weighted average; secondary school grades and finally whether a student receives a scholarship based on athletic or artistic abilities.

A good measure and one that will be a constant focus point throughout this research report is GPA. A detailed description regarding GPA will be discussed in the next section.

A study done by le Roux et al. in [8] concluded that an access test (with respect to this research report access tests would equate to the NBT) provides a more accurate indication of success rates amongst students in comparison to the averages that students achieve in Grade 12. For instance when the 2014 study of the same title was conducted by Pretorius [13], the results revealed that for First-year students the weighted average was 60 percent (only using June data as that was the available data at the time), the NBT average equaled 55 percent and Grade 12 averages were equal to 71 percent, hence, NBT scores are a little bit conservative (under estimating), but closer, in predicting student performance at tertiary level as opposed to Grade 12 marks which is over estimating the tertiary performance.

Success is however, not solely based on marks. A student's interaction with the tertiary institution [15], commitment and persistence to completing the chosen degree [16], student retention [14], degree choices and a student's ability to work with other individuals [16] are a few factors that also contribute to student success.

The approach of student success is not the only variable of significance as throughput rates should also be investigated. Throughput rates are per definition, whether a student obtains and thus, graduates from UP (or any other tertiary institution) having completed their degree in the quantified time period as stipulated by UP (or any other tertiary institution). Furthermore, if the findings (upon completion of evaluating the data) disclose that a certain stream of students are indeed considered to be more successful, consideration must be given to the number of students allowed into the Faculty via the different streams in order to maximise throughput.

As aforementioned, throughput rates refers to the time that it takes a student to graduate from their respective tertiary institutions. Students enrolled at South African tertiary institutions, that graduate in the prescribed time-period is approximately 15 percent [9], which is regarded as being one of the lowest on an international scale.

In addition, the National Planning Committee (NPC) have mentioned that even though there has been substantial improvements in the number of students that are enrolled in tertiary institutions [3], there are still numerous challenges. One of these challenges is that of resources [9]. Even with various shortcomings (for example financial) that are present in tertiary institutions at a national level, there is a need for improving opportunities for students as well as the success of students [2].

Diverse enrolment in tertiary institutions should also be examined. According to Akojee and Nkomo [1],

approximately 53 percent of students enrolled at tertiary institutions across South Africa in the year 2002 were female. In the findings of the research report done last year (2014), 55 percent of First-year students enrolled at UP are female [13].

The current enrolment strategies (streams) that are used by the Faculty of EMS at UP can also be compared to those used at other national as well as international tertiary institutions.

This research report will try and emulate these approaches in order to predict student success at UP. Furthermore, section 2 (background theory) will explain GPA in detail and the techniques that will be applied. Application and models used to try and predict student success will be explained in section 3. A short summary of the results will form section 4. Finally, a conclusion (section 5) will be given that will summarise and highlight the main findings and conclusions in the research report. Moreover, further research and other applications that can be pursued to achieve maximum student success for the different streams, the findings and deduces which streams lead to student success (if any).

2 Brief summary of the Background Theory

A good measure and variable that will be focused on in this research report is that of GPA. This is due to the fact that some other measures do not take the difficulty of modules into consideration. GPA makes provision for difficulty by using a credits system. The credits system is based on the idea of notional hours. Credits per definition is a way of computing several attainable learning outcomes, at any one of ten levels as specified by the National Qualifications Framework (NQF), in terms of notional hours [5]. Notional hours can be defined as the estimated learning time that it takes an average student to achieve 50 percent for a module [5]. For example EKN 110 (course code for economics at the 100 level at UP) comprises of ten credits which means that an average student needs to spend 100 (ten credits multiplied by notional hours, equivalent to ten hours, i.e. 10 x 10 = 100) hours in order to pass the module with a mark of 50 percent. This includes contact lectures, practicals, tutorials and all assessments as well as the time the student spent on preparation for all activities within the module. One credit equates to ten notional hours. Furthermore, different year levels have different credit values, thus, as mentioned previously EKN 110 carries ten credits whereas EKN 214 (Economics presented at the 200 level) comprises of 16 credits and lastly, EKN 310 (300 level economics) has a weight of 20 credits. All these values are based on the credit system as implemented by UP. Therefore, it is deduced that the credits for a particular module field increases as the year level increases, thus, it compensates for the increase in difficulty as well as volume within modules.

UP takes the aforementioned method into account, for all modules when computing a student's GPA. Along with this, when a student is awarded a supplementary exam, provision is made when a student's GPA is calculated.

GPA will be compared to various other variables such as that of the NBT, as to where First-year marks are over-or under-estimated. Scholars within the South African schooling system are required to write these NBT's which comprise of a mathematical, academic literacy and finally, a quantitative literacy component. Even though the NBT's were a directive of the Higher Education of South Africa, the NBT's are conducted by an independent advisory panel [10]. The advisory panel ensures that the quality of the tests that are written by scholars are in line with an international level, more specifically that of the United States of America, as the tests are revised with a quality guarantee from the Assessment Systems Corporation in Michigan as well as the Educational Testing Services at Princeton [10]. As opposed to just using a student's school average, the NBT's could potentially provide a more accurate reflection of a student's academic abilities for tertiary institutions. Furthermore, according to MacGregor [10], the purpose of these benchmark tests is to evaluate the proficiency of students at the First-year level (leaving school and entering the university environment) in academic aspects such as mathematics and literacy, in order to compare the standards between final year secondary schooling outcomes (final Grade 12 marks) and the minimum admission requirements of a university degree. Hence, these tests are supposed to provide tertiary institutions with the necessary information to be of assistance in the improvement of curriculum's [10] as well as encouraging faculties to accommodate the educational requirements of students. In section 3 of this research project, the NBT averages will be computed and examined to measure student success using the data of UP students. On a larger scale, throughout South African institutions, the results of the NBT's have been concerning, extremely low mathematics abilities (only

seven percent of scholars have the necessary skills to be able to register for a mathematics degree), with the other two components of the tests fairing slightly better [10], raises the question of the schooling system in South Africa. Stellenbosch University has its own unique test, known as the Access Test (AT) and have required all potential students to write the respective tests depending on module selections, to determine the readiness of a student to study at the institution, based on numeracy, language and thinking skills as well as subject specific areas including mathematics and physical science [12]. In terms of a prediction model used at Stellenbosch University constitutes the performance of both quantitative and qualitative factors in determining whether a student would pass or fail [12]. Thus, perhaps it might be more reflective if tertiary institutions imposed their own entry level tests.

Various approaches will be used to determine the success of students, such as regression analysis and paired t-tests. Regression analysis, in general, can be defined as the dependence of a single variable (also known as the dependent variable) on other variables (which could equate to one or more variables, these are also known as the independent or explanatory variables) in order to attempt making a prediction of the mean value of the dependent variable based on the values of the known independent variables, usually with regards to repeated sampling [6]. Furthermore, Kutner et al. in [7], go on to discuss regression analysis in terms of a statistical relation, firstly, where there is an inclination of the dependent variable to contrast to that of the independent variable in a systematic way. Secondly, there is a scattering of various points around a curve that has a statistical relation [7]. An assumption made is that for every level of the independent variable it is possible to compute a corresponding probability distribution of the dependent variable [7]. Hence, in regression analysis the focus is on using statistical dependence amongst variables, simply a statistical relation, which is indicative of stochastic variables [6], meaning that there is not a perfect relation between these variables, thus, concluding that the observations of these variables would not lie directly on the regression curve or line (which ever turns out to be more appropriate) [7]. Therefore, regression analysis will give assistance to the main three purposes thereof, namely, control, description and prediction or estimation [6]. Regression parameters and the dependent variable will be determined in order to perform regression analysis using the data that has been provided.

Correlation analysis will also be examined which is broadly used as a measure of the linear relationship between any two variables, be it between the independent variable and the dependent variable or between any two dependent variables [6]. When using correlation analysis there is no discrepancy between the dependent and the independent variables, hence, they are symmetric and the assumption that is made is that they are both stochastic variables [6]. In addition, the correlation coefficient, represented by the symbol r, is as mentioned indicative of the strength of the linear relationship between variables. The coefficient can take on any value between -1 and 1, where for instance, -1 represents a perfect negative linear relationship, -0.9 a strong negative linear relationship, -0.1 a weak negative linear relationship, a r value equal to zero equates to no linear relationship between the variables. The opposite becomes true for the positive values. Hence, 0.1 is a weak positive linear relationship, 0.9 a strong positive linear relationship and finally a r value equal to 1 represents a perfect positive linear relationship.

Finally, paired t-tests will also be used in the analysis when doing hypothesis testing. It will be used to determine the difference between means and variances of the grouped data. An assumption made is that the differences are normally distributed [11]. Once the difference of the means have been computed, usually denoted by D_i, then a normal one sample t-test can be performed to test the various hypotheses [11].

The reader is referred to Pretorius [13], research entitled "Improving the enrolment strategy in the Faculty of Economic and Management Sciences through an inquiry into the throughput rates of diverse enrolment and transfer streams" for more information regarding the background theory, for instance the scale at which APS scores are calculated, subjects offered at schooling level etc.

3 Application

The application will be applied to First-year data of students that are enrolled at UP as second years in 2015 and is similar to the 2014 study when they were first years. The percentage of students in the Faculty of EMS enrolled across the four streams are given in Figure 1. The four admittance types are as follows: New (2014 was the first year of study at UP), Readmitted, Returning and finally Transferring students. From Figure

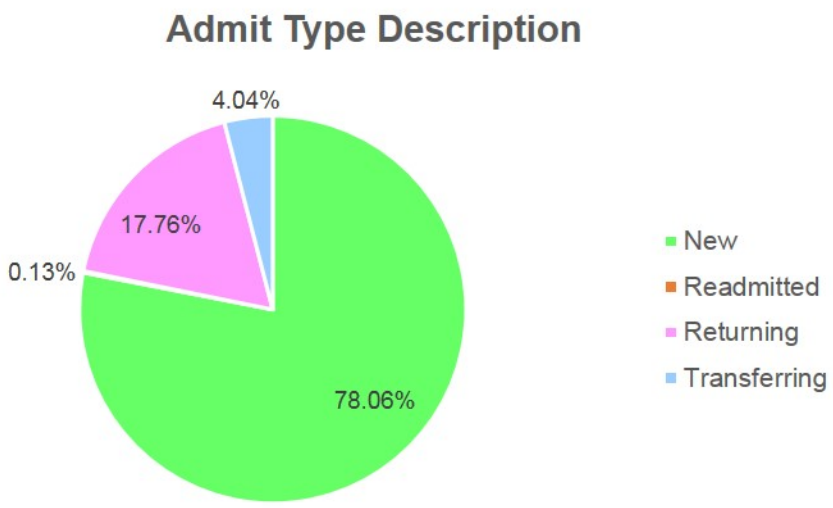


Figure 1: Admittance Type Description of Students enrolled at UP

1 it can be seen that the majority of enrolments are those of New students, followed by Returning students (a percentage of 17.76). Transferring students into the Faculty of EMS (having previously been enrolled in another Faculty within UP) are classified third in terms of number of students. Finally, the Readmitted students represents a mere 0.13 percent of all the students. Upon further investigation it was revealed that the students enrolled for the Extended Programme (where students complete their degree in a period of four years) were embedded under the main stream category headings of New, Readmitted, Returning and Transferring. Hence, the analysis shows that 104 of 1 812 of New students were actually enrolled for the Extended (Four-year) Programme. Furthermore, no students who were readmitted into the Faculty of EMS then re-enrolled for the Extended Programme, since any student is only allowed to enrol once for the Extended Programme. One Extended Programme student was a Returning stream student and finally, two students of 91 from the Transferring stream applied for the Extended Programme. It should be noted that, with reference to Figure 1 that from this point onwards for the purpose of statistical analysis provision will be made for the stream entitled Extended Programme, i.e. separate analysis will be performed for this respective stream. In addition, five streams (New, Readmitted, Returning, Transferring and Extended Programme) will now be used as opposed to the original four main streams (New, Readmitted, Returning and Transferring) in order to provide the most accurate results.

One of the techniques that will be used in the application is that of regression analysis along with performing a Kruskal-Wallis test. ANOVA (Analysis of Variance) tables will also be used with regard to comparing the various streams in an attempt to distinguish whether a certain stream, if any, does indeed lead to success. These approaches will be helpful in determining whether the respective hypotheses, specifically whether the null hypothesis (H_0) stating that the performance of students in the different streams are equal, can be rejected or not.

Note: due to missing data under certain headings (for instance, Grade 11 averages etc.) within the study, certain streams maybe excluded in a specific set of analysis as a result of this reason. On a larger scale it is difficult to account for all the data values as well as categorical data (such as degree choice) attributed to every single student. The reader will be informed when and if there is a possibility that a certain stream was excluded.

Additional note: Figures 3 through 19 were generated using SAS® software[1]. The corresponding SAS®

[1]The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA

software code can be found in the Appendix and the figures are of SAS® software output for the applicable data analysis.

3.1 Basic descriptive statistics

3.1.1 Enrolment streams

In applying basic descriptive statistics, namely averages, to First-year data, GPA, Grade 11 and Grade 12 together with the NBT averages, will be indicative as to the performance of the various streams. This however, will not be indicative as to whether or not a specific stream leads to more success amongst students. With regards to averages, see Table 1 below, the students in the category entitled New performed the best in Grade 11 as well as having the highest GPA average (58.2 which is slightly greater than the GPA average of 57.06 as achieved in the Readmitted category). The worst GPA performance was an average of 45.46 for the Extended Programme stream whilst the Returning and Transferring streams had a GPA average of 47.82 and 52.91, respectively. Furthermore, across the various streams the averages for Grade 11 were very similar to that of the mean of the respective streams' Grade 12 performance. For instance, when examining the Returning stream, the Grade 11 average was 66.93 whereas the average for the student's Grade 12 performance was 67.19. It can be seen that the trend is similar when comparing the Grade 11 and Grade 12 performance across the different streams, with the exception of the Readmitted students because several values for students are missing or not made available. Thus, it was not possible to derive an average for the Readmitted students for Grade 11 and NBT, hence, in the case of examining these averages this group of students were not part of the analysis. One important conclusion that can be drawn from Table 1 is that of GPA and NBT when comparing the averages. It can be established that the NBT average is more accurate (consistent) in determining how a student will perform at tertiary level, in particular at UP, than using either the averages of Grade 11 or Grade 12 since these averages could possibly over-estimate a student's performance at university. Consequently, the GPA measure maybe conservative and under-estimates a student's success level but is more accurate in predicting the First-year performance that students can expect. There will be exceptions and external factors cannot be ignored entirely. Averages should be treated with caution as it could potentially lead to inaccurate interpretation of the main objective of this study. A more in-depth analysis will take place in the proceeding sections.

	New	**Readmitted**	**Returning**	**Transferring**	**Extended Programme**
Grade 11	72.35	Not available	66.93	65.00	63.07
Grade 12	72.56	76.25	67.19	62.89	63.46
NBT	56.87	Not available	53.15	57.95	49.79
GPA	58.20	57.06	47.82	52.91	45.46

Table 1: Averages across the different streams

3.1.2 GPA Scores Compared

The GPA averages across the five main streams have been calculated for the first semester as well as for the first year. Both these methods are for the year 2014, hence, it reflects the marks achieved by the First-year students. The GPA for the first semester was calculated using a weighted average method. The reader is referred to Pretorius [13] for the method of calculation. The first year GPA scores for this study were provided by BIRAP and were calculated using a similar approach to that done by Pretorius. A slight difference in the GPA scores that were provided is when a student is awarded a supplementary examination. For instance, if a student is awarded a supplementary exam after achieving a semester mark of 45 percent and then achieves a lower mark in the supplementary examination, say, 42 percent, the GPA score is then calculated using the higher of the two marks. In the case of this particular student it will be the semester mark and not the supplementary examination mark which will be used. In addition, GPA scores are calculated on an annual

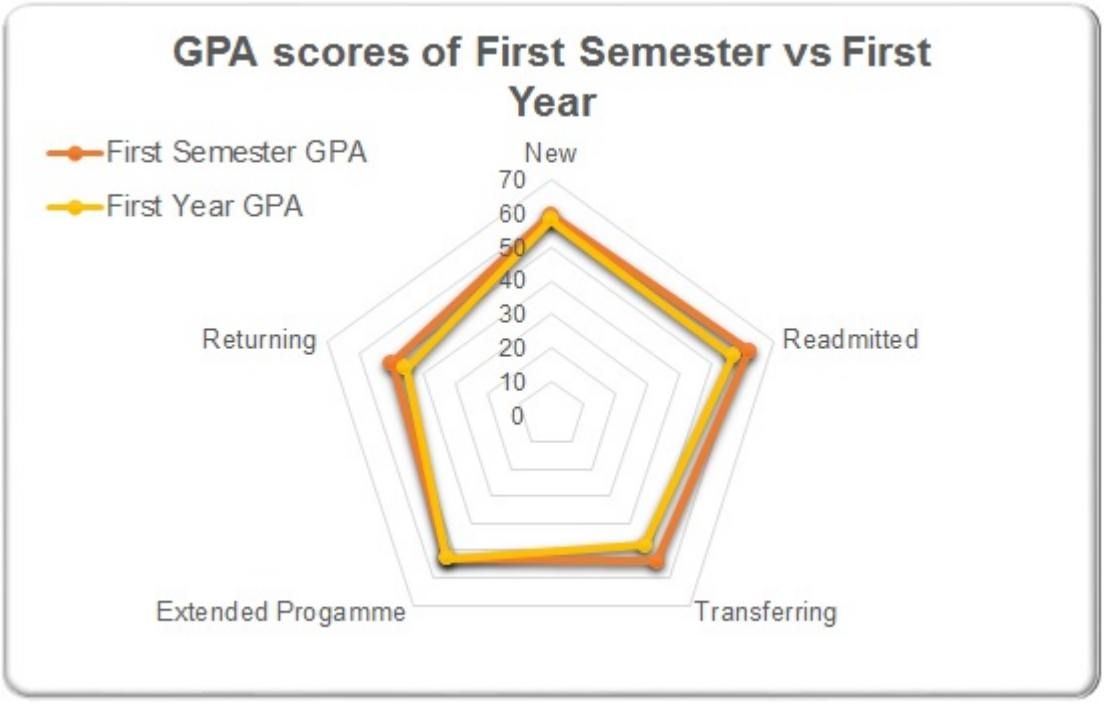


Figure 2: GPA Scores Compared

basis, i.e. all modules, taken over both semesters, enrolled for by a student will comprise the student's GPA score. Moreover, a cumulative weighted GPA is also computed. This means that (as an example) the academic record of a student enrolled for a BCom Statistics degree, where the prescribed period of completion is three years, will after the first year have a weighted average % for the term and a cumulative weighted average. After the second year of study, a new weighted average % for the term is calculated (using the modules enrolled for in the year of study). The cumulative weighted average will now comprise of the weighted average over both years of study. This procedure will then be used for the third year of study and so forth. At UP the weighted average is computed for a year and is thus, unavailable for semesters. A term (year) at UP comprises of two semesters. The use of the word 'term' makes reference to the year of study. The terminology used is specific to UP. In order to avoid confusion, the terminology used by UP for 'weighted average % for the term' equates to the terminology 'GPA scores' used in this report.

Figure 2, shows that when the comparison of GPA scores is made for the different streams there is minimal difference in the averages. Hence, the performance of students was relatively similar between the two semesters. The average from first semester to first year actually improved in the Extended Programme stream. Conversely, there was a slight decrease in the average of the other four streams. The biggest variation in averages was observed in the Transferring stream.

The reader should remain aware of the difference in the calculation methodology used. This may play a role in the obtained averages when making comparisons.

3.2 Enrolment Streams Compared

The different streams are compared to one another by setting up a simple one-way ANOVA table using SAS® software. This analysis will be conducive to see if any stream performs better than the other. The streams will be compared in a between group fashion. The null and alternate hypotheses for all the between group options will be similar. The enrolment streams that are compared are classified into five levels, namely the admittance types.

Note, due to missing observations it is possible that a few students were left out when modeling the

provided data. In other words, not the full set of data provided was utilised.

Before the between group enrolment stream analysis is evaluated, further investigation into the procedure that was applied will be provided. Results will be given regarding the goodness of fit of the model, the variation exposure of the model and if the overall model is of any statistical significance. This information is provided in the form of an ANOVA table using SAS® software. This procedure makes use of ordinary least squares (OLS) as a method in order to fit general linear models.

The observed R^2 between GPA scores and Admittance type is equal to 0.058204. Hence, this is indicative that 5.8204 percent of the variation in the dependent variable (GPA) is explained by the model (i.e. the independent variable, Admittance type). R^2 is also an indication of the goodness of fit of the model. The R^2 value for this model would suggest that it is by no means a good fit, however, R^2 should be treated with caution along with the interpretations thereof.

The overall statistical significance of the model can be evaluated by computing the F statistic from the F-test. The null hypothesis therefore, would be as follows:

H_0: the overall model is not statistically significant against the alternate hypothesis,

H_a: the overall model is statistically significant. The F-value for this model is 35.98 with a corresponding p-value < 0.0001. Hence, since the (p-value < 0.0001) $<$ ($\alpha = 0.05$), it means that the null hypothesis is rejected at the 5 percent level of significance. Thus, concluding that the overall model is significant. If the test was done on a 1 percent level of significance ($\alpha = 0.01$), the null hypothesis would still be rejected in favour of the alternate hypothesis, resulting in the overall model being highly statistically significant, due to the very small p-value. The reader is referred to Figure 3.

In comparing the enrolment streams both the Tukey and Scheffe methods will be evaluated, since the means are being compared and the test is done to see whether the means of the different classes of the independent variable (Admittance type) are in fact similar. The Scheffe method is that of a multiple comparison procedure where Tukey refers to a studentised range test. As Figures 6 and 7 alludes to warnings it should be noted that both these methods control for Type I (type 1) errors in an experimentwise solution. However, the Scheffe method leads to a higher Type II (type 2) error rate than the Tukey method for all between group assessments. Type I errors can be defined as rejecting the null hypothesis when in actual fact the null hypothesis is correct. A Type II error is committed when the null hypothesis is not rejected while in fact the null hypothesis is not correct. The reason for using both Tukey and Scheffe is that the Tukey approach is more accurate in determining confidence intervals, due to the elimination of committing a Type II error. However, when evaluating the grouping (given by alphabetic letters in SAS® software), Scheffe leads to better results. The level of significance or α level for all the between group comparisons is 0.05, this is the same level used to calculate the confidence intervals/limits. The simulated data also provides information regarding the confidence level as well as the difference between the means. Note, the difference in means of the two methods result in the exact same values, the exception is that of the confidence intervals. Confidence intervals can be interpreted as upon repeated samples, 95 percent of the intervals will include the true mean value (zero, in this case).

All the proceeding analysis and evaluation (in section 3.2.1 to 3.2.10) refers to the following Figures:

- 4 (analysis relating to Tukey's Studentised Range test)
- 5 (evaluation of data with regards to Scheffe's test)
- 6 (Tukey's Grouping method of GPA scores)
- 7 (Scheffe's Grouping method of GPA scores)

3.2.1 New students vs Extended Programme students

The between group comparison of New students versus the students enrolled for the Extended Programme will be evaluated. The null hypothesis is described as follows:

$H_0 : \mu_{new} - \mu_{extended} = 0$ (New students' average performance is statistically the same as the average performance of Extended Programme students) whereas the alternate hypothesis (H_a) is: not all the mean differences ($\mu_{new} - \mu_{extended}$) are equal to zero. Note: μ refers to the mean; extended makes reference to the Extended Programme. From Figure 4 it can be concluded, using the Tukey method that the difference

Comparing the Admittance type of first year students
Analysis of Variance Table using the glm

The GLM Procedure

Class Level Information

Class	Levels	Values
admittance_type	5	Extended Programme New Readmit Returning Transferring

Number of Observations Read	2334
Number of Observations Used	2334

Comparing the Admittance type of first year students
Analysis of Variance Table using the glm

The GLM Procedure

Dependent Variable: gpa gpa

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	49066.7081	12266.6770	35.98	<.0001
Error	2329	793949.4446	340.8971		
Corrected Total	2333	843016.1527			

R-Square	Coeff Var	Root MSE	gpa Mean
0.058204	33.25816	18.46340	55.51540

Source	DF	Type I SS	Mean Square	F Value	Pr > F
admittance_type	4	49066.70810	12266.67703	35.98	<.0001

Source	DF	Type III SS	Mean Square	F Value	Pr > F
admittance_type	4	49066.70810	12266.67703	35.98	<.0001

Figure 3: Admittance types compared

between the means is 12.7417 with a confidence interval of (7.7187 ; 17.7647). Hence, because zero does not fall within the interval it means that the null hypothesis is rejected at the 5 percent level of significance. Thus, this between group comparison between these two streams deems to be significant and indicates that the average performance of New students is in actual fact better than those enrolled in the Extended Programme, since the difference yielded a positive answer. Scheffe's multi comparisons yields the following results (using the same null and alternate hypotheses): difference of means is the exact same as the Tukey method with a slight difference being observed with regards to the confidence interval which is (7.0696 ; 18.4138). Thus, the null hypothesis will still be rejected.

3.2.2 New students vs Transferring students

New students' performance for First-year will be compared to that of Transferring students. This will distinguish if there is an apparent better level of success within one of these streams or not. Once again the null and alternate hypotheses are:

$H_0 : \mu_{new} - \mu_{transferring} = 0$ (New students' average performance is the same as the average performance of Transferring students)

H_a : not all the mean differences are equal

According to the Tukey method, the difference of means between these two streams is 5.2896. The 95 percent confidence interval of this between group comparison is (-0.1906 ; 10.7698). Since the value of zero falls within this confidence interval, it results in the null hypothesis not being rejected at the 5 percent level of significance. Hence, the comparison of the two streams is not statistically significant and can in essence be left out of the model. It also provides the reasoning that New students do not tend to perform better than Transferring students at a First-year level when using the variable GPA. Scheffe's method leads to confidence intervals equal to (-0.8987 ; 11.4780). A wider interval means that there is more acceptance to an error being made. The between group comparison is still statistically insignificant.

3.2.3 New students vs Readmitted students

The third between group comparison is that of New students versus students that were Readmitted into the Faculty of EMS. The Readmitted students account for the lowest entrant numbers as in this study, only 3 students that are part of the study are in this category. The hypotheses can be described as follows:

$H_0 : \mu_{new} - \mu_{readmitted} = 0$ (New students' average performance is the same as the average performance of Readmitted students)

H_a : not all the mean differences are equal

Tukey's and Scheffe's methods reveals a difference in means of 1.1407, with a confidence interval of (-27.9853 ; 30.2667) and (-31.7492 ; 34.0306) for the methods, respectively. Since the value of zero falls inside both of these confidence intervals, the outcome is that the null hypothesis is not rejected. Therefore, the streams entitled New and Readmitted are not statistically significantly different. This suggests that New students do not necessarily perform better than Readmitted students and vice versa.

3.2.4 New students vs Returning students

Finally, the last between group comparison that involves New students is that with Returning students. The analysis done will assess whether there are higher GPA scores, if any, amongst these two streams. The null and alternate can be given as:

$H_0 : \mu_{new} - \mu_{returning} = 0$ (New students' average performance is the same as the average performance of Returning students)

H_a : not all the mean differences are equal

The confidence intervals for the Tukey and Scheffe methods are respectively given by the following intervals, (7.6585 ; 13.1127) and (7.3061 ; 13.4651). The difference of means between the New students and Returning students grouping equals 10.3856. It can be seen that the mean value level of zero does not fall within both the confidence intervals which leads to the conclusion that the null hypothesis is rejected in favour of the alternate hypothesis at the 5 percent level of significance. The between group comparison is

statistically significant. Hence, New students perform better than Returning students, in other words they have higher GPA scores than Returning students.

3.2.5 Readmitted students vs Returning students

When the Readmitted students are compared to those that are Returning to UP and more specifically to the Faculty of EMS the following results were observed, a difference in means level of 9.2449. The confidence interval when the Tukey method was performed is (-19.9576 ; 38.4474) whilst the Scheffe method yields a confidence interval of (-23.7314 ; 42.2211). The above was achieved using the following hypotheses:

$H_0 : \mu_{readmitted} - \mu_{returning} = 0$ (Readmitted students' average performance is the same as the average of Returning students)

H_a : not all the mean differences are equal

The mean value equal to zero falls within the intervals thus, meaning that the null hypothesis is not rejected. Therefore, the between group comparison reveals that the streams are indeed statistically insignificant and that Readmitted students do not have statistically better GPA scores than Returning students. The opposite also holds true, meaning if the null hypothesis now read: $H_0 : \mu_{returning} - \mu_{readmitted} = 0$ (Returning students' average performance is the same as the average of Readmitted students), the same conclusion can be made.

3.2.6 Readmitted students vs Transferring students

The next between group comparison will detect whether Readmitted students or students that are Transferring to the Faculty of EMS from another faculty within UP do indeed have statistically higher GPA scores than one another. This will be analysed using confidence intervals (both Tukey and Scheffe), where the hypotheses are:

$H_0 : \mu_{readmitted} - \mu_{transferring} = 0$ (Readmitted students' average performance is the same as the average performance of Transferring students)

H_a : not all the mean differences are equal

The confidence intervals are (-25.4380 ; 33.7358) and (-29.2614 ; 37.5592). The analysis reveals three important notions, firstly that the null hypothesis is not rejected at the 5 percent level of significance. Secondly, that the comparison between these two streams is statistically insignificant and lastly, it can be concluded that Readmitted students do not have a better performance than Transferring students with respect to GPA scores.

3.2.7 Readmitted students vs Extended Programme students

The last between group comparison involving Readmitted students is this stream versus students that are enrolled for the four-year Extended Programme. The variable of interest is GPA scores. The null and alternate hypotheses can be described as follows:

$H_0 : \mu_{readmitted} - \mu_{extended} = 0$ (Readmitted students' average performance is the same as the average performance of Extended Programme students)

H_a : not all the mean differences are equal

Note: extended in this context refers to the Extended Programme. The confidence interval using Tukey's method is (-17.9046 ; 41.1066) whereas (-21.7176 ; 44.9195) are the lower and upper limits, respectively using Scheffe's method. The difference between the mean levels for these two streams is 11.601. The confidence intervals can be used to not reject the null hypothesis as the mean value under consideration (that is, a mean value = 0) falls within the intervals. Hence, the comparison is statistically insignificant and could in essence be left out of model comparisons. Therefore, Readmitted students do not statistically perform better than those students who are enrolled in the Extended Programme.

3.2.8 Returning students vs Transferring students

In this subsection, analysis will be evaluated to determine whether the between group comparison between Returning students and Transferring students does indeed lead to higher First-year GPA scores or not. The

hypotheses that are used are given by the following null and alternate hypotheses:

$H_0 : \mu_{returning} - \mu_{transferring} = 0$ (Returning students' average performance is the same as the average performance of Transferring students)

H_a : not all the mean differences are equal

In applying Tukey's method to confidence intervals the following is obtained (-5.096 ; 0.7773) whilst (-11.7282 ; 1.5362) is obtained using Scheffe's method. As can be seen the mean value = 0 just falls into the confidence interval limits. Concluding, the null hypothesis is not rejected at the 5 percent level of significance. Thus, the comparison is in actual fact statistically insignificant. Hence, Returning students do not have higher First-year GPA scores than Transferring students and vice versa.

3.2.9 Returning students vs Extended Programme students

The performance level of First-year students who are Returning versus students that are part of the Faculty of EMS's Extended Programme will be investigated. This is to determine and give guidance to the Faculty of EMS if there is higher GPA scores between these two streams. The results when simulated using SAS® software were as follows:

- Difference between means: 2.3561
- Tukey confidence interval: (-3.093 ; 7.8052)
- Scheffe confidence interval: (-3.7972 ; 8.5094)

Therefore, the null and alternate hypotheses are as follows:

$H_0 : \mu_{returning} - \mu_{extended} = 0$ (Returning students' average performance is the same as the average performance of Extended Programme students)

H_a : not all the mean differences are equal

Note: extended makes reference to the Extended Programme. The mean value equal to zero falls within the above confidence intervals and thus, do not reject the null hypothesis at the 5 percent level of significance. It appears that the comparison is of no statistical significance. Hence, Returning students do not necessarily perform better than the Extended Programme students.

3.2.10 Transferring students vs Extended Programme students

Finally, the last between group comparison that will be done in this report is that of the performance of Transferring students versus the performance of the Extended Programme students at a First-year level. The null and alternate hypotheses are similar to those used previously with:

$H_0 : \mu_{transferring} - \mu_{extended} = 0$ (Transferring students' average performance is the same as the average performance of Extended Programme students)

H_a : not all the mean differences are equal

Note: extended makes reference to the Extended Programme. The lower limit of the confidence interval equals 0.2210 with a corresponding upper limit equaling 14.6831. Results refer to the Tukey method. The Scheffe method yields a lower confidence limit of -0.7134 and an upper limit of 15.6176. Depending on which method is applied, the rejection of the null hypothesis and conclusion will differ somewhat. This is due to the Type II errors that play a role (in the Scheffe method). Therefore, the Tukey method explained: the mean value of zero does not fall in the confidence interval, thus, the comparison is statistically significant and reveals that Transferring students do indeed perform better than the Extended Programme students at a First-year level, with GPA scores as the dependent variable of interest. Conversely, The Scheffe method yields results in which the null hypothesis is not rejected at the 5 percent level of significance, meaning that the comparison between these two streams is statistically insignificant, hence, there is no difference in performance from either of these groups.

The GLM Procedure

Tukey's Studentized Range (HSD) Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	2329
Error Mean Square	340.8971
Critical Value of Studentized Range	3.86068

Comparisons significant at the 0.05 level are indicated by ***.

admittance_type Comparison		Difference Between Means	Simultaneous 95% Confidence Limits		
New	- Readmit	1.1407	-27.9853	30.2667	
New	- Transferring	5.2896	-0.1906	10.7698	
New	- Returning	10.3856	7.6585	13.1127	***
New	- Extended Programme	12.7417	7.7187	17.7647	***
Readmit	- New	-1.1407	-30.2667	27.9853	
Readmit	- Transferring	4.1489	-25.4380	33.7358	
Readmit	- Returning	9.2449	-19.9576	38.4474	
Readmit	- Extended Programme	11.6010	-17.9046	41.1066	
Transferring	- New	-5.2896	-10.7698	0.1906	
Transferring	- Readmit	-4.1489	-33.7358	25.4380	
Transferring	- Returning	5.0960	-0.7773	10.9692	
Transferring	- Extended Programme	7.4521	0.2210	14.6831	***
Returning	- New	-10.3856	-13.1127	-7.6585	***
Returning	- Readmit	-9.2449	-38.4474	19.9576	
Returning	- Transferring	-5.0960	-10.9692	0.7773	
Returning	- Extended Programme	2.3561	-3.0930	7.8052	
Extended Programme	- New	-12.7417	-17.7647	-7.7187	***
Extended Programme	- Readmit	-11.6010	-41.1066	17.9046	
Extended Programme	- Transferring	-7.4521	-14.6831	-0.2210	***
Extended Programme	- Returning	-2.3561	-7.8052	3.0930	

Figure 4: Tukey's Studentised Range Test for GPA score

The GLM Procedure

Scheffe's Test for gpa

NOTE: This test controls the Type I experimentwise error rate, but it generally has a higher Type II error rate than Tukey's for all pairwise comparisons.

Alpha	0.05
Error Degrees of Freedom	2329
Error Mean Square	340.8971
Critical Value of F	2.37575

Comparisons significant at the 0.05 level are indicated by ***.

admittance_type Comparison		Difference Between Means	Simultaneous 95% Confidence Limits		
New	- Readmit	1.1407	-31.7492	34.0306	
New	- Transferring	5.2896	-0.8987	11.4780	
New	- Returning	10.3856	7.3061	13.4651	***
New	- Extended Programme	12.7417	7.0696	18.4138	***
Readmit	- New	-1.1407	-34.0306	31.7492	
Readmit	- Transferring	4.1489	-29.2614	37.5592	
Readmit	- Returning	9.2449	-23.7314	42.2211	
Readmit	- Extended Programme	11.6010	-21.7176	44.9195	
Transferring	- New	-5.2896	-11.4780	0.8987	
Transferring	- Readmit	-4.1489	-37.5592	29.2614	
Transferring	- Returning	5.0960	-1.5362	11.7282	
Transferring	- Extended Programme	7.4521	-0.7134	15.6176	
Returning	- New	-10.3856	-13.4651	-7.3061	***
Returning	- Readmit	-9.2449	-42.2211	23.7314	
Returning	- Transferring	-5.0960	-11.7282	1.5362	
Returning	- Extended Programme	2.3561	-3.7972	8.5094	
Extended Programme	- New	-12.7417	-18.4138	-7.0696	***
Extended Programme	- Readmit	-11.6010	-44.9195	21.7176	
Extended Programme	- Transferring	-7.4521	-15.6176	0.7134	
Extended Programme	- Returning	-2.3561	-8.5094	3.7972	

Figure 5: Scheffe's Test for GPA scores

The GLM Procedure

Tukey's Studentized Range (HSD) Test for gpa

NOTE: This test controls the Type I experimentwise error rate, but it generally has a higher Type II error rate than REGWQ.

Alpha	0.05
Error Degrees of Freedom	2329
Error Mean Square	340.8971
Critical Value of Studentized Range	3.86068
Minimum Significant Difference	19.043
Harmonic Mean of Cell Sizes	14.01178

NOTE: Cell sizes are not equal.

Means with the same letter are not significantly different.

Tukey Grouping	Mean	N	admittance_type
A	58.203	1708	New
A			
A	57.062	3	Readmit
A			
A	52.913	89	Transferring
A			
A	47.817	427	Returning
A			
A	45.461	107	Extended Programme

Figure 6: Tukey Grouping of GPA scores

The GLM Procedure

Scheffe's Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	2329
Error Mean Square	340.8971
Critical Value of F	2.37575
Minimum Significant Difference	21.504
Harmonic Mean of Cell Sizes	14.01178

NOTE: Cell sizes are not equal.

Means with the same letter are not significantly different.

Scheffe Grouping	Mean	N	admittance_type
A	58.203	1708	New
A			
A	57.062	3	Readmit
A			
A	52.913	89	Transferring
A			
A	47.817	427	Returning
A			
A	45.461	107	Extended Programme

Figure 7: Scheffe Grouping of GPA scores

3.3 Independent samples

This section will evaluate whether or not a certain matric authority description or simply put, school type plays a role in determining student success at the tertiary level. Students will be divided into four groups; Cambridge, State School, IEB and Foreign Country, in accordance with the school that they attended. State School refers to students who attended public schools in South Africa, i.e. the nine provincial education departments. Secondly, IEB (Independent Examinations Board) refers to private schools within the South African schooling system. Certain assumptions are made, such as if a student attended 'School of Tomorrow' or 'International Bacalaureat', are considered to be IEB. Foreign Country makes reference to students who did not attend school in South Africa.

The application done uses the Kruskal-Wallis test, this is due to the fact that the analysis of matric authority description contains a single independent variable with two or more levels and a dependent variable with a non-normal distribution. In the study and the aforementioned paragraph, the independent variable is thus, the matric authority description, with four levels, namely, Cambridge, State School, IEB and finally Foreign Country. Furthermore, the dependent variable of choice is the GPA scores, this is in accordance with what this report is aiming to achieve. Hence, the Kruskal-Wallis test is the best option. The Kruskal-Wallis test is a nonparametric (or distribution free) version of a one-way ANOVA table. In addition the Kruskal-Wallis is a generalised form of the Wilcoxon-Mann-Whitney. Assumptions that are made include that the data (divided data) comes from the same distribution with each having a different mean value. This is apparent from section 3.1.1. Secondly, it is assumed that the variable GPA is not normally distributed. The Kruskal-Wallis test is based on ranking the data. It should be noted that due to certain information being unavailable, such as the matric authority description, a few students have been left out of this particular analysis.

3.3.1 Krsukal - Wallis Test

The following hypothesis will be tested with reference to the main objective of this section (Kruskal-Wallis, see Figure 8):

H_0: there is no statistical difference between the different matric authority description against the alternate hypothesis (H_a) there is statistical significance between the different matric authority description. The test will be performed at a 5 percent level of significance, in other words, $\alpha = 0.05$. The Kruskal-Wallis test has a χ^2 (Chi-Square) value equal to 9.6232. The degrees of freedom equates to three, this is computed as the number of parameters less one. In this case, the parameters are the four classified group types, i.e. the matric authority description. The corresponding p-value of the Kruskal-Wallis test is 0.0221. Hence, since the (p-value = 0.0221) $<$ ($\alpha = 0.05$), it means that the null hypothesis (H_0) is rejected at the 5 percent level of significance. It can therefore, be concluded that there is statistical significance amongst the matric authority description, meaning that based on a certain school type a student attended could lead to those students being more successful at a tertiary level. This analysis is all based upon Figure 8.

Further analysis can be of interest. The ANOVA table classified by matric authority description for the variable GPA is provided and it can be concluded that students who attended Cambridge had the highest mean, equal to 62.13. The lowest mean GPA value was 52.85, students who attended school outside South Africa. Once again averages (mean levels) should be treated with caution.

The overall model, where the null and alternate hypotheses are as follows:

H_0: the overall model is not statistically significant;

H_a: the overall model is statistically significant.

This test is computed using an F-test with a 5 percent level of significance. The F-value = 3.1775 with a corresponding p-value = 0.0232 (See Figure 8). Hence, since the (p-value = 0.0232) $<$ ($\alpha = 0.05$), it means that the null hypothesis is rejected at the 5 percent level of significance. Thus, the overall model is significant.

Figure 8 also identifies the Wilcoxon, Rank Sums, where information is provided for the different matric authority descriptions in terms of what the expected value and standard deviations are under the null hypothesis (no statistical difference between the matric authority description). It should be noted (alluded to using SAS® software) that if there is a tie in the ranks, when ranking the GPA scores, ties would mean that average scores are applied. From looking at the observed mean scores of the Wilcoxon Scores (Rank Sums) for GPA, the rankings would be:

1. Cambridge (1409.875)
2. IEB (1169.93548)
3. State School (1141.60438)
4. Foreign country (1072.375)

Rankings are based on the mean scores in ascending order, i.e from largest to smallest. From the outset it would seem that there is a possible difference in performance between students who attended Cambridge and those that attended school outside the South African boarders. Comparison of mean differences will be performed to determine if the matric authority description differs.

Nonparametric test for Matric Authority

The NPAR1WAY Procedure

Analysis of Variance for Variable gpa
Classified by Variable matric_authority_description

matric_ authority_ description	N	Mean
Cambridge	56	62.130518
State School	1782	55.119640
Foreign Country	32	52.849813
IEB	434	56.564912

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Among	3	3434.524307	1144.841436	3.1775	0.0232
Within	2300	828679.923430	360.295619		

Average scores were used for ties.

Nonparametric test for Matric Authority

The NPAR1WAY Procedure

Wilcoxon Scores (Rank Sums) for Variable gpa
Classified by Variable matric_authority_description

matric_ authority_ description	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
Cambridge	56	78953.0	64540.0	4917.1968	1409.87500
State School	1782	2034339.0	2053755.0	13366.4021	1141.60438
Foreign Country	32	34316.0	36880.0	3736.8406	1072.37500
IEB	434	507752.0	500185.0	12485.0730	1169.93548

Average scores were used for ties.

Kruskal-Wallis Test

Chi-Square	9.6232
DF	3
Pr > Chi-Square	0.0221

Figure 8: Wilcoxon Scores and the Kruskal - Wallis test

3.3.2 van der Waerden normal scores

From Figure 9, the van der Waerden normal scores are also provided. This test shows whether or not the distribution of the population functions are equal. Hence, in the context of this report this would correlate to whether the matric authority description populations come from the same distribution. In other words, do

Nonparametric test for Matric Authority

The NPAR1WAY Procedure

Van der Waerden Scores (Normal) for Variable gpa
Classified by Variable matric_authority_description

matric_ authority_ description	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
Cambridge	56	21.342261	0.0	7.351650	0.381112
State School	1782	-33.877960	0.0	19.983969	-0.019011
Foreign Country	32	-4.433784	0.0	5.586912	-0.138556
IEB	434	16.969482	0.0	18.666303	0.039100

Average scores were used for ties.

Van der Waerden One-Way Analysis

Chi-Square	10.1659
DF	3
Pr > Chi-Square	0.0172

Figure 9: van der Waerden Normal Scores

Cambridge, State Schools, IEB and Foreign schooling all follow the same distribution, irrespective of whether that distribution is known or unknown. However, the Kruskal-Wallis test was performed which makes the assumption that the distributions are not normally distributed, whereas the van der Waerden makes use of a normal distribution. The van der Waerden test converts the ranks as determined by the Kruskal-Wallis test into quantiles. These quantiles follow a normal distribution. The hypotheses for this test can be given as follows:

H_0: all the population distribution functions are the same

H_a: at least one of the population distribution functions has larger observations than at least one of the other population distribution functions.

An explanation of the alternate hypothesis is for instance, the matric authority description labeled Cambridge may yield higher GPA scores (in normally distributed quantiles) than say the IEB category. Continuing the example using numerical values, perhaps Cambridge yields a value that is equal to it's rank (say one) whereas IEB might be ranked fourth thus, having the lowest GPA scores.

In evaluating the hypotheses, the van der Waerden test statistic equals 10.1659 with three degrees of freedom. The degrees of freedom are calculated as the number of parameters (Cambridge, State School, IEB and finally, Foreign Country schooling) less one. The test statistic is given by a χ^2 (Chi-Square) value. The corresponding p-value is equal to 0.038. The p-value is evaluated against a 5 percent level of significance ($\alpha = 0.05$). Since, the (p-value $= 0.0380$) $<$ ($\alpha = 0.05$), it leads to the conclusion that the null hypothesis is rejected at the 5 percent level of significance in favour of the alternate hypothesis. This means that the four different matric authority descriptions are indeed from population functions that come from different distributions. Thus, there is an indication that at least one population yields different (possibly higher) GPA scores than at least one of the remaining three populations.

3.3.3 Wilcoxon scores

The interpretation of the boxplot of Wilcoxon scores for GPA scores given in Figure 10 will be discussed. As can be seen, the boxplot of Foreign Country is expanded, in other words is quite stretched out, this is an indication that within the category there is a wide difference in GPA scores obtained by students. Furthermore, it can be seen that the boxplot of State School is slightly lower than that of Cambridge, this advocates that there maybe a difference in GPA scores between the two groups, however, this may not hold true as there is only a slight difference in the placing of the boxplot. In addition, there are obvious differences in the boxplots of Cambridge and Foreign Country. In other words, the length of the whiskers of

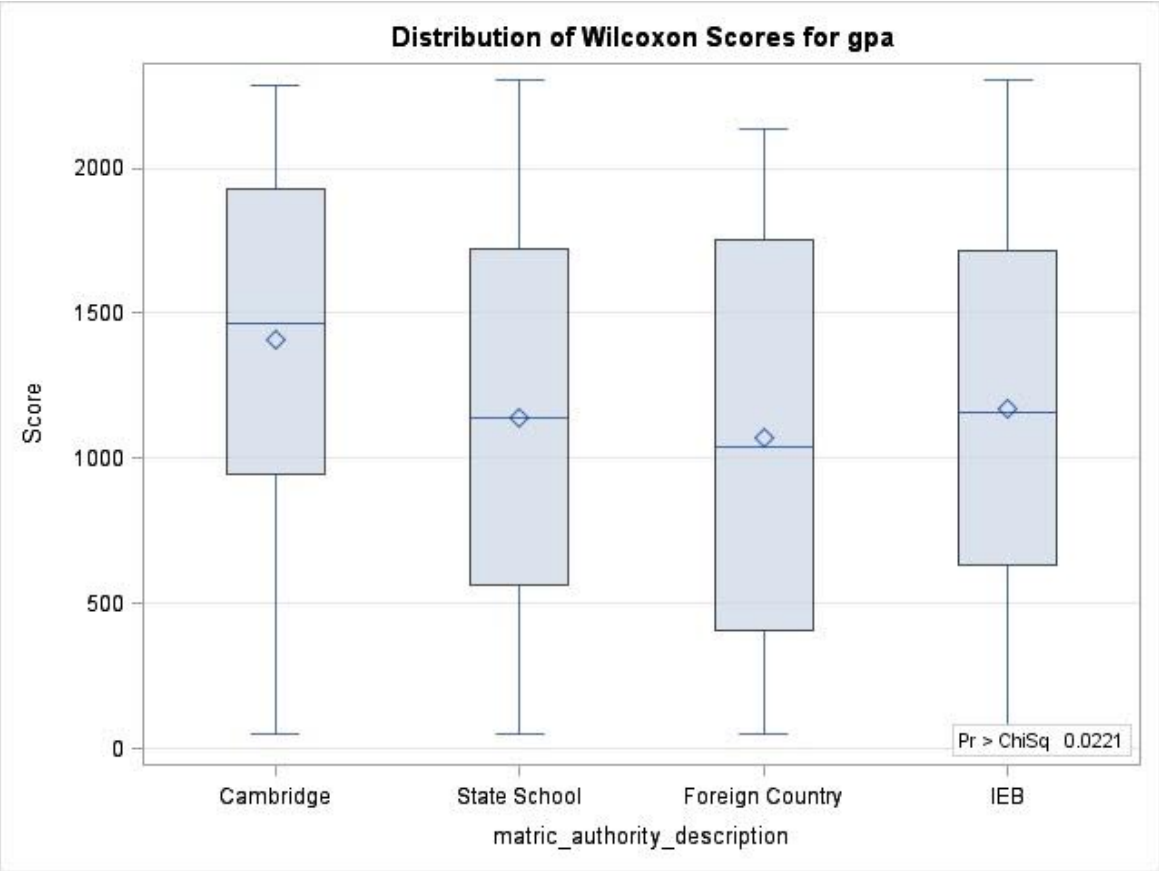


Figure 10: Boxplot of Wilcoxon scores for GPA

the respective boxplots as well as the box itself. Due to this obvious difference, further investigation will be explored in order to see if there is in fact a difference between the GPA scores of those two groups. It can also be seen that State Schools and IEB have very similar median values (50th percentile) for the Wilcoxon scores of GPA, this could be indicative that these are roughly the same as the overall median level, but these independent variables maybe from different population distributions. This will also be investigated further to figure out if the population distributions are indeed different. Finally, none of the boxplots reveal that the four sections are uneven in size. What is meant by this is that from the 5th percentile to the 25th percentile is not all that unequal to the next section (25th percentile to the 50th percentile) and so forth. The exception being that of Cambridge but further investigation reveals that the sections of the 25th percentile to the 50th percentile and between the 50th - 75th percentiles are relatively equal in size. The p-value is also given which is 0.0221.

Knowing that differences were detected, further investigation was conducted to evaluate which populations of matric authority description are contrasting.

3.3.4 Matric authority descriptions compared

Using Figure 11, it can be seen that there is a distinct difference between the GPA score performance of students that attended Cambridge as opposed to those that attended school elsewhere. This can be seen from the Scheffe grouping which indicates Cambridge is assigned a letter 'A' whilst Foreign Country is assigned the letter 'B'. This shows that they are statistically significantly different. Furthermore, there is no statistical significance between the following matric authority descriptions:

- Cambridge, IEB and State School (all grouped with the alphabetic letter 'A')

Nonparametric test for Matric Authority

The GLM Procedure

Scheffe's Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	2300
Error Mean Square	360.2956
Critical Value of F	2.60877
Minimum Significant Difference	8.5601
Harmonic Mean of Cell Sizes	76.96384

NOTE: Cell sizes are not equal.

Means with the same letter are not significantly different.

Scheffe Grouping	Mean	N	matric_ authority_ description
A	62.131	56	Cambridge
A			
B A	56.565	434	IEB
B A			
B A	55.120	1782	State School
B			
B	52.850	32	Foreign Country

Figure 11: Matric Authority descriptions compared

- IEB, State School and Foreign Country (all grouped with the alphabetic letter 'B')

Hence, it can be concluded that the First-year performance of students that attended Cambridge is indeed better than the performance of students that did not attend the South African schooling systems (i.e Foreign Country students).

3.4 Success prediction

This section consists of various procedures that were performed to determine the best level of success amongst the streams, if any. The dependent variable used in this analysis is GPA. Categorical variables that were used in the models is the admittance type descriptions (i.e. the streams), gender description (i.e. female or male) and finally, citizenship country description type (i.e. South Africa or International). International comprises of all students who are not South African. In Pretorius [13], a graphical representation of the citizenship country description is provided. Along with these independent variables others include, NBT mathematics, NBT academic literacy, NBT quantitative literacy as well as Grade 11 APS and Grade 12 APS. Provisional admittance into UP is based on the APS score achieved by potential students in Grade 11. The final decision made is based on Grade 12 APS scores. For these reasons these variables were included. It should be noted that the proceeding models are built only using information that was available for all the aforementioned variables. In other words, due to non-response in certain fields only 845 students out of the study total of 2 359 students were eligible. Furthermore, due to information regarding the Readmitted stream also being unavailable, the stream was thus, also excluded.

Note: the level of significance (α) used in all the models is 5 percent, unless otherwise stated.

3.4.1 Complete model

This model makes use of the variables defined above, where the independent variables consists of the categorical variables as well as the three NBT categories and the two APS groups. In the overall model (not taking dummy variables into account), with reference to Figure 12, it can be deduced that the categorical variables have different levels. The following hypotheses can be used to describe the overall model:

H_0: the overall model is not statistically significant

H_a: the overall model is statistically significant

The p-value obtained is < 0.0001. Hence, the following two important notions can be made:

1. The null hypothesis (H_0) is rejected in favour of the alternate hypothesis. This decision is made on the basis that the p-value < 0.001 is less than the level of significance ($\alpha < 0.05$).

2. The overall model is indeed statistically significant.

The R^2 value equals 0.285860, meaning that 28.59 percent of the variation in GPA scores can be explained by the model (i.e. all the independent variables). R^2 is also used as a measure for goodness of fit. This means how well does a model fit, hence, the goodness of fit should be interpreted with caution.

The SAS System

The GLM Procedure

Class Level Information

Class	Levels	Values
admit_type	4	Extended_programme New Returning Transferring
gender_description	2	Female Male
citizenship_country_description	2	International South Africa

Number of Observations Read 845
Number of Observations Used 845

The SAS System

The GLM Procedure

Dependent Variable: gpa gpa

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	59	77078.9490	1306.4229	5.33	<.0001
Error	785	192559.6355	245.2989		
Corrected Total	844	269638.5845			

R-Square	Coeff Var	Root MSE	gpa Mean
0.285860	27.51280	15.66202	56.92631

Figure 12: Overall Model

Furthermore, dummy variables are used due to categorical variables being included in the model. The number of dummy variables is calculated as the number of levels for a certain categorical value less one. For example, in this model there are four levels under the categorical variable of admittance type, hence three dummy variables will be used. The same applies to the other categorical variables. SAS® software refers to these as 'class' (see Figure 12).

The following are used as the base dummy variable under the respective class of admittance type, gender description and citizenship country description:

- Transferring
- Male

- South Africa

A base dummy variable implies that the parameter is set equal to zero in the matrix. For example, the representation of the admittance type using dummy variables is given in Table 2:

	x_1	x_2	x_3
New	1	0	0
Returning	0	1	0
Readmitted	0	0	1
Transferring	0	0	0

Table 2: Example of Dummy variables

An identical process will be applied when using dummy variables for gender description as well as citizenship country description. In the case of gender, as mentioned, the base dummy variable is male hence, male will be represented by a zero whereas female will be a one. Similarly, South African students are considered to be the based variable, thus, equals zero whilst International students are assigned a one value.

By making use of using SAS® software (see Figure 13), the choice of base dummy variables was made automatically. This allows for obtaining parameters that are unbiased and not linear combinations of each other. Furthermore, in order to perform regression analysis on the data, a full-rank parameterisation is needed. Full-rank parameterisation can be described as being able to estimate parameters even when categorical variables are included in the model. Hence, a standard regression procedure was not applied because there are categorical variables included in the model. Certain interactions between the variables can also be analysed. Interactions refers to for example, a student that is female, enrolled for the Extended Programme and citizenship is International etc.

The null and alternate hypotheses for all the parameters in general, excluding the interaction terms, can be described as:

H_0: The variable is not statistically significant; against the alternate hypothesis of,

H_a: The variable is statistically significant

In order to be able to interpret the individual p-values, the other variables have to already be included in the model and remain constant.

Referring to Figure 13, it shows that the p-values for all the streams are less than the level of significance ($\alpha = 0.05$). The p-values of 0.0017, 0.0122 and 0.0053 are for the Extended Programme, New and Returning streams, respectively. This means that the null hypothesis is rejected at the 5 percent level of significance. Hence, the variables are statistically significant and need to be included in the model. Furthermore, it can be concluded (using p-values) that the gender description variable is statistically significant. The citizenship country description (p-value = 0.2221) of students is not statistically significant and can in essence be left out of the model. Hence, it means that whether students are from South Africa or are International students is of no significance. Thus, this would not lead to higher levels of success amongst students.

When the three NBT categories are analysed, namely academic literacy, quantitative literacy and mathematics, none of them are statistically significant. This is as a result of the null hypothesis not being rejected at the 5 percent level of significance. The p-values are thus, in excess of the level of significance. These variables can also be left out of the model.

Furthermore, using the above null and alternate hypotheses, the Grade 12 APS score is statistically significance. This is since the (p-value = 0.0122) $<$ ($\alpha = 0.05$). Hence, the null hypothesis is rejected in favour of the alternate hypothesis. Conversely, the Grade 11 APS is not statistically significant.

The interaction terms, generally speaking, have the following null and alternate hypotheses:

H_0: The interaction between the variables is not statistically significant

H_a: The interaction between the variables is statistically significant

The following interaction terms (with reference to Figure 13) are statistically significant:

- Extended Programme, Female, International (p-value = 0.0207)
- New, Female, International (p-value = 0.0268)

The SAS System

The GLMSELECT Procedure
Least Squares Model (No Selection)

Parameter Estimates

Parameter		DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept		1	70.335000	18.280699	3.85	0.0001
admit_type	Extended_programme	1	-39.980554	12.663968	-3.16	0.0017
admit_type	New	1	-29.949314	11.916918	-2.51	0.0122
admit_type	Returning	1	-33.199701	11.884965	-2.79	0.0053
gender_description	Female	1	-31.828361	10.649881	-2.99	0.0029
citizenship_country_	International	1	-35.375401	28.953562	-1.22	0.2221
nbt_academic_literac		1	0.109582	0.118737	0.92	0.3563
nbt_quantitative_lit		1	-0.141985	0.102985	-1.38	0.1684
nbt_maths		1	-0.036566	0.108121	-0.34	0.7353
aps_grade_11		1	-0.500296	0.337360	-1.48	0.1385
aps_grade_12		1	0.959922	0.381937	2.51	0.0122
admit_*gender*citize	Extended_programme Female International	1	73.698212	31.787548	2.32	0.0207
admit_*gender*citize	New Female International	1	-16.580887	7.473042	-2.22	0.0268
admit_*gender*citize	Returning Female International	1	30.878976	22.175614	1.39	0.1642
nbt_ac*admit_*gender	Extended_programme Female	1	-0.017036	0.416617	-0.04	0.9674
nbt_ac*admit_*gender	New Female	1	-0.049315	0.140567	-0.35	0.7258
nbt_ac*admit_*gender	Returning Female	1	0.102518	0.241241	0.42	0.6710
nbt_qu*admit_*gender	Extended_programme Female	1	0.192368	0.407502	0.47	0.6370
nbt_qu*admit_*gender	New Female	1	0.104213	0.126769	0.82	0.4113
nbt_qu*admit_*gender	Returning Female	1	-0.254337	0.194659	-1.31	0.1917
nbt_ma*admit_*gender	Extended_programme Female	1	0.732536	0.444705	1.65	0.0999
nbt_ma*admit_*gender	New Female	1	0.180551	0.120182	1.50	0.1334
nbt_ma*admit_*gender	Returning Female	1	0.367175	0.195846	1.87	0.0612
aps_gr*admit_*gender	Extended_programme Female	1	0.863261	1.305614	0.66	0.5087
aps_gr*admit_*gender	New Female	1	0.524546	0.415636	1.26	0.2073
aps_gr*admit_*gender	Returning Female	1	0.495181	0.630438	0.79	0.4324
aps_gr*admit_*gender	Extended_programme Female	1	-0.501026	1.301782	-0.38	0.7004
aps_gr*admit_*gender	New Female	1	0.275969	0.454103	0.61	0.5435
aps_gr*admit_*gender	Returning Female	1	0.337987	0.684688	0.49	0.6217
nbt_ac*admit_*citize	Extended_programme International	1	4.753984	2.420240	1.96	0.0498
nbt_ac*admit_*citize	New International	1	0.141220	0.357547	0.39	0.6930
nbt_ac*admit_*citize	Returning International	1	-0.634500	1.801017	-0.35	0.7247
nbt_qu*admit_*citize	Extended_programme International	1	-1.933650	1.293303	-1.50	0.1353
nbt_qu*admit_*citize	New International	1	-0.144140	0.245273	-0.59	0.5569
nbt_qu*admit_*citize	Returning International	1	0.420644	0.737126	0.57	0.5684
nbt_ma*admit_*citize	Extended_programme International	1	0.332076	1.028052	0.32	0.7468
nbt_ma*admit_*citize	New International	1	-0.039822	0.216733	-0.18	0.8543
nbt_ma*admit_*citize	Returning International	1	0.169592	1.865965	0.09	0.9276
aps_gr*admit_*citize	Extended_programme International	1	3.362211	2.954006	1.14	0.2554
aps_gr*admit_*citize	New International	1	-1.007973	0.906829	-1.11	0.2667
aps_gr*admit_*citize	Returning International	1	-0.206154	2.844932	-0.07	0.9423
aps_gr*admit_*citize	Extended_programme International	1	-9.478669	4.462641	-2.12	0.0340
aps_gr*admit_*citize	New International	1	2.383989	1.085846	2.20	0.0284
aps_gr*admit_*citize	Returning International	1	1.309095	3.897248	0.34	0.7370

Figure 13: Overall Model including interaction terms

- NBT academic literacy, Extended Programme, International (p-value = 0.0498)
- APS Grade 12, Extended Programme, International (p-value = 0.034)
- APS Grade 12, New, International (p-value = 0.0284)

These interactions are statistically significant because the p-values are less than the level of significance. Hence, the null hypothesis is rejected in favour of the alternate hypothesis. It can furthermore, be concluded that the rest of the interaction terms are not statistically significant.

3.4.2 Regression of numeric data

A classical linear regression analysis was performed on the data under the assumption of no streams. Thus, if streams were to be ignored the analysis that is evaluated is to try and answer the question of which variable can be the used as the best predictor of student success at tertiary institutions, specifically UP. Similar analysis as done previously will now be evaluated. With regards to testing whether or not the overall model is significant, it is computed using the F-test (F^*) and the corresponding p-value. The null and alternate hypotheses for the overall model is as follows:

Null hypothesis, H_0: the overall model is not statistically significant

Alternate hypothesis, H_a: the overall model is statistically significant

The regression model's F-value (F^*) = 33.58 and has a corresponding p-value < 0.0001 (See Figure 14). It can thus, be concluded that since the (p-value < 0.0001) $<$ ($\alpha = 0.05$), the null hypothesis is rejected at the 5 percent level of significance. This is indicative of the overall model being statistically significant. Furthermore, the R^2= 0.1667, which means that 16.7 percent of the variation in the dependent variable, GPA scores, is explained by the model (i.e. the independent variables). The independent variables in this regression model include, NBT academic literacy, NBT quantitative literacy, NBT mathematics and then the Grade 11 and Grade 12 APS scores.

For simplicity the various parameters will be defined:

- β_0 is the intercept term in simple regression analysis
- β_1 corresponds to be a representation of NBT academic literacy (X_1)
- β_2 corresponds to be a representation of NBT quantitative literacy (X_2)
- β_3 corresponds to be a representation of NBT mathematics (X_3)
- β_4 corresponds to be a representation of Grade 11 APS (X_4)
- β_5 corresponds to be a representation of Grade 12 APS (X_5)

Therefore, the Population Regression Function (PRF) using the parameters as given in the model using the parameters defined above (not in general Y and X formation) is:

$$\begin{aligned} GPA &= \beta_0 + \beta_1 NBTacademicliteracy + \beta_2 NBTquantitativeliteracy + \beta_3 NBTmathematics \\ &\quad + \beta_4 APSgrade11 + \beta_5 APSgrade12 + \mu_i \end{aligned}$$

where μ_i is the residuals in the model.

The Sample Regression Function (SRF) is then given by:

$$\begin{aligned} \hat{GPA} &= \hat{\beta}_0 + \hat{\beta}_1 NBTacademicliteracy + \hat{\beta}_2 NBTquantitativeliteracy + \hat{\beta}_3 NBTmathematics \\ &\quad + \hat{\beta}_4 APSgrade11 + \hat{\beta}_5 APSgrade12 \end{aligned}$$

The estimated regression model for the complete model can be written as follows (see Figure 14):

$$\begin{aligned} \hat{GPA} &= 5.50029 + 0.09526NBTacademicliteracy - 0.18966NBTquantitativeliteracy \\ &\quad +0.13285NBTmathematics - 0.04845APSgrade11 + 1.52781APSgrade12 \end{aligned}$$

A simple interpretation for NBT mathematics is: for a unit increase in GPA score, there will be a 0.13285 unit increase in NBT mathematics, provided that the other variables are in the model and remain constant. Similarly, the interpretation of APS Grade 12 is: for a unit increase in the GPA score, Grade 12 APS will increase by 1.52781 units, given the other variables are in the model and remain constant. This could also be stated from the side of the independent variables namely: provided all the variables in the model remain constant an increase of one unit in the NBT mathematics will result in an increase of 0.13285 units in the GPA score etc. The interpretation for the other variables will be the same with just the parameter estimate changing (see Figure 14).

To test whether the individual variables are statistically significant or not a t-test is performed. The corresponding p-value will help reject or not reject the null hypothesis. Therefore, in general, the null and alternate hypothesis can be defined as:

H_0: $\beta = 0$

H_a: $\beta \neq 0$

Hence, for the NBT academic literacy variable the null and alternate hypothesis is:

H_0: $\beta_1 = 0$

H_a: $\beta_1 \neq 0$

From Figure 14, the p-value for $\hat{\beta}_1 = 0.1371$, this p-value is greater than the 5 percent level of significance. Therefore, since this is true, the null hypothesis is not rejected. Hence, the academic literacy parameter is not significant and can be left out of the model.

Similarly, the null and alternate hypothesis for the Grade 11 APS is as follows:

H_0: $\beta_4 = 0$

H_a: $\beta_4 \neq 0$

With reference to Figure 14, the observed p-value for $\hat{\beta}_4 = 0.7903$, where the p-value is greater than $\alpha = 0.05$. Thus, the null hypothesis cannot be rejected at the 5 percent level of significance. Therefore, $\hat{\beta}_4$ (Grade 11 APS) is not a statistically significant parameter and can be left out of the model.

The same approach can be used for all the parameters. A summary of the variables would be that the aforementioned two parameters are not statistically significant and can be left out of the model. On the other hand, the rest of the variables are significant. This is because the respective p-values are less than the 5 percent level of significance. Hence, the null hypothesis is rejected in favour of the alternate hypothesis.

The smallest p-value (< 0.0001) corresponds to the Grade 12 APS.

Further regression analysis was performed on the separate streams, in order to determine which parameter could potentially lead to higher success rates. This could help the Faculty of EMS make an informed decision based on which variable (Grade 11 APS, Grade 12 APS or the individual NBT groups) could produce more successful students within the various streams. It was however, not possible to perform regression models on two of the five streams. The reasons are that due to missing data, the Readmitted stream was excluded (as mentioned earlier in this section). Linear combinations were picked up with the Transferring stream. Linear combinations means that in matrix form, mathematical operations (for example, addition, multiplication etc.) performed on one column will lead to the same values in another column. A numerical example in matrix form follows:

$$\begin{bmatrix} col(A) & col(B) & col(C) \\ 1 & 3 & 2 \\ 2 & 1 & 4 \\ 5 & 7 & 10 \end{bmatrix}$$

From the above matrix, column A and column C are linear combinations of one another. This is because if every element in column A is multiplied by a constant equal to two then the resulting column would correspond to the elements in column C, i.e. 1 x 2 = 2; 2 x 2 = 4 and 5 x 2 = 10.

The SAS System

The REG Procedure
Model: MODEL1
Dependent Variable: gpa gpa

Number of Observations Read	845
Number of Observations Used	845

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	44962	8992.31140	33.58	<.0001
Error	839	224677	267.79145		
Corrected Total	844	269639			

Root MSE	16.36433	R-Square	0.1667
Dependent Mean	56.92631	Adj R-Sq	0.1618
Coeff Var	28.74652		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	5.50029	5.15090	1.07	0.2859
nbt_academic_literacy	nbt_academic_literacy	1	0.09526	0.06402	1.49	0.1371
nbt_quantitative_literacy	nbt_quantitative_literacy	1	-0.18966	0.05536	-3.43	0.0006
nbt_maths	nbt_maths	1	0.13285	0.04644	2.86	0.0043
aps_grade_11	aps_grade_11	1	-0.04845	0.18217	-0.27	0.7903
aps_grade_12	aps_grade_12	1	1.52781	0.20097	7.60	<.0001

Figure 14: Regression of the Numerical variables

The above holds true for students in the Transferring stream. The linear combinations are specific to NBT mathematics, Grade 11 APS and Grade 12 APS. Certain estimates are also considered to be biased. The output is not given in the report but the coding in the Appendix can be used to simulate the results.

Regression analysis performed on the New stream resulted in the following (see Figure 15). Firstly, the R^2 (goodness of fit) value equals 0.1844, meaning that 18.44 percent of the variation in the dependent variable (GPA score) is explained by the model (i.e. the independent variables).

To assess the overall significance of the model the following hypotheses need to be tested:

H_0: the overall model is not statistically significant

H_a: the overall model is statistically significant

The respective p-value of the F-test statistic (F = 24.37) is < 0.0001. Concluding, the null hypothesis is rejected at the 5 percent level of significance, meaning that the overall model is highly statistically significant. Furthermore, the evaluation of the parameters yields the following results:

- The variables, namely, Grade 11 APS, NBT academic literacy and NBT quantitative literacy are not statistically significant and it would be of no difference to the model if they were left out
- APS Grade 12 and NBT mathematics are parameters that are statistically significant

The above was tested using the following hypotheses (in general):

H_0: $\beta = 0$ (β, any parameter is equal to zero)

H_a: $\beta \neq 0$

Hence, testing whether or not NBT quantitative literacy ($\hat{\beta}_2$) is statistically significant or not, the hypotheses are:

H_0: $\beta_2 = 0$

H_a: $\beta_2 \neq 0$

Evaluating the appropriate parameter (i.e. $\hat{\beta}_2$), the t-test value equals -0.79, with a corresponding p-value = 0.4326. The t-test statistic is computed using the following formula:

$$t^* = \frac{\hat{\beta}_2 - \beta_2}{se\left(\hat{\beta}_2\right)}$$

Regression analysis for the New stream 1

The REG Procedure
Model: MODEL1
Dependent Variable: gpa gpa

Number of Observations Read 545
Number of Observations Used 545

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	24847	4969.43613	24.37	<.0001
Error	539	109916	203.92513		
Corrected Total	544	134763			

Root MSE	14.28024	R-Square	0.1844
Dependent Mean	60.77749	Adj R-Sq	0.1768
Coeff Var	23.49593		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.86032	6.67803	0.13	0.8975
nbt_academic_literacy	nbt_academic_literacy	1	0.08087	0.06765	1.20	0.2324
nbt_quantitative_literacy	nbt_quantitative_literacy	1	-0.04950	0.06302	-0.79	0.4326
nbt_maths	nbt_maths	1	0.09765	0.04952	1.97	0.0491
aps_grade_11	aps_grade_11	1	-0.02561	0.19874	-0.13	0.8975
aps_grade_12	aps_grade_12	1	1.53288	0.23928	6.41	<.0001

Figure 15: Regression Analysis for the New stream

where se refers to the standard error. For $\hat{\beta}_2$ using the formula and the relevant output the t-test statistic is:

$$t^* = \frac{-0.0495 - 0}{0.06302} = -0.79$$

Similarly, the t-test statistic can be computed in this fashion for all the parameters.

Evaluating $\hat{\beta}_2$, since the (p-value $= 0.4326$) $\not<(\alpha = 0.05)$, the null hypothesis is not rejected at the 5 percent level of significance. Hence, it can be concluded that NBT quantitative literacy is not a statistically significant parameter and can be left out of the model.

In the same way, the parameter (Grade 12 APS) can be tested. The corresponding null and alternate hypotheses are as follows:

H_0: $\beta_5 = 0$

H_a: $\beta_5 \neq 0$

From Figure 15, one can see that the t-test statistic $(t^*) = 6.41$. Due to a relatively large t^* and small standard error, the expectation is that the p-value will be small. Upon evaluation the p-value for $\hat{\beta}_5$ is less than 0.0001. Hence, because the p-value is less than the 5 percent level of significance, the null hypothesis is rejected in favour of the alternate hypothesis. Meaning that the Grade 12 APS is statistically significant and therefore, plays a role in the model.

Evaluation of the regression analysis in determining which parameter is more accurate in shaping GPA scores at tertiary level for students enrolled for the Extended Programme will now take place.

An overview of the model is provided in Figure 16. It shows that the F-statistic value of the model is equal to 2.3, this is a relatively small number that is obtained. The corresponding p-value is equal to 0.0539. If the test is evaluated against a 5 percent level of significance (i.e $\alpha = 0.05$) this would imply that the overall model is in actual fact not statistically significant. Furthermore, the model has a R^2 value equal to 0.143, which can be interpreted as 14.3 percent of the variation in GPA scores is explained by the model (the independent variables).

In the interest of the aim of this section, to determine which parameter may lead to enhanced performance at UP the independent parameters (the three individual NBT components as well as the two APS scores) will be assessed. Two of the six parameters will be investigated, one showing a parameter that is insignificant and one where the variable is statistically significant and is thus, crucial to the importance of the model. It should

Regression Analysis for the Extended Programme stream

The REG Procedure
Model: MODEL1
Dependent Variable: gpa gpa

Number of Observations Read	75
Number of Observations Used	75

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	6316.37677	1263.27535	2.30	0.0539
Error	69	37857	548.64884		
Corrected Total	74	44173			

Root MSE	23.42325	R-Square	0.1430
Dependent Mean	46.75940	Adj R-Sq	0.0809
Coeff Var	50.09315		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	-14.97781	40.67727	-0.37	0.7138
nbt_academic_literacy	nbt_academic_literacy	1	-0.18693	0.31874	-0.59	0.5595
nbt_quantitative_literacy	nbt_quantitative_literacy	1	-0.25403	0.28386	-0.89	0.3740
nbt_maths	nbt_maths	1	0.05660	0.29653	0.19	0.8492
aps_grade_11	aps_grade_11	1	-0.14734	0.93780	-0.16	0.8756
aps_grade_12	aps_grade_12	1	3.08659	1.28925	2.39	0.0194

Figure 16: Regression Analysis for the Extended Programme stream

be noted that the same setting up of hypotheses and conclusions can be made for the remaining parameters. Firstly, if NBT mathematics is used as an example, upon first inspection, intuition would suggest that the parameter is statistically insignificant due to the high p-value that is observed. Hence, the first step is to set up the null and alternate hypotheses. These are given below:

H_0: $\beta_3 = 0$

H_a: $\beta_3 \neq 0$

The p-value for $\hat{\beta}_3$ equals 0.8492, leads to the null hypothesis being rejected at the 5 percent level of significance. Concluding, that the parameter (NBT mathematics) is indeed not statistically significant and can be left out of the model. Other parameters that are statistically insignificant (same procedure applied) include, NBT academic literacy, NBT quantitative literacy as well as Grade 11 APS scores. The conclusion can be made due to all these parameters experiencing high p-values upon visual review.

Conversely, a parameter that is statistically significant is the Grade 12 APS scores. The parameter is symbolised by $\hat{\beta}_5$. The null and alternate hypotheses are given by:

H_0: $\beta_5 = 0$

H_a: $\beta_5 \neq 0$

Hence, the respective t-test value = 2.39 with a corresponding p-value = 0.0194. Since (p-value = 0.0194) $<$ ($\alpha = 0.05$), the null hypothesis is rejected at the 5 percent level of significance. This results in Grade 12 APS being statistically significant.

Finally, the last stream where classical linear regression analysis was performed was on the Returning students. The estimated regression model (SRF) can be written as (see Figure 17):

$$\begin{aligned} \hat{GPA} &= 38.26175 + 0.15531 NBTacademicliteracy - 0.2665 NBTquantitativeliteracy \\ &\quad + 0.15401 NBTmathematics - 0.04505 APSgrade11 + 0.40795 APSgrade12 \end{aligned}$$

Furthermore, if NBT mathematics is to be interpreted, it means that for a unit increase in GPA, NBT mathematics will increase by 0.15401 units where all the other parameters are in the model but also remain constant. Another example is for a unit increase in the GPA scores, Grade 11 APS will decline by 0.04505 units whilst the other parameters are already in the model and held constant.

The null and alternate hypotheses for the significance for the overall model is as follows:

Regression Analysis for the Returning stream

The REG Procedure
Model: MODEL1
Dependent Variable: gpa gpa

Number of Observations Read	222
Number of Observations Used	222

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	2491.99524	498.39905	1.69	0.1392
Error	216	63861	295.65352		
Corrected Total	221	66353			

Root MSE	17.19458	R-Square	0.0376
Dependent Mean	50.81214	Adj R-Sq	0.0153
Coeff Var	33.83951		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	38.26175	10.26597	3.73	0.0002
nbt_academic_literacy	nbt_academic_literacy	1	0.15531	0.14307	1.09	0.2789
nbt_quantitative_literacy	nbt_quantitative_literacy	1	-0.26650	0.10956	-2.43	0.0158
nbt_maths	nbt_maths	1	0.15401	0.11157	1.38	0.1689
aps_grade_11	aps_grade_11	1	-0.04505	0.36467	-0.12	0.9018
aps_grade_12	aps_grade_12	1	0.40795	0.42140	0.97	0.3341

Figure 17: Regression Analysis for the Returning Stream

H_0: the overall model is not statistically significant

H_a: the overall model is statistically significant

Thus, from Figure 17, it is shown that the p-value for the F-test statistic is equal to 0.1392. This means that since the (p-value = 0.1392) $\not<(\alpha = 0.05)$, the null hypothesis is not rejected at the 5 percent level of significance. It can then be concluded that the overall model is highly insignificant. Further analysis, excluding the intercept term, NBT academic literacy (p-value = 0.2789), NBT mathematics (p-value =0.1689), Grade 11 APS (p-value = 0.9018) and Grade 12 APS (p-value = 0.3341) are all in excess of the 5 percent level of significance. Hence, all these parameters are statistically insignificant. The only parameter that is statistically significant and cannot be removed from the model is that of NBT quantitative literacy. The null and alternate hypotheses for β_2 are:

H_0: $\beta_2 = 0$

H_a: $\beta_2 \neq 0$

The appropriate p-value is equal to 0.0158, with the value being less than $\alpha = 0.05$ (5 percent level of significance), the null hypothesis is rejected in favour of the alternate hypothesis. Thus, NBT quantitative literacy is statistically significant.

3.4.3 South African students vs International students

Further analysis was conducted based on the results in section 3.4.1 with regards to the performance of South African students versus their International counterparts. The same procedure that was done in section 3.2 was applied. The model included the citizenship country description (discussed in this subsection) and the gender description (discussed in subsection 3.4.4).

The null and alternate hypothesis can be described as follows:

$H_0 : \mu_{sa} - \mu_{int} = 0$ (the mean GPA scores of South African students is the same as the mean GPA scores of International students)

$H_a : \mu_{sa} - \mu_{int} \neq 0$

Note: SA makes reference to South African students and int refers to International students.

Figure 18 reveals that the confidence interval using the Tukey method is (-4.6814 ; 5.5876). The mean value equal to zero falls within the confidence interval which means that the between group comparison is insignificant as the null hypothesis is rejected at the 5 percent level of significance. Another measure to show

that the comparison is insignificant is due to a positive difference between means value. Hence, South African students do not necessarily perform better than international students.

Furthermore, the Scheffe grouping leads to the same conclusion as the variables are not significant at the 5 percent level of significance since they are grouped using the same alphabetical letter.

3.4.4 Gender: female vs male students

When gender of the students in the study were evaluated the following became apparent. The null and alternate hypothesis are:

$H_0 : \mu_{female} - \mu_{male} = 0$ (the mean GPA scores of female students is the same as the mean GPA scores of male students)

$H_a : \mu_{female} - \mu_{male} \neq 0$

With reference to Figure 19, the confidence interval is (8.2024 ; 13.6022) with a difference between means equal to 10.9023. Therefore, four main conclusions can be made:

1. The mean value of zero does not fall within the interval
2. The null hypothesis is rejected at the 5 percent level of significance
3. The between group comparison is statistically significant
4. Female students perform better than male students when GPA score is used as the dependent variable.

The above refers to the Tukey Studentised Range Test. The Scheffe grouping reveals that the two variables (male and female) are assigned different alphabetic letters. This is indicative of the variables being statistically significant at the 5 percent level of significance.

4 Results

A brief summary of the results as seen in the application will follow in this section.

Taking averages into account (with caution of course) it seems as if the better First-year performance would be those students that are classified in the New students group. This is based on the comparison of the GPA scores, which is of importance as it ties in with the main objective of this research report. The averages for GPA scores (for the first year) amongst the streams were also compared to the average GPA scores achieved in the first semester. Even though there were in some instances slight improvements (Extended Programme) and declines for the other four streams (New, Transferring, Readmitted and Returning), the performance of First-year students remained relatively constant over the first year.

When the various streams are compared in a between group manner, the following becomes apparent. Firstly, the performance of New students is better than the performance of both the Returning stream as well as the Extended Programme stream at a First-year level with the variable of interest being the First-year GPA scores. These results were achieved with performing difference between means with the option of examining both the Tukey and Scheffe methods. Furthermore, when the Tukey method was applied, it revealed that Transferring students performed better than students enrolled in the Faculty of EMS's four-year Extended Programme. SAS® software output (see Figures 7 or 6) makes use of *** for a visualisation of the streams where between group comparisons are statistically significant at the 5 percent level of significance. In addition, both the Tukey and Scheffe methods provide a grouping of the means with alphabetic letters. From Figure 7, all the admittance types are grouped using the letter 'A', this suggests that the means are not statistically different from one another.

The Kruskal-Wallis test identified that students who matriculated under the different matric authorities (Cambridge, IEB, State School and Foreign Country) performed differently from one another in terms of GPA scores in the First-year. This was due to the fact that the null hypothesis was rejected in favour of the alternate hypothesis as the p-value when the test was performed was less than the level of significance which was set at 5 percent (i.e. $\alpha = 0.05$). Rankings were also determined in terms of Wilcoxon scores, more specifically the ranking of mean scores. Cambridge was ranked first, followed by IEB, State School was ranked in third place and the final ranking was that of Foreign Country. Further investigation revealed that students

The GLM Procedure

Tukey's Studentized Range (HSD) Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	843
Error Mean Square	319.8454
Critical Value of Studentized Range	2.77579

Comparisons significant at the 0.05 level are indicated by ***.

citizenship_country_description Comparison	Difference Between Means	Simultaneous 95% Confidence Limits	
South Africa - International	0.4531	-4.8694	5.7756
International - South Africa	-0.4531	-5.7756	4.8694

The SAS System

The GLM Procedure

Scheffe's Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	843
Error Mean Square	319.8454
Critical Value of F	3.85251
Minimum Significant Difference	5.3225
Harmonic Mean of Cell Sizes	86.99172

NOTE: Cell sizes are not equal.

Means with the same letter are not significantly different.

Scheffe Grouping	Mean	N	citizenship_country_description
A	56.951	799	South Africa
A			
A	56.498	46	International

Figure 18: Country Citizenship Compared

The GLM Procedure

Tukey's Studentized Range (HSD) Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	843
Error Mean Square	297.6763
Critical Value of Studentized Range	2.77579

Comparisons significant at the 0.05 level are indicated by ***.

gender_description Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
Female - Male	10.9023	8.2022	13.6023	***
Male - Female	-10.9023	-13.6023	-8.2022	***

The GLM Procedure

Scheffe's Test for gpa

NOTE: This test controls the Type I experimentwise error rate.

Alpha	0.05
Error Degrees of Freedom	843
Error Mean Square	297.6763
Critical Value of F	3.85251
Minimum Significant Difference	2.7
Harmonic Mean of Cell Sizes	314.613

NOTE: Cell sizes are not equal.

Means with the same letter are not significantly different.

Scheffe Grouping	Mean	N	gender_ description
A	59.623	636	Female
B	48.721	209	Male

Figure 19: Gender Compared

who attended Cambridge had a significant better First-year performance than students that attended schools in Foreign Countries. The variable of interest was once again the GPA score. Hence, Cambridge educated students have a more accurate measure of performance at the tertiary level at UP (this is based solely on mean scores). The visualisation interpretation of this is given by the different alphabetic letters in Figure 11. Same letter descriptions like only 'A' or 'B' implies that the comparisons are not statistically significantly different from one another but different letter descriptions like 'A' or 'B' implies that the comparisons are statistically significantly different from one another. In other words, matric authority descriptions (Cambridge, IEB, State School and Foreign Country) that have the same grouping letter are not statistically significant from one another. As was seen in subsection 3.3.4, Cambridge, IEB and State Schools are not significantly different from each other. Likewise, IEB, State School and Foreign Country are not significantly different from one another. Hence, the only lettering that is different is that observed between Cambridge and Foreign Country.

The van der Waerden normal scores were also tested. The conclusion is that the four independent levels (Cambridge, IEB, State School and Foreign Country) encompassed under the matric authority descriptions is that at least one of the population distribution functions yield higher observation values (GPA scores) than at least one of the three remaining distributions. Hence, the population distribution for Cambridge possibly yields higher GPA scores than the population distribution function of Foreign Country. It has also been shown that these population distribution functions follow different distributions.

Classical linear regression was performed to determine which parameter (NBT academic literacy, NBT quantitative literacy, NBT mathematics, Grade 11 APS or Grade 12 APS) would be the best predictor of student performance at UP, especially at a First-year level. The first regression that was evaluated was if student streams were ignored and students were pooled into a single category, then the three parameters that were significant included NBT mathematics, NBT quantitative literacy and Grade 12 APS. The best predictor however, would be Grade 12 APS as this parameter had the lowest p-value (< 0.0001). Hence, if a decision has to be made as to what the best way is in accepting students into the Faculty of EMS it would be on the performance of Grade 12 marks that are converted into an APS score. Furthermore, when the analysis was done on the separate streams, namely, New students, students enrolled for the Extended Programme and students that are Returning to the Faculty of EMS at UP the following was observed: New students' most accurate predictor of First-year performance is Grade 12 APS scores (p-value < 0.0001). Grade 12 APS scores and NBT quantitative literacy are the best predictors for the Extended Programme and Returning streams, respectively. Hence, if an overall consensus is made the best predictor and most accurate reflection of a student's First-year performance is then likely to be the Grade 12 APS.

Comparisons were also done on whether there is a difference in the performance of South African students versus International students however, no significant differences were found. Furthermore, analysis revealed that female students perform better than male students at a First-year level when analysing GPA scores.

5 Conclusion

In concluding, in order to improve the throughput rates with students graduating in the specified time period (usually three years at the undergraduate level) from the Faculty of EMS at UP the following evidence should be taken into account. Firstly, Grade 11 APS scores in any stream do not play a vital role in determining potential success among students, the analysis showed that this parameter is highly insignificant in the majority of the models. Hence, other possible ways to provisionally accept students should be considered. A recommendation could be having an entrance examination that is set up by UP staff. This will provide a more accurate reflection as to how students might perform in their First-year as well as subsequent years of study. As in many cases, irrespective of the enrolment streams and when streams were considered, Grade 12 APS still provides the ultimate decision on acceptance of students as this parameter proved to be highly significant. Currently, at UP the final decision is based on a student's matric results that are converted into an APS score. Concerning is the writing of the NBT tests, in all three categories, with the exception of NBT quantitative literacy (Returning stream) the parameter is insignificant and does not impact on the model at all. Hence, these parameters could even in essence be left out of the model. Thus, evaluating averages should be done so with caution as when the averages were calculated, it suggested that the NBT marks are the best predictor to estimate a students performance at tertiary level. Further analysis then proved that this was in

actual fact not the case and could lead to an under-or-over-estimation of student performance. What was evident is that students performed relatively identically between the two semesters when the averages for the GPA scores were computed, with the only slight increase between the two semesters being observed in the Extended Programme stream. The changes were however, minimal irrespective of whether it was an increase or a decrease.

Between group comparisons were performed and when analysed the results were not clear cut as to say a certain stream is head and shoulders above the rest. There was an indication that New students have a slight advantage in First-year performance as they did perform better than both the Returning and Extended Programme students. Furthermore, Transferring students also had higher GPA scores than those students enrolled for the Extended Programme. This is by no means an indication that the weakest stream is the Extended Programme. For a conclusive answer to be made regarding this further tests will have to be performed such as how these students then adjust when they continue with their studies in a more mainstream environment. Further investigation and recommendations will be to assess their Second-year GPA performance as well as seeing what the throughput rate is for this specific stream. Subsequently, this can be evaluated across all the streams. Hence, the Faculty of EMS can perhaps reserve a few more places for New students than for the other streams.

One of the major shortfalls of this research report is that the models did not fit the data well with low R^2 values. This questions the reliability of the models. Another shortfall that was experienced when trying to compute a more complete model was that not all the information for the variables was available for every student. Hence, less than 50 percent of the observations could be used to make predictions to help improve the enrolment strategies within the Faculty. This might have also been a factor contributing to the poor model fit. Even though the fit of models should be treated with caution, this still remains a concern.

Furthermore, this research report evaluated the matric type authority and found that Cambridge schooled students do perform slightly better than students that had other schooling backgrounds. It also came to light that these independent variables (Cambridge, IEB, State School and Foreign Country) come from different population distributions. It might be of worth to investigate just what population distributions these variables follow in order to do future research on these populations. In fact it might be of significance to test what distribution various variables (for instance, Grade 12 APS, Grade 11 APS, the NBT categories etc) follow, if the appropriate distribution can be identified it might have a significant bearing on the analysis technique applied to the data. The Faulty of EMS however, cannot make a decision on whether or not to accept students based on this as this would not be in line with social responsibilities. Moreover, the same could be said about basing a decision of acceptance on gender. Even though it was clear from the analysis that female students yield higher GPA scores than male students. Similarly, it should be noted that there was no significance in the performance of South African students versus International students.

Future progress of this study will make allowance to determine the exact throughput rate of students. Hence, a better prediction will be readily available to see exactly what the graduation rate, within the specified time period, is observed within the Faculty of EMS. The procedure will be applied on a much smaller scale, but this would give a representative sample of the population (all students enrolled in the Faculty). This however, will only be able to come to fruition in 2016, as this is when the current group of students in this study will be in their final year, provided that all modules (including third year modules) are passed.

References

[1] Salim Akoojee and M Nkomo. Access and quality in South African higher education: the twin challenges of transformation. *South African Journal of Higher Education*, 21(3):385–399, 2007.

[2] S Badat, M Price, and Mabelebele. HESA presentation to the portfolio committee on higher education and training. Technical report, Higher Education South Africa, 2014.

[3] Saleem Badat. Higher education, transformation and lifelong learning. Technical report, University of the Western Cape, South Africa, 2013.

[4] Wayne J Camara and Gary Echternacht. The sat and high school grades: Utility in predicting success in college. research notes. Technical report, College Entrance Examination Board, New York, NY., 2000.

[5] Brian Eagon Forbes. Assessment strategies for work-integrated learning at higher education institutions. *SAQA Bulletin*, 6(2):49–66, 2004.

[6] Damodar N. Gujarati and Dawn C. Porter. *Basic Econometrics*. McGraw-Hill / Irwin, fifth edition, 2009.

[7] Michael H. Kutner, Christopher J. Nachtsheim, and John Neter. *Applied Linear Regression Models*. McGraw-Hill / Irwin, 2004.

[8] NJ Le Roux, A Bothma, and HL Botha. Statistical properties of indicators of first-year performance at university. *ORiON*, 20(2):161–178, 2004.

[9] Moeketsi Letseka and Simeon Maile. *High university drop-out rates: A threat to South Africa's future*. Human Sciences Research Council Pretoria, 2008.

[10] Karen MacGregor. South Africa: Shocking results from university tests. *University World News*, 16, 2009.

[11] Irwin Miller and Marylees Miller. *John E. Freund's Mathematical Statistics with Applications*. Pearson Prentice Hall, seventh edition, 2004.

[12] C Nel and L Kistner. The national senior certificate: Implications for access to higher education. *South African Journal of Higher Education*, 23(5):953–973, 2009.

[13] E Pretorius. Improving the enrolment strategy in the faculty of economic and management sciences through an inquiry into the throughput rates of diverse enrolment and transfer streams. Honours Research Report, 2014.

[14] Alan Seidman. *College student retention: Formula for student success*. Greenwood Publishing Group, 2005.

[15] James Taylor, Rui Brites, Fernanda Correia, Minoo Farhangmehr, Brites Ferreira, Maria de Lourdes Machado, Cláudia Sarrico, and Maria José Sá. Strategic enrolment management. *Higher Education Management and Policy*, 20(1):1–17, 2008.

[16] Mantz Yorke and Bernard Longden. *Retention and student success in higher education*. McGraw-Hill International, 2004.

6 Appendix

The appendix contains SAS® software code.[2]

SAS® software code for Figures 3-7

```
options nodate nodate pageno = 1;

****Importing Admittance Type into SAS***;
proc import out = admit
        datafile = 'C:\Users\Claudia\Desktop\Varsity Modules-2015\
                                STK 795\Data\Admit_type.xls'
         dbms = excel replace;
        getnames = yes;
        sheet = 'Admit';
run;

title1 'Comparing the Admittance type of first year students';
title2 'Analysis of Variance Table using the glm ';
****Performing the GLM procedure****;

proc glm data = admit; ***glm uses ordinary least squares in order to
                                                fit general linear models***;
        class admittance_type;
        model gpa = admittance_type;
        means admittance_type /scheffe lines cldiff tukey;
***Both the Scheffe and Tukey comparisons of
        mean differences will be applied***;
run;
```

SAS® software code for Figures 8-11

```
options nodate ps = 10000;

****Importing data****;
proc import out = school
        datafile = 'C:\Users\Claudia\Desktop\Varsity Modules-2015\
                                STK 795\Data\Matric_Authority_Description.xls'
         dbms = excel replace;
        getnames = yes;
        sheet = 'Matric';
run;
title1 'Nonparametric test for Matric Authority';

ods graphics on;
proc npar1way data = school plots(only)=wilcoxonboxplot;
        class matric_authority_description;
        var gpa;
run;
ods graphics off;

proc glm data = school;
        class matric_authority_description;
```

[2]The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA

```
        model gpa = matric_authority_description;
        means matric_authority_description / scheffe
                                                    lines tukey cldiff;
run;
```

SAS® software code for Figures 12-13

```
options pageno = 1 nodate;

data regression;
set reg;
****Computing the ANOVA section of the model****;
proc glm data = regression;
class admit_type gender_description citizenship_country_description;
model gpa = admit_type gender_description citizenship_country_description
nbt_academic_literacy nbt_quantitative_literacy nbt_maths
aps_grade_11 aps_grade_12
admit_type*gender_description*citizenship_country_description
admit_type*gender_description*nbt_academic_literacy
admit_type*gender_description*nbt_quantitative_literacy
admit_type*gender_description*nbt_maths
admit_type*gender_description*aps_grade_11
admit_type*gender_description*aps_grade_12
admit_type*citizenship_country_description*nbt_academic_literacy
admit_type*citizenship_country_description*nbt_quantitative_literacy
admit_type*citizenship_country_description*nbt_maths
admit_type*citizenship_country_description*aps_grade_11
admit_type*citizenship_country_description*aps_grade_12;
run;

***Using dummy variables to evaluate significance of the parameters
and to see which is the best estimate of success***;
ods graphics on;
proc glmselect data = regression;
        ods select parameterestimates;
class admit_type gender_description citizenship_country_description
        / param = ref;
**most of the possible **;
model gpa = admit_type gender_description citizenship_country_description
nbt_academic_literacy nbt_quantitative_literacy
nbt_maths aps_grade_11 aps_grade_12
admit_type*gender_description*citizenship_country_description
admit_type*gender_description*nbt_academic_literacy
admit_type*gender_description*nbt_quantitative_literacy
admit_type*gender_description*nbt_maths
admit_type*gender_description*aps_grade_11
admit_type*gender_description*aps_grade_12
admit_type*citizenship_country_description*nbt_academic_literacy
admit_type*citizenship_country_description*nbt_quantitative_literacy
admit_type*citizenship_country_description*nbt_maths
admit_type*citizenship_country_description*aps_grade_11
admit_type*citizenship_country_description*aps_grade_12
/ selection = none;
run;
```

```
ods graphics off;
```

SAS® software code for Figures 14-17. Note: the final set of SAS® software coding within this subsection does not have a corresponding Figure.

```
data regression;
set reg;

proc reg data = regression;
        model gpa = nbt_academic_literacy nbt_quantitative_literacy nbt_maths
                                aps_grade_11 aps_grade_12;
run;


  options nodate pageno = 1 ls = 115;

****New****;
proc import out = new
        datafile = 'C:\Users\Claudia\Desktop\Varsity Modules-2015\STK 795\
                                Data\regression.xls'
         dbms = excel replace;
        getnames = yes;
        sheet = 'New';
run;
title 'Regression analysis for the New stream';

proc reg data = new;
        model gpa = nbt_academic_literacy nbt_quantitative_literacy nbt_maths
                                aps_grade_11 aps_grade_12;
run;

****Extended Programme****;

option nodate ps = 10000 pageno = 1 ls = 136;

data extended_programme;
set reg;

title 'Regression Analysis for the Extended Programme stream';

proc reg data = extended_programme;
        model gpa = nbt_academic_literacy nbt_quantitative_literacy nbt_maths
                                aps_grade_11 aps_grade_12;
run;

options nodate ps = 10000 pageno = 1 ls = 136;

data returning;
set reg;

title 'Regression Analysis for the Returning stream';
proc reg data = returning;
        model gpa = nbt_academic_literacy nbt_quantitative_literacy nbt_maths
                                aps_grade_11 aps_grade_12;
```

```
run;

****Transferring****;
/*proc import out = transferring;
        datafile = 'C:\Users\Claudia\Desktop\Varsity Modules-2015\STK 795\
                                Data\reg_transferring.xls'
         dbms = excel replace;
        getnames = yes;
        sheet = 'Transferring';
run;
*/
data transferring;
set reg;

proc reg data = transferring;
        model gpa = nbt_academic_literacy nbt_quantitative_literacy nbt_maths
                                aps_grade_11 aps_grade_12;
run;
```

SAS® software code for Figure 18

```
options nodate pageno = 1;

data citizenship;
set reg;

proc glm data = citizenship;
class citizenship_country_description;
model gpa = citizenship_country_description;
means citizenship_country_description /scheffe lines tukey cldiff;
run;
```

SAS® software code for Figure 19

```
options nodate pageno = 1;

data gender;
set reg;

proc glm data = gender;
class gender_description;
model gpa = gender_description;
means gender_description /scheffe tukey cldiff lines;
run;
```

An overview of local optimum designs for nonlinear response models

Wanda Ndamse 12290085

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor(s): Dr LK Debusho

Department of Statistics, University of Pretoria

2 November 2015

Abstract

The main problem in calculating an optimum design for a nonlinear response model is that, the information matrix and so the optimality criterion are functions of unknown parameters in the model. In this essay the subject of optimum design is first covered for linear models, noting important optimality criteria that can also be used on nonlinear models. The concept of optimum designs for linear models is somewhat also applicable for nonlinear models, however, with changes in some intrinsic part of nonlinear models. Locally optimum design approach is used to calculate the designs by introducing the best guess for the unknown parameters.

Declaration

I, *Wanda Ndamse,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Wanda Ndamse

_____
Dr Legesse Kassa Debusho

_____
Date 2 November 2015

Acknowledgments

Having been grown in a family where all my grandparents were teachers, I think I was bound to at least go as far as I have gone in my academic career even though the path has not been easy. With the presence of my parents I can atleast say the path has shed some light during darker times. I thank my parents for accepting me as an absentee child during these past winter holidays, I hope that by the end of this academic year they will see the sacrifices I had to take yielding results. I give special thanks to Johannesburg Stock Exchange (JSE) for funding me for the past three years, I really would have not been where I am if it was not for their investment in my academic career. Finally, I thank my supervisor Dr. Legesse Kassa Debusho who has been my only spark of hope this year. I never thought I would understand this subject matter at such a level, even though I have only touched the tip of an iceberg in terms of optimum design in general. I have truly been in great and capable hands that guided me in a professional manner towards heading for the end goal. I thank also my LORD for laying His hand over me every time I thought it was time to give up.

Contents

List of Figures

List of Tables

1 Introduction

It is well known in general that anything pertaining to nature is subject to the theory of complex adaptive system. As times change, things tend to change with their bionomics, that is why pattern, relationship and iteration have a curvature element to suite the change in their ecological system. We can then note that very few things are linear by nature, in reality, the theory of non-linearity dominates the sphere of science.

In this essay we study nonlinear models considering optimum designs for efficient estimation of parameters. In practice, this field of research is particularly important to pharmaceutical companies who are involved in various clinical trial experiments, for example, they may do experimental trials on dose modification tests or clinical trials on patience with a certain infection. Consider for example a design space denoted by χ, with the response being to know the main cause of breast cancer. Some parameters that may be of interest in this model could be regression coefficients that associate estrogen exposure, genetics, alcohol abuse etc with the response variable. It would rather be difficult to formulate the best active drug to cure the infection if such a fundamental problem of unknown parameter estimates arises as we see in most nonlinear models.

The objective of this study therefore is to calculate optimum designs for efficient estimation of unknown parameters in nonlinear models. As the information matrix for parameters and so the optimality criteria are functions of the unknown parameters, optimum designs are calculated numerically for a given set of values of parameters, optimum designs are therefore locally optimum. In Section 2, we introduce estimation of parameters and information matrix under the nonlinear model setting. We illustrate how the information matrix is a function of the unknown parameter(s) using the maximum likelihood estimate. In Section 3, we then introduce the theory of optimum design where we consider all the aspects that qualifies a design to be optimum, noting that some conditions may not always apply to nonlinear model. An application where the local approach is applied to solve the problem of unknown parameters as well as optimum design points is considered in Section 4. Section 5, will then be conclusion where we summarize the essay, state the problem areas and note alternative methods for solving optimum designs.

2 Estimation of parameters

Before building on the concept of optimum designs for nonlinear models, we first need to explain the problem arising from estimating the parameters. By considering the maximum likelihood (ML) estimation, it will then be easy to see that parameters can easily be estimated for linear, or nonlinear transformed models, rather than for intrinsically nonlinear models.

Suppose our nonlinear regression model at each time point t observes a response y_i,

$$y_i = \eta(t_i, \theta) + \varepsilon_i \tag{1}$$

where η is a nonlinear function, θ is a $(p \times 1)$ vector of unknown parameters, and ε is a random error such that $\varepsilon \sim N(0, \sigma^2)$.

Since we are interested in estimating the parameter θ, we consider the ML estimation when considering optimum designs for nonlinear models, this is mainly because of the convenience of the likelihood function that can be maximized as a log likelihood function. Consider the following expression of the ML estimation.

Let $t = (t_1, ..., t_n)^T$ be a vector of independent, identically distributed random sample with a probability density function (pdf) $f(t_i; \theta)$. The likelihood function of θ is then given by

$$L(\theta|t) = \prod_{i=1}^{n} f(t_i|\theta). \tag{2}$$

The log-likelihood function of expression (2) is given by,

$$log\, L(\theta|t) = \sum_{i=1}^{n} log\,(f(t_i|\theta)). \tag{3}$$

As a natural logarithm is monotonically increasing, maximizing $L(\theta)$ is equivalent to maximizing $log\, L(\theta)$.

The pdf $f(t;\theta)$ is a regular, the first derivative of the log-likelihood function results in what is known as the score vector, denoted

$$s(\theta|t) = \frac{\partial log\, L(\theta|t)}{\partial \theta}. \tag{4}$$

The second derivatives of the log-likelihood is called the Hessian matrix

$$H(\theta|t) = \frac{\partial log\, L(\theta|t)}{\partial \theta}\frac{\partial log\, L(\theta|t)}{\partial \theta^T} = \begin{vmatrix} \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1^2} & \cdots & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1 \partial \theta_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_p \partial \theta_1} & \cdots & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_p^2} \end{vmatrix}. \tag{5}$$

Two identities therefore are satisfied by the moments of the score vector, that is:

1. The ML estimator of θ can be found by solving the equation $s(\hat{\theta}|t) = 0$. In practice, for nonlinear models this equation is solved by numerical iteration using, for example, the Newton Raphson method[3].

2. "The information matrix for the parameters is equal to minus the expected value of the matrix of second-order derivatives of the log-likelihood function, where the derivatives are with respect to the parameters"[3]. Since the variance is the inverse of the information matrix we can therefor express $\mathrm{Var}(s(\theta|t)) = E\left\{s(\theta|t)\, s(\theta|t)^T\right\} = -E\left\{\left(\frac{\partial log\, L(\theta|t)}{\partial \theta}\frac{\partial log\, L(\theta|t)}{\partial \theta^T}\right)\right\}$.

The $p \times p$ matrix $-E\left\{\left(\frac{\partial log\, L(\theta|t)}{\partial \theta}\frac{\partial log\, L(\theta|t)}{\partial \theta^T}\right)\right\}$ is called the expected Fisher information matrix[3]. Note that the information matrix is a function of the unknown parameter(s).

3 Theory of optimum design

3.1 Optimum design

Continuous vs exact designs

It is important to note the difference between exact and continuous design and their corresponding notation since this theory lays a fundamental concept in optimum design. If we have n distinct design points in a design space, denoted χ, with a weight also called design weight, w_i, associated to observations at x_i, $i = 1, 2, \ldots, m$ then this design can be viewed as a measure ξ on the design space χ and can be described as:

$$\xi = \begin{pmatrix} x_1, x_2, ..., x_m \\ w_1, w_2, ..., w_m \end{pmatrix}. \tag{6}$$

If $w_i = r_i/n$ where r_i is a replication at x_i and $\sum_{i=1}^{m} r_i = n$, then ξ is called an exact design on χ. Thus for exact design, ξ takes on values of w_i which are multiples of $1/n$, and defines an exact design on χ. On the other hand, if we remove the restriction that w_i be a multiple of $1/n$, we can extend this idea to a design measure which satisfies $\xi(x) \geq 0$, $x \in \chi$ and $\int_\chi \xi(dx) = 1$[2]. Furthermore, expression (6) has conditions $0 \leq w_i \leq 1$ and $\sum_{i=1}^{n} w_i = 1$. Hence the measure ξ is referred to as the continuous design on χ. This concept of continuous design, pioneered by Kiefer (1959), is very popular, mainly because of the the continuity, and comfort of convexity it offers [11]. The nature of designs that will be considered furthermore are mostly continuous designs, and hence more emphases will be given to it. One should note that it is sometimes possible to obtain an exact design from an continuous design set by rounding[18], thus, it is possible to theoretically compute an continuous design and practically present an exact design.

The information matrix for the continuous design is given by[2]

$$
\begin{aligned}
M(\xi) &= \int_{\chi} f(x)\, f^T(x)\, \xi(dx) \\
&\approx \sum_{i=1}^{n} w_i\, M(\bar{\xi}_i) \\
&\approx \sum_{i=1}^{n} w_i\, f(x_i)\, f^T(x_i). \qquad (7)
\end{aligned}
$$

3.2 Experimental design

In the field of experimental design, we are generally interested and concerned with the analysis of data generated from an experiment. It would be desirable to select optimum points that best describe the experimental outcome at reliable and reproducible conclusions[19]. An optimum design would be that which minimize costs, accommodate different factors, and enable designs to be optimized when the design-space is contained[2]. Generally an experiment is formulated to answer certain questions which must be derived from an observed data. A model must then be formulated to best explain the answer from limited points which we refer to them as design points.

One of the processes of obtaining optimum design points is through observing applications that will eventually result in a minimum variance. The smaller the variance the more precise the estimate of parameter is [5]. Minimizing the variance can be achieved through several letter optimality criterion such as A-, D- and E-optimality, which is a subject of the following section.

3.3 Optimality criteria

Optimality criterion is represented by a solitary value that encapsulate the performance of a design in terms of how good it is. It is also minimized by an optimum design[10]. In programming terminology there are namely two types of criteria, however, we are only going to focus on one, which is the information-based criteria. Although there are numerous letter optimality criteria, one may desire to use A-, D- or E-optimality which can be statistically interpreted in terms of the information matrix $M(\xi)$. These optimality criteria differ in terms of how they minimized the variance.

A-optimality - minimizes the trace of the information matrix $M^{-1}(\xi)$. It can also be defined using a non-zero eigenvalue, $\lambda_1, ..., \lambda_p$ of $M(\xi)$, where the eigenvalue of $M^{-1}(\xi)$ is $\frac{1}{\lambda_1}, ..., \frac{1}{\lambda_p}$ [2]. Thus the sum of the variance is minimized through:

$$
min \sum_{i=1}^{p} \frac{1}{\lambda_i}. \qquad (8)
$$

E-optimality - is utilized when one is interested in estimating normalized linear functions of the parameter. The variance of the least well-estimated linear combination of $a^T\hat{\theta}$ with $a^Ta = 1$ is minimized[2]:

$$
min\, max \frac{1}{\lambda_i}. \qquad (9)
$$

D-optimality - is one of the commonly used criterion, mainly because of the properties it holds. It can be extended to several useful extensions such as the D_A-optimality and the D_S-optimality criterion.

Basically it maximizes the determinant of the information matrix, that is equivalent to minimizing the determinant of the variance-covariance matrix of the parameter estimates[2]. The determinant can thus be regarded as a general measure of the size of the information matrix[10]. Maximization of $M(\xi)$ is equivalent to maximizing the $log\, M(\xi)$, and the design ξ^* is said to be D-optimum if it maximizes the function $log\, M(\xi)$. This is also equivalent to minimizing the generalized variance of the parameter estimates[2], that is

$$
min \prod_{i=1}^{p} \frac{1}{\lambda_i}. \qquad (10)
$$

The D_S-optimality is appropriate when there are only a subset of parameters which are of interest, they ought to be estimated as precisely as possible with the other parameters treated as nuisance[2]. The only difficulty one may encounter with D_S-optimum designs is that $M(\xi^*)$ can be a singular.

It would also be worthwhile to mention the D-efficiency under the D-optimality criterion. D-efficiency basically compares an arbitrary design ξ with that of an optimum design ξ^*. Efficiencies are a way in which we can quantify the goodness of designs[12]. If the ratio is 100%, this means that an arbitrary design is no different to an optimum design, however, in practice this is usually not the case. D-efficiency of any arbitrary design ξ is defined as[2]

$$D_{eff} = \left\{ \frac{|M(\xi)|}{|M(\xi^*)|} \right\}^{\frac{1}{p}} \tag{11}$$

with p being the number of parameters in the model. An efficient design will have the smallest variance, thus outperform any design with a variance that is not smaller. This comparison would then put to conclusion the subject of optimum design, because after this comparison we would now know which design is an optimum design by reviewing it with the conditions in the General Equivalence Theorem if possible, otherwise review an optimum design by observing its graphical presentation.

3.4 General Equivalence Theorem

The General Equivalence Theorem lists very important conditions which are used to inquire if a design is optimum. In continuous designs we consider minimization or maximization of the general measure of the information matrix $\Psi\{M(\xi)\}$ which is under assumptions of continuity, compactness of the design space χ and convexity and differentiability of Ψ[2]. The assumption of convexity alone is of great importance as it is known that well known examples of convex functions are quadratic and exponential for t, $y \epsilon R^n$[14]. Other important properties of convexity is that local optimality guarantees global optimality, thus, for convex problems, any locally optimum point is globally optimum[14].

The General Equivalence Theorem can be observed as an effect of the evident that derivatives are zero at a minimum of a function[17], of which that function depends on the design measure ξ through the information matrix $M(\xi)$ [2]. We then let the measure $\bar{\xi}$ put mass at the point x and let the measure $\xi^{'}$ be given by

$$\xi^{'} = (1-\alpha)\xi + \alpha\bar{\xi}. \tag{12}$$

Then from equation (12), we can express

$$M(\xi^{'}) = (1-\alpha)M(\xi) + \alpha M(\bar{\xi}). \tag{13}$$

The derivative of Ψ in the direction $\bar{\xi}$ is

$$\phi(x,\xi) = \lim_{\alpha \to 0} \frac{1}{\alpha}[\Psi\{(1-\alpha)M(\xi) + M(\bar{\xi})\} - \Psi\{M(\xi)\}]. \tag{14}$$

We refer to the above equation as the directional derivative.

The conditions of the optimum design ξ^*, are then stated by the General Equivalence Theorem[17]:

1. The design ξ^* minimizes $\Psi\{M(\xi)\}$
2. The minimum of $\phi(x,\xi^*) \geq 0$
3. The derivative $\phi(x,\xi^*)$ achieves its minimum at the points of the design.

With this theorem one can now check if designs are optimum or not, noting that if one condition is met by a specific design in question, other condition ought to be satisfied as well. This theorem however does not mention anything about the number of support points in a design. A specific bound number can however be found from the nature of the information matrix $M(\xi)$, which is a symmetric $p \times p$ matrix[2]. The information matrix is additive by nature, it can be represented as a weighted sum, of at most, $\frac{p(p+1)}{2}$ information matrices $m(\bar{\xi}_i)$, where $\bar{\xi}_i$ puts unit weight at the support point x_i[2]. It is usual for optimum designs to contain fewer points. D-optimum designs contain p points, each with weight $\frac{1}{p}$[17] if the parameters are subject to

linearity, however, for nonlinear models this may not always be the case as Figure 5 and 6 will illustrate in the proceeding section.

4 Application using the local optimum approach

4.1 Model specification

Consecutive first-order reaction model. Consecutive reaction can be seen as a chain reaction in which, for example, considering the food chain in an ecosystem, the end product of what the earth produces is the initial material that herbivorous animals depend on for life, herbivorous animals will then be the initial material that carnivorous animals feed on, and so the chain goes on. Consecutive reaction is a chemical process undergo a similar reaction process as the ecosystem, however, only considering chemical reactions. One may now understand that the end product of a chemical reaction is only the initial material for another chemical reaction.

If we consider an irreversible two consecutive first order reaction, we may illustrate it as: $A \overset{\theta_1}{\to} B \overset{\theta_2}{\to} C$, where A, B and C are certain initial chemical substances. If one desires to calculate the chemical concentration of B at time t, given that the concentration of A is 1, and $\theta_1 > \theta_2 > 0$, we may present the following model:

$$\eta(t,\theta) = [B] = \frac{\theta_1}{\theta_1 - \theta_2}\left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right), \tag{15}$$

of which the ML function is:

$$\begin{aligned} L(\theta|t) &= \prod_{i=1}^{n} \frac{\theta_1}{\theta_1 - \theta_2}\left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right) \\ log\, L(\theta|t) &= n\, log\,(\frac{\theta_1}{\theta_1 - \theta_2}) + \sum log\ \left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right) \end{aligned}$$

with the score vector of θ_1 and θ_2:

$$\frac{\partial log\, L(\theta|t)}{\partial \theta_1} = \frac{n}{\theta_1} - \frac{n}{\theta_1 - \theta_2} + \sum \frac{t\exp(-\theta_1 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}, \tag{16}$$

and

$$\frac{\partial log\, L(\theta|t)}{\partial \theta_2} = \frac{n}{\theta_1 - \theta_2} - \sum \frac{t\exp(-\theta_2 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}. \tag{17}$$

From the above score vector, a 2×2 hessian matrix can now be derived through the second derivatives of the above equations,

$$\begin{aligned} \eta(\theta|t) &= \frac{\partial log\, L(\theta|t)}{\partial \theta} \frac{\partial log\, L(\theta|t)}{\partial \theta^T} \\ &= \begin{pmatrix} \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1^2} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1 \partial \theta_2} \\ \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2^2} \end{pmatrix} \end{aligned}$$

where

$$\frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1^2} = \frac{n}{(\theta_1 - \theta_2)^2} - \frac{n}{\theta_1^2} - \sum \frac{t^2}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}\left(\exp(-\theta_1 t) - \frac{\exp(-2\theta_1 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}\right),$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1 \partial \theta_2} = \frac{-n}{(\theta_1 - \theta_2)^2} + \sum \left(\frac{t^2 \exp(-\theta_2 t - \theta_1 t)}{(\exp(-\theta_2 t) - \exp(-\theta_1 t))^2}\right),$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2 \partial \theta_1} = \frac{-n}{(\theta_1 - \theta_2)^2} + \sum \left(\frac{t^2 \exp(-\theta_1 t - \theta_2 t)}{(\exp(-\theta_2 t) - \exp(-\theta_1 t))^2}\right),$$

and

$$\frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2^2} = \frac{n}{(\theta_1-\theta_2)^2} - \sum \frac{t^2}{\exp(-\theta_2 t)-\exp(-\theta_1 t)}\left(\exp(-\theta_1 t) - \frac{\exp(-2\theta_2 t)}{\exp(-\theta_2 t)-\exp(-\theta_1 t)}\right).$$

The parameter θ_1 in this model is considered a constant absorption rate, while parameter θ_2 is a constant elimination rate[16]. Depending on what kind of two consecutive first order reaction is of question, the parameter values will be exponentially distributed differently over time for every different model. It is than important to accurately estimate the parameters for the model to be as efficient as possible.

John W. Mauger[13] studies the degradation of hydro-cortisone sodium succinate which is followed in aqueous systems buffered at pH values of 6.9, 7.2 and 7.6 at 70 degrees Celsius. Figure 1 illustrates the reaction occurrence of material [A], [B] and [C] at a pH value of 6.9 at 70 degrees Celsius. Mauger described the decomposition pathway using a two-step irreversible consecutive first order reaction where a percent concentration of steroid ester is the first initial material [A], steroid alcohol is the second material [B], and products devoid of the 17-dihydroxyacetone side chain being the third material [C]. The two-step sequence is presented as $ester \xrightarrow{\theta_1} alcohol \xrightarrow{\theta_2} degradation\ products$, with their time course measured at a pH value of 6.9 at 70 degrees Celsius. The main aim of Mauger was to use reaction rates as predictive tools for evaluating the stability of therapeutically effective drugs in solution.

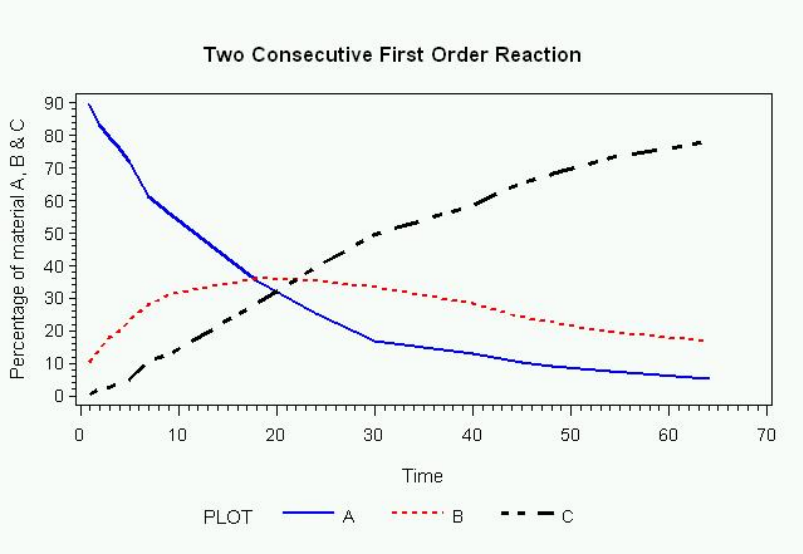


Figure 1: Two consecutive first order reaction

We can see that when material [B] is introduced, material [A] starts to deteriorate, this reaction forming other products which are referred to in the text as moieties devoid of the 17-dihydroxyacetone side chain which are represented by symbol [C]. The decomposition of [B] further causes the formation of [C] to a level of about 75%. The effect of the parameters are of paramount significance in determining the mechanisms and magnitude of the rate[13].

Compartmental model. In pharmacokinetics, compartmental models are considered for theoretical purposes to better understand how and what the body does with dose of drugs. When a drug enters a body it leaves the administration site to enter the central compartment, the drug is then digested to peripheral compartments until it is fully absorbed or until it leaves the body, this process is illustrated by Figure 2. Compartmental modeling can help scientists understand mathematically how the body processes dose of drugs and which drug regimens are effective. Although the mathematical approach only gives approximations, it helps one understand pharmacokinetics, pharmacodynamics, and other biological systems to a greater extent in practice.

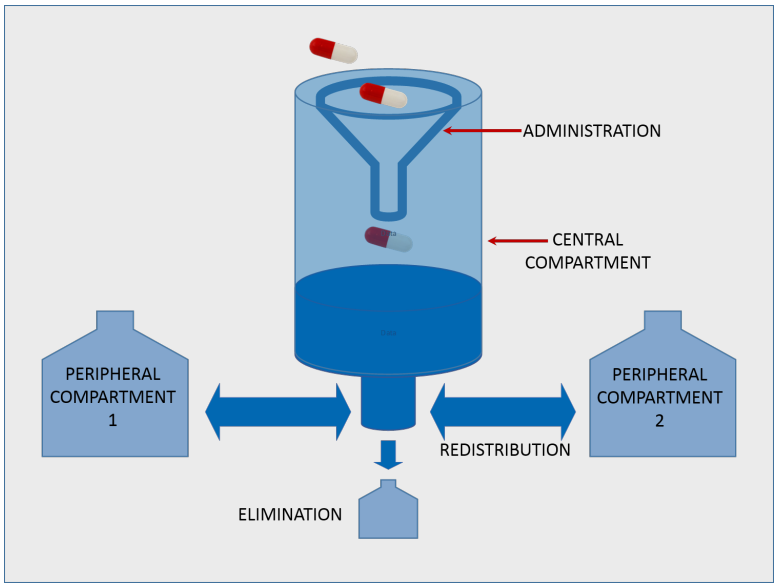


Figure 2: Compartmental model [15] (modified)

Consider the following compartmental model for optimum design,

$$\eta(t,\theta) = \theta_3 \left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right), \tag{18}$$

which has a maximum likelihood function of:

$$\begin{aligned} L(\theta|t) &= \prod_{i=1}^{n} \theta_3 \left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right) \\ log\, L(\theta|t) &= n \log \theta_3 + \sum log \left(\exp(-\theta_2 t) - \exp(-\theta_1 t)\right) \end{aligned}$$

with score vectors:

$$\frac{\partial log\, L(\theta|t)}{\partial \theta_1} = \sum \frac{t \exp(-\theta_1 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}, \tag{19}$$

$$\frac{\partial log\, L(\theta|t)}{\partial \theta_2} = -\sum \frac{t \exp(-\theta_2 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}, \tag{20}$$

and

$$\frac{\partial log\, L(\theta|t)}{\partial \theta_3} = \frac{n}{\theta_3}. \tag{21}$$

A 3×3 hessian matrix is expressed as:

$$\eta(\theta|t) = \frac{\partial log\, L(\theta|t)}{\partial \theta} \frac{\partial log\, L(\theta|t)}{\partial \theta^T}$$

$$= \begin{pmatrix} \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1^2} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1 \partial \theta_2} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1 \partial \theta_3} \\ \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2^2} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_2 \partial \theta_3} \\ \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_3 \partial \theta_1} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_3 \partial \theta_2} & \frac{\partial^2 log\, L(\theta|t)}{\partial \theta_3^2} \end{pmatrix},$$

where

$$\frac{\partial^2 log\, L(\theta|t)}{\partial \theta_1^2} = \sum \frac{t^2}{\exp(-\theta_2 t) - \exp(-\theta_1 t)} \left(\frac{\exp(-2\theta_1 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)} - \exp(-\theta_1 t) \right),$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_1\partial\theta_2} = \sum \frac{t^2 \exp(-\theta_1 t - \theta_2 t)}{(\exp(-\theta_2 t) - \exp(-\theta_1 t))^2},$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_1\partial\theta_3} = 0,$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_2\partial\theta_1} = \sum \frac{t^2 \exp(-\theta_1 t - \theta_2 t)}{(\exp(-\theta_2 t) - \exp(-\theta_1 t))^2},$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_2^2} = \sum \frac{-t^2}{\exp(-\theta_2 t) - \exp(-\theta_1 t)} \left(\exp(-\theta_2 t) + \frac{\exp(-2\theta_2 t)}{\exp(-\theta_2 t) - \exp(-\theta_1 t)}\right),$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_2\partial\theta_3} = 0,$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_3\partial\theta_1} = 0,$$

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_3\partial\theta_2} = 0,$$

and

$$\frac{\partial^2 log\, L(\theta|t)}{\partial\theta_3^2} = \frac{-n}{\theta_3^2}.$$

The parameters for this model can be defined as rates of change from one compartment to another. How the body decomposes the drug from one compartment to another is usually due to the exponential relationship with the rates of change over time.

For further reading, Mauger (1968)[13] gives an extensive explanation for the two consecutive first order reaction, and Fresen's (1984)[7] explains the compartmental model to a greater extent.

4.2 Parameter sensitivity

A fundamental technique for parameter sensitivity that was applied for both of the above models in this essay is partial differentiation. Using the SAS® software[1], one can apply the PROC NLIN procedure which uses the Taylor expansion series to generate estimated values for the parameters. With the estimated parameters we apply partial differentiation for all the θ values in question, then plot a graph that illustrates the partially differentiated equation over time. Figure 3 illustrates the plot against time for the differentiated equation. Note how both the plotted lines decrease over time in Figure 3, this illustrates that for very large values of time, the experiment will not be informative since the lines decrease over time [2].

The design points as well as the D-optimum criteria may be obtained by a PROC OPTEX procedure in the SAS. The search method that was specified in the OPTEX procedure was the modified Federov algorithm. Out of all the other search methods, this algorithm usually finds better design points, however, for the two models, all the search algorithm methods resulted in the same design points. Figure 4 illustrates the design points at time 3 and 9.275 at which measurements may be taken since these points yield optimum designs. These design points are also D-optimum as illustrated by the black line tangent to the curve at the black dots, this means that at these time points, a minimum variance-covariance is obtained over all other time points.

obs	*d*1	*d*2	*time*
1	0.08064	−1.61627	3.000
2	−0.04785	−3.58807	9.275

Table 1: design point for a two consecutive first order reaction

[1]The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NYC, USA.

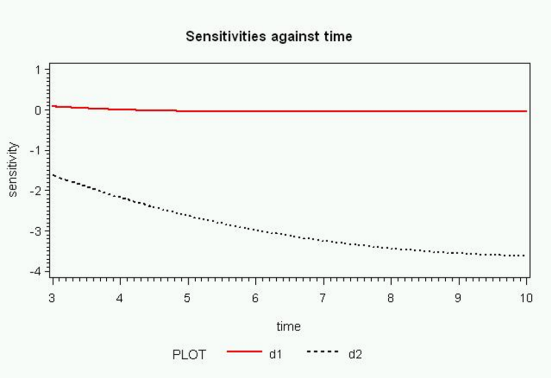


Figure 3: two consecutive first-order reaction, sensitivity against time.

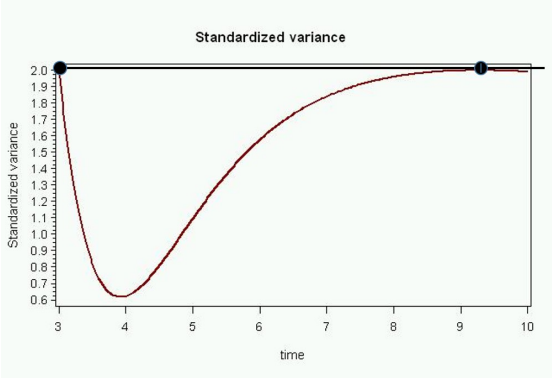


Figure 4: two consecutive first-order reaction, variance function $d(x, \xi^*, \theta)$, and local D-optimum design. Maximum value of 2 occurs at design point with time 3 and 9.275.

The compartmental model in Figure 5 and 6 posed a great challenged in estimating. This is a common challenge to all models with three parameters or more. Partial differentiation is applied for the three parameters and plotted in Figure 5 over time. Differentiation of the first parameter denoted by an orange line can be observed as increasing at a decreasing rate bound to reach a maximum then eventually decreasing at a later time period. The other differentiation plots can be observed as increasing at a decreasing rate to a maximum, then decreases towards zero. The third parameter, although it can not be observed clearly in the graph, it increases to a maximum below 1 then decreases towards zero as time increases. We can therefore note once again that it would not be ideal to perform the experiments at very small or very large values of time, since the experiment may be informative.

With the graphic illustration of Figure 6, it could be possible to assume one of the parameter to be nuisance in the model, since there are three design points as illustrated in Table 2, but only two design points are optimal as Figure 6 visually illustrates. It is unfortunate that we can not always use the General Equivalence Theorem to confirm the optimum designs for a nonlinear model since it is numerically difficult to derive. We can conclude this problem as one that is uncertain, since we find near optimum points but do not see three design points corresponding with three D-optimum points at a standardized variance of 3, as would be the case for a linear model. One possible solution to such a problem would be to obtain prior point estimates θ_0 or prior distribution for θ based on past experience[2]. Another approach could be to use the locally c-optimum design approach where the area under the curve, time to maximum concentration, and maximum concentration give an in-depth approach to solving for optimum designs.

obs	$d1$	$d2$	$d3$	*time*
1	6.7767	4.37441	−0.31775	1.250
2	17.2518	4.14104	−0.53195	4.075
3	22.9405	1.97717	−0.49514	7.000

Table 2: design points for a compartmental model.

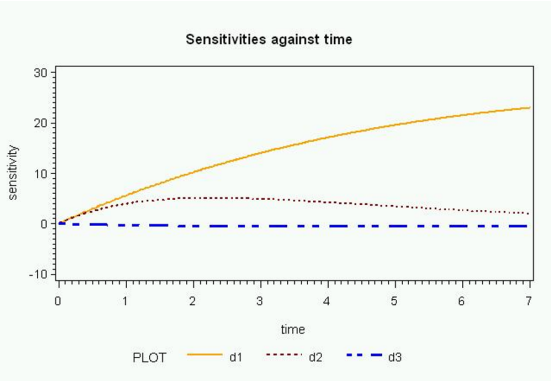


Figure 5: compartmental model, sensitivity against time.

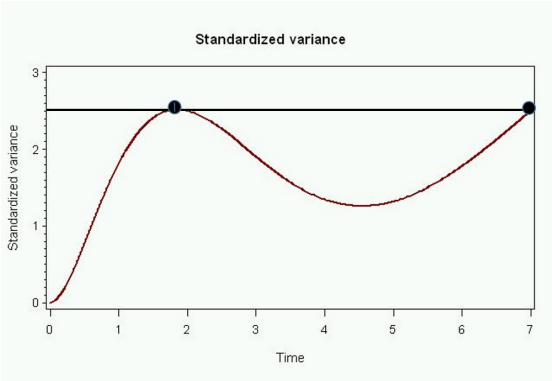


Figure 6: compartmental model, variance $d(x, \xi^*, \theta)$, and local D-optimum design.

5 Discussion

The importance of optimum design cannot be disputed in scientific studies, its results have been found to be useful in solving practical experimental problems in experimental designs. The main aim of this essay has been to review local optimum designs for nonlinear response models, of which the D-optimality criteria has been used. The statistical approach seeks to find optimum designs through observing points that have minimum variance given a design region χ. Since one cannot avoid random error in experimental designs, statistical methods are then essential for efficient design and analysis[2]. Not only do we have a problem of random errors, but nonlinear models have a problem of unknown parameter estimates. A method that was of use in this essay was the local approach which was executed using PROC NLIN in SAS, alternative and more efficient methods that can be of use are the Bayesian optimum design and sequential experimental design.

Optimum designs for nonlinear models may pose a challenge of not only unknown parameter estimates but also verifying if designs are optimum since the General Equivalence Theorem can be sometimes challenging to mathematically prove. Biologist on the other hand may require reasoning that cannot be explained by simple statistical approaches, therefor this may require one to diverge from the simple optimum design calculation process and use other mathematical applications such as those used by Anthony Atkinson (1993)[1]. Further research avenue that are possible in this niche research environment can possibly be application of optimal designs for models with three or more unknown parameter estimates.

References

[1] AC Atkinson, K Chaloner, AM Herzberg, and J Juritz. Optimum experimental designs for properties of a compartmental model. *Biometrics*, pages 325–337, 1993.

[2] AC Atkinson, AN Donev, and RD Tobias. *Optimum Experimental Designs, with SAS*. Oxford University Press Oxford, 2007.

[3] Adelchi Azzalini. *Statistical Inference Based on the Likelihood*, volume 68, chapter 3, pages 64–73. CRC Press, 1996.

[4] MPF Berger and W Wong. *An Introduction to Optimal Designs for Social and Biomedical Research*, volume 83. John Wiley & Sons, 2009.

[5] PL Bonate and DR Howard. *Pharmacokinetics in Drug Development: Advances and Applications*, volume 3, chapter 8, pages 176–177. Springer Science & Business Media, 2011.

[6] H Dette, VB Melas, and A Pepelyshev. Optimal designs for a class of nonlinear regression models. *Annals of Statistics*, pages 2142–2167, 2004.

[7] J Fresen. Aspects of bioavailiablity studies. Master's thesis, University of Cape Town, Rondebosch, Cape Town, 7700, South Africa, 1984.

[8] William H Greene. *Econometric Analysis*, chapter 17. Pearson Education India, 2003.

[9] DN Gujarati and DC Porter. *Basic Econometrics*, chapter 14, pages 525–530. McGraw-Hill Education, 1221 Avenue of the Americas, New York, NY, 10020, fifth edition, October 2008.

[10] SAS Intsitute Inc. *SAS OnlineDoc®, Version 8*. SAS Intsitute Inc, Campus Drive, Cary, North Carolina 27513, September 1999.

[11] J Kiefer. Optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 272–319, 1959.

[12] WF Kuhfeld. Efficient experimental designs using computerized searches. *Research Paper Series, SAS Institute Inc*, 1997.

[13] John William Mauger. A mathematical analysis of an irreversible consecutive first order reaction using hydrocortisone sodium succinate as a model. Master's thesis, University of Rhode Island, 45 Upper College Rd, Kingston, RI 02881, United States, 1968.

[14] K Murota. *Discrete Convex Analysis*, chapter 1, pages 9–10. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematic, illustrated edition, 2003.

[15] R Rowland and P Squires. Pharmacokinetics - an overview. *AnaesthesiaUK*, November 2011.

[16] CG Swain. The kinetic analysis of consecutive irreversible first order reactions. *Journal of the American Chemical Society*, 66(10):1696–1700, 1944.

[17] J Warfield. *Sequential Optimal Designs with Unknown Link Function*, chapter 1, pages 1–8. ProQuest, 2008.

[18] A Wilhelm. How to Obtain Efficient Exact Designs from Optimal Approximate Designs. In Professor Dr. Albert Prat, editor, *COMPSTAT*, pages 495–500. Springer, Physica-Verlag HD, 1996.

[19] M Yang, S Biedermann, and E Tang. On optimal designs for nonlinear models: a general and efficient algorithm. *Journal of the American Statistical Association*, 108(504):1411–1420, 2013.

Appendix

SAS® software used to obtain all relevant results:

```
/*************************************************************
        Model1 Two consecutive first order reaction
*************************************************************/
data ccc;
        set cons;

goptions reset=all;
axis1 label=(angle=90 'Percentage of material A, B & C');
axis2 label=('Time');
symbol1 i=join width=2 color=blue ;
symbol2 i=join line=2 width=2 color=red ;
symbol3 i=join line=42 width=2 color=black ;
title2 "Two Consecutive First Order Reaction";
proc gplot data=ccc;
        plot (A B C)*time / overlay haxis=axis2 legend vaxis=axis1;
run;
title2;

proc nlin data=ccc ;
        parameters b1=1 b2=0.1;
        model b=(b1/b1-b2)*(exp(-b2*time)-exp(-b1*time));
run;

proc iml;
/* calculation */
p_b1=1; p_b2=0.1;

do t=3 to 10 by 0.025;
        d1 = d1 //((t*p_b1*exp(-t*p_b1))-exp(-t*p_b1)+exp(-t*p_b2))/(p_b1-p_b2)
                -(p_b1*(exp(-t*p_b2)-exp(-t*p_b1)))/(p_b1-p_b2)**2;
        d2 = d2 //(p_b1*(exp(-t*p_b2)-exp(-t*p_b1))) / ((p_b1-p_b2)**2)
                -(t*p_b1*exp(-t*p_b2))/(p_b1-p_b2);

        ti=ti//t*1;
end;

q=d1||d2||ti;
create wanda from q[colname={'d1' 'd2' 'time'}];
append from q;
/*****Sensitivity Plot*****/
goptions reset=all i=join;
axis1 label=(angle=90 'sensitivity') ;
axis2 label=('time') minor=(number=0.9) order=3 to 10 by 1 ;
symbol1 color=red width=2 ;
symbol2 color=black width=2 line=2;
title2 "Sensitivities against time";
proc gplot data=wanda;
   plot (d1 d2)*Time / overlay legend vaxis=axis1 haxis=axis2;
run;
title2;
```

```
title2 "Optimum Design Criteria";
proc optex data=wanda coding=none;
   model d1 d2 / noint;
   id Time;
   generate n=2 method=m_fedorov niter=1000 keep=10;
   output out=Design;
run;
title2;

title2 "Design Points";
proc sort  data=Design;
        by time;
proc print data=Design; /* 3, 9.275 are my two design points*/
run;
title2;

data haha;
        set Design;
        y = rannor(1);
data haha;
        set haha wanda;
proc reg data=haha noprint;
   model y = d1 d2 / noint;
   output out=d h=h;/*h is the hat matrix or projection matrix*/
   run;
data d;
        set d;
    d = 2 * h; /* 2 = NObs in Design */
goptions reset=all i=join;
axis1 label=(angle=90 'Standardized variance') ;
axis2 label=('Time');
symbol1 color=darkred width=2;
title2 "Standardized variance";
proc gplot data=d;
   where (y = .);
   plot d*Time / vaxis=axis1;
run;
title2;title1;
```

```
/*************************************************************
                Model2 Compartmental model
*************************************************************/
data ddd;
set comp;
proc nlin data=ddd noprint;
        parameters b1=0.08754 b2=0.437717 b3=6.048279;
        model CONCENTRATION=b3*(exp(-b1*time)-exp(-b2*time));
        run;

proc iml;
/* calculation */
p_b1=0.08754; p_b2=0.437717; p_b3=6.048279;

do t=0 to 7 by 0.025;
        d1 = d1 // (p_b3*t*exp(-t* p_b1));
        d2 = d2 // (p_b3*t*exp(-t* p_b2));
        d3 = d3 // (exp(-t* p_b2)-exp(-t* p_b1));
        ti=ti//1*t;
end;
dd1=d1##2||t;
dd2=d2##2||t;
dd3=d3##2||t;
call sort(dd1,1);
max_timedd1=dd1[281,];
call sort(dd2,1);
max_timedd2=dd2[281,];
call sort(dd3,1);
max_timedd3=dd3[281,];
**print max_timedd1 max_timedd2 max_timedd3, dd1, dd2, dd3;
q=d1||d2||d3||ti;
create wanda from q[colname={'d1' 'd2' 'd3' 'time'}];
append from q;

goptions reset=all i=join;
axis1 label=(angle=90 'sensitivity') ;
axis2 label=('time') minor=(number=0.9) order=0 to 7 by 1;
symbol1 color=orange line=1 width=2 ;
symbol2 color=darkred line=2 width=2 ;
symbol3 color=blue line=42 width=2;
title2 "Sensitivities against time";
proc gplot data=wanda;
   plot (d1 d2 d3)*time / overlay legend vaxis=axis1 haxis=axis2;
run;
title2;

title2 "Optimum Design Criteria";
proc optex data=wanda coding=none;
        model d1 d2 d3 / noint;
        id time;
        generate n=3 method=m_fedorov niter=1000 keep=10;
        output out=Design;
run;
```

```
title2;

title2 "Design Points";
proc sort  data=Design;
        by time;
proc print data=Design; /*Design points at time ()*/
run;
title2;
/************************************************/
data jabu;
        set Design;
        y = rannor(1);
data jabu;
        set jabu wanda;
proc reg data=jabu noprint;
   model y = d1 d2 / noint;
   output out=d h=h;
   run;
data d;
        set d;
        d = 3 * h;
   /* 3 = NObs in Design */
proc print data=d noprint;
run;
goptions reset=all i=join;
axis1 label=(angle=90 'Standardized variance') ;
axis2 label=('Time');
symbol1 color=darkred width=2;
title2 "Standardized variance";
proc gplot data=d;
   where (y = .);
   plot d*Time / vaxis=axis1 haxis=axis2;
run;
title2;title1;
```

Compressed Sensing and Statistical Preprocessing of fMRI Data

Altus Coetzee 12023842

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr. IN Fabris-Rotelli

Department of Statistics, University of Pretoria

2 November 2015

Abstract

In this paper research is done in the field of Functional Magnetic Resonance Imaging (fMRI). The statistical methods used to the adjust the sequences of fMRI images accumulated during such a study are investigated and explained. Sparsity is assumed for these images and compressive sensing applications investigated. Finally an application is done where a limited number of measurements are sampled from such an assumed sparse image and a reconstruction done with enlightening results, which can be implemented in future MRI data.

Declaration

I, *Marthinus Albertus Coetzee,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Student: Martinus Albertus Coetzee

_____
Supervisor: Inger Fabris-Rotelli

_____
2 November 2015

Acknowledgements

The author would like to give much needed thanks to the supervisor and co-author, Dr Inger Fabris-Rotell, who assisted and supported greatly in the compilation of this honours project. I would like to thank Dr Colin Turner, Radiologist at Mediclinic Nelspruit who gave me much needed advice and insight into this research topic and also provided me with the MR images used in the report. This research was supported by a South African Research Chair Initiative (SARChI) bursary awarded to the Department of Statistics at the University of Pretoria. This work is based on the research supported in part from the National Research Foundation of South Africa for the Grant 90315. Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard.

Contents

List of Figures

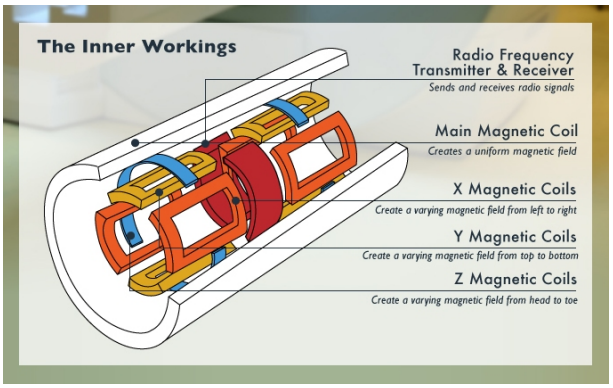


Figure 1: The components of a MRI scanner.

1 Introduction

Functional magnetic resonance imaging (fMRI) is a method of visualizing how physical - and neuronal[1] processes are related. This can be performed without safety concerns on a wide group of individuals and is a popular method because of its non-invasive nature. [20]

fMRI stems from magnetic resonance imaging (MRI). MRI is a safe an non-invasive method to obtain high quality medical images of tissues and organs in the human body [17]. A MRI scanner is a tube-like structure with a very strong magnet. An electric current is passed through this magnet which produces a strong magnetic field causing the various nuclei present in the human body of the subject under observation to align. Magnetic pulses, called radio frequency (RF) pulses, are omitted by the scanner that penetrates these areas of aligned nuclei, known as slices, and causes the aligned nuclei to deviate from their aligned state. Removal of this RF pulse causes an incentive for these nuclei to return to their aligned positions, inducing a current picked up by a receiver coil placed over the area under consideration, providing the MR signal. Using mathematical procedures this MR signal is processed to ultimately produce high quality images of the human body, which are combined to obtain an overall image sequence of the area under consideration. [17, 20] Figure[2] 1 shows the basic components in a MRI scanner.

There are various neuroimaging methods that aid in the research of neuronal activity [17]. This paper will focus on the method of fMRI.

During an fMRI procedure the subject is placed inside a MRI scanner and instructed to perform mental tasks purposed to manipulate mental processes [20]. When the neurons in the brain become active as result of these exercises, they consume oxygen provided by blood flow. Blood flow to the area of neuronal activity is increased, known as hemodynamic response, beyond what is needed to replenish the neurons of oxygen [20]. An excess of oxygen-rich blood is formed relative to non-active areas of the brain [20]. Oxygenated - and DE-oxygenated blood each have different magnetic properties, known respectively as diamagnetic and paramagnetic [17]. Neuronal activity is followed by an increase in oxygenated blood flow through that area [20]. This increase in blood flow is greater than is needed to replenish the activated cells of oxygen. Due to its magnetic property this excess oxygen rich blood increases the MR signal, while DE-oxygenated blood does the opposite [17]. This phenomenon can be viewed better by looking at Figure[3] 2. This is known as Blood Oxygen Level Dependent (BOLD) signal [20].

As result the series of acquired MR images display the differences in the measured MR signal, which is then used by specialists to pinpoint areas of brain activity [17, 20]. This signal produces images of the subject's brain where oxygen-rich and oxygen-poor areas show up in different colours. Figure[4] 3 shows how this difference in signal produces blue and red areas, that represents the different levels of blood oxygen.

The wealth of data collected by a fMRI procedure have to be analyzed before conclusions are made.

[1] Relating to a nerve cell or neuron, by Cambridge Dictionaries Online
[2] Image courtesy of www.howtolearn.com
[3] Image courtesy of www.nature.com
[4] Image courtesy of www.tm-ireland.org

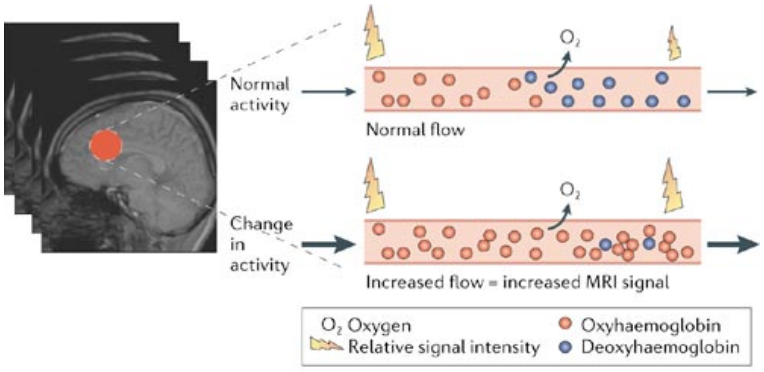


Figure 2: Change in MRI signal caused by a change in neuronal blood flow.

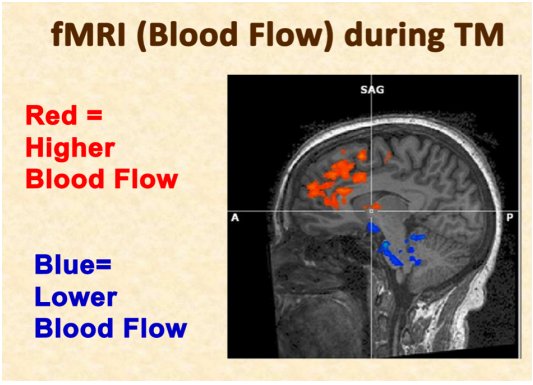


Figure 3: Colours on fMRI scan showing the presence or absence of neuronal activity.

Preprocessing techniques have to be performed on this data to make it more suitable for the analysis process. There is great variability present in the data that can easily swamp the small changes in BOLD signal originating from neuronal activity. This variability can be classified into controllable - and non-controllable variation. Preprocessing steps remove as much controllable variation as possible form the raw fMRI data before analysis can be done. [4, 20]

This paper focuses on the preprocessing steps implemented on fMRI data.

2 Preprocessing theory in fMRI

Functional magnetic resonance imaging is a revolutionary technique that is used to capture brain activity [20] in patients performing mental tasks the purpose of which to elicit mental responses. Brain nuclei are energized by the strong magnetic field produced by an electromagnet in a MR scanner and a MR signal is measured, strengthened by the presence of oxygen-rich blood following neuronal activity. As a result areas of brain activity can be identified on the sequence images obtained from such a procedure. Analysis has to be done on the large amount of complex data produced by such an experiment and a number of software packages are used for this process.

Statistical Parametric Mapping (SPM)

This was the first software package widely used in fMRI analysis. It has a MATLAB command line interface with scripting abilities [18] and source code is available for use. With limited visualization capabilities it remains one of the most popular packages to use today since it is freely provided online and easy to interpret.

FMRIB Software Library (FSL)

A software package developed at Oxford University with powerful visualization capabilities and a fast analysis process by using computing clusters[5]. Commands are executed via a command line or via a Graphical User Interface (GUI).

Analysis of Functional NeuroImages (AFNI)

A powerful software package in terms of its visualization capabilities. Command line prompts [18] or a GUI is used in the implementation of this software on fMRI sequences and source code is provided for use.

BrainVoyager

This software package was developed at Brain Innovation and is the mainly used package in the commercial environment. It provides a fully GUI with some scripting [18] abilities. It is known for its user friendliness and refined user interface.

Freesurfer

This is a freely provided toolbox [9] used in the analysis process of fMRI data. It can be run on various software platforms and is an open-source software package.

The rapid development of medical technology and the increasing demand to relate biological structure to function [1] has led to an ever increasing demand for imaging capabilities in the fields of medicine and psychology. An image is a graphical representation of a matrix, X, of numbers that represents different greyscale values as shown in Figures 5[6] and 4[7] and defined in the definition below.

Definition 1. A greyscale image is a function on a pixel domain $X \subseteq \mathbb{Z}^2 : f(x, y) : X \to \{0, 1, \cdots, 255\}$.

[5] A collection of computers connected to function as a single unit on a single task

[6] Image courtesy of www.tutorialspoint.com

[7] Image courtesy of py.processing.org

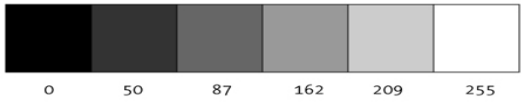


Figure 4: An indication of the the greyscale intensities adopted by the pixel domain X.

Figure 5: A greyscale image where the indicated region is represented in a matrix with values ranging from 0 to 255, representing different greyscale intensities.

$$X_1 = \begin{bmatrix} 235 & 215 & 193 & 191 & 201 & 206 & 202 & 203 & 204 \\ 180 & 236 & 209 & 210 & 196 & 199 & 199 & 214 & 205 \\ 114 & 218 & 202 & 198 & 189 & 197 & 213 & 218 & 195 \\ 12 & 120 & 255 & 235 & 202 & 207 & 200 & 204 & 195 \\ 25 & 40 & 120 & 174 & 242 & 228 & 208 & 202 & 194 \\ 46 & 61 & 52 & 62 & 163 & 217 & 218 & 220 & 215 \\ 36 & 51 & 69 & 72 & 102 & 141 & 146 & 166 & 185 \\ 37 & 45 & 51 & 54 & 93 & 126 & 115 & 109 & 123 \\ 44 & 43 & 29 & 50 & 108 & 143 & 139 & 126 & 116 \\ 48 & 51 & 39 & 55 & 105 & 136 & 136 & 129 & 110 \\ 51 & 55 & 44 & 57 & 102 & 131 & 135 & 132 & 124 \end{bmatrix}$$

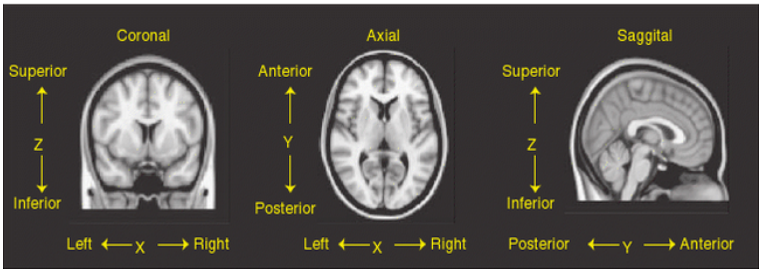


Figure 6: The axes assigned to MR images in the neurological environment.

The visual quality of captured images can be enhanced by performing image sharpening on it. Captured images often show a loss of quality along the edges and which can be corrected using image sharpening. In [16] it is assumed that pixels in a fixed neighbourhood of an image have a common mean, μ_0, and variance, σ_0^2. The pixels in areas of low contrast within the image tend to be more or less the same as μ_0, whereas pixels in high contrast areas, which is along the edges of an image, deviates from μ_0 and this causes the loss of quality along the edges of an image. To sharpen the edges of an image different filters are passed over the image to extract information which is then again applied to the original image to obtain an enhanced image.

2.1 Spatial Transformations

The area occupied by an image itself is referred to as the spatial domain and the methods applied in this domain work by manipulating the pixels or voxels of an image [12]. A transformed image is obtained that is corrected of some anomalies borne by the original image. During fMRI studies a sequence of brain images are captured of individuals. A spatial transformation is then applied to realign this sequence of images in the best possible way to correct for motion related artifacts and the various shapes of human brains [8] to aid in the comparing and statistical analysis of such data across individuals. Its major advantage is rooted in concluding an overall result [2] after being applied to images from multiple individuals. fMRI images are registered in two steps. Registration is the first step involving determining the parameters to be used in the transformation through a transformation model. Secondly a transformation is applied based on the determined parameters. The number of parameters determines the complexity [20] of a transformation. A manipulation to an image is made by each parameter and a model with many parameters would result in a more complex transformation. Once the parameters are determined a resampling of the image should be done to apply the transformation [2]. Voxel coordinates of the original image are transformed into a new space called a standard space and various forms of interpolation are then used to determine the intensity of the transformed voxels from the original voxels. In this standard space it is possible to cross-examine fMRI images from different individuals and studies to add to the value of the analysis process.

MRI images are captured as three-dimensional representations of physical objects. To universally distinguish locations in such an image, it is assigned a X, Y and $Z-axis$ respectively, as in Figure 6[8], indicating left to right, anterior(forward) to posterior(backward) and superior(upward) to inferior(downward) directions.

2.1.1 Transformation models

Affine transformation model This model focuses on linearly manipulating fMRI images. It is a simple transformation model and transformations such as rotations, scaling, shearing and translation are possible as shown in Figure 7[9]. It may occur that all the transformations are performed but this is not always the case. It can be naturally assumed that the shape and size of a subject's head remains constant during the duration of a fMRI study. Following this assumption motion defects can be corrected by applying translations and rotations to the sequence of images as in Figure 7.

[8] Image courtesy of [20]
[9] Image courtesy of [20]

The original coordinates are linearly transformed to new coordinates. A transformation matrix is applied to the original coordinates which makes it possible.

Consider a two-dimensional coordinate system $(x, y) \in \mathrm{X} \subseteq \mathbb{Z}^2$, i.e. x and y denotes the row - and column coordinates of an image in matrix notation. Let C_1 and C_2 denoted the original - and transformed coordinates respectively and T the transformation matrix, so that,

$$\mathrm{c}_2 = \mathrm{T}\mathrm{c}_1$$

with,

$$\mathrm{c}_1 = \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

for $(x_i, y_i) \in \mathrm{X}$.

The transformation matrix, T, varies according to the transformation to be done, for example,

- translation

$$\mathrm{T}_{trans} = \begin{bmatrix} 1 & 0 & \mathrm{T}_x \\ 0 & 1 & \mathrm{T}_y \\ 0 & 0 & 1 \end{bmatrix}$$

with T_x and T_y denoting the translation in the x - and y direction.

- rotation

$$\mathrm{T}_{rot} = \begin{bmatrix} \cos(\theta) & \text{-}\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

with θ denoting the angle of rotation.

- scaling

$$\mathrm{T} = \begin{bmatrix} \mathrm{s}_x & 0 & 0 \\ 0 & \mathrm{s}_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

with s_x and s_y denoting the scale of magnification in the x - and y direction.

- shearing:

$$\mathrm{T} = \begin{bmatrix} 1 & \mathrm{Sh}_x & 0 \\ \mathrm{Sh}_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

with Sh_x and Sh_y denoting the shearing in the x - and y direction.

Piecewise linear transformations fMRI images are decomposed into smaller sections where-after linear transformations are applied to each section as necessary. It is therefore a generalization of affine transformations.

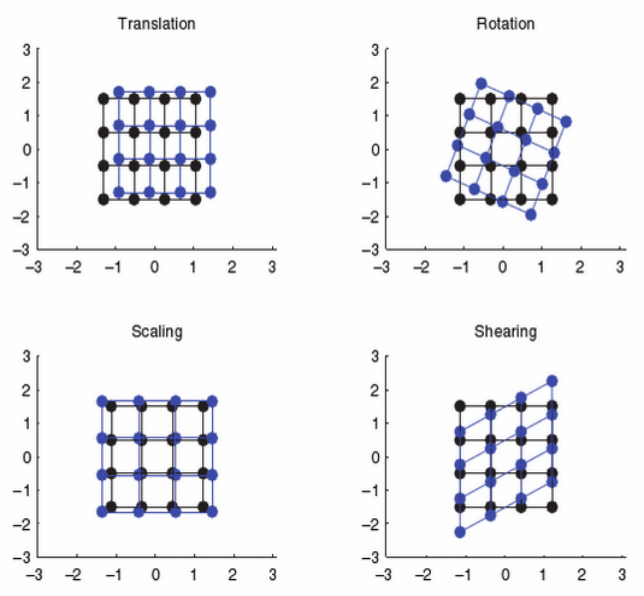


Figure 7: The linear transformations possible when an affine transformation model is used.

Rigid body transformations Rigid body transformations are used in images from a single subject and is also a subset of affine transformations. This typically involves a translation and rotation about orthogonal axes. The order in which this is carried out is important.

A translation is done by matrix,

$$\begin{bmatrix} 1 & 0 & 0 & \mathrm{T}_x \\ 0 & 1 & 0 & \mathrm{T}_y \\ 0 & 0 & 1 & \mathrm{T}_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

with T_x, T_y and T_z denoting a translation in the x -, y - and z direction respectively.

A rotation is done by matrices,

$$\mathrm{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & \sin(\theta) & 0 \\ 0 & \text{-}\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathrm{B} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ \text{-}\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } \mathrm{C} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 & 0 \\ \text{-}\sin(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

with A, B and C denoting rotations of θ radians in the x -, y - and z direction respectively.

Nonlinear transformations Nonlinear transformations allow much greater manipulation capabilities than affine transformations. With nonlinear transformations it is possible to achieve any transformation of image voxels instead of only transformation in a linear fashion as with affine transformation. Transformed images can be viewed in a higher dimensional form adding to the complexity of such a procedure.

In general linear transformations are much more fit to transform whole images while nonlinear transformations focuses more on local transformations within an image.

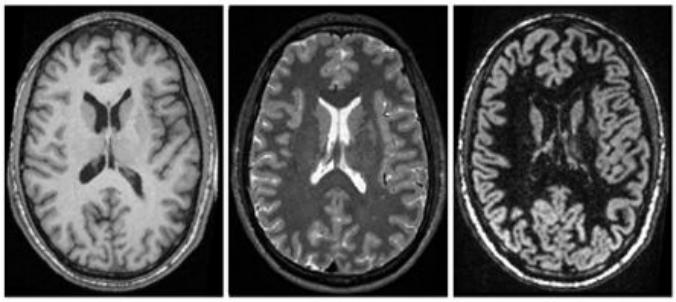


Figure 8: MR images with different contrasts as a result of brain matter responding differently to varying intensities of the scanner.

2.1.2 Cost functions

Cost functions as described in [20] emphasize degree of similarity between fMRI images, to assist in determining the transformation resulting in the best alignment between these images and is indicated by C. Well-aligned images are inferred by a small cost function with the inverse applying to badly aligned images. Alignment refers to two image's similarities and the degree to which they can be compared in a typical study. A reference image is chosen from the sequence taken and comparisons are made to it. Registering images, as defined before, plays an important role in the choice of cost function. Naturally with neural images of the same subject, defects concerning different shapes and sizes of the brain do not occur and the cost functions should only determine the similarity between the intensities of two images. Registering such images are referred to as within-modality registration. Between-modality registration on the other hand refers to registering fMRI images that are not from the same subject and have different contrasts as shown in Figure 8[10]. Images are registered in a geometrical way as stated in [14]. A cost function should be chosen to identify the degree of similarities across these different images. Jenkinson et al. confirms in [14] that a study done by West et al. in [23] showed that cost functions focusing on intensities rather than the geometrics of images are more effective. A list of cost functions follow.

Least squares This cost function is very commonly used in within-modality registration. It measures the squared deviation of voxel intensities from two images.

A formula is given by,

$$\mathrm{C} \quad = \quad \sum_{i=1}^{n} (V_i - W_i)^2$$

with $V_i$and W_i denoting the intensity of the i^{th} voxel in images V and W. C can take on any value greater than zero [20], excluded.

Normalized correlation Correlation identifies the linear relationship between two variables. As suggested by the name, normalized correlation measures the linear relationship between the intensities of voxels within the compared images.

A formula is given by,

$$\mathrm{C} \quad = \quad \frac{\sum_{i=1}^{n} (V_i W_i)}{\sqrt{\sum_{i=1}^{n} V_i^2} \sqrt{\sum_{i=1}^{n} W_i^2}}$$

with $V_i$and W_i denoting the intensity of the i'th voxel in images V and W. C can take on any value in the range of -1 to 1 [2, 20], both excluded.

[10]Image courtesy of www.dzne.de

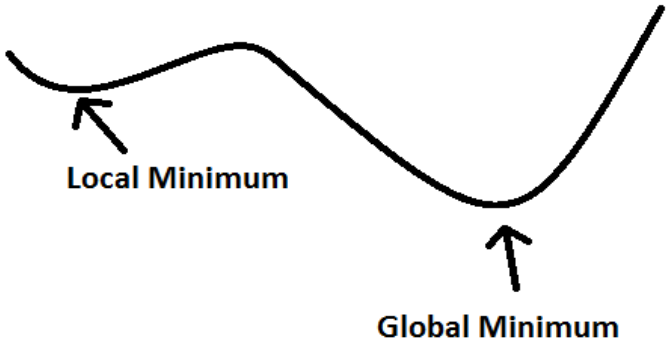


Figure 9: A pictorial representation of local minima: a local minimum is determined rather than the preferred global minimum.

Mutual information This cost function is based on the concept of entropy. Entropy describes the uncertainty inherent in a signal and should be minimized as far as possible. The use of this cost function is not restricted to within - or between modality registration, although it is more useful for between modality registration. Mutual information measures the similarity between images and it reaches a maximum when the entropy between images are at a minimum. This is used in software packages such as FSL, AFNI and and SPM. [20]

Correlation ratio The correlation ratio is a measure of the relationship between the variances of images. Ideally this should be zero. It can be used in both within - and between modality registration, and is implemented in the software package FSL. [20]

2.1.3 Resolving the parameters for the transformation

In order to determine the best transformation the parameters need to be estimated. For this an optimization technique is implemented. In [3] optimization is described as a method whereby an iterative procedure is used after choosing an initial estimate. For more information on optimization methods refer to [22]. Parameters are to be determined that would result in the minimal cost function. Using optimization methods can lead to a problem known as local minima. Local minima describes the problem, shortly illustrated in Figure 9[11], that suboptimal parameters can be determined as a result of the optimization method getting caught in the different local minima existing in a high dimensional case [20]. By working with smooth images the number of local minima are decreased as reported in [3].

Regularization

Most real world problems that involve solving some subset of parameters $\mathbf{x} \in \mathbb{R}^N$ from measured data $\mathbf{y} \in \mathbb{R}^m$, also known as inverse problems [7], can be formulated as

$$\mathbf{A}\mathbf{x} = \mathbf{y} \text{ where } \mathbf{A} \in \mathbb{R}^{mN} \tag{1}$$

and could be ill-posed. An inverse problem, such as (1), is said to be ill-posed if any of the following conditions are not met [13, 19] :

- There exists a solution.
- This solution is idiomatic.
- There exists a continuous dependence between the data and this solution.

[11] Image courtesy of blog.demofox.org

Ill-posed problems lead to solutions of (1) that are unstable and non-unique. Regularization techniques aim to convert problems that are ill-posed to well-posed problems, satisfying the three conditions above, which would address the issues of *existence*, *uniqueness* and *continuity* at hand.

Sparse Optimization

During an experiment data is sampled [11] from $\mathbf{x} \in \mathbb{R}^N$ and is represented in the vector $\mathbf{y} \in \mathbb{R}^m$. It can be of interest to extract the original, unknown signal from the measured data and this problem can be written as a well-known linear system,

$$\mathbf{Ax} = \mathbf{y} \text{ with } \mathbf{A} \in \mathbb{R}^{mN} \tag{2}$$

which needs to be solved for $\mathbf{x}$ [21].

CASE 1 If $m = N$:

An exact solution of the form $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$ can be determined, since the inverse of A exists.

CASE 2 If $m > N$:

The linear system is over-determined, the columns of A are of full rank [15] and a solution of the form $\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}$ is obtained.

CASE 3 If $m < N$:

The linear system is now said to be under-determined, and has infinitely many solutions. Normally it is not possible to get a unique solution, **x,** using conventional methods if no extra information is available. Clearly other methods needs to be sought in order to guarantee a unique solution.

Now consider a signal $\mathbf{x} = (x_1, x_2, \cdots, x_N) \in \mathbb{R}^N$. This signal $\mathbf{x}$ can be expressed as

$$\mathbf{x} = \sum_{i=1}^{N} w_i \psi_i \tag{3}$$

where $\Psi := [\psi_1, \psi_2, \cdots \psi_N]$ is a $N \times N$ basis matrix [5], with each ψ_i representing a $N \times 1$ orthonormal basis of $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{w} : N \times 1$ consisting of weighting coefficients $w_i = \langle \mathbf{x}, \psi_i \rangle : i = 1, 2, \cdots, N$. The signal can be represented by $\mathbf{x}$ or $\mathbf{w}$, each in the time - and $\mathbf{\Psi}$ domain respectively. The assumption that $\mathbf{x}$ is sparse implies the existence of a basis $\mathbf{\Psi}$, such that (3) can represented by only a small number of significant coefficients, meaning a small number of non-zero/large coefficients. Assuming that there are S of these significant coefficients, results in the other $N - S$ coefficients being negligible. Therefore it is now possible to express this signal as a linear combination of $S \ll N$ basis vectors, since only S of the w_i's in (3) are sufficiently large, so that $\mathbf{x}$ is now approximately,

$$\tilde{\mathbf{x}} = \sum_{i=1}^{S} w_i \psi_i \tag{4}$$

which is known as the sparse representation of $\mathbf{x}$. It is clear that the most salient/useful information in $\mathbf{x}$ is expressed by $\tilde{\mathbf{x}}$.

Extracting an unknown signal [11] from measured data as explained by (2) when the number of measurements, m, is less than the required size N of the signal, seems to contradict logical reasoning, however this is indeed possible when it is assumed that $\mathbf{x}$ is sparse and can be done using several competent algorithms. Designing the $m \times N$ measurement - or sensing matrix $\mathbf{A}$, is also very important and complex [5], as it needs to preserve all the salient information present in the sparse signal during the dimensionality reduction from $\mathbb{R}^N$ to $\mathbb{R}^m$. This process of extracting an unknown, sparse signal from measurements made is known as sparse optimization.

To obtain a sparse representation, $\tilde{\mathbf{x}}$, of a vector $\mathbf{x}$, intuitively requires finding the minimum number of nonzero entries in $\mathbf{x}$ so that $\mathbf{Ax} = \mathbf{b}$ is true. That is finding as in [24],

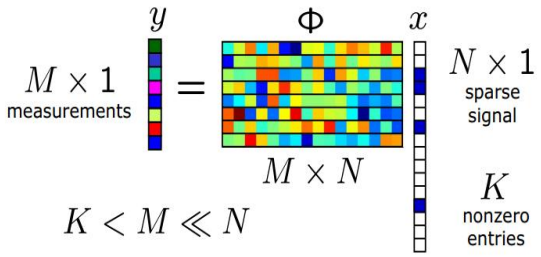


Figure 10: A pictorial illustration of a sparse signal. As indicated this signal is sparse with K = 4 and the measurement matrix is represented by $\mathbf{\Phi}$ in this case.

$$\underset{\mathbf{x}\in\mathbb{R}^N}{min}\ \{\|\mathbf{x}\|_0 :\ \mathbf{Ax} = \mathbf{b}\} \tag{5}$$

where $\|.\|_0$ is a metric that indicates the number of non-zero entries is $\mathbf{x}$. It can be problematic to solve (5) and Candes et al. [6] propose the following alternative problem, know as the *Basis Pursuit* problem,

$$\underset{\mathbf{x}\in\mathbb{R}^N}{min}\ \{\|\mathbf{x}\|_1 :\ \mathbf{Ax} = \mathbf{b}\} \tag{6}$$

where $\|.\|_1$ is a metric that returns the sum of the absolute entry values of $\mathbf{x}$, a good alternative to (5) since it will give an accurate representation of the non-zero entries present. This is a convex optimization problem and allows a sparse $\mathbf{x}$ to be uniquely obtained from $\mathbf{y}$.

Magnetic Resonance Imaging (MRI) and Sparsity

MR images of high resolution are obtained by a costly, lengthy and uncomfortable measurement procedure [10, 11]. Patients undergoing such a procedure are required to lie completely still in a cramped and noisy environment, which can not always be expected. Sparsity can prove to be invaluable for MRI, as this could curb the time necessary [24] for such a procedure by only taking the least required amount measurements [10] from the patients involved, which may result in images of sufficient quality. It is important to note that MR images can be represented sparsely in the wavelet domain [24], meaning a linear combination of the wavelet basis, W, and its sparse representation [11] can be constructed. It is now possible to reconstruct an image of much better quality from the images of poor quality with a well-designed measurement matrix $\mathbf{A}$, which can be done algorithmically. This can be related in a problem such as (2) where,

$$\mathbf{A} = \mathcal{LF}$$

with,

- $\mathcal{L}$ a linear mapping, and,
- $\mathcal{F}$ a discrete Fourier transform.

It is important to note that the above theory is also applicable to fMRI data, which is investigated in this research paper.

3 Application

A MATLAB program[12] was used to reconstruct a brain MR image using a sample ($S\%$) of the original image values, sampled via a finite number of radial lines (L) in the Fourier domain of that image. The original

[12] Justin Romberg: $l1-$magic from statweb.stanford.edu/~candes/l1magic/

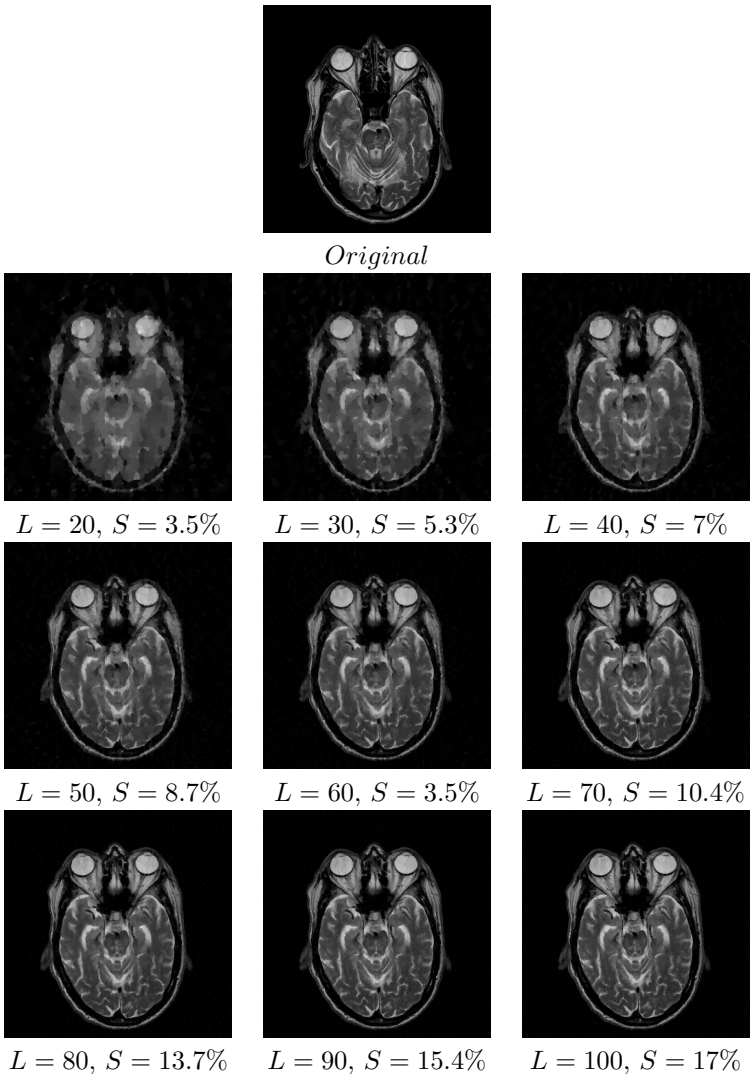


Figure 11: The original brain MR image (top), followed by 9 images where L is increased from 20 to 100 in increments of 10. The percentage of sampled points S, to the original number of pixels, is also indicated.

image[13] is assumed sparse in the Fourier domain. This is done for $L = 20, 30, 40, ..., 100$. It is interesting to view the reconstructed image quality as the number of radial lines taken - which are relatively small - increases. This is a property that can be very useful in MRI and fMRI data. See Figure 11. It can be seen that the quality is almost identical to that of the original image from $L = 70$. The results of this procedure are shown in Figure 11.

[13]This image was provided by Dr Colin Turner, Radiologist at Mediclinic Nelspruit.

4 Conclusion

In this report research was done in the field of fMRI. Attention was given to the possibility of improving the efficiency the process of obtaining MRI images, whilst still maintaining image quality. Sparsity was investigated which can be seen as a building block for reaching the before-mentioned goal. An application was done where a limited number of measurements taken from a brain MR image, assuming a certain level of sparsity of the original, are used to reconstruct the full-size original image with very good results. This is a very applicable to the field in question and as such it is much needed to do further research into this and develop a much needed, sound application using the assumption of sparsity.

References

[1] Michael D Abr'amoff, Paulo J Magalhães, and Sunanda J Ram. Image processing with imagej. *Biophotonics International*, 11(7):36–43, 2004.

[2] John Ashburner and Karl J Friston. Spatial transformation of images. *Human Brain Function*, pages 43–58, 1997.

[3] John Ashburner and Karl J Friston. Nonlinear spatial normalization using basis functions. *Human Brain Mapping*, 7(4):254–266, 1999.

[4] F. Gregory Ashby. *Statistical Analysis of fMRI data*. MIT press, 2011.

[5] Richard Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4), 2007.

[6] Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, 2006.

[7] Zhe Chen and Simon Haykin. On different facets of regularization theory. *Neural Computation*, 14(12):2791–2846, 2002.

[8] Christos Davatzikos. Spatial transformation and registration of brain images using elastically deformable models. *Computer Vision and Image Understanding*, 66(2):207–222, 1997.

[9] Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

[10] Massimo Fornasier and Holger Rauhut. Compressive sensing. In *Handbook of Mathematical Methods in Imaging*, pages 187–228. Springer, 2011.

[11] Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Springer, 2013.

[12] Rafael C Gonzalez, Richard Eugene Woods, and Steven L Eddins. *Digital Image Processing Using MATLAB*. Pearson Education India, 2004.

[13] J Hadamard. *Lectures on the Cauchy Problem in Linear Partial Differential Equations*. Yale University Press, New Howan, 1923.

[14] Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved optimization for the robust and accurate linear registration and motion correction of brain images. *Neuroimage*, 17(2):825–841, 2002.

[15] Seung-Jean Kim, Kwangmoo Koh, Michael Lustig, Stephen Boyd, and Dimitry Gorinevsky. An interior-point method for large-scale l1-regularized least squares. *Selected Topics in Signal Processing, IEEE Journal of*, 1(4):606–617, 2007.

[16] Jong-Sen Lee. Refined filtering of image noise using local statistics. *Computer Graphics and Image Processing*, 15(4):380–389, 1981.

[17] Martin A Lindquist. The statistical analysis of fMRI data. *Statistical Science*, 23(4):439–464, 2008.

[18] TR Oakes, T Johnstone, KS Ores Walsh, LL Greischar, AL Alexander, AS Fox, and RJ Davidson. Comparison of fMRI motion correction software tools. *Neuroimage*, 28(3):529–543, 2005.

[19] T Paggio and V Tome. Ill-posed problems and regularization analysis in early vision. Technical Report A.I Memo 773 C.B.I.P. Paper 001, Massachusetts Institute of Technology, AI Laboratory and Center for Biological Information Processing, Whitaker College, 1984.

[20] Russell A Poldrack, Jeanette A Mumford, and Thomas E Nichols. *Handbook of Functional MRI Data Analysis*. Cambridge University Press, 2011.

[21] David Poole. *Linear Algebra: A Modern Introduction*. Cengage Learning, 2014.

[22] William H Press. *Numerical Recipes 3rd edition: The Art of Scientific Computing*. Cambridge University Press, 2007.

[23] Jay West, J Michael Fitzpatrick, Matthew Y Wang, Benoit M Dawant, Calvin R Maurer Jr, Robert M Kessler, Robert J Maciunas, Christian Barillot, Didier Lemoine, André Collignon, et al. Comparison and evaluation of retrospective intermodality brain image registration techniques. *Journal of Computer Assisted Tomography*, 21(4):554–568, 1997.

[24] Wotao Yin and Yin Zhang. Extracting salient features from less data via l1-minimization. *SIAG/OPT Views-and-News*, 19(1):11–19, 2008.

Credit scoring using non-parametric Gaussian Mixture Models

Charl Arthur Henry Cowley 11073617

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr. C.M. van der Walt

Department of Statistics, University of Pretoria

November 2, 2015

Abstract

Credit scoring is used to discern good debtors from bad debtors. This paper investigates the accuracy of one-class and two-class Gaussian Mixture Model (GMM) classifiers. We also compare the accuracy of two-class GMM classifiers that make use of Bayes' rule to two-class GMM's that do not have Bayesian priors. The comparison is done by comparing the Area under the ROC curves (AUC) for GMM's with one to 20 mixtures over four different covariance structures (diagonal, spherical, tied and full) on the German Credit Data set. We also take note of the comparative accuracy between the parametric GMM (a GMM with one mixture) and the non-parametric GMM (a GMM with more than one mixture).

Declaration

I, *Charl Arthur Henry Cowley,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Charl Arthur Henry Cowley

_____
Dr. Christiaan van der Walt

_____
Date

Acknowledgements

Dr. Christiaan van der Walt, without whom I would not have achieved much success in understanding this topic.
My parents and sister, Henry, Marleen and Marlise Cowley, for being continuous sources of inspiration and motivation.
Nadia van Staden, my number one.

Contents

List of Figures

List of Tables

1 Introduction

Forecasting financial risk has become one of the seminal research topics in statistics. Credit scoring is the specific application of forecasting financial risk with regards to consumer lending. Credit scoring is important in so far as it attempts to predict the behavioural patterns of consumers and specifically whether or not affording debt to a customer will lead to a delinquency. A statistical credit scoring model is trained by applying statistical techniques to large amounts of historical debtor data. Training statistical models from data also makes credit scoring an ideal application of data mining techniques [3].

There typically exist two types of credit scoring models, namely application scoring and behavioural scoring. Application scoring attempts to assign a score to a new loan applicant based on historical data and existing credit records. Behavioural scoring attempts to determine the treatment of existing debtors by analysing the debtor's latest spending patterns [29].

Credit scoring is an important device for various types of institutions such as banks, insurance companies and government departments and may even extend to landlords and general retailers. These institutions use credit scores to set limits on the amount of debt afforded as well as determining the interest rate at which the debtor is most likely not to default [29]. Thereby allowing them to better manage risk and simultaneously maximise profits.The ultimate goal of credit scoring, however, is to group individuals together who show a high propensity to repay their debt and separate them from the individuals with a high probability of default (PD).

This requires lenders to use a classification approach in trying to model the PD. The standard approach to classification is to estimate the class-conditional probability density function (pdf) of every class in a data set and to a.) assign a new data sample to the class with the highest probability or to b.) find thresholds that distinguish the separate classes from each other. This approach is known as two-class classification. Thereby debtors below a certain threshold PD are to be classified positively and the rest that are above this threshold, which is determined by means of cross-validation, to be classified negatively.

There are three main approaches to one-class classification (OCC), namely density estimation, the boundary method and the reconstruction method [28]. One can use OCC methods to estimate a class-conditional pdf for only the positive data (the non-default data) and, consequently, decide whether a new sample should be added to this class based on its likelihood score. A common problem facing modellers is the imbalance in debtor data due to the low percentage of debtors that default on debt compared to those who do not. This is formally known as the low-default portfolio (LDP) problem. In the context of credit scoring, the accuracy of two-class classification methods decrease, due to the LDP problem. It has been shown by [18] that OCC obtains better results than two-class classification in the presence of data with a high imbalance ratio.

We aim to investigate whether the form of the underlying class-conditional pdf has an impact on the effectivity of the estimation of the PD if we use a non-parametric Gaussian Mixture Model (GMM) approach (two-class classification) as opposed to a standard Gaussian approach (OCC). We also aim to determine whether the use of Bayes' rule (where the class prior is the proportion of samples belonging to a class) combined with the standard approach to classification can lead to two-class classification methods obtaining better results than if no prior is used for four different covariance structures in the German Credit data set.

Chapter 2 presents a review of the LDP problem, the classical approach to Gaussian classification and GMM's. We also elaborate on some problems that arise when initialising the parameters of GMM's. Chapter 3 presents an introduction on the German Credit data set, how the proposed experiment is to be executed. Chapter 4 presents the consequent results. Finally, conclusions are given and discussed, along with recommendations for further research.

2 Background Theory

2.1 The LDP problem: prior research

There exist various examples in the literature of classification techniques and algorithms that have been adapted to build models specifically focused on credit scoring including, among others, [10], [2], [21] and [15]. Very few examples, where the LDP problem is assessed, focus on comparing the predictive performance of different methods of classification. These papers, such as [25] - who used a most prudent estimation principle - instead focused on whether the models suggested were valid for addressing the LDP problem.

Due to the prevalence of the LDP problem it is very difficult to choose a standard classification technique to use for credit scoring. Furthermore, no definitive benchmarking study on the performance of classification techniques is said to currently exist in the literature for a general data set. The issues surrounding the LDP problem are further exacerbated by the various conflicting findings that are being made on the subject. Baesens et al. [3], for example, found that for some specific data sets, the difference in the performance of complex state-of-the-art classifiers and traditional simpler classifiers not to be statistically significant. Brown and Mues [9] on the other hand, found that the traditional classifiers fared considerably worse than the modern classifiers in instances where the data was more unbalanced.

Kennedy et al. [18] and Lee and Cho [19] found that the LDP problem can best be addressed by only using OCC methods in cases of extreme data imbalance. That is, when less than 1% of debtor data indicates default.

Another potential solution to the LDP problem, is oversampling, which would be done to correct for bias in a data set. Kennedy et al. [18] duplicated the results of Bellotti and Crook [5] when they found that oversampling does not lead to an improvement of the performance of the best classifiers. They did, however, find that the adjustment of the threshold that separates two classes or serves as an upper bound for an OCC method, led to much improved results. The selection of an optimised threshold has been discussed at length [3], but the impact that it has on the predictive power of classifiers is strangely neglected.

It was also shown by Juszczak et al. [16] that OCC methods are much more consistent than two class classifiers in cases where population shift occur. This will typically occur when debtor behaviour changes over long time periods and also in volatile market segments, such as microcredit.

2.2 Alternative fields of application for classification methods

Classification methods have many and varied applications. An ensemble of classifiers can typically be used in the insurance industry to monitor and update the underwriting of life-insurance policies [8]. Pattern recognition can be used to identify fraudulent behaviour and has various modern applications of which facial recognition on social media platforms is one of many growing research topics [27]. Classification is also encountered in daily interactions such as whether a new e-mail ought to be labelled as spam and also in filtering the relevance of internet search results. Classification can also perform highly specialised scientific tasks, including the clustering of genes in biology and the classification of brain tissue in MRI scans [30].

Gaussian Mixture Models, in particular, have been applied to classifying skin colour by means of the EM algorithm (similarly to this paper) [33] and in speech recognition by using Hidden Markov Models (HMM's) [11].

2.3 The classical Gaussian approach to classification

The classical Gaussian approach to classification is a density estimation method which makes use of the fact that the underlying data is generated from a unimodal d-dimensional multivariate normal/ Gaussian

distribution.

$$f(y) \quad = \quad (2\pi)^{-\frac{d}{2}}|\Sigma|^{-\frac{1}{2}}\exp\left\{-\frac{1}{2}(y-\mu)^T\Sigma^{-1}(y-\mu)\right\}, -\infty < y < \infty$$

The classification of an object z is dependent on a certain threshold valueθ. The Mahalanobis distance of z to the training data set distribution,

$$f(z) \quad = \quad (z-\mu)^T\Sigma^{-1}(z-\mu)$$

where μ is the mean and Σ the covariance matrix of the training data set (obtained through an EM approach), is compared to θ. A classification decision is made based on this comparison. Despite being one of the simplest OCC methods available, the performance of this method is severely lacking when the normality assumption is not met, since it introduces a large bias [28].

2.4 Gaussian Mixture Models

GMM's are used primarily for density estimation. A GMM is a density function represented as the weighted sum of a number of individual Gaussian densities. GMM's are, consequently, highly effective for representing multi-modal data. The number of individual Gaussians are chosen by looking at a graphical representation of your data and estimating the number of clusters of data. The density function of a GMM that consists of n i.i.d. Gaussian densities, $\phi(.)$, is given by

$$f(\mathbf{x}) \quad = \quad \sum_{i=1}^{n}\sum_{j=1}^{k}\omega_j\phi(y_i|\mu_j,\Sigma_j)$$

where ω_i, μ_i and Σ_i are the mixing coefficient, mean and covariance of each individual Gaussian density, respectively [7]. It is important to note that a GMM with only one mixture reduces to a unimodal d-dimensional multivariate normal/ Gaussian distribution and is also known as a parametric GMM, while a GMM with more than one mixture is known as a non-parametric GMM. The parameters of a GMM are estimated by training data with a number of algorithms. One must, however, take note of the fact that GMM's require a large amounts of data for effective and efficient training [28]. Quite often, initial values are obtained using K-means and K-means++ clustering.

2.5 Initial values with K-means

Finding optimal initial values for a GMM is an age-old chicken-egg conundrum that has been discussed in the literature by[22], [20], [23] and [6]. GMM's are normally initialised with K-means clustering since K-means and GMM are both unsupervised classification methods. K-means can be considered a special case of the Expectation Maximisation (EM) algorithm since it proceeds in two steps which correspond to the E and M steps of the EM algorithm. K-means can easily be optimised by making use of Lloyd's algorithm [17] and despite having been developed in 1957, it is still being widely used in practice today due to it achieving EM accuracy of up to 99.91% and delivering computational performance comparable to modern techniques at the fraction of the cost [26].

2.6 The EM algorithm and Gaussian Mixture Models

Once we have obtained initial parameter values, say $\omega^{(0)}$, $\mu^{(0)}$ and $\Sigma^{(0)}$ we calculate the initial log-likelihood $L^{(0)} = \sum_{i=1}^{n}\log\left(\sum_{j=1}^{k}\omega_j^{(0)}\phi(y_i|\mu_j^{(0)},\Sigma_j^{(0)})\right)$. The EM algorithm can be used iteratively to train the GMM to

the data by maximising the log-likelihood functions of the pdf of the GMM with respect to each parameter. Where after the initial value of the log likelihood is evaluated. In the Expectation (E) step of EM, the current parameter values are used to determine the posterior probability of each individual data point. That is, we calculate $\gamma_{ij}^{(m)} = \frac{\omega_j^{(m)}\phi(y_i|\mu_j^{(m)},\Sigma_j^{(m)})}{\sum_{l=1}^{k}\omega_l^{(m)}\phi(y_i|\mu_l^{(m)},\Sigma_l^{(m)})}$ and $n_j^{(m)} = \sum_{i=1}^{n}\gamma_{ij}^{(m)}$. In other words, we evaluate the responsibilty that each individual Gaussian assumes to explain that data point. In the Maximisation (M) step, these probabilities are used in updating formulae to give better estimates for the parameters. The updating formulae for the parameters are

$$\begin{aligned}
\omega_j^{(m+1)} &= \frac{n_j^{(m)}}{\sum_{j=1}^{k} n_j^{(m)}} = \frac{n_j^{(m)}}{n} \\
\mu_j^{(m+1)} &= \frac{1}{n_j^{(m)}}\sum_{i=1}^{n}\gamma_{ij}^{(m)} y_i \\
\Sigma_j^{(m+1)} &= \frac{1}{n_j^{(m)}}\sum_{i=1}^{n}\gamma_{ij}^{(m)}(y_i - \mu_j^{(m+1)})(y_i - \mu_j^{(m+1)})^T
\end{aligned}$$

After each iteration, the log-likelihood is again evaluated, $L^{(m)} = \sum_{i=1}^{n}\log\left(\sum_{j=1}^{k}\omega_j^{(m)}\phi(y_i|\mu_j^{(m)},\Sigma_j^{(m)})\right)$ Once the difference between the log-likelihood of two iterations are below a sufficient threshold, $|L^{(m+1)} - L^{(m)}| < \delta$, for some δ, the algorithm is said to have converged and is ended [7]. (See Appendix A for a proof of how these updating formulae are obtained) EM does, however, only converge to local optima and has been shown to be less effective when data is multi-modal [31]. Research has been conducted by [1], [30] and [31] exploring other methods that estimate the true values of the parameters for such higher dimensional cases. For our purposes, however, the EM algorithm will be sufficient for training the GMM to the data set.

2.7 Bayes' Rule

Given the events A and B, Bayes' theorem/ rule can be stated as

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)}$$

This means that the conditional probability of an event A occuring, given that another event B is true, can be calculated from prior knowledge of the probability of each event occurring (A and B) and the conditional probability that the event B occurs, given that A is true [4]. We know that the German Credit data set consists of 700 positive observations and 300 negative observations. We hope that by incorporating these weights of 0.7 and 0.3 into a two-class GMM classifier we will be able to show a significant improvement in the classification performance thereof.

3 Application

3.1 Description of German Credit data set

We will be using the German Credit data set[1]. There currently exist two formats of this dataset. An original data set - containing only categorical/ symbolic attributes - that has been available for use since 1994.

[1] The data set is obtained from: https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29

Strathclyde University adapted the original data set for algorithms that require numerical attributes. We will be using this altered data set. This set consists of qualitative (credit history, purpose of debt application, present employment status, gender, properties owned and housing status) as well as the new numerical attributes (credit amount, instalment rate as percentage of income, age in years, number of dependants and number of existing credit) which include various indicator variables. These indicator variables have been changed from categorical variables to integer values in order to facilitate its use in the EM algorithm, since EM requires numerical values. An important aspect of this data set is that it requires the use of a specific confusion matrix. In a confusion matrix, the ij'th element is the penalty for classifying an instance of class j as class i. In the credit scoring context, it is the cost incurred by a lender for wrongly classifying a solvent debtor as a possible debtor and vice versa.

3.2 Previous papers on the German Credit data set

The German Credit data set is popular with experiments that aim to showcase how a particular classification method can be refined. Eggermont et al.[12], for example, used the decision tree construction algorithm C4.5 to refine and reduce the sizes of search spaces. O' Dea et al. [24] presented a possible solution to classification problems through a combination of feature selection and neural networks and Ekin et al. [13] compared classic distance-based classification methods (such as K-means clustering) to other modern methods. Interestingly, they found that distance-based methods performed on similar and higher levels than other classification methods and were much easier to implement. Most significantly, though, they found that distance-based classifying methods were particularly robust. Wang et al. [32] used the German Credit data set to test a proposed method by which changes in behavioural patterns could be detected. An old classifier representing historical knowledge was compared to a new classifier that kept track of the latest behaviour. All these papers utilised the German Credit data set to compare and refine their chosen classification method.

3.3 Description of Experimental Design

The experiments proposed will be implemented in the *Scientific PYthon Development EnviRonment*, or *Spyder*, due to its user-friendly interface, powerful interactive development environment (IDE) for the *Python* language and its advanced editing, testing, debugging and introspection abilities.

As stated previously, we want to determine whether a two-class GMM classifier, can perform better than a one-class GMM classifier over different class imbalances and also compare the classification accuracy of two-class GMM's that utilise a Bayesian prior with two-class GMM's that do not use a Bayesian prior. In order to achieve this comparison, we will perform two experiments.

3.3.1 Initialisation of GMM

As discussed above, the weight, mean and covariance parameters of a GMM are regularly initialised by making use of k-means and k-means++ clustering. For simplicity sake we will be making use of random initialisations of the parameters. The Python scientific package, *sklearn*, which provides relevant tools for machine learning, data mining and data analysis, contains a module called *mixture.GMM*. This module provides an argument for the number of initialisations that should be done, before starting EM for the GMM. We will be using ten initialisations.

3.3.2 Experiment 1: Comparison of one and two-class GMM classifiers

Firstly, we will compare one-class and two-class GMM classifiers over different covariance structures for one to 20 mixtures. The module will be used to repeatedly fit the relevant GMM's with the different amount of mixtures and also requires an argument that specifies the type of covariance structure to use. The options are diagonal, spherical, tied and full covariances.

- Diagonal covariance structure - assigns a diagonal matrix to each component of the GMM. This means that the predictors in the model are uncorrelated.
- Spherical covariance structure - this is the most parsimonious covariance specification, as it reduces the entire covariance structure of a component to one single variable parameter.
- Tied covariance structure - assigns identical covariance structures to all components of the GMM.
- Full covariance structure - assigns a full matrix for each component to each component of the GMM and allow for correlated predictors. This specification can lead to overfitting of a model.

3.3.3 Experiment 2: Comparison of two-class GMM classifiers with and without Bayesian priors

The second experiment will compare the performance of a two-class GMM classifier with a Bayesian prior versus a two-class GMM classifier without a Bayesian prior. In *Python*, we will provide a condition in the code by means of a boolean operator which will execute two-class classification with the Bayesian prior if the operator is set to "True" and will execute two-class classification without the Bayesian prior if the operator is set to "False". Thereby, we will answer the question as to whether the Bayesian prior has an impact on two-class classifier performance.

3.4 Methods of comparison

We have mentioned the concept of classifier performance and comparison at length and will now elaborate on how this is done by means of Receiver Operating Characteristic (ROC) curves and the area under the ROC curve (AUROC or AUC).

3.4.1 ROC curves and AUC

ROC is a metric that is used to evaluate the quality of classifier output and is typically used in analysing dichotomous classification, such as credit scoring. A ROC curve is a graphical representation of the ROC of a classifier for different threshold values and is of particular use where algorithms, such as EM for GMM's, are evaluated. This means that a ROC curve will be an ideal measure to evaluate the output of our experiments defined previously.

In dichotomous classification, there are four possible outcomes, namely true positive (TP), false positive (FP), false negative (FN) and true negative (TN). A true positive classification occurs when an element of the positive class (in an OCC example) is correctly classified in the positive class. The TPR (also known as sensitivity or recall, as it is referred to in machine learning nomenclature) is then the ratio of the positive class elements correctly classified to the total number of positive elements. A false positive classification occurs when an element of the negative class is incorrectly classified in the positive class. The FPR (or false alarm rate) is then the ratio of the negative class elements incorrectly classified to the total number of negative elements. ROC curves typically feature the true positive rate (TPR) of classification on the Y axis and the false positive rate (FPR) of classification on the X axis .

Figure(1) shows a summary of the important measures obtained from a ROC curve by means of a confusion matrix.

The optimal point of classification is where the TPR is equal to 1 and the FPR is equal to 0. While this is not a very realistic outcome in practice, the severity of the slope of the ROC curve is a very good indication of classifier performance, since a steep ROC curve indicates a greater TPR and a lower FPR.

As we have seen, the ROC curve contains many metrics to evaluate classifier output quality, but in order to compare classifiers it would be desirable to reduce ROC performance from a two-dimensional portrayal of classifier performance into a single numerical value that describes the expected performance. A common

		True class	
		Positive	Negative
Hypothesised class	Yes	True positives (TP)	False positives (FP)
	No	False negatives (FN)	True negatives (TN)
	Col totals	P	N

$$
\begin{aligned}
TPR &= \frac{TP}{P} \\
FPR &= \frac{FP}{N} \\
specificity &= 1 - FPR \\
precision &= \frac{TP}{TP + FP} \\
&= (positive - predictive - value) \\
accuracy &= \frac{TP + TN}{P + N} \\
F - measure &= \frac{2}{\frac{1}{precision} - \frac{1}{recall}}
\end{aligned}
$$

Figure 1: Confusion matrix/ contingency table and metrics used with ROC curves

scalar value which will provide this information, is the area under the ROC curve (AUC). AUC is a useful and easy value to obtain and possesses a vital statistical property in that the AUC of a classifier is approximately equal to the probability that the classifier will rank a random positive element higher than a corresponding negative element[14].

For our experiments we will calculate the AUC of the one-class GMM classifiers and two-class GMM classifiers for all the mixtures over each covariance structure.

Since FPR and TPR return values between 0 and 1, AUC will also be a scalar value between 0 and 1. An AUC of 1 represents a perfect test, while 0.5 indicates a worthless test. A general guide to classifying the accuracy of a classifier, is the conventional academic point system:

- $0.9 < AUC \leq 1$: Excellent classifier
- $0.8 < AUC \leq 0.9$: Good classifier
- $0.7 < AUC \leq 0.8$: Moderate classifier
- $0.6 < AUC \leq 0.7$: Poor classifier
- $0.5 < AUC < 0.6$: Fail, very bad classifier, it would not make practical sense to use this classifier.

4 Results

4.1 Experiment 1: Comparison of one and two-class GMM classifiers

4.1.1 Diagonal covariance

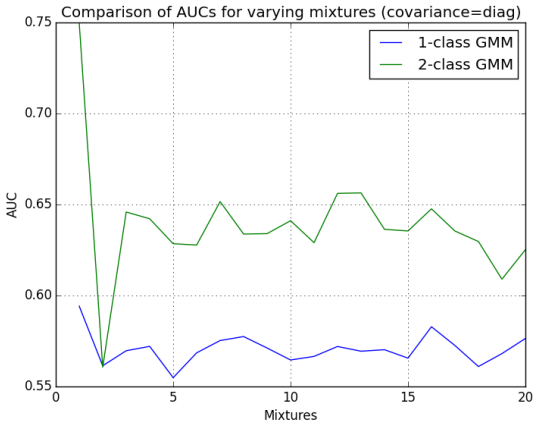


Figure 2: Comparison of AUC's for varying mixtures with Diagonal covariance structure

The two-class GMM classifier has very volatile performance when a diagonal covariance structure is assigned. While it outperforms the one-class GMM classifier for almost all mixtures, its overall performance is mediocre. We do, however, observe that a two-class GMM classifier with one mixture and a diagonal covariance structure achieves classifier performance in excess of 0.75 and can conclude that it performs admirably. It is important to note that a two-class GMM classifier with one mixture, a diagonal covariance structure and a Bayesian prior, is equivalent to a Naive Bayes classifier. The improved performance of this classifier echoes the findings of Kennedy et al. [18] that the Parzen classifier, which is similar to the Naive Bayesian classifier and is hence an extension of a GMM classifier, achieves higher harmonic mean performance than the GMM classifier.

4.1.2 Spherical covariance

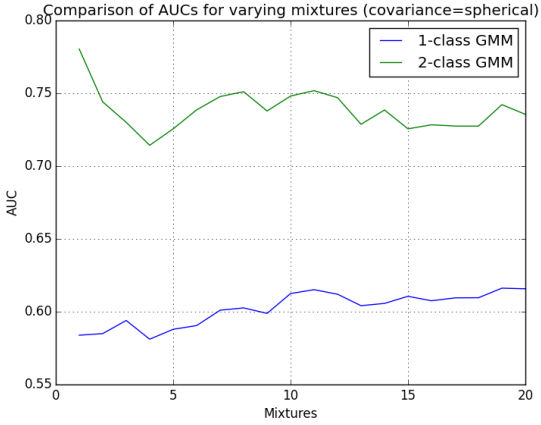


Figure 3: Comparison of AUC's for varying mixtures with Spherical covariance structure

It is again evident that the two-class GMM classifier outperforms the one-class GMM classifier and that it performs consistently in a moderate area of classification accuracy. It can additionally be observed that the two-class GMM classifier has much less volatile performance when a spherical covariance structure is assigned rather than a diagonal structure covariance structure. The performance of the one-class GMM is again abysmal to poor, although a slight improvement is evident as the number of mixtures increase.

4.1.3 Tied covariance

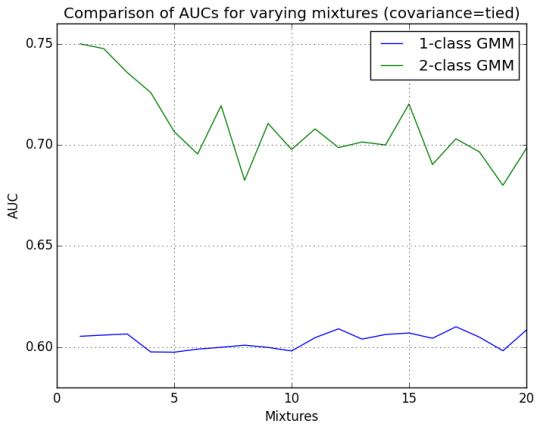


Figure 4: Comparison of AUC's for varying mixtures with Tied covariance structure

A similar pattern to the previous covariance structures follow for Tied covariances. We also see a steady decrease in two-class GMM classifier performance as the number of mixtures increase.

4.1.4 Full covariance

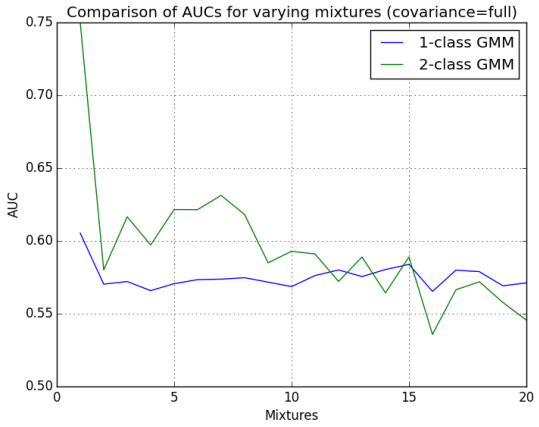


Figure 5: Comparison of AUC's for varying mixtures with Full covariance structure

Figure 5 clearly shows that two-class GMM performance deteriorates profoundly as the number of mixtures increase. We also observe one extreme classifier performance in excess of 0.75 for a two-class GMM with one mixture. A two-class GMM classifier with one mixture and a full covariance structure is equivalent to a simple Gaussian classifier. Kennedy et al. [18] also found that the simple Gaussian classifier achieves better results than GMM's with more mixtures.

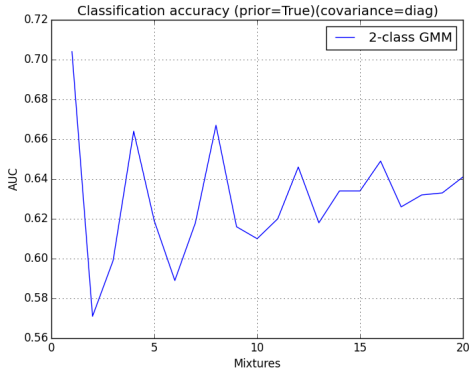
4.1.5 Summary of optimal AUC's

Covariance structure	One-class	Two-class
Diagonal	0.595	0.75
Spherical	0.62	0.78
Tied	0.61	0.75
Full	0.605	0.76

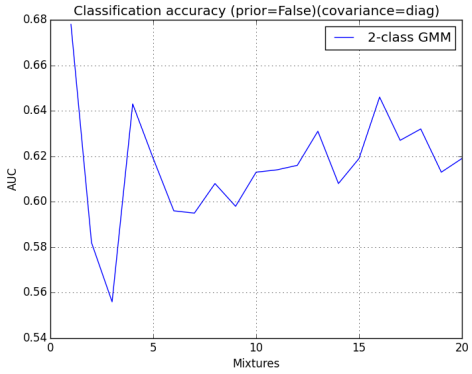
Table 1: Optimal AUC of one-class and two-class GMM classifiers for each covariance structure

4.2 Experiment 2: Comparison of two-class GMM classifiers with and without Bayesian priors

4.2.1 Diagonal covariance



(I)

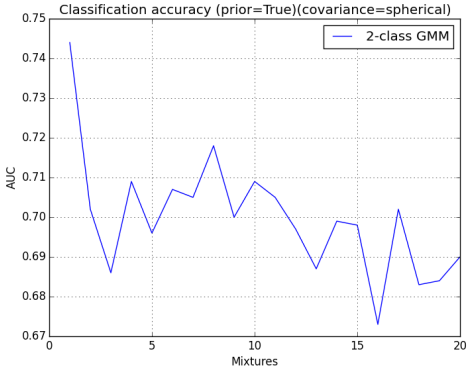


(II)

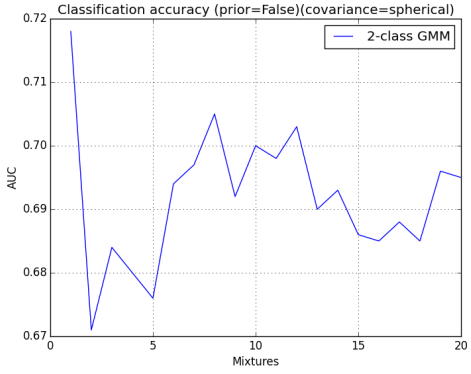
Figure 6: Classification accuracy of two-class GMM classifiers with (Prior=True) and without (Prior=False) Bayesian priors with Diagonal covariance structure

We observe a volatile similar pattern between the two classifiers in Figure 6 where the performance of the classifier with the Bayesian prior (I) is at a slightly higher level than that of the classifier without a prior (II) across all mixtures.

4.2.2 Spherical covariance



(I)

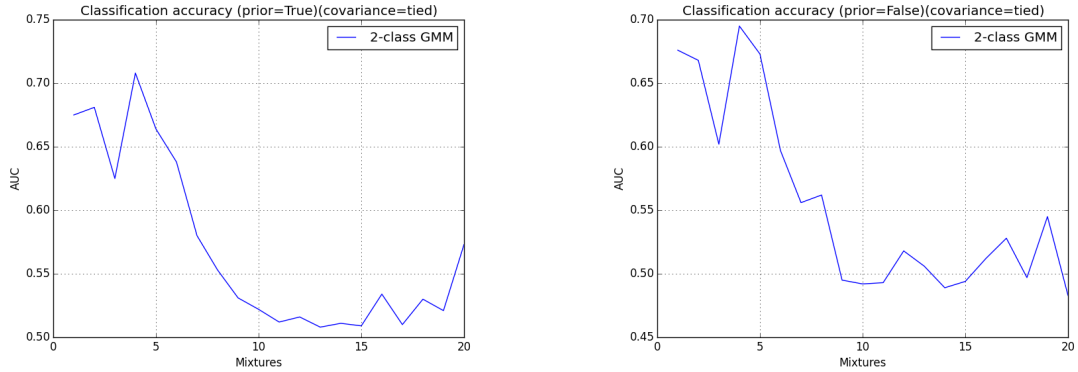


(II)

Figure 7: Classification accuracy of two-class GMM classifiers with (Prior=True) and without (Prior=False) Bayesian priors with Spherical covariance structure

Again we observe an effect where the pattern of performance for the two classifiers are similar, but the best performance of the classifier with the Bayesian prior (I) is approximately 0.02 points higher than the classifier without the Bayesian prior(II). The performance seen in (II) is somewhat less volatile than (I).

4.2.3 Tied covariance

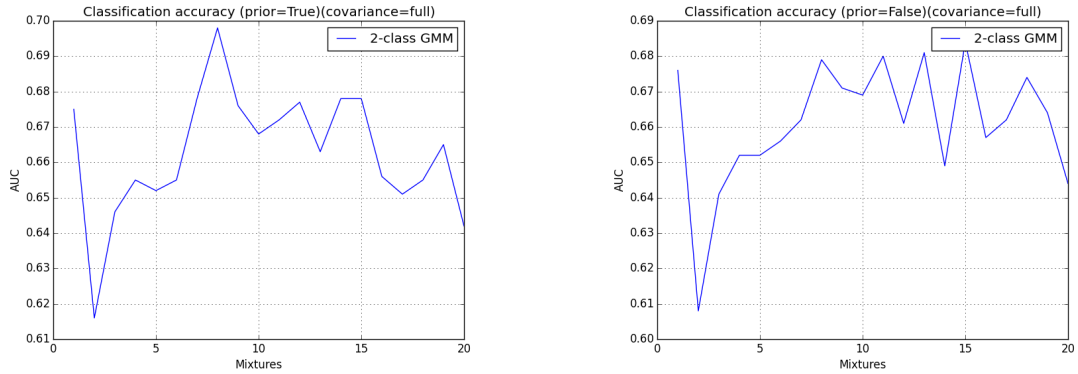


(I) (II)

Figure 8: Classification accuracy of two-class GMM classifiers with (Prior=True) and without (Prior=False) Bayesian priors with Tied covariance structure

The same level effect is again observed for tied covariance structures. We do, however, also observe a severe deterioration in classifier performance as the number of mixtures increase.

4.2.4 Full covariance



(I) (II)

Figure 9: Classification accuracy of two-class GMM classifiers with (Prior=True) and without (Prior=False) Bayesian priors with Full covariance structure

For a full covariance structure, the performance of two-class GMM classifiers is at a consistently higher level than for all the other covariance structures and the classifier performance does not deteriorate as the number of mixtures increase.

4.2.5 Summary of optimal AUC's for two-class GMM classifiers

Covariance structure	Prior	No Prior
Diagonal	0.708	0.679
Spherical	0.745	0.718
Tied	0.705	0.698
Full	0.695	0.685

Table 2: Optimal accuracy (AUC) of two-class GMM classifiers with and without Bayesian priors for each covariance structure

5 Conclusion

We wanted to know whether two-class GMM classifiers can perform better than one-class GMM classifiers. It was confirmed that for almost all number of mixtures across the four covariance structures studied, the two-class GMM classifier performed much better than the one-class GMM classifier. The best two-class GMM classifiers were equivalent to a Naive Bayes classifier and a simple Gaussian classifiers (both of which are parametric GMM's). The best covariance structure for consistent classifier performance, was the spherical covariance structure, which is by far the most parsimonious. In a credit scoring context, this is a useful finding, since the computational effort required in practice will always need to be minimised. It is also worth noting that the covariance structure which delivered the worst classifier performance, was the tied covariance structure. This seems to make intuitive sense in a credit scoring context, since no two loan applicants will have identical attributes and that to assume equal variances negates this individuality of applicants. The full covariance structure delivered very volatile results, which again confirms that allowing for correlation in data leads to results that are difficult to interpret.

We also wanted to know whether the introduction of a Bayesian prior can improve the performance of a two-class GMM classifier. We observed a very small improvement of approximately 0.02 points in the performance of the classifer when the prior was added over all mixtures for all the covariance structures studied. In a credit scoring context, the implementation of a Bayesian prior will mean that two more applicants out of every 100 will be correctly afforded credit.While this might seem relatively insignificant, the financial and economic implications of two more correct classifications could be immense.

The performance of the non-parametric GMM classifiers are, however, not very accurate in classifying the German credit data set. The performance of the non-parametric GMM classifiers can be improved in further studies by initialising the parameters with k-means clustering (as discussed previously), increasing the number of EM iterations (Python defaults to 100 iterations) or selecting appropriate parameters by tuning the GMM through information criteria comparison. Two prevalent information criteria are Bayes Information Criteria and Akaike's Information Criteria. Lower AIC and BIC values indicate models that are a better fit for the data. The literature suggests that with greater amounts of data, GMM classifiers can be trained more effectively, but unless the data is normally distributed, the performance of the GMM classifier is unlikely to improve.

The underlying distribution of the German credit data set presents a final interesting finding in so far as the improved classifier performance of parametric GMM's (GMM's with one mixture) in comparison with non-parametric GMM's (GMM's with more than one mixture) is concerned. This would suggest that a single Gaussian density fits the data sufficiently well. It can be shown relatively easily whether the data is normally distributed, and the good performance of the parametric GMM classifiers in this paper suggest that the data might be distributed as such.

References

[1] A Anandkumar, R Ge, D Hsu, SM Kakade, and M Telgarsky. Tensor decompositions for learning latent variable models. *The Journal of Machine Learning Research*, 15(1):2773–2832, 2014.

[2] B Baesens, M Egmont-Petersen, R Castelo, and J Vanthienen. Learning Bayesian network classifiers for credit scoring using Markov chain Monte Carlo search. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on Pattern Recognition*, volume 3, pages 49–52. IEEE, 2002.

[3] B Baesens, T Van Gestel, S Viaene, M Stepanova, J Suykens, and J Vanthienen. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, 54(6):627–635, 2003.

[4] LJ Bain and M Engelhardt. *Introduction to Probability and Mathematical Statistics*. Brooks/Cole, 1987.

[5] T Bellotti and J Crook. Credit scoring with macroeconomic variables using survival analysis. *Journal of the Operational Research Society*, 60(12):1699–1707, 2009.

[6] C Biernacki, G Celeux, and G Govaert. Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian Mixture Models. *Computational Statistics & Data Analysis*, 41(3):561–575, 2003.

[7] CM Bishop. *Pattern Recognition and Machine Learning*, volume 4. Springer New York, 2006.

[8] P Bonissone, N Eklund, and K Goebel. Using an ensemble of classifiers to audit a production classifier. In *Multiple Classifier Systems*, pages 376–386. Springer, 2005.

[9] I Brown and C Mues. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, 39(3):3446–3453, 2012.

[10] VS Desai, JN Crook, and GA Overstreet. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95(1):24–37, 1996.

[11] VV Digalakis, D Rtischev, and LG Neumeyer. Speaker adaptation using constrained estimation of Gaussian mixtures. *Speech and Audio Processing, IEEE Transactions on*, 3(5):357–366, 1995.

[12] J Eggermont, JN Kok, and WA Kosters. Genetic programming for data classification: Partitioning the search space. In *Proceedings of the 2004 ACM Symposium on Applied Computing*, pages 1001–1005. ACM, 2004.

[13] O Ekin, PL Hammer, A Kogan, and P Winter. Distance-based classification methods. *Information Systems and Operational Research*, 37:337–352, 1999.

[14] T Fawcett. ROC graphs: Notes and practical considerations for researchers. *Machine Learning*, 31:1–38, 2004.

[15] C Huang, M Chen, and C Wang. Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*, 33(4):847–856, 2007.

[16] P Juszczak, NM Adams, DJ Hand, C Whitrow, and DJ Weston. Off-the-peg and bespoke classifiers for fraud detection. *Computational Statistics & Data Analysis*, 52(9):4521–4532, 2008.

[17] T Kanungo, DM Mount, NS Netanyahu, CD Piatko, R Silverman, and AY Wu. An efficient k-means clustering algorithm: Analysis and implementation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):881–892, 2002.

[18] K Kennedy, B MacNamee, and SJ Delany. Using semi-supervised classifiers for credit scoring. *Journal of the Operational Research Society*, 64(4):513–529, 2012.

[19] H Lee and S Cho. Focusing on non-respondents: Response modeling with novelty detectors. *Expert Systems with Applications*, 33(2):522–530, 2007.

[20] R Maitra. Initializing partition-optimization algorithms. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 6(1):144–157, 2009.

[21] D Martens, B Baesens, T Van Gestel, and J Vanthienen. Comprehensible credit scoring models using rule extraction from support vector machines. *European Journal of Operational Research*, 183(3):1466–1476, 2007.

[22] M Meila and D Heckerman. An experimental comparison of several clustering and initialization methods. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 386–395. Morgan Kaufmann Publishers Inc., 1998.

[23] V Melnykov and I Melnykov. Initializing the EM algorithm in Gaussian mixture models with an unknown number of components. *Computational Statistics & Data Analysis*, 56(6):1381–1395, 2012.

[24] P ODea, J Griffith, and C ORiordan. Combining feature selection and neural networks for solving classification problems. In *Proceedings of 12th Irish Conference on Artificial Intelligence and Cognitive Science*, pages 157–166. Citeseer, 2001.

[25] K Pluto and D Tasche. Estimating probabilities of default for low default portfolios. In *The Basel II Risk Parameters*, pages 75–101. Springer, 2011.

[26] G Singh, A Panda, S Bhattacharyya, and T Srikanthan. Vector quantization techniques for GMM based speaker verification. In *Acoustics, Speech, and Signal Processing, 2003 IEEE International Conference on*, volume 2, pages II–65. IEEE, 2003.

[27] Z Stone, T Zickler, and T Darrell. Toward large-scale face recognition using social network context. *Proceedings of the IEEE*, 98(8):1408–1415, 2010.

[28] DMJ Tax. *One-class classification*. PhD thesis, TU Delft, Delft University of Technology, 2001.

[29] LC Thomas, DB Edelman, and JN Crook. *Credit Scoring and its Applications*. SIAM, 2002.

[30] J Tohka, E Krestyannikov, I Dinov, D Shattuck, U Ruotsalainen, and A Toga. Genetic algorithms for finite mixture model based tissue classification in brain MRI. In *Proceedings of European Medical and Biological Engineering Conference*, volume 11, pages 4077–4082, 2005.

[31] N Vlassis and A Likas. A greedy EM algorithm for Gaussian mixture learning. *Neural Processing Letters*, 15(1):77–87, 2002.

[32] K Wang, S Zhou, AW-C Fu, and JX Yu. Mining Changes of Classification by Correspondence Tracing. In *Proceedings of the Third SIAM International Conference on Data Mining*, pages 95–106. SIAM, 2003.

[33] M Yang and N Ahuja. Gaussian Mixture Models for human skin color and its applications in image and video databases. In *Proceedings of SPIE: Storage and Retrieval for Image and Video Databases VII*, pages 458–466. International Society for Optics and Photonics, 1998.

Appendices

Appendix A: The EM algorithm for Gaussian Mixture Models

Before we start, we derive a useful result that will be used later

Result 1

Let the complete data X consist of n i.i.d. samples $X_1, ..., X_n$such that $p(x|\theta) = \prod_{i=1}^{n} p(x_i|\theta)$ for all $x \epsilon \mathcal{X}$ for all . Also, let $y_i = T(x_i), i = 1, ..., n$ then

$$Q(\theta|\theta^{(m)}) = \sum_{i=1}^{n} Q_i(\theta|\theta^{(m)})$$

where

$$Q_i(\theta|\theta^{(m)}) = E_{X_i|y_i,\theta^{(m)}}[\log p(X_i|\theta)], i = 1, ..., n$$

Since

$$\begin{aligned} Q(\theta|\theta^{(m)}) &= E_{X|y,\theta^{(m)}}[\log p(X|\theta)] \\ &= E_{X|y,\theta^{(m)}}[\log \left(\prod_{i=1}^{n} p(X_i|\theta) \right)] \\ &= E_{X|y,\theta^{(m)}}[\sum_{i=1}^{n} \log p(X_i|\theta)] \\ &= \sum_{i=1}^{n} E_{X_i|y,\theta^{(m)}}[\sum_{i=1}^{n} \log p(X_i|\theta)] \\ &= \sum_{i=1}^{n} E_{X_i|y_i,\theta^{(m)}}[\sum_{i=1}^{n} \log p(X_i|\theta)] \end{aligned}$$

because $p(x_i|y, \theta^{(m)}) = p(x_i|y_i, \theta^{(m)})$ due to the assumption of i.i.d samples and since $y_i = T(x_i), i = 1, ..., n$ and Bayes' Rule. □

Now, we proceed to prove EM for GMM's:

Given n i.i.d samples $y_1, ..., y_n \epsilon \mathbb{R}^d$from a Gaussian Mixture Model (GMM) with k components, consider the problem of estimating the set of its parameters $\theta = \{\omega_j, \mu_j, \Sigma_j; j = 1, ..., k\}$.Let the density of each Gaussian be

$$\phi(y|\mu, \Sigma) \triangleq (2\pi)^{-\frac{d}{2}} |\Sigma|^{-\frac{1}{2}} \exp\{-\frac{1}{2}(y - \mu)^T \Sigma^{-1} (y - \mu)\}$$

and define $\gamma_{ij}^{(m)}$to be your "guesstimate"of the probability that the i'th sample belongs to the j'th Gaussian component at the m'th iteration, that is,

$$\begin{aligned} \gamma_{ij}^{(m)} &\triangleq P(Z_i = j|Y_i = y_i, \theta^{(m)}) \\ \gamma_{ij}^{(m)} &= \frac{\omega_j^{(m)} \phi(y_i|\mu_j^{(m)}, \Sigma_j^{(m)})}{\sum_{l=1}^{k} \omega_l^{(m)} \phi(y_i|\mu_l^{(m)}, \Sigma_l^{(m)})} \end{aligned}$$

such that $\sum_{j=1}^{k} \gamma_{ij}^{(m)} = 1$.

Firstly, we have that

$$\begin{aligned} Q_i(\theta|\theta^{(m)}) &= E_{Z_i|y_i,\theta^{(m)}}[\log p_X(y_i, Z_i|\theta)] \\ &= \sum_{j=1}^{k} \gamma_{ij}^{(m)} \log p_X(y_i, j|\theta) \\ &= \sum_{j=1}^{k} \gamma_{ij}^{(m)} \log \omega_j \phi(y_i|\mu_j, \Sigma_j) \\ &= \sum_{j=1}^{k} \gamma_{ij}^{(m)} \left(\log \omega_j - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2}(y_i - \mu_j)^T \Sigma_j^{-1} (y_i - \mu_j) \right) + constant \end{aligned}$$

From Result 1and by dropping the constant term, it then follows that

$$Q(\theta|\theta^{(m)}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \gamma_{ij}^{(m)} \left(\log \omega_j - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2}(y_i - \mu_j)^T \Sigma_j^{-1} (y_i - \mu_j) \right)$$

This completes the Expectation (E) step of EM.

Now, let $n_j^{(m)} = \sum_{i=1}^{n} \gamma_{ij}^{(m)}$, which means that we can rewrite $Q(\theta|\theta^{(m)})$ as

$$Q(\theta|\theta^{(m)}) = \sum_{j=1}^{k} n_j^{(m)} \left(\log \omega_j - \frac{1}{2} \log |\Sigma_j| \right) - \frac{1}{2} \left(\sum_{i=1}^{n} \sum_{j=1}^{k} \gamma_{ij}^{(m)} (y_i - \mu_j)^T \Sigma_j^{-1} (y_i - \mu_j) \right)$$

To complete the Maximisation (M) step, we need to maximise $Q(\theta|\theta^{(m)})$ over all values of θ, subject to the constraints of $\sum_{j=1}^{k} \omega_j = 1, \omega_j \geq 0, j = 1, ..., k$ and $\Sigma_j \succ 0, j = 1, ..., k$ (which means that Σ_j is positive definite). Rather than maximising the log-likelihood function

$$L(\theta) = \sum_{i=1}^{n} \log \left(\sum_{j=1}^{k} \omega_j \phi(y_i|\mu_j, \Sigma_j) \right)$$

we form a Lagrange multiplier, to solve for the optimal weights, that is ω_j.

$$J(\omega, \lambda) = \sum_{j=1}^{k} n_j^{(m)} (\log \omega_j) + \lambda \left(\sum_{j=1}^{k} \omega_j - 1 \right)$$

The optimal weights satisfy

$$\frac{\partial J}{\partial \omega_j} = \frac{n_j^{(m)}}{n} + \lambda = 0, j = 1, ..., k$$

Combining the above with the constraint of $\sum_{j=1}^{k} \omega_j = 1$ it follows that

$$\omega_j^{(m+1)} = \frac{n_j^{(m)}}{\sum_{j=1}^{k} n_j^{(m)}} = \frac{n_j^{(m)}}{n}$$

To solve for the mean vectors, that is, μ_j, we let

$$\frac{\partial Q(\theta|\theta^{(m)})}{\partial \mu_j} = \Sigma_j^{-1}\left(\sum_{i=1}^{n}\gamma_{ij}^{(m)}y_i - n_j^{(m)}\mu_j\right) = 0$$

which gives

$$\mu_j^{(m+1)} = \frac{1}{n_j^{(m)}}\sum_{i=1}^{n}\gamma_{ij}^{(m)}y_i$$

To solve for the covariance matrices, that is, Σ_j, we let

$$\begin{aligned}\frac{\partial Q(\theta|\theta^{(m)})}{\partial \Sigma_j} &= -\frac{1}{2}n_j^{(m)}\frac{\partial}{\partial \Sigma_j}\log|\Sigma_j| - \frac{1}{2}\sum_{i=1}^{n}\gamma_{ij}^{(m)}\frac{\partial}{\partial \Sigma_j}(y_i-\mu_j)^T\Sigma_j^{-1}(y_i-\mu_j)\\ &= -\frac{1}{2}n_j^{(m)}\Sigma_j^{-1} + \frac{1}{2}\sum_{i=1}^{n}\gamma_{ij}^{(m)}\Sigma_j^{-1}(y_i-\mu_j)(y_i-\mu_j)^T\Sigma_j^{-1}\\ &= 0,\ j=1,...,k\end{aligned}$$

which gives

$$\Sigma_j^{(m+1)} = \frac{1}{n_j^{(m)}}\sum_{i=1}^{n}\gamma_{ij}^{(m)}(y_i-\mu_j^{(m+1)})(y_i-\mu_j^{(m+1)})^T$$

Appendix B: Code

```
# -*- coding: utf-8 -*-
"""

"""
import numpy as np
from sklearn.cross_validation import StratifiedKFold
from sklearn import mixture
from sklearn.metrics import confusion_matrix
from sklearn import metrics
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt

#Settings

#TODO: run for all cov_types
#run for DO_PRIOR=True and DO_PRIOR=False
MAX_MIXTURES=20
mixtures=range(1,1+MAX_MIXTURES,1)

#cov_types=['diag','spherical','full','tied']
COV_TYPE='tied'
NINIT=10                    #number of initialisations to test for GMM
DO_AUC=True                 #plot AUC curve
DO_ROC=False                #plot ROC curve for each mixture
DO_CLASSIFICATION=True      #do two class classification
DO_PRIOR=False              #use prior on two class classification likelihoods
#
    ------------------------------------------------------------------------------


#Import German Credit Data
fnm='C:\Users\Charl Cowley\Desktop\gmm\german.data-numeric.csv'

data=np.genfromtxt(fnm,delimiter=',')
#class 1 = positive (approve application)
#class 2 = negative (reject application)

X=data[:,0:-1]
if 1:#z-score data
    X=StandardScaler().fit_transform(X)
y=data[:,-1]
n1=len(np.where(y==1)[0])
n2=len(np.where(y==2)[0])
N=len(y)
skf=StratifiedKFold(y,n_folds=10,shuffle=True,random_state=0)

auc_one_class=[]
auc_two_class=[]
acc_two_class=[]

for M in mixtures:
```

```
50      print 'mixtures=%g'%M
51
52      lh1=[]
53      lh2=[]
54      log_lh1=[]
55      log_lh2=[]
56      clabs=[]
57      for train_index, test_index in skf:#cross validation
58          X_train, X_test = X[train_index], X[test_index]
59          y_train, y_test = y[train_index], y[test_index]
60
61          #Split data by classes
62          train_index_c1=np.where(y_train==1)[0]
63          train_index_c2=np.where(y_train==2)[0]
64
65          X_train_c1=X_train[train_index_c1,:]
66          X_train_c2=X_train[train_index_c2,:]
67          y_train_c1=y_train[train_index_c1]
68          y_train_c2=y_train[train_index_c2]
69
70          #Fit a GMM with M mixtures
71          #for class 1
72          #TODO: train GMM with mixture.GMM on X_train_c1 for each covariance
                type
73          gmm1=mixture.GMM(n_components=M, covariance_type=COV_TYPE, n_init=
                NINIT)
74          gmm1.fit(X_train_c1)
75          #Change n_iter?
76
77          #for class 2
78          #TODO: train GMM with mixture.GMM on X_train_c2 for each covariance
                type
79          gmm2=mixture.GMM(n_components=M, covariance_type=COV_TYPE, n_init=
                NINIT)
80          gmm2.fit(X_train_c2)
81
82          #Likelihoods for each class
83          lh_c1=np.power(10,gmm1.score_samples(X_test)[0])
84          lh_c2=np.power(10,gmm2.score_samples(X_test)[0])
85
86          #Log likelihoods for each class
87          log_lh_c1=gmm1.score_samples(X_test)[0]
88          log_lh_c2=gmm2.score_samples(X_test)[0]
89
90          #Append all likelihood scores
91          lh1=lh1+lh_c1.tolist()
92          lh2=lh2+lh_c2.tolist()
93          log_lh1=log_lh1+log_lh_c1.tolist()
94          log_lh2=log_lh2+log_lh_c2.tolist()
95
96          clabs=clabs+y_test.tolist()
97
98
```

```
99
100     #1-class classification
101
102     #Use class 1 (Positive class) in model
103     #Calculate AUC for each number of mixtures
104     if DO_AUC:#AUC
105         #calculate ROC curve from lh1 - thus use model 1 (majority class model
                ) as underlying model
106         #TODO: calculate ROC curve with metrics.roc_curve, hint: pos_label=1
                since the likelihoods of model 1 are used
107         fpr,tpr,thresholds=metrics.roc_curve(clabs,log_lh1,pos_label=1)
108
109         #calculate AUC from ROC curve
110         auc=metrics.auc(fpr, tpr)
111         auc_one_class.append(auc)
112
113     if DO_ROC:#ROC
114         plt.figure()
115         plt.plot([0, 1], [0, 1], 'k--')#base line
116         plt.plot(fpr,tpr)
117
118         plt.xlim([0.0, 1.0])
119         plt.ylim([0.0, 1.05])
120         plt.xlabel('False Positive Rate')
121         plt.ylabel('True Positive Rate')
122         plt.title('ROC curve - 1-class GMM (M=%g)(covariance=%s)'%(M,COV_TYPE)
                )
123         plt.legend()
124         plt.grid()
125         plt.savefig('./results/ROC_1class_M%g_cov%s.png'%(M,COV_TYPE))
126
127
128     #2-class classification
129
130     if DO_PRIOR:#only applicable to two class classification
131         p1=float(n1)/N
132         p2=float(n2)/N
133         for itr in range(len(lh1)):
134             lh1[itr]=p1*lh1[itr]
135             lh2[itr]=p2*lh2[itr]
136
137     if DO_CLASSIFICATION:#classification
138         plabs=[]#predicted labels
139         for itr in range(len(clabs)):
140             if lh1[itr]>lh2[itr]:
141                 plabs.append(1)#classify as class 1
142             else:
143                 plabs.append(2)#classify as class 2
144         cfm=confusion_matrix(clabs,plabs)
145         print cfm
146
147         acc=float(cfm[0,0]+cfm[1,1])/len(clabs)
148         acc_two_class.append(acc)
```

```
149
150
151     if DO_AUC:#AUC
152         #to generate an ROC curve for the two class classifier, use the
153         #likelihoods of both models and make each row sum to 1
154         lhs=np.vstack([lh1,lh2]).transpose()#stack likelihoods of two models
155         lhs_row_sum=np.sum(lhs,axis=1)
156         lhs[:,0]=np.divide(lhs[:,0],lhs_row_sum)
157         lhs[:,1]=np.divide(lhs[:,1],lhs_row_sum)
158         if 0:
159             lhs_row_sum=np.sum(lhs,axis=1)#test that all row now sum to 1
160
161         #calculate ROC curve from lhs[:,0] - thus use model 1 (majority class
                model) as underlying model
162         #TODO: calculate ROC curve with metrics.roc_curve, hint: pos_label=1
                since the normalised likelihoods of model 1 are used
163         fpr,tpr,thresholds=metrics.roc_curve(clabs,lhs[:,0],pos_label=1)
164
165         #calculate AUC from ROC curve
166         auc=metrics.auc(fpr, tpr)
167         auc_two_class.append(auc)
168
169     if DO_ROC:#ROC
170         plt.figure()
171         plt.plot([0, 1], [0, 1], 'k--')#base line
172         plt.plot(fpr,tpr)
173
174         plt.xlim([0.0, 1.0])
175         plt.ylim([0.0, 1.05])
176         plt.xlabel('False Positive Rate')
177         plt.ylabel('True Positive Rate')
178         plt.title('ROC curve - 1-class GMM (covariance=%s)'%COV_TYPE)
179         plt.legend()
180         #plt.show()
181         plt.grid()
182         plt.savefig('./results/ROC_2class_M%g_cov%s.png'%(M,COV_TYPE))
183
184 #   ------------------------------------------------------------------------


185 #AUC comparison
186 if DO_AUC:
187     plt.figure()
188     plt.plot(mixtures,auc_one_class,label='1-class GMM')#,'k-'
189     plt.plot(mixtures,auc_two_class,label='2-class GMM')#,'r-'
190     plt.xlabel('Mixtures')
191     plt.ylabel('AUC')
192     plt.title('Comparison of AUCs for varying mixtures (covariance=%s)'%
            COV_TYPE)
193     plt.legend()
194     plt.grid()
195     plt.savefig('./results/AUC_mixtures%g-%g_cov%s.png'%(mixtures[0],mixtures
            [-1],COV_TYPE))
```

```
196
197 if DO_CLASSIFICATION:
198     plt.figure()
199     plt.plot(mixtures,acc_two_class,label='2-class GMM')#,'k-'
200
201     plt.xlabel('Mixtures')
202     plt.ylabel('AUC')
203     plt.title('Classification accuracy (prior=%s)(covariance=%s)'%(DO_PRIOR,
            COV_TYPE))
204     plt.legend()
205     plt.grid()
206     plt.savefig('./results/acc_mixtures%g-%g_DO_PRIOR%g_cov%s.png'%(mixtures
            [0],mixtures[-1],DO_PRIOR,COV_TYPE))
207
208 #
    ------------------------------------------------------------------------------
```

A study of the moment generating functions of the generalized $\kappa - \mu$ and $\eta - \mu$ distributions in wireless systems

Micaela Giacovazzi 12114988

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mr. J.T. Ferreira, Co-supervisor: Prof. A. Bekker

Department of Statistics, University of Pretoria

2 November 2015

Abstract

In generalised fading models, the $\kappa - \mu$ and $\eta - \mu$ distribution is known for their encompassing nature, having many well-known distributions as special cases. In this study, the $\kappa - \mu$ and $\eta - \mu$ distribution is investigated, taking a particular interest in their moment generating functions (mgf) and the derivation thereof in closed form. The use of the mgf in the calculation of the average bit error rate (a popular performance metric in fading models) is highlighted, with emphasis on the ease of computation with these closed form mgfs.

Keywords: average bit error rate, fading channel, $\eta-\mu$ distribution, $\kappa-\mu$ distribution, signal-to-noise ratio

Declaration

I, *Micaela Armanda Giacovazzi,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_ _

Micaela Armanda Giacovazzi

_ _

Mr. J.T. Ferreira

_ _

Prof. A. Bekker

_ _

Date

Acknowledgments

I would like to express my deepest gratitude towards my parents, Ermano and Eugenia Giacovazzi, and Francois Strijdom who have shown me an incredible amount of support and encouragement throughout the course of this research project. I would also like to extend my gratitude towards Mr. Johan Ferreira and Prof. Andriette Bekker whose nurturing hands and guidance have been a invaluable gift to me.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background

A fading channel is a type of channel that experiences signal degradation from the transmitter to the receiver in a communication environment. In wireless systems specifically, fading stems from multipath induced fading or shadow fading. Multipath induced fading refers to the generated signals from the transmitter that reach the receiver by two or more paths. Shadow fading, however, is caused by the obstruction of objects in the propagation path between receiver and transmitter. In wireless systems, multipath fading and shadow fading occur simultaneously leading to the phenomenon referred to as composite fading [3].

The use of system performance measures such as the Signal-to-Noise Ratio (SNR) and the average bit error rate (ABER) assist in the identification of the behaviour of the signal over the propagation path between transmitter and the receiver. The SNR is measured at the output of the receiver [8]. A high SNR can indicate either a high level of signal strength or the absence of noise in the propagation path, or both. A low SNR could be interpreted as a weakness of the signal being transmitted or a high level of noise present in the propagation path, or both. The latter performance measure, ABER, is the most descriptive about the nature of the fading channel [8]. The ABER refers to the expected value of the bit error rate. Bit errors are pieces of data that have been altered from their original state upon arrival at the receiver.

There are known models that have been previously explored in literature which best characterize a fading channel. These models include: 1) Rayleigh Fading model, 2) Rician Fading model, 3) Nakagami Fading model, 4) Weibull Fading model, and 5) Log-Normal Shadowing model. Some of these models will be explored in this project. We will focus particularly on the appropriaten use of the $\kappa-\mu$ distribution and $\eta-\mu$ distribution in a fading environment. The $\eta-\mu$ distribution's advantage above the $\kappa-\mu$ distribution is that it better represents the small-scale variation of the fading signal in a non-line-of-sight environment [9]. Line-of-sight (LOS) refers to the path the signal travels from transmitter to receiver, this path is not obstructed by any objects that may cause interference. On the other hand, non-line-of-sight (NLOS) suggests the alternative, that is the partial or full presence of objects that can cause interruptions of signal projection in the path.

Common attributes shared among the mentioned existing fading models are a positive domain, a heavy tail and a positively skewed curve. The domain of a fading model represents the instantaneous SNR. The SNR value comprises of a signal strength value and a noise value. Neither value can be negative since the absence of strength implies a strength value of zero, similarly for the value of noise. [8] illustrates how the pdf of the SNR distribution can be derived via a simple transformation. Thus; it is important to take note that the pdf of the SNR of a model may not necessarily be the same pdf as that of the fading channel. The $\kappa-\mu$ and $\eta-\mu$ distributions are different generalizations of SNR distributions which contains many well-known fading channels' SNR distributions. The table below summarizes common SNR distributions for different fading channels.

Fading Channel	Fading parameter	Pdf ($f_\gamma(\gamma)$)
Rayleigh		$\frac{1}{\bar{\gamma}}\exp\left(-\frac{\gamma}{\bar{\gamma}}\right)$
Nakagami-m	$\frac{1}{2}\leq m$	$\frac{m^m\gamma^{m-1}}{\bar{\gamma}^{m+1}\Gamma(m)}\exp\left(-\frac{m\gamma}{\bar{\gamma}}\right)$
Hoyt	$0\leq q\leq 1$	$\left(\frac{1+q^2}{2q\bar{\gamma}}\right)\exp\left(-\frac{(1+q^2)^2\gamma}{4q^2\bar{\gamma}}\right)I_0\left(\frac{\left(1-q^4\right)\gamma}{4q^2\bar{\gamma}}\right)$
Rice	$0\leq n$	$\frac{(1+n^2)}{\exp(n^2)\bar{\gamma}}\exp\left(-\frac{(1+n^2)\gamma}{\bar{\gamma}}\right)I_0\left(2\sqrt{\frac{n^2(1+n^2)\gamma}{\bar{\gamma}}}\right)$

Table 1: Probability density functions (pdf) of the SNR per symbol γ for some well-known fading channels

1.2 Literature Review

The $\kappa-\mu$ distribution and the $\eta-\mu$ distribution were presented in [9]. It was shown that special cases of the $\kappa-\mu$ distribution include the Rice distribution, the Nakagami-m distribution and the Rayleigh distribution, and special cases of the $\eta-\mu$ distribution are the Hoyt distribution and also the Nakagami-m distribution and the Rayleigh distribution. The objective of this paper was to produce a general fading model, and to describe, parametrize and fully characterize the corresponding signal in terms of measurable physical parameters. [9] also contributed to the attainment of exact and closed-form moment-based estimators for the parameters of the $\kappa-\mu$ and $\eta-\mu$ distribution and a proposal for practical procedures to apply the distributions.

[6] proposed the existence of the moment generating functions (mgf) of the $\kappa-\mu$ and $\eta-\mu$ distributions in closed form. The need to obtain mgfs in terms of elementary functions stems directly from the need to calculate the average bit error rates (ABER) in fading channels. In this article, the relation between the modified Bessel function and the modified Bessel function of the first kind was used to derive said expressions (see Appendix: Definition 2). It was shown that for a large variety of practical applications, ABER formulas for systems operating in $\kappa-\mu$ and $\eta-\mu$ radio channels can be expressed either in terms of elementary functions or in terms of finite-range integrals of elementary functions, which is highly desirable because it is computationally simple to calculate.

[5] similarly proposed novel expressions for the mgfs of the generalized $\kappa-\mu$ and $\eta-\mu$ distribution. The motivation similarly comes from the applicability in the derivation of several performance measures such as the average error rate (AER). Well-known Meijer's G-functions were used to derive these new expressions as opposed to the Bessel function relation used by [6]. Meijer's G-functions are easily implemented by using appropriate computing software. When plotting the AER against the average SNR, it was observed that by keeping μ constant, an increase of κ and η implies in an improvement of the system performance. In a similar fashion, having kept η or κ constant, the performance improves as μ increases, as the AER decreases.

Al-Ahmadi and Yanikomeroglu [3] proposed an adjusted form of the expressions of the parameters of the approximating gamma distribution for modeling composite fading channels. Modeling composite fading channels is an important part of the analysis of several wireless communication problem such as interference analysis in cellular systems and performance analysis of network. It was shown in [3] that the generalized K probability density function (pdf) is approximated by the gamma pdf. The moment matching method was used to obtain the desired results, an adjustable form of the expressions. It is done by matching the first two positive moments, to overcome the arising numerical and/or analytical limitations of higher order moment matching. The results indicated that the introduction of the adjusted results in gamma pdf closely approximate the generalized-K distribution in both the lower and upper tail regions and can further approximate the distribution of the sum of independent generalized-K random variables in these regions.

The utility of the gamma distribution for shadow fading, in both terrestrial and satellite channels, using empirical data has been shown in [1]. The authors investigated this area because the mathematical form of the lognormal pdf is not convenient for the analytic calculations that arise in connection with shadow fading in wireless channels. The lognormal model of shadow fading cannot produce easy-to-use expressions for many performance measures such as average symbol error rate. The paper first demonstrates the utility of the gamma pdf for shadow fading in terrestrial and satellite channels using empirical data. For terrestrial channels, the data collected in the urban and suburban areas were employed. For the satellite channels, extensive empirical information that had been previously published was used. Secondly, it was shown how the application of the gamma pdf, in conjunction with the Rayleigh model of multipath fading, resulted in closed-form expressions for key system performance measures in the interested channels.

1.3 Objective and aims

The objective and aims of this study is to sufficiently:

- illustrate the pdf of the $\kappa - \mu$ and $\eta - \mu$ distribution for different values of κ and different values of η, respectively;
- derive the mgf of the $\kappa - \mu$ and $\eta - \mu$ distribution in closed form;
- evaluate the ABERs as an expression of the mgf of the $\kappa - \mu$ and $\eta - \mu$ distribution, respectively, and
- illustrate the relationship between the ABER and the SNR, using the $\kappa - \mu$ and $\eta - \mu$ distribution as the underlying fading models.

1.4 Outline of this study

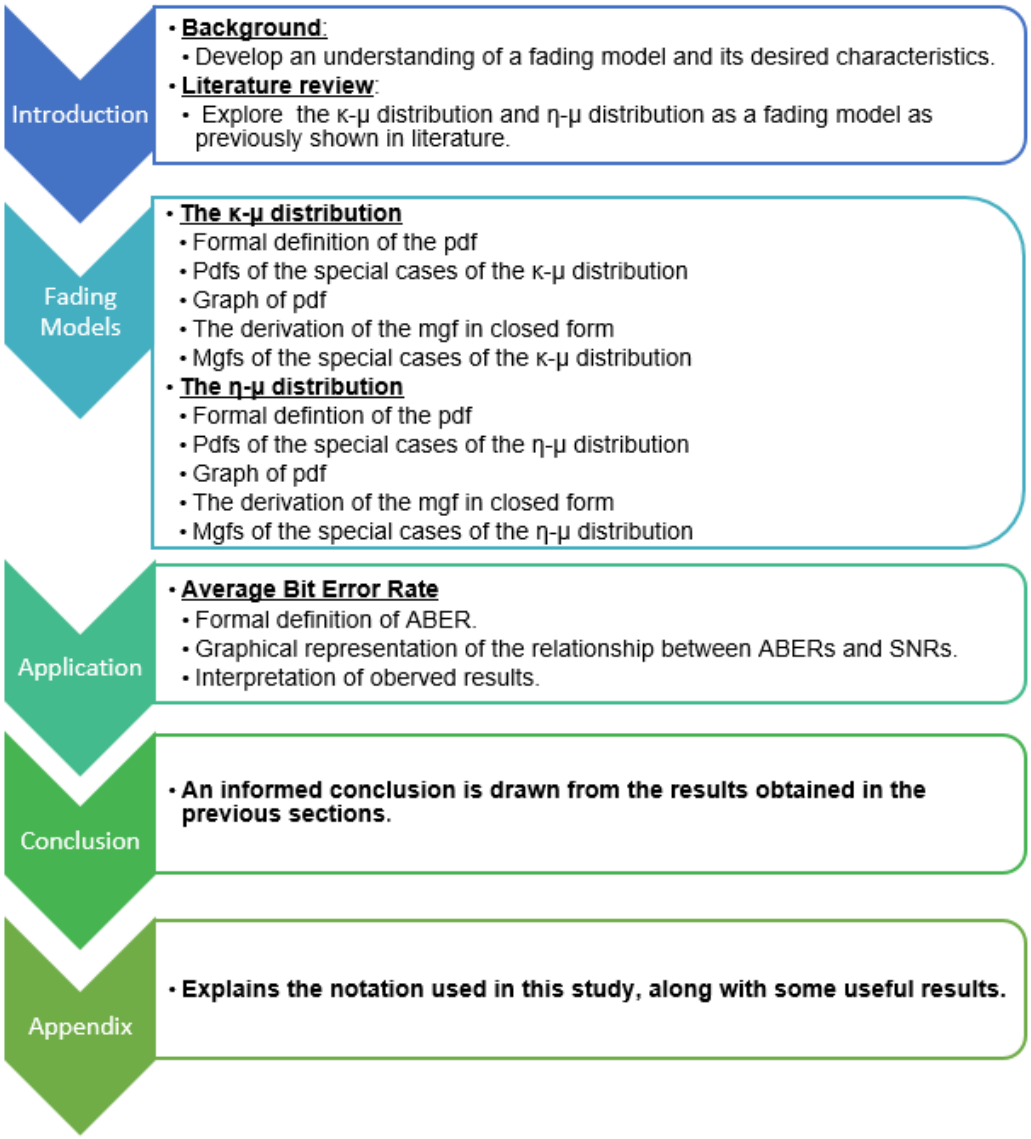


Figure 1: Outline of study

2 Fading Distributions

2.1 The $\kappa - \mu$ distribution

This section covers some important theoretical results regarding the $\kappa-\mu$ distribution. Yacoub (2007) derived this distribution from two mutually independent Gaussian random variables; each describing in-phase - and quadrature components of the fading channel. See Yacoub (2007) [9] for more detail.

2.1.1 Probability density function

A random variable γ is said to have the $\kappa - \mu$ distribution if it has the following pdf:

$$f_{\gamma_{\kappa-\mu}}(\gamma) = \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}\gamma^{\frac{\mu-1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right)I_{\mu-1}\left(2\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right);\gamma>0 \tag{1}$$

where $\mu = \frac{E^2[\gamma]}{2var[\gamma]}\left[\frac{1+2\kappa}{(1+\kappa)^2}\right]$, and $\kappa > 0$ is the ratio of the total power of the dominant components to that of scattered waves. γ denotes the signal-to-noise ratio, and $\bar{\gamma} = E(\gamma)$ denotes the average SNR. $I_\alpha(.)$ is the modified Bessel function of the first kind of order α (see Appendix: Definition 2).
Using the relation [9]:

$$I_{v-1}(z) \approx \frac{\left(\frac{z}{2}\right)^{v-1}}{\Gamma(v)}, \tag{2}$$

for small z.
The pdf (1) can also be written as:

$$\begin{aligned} f_{\gamma_{\kappa-\mu}}(\gamma) &= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}\gamma^{\frac{\mu-1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right)\frac{1}{\Gamma(\mu)}\left(\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right)^{\mu-1} \\ &= \frac{\mu\mu^{\mu-1}(1+\kappa)^{\frac{\mu+1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}\kappa^{\frac{\mu-1}{2}}\gamma^{\frac{\mu-1}{2}}\gamma^{\frac{\mu-1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}\Gamma(\mu)}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right) \\ &= \frac{\mu^\mu(1+\kappa)^\mu\gamma^{\mu-1}}{\exp(\mu\kappa)\bar{\gamma}^{\mu+1}\Gamma(\mu)}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right). \end{aligned} \tag{3}$$

Some special cases of this distribution is presented next.

Corollary 1: By setting $\mu = 1$ in (1), the pdf reduces to the Rice channel's SNR distribution with pdf

$$f_{\gamma_{Rice}}(\gamma) = \frac{(1+\kappa)}{\exp(\kappa)\bar{\gamma}}\exp\left(-\frac{(1+\kappa)\gamma}{\bar{\gamma}}\right)I_0\left(2\sqrt{\frac{\kappa(1+k)\gamma}{\bar{\gamma}}}\right),\gamma>0 \tag{4}$$

with $\kappa = n^2 > 0$ as the Rice parameter [8] and $\bar{\gamma} = E(\gamma)$ denotes the average SNR.

Corollary 2: By setting $\kappa = 0$ and $\mu = m$ in (3); the pdf in (3) reduces to the Nakagami-m channel's SNR distribution with pdf

$$f_{\gamma_{Nakagami-m}}(\gamma) = \frac{\mu^\mu\gamma^{\mu-1}}{\bar{\gamma}^{\mu+1}\Gamma(\mu)}\exp\left(-\frac{\mu\gamma}{\bar{\gamma}}\right),\gamma>0 \tag{5}$$

which is a gamma distribution with parameters μ and $\frac{\mu}{\bar{\gamma}}$ and $\bar{\gamma} = E(\gamma)$ denotes the average SNR [8].

Corollary 3: By setting $\kappa = 0$ and $\mu = 1$ in (3), then the pdf in (3) reduces to the Rayleigh channel's SNR distribution with pdf

$$f_{\gamma_{Rayleigh}}(\gamma) \quad = \quad \frac{1}{\bar{\gamma}} \exp\left(-\frac{\gamma}{\bar{\gamma}}\right), \gamma > 0 \tag{6}$$

which is an exponential distribution with parameter $\frac{1}{\bar{\gamma}}$ and $\bar{\gamma} = E(\gamma)$ denotes the average SNR [8].

An illustration of the pdf (1) for different arbitrary parameter values is given below.

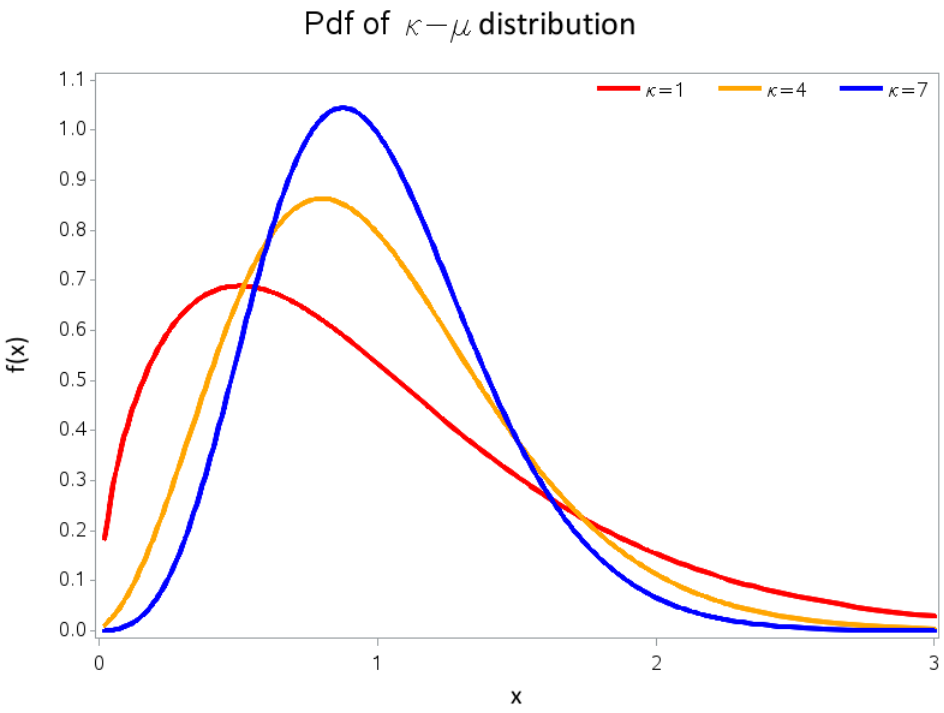


Figure 2: The $\kappa - \mu$ pdf (1) for fixed value of $\mu = 1.5$ and different values of κ

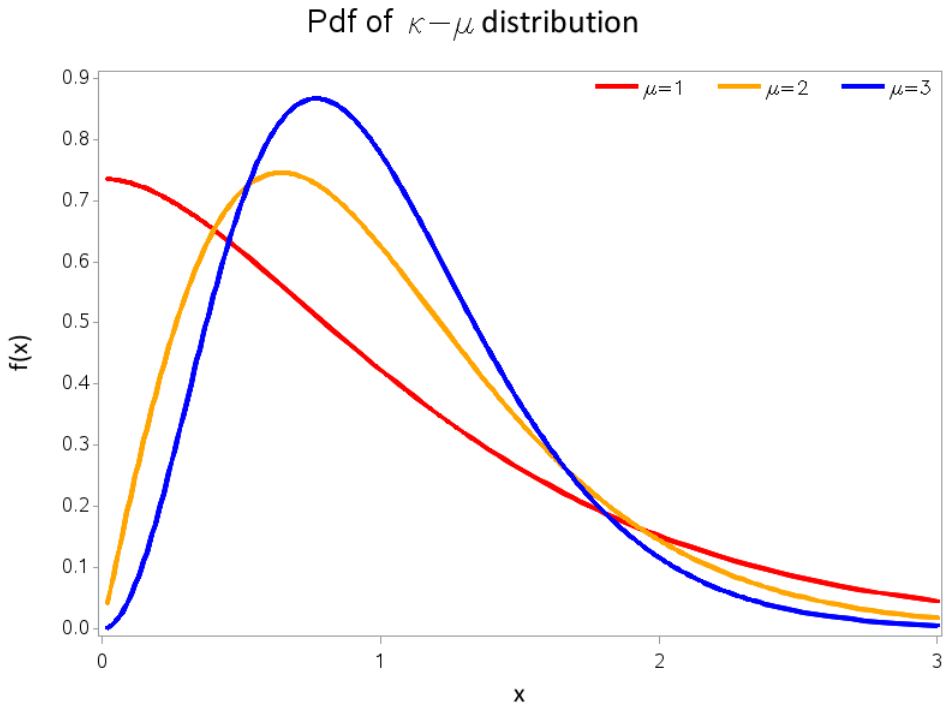


Figure 3: The $\kappa - \mu$ pdf (1) for fixed value of $\kappa = 1$ and different values of μ

2.1.2 Moment generating function

The following theorem gives the moment generating function of the $\kappa - \mu$ distribution.

Theorem: Suppose γ follows the $\kappa - \mu$ distribution with pdf (1). Then the moment generating function of γ is given by

$$M_{\gamma_{\kappa-\mu}}(s) \quad = \quad \left[\frac{\mu(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu} \exp\left[\mu^2 \frac{\kappa(1+k)}{s\bar{\gamma}+\mu(1+\kappa)} - \mu\kappa\right] \tag{7}$$

where $\mu = \frac{E^2[\gamma]}{2var[\gamma]}\left[\frac{1+2\kappa}{(1+\kappa)^2}\right]$, and $\kappa > 0$ is the ratio of the total power of the dominant components to that of scattered waves. γ denotes the signal-to-noise ratio, and $\bar{\gamma} = E(\gamma)$ denotes the average SNR.

Proof:
Consider

$$M_{\gamma_{\kappa-\mu}}(s) \quad = \quad E\{\exp(-s\gamma)\}$$

Using the definition of a mgf (see Appendix: Definition 1), we obtain:

$$= \int_0^{\infty} \exp(-s\gamma) f_{\gamma_{\kappa-\mu}}(\gamma) d\gamma$$

Using the pdf of the $\kappa - \mu$ distribution given by (1), it follows that $M_{\gamma_{\kappa-\mu}}(s)$ is equal to

$$
\begin{aligned}
&= \int_0^\infty \exp(-s\gamma)\frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}\gamma^{\frac{\mu-1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right)I_{\mu-1}\left(2\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right)d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp(-s\gamma)\gamma^{\frac{\mu-1}{2}}\exp\left(-\frac{\mu(1+\kappa)\gamma}{\bar{\gamma}}\right)I_{\mu-1}\left(2\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right)d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}I_{\mu-1}\left(2\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right)d\gamma.
\end{aligned}
$$

Using the series expansion of the modified Bessel function of the first kind (see Appendix: Definition 2), we obtain:

$$
\begin{aligned}
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}\sum_{j=0}^\infty \frac{1}{j!\Gamma(\mu+j)}\left(\frac{2\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}}{2}\right)^{\mu-1+2j} d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}\sum_{j=0}^\infty \frac{\left[\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right]^{\mu-1+2j}}{j!\Gamma(\mu+j)}d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}\left[\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right]^{\mu-1}\sum_{j=0}^\infty \frac{\left[\mu\sqrt{\frac{\kappa(1+\kappa)\gamma}{\bar{\gamma}}}\right]^{2j}}{j!\Gamma(\mu+j)}d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}\left[\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}\gamma^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\right]\sum_{j=0}^\infty \frac{\left[\mu^{2j}\frac{\kappa^j(1+\kappa)^j\gamma^j}{\bar{\gamma}^j}\right]}{j!\Gamma(\mu+j)}d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\frac{\mu-1}{2}}\gamma^{\frac{\mu-1}{2}}\sum_{j=0}^\infty \frac{\left[\mu^{2j}\frac{\kappa^j(1+\kappa)^j\gamma^j}{\bar{\gamma}^j}\right]}{j!\Gamma(\mu+j)}d\gamma \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\sum_{j=0}^\infty \frac{\left[\mu^{2j}\frac{\kappa^j(1+\kappa)^j}{\bar{\gamma}^j}\right]}{j!\Gamma(\mu+j)}\int_0^\infty \exp\left[-\gamma\left(s+\frac{\mu(1+\kappa)}{\bar{\gamma}}\right)\right]\gamma^{\mu-1+j}d\gamma.
\end{aligned}
$$

Using the definition of the gamma integral (see Appendix: Definition 4), we obtain:

$$
\begin{aligned}
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\sum_{j=0}^\infty \frac{\left[\mu^{2j}\frac{\kappa^j(1+\kappa)^j}{\bar{\gamma}^j}\right]}{j!\Gamma(\mu+j)}\left[\frac{1}{s+\frac{\mu(1+\kappa)}{\bar{\gamma}}}\right]^{\mu+j}\Gamma(\mu+j) \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\sum_{j=0}^\infty \frac{\left[\mu^{2j}\frac{\kappa^j(1+\kappa)^j}{\bar{\gamma}^j}\right]}{j!}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu+j} \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\sum_{j=0}^\infty \frac{\left[\mu^2\frac{\kappa(1+\kappa)}{\bar{\gamma}}\right]^j}{j!}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{j} \\
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\sum_{j=0}^\infty \frac{\left[\mu^2\frac{\kappa(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right]^j}{j!}.
\end{aligned}
$$

Using the series expansion of $e(.)$ (see Appendix: Definition 5), we obtain:

$$\begin{aligned}
&= \frac{\mu(1+\kappa)^{\frac{\mu+1}{2}}}{\kappa^{\frac{\mu-1}{2}}\exp(\mu\kappa)\bar{\gamma}^{\mu+1}}\mu^{\mu-1}\frac{\kappa^{\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu-1}{2}}}{\bar{\gamma}^{\frac{\mu-1}{2}}}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\exp\left[\mu^2\frac{\kappa(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right] \\
&= \mu^{1+\mu-1}\kappa^{\frac{\mu-1}{2}-\frac{\mu-1}{2}}(1+\kappa)^{\frac{\mu+1}{2}+\frac{\mu-1}{2}}\bar{\gamma}^{-\frac{\mu+1}{2}-\frac{\mu-1}{2}}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\exp\left[\mu^2\frac{\kappa(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right]\exp(-\mu\kappa) \\
&= \mu^{\mu}(1+\kappa)^{\mu}\bar{\gamma}^{-\mu}\left[\frac{\bar{\gamma}}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\exp\left[\mu^2\frac{\kappa(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right]\exp(-\mu\kappa) \\
&= \left[\frac{\mu(1+\kappa)}{s\bar{\gamma}+\mu(1+\kappa)}\right]^{\mu}\exp\left[\mu^2\frac{\kappa(1+k)}{s\bar{\gamma}+\mu(1+\kappa)}-\mu\kappa\right]
\end{aligned}$$

which leaves the final result. ■

Next, some special cases of the mgf in (7) is considered.

Corollary 4: Suppose γ follows the Rice channel's SNR distribution with pdf (4). Then the mgf of γ is given by

$$M_{\gamma}(s) = \left[\frac{(1+\kappa)}{s\bar{\gamma}+(1+\kappa)}\right]\exp\left[\frac{\kappa(1+k)}{s\bar{\gamma}+(1+\kappa)}-\kappa\right] \tag{8}$$

with $\kappa = n^2 > 0$ as the Rice parameter and $\bar{\gamma} = E(\gamma)$ denotes the average SNR..

Corollary 5: Suppose γ follows the Nakagami-m channel's SNR distribution with pdf (5). Then the moment generating function of γ is given by

$$M_{\gamma}(s) = \left[\frac{\mu}{s\bar{\gamma}+\mu}\right]^{\mu} \tag{9}$$

where $\mu = m > 0$ as the Nakagami-m parameter.

Corollary 6: Suppose γ follows the Rayleigh channel's SNR distribution with pdf (6). Then the mgf of γ is given by

$$M_{\gamma}(s) = \frac{1}{1-\bar{\gamma}s}. \tag{10}$$

2.2 The $\eta - \mu$ distribution

2.2.1 Probability density function

A random variable γ is said to have the $\eta - \mu$ distribution if it has the pdf given by [6]

$$f_{\gamma_{\eta-\mu}}(\gamma) = \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}}h^{\mu}\gamma^{\mu-\frac{1}{2}}}{\Gamma(\mu)H^{\mu-\frac{1}{2}}\bar{\gamma}^{\mu+\frac{1}{2}}}\exp\left(-\frac{2\mu\gamma h}{\bar{\gamma}}\right)I_{\mu-\frac{1}{2}}\left(\frac{2\mu H\gamma}{\bar{\gamma}}\right)\ \gamma > 0 \tag{11}$$

where γ denotes the signal-to-noise ratio, $\bar{\gamma} = E(\gamma)$ is the average SNR, $\Gamma(.)$ is the gamma function, $I_{\alpha}(.)$is the modified Bessel function of the first kind of order α and $\mu = \frac{E^2[\gamma]}{2var[\gamma]}\left[1 + \left(\frac{H}{h}\right)^2\right]$. The parameters h and H have different structures depending on the real-life scenario, and briefly set-out below.
Format 1:
$H = \frac{\left(\eta^{-1}-\eta\right)}{4}$ and $h = \frac{\left(2+\eta^{-1}+\eta\right)}{4}$ where $0 < \eta < \infty$ is the power ratio of the in-phase and quadrature scattered waves in each multipath cluster.
Format 2:
$H = \frac{\eta}{(1-\eta^2)}$ and $h = \frac{1}{(1-\eta^2)}$ where $-1 < \eta < 1$ is the correlation between the in-phase and quadrature scattered waves in each mulitpath cluster.
Using the relation as in (2):

$$I_{\mu-\frac{1}{2}}\left(\frac{2\mu^{1+\gamma}}{\bar{\gamma}}\right) = \left(\frac{\mu^{\mu-\frac{1}{2}}H^{\mu-\frac{1}{2}}\gamma^{\mu-\frac{1}{2}}}{\bar{\gamma}}\right)\frac{1}{\Gamma\left(\mu+\frac{1}{2}\right)},\ \gamma > 0$$

the pdf (11) can also be written as:

$$\begin{aligned} f_{\gamma_{\eta-\mu}}(\gamma) &= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}}h^{\mu}\gamma^{\mu-\frac{1}{2}}}{\Gamma(\mu)H^{\mu-\frac{1}{2}}\bar{\gamma}^{\mu+\frac{1}{2}}}\exp\left(-\frac{2\mu\gamma h}{\bar{\gamma}}\right)I_{\mu-\frac{1}{2}}\left(\frac{2\mu H\gamma}{\bar{\gamma}}\right) \\ &= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}}h^{\mu}\gamma^{\mu-\frac{1}{2}}\mu^{\mu-\frac{1}{2}}H^{\mu-\frac{1}{2}}\gamma^{\mu-\frac{1}{2}}}{\Gamma(\mu)H^{\mu-\frac{1}{2}}\bar{\gamma}^{\mu+\frac{1}{2}}\bar{\gamma}^{\mu-\frac{1}{2}}\Gamma\left(\mu+\frac{1}{2}\right)}\exp\left(-\frac{2\mu\gamma h}{\bar{\gamma}}\right) \\ &= \frac{2\sqrt{\pi}\mu^{2\mu}h^{\mu}\gamma^{2\mu-1}}{\Gamma(\mu)\Gamma\left(\mu+\frac{1}{2}\right)\bar{\gamma}^{2\mu}}\exp\left(-\frac{2\mu\gamma h}{\bar{\gamma}}\right). \end{aligned} \tag{12}$$

Some special cases of this distribution is presented next.

Corollary 7: By setting $\mu = \frac{1}{2}$ in (12), the pdf reduces to the Hoyt channel's SNR distribution with pdf

$$\begin{aligned} f_{\gamma_{Hoyt}}(\gamma) &= \frac{2\sqrt{\pi}\left(\frac{1}{2}\right)h^{\frac{1}{2}}}{\Gamma\left(\frac{1}{2}\right)\Gamma\left(\frac{1}{2}+\frac{1}{2}\right)\bar{\gamma}}\exp\left(-\frac{\gamma h}{\bar{\gamma}}\right) \\ &= \frac{h^{\frac{1}{2}}}{\bar{\gamma}}\exp\left(-\frac{\gamma h}{\bar{\gamma}}\right),\ \gamma > 0 \end{aligned} \tag{13}$$

with the Hoyt parameter given by $-\frac{1-\eta}{1+\eta}$ in format 1 or $-\eta$ in format 2 and $\bar{\gamma} = E(\gamma)$ is the average SNR.

Corollary 8: By setting $h = 1$ and $H = 1$ in (13), the pdf reduces to the Rayleigh channel's SNR distribution with pdf

$$f_{\gamma_{Rayleigh}}(\gamma) = \frac{1}{\bar{\gamma}}\exp\left(-\frac{\gamma}{\bar{\gamma}}\right),\ \gamma > 0 \tag{14}$$

which is an exponential distribution with parameter $\frac{1}{\bar{\gamma}}$ and $\bar{\gamma} = E(\gamma)$ is the average SNR.
An illustration of the pdf (11) for different arbitrary parameter values are given below.

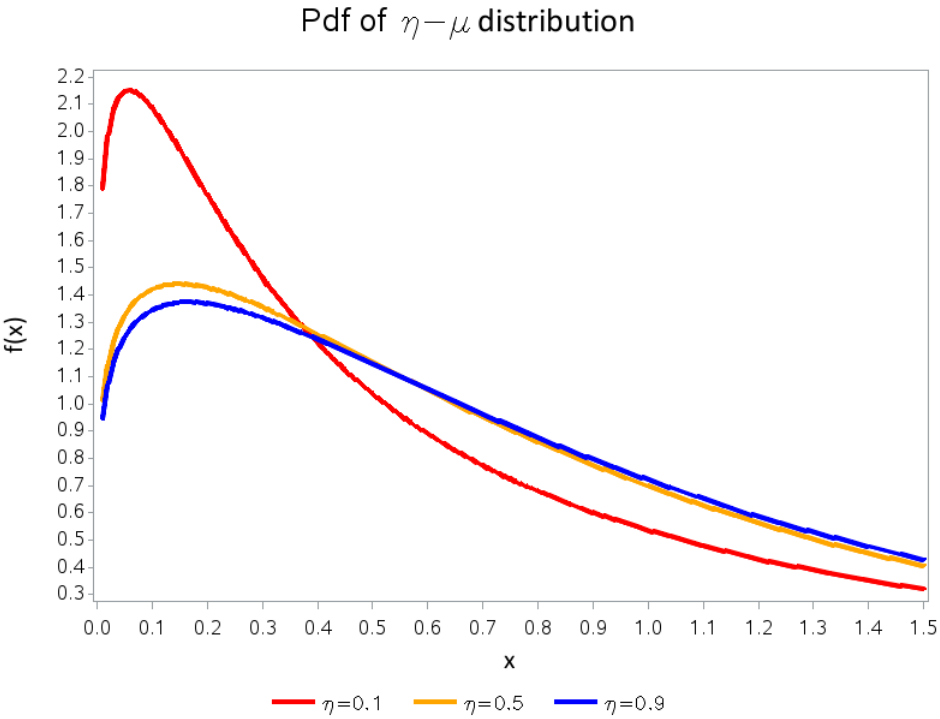


Figure 4: The $\eta - \mu$ pdf (11) for fixed value of $\mu = 0.6$ and different values of η; format 1

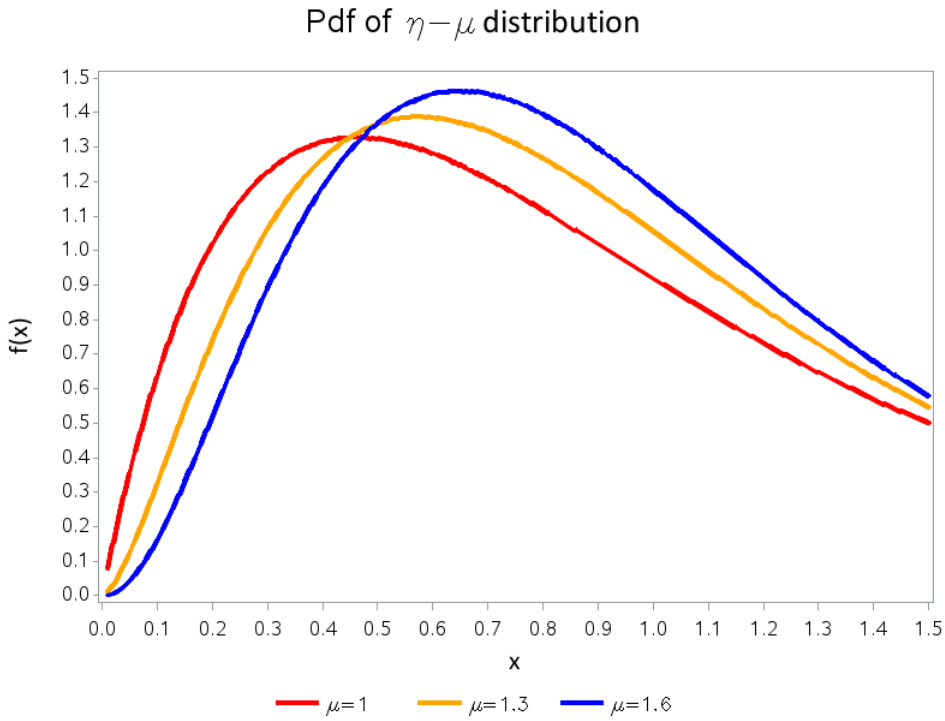


Figure 5: The $\eta - \mu$ pdf (11) for fixed value of $\eta = 0.5$ and different values of μ; format 1

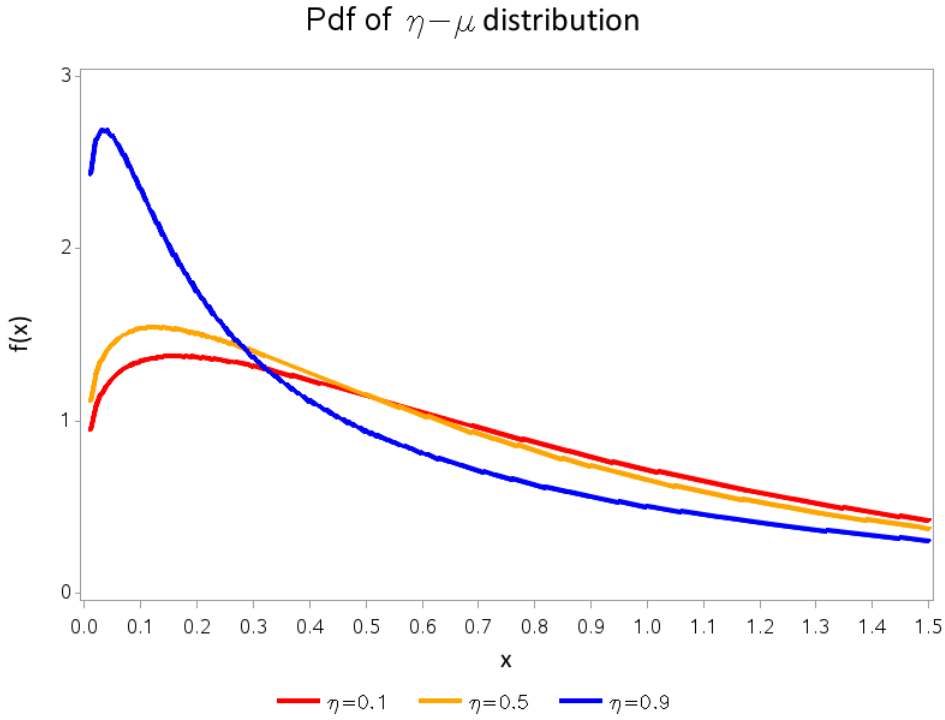


Figure 6: The $\eta - \mu$ pdf (11) for fixed value of $\mu = 0.6$ and different values for η; format 2

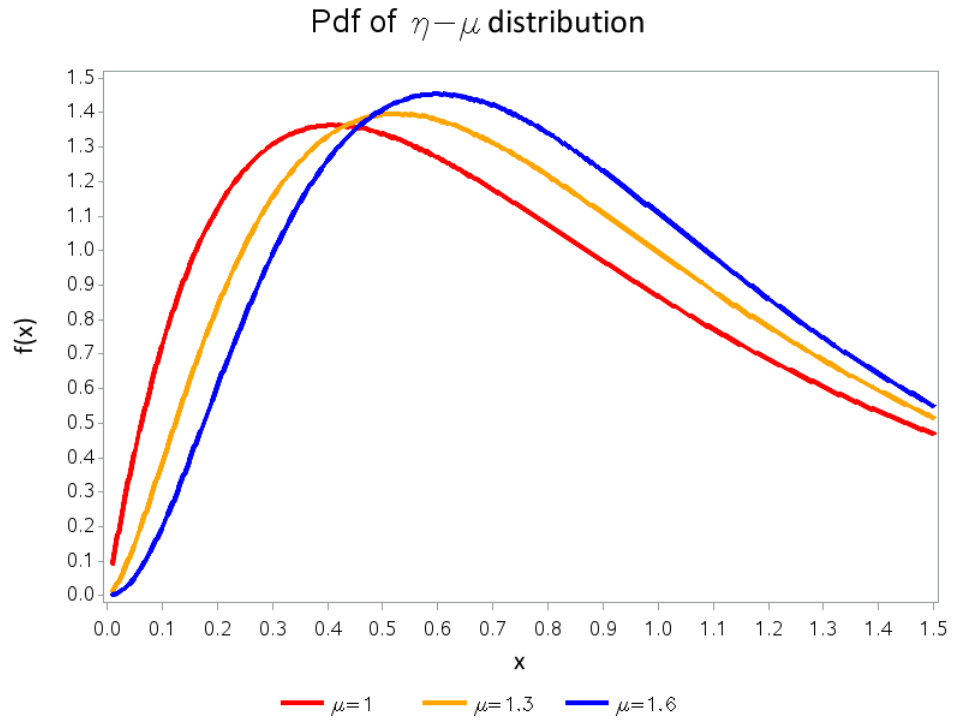


Figure 7: The $\eta-\mu$ pdf (11) for fixed value of $\eta = 0.5$ and different values of μ; format 2

2.2.2 Moment generating function

The following theorem gives the mgf of the $\eta-\mu$ distribution.

Theorem: Suppose γ follows the $\eta-\mu$ distribution with pdf (11). Then the mgf of γ is given by

$$M_{\gamma_{\eta-\mu}}(s) = \frac{4\mu^2 h}{(2(h-H)\mu+s\bar{\gamma})(2(h+H)\mu+s\bar{\gamma})} \tag{15}$$

where γ denotes the signal-to-noise ratio, $\bar{\gamma} = E(\gamma)$ is the average SNR, $\Gamma(.)$ is the gamma function, $I_\alpha(.)$is the modified Bessel function of the first kind of order α and $\mu = \frac{E^2[\gamma]}{2var[\gamma]}\left[1+\left(\frac{H}{h}\right)^2\right]$. The parameters h and H have different structures depending on the real-life scenario, and briefly set-out below.
Format 1:
$H = \frac{(\eta^{-1}-\eta)}{4}$ and $h = \frac{(2+\eta^{-1}+\eta)}{4}$ where $0 < \eta < \infty$ is the power ratio of the in-phase and quadrature scattered waves in each multipath cluster.
Format 2:
$H = \frac{\eta}{(1-\eta^2)}$ and $h = \frac{1}{(1-\eta^2)}$ where $-1 < \eta < 1$ is the correlation between the in-phase and quadrature scattered waves in each mulitpath cluster.

Proof:
Consider

$$M_{\gamma_{\kappa-\mu}}(s) = E\{\exp(-s\gamma)\}.$$

Using the definition of a mgf (see Appendix: Definition 1) we obtain:

$$= \int_0^\infty \exp(-s\gamma) f_{\gamma_{\eta-\mu}}(\gamma) d\gamma$$

and using the pdf of the $\eta - \mu$ distribution given by (11), it follows that the mgf equals

$$= \int_0^\infty \exp(-s\gamma) \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu \gamma^{\mu-\frac{1}{2}}}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \exp\left(-\frac{2\mu\gamma h}{\bar{\gamma}}\right) I_{\mu-\frac{1}{2}}\left(\frac{2\mu H\gamma}{\bar{\gamma}}\right) d\gamma.$$

Using the series expansion of the modified Bessel function of the first kind (see Appendix: Definition 2), we obtain:

$$= \int_0^\infty \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu \gamma^{\mu-\frac{1}{2}}}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \exp\left[-\gamma\left(s + \frac{2\mu h}{\bar{\gamma}}\right)\right] \sum_{k=0}^\infty \frac{1}{k!\Gamma(\mu+\frac{1}{2}+k)} \left(\frac{\frac{2\mu H\gamma}{\bar{\gamma}}}{2}\right)^{\mu-\frac{1}{2}+2k} d\gamma$$

Assuming $(\mu + \frac{1}{2} + k) \in N$, we obtain

$$= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \int_0^\infty \gamma^{\mu-\frac{1}{2}} \exp\left[-\gamma\left(\frac{s\bar{\gamma}+2\mu h}{\bar{\gamma}}\right)\right] \left(\frac{\mu H\gamma}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \left(\frac{\mu H\gamma}{\bar{\gamma}}\right)^{2k} d\gamma$$

$$= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \int_0^\infty \gamma^{\mu-\frac{1}{2}} \exp\left[-\gamma\left(\frac{s\bar{\gamma}+2\mu h}{\bar{\gamma}}\right)\right] \left(\frac{\mu H\gamma}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \left(\frac{\mu H\gamma}{\bar{\gamma}}\right)^{2k} d\gamma$$

$$= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \left(\frac{\mu H}{\bar{\gamma}}\right)^{2k} \int_0^\infty \gamma^{\mu-\frac{1}{2}} \exp\left[-\gamma\left(\frac{s\bar{\gamma}+2\mu h}{\bar{\gamma}}\right)\right] \gamma^{\mu-\frac{1}{2}} \gamma^{2k} d\gamma$$

$$= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \left(\frac{\mu H}{\bar{\gamma}}\right)^{2k} \int_0^\infty \gamma^{2\mu-1+2k} \exp\left[-\gamma\left(\frac{s\bar{\gamma}+2\mu h}{\bar{\gamma}}\right)\right] d\gamma$$

Using the definition of the gamma integral (see Appendix: Definition 4), we obtain:

$$= \frac{2\sqrt{\pi}\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \left(\frac{\mu H}{\bar{\gamma}}\right)^{2k} \left(\frac{\bar{\gamma}}{s\bar{\gamma}+2\mu h}\right)^{2\mu+2k} \Gamma(2\mu+2k)$$

$$= \frac{\mu^{\mu+\frac{1}{2}} h^\mu}{\Gamma(\mu) H^{\mu-\frac{1}{2}} \bar{\gamma}^{\mu+\frac{1}{2}}} \left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}} \sum_{k=0}^\infty \frac{1}{k!(\mu-\frac{1}{2}+k)!} \left(\frac{\mu H}{\bar{\gamma}}\right)^{2k} \left(\frac{\bar{\gamma}}{s\bar{\gamma}+2\mu h}\right)^{2\mu+2k} 2\sqrt{\pi}\Gamma(2(\mu+k)).$$

Using Legendre's duplication formula (see Appendix: Definition 6), we obtain:

$$
\begin{aligned}
&= \frac{\mu^{\mu+\frac{1}{2}}h^{\mu}}{\Gamma(\mu)H^{\mu-\frac{1}{2}}\bar{\gamma}^{\mu+\frac{1}{2}}}\left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}}\sum_{k=0}^{\infty}\frac{1}{k!(\mu+k-\frac{1}{2})!}\left(\frac{\mu H}{\bar{\gamma}}\right)^{2k}\left(\frac{\bar{\gamma}}{s\bar{\gamma}+2\mu h}\right)^{2\mu+2k}2^{2(\mu+k)}\Gamma(\mu+k)\Gamma\left(\mu+k+\frac{1}{2}\right)\\
&= \frac{\mu^{\mu+\frac{1}{2}}h^{\mu}}{\Gamma(\mu)H^{\mu-\frac{1}{2}}\bar{\gamma}^{\mu+\frac{1}{2}}}\left(\frac{\mu H}{\bar{\gamma}}\right)^{\mu-\frac{1}{2}}\sum_{k=0}^{\infty}\left(\frac{\mu H}{\bar{\gamma}}\right)^{2k}\left(\frac{\bar{\gamma}}{s\bar{\gamma}+2\mu h}\right)^{2\mu+2k}2^{2(\mu+k)}\left[\Gamma(\mu+k)\frac{1}{\Gamma(\mu)}\frac{1}{k!}\right]\\
&= \left[\frac{4\mu^2 h}{(s\bar{\gamma}+2\mu h)^2}\right]^{\mu}\sum_{k=0}^{\infty}\left[\frac{2\mu H}{(s\bar{\gamma}+2\mu h)}\right]^{2k}\left(\frac{(k+\mu-1)!}{k!(\mu-1)!}\right)\\
&= \left[\frac{4\mu^2 h}{(s\bar{\gamma}+2\mu h)^2}\right]^{\mu}\left[\frac{(s\bar{\gamma}+2\mu h)^2}{(s\bar{\gamma}+2\mu h)^2-4^2\mu^2H^2}\right]^{\mu}\sum_{k=0}^{\infty}\left[\begin{array}{c}k+\mu-1\\ \mu-1\end{array}\right]\left[\frac{4\mu^2H^2}{(s\bar{\gamma}+2\mu h)^2}\right]^{k}\left[\frac{(s\bar{\gamma}+2\mu h)^2+4\mu H^2}{(s\bar{\gamma}+2\mu h)^2}\right]^{\mu}.
\end{aligned}
$$

Using the definition of the pmf of a negative binomial distribution (see Appendix: Definition 3), we obtain

$$
\begin{aligned}
&= \left[\frac{4\mu^2 h}{(s\bar{\gamma}+2\mu h)^2}\right]^{\mu}\left[\frac{(s\bar{\gamma}+2\mu h)^2}{(s\bar{\gamma}+2\mu h)^2-4\mu^2H^2}\right]^{\mu}\\
&= \left(\frac{4\mu^2 h}{(s\bar{\gamma}+2\mu h)^2-4H^2\mu^2}\right)^{\mu}\\
&= \left(\frac{4\mu^2 h}{4h^2\mu^2-4H^2\mu^2+4h\mu s\bar{\gamma}+s^2\bar{\gamma}^2}\right)^{\mu}\\
&= \left(\frac{4\mu^2 h}{4h^2\mu^2-4H^2\mu^2+2h\mu s\bar{\gamma}-2H\mu s\bar{\gamma}+2h\mu s\bar{\gamma}+2H\mu s\bar{\gamma}+s^2\bar{\gamma}^2}\right)^{\mu}\\
&= \left(\frac{4\mu^2 h}{(2h\mu-2H\mu)(2h\mu+2H\mu)+s\bar{\gamma}(2h\mu-2H\mu)+s\bar{\gamma}(2h\mu+2H\mu)+s^2\bar{\gamma}^2}\right)^{\mu}\\
&= \left(\frac{4\mu^2 h}{(2h\mu-2H\mu+s\bar{\gamma})(2h\mu+2H\mu+s\bar{\gamma})}\right)^{\mu}\\
&= \left(\frac{4\mu^2 h}{(2(h-H)\mu+s\bar{\gamma})(2(h+H)\mu+s\bar{\gamma})}\right)^{\mu}
\end{aligned}
$$

Which leaves the final result. ■

Corollary 9: Suppose γ follows the Hoyt channel's SNR distribution with pdf (13). Then the moment generating function of γ is given by

$$
M_{\gamma}(s) = \left(\frac{h}{((h-H)+s\bar{\gamma})((h+H)+s\bar{\gamma})}\right) \tag{16}
$$

with the Hoyt parameter given by $-\frac{1-\eta}{1+\eta}$ in format 1 or $-\eta$ in format 2 and $\bar{\gamma}=E(\gamma)$ is the average SNR.

Corollary 10: Suppose γ follows the Rayleigh channels's SNR distribution with pdf (14). Then the moment generating function of γ is given by

$$
M_{\gamma}(s) = \frac{1}{1-\bar{\gamma}s}.
$$

where $\bar{\gamma}=E(\gamma)$ is the average SNR.

3 Application

The basic unit of data in a communication environment is called a bit. The word 'bit' is derived from the combination of the words 'binary' and 'unit' indicating that a bit can only take on one of two values, usually 1 or 0. Bit errors are bits of a data stream that are received over a communication channel from a transmitter but have been altered due to some kind of interference.

A bit error test can be simply illustrated as:
Transmitted bit sequence:
0 1 1 0 0 0 1 0 1,
Received bit sequence:
1 1 1 1 0 1 0 0 1,
the number of bit errors (those indicated) is in this illustration is four (4). The BER is 4 incorrect bits divided by ten (10) total transferred bits, results in $\frac{4}{10} = 0.4$ or 40.

ABERs measure how effectively the receiver is able to decode the transmitted data. The ABERs can be evaluated as [8]:

$$P_{aver} = \frac{1}{2} M_{\gamma}\left(C\frac{E_b}{N_o}\right) \tag{17}$$

where the ratio of the energy per bit to the noise spectral density $\frac{\mathbf{E_b}}{\mathbf{N_o}}$ defines the transmit SNR.
For a large variety of applications, the conditional bit error probability $P_b(E|\gamma)$ is expressed as [8]:

$$P_b(E|\gamma) = a_m Q\left(b_m\sqrt{\gamma}\right) \tag{18}$$

where Q is the Gaussian Q-function, a_m and b_m are parameters that depend on the specific modulation-detection combination and transmit SNR. However, a more accurate and representative form of the ABER is given by utilizing a finite integral representation of $Q(.)$. Thus on the basis of a finite integral representation of $Q(.)$ the ABER can also be evaluated as [8]:

$$P_{aver} = \frac{a_m}{\pi}\int_0^{\frac{\pi}{2}} M_{\gamma}\left(\frac{b_m^2}{2sin^2\theta}\right) d\theta \tag{19}$$

where $M_{\gamma}(.)$ represents the mgf of the SNR distribution of the fading channel.
In figures 4 - 6, we present numerical estimates obtained by using (7), (15), (with $\bar{\gamma}$ =1) and (19) for the case of coherent detection of binary phase shift keying (BPSK) [6]. Under these conditions, $a_m = 1$ and $b_m^2 = \frac{\mathbf{2E_b}}{\mathbf{N_o}}$ [8]. The curves in figure 4 and figure 5 are given for confirmation of correctness of our derivations since they represent results that have been already reported [5]. Using the closed form expression of the mgf, as in (7) and (15), is both computationally and analytically advantageous. The investigation here yields similar results compared to that of [5].

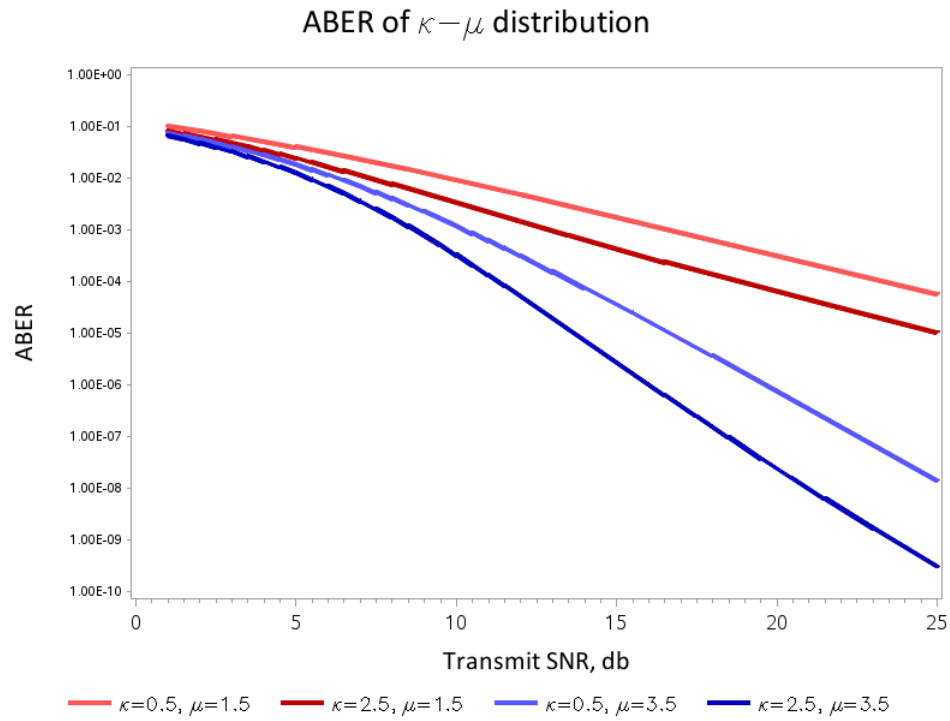


Figure 8: ABERs (19) of coherently detected BPSK in $\kappa-\mu$ radio channels

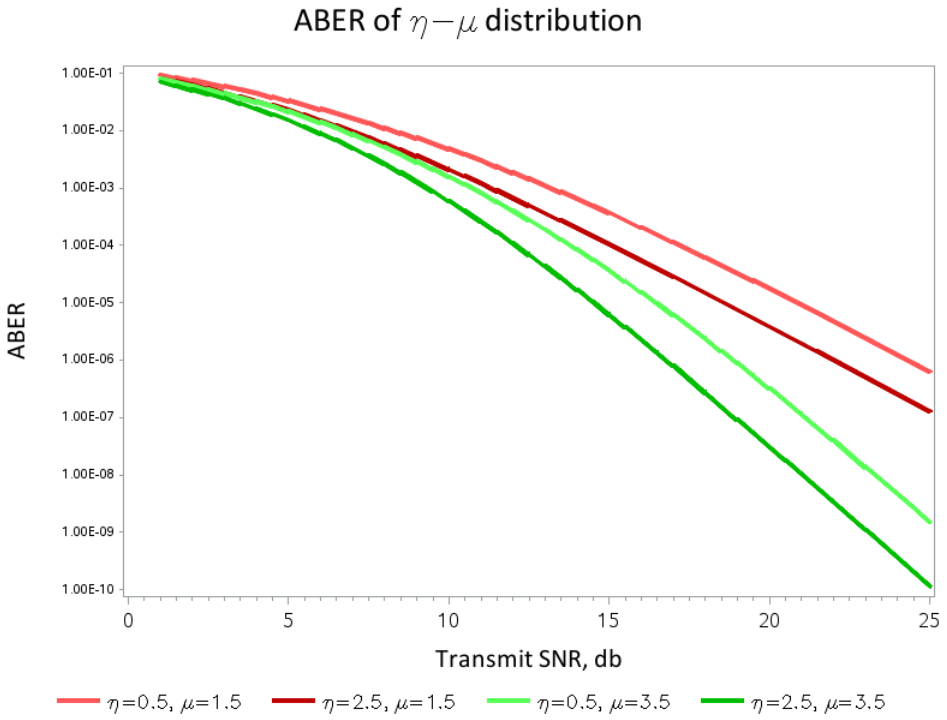


Figure 9: ABERs (19) of coherently detected BPSK in $\eta-\mu$ radio channels; format 1

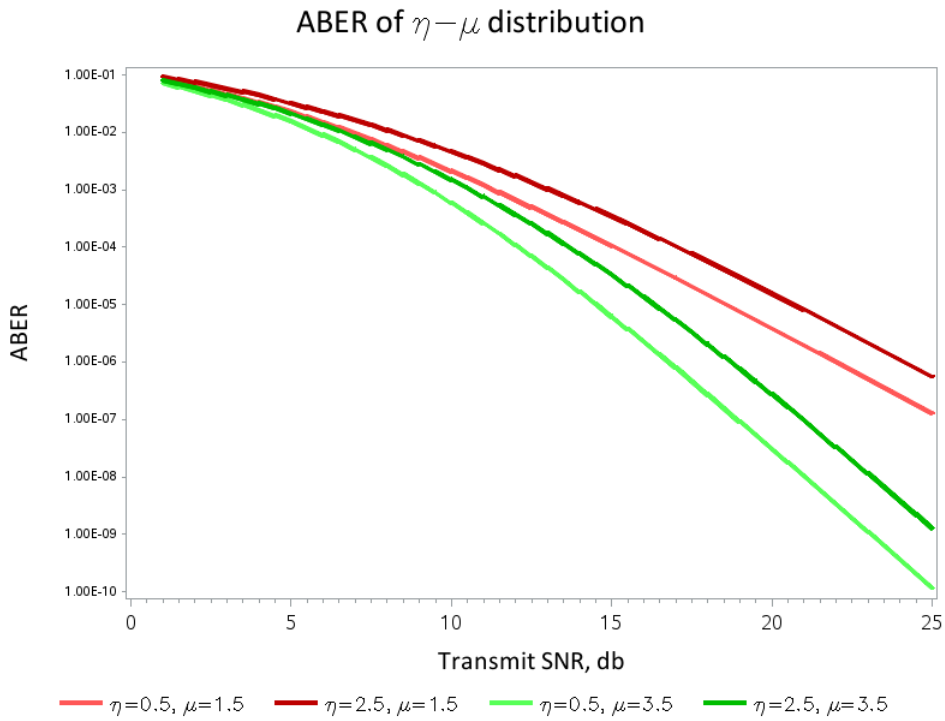


Figure 10: ABERs (19) of coherently detected BPSK in $\eta-\mu$ radio channels; format 2

The ABER decreases as the SNR increases. For a fixed value of κ, as μ increases the slope of the graph becomes steeper. The effects of η on the average error rates in radio channels modeled by two formats also differ. In contrast with the format 1, we observe that the error rate performance of the coherently detected BPSK in a $\eta-\mu$ radio channel of the format 2 improves as η decreases.

4 Conclusion

This project saw the systematic description of the $\kappa-\mu$ and $\eta-\mu$ distribution as generalised distributions of some well-known fading models. By deriving the mgfs of these distributions, their use to evaluate an important performance measure, namely ABER, was highlighted due to the ease of use. Some special cases of the distributions as well as their mgfs were discussed, and the ABER investigated.

5 Future work

There is plentiful scope to extend the research of this project to that of noncentral-type SNR distributions- and fading channels. Furthermore, composite fading model - where the model incorporates both fading - and shadowing, and the evaluation of the ABER of these models via their respective mgfs can also be investigated. In addition, other performance measures such as the outage probability of the model, can also be investigated.

References

[1] A. Abdi and M. Kaveh. A comparative study of two shadow fading models in ultrawideband and other wireless systems. *IEEE Transactions on Wireless Communications*, 2011.

[2] M. Abramowitz and I. Stegun. *Handbook of Mathematical Functions*, volume 10 issue 5. Tenth Printing, 1964.

[3] S. Al-Ahmadi and H. Yanikomeroglu. On the approximation of the generalized-k distribution by a gamma distribution for modeling composite fading channels. *IEEE Transactions on Wireless Communications*, 9 issue 2, 2010.

[4] L. J. Bain and M. Engelhardt. *Introduction to probability and mathematical statistics*, volume 4. Duxbury Press Belmont, CA, 1992.

[5] D. Benevides da Costa and M. D. Yacoub. Moment generating functions of generalized fading distributions and applications. *IEEE Communications Letters*, 12(2):112–114, 2008.

[6] N. Y. Ermolova. Moment generating functions of the generalized η-μ and k-μ distributions and their applications to performance evaluations of communication systems. *IEEE Communications Letters*, 12(7):502–504, 2008.

[7] A.P. Prudnikov, Y. A Brychkov, and O. Marichev. Integrals and series: Vol. i: Elementary functions; vol. 2: Special functions, 1986.

[8] M. K. Simon and M. Alouini. *Digital Communication over Fading Channels*, volume 95. John Wiley & Sons, 2005.

[9] M. D. Yacoub. The κ-μ distribution and the η-μ distribution. *IEEE Antennas and Propagation Magazine*, 49(1):68–81, 2007.

Appendix

Definition 1: [8]
Let γ be a random variable with pdf $f_\gamma(\gamma)$, then the expected value

$$\begin{aligned} M_\gamma(s) &= E\{\exp(-s\gamma)\} \\ &= \int_0^\infty \exp(-s\gamma) f_\gamma(\gamma) d\gamma \end{aligned}$$

is called the moment generating function (mgf) of γ if this expected value exists for all values of s in some interval of the form -g $<$ s $<$ g for some g $>$ 0.
Remark: This definition of a mgf differs from the traditional context of a mgf which does not include the negative in it's expression.

Definition 2: [7]
The modified Bessel function of the first kind is defined b

$$\begin{aligned} I_v(\gamma) &= \sum_{k=0}^{\infty} \frac{\left(\frac{\gamma}{2}\right)^{v+2k}}{k!\Gamma(v+k+1)} \\ &= \left(\frac{\gamma}{2}\right)^v \sum_{k=0}^{\infty} \frac{1}{k!\,(v+k)!} \left(\frac{\gamma}{2}\right)^{2k} \end{aligned}$$

where $v \in \mathbb{R}$ and $\Gamma(.)$ is the gamma function.

Definition 3: [4]
A random variable γ is said to follow a negative binomial distribution denoted $\gamma \sim NB(p,r)$, if γ has probability mass function (pmf)

$$f_{\gamma_{\kappa-\mu}}(\gamma) = \begin{pmatrix} k+r-1 \\ k \end{pmatrix} (1-p)^r p^k$$

where $k = 0, 1, 2, 3, \ldots$, $0 \leq p \leq 1$, $0 < r$ and $\begin{pmatrix} k+r-1 \\ k \end{pmatrix} = \frac{(k+r-1)!}{(k)!(r-1)!}$

Definition 4: [4]

$$\int_0^\infty x^{\beta-1} \exp(-\alpha x)\, dx = \frac{1}{\alpha^\beta}\Gamma(\beta)$$

for $\alpha, \beta > 0$.
Definition 5: [4]
The series expansion of e can be written as

$$e^x = \sum_{j=0}^{\infty} \frac{x^j}{j!}$$

Definition 6: [2]
The Legendre's duplication formula is defined as

$$2\sqrt{\pi}\Gamma(2z)\frac{1}{2^{2z}} \quad = \quad \Gamma(z)\Gamma(z+\frac{1}{2})$$

where $\Gamma(.)$ is the gamma function.

SAS Code:

The $\eta - \mu$ pdf for fixed value of $\eta = 0.5$ and different values of μ; format 1

```
*pdf of eta-mu distribution;
*fixed mu;
*format 1;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;

x_range = 1.5;
x_vec = do(0,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);


do mu=1 to 1.6 by 0.3;
eta = 0.5;

*format 1;
big_h = (1/eta - eta)/4;
small_h = (2+1/eta+eta)/4;

v1 = mu+0.5;
v2 = mu-0.5;
dens = 0;

  do i = 1 to nrow(x_vec);
    x = x_vec[i];
        pi = constant('pi');
    f1 = 2 * pi * mu**v1 * small_h**mu / (gamma(mu)*big_h**v2 * gamb**v1);
    f2 = exp(-2*mu*small_h*x/gamb)* x**v2;
    f3 = IBESSEL(mu-0.5,2*mu*big_h*x/gamb,0);
    dens_val = f1*f2*f3;

    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;

varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'h-m' f=calibri ' distribution';
symbol1  c=red width=3;
symbol2  c=orange width=3;
```

```
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "m=1" "m=1.3" "m=1.6")
position = (bottom center outside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

The $\eta - \mu$ pdf for fixed value of $\eta = 0.5$ and different values of μ; format 2

```
*pdf of eta-mu distribution;
*fixed eta;
*format 2;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;

x_range = 1.5;
x_vec = do(0,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);

do mu=1 to 1.6 by 0.3;
eta = 0.5;

*format 2;
big_h = eta/(1-eta**2);
small_h = 1/(1-eta**2);


v1 = mu+0.5;
v2 = mu-0.5;
dens = 0;

  do i = 1 to nrow(x_vec);
    x = x_vec[i];
        pi = constant('pi');
    f1 = 2 * pi * mu**v1 * small_h**mu / (gamma(mu)*big_h**v2 * gamb**v1);
    f2 = exp(-2*mu*small_h*x/gamb)* x**v2;
    f3 = IBESSEL(mu-0.5,2*mu*big_h*x/gamb,0);
    dens_val = f1*f2*f3;

    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;
```

```
varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'h–m' f=calibri ' distribution' ;
symbol1  c=red width=3;
symbol2  c=orange width=3;
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "m=1" "m=1.3" "m=1.6")
position = (bottom center outside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

The $\eta-\mu$ pdf for fixed value of $\mu = 0.6$ and different values of η; format 1

```
*pdf of eta–mu distribution;
*fixed mu;
*format 1;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;

x_range = 1.5;
x_vec = do(0,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);

do eta = 0.1 to 0.9 by 0.4;


*format 1;
big_h = (1/eta - eta)/4;
small_h = (2+1/eta+eta)/4;

mu=0.6;
v1 = mu+0.5;
v2 = mu-0.5;
dens = 0;

  do i = 1 to nrow(x_vec);
    x = x_vec[i];
        pi = constant('pi');
    f1 = 2 * pi * mu**v1 * small_h**mu / (gamma(mu)*big_h**v2 * gamb**v1);
    f2 = exp(-2*mu*small_h*x/gamb)* x**v2;
    f3 = IBESSEL(mu-0.5,2*mu*big_h*x/gamb,0);
    dens_val = f1*f2*f3;
```

```
    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;

varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'h-m' f=calibri ' distribution' ;
symbol1  c=red width=3;
symbol2  c=orange width=3;
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "h=0.1" "h=0.5" "h=0.9")
position = (bottom center outside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

The $\eta-\mu$ pdf for fixed value of $\mu = 0.6$ and different values of η; format 2

```
*pdf of eta-mu distribution;
*fixed mu;
*format 2;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;

x_range = 1.5;
x_vec = do(0,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);

do eta = 0.1 to 0.9 by 0.4;

*format 2;
big_h = eta/(1-eta**2);
small_h = 1/(1-eta**2);

mu=0.6;
v1 = mu+0.5;
v2 = mu-0.5;
dens = 0;
```

```
  do i = 1 to nrow(x_vec);
    x = x_vec[i];
        pi = constant('pi');
    f1 = 2 * pi * mu**v1 * small_h**mu / (gamma(mu)*big_h**v2 * gamb**v1);
    f2 = exp(-2*mu*small_h*x/gamb)* x**v2;
    f3 = IBESSEL(mu-0.5,2*mu*big_h*x/gamb,0);
    dens_val = f1*f2*f3;

    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;

varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'h-m' f=calibri ' distribution' ;
symbol1  c=red width=3;
symbol2  c=orange width=3;
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "h=0.1" "h=0.5" "h=0.9")
position = (bottom center outside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

The $\kappa-\mu$ pdf for fixed value of $\mu = 1.5$ and different values of κ

```
*pdf of kappa-mu distribution;
*For fixed mu;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;
kappa = 0.1;

x_range = 3;
x_vec = do(0.01,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);

do kappa =1 to 7 by 3;

mu = 1.5;
```

```
v1 = (mu+1)/2;
v2 = (mu-1)/2;
dens = 0;

  do i = 1 to nrow(x_vec);
    x = x_vec[i];
    f1 = mu * (1+kappa)**v1 / (kappa**v2*exp(mu*kappa)*gamb**v1);
    f2 = exp(-mu*(1+kappa)*x/gamb)* x**v2;
    f3 = IBESSEL(mu-1,2*mu*sqrt(kappa*(1+kappa)*x/gamb),0);
    dens_val = f1*f2*f3;

    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;

varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'k-m' f=calibri ' distribution';
symbol1  c=red width=3;
symbol2  c=orange width=3;
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "k=1" "k=4" "k=7")
position = (top right inside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

The $\kappa-\mu$ pdf for fixed value of $\kappa = 1$ and different values of μ

```
*pdf of kappa-mu distribution;
proc iml;
exp_gamsq = 2;
var_gam = 1;
gamb=1;
kappa = 1;

x_range = 3;
x_vec = do(0.01,x_range,0.01)';
complete_dens = j(nrow(x_vec),1,0);

do mu =1 to 3 by 1;
```

```
v1 = (mu+1)/2;
v2 = (mu-1)/2;
dens = 0;

  do i = 1 to nrow(x_vec);
    x = x_vec[i];
    f1 = mu * (1+kappa)**v1 / (kappa**v2*exp(mu*kappa)*gamb**v1);
    f2 = exp(-mu*(1+kappa)*x/gamb)* x**v2;
    f3 = IBESSEL(mu-1,2*mu*sqrt(kappa*(1+kappa)*x/gamb),0);
    dens_val = f1*f2*f3;

    dens = dens // dens_val;
  end; *i;
    dens = dens[2:nrow(dens),];
  complete_dens = complete_dens || dens;
end; *k;

x_vec = x_vec[2:nrow(x_vec),];
complete_dens = complete_dens[2:nrow(complete_dens),2:ncol(complete_dens)] || x_vec;

varnames = 'fx1' || 'fx2' || 'fx3' || 'x';
create graph from complete_dens[colname=varnames];
append from complete_dens;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black 'Pdf of ' f=greek 'k-m' f=calibri ' distribution';
symbol1  c=red width=3;
symbol2  c=orange width=3;
symbol3  c=blue width=3;

axis1 label=(f=calibri h=2 "x") value=(h=1.5) minor = none;
axis2 label=(f=calibri a=90 h=2 "f(x)") value=(h=1.5) minor=none ;
legend1 label=none value=(h=1.5 f=greek "m=1" "m=2" "m=3")
position = (top right inside);
proc gplot data=graph;
plot fx1*x fx2*x fx3*x/ overlay vaxis=axis2 haxis=axis1 legend=legend1;
run;
quit;
```

ABERs of coherently detected BPSK in $\kappa-\mu$ radio channels

```
*mgf of kappa mu distribution;
proc iml;
gamb = 1;

am = 1;
pi = constant('pi');
step = 0.001;

kappa_range = do(0.5,2.5,2)';
mu_range = do(1.5,3.5,2)';
s_range = do(1,25,0.5)';
```

```
mgf = j(nrow(s_range),1,0);

do m = 1 to nrow(mu_range);
mu = mu_range[m];
 do k = 1 to nrow(kappa_range);
 kappa = kappa_range[k];
  do s = 1 to nrow(s_range);
  val1 = 0;
   do paai = 0.001 to (pi/2) by step;
          SNR=10**(s_range[s]/10);
          sval = SNR/sin(paai)**2;
          A = (mu*(1+kappa)/(mu*(1+kappa)+sval*gamb) )**mu;
          B = exp( (mu**2)*kappa*(1+kappa)/(mu*(1+kappa)+sval*gamb) -mu*kappa);
          p = A*B;

      val1 = val1 + step*p;
    end; *paai;
        mgf[s,1] = am/pi * val1;
   end; *s;
        compl_mgf = compl_mgf || mgf;
  end; *m;
end; *k;

compl_mgf = compl_mgf || s_range;

varnames = 'mk1' || 'mk2' || 'mk3' || 'mk4' || 'snr';
create graph from compl_mgf[colname=varnames];
append from compl_mgf;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black f=calibri 'ABER of ' f=greek 'k-m' f=calibri ' distribution';
symbol1  c="light red" width=3;
symbol2  c="dark red" width=3;
symbol3  c="light blue" width=3;
symbol4  c="dark blue" width=3;

axis1 label=(f=calibri h=2 "Transmit SNR, db") order=0 to 25 by 5
                                                value=(f=cmr10 h=1.5) ;
axis2 logbase=10 logstyle=expand label=(f=calibri a=90 h=2 "ABER")
minor=none ;
legend1 label=none value=(h=1.5 f=greek "k=0.5, m=1.5" "k=2.5, m=1.5"
"k=0.5, m=3.5" "k=2.5, m=3.5")
position = (bottom center outside);
proc gplot data=graph;
plot mk1*snr mk2*snr mk3*snr mk4*snr/ overlay vaxis=axis2 haxis=axis1
legend=legend1;
run;
quit;
```

ABERs of coherently detected BPSK in $\eta - \mu$ radio channels; format 1

```
*mgf of eta mu distribution;
proc iml;
```

```
gamb = 1;

am = 1;
pi = constant('pi');
step = 0.01;

eta_range = do(0.1,0.8,0.7)';
mu_range = do(1.5,2.5,1)';
s_range = do(1,25,0.5)';
mgf = j(nrow(s_range),1,0);

do m = 1 to nrow(mu_range);
mu = mu_range[m];
 do e = 1 to nrow(eta_range);
 eta = eta_range[e];
 big_h = (1/eta - eta)/4;
 small_h = (2+1/eta+eta)/4;
  do s = 1 to nrow(s_range);
  val1 = 0;
   do paai = 0.001 to (pi/2) by step;
          SNR=10**(s_range[s]/10);
          sval = SNR/sin(paai)**2;
          A = (  4*mu**2*small_h
/(  (2*(small_h-big_h)*mu + sval*gamb)  * (2*(small_h+big_h)*mu+sval*gamb)))**mu;

      val1 = val1 + step*A;
    end; *paai;
        mgf[s,1] = am/pi * val1;
   end; *s;
        compl_mgf = compl_mgf || mgf;
  end; *m;
end; *k;

compl_mgf = compl_mgf || s_range;

varnames = 'mk1' || 'mk2' || 'mk3' || 'mk4' || 'snr';
create graph from compl_mgf[colname=varnames];
append from compl_mgf;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black f=calibri 'ABER of ' f=greek 'h-m' f=calibri ' distribution';
symbol1  c="light red" width=3;
symbol2  c="dark red" width=3;
symbol3  c="light green" width=3;
symbol4  c="dark green" width=3;

axis1 label=(f=calibri h=2 "Transmit SNR, db") order=0 to 25 by 5
value=(f=cmr10 h=1.5) ;
axis2 logbase=10 logstyle=expand label=(f=calibri a=90 h=2 "ABER")
minor=none ;
legend1 label=none value=(h=1.5 f=greek "h=0.5, m=1.5" "h=2.5, m=1.5"
"h=0.5, m=3.5" "h=2.5, m=3.5")
```

```
position = (bottom center outside);
proc gplot data=graph;
plot mk1*snr mk2*snr mk3*snr mk4*snr/ overlay vaxis=axis2 haxis=axis1
legend=legend1;
run;
quit;
```

ABERs of coherently detected BPSK in $\eta - \mu$ radio channels; format 2

```
*mgf of eta mu distribution;
proc iml;
gamb = 1;

am = 1;
pi = constant('pi');
step = 0.001;

eta_range = do(0.1,0.8,0.7)';
mu_range = do(1.5,2.5,1)';
s_range = do(1,25,0.5)';
mgf = j(nrow(s_range),1,0);

do m = 1 to nrow(mu_range);
mu = mu_range[m];
 do e = 1 to nrow(eta_range);
 eta = eta_range[e];
 big_h = eta/(1-eta**2);
 small_h = 1/(1-eta**2);
  do s = 1 to nrow(s_range);
  val1 = 0;
   do paai = 0.001 to (pi/2) by step;
          SNR=10**(s_range[s]/10);
          sval = SNR/sin(paai)**2;
          A = (  4*mu**2*small_h
/(  (2*(small_h-big_h)*mu+sval*gamb)  * (2*(small_h+big_h)*mu+sval*gamb)))**mu;

      val1 = val1 + step*A;
    end; *paai;
        mgf[s,1] = am/pi * val1;
   end; *s;
        compl_mgf = compl_mgf || mgf;
  end; *m;
end; *k;

compl_mgf = compl_mgf || s_range;

varnames = 'mk1' || 'mk2' || 'mk3' || 'mk4' || 'snr';
create graph from compl_mgf[colname=varnames];
append from compl_mgf;
quit;

goptions reset=all i=join ftext=calibri;
title h=2.5 c=black f=calibri 'ABER of ' f=greek 'h-m' f=calibri ' distribution';
symbol1  c="light red" width=3;
```

```
symbol2  c="dark red" width=3;
symbol3  c="light green" width=3;
symbol4  c="dark green" width=3;

axis1 label=(f=calibri h=2 "Transmit SNR, db") order=0 to 25 by 5
value=(f=cmr10 h=1.5) ;
axis2 logbase=10 logstyle=expand label=(f=calibri a=90 h=2 "ABER")
minor=none ;
legend1 label=none value=(h=1.5 f=greek "h=0.5, m=1.5" "h=2.5, m=1.5"
"h=0.5, m=3.5" "h=2.5, m=3.5")
position = (bottom center outside);
proc gplot data=graph;
plot mk1*snr mk2*snr mk3*snr mk4*snr/ overlay vaxis=axis2 haxis=axis1
legend=legend1;
run;
quit;
```

A rating system for rugby teams from multiple leagues

Rion Jansen 12108589

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisors: Dr. Paul J. van Staden, Dr. Inger Fabris-Rotelli

Department of Statistics, University of Pretoria

02 November 2015

October 4, 2015

Abstract

No current system exists to rank rugby teams across multiple leagues. In this report a ranking system to rate rugby teams in multiple rugby leagues, with the intention to measure their relative strength toward each other, was set up. Applying it to past results to get a current rating for the rugby teams can also lead to predicting the winner of a match before the match is played. This system will be applied on past results for teams from three different rugby leagues. An interactive and automated program was developed in SAS/IML for this purpose. A sensitivity analysis was also conducted.

Declaration

I, *Rion Jansen,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Rion Jansen

_____
Supervisors: *Dr. Paul J. van Staden, Dr. Inger Fabris-Rotelli*

_____
Date

Contents

List of Figures

List of Tables

1 Introduction

Creating a consistent, objective and fair rating system in rugby has many advantages when measuring the strength of different teams relative to each other and predicting future outcome of matches yet to take place. The rating system discussed in this report will assign a numerical value to each rugby team in a league, across different leagues, based on their past performances and recent results obtained, rating them from the best team to the worst team with higher ratings reflecting a better performing rugby team than a team with a lower rating.

In rugby, as in many sports, the resulting outcome of a match can be a win, draw or a loss for the each of the two teams. If say, team A wins, team B will automatically lose the match and visa versa, and a draw is an outcome where both teams end up on the same number of points at the end of the match. In a sport with a very low percentage of matches resulting in draws, such as rugby, the chances of predicting a result beforehand, at random, results in a 50/50 chance of guessing the outcome right. With the inclusion of a fair rating system for the rugby teams, the higher rated team should stand a higher chance of winning a match against a lower ranked team and thus, in turn, break the 50/50 chance of being right, and give the predictor a better chance of guessing the correct outcome [7].

In this report, background theory will first be discussed in section 2 to establish a footing for the rest of the paper. The rating system will be constructed in section 3 making use of techniques like standardizing the scores of the matches and exponential smoothing. After that the coding algorithm is explained in section 4 and then the results will be reported on in section 5 with a conclusion following in section 6.

2 Literature Review

Existing Ranking Systems for Rugby

An existing rating system[1] that has been used since 2003 by World Rugby, previously known as the International Rugby Board (IRB), to measure the strength of men's national rugby teams playing international rugby Tests in Rugby Union. The system rewards (or deducts) points for winning (or losing) a Test match[2] in a system that exchanges points between the two nations playing. The system takes into account four different factors based on:

- Where the Test match takes place (home, away or neutral ground),
- The difference in the rating of the two teams before the Test match,
- The final score result of the Test match,
- The importance of the match (e.g. Rugby World Cup finals will result in double point exchanges).

A Test match in which either side wins (or loses) by more than 15 points will result in a higher exchange of rating points. The range of values a team's rating can take on is from 0 up to 100. This system also gives more weight to more recent results, so past results will be superseded by the most recent outcomes, giving a more accurate depiction of a team's current strength. Currently there are 102 national teams that are recognized by the system as Member Unions. If a new country becomes a World Rugby Full Member Union they start off with a ranking of 30 points. One attribute of the system is that countries can fall from the top to the bottom in less than 20 matches. Countries that don't compete for a while will have their ratings removed from the system but as soon as they start playing again they will continue from their previous ranking. When countries merge to create a new rugby team they will automatically receive the higher of the applicable ratings. When countries split the new teams will have a lower rating than the original country's rating, decided based on a range of factors. One major shortfall of the system is that there is no credit given to say a lower rated team, narrowly losing to a much higher rated team, thus not giving a true reflection of a team's relative strength.

[1] http://www.worldrugby.org/rankings/explanation

[2] A Test match in rugby union is an international match, where the two teams are recognized by those countries' national governing bodies. These teams are usually the national senior rugby teams.

Within a league they utilize a log system where, at the start of the season, every team starts over on a score of zero. The teams accumulate points based on their performance throughout the season. No points are deducted for a loss. At the end of the season, either playoffs are played and a winner for the season is decided in a final, or in some leagues the team at the top of the log will be crowned the champions. This current system does not reflect the strength of a team in the early stages of the competition, and also does not give a perspective of whether the accumulated points were obtained a long while back and that the team is currently struggling to get a win. Another shortfall of this system is that it also does not have any predictive value of future outcomes, hence leading to the reason for the rating system constructed in this report.

Ranking Systems in Other Sports

Almost all sports take interest in rating competitors and take on varying techniques and have different underlying assumptions. In Australian Rules football (a sport with some of the same characteristics as rugby) approximately two months before the start of the 1981 season, a simple rating model was constructed that was based on an adjustive scheme [5], similar to the ELO system used by the World Chess Federation at the time, developed by chess Master Arpad Elo [1]. The Australian Rules football model uses exponential smoothing to adjust the ratings [5]. This technique has been used in rating participants of other sports like rating tennis players [**?**].

Exponential Smoothing

Exponential smoothing has also been used in time series analysis, which is the study of a sequence of observations, arranged in the order of the time of their outcome[3]. Exponential Smoothing has been used as a forecasting method to create prediction intervals. Another very important property of exponential smoothing is robustness [6]. Exponential smoothing is a technique which originated in the 1950s from the work of Robert G. Brown. He originally started his work on exponential smoothing working as an analyst for the United States Navy during World War II, where he was assigned the task of developing a tracking model for fire-control information on the location of submarines. This tracking model was a simple exponential smoothing model of continuous data. The model was later extended to fit more complex cases like trends and seasonality and also to work on discrete data [8]. He later went on to publish [4] in the Journal of Operations Research where many important results are described. There are many advantages gained by using these techniques. For example one can shorten the files on historical data, and also simplify the number of calculations [3]. Exponentially weighted moving average (EWMA) control charts are used in statistical process control, and takes into account all prior information available on the variables through a type of exponential smoothing. EWMA charts are very useful in monitoring the process mean [8]. EWMA charts are also robust under certain conditions [2].

This technique of smoothing will be used in our rating system where we assume a simple case with no trend or seasonality of the data hence giving rise to the following equation:

$$R_t = \alpha M_t + (1 - \alpha) R_{t-1}$$

where R_t is the new rating after a match has taken place for a team, M_t is the score obtained, using the system, for the match played at time t, R_{t-1} is the team's rating before the match has taken place, and finally $\alpha \in [0, 1]$ is the constant smoothing parameter for the system. Other factors and assumptions used in the model will be discussed later in the report.

3 The Rating System

The aim of the system proposed in this report is to rate all the teams playing in the Pro 12, Top 14 and Premiership league in Europe in a fair manner. We will refer to these three leagues as our three main leagues. Results from the Anglo Welsh cup, Rugby Challenge Cup, Rugby Championship Cup and also the Euro Rugby Championship Playoffs will be used together with the results from the Pro 12, Top 14 and

[3] http://www.investopedia.com/terms/t/timeseries.asp

Premiership to rank the teams. The reason is to take into account as many matches as possible, and also take advantage from the matches being played where teams from different leagues play each other. The algorithm constructed will first rate the teams and then they will be ranked after that from the highest rating, being the best team, to the lowest rating, being the worst team. The system will make use of the following parameters:

- The home team of the match will be referred to as HT, and the away team as AT.
- The home team's score will be called the HTS and the away team's score will be called ATS.
- A maximum Points difference $MPD = 50$ which will be used to alter the scores where the one team beats another team by more than 50 points. This is only to make sure that some "freak" results don't have too high of an impact on ratings. In some cases a team might lose by a large nmber of points not because of their lack of skill or strength but rather because of some other factor like they are resting all their best players before more important games in the coming weeks. For example a team that is assured of a home semi-final spot might not play a full strength team because of the risk of injury to key players and also wanting to rest certain players. This might result in the other team beating them by a large amount of points. For these kinds of reasons we do not want these results to have too high impact on the ratings thus the maximum point difference will be set at 50.
- An average $AVE = 50$ around which all the scores and ratings will be centered.
- An upper limit $UL = 100$ which is used when standardizing the scores of the home and away team for a match.
- A multiplier $M = 5$ which will be used when calculating the initial ratings of the teams.
- Exponential weight $\alpha = 0.1$ that will be utilized when updating the ratings after a match for the exponential smoothing step.
- Average values for new clubs $AVE_{new} = 40$. The reason for the value of 40 is that if a new team enters a league (because of say relegation of a team or teams, or the expansion of the league) it would be unfair to the established teams who have prior ratings that averaging around 50 to start the new team on the average value of 50 because they will most probably be a weaker team if they are a new team due to the nature of the sport. Thus we now start them on a slightly lower value of 40.

The algorithm will now be constructed that is used to rate the teams, using Figure 1: Calculating Initial Ratings, and Figure 2: Rating Throughout the Season, to aid in the understanding of the steps.

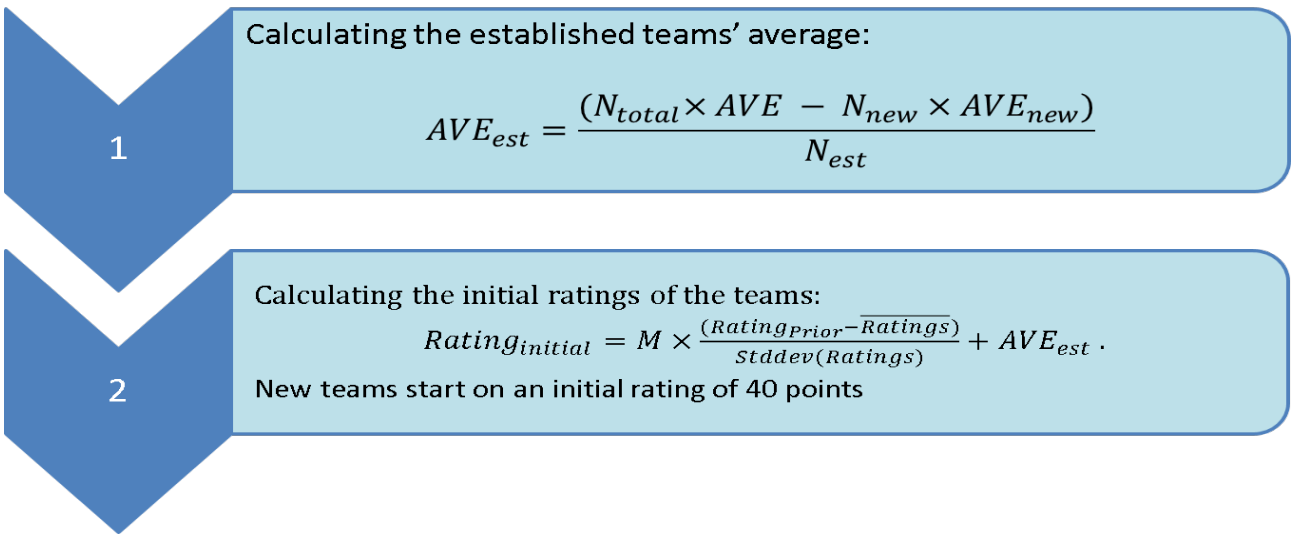


Figure 1: Calculating initial ratings

1. To calculate the initial ratings for the season we first need to calculate our average value for established teams AVE_{est}. This will be the readjusted scores of the teams who were playing in one of the three main leagues in the previous season and who are also playing in the current season. We need to know the number of new clubs joining the league without a prior rating, say N_{new}, that are starting on $AVE_{new} = 40$ points. We also need to know the number of established clubs in the league, say N_{est} and also the total number of teams in the league, sayN_{total}. To calculate AVE_{est}:

$$AVE_{est} = [(N_{total})(AVE) - (N_{new})(AVE_{new})]/N_{est}$$

2. Now we will standardize the ratings obtained in the previous season (this only applies to the teams who have a prior rating). This is calculated as:

$$Rating_{Initial} = M \times \frac{(Rating_{Prior} - \bar{Ratings})}{Stddev(Ratings)} + AVE_{est}$$

where $\bar{Ratings}$ is the average of the ratings of all the teams with a known prior rating from the previous season, and $Stddev(ratings)$ is the Standard deviation of the ratings of all the teams who had a known rating in the previous year. If a team does not have a known rating obtained from the previous year, for example the team is a new entry to the league, then they will receive an initial rating of $AVE_{new} = 40$ points.

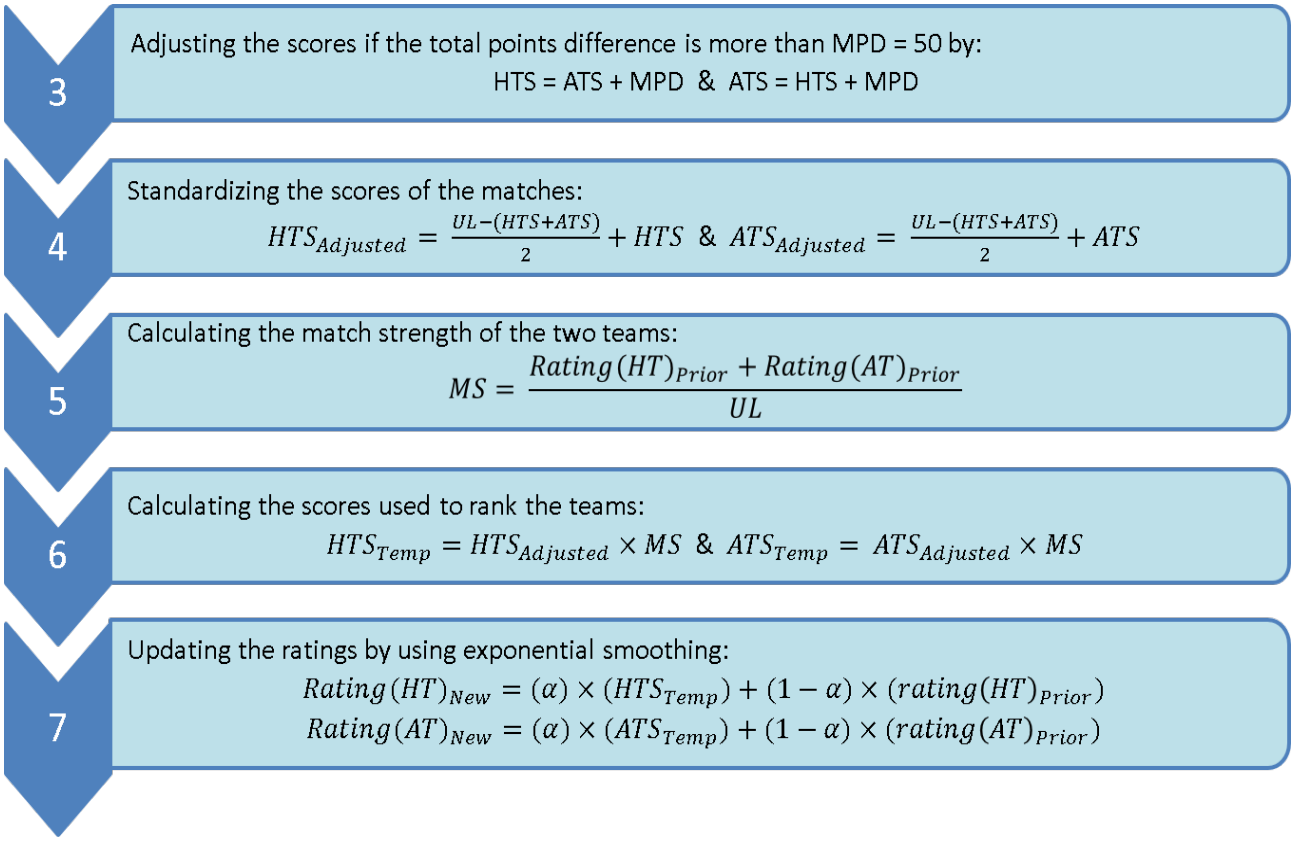


Figure 2: Rating throughout the season

3. Now each match's scores will be standardized but first we check to see if the total points difference $|HTS - ATS|$ is not more than our $MPD = 50$. For the home team: If $HTS - ATS$ is larger than MPD, then $HTS = ATS + MPD$, otherwise the home team's score stays the same. For the away

team: If $ATS - HTS$ is larger than MPD, then $ATS = HTS + MPD$, otherwise the away team's score stays the same.

To aid in the understanding of the system, we will use the final of the Anglo-Welsh Cup 2014/15. The two teams that played in the final on the22^{nd} of March 2015, at Franklin's Gardens, Northampton, were the Exeter Chiefs and Saracens. The venue was a neutral venue since it was not played at either of the team's home stadiums. For illustration purposes we assume that Saracens were the home team and Exeter Chiefs the away team. The final score was Saracens 23 - 20 Exeter Chiefs. We also need the ratings of the two teams before the match as this is used in a later calculation. We now use ratings of 59.01 and 59.59 for the Exeter Chiefs and Saracens respectively (these values have been calculated by our proposed system). For this game the points difference was a mere 3 points, so this calculation would only show that the two team's scores stay the same, since the points difference was not more than 50.

4. At the end of every match played, after the points difference calculation has been completed, the home team score and away team score will be adjusted in the following manner.
   For the home teams we have $HTS_{Adjusted} = \frac{UL-(HTS+ATS)}{2} + HTS$,
   and for the away teams we have $ATS_{Adjusted} = \frac{UL-(HTS+ATS)}{2} + ATS$.
   This means that the match scores for the example that we are considering would be adjusted in the following manner:
$$HTS_{Adjusted} = \frac{100 - (23 + 20)}{2} + 23 = 51.5,$$
$$ATS_{Adjusted} = \frac{100 - (23 + 20)}{2} + 20 = 48.5.$$

5. Next the scores will again be adjusted to take into account the strength of the two teams playing against each other. We will make use of the latest rating of the two teams prior to the match. The home team rating will be referred to as $Rating(HT)_{Prior}$, and the away team rating will be referred to as $Rating(AT)_{Prior}$. We will now define a match strength variable as
$$MS = \frac{Rating(HT)_{Prior} + Rating(AT)_{Prior}}{UL}$$
   The match strength for the match being considered will then be
$$MS = \frac{59.01 + 59.59}{100} = 1.186$$

6. Now we multiply the adjusted scores of the two teams which we calculated in step 4, with the MS that we calculated in step 5, to create two new temporary variables which will be used to update our ratings:
$$HTS_{Temp} = (HTS_{Adjusted})(MS),$$
$$ATS_{Temp} = (ATS_{Adjusted})(MS).$$
   For the match in consideration we now have:
$$HTS_{Temp} = 51.5 \times 1.186 = 61.079,$$
$$ATS_{Temp} = 48.5 \times 1.186 = 57.521.$$

7. To update the ratings after the match for the two teams, we finally make use of exponential smoothing and obtain, for the home team:

$$Rating(HT)_{New} = \alpha(HTS_{Temp}) + (1-\alpha)(Rating(HT)_{Prior}).$$

For the away team the calculation yields:

$$Rating(AT)_{New} = \alpha(ATS_{Temp}) + (1-\alpha)(Rating(AT)_{Prior}).$$

Now after the match we will need to update the ratings of the two teams.

$$Rating(HT)_{New} = 0.1(61.079) + (0.9)(59.59) = 59.7389,$$

$$Rating(AT)_{New} = 0.1(57.521) + (0.9)(59.01) = 58.8611.$$

8. Finally, at the end of each week in the competition we take these ratings for all the matches played and rank them from largest to smallest to get our team ranking list.

Steps 3 to 7 will be repeated after every match. This will make sure that the ratings of the teams playing are up to date after every match played by doing the calculations directly after the match and not say at the end of the week. This is important because if a team plays more than one match per week, the rating prior to say the second match in that week needs to be the latest reflection of their strength. So say for example a team plays two matches in a week, their rating prior to the second match should already take into account the result of the first match of the week.

4 Programming

In this chapter the programming and the data collection will be explained thoroughly. The results from the 2009/2010 season, up to the 2014/2015 season for the following leagues will be used:

- Premiership Rugby
- Pro 12
- Top 14
- Anglo-Welsh Cup
- European Rugby Challenge Cup
- European Rugby Champions Cup Play-offs (the matches only date back to 2014 so only those will be used)
- European Rugby Champions Cup

The goal is to rate the teams from the three main leagues, those are the twelve teams currently playing in Premiership Rugby, the twelve teams currently playing in the Pro 12 and the fourteen teams currently playing in the Top 14. The results of the seven leagues and play-offs were obtained from the ESPN website[4]. The matches of the other four leagues that are being utilized will only be taken into consideration when any two teams who play in one of the three main leagues, play against each other. A data set was created containing the dates of the matches played, the home team's name and their score, the away team's name and their score, and also the league from which the result came from.

The program was coded in Statistical Analysis Software (SAS), in iterative matrix language (IML). The following steps describe the program code:

[4] http://www.espn.co.uk/rugby/

1. Read in all the match data. Create a variable say C which is equal to the number of matches that we are taking into account.

2. All the parameters of the model are now specified as follows:

The maximum points difference per match score will be $mDiff = 50$.
The upper limit is $UL = 100$.
Exponential Weight is $EW = 10$.
The average rating points of all clubs is $AvP = 50$.
The multiplier is $Mul = 5$.
The number of teams being ranked is $nTeam = 38$.
Alpha the is $\alpha = EW/100$.

1. In this step we assign rating values to the teams who had a rating in the previous year.

2. To determine the established team's average we have -

$$EstCAvp = \frac{(nTeam \times AvP - nNewTeam \times NewCAvp)}{nEstTeam}$$

3. After the amount of new teams have been determined, the remainder ($nTeam - nNewTeam = 38 - nNewTeam$) teams will be seen as established teams. Then the initial rating scores for the established teams will be standardized by

$$StartRank = Mul\frac{(StartTemp - Ave)}{Stddev} + EstCAvp$$

where $Stddev$ is the standard deviation of the values in $StartTemp$, if this value is zero then we set it equal to one.

4. Standardizing the match scores around the average point. See Algorithm 1 : Standardizing

---

Algorithm 1 Standardizing

(a) To determine the established team's average we have -

$$EstCAvp = \frac{(nTeam \times AvP - nNewTeam \times NewCAvp)}{nEstTeam}$$

if $(pDiff) \geq (mDiff)$ then $HPADJ = (AP + mDiff)$, or
if $(pDiff) \leq (-1) \times (mDiff)$ then $HPADJ = (HP + mDiff)$.

(b) Let the match points total be $MPT = (HPADJ + APADJ)$.

$$ScoreTempA = \frac{UL - MPT}{2} + HPADJ$$

$$ScoreTempA = \frac{UL - MPT}{2} + APADJ$$

---

5. Move the standardized scores of each match into a matrix with the row being equal to the match number (sorted chronologically from oldest to newest) and the columns corresponding to the two team's position in the matrix (sorted alphabetically)

6. To determine the ranking update dates find the first Sunday before the first match is played and also the next Sunday after the final match of interest. The reason for the Sundays is that the new rating scores and rankings will be published each Sunday after all the matches on that Sunday is completed. Next we determine what the number of weeks is that the results span over.

7. Next we find out which match is in which week

8. In this step we count many matches have been played up to the corresponding week

9. In step twelve, steps five to seven as mentioned in section 3 in this report, will be used. See Algorithm 2: Exponential Smoothing

Algorithm 2 Exponential Smoothing

do {i = 1 to the total number of matches}
{set Ranktemp1 and ranktemp as the ratings of all the teams before the match}
{calculate the match strength variable}
{create scoretemp as the standardized scores multiplied by the match strength}
{put the two scoretemp values in their places inside the ranktemp1 vector corresponding to the columns of the two teams}
{let

$$Ranktemp3 = \alpha \times Scoretemp + (1 - \alpha) \times Ranktemp2$$

and let Ranktemp3 be added as a new line beneath Ranktemp1}
end do

10. To calculate the ratings at the end of each Sunday, we go through the total number of weeks that this season has continued for and only select the rows of Ranktemp1 that are the last of the week and put it into the *Rating* matrix.

11. Finally create a matrix *Rank* used to rank the ratings in the matrix *Rating* in each row. This will be the ranks of all the teams after each week.

5 Application and Findings

Sensitivity Analysis

Now we will consider the effects that different α and AVE_{new} values for new teams have on the system. First we shall make use of a win percentage test. This test is used to tell us what percentage of the time a team wins, given that they are rated higher than the other team. The test works as follows:

First we calculate the home team's net rank and net score. We focus on the home team only since the test will show the exact same result if the focus was on the away team. Let $HomeTeamRatingNet = Rating(HT) - Rating(AT)$ and let $HTScoreNet = Score(HT) - Score(AT)$. Secondly we let

$$Outcome = \begin{cases} 1 & if\,(HomeTeamRatingNet \geqslant 0 \& HTScoreNet \geqslant 0) \\ & or\,(HomeTeamRatingNet < 0 \& HTRankNet < 0)\ . \\ 0 & \text{otherwise} \end{cases}$$

We calculate this after each match and let *CountOutcome* be the sum of the *Outcome* variable for each match. Finally we divide *CountOutcome* by the total number of matches and multiply that by 100 to obtain a percentage.

Another test that was applied is to measure Pearson's correlation coefficient between the $HTRankNet$ and $HTScoreNet$ with the formula

$$r = \frac{\sum HomeTeamRatingNet \times HTScoreNet}{\sqrt{\sum(HomeTeamRatingNet^2) \times \sum(HTScoreNet^2)}}.$$

This test will be used to see if we find a correlation between the net difference in the ratings prior to the match of the teams and the standardized scores obtained in the match. If there is a positive correlation we

can see that higher rated teams also tend to score more when playing lower rated teams, proportionate to the difference in their ratings.

We will examine the win percentages for different combinations of α and starting values for new teams that we apply from the 2009/10 season up to the 2014/15 season for all of these combinations.

In Table 1 the win percentage test is conducted and in Table 2 the Pearson's correlation coefficient is calculated for the 2014/15 season.

		Starting	values	for	new	teams
		30	35	40	45	50
α	0.05	63.68	64.01	63.84	63.52	62.70
	0.1	64.33	64.66	63.84	63.52	63.52
	0.15	62.38	62.38	62.05	61.89	62.05
	0.2	62.05	61.73	61.88	62.05	61.56

Table 1: Win percentages sensitivity analysis

		Starting	values	for	new	teams
		30	35	40	45	50
α	0.05	0.503	0.510	0.508	0.494	0.460
	0.1	0.511	0.513	0.509	0.499	0.479
	0.15	0.502	0.501	0.496	0.487	0.472
	0.2	0.486	0.484	0.479	0.471	0.459

Table 2: Correlation sensitivity analysis

In this analysis we see that the value of $\alpha = 0.1$ and a $AVE_{new} = 35$ for new teams perform the best because the accuracy and the correlation is the highest for this combination out of all the combinations. We will however not start new teams on an initial value of 35 because this might make it harder for a good new team to reach a high rating really quickly. So a starting value of 40 will be used for new teams so that they might see progress much more quickly and not be demotivated from being on a lower rating.

Findings

When applying the model to the data, with $\alpha = 0.1$ and the AVE_{new} for new teams as 40, on the 38 teams from the 2010/11 season (36 teams in the 2009/10 season), we make use of the three finals played on the 31st of May 2014.

The first final, that of the Aviva Premiership, was played between the Northampton Saints (NSA) and Saracens (SAR), with ratings prior to the game of 62.16 and 62.31 respectively. The final score of the match was $NSA\ 24 - 20\ SAR$ so the lower rated team won the match but we see that the two team's ratings were very close to each other so it is hard to really tell who is a better team. After the match the ratings were adjusted and the Northampton Saints moved up to a rating of 62.41 and Saracens dropped down to 62.05.

In the second final, the Pro 12 Grand Final, Leinster (LEI) faced off against the Glasgow Warriors (GLA) and the two teams had ratings of 59.38 and 55.83 before the match respectively. The final score was $LEI\ 34 - 12\ GLA$ and we see that the higher rated team won the match. After the match the ratings were updated and Leinster's rank jumped up to 60.47 and Glasgow's rank fell down to 54.74.

In the Top 14 final, Toulon (TOU) played Castres (CAS) and both teams had ratings before the match of 59.76 and 53.98 respectively. The final score was $TOU\ 18 - 10\ CAS$ so once again the higher rated team won the match. The ratings adjusted to become 59.92 and 53.81 for Toulon and Castres respectively.

The win percentage test was conducted again but this time it was elaborated a bit to test different strengths of teams. The seasons used were those from 2010/11 up to the 2014/15 season, since in the 2009/10 season, all the teams started the season on a rating score of 50 points so no real useful information would be acquired when conducting this test in the season.

The test now works as follows:

- Again we let $HomeTeamRatingNet = Rating(HT) - Rating(AT)$ and let $HTScoreNet = Score(HT) - Score(AT)$.
- Now let $CountOutcome$ be the number of times that $HomeTeamRatingNet \geqslant i$ where $i = 0, 3, 6, 9, 12, 15$ and $HTScoreNet \geqslant 0$, or if $HomeTeamRatingNet < -i$ where $i = 0, 3, 6, 9, 12, 15$ and $HTScoreNet < 0$ for all the games played. Again we divide $CountOutcome$ by the total number of matches that had $HomeTeamRatingNet \geqslant i$ or $HomeTeamRatingNet < -i$ and multiply that by 100 to obtain a percentage.

The results of the elaborated win percentage test is summarized in Table 3

			Season		
Home Team Rating Net(+/-) difference	2010/11	2011/12	2012/13	2013/14	2014/15
0	65.18	62.58	71.74	65.30	63.84
3	68.27	67.44	75.43	70.68	66.16
6	70.82	72.22	79.81	73.63	72.64
9	70.65	75.64	80	77.59	77.30
12	75.61	83.75	80.53	78.57	86.02
15	84.85	90	85.45	82.93	87.27

Table 3: Win Percentages

Now we will apply the Pearson's correlation coefficient on $HTRankNet$ and $HTScoreNet$ again with $\alpha = 0.1$ and $AVE_{new} = 40$.

The results obtained are tabulated in Table 4.

Season	2010/11	2011/12	2012/13	2013/14	2014/15
r	0.4000247	0.4216834	0.5104757	0.4641594	0.509069

Table 4: Correlation

To illustrate the effect the system has on a particular team's rating and ranks throughout a season, we can look at a few matches played, involving the Irish team Munster who played in the Pro 12 and the European Rugby Champions Cup. Their progress is summed up in Table 5.

Date of Match	16-08-14	05-09-14	12-09-14	19-09-14
Munster- Rank (Ranting)	6 (56.40)	8 (55.68)	10 (55.50)	9 (56.06)
Opponent	Sale Sharks	Edinburgh	Benetton Treviso	Zebre
Opponent's Rank (Rating) - before the match	16 (51.97)	29 (46.04)	36 (41.44)	36 (41.50)
Score: Munster - Opponent	27 - 26	13 - 14	21 - 10	31 - 5

Date of Match	23-09-14	10-10-14	23-05-15	30-05-15
Munster- Rank (Ranting)	8 (55.72)	7 (56.51)	3 (59.40)	4 (59.31)
Opponent	Ospreys	Scarlets	Ospreys (semi-final)	Glasgow (final)
Opponent's Rank (Rating) - before the match	11 (54.92)	1 (61.2)	13 (54.18)	8 (56.83)
Score: Munster - Opponent	14 - 19	17 - 6	21 - 18	13 - 31

Table 5: Munster Progress

In Figure 3 and 4 the progress of all the teams playing in the PRO 12 can be seen by looking at their ratings and rankings (in the rankings a lower ranked team is seen as a better team) throughout the duration of the 2014/15 season.

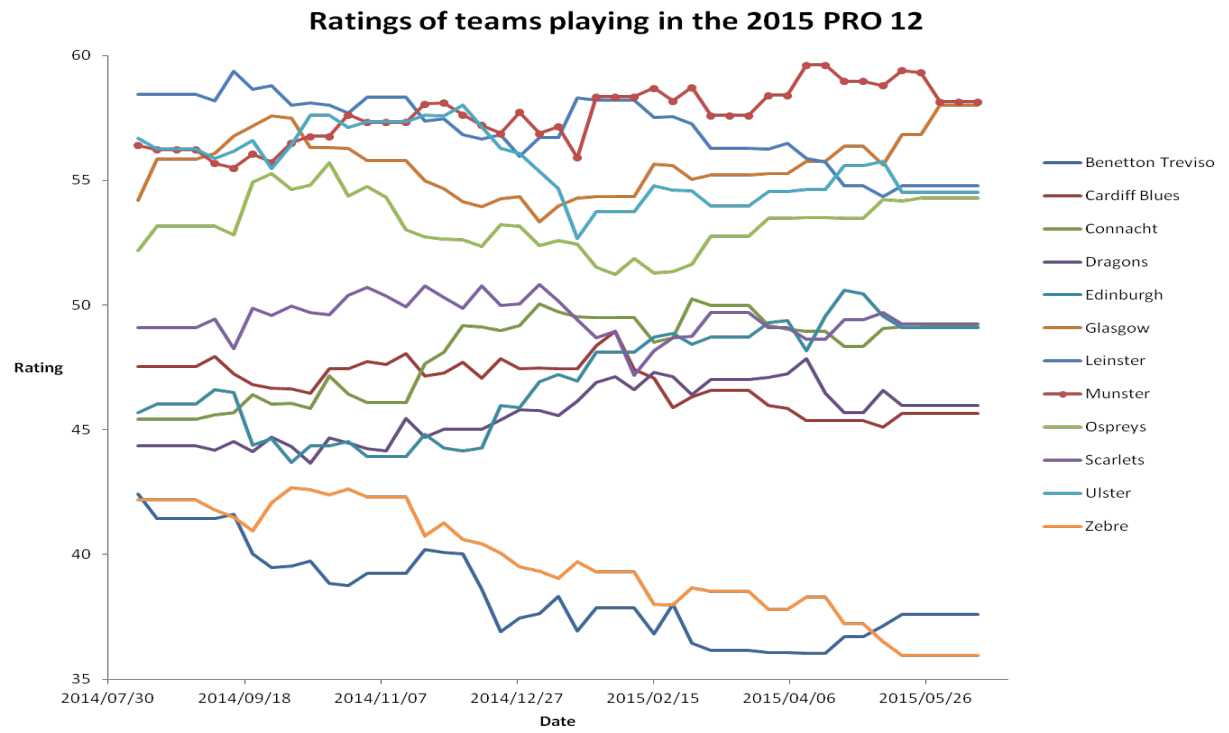


Figure 3: Ratings of teams playing in the 2015 PRO 12

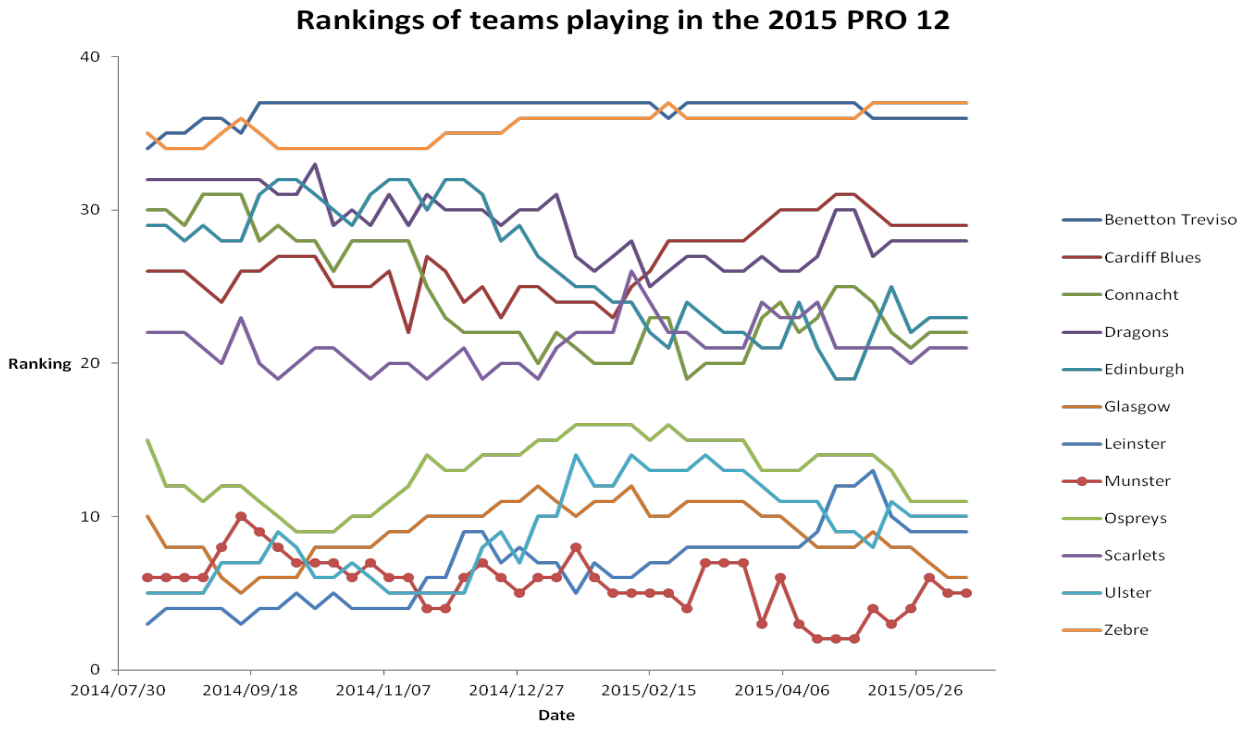


Figure 4: Rankings of teams playing in the 2015 PRO 12

6 Conclusion

In this report an accurate, unbiased and consistent rating system for rugby teams across multiple leagues was constructed and tested by calculating win percentages and correlations for the home team. The system has a good accuracy, which is around 66%, as seen from table 3, and the accuracy is even higher when a team has a bigger rating gap over the other team when predicting a winner for the match. There are also more possible ways to improve the system in the future and to enhance the prediction capabilities of the system. For example allowing an adjustment for a team touring far away from home and for a long duration, or even looking at upwards and downwards trends when examining the movement of a particular teams rating points. This can lead to a system with more capabilities in the future.

References

[1] W. H. Batchelder and N. J. Bershad. The statistical analysis of a Thurstonian model for rating chess players. *Journal of Mathematical Psychology*, 19(1):39–60, 1979.

[2] C. M. Borror, D. C. Montgomery, and G. C. Runger. Robustness of the EWMA control chart to non-normality. *Journal of Quality Technology*, 31(3):309–316, 1999.

[3] R. G. Brown. *Statistical Forecasting for Inventory Control*. McGraw/Hill, New York, 1959.

[4] R. G. Brown, R. F. Meyer, and D. D'Esopo. The Fundamental Theorem of Exponential Smoothing. *Operations Research*, 9(5):673–687, 1961.

[5] S. R. Clarke. Computer forecasting of Australian Rules Football for a daily newspaper. *Journal of the Operational Research Society*, 44(8):753–759, 1993.

[6] E. S. Gardner. Exponential smoothing: The state of the art – Part II. *International Journal of Forecasting*, 22(4):637–666, 2006.

[7] S. Ray. The methodology of officially recognized international sports rating systems. *Journal of Quantitative Analysis in Sports*, 7(4):1–22, 2011.

[8] S. H. Steiner. Exponentially weighted moving average control charts with time-varying control limits and fast initial response. *Journal of Quality Technology*, 31(1):75–86, 1999.

Appendix

When running the program, all the consecutive seasons have to be run in the correct order since any initial ratings are dependent on the last ratings of the preceding season. For this program only the input data set has to be changed each time, in order to calculate all the ratings.

```
proc iml;
   */*************************************************************************/;
   */***** Step 1 *****/*;
   */ Reading In */;
   use RATING.S14152 ;
   read all var {MatchNo} into MNo;
   read all var {Date} into DATE;
   read all var {Matchtype} into Matchtype;
   read all var {HomeTeam} into HT;
   read all var {AwayTeam} into AT;
   read all var {HomeTeamScore} into HP;
   read all var {AwayTeamScore} into AP;
   read all var {League} into League;
   C = nrow(DATE); */ Total Number Of Matches Played */;
   */*************************************************************************/;
   */ Sorting out unwanted teams */;
   */*************************************************************************/;
   */***** Step 2 *****/*;
   */ Values */;
   mdiff = 50;
   UL = 100; */ Upper Limit */;
   EW = 10; */ Exponental Weight */;
   DB = 10; */ Devide by (LEAGUE POINT = -1 TO 1: DIVIDE BY 10) */;
   AdP = 1; */ Add Point (LEAGUE POINT = 0 TO 2: ADD 1) */;
   ADJ = DB/10; */ To rebalance ratios */;
   AvP = 50; */ Average Point (all clubs) */;
   Mul = 5; */ Multiplier */;
   nTeam = 38; */ Number of Teams */;
   Check= 'y'; */ To check if we incorparate the teams
   relative strength while rating (y/n) */;
   Alpha = EW/100;
   res = 1-alpha;
   NewCAvp = 40;
   */*************************************************************************/;
   Allteams = HT//AT;
   u = unique(Allteams);
   Heading = { 'DATE' };
   Heading = Heading||u;
   */*************************************************************************/;
   */***** Step 3 *****/*;
   */ Starting Values */;
   StartTemp = J(1,nteam,.);
   use RATING.rating14 ;
   read all into RatingOLD;
   close RATING.rating14;
   use Rating.TEAMS14 ;
   read ALL VAR _CHAR_ into headingOLD;
```

```
close Rating.TEAMS14;
headingOLD = "DATE"|| headingOLD;
lastrating = RatingOLD[nrow(RatingOLD),];
do i=2 to ncol(heading);
do j=1 to ncol(HeadingOLD);
tempName1 = Heading[1,i];
tempName2 = HeadingOld[1,j];
if tempName1=tempName2
then Starttemp[1,i-1]=lastRating[1,j];
end;
end;
*/Print lastrating[colname=headingOLD];
*/print starttemp[colname=u];
*/*************************************************************************/;
nNewTeam = 0;
do i = 1 to nteam;
if StartTemp[1,i] = . then nNewTeam=nNewTeam +1;
end;
nEstTeam= nTeam - nNewTeam;
*/***** Step 4 *****/*;
EstCAvp = (nTeam*AVP-nNewTeam*NewCAvp)/nEstTeam;
*/*************************************************************************/;
*/***** Step 5 *****/*;
*/ CALCULATION (Standerdizing Starting Values around Average Point)*/;
ave = StartTEMP[,:];
varsum = 0;
ncount = 0;
do i = 1 to nTeam;
if StartTemp[1,i] ^= . then;
do;
varsum = varsum + ((StartTemp[1,i]-ave)*(StartTemp[1,i]-ave));
ncount = ncount+1;
end;
end;
if ncount = 0 then;
do;
ncount=nteam;
stddev = 1;
end;
else;
do;
var = varsum/(ncount-1);
stddev = sqrt(var);
end;
StartRank = J(1,nTeam,0);
if stddev = 0 then;
stddev = 1;
else;
do i=1 to nTeam;
StartRank[1,i] = MUL*(StartTemp[1,i]-Ave)/stddev +EstCAvp;
end;
do i = 1 to nteam;
if StartTemp[1,i] = . then StartRank[1,i]= NewCAvp;
```

```
end;
*/*************************************************************************/;
*/***** Step 6 Part a *****/*;
*/ CALCULATION (Standerdizing Match SCORES around Average Point)*/;
pDiff = J(c,1,.);
pDiff = HP-AP;
HPADJ = J(c,1,.);
HPADJ = HP;
APADJ = J(c,1,.);
APADJ = AP;
do i = 1 to c;
if pDiff[i,1]> Mdiff then HPADJ[i,1] = AP[i,1]+Mdiff;
if pDiff[i,1]< (-1)*(Mdiff) then APADJ[i,1] = HP[i,1]+Mdiff;
end;
*/***** Step 6 Part b *****/*;
ScoreTempa=J(c,2,.);
MPT = HPADJ+APADJ; */ Match Points Total */;
do i=1 to c;
ScoreTempa[i,1] = ((UL-MPT[i,1])/2)+HPADJ[i,1];
ScoreTempa[i,2] = ((UL-MPT[i,1])/2)+APADJ[i,1];
end;
*/*************************************************************************/;
*/***** Step 7 *****/*;
SCORES = J(c,nTeam,0);
NAME = J(c,2,.);
d = 2*c+1;
Playing = J(d,2,.);
do i = 1 to c;
t1 = 0;
t2 = 0;
do j = 1 to ncol(U);
if HT[i,1] = u[1,j] then t1=j;
end;
do k = 1 to ncol(U);
if AT[i,1] = u[1,k] then t2=k;
end;
SCORES[i,t1]=ScoreTempa[i,1];
SCORES[i,t2]=ScoreTempa[i,2]; */ Note: using standerdized scores */;
NAME[i,1] = t1;
NAME[i,2] = t2;
Playing[2*i-1,1]= name[i,1];
Playing[2*i,1] = name[i,2];
Playing[2*i-1,2]= League[i,1];
Playing[2*i,2] = League[i,1];
end;
*/*************************************************************************/;
*/***** Step 8 *****/*;
*/ Ranking Update Dates */;
aDate = Date[1,1];
eDate = Date[c,1];
bDate = intnx('week', aDate, 0,'same');
*/ ddate = last Sunday before the first Matches */;
dDate = intnx('week', bDate, 0,'B');
```

```
if aDate=dDate then dDate = intnx('week', bDate, -1,'B');
*/ gdate = first Sunday after the last Matches */;
gDate = intnx('week', eDate, +1,'B');
*/ fdate = total number of weeks */;
fDate = (gDate-dDate)/7;
RankDates=dDate;
do i=1 to fdate;
date1 = intnx('week', dDate , i,'same');
RankDates = RankDates//date1;
end;
/*print adate[format=date9.] edate[format=date9.] RankDates[format=date9.];*/
**/*************************************************************************/;
*/***** Step 9 *****/*;
*/ Find out which match is in which week */;
WD = J(c,fdate+1,.);
do i=1 to fdate+1;
do j=1 to c;
WD[j,i] = RankDates[i,1];
end;
end;
MD = Date;
Do i=2 to fdate+1;
MD = MD||date;
end;
DateTemp = MD-WD;
WeekFit =J(c,1,.);
do i=1 to fdate+1;
do k = 1 to c;
if DateTemp[k,i] <= 0 & DateTemp[k,i] > -7
then weekFit[k,1]= i-1;
end;
end;
nWeeks=max(WeekFit); */ CALCULATING THE NUMBER OF WEEKS
THAT WE ARE CALCULATING RATINGS FOR */;
*/*************************************************************************/;
*/***** Step 10 *****/*;
t = gdate-ddate; */ Last Rank Date Up To First */;
tempD= J(t+1,1,.);
indD = J(t+1,1,.);
DMC = 0; */ Daily Match Count */;
do i = 1 to t+1;
tempd[i,1]= dDate+i-1; */ ddate - first ranking sunday */;
end;
do i = 1 to t+1;
DMC = 0;
do j = 1 to c;
if tempD[i,1]= date[j,1] then DMC= DMC +1;
end;
indD[i,1]= DMC; */ Number of matches on every date,
from the first rank date up to the last */;
end;
*/*************************************************************************/;
count = 1;
```

```
vec1 = J(nWeeks+1,1,.);
do i=1 to nWeeks+1;
aa1 = dDate;
bb1 = RankDates[i,1];
do j = 1 to c;
cc1 = Date[j,1];
dif1 = bb1-cc1;
*/ * * * * * * * * * * /*;
if (dif1 >=0) & (dif1<7) then count = count+1;
end;
vec1[i,1] = count;
end;
*/*************************************************************************/;
*/*************************************************************************/;
*/***** Step 11 *****/*;
RANKtemp1 = J(c,nTeam,.);
RANKtemp2 = J(1,nTeam,.);
RANKtemp3 = J(1,nTeam,.);
ScoreTemp1 = J(1,nTeam,.);
ScoreTemp2 = J(1,nTeam,0);
ScoreTemp = J(1,nTeam,0);
GameSt = J(1,2,.);
Rank1 = J(t+1,nTeam,.);
RankTemp1 = StartRank//RankTemp1;
Countout=0;
sumprod=0;
sum1=0;
sum2=0;
testv = 16;
countmatrix = J((testv+1),3,0);
do i = 0 to (testv);
countmatrix[i+1,1]=i;
end;
do i = 1 to c;
RankTemp2[1,] = RankTemp1[i,];
*/ * * * * * * * * * * /*;
ScoreTemp1[1,] = RankTemp1[i,];
ScoreTemp1[1,NAME[i,1]] = 0;
ScoreTemp1[1,NAME[i,2]] = 0;
MatchSt = 1;
If Check = 'y' then;
do;
GameSt[1,1] = RankTemp2[1,NAME[i,1]];
GameSt[1,2] = RankTemp2[1,NAME[i,2]];
MatchSt = (GameSt[1,1]+GameSt[1,2])/UL;
end;
else;
if Check = 'n' then MatchSt=1;
ScoreTemp2[1,] = SCORES[i,]*MatchSt;
ScoreTemp[1,] = ScoreTemp1[1,]+ScoreTemp2[1,];
*/ * * * * * * * * * * /*;
RankTemp3[1,] = Alpha*ScoreTemp[1,] + (1-Alpha)*RankTemp2[1,];
RankTemp1[i+1,] = RankTemp3;
```

```
*/*****************************************************************/;
*/* Outcome Measure */*;
HTRankNet = GameSt[1,1]-GameSt[1,2];
HTScoreNet = Scores[i,Name[i,1]]-Scores[i,Name[i,2]];
if HTRankNet >=0 && HTScoreNet >=0 then Countout=Countout+1;
if HTRankNet <0 && HTScoreNet <0 then Countout=Countout+1;
do j=0 to testv;
if HTRankNet >= j then countmatrix[j+1,2]=countmatrix[j+1,2]+1 ;
if HTRankNet <(-1)*j then countmatrix[j+1,2]=countmatrix[j+1,2]+1;
if HTRankNet >= j && HTScoreNet >=0 then countmatrix[j+1,3]=countmatrix[j+1,3]+1;
if HTRankNet <(-1)*j && HTScoreNet <0 then countmatrix[j+1,3]=countmatrix[j+1,3]+1;
end;
sumprod = sumprod + HTRankNet*HTScoreNet;
sum1 = sum1+ HTRankNet*HtRankNet;
sum2 = sum2+ HTScoreNet*HtScoreNet;
*/*****************************************************************/;
end;
ratio1415 = J(testv,2,.);
do i = 1 to testv;
ratio1415[i,1] = i-1;
ratio1415[i,2] = countmatrix[i,3]*(100/(countmatrix[i,2]));
end;
corr1415 = sumprod/(sqrt(sum1)*sqrt(sum2));
/*PRINT ALPHA NewCAvp;*/
/*print ratio1415 corr1415;*/
*/*****************************************************************/;
*/***** Step 12 *****/*;
Rating = J(nWeeks,nTeam,.);
Rating = StartRank//Rating;
Do k = 2 to nWeeks+1;
a = vec1[k,1];
Rating[k,]= ranktemp1[a,];
end;
ave = Rating[,:];
*/ print RankDates[format=date9.] Rating[colname=u];
*/*****************************************************************/;
*/***** Step 13 *****/*;
*/ Ranking */;
RANKS = RANKtie(-Rating[1,]);
do i = 2 to nWeeks+1;
RANKSweek = ranktie(-Rating[i,]);
RankS = Ranks//RANKSweek;
end;
RankS = RankDates||RankS;
Rating = RankDates||Rating;
*/ print RankDates[format=date9.] RANKS[colname=Heading];
*/*****************************************************************/;
create Rating.Rating15 from Rating [colname=Heading];
Append from Rating;
close Rating.Rating15;
create Rating.TEAMS15 from U[colname=U];
Append from U;
close Rating.TEAMS15;
```

```
create Rating.Ranks15 from Ranks [colname=Heading];
Append from Ranks;
close Rating.Ranks15;
quit;
PROC EXPORT DATA= RATING.RATING15
OUTFILE= "C:\Users\Rion\Google Drive\Hons\Research Report\DA
TA\Ratings Out 15.xls"
DBMS=EXCEL LABEL REPLACE;
NEWFILE=YES;
RUN;
PROC EXPORT DATA= RATING.Ranks15
OUTFILE= "C:\Users\Rion\Google Drive\Hons\Research Report\DA
TA\Ranks Out 15.xls"
DBMS=EXCEL LABEL REPLACE;
NEWFILE=YES;
RUN;
```

Random walk on a clock

Lerato Langa 12244679

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mr TM Loots

Department of Statistics, University of Pretoria

2 November 2015

Abstract

This report is based on the study of a random walk on a clock, with the aim of answering the following questions. How long it will take this random walk to occupy each state at least once. As well as identifying which of these states will be visited the most on average. The states are denoted by the numbers $1, 2, \ldots, 12$ representing the position on the clock. In this report a number of methods will be proposed to obtain a solution to the two questions posed.

Declaration

I, *Lerato Moshidi Langa,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Lerato Moshidi Langa

_____
Mr Theodor Loots

_____
Date

Acknowledgements

I would like to acknowledge my supervisor Mr TM Loots for his guidance and support during the compilation of this research report.

Contents

List of Figures

List of Tables

1 Introduction

The random walk model can be used to model a variety of real-world problems. Examples of these include the modelling of gambling problems, stock prices and migration of animals. In particular this report will investigate the study of random walks on a clock. A number of questions could be posed in this study, examples of these can be found in the following blog: [8]. These questions assist us in understanding the behaviour of the particular random walk process. From [1] the following question was posed. "Stand on a large clock, say on 1. Now flip a coin and move ahead one hour if the coin turns up heads, and back one hour otherwise. Keep repeating the process until you have stood on all 12 numbers. (1) How long on average did this random walk take? Furthermore, (2) which state(s) will be visited the most? Now if we generalize to clocks with p positions, how does the expected time vary?" A number of methods will be presented as possible ways to answer both of the questions posed. This process could be simply interpreted as a game with p equally spaced points around a circle. Now suppose there are five players. Each player should place a chip on this clock, after 50 steps the player whose point was visited the most within those 50 steps wins a prize of R1. How would a player choose which point to place his/her chip given a specified starting point? Would this starting point even have an effect on what value is observed the most? This research report aims to come up with solutions to those questions, with the relevant mathematical and/or theoretical backing. In some instances theoretical results will be used and other cases simulations will be used to model the given process as well as combinations of these. The reader should note that in the case of this report we will focus on a 12 state clock. This report will look into simulations of the random walk, Markov chains as well as directional statistics to answer both questions.

2 Simple random walk

2.1 Background Theory

According to [7] a random walk process is a non-stationary time-series process defined by

$$Z_t = Z_{t-1} + \epsilon_t$$

where ϵ_t denotes a white noise process with mean, $\mu_\epsilon = 0$ and variance $\sigma_\epsilon^2 = 1$ and Z_t denotes the position on the clock at time t. The value of the error terms is given by $\epsilon_t = 1$ or $\epsilon_t = -1$, depending on whether the clock is moving clockwise or anti-clockwise. This process can be simplified to

$$Z_t = Z_0 + \sum_{t=1}^{n} \epsilon_t$$

i.e. the position of the clock at time t is given by the sum of the starting position and a simple random walk. From this it is clear that the expected value of Z_t is given by Z_0. This process will generate the described clock setting. In order answer (1) proofs from [1] and [5] were considered. From [1] the following result was given:

Let $S_0 = 0$ and let $S_k = X_1 + \ldots + X_k$ for $k > 0$ where X_i are independent, identically distributed variables with $X_i = 1$ and $X_i = -1$ each with probability $\frac{1}{2}$. Let $t_n = min\{k : N = n, n \leq p\}$ be the first time that we have visited n distinct states then

$$E(t_p) = \frac{p(p-1)}{2}$$

Proof

Let t =min$\{k : S_k = -A$ or $S_k = B\}$ for some fixed integers $A, B > 0$. Let V_k = the collection of all visited states up to time k. Let N =the number of elements in V_k. It can be seen that $t_1 = 0$, and $t_2 = 1$. A key realisation is that one can easily get from t_n to t_{n+1}. Now consider the situation at time t_n for $n < p$. Suppose that n states have been visited and we are at one extreme of $V_t(n)$. If we consider the case where say we are at 0 and have visited positions$1, 2, \ldots, n-1$. In order to calculate t_{n+1}–t_n we want to know the

first time we hit either -1 or n (for $n = p$, these are the same vertex). Therefore from fact stated above, it follows that: $E(t_{n+1}–t_n) = n$ for $n < p$

This yields the following: $E(t_n) = 1 + \ldots + (n-1) = n(n-1)/2$ for $n \leq p$

In particular: $E(t_p) = p(p-1)/2$

From this proof it can be said that the average time it takes to occupy each state at least once is given by where p denotes the number of finite states. The aim is to back up this theoretical result with the use of simulations. In a similar manner the second question will also be answered. The effect that different starting times has on which state is visited the most will be investigated as well as whether that effect will affect the behaviour of the process in the long run.

2.2 Application

Firstly we will look at the 12 step clock starting at position 12. A thousand random values are generated to model the process. A histogram is plotted and compared with a normal curve to try and fit an appropriate distribution. The sample size is then made large. It is found that the process converges to a uniform distribution i.e. if the process is performed for long enough it is equally likely to observe each of the 12 states. The effect of the starting point wears off. In this section both (1) and (2) will be answered through simulations. Let us consider (1), for this the random walk is generated as well as a counter vector for each position on the clock. The counter counts how many times each of points on the clock are visited. The counter will stop when the counter value for each state is greater than one. The sum of the elements of the counter vector gives the number steps it took for all 12 states to be visited. This process can be repeated 1000 times each time recording the number of steps taken to visit all the states. An estimate of the average number of steps can be found from this. The result of this is given below. Secondly one must investigate which of the states will be observed the most for a fixed n steps. A Program_2 was written to find the state that is most frequently visited for $n = 150$ at three different starting points. The results from the seed value of 10 are presented below.

State	$Z_0 = 3$	$Z_0 = 6$	$Z_0 = 12$
	Frequency	**Frequency**	**Frequency**
1	10	8	22
2	20	6	21
3	21	5	16
4	22	10	10
5	21	20	4
6	16	21	7
7	10	22	8
8	4	21	6
9	7	16	5
10	8	10	10
11	6	4	20
12	5	7	21

Table 1: Frequencies of all 12 states for different starting values

From the table above it was observed that for $Z_0 = 3$ and $Z_0 = 12$ that the states the 2 states above and below the starting value occurs the most. However for $Z_0 = 6$ this is not the case, in this instance the values 10 and 11 occur the most. These values are not 2 states from the starting value Z_0, the same result was found for other simulations with starting points 3 and 12. So which state(s) will be visited the most? In order to find the state(s) visited the most a histogram was fitted to the given frequency distributions and compared with the normal, exponential, gamma, weibull and lognormal distribution curves with the aim of fitting a known distribution to the frequency distribution. The result of this is given below:

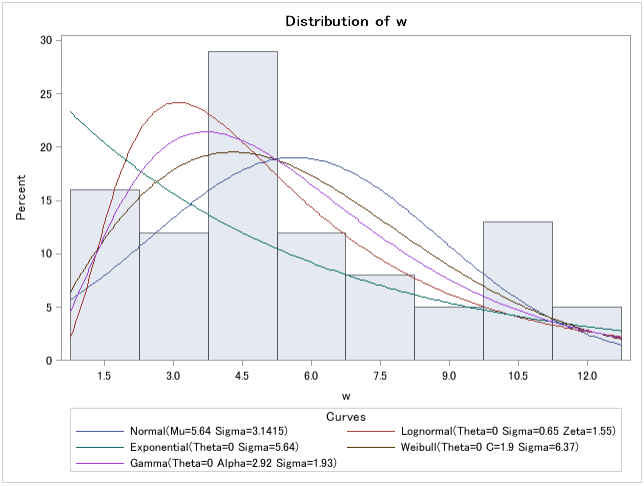


Figure 1: Frequency distribution for $Z_0 = 3$

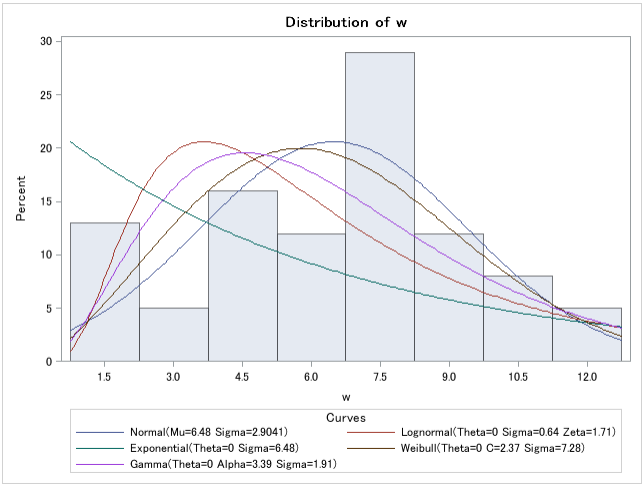


Figure 2: Frequency distribution for $Z_0 = 6$

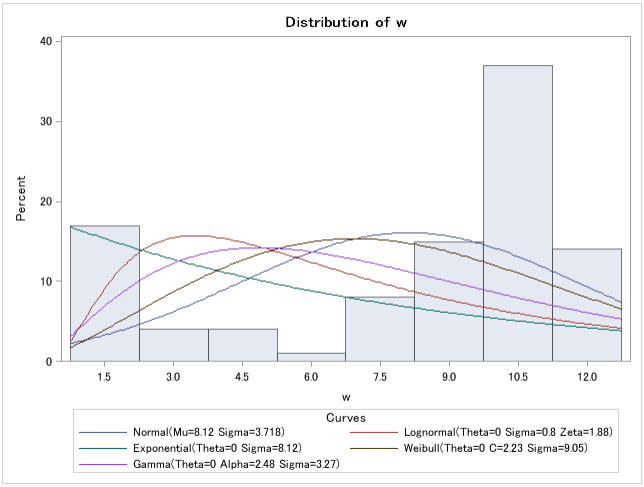


Figure 3: Frequency distribution for $Z_0 = 12$

The histograms show that the distribution of the frequency does not follow any of the known distributions well enough for $n = 150$. Therefore an alternative method could be used to fit a distribution to the frequencies.

3 Markov chain approach

3.1 Background Theory

Markov chains are stochastic models which describe a sequence of probable events, which depend only on the previous event attained. For some discrete state space S, the transition probability matrix is defined as follows

$$P(i,j) = P[Z = j | Z_{n-1} = i], i, j \in S$$

The random walk on a clock can be described by a transition probability matrix with state space $S = \{1, 2, 3, \ldots, 12\}$. We will consider the transition probability matrix generated by this random walk. The described matrix is dependent only on the previous state visited. Markov chain theory presented in [5] will be used to determine the behaviour of this process. The study of first passage times as well as limiting distributions will be considered to determine the long and short term behaviour of the Markov chain. The first passage time can be defined as the number of steps taken for a process to go from state i to state j. Let F_{ij} denote the first passage time of the transition from i to j then its distribution is denoted by $f_{ij}^{(n)}$where

$$f_{ij}^{(n)} = P[Z_n = j, X_r \neq j, r = 1, 2, \ldots, n-1 | Z_0 = i] = P[F_{ij} = n]$$

where n denotes the number of steps taken. This definition can be used to answer (1). As the average amount of time taken to visit each state at least once can be viewed as a sum of first passage times for a given fixed value of the state space S. Moving to the second question, limiting distributions will be used to see which state is visited the most in the long run. The following condition should be met to find the limiting distribution of a Markov chain. The transition probability matrix P must be as follows:

1. Aperiodic i.e. have a period of one where the period is defined as $d_i = gcd\{n : n \geq 1, p_{ij}^{(n)} > 0\}$ where $p_{ij}^{(n)}$ is the $ij - th$ term of P^n and $n \in Z$.
2. Irreducible i.e have one equivalence class.
3. Have a finite number of states.

3.2 Application

The 12-state random walk on a clock is represented by the following transition probability matrix

$$P = \begin{bmatrix} 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 \\ 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 \\ 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \end{bmatrix}$$

The transition probability matrix has a finite number of states , is irreducible but has a period of 2 and hence is not aperiodic. Therefore a limiting distribution cannot be found for this Markov chain. Thus this method cannot be used to answer (2). However [2] developed theory that P should be constructed in a different manner based on the fact that the probability of observing a head or tail in a balanced coin toss is not exactly 0.5. The argument is based on an element of randomness that affects the coin toss. For example one could toss a coin and it lands right on the side and neither a head or tail will be observed, hence there arguably does exist a very small probability of remaining in the same state based on this argument. The new transition probability matrix of this process is now defined by:

$$P_{new} = \begin{bmatrix} 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 \\ 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 & 0.4995 \\ 0.4995 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4995 & 0.001 \end{bmatrix}$$

This new transition probability matrix is aperiodic, irreducible and has a finite number of states thus a limiting distribution can be calculated for the newly defined process. The limiting distribution,Π is defined as $\Pi = \lim_{n\to\infty} P^n$. Using the SAS simulation provided in Program_1 the limiting distribution was found to be as follows:

$$\Pi = \begin{bmatrix} \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \end{bmatrix}$$

where $\frac{1}{12} : 12 \times 1$ vector with all elements equal to $\frac{1}{12}$. The limiting distribution says that as the process is performed an infinite number of times each state will be equally likely to occur. This in effect means that the process converges to a discrete uniform distribution over the interval $[1, 12]$, hence the limiting distribution has a discrete uniform distribution.

4 Directional Statistics

4.1 Background Theory

According to [6], directional statistics is mainly concerned with statistics on a unit vectors in a plane or three dimensional spaces. Thus samples are usually taken from circles or spheres. Circular data is usually collected from a clock or compass. This data is analysed differently than data on a real line. For instance, if two boats are traveling at 350 and 10 degrees respectively. What is the average direction these boats are traveling? When calculating this mean arithmetically, it is found that on average these boats are traveling

Direction (in degrees) clockwise from north									
8	9	13	13	14	18	22	27	30	34
38	38	40	44	45	47	48	48	48	48
50	53	56	57	58	58	61	63	64	64
64	65	65	68	70	73	78	78	78	83
83	88	88	88	90	92	92	93	95	96
98	100	103	106	113	118	138	153	153	155
204	215	223	226	237	238	243	244	250	251
257	268	285	319	343	350				

Figure 4: Orientation of 76 female turtles after laying eggs

at 180 degrees which intuitively doesn't make sense. Why would two forward directions result in a backward direction? Directional statistics assists in solving questions of this nature by considering the impact the circular shape has on data. This section only looks at question (2), we will be considering the forms of frequency distributions and the properties of these.

4.1.1 Frequency Distributions

Unimodal Distributions In answering (2) one is basically looking at the number which occurs most frequently, this is also known as the mode. In directional statistics the mode is seen as the direction visited the most. Unimodal distributions refer to distributions with only a single mode.

Multimodal Distributions Multimodal distributions occur when a distribution has two or more modes. However it is rare to find more than two modes. An example of such is found 4 of [6]. It shows the direction that 76 female turtles moved after laying their eggs on the beach. The circular plot of this shows that there is a dominant mode and a subsidiary mode. That shows the turtles have a preferred direction but a minority of the group prefers another direction. Determining whether the distribution on the clock is unimodal or multimodal can be used to give direction on whether the process can be fitted to a known discrete circular distribution.

4.1.2 Discrete Circular Distributions

Lattice distribution From [6] consider a discrete distribution with

$$P[\theta = v + \frac{2\pi r}{m}] = p_r, r = 0, 1, \ldots, m-1$$

and

$$p_r \geq 0, \sum_{r=0}^{m-1} p_r = 1.$$

The points $v + \frac{2\pi r}{m}$ are the vertices of an m sided regular polygon inscribed in the unit circle. If all the weights are equal then

$$p_r = \frac{1}{m}.$$

This is referred to as the discrete uniform distribution. If $m = 37$ this represents a roulette table described in [3].

Wrapped distributions As defined in [6] a wrapped distribution is given by wrapping a given distribution on a line onto a circle of unit radius. That is for a random variable Z on a line, the corresponding wrapped distribution is given by

$$Z_w = Z(mod 2\pi).$$

Therefore if Z has a distribution function F then distribution function of Z_w is given by F_w where

$$F_W(\theta) = \sum_{k=-\infty}^{\infty} \{F(\theta + 2\pi k) - F(2\pi k)\},$$

$$\text{for } 0 \leq \theta \leq 2\pi.$$

In particular if Z has a density function f then the corresponding probability density function $f_w$of $Z_w$is given by

$$f_w(\theta) = \sum_{k=-\infty}^{\infty} f(\theta + 2\tau k).$$

The results above hold for discrete random variables.

4.2 Application

The random walk process will be generated in the same way described previously, but in this case the state numbers will be replaced by equally spaced angles, spaced $\frac{\pi}{6}$ radians apart. The clock is given in the figure below

Figure 5: Clock in radians

In order to answer the first question we consider the same random walk process just with the states replaced by radians. In the case of Program_3 degrees were used instead of radians for interpretation purposes. The aim of this section is to represent the data as circular data and fit the appropriate distribution as well as explain the convergence to a discrete uniform distribution found in the initial simulation of the process. Program_3 generates the same frequency distributions as those given in 12 and 3 as the same data is used just relabeled. Hence results similar as 1 are obtained .

5 Conclusion

In this report the following topics: Markov chains, random walk simulations and directional statistics were discussed in order to solve the two questions posed in the introduction. The Markov chain method proved

to be successful in doing this for (1) but presented limitations in answering (2) as a limiting distribution could not be found for the transition probability matrix that represents the process. However through the adaptation of the initial transition probability matrix it was found that the process converges to a discrete uniform distribution. Simulations of the random walk process resulted in the same realisation . Hence it can be concluded that as the process is repeated an infinite amount of times the effect of the starting point dies out and the probability of visiting each state is equal. To answer question (1) the generated simulations were compared to theoretical results given from [4] as well as the proofs presented on [1]. Simulations show that the average amount of steps it takes to visit each state at least once tends to $\frac{p(p-1)}{2}$ which was the result obtained theoretically. In the case of this report the average amount of steps to visit all the states at least once is approximately given by $\frac{12(11)}{2} = 66$. Through simulations it was found that the average numbers of steps ranges in the interval $[68, 75]$. This differs slightly from the theoretical result of 66 but is still close to the lower limit of the interval hence the average number of steps taken could be represented as $\frac{p(p-1)}{2} + r$, where $r \in (0, 10)$ and is due to the randomness of generating the process. Although this report has managed to answer (1) successfully, (2) was not answered fully and additional methods could be investigated to fully answer (2). The reader could also consider a clock with an arbitrary k number of states and the effect of this on the process.

References

[1] John. D Cook. Random walk on a clock, September 2013. Blog[Online] www.johndcook.com/blog/2013/09/30/random-walk-on-a-clock/. Accessed: 21-04-21.

[2] Phillip I Good. *Introduction to Statistics Through Resampling and R.* John Wiley & Sons, 2 edition, 2013.

[3] James M Hill and Chandra M Gulati. The random walk associated with the game of roulette. *Journal of Applied Probability*, 18(4):931–936, December 1981.

[4] Oliver Knill. *Probability Theory and Stochastic Processes with Applications.* Overseas Press India Private Limited, 2009.

[5] Gregory F. Lawlier. *Introduction to Stochastic Processes.* Chapman & Hall/CRC, 2006.

[6] Kanti V Mardia and Peter E Jupp. *Directional Statistics*, volume 494. John Wiley & Sons, 2009.

[7] Brockwell PJ and Davis RA. *Introduction to Time Series and Forecasting.* Springer Series in Statistics, 2002.

[8] Micheal Shiwdawu. Random walk on a clock, February 2014. Blog [Online] http://michaelshidawu.com/?p=55. Accessed: 21-04-2015.

Appendix

Program_1

```
/*Calculating the limiting distribution*/
proc iml;
n=1000000;

/*transition probability matrix*/
p={ 0.001 .4995 0 0 0 0 0 0 0 0 0 .4995,
    .4995 0.001 .4995 0 0 0 0 0 0 0 0 0,
    0  .4995 0.001 .4995 0 0 0 0 0 0 0 0,
0  0 .4995 0.001 .4995 0 0 0 0 0 0 0,
0  0 0 .4995 0.001 .4995 0 0 0 0 0 0,
0  0 0 0 .4995 0.001 .4995 0 0 0 0 0,
0  0 0 0 0 .4995 0.001 .4995 0 0 0 0,
0  0 0 0 0 0 .4995 0.001 .4995 0 0 0,
0  0 0 0 0 0 0 .4995 0.001 .4995 0 0,
0  0 0 0 0 0 0 0 .4995 0.001 .4995 0,
0  0 0 0 0 0 0 0 0 .4995 0.001 .4995,
.4995 0 0 0 0 0 0 0 0 0 .4995 0.001}
;
/*Formula for limiting distribution*/
Pn=p**n;
print Pn;
quit;
```

Program_2

```
/*Finding the frequency distribution*/
proc iml;
n=150;
p=0.5;
seed=0;
b= j(n, 1, .);
w=j(n,1 ,.);
x=3;
do i=1 to n;
b[i]=ranuni(seed);
if  b[i]<0.5 then w[i]=x-1;
if b[i]>0.5 then w[i]=x+1;
if w[i] > 12 then w[i]=1;
if w[i]<1 then w[i]=12;
x=w[i] ;
end;

/*reading data from a vector into a data set*/
create new var{'w'};
append ;
close new;
quit;
proc univariate data=new;
var w;
```

```
histogram w/ normal
exponential
gamma
weibull
lognormal;
run;
proc freq data =new;
run;
```

Program_3

```
/*Calculating the average number of steps taken to visit each state at least once*/
proc iml;
it=10000;/*number of iterations*/
n=150;/*maximum number of random walk components*/
p=0.5;/*probability of moving either clockwise or anti-clockwise*/
b=j(n,1,.);
rw=j(n,1,.);/*random walk component*/
seed=0;
x=10; /*starting point*/
do i = 1 to it;
counter=j(12,1,0); /*generating points on the clock*/
do j=1 to n;
b[j]=ranuni(seed);
if b[j]<0.5 then rw[j]=x-1;
if b[j]>0.5 then rw[j]=x+1;
if rw[j]<1 then rw[j]=12;
if rw[j]>12 then rw[j]=1;

/*calculating frequency distribution*/
do k = 1 to 12 until(freq>=j(12,1,1));
count = j(12,1,0);
if rw[j]=k then count[k]=1;
counter=counter||count;
freq=counter[,+];
end;
x=rw[j];
end;
freq1=counter[,+];
sum=freq1[+];
sums=sums//sum;
average=(1/it)*sums[+];
end;
print average;
```

Program_4

```
proc iml;
n=150;
p=0.5;
b= j(n, 1, .);
w=j(n,1 ,.);
x=30;
```

```
do i=1 to n;
b[i]=ranuni(10);
if b[i]<0.5 then w[i]=x-30;
if b[i]>0.5 then w[i]=x+30;
if w[i] > 360 then w[i]=30;
if w[i]<30 then w[i]=360;
x=w[i] ;
end ;
print w;
create new var{'w'};
append ;
close new;
quit;
proc univariate data=new;
var w;
histogram w/ normal
exponential
gamma
weibull
lognormal;
run;
proc freq data =new;
run;
```

Penalized maximum likelihood bandwidth estimation

Lara-Jayne Lauryssen 12075834

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor: Mr M.T.Loots

Department of Statistics, University of Pretoria

02 November 2015

Abstract

In exponential families, it has been shown that the likelihood function may be penalized by Jeffreys invariant prior in reducing the bias of maximum likelihood estimators. This method will be described and then applied to the maximum likelihood bandwidth estimator assuming a Gaussian kernel, in a kernel density estimation setting.

Declaration

I, *Lara-Jayne Lauryssen,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Lara-Jayne Lauryssen

_____
Theodor Loots

_____
Date

Acknowledgments

This work is based on the research supported in part by the National Research Foundation (NRF) of South Africa for the grant, Unique Grant No. 94108. The financial assistance of the NRF towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF; which does not accept any liability in this regard.

Contents

List of Figures

List of Tables

1 Introduction

Penalized maximum likelihood estimation was developed by Firth in 1992 through to 1993, as a bias reduction method for maximum likelihood estimates in generalized linear models [7, 4]. In this report the properties of this method will be explored, to obtain finite estimates for θ using a modification of the score function [6]. In normal parametric problems, to modify the score function the first order term is removed from the asymptotic bias of the maximum likelihood estimates [4]. For smoothing curves this method is introduced and therefore the penalized maximum likelihood estimator is a smoothing parameter that approaches zero [3]. Penalization can be thought of as a technique that introduces some degree of tolerable bias, to achieve a smaller variability of parameter estimates, thus a smaller variance is achieved by introducing a degree of bias to the model [2].

For models with categorical predictors, penalization can be used as another way for finding estimates of these regression coefficients without fitting noise. This is a technique used for predicting models where adjustments for over fitting are directly incorporated into the model as opposed to shrinking it afterward [9]. Up until now the penalization approach has not often been used for empirical data, but does however lead to reduced prediction errors and may also cause shrinkage in individual predictors to differ for over optimism without forfeiting substantial discriminative precision of the model [9]. Penalized maximum likelihood estimation should not be confused with weighted maximum likelihood estimation, the reason being that for weighted maximum likelihood estimation every observation and not the individual predictors, are weighted depending on some available characteristics [5].

A direct relationship can be seen between the kernel density estimation and a specific penalization method of density estimation, for example roughness penalties [8]. For this specific penalty method the average is taken with respect to the bandwidth parameter and solutions can be categorised as weighted average Gaussian kernel density estimates [8]. As the degrees of freedom in a model get smaller the penalty factor increases, therefore the regression fit gets flatter and the confidence boundaries become narrower [5]. Penalization is a technique for avoiding complications with the stability of parameter estimates occurring when likelihood is reasonably flat. This therefore makes finding the maximum likelihood estimate difficult when using standard approaches [2].

The method of penalized maximum likelihood estimation is extensively used in epidemiology (where incidence, distribution and possible control of diseases and other factors relating to health are studied) [2]. In realistically sized epidemiological studies, sparse-data complications can be a result if data is classified as being flexible enough, which therefore requires the use of penalization, semi-parametric modelling or some blend of these methods which are more involved than standard maximum likelihood [2].

2 Basic Idea

A comparison of the standard maximum likelihood and penalized maximum likelihood estimation will be looked at here. With the standard maximum likelihood estimation procedure the joint density function is first specified for all observations, which is given by:

$$f(x_1, x_2, ...x_n|\theta) = f(x_1|\theta) \times f(x_2|\theta) \times ..... \times f(x_n|\theta)\cdot$$

The observed values $x_1, x_2, \ldots, x_n$ can be considered fixed parameters of this function where θ will be the functions parameter(s) and is to be estimated. The likelihood function is then given by:

$$\begin{aligned} L(\theta; x_1, ...x_n) &= f(x_1, x_n, ...x_n|\theta) \\ &= \prod_{i=1}^{n} f(x_i|\theta) \end{aligned}$$

The derivative of this likelihood function is calculated and set equal to zero to calculate the maximum likelihood estimator i.e. $\frac{\partial}{\partial\theta} L(\theta) = 0$. Any value of θ that maximises $L(\theta)$ will also maximise the log-likelihood, $\ln L(\theta) = l(\theta)$ also known as the score function. Therefore maximum likelihood estimates are obtained by

solving the following score equation

$$\frac{\partial l(\theta)}{\partial \theta} \equiv U(\theta) = 0$$

This approach is used as it is usually the desired method of calculating. The small bias of these estimates is as a result of the combined effect of curvature and unbiasedness of the score function [6].

In penalized maximum likelihood, involving parameters of the exponential family, the penalized likelihood function is given by:

$$L(\theta)^* = L(\theta)|I(\theta)|^{0.5}.$$

Therefore the penalized log likelihood is equal to

$$l^*(\theta) = l(\theta) + 0.5 \ln |I(\theta)| \tag{1}$$

where the penalty function is $|I(\theta)|^{0.5}$ and is known as Jeffreys invariant prior, where $I(\theta)$ is the Fisher information matrix which can be calculated as the variance of the score function. As an example the Fisher information matrix will be derived for the normal distribution. The PDF of the normal distribution is given by

$$f(x|\mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

To be able to calculate the Fisher information matrix for a univariate distribution we first need to find the logarithm which is

$$\ln f(x|\mu, \sigma^2)) = -\frac{1}{2}\ln(2\pi) - \frac{1}{2}\ln\sigma^2 - \frac{1}{2\sigma^2}(x-\mu)^2$$

then the partial derivatives need to be calculated

$$\begin{aligned}
\frac{\partial}{\partial \mu} \ln f(x|\mu, \sigma^2) &= \frac{1}{\sigma^2}(x-\mu), \\
\frac{\partial}{\partial \sigma^2} \ln f(x|\mu, \sigma^2) &= -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}(x-\mu)^2.
\end{aligned}$$

The Fisher score is defined as

$$\begin{aligned}
U(\mu) &= \frac{\partial \ln L(\mu, \sigma^2; x)}{\partial \mu} \\
U(\sigma^2) &= \frac{\partial \ln L(\mu, \sigma^2; x)}{\partial \sigma^2}
\end{aligned}$$

for the normal distribution, this becomes

$$g(\mu, \sigma^2; x) = \begin{pmatrix} \frac{1}{\sigma^2}(x-\mu) \\ -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}(x-\mu)^2 \end{pmatrix}.$$

The product of the Fisher score and its transposition is given as

$$\begin{pmatrix} \frac{1}{\sigma^2}(x-\mu) \\ -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}(x-\mu)^2 \end{pmatrix} \left(\frac{1}{\sigma^2}(x-\mu) - \frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}(x-\mu)^2 \right) =$$

$$\begin{pmatrix} \frac{1}{\sigma^4}(x-\mu) & -\frac{1}{2\sigma^4}(x-\mu) + \frac{1}{2\sigma^6}(x-\mu)^3 \\ -\frac{1}{2\sigma^4}(x-\mu) + \frac{1}{2\sigma^6}(x-\mu)^3 & \frac{1}{4\sigma^4} - \frac{1}{2\sigma^6}(x-\mu)^2 + \frac{1}{4\sigma^8}(x-\mu)^4 \end{pmatrix} =$$

$$\begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix}$$

where $g_{12} = g_{21}$. To calculate the Fisher information matrix the expected values of all g_{ij} need to be determined [1]. Therefore in this case

$$E\left[g_{11}\right] = E(\frac{1}{\sigma^4}(x-\mu)^2) = \frac{1}{\sigma^2},$$
$$E\left[g_{12}\right] = E(-\frac{1}{2\sigma^4}(x-\mu) + \frac{1}{2\sigma^6}(x-\mu)^3) = 0,$$
$$E\left[g_{22}\right] = E\left(\frac{1}{4\sigma^4} - \frac{1}{2\sigma^6}(x-\mu)^2 + \frac{1}{4\sigma^8}(x-\mu)^4\right) = \frac{1}{2\sigma^4}.$$

Therefore the Fisher information matrix for the normal distribution is

$$\begin{bmatrix} \frac{1}{\sigma^2} & 0 \\ 0 & \frac{1}{2\sigma^4} \end{bmatrix}.$$

Firth suggested to base estimation on the modified score equation [4]. He showed that by using this modification the $O(n^{-1})$ bias of maximum likelihood estimates is removed [6]. The modified score function is related to both the penalized log likelihood, $\ln L(\theta)^* = \ln L(\theta) + 0.5 \ln |I(\theta)|$ and likelihood function, $L(\theta)^* = L(\theta)|I(\theta)|^{0.5}$. In exponential families with canonical parametrization the idea is to penalize the likelihood by the Jeffreys invariant prior. However the approach to binomial logistic models, Poisson log linear models and certain other generalized linear models is different, for these models the Jeffreys prior penalty is applied in typical regression software where a system of iterative modifications are made to the data [4].

3 Application

Kernel density estimation (KDE) is a non-parametric approach to approximating the probability density function of a random variable. In statistics, this is an imperative data smoothing problem where implications about the population are established from a finite data sample. The kernel density estimator is given by

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right).$$

where $K(\cdot)$ is the kernel which is a non-negative function that integrates to 1 and has a mean of 0 and h is a smoothing parameter called the bandwidth. For this application we assume the Gaussian kernel, so that the estimator becomes

$$\hat{f}_h(x) = \frac{1}{\sqrt{2\pi}nh}\sum_{i=1}^{n} e^{-\frac{(x-x_i)^2}{2h^2}}.$$

The only parameter in this model is the bandwidth h. The likelihood and log-likelihood functions, are respectively given by

$$\begin{aligned} L(h) =& \prod_{j=1}^{n} \hat{f}_h(x_j) \\ =& \frac{1}{(2\pi)^{\frac{n}{2}}(nh)^n} \prod_{j=1}^{n}\sum_{i=1}^{n} e^{-\frac{(x_j-x_i)^2}{2h^2}}, \end{aligned}$$

$$\begin{aligned} l(h) = \ln L(h) =& \ln \frac{1}{(2\pi)^{\frac{n}{2}}(nh)^n} + \sum_{j=1}^{n} \ln \sum_{i=1}^{n} e^{-\frac{(x_j-x_i)^2}{2h^2}} \\ =& -\frac{n}{2}\ln(2\pi) - n\ln n - n\ln h + \sum_{j=1}^{n} \ln \sum_{i=1}^{n} e^{-\frac{(x_j-x_i)^2}{2h^2}}. \end{aligned} \tag{2}$$

To calculate h that will maximise the likelihood function, the derivative of the log-likelihood function needs to be found with respect to h.

$$\begin{aligned}\frac{\partial}{\partial h}l(h) &= \frac{\partial}{\partial h}\left[-\frac{n}{2}\ln(2\pi) - n\ln n - n\ln h + \sum_{j=1}^{n}\ln\sum_{i=1}^{n}e^{-\frac{(x_j-x_i)^2}{2h^2}}\right] \\ &= -\frac{n}{h} + \sum_{j=1}^{n}\frac{\sum_{i=1}^{n}\frac{x_j-x_i}{h^2}e^{-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}} \\ \therefore h &= \frac{1}{n}\sum_{j=1}^{n}\frac{\sum_{i=1}^{n}(x_j-x_i)\,e^{-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}}\end{aligned}$$

since h cannot be solved explicitly this solution is not very useful. Therefore a search optimisation approach needs to be used when finding the h value that maximises the log-likelihood function. When calculating the values of the log-likelihood function it is important to note, that the leave-one-out approach needs to be used in order to avoid h converging to zero. The steps involved in this search optimisation approach are as follows:

1. Choose an interval i.e $[a, b]$ where a is the lower limit and b is the upper limit.
2. Divide this interval into 10 equal parts.
3. Calculate the value of the log-likelihood function at each of these 10 points.
4. Save the maximum value and its index, at position i in the vector of calculated log-likelihood values.
5. Create a new interval $[a^*, b^*]$ so that the indices of a^* and b^* are $i-1$ and $i+1$ respectively (given that i wasn't on either end-point of the vector). If it was on the end point, set a^* and b^*equal to that end point.
6. Repeat these steps from 2 until desired accuracy is achieved.

To obtain the Fisher information matrix given by $I(h) = E_h\left[\left[\frac{\partial \ln \hat{f}_h(x)}{\partial h}\right]^2\right]$, the procedure outlined in section 2 needs to be applied. The logarithm of the kernel density estimator is

$$\ln \hat{f}_h(x) = \ln\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}nh}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2} \quad .$$

The partial derivative with respect to h, is then given by,

$$\begin{aligned}\frac{\partial \ln \hat{f}_h(x)}{\partial h} &= \frac{\partial}{\partial h}\ln\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}nh}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2} \\ &= \frac{\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}nh^2}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\left(\frac{x-x_i}{h}-1\right)}{\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}nh}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}} \\ &= \frac{1}{h}\left[\frac{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}} - 1\right].\end{aligned}$$

Therefore

$$\begin{aligned}\left[\frac{\partial \ln \hat{f}_h(x)}{\partial h}\right]^2 &= \frac{1}{h^2}\left[\frac{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}} - 1\right]^2 \\ &= \frac{1}{h^2}\left[\left(\frac{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}}\right)^2 - 2\frac{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}}{\sum_{i=1}^{n}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}} + 1\right]\end{aligned}$$

so that

$$
\begin{aligned}
E_h\left[\left[\frac{\partial \ln \hat{f}_h(x)}{\partial h}\right]^2\right] &= \int_{-\infty}^{\infty} \hat{f}_h(x)\left[\frac{\partial \ln \hat{f}_h(x)}{\partial h}\right]^2 dx \\
&= \frac{1}{h^2}\left[\frac{1}{\sqrt{2\pi}h}\int_{-\infty}^{\infty}\frac{\left(\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}\right)^2}{\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}}dx\right. \\
&\quad \left. -2\int_{-\infty}^{\infty}\sum_{i=1}^{n}\frac{1}{\sqrt{2\pi}h}e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}dx + 1\right] \\
&= \frac{1}{h^2}\left[\frac{1}{\sqrt{2\pi}h}\int_{-\infty}^{\infty}\frac{\left(\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}\right)^2}{\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}}dx + 1\right]. \qquad (3)
\end{aligned}
$$

This therefore is the Fisher information matrix for the Gaussian kernel density estimator. Recall from (1) that the penalized log likelihood is given by $l^*(\theta) = l(\theta) + 0.5\ln|I(\theta)|$. The penalized log-likelihood using (2) and (3) is therefore,

$$
\begin{aligned}
l^*(h) &= l(h) + \frac{1}{2}\ln E_h\left[\left[\frac{\partial \ln \hat{f}_h(x)}{\partial h}\right]^2\right] \\
&= -n\left(\frac{\ln(2\pi)}{2} + \ln n + \ln h\right) + \sum_{j=1}^{n}\ln\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{(x-x_i}{h}\right)^2} \\
&\quad + \frac{1}{2}\ln\frac{1}{h^2} + \frac{1}{2}\ln\left[\frac{1}{\sqrt{2\pi}h}\int_{-\infty}^{\infty}\frac{\left(\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}\frac{x-x_i}{h}\right)^2}{\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{x-x_i}{h}\right)^2}}dx + 1\right].
\end{aligned}
$$

The above procedure has been applied using SAS® software[1](all relevant SAS® software programs used in this report can be found in the appendix and the output from these programs will be shown and discussed in this section). The log-likelihood function was optimised and a penalized log-likelihood function calculated, in order to find optimal bandwidth values for both standard and penalized kernel density estimation. The data found in table 1 was generated using program 1 (see appendix) that makes use of a search optimisation algorithm in order to find the optimal bandwidth values. The convergence of these values that can be noticed in table 1, is the approach that was used in selecting a bandwidth value that will be applied in further application.

[1]The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.

Search Optimisation of Log-Likelihood Function.		Search Optimisation of Penalized Log-Likelihood Function	
Bandwidth	Value of Log-Likelihood Function	Bandwidth	Value of Penalized Log-Likelihood Function
0.300000000000000	-1445.6622227393300	0.300000000000000	-1442.2472022728500
0.300000000000000	-1445.6622227393300	0.300000000000000	-1442.2472022728500
0.300000000000000	-1445.6622227393300	0.298000000000000	-1442.2468239776500
0.299600000000000	-1445.6622172759400	0.298400000000000	-1442.2467950127400
0.299760000000000	-1445.6622131760800	0.298400000000000	-1442.2467950127400
0.299760000000000	-1445.6622131760800	0.298416000000000	-1442.2467949434200
0.299756800000000	-1445.6622131759400	0.298422400000000	-1442.2467949391600
0.299758080000000	-1445.6622131755900	0.298421120000000	-1442.2467949389500
0.299758336000000	-1445.6622131755800	0.298421376000000	-1442.2467949389400
0.299758233600000	-1445.6622131755800	0.298421273600000	-1442.2467949389300
0.299758236160000	-1445.6622131755800	0.298421299200000	-1442.2467949389400
0.299758238208000	-1445.6622131755800	0.298421298432000	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298432000	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412800	-1445.6622131755800	0.298421298401280	-1442.2467949389300
0.299758238412790	-1445.6622131755800	0.298421298401280	-1442.2467949389300

Table 1: Log-Likelihood vs Penalized Log-Likelihood Search Optimisation

In table 2 comparisons have been made between the log-likelihood and penalized log-likelihood function values for a range of bandwidth estimates. From this table it can clearly be observed that the penalized log-likelihood function is greater than the log-likelihood function. This should be the expected result when taking previously discussed theoretical aspects of this topic into consideration, since the penalized log-likelihood function is calculated by summing the penalty and log-likelihood function. In figure 1 these values have been plotted and this result can again be observed.

Comparison Between Log-Likelihood and Penalized Log-Likelihood			
Bandwidth	Value of Log-Likelihood Function	Penalty	Value of Penalized Log-Likelihood Function
0.025000000000000	-1560.8959532813500	5.560910674228800	-1555.3350426071200
0.050000000000000	-1478.1012405926600	4.544067210305000	-1473.5571733823600
0.075000000000000	-1463.0553982245000	3.990741522740560	-1459.0646567017600
0.100000000000000	-1456.7095274932200	3.722076229416410	-1452.9874512638100
0.125000000000000	-1453.0828745112900	3.594758533837190	-1449.4881159774500
0.150000000000000	-1450.6267562140200	3.532014455730630	-1447.0947417582900
0.175000000000000	-1448.8492534948500	3.495885047159390	-1445.3533684476900
0.200000000000000	-1447.5662423665600	3.471602899933940	-1444.0946394666300
0.225000000000000	-1446.6723734796500	3.453480876888990	-1443.2188926027600
0.250000000000000	-1446.0901770181600	3.438878126826010	-1442.6512988913300
0.275000000000000	-1445.7647671921700	3.426345052486790	-1442.3384221396900
0.300000000000000	-1445.6622227393300	3.415020466484520	-1442.2472022728500
0.325000000000000	-1445.7650839588900	3.404380051235420	-1442.3607039076600
0.350000000000000	-1446.0668320652600	3.394102993119300	-1442.6727290721500
0.375000000000000	-1446.5672149694600	3.383991927412580	-1443.1832230420500
0.400000000000000	-1447.2690858117500	3.373923852260300	-1443.8951619594900
0.425000000000000	-1448.1766160213300	3.363820734132330	-1444.8127952872000
0.450000000000000	-1449.2944296352600	3.353632359560380	-1445.9407972757000
0.475000000000000	-1450.6272193247200	3.343326414501230	-1447.2838929102200
0.500000000000000	-1452.1795528063500	3.332882648970780	-1448.8466701573800

Table 2: Comparison between Log-likelihood and Penalized log-likelihood

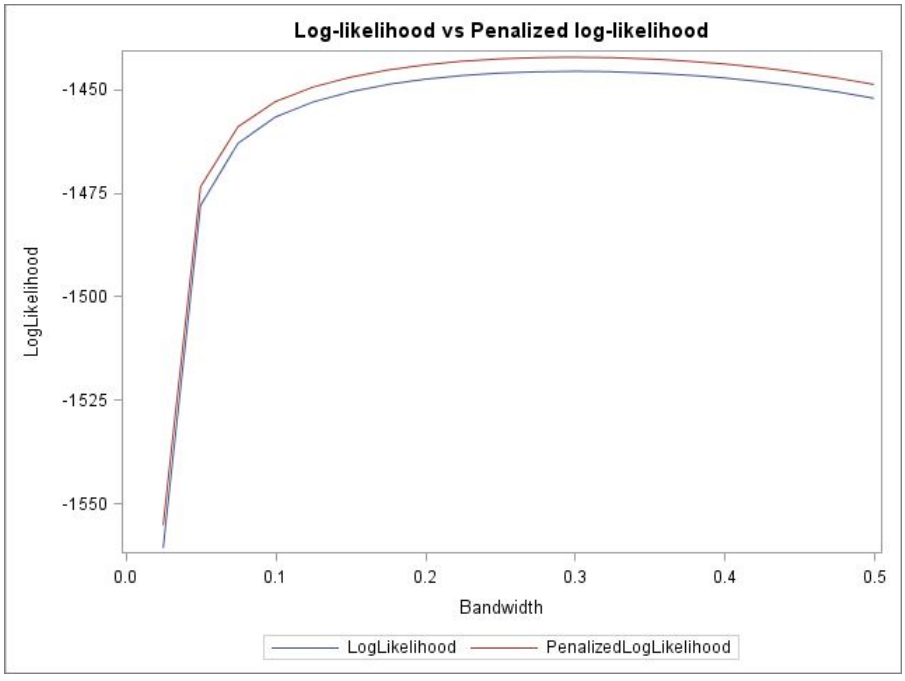


Figure 1: Log-likelihood vs Penalized log-likelihood

The graph in figure 2 was generated using program 2(see appendix). From this graph it is observed that the penalization approach to bandwidth estimation seems to have no effect on kernel density estimation, as

these two density plots overlap exactly. The normal distribution curve in this case was simply used as a reference point.

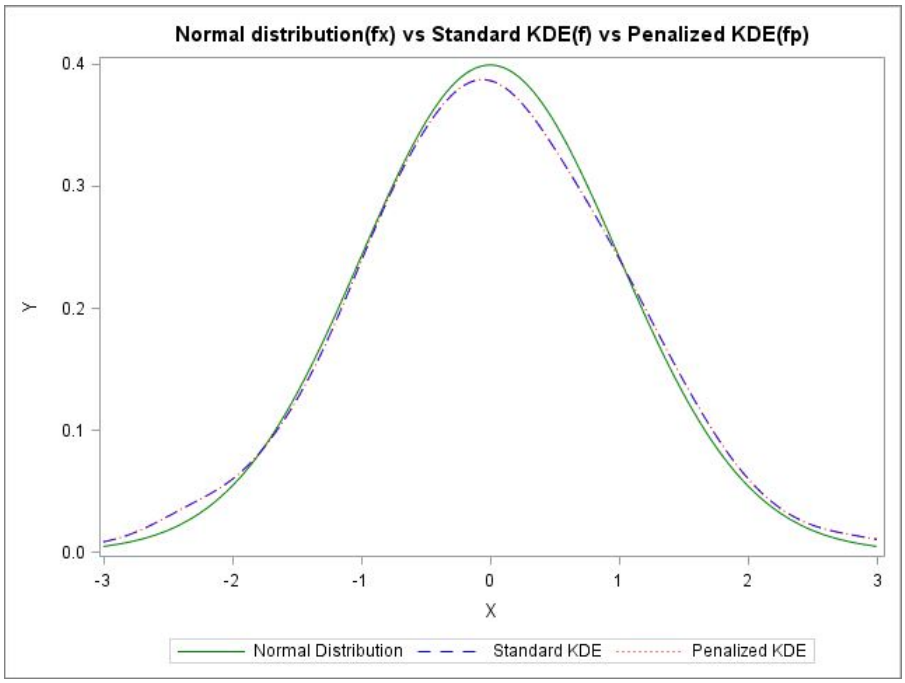


Figure 2: Normal distribution vs Standard KDE vs Penalized KDE

4 Conclusion

In this report the method of penalized maximum likelihood estimation was applied to estimate the kernel density bandwidth. The results observed throughout this report have shown that the penalized maximum likelihood approach to bandwidth estimation does in fact yield approximately the same curve as what the standard maximum likelihood approach did. In conclusion suggesting that using the penalization approach for bandwidth estimation in kernel density had no effect on the smoothness of the curve. These results are not those which had been expected however research can be continued using different data. This conclusion however does not necessarily indicate that the penalization approach does not yield better results than the usual approach, however this comparison has not been investigated in this report and can be explored further in future research.

References

[1] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[2] Stephen R Cole, Haitao Chu, and Sander Greenland. Maximum likelihood, profile likelihood, and penalized likelihood: a primer. *American Journal of Epidemiology*, 179(2):252–260, 2014.

[3] Ronaldo Dias. Nonparametric estimation:smoothing and visualization. Technical report, Universidade Estadual de Campinas, Brazil, 2011.

[4] David Firth. Bias reduction of maximum likelihood estimates. *Biometrika*, 80(1):27–38, 1993.

[5] Frank E Harrell. *Regression Modeling Strategies*. Springer Science & Business Media, 2001.

[6] Georg Heinze. The application of Firth's procedure to Cox and logistic regression. Technical report, University of Vienna, 2001.

[7] Georg Heinze and Meinhard Ploner. A SAS macro, S-PLUS library and R package to perform logistic regression without convergence problems. Technical report, Medical University of Vienna, Vienna, 2004.

[8] Roger Koenker, Ivan Mizera, and Jungmo Yoon. What do kernel density estimators optimize? *Journal of Econometric Methods*, 1(1):15–22, 2012.

[9] KGM Moons, A Rogier T Donders, EW Steyerberg, and FE Harrell. Penalized maximum likelihood estimation to directly adjust diagnostic and prognostic prediction models for overoptimism: a clinical example. *Journal of Clinical Epidemiology*, 57(12):1262–1270, 2004.

5 Appendix

***Program* 1 :**

```
proc iml;
reset noautoname;

x = rannor(j(1000,1,1));

Silverman = std(x)*(4/(3*nrow(x)))**(1/5);

start LogLikelihood(h) global(x);
n = nrow(x);
loglike = -n*log(sqrt(2*constant('PI'))*n*h);

do j = 1 to n;
lltemp = 0;
do i = 1 to n;
if i = j then lltemp = lltemp+0;
else lltemp = lltemp+exp(-(1/2)*((x[j]-x[i])/h)**2);
end;
loglike = loglike+log(lltemp);
end;
return(loglike);
finish LogLikelihood;


/*Find the maximum with a simple search algorithm*/
result = j(20,2);
y = j(10, 2);
a = 0;
b = 0.5;
do j = 1 to nrow(result);
do i = 1 to nrow(y);
y[i,1] = a+(b-a)*(i/nrow(y));
y[i,2] = LogLikelihood(y[i,1]);
end;

if y[<:>,2] = 1 then do;
a = y[y[<:>,2],1];
b = y[y[<:>,2]+1,1];
end;
else if y[<:>,2] = nrow(y) then do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2],1];
end;
else do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2]+1,1];
end;
result[j,] = y[y[<:>,2],];
end;
```

```
c = {"Bandwidth" "Value of Log-Likelihood Function"};

print result [label="Search Optimisation of the Log-Likelihood Function"
colname=c format=19.15];


print Silverman [label="Silverman's Rule of Thumb"];


start mLogLikelihood;
/*Calculate Fisher Information*/
start FisherInt(xi) global(x, h);
n = nrow(x);
numerator = 0;
denominator = 0;
do i = 1 to n;
numerator = numerator+exp(-(1/2)*((xi-x[i])/h)**2)*(xi-x[i])/h;
denominator = denominator+exp(-(1/2)*((xi-x[i])/h)**2);
end;
return((numerator**2)/denominator);
finish FisherInt;

z = {.M .P};
call quad(r, "FisherInt", z) scale=h;

mll = LogLikelihood(h)+(1/2)*log(1/h**2)+(1/2)*log((1/(sqrt(2*constant('PI'))*h))
*r+1);
finish mLogLikelihood;


/*Find the maximum with a simple search algorithm*/
result = j(20,2);
y = j(10, 2);
a = 0;
b = 0.5;
do j = 1 to nrow(result);
do i = 1 to nrow(y);
h = a+(b-a)*(i/nrow(y));
y[i,1] = h;
run mLogLikelihood;
y[i,2] = mll;
end;
if y[<:>,2] = 1 then do;
a = y[y[<:>,2],1];
b = y[y[<:>,2]+1,1];
end;
else if y[<:>,2] = nrow(y) then do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2],1];
end;
else do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2]+1,1];
```

```
end;
result[j,] = y[y[<:>,2],];
end;

c = {"Bandwidth" "Value of Penalized Log-Likelihood Function"};

print result [label="Search Optimisation of the Penalized Log-Likelihood
Function" colname=c format=19.15];


/*Compare the penalisation approach*/
compare = j(20, 4);
do ii = 1 to nrow(compare);
h = 0.5*(ii/nrow(compare));
run mLogLikelihood;
compare[ii,1] = h;
compare[ii,2] = LogLikelihood(h);
compare[ii,4] = mll;
compare[ii,3] = compare[ii,4]-compare[ii,2];
end;

c = {"Bandwidth" "Value of Log-Likelihood Function" "Penalty"
"Value of Penalized Log-Likelihood Function"};
goptions nobs
print compare [label="Comparison Between Log-Likelihood and Penalized
Log-Likelihood Functions" colname=c format=19.15];

create compare from compare[colname={"Bandwidth" "LogLikelihood"
"Penalty" "PenalizedLogLikelihood"}];
append from compare;
close compare;

quit;

title 'Log-likelihood vs Penalized log-likelihood';
proc sgplot data=compare;
series x = bandwidth y=LogLikelihood;
series x = bandwidth y=PenalizedLogLikelihood;
run;
ods _all_ close;
```

***Program* 2 :**

```
proc iml;
n=1000;
mu = rannor(j(n,1,1));
h=0.299758238412790;
hp=0.298421298401280;*penalized;

*normal distribution;
start std_norm(x);
fx=(1/(sqrt(2*constant('pi'))))*exp(-(1/2)*(x)##2));
return (fx);
finish std_norm;
```

```
*kernel density estimation;
start kde_norm(x) global(n,h,mu);
f=0;
do i =1 to n;
f=f+(1/(sqrt(2*constant('pi'))*n*h)*exp(-(1/2)*((x-mu[i])/h)##2));
end;
return (f);
finish kde_norm;

*applying penalized bandwidth estimator;
start kde_pen(x) global(n,hp,mu);
fp=0;
do k=1 to n;
fp=fp+(1/(sqrt(2*constant('pi'))*n*hp)*exp(-(1/2)*((x-mu[k])/hp)##2));
end;
return (fp);
finish kde_pen;

do j=-3 to 3 by 0.01;
x=x//j;
fx=fx//std_norm(j);
f=f//kde_norm(j);
fp=fp//kde_pen(j);
end;

x_kde_y=x||fx||f||fp;

create set from x_kde_y[colname={x fx f fp}];
append from x_kde_y;
quit;

title 'Normal distribution(fx) vs Standard KDE(f) vs Penalized KDE(fp)';
proc sgplot data=set;
yaxis label=" Y ";
series x=x y=fx/legendlabel="Normal Distribution"
lineattrs=(pattern=solid thickness=0.2 color=Green);
series x=x y=f/ legendlabel="Standard KDE"
Lineattrs=(pattern=dash thickness=0.2 color=blue);
series x=x y=fp/legendlabel="Penalized KDE"
Lineattrs=(pattern=dot thickness=0.2 color=red);
run;
```

Comparison of image metrics for greyscale image segmentation

Christine Papavarnavas
10192205

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr I Fabris-Rotelli
Department of Statistics, University of Pretoria

2 November 2015

Abstract

This report outlines image processing techniques for image comparison which provides effective approximations between the true/original image and a processed image. The development and improvement of quality assessment techniques that attempt to replicate the characteristics of the human visual system is essential for the field of image processing.

Declaration

I, *Christine Papavarnavas,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Christine Papavarnavas

_____
Dr. I. Fabris-Rotelli

_____
Date

Acknowledgements

- Mr. A Lau for his valuable assistance in the development of the metric algorithms.
- This work is based on the research supported in part from the National Research Foundation of South Africa for the Grant 90315. Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard.

Contents

List of Figures

List of Tables

1 Introduction

A greyscale image is an image that has various ranges of monochromatic shades i.e. shades from black to white. Furthermore the pixels of greyscale images contain information because each and every pixel has a specific luminance value. An image can be regarded as a function, $f : S \subseteq \mathbb{Z}^2 \rightarrow \{0, ..., 255\}$ for greyscale images and $f : S \subseteq \mathbb{Z}^2 \rightarrow \{0, 1\}$ for binary images. The luminance value of a pixel refers to the degree of brightness or intensity that is determined using a scale from black (0 intensity) to white (full intensity i.e. 255 for greyscale images and 1 for binary images). We represent an image as a matrix, see Figure 1. Figure 2 is a typical graphical repersentation of the surface of an image.

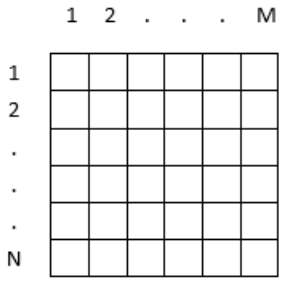


Figure 1: A matrix representing an image $f : S \subseteq \mathbb{Z}^2$ of the size $N \times M$ pixels

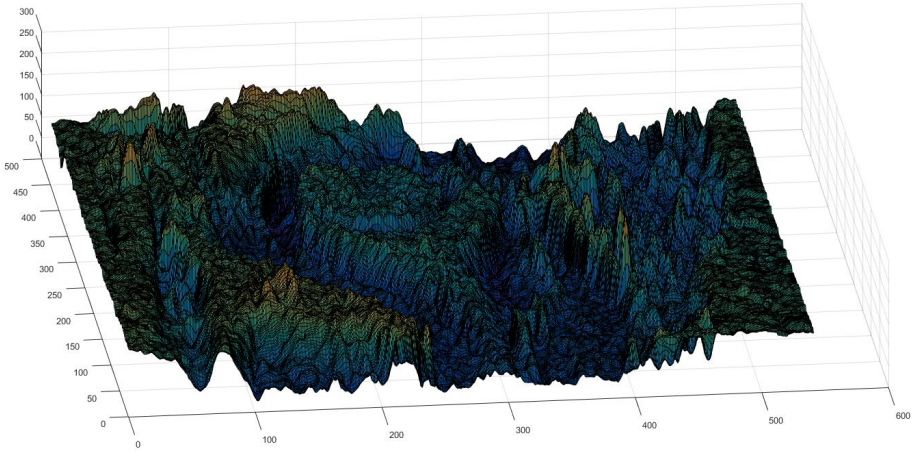


Figure 2: Surface representation of an image

It should be noted that this report will focus on image comparison/error measures. Error measures are used to assess the difference between two images by calculating the distance between the two images' object pixels or picture functions. An image metric will measure the 'distance' between two images f and g, that is $d(f, g)$. Hence error measures can be regarded as an approach in order to evaluate the numerical difference between two images for comparison purposes. There exist objective and subjective methods that are used to assess the image quality that mimic characteristics of the human visual system. The main goal of the methods is to account for the visible errors that are present in the images, a reference image and a distorted image, that are under comparison. Subjective methods for evaluation are considered time-consuming, costly and inconvenient since it requires human involvement either through opinion or a ranking scale marked with adjectives such as 'Bad', 'Poor', 'Neutral', 'Good' and 'Excellent'. Objective methods are usually computer algorithm based which requires input data to automatically calculate numerical values based on complex formulae since computations become straightforward. The algorithm or model can be monitored, optimized

and adjusted accordingly until satisfactory results are obtained. Therefore objective methods will be used to determine the visual differences for image comparison metrics.

In the past research has been invested in binary error measures Δ_b by Baddeley et al[3, 4, 57]. However a greyscale measure Δ_g that allows for an innovative metric for greyscale image comparison, aims to improve the commonly used greyscale comparison measure and the root-mean-square error [57]. Since the focus of this report is Δ_g of Baddeley et al [57], a greyscale measure, the images used for comparison purposes are hence greyscale images. Another metric, the Structural SIMilarity (SSIM) Index is a structural similarity quality measure that is based on the formation of an image that is utilized in image quality assessment [53]. In this report both the Δ_g and the SSIM metric will be analyzed with respect to implementation, application and error sensitivity for certain types of distortions using different image processing techniques. In this report the results of applications of mainly the metrics Δ_g and SSIM will be discussed with regard to the comparison of different images using different image comparison processes. It is important to note that there are a variety of image processing techniques. Image processing techniques include, for example, dilation and erosion [30, 14, 23] , smoothing [32, 33, 13, 60, 36], compression [29, 51, 25], segmentation [63, 22, 59] and sharpening [44, 49]. Image processing techniques have global effects on images and in greyscale images create unrestricted errors since all the grey levels of the greyscale image may be affected. However segmentation techniques in image processing that isolate a greyscale image into numerous segments in order to reduce the information for straightforward pixel analysis will be integrated into the application of the metric Δ_g for greyscale image comparison. Segmentation can be classified into five subcategories namely - region based, edge based, threshold, feature based clustering and model based [22].

In this report image comparison techniques will be analyzed over the ICM segmentation techniques [12].

2 Literature Review

2.1 Image Metrics

BINARY METRICS

In the past image comparison was performed by calculating the correlation function, the root of the mean square (RMS) or the signal to noise ratio until binary error measures were established [57]. The application of binary error metrics requires the calculation of numeric measure (i.e. value) between two binary images in order to determine the discrepancies. Binary error metrics have proved to be important in the quality of image processing algorithms especially in the edge detection and classification or segmentation. Furthermore binary metrics have been compared with the misclassification error rate [3, 57], Pratts figure of merit [3] and the Sobolev norm [57].

A GREYSCALE METRIC

We consider the greyscale metric introduced by Wilson et al [57] and Baddeley et al [57]. Although a greyscale error metric is an extension of a binary error metric the main aim of metrics in applications for image comparison is to use a metric that generates the optimal topology when working with images [57]. The greyscale metric of Wilson et al [57] has furthermore been tailored to reduce sensitivity due to changes of pixels at large distances. Image processing techniques that result in global effects on the image such as dilation/erosion, smoothing, compression and added noise have been applied to images and their effect on the greyscale metric's effectiveness has been observed [57]. It has been documented that the global distortions of images that had compression/dilation techniques applied, image comparison verified that there was hardly any difference between the errors detected by Δ_g and those detected by both the RMS and Sobolev norm. Nonetheless Δ_g's response is considered as more optimal than the RMS response for certain types of distortions. It is also important to note that Δ_b and Δ_g metrics yield different results for the comparison of two binary images, however when both Δ_b and Δ_g measure are normalized the results for the comparisons are similar. Also different distance measures, including Δ_g, for points in an image grid have been tested to show the relevance of the relationship between changes in grey levels with respect to changes in pixel positioning.

The measure Δ_g has also had the potential possible application in image restoration algorithms to determine the maximum number of cycles and highlight changes in a series of medical images [57].

THE SSIM METRIC

The goal in image processing applications is to essentially use a metric that, according to [53], optimizes specifically the algorithms and parameter settings of image processing systems by efficiently monitoring and adjusting simultaneously the image quality. Thus image processing applications require a metric that is ideal for benchmarking image processing algorithms and methods. This metric is based on the ability of the HVS to extract structural information from an image or scene and is thus an excellent objective measure of the perceived image quality. SSIM components such as the mean, variance, and cross-correlation between the reference and the test image contribute to SSIM's quality evaluation of images. Simulation results verify that the SSIM index is better than peak signal-to-noise ratio (PSNR) and mean square error (MSE). Image quality assessment uses the SSIM index and for the overall quality evaluation of the image in comparison use the measure defined as the mean SSIM (MSSIM) and is known to produce values that have uniformity with the qualitative visual appearance. It essential to note that numerous advancements and modifications have been made to the SSIM index to account for different image comparison purposes and applications for example:

- ESSIM and GSSIM [9, 10]: Although it has been proven that the SSIM index fails in yielding useful results for images that are badly blurred, improved methods were developed: the Edge-based Structural Similarity Index (ESSIM) and the Gradient-Based Structural Similarity (GSSIM). Both the ESSIM and the GSSIM incorporate the characteristics of the HVS to extract structural information i.e. for the human to 'capture' a scene. Hence ESSIM and GSSIM regard edge information as the most important structural information of an image. The structure comparison in the SSIM equation is replaced by an edge based structure comparison equation for ESSIM. The contrast and structure comparison equations are replaced in the SSIM formula with a gradient-based contrast and structure comparison equation for GSSIM. Both ESSIM and GSSIM are useful for the image quality assessment especially for blurred images since both are more consistent with HVS in comparison to the SSIM and PSNR. ESSIM and the GSSIM have proven to have an optimal performance in image comparison for Gaussian blurred images, compared to the PSNR and the SSIM.

- The SSIM Index Map for image enhancement [54]: The HVS is attracted to the image regions that are low quality and that can potentially affect the quality evaluation of the entire image. The SSIM index map can be used as a local perceptual quality indicator. The maximum of minimal structural similarity criterion scheme is introduced to enhance the quality at the lowest quality region present in the image, therefore improving the worst case scenario. Hence using this approach the coded image has a uniform quality over the image space.

- MSSIM [55]: Disadvantages regarding the SSIM index exist since it is single scale method; hence Multi-Scale Structural Similarity (MSSIM) has been introduced as it provides flexibility by allowing for variations of the viewing conditions. One such variation that MSSIM allows for is the image synthesis tactic that essentially calibrates parameters that assess and measure the relative importance between different scales, to obtain results closer to HVS than what the SSIM yields. Experimental tests have confirmed that the MSSIM index with the appropriate parameter settings outperforms numerous metrics previously discussed including the SSIM index. The MSSIM component for cross correlation that specifically evaluates pixel values across image scale to provide an indication of how well the edges of images selected for comparison match. Hence for both the SSIM and the MSSIM index it has been verified that the image that maximized the cross correlation component with respect to a reference image has similar (identical) edge information. It has also been noted that MSSIM is crude measure that requires further development to obtain a more systematic approach which will produce a wider range of applications.

- PIQA [16]: The PSNR, SNR and MSE display changes in the image quality and hence are not an accurate representation of Human Visual System (HVS). The Perceptual Image Quality Assessment

(PIQA) combined with visual perceptual masking was introduced to provide a possible solution to the problem. PIQA has three similarity comparision measures namely: the luminance , the structure and the contrast comparison measure (same as in SSIM) which assists in detecting flat or edge region changes. PIQA aims to refine and enhance the ability of identifying the structural information in both blurred and noisy images by using a structural tensor to encode the structural information. The performance of the PIQA in image databases can be considered more superior to MSSIM, in specific applications.

2.2 Image Processing Techniques

We introduce the basics of image segmentation which will be used for the comparison of the image metrics in the application section.

Segmentation can be regarded as an essential modern day image processing tool that allows for automatic image analysis and is regarded as an essential step of low-level vision of the image processing area [63]. Image segmentation creates objects from grouped individual pixels to essentially obtain more information from an image which can be regarded as a form of data mining. Segmentation methods are classified into three groups according to their ability to segment effectively, namely [63, 61]:

- Analytical (applicable only for evaluating segmentation properties that are algorithmic or implementation based)
- Empirical goodness
- Empirical discrepancy

Experimental results regarding these three groups for image comparison provide a ranking system of each segmentation method's evaluation ability. It is important to note that irrespective of the numerous survey papers published regarding segmentation techniques, a single algorithm is not applicable to a broad spectrum of images [63]. Furthermore it should also be noted that specific applications of segmentation techniques require unique algorithms depending on the application and the goals/intention of the researcher/s. The performance evaluation of segmentation algorithms is a crucial topic in ongoing segmentation research. Few research efforts that focus on evaluation methods concentrate mainly on designing innovative evaluation methods and seldom attempt to classify the existence of the different evaluation methods [62, 61]. The performance of the segmentation algorithms are usually assessed according to discrepancy measures or comparison with existing references or measured by goodness parameters. The empirical method is regarded as appropriate in comparison to the analytical method for assessing the performance evaluation and the discrepancy method (a subcategory of the empirical methods) is preferred, to the goodness methods, for the objective assessment of the segmentation algorithms. However all the properties of segmentation algorithms cannot be obtained from analytical studies. This is due to the general lack of theory for image segmentation [24]. A potential classification scheme for segmentation algorithms has been introduced [63, 61].

K-means segmentation obtains cluster centers that essentially minimize the sum of squares (SSE) distances from each individual data point clustered to its cluster center [26] i.e. the center that is closest to it. Each observation in the k clusters can then be regarded as being associated with the nearest mean of a certain cluster. The disadvantages of the $k-$means algorithm are that it is dependent on the initialization, it is extremely sensitive to outliers and can ideally only deal with clusters that have a spherical symmetrical point distribution. Hence the adaptive $k-$means algorithm was proposed [58]. Lloyd's algorithm for $k-$means provides an efficient implementation. The adaptive $k-$means clustering algorithm is regarded useful for image segmentation since it has the ability of further segmenting the regions of intensity distributions that vary in smoothness. The alternative hard $k-$means (AHKM) is a robust metric that can be used to replace the classical Euclidean norm in the $k-$means clustering [58]. The AHKM metric in comparison to the Euclidean norm has been proven in research to be more robust to noise and outliers and furthermore tolerate clusters of unequal sizes [58, 19]. Experimental results based on numerical calculations have also noted that the AHKM has a more favorable performance compared to the hard $k-$means. Therefore the AKHM metric

in conjunction with the AHKM algorithm have been noted and recommended for applications that involve cluster algorithms to image segmentation (especially MRI) [58].

The Iterated Conditional Modes (ICM) algorithm has been used for applications of segmentation and consequently simulation annealing within the various segmentation categories. The ICM algorithm has initially been used for image segmentation since it is both reliable and not computation intensive. Furthermore simulated annealing optimizes sampling within segmented categories. The results obtained for optimized sampling from using the ICM algorithm in conjunction with simulated annealing have proved to produce more powerful optimal prospective sampling schemes designs [12].

Other examples of segmentation techniques are simulated annealing [27, 7, 18], thresholding [59, 45, 47], watershed transform [20, 14, 48, 31, 43, 50], model based segmentation [52, 28, 37] and multi-scale segmentation [39, 6, 11, 34].

3 Image Metrics

One of the main aims of image processing is to obtain a visually smoother image by removing the noise element present. The noise element may be due to numerous factors noted in Wang et al [53] for example acquisition, processing, compression, storage, transmission and reproduction of the image. A subjective evaluation approach is the referred to as the human visual investigation. However in order for the evaluation approach to be objective qualitative methods are preferred and thus used instead. Furthermore the qualitative evaluation approach can be subdivided into three categories [53]:

- Full reference, when there is a complete reference image that is undistorted available with known certainty.
- Reduced-reference, where only a portion of the reference image can be accounted for as known and
- No-reference, where the reference image is unknown.

3.1 Binary Metric

Wilson et al [57] noted that comparisons between binary images were numerically made using one of two techniques, which have been noted to originate from the study of edge detection algorithms, namely:

- Counting the number of pixels that are incorrectly present in the image considered for comparison purposes i.e. the number of false positives with respect to an edge and the number of missed edges, are counted and recorded, and
- Measuring the localization of these errors that were recorded by obtaining the difference i.e. how close the response to the edge was in the comparison.

However numerous complications regarding these methods and their disadvantages were presented in [3] which noted that although these methods presented reasonable measures of errors, the errors that were calculated using the binary error measures were not accurate. The Δ_b metric presented and discussed in Wilson et al [57] is the Baddeley's Δ_b presented in [3]. The Baddeley's Δ_b metric derived is a measure that has been noted to satisfy the necessary axioms of a metric [5] and in essence calculates the distance between the two sets of pixels in images f and g, that is $d(f, g)$. Note that $d(f, g)$ can be regarded as the measure required to evaluate the numerical difference between the two images f and g respectively. Both Matheron and Serra [46, 38] propose that myopic topology is ideally the best topology for the study of binary images. The myopic typology established by Matheron and Serra is generated by implementing the Hausdorff metric on K', where K' denotes the class of all nonempty compact sets present in $\mathbb{R}^2$ [5]. The Hausdorff metric is defined as follows:

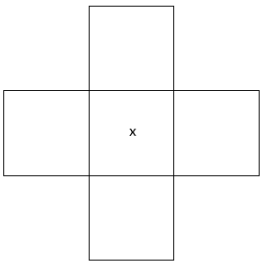


Figure 3: An illustration of $4-$connectivity for pixel x

Definition 1. For the metric space (X, d) and any two sets A, $B \in K'$ the Hausdorff metric is:

$$H(A,B) = \sup_{x \in X} |d(x,A) - d(x,B)|$$

where $d(x,A) = \inf\{d(x,a) : x \in X,\, a \in A\}$ and for $4-$connectivity (see Figure 3) we use the cityblock metric where $d(x,a)$ is defined as $d(x,a) = |x_1 - a_1| + |x_2 - a_2|$ for $x = (x_1,\, x_2)$ and $a = (a_1,\, a_2)$. The discrete version of the Hausdorff metric, namely Baddeley's Δ_b metric introduced in Wilson et al [57], has been implemented in applications for the comparison of binary images, to provide an improvement and resolve the complications associated with existing metrics that have been previously used for binary image comparison purposes. Let $X \subseteq \mathbb{Z}^2$ be a binary image with A, $B \subseteq X$ now discrete sets of $\mathbb{Z}^2$. Baddeley's Δ_b metric is defined as follows:

$$\Delta_b(A,B) = \left[\frac{1}{card(X) \times c^p} \sum_{x \in X} |d^*(x,A) - d^*(x,B)|^p \right]^{\frac{1}{p}} \text{ for } 1 \leq p \leq \infty,$$

$$\Delta_b(f_1,f_2) = \left[\frac{1}{card(X) \times c^p} \left(\sum_{x \in X} |d^*(x,A_1) - d^*(x,A_2)|^p + \sum_{x \in X} |d^*(x,B_1) - d^*(x,B_2)|^p \right) \right]^{\frac{1}{p}}$$

where $card(X)$ is the number of elements in X, $d^*(x,A) = \min\{\{d(x,a) : a \in A\},\, c\}\}$ on $(X \subseteq \mathbb{Z}^2, d)$, $d(x_1,x_2) = \min\{k : k = card(P)\}$ where P is a path (see Figure 4) between x_1 and x_2 via 4-connectivity, c is a bounding constant ensuring no points further than paths of length c contribute to the metric and where A_1 be the set of black pixels in f_1 and B_1 be the set of white pixels in f_1 and respectively let A_2 be the set of black pixels in f_2 and B_2 be the set of white pixels in f_2. The parameters p and c are used to denote the tradeoff between the localization error and misclassification error respectively. For $p = 1$, Δ_b results in the average of the errors i.e. the distance transform discrepancies at each pixel in the image. Whereas $p = 2$ results in the root mean square (RMS) error. It is important to note that there is a direct relationship between larger errors and the value of p, since larger values of p emphasize greater errors since Δ_b is equivalent to the Hausdorff metric as $p \to \infty$. In summary Δ_b evaluates the shortest distance between every pixel $x \in X$ with respect to the two sets A, $B \subseteq X$. For example consider some possible paths for 4-connectivity of $d(x_1,x_2)$ within Figure 4. In Figure 4, the purple path illustrated represents a possible path. The red path is the shortest possible path and is regarded as the path with the shortest distance that is included in the metric calculations, $d(x_1,x_2) = \min\{k : k = card(P)\}$.

Example

The following example illustrates an application of Baddeley's Δ_b metric.

Consider two binary images f_1 and f_2 shown in Figures 5 and 6 respectively, each a 10×10 matrix with domain $X = \{(i,j) : i,\, j = 1, 2 \ldots 10\}$. Let A_1 be the set of black pixels in f_1 and B_1 be the set of white pixels in f_1 and respectivelworst casey let A_2 be the set of black pixels in f_2 and B_2 be the set of white pixels in f_2.

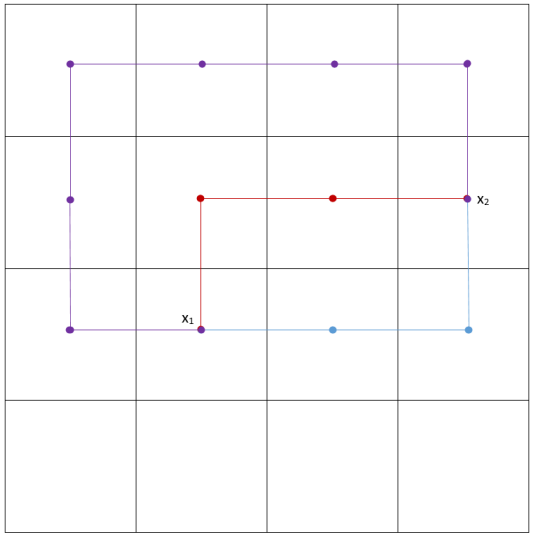


Figure 4: Some possible paths using 4-connectivity between x_1 and x_2

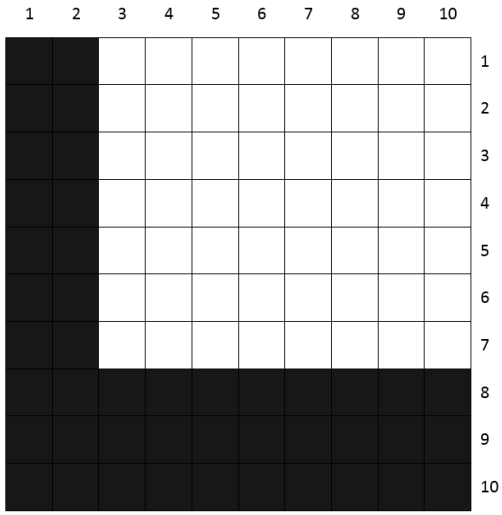


Figure 5: Image f_1 with the set of black pixels A_1 and white pixels B_1

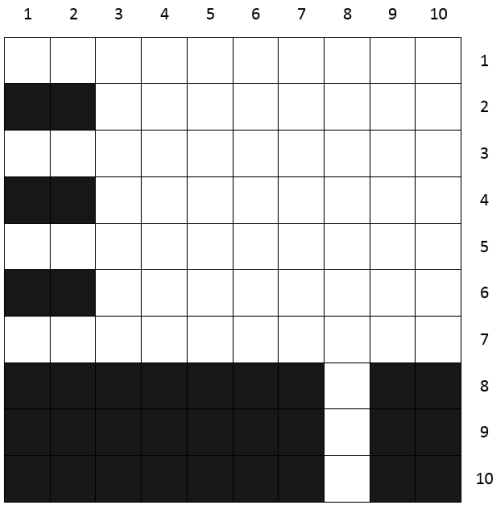


Figure 6: Image f_2 with the set of black pixels A_2 and white pixels B_2

Note that for binary images $f_1 : S \subseteq \mathbb{Z}^2 \to \{0,1\}$ and $f_2 : S \subseteq \mathbb{Z}^2 \to \{0,1\}$ to calculate $\Delta_b(A_1, A_2)$ we set $p = 1$ and $c = 3$:

$$\Delta_b(A_1, A_2) = \left[\frac{1}{100 \times 3^1} \sum_{x \in X} |d^*(x, A_1) - d^*(x, A_2)| \right]$$

where

$$d^*(x, A_i) = \min\{d(x, A_i\}, c = 3\} = \min\{\min\{d(x, a) : a \in A_i\}, c = 3\}\}, \; i = 1, 2.$$

Then

- For $x_{11} = (1, 1)$:
  $d^*(x_{11}, A_1) = \min\{\min\{d(x_{11}, a) : a \in A_1\}, c = 3\}\} = \min\{0, 3\} = 0$
  $d^*(x_{11}, A_2) = \min\{\min\{d(x_{11}, a) : a \in A_2\}, c = 3\}\} = \min\{1, 3\} = 1$ and
  $|d^*(x, A_1) - d^*(x, A_2)| = |0 - 1| = 1$

- For $x_{21} = (2, 1)$:
  $d^*(x_{21}, A_1) = \min\{\min\{d(x_{21}, a) : a \in A_1\}, c = 3\}\} = \min\{0, 3\} = 0$
  $d^*(x_{11}, A_2) = \min\{\min\{d(x_{21}, a) : a \in A_2\}, c = 3\}\} = \min\{0, 3\} = 0$ and
  $|d^*(x, A_1) - d^*(x, A_2)| = |0 - 0| = 0$

- For $x_{36} = (3, 6)$:
  $d^*(x_{36}, A_1) = \min\{\min\{d(x_{36}, a) : a \in A_1\}, c = 3\}\} = \min\{4, 3\} = 3$
  $d^*(x_{36}, A_2) = \min\{\min\{d(x_{36}, a) : a \in A_2\}, c = 3\}\} = \min\{5, 3\} = 3$ and
  $|d^*(x, A_1) - d^*(x, A_2)| = |3 - 3| = 0$

Therefore, continuing in this manner
$\Delta_b(A_1, A_2) = \left[\frac{1}{100 \times 3^1} \sum_{x \in X} |d^*(x, A_1) - d^*(x, A_2)| \right]$
$= \frac{1}{100 \times 3^1} \; [(1+1+1+1+0+0+0+0+0+0) + (0+0+0+0+0+0+0+0+0+0)+$
$(1+1+1+1+0+0+0+0+0+0) + (0+0+0+0+0+0+0+0+0+0)+$
$(1+1+1+1+0+0+0+0+0+0) + (0+0+0+0+0+0+0+0+0+0)+$
$(1+1+1+1+0+0+0+0+0+0) + (0+0+0+0+0+0+0+1+0+0)+$

$(0+0+0+0+0+0+0+1+0+0)+(0+0+0+0+0+0+0+1+0+0)]$
$=\frac{19}{300}$
$=0.0633$

Similarly to calculate to calculate $\Delta_b(B_1, B_2)$

- For $x_{11} = (1,1)$:
  $d^*(x_{11}, B_1) = \min\{\min\{d(x_{11}, b) : b \in B_1\}, c = 3\}\} = \min\{2, 3\} = 2$
  $d^*(x_{11}, B_2) = \min\{\min\{d(x_{11}, b) : b \in B_2\}, c = 3\}\} = \min\{0, 3\} = 0$ and
  $|d^*(x, B_1) - d^*(x, B_2)| = |2 - 0| = 2$

- For $x_{21} = (2,1)$:
  $d^*(x_{21}, B_1) = \min\{\min\{d(x_{21}, b) : b \in B_1\}, c = 3\}\} = \min\{2, 3\} = 2$
  $d^*(x_{11}, B_2) = \min\{\min\{d(x_{21}, b) : b \in B_2\}, c = 3\}\} = \min\{1, 3\} = 1$ and
  $|d^*(x, B_1) - d^*(x, B_2)| = |2 - 1| = 1$

- For $x_{36} = (3,6)$:
  $d^*(x_{36}, B_1) = \min\{\min\{d(x_{36}, b) : b \in B_1\}, c = 3\}\} = \min\{0, 3\} = 0$
  $d^*(x_{36}, B_2) = \min\{\min\{d(x_{36}, b) : b \in B_2\}, c = 3\}\} = \min\{0, 3\} = 0$ and
  $|d^*(x, B_1) - d^*(x, B_2)| = |0 - 0| = 0$

Therefore, continuing in this manner

$\Delta_b(B_1, B_2) = \left[\frac{1}{100\times 3^1} \sum_{x\in X} |d^*(x, B_1) - d^*(x, B_2)|\right]$
$= \frac{1}{100\times 3^1} [(2+1+0+0+0+0+0+0+0+0)+(1+0+0+0+0+0+0+0+0+0)+$
$(2+1+0+0+0+0+0+0+0+0)+(1+0+0+0+0+0+0+0+0+0)$
$(2+1+0+0+0+0+0+0+0+0)+(1+0+0+0+0+0+0+0+0+0)$
$(2+1+0+0+0+0+0+0+0+0)+(2+1+0+0+0+0+0+1+0+0)$
$(1+1+0+0+0+0+1+2+1+0)+(0+0+0+0+0+1+2+3+2+1)$
$=\frac{34}{300}$
$=0.1133$

Therefore,

$\Delta_b(f_1, f_2) = \left[\frac{1}{100\times 3^1} \left(\sum_{x\in X} |d^*(x, A_1) - d^*(x, A_2)|^1 + \sum_{x\in X} |d^*(x, B_1) - d^*(x, B_2)|^1\right)\right]^{\frac{1}{1}}$
$=\left[\frac{1}{100\times 3^1} \sum_{x\in X} |d^*(x, A_1) - d^*(x, A_2)|^1 + \frac{1}{100\times 3^1} \sum_{x\in X} |d^*(x, B_1) - d^*(x, B_2)|^1\right]^{\frac{1}{1}}$
$=\frac{19}{300}+\frac{34}{300}$
$=0.1767$

The smaller the values of $\Delta_b(A_1, A_1)$ and $\Delta_b(B_1, B_1)$ concurrently the closer the binary image similarity. Note it can be shown similarly that $\Delta_b(A_1, A_1) = 0$ and $\Delta_b(B_1, B_1) = 0$. This property is essential for any valid metric.

Results of Δ_b applications

Example 1. Considering the images f_1 and f_2 depicted in Figures 5 and 6 respectively by applying the `Matlab` code provided in the Appendix we obtain $\Delta_b(f_1, f_2) = 0.1767$ as calculated previously using the definition defined for Baddeley's Δ_b metric.

Example 2. However considering the case where the image in Figure 5 is compared to itself, applying the `Matlab` code provided in the Appendix we obtain that the Δ_b metric for the black pixels when f_1 is compared with f_1 is calculated as $\Delta_b(f_1, f_2) = 0$. The results obtained indicate that the images compared are identical and furthermore verifies that Δ_b metric is a valid metric.

Example 3. Similarly when the image in Figure 6 is compared with itself the results obtained for the Δ_b metric are given as $\Delta_b(f_1, f_2) = 0$. The results once again indicate that the images compared are identical and verifies that Baddeley's Δ_b metric is a valid metric.

Example 4. The following conditions are assumed when comparing a pure black image and a pure white image. Suppose that B is a black image and A is a white image. Therefore it is clear that $d^*(x, A) =$

0 , since $x \in A$ and we need to define $d^*(x, A^c) = 1$ since $A^c = \emptyset$. Thus when a f_1 and f_2 are defined to be a 10×10 white matrix (i.e. matrix of ones, full intensity) and a 10×10 black matrix (i.e. matrix of zeros, zero intensity). The following result is obtained for Baddeley's Δ_b metric, $\Delta_b(f_1, f_2) = 0.6667$. The result is approximately similar to the results obtained in [3]. Since the calculated $\Delta_b(f_1, f_2)$ can be regarded as closer to 1 than 0 and since it follows that when Baddeley's Δ_b metric is equal to one is indicative of the fact that each and every pixel in the respective images f_1 and f_2 differ significantly.

3.2 Greyscale Metric

The greyscale metric Δ_g established by Wilson et al [57] is an extension of Baddeley's Δ_b metric [57]. The metric Δ_b was extended to greyscale images in Wilson et al in order to preserve the theoretical basis that Δ_b was defined for and hence establish the greyscale metric Δ_g which generates the optimal topology that is appropriate especially for dealing with a variety of images.
Wlison et al [57] consider two greyscale images f and g and define the distance between the subgraphs as follows:

Definition 2. For a greyscale image $f : X \to Y = \{0, 1, \ldots, 255\}$, the subgraph Γ_f of f is defined as the set of all points in $\mathbb{Z}^3$ that lie between the graph of f and the plane:

$$\Gamma_f = \left\{(x, y) : x \in X \subset \mathbb{Z}^2,\, y \in Y \text{ and } y \leq f(x)\right\}.$$

We then define the distance between points in a subgraph as:

$$d((x,\, y),\, (x',\, y')) = \max\{d(x,\, x'),\, |y - y'|\} \tag{1}$$

where $d(x,\, x')$ is the same path definition as before.

We further define the upper level for a greyscale image:

Definition 3. The upper level at y for a greyscale image f, is denoted as

$$X_y(f) = \{\, x :\, f(x) \geq y \,\}$$

so that $X_y(f)$ is a subset of the domain X that is formed by thresholding f at y.

We now define the distance from a point in X to the set $X_y(f)$.

Definition 4. The distance from a point $x \in X$ to the set $X_y(f)$ for a chosen y is defined as follows:

$$d(x,\, X_y(f)) = \min\{d(x,\, x'),\, x' \in X_y(f)\}. \tag{2}$$

The next distance we define establishes a function that yields the distance between a point $(x, y) \in X \times Y$ i.e. the whole volume $X \times \{0,\, 1,\, \ldots,\, 255\}$, and the subgraph $\Gamma_f \subseteq X \times Y$:

$$\begin{aligned} d((x,\, y),\, \Gamma_f) &= \min\{d((x,\, y),(x,'\, y')) : (x',\, y') \in \Gamma_f\} = \min_{y' \in Y}\left\{\min_{x' \in X_{y'}(f)}\left[\max\{d(x,\, x'),\, |y - y'|\}\right]\right\} \\ &= \min_{y' \in Y}\left\{\max\left[\min_{x' \in X_{y'}(f)}\{d(x,\, x'),\, |y - y'|\}\right]\right\} = \min_{y' \in Y}\{\max[d(x,\, X_{y'}(f)),\, |y - y'|]\}. \end{aligned}$$

Wilson et al [57] continue to establish the greyscale metric by reducing the number of intensity levels that are investigated to determine the minimum, by bounding $d((x,y),\Gamma_f)$ with a constant $c>0$, therefore reducing the sensitivity of Δ_g. The mathematical formula involved with reducing the number of intensity levels to establish Δ_g is:

$$d^*\left((x,\,y)\,,\,\Gamma_f\right)=\min\left\{d\left((x,\,y)\,,\Gamma_f\right),\,c\right\}=\min\left\{\min_{y'\in Y}\left\{\max\left\{d\left(\left(x,\,X'_y\left(f\right)\right)\right),\,|y-y'|\right\}\right\},c\right\}. \tag{3}$$

Therefore reducing the number of intensity levels that need to be checked to determine the minimum in equation (3) by additionally letting:

$$d^*\left((x,\,y)\,,\,\Gamma_f\right)=\min_{y':|y-y'|\leq c}\left(\max\left\{d\left(x,\,X'_y\left(f\right)\right),\,|y-y'|\right\},\,c\right).$$

The bounded distance between the subgraphs of two images f and g at the point (x,y) is then denoted as:

$$\left|d^*\left((x,\,y)\,,\,\Gamma_f\right)-d^*\left((x,\,y)\,,\,\Gamma_g\right)\right|.$$

Definition 5. Suppose that two images f and g are defined on the pixel matrix X that have the same number of possible greylevels. Define Γ_f and Γ_g as the subgraphs of the respective images f and g, then the greyscale metric, Δ_g is defined as:

$$\Delta_g(\Gamma_f,\,\Gamma_g)=\left[\frac{1}{n(X)\times n(Y)\times 256\times c^p}\sum_{x\in X}\sum_{y\in Y}\left|d^*\left((x,\,y)\,,\,\Gamma_f\right)-d^*\left((x,\,y)\,,\,\Gamma_g\right)\right|^p\right]^{\frac{1}{p}}\quad\text{for } 1\leq p<\infty.$$

The formula established for Δ_g is applicable to all greyscale images and but is only applicable to special cases of binary images. Wilson et al [57] stresses that the converse is not true for the binary image metric, Δ_b.

Results of Δ_g applications with comparison to Baddeley's Δ_b metric

Example 5. Considering a similar example in the application section of [3] we use two greyscale images f and g depicted in the respective Figures 7 and 8. Notably f and g are referred to as chessboard images. The Δ_g metric calculated by implementing the `Matlab` code in the Appendix yielded $\Delta_g\left(\Gamma_f,\Gamma_g\right)=0.0013$ indicating that images f and g are similar since the volumes under the subgraphs of the images f and g are similar. However applying the `Matlab` code for Baddeley's Δ_b metric to the images the results obtained are $\Delta_b\left(f,g\right)=0.6667$, since the calculated Baddeley's Δ_b metric is close to 1, it indicates that the images are structurally different.

Example 6. However considering the case where image f is compared to itself, applying the `Matlab` code provided in the Appendix the Δ_g metric is calculated to be $\Delta_g\left(\Gamma_f,\Gamma_f\right)=0$. Hence this result emphasizes, as in the case with Baddeley's Δ_b metric, that the images compared are identical since there is no structural difference present in the images compared and that Δ_g is therefore a valid metric.

Example 7. Similarly as in Example 6 when image g is compared to itself, the Δ_g metric is computed as $\Delta_g\left(\Gamma_g,\Gamma_g\right)=0$. Once again the result confirms that Δ_g is a valid metric and that the images compared are identical i.e. no structural difference.

Example 8. The results obtained when comparing firstly images f, a 10×10 matrix of zeros with g, a 10×10 matrix of ones was $\Delta_g=0.0039$. However when comparing f with a 10×10 greyscale image g where the intensity at each pixel position is 255 i.e. full intensity (a matrix of elements with intensity value of 255 for each pixel position) $\Delta_g=0.9922$. Thus the more similar the volumes of the subgraphs of the compared images are, the closer the calculated Δ_g metric is to zero. Hence indicating that the images are not structurally different.

1	2	3	4	5	6	7	8	9	10	
0	1	0	1	0	1	0	1	0	1	**1**
0	1	0	1	0	1	0	1	0	1	**2**
0	1	0	1	0	1	0	1	0	1	**3**
0	1	0	1	0	1	0	1	0	1	**4**
0	1	0	1	0	1	0	1	0	1	**5**
0	1	0	1	0	1	0	1	0	1	**6**
0	1	0	1	0	1	0	1	0	1	**7**
0	1	0	1	0	1	0	1	0	1	**8**
0	1	0	1	0	1	0	1	0	1	**9**
0	1	0	1	0	1	0	1	0	1	**10**

Figure 7: Image f with a checkerboard of pixel intensities of 0's and 1's.

1	2	3	4	5	6	7	8	9	10	
1	0	1	0	1	0	1	0	1	0	**1**
1	0	1	0	1	0	1	0	1	0	**2**
1	0	1	0	1	0	1	0	1	0	**3**
1	0	1	0	1	0	1	0	1	0	**4**
1	0	1	0	1	0	1	0	1	0	**5**
1	0	1	0	1	0	1	0	1	0	**6**
1	0	1	0	1	0	1	0	1	0	**7**
1	0	1	0	1	0	1	0	1	0	**8**
1	0	1	0	1	0	1	0	1	0	**9**
1	0	1	0	1	0	1	0	1	0	**10**

Figure 8: Image g with a checkerboard of pixel intensities of 1's and 0's.

3.3 Structural similarity icalculated tends to be a value is not significantly different from ndex (SSIM)

The structural similarity index (SSIM) is introduced in order to provide a framework for image quality measure and assessment based primarily on the adaptive capabilities of the human visual system (HVS) to extract structural information from an image or viewing field, providing a measure for the change in the structural information present in the image and hence is an optimal approximation to the perceived image distortion [53].

The structural similarity (SSIM) index defined in Wang et al [53] focuses predominantly on the luminance of the surface of an object that is observed. The surface luminance of the observed object is defined as the product of the illumination and the reflectance, however the object structures present in the image are regarded as independent of the illumination. Since the aim of the SSIM index is to investigate specifically the structural information present in the image, the influence of the illumination within the image needs to separated. Wang et al [53] consider image structural information as the qualities or characteristics that represent the structure of the object within image, that are independent from the average luminance and contrast. Local luminance and local contrast need to be defined for application purposes since the luminance and contrast present in an image vary considerably.

In order to introduce the structural similarity index it is important to note that images are regarded as highly structured since the pixels of an image usually have strong dependencies. The dependencies contain essential information regarding the structure of the image i.e. information regarding the objects that are present with the image. The SSIM algorithm approach emphasizes that SSIM has three elementary comparisons namely luminance, contrast and structure that is required for the task of computing the similarity measurement. Suppose that f and g are two images, with one of the images, f is considered to have flawless or ideal quality, so that the similarity measure acts as a quality measurement of image g [53]. The SSIM algorithm is applied locally to the image. The algorithm requires for the image quality assessment image statistical features, image distortions and the localized quality measurement that yield relevant information regarding the quality degradation of the image. The application of the SSIM algorithm incorporates the calculation of the local statistics μ_{f_8} , σ_{f_8} and $\sigma_{f_8 g_8}$ (defined below). The local statistics are computed using an 8×8 matrix that moves over the entire image pixel by pixel. At each step i.e. each pixel by pixel movement, the SSIM index is calculated. The SSIM is defined and obtained as follows in Wang et al [53]:

- Firstly the luminance of each 8×8 image block, the mean intensity is calculated:

$$\mu_{f_8} = \frac{1}{N} \sum_{i=1}^{N} f_8 (x_i).$$

  The luminance comparison function is then denoted as $l(f_8, g_8)$ and is regarded as a function of μ_{f_8} and μ_{g_8}.

- Remove the mean intensity from the image. The resultant image is defined as $f - u_{f_8}$

- The image contrast unbiased estimate is defined as

$$\sigma_{f_8} = \left(\frac{1}{N-1} \sum_{i=1}^{N} (f (x_i) - \mu_{f_8})^2 \right)^{\frac{1}{2}}.$$

  The function $c(f_8, g_8)$ is then defined as the contrast comparison between f and g and is a function of σ_{f_8} and σ_{g_8}.

- Furthermore the image is normalized by dividing by the standard deviation of the particular image considered, in order to obtain a unit standard deviation. Thereafter the structure comparison function $s(f_8, g_8)$ is conducted on the normalized images, namely $\frac{(f_8 - \mu_{f_8})}{2}$ and $\frac{(g_8 - \mu_{g_8})}{2}$.

- The overall similarity measure is obtained by combing the independent results as follows

$$S\left(f_8,\, g_8\right) = f\left(l\left(f_8,\, g_8\right),\, c\left(f_8, g_8\right),\, s\left(f_8,\, g_8\right)\right).$$

The luminance comparison $l\left(f_8,\, g_8\right)$, is explicitly defined as

$$l\left(f_8,\, g_8\right) = \frac{2\mu_{f_8}\mu_{g_8} + C_1}{\mu_{f_8}^2 + \mu_{g_8}^2 + C_1}$$

where $C_1 = (K_1 L)$ is included to avoid instability when the term $\mu_{f_8}^2 + \mu_{g_8}^2$ is close to zero. Note that L is the range of the pixel values i.e. for greyscale images the range is 255 and 1 for binary images, and $K_1 \ll 1$ is a constant. Another formulation of $l\left(f_8,\, g_8\right)$ taking into account the sensitivity of the HVS to changes in the relative luminance is established by defining R as the luminance change that is relative to the background luminance. The luminance of the distorted image is defined as $\mu_{g_8} = (1+R)\,\mu_{f_8}$, substituting this into $l\left(f_8,\, g_8\right)$ obtain the luminance comparison as

$$l\left(f_8,\, g_8\right) = \frac{2\left(1+R\right)}{1 + \left(1+R\right)^2 + \frac{C_1}{\mu_{f_8}^2}}$$

.

- The contrast comparison function is similarly defined as

$$c\left(f_8,\, g_8\right) = \frac{2\sigma_{f_8}\sigma_{g_8} + C_2}{\sigma_{f_8}^2 + \sigma_{g_8}^2 + C_2}$$

where $C_2 = \left(K_2 L\right)^2$ and $K_2 \ll 1$. The function $c\left(f_8,\, g_8\right)$ has a feature that is regarded as consistent with the human visual system (HVS) being less sensitive to contrast change.
Finally the structure comparison $s\left(f_8,\, g_8\right)$ is calculated.

$$s\left(f_8,\, g_8\right) = \frac{\sigma_{f_8 g_8} + C_3}{\sigma_{f_8}\sigma_{g_8} + C_3}$$

where C_3 is a constant and $\sigma_{f_8 g_8}$ is the correlation between f_8 and g_8 and is defined as

$$\sigma_{f_8 g_8} = \frac{1}{N-1}\sum_{i=1}^{N}\left(f\left(x_i\right) - \mu_{f_8}\right)\left(g\left(x_i\right) - \mu_{g_8}\right).$$

The structure comparison $s\left(f_8,\, g_8\right)$ can take on negative values.

- The comparisons defined as the functions $l\left(f_8,\, g_8\right)$, $c\left(f_8,\, g_8\right)$ and $s\left(f_8,\, g_8\right)$ establish the similarity measure, the structural similarity measure (SSIM), between the 8×8 image blocks in f and g,

$$SSIM\left(f_8,\, g_8\right) = \left[l\left(f_8,\, g_8\right)\right]^{\alpha} \cdot \left[c\left(f_8,\, g_8\right)\right]^{\beta} \cdot \left[s\left(f_8,\, g_8\right)\right]^{\gamma} \tag{4}$$

where $\alpha > 0$, $\beta > 0$ and $\gamma > 0$. The parameters α, β and γ are used to adjust the relative importance of the functions.

- The functions $l\left(f_8,\, g_8\right)$, $c\left(f_8,\, g_8\right)$ and $s\left(f_8,\, g_8\right)$ satisfy the three properties required by the structural similarity index (SSIM) - symmetry, boundedness and the unique maximum [53].
The SSIM is defined specifically as in [53] the simplified form with $\alpha = \beta = \gamma = 1$ and $C_3 = \frac{C_2}{2}$ for each 8×8 image block

$$SSIM\left(f_8,\, g_8\right) = \frac{\left(2\mu_{f_8}\mu_{g_8} + C_1\right)\left(2\sigma_{f_8 g_8} + C_2\right)}{\left(\mu_{f_8}^2 + \mu_{g_8}^2 + C_1\right)\left(\sigma_{f_8}^2 + \sigma_{g_8}^2 + C_2\right)}.$$

- However in practice an overall quality measure for the whole image is needed, thus the mean of the SSIM index is used, the MSSIM

$$MSSIM\left(f,\, g\right) = \frac{1}{N}\sum_{j=1}^{L} SSIM\left(f_{8_j},\, g_{8_j}\right)$$

where f and g are the reference (perfect quality) and distorted image respectively and f_{8_j} and g_{8_j} are the contents of the image at the j^{th} 8×8 image block and L is the number of 8×8 image blocks that are present in the image.
The SSIM measure is applied to each pixel in the image using an 8×8 image blocks and the average of the SSIM values is computed as the MSSIM, the mean structural similarity index. Notably the closer the MSSIM value is to 1 is representative of a stronger similarity. Depending on the application it is possible to calculate the weighted average of the different samples in the SSIM map.

Wang et al [53] introduce the SSIM measure in order to penalize the errors present in the image based on the visibility of the particular error, in order to emulate the human visual system as close as possible.

3.3.1 Advances on the SSIM Measure

Edge based structural similarity

Chen et al [9] acknowledged that numerous researchers have concluded that the HVS i.e. the human eye is highly sensitive to an image's edge and contour information since the edge and contour information of the image can be regarded as essential information regarding the image's structure that the human eye requires in order to capture the scene. The edge based structural similarity (ESSIM) is an improvement of the SSIM algorithm. The ESSIM compares the edge information of the distorted image block with the original perfect quality image block, thus replacing $s\left(f_8,\, g_8\right)$ in the SSIM defined in equation (4) by $e\left(f_8,\, g_8\right)$, the edge based structure comparison which takes into account edge distortion. The ESSIM is defined as:

$$ESSIM\left(f_8,\, g_8\right) = \left[l\left(f_8,\, g_8\right)\right]^{\alpha} \cdot \left[c\left(f_8,\, g_8\right)\right]^{\beta} \cdot \left[e\left(f_8,\, g_8\right)\right]^{\gamma}.$$

Gradient based structural similarity

Chen et al [10] emphasize that the human visual system is very sensitive to an image's edge and contour information since important structural information for the image can be obtained from the edge and contour information which enables the human visual system to capture the scene. The gradient based structural similarity (GSSIM) compares the distorted image block and the perfect quality image's edge information. It replaces both $c\left(f_8,\, g_8\right)$, the contrast comparison, and $s\left(f_8, g_8\right)$, the structure comparison, in the SSIM index defined in equation (4). The functions $c_g\left(f_8,\, g_8\right)$ and $s_g\left(f_8, g_8\right)$ are the gradient contrast comparison and gradient structure comparison that replace $c\left(f_8,\, g_8\right)$ and $s\left(f_8, g_8\right)$ respectively. The GSSIM is defined as:

$$GSSIM\left(f_8,\, g_8\right) = \left[l\left(f_8,\, g_8\right)\right]^{\alpha} \cdot \left[c_g\left(f_8,\, g_8\right)\right]^{\beta} \cdot \left[e_g\left(f_8,\, g_8\right)\right]^{\gamma}.$$

4 Segmentation

Segmentation is simply the process that partitions the domain of an image into systematically interpretable regions. These non-overlapping regions are regarded as a set, thus the union of the non-overlapping regions is the entire true image[61]. Optimal image segmentation implies that the regions of an image segmentation application need to be uniform and homogeneous with regard to any image characteristic consideration such as grey tone or texture. The main goal of segmentation is to decompose the image into partitions that are essentially meaningful with regard to a specific application. Currently there exists a comprehensive and innovative variety of image segmentation techniques that have emerged in the field of image processing. Image

segmentation techniques are considered to be classified for either broad/general purpose image comparison applications or specifically designed specialized categories of images. Therefore segmentation is application specific since the task required determines the models.
Image segmentation techniques can be regarded as either edge based detection, region or model based or feature based clustering [22].

Image segmentation can be referred to as a clustering process, however there is a distinct difference between image segmentation and clustering. Clustering is defined in [22]. Clustering with respect to image pixels is defined as the grouping or dividing an image consisting of a finite number of pixels into smaller groups or sets of pixels based on distinguishable characteristics.
Segmentation that implements clustering has two distinct approaches:

- Approach 1: The steps involved include firstly divisive clustering i.e. top-down clustering [17], partitioning and splitting of the image. Thus the image is recursively partitioned into numerous regions/components
- Approach 2: The steps involved include agglomerative clustering [2, 40], grouping and merging. Therefore beginning with small regions/components i.e a small partition of image pixels and recursively merging the existing clusters.

Image segmentation techniques for edge based segmentation evaluation is classified in [61] based on the method implemented namely:

- Pixel based methods:
  Is a method whereby pixels with similar characteristics for instance colour or texture are grouped into an interpretable region. Examples of pixel based methods include clustering, adaptive k-means[19] and histogram thresholding[22].
- Region based methods:
  This method defines objects present within the image into pixel regions that display homogeneous features/characteristics. Examples of region based methods that group pixels based on pixel similarities are the split-and-merge and the region growing technique[22].
- Boundary based methods:
  Are distinctly different compared to the pixel based methods and region based methods since interpretable regions are defined as pixels that are surrounded by closed boundaries which are present within the image. Boundary based methods are advantageous since it allows for the occurrence of significant characteristic variations of pixels within a closed boundary, examples include edge flow[35].

Segmentation is regarded as a critical in image processing since segmentation in effect is an image processing technique that considers interpretable regions of pixels classified according homogeneous image characteristics instead of evaluating each and every pixel present within the evaluated image. Segmentation is typically implemented in image processing applications in order to locate objects and boundaries present within an image. The various image segmentation techniques listed have advantages and disadvantages noted and discussed in detail in [22, 61]. Segmentation techniques have applications predominately involved in medical diagnosis [22, 21]. First we introduce the simplest segmentation algorithm namely the k-means. We need the definition of the Euclidean norm therein and thus define it below.

In order to define the various metrics and image processing techniques referred to in this report an appropriate distance measure needs to be chosen. The Euclidean norm is a widely used distance measure. On the p -dimensional Euclidean space, $\mathbb{R}^p$ the of length of the vector $x = (x_1, x_2, \ldots, x_p)$ is defined as

$$\| x \|_2 = \sqrt{x_1^2 + x_2^2 + \ldots + x_p^2}.$$

The formula gives the distance from the origin to the point x [5]. However for image processing metrics and techniques since the distance will be measured between pixels from different images the formula is altered accordingly:

$$\Delta(\mathbf{z}_i - \mathbf{z}) = \| \mathbf{z}_i - \mathbf{z} \|_2 = \sqrt{(z_{i1} - z_1)^2 + (z_{i2} - z_2)^2 + \ldots + (z_{ip} - z_p)^2}.$$

4.1 k-means algorithm

The k-means algorithm, also referred to as the k-means clustering algorithm, is an essential algorithm that has been implemented in image processing. The k-means algorithm considers a set of pixels, with an integer k representing the number of groups that is required to classify each pixel in an image into an appropriate cluster. The k-means algorithm therefore determines a set of k points in $\mathbb{R}^p$, referred to as cluster centers, that minimize the Euclidean distance between each image pixel to the nearest cluster center. The main aim of the k-means is to minimize the maximum distance from each image pixel to its closest cluster center. It is important to note that the k-means algorithm is sensitive to noise and outliers. Furthermore the k-means algorithm ultimately aims to minimize the within cluster variability given that there exists k clusters.

- Step 1: Initialization
  If given p pixels $x_1, x_2, \ldots, x_p$ partition the pixels into k clusters defined as $S = \{S_1, S_2, \ldots, S_k\}$. Determine and define the k cluster centers for each cluster partitioned (initial centers can be chosen randomly but affect the convergence of the algorithm). Denote the k clusters as $\omega_1, \omega_2, \ldots \omega_k$ and the mean of each cluster as $\mu_1, \mu_2, \ldots, \mu_k$.

- Step 2: Classification
  For each of the image pixels represented in the image calculate the Euclidean distance between each image pixel and the mean of the respective cluster centers. From these calculations the nearest cluster center is determined and thus each image pixel is included in the cluster that related to the nearest cluster center. The within cluster variability is measured by the sum of square errors (SSE) defined as

  $$SSE = \sum_{i=1}^{k} \sum_{j=1}^{n_i} \left\| \mathbf{x_j^{(i)}} - \mu_\mathbf{i} \right\|_2^2.$$

  The notation n_i represents the number of pixels in the i^{th} cluster and $\mathbf{x_j^{(i)}}$ $\left(\text{i.e. } x_1^{(i)}, x_2^{(i)}, \ldots, x_{n_i}^{(i)}\right)$ the j^{th} pixel in the i^{th} cluster. In order to achieve minimizing the SSE the problem therefore reduces in order to minimize the each and every individual term for each $\mathbf{x_j^{(i)}}$, the j^{th} pixel in the i^{th} cluster:

  $$\left\| \mathbf{x_j^{(i)}} - \mu_\mathbf{i} \right\|_2^2 = \left(\mathbf{x_j^{(i)}} - \mu_\mathbf{i} \right) \bullet \left(\mathbf{x_j^{(i)}} - \mu_\mathbf{i} \right).$$

  Since the SSE is the Euclidean distance assign each pixel to the cluster whose mean yields the smallest term in the SSE, that is, cluster i at iteration t is such that:

  $$S_i^{(t)} = \left\{ x_j : \left\| \mathbf{x_j^{(i)}} - \mu_\mathbf{i} \right\|_2^2 \leq \left\| \mathbf{x_j^{(i)}} - \mu_\mathbf{p} \right\|_2^2, p = 1, 2, \ldots k \right\}$$

- Step 3: Cluster center calculation
  For each cluster created in step 2, the cluster center is recalculated i.e. update the cluster center for each cluster:

  $$\mu_i^{updated} = \frac{1}{card\left(S_i^{(t)}\right)} \sum_{x_j \in S_i^{(t)}} x_j$$

- Step 4: Convergence condition
  Steps 2 and 3 of the algorithm are repeated until the convergence condition is satisfied. The convergence condition dictates that the number iterations, repeating steps 2 and 3, are stopped only once there exists no observable change between the image pixels present in the various clusters or when the difference between the cluster centers at consecutive iterations is smaller than a given threshold.

4.2 Fuzzy k-means

The k-means algorithm has been noted to be an image segmentation process that is also referred to as hard clustering since the partitioning of the pixels into k mutually exclusive clusters is implemented so that the pixels in each cluster simultaneously remain as close as possible to the other surrounding pixels, however a substantially far distance apart from pixels in different clusters [19]. The k-means algorithm is referred to as hard clustering since each pixel can only belong to one cluster. The fuzzy k-means (FKM), based on Ruspini Fuzzy clustering theory which was proposed in 1980's [19] is an alternative to the standard k-means algorithm that incorporates a parameter that represents the degree of fuzziness with respect to the cluster assignments. The fuzzy parameter is denoted as m. The clusters in FKM algorithm are constructed according to the distance between the pixels and the cluster centers that are present in each cluster. FKM is referred to as a data clustering technique whereby the data set of pixels are grouped into k clusters, where every data point in the dataset is associated with every single cluster and therefore every data point has simultaneously a degree of belonging to every cluster. Although the FKM clustering techniques is mainly based on the fuzzy parameter m that represents the fuzzy behavior present in the data set of pixels, the algorithm provides a technique which can be considered as natural since it yields a clustering whereby the weights have a natural interpretation that is not probabilistic. It has been noted that the FKM algorithm is similar in structure, behavior and comparably similar results to the k-means algorithm, and [19] concludes that the k-means algorithm is superior to the FKM algorithm due to the additional computational time required to determine the fuzzy measures that are involved in the algorithm.

- Step 1: Initialization
  Assign each pixel x_p a fuzzy coefficient, $\omega_i(x_p)$ that represents the degree of association of x_p with cluster i such that $\sum_{i=1}^{k} \omega_i(x_p) = 1$ holds for each x_p. A large coefficient $\omega_i(x_p)$ is indicative of a stronger association with the respective cluster. Initially uniform weights are assigned.

- Step 2: Cluster center calculation
  The cluster centers are computed using the formula

$$\mu_i = \frac{\sum_{x_p} \omega_i(x_p)^m x_p}{\sum_{x_p} \omega_i(x_p)^m}$$

  where the notation m represents the fuzzy exponent. The fuzzy exponent is usually set to $m = 2$.

- Step 3: Updating the coefficients
  The fuzzy coefficients are then updated. The formula for updating the fuzzy coefficients for each pixel x_p and $i = 1, 2, \ldots, k$ can be regarded as the inverse distance from the pixel to the cluster

$$w_i(x_p) = \left(\sum_{j=1}^{K} \left(\frac{\|\mu_i - x_p\|_2}{\|\mu_j - x_p\|_2} \right)^{\frac{2}{(m-1)}} \right)^{-1}.$$

- Step 4: Convergence condition
  If the condition for convergence is satisfied when the coefficients no longer change significantly, that is the Euclidean is less than some ε, the algorithm will stop. The $\varepsilon > 0$ is an arbitrarily small positive number often referred to as the threshold. If the condition for convergence is not satisfied then the algorithm will repeat steps 2 to 3 until the convergence criteria is satisfied [8].

4.3 Thresholding

Thresholding is an image processing technique that is regarded as one of the simplest techniques of image segmentation. Thresholding is essentially a method by which a greyscale image is converted into a binary

image [1, 41]. In the simplest application of thresholding the individual pixels present in an image with an pixel intensity value that is greater than a chosen threshold value are labeled as "object" or foreground pixels, [41]. Since thresholding creates a binary image from a greyscale image all the pixels that have intensities below the specified threshold level, the background pixels, are represented by a black pixel and similarly all the pixels that are considered as the foreground of the image, are represented by a white pixel. This allocation is defined as threshold above. Threshold below considers pixels that are below the threshold to be the foreground, essentially the opposite of threshold above. Note that since the greyscale image is converted to a binary image, using the threshold above allocation each background pixel which is a black pixel is assigned a value of 0 and all the "object" pixels which are white are assigned a value of 1. Therefore the binary image is finally created from the greyscale image by colouring each pixel either black or white depending on the image pixel's labels [41].
A variety of thresholding techniques are highlighted in [21] and include:

- Band thresholding:
  The band thresholding technique is a technique in which the foreground intensities are between two threshold values.
- P-tile thresholding:
  This method involves the percentage of the image pixels which originate from the objects. The threshold value is thus chosen as a percentile of the cumulative sum of the pixel intensities.
- Optimal thresholding:
  The threshold value is chosen to be statistically optimal.
- Adaptive thresholding:
  The image is partitioned into sub-images and the threshold value for each sub-image is selected. Therefore the threshold value will vary depending on the location in the image.

A problem that arises for the various thresholding techniques discussed is determining an appropriate threshold value. The examination of the image's histogram is regarded as the simplest approach to determining the threshold value and is based on the grey level frequency distribution of the grey level image f and assigning the threshold a position in a valley that is situated between two modes/peaks of the resultant histogram [22]. Due to the simplicity of this thresholding technique, it is only effective for simple images.

Since determining the optimal threshold value is problematic an iterative method called iterative threshold selection was suggested by Ridler and Calvard in order to find the optimal threshold [42]. The iterative threshold selection is a simplistic technique which can be regarded as a special case of the k-means clustering. The iterative threshold selection supposes an object is present within the image, without any prior knowledge of the object's position in the image. Firstly it is important to note that the image is assumed to have two sections. The two sections are referred to as the background and the foreground, which contains the object. The iterative threshold selection algorithm then iteratively calculates the optimal threshold as follows:

- Step 1
  Assume that the four corners of the image represent the background $b = \{x : x_i \text{ is the background pixel}\} = \left\{ x_i^{(b)} \right\}$ and the rest of the image is the foreground $f = \{x : x_i \text{ is the foreground pixel}\} = \left\{ x_i^{(f)} \right\}$. For example the corners of width and height that are 10% of the image width and height could be used.
- Step 2
  The mean intensity of the background is computed, denoted as $\mu_b^{(0)} = \frac{1}{card(b)} \sum x_i^{(b)}$. Similarly the mean intensity for the foreground is computed and denoted as $\mu_f^{(0)} = \frac{1}{card(f)} \sum x_i^{(f)}$.
- Step 3
  The threshold is calculated as the average of the background and foreground intensities, determined in step 2 i.e.

$$T^{(0)} = \frac{\mu_b^{(0)} + \mu_f^{(0)}}{2}.$$

- Step 4
  The true image is segmented using the threshold calculated in step 3.
- Step 5
  Steps 2 to 4 are repeated. However note that $\mu_b^{(t)}$and $\mu_f^{(t)}$ are the mean intensities of the background and foreground respectively as a result of the t^{th} iteration and $T^{(t)}$ is defined as the threshold value as a consequence of the occurrence of the t^{th} iteration.
- Step 6
  The process is continued for either a predetermined number of iterations or until the threshold value calculated in step 3 changes by only a negligible amount for instance less than 0.5.

Although the iterative selection method initially commences with step 1, it can be replaced by a variety of different initializations for example threshold the true image using the value for the mean intensity or any other appropriate value or a random value that is left to the discretion of the statistician.

4.3.1 Adaptive thresholding

Extensive research has been done in order to resolve the problem of the uneven illumination present in an image. Adaptive thresholding is more sophisticated than conventional thresholding since it provides a possible solution to the uneven illumination that is present across an image that usually arises from strong illumination [56].
Conventional thresholding segments the true image incorporates a specific threshold operator defined as the average of the background and foreground intensities which is regarded as a global threshold for all pixels since it uses a fixed threshold for all pixels. Whereas in adaptive thresholding the threshold value is referred to as changing dynamically over the image since adaptive thresholding selects a unique threshold for each pixel based on the range of intensity values that is associated with the pixel's local neighborhood. More simply adaptive thresholding is defined as a thresholding technique in which the thresholding value differs, as a result of the position of the pixel in the image which is being thresholded. The threshold value may change for each and every individual pixel or remain constant over a sub-region of the image. Adaptive thresholding is similar to conventional thresholding since a binary image is created from a greyscale or colour image that represents the segmentation.

4.3.1.1 The Wellner Algorithm

The text and figures provided in this section were adapted from [21].

Wellner suggests a simple algorithm for adaptive thresholding which is founded on the moving average of the pixel intensities [56].

Define $s \in \mathrm{N}$ as the parameter that specifies the local window size and define $t \in (0,1)$ as the parameter that specifies the relative threshold value. Furthermore let $\Lambda(i,j,s)$ be the pixel intensities' moving average present in the local window that has a size of s around pixel (i,j) [1]. A greyscale image is then transformed into a binary image, β, by considering the following function:

$$\beta(i,j) = \begin{cases} 1 & \text{if } I(i,j) < (1-t)\,\Lambda(i,j,s) \\ 0 & \text{otherwise.} \end{cases}$$

The basic structure of the algorithm is to run through the image while simultaneously computing the moving average of the last s pixels that are observed. However when the pixel in the image has a value that is significantly less than the computed average it is immediately set to black, otherwise it is left white [56]. Note that usually in practice only one pass through the image is usually sufficient and the simplicity of the algorithm condenses the computational time.

[1] Note that in Sections 1.1- 1.2 a pixel was denoted as x_i for simplicity.

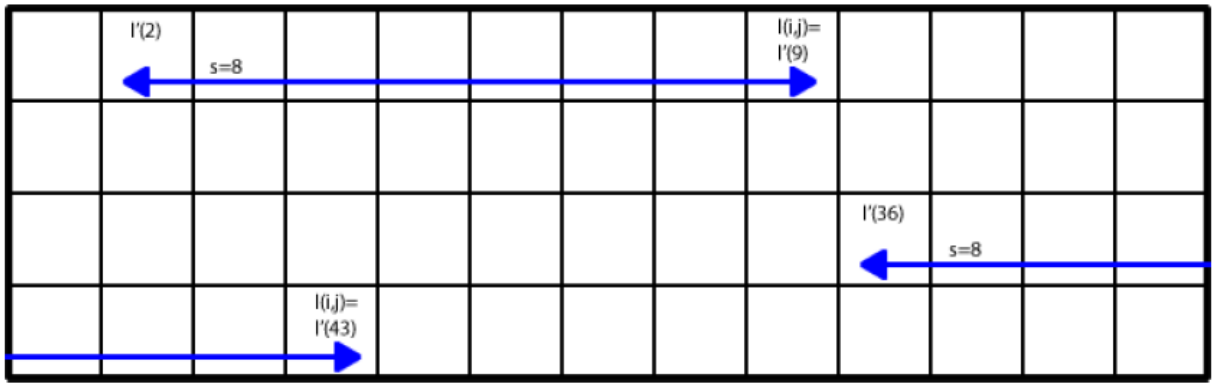


Figure 9: The Wellner algorithm utilizing the last s observations in order to compute the moving average of the pixel intensities obtained from [21].

Choices of $\Lambda(i,j,s)$

Wellner discusses a few appropriate choices for the function defined as $\Lambda(i,j,s)$ [56]. Wellner firstly suggests that the true image is treated as a single row of pixels. Essentially if the image is of size $n \times m$ then each row of the true image is taken and appended to the previous row, thereby creating an $1 \times nm$ vector denoted as I'. Thereafter the moving average is defined as

$$\Lambda_1(i,j,s) = \frac{\sum_{k=0}^{s-1} I'((n-1)\,i + j - k)}{s}$$

this formulation of $\Lambda_1(i,j,s)$ represents the average of the last s observations. Figure 9 gives an indication of how Λ_1 is calculated.

In order to improve the computational time of the algorithm Wellner suggested using an approximation of Λ_1. The approximation requires that the subtraction of an s^{th} of the cumulative sum of the pixel intensities and after the subtraction adds the value of the current pixel intensity. The computational time is reduced since for every pixel the moving average is not required to be recalculated. In [56] Wellner also demonstrates that this approach is equivalent to the exponential weighted average of all the pixels up to and including the current pixel. Consider:

$$\Lambda_2(i,j,s) = \sum_{k=0}^{(n-1)i+j-1} \left(1 - \frac{1}{s}\right) I'((n-1)\,i + j - k),$$

Λ_1 and Λ_2 only take into consideration the pixels that are referred to as trailing pixels, when thresholding the current pixel. Wellner modified A_1 to include the pixels located on either side of the pixel that is being thresholded. The inclusion of the surrounding pixels ensures that the edges of the objects present in the image are more accurately detected. The moving average centered around the $(i,j)^{th}$ pixel in image I i.e. $I(i,j)$ is defined by Wellner as:

$$\Lambda_3(i,j,s) = \frac{\sum_{k=0}^{s} I'\left((n-1)\,i + j + \frac{s}{2} - k\right)}{s}.$$

The definition defining Λ_3 is not perfectly centered around the threshold pixel. However it is important to note that the value of s needs to be even, since the pixel indies are whole numbers. For example the pixel with the notation $I(33.5, 6)$ is illogical. Since the definition of Λ_3 is not perfectly centered around the threshold pixel, Λ_3 is redefined to center the average of the moving pixels around the threshold pixel as:

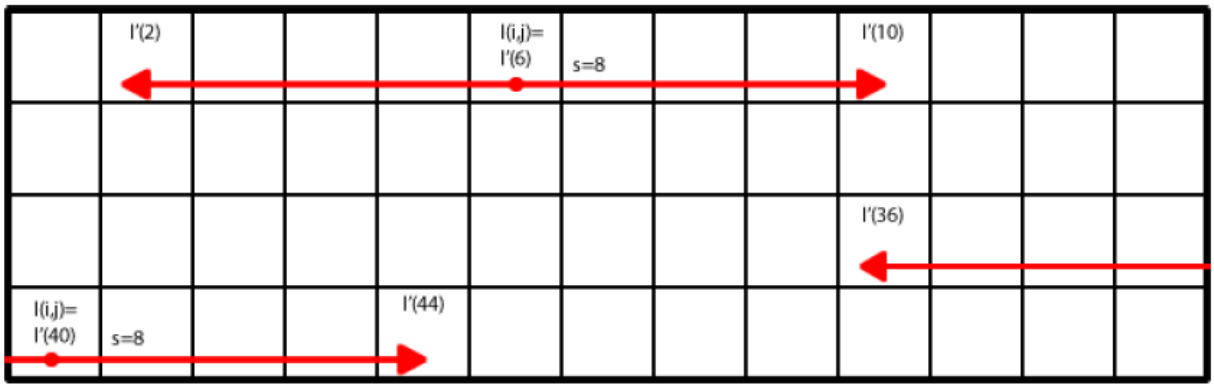


Figure 10: The Wellner algorithm utilizing the s observations that are located around the thresholded pixel in order to calculate the moving average of the pixel intensities obtained from [21].

$$\Lambda_3(i,\, j,\, s) = \left(\frac{\sum_{k=0}^{s} I'\left((n-1)\, i + j + \frac{s}{2} - k\right)}{s+1} \right).$$

After the implementation of this formula, $s+1$ pixels are contained in the window. Figure 10 indicates how $\Lambda_3$is calculated.

A problem associated with the Wellner algorithm is that each row is treated separately. Furthermore the moving average does not consider the pixels that are located above or below the thresholded pixel. The problem is resolved by Wellner by introducing a proposed formulation of the moving average:

$$\Lambda_4(i,\, j,\, s) = \frac{\Lambda(i,\, j,\, s) + \Lambda(i-1,\, j,\, s)}{2}.$$

In the formula the notation Λ refers to any of the moving averages defined. Wellner specifically chooses the moving average defined in Λ_2. Although the Wellner algorithm improves the results, it is regarded as a computational intensive technique since the algorithm requires that previous Λ values are kept and stored.

Problem of the line-end

During the initialization of the Wellner algorithm, the image represented as a matrix is reshaped in order to create a vector that appends the rows to the prior rows which may yield inferior or unsatisfactory results near the image's edges. Referring to Figure 11 it has been noted that the there are two pixels that are marked by the stars. Furthermore note that these pixels marked by stars have no relation to each other. Therefore it is not recommended nor sensible to use the information from the one pixel in order to make a decision regarding the other pixel's threshold value. The application of the Ox Plough method which is usually used to transverse through an image is suggested in an attempt to resolve the problem discussed. The Ox Plough method assures that pixels that are located nearby the image's edges can be more accurately thresholded. The Ox Plough method is illustrated in Figure 11.

Choosing s and t

Wellner established through empirical studies that the optimal results over a range of images required for an image of size $n \times m$ set the parameters:

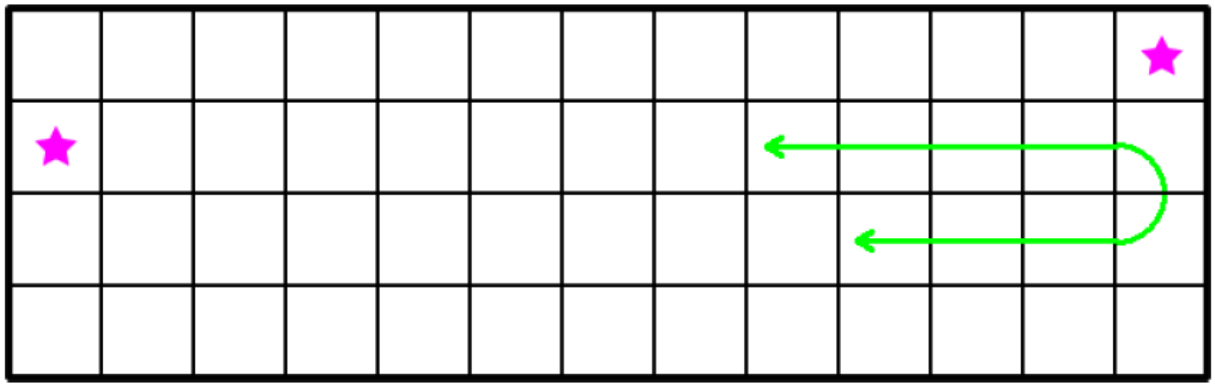


Figure 11: The line-end problem solved by applying the Ox Plough method. In the initialization of the Ox Plough method, firstly move from the left to the right and thereafter from the right to the left when the end of a pixel row is reached. Therefore only neighbouring pixels are used to compute the required moving averages of the intensities obtained from [21]..

- $s = \frac{m}{8}$ and
- $t = 0.15$

4.4 Spatial Clustering

The spatial clustering procedure takes into consideration simultaneously the spatial distribution of both the measurements and the distribution of the measurements present in the measurement space. The Iterated Conditional Mode (ICM) algorithm allows for the spectral features and the spatial information of an image are taken into consideration in adequate image segmentation [12]. The ICM algorithm can be regarded as an image restoration technique since the condition of an image may be corrupted as a result of noisy transmissions or due to human negligence. For example a dirty camera lens. The main aim of the ICM algorithm is thus to restore the image to its true state. Assumptions made when implementing the ICM algorithm include firstly that neighboring pixels tend to have the same pixel value and secondly that every pixel is independently corrupted with an associated probability [12]. The ICM algorithm that was presented in [12] is regarded as simplistic enough to simultaneously illustrate improved segmentation and applies the k-means algorithm in the initialization step of the ICM algorithm. Before the ICM algorithm is defined it is fundamental to note the following notation, for a segmented image I with N pixels where the pixels are represented by x_{ij} the image is segmented into K clusters denoted as $C_1^{(\alpha)}, C_2^{(\alpha)}, \ldots, C_K^{(\alpha)}$, where α is defined as the number of iterations. The ICM algorithm is defined as follows:

Step 1

Apply the k-means algorithm in order to determine the initial cluster mean vectors $\mu_k^{(0)}$ for the k clusters $k = 1, 2, \ldots, K$.

Step 2

For each of the k clusters assign pixel x_{ij} to the minimum of

$$\left(x_{ij} - \mu_k^{(\alpha)}\right)^T \left(x_{ij} - \mu_k^{(\alpha)}\right) - \beta\nu^{(\alpha)} N_{ij}^{(\alpha)}(k)$$

where

- β represents the spatial penalization or correction parameter, as defined in equation (1.5) of [12] ,
- $\nu^{(\alpha)}$ is the within cluster variance formulated as:

$$\nu^{(\alpha)} = \frac{1}{N}\sum_{k=1}^{N}\sum_{(i,j)\in C_k^{(\alpha)}}\left(f_{ij}-\mu_k^{(\alpha)}\right)^T\left(f_{ij}-\mu_k^{(\alpha)}\right)$$

- $N_{ij}^{(\alpha)}(k)$ is defined as the number of neighbours of pixel x_{ij} that are presently classified in cluster k at the α^{th} iteration.

Step 3
Recalculate and update the mean cluster vectors

$$\mu_k^{(\alpha)} = \frac{1}{N_k^{(\alpha)}}\sum_{(i,j)\in C_k^{(\alpha)}} x_{ij}.$$

Step 4
Steps 2 and 3 are repeated until convergence i.e. little or no change occurs.

The optimum value of β is chosen as 2.5 [15].

5 Application

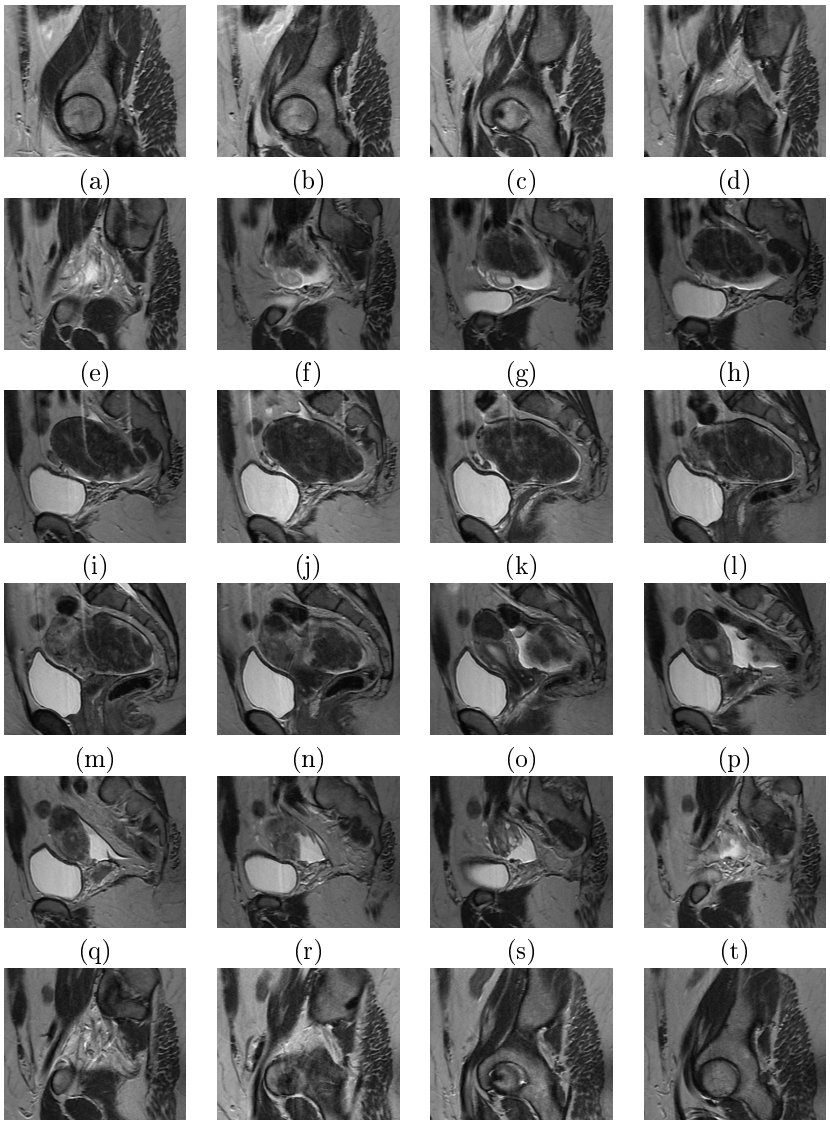


Figure 12: Greyscale Magnetic Resonance Imaging (MRI) images

5.1 Applications using standard thresholding and the Wellner's algorithm

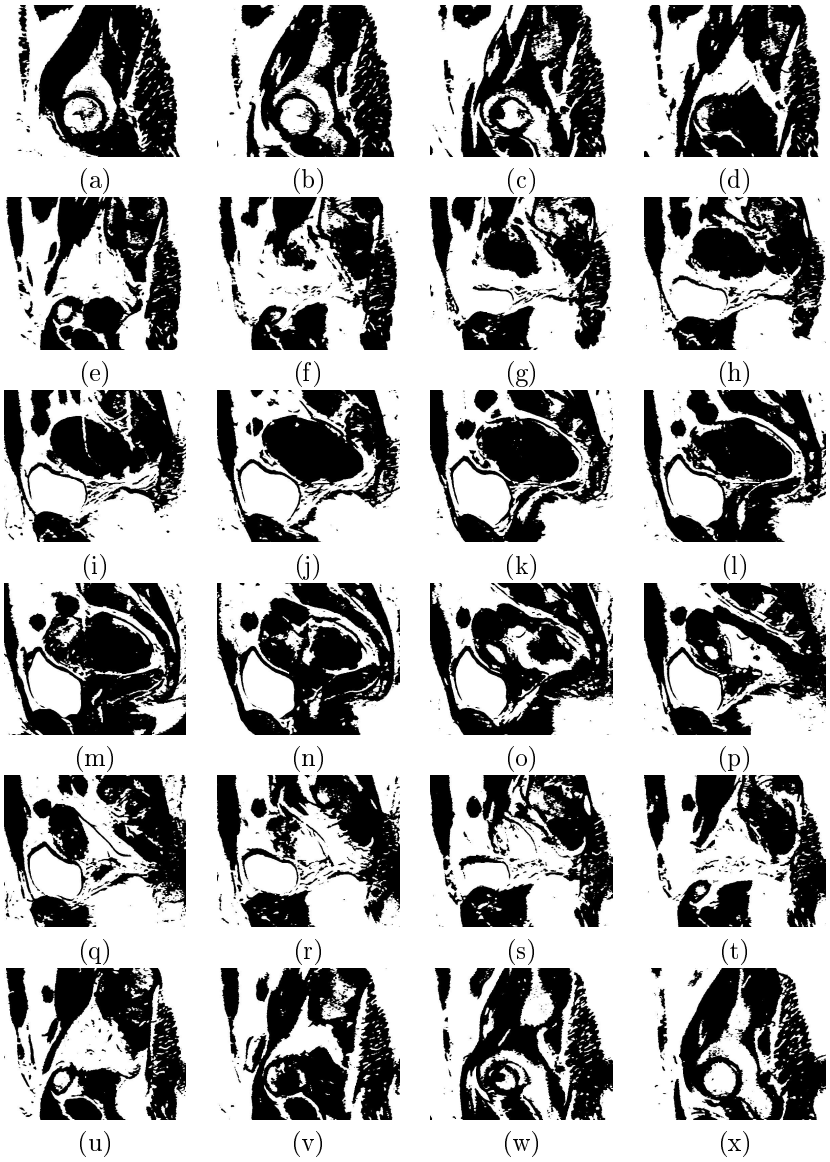


Figure 13: MRI images with standard thresholding

Figure 14: MRI images with Wellner

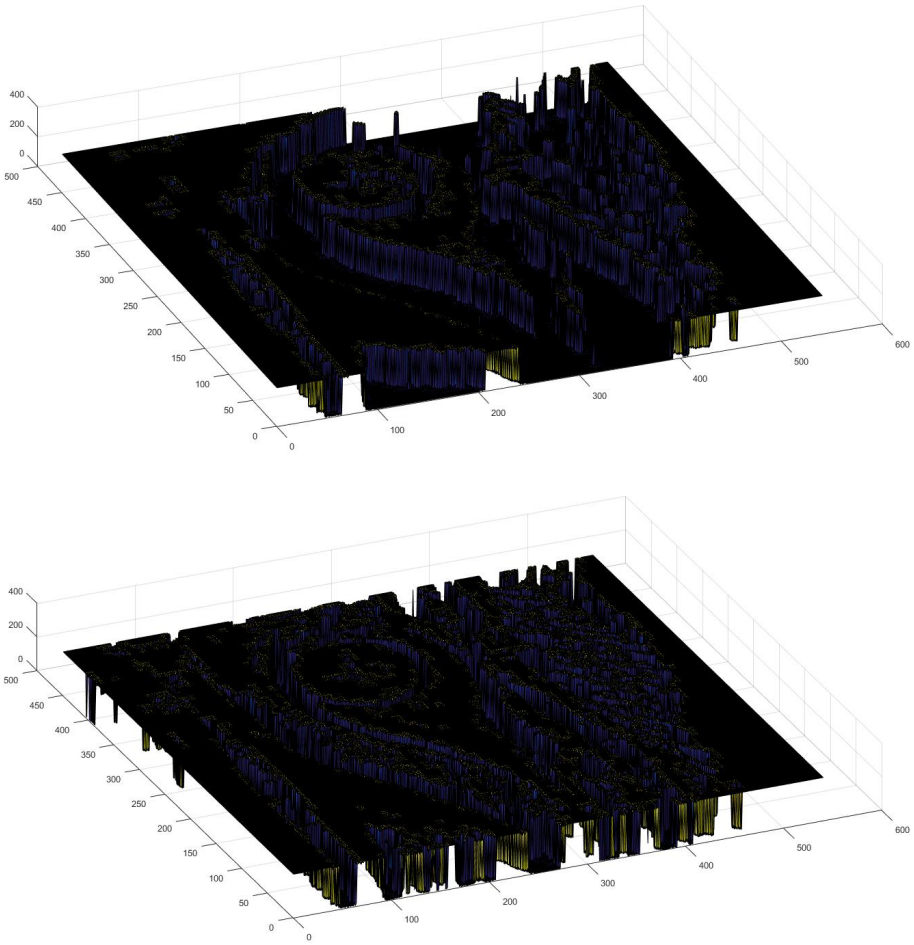


Figure 15: Surface representation of image (a) applying standard thresholding and the Wellner algorithm respectively

Images compared	Δ_b with standard thresholding	SSIM with standard thresholding	Δ_b with Wellner	SSIM with Wellner
1. (a) and (b)	0.3338	0.3173	0.3977	0.2069
2. (b) and (c)	0.3671	0.2624	0.4048	0.1936
3. (c) and (d)	0.4217	0.2092	0.4371	0.1551
4. (d) and (e)	0.3757	0.2605	0.4547	0.1377
5. (e) and (f)	0.3499	0.2751	0.4217	0.1603
6. (f) and (g)	0.3344	0.3023	0.3959	0.2027
7. (g) and (h)	0.3194	0.3309	0.3858	0.2176
8. (h) and (i)	0.3036	0.3645	0.3667	0.2545
9. (i) and (j)	0.2817	0.4141	0.3354	0.3060
10. (j) and (k)	0.2839	0.4305	0.3480	0.2874
11. (k) and (l)	0.2848	0.4244	0.3311	0.3037
12. (l) and (m)	0.3088	0.3915	0.3611	0.2715
13. (m) and (n)	0.3176	0.3767	0.3420	0.3035
14. (n) and (o)	0.3381	0.3514	0.3687	0.2621
15. (o) and (p)	0.3470	0.3232	0.3593	0.2682
16. (p) and (q)	0.3411	0.2987	0.3495	0.2644
17. (q) and (r)	0.3407	0.2880	0.3256	0.2775
18. (r) and (s)	0.3316	0.3141	0.3606	0.2364
19. (s) and (t)	0.3214	0.3112	0.3972	0.1977
20. (t) and (u)	0.3115	0.3284	0.4000	0.1867
21. (u) and (v)	0.3838	0.2601	0.4535	0.1431
22. (v) and (w)	0.4299	0.2257	0.4401	0.1641
23. (w) and (x)	0.3732	0.2925	0.4048	0.2042

Table 1: Results of standard thresholding and the Wellner Algorithm

- We apply standard thresholding and the Wellner algorithm to the greyscale MRI images represented in Figure 12.The results are shown in Figure 13 and Figure 14, we then compare the Δ_b and SSIM metric for the thresholded images. The application results for the values of the Δ_b metric for the thresholding technique are shown in Table 1 and Figure 16. The measures increase significantly at comparisions 1, 2 and 3. The increase in the value of the Δ_b metric in Comparison 1 is arguably due to the influence of the structure of image (b) in Figure 13 especially since a larger number of white pixel areas surround the round joint object in image (b) than in image (a). The increase in the Δ_b metric in Comparison 2 is as a result of a decrease in the number of white pixel areas and subsequent increase in black pixels around the joint object that is a significantly less prominent structure in image (c), irrespective of the relative increase in the structural size of various muscle tissue regions in image (c), that have a significantly more white pixels present in the regions of increase. The Δ_b value of the metric for comparisons 1 and 2 increased due to the change in the image structure as a result of significant changes in the proportion of black and white pixel regions within the consecutive images. Notably the images are relatively similar in structure and the object position of the respective joint/muscle objects present within the consecutive sequence of the standard thresholded MRI images in Figure 13. The graph for the application of standard thresholding in Figure 16 indicates that for Comparison 1 and 2 there is a relatively small increase in the value of the Δ_b metric. Considering the images in Comparison 1 and 2 in Figure 13 it can be visually seen that the proportion of white pixels increases in certain areas, justifying the resultant increase in the value of the Δ_b metric between Comparison 1 and 2. However considering the graph in Figure 16 and the results in Table 1 for standard thresholding and referring to image Comparison 3. (c) and (d) indicates that there is a significant increase in the metric value for Δ_b. This significant increase is due the sudden increase in the proportion of black pixels and consequent decrease in the proportion of white pixels that influence the structure of the image.

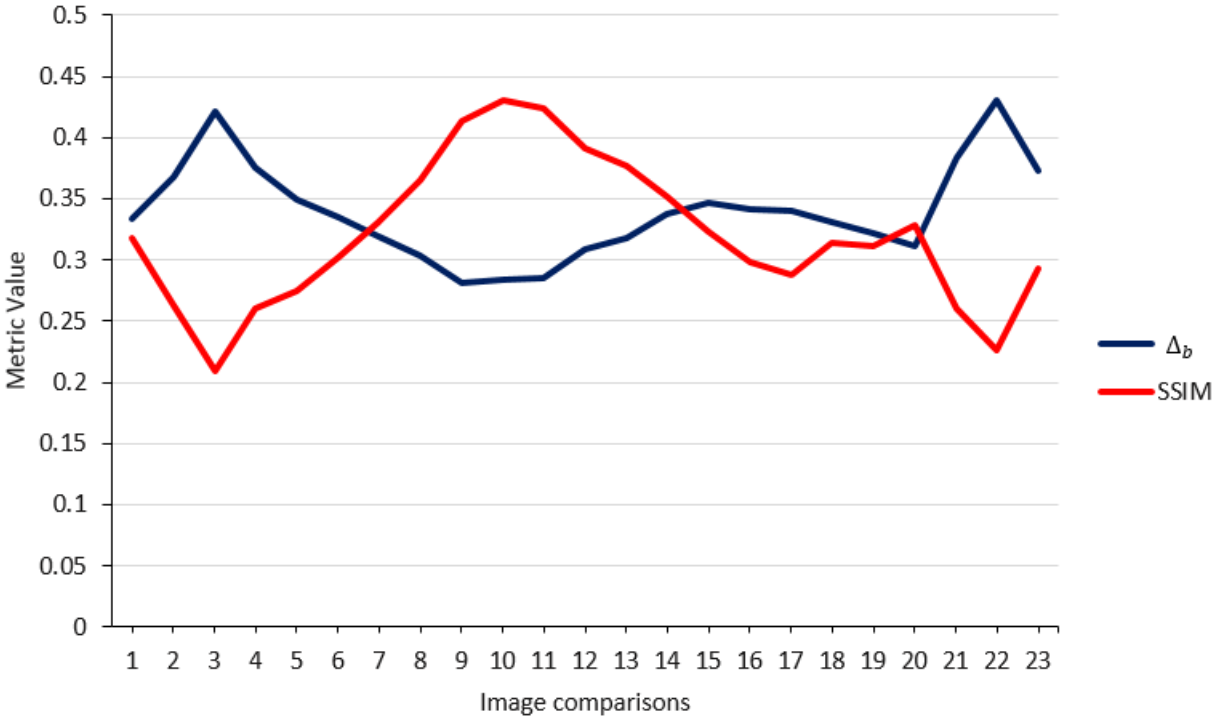


Figure 16: Measure values for the metrics using standard thresholding

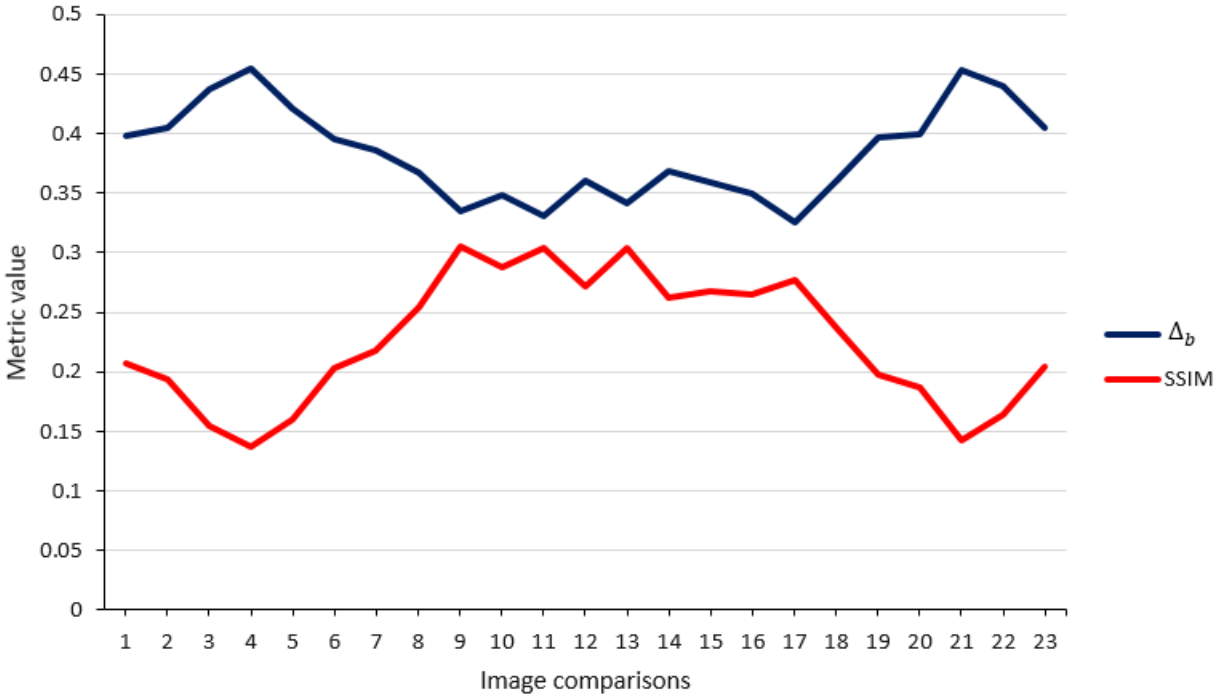


Figure 17: Measure values for the metrics using the Wellner algorithm

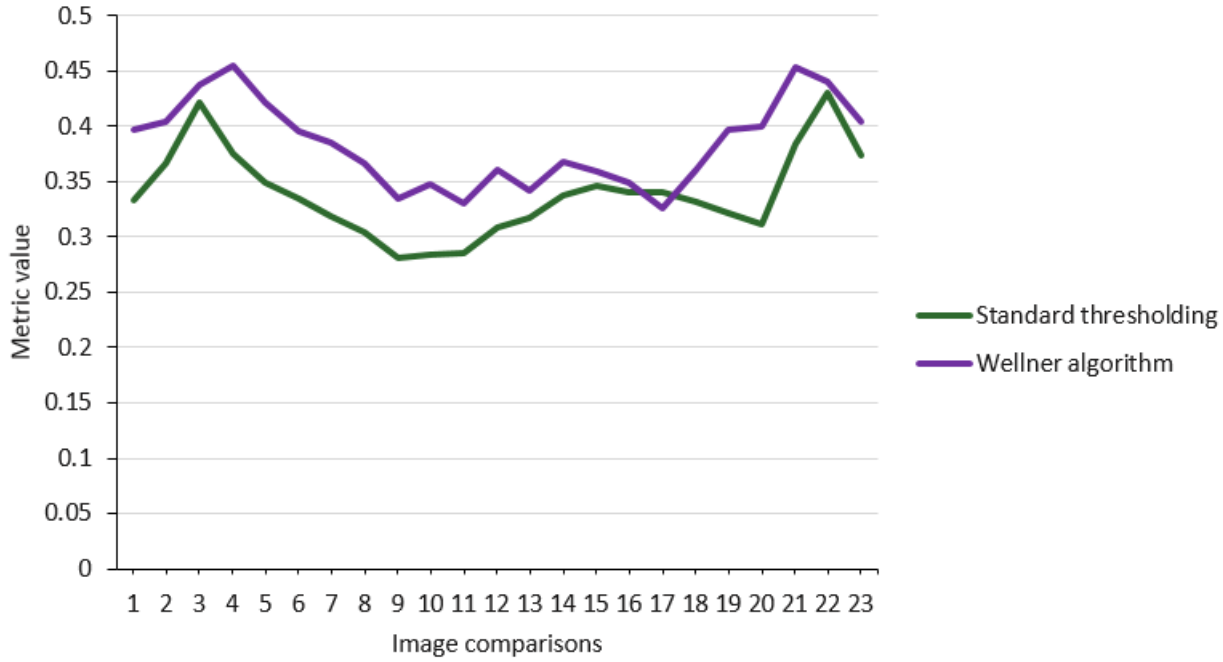


Figure 18: Measure values for the Δ_b metric applying thresholding segmenation techniques

Figure 19: Measure values for the SSIM metric applying thresholding segmentation techniques

- The graphs representing the Δ_b and SSIM metric results for the standard thresholding and Wellner's algorithm in Figures 16 and 17 are mirror graphs or alternatively the lines representing the Δ_b and SSIM metric in each respective graph are symmetric to each other. The symmetry of the lines representing the Δ_b and SSIM metric is due to the objective of the metrics under consideration. Since the main objective of metrics is to account for visible errors that are present in images, a reference image and the distorted image that are under comparison. Baddeley's Δ_b metric is a numerical measure between binary images to determine discrepancies i.e. errors. The SSIM metric is however a measure of the luminance of the surface of an object that is observed and predominantly aims to assess the structural information present within the image. The symmetry of the lines representing the results for Baddeley's Δ_b and SSIM metric is due to the proportion of the black and white pixels within the image, that affect the luminance present within an image and hence the value of the SSIM metric. The greater the proportion of white pixels compared to black pixels results in a greater SSIM value, and the position or location of the black and white pixels present influences the structure of an image i.e. the appearance of the image to the HVS, thus influencing the numerical value of Baddeley's Δ_b metric since the greater the structural difference between compared images, the greater the numerical value obtained for the Δ_b metric. This is furthermore illustrated by the surface representation of the image under the thresholding techniques as shown in Figure 15. Hence there is an inverse relationship between Baddeley's Δ_b metric and the SSIM metric, since a greater variation in luminance between compared images (i.e. the proportion of white pixels present in the images, significantly influences the structural information of the images) yields a greater Δ_b value and a lesser associated SSIM value.

- Referring to Figures 16 and 17 the peaks present in the respective standard and Wellner's algorithm are discussed. In Figure 16 the peaks occur at image Comparison 3 and 22 for Baddeley's Δ_b metric and for the SSIM metric at image Comparison 10. Corresponding to the peaks in image comparisons 3 and 22 an observable similarity of the consecutive images that are compared is that the proportion of black pixels is greater than the proportion of white pixels. Whereas for image Comparison 10 it is observed that a greater proportion of the consecutive images are regions of white pixels. Furthermore considering image comparisons 3 and 22 for standard thresholding, the consecutive images have significant variation in the location and regions of black pixels that directly influence the structure of the image, since the consecutive images are observably different. Thus the significant change in the structural information for the consecutive standard thresholding images is due to the increase in the proportion of the black pixels and subsequently their location which influences the intricate details of the image's structure contributing to a significantly higher Δ_b value. However the peak located at Comparison 10 for the SSIM metric in Figure 16 is irrespective of the proportion of black pixels which is significant even though the consecutive images have a greater portion of white pixels, the structural information of the image remains unchanged hence resulting in a higher SSIM. The greater SSIM metric value is due to the observable luminance i.e. white pixels that are predominantly focused on the surface of the observed joint object present in the consecutive images. The application of the Wellner's algorithm to the greyscale images, in Figure 14, results in converted images that have a significant proportion of white pixels and greater details highlighted in comparison to images with the standard thresholding applied as in Figure 13. Figure 17 has peaks observed at image comparisons 4 and 21 for Baddeley's Δ_b metric when considering the Wellner's algorithm. Referring to Figure 14 the consecutive images in image comparisons 4 and 21 have a significant proportion white pixels as opposed to black pixels. The peaks in the Δ_b metric values are due to a significant change in the structure of the consecutive images since there are greater regions of white pixels that alter the structural information present within the compared images.

- Figures 18 and 19 indicate that the respective Δ_b and SSIM metrics display a similar trend for the sequence of image comparisons for both the standard thresholding and the Wellner's algorithm. In Figure 18, the Wellner's algorithm displays slightly greater values for the Δ_b metric than standard thresholding, since referring to Figure 13 the images as a result of standard thresholding have a greater proportion of black pixels in comparison to the images obtained by implementing the Wellner's algorithm which have a greater proportion of white pixels - see Figure 14. The image comparisons for the Wellner's algorithm yield greater Δ_b values due to a greater variation in the structural information present in

the consecutive images. The structural information is influenced by the greater proportion of white pixel regions and intricate details present in the consecutive Wellner's algorithm images, see Figure 14. Whereas in Figure 13, the sequence of standard thresholding images obtained have considerably greater black pixel regions and minimal object structure detail. The SSIM metric values display a similar trend for the sequence of image comparisons for both the standard thresholding and the Wellner's algorithm, however some differences are noted refer to Table 1. In Figure 14 the surface of the joints/objects observed have a greater luminance, since there are greater regions of white pixels providing luminance to the surface of the observed object in comparison to the consecutive images in Figure 13. Hence the values for the SSIM metric are greater for the standard thresholding technique.

5.2 Applications using the ICM

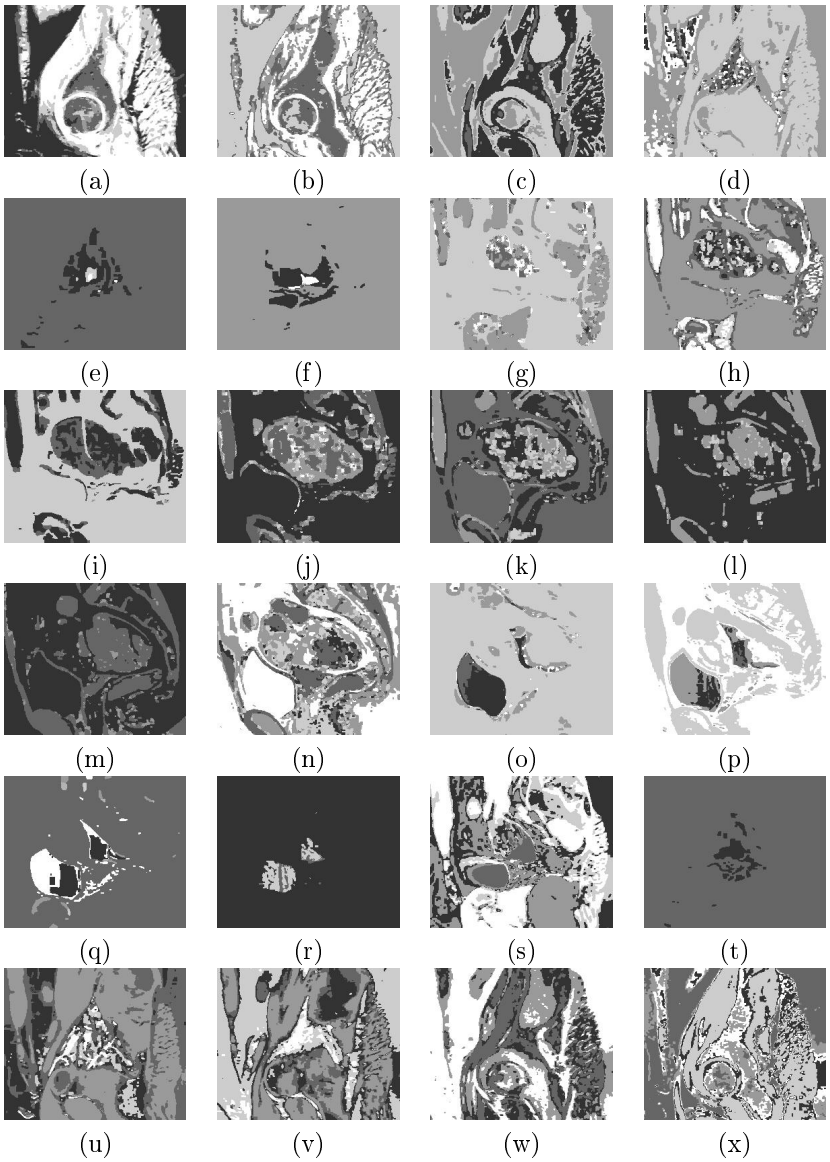


Figure 20: MRI images with ICM

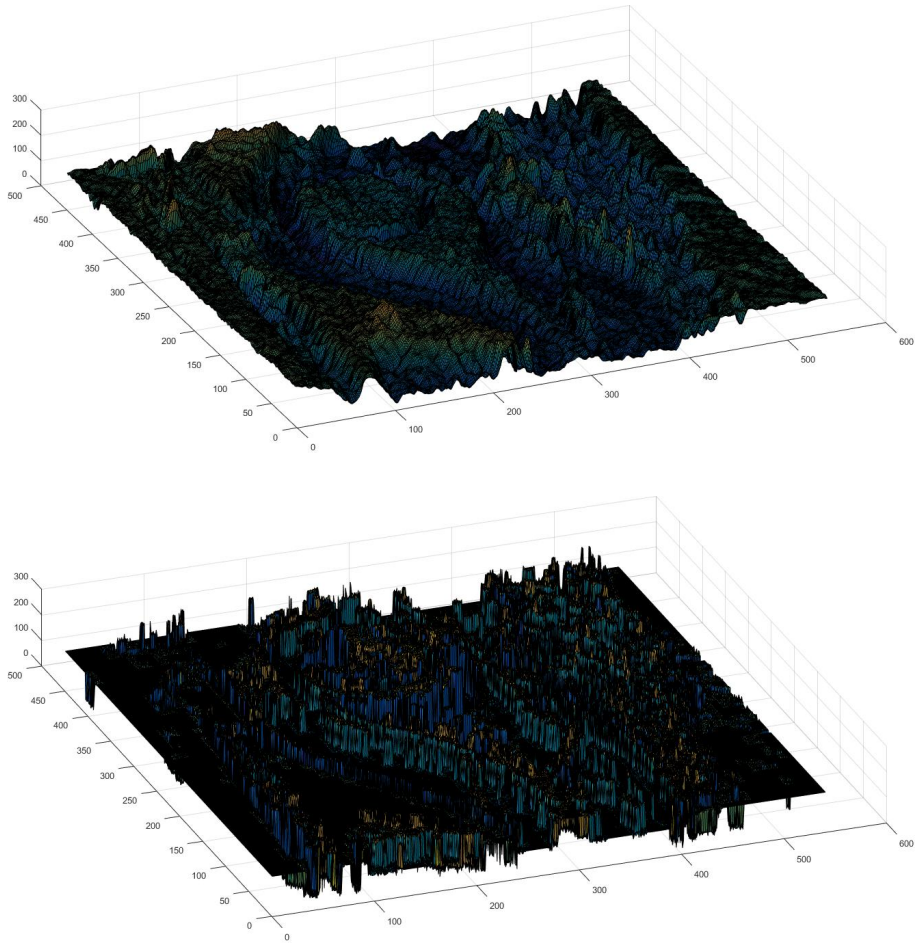


Figure 21: Surface representation of (a) for the greyscale and ICM algorithm respectively

Images compared	Δ_g for greyscale images	SSIM for greyscale images	Δ_g for ICM images	SSIM for ICM images
1. (a) and (f)	0.1960	0.2128	0.3378	0.2656
2. (k) and (l)	0.0846	0.9857	0.1991	0.3198
3. (m) and (n)	0.0702	0.4780	0.4741	0.0911
4. (a) and (x)	0.1213	0.3129	0.2481	0.1608

Table 2: Results for greyscale images and ICM images

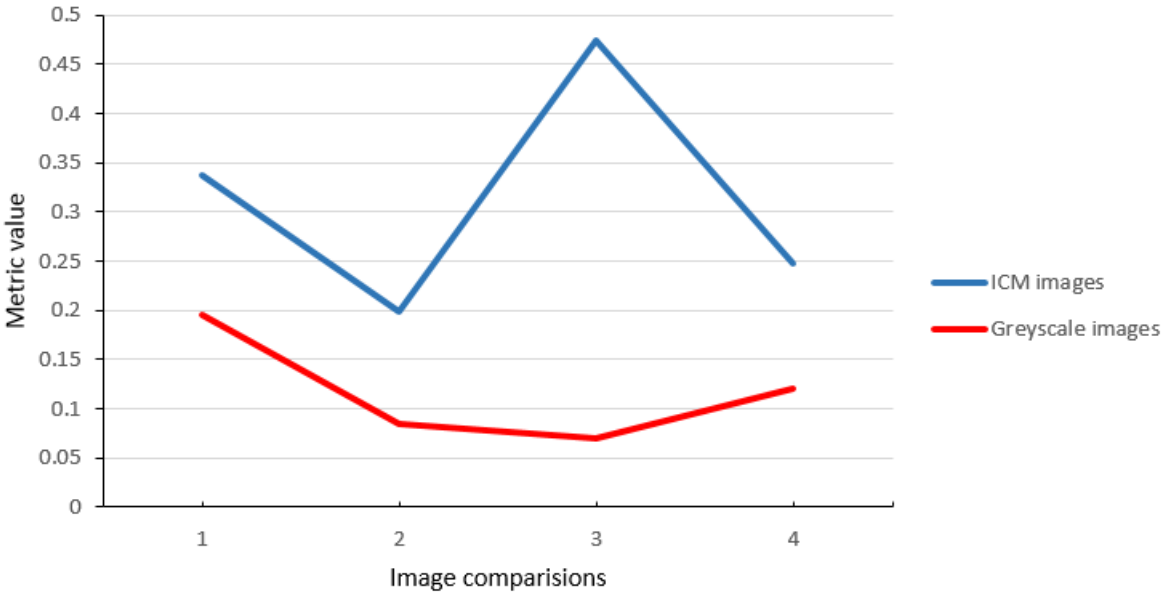


Figure 22: Measure values for the Δ_g metric

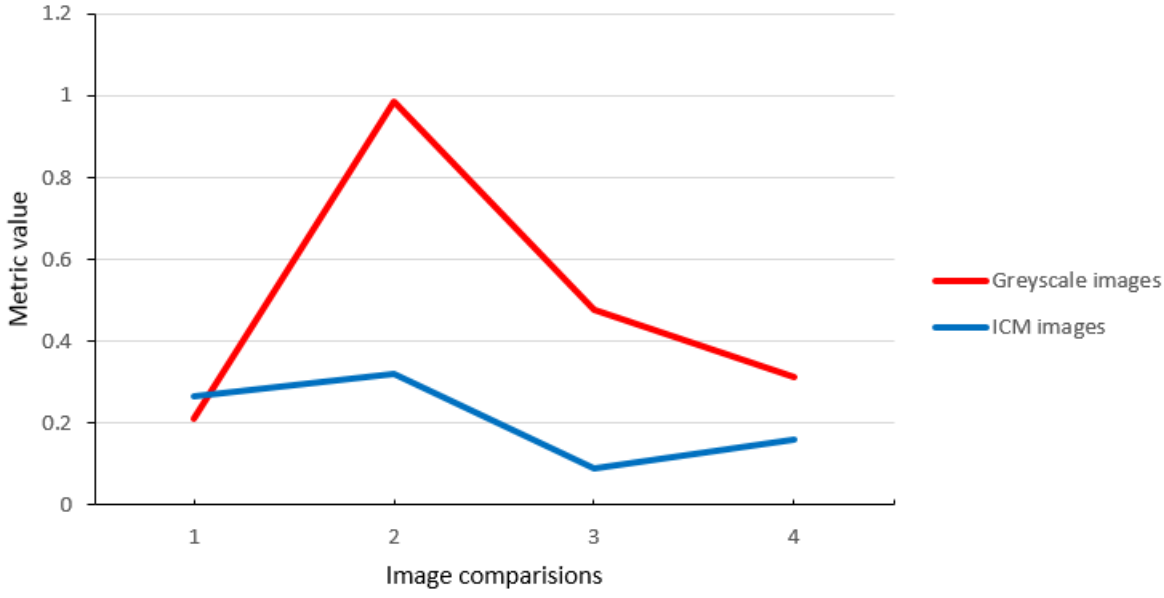


Figure 23: Measure values for the SSIM metric

- The metric results Δ_g and the SSIM, obtained for greyscale and ICM images are represented in Table 2. The Δ_g metric requires extensive computational time due to the theoretical complexity implemented in the `Matlab` algorithm which a disadvantage to implementing the Δ_g metric. Thus alternative metrics that are more efficient should be considered for application purposes. An efficient metric that yields instantaneous results (and is thus regarded as optimal) and has an objective that focuses on evaluating the overall quality of compared images is the SSIM metric.

- The results indicate for Δ_g that the closer the metric value is to 0 the more similar the images. However a Δ_g value close to 1 indicates that the compared images are different. The ICM algorithm is an image processing technique that is used for image segmentation and implements the $k-$means algorithm. Referring to Table 2 , the values obtained for the Δ_g metric, in the case of image Comparison 1, indicate that the images compared are relatively similar since the value of Δ_g is close to 0 as apposed to 1. Notably the Δ_g metric value obtained for the ICM image comparison is greater than the numerical value obtained for greyscale images since the ICM converted images in Figure 20 are observed as significantly different. This is illustrated in the surface representation of the image as shown in Figure 21.

- Referring to the results obtained by comparing the results of Δ_g for greyscale MRI images and Δ_g for ICM images, the results obtained for Δ_g are greater for the ICM thresholded images in Figure 20 than the greyscale MRI images in Figure 12. This is due to the fact that the greyscale MRI images converted using the ICM algorithm are no longer structurally identical to the initial greyscale images. The ICM algorithm has a clustering characteristic due to the application of the the $k-$means algorithm which influences how various regions present in the resultant ICM images are represented. Since the ICM algorithm represents regions that have similar pixel intensities as specific monochromatic shades of black, grey and white; causing the resultant ICM images to be significantly different to the initial greyscale images presented in Figure 12. The ICM algorithm yields greater Δ_g numeric values since the images compared tend to be significantly different due to the pixel intensity classification of the various regions present in the images. This indicates that the compared ICM images are significantly different as the values of Δ_g are closer to 1 whereas the greyscale images are relatively similar, since the Δ_g values are relatively close to 0 as seen in Table 2 and Figure 22. The peak in Figure 22 is due the classification of the pixel intensities from implementing the ICM algorithm, since the ICM images in Comparison 3 for Figure 20 differ greatly with respect to the pixel intensity classification even though the images are sequential.

- The SSIM metric measures the luminance of the surface of the observed object and investigates the structural information present within the image. The SSIM metric value for the ICM images is generally lower than the SSIM value for the greyscale images as the luminance of the observed object in Figure 20 is less than that observed in Figure 12, justifying the SSIM values in Table 2. The significant difference in the SSIM values is due to the ICM algorithm's pixel intensity classification for various regions present in the image. Images with the same pixel intensities are represented with the same monochromatic shade which reduces the variety of luminance values present in the image, altering the structural information that the SSIM metric requires for calculations. The peak represented in Figure 23 for image Comparison 2 emphasizes this fact.

6 Conclusion

The greyscale metric Δ_g introduced and established by Wilson et al [57] is regarded as an error metric that is an extension of the binary metric, Δ_b that yields similar results when both the Δ_b and Δ_g metric are normalized. However the disadvantages associated with implementing the Δ_g metric disrepute it as an optimal error sensitivity metric. The disadvantages associated with Wilson et al's [57] Δ_g metric are the theoretical complexities involved in the algorithm and the computational time required to obtain the desired output. Due to the intensive computational time involved when implementing the Δ_g metric, the applications presented in Section 5 selected only specific corresponding greyscale and ICM images instead of implementing the Δ_g algorithm on the consecutive sequence of greyscale and ICM images represented in Figures 12 and 20 respectively. The intense computational time of the Δ_g metric is due to the aim of this error metric, which is to show the relevance of the relationship between the changes in grey levels with respect to the pixel positioning. This is accounted for when the subgraphs Γ_f and Γ_g are computed for the comparison of images f and g, defined in Subsection 3.2. However the Δ_g metric is appropriate for applications that involve binary and greyscale images whereas Baddeley's Δ_b metric is only applicable for binary images. The error metric that is regarded as optimal throughout the applications presented in Section 5 is the SSIM metric. The SSIM metric yielded instantaneous results irrespective of applying the standard thresholding,

Wellner's algorithm and ICM thresholding techniques. The SSIM metric is an error metric implemented in the image quality assessment to obtain an overall quality evaluation of the compared images and is regarded as an optimal error metric since it satisfies the requirements given in [53]. That is the ability to monitor and adjust the image quality, optimize specifically the algorithms and parameter settings of image processing systems and also to benchmark the image processing algorithms and methods. Thus the SSIM metric is arguably an ideal, optimal and unsurpassed error metric in comparison to Baddeley's Δ_b and Wilson et al's Δ_g metric. Further creative exploration of the Edge-Based Structural Similarity Index (ESSIM), Gradient-Based Structural Similarity Index (GSSIM) and Muli-Scale Structural Similarity (MSSIM) that focus on the concepts of error metrics, image processing techniques and the structural information present in an image, that drive innovative success, should be investigated.

References

[1] M Agarwal and V Singh. A methodological survey and proposed algorithm on image segmentation using genetic algorithm. *International Journal of Computer Applications*, 67(16):7–17, 2013.

[2] M Amadasun and RA King. Low-level segmentation of multispectral images via agglomerative clustering of uniform neighbourhoods. *Pattern Recognition*, 21(3):261–268, 1988.

[3] AJ Baddeley. An error metric for binary images. In *Proceedings of the International Workshop on Robust Computer Vision*, pages 59–78. Wichmann Verlag, Karlsruche, 9-11 March 1992 1992.

[4] AJ Baddeley. Errors in binary images and an l_p version of the Hausdorff metric. *Nieuw Archief voor Wiskunde*, 10(4):157–183, 1992.

[5] RG Bartle. *The Elements of Real Analysis*, volume 2. Wiley New York, 1964.

[6] C Burnett and T Blaschke. A multi-scale segmentation/object relationship modelling methodology for landscape analysis. *Ecological Modelling*, 168(3):233–249, 2003.

[7] P Carnevali, L Coletti, and S Patarnello. Image processing by simulated annealing. *IBM Journal of Research and Development*, 29(6):569–579, 1985.

[8] C-T Chang, Jim ZC Lai, and M-D Jeng. A fuzzy k-means clustering algorithm using cluster center displacement. *Journal of Information Science and Engineering*, 27(3):995–1009, 2011.

[9] G-H Chen, C-L Yang, L-M Po, and S-L Xie. Edge-based structural similarity for image quality assessment. In *Acoustics, Speech and Signal Processing, 2006. 2006 IEEE International Conference on*, volume 2, pages II–II. IEEE, 2006.

[10] G-H Chen, C-L Yang, and S-L Xie. Gradient-based structural similarity for image quality assessment. In *Image Processing, 2006 IEEE International Conference on*, pages 2929–2932. IEEE, 2006.

[11] JC De Anda, XZ Wang, and KJ Roberts. Multi-scale segmentation image analysis for the in-process monitoring of particle shape with batch crystallisers. *Chemical Engineering Science*, 60(4):1053–1065, 2005.

[12] P Debba, A Stein, F van der Meer, E Carranza, and A Lucieer. Field sampling from segmented image. In Spinger Berlin Heidelberg, editor, *Computational Science and Its Applications, Lecture Notes in Computer Science*, volume 5072, pages 756–768, ICCSA 2008.

[13] G Deng and LW Cahill. An adaptive Gaussian filter for noise reduction and edge detection. In *Nuclear Science Symposium and Medical Imaging Conference, 1993., 1993 IEEE Conference Record.*, pages 1615–1619. IEEE, 1993.

[14] ER Dougherty and RA Lotufo. *Hands-on Morphological image processing*, volume 71. SPIE press Bellingham, 2003.

[15] I Fabris-Rotelli and J-F Greeff. The application of iterated conditional modes to feature vectors of the discrete pulse transform of images. In *Proceedings of the 23nd Annual Symposium of the Pattern Recognition Association of South Africa*, pages 149 – 156, November 2012.

[16] X Fei, L Xiao, Y Sun, and Z Wei. Perceptual image quality assessment based on structural similarity and visual masking. *Signal Processing: Image Communication*, 27(7):772–783, 2012.

[17] DA Forsyth and J Ponce. *Computer Vision: A Modern Approach*. Prentice-Hall Englewood Cliffs, 2002.

[18] N Friedland and D Adam. Automatic ventricular cavity boundary detection from sequential ultrasound images using simulated annealing. *Medical Imaging, IEEE Transactions on*, 8(4):344–353, 1989.

[19] S Ghosh and SK Dubey. Comparative analysis of K-means and fuzzy C-means algorithms. *International Journal of Advanced Computer Science and Applications*, 4(4):35–39, 2013.

[20] V Grau, AUJ Mewes, M Alcaniz, R Kikinis, and SK Warfield. Improved watershed transform for medical image segmentation using prior information. *Medical Imaging, IEEE Transactions on*, 23(4):447–458, 2004.

[21] J-F Greeff. An overview of the statistical pattern recognition techniques. Hounours Research Report, The University of Pretoria, 2012.

[22] RM Haralick and LG Shapiro. Image segmentation techniques. In *1985 Technical Symposium East*, volume 548, pages 2–9. International Society for Optics and Photonics, 1985.

[23] RM Haralick, SR Sternberg, and X Zhuang. Image analysis using mathematical morphology. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-9(4):532–550, 1987.

[24] RM Haralock and LG Shapiro. *Computer and Robot Vision*. Addison-Wesley Longman Publishing Co., Inc., 1991.

[25] EW Jacobs, Y Fisher, and RD Boss. Image compression: A study of the iterated transform method. *Signal Processing*, 29(3):251–263, 1992.

[26] T Kanungo, DM Mount, NS Netanyahu, CD Piatko, R Silverman, and AY Wu. An efficient k-means clustering algorithm: Analysis and implementation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(7):881–892, 2002.

[27] B Karasulu and S Korukoglu. A simulated annealing-based optimal threshold determining method in edge-based segmentation of grayscale images. *Applied Soft Computing*, 11(2):2246–2259, 2011.

[28] A Kelemen, G Székely, and G Gerig. Elastic model-based segmentation of 2-D and 3-D neuroradiological data sets. *Medical Imaging, IEEE Transactions on*, 18(10):828–839, 1999.

[29] M Kunt, M Benard, and R Leonardi. Recent results in high-compression image coding. *Circuits and Systems, IEEE Transactions on*, 34(11):1306–1336, 1987.

[30] J Liang, J. Piper, and J-Y Tang. Erosion and dilation of binary images by arbitrary structuring elements using interval coding. *Pattern Recognition Letters*, 9(3):201–209, 1989.

[31] JS Lim. *Two-dimensional Signal and Image Processing*, volume 1. Englewood Cliffs, NJ, Prentice Hall, 1990.

[32] T Lindeberg. Scale-space for discrete signals. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(3):234–254, 1990.

[33] T Lindeberg and J Gårding. Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. *Image and Vision Computing*, 15(6):415–434, 1997.

[34] C Lorenz, I-C Carlsen, TM Buzug, C Fassnacht, and J Weese. Multi-scale line segmentation with automatic estimation of width, contrast and tangential direction in 2D and 3D medical images. In J Troccaz, E Grimson, and R Mosges, editors, *First Joint Conference Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery Grenoble*, pages 233–242. Springer, March 19-22 1997.

[35] WY Ma and BS Manjunath. Edge flow: a framework of boundary detection and image segmentation. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 744–749. IEEE, 1997.

[36] R Maini and H Aggarwal. Study and comparison of various image edge detection techniques. *International Journal of Image Processing*, 3(1):1–11, 2009.

[37] J Mao and AK Jain. Texture classification and segmentation using multiresolution simultaneous autoregressive models. *Pattern Recognition*, 25(2):173–188, 1992.

[38] G Matheron. *Random Sets and Integral Geometry*. John Wiley and Sons, 1975.

[39] SK Maxwell, GL Schmidt, and JC Storey. A multi-scale segmentation approach to filling gaps in Landsat ETM+ SLC-off images. *International Journal of Remote Sensing*, 28(23):5339–5356, 2007.

[40] PP Mohanta, DP Mukherjee, and ST Acton. Agglomerative clustering for image segmentation. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 1, pages 664–667. IEEE, 2002.

[41] SR Patil, R Chavan, A Shinde, TJ Joy, and N Wakale. Intelligent cooking providing automatic time and temperature setting using image processing with wide range of recipes for microwave ovens. *International Journal of Science and Modern Engineering*, 2(1):11–16, 2013.

[42] TW Ridler and S Calvard. Picture thresholding using an iterative selection method. *IEEE Transactions on Systems, Man and Cybernetics*, 8(8):630–632, 1978.

[43] JC Russ and RP Woods. The image processing handbook. *Journal of Computer Assisted Tomography*, 19(6):979–981, 1995.

[44] F Russo. An image enhancement technique combining sharpening and noise reduction. *Instrumentation and Measurement, IEEE Transactions on*, 51(4):824–828, 2002.

[45] PK Sahoo, SAKC Soltani, and AKC Wong. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2):233–260, 1988.

[46] J Serra. Introduction to mathematical morphology. *Computer Vision, Graphics, and Image Processing*, 35(3):283–305, 1986.

[47] M Sezgin and B Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–168, 2004.

[48] J Sijbers, P Scheunders, M Verhoye, A Van der Linden, D Van Dyck, and E Raman. Watershed-based segmentation of 3D MR data for volume quantization. *Magnetic Resonance Imaging*, 15(6):679–688, 1997.

[49] W-J Song. Method for enhancing image data by noise reduction or sharpening, November 1988. US Patent 4,783,840.

[50] M Sonka, V Hlavac, and R Boyle. *Image Processing, Analysis, and Machine Vision*. Cengage Learning, 2014.

[51] DS Taubman and MW Marcellin. *JPEG2000 Image Compression Fundamentals, Standards and Practice: Image Compression Fundamentals, Standards, and Practice*, volume 1. Springer Science and Business Media, 2002.

[52] K. Van Leemput, F Maes, D Vandermeulen, and P Suetens. Automated model-based tissue classification of MR images of the brain. *Medical Imaging, IEEE Transactions on*, 18(10):897–908, 1999.

[53] Z Wang, AC Bovik, HR Sheikh, and EP Simoncelli. Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, April 2004.

[54] Z Wang, Q Li, and X Shang. Perceptual image coding based on a maximum of minimal structural similarity criterion. In *IEEE International Conference Image Processing*, volume 2, pages II–121. IEEE, 2007.

[55] Z Wang, EP Simoncelli, and AC Bovik. Multiscale structural similarity for image quality assessment. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*, volume 2, pages 1398–1402. IEEE, November 2003.

[56] PD Wellner. Adaptive thresholding for the digitaldesk. Technical Report EPC1993-110, Xerox, 1993.

[57] DL Wilson, AJ Baddeley, and RA Owens. A new metric for grey-scale image comparison. *International Journal of Computer Vision*, 24(1):5–17, 1997.

[58] K-L Wu and M-S Yang. Alternative c-means clustering algorithms. *Pattern recognition*, 35(10):2267–2278, October 2002.

[59] B Yogamangalam, Karthikeyan R, and B Karthikeyan. Segmentation techniques comparison in image processing. *International Journal of Engineering and Technology*, 5(1):307–313, 2013.

[60] IT Young and J Van Vliet. Recursive implementation of the Gaussian filter. *Signal Processing*, 44(2):139–151, 1995.

[61] H Zhang, JE Fritts, and SA Goldman. Image segmentation evaluation: A survey of unsupervised methods. *Computer Vision and Image Understanding*, 110(2):260–280, 2008.

[62] Y Zhang and JJ Gerbrands. Segmentation evaluation using ultimate measurement accuracy. In *Information Theory Group, Department of Electrical Engineering*, pages 449–460. International Society for Optics and Photonics, 1992.

[63] YJ Zhang. A survey on evaluation methods for image segmentation. *Pattern Recognition*, 29(8):1335–1346, 1996.

Appendix

Metric code

Baddeley's binary code

Main Program

The function `baddeley` calculates Baddeley's Δ_b metric as defined in Section 3.1.

$$\Delta_b(f_1, f_2) = \left[\frac{1}{card(X) \times c^p} \left(\sum_{x \in X} |d^*(x, A_1) - d^*(x, A_2)|^p + \sum_{x \in X} |d^*(x, B_1) - d^*(x, B_2)|^p \right) \right]^{\frac{1}{p}} \quad 1 \leq p \leq \infty.$$

The `Matlab` code requires that c and p are specified before the function Baddeley is initialized. The research report's results have been computed for $c = 3$ and $p = 1$.

```
function d = baddeley(f1, f2, c, p)

%Input variables:
%f1: image f
%f2: image g
%c=3, c is the bounding constant ensuring no points further than the paths
%     of length c contribute to the metric.
%p=1, p controls the relative weight of errors of different magnitudes

%Output Variables:
%baddeley: Baddeley's binary metric which is a measure that evaluates the
%   numerical distance between f1 and f2 i.e.the function calculates
%   Baddeley's metric for the two images being compared.
%   This is between the white and black of the respective images


[W1, B1] = setblackorwhite(f1); %See Above for set...white function description.
[W2, B2] = setblackorwhite(f2);
ww1 = size(W1); %number of white pixels in image f1, which is a matrix
ww2 = size(W2); %number of white pixels in image f2
bb1 = size(B1); %number of black pixels in image f1
bb2 = size(B2); %number of black pixels in image f2
[n1,m1] = size(f1); %the number of rows and col in image f1
[n2,m2] = size(f2); %the number of rows and col in image f2

%Checking images sizes are the same
if size(f1) == size(f2)
    %For the comparision of the white pixels of image f1 and f2
    dw = 0; %initialize dw=0
    db = 0;
    for i = 1 : n1
        for j = 1 : m1
            x = [i j]; %pixel position

            %Applying Baddeley's metric for white pixels
            if isempty(W1) == 1
                distx_W1 = 1;
```

```
            else [ distx_W1 ] = min_dist (x , W1);
            end
            if isempty(W2) == 1
                distx_W2 = 1;
            else [ distx_W2 ] = min_dist (x , W2);
            end

        star_W1 = min(distx_W1 ,c);
        star_W2 = min(distx_W2 ,c);

            dw = dw + (abs(star_W1 - star_W2))^(p);

            %Applying Baddeley's metric for black pixels
            if isempty(B1) == 1
                distx_B1 = 1;
            else [ distx_B1 ] = min_dist (x , B1);
            end
            if isempty(B2) == 1
                distx_B2 = 1;
            else [ distx_B2 ] = min_dist (x , B2);
            end

         star_B1 = min(distx_B1 ,c);
         star_B2 = min(distx_B2 ,c);

            db = db + (abs(star_B1 - star_B2))^(p); %loop for the sum
        end
    end
    d = ((dw+db)/(n1*m1*c^p))^(1/p);
    else disp('Please check the matrix sizes: f1 and f2 must be the same size');
end
```

The function `setblackorwhite` which is called in the function `baddeley` determines the set of black and white pixel positions present in the respective images for which the Baddeley's Δ_b metric is calculated.

```
function [W, B] = setblackorwhite(f)

%Input variable
%f: an image

%Output variable
%setblackorwhite: determines the set of black and white pixels present in
%  the image i.e. extracts all the black and white pixel
%  positions

ind_b = find(f == 0);
ind_w = find(f == 1);

[ B1 , B2 ] = ind2sub([n,m],ind_b);
[ W1 , W2 ] = ind2sub([n,m],ind_w);

B = [ B1 , B2 ];
W = [ W1 , W2 ];
```

Lastly the function `min_dist` is required to be computed in order to determine Δ_b. The function `min_dist` determines $d^*(x, A) = \min\{d(x, a) : a \in A\}$ where $d(x, a) = |x_1 - a_1| + |x_2 - a_2|$ and $x = (x_1, x_2)$and $a = (a_1, a_2)$.

```
function [ distx_A ] = min_dist ( x , A )

%Input variables:
%x: all the pixels present in the image, x=(x_1, x_2)
%A: the set of pixel positions, depending on prior code,
%   it is the set of black or set of white pixel positions

%Output variable:
%min_dist: calculates the minimum distance from the current pixel posistion x
%   to A- the set of black or white pixel positions
%       and determines the minimum calculated distance

x_dist = abs(x(1) - A(:,1)) + abs(x(2) - A(:,2));

distx_A = min(x_dist);

end
```

Wilson et al's greyscale code

Main program

The function `delta_g_subf_g` determines Wilson's Δ_g metric as defined in Section 3.2 .

$$\Delta_g(\Gamma_f, \Gamma_g) = \left[\frac{1}{n(X) \times n(Y) \times 256 \times c^p} \sum_{x \in X} \sum_{y \in Y} |d^* ((x, y), \Gamma_f) - d^* ((x, y), \Gamma_g)|^p \right]^{\frac{1}{p}} \text{ for } 1 \leq p < \infty.$$

In essence the function `delta_g_subf_g` determines the Δ_g metric for different subgraphs for images f and g.

```
function metric_g = delta_g_subf_g( f, g, c, p)

%Function determines the greyscale metric for the different subgraphs for f and g

%Input variables:
%f: image f
%g: image g
%c=3, c is the bounding constant ensuring no points further than the paths
%     of length c contribute to the metric.
%p=1, p controls the relative weight of errors of different magnitudes

%Output variables:
%delta_g_subf_g: determines the greyscale metric which is the distance between two
%     greyscale images f and g, which is further defined as the
%     distance between the respective subgraphs of f and g


sum=0;
```

```
test=0;
gammaf=Gamma_f(f,(0:255));
gammag=Gamma_f(g,(0:255));

message = 'Finished with the Gammas'

map_in = Map_Maker(c);
s_map = size(map_in);
cen = s_map(1) - (s_map(1) - 1)/2;
[I,J] = find(map_in == 1);
M_full = transpose([I-cen,J-cen]);

if size(f) == size(g)
for i = 1 : size(f,1)
    i
    for j = 1 : size(f,2)
        j

        for y = 0 : 255
            df_star = min(wholevolsub_g(i, j, y, M_full, size(f), gammaf),c)
            dg_star = min(wholevolsub_g(i, j, y, M_full, size(g), gammag),c)
            abs(df_star-dg_star)
            sum = sum +  (abs(df_star-dg_star))^p
            test= test + 1;
        end
    end
end
metric_g=(1/(size(f,1)*size(f,2)*(256)*c)*sum)^(1/p)
else disp('Please check the matrix sizes: f and g must be the same size');
end
```

The function `Gamma_f` is required by the function `delta_g_subf_g` in order to determine the subgraph Γ_f and Γ_g for images f and g.

```
function [ Gam_f ] = Gamma_f( f , Y )

%Input variables:
%f: an image
%Y: is the range of greyscale luminance values i.e. Y={0,1,2...,255}

%Output variables:
%Gamma_f: determines the subgraphs for the respective images compared i.e.
%  Gamma_f determines a subgraph for image f and image g

size_f = size(f);

Gam_f = zeros(size_f(1)*size_f(2)*numel(Y),3);
g = 1;
for k = 1 : numel(Y);
    ind = find( f >= Y(k) );
    [I,J] = ind2sub(size_f,ind);

    Gam_f(g:g+numel(ind)-1,:) = [ [I,J] , ones(numel(ind),1)*Y(k) ];
```

```
    g = g + numel(ind);
end

Gam_f = Gam_f(1:sum(Gam_f(:,1) > 0),:);

end
```

The function `Map_Maker` creates representative design of a c-connectivity structure, in the report's case since $c = 3$. The 3-connectivity structure moves through every pixel within the image to decrease the amount of computations.

```
function [ map ] = Map_Maker( lim )

%Input variable:
%lim: parameter that determines the size of the the c-connectivity structure that
%     moves through every pixel within the image

%Output variable:
%Map_Maker: creates c-connectivity structure that moves through every pixel
%    present in the image, essentially to decrease computations

n = 2*lim+1;
map = zeros(n,n);
for i = 1 : n
    if i < lim + 1
        map(i,(lim+2-i:lim+i)) = 1;
    elseif i > lim + 1
        k = n - i + 1;
        map(i,(lim+2-k:lim+k)) = 1;
    else
        map(i,(1:n)) = 1;
    end
end

end
```

The function `wholevolsub_g` determines the distance between a point $(x, y) \in X \times Y$ i.e. the whole volume $X \times \{0, 1, \ldots, 255\}$, and the subgraph $\Gamma_f \subseteq X \times Y$:

```
function dis_0=wholevolsub_g(i1, j1, y1, M_full, size_f, gammaf)

%Input variables:
%Note that x=(x1,x2) the pixel positions present in an image
%i1: x1
%j1: x2
%y1: the greyscale luminance value for pixel position x=(x1,x2)
%M_full:
%size_f: the dimensions of the considered image
%gammaf: subgraph of image

%Output variables:
%wholevolsub: determines the distance between a point (x,y) i.e. (i1, j1)
%      the subgraph
```

```
%Determining the values at which the frame can take for incomplete maps.
D_H = (1 : size_f(2));
D_V = (1 : size_f(1));

TestM = M_full + repmat([i1;j1],1,size(M_full,2));
loc_i = unique(intersect(TestM(1,:),D_V));
loc_j = unique(intersect(TestM(2,:),D_H));

gammaf_new = gammaf( (gammaf(:,1) <= max(loc_i)) & (gammaf(:,1) >= min(loc_i))
& (gammaf(:,2) <= max(loc_j)) & (gammaf(:,2) >= min(loc_j)), : );

y_diff = abs(gammaf_new(:,3)-y1);
dx_x_dash =  abs(i1 - gammaf_new(:,1)) + abs(j1 - gammaf_new(:,2));

true_dis = (dx_x_dash > y_diff).*dx_x_dash + (dx_x_dash <= y_diff).*y_diff;
dis_0 = min(true_dis);

end
```

The function `subgdist` calculates the distance between the points in a subgraph defined by the formula $d\left((x,\, y)\,,\,(x',\, y')\right) = \max\left\{d\left(x,\, x'\right)\,,\,|y - y'|\right\}$.

```
%STEP 3
function subgdist  = subgraphdist( i1, j1, y1, i2, j2, y2)


%Input variables:
%Note that x=(x1,x2) the pixel positions present in an image 1 and x'=(x1',x2')
%i1: x1 in image f
%i2: x1 in image f
%y1: the greyscale luminance value for pixel position x=(x1,x2)
%i1: x1 in image g
%i2: x1 in image g
%y2: the greyscale luminance value for pixel position x'=(x1',x2')

%Output variables:
%subgdist: determines the distance between points in a subgraph
%          i.e. The function subgdist defines the distance between points in
%          a subgraph def2

ydis=abs(y1-y2);
d=fourconnecdistgrey([i1,j1],[i2,j2]);
subgdist = max(d , ydis);
```

The function `fourconnecdisgrey` determines the distance between pixels within an image utilizing the c-connectivity, where $c = 3$.

```
function d = fourconnecdistgrey(x,a)

%Input variables:
%x: all the pixels present in the image, x=(x_1, x_2)
%a: the set of pixel positions
```

```
%Output variables
%d: is the distance between two pixels using 4 connectivity


%d is the istance between two pixels using 4 connectivity
%x = (x_1,x_2), a = (a_1,a_2)

d = abs(x(1)-a(1)) + abs(x(2)-a(2));
end
```

SSIM code

The function `mssim` determines the SSIM index defined in Wang et al [53] as defined in Section 3.3

$$SSIM\left(f_8,\, g_8\right) = \frac{\left(2\mu_{f_8}\mu_{g_8} + C_1\right)\left(2\sigma_{f_8 g_8} + C_2\right)}{\left(\mu_{f_8}^2 + \mu_{g_8}^2 + C_1\right)\left(\sigma_{f_8}^2 + \sigma_{g_8}^2 + C_2\right)}.$$

Main program

```
function [mssim, ssim_map] = ssim(img1, img2, K, window, L)

% ========================================================================
% SSIM Index with automatic downsampling, Version 1.0
% Copyright(c) 2009 Zhou Wang
% All Rights Reserved.
%
% ----------------------------------------------------------------------
% Permission to use, copy, or modify this software and its documentation
% for educational and research purposes only and without fee is hereby
% granted, provided that this copyright notice and the original authors'
% names appear on all copies and supporting documentation. This program
% shall not be used, rewritten, or adapted as the basis of a commercial
% software or hardware product without first obtaining permission of the
% authors. The authors make no representations about the suitability of
% this software for any purpose. It is provided "as is" without express
% or implied warranty.
%----------------------------------------------------------------------
%
% This is an implementation of the algorithm for calculating the
% Structural SIMilarity (SSIM) index between two images
%
% Please refer to the following paper and the website with suggested usage
%
% Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image
% quality assessment: From error visibility to structural similarity,"
% IEEE Transactios on Image Processing, vol. 13, no. 4, pp. 600-612,
% Apr. 2004.
%
% http://www.ece.uwaterloo.ca/~z70wang/research/ssim/
%
% Note: This program is different from ssim_index.m, where no automatic
% downsampling is performed. (downsampling was done in the above paper
```

```
% and was described as suggested usage in the above website.)
%
% Kindly report any suggestions or corrections to zhouwang@ieee.org
%
%----------------------------------------------------------------------
%
%Input : (1) img1: the first image being compared
%        (2) img2: the second image being compared
%        (3) K: constants in the SSIM index formula (see the above
%            reference). default value: K = [0.01 0.03]
%        (4) window: local window for statistics (see the above
%            reference). default window is Gaussian given by
%            window = fspecial('gaussian', 11, 1.5);
%        (5) L: dynamic range of the images. default: L = 255
%
%Output: (1) mssim: the mean SSIM index value between 2 images.
%            If one of the images being compared is regarded as
%            perfect quality, then mssim can be considered as the
%            quality measure of the other image.
%            If img1 = img2, then mssim = 1.
%        (2) ssim_map: the SSIM index map of the test image. The map
%            has a smaller size than the input images. The actual size
%            depends on the window size and the downsampling factor.
%
%Basic Usage:
%   Given 2 test images img1 and img2, whose dynamic range is 0-255
%
%   [mssim, ssim_map] = ssim(img1, img2);
%
%Advanced Usage:
%   User defined parameters. For example
%
%   K = [0.05 0.05];
%   window = ones(8);
%   L = 100;
%   [mssim, ssim_map] = ssim(img1, img2, K, window, L);
%
%Visualize the results:
%
%   mssim                        %Gives the mssim value
%   imshow(max(0, ssim_map).^4)  %Shows the SSIM index map
%======================================================================


if (nargin < 2 || nargin > 5)
   mssim = -Inf;
   ssim_map = -Inf;
   return;
end

if (size(img1) ~= size(img2))
   mssim = -Inf;
   ssim_map = -Inf;
```

```
   return;
end

[M N] = size(img1);

if (nargin == 2)
   if ((M < 11) || (N < 11))
   mssim = -Inf;
   ssim_map = -Inf;
      return
   end
   window = fspecial('gaussian', 11, 1.5); %
   K(1) = 0.01; % default settings
   K(2) = 0.03; %
   L = 255;                                  %
end

if (nargin == 3)
   if ((M < 11) || (N < 11))
   mssim = -Inf;
   ssim_map = -Inf;
      return
   end
   window = fspecial('gaussian', 11, 1.5);
   L = 255;
   if (length(K) == 2)
      if (K(1) < 0 || K(2) < 0)
   mssim = -Inf;
    ssim_map = -Inf;
    return;
      end
   else
   mssim = -Inf;
    ssim_map = -Inf;
   return;
   end
end

if (nargin == 4)
   [H W] = size(window);
   if ((H*W) < 4 || (H > M) || (W > N))
   mssim = -Inf;
   ssim_map = -Inf;
      return
   end
   L = 255;
   if (length(K) == 2)
      if (K(1) < 0 || K(2) < 0)
   mssim = -Inf;
    ssim_map = -Inf;
    return;
      end
   else
```

```
   mssim = -Inf;
    ssim_map = -Inf;
   return;
   end
end

if (nargin == 5)
   [H W] = size(window);
   if ((H*W) < 4 || (H > M) || (W > N))
   mssim = -Inf;
   ssim_map = -Inf;
      return
   end
   if (length(K) == 2)
      if (K(1) < 0 || K(2) < 0)
   mssim = -Inf;
    ssim_map = -Inf;
    return;
      end
   else
   mssim = -Inf;
    ssim_map = -Inf;
   return;
   end
end


img1 = double(img1);
img2 = double(img2);

% automatic downsampling
f = max(1,round(min(M,N)/256));
%downsampling by f
%use a simple low-pass filter
if(f>1)
    lpf = ones(f,f);
    lpf = lpf/sum(lpf(:));
    img1 = imfilter(img1,lpf,'symmetric','same');
    img2 = imfilter(img2,lpf,'symmetric','same');

    img1 = img1(1:f:end,1:f:end);
    img2 = img2(1:f:end,1:f:end);
end

C1 = (K(1)*L)^2;
C2 = (K(2)*L)^2;
window = window/sum(sum(window));

mu1   = filter2(window, img1, 'valid');
mu2   = filter2(window, img2, 'valid');
mu1_sq = mu1.*mu1;
mu2_sq = mu2.*mu2;
mu1_mu2 = mu1.*mu2;
```

```
sigma1_sq = filter2(window, img1.*img1, 'valid') - mu1_sq;
sigma2_sq = filter2(window, img2.*img2, 'valid') - mu2_sq;
sigma12 = filter2(window, img1.*img2, 'valid') - mu1_mu2;

if (C1 > 0 && C2 > 0)
   ssim_map = ((2*mu1_mu2 + C1).*(2*sigma12 + C2))./
               ((mu1_sq + mu2_sq + C1).*(sigma1_sq + sigma2_sq + C2));
else
   numerator1 = 2*mu1_mu2 + C1;
   numerator2 = 2*sigma12 + C2;
denominator1 = mu1_sq + mu2_sq + C1;
   denominator2 = sigma1_sq + sigma2_sq + C2;
   ssim_map = ones(size(mu1));
   index = (denominator1.*denominator2 > 0);
   ssim_map(index) = (numerator1(index).*numerator2(index))./
                        (denominator1(index).*denominator2(index));
   index = (denominator1 ~= 0) & (denominator2 == 0);
   ssim_map(index) = numerator1(index)./denominator1(index);
end

mssim = mean2(ssim_map);

return
```

Thresholding code

Standard thresholding and Wellner's algorithm

Code courtesy of [21].

Main program

The function `wellner` outputs segmentation images that are result from the standard thresholding and Wellner's algorithm defined in Section 4.3. The image outputted from the the `Matlab` code defined as `img` yields a resultant Wellner algorithm image. Similarly `img3` in the `Matlab` code produces the segmented image from the standard thresholding technique.

```
%Input variable:
%img: the image

%Output variable:
%wellner: outputs an image that has been segmented that results from the
%  Wellner algorithm
%If img is requested as the output for Wellner, obtain the Wellner algorithm segmented image
%and if img3 is requested as the output for Wellner obtain the segmented image with
%standard thresholding

%% import image

out=wellner(img)
if size(size(img)) ~= 2
    img = rgb2gray(img);
end
```

```
img = double(img);

ws = round(size(img,2)/8); %%must be even
if mod(ws,2) ~= 0
    ws = ws + 1;
end

t = 15; %%percentage

%% threshold

n = size(img,1);
m = size(img,2);

bw = zeros(n,m);

for i = 1:n

    if mod(i,2) ~=0
        jStart = 1;
        jEnd = m;
        jBy = 1;
    else
        jStart = m;
        jEnd = 1;
        jBy = -1;
    end


    for j = jStart:jBy:jEnd

        l = j-ws/2;
        u = j+ws/2;

        gs = img(i,max(1,l):min(m,u));

        aveGs = sum(gs) / size(gs,2);

        if img(i,j) > aveGs * (100-t)/100
            bw(i,j) = 1;
        end

    end

end

%%

tt = graythresh(uint8(img));
tt = tt*255;

img2 = img;
img3 = img2;
```

```
for i = 1:size(img2,1)
    for j = 1:size(img2,2)
        if img2(i,j) < tt

            img3(i,j) = 0;
        else
            img3(i,j) = 255;
        end
    end
end

subplot(131)
imshow(uint8(img))
subplot(132)
imshow(uint8(img3))
subplot(133)
imshow(bw)
```

ICM algorithm code

Main program

The function `U` outputs an image that results from the ICM thresholding technique defined in Section 4.4. Code courtesy of Dr. Fabris-Rotelli.

```
function U = main(images,c,beta)

%Input variables:
%images: array of images at specific scales/values
%c: number of classes
%beta: smoothing parameter

%Output variables:
%main

[s1,s2,n] = size(images);
%s1: number of rows in each image
%s2: number of columns in each image
%n: number of images in scale space

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%colours for each class%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
colourdiff = 256/c;
colour = zeros(c,1);
for g = 1 : c % significantly different colours for each class;
    colour(g,1) = colourdiff*g;
end
colour
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%initial cluster centres: as pixel positions;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%initially random cluster centres: placed proportionally on the image left diagonal;
mu = zeros(c,2); %rows = clusters;
for k = 1 : c
    mu(k,:) = [floor(s1/(2*c))*(2*(k-1)+1), floor(s2/(2*c))*(2*(k-1)+1)];
end

mu = KMC(images,c,mu); %update the initial cluster centres using k-means
                        %clustering algorithm;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%cluster centres: as images values;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
newmu = zeros(n,c); %columns = cluster centres;
for k = 1 : c
    newmu(:,k) = images(mu(k,1),mu(k,2),:);
end
mu = newmu;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%First minimization%%second term not included since N_ij^\alpha(k) = 0%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
U = zeros(s1,s2);
for i = 1 : s1
    for j = 1 : s2
        cluster = 1; %initially cluster 1 gives the minimum for pixel (i,j);
        for k = 1 : n
            vec(k) = images(i,j,k) - mu(k,cluster); %#ok<AGROW>
        end
        value = transpose(vec')*vec';
        for k = 2 : c
            for h = 1 : n
                newvec(h) = images(i,j,h) - mu(h,k); %#ok<AGROW>
            end
            newvalue = transpose(newvec')*newvec';
            if newvalue < value
                cluster = k;
                value = newvalue;
            end
        end
        U(i,j) = cluster;
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
diff = norm(mu,2) %inital stop criterion;
v = variance(s1,s2,images,mu,U,n); %initial variance;
Nij = count(U,c,s1,s2); %initial count;
total = 0;
while diff > 1.004 && total < 30
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
    %Minimization Step%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for i = 1 : s1
        for j = 1 : s2
            cluster = 1; %initially cluster 1 does the minimizing for pixel (i,j);
            for k = 1 : n
                vec(k) = images(i,j,k) - mu(k,cluster);
            end
            vvalue = transpose(vec')*vec';
            value = vvalue - beta*v*Nij(i,j,cluster);
            for k = 2 : c
                for h = 1 : n
                    vec(h) = images(i,j,h) - mu(h,k);
                end
                vvalue = transpose(vec')*vec';
                newvalue = vvalue - beta*v*Nij(i,j,k);
                if newvalue < value
                    cluster = k;
                    value = newvalue;
                end
            end
            U(i,j) = cluster;
        end
    end
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %updating centroids%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    newmu = updatecentre(images,U,c,s1,s2,n);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %updating the variance%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    v = variance(s1,s2,images,mu,U,n);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %updating count%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    Nij = count(U,c,s1,s2);
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    newdiff = norm(newmu - mu,2)
    mu = newmu;
    total = total + 1;
    if newdiff == diff
        total = 20;
    end
    diff = newdiff;
end
U = finalclassify(U,colour,s1,s2);
% for i = 1 : s1
%     for j = 1 : s2
%         if i == 1 || i == s1 || j == 1 || j == s2
```

```
%             U(i,j) = colour(1);
%         end
%     end
% end
mmshow(U);
```

The function `KMC`:

```
function cen = KMC(images,c,cen)

%Input variables:
%images: array of images at specific scales/values
%c: number of clusters
%cen : cluster centers - Initially chosen proportionally along the left
%diagonal;

%Output variables:
%KMC

[s1,s2,n] = size(images);
%s1: number of rows in each image
%s2: number of columns in each image
%n: number of images in scale space

N = 0;
d0 = distance(images,cen,c,s1,s2); %s1 x s2 x c matrix, third dimension
                                   %indicates distance to which centroid;
U0 = classify(d0,s1,s2,c);
v = 0;
while (N < 100) && (v < c)
    newcen = centroids(U0,c,cen);
    d = distance(images,newcen,c,s1,s2);
    U = classify(d,s1,s2,c);
    v = movement(newcen,cen,c) %v = c implies no movement of centroids
                               %i.e. convergence;
    U0 = U;
    N = N + 1;
    cen = newcen;
end

figure
UU = colour(U,s1,s2,c);
mmshow(UU);
```

The function `distance`:

```
function d = distance(images,cen,c,s1,s2)
%distance is caluclated as an absolute sum;

%Input variables:
%images: array of images at specific scales/values
%cen: centroid centres c x 2 as pixel positions
%c: number of clusters
%s1: number of rows in each image
```

```
%s2: number of columns in each image

%Output variables:
%distance is caluclated as an absolute sum;

d = zeros(s1,s2,c); %3D matrix with distances to each centroid in the third dimension;
for i = 1 : s1
    for j = 1 : s2
        for k = 1 : c
            d(i,j,k) = sum(abs(images(i,j,:) - images(cen(k,1),cen(k,2),:)));
        end
    end
end
```

The function `classify`:

```
function U = classify(d,s1,s2,c)

%Input variables:
%d: distance matrix s1 x s2 x c
%s1: number of rows in each image
%s2: number of columns in each image
%c: number of clusters

%Output variables:
%classify

U = zeros(s1,s2,c);

for i = 1 : s1
    for j = 1 : s2
        [m,mm] = min(d(i,j,:));%m: min values; mm: min indices;
        U(i,j,mm) = 1;%indicates which centroid (i,j) is closest too;
    end
end
```

The function `centroids`:

```
function newcen = centroids(U,c,cen)

%Input variables:
%U: current clustering
%c: number of clusters
%cen: centroid centres c x 2 as pixel positions

%Output variables:
%newcen: calculates centroids of the clusters

newcen = zeros(c,2);%pixel positions of the new centroids;

for k = 1 : c
    if sum(sum(U(:,:,k))) ~= 0 %if there are pixels in the cluster;
        newcen(k,:) = mmblob(U(:,:,k),'centroid', 'data');
    else %leave the center as is
```

```
        newcen(k,:) = cen(k,:);
    end
end
```

The function `movement`:

```
function v = movement(cen,newcen,c)
%how many centers do not change;

Input variables:
%cen: centroid centres c x 2 as pixel positions
%newcen: calculates centroids of the clusters
%c: number of clusters

%Output variables:
%movement: determines how many centers do not change




v = zeros(c,1);
for k = 1 : c
    v(k,1) = isequal(cen(k,:),newcen(k,:));
end
v = sum(v);
```

The function `colour`:

```
function UU = colour(U,s1,s2,c)

%Input variables:
%U: current clustering
%s1: number of rows in each image
%s2: number of columns in each image
%c: number of clusters

%Output variables:
%colour



UU = zeros(s1,s2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%colours for each class%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
colourdiff = 256/c;
colour = zeros(c,1);
for g = 1 : c % significantly different colours for each class;
    colour(g,1) = colourdiff*g;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%assigning colours to each class
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i = 1 : s1
    for j = 1 : s2
        for k =  1 : c
            if U(i,j,k) == 1
                UU(i,j) = colour(k,1);
            end
        end
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

The function `variance`:

```
function v = variance(s1,s2,images,mu,U,nn)

%Input variables:
%s1: number of rows in each image
%s2: number of columns in each image
%images: array of images at specific scales/values
%mu: c cluster means of dimension n each;
%U: current clustering;
%nn: number of clusters;

%Output variables:
%variance


N = s1*s2; %total number of pixels;
v = 0;
for i = 1 : s1
    for j = 1 : s2
        for k = 1 : nn
            vec(k) = images(i,j,k) - mu(k,U(i,j));
        end
        value = transpose(vec')*vec';
        v = v + value;
    end
end

v = v/N;
```

The function `count`:

```
function Nij = count(U,c,s1,s2)

%Input variables:
%U: current clustering
%c: number of clusters
%s1: number of rows in each image
%s2: number of columns in each image

%Output variables:
```

```
%count

Nij = zeros(s1,s2,c); %number of neighbours of each pixel in cluster c;
for i = 1 : s1
    for j = 1 : s2
        y = ICMnbr(i,j,s1,s2); %neighbours of pixel (i,j) via 4- or 8-connectivity;
        for k = 1 : size(y,1)
            Nij(i,j,U(y(k,1),y(k,2))) = Nij(i,j,U(y(k,1),y(k,2)))+1;
        end
    end
end
```

The function `ICMnbr`:

```
function y = ICMnbr(i,j,N,M)

%Input variables:
%pixel (i,j)
%N: number of rows in the image
%M: number of columns in the image
%y: neighbours of pixel (i,j) (4-connectivity)

%Output variables:
%ICMnbr

k=0;
if i>1
    k=k+1;
    y(k,1)=i-1;y(k,2)=j;
end
if j>1
    k=k+1;
    y(k,1)=i;y(k,2)=j-1;
end
if i<N
    k=k+1;
    y(k,1)=i+1;y(k,2)=j;
end
if j<M
    k=k+1;
    y(k,1)=i;y(k,2)=j+1;
end

% %Include these as well for 8-connectivity%
% if i > 1 && j > 1
%     k = k+1;
%     y(k,1) = i-1; y(k,2) = j-1;
% end
% if i > 1 && j < M
%     k = k+1;img3
%     y(k,1) = i-1; y(k,2) = j+1;
% end
% if j > 1 && i < N
```

```
%     k = k+1;
%     y(k,1) = i+1; y(k,2) = j-1;
% end
% if i < N && j < M
%     k = k+1;
%     y(k,1) = i+1; y(k,2) = j+1;
% end
```

The function `updatecenter`:

```
function newmu = updatecentre(images,U,c,s1,s2,n)

%Input variables:
%images: scale space
%U: current clustering
%c: number of clusters
%s1: number of rows in each image
%s2: number of columns in each image
%n: number of images in scale space
img3
%Output variables:
%updatecentre


newmu = zeros(n,c);
totals = zeros(c,1); %number of pixels in each cluster;

for i = 1 : s1
    for j = 1 : s2
        totals(U(i,j)) = totals(U(i,j)) +1;
        for h = 1 : n
            newmu(h,U(i,j)) = newmu(h,U(i,j)) + images(i,j,h);
        end
    end
end
for k = 1 : c
    if totals(k) ~= 0
        newmu(:,k) = newmu(:,k)/totals(k);
    end
end
```

The function `finalclassify`:

```
function U = finalclassify(U,colour,s1,s2)

%Input variables:
%U: current clustering
%colour
%s1: number of rows in each image
%s2: number of columns in each image

%Output variables:
%finalclassify;
```

```
for i = 1 : s1
    for j = 1 : s2
        U(i,j) = colour(U(i,j));
    end
end
```

Wrapped distributions

Clemence Kwinje 11327482

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mr. M.T Loots

Department of Statistics, University of Pretoria

2 November 2015

Abstract

Directional statistics is another avenue of statistics which caters for directional random variables. Directional random variables are variables which rotates in circles of unit radius. Directional random variables are also those variables measured with both magnitude and direction from the point of origin. Wrapped distributions are the transformed distributions from our ordinary distributions to cater for directional random variables. Some of the stipulated instruments used to measure these variables are, the compass and the clock. In this report, an analysis and overview of wrapped distributions are done.

Declaration

I, *Clemence Rangarirai Kwinje,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Clemence Rangarirai Kwinje

_____
Theodor Loots

___2 November 2015_____

Acknowledgements

Special thanks to my supervisor, Theodor Loots, for his optimum involvement. He made this research possible.

Contents

1 Introduction

Studies of directional statistics started in the mid 18th century. The proposition to test for the uniformity of unit vectors on circles using normal vectors was established in 1743 by D. Bernoulli [9]. The distribution of variables on the sphere using a characterization of the normal distribution of the variable x on the real line was established by Von Mises in 1918 [9]. The resulting probability density function is given below

$$g(\theta, \mu, \kappa) = (1/(2\pi I(\kappa))) * exp(cos(\theta - \mu))$$

where I is the initial caliber of the Bessel function which denotes the modified Bessel function of the first kind and order 0, which is given by

$$I = (1/2\pi)(\int_0^{2\pi} exp(kcos(\theta)))d\theta.$$

This is the probability density function of the Von Mises M(μ,κ) with parameters μ and κ [9].

Directional or circular data comes from various arenas. The compass and the clock are the main primary instruments for measuring directional statistics. Directions of birds moving from one place to another, Protractor and spirit level readings are other examples of directional statistics. Arrival times of presidents at a summit, or times of the year or month of certain events is an example of clock measurements. Directional data can be regarded as any part of a circle with a radius equal to one or of a vector with length equal to one [5]. It is important to set the starting point and direction of the circle, from which all directional observations can then be attained by measuring the angle from the starting point to any point on the circle [5]. Directional or circular data is usually measured in degrees. However, it is sometimes important to measure in radians.

Since the surface of our planet is almost in a shape of a sphere, spherical statistics emanates in massive proportions in the sciences of our planet earth. Take for example, a very delicate part of an earthquake, the epicenter, which is found on our planet's surface, vertically above the emanating point of the quake [8]. Some spherical/circular statistical data where the results are distinct points on the earth's surface, evolves in the approximation of the relative rotations of tectonic plates [8].

Directional statistics data results are also implemented to hypothesize where palaeomagnetic fields will be accelerating towards [2]. The foundation of analysis of palaeomagnetic was a massive accelerator in the growth of the study of directional/circular observation [2]. Directional data emanates in the invention of different geological procedures, since these encompasses maneuvering matter from one designated destination to another [2].

Wind directions brings to the fore a real or basic sense of directional or circular statistical data [3]. A distribution of wind direction may emanate either as a marginal distribution of the wind speed and direction, or as a conditional distribution for a given speed [3]. Some directional or circular data which comes from meteorology encompasses the times of hurricane occurrence in a day [3].

The study of animal migrations has led to the study of directional or circular statistics [1]. Usual questions are (i) whether the animals actually maneuver in a designated direction and (ii) whether there exists a uniform distribution in the directions of travel [1]. Solutions to the concluding questions are very important in efforts to see if the animals use any clues, such as the moon's direction.

Circular statistics data spearheaded the evolution of one of the most important distributions in fractional parts of atomic weights [9]. Prior to the innovation of isotopes, it was trialed that measured atomic weights are integers subject to error [9]. Von Mises (1918) suggested examining the inference by testing whether or not the complementary distributions on the circle has a mode at 0 degrees [9]. Circular data is also used in the development of mental maps which human beings implement to present their surroundings [9].

Unfortunate events of a certain disease such as deaths (or stroke because of the disease) at different periods of year gives directional data. Directional statistical data can also emanate from vector cardiology [8]. In the context of vector cardiograms arenas, data concerning electrical activity in a heart beat is explained in terms of near-planar orbit in three dimensional space [8].

Information on distance is not readily available for astronomical objects. A lot of observations are of points on the celestial sphere, and so provide directional data [9]. Orbits of planets, with a specified order of rotation, can be seen as points on the sphere [9]. This further enhances the importance of directional statistics.

2 Background theory

The wrapping method is one of the many ways used to study directional data. If a known distribution on the real line is given then, it is possible to wrap that distribution on the circumference of a circle with radius equal to one. Literally, it implies that if the random variable X has distribution function $F(x)$, then Xw will be the wrapped random variable of the wrapped distribution defined by

$$Xw = X(mod(2\pi))$$

with the distribution function of Xw as

$$Fw(\theta) = \sum_{k=1}^{\infty}[F(\theta + 2\pi k) - F(2\pi k)], \quad k = 0, \pm 1, \pm 2, \pm 3, ...$$

We are accumulating probability over all the overlapping points $x = \theta, \theta \pm 2\pi, \theta \pm 4\pi, ...$, using the above approach [6]. So if $g(\theta)$ represents a density function which is circular and $f(x)$ is the density function of the random variable X, we have

$$g(\theta) = \sum_{k=1}^{\infty} f(\theta + 2\pi k)$$

All wrapped distributions can be built by the given technique. In particular, if a random variable, X, attains a concentrated distribution on the points

$$x = k/(2\pi m), \quad k = 0, \pm 1, \pm 2, \pm 3....$$

and m is an integer, the probability function of Xw is as follows

$$P(Xw = ((2\pi rk)/m) = \sum_{k=1}^{\infty} p(r + km) \quad k = 0, 1, 2, 3.....m - 1,$$

where p is a function of the probability of the random variable X [6]. Since there is no general way of calculating moments, specific distributions are going to be used to explain how we attain those methods of moments.

3 Wrapped distributions

Three wrapped distributions will be investigated specifically wrapped geometric, wrapped discrete skew Laplace and wrapped t-distribution.

3.1 Wrapped geometric distribution

Take into consideration the geometric distribution on positive integers with parameter ξ [4]. The probability mass function of the geometric distribution is given by

$$p(x; \xi) = (1 - \xi)^x \xi, \quad x = 0, 1, 2, ...; \xi > 0.$$

Now, the probability function of Xw is set as

$$P(Xw = \frac{2\pi r}{m}) = \sum_{k=-\infty}^{\infty} p(r + km; \xi), \quad r = 0, 1, 2, 3, ..., m - 1$$

This implies that,

$$P(Xw = \frac{2\pi r}{m}) = \sum_{k=0}^{\infty}(1 - \xi)^{r+km}\xi = \frac{\xi(1 - \xi)^r}{1 - (1 - \xi)^m}$$

whereas m is an element of natural numbers and $r = 0, 1, 2, 3, ..., m-1$ [4]. Repeatedly,

$$\sum_{k=0}^{m-1} \frac{\xi(1-\xi)^r}{1-(1-\xi)^m} = 1.$$

Indeed, therefore, $P(\cdot)$ is a probability mass function.

Lets say $\varphi(t)$ is the characteristic function of a random variable X, which is linear, thus the characteristic function of Xw will be given as $\varphi(p)$ [4]. This then, implies that, for a wrapped distribution, we ought to attain

$$\begin{aligned} \varphi(p) &= \sum_{r=0}^{m-1} \frac{\xi(1-\xi)^r e^{i2\pi\frac{rp}{m}}}{1-(1-\xi)^m} \quad p = 0, \pm1, \pm2, \pm3, ... \\ &= \frac{\xi}{1-(1-\xi)e^{\frac{i2\pi p}{m}}} \\ &= \frac{\xi}{y-iz}, \end{aligned}$$

where $y = 1-(1-\xi)cos(\frac{2\pi p}{m})$ and $z = (1-\xi)sin(\frac{2\pi p}{m})$, where p is not equal to $0(\mathrm{mod} m)$ [4].

By explaining further, $\varphi(p)$, we will attain

$$\begin{aligned} \varphi(p) &= \xi\{(1-(1-\xi)cos(\frac{2\pi p}{m}))^2 + ((1-\xi)sin(\frac{2\pi p}{m}))^2\}^{\frac{-1}{2}} e^{iarctan\{\frac{(1-\xi)sin(\frac{2\pi p}{m})}{1-(1-\xi)cos(\frac{2\pi p}{m})}\}} \\ &= \phi_p e^{i\gamma_p}, \end{aligned}$$

wher

$$\phi_p = \xi\{(1-(1-\xi)cos(\frac{2\pi p}{m}))^2 + ((1-\xi)sin(\frac{2\pi p}{m}))^2\}^{\frac{-1}{2}}$$

and

$$\gamma_p = arctan\{\frac{(1-\xi)sin(\frac{2\pi p}{m})}{1-(1-\xi)cos(\frac{2\pi p}{m})}\}.$$

With the above equation we get the p^{th} trigonometric moment of the wrapped geometric distribution as follows

$$\varphi_p = \alpha_p + i\beta_p,$$

where

$$\alpha_p = E(cosp\theta) = \phi_p cos\gamma_p$$

and

$$\beta_p = E(sinp\theta) = \phi_p sin\gamma_{p.}$$

Central trigonometric moments are as follows

$$\alpha bar_p = \phi_p cos(\gamma_p - p\gamma_1),$$

$$\beta bar_p = \phi_p sin(\gamma_p - p\gamma_1)$$

The distance of the resultant vector, $\phi = \phi_1$ is

$$\begin{aligned} \phi &= \sqrt{\alpha_1^2 + \beta_1^2} \\ &= \frac{\xi}{(1+(1-\xi)^2 - 2(1-\xi)cos\frac{2\pi}{m})^{\frac{1}{2}}} \\ &= \frac{\xi}{((\xi)^2 + 2(1-\xi)(1-cos\frac{2\pi}{m}))^{\frac{1}{2}}}. \end{aligned}$$

The average direction is

$$\gamma = arctan\frac{(1-\xi)sin(\frac{2\pi}{m})}{1-(1-\xi)cos(\frac{2\pi}{m})}.$$

The circular square of deviations, V_1 is as follows

$$V_1 = 1-\phi$$

$$= 1-\frac{\xi}{(1+(1-\xi)^2-2(1-\xi)cos(\frac{2\pi}{m}))^{\frac{1}{2}}}.$$

The circular standard deviation is given below

$$\begin{aligned}\sigma_0 &= \sqrt{-2lnp} \\ &= \sqrt{ln\frac{(\xi)^2+2(1-\xi)(1-cos\frac{2\pi}{m})}{(\xi)^2}}.\end{aligned}$$

Skewness is measured as follows

$$\begin{aligned}\mu_1^0 &= \frac{\beta bar_2}{V_1^{1.5}} \\ &= \frac{\phi_2 sin(\gamma_2-2\gamma_1)}{(1-\frac{\xi}{((\xi)^2+2(1-\xi)(1-cos\frac{2\pi}{m}))^{\frac{1}{2}}})^{1.5}}.\end{aligned}$$

Kurtosis is measured by

$$\mu_2^0 = \frac{\phi_2 cos(\gamma_2-2\gamma_1)-\phi^4}{V_0^2}.$$

3.1.1 Method of moments

Allow $\Theta = (\Theta_1, \Theta_2, \Theta_3, ...\Theta_n)$ to be a sample which is random and has n elements. Let the sample come from a wrapped geometric distribution with parameters ξ and m [4]. Parameter estimates are attained by setting sample moments equal to the corresponding population moments. We already have ascertained the p^{th} sample trigonometric moment about the zero direction, $m_p^{'} = \alpha_p + ib_p$, where

$$\alpha_p = \frac{1}{n}\sum_{j=1}^{n} cos(p\Theta_j),$$

$$b_p = \frac{1}{n}\sum_{j=1}^{n} sin(p\Theta j).$$

Equating a_p to $\alpha_p$and b_p to β_p, gives

$$a_p = \frac{\xi}{((\xi)^2+2(1-\xi)(1-cos\frac{2\pi}{m}))^{0.5}} * cos\{tan^{-1}\frac{(1-\xi)sin(\frac{2\pi p}{m})}{1-(1-\xi)cos(\frac{2\pi p}{m})}\}$$

and

$$b_p = \frac{\xi}{((\xi)^2+2(1-\xi)(1-cos\frac{2\pi p}{m}))^{0.5}} * sin\{tan^{-1}\frac{(1-\xi)sin(\frac{2\pi p}{m})}{1-(1-\xi)cos(\frac{2\pi p}{m})}\}.$$

where $p \neq 0 \pmod m$ [4]. Solving these two equations for set values of m and p, which we can equate to 1, we obtain

$$\widehat{\xi} = \frac{\hat{R}^2 cos(\frac{2\pi}{m}) \pm \hat{R}\sqrt{\hat{R}^2 cos(\frac{2\pi}{m}) - 2(\hat{R}^2-1)cos(\frac{2\pi}{m})}}{\hat{R}^2-1}$$

where $R = \sqrt{a_1^2+b_1^2}$, is the average of the resultant length of the sample [4].

3.2 Wrapped Laplace distribution on integers

The discrete Laplace distribution was invented by Jupp and Mardia in 1999, who managed to come up with a discrete analogue of the normal distribution [6]. The discrete normal random variable X has a probability mass function which can be written as

$$P(X = s) = \frac{f(s)}{f(l)} \quad s = 0, \pm 1, \pm 2, ...$$

The f in the equation above represents the probability density function of the general normal distribution with parameters μ and σ^2. For any random variable that is continuous, W on the real number line, we can attain a random variable X that belongs to the set of integers. This is done using the equation above. If the skew Laplace density functions given below

$$f(w) = \frac{1}{\sigma}\frac{\kappa}{1+\kappa^2}e^{-\frac{|x|}{\kappa\sigma}}, w < 0$$

or

$$f(w) = \frac{1}{\sigma}\frac{\kappa}{1+\kappa^2}e^{-\frac{\kappa|x|}{\sigma}}, w > 0.$$

for κ greater than zero, are inserted into the equation above, the resulting probability mass function of the attained discrete distribution takes an explicit form in terms of the parameters $p_1 = e^{-\frac{\kappa}{\sigma}}$ and another one $q_1 = e^{-\frac{1}{\kappa\sigma}}$ [6].

A variable X which is random, has a distribution that is called discrete skew Laplace with parameters q_1 an element of the set (0,1) and p_1 an element of the set (0,1), if and only if

$$\begin{aligned} f(s) &= P[X = s] \\ &= \frac{(1-p_1)(1-q_1)}{1-p_1q_1}p_1^{\kappa} \end{aligned}$$

for $s = 0, 1, 2, 3, ...$, or

$$\begin{aligned} f(s) &= P[X = s] \\ &= \frac{(1-p_1)(1-q_1)}{1-p_1q_1}q_1^{|\kappa|} \end{aligned}$$

for $s = 0, -1, -2, -3, ...$ [6].

The random variable X has a characteristic function written as follows

$$\varphi(r) = \frac{(1-p_1)(1-q_1)}{(1-p_1e^{ir})(1-q_1e^{ir})},$$

where r is an element or real numbers [6].

As said earlier on, we are going to concentrate on wrapping the discrete skew Laplace distribution on integers values, thus for $Z = 0, \pm 1, \pm 2, \pm 3....$ This is done on a circle with radius equal to one. From the introduction, it is known that the reduction modulo 2π does the job of wrapping the straight line onto the circle [6]. The reduction modulo $2\pi c$ (given that c is an integer greater than zero) does also the job of wrapping the integers onto the family of c^{th} root of 1, which is seen as a subgroup of the circle. This implies that, if W is a random variable which belongs to the integers set, then Θ, defined by

$$\Theta = 2\pi W(mod 2\pi c),$$

is a variable, which is random, on the lattice $\frac{2\pi t}{c}$, for $t = 0, 1, 2, 3, ..., c-1$, on the circle [6].

Now, if W attains a discrete skew laplace distribution that has parameters p_1 and q_1, then the wrapped variable (random) $\Theta = \frac{2\pi t}{c}$ has the probability distribution function that is given as follows

$$\begin{aligned} P(\Theta = \frac{2\pi t}{c}) &= \sum_{s=-\infty}^{\infty} p(t+sc) \\ &= \sum_{s=-\infty}^{\infty} \frac{(1-p^*)(1-q^*)q^{*|t+sc|}}{1-p^*q^*} + \frac{(1-p^*)(1-q^*)}{1-p^*q^*}p^{*t} + \sum_{s=-\infty}^{\infty} \frac{(1-p^*)(1-q^*)p^{*(t+sc)}}{1-p^*q^*} \end{aligned}$$

for $t = 0, 1, 2, 3, ..., c-1$, where $p^* = p_1 (\text{mod } 2\pi c)$ and $q^* = q_1 (\text{mod } 2\pi c)$

$$\begin{aligned} &= \frac{(1-p^*)(1-q^*)}{1-p^*q^*}[\sum_{s=-\infty}^{-1} q^{*-(t+sc)} + p^* + \sum_{s=1}^{\infty} p^{*(t+sc)}] \\ &= \frac{(1-p^*)(1-q^*)}{1-p^*q^*}[(1-p^{*c}) + p^{*t}(1-p^{*c})(1-q^{*c}) + p^{*(t+c)}(1-q^{*c})] \\ &= \frac{(1-p^*)(1-q^*)}{1-p^*q^*}[\frac{q^{*(c-t)}(1-p^{*c}) + p^{*t}(1-q^{*c})}{(1-p^{*c})(1-q^{*c})}], \end{aligned}$$

for $t = 0, 1, 2, 3, ..., c-1$ and p^*, q^* being elements of the set (0,1) [6].

We also have

$$\sum_{t=0}^{\infty} p_v(\Theta) = 1.$$

This means that $P_v(\cdot)$ is defined as a probability distribution.

A random variable Θ, which is angular, follows a wrapped skew laplace distribution on integers with $p^*, q^*$and c as parameters, and its probability mass function is given as

$$p_v(\Theta) = \frac{(1-p^*)(1-q^*)}{1-p^*q^*}[\frac{q^{*c-t}(1-p^{*c}) + p^{*t}(1-q^{*c})}{(1-p^{*c})(1-q^{*c})}]$$

for $t = 0, 1, 2, 3, ...c-1$ and p^*, q^* being elements of the set (0,1) and it is represented as $WDSL(p^*, q^*, c)$ [6].

3.2.1 Special scenarios

Two special cases do materialize if either p^* or q^* approaches zero. The first one is $\Theta \sim WDSL(p^*, 0, c)$ which is a wrapped geometric distribution with the probability mass function given below

$$P(\Theta = \frac{2\pi t}{c}) = \frac{(1-p^*)p^{*t}}{1-p^{*c}},$$

where p^*an element of the set (0,1) and $t = 0, 1, 2, 3....., c-1$ [6].

The second special case is $\Theta \sim WDSL(0, q^*, c)$ which also, is a wrapped geometric distribution with the probability mass function

$$P(\Theta = \frac{2\pi t}{c}) = \frac{(1-q^*)q^{*-t}}{1-q^{*c}},$$

for $q^* \in$ (0,1) and $t = 0, 1, 2, 3, ..., c-1$ [6].

If it happens that $p^* = q^*$ then we attained a probability mass function of a wrapped discrete Laplace distribution which is given below

$$P(\Theta = \frac{2\pi t}{c}) = \frac{(1-p^*)(p^{*c-t} + p^{*t})}{(1+p^*)(1-p^{*c})},$$

for $t = 0, 1, 2, 3, ...c-1$ [6].

3.2.2 The characteristic function and the trigonometric moments

$F(\Theta)$, the distribution function of the wrapped discrete skew Laplace distribution with p^*, q^* and c as parameters, is given as follows

$$\begin{aligned} P(k) &= \frac{(1-p^*)(1-q^*)}{(1-p^*q^*)(1-p^{*c})(1-q^{*c})} \sum_{t=0}^{c-1}[q^{*c}(1-p^{*c})(\frac{k}{q^*})^t + \{(1-p^{*c})(1-q^{*c}) + p^{*c}(1-q^{*c})\}(kp^*)^t] \\ &= \frac{(1-p^*)(1-q^*)}{(1-p^*q^*)(1-p^{*c})(1-q^{*c})}[\frac{(1-p^{*c})(q^{*c}-k^c)q^*}{q^*-k} + \frac{(1-(kp^*)^c)(1-q^{*c})}{1-kp^*}] \end{aligned}$$

The above equation is also the probability generating function of $WDSL(p^*, q^*, c)$. We also have $P(1) = 1$, when $k = e^{i\frac{2\pi n}{c}}$ we have

$$P(e^{\frac{i2\pi n}{c}}) = \frac{(1-p^*)(1-q^*)}{(1-pe^{\frac{i2\pi n}{c}})(1-qe^{-\frac{i2\pi n}{c}})}.$$

If a linear random variable Y, has a characteristic function $\varphi(r)$, then $\varphi(n)$, is the characteristic function of a wrapped random variable, Y_w, for $n = 0, \pm1, \pm2, \pm3....$ [6]. For the wrapped discrete skew Laplace distribution, we attain

$$\begin{aligned}
\varphi(n) &= E(e^{in\Theta}), \quad n = 0, \pm1, \pm2, \pm3, ... \\
&= E(e^{\frac{in2\pi t}{c}}), \quad t = 0, 1, 2, 3, ..., c-1 \\
&= \sum_{t=0}^{c-1} \frac{(1-p^*)(1-q^*)}{(1-p^*q^*)(1-p^{*c})(1-q^{*c})}[q^{*c-t}(1-p^{*c}) + p^{*t}(1-p^{*c})(1-q^{*c}) + p^{*c+t}(1-q^{*c})]e^{\frac{in2\pi t}{c}}
\end{aligned}$$

which simplifies to

$$\varphi(n) = \frac{(1-p^*)(1-q^*)}{(1-p^*e^{\frac{i2\pi n}{c}})(1-q^*e^{\frac{i2\pi n}{c}})},$$

for $n = 0, \pm1, \pm2, \pm3, ...,n \neq 0 \pmod m$ [6].

We also have that

$$\begin{aligned}
\varphi(n) &= \frac{(1-p^*)(1-q^*)}{1-p^*e^{\frac{i2\pi n}{c}} - q^*e^{\frac{i2\pi n}{c}} + p^*q^*} \\
&= \frac{(1-p^*)(1-q^*)[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c}) + i(p^*-q^*)sin(\frac{2\pi n}{c})]}{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2},
\end{aligned}$$

the above equation boils down to

$$\varphi(n) = \alpha_n + i\beta_n,$$

for

$$\alpha_n = \frac{(1-p^*)(1-q^*)[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]}{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2}$$

and

$$\beta_n = \frac{(1-p^*)(1-q^*)(p^*-q^*)sin(\frac{2\pi n}{c})}{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2}.$$

The n^{th} trigonometric moment of the wrapped discrete skew laplace distribution with parameters p^*, q^* and c, is given by

$$\varphi_\Theta(n) = \frac{(1-p^*)(1-q^*)}{(1-p^*e^{\frac{i2\pi n}{c}})(1-q^*e^{\frac{i2\tau n}{c}})}.$$

The equation above can also be written in this context,

$$\begin{aligned}
\varphi_\Theta(n) &= (1-p^*)(1-q^*)\{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2\}^{-\frac{1}{2}} e^{itan^{-1}(\frac{(p^*-q^*)sin(\frac{2\pi n}{c})}{1+p^*q^*-(p^*+q^*)cos(\frac{2\pi n}{c})}),} \\
&= \tau_n e^{i\upsilon_n}
\end{aligned}$$

where $\tau_n \in (0, 1)$ is called the p^{th} average resultant length and $\upsilon_n \in [0, 2\pi)$ is called the p^{th} average direction for n=1,2,3,...,

$$\tau_n = (1-p^*)(1-q^*)\{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2\}^{-\frac{1}{2}}$$

and

$$\upsilon_n = tan^{-1}(\frac{(p^*-q^*)sin(\frac{2\pi n}{c})}{1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})}).$$

The length of the average resultant vector is given a

$$\begin{aligned}\tau &= \tau_1 \\ &= \sqrt{\alpha_1^2+\beta_1^2} \\ &= \frac{(1-p^*)(1-q^*)}{\sqrt{[1+p^*q^*-(p^*+q^*)cos(\frac{2\pi n}{c})]^2+[(p^*-q^*)sin(\frac{2\pi n}{c})]^2}},\end{aligned}$$

the average direction is given by

$$\begin{aligned}\upsilon_n &= \upsilon_1 \\ &= tan^{-1}(\frac{\beta_1}{\alpha_1}) \\ &= tan^{-1}(\frac{(p^*-q^*)sin(\frac{2\pi}{n})}{1+p^*q^*-(p^*+q^*)cos(\frac{2\pi}{c})}).\end{aligned}$$

We can also attain the circular variance which is

$$\begin{aligned}V_0 &= 1-\tau \\ &= 1-\frac{(1-p^*)(1-q^*)}{\sqrt{[1+p^*q^*-(p^*+q^*)cos(\frac{2\pi n}{c})]^2+[(p^*-q^*)sin(\frac{2\pi n}{c})]^2}},\end{aligned}$$

and the circular standard deviation is attained as follows

$$\begin{aligned}\sigma_0 &= \sqrt{-2ln\tau} \\ &= \sqrt{ln[\frac{[1+p^*q^*-(p^*+q^*)cos(\frac{2\pi n}{c})]^2+[(p^*-q^*)sin(\frac{2\pi n}{c})]^2}{[(1-p^*)(1-q^*)]^2}]}.\end{aligned}$$

Skewness is measured by

$$\eta_1^0 = \frac{\beta_2 bar}{V_0^{\frac{3}{2}}},$$

given that $\beta_2 bar = e = E[sin2(\Theta-\upsilon)]$ [6].

Kurtosis is also measured by

$$\eta_2^0 = \frac{\alpha_2 bar-(1-V_0)^4}{V_0^2},$$

given that $\alpha_2 bar = E[cos2(\Theta-\upsilon)]$ [6].

3.2.3 Method of moments

Allow $\Theta_1, ..., \Theta_l$ to represent a sample of size n, the sample should be random. The random sample is taken from the wrapped discrete skew Laplace distribution with the parameters $p^*, q^*$and c. It follows that the n^{th} sample trigonometric moment about the zero direction is

$$t_n' = a_n + ib_n$$

where

$$a_n = \frac{1}{l}\sum_{i=1}^{l} cos(n\Theta_i)$$

and

$$b_n = \frac{1}{l}\sum_{i=1}^{l} sin(n\Theta_i).$$

The population moments that corresponds to the sample moments above is

$$\varphi(n) = \alpha_n + i\beta_n.$$

Matching the population moments to the sample moments, we attain $\alpha_n = a_n$and $\beta_n = b_n$ for $n = 1, 2, 3, ...$[6]. We then, have that

$$a_1 = \frac{(1-p^*)(1-q^*)[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi}{c})]}{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi}{c})]^2}$$

and

$$b_1 = \frac{(1-p^*)(1-q^*)(p^*-q^*)sin(\frac{2\pi}{c})}{[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi n}{c})]^2 + [(p^*-q^*)sin(\frac{2\pi n}{c})]^2}$$

Exploiting the equations above, and for a constant value of t we can ascertain the estimates for p^* and q [6].

Through the division of a_1 by b_1 we get

$$\frac{a_1}{b_1} = \frac{1+p^*q^* - (p^*+q^*)cos(\frac{2\pi}{c})}{(p^*-q^*)sin(\frac{2\pi}{c})}$$

which gives

$$a_1(p^*-q^*)sin(\frac{2\pi}{c}) = b_1[1+p^*q^* - (p^*+q^*)cos(\frac{2\pi}{c})]$$

which finally gives

$$p = \frac{b_1q^*cos(\frac{2\pi}{c}) - a_1q^*sin(\frac{2\pi}{c}) - b_1}{b_1q^* - b_1cos(\frac{2\pi}{c}) - a_1sin(\frac{2\pi}{c})}$$

By substituting the "p^*" value in terms of "q^*" in the equation for a_1 will make us attain an equation in terms of "q^*" and thus "p" [6].

3.3 Wrapped t-distribution

Allow the random variable $Y \sim t(s)$. This means that Y is a random variable which is linear in nature and it follows a Student's t distribution with s degrees of freedom. Under normal circumstances the degrees of freedom s is greater than zero and more often than not, it is in integer form. The random variable Y has the density function which is as follows:

$$f(y;s) = c(1+\frac{y^2}{s})^{-\frac{s+1}{2}},$$

where y is an element of the real numbers and $c = \frac{\Gamma(\frac{s+1}{2})}{(\Gamma(\frac{s}{2})\sqrt{\pi s})}$ [10].

Now, consider $Z = \mu + \delta Y$ [10]. The random variable Z is scaled and re-centered with new center μ and scaled δ times, where μ is an element of real numbers and δ is any number greater than zero. The random variable Z is wrapped onto the circle of radius equal to one as follows:

$$\theta = Z(mod(2\pi)).$$

Thus, θ is a circular random variable that is called the wrapped random variable which follows the wrapped t-distribution with the density function given as

$$f(\Theta;\mu_0,\delta,s) = \frac{c}{\delta}\sum_{p=-\infty}^{\infty}(1+\frac{(\Theta+2\pi p-\mu_0)^2}{\delta^2 s}),$$

where Θ is between 0 and 2π. The density of θ is symmetric about the average direction μ_0 since the density function above consist of the infinite summation which is symmetrically monotonic decreasing function about its mean μ_0.[10]

3.3.1 Special scenarios

The amazing feature of the wrapped t-distribution is that the degrees of freedom, s, is interpreted in many different ways. Particularly, when s=1, the density function given above will reduce to the density of the heavy-tailed wrapped Cauchy distribution [10]. If s turns to infinity, the limiting distribution is the light-tailed normal distribution. If s is between 0 and 1, we will have a more heavier-tailed Cauchy distribution since the density function given above will be more peaked than normal [10].

The density function of a wrapped t random variable, θ, can only be stated as an infinite summation. This means that there are no certain scenarios where the density function of the wrapped random variable can be manipulated to a finite summation [10]. If s approaches zero and δ approaches infinity, the estimation of the density given above will become increasingly difficult because of the large numbers of the central terms in the infinite summation that be must included. In the events of smaller values of s and δ greater than one, only a few number terms are needed which makes the estimation much easier [10].

3.3.2 The characteristic function and trigonometric moments

Lebedev and Hurst studied the characteristic function of the random variable Y which belongs to the t-distribution and it is linear [10]. Implementing how Hurst presented his work, for $v > 0$,

$$\begin{aligned}\Phi_Y(v) &= E(e^{ivY}) \\ &= \frac{K_{S/2}(v\sqrt{s})(v\sqrt{s})^{(s/2)}}{\Gamma(s/2)2^{(s/2)-1}},\end{aligned}$$

where

$$K_\omega(y) = \frac{1}{2}(\int_0^\infty t^{\omega-1}e^{\{-\frac{y}{2}(t+\frac{1}{t})\}}dt),$$

ω is an element of a real number set, $y > 0$, is an integral of the third type and order ω [10]. Putting to use the idea of Mardia & Jupp, the members of the characteristic function $[\phi_\theta(m) : p = 0, \pm1, \pm2, ...]$ of $\theta$are as follows, when $\mu_0 = 0$, by

$$\phi_\theta(m) = \frac{K_{s/2}(m\delta\sqrt{s})(m\delta\sqrt{s})^{s/2}}{\Gamma(s/2)2^{(s/2)-1}}.$$

The distribution is symmetric about $\mu_0 = 0$, which then leads to the trigonometric moments

$$\beta_m = E(sin(m\theta)) = 0$$

and

$$\alpha_m = E(cos(m\theta)) = \phi_\theta(m).$$

Particularly, the two first cosine moments are given below as

$$\begin{aligned}\alpha_1 &= \rho \\ &= \frac{K_{s/2}(\delta\sqrt{s})(\delta\sqrt{s})^{s/2}}{\Gamma(s/2)2^{(s/2)-1}}\end{aligned}$$

and

$$\alpha_2 = \frac{K_{s/2}(2\delta\sqrt{s})(2\delta\sqrt{s})^{s/2}}{\Gamma(s/2)2^{(s/2)-1}}.$$

Kato and Shimizu presents the trigonometric moments, characteristic function and the density function of a wrapped t variable with integer degrees of freedom, which is correct but is not always the case. This is because there are some instances where the degrees of freedom are not strictly integers.

3.3.3 Method of moments

Allow $\Theta = (\Theta_1, ..., \Theta_n)$ to represent a random sample of magnitude n from the wrapped distribution with parameters μ, δ, s distribution. Bluntly speaking, for the method of moments approximates of the parameters are attained by solving the system of equations which comes from equating $\mu_0$to the sample mean direction, Θ^-, $\alpha_1$to the resultant length, R^-, and α_2 to

$$\alpha_2^- = \frac{1}{n}\sum_{i=1}^{n} cos2(\Theta_i - \Theta^-).$$

If, by any chance, α_2^- is negative, then no solution will exists for the last equation [10]. Observations from simulations, for n fixed, the percentage of samples with negative α_2^- values, m fluctuates with fluctuating dispersion (δ) . In actual sense, when $\delta = 10$, m is greater than 0.37 for n greater than or equal to 5. For small samples sizes like n less than 10 less dispersed populations, δ less than 1, m increases as s turns to zero. In another context, for large samples, negative values of α_2^- are not easily attained from samples taken from populations with δ less than 1. With this potential problem of attaining negative α_2^-- values, the method of moments estimation cannot, in general be recommended [10].

3.3.4 Maximum likelihood estimation

Estimation using maximum likelihood shrinks to the numerical optimization of the log-likelihood function

$$l(\mu_0, \delta, s, \Theta) = -n(log(\delta) + \frac{1}{2}log(s) + log(B(\frac{v}{2}, \frac{1}{2})) + \sum_{i=1}^{n} log(\sum_{m=-1}^{\infty}(1 + \frac{(\Theta_i + 2\pi m - \mu_0)^2}{\delta^2 s})^{-\frac{s+1}{2}}).$$

The gradient based method of maximizing the equation above is rather cumbersome but possible [10]. In order to estimate the parameters, we find the partial derivative with respect to the parameter needed to be estimated. Equate that partial derivative to zero and then solve for the parameter to be estimated.

4 Conclusion

In this research report, analysis and an overview of directional statistics under the subtopic, wrapped distributions, was done. Transformations of usual distributions to wrapped ones was illustrated for three distributions namely geometric, Laplace and t distribution. All steps of changing these distributions to wrapped ones was shown in detail. There is no general way of wrapping the usual distributions, so each distribution has its unique way of being wrapped. Parameter estimation for the wrapped distributions was also done in this research report. It was noted that generally, the method of moments estimation estimates the parameters easily. Unfortunately there are some other instances where method of moments estimation is not recommended. It is seen that for a wrapped t distribution, method of moments estimation does not yield estimates. In this circumstance maximum likelihood estimation will then be employed.

Even though it is a cumbersome task, future endeavours in this arena of wrapping distributions should derive the maximum likelihood functions for almost all wrapped distributions so that comparison of estimates could be done in order to get the best estimates. This is because as it stands, we only base with one source of parameter estimation and it is not known if it is the best one or not.

References

[1] Reinaldo B Arellano-Valle, Héctor W Gómez, and Fernando A Quintana. A new class of skew-normal distributions. *Communications in Statistics-Theory and Methods*, 33(7):1465–1480, 2004.

[2] Martin Ehler and Jennifer Galanis. Frame theory in directional statistics. *Statistics & Probability Letters*, 81(8):1046–1051, 2011.

[3] Fisher N I. *Statistical Analysis of Circular Data*. Cambridge University, 1993.

[4] Sophy Jacob and K Jayakumar. Wrapped geometric distribution: A new probability model for circular data. *Journal of Statistical Theory and Applications*, 12(4):348–355, 2013.

[5] Sreenivasa Rao Jammalamadaka and Tomasz J Kozubowski. New families of wrapped distributions for modeling skew circular data. *Communications in Statistics-Theory and Methods*, 33(9):2059–2074, 2004.

[6] K Jayakumar and Sophy Jacob. Wrapped skew Laplace distribution on integers: A new probability model for circular data. *Open Journal of Statistics*, 2(01):106, 2012.

[7] PE Jupp and KV Mardia. A unified view of the theory of directional statistics, 1975-1988. *International Statistical Review/Revue Internationale de Statistique*, pages 261–294, 1989.

[8] KV Mardia. Directional statistics in geosciences. *Communications in Statistics-Theory and Methods*, 10(15):1523–1543, 1981.

[9] KV Mardia. Directional statistics and shape analysis. *Journal of Applied Statistics*, 26(8):949–957, 1999.

[10] Arthur Pewsey, Toby Lewis, and MC Jones. The wrapped T family of circular distributions. *Australian & New Zealand Journal of Statistics*, 49(1):79–91, 2007.

Compound weighted Poisson distribution

Zwelakhe Magagula 10169106

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr R Ehlers

Department of Statistics, University of Pretoria

2 November 2015 (Final draft)

Abstract

In this essay a special case of random sum distributions is considered. We use the weighted Poisson distribution as the count distribution together with the geometric distribution as the compounding distribution. The extent of variability of the compound weighted distribution obtained for different weight functions is discussed using common measures of variability like the Fisher index of dispersion as well as the factorial moment to mean measure.

Declaration

I, *Zwelakhe Lindokuhle Magagula,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Zwelakhe Lindokuhle Magagula

_____
Dr R Ehlers

_____
Date

Contents

1 Introduction

In this section a brief explanation of the underlying theory needed for compound weighted Poisson distribution and the study of its properties is given.

1.1 Random sum distribution

A random sum distribution is a special case of mixture distributions. It is defined by considering a sequence of independently identically distributed random variables X_i, i = 1,2,... with probability mass function p_i=$P[X_i = i]$. Consider the sum S=$\sum_{i=1}^{N} X_i$ where N is a non-negative integer random variable independent of the sequence $\{X_i\}$. The distribution of S is known as the random sum distribution. A practical example of the random sum model is the collective risk model in insurance, where the random sum S is used to denote aggregate claim amount with N being the number of claims and the X_i's denoting the individual claim amounts. Another example is found in entomology where N denotes the number of female insects in a specific region and the random variable X_i denotes the number of eggs laid by the i^{th} female, so that S is the total number of eggs laid in this region. For the random sum we can find a general expression for its expected value and variance by making use of some identities of conditional expectation given in [4]. To find $E[S]$, apply the following identity

$$E[S] = E_N[E[S \mid N]].$$

From this it follows that

$$E[S \mid N = n] = E[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} E[\, X_i] = nE[X_i].$$

Therefore,

$$E[S] = E[NE[X_i] = E[N]E[X_i]. \tag{1}$$

Similarly for $var[S]$, we use the identity:

$$var[S] = E_N[var[S \mid N]] + var[E[S \mid N]].$$

Since X_i, i = 1,2,...,N are independent consider

$$var[S \mid N = n] = var[\sum_{i=1}^{n} X_i] = n \sum_{i=1}^{n} var[X_i]$$

and so

$$var[S] = E[N(var[X_i])] + var[NE[X_i]].$$

i.e.

$$var[S] = E[N][var[X_i]] + var[N][E[X_i]]^2. \tag{2}$$

1.2 Weighted distributions

In this subsection we give a brief account on weighted distributions. The idea of weighted distributions is discussed in detail in [9, 4]. Suppose Y is a non-negative random variable with probability density function (pdf) $f(y)$. Suppose y is a realization of Y under $f(y)$and is observed with probability proportional to the weight function $w(y)$. Then the distribution of the observed sample is given by

$$f^w(y) = \frac{w(y)f(y)}{E[w(Y)]} \tag{3}$$

where $E[w(Y)]$ is a normalizing factor. The random variable Y^w with pdf $f^w(y)$ is the weighted version of Y. Consider the simple weight function $w(y) = y$, where $Y \sim POI(\lambda)$. It follows that

$$f(y) \quad = \quad \frac{e^{-\lambda}\lambda^y}{y!},$$

and

$$E(Y) \quad = \quad \lambda.$$

For the given weight function $w(y) = y$ and from (3) it follows that

$$f^w(y) = \frac{w(y)f(y)}{E[w(Y)]} \quad = \quad \frac{ye^{-\lambda}\lambda^y}{y!E(Y)} = \frac{e^{-\lambda}\lambda^{y-1}}{(y-1)!},$$

where $(Y^w - 1) \sim POI(\lambda)$.
The weighted random variable Y^w is stochastically greater than the variable Y if the weight function $w(y)$ is a monotone increasing function, and smaller than the variable Y if the weight function is monotone decreasing. Consequently the expected value of the weighted version is greater on smaller than that of the original if the weight function is monotone increasing or decreasing respectively (see[11, 10]).

1.3 Dispersion and measures of dispersion

Dispersion (also known as variability or spread) is a very important study in distribution theory, be it in theoretical aspects or in the context of an underlying statistical sample. Dispersion enables us to get information about data or a certain distribution. Many problems arise from variability being large. Thus an evident extent of variability helps in being able to handle the problem properly. Moreover, dispersion is a good measure of randomness of a data. In this paper we consider two measures of dispersion, namely the Fisher index and factorial moment to mean measure.

1.3.1 Fisher index of dispersion

A prominent measure of variability is the Fisher index of dispersion given in detail by [5]. The Fisher index of dispersion is defined as follows:

$$FI(X) = \frac{Var(X)}{E(X)}. \tag{4}$$

It is loosely termed, the variance to mean ratio, which is a normalized measure of dispersion. It is clear that the Fisher index of dispersion is only defined when the $E[X]$ is non-zero. A distribution is over-dispersed if $FI[X] > 1$. Conversely we say it is under-dispersed if $FI[X] < 1$ and equi-dispersed if $FI[X] = 1$. The ideas behind over-dispersion and under dispersion is also discussed in length in [12].

1.3.2 Factorial moment to mean measure

Another prominent measure of dispersion outlined in [3] and is used in this paper is the factorial moment to mean measure which measures variability like the Fisher index. The factorial moment to mean of order r is defined as follows:

$$I_r(X) = \frac{E[X(X-1)...(X-r+1)]}{[E(X)]^r} \qquad for\ r = 2, 3, ... \tag{5}$$

In the case of count data, for example where X is a Poisson distributed random variable, $I_2(X)$ is non-negative. The theorem below shows how the factorial moment to mean measure is related to the Fisher index measure of dispersion in the case of count data.

Theorem 1.1
For $r = 2$ in and X a random variable for count data i) $FI(X) > 1$ if and only if $I_2(X) > 1$,
ii)$FI(X) = 1$ if and only if $I_2(X) = 1$,
iii)$FI(X) < 1$ if and only if $I_2(X) < 1$.

Proof
It follows from (4), (5) and the fact that $var(X) = E(X^2) - [E(X)]^2$ that

$$\begin{aligned} FI(X) &= \frac{Var(X)}{E(X)} \\ &= \frac{E[X(X-1)] + E(X) - [E(X)]^2}{E(X)} \\ &= I_2(X)E(X) + 1 - E(X) \\ &= 1 + E(X)[I_2(X) - 1]. \end{aligned}$$

The result follows from this and the fact that $E(X) \geq 0$.

■

1.4 Probability generating function

Definition 1.4.1
If X is a discrete random variable with values in the non negative integers with probability mass function p_i=$P[X_i = i]$. The probability generating function of X is defined as:

$$u(z) = E[z^X] = \sum_{i=0}^{\infty} p_i z^i \ . \tag{6}$$

In this paper we make use of probability generating functions to calculate moments for different distributions (See [1]). Below is a result (that we prove informally) for moments calculated from a probability generating function.

Theorem 1.4.1
For X a random variable with $u^{(r)}(1)$ the rth derivative of its probability generating function at z=1 it follows that
$u^{(r)}(1) = E[X(X-1)...(X-r+1)]$

Proof
From (6)

$$\begin{aligned} u^{(r)}(z) &= \frac{d^r}{dz^r} u(z) \\ &= \frac{d^r}{dz^r} [\sum_{i=0}^{\infty} p_i z^i] \\ &= \sum_{i=0}^{\infty} \frac{d^r}{dz^r} p_i z^i \\ &= \sum_{i=0}^{\infty} p_i i(i-1)...(i-r+1) z^{i-r}. \end{aligned}$$

For $z = 1$ it follows that

$$\begin{aligned} u^{(r)}(1) &= \sum_{i=0}^{\infty} p_i i(i-1)...(i-r+1) \\ &= E[X(X-1)...(X-r+1)]. \end{aligned}$$

■

The consequence of this result is that

$$u'(1) = E[X] \tag{7}$$

and

$$u''(1) = E[X(X-1)] = E[X^2] - E[X].$$

Thus it follows that

$$E[X^2] = u''(1) + u'(1). \tag{8}$$

From (7) and (8) and the fact that $var(X) = E(X^2) - [E(X)]^2$ we can calculate the expected value and variance of a random variable.

1.5 Outline of study

The Poisson distribution is used in practice as a standard model for count data, for example in the random sum $S = \sum_{i=1}^{N_\lambda} X_i$, $N_\lambda \sim POI(\lambda)$ and X_i, $i = 1, 2, ...N_\lambda$ a sequence of independently identically distributed random variable. However it is an equi-dispersed model. For example, for N_λ, $E(N_\lambda) = Var(N_\lambda) = \lambda$ and the Fisher index of dispersion is $FI(N_\lambda) = \frac{Var(N_\lambda)}{E(N_\lambda)} = 1$. In this research two generalizations of the Poisson distribution, namely the compound Poisson distribution and the weighted Poisson distribution are combined to construct the more flexible compound weighted Poisson distribution. The Fisher index of dispersion [5] and factorial to mean measure [3] are derived for this distribution, and compared for cases when using different weight function and the geometric compounding distribution.

2 Compound weighted Poisson distribution

In this section two generalizations of the Poisson distribution namely the compound Poisson and weighted Poisson distribution are discussed and then combined to construct the more flexible compound weighted Poisson distribution. Measures of dispersion as derived in Section 1.3 are calculated for each of these distributions.

2.1 Compound Poisson distribution

One of the generalizations of the Poisson distribution is the compound Poisson distribution. This is a special case of the random sum distributions which was described in Section 1.1 and is given in much detail in [4]. Consider $S = \sum_{i=1}^{N_\lambda} X_i$, and let X_i, i = 1,2,...,N_λ be a sequence of independently identically distributed random variable and N_λ is a non-negative integer random variable independent of the sequence $\{X_i\}$. Suppose that N_λ follows a Poisson distribution with parameter λ. Then S follows a compound Poisson distribution with parameter λ and $F(x)$. $F(x)$ represents a general distribution for X_i. The distribution of the $X_i s$ is referred to as the compounding distribution, and just like every random variable, we can calculate its moments. To calculate the moments of S, conditional expectation results are used. From (1) and the fact that $N_\lambda \sim POI(\lambda)$ it follows that

$$E[S] = E[N_\lambda]E[X_i] = \lambda E[X_i]. \tag{9}$$

Similarly from (2)

$$\begin{aligned} var[S] &= E[N][var[X_i]] + var[N][E[X_i]]^2 \\ &= \lambda E[X_i^2]. \end{aligned} \tag{10}$$

From (4), (9) and (10) the Fisher index for the compound Poisson distribution is given by

$$FI[S] = \frac{Var[S]}{E[X]}$$
$$= \frac{E[X^2]}{E[X]}.$$

Since $I_2[X] \geq 0$ it can be seen from the Fisher index above that the compound Poisson distribution is over-dispersed ($FI(S) > 1$) and will be equi-dispersed if ($FI(S) = 1$) that is if and only if $I_2[X] = 0$ that is if and only if $E[X^2] = E[X]$.
Also from (5), (9) and (10) the factorial moment to mean measure for the compound Poisson distribution is

$$I_2[S] = \frac{E[S^2] - E[S]}{[E[S]]^2}$$
$$= \frac{var[S] + [E[S]]^2 - E[S]}{[E[S]]^2}$$
$$= \frac{\lambda E[X_i^2] + [\lambda E[X_i]]^2 - \lambda E[X_i]}{[\lambda E[X_i]]^2}$$
$$= \frac{E[X_i^2]}{\lambda [E[X_i]]^2} + 1 - \frac{1}{\lambda E[X_i]}.$$

From $I_2[S]$ above and Theorem 1.1, $I_2[S] = 1$ that is the distribution of S is equi-dispersed if and only if $E[X^2] = E[X]$.

2.2 Weighted Poisson distribution

Another commonly used generalization of the Poisson distribution is the weighted Poisson distribution. In Section 1.2 the general case for weighted distributions was discussed. Consider the random variable N_λ following a Poisson distribution with Poisson parameter λ and a non-negative weight function $w(n)$. Then from (3) N_λ^w is the weighted version of the Poisson random variable N_λ with probability mass function (pmf) given by

$$P[N_\lambda^w = n] = \frac{w(n)}{E[w(N_\lambda)]} \frac{\lambda^n e^{-\lambda}}{n!}, \tag{11}$$

where $E[w(N_\lambda)] = \sum_{n=0}^{\infty} w(n) \frac{\lambda^n e^{-\lambda}}{n!} < \infty$ is a normalizing constant. The weighted Poisson distribution and its properties is discussed in [2].
To measure the extent of variability for the weighted Poisson distribution $FI(N_\lambda^w)$, the Fisher index and $I_2(N_\lambda^w)$, the factorial moment to mean measure are calculated by finding moments for N_λ^w using the probability generating function given in (6) together with (7) and (8).

From (6) and (9)

$$
\begin{aligned}
u_{N_\lambda^w}(z) =& \sum_{n=0}^{\infty} z^n \left[\frac{w(n)}{E[w(N_\lambda)]} \frac{\lambda^n e^{-\lambda}}{n!} \right] \\
&= \frac{1}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} w(n) \frac{(\lambda z)^n e^{-\lambda}}{n!} e^{-\lambda z} e^{\lambda z} \\
&= \frac{1}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} w(n) \frac{(\lambda z)^n e^{-\lambda(1-z)}}{n!} e^{-\lambda z} \\
&= \frac{e^{-\lambda(1-z)}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} w(n) \frac{(\lambda z)^n}{n!} e^{-\lambda z} \\
&= \frac{E[w(N_{\lambda z}]}{E[w(N_\lambda)]} e^{-\lambda(1-z)}, \qquad (12)
\end{aligned}
$$

where $N_{\lambda z} \sim$Poisson(λz).

From (7) and (12) it follows that

$$
E[N_\lambda^w] = \lambda \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]}. \qquad (13)
$$

Similarly it follows from (8) and (12) that

$$
\begin{aligned}
E[(N_\lambda^w)^2] &= u''_{N_\lambda^w}(1) + u'_{N_\lambda^w}(1) \\
&= E[N_\lambda^w] + \frac{1}{E[w(N_\lambda)]} \sum_{n=o}^{\infty} w(n) \frac{n(n-1)(\lambda z)^{n-2} e^{-\lambda}}{n!} \lambda^2 \mid_{z=1} \\
&= E[N_\lambda^w] + \frac{1}{E[w(N_\lambda)]} \sum_{n=o}^{\infty} w(n) \frac{\lambda e^{-\lambda}}{(n-2)!} \lambda^2 \\
&= \lambda \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]} + \lambda^2 \frac{E[w(N_\lambda + 2)]}{E[w(N_\lambda)]}. \qquad (14)
\end{aligned}
$$

Consequently the variance of N_λ^w is given by the identity

$$
Var[N_\lambda^w] = \lambda \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]} + \lambda^2 \frac{E[w(N_\lambda + 2)]}{E[w(N_\lambda)]} - \lambda^2 [\frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]}]^2. \qquad (15)
$$

Thus the Fisher index for the weighted Poisson distribution is given by

$$
\begin{aligned}
FI[N_\lambda^w] &= \frac{Var[N_\lambda^w]}{E[N_\lambda^w]} \\
&= 1 + \lambda \left[\frac{E[w(N_\lambda + 2)]}{E[w(N_\lambda + 1)]} - \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]} \right]. \qquad (16)
\end{aligned}
$$

From [7] we read that the weighted Poisson distribution accounts for both over-dispersion and under-dispersion in this case.
Now the factorial to mean measure of order 2 can be calculated as:

$$I_2[N_\lambda^w] = \frac{E[(N_\lambda^w)^2] - E[N_\lambda^w]}{[E[N_\lambda^w]]^2}$$
$$= \frac{E[w(N_\lambda + 2)]}{[E(w(N_\lambda + 1))]^2} E[w(N_\lambda)].$$

2.3 Compound weighted Poisson distribution

Two of the most prominent generalizations of the Poisson distribution namely the compound Poisson distribution and the weighted Poisson distribution were discussed in the previous sections. In this section we combine the two generalizations to obtain an optimal and flexible family of distributions, the compound weighted Poisson distribution. Let $S^w = \sum_{i=1}^{N_\lambda^w} X_i$, with a given weight function $w(n), n = 0, 1, ...$ be a random sum, where N_λ^w is a non-negative random variable which has a weighted Poisson distribution, independent of the sequence of independently identically distributed random variables X_i. Then the distribution of the random sum S^w is called a compound weighted Poisson distribution. In this section we continue to make use of probability generating functions to calculate the moments of the random variable S^w, which are use to measure the extent of variability of the distribution of S^w.

Consider again the sequence of independently identically distributed random variables X_i, i = 1,2,... with probability mass function p_i=$P[X_i = i]$ and probability generating function given by:

$$u_1(z) = E[z^X]. \tag{17}$$

The probability generating function of S^w is given by

$$u_{S^w}(z) = E[z^{S^w}]$$
$$= \sum_{n=0}^{\infty} E[z^{X_1+X_2+...+X_n} \mid N_\lambda^w = n] P[N_\lambda^w = n]$$
$$= \sum_{n=0}^{\infty} E[z^{X_1}] E[z^{X_2}] ... E[z^{X_n}] \; P[N_\lambda^w = n]$$
$$= \sum_{n=0}^{\infty} [E(z^{X_1})]^n \; P[N_\lambda^w = n].$$

From the above, (11) and (17) it follows that

$$
\begin{aligned}
u_{S^w}(z) =& \sum_{n=0}^{\infty} [u_1(z)]^n \; P[N_\lambda^w = n] \\
=& \sum_{n=0}^{\infty} [u_1(z)]^n \left[\frac{w(n)}{E[w(N_\lambda)]} \frac{\lambda^n e^{-\lambda}}{n!} \right] \\
=& \sum_{n=0}^{\infty} \left[\frac{w(n)}{E[w(N_\lambda)]} \frac{[\lambda u_1(z)]^n e^{-\lambda}}{n!} \right] \\
=& \sum_{n=0}^{\infty} \left[\frac{w(n)}{E[w(N_\lambda)]} \frac{[\lambda u_1(z)]^n e^{-\lambda}}{n!} \right] \frac{e^{-\lambda u_1(z)}}{e^{-\lambda u_1(z)}} \\
=& \frac{e^{-\lambda(1-u_1(z))}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{[\lambda u_1(z)]^n e^{-\lambda u_1(z)}}{n!} \right] \\
=& \frac{E[w(N_{\lambda u_1(z)})]}{E[w(N_\lambda)]} e^{-\lambda(1-u_1(z))}, \qquad (18)
\end{aligned}
$$

where $N_{\lambda u_1(z)} \sim POI(\lambda u_1(z))$.
In order to measure variability we calculate from (18) the expected value and variance of this weighted random variable S^w. By differentiating the pgf given by (18) above we get:

$$
\begin{aligned}
u'_{S^w}(z) &= \frac{d}{dz} \left\{ \sum_{n=0}^{\infty} \left[\frac{w(n)}{E[w(N_\lambda)]} \frac{[\lambda u_1(z)]^n e^{-\lambda}}{n!} \right] \right\} \\
&= \frac{\lambda e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{[\lambda u_1(z)]^{n-1} n}{n!} u'_1(z) \right] \\
&= \frac{\lambda e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{[\lambda u_1(z)]^{n-1}}{(n-1)!} u'_1(z) \right]. \qquad (19)
\end{aligned}
$$

The expected value of S^w follows from the above by substituting of $z = 1$ as derived in (7) . From (7) and (17) we also know $u'_1(1) = E[X]$ and $u_1(1) = 1$ thus it follows that

$$
E[S^w] = \frac{\lambda E[X]}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{\lambda^{n-1} e^{-\lambda}}{(n-1)!} \right] = \lambda E(X) \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]}. \qquad (20)
$$

Similarly the second derivative of the pgf in (18) is given by

$$
u''_{S^w}(z) = \frac{\lambda^2 e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{[\lambda u_1(z)]^{n-2}}{(n-2)!} [u'_1(z)]^2 \right] + \frac{\lambda e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{[\lambda u_1(z)]^{n-1}}{(n-1)!} u''_1(z) \right].
$$

Substituting $z = 1$ into the above expression and making use of (6), (7) and (8) we have that

$$
\begin{aligned}
u''_{S^w}(1) &= \frac{\lambda^2 e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{\lambda^{n-2}}{(n-2)!} [E(X)]^2 \right] + \frac{\lambda e^{-\lambda}}{E[w(N_\lambda)]} \sum_{n=0}^{\infty} \left[w(n) \frac{\lambda^{n-1}}{(n-1)!} \left[E(X^2) - E(X) \right] \right] \\
&= \lambda^2 [E(X)]^2 \frac{E[w(N_\lambda + 2)]}{E[w(N_\lambda)]} + \lambda [E(X^2) - E(X)] \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]}. \qquad (21)
\end{aligned}
$$

From (8), (19) and (21) it follows that

$$
\begin{aligned}
E[(S^w)^2] &= u''_{S^w}(1) + u'_{S^w}(1) \\
&= \lambda^2 [E(X)]^2 \frac{E[w(N_\lambda + 2)]}{E[w(N_\lambda)]} + \lambda E(X^2) \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]}.
\end{aligned}
$$

For the variance, we obtain

$$Var(S^w) = \lambda E(X^2)\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]} + \lambda^2[E(X)]^2\left\{\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda)]} - \left[\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right]^2\right\}.$$

From (20) we know that $\lambda\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]} = \frac{E(S^w)}{E(X)}$, then variance of S^w simplifies to

$$Var(S^w) = \frac{E(X^2)}{E(X)}E(S^w) + \lambda^2[E(X)]^2\left\{\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda)]} - \left[\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right]^2\right\}. \tag{22}$$

The Fisher index for the distribution of S^w becomes

$$FI(S^w) = \frac{Var(S^w)}{E(S^w)} = \frac{E(X^2)}{E(X)} + \lambda[E(X)]\left\{\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda+1)]} - \frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right\}. \tag{23}$$

Substitution (16) into (23) gives

$$FI(S^w) = \frac{Var(S^w)}{E(S^w)} = \frac{E(X^2)}{E(X)} + [E(X)]\left[\frac{Var[N_\lambda^w]}{E[N_\lambda^w]} - 1\right]. \tag{24}$$

From (24) above we see that

$$FI(S^w) = FI(S) + E(X)[FI(N_\lambda^w) - 1].$$

Where S has a compound Poisson distribution discussed in Section 3.1 and N_λ^w has the weighted Poisson distribution discussed in Section 3.2.

From (5), (23) and (24) it follows that the factorial moment to mean measure for S^w is given by

$$I_2(S^w) = \frac{\lambda[E(X)]^2\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda)]} + \lambda[E(X^2)-E(X)]\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}}{\left[\lambda E(X)\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right]^2}.$$

Next we prove a result that is given in [8], which gives a necessary and sufficient condition for over-dispersion and under-dispersion of S^w in terms of the compounding distribution. This is a useful result in the next section, where we consider a specific compounding distribution and derive its properties.

Theorem 2.3.1
The Let S^w be a compound weighted Poisson variable. Then $FI(S^w) > 1$ or $FI(S^w) < 1$ if and only if

$$\frac{Var[N_\lambda^w]}{E[N_\lambda^w]} \lessgtr 1 - I_2(X). \tag{25}$$

Proof
Suppose the variable S^w is over-dispersed, that is $\frac{Var(S^w)}{E(S^w)}>1$. From (24) we have that

$$FI[S^w] = \frac{E(X^2)}{E(X)} + [E(X)]\left[\frac{Var[N_\lambda^w]}{E[N_\lambda^w]} - 1\right] \quad > \quad 1.$$

Simplifying the above inequality we get

$$FI[N_\lambda^w] = \frac{Var[N_\lambda^w]}{E[N_\lambda^w]} \quad > 1 - \frac{E(X^2)-E(X)}{[E(X)]^2} \quad = 1 - I_2(X).$$

The result for under-dispersion follows in a similar way.

Conversely if we suppose that the inequalities given in (25) hold, substitution of $I_2(X)$ and algebraic simplification lead back (24) giving inequalities that prove for over-dispersion or under dispersion of S^w.

■

2.4 Summary of results

The table below gives a summary of the results describing different measures of dispersion calculated in Section 2.1 to 2.3.

Variable	Fisher index	Factorial moment to mean measure
S	$\frac{E[X^2]}{E[X]}$	$\frac{E[X_i^2]}{\lambda[E[X_i]]^2} + 1 - \frac{1}{\lambda E[X_i]}$
N_λ^w	$1 + \lambda \left[\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda+1)]} - \frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]} \right]$	$\frac{E[w(N_\lambda+2)]}{[E(w(N_\lambda+1))]^2} E[w(N_\lambda)]$
S^w	$\frac{E(X^2)}{E(X)} + \lambda[E(X)] \left\{ \frac{E[w(N_\lambda+2)]}{E[w(N_\lambda+1)]} - \frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]} \right\}$	$\frac{\lambda[E(X)]^2 \frac{E[w(N_\lambda+2)]}{E[w(N_\lambda)]} + \lambda[E(X^2)-E(X)] \frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}}{\left[\lambda E(X) \frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right]^2}$

Table 1: Results: Fisher index and factorial to mean measure for S, $N_\lambda^w$and S^w

3 Application to special cases

In Section 2.3 the weighted Poisson distribution was used as the count distribution in the random sum. In this section we expand this theory by focusing on the geometric distribution as the compounding distribution. Different cases of weight functions for the weighted Poisson distribution are also studied and the measures of dispersion discussed in Section 1.3 are calculated.

3.1 Geometric compounding distribution

Consider the compound weighted Poisson distribution $S^w = \sum_{i=1}^{N_\lambda^w} X_i$, where the sequence of random variables X_i, i = 1,2,...,N_λ^w are geometrically distributed with parameter $(1-\mu)$ and N_λ^w has the weighted Poisson distribution. The purpose here is to investigate the variability of the distribution of S^w for different weight functions $w(n)$ of N_λ^w. Three weight functions will be considered, namely $w(n) = C$, $w(n) = n$ and $w(n) = \frac{1}{1+n}$. For each case the Fisher index in (24) will be calculated and results will be compared graphically.
The probability mass function is given by

$$p_i = P[X_i = i] = \mu^{i-1}(1-\mu), \qquad i = 1, 2, ... \tag{26}$$

The probability generating function for the geometric distribution is given by

$$\begin{aligned} u_1(z) &= \sum_{\forall i} z^i \mu^{i-1}(1-\mu) \\ &= z(1-\mu)\sum_{\forall i}(z\mu)^{i-1} \\ &= \frac{z(1-\mu)}{1-\mu z}. \end{aligned} \tag{27}$$

From (7), (8) and (27)

$$\begin{aligned} E[X] = u_1'(1) &= \left[\frac{1-\mu}{1-z\mu} + \frac{(1-\mu)\mu z}{(1-z\mu)^2}\right] |_{z=1} \\ &= \frac{1}{1-\mu}. \end{aligned}$$

and

$$\begin{aligned} E[X^2] &= u''(1) + u'(1) \\ &= \left[\frac{\mu(1-\mu)}{(1-z\mu)^2} + \frac{\mu(1-\mu)}{(1-z\mu)^2} + \frac{2\mu^2 z(1-\mu)}{(1-z\mu)^3}\right] |_{z=1} + \frac{1}{1-\mu} \\ &= \frac{1+\mu}{(1-\mu)^2}. \end{aligned} \tag{28}$$

From the above and (5) we have that the factorial moment to mean measure for the geometric distribution $I_2(X) = \frac{E[X(X-1)]}{[E(X)^2]} = 2\mu$. Applying the result in Theorem 2.3.1 (see (19)), it follows that $FI(S^w) \lesseqgtr 1$ if and only if $FI[N_\lambda^w] \lesseqgtr 1 - 2\mu$.
Below we derive properties of the distributions that arise when using the weight functions $w(n) = C$, $w(n) = n$ and $w(n) = \frac{1}{1+n}$, in the compound weighted Poisson distribution with geometric compounding distribution.

3.1.1 Case 1 : constant weight function $w(n) = C$.

It follows from (11) that

$$P[N_\lambda^w = n] = \frac{w(n)}{E[w(N_\lambda)]}\frac{\lambda^n e^{-\lambda}}{n!} = \frac{\lambda^n e^{-\lambda}}{n!}.$$

Thus the random variable N_λ^w has a Poisson distribution with parameter λ. The distribution of S^w is a compound weighted Poisson with geometric compounding distribution and is known as the Pólya–Aeppli distribution with parameters λ and μµdenoted as $PA(\lambda, \mu)$ (see [6]).
The probability generating function for S^was given by (18). It follows that for the constant weight function it simplifies to,

$$u_{S^w}(z) = e^{-\lambda(1-u_1(z))},$$

where $u_1(z)$ given by (27) is the probability generating function of the geometric distribution.
From (20) and (22)

$$E[S^w] = \lambda E(X) = \frac{\lambda}{1-\mu}.$$

and

$$Var(S^w) = \frac{E(X^2)}{E(X)}E(S^w) = \frac{\lambda(1+\mu)}{(1-\mu)^2}.$$

Thus for the Pólya–Aeppli distribution the Fisher index is given by

$$FI(S^w) = \frac{Var(S^w)}{E(S^w)} = \frac{1+\mu}{1-\mu} = 1 + \frac{2\mu}{1-\mu} > 1, \tag{29}$$

which implies that the Pólya–Aeppli distribution is over-dispersed.

3.1.2 Case 2 : weight function $w(n) = n$.

It follows from (1) that

$$P[N_\lambda^w = n] = \frac{w(n)}{E[w(N_\lambda)]}\frac{\lambda^n e^{-\lambda}}{n!} = \frac{\lambda^{n-1} e^{-\lambda}}{(n-1)!}.$$

The random variable N_λ^w has a size-biased distribution (see [10]) and $N_\lambda^w = 1 + N_\lambda$ where $N_\lambda \sim POI(\lambda)$. Thus the mean and variance for the weighted Poisson distribution are given by

$$E(N_\lambda^w) = \lambda + 1 \qquad Var(N_\lambda^w) = \lambda$$

The probability generating function for S^w is given by (18) simplifies to

$$u_{S^w}(z) = e^{-\lambda(1-u_1(z))},$$

where $u_1(z)$ is given by (27).
From (20)

$$E[S^w] = \lambda E(X)\frac{\lambda+1}{\lambda} = \frac{1+\lambda}{1-\mu}.$$

The variance follows from (22)

$$Var(S^w) = \frac{E(X^2)}{E(X)}E(S^w) + \lambda^2[E(X)]^2\left\{\frac{E[w(N_\lambda+2)]}{E[w(N_\lambda)]} - \left[\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]}\right]^2\right\} = \frac{\mu + \lambda(1+\mu)}{(1-\mu)^2}.$$

Thus for the size biased distribution the Fisher index is given by

$$FI(S^w) = \frac{Var(S^w)}{E(S^w)} = \frac{\mu + (1+\mu)\lambda}{(1-\mu)(1+\lambda)} = \frac{1+\mu}{1-\mu} - \frac{1}{(1-\mu)(1+\lambda)} < \frac{1+\mu}{1-\mu}. \tag{30}$$

From (29) and (30) we see that the compound size based Poisson distribution with geometric compounding distribution is under-dispersed with respect to the Pólya–Aeppli distribution.

3.1.3 Case 3: weight function $w(n) = \frac{1}{n+1}$.

For the weight function $w(n) = \frac{1}{n+1}$, $n = 0, 1, ...$ it follows that

$$\begin{aligned} E(w(N_\lambda)) &= E\left(\frac{1}{N_\lambda + 1}\right) \\ &= \sum_{n=0}^{\infty} \frac{1}{n+1} \frac{\lambda^n e^{-\lambda}}{n!} \\ &= \frac{1}{\lambda} \sum_{n=0}^{\infty} \frac{\lambda^{n+1} e^{-\lambda}}{(n+1)!} \\ &= \frac{1}{\lambda}[1 - e^{-\lambda}]. \end{aligned} \tag{31}$$

From (11)

$$P[N_\lambda^w = n] = \frac{w(n)}{E[w(N_\lambda)]} \frac{\lambda^n e^{-\lambda}}{n!} = \frac{e^{-\lambda}}{1 - e^{-\lambda}} \frac{\lambda^{n+1}}{(n+1)!}.$$

From (20)

$$E(N_\lambda^w) = \lambda \frac{E[w(N_\lambda + 1)]}{E[w(N_\lambda)]},$$

where the expression for $E[w(N_\lambda)]$ is given in (31).
To find an explicit expression for this expected value we first find the $E[w(N_\lambda + 1)] = E\left[\frac{1}{N+1}\right]$.

$$\begin{aligned} E[w(N_\lambda + 1)] = E\left[\frac{1}{N+1}\right] &= \sum_{n=0}^{\infty} \frac{1}{n+1} \frac{\lambda^n e^{-\lambda}}{n!} \frac{n+1}{n+1} \\ &= \frac{e^{-\lambda}}{\lambda} \sum_{n=0}^{\infty} \frac{(n+1)\lambda^{n+2}}{(n+2)!} \\ &= \frac{e^{-\lambda}}{\lambda} \left[\sum_{n=0}^{\infty} \frac{\lambda^{n+2}}{(n+2)!} + \sum_{n=0}^{\infty} \frac{n\lambda^{n+2}}{(n+2)!}\right]. \end{aligned} \tag{32}$$

Since

$$\sum_{n=0}^{\infty} \frac{e^{-\lambda}\lambda^n}{n!} = 1 = e^{-\lambda}(1 + \frac{\lambda}{1} + \frac{\lambda^2}{2!} + \frac{\lambda^3}{3!} + ...).$$

We have that

$$\sum_{n=0}^{\infty} \frac{\lambda^{n+2}}{(n+2)!} = e^{\lambda} - 1 - \lambda. \tag{33}$$

Since

$$\begin{aligned} E(N-2) =& \sum_{n=0}^{\infty} \frac{(n-2)e^{-\lambda}\lambda^n}{n!} \\ &= e^{-\lambda}(-2-\frac{\lambda}{1}+0+\frac{\lambda^3}{3!}+\frac{2\lambda^4}{4!}+...) \\ &= e^{-\lambda}(-2-\lambda+\sum_{n=0}^{\infty}\frac{n\lambda^{n+2}}{(n+2)!}). \end{aligned}$$

It follows that

$$\sum_{n=0}^{\infty}\frac{n\lambda^{n+2}}{(n+2)!} = (\lambda-2)e^{\lambda}+2+\lambda. \tag{34}$$

Substituting (33) and (34) into (32) we then have that

$$\begin{aligned} E[w(N_\lambda+1)] &= \frac{e^{-\lambda}}{\lambda}\left[e^{-\lambda}-1-\lambda+(\lambda-2)e^{\lambda}+2+\lambda\right] \\ &= \frac{e^{-\lambda}}{\lambda}\left[1-e^{\lambda}+\lambda e^{\lambda}\right]. \end{aligned} \tag{35}$$

From (14), (31) and (35) it follows that

$$\begin{aligned} E(N_\lambda^w) &= \lambda\frac{E[w(N_\lambda+1)]}{E[w(N_\lambda)]} \\ &= \lambda\frac{e^{-\lambda}}{\lambda}\left[1-e^{\lambda}+\lambda e^{\lambda}\right]\frac{\lambda}{1-e^{-\lambda}} \\ &= \frac{e^{-\lambda}+\lambda-1}{1-e^{-\lambda}}. \end{aligned}$$

In a similar way the variance of N_λ^w can be derived and found to be

$$Var(N_\lambda^w) = \frac{\lambda[1-(1+\lambda)e^{-\lambda}]}{(1-e^{-\lambda})^2}.$$

Moreover from (18) and (31) the probability generating function of S^w is given by

$$u_{S^w}(z) = \frac{1-e^{-\lambda u_1(z)}}{u_1(z)[1-e^{-\lambda}]}e^{-\lambda(1-u_1(z))}. \tag{36}$$

Thus from (7), (8) and (36) it follows that the expected value and variance of S^w are given by (see [8])

$$\begin{aligned} E[S^w] &= \frac{\lambda-1+e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})}, \\ Var[S^w] &= \frac{(1+\mu)(1-e^{-\lambda})(\lambda-1+e^{-\lambda})+(1-e^{-\lambda})^2-\lambda^2e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})^2}. \end{aligned}$$

From (4) we have that the Fisher index for this distribution is

$$FI(S^w) = \frac{Var[S^w]}{E[S^w]} = \frac{1+\lambda}{1-\mu} + \frac{(1-e^{-\lambda})^2 - \lambda^2 e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})(\lambda - 1 + e^{-\lambda})}.$$

.

Since the second term of the expression is positive, the distribution of S^w is over-dispersed with respect to the Pólya–Aeppli distribution.

3.2 Results

3.2.1 Theoretical results

The summary of results given and discussed in Section 3.1 are given in Table 2 below.

$w(n)$	$P[N_\lambda^w = n]$	$E[S^w]$	$Var[S^w]$	$FI[S^w]$
c	$\frac{\lambda^n e^{-\lambda}}{n!}$	$\frac{\lambda}{1-\mu}$	$\frac{\lambda(1+\mu)}{(1-\mu)^2}$	$\frac{1+\mu}{1-\mu}$
n	$\frac{\lambda^{n-1} e^{-\lambda}}{(n-1)!}$	$\frac{1+\lambda}{1-\mu}$	$\frac{\mu+\lambda(1+\mu)}{(1-\mu)^2}$	$\frac{1+\mu}{1-\mu} - c_1$
				$c_1 = \frac{1}{(1-\mu)(1+\lambda)}$
$\frac{1}{n+1}$	$\frac{e^{\lambda}}{1-e^{-\lambda}} \frac{\lambda^{n+1}}{(n+1)!}$	$\frac{\lambda - 1 + e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})}$	$\frac{(1+\mu)(1-e^{-\lambda})(\lambda - 1 + e^{-\lambda})}{(1-\mu)(1-e^{-\lambda})^2} + a_1$	$\frac{1+\lambda}{1-\mu} + c_2$
			$a_1 = \frac{(1-e^{-\lambda})^2 - \lambda^2 e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})^2}$	$c_2 = \frac{(1-e^{-\lambda})^2 - \lambda^2 e^{-\lambda}}{(1-\mu)(1-e^{-\lambda})(\lambda - 1 + e^{-\lambda})}$

Table 2: Summary of results for the compound weighted Poisson distribution with geometric compounding distribution for different weight functions.

Consider the graphs below

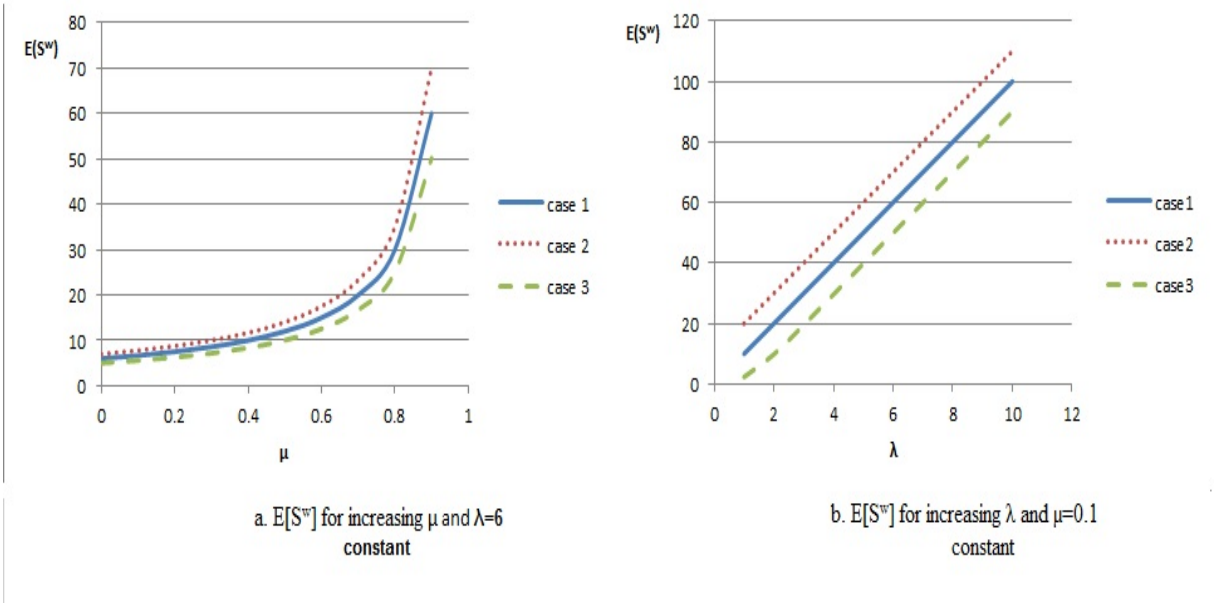


Figure 1: Graphs of $E[S^w]$ for the three cases (a) increasing μ and (b) increasing λ

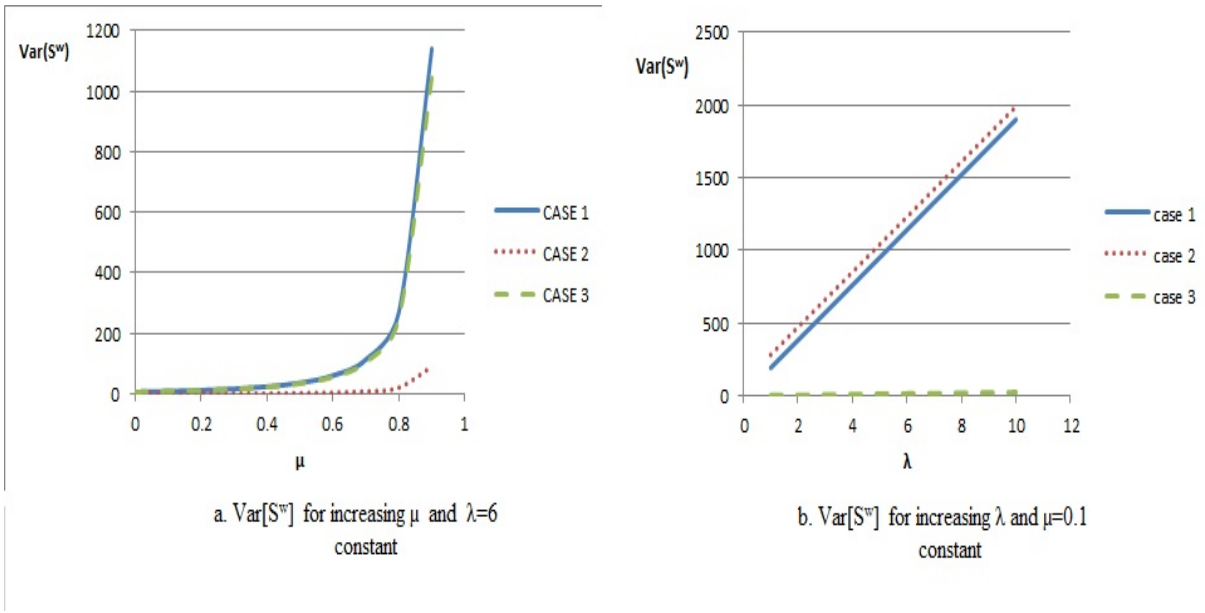


Figure 2: Graphs of $Var[S^w]$ for the three cases (a) increasing μ and (b) increasing λ

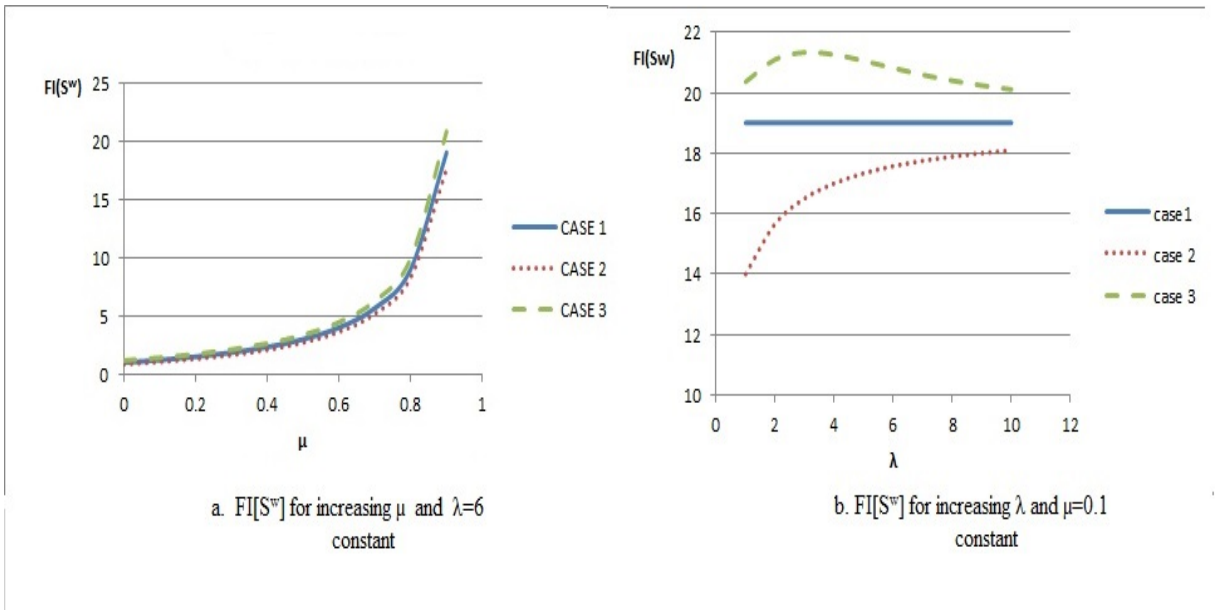


Figure 3: Graphs of $FI[S^w]$ for the three cases (a) increasing μ and (b) increasing λ

Figures 1 to 3 give a graphical representation of the results summarized in Table 2. For each case $E[S^w]$,$Var[S^w]$ and $FI[S^w]$are respectively plotted against μ (keeping $\lambda = 6$ constant) and against λ (keeping $\mu = 0.1$,constant). Both $E[S^w]$ and $Var[S^w]$ increase when the parameters μ and λ increase. In Section 3.1 it was noted that for case 1 the Pólya–Aeppli distribution is over-dispersed with respect to the Poisson distribution. Evaluating the other cases with respect to the Pólya–Aeppli distribution shows that case 2 ($w(n) = n$) is under-dispersed, that is $FI_{case2}[S^w] < FI_{case1}[S^w]$ and case 3 ($w(n) = \frac{1}{n+1}$) is over-dispersed, that is

$FI_{case3}[S^w] > FI_{case1}[S^w]$. That is $FI_{case2}[S^w] < FI_{case1}[S^w] < FI_{case3}[S^w]$. In Figure 3 we observe that the graph of case 2 plots below that of case one while that case 1, while that of case 3 plots above. This agrees with our previous observation.

3.2.2 Empirical results

In this section empirical results simulated by using SAS are presented. For each of the three cases 100000 values of $S^w = \sum_{i=1}^{N_\lambda^w} X_i$, were simulated where X_i is a geometric random variable with parameter (1-μ)=0.1 and N_λ^w has a weighted Poisson distribution with parameter $\lambda = 6$. $E[S^w]$, $Var[S^w]$ and $FI[S^w]$ for these simulated values are summarized for each case in Table 3. The expected value, variance are given by the proc means procedure in SAS (see program in appendix) and Fisher index of S^w is calculated using (4).

Empirical results				Theoretical results			
Case	$E[S^w]$	$Var[S^w]$	$FI[S^w]$	Case	$E[S^w]$	$Var[S^w]$	$FI[S^w]$
Case 1	59.56	1159.99	19.64	Case 1	60	1140	19
Case 2	70.07	1228.19	17.53	Case 2	70	1247.06	17.82
Case 3	50.33	1063.44	21.13	Case 3	49.90	1073.7	21.52

Table 3: Summary of simulation results compared to theoretical results

As expected, from Table 3 we observe that the empirical values are close to the theoretical values. From the empirical results we also note that for case 1 the Pólya–Aeppli distribution the Fisher index is 19.64. This suggests that the distribution of S^w for this case is over-dispersed. Now with respect to the Pólya–Aeppli distribution, it is noted that case 2 which records a Fisher index of 17.53 is under-dispersed, and case 3 with Fisher index 21.13 is over-dispersed. Recall, discussed in Section 3.1 is the theory of the Fisher index of each of the three cases. Given below is a graphical representation of the simulated data, in the form of a histogram for each case.

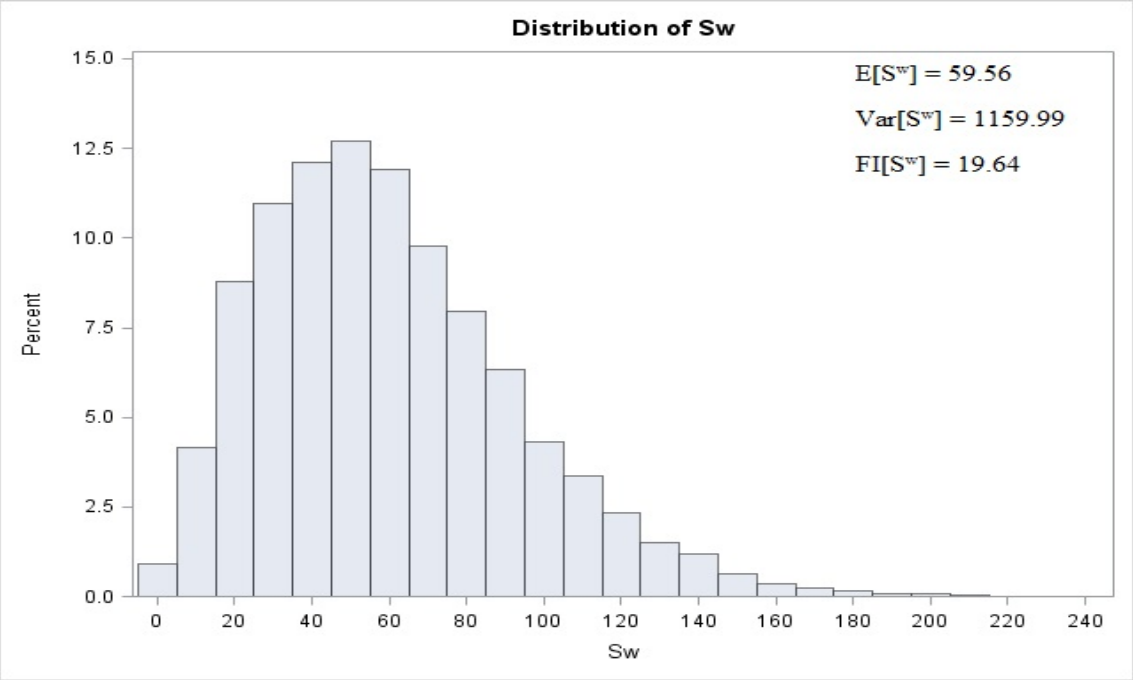


Figure 4: Case 1 simulation bar chart

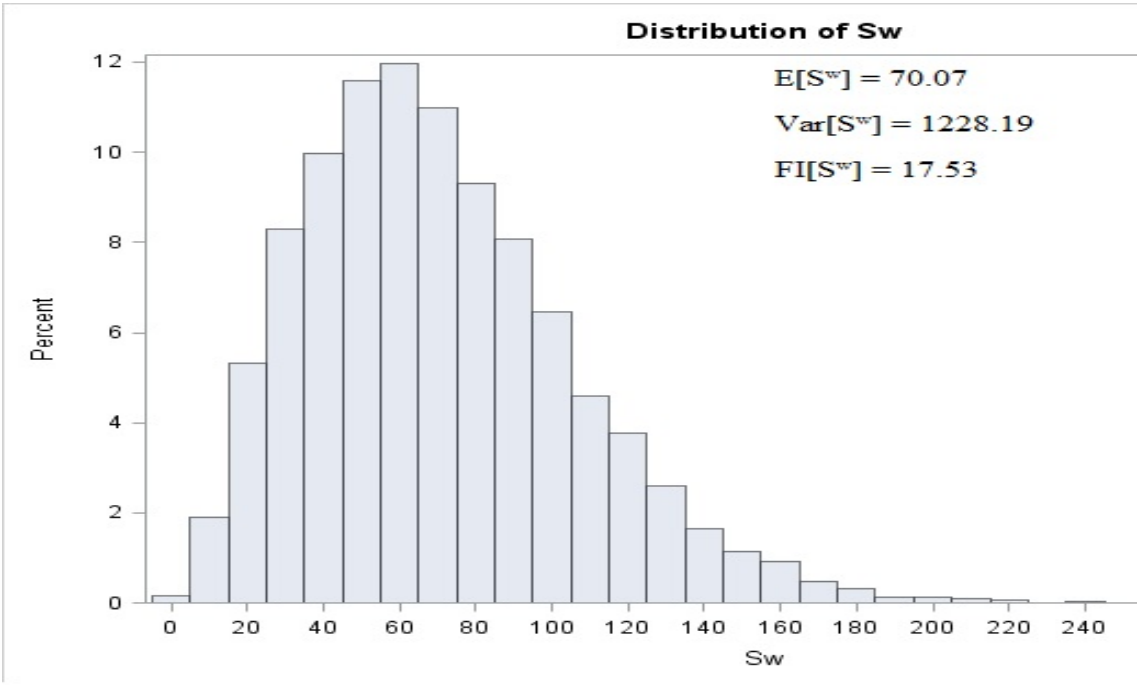


Figure 5: Case 2 simulation bar chart

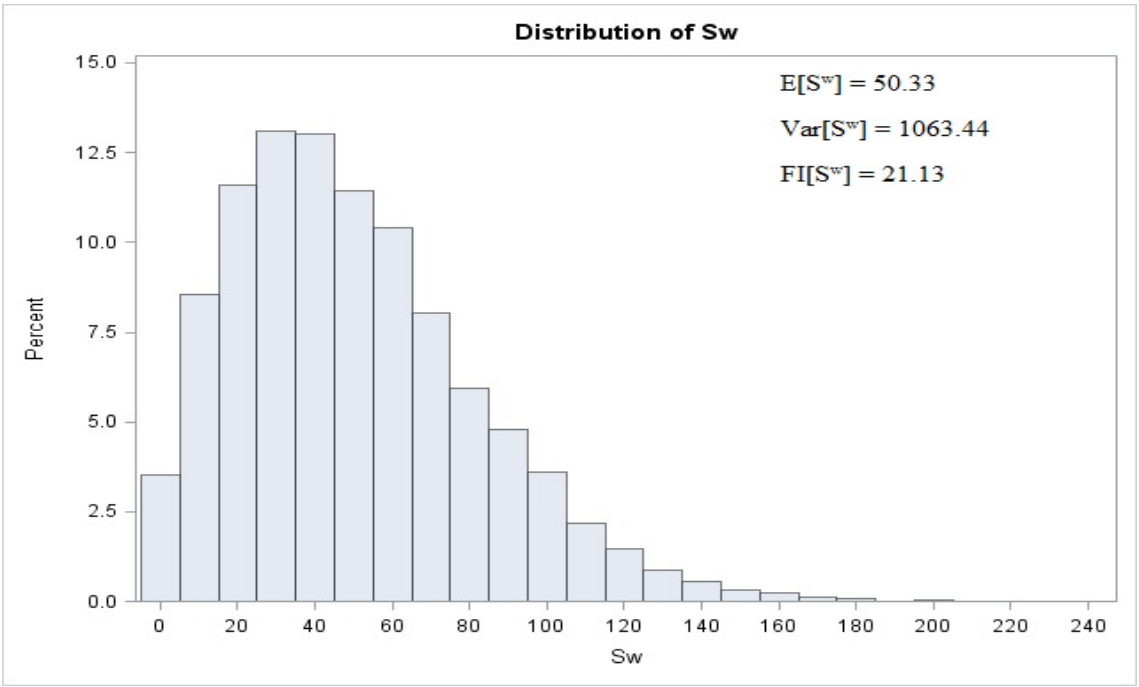


Figure 6: Case 3 simulation bar chart

4 Conclusion

The objective of this study was to construct a more flexible distributions in terms of dispersion that can be used for count data in the random sum $S = \sum_{i=1}^{N_\lambda} X_i$, $N_\lambda \sim POI(\lambda)$ and X_i, $i = 1, 2, ...N_\lambda$ a sequence of independently identically distributed random variable. The restriction of the Poisson which is frequently used for count data is that it is equi-dispersed and when considering $X_i \sim GEO(1-\mu)$, the random sum has the Pólya–Aeppli distribution, which is over-dispersed, that is $FI(S) = \frac{Var(S)}{E(S)} > 1$. The compound weighted Poisson distribution that was constructed in this essay by combining the compound Poisson and the weighted Poisson distribution, has the flexibility that depending on the choice of weight function the resulting distribution could be over-dispersed or under-dispersed in relation to the Pólya–Aeppli distribution.

References

[1] Joseph Abate and Ward Whitt. Numerical inversion of probability generating functions. *Operations Research Letters*, 12(4):245–251, 1992.

[2] N Balakrishnan and Tomasz J Kozubowski. A class of weighted Poisson processes. *Statistics & Probability Letters*, 78(15):2346–2352, 2008.

[3] Dennis D Boos and Cavell Brownie. Comparing variances and other measures of dispersion. *Statistical Science*, pages 571–578, 2004.

[4] C Campbell, B Read, N Balakrishnan, and B Vidakovic. *Encyclopedia of Statistical Sciences*. Wiley, 2006.

[5] RA Fisher. The effect of methods of ascertainment upon the estimation of frequencies. *Annals of eugenics*, 6(1):13–25, 1934.

[6] Norman L Johnson, Adrienne W Kemp, and Samuel Kotz. *Univariate discrete distributions*, volume 444. John Wiley & Sons, 2005.

[7] Célestin C Kokonendji, Dominique Mizere, and Narayanaswamy Balakrishnan. Connections of the Poisson weight function to overdispersion and underdispersion. *Journal of Statistical Planning and Inference*, 138(5):1287–1296, 2008.

[8] Leda D Minkova and N Balakrishnan. Compound weighted Poisson distributions. *Metrika*, 76(4):543–558, 2013.

[9] Ganapati P Patil. *Weighted distributions*. Wiley Online Library, 2002.

[10] Ganapati P Patil and Calyampudi R Rao. Weighted distributions and size-biased sampling with applications to wildlife populations and human families. *Biometrics*, pages 179–189, 1978.

[11] GP Patil, CR Rao, and MV Ratnaparkhi. On discrete weighted distributions and their use in model choice for observed data. *Communications in Statistics-Theory and methods*, 15(3):907–918, 1986.

[12] Evdokia Xekalaki. Under-and overdispersion. *Encyclopedia of actuarial science*, 2006.

Appendix

```
******************** case 1 simulation******************;
data simulation1;
do j=1 to 100000;
N_lamda=ranpoi(0,6);
Sw=0;
do i=1 to n_lamda;
x= RAND('geometric',0.1);
  Sw= Sw+x;
 end;
output;
end;
proc means n  mean var data= simulation1 maxdec=4;
var Sw;
run;
Proc univariate plot  noprint data=simulation1 ;
var sw;
histogram / midpoints = 0 to 300 by 10;
run;

******************** case 2 simulation******************;
data simulation2;
do j=1 to 100000;
N_lamda=ranpoi(0,6)+1;
Sw=0;
do i=1 to n_lamda;
  x= RAND('geometric',0.1);
Sw= Sw+x;
end;
output;
end;
proc means n  mean var data= simulation2 maxdec=4;
var Sw;
run;
Proc univariate plot  noprint data=simulation2 ;
var sw;
histogram / midpoints = 0 to 300 by 10;
run;

******************** case 3 simulation******************;
data simulation3;
do j=1 to 100000;
u=ranuni(0); N_lamda=-1; psum=0; ind=0; lamda=6;
do until (ind=1);
N_lamda=N_lamda+1;
p=exp(-lamda)/(1-exp(-lamda))*(lamda**(N_lamda+1))/fact(N_lamda+1); psum1=psum;
psum=psum+p;
if psum1<u<=psum then ind=1;
end;
Sw=0;
do i=1 to n_lamda;
 x= RAND('geometric',0.1);
```

```
   Sw= Sw+x;
end;
output;
end;
proc means n  mean var data= simulation3 maxdec=4;
var Sw;
run;
Proc univariate plot  noprint data=simulation3 ;
var sw;
histogram / midpoints = 0 to 300 by 10;
run;
```

Kernel density estimation: kernel comparisons and bandwidth selection

Mbavhalelo Innocent Masetla 12333710

STK795 Research Report

Submitted in partial fulfillment of the degree BCom (Hons) Statistics

Supervisor(s): Mr MT Loots

Department of Statistics, University of Pretoria

02 November 2015

Abstract

The aim of this research report is to compare different kernels with a specific focus on the Gaussian kernel. Bandwidth selection will be explored as it determines the smoothness of the kernel and affects whether data collapses or clusters. In particular, the Gaussian kernel will be explored using the method of maximum likelihood for the bandwidth estimator and will be extended to other applications, such as the Cauchy kernel.

Declaration

I, *Mbavhalelo Innocent Masetla,* declare that this report, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Mbavhalelo Innocent Masetla

_____
Mr MT Loots

_____
Date

Acknowledgments

This work is based on the research supported in part by the National Research Foundation" (NRF) of South Africa for the grant, Unique Grant No. 94108. The financial assistance of the NRF towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF; which does not accept any liability in this regard.

Contents

List of Figures

List of Tables

1 Introduction

Kernel density estimation (KDE) is a non-parametric data smoothing problem where a point of some distribution is estimated as a weighted average of its similarities with other points from that distribution. To approximate the kernel, the use of mathematical descriptions of the models tested in the form of probability density functions (pdfs) are used to aid in the generation of new observations based on previous observations [2]. These observations are used to statistically analyze the finite population so as to make statistical inferences. The inferences are then used to construct the distribution that determines the type of kernel, such as an Epanechnikov kernel or Quartic kernel, using the mathematical properties of the pdf since it acquires the smoothness properties of the selected kernel [4].

The smoothness of the density function is determined by the bandwidth of the kernel, h^2, illustrating that the method of selecting an appropriate bandwidth is crucial for each point as it may affect the degree of smoothness of the pdf, causing the data to cluster or collapse. There are several methods of selecting a bandwidth, including the "Quick and Dirty" methods of selection that are based on AMISE (Asymptotic Mean Integrated Squared Errors), the cross validation methods that minimize the integrated squared errors by using the method of moments and other methods of selection that are available but do not form part of this research report [3]. The resulting estimate of the kernel is influenced by the selected bandwidth as it is a scaling parameter used to aid with variations in the given sample and should aim to mirror the behavior of the pdf used.

The analysis of KDE is to test whether the estimation can be improved using other parametric kernels to test whether better estimation densities can be found through extensions of the various processes of KDE. The maximum likelihood bandwidth estimator that will be used in the estimation process.

2 Maximum likelihood bandwidth estimator

The use of non-parametric estimation in kernel density estimation is to avoid the assumption of a prior distribution on data and allows the data to be freed from any distributional assumptions. Since the distribution of the data used is unknown, estimating the probability density function, $p(\mathbf{x})$, from a set of observed data samples $\mathbf{X}$ becomes essential to avoid the use of an unspecified pdf which uses unknown parameters. The dataset used to estimate the density function uses samples that are independently and identically distributed from the unknown pdf, $p(\mathbf{x})$, with

$$\hat{p}(x) = \frac{1}{N} \sum_{i=1}^{N} K_H(x, x_i) \tag{1}$$

where H = bandwidth matrix ; $K_H(.)$ = kernel function ; x = data point where density is estimated and x_i= kernel centered at point i .

For the pdf to be recognized as the kernel function and inherit the smoothness properties of the kernel function, the kernel function must satisfy the following conditions;

- $\int_{-\infty}^{\infty} K_H(x, x_i) = 1$
- $K_H(.) \geq 0$

with the H matrix being fixed across all the kernels. Due to a fixed H matrix, the kernel estimators with variable scale will not accurately achieve data modeling. This may cause data dense areas to be over smoothed by the estimator and cause under smoothing in areas where there is little data. To address the varied smoothness, variable kernel density estimation was introduced and uses H_i instead of a single H [4]. This version of density estimation uses an adaptive bandwidth for an estimator to avoid over smoothing and under smoothing of data dense areas and data sparse areas respectively. It is an effective technique when using a multidimensional sample space, which does not form part of this report.

Since KDE is based on unknown distributions, the goodness-of-fit measures used for these distribution is based on the maximum likelihood approach where the product of the likelihood of each data point belonging to the estimated distribution is maximized. The likelihood function is defined as:

$$L_H(\mathbf{X}) = \prod_{i=1}^{N} p_H(x_i) \tag{2}$$

where $p_H(x_i)$ = unknown pdf.

The likelihood function, using a log monotonic transformation, can be optimized by using the maximum likelihood criterion and is then the objective function used for the maximum likelihood approach. The monotonically transformed likelihood function is given by

$$ln(L_H(\mathbf{X}) = \sum_{i=1}^{n} ln[p_H(x_i)]. \tag{3}$$

The use of the log-likelihood makes the maximization process simpler and it is similar to maximizing the standard likelihood function since it is a monotonic transformation where the given order is preserved. However, the maximum likelihood criterion's shortcoming is that there is a possibility of the likelihood tending to an infinity solution if the bandwidth tends to zero. This problem can be addressed by the use of the "leave-one-out" likelihood estimation to avoid reaching a solution where the likelihood tends to infinity. The "leave-one-out" likelihood estimated pdf is given by

$$p_{H(-i)}(x_i) = \frac{1}{N-1} \sum_{j \neq i}^{n} K_{H_j}(x_i - x_j | H_j).$$

The "leave-one-out" objective function is given by

$$ln(L_{H(-i)}(\mathbf{X})) = \sum_{i=1}^{N} ln[p_{H(-i)}(x_i)], \tag{4}$$

where optimizing the objective function above with respect to the bandwidth matrix removes the infinity solution that the maximum likelihood criterion would have giving when the bandwidth tends to zero.

The leave-one-out formulation can be used with kernel bandwidth estimation where the estimators are derived from the leave-one-out maximum likelihood by either maximizing the log-likelihood objective function shown in equation (4), which results in the use of the maximum leave-one-out likelihood, or minimizing some sample entropy with respect to the kernel bandwidths, which results in a minimum leave-one-out entropy estimator that uses different kernel matrices since the estimate is dependent on the derivative of the kernel function. Since this estimator uses partial derivatives for each bandwidth, it does not form part of this report.

The maximum leave-one-out likelihood uses the same maximum likelihood framework as the minimum leave-one-out estimator but differs from the minimum leave-one-out estimator in that it does not use partial derivatives when the maximum likelihood objective function is optimized with respect to the bandwidths, which implies that it uses one bandwidth for kernels within the same neighborhood.

3 Application

3.1 Gaussian kernel

The Gaussian kernel can be expressed as the standard univariate normal density function, given by the pdf

$$p(\mathbf{x}) = \frac{1}{\sqrt{2\pi}nh} \sum_{i=1}^{n} \exp -\frac{1}{2} \left(\frac{x_i - x}{h} \right)^2 \tag{5}$$

which can be used in formulating the likelihood function given by

$$\begin{aligned} L(\mathbf{X}|h) &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}nh} \sum_{i=1}^{n} \exp -\frac{1}{2} \left(\frac{x_j - x_i}{h} \right)^2 \\ &= \frac{1}{(2\pi)^{n/2}(nh)^n} \prod_{i=1}^{n} \sum_{i=1}^{n} \exp -\frac{1}{2} \left(\frac{x_j - x_i}{h} \right)^2 \end{aligned} \tag{6}$$

and represents the joint density function of the random variable. Using a monotonic transformation, the log-likelihood of the function can be shown as

$$\begin{aligned} l(\mathbf{X}|h) &= ln\left(\frac{1}{(2\pi)^{n/2}(nh)^n}\prod_{j=1}^{n}\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2\right) \\ &= ln\left(\frac{1}{(2\pi)^{n/2}(nh)^n}\right)+ln\left(\prod_{j=1}^{n}\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2\right) \\ &= -ln\left((2\pi)^{n/2}(nh)^n\right)+\sum_{j=1}^{n}ln\left(\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2\right) \\ &= -n\left(\frac{ln(2\pi)}{2}+ln(n)+ln(h)\right)+\sum_{j=1}^{n}ln\left(\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2\right) \end{aligned} \tag{7}$$

where $l(\mathbf{X}|h) = ln[L(\mathbf{X}|h)]$. To determine the maximum likelihood estimator for h, equation (7) must be partially differentiated with respect to h such that

$$\begin{aligned} \tfrac{\delta}{\delta h}[l(\mathbf{X}|h)] &= \tfrac{\delta}{\delta h}[-n\left(\tfrac{ln(2\pi)}{2}+ln(n)+ln(h)\right)+\textstyle\sum_{j=1}^{n}ln\left(\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2\right)] \\ &= -\tfrac{n}{h}+\textstyle\sum_{j=1}^{n}\frac{\sum_{i=1}^{n}\frac{x_j-x_i}{h^2}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}{\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}. \end{aligned} \tag{8}$$

From equation (8), h can be expressed as follows;

$$h=\frac{1}{n}\sum_{j=1}^{n}\frac{\sum_{i=1}^{n}(x_j-x_i)\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}{\sum_{i=1}^{n}\exp-\frac{1}{2}\left(\frac{x_j-x_i}{h}\right)^2}. \tag{9}$$

Since h cannot be solved explicitly, a simple value search optimisation approach will be followed that maximises the log-likelihood by way of the leave-one-out method. The value search results show that the value of h converges:

Search Optimisation of the Modified Log-Likelihood Function	
Bandwidth	Value of Modified Log-Likelihood Function
0.300000000000000	−1442.2472022728500
0.300000000000000	−1442.2472022728500
0.298000000000000	−1442.2468239776500
0.298400000000000	−1442.2467950127400
0.298400000000000	−1442.2467950127400
0.298416000000000	−1442.2467949434200
0.298422400000000	−1442.2467949391600
0.298421120000000	−1442.2467949389500
0.298421376000000	−1442.2467949389400
0.298421273600000	−1442.2467949389300
0.298421299200000	−1442.2467949389300
0.298421298432000	−1442.2467949389300
0.298421298432000	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300
0.298421298401280	−1442.2467949389300

Table 1: Gaussian bandwidth value search results

Using the h from the value search in SAS®software [1], an estimated KDE function from the Gaussian distribution is created in the Gaussian SAS®program and plots the following figure:

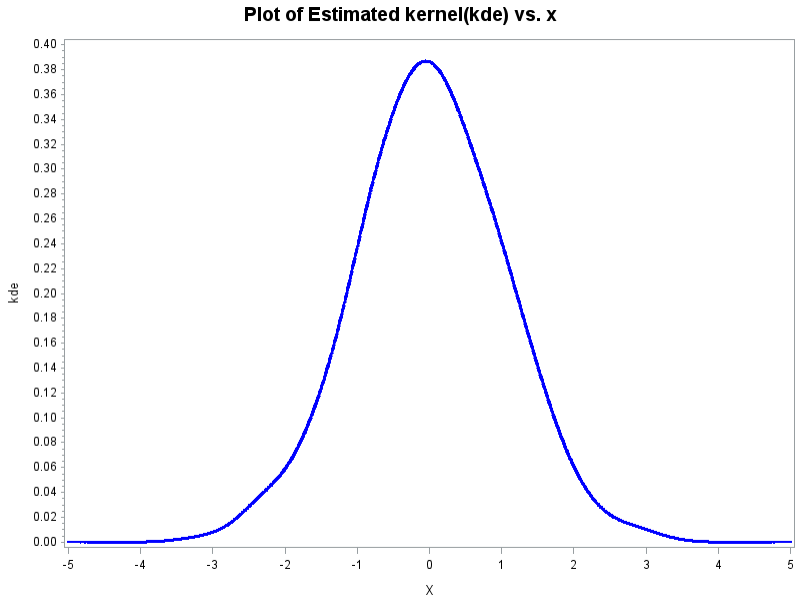


Figure 1: Plot of X against the estimated Gaussian kernel (KDE)

The plot shows that the estimated KDE function has a similar shape to the standard Gaussian kernel

[1] The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.

with $\mu = 0$ and $\sigma^2 = 1$. To show the similarity, a plot of an overlay of the estimated Gaussian KDE and the standard normal KDE is given:

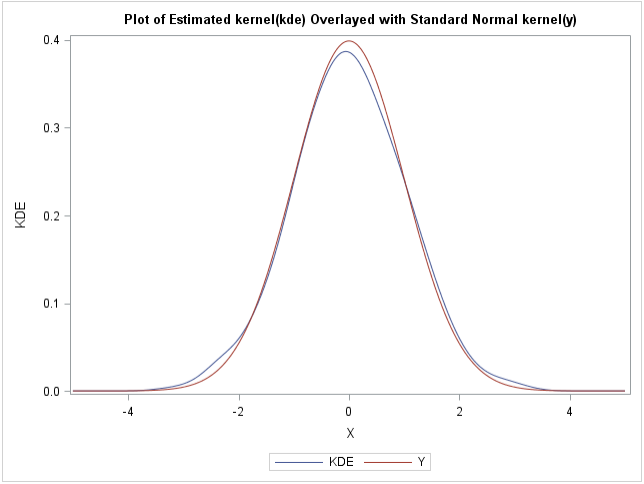


Figure 2: Overlay of the estimated Gaussian kernel and the standard normal kernel

The plot in Figure 2 shows that the estimation method of the bandwidth (h) helps in the generation of a kernel density function that is close to being a standard normal KDE. The difference between the estimated KDE function and the standard normal KDE function can be accounted for by the bandwidth since the estimated bandwidth of the estimated KDE function is less than that of a standard normal KDE function.

3.2 Extension to the Cauchy kernel

The extension of the maximum likelihood estimator is used on the Cauchy kernel which is a standard Cauchy distribution used for a random variable that is the ratio of two independent standard normal variables and lacks defined moments [1], with the probability distribution given as

$$f\left(\mathbf{x}|h\right) = \frac{1}{\pi nh}\sum_{i=1}^{n}\left(\frac{1}{1+\left(\frac{x-x_i}{h}\right)^2}\right) \tag{10}$$

where h=scale parameter and x=location parameter. The likelihood function can thus be expressed as a product of the probability density functions, expressed as

$$\begin{aligned} L(\mathbf{X}|h) &= \prod_{j=1}^{n}\frac{1}{\pi nh}\sum_{i=1}^{n}\left(\frac{1}{1+\left(\frac{x_j-x_i}{h}\right)^2}\right) \\ &= \frac{1}{(\pi nh)^n}\prod_{j=1}^{n}\sum_{i=1}^{n}\left(\frac{1}{1+\left(\frac{x_j-x_i}{h}\right)^2}\right). \end{aligned} \tag{11}$$

Transforming the likelihood function into a log-likelihood function and using $l(\mathbf{X}|h) = ln[L(\mathbf{X}|h)]$, equation (11) becomes

$$l(\mathbf{X}|h) = -nln\left(\pi nh\right) + \sum_{j=1}^{n} ln \sum_{i=1}^{n}\left(\frac{1}{1+\left(\frac{x_j-x_i}{h}\right)^2}\right). \tag{12}$$

Similar to the Gaussian application, the bandwidth (h) is estimated using a value search approach that maximises the log-likelihood by way of the leave-one-out method. The following table shows the convergence of h:

Search Optimisation of the Log-Likelihood Function	
Bandwidth	Value of Log-Likelihood Function
0.320000000000000	−2670.2560118489800
0.324000000000000	−2670.2481684030600
0.324800000000000	−2670.2479867137800
0.324720000000000	−2670.2479842846700
0.324720000000000	−2670.2479842846700
0.324716800000000	−2670.2479842825800
0.324717440000000	−2670.2479842824200
0.324717568000000	−2670.2479842824200
0.324717568000000	−2670.2479842824200
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588480000	−2670.2479842824100
0.324717588479980	−2670.2479842824100
0.324717588479980	−2670.2479842824100

Table 2: Cauchy bandwidth value search results

Using the h value from the value search, the estimated KDE function for the Cauchy distribution is created in the SAS® software and plots the following:

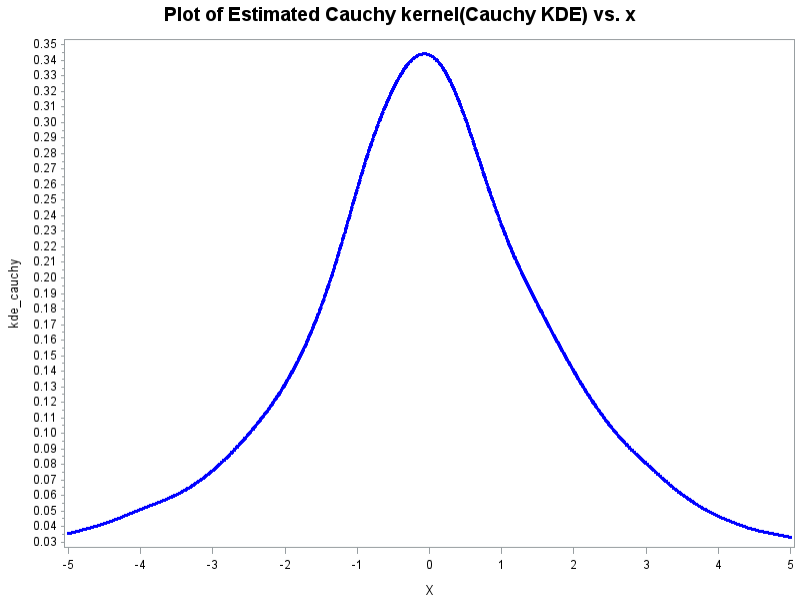


Figure 3: Plot of X against the estimated Cauchy kernel (KDE)

An overlay of the estimated Cauchy KDE function and the theoretical Cauchy KDE function is given by the following plot:

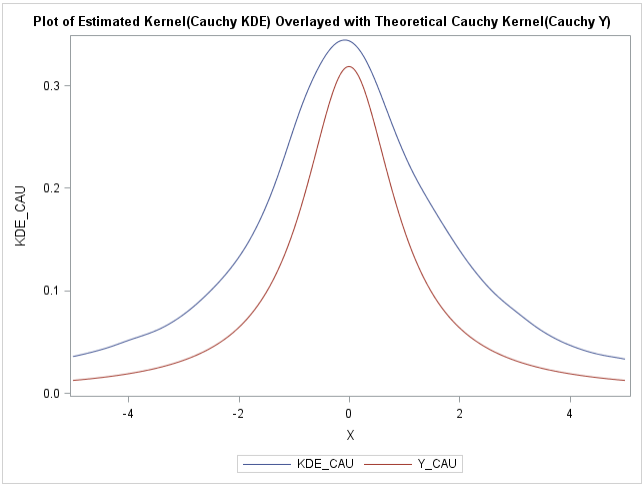


Figure 4: Overlay of the estimated Cauchy KDE and the theoretical Cauchy KDE

which shows that the theoretical Cauchy KDE is heavily tailed compared to the estimated Cauchy KDE function.

3.3 Comparing the Gaussian KDE function and the Cauchy KDE function

3.3.1 Gaussian KDE using Cauchy distributed data

Using a similar setup as in sections 3.1 and 3.2, the Gaussian kernel function is estimated using Cauchy distributed data with the purpose of comparing the estimated Gaussian kernel function and the standard normal kernel function to the theoretical Cauchy kernel function. The method used to generate the necessary plots involved generating Cauchy distributed data and using the data to find the optimal bandwidth h through a value search approach so that the h can be used in the respective estimated kernel functions.Using the SAS® software, the following plot was generated:

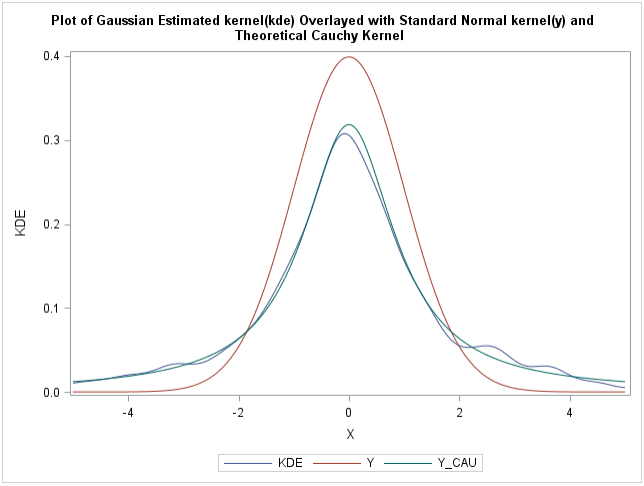


Figure 5: Overlay of the estimated Gaussian kernel, standard normal kernel and the theoretical Cauchy kernel

Figure 5 shows that the standard normal kernel would not be ideal for Cauchy distributed data as it is heavily tailed compared to the theoretical Cauchy kernel, indicating that the Cauchy data is undersmoothed

as compared to the Cauchy kernel. The estimated Gaussian kernel can be used to estimate Cauchy distributed data as it is similar to the theoretical Cauchy kernel, even though it is not smooth on the tail.

3.3.2 Cauchy KDE using standard normal data

The Cauchy kernel function is estimated using standard normal data with the purpose of comparing the estimated Cauchy kernel function and the theoretical Cauchy kernel function to the standard normal kernel function. Using a similar method to the method described in section 3.3.1, the following plot was generated in the SAS® software:

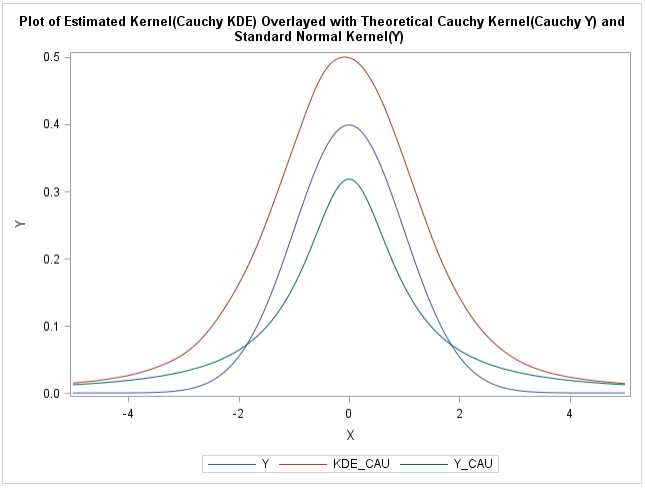


Figure 6: Overlay of the estimated Cauchy kernel, theoretical Cauchy kernel and the standard normal kernel

Figure 6 shows that the theoretical Cauchy kernel has flatter tails than the estimated Cauchy kernel and the standard normal kernel, illustrating that the theoretical Cauchy places more weight on the data on the tails of the kernel than the other kernels. It accomodates more data than the standard normal, which places more weight to the center of the data distribution. Therefore, the Cauchy kernel seems to be a better choice of kernel for a larger sample size.

4 Conclusion

The kernel choice in the application section yields different results depending on the distribution of the data used, given a specified bandwidth. As noted, a Cauchy kernel illustrates more variability and accommodates more data since the Cauchy kernel is not as heavily tail as the standard normal kernel. The Gaussian kernels have heavier tails, indicating that the weight functions places more emphasis on the data centred around the mean level than on data towards the tails. Data is clustered around the mean when using the Gaussian kernel. As shown in section 3.3.1, the estimated Gaussian kernel has a similar shape to that of a Cauchy kernel, which show a higher variability and accomodate more data than the standard normal kernel for Cauchy distributed data. For the normally distributed data, the theoretical Cauchy kernel can be used as its variability allows for the consideration of more data. In order to determine whether other kernels are better than the Gaussian kernel in terms of estimation, more extensive tests need to be conducted using different kernels and different data from other distributions.

References

[1] Catherine Forbes, Merran Evans, Nicholas Hastings, and Brian Peacock. *Statistical Distributions*. John Wiley & Sons, 4 edition, 2011.

[2] Bernard W Silverman. *Density Estimation for Statistics and Data Analysis*, volume 26. CRC press, 1986.

[3] Berwin A Turlach. *Bandwidth Selection in Kernel Density Estimation: A Review*. Université catholique de Louvain, 1993.

[4] Christiaan Maarten van der Walt. Maximum-likelihood kernel density estimation in high-dimensional feature spaces. *North-West University, PhD*, 2014.

5 Appendix

Gaussian SAS®program:

```
proc iml;
n=1000;
mu = rannor(j(n,1,1));
h=0.299758238207970; /*from Theo's code*/

*Silverman = std(x)*(4/(3*nrow(x)))**(1/5);

start kde_norm(x) global(n,h,mu);
norm_pdf=0;
do i =1 to n;
norm_pdf=norm_pdf+
        (1/(sqrt(2*constant('pi'))*n*h)*exp(-(1/2)*((x-mu[i])/h)##2));
end;
return (norm_pdf);
finish kde_norm;

start std_norm(x);
y=(1/(sqrt(2*constant('pi'))))*exp(-(1/2)*(x)##2));
return (y);
finish std_norm;


**Generate x's from -n to n (grid)**;
do j=-5 to 5 by 0.01;
if j=-5 then x=j;
else x=x//j;
if j=-5 then kde=kde_norm(j);
else kde=kde//kde_norm(j);
if j=-5 then y=std_norm(j);
else y=y//std_norm(j);
end;
*xrow=nrow(x);
*fxro=nrow(kde);
*yrow=nrow(y);
x_kde_y=x||kde||y;

*print x_kde_y;
create gaus from x_kde_y[colname={x kde y}];
append from x_kde_y;
quit;

goptions reset=all i=stepsj ;
axis1 label=(angle=90 'kde');
axis2 label=(angle=90 'y');
axis3 label=('X') ;
symbol1 color=blue width=2 ;
title1 'Plot of Estimated kernel(kde) vs. x';
proc gplot data=gaus;
plot kde*x / vaxis=axis1 haxis=axis3;
run;
```

```
goptions reset=title symbol;
symbol1 color=red width=2 ;
title2 'Plot of Standard Normal kernel(y) vs. x';
proc gplot data=gaus;
plot y*x /vaxis=axis2 haxis=axis3;
run;

goptions reset=all;
axis1 label=(angle=90 'kde y');
title3
        'Plot of Estimated kernel(kde) Overlayed with Standard Normal kernel(y)';
proc sgplot data=gaus;
series x=x y=kde;
series x=x y=y;
run;
```

Cauchy SAS®program:

```
proc iml;
n=1000;
mu = rancau(j(n,1,1));
h=0.324717588479980; /*from Theo's code*/

*Silverman = std(x)*(4/(3*nrow(x)))**(1/5);

start kde_cauchy(x) global(n,h,mu);
cauchy_pdf=0;
do i =1 to n;
cauchy_pdf=cauchy_pdf+
        ((1/(constant('pi')*n*h)*(1/(1+(((x-mu[i])##2)/h)))));
end;
return (cauchy_pdf);
finish kde_cauchy;

start std_cauchy(x);
y=(1/(constant('pi'))*(1/(1+((x)##2))));
return (y);
finish std_cauchy;


**Generate x's from -n to n (grid)**;
do j=-5 to 5 by 0.01;
if j=-5 then x=j;
else x=x//j;
if j=-5 then kde_cau=kde_cauchy(j);
else kde_cau=kde_cau//kde_cauchy(j);
if j=-5 then y_cau=std_cauchy(j);
else y_cau=y_cau//std_cauchy(j);
end;
*xrow=nrow(x);
*fxro=nrow(kde);
*yrow=nrow(y);
x_kdecau_ycau=x||kde_cau||y_cau;
```

```
*print x_kdecau_ycau;
create cauchy from x_kdecau_ycau[colname={x kde_cau y_cau}];
append from x_kdecau_ycau;
quit;

goptions reset=all i=stepsj ;
axis1 label=(angle=90 'kde_cauchy');
axis2 label=(angle=90 'y_cauchy');
axis3 label=('X') ;
symbol1 color=blue width=2 ;
title1 'Plot of Estimated Cauchy kernel(Cauchy KDE) vs. x';
proc gplot data=cauchy;
plot kde_cau*x / vaxis=axis1 haxis=axis3;
run;

goptions reset=title symbol;
symbol1 color=red width=2 ;
title2 'Plot of Theoretical Cauchy kernel(Cauchy Y) vs. x';
proc gplot data=cauchy;
plot y_cau*x /vaxis=axis2 haxis=axis3;
run;

goptions reset=all;
axis1 label=(angle=90 'kde y');
title3
        'Plot of Estimated Kernel(Cauchy KDE) Overlayed with Theoretical Cauchy Kernel(Cauc
proc sgplot data=cauchy;
series x=x y=kde_cau;
series x=x y=y_cau;
run;
```

Cauchy on normal data SAS®program:

```
proc iml;
**Kernel Estimation Of Cauchy Using Normally Distribtued Data**;

******Normally distributed Data*******;
n=1000;
reset noautoname;
x = rannor(j(1000,1,1));
Silverman = std(x)*(4/(3*nrow(x)))**(1/5);

*********Calculation of h for Normal********;
start nLogLikelihood(h) global(x);
n = nrow(x);
loglike = -n*log(sqrt(2*constant('PI'))*n*h);
do j = 1 to n;
lltemp = 0;
do i = 1 to n;
if i = j then lltemp = lltemp+0;
else lltemp = lltemp+exp(-(1/2)*((x[j]-x[i])/h)**2);
end;
loglike = loglike+log(lltemp);
```

```
end;
return(loglike);
finish nLogLikelihood;


/*Find the maximum with a simple search algorithm*/
result = j(20,2);
y = j(10, 2);
a = 0;
b = 0.5;

do j = 1 to nrow(result);
do i = 1 to nrow(y);
y[i,1] = a+(b-a)*(i/nrow(y));
y[i,2] = nLogLikelihood(y[i,1]);
end;

if y[<:>,2] = 1 then do;
a = y[y[<:>,2],1];
b = y[y[<:>,2]+1,1];
end;
else if y[<:>,2] = nrow(y) then do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2],1];
end;
else do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2]+1,1];
end;
result[j,] = y[y[<:>,2],];
end;

c = {"Bandwidth" "Value of Log-Likelihood Function"};

print result
        [label="Search Optimisation of the Log-Likelihood Function" colname=c format=19.15

h=result[20,1]; /*the value h converges to*/
print h;

**Estimated Cauchy Kernel  & Theoretical Cauchy Kernel**;
mu = rannor(j(n,1,1));
start kde_cauchy(x) global(n,h,mu);
cauchy_pdf=0;
do i =1 to n;
cauchy_pdf=cauchy_pdf+((1/(constant('pi')*n*h)*(1/(1+(((x-mu[i])##2)/h)))));
end;
return (cauchy_pdf);
finish kde_cauchy;

start std_cauchy(x);
y=(1/(constant('pi'))*(1/(1+((x)##2))));
return (y);
```

```
finish std_cauchy;

**Standard Normal Kernel**;
start std_norm(x);
y=(1/(sqrt(2*constant('pi'))))*exp(-(1/2)*(x)##2));
return (y);
finish std_norm;

**Generate x's from -n to n (grid)**;
do j=-5 to 5 by 0.01;
if j=-5 then x=j;
else x=x//j;
if j=-5 then kde_cau=kde_cauchy(j);
else kde_cau=kde_cau//kde_cauchy(j);
if j=-5 then y_cau=std_cauchy(j);
else y_cau=y_cau//std_cauchy(j);
if j=-5 then y=std_norm(j);
else y=y//std_norm(j);
end;
*xrow=nrow(x);
*fxro=nrow(kde);
*yrow=nrow(y);
x_kdecau_ycau_y=x||kde_cau||y_cau||y;

*print x_kdecau_ycau_y;
create cauchy from x_kdecau_ycau_y[colname={x kde_cau y_cau y}];
append from x_kdecau_ycau_y;
quit;

goptions reset=all i=stepsj ;
axis1 label=(angle=90 'kde_cauchy');
axis2 label=(angle=90 'y_cauchy');
axis3 label=('X') ;
symbol1 color=blue width=2 ;
title1
        'Plot of Estimated Cauchy kernel(Cauchy KDE) on normally distributed data';

/*proc gplot data=cauchy;
plot kde_cau*x / vaxis=axis1 haxis=axis3;
run;

goptions reset=title symbol;
symbol1 color=red width=2 ;
title2 'Plot of Theoretical Cauchy kernel(Cauchy Y) vs. x';
proc gplot data=cauchy;
plot y_cau*x /vaxis=axis2 haxis=axis3;
run;*/

goptions reset=all;
title3
        'Plot of Estimated Kernel(Cauchy KDE) Overlayed with
        Theoretical Cauchy Kernel(Cauchy Y) and Standard Normal Kernel(Y)';
proc sgplot data=cauchy;
```

```
series x=x y=y;
series x=x y=kde_cau;
series x=x y=y_cau;
run;
```

Normal on data CauchySAS® program:

```
proc iml;
**Kernel Estimation Of Normal Using Cauchy Distribtued Data**;

******Cauchy distributed Data*******;
n=1000;
reset noautoname;
x = rancau(j(1000,1,1));

**Calculation of h for Cauchy Distribution**;
start cLogLikelihood(h) global(x);
n = nrow(x);
loglike = -n*log(constant('PI')*n*h);
do j = 1 to n;
lltemp = 0;
do i = 1 to n;
if i = j then lltemp = lltemp+0;
else lltemp = lltemp+1/(1+((x[j]-x[i])/h)**2);
end;
loglike = loglike+log(lltemp);
end;
return(loglike);
finish cLogLikelihood;


/*Find the maximum with a simple search algorithm*/
result = j(20,2);
y = j(10, 2);
a = 0;
b = 0.5;
do j = 1 to nrow(result);
do i = 1 to nrow(y);
y[i,1] = a+(b-a)*(i/nrow(y));
y[i,2] = cLogLikelihood(y[i,1]);
end;

if y[<:>,2] = 1 then do;
a = y[y[<:>,2],1];
b = y[y[<:>,2]+1,1];
end;
else if y[<:>,2] = nrow(y) then do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2],1];
end;
else do;
a = y[y[<:>,2]-1,1];
b = y[y[<:>,2]+1,1];
end;
```

```
result[j,] = y[y[<:>,2],];
end;

c = {"Bandwidth" "Value of Log-Likelihood Function"};

print result
        [label="Search Optimisation of the Log-Likelihood Function" colname=c format=19.15

h=result[20,1]; /*the value h converges to*/
*print h;

**Estimated Gaussian Kernel  & Standard Gaussian Kernel**;
mu = rancau(j(n,1,1));
start kde_norm(x) global(n,h,mu);
norm_pdf=0;
do i =1 to n;
norm_pdf=norm_pdf+
        (1/(sqrt(2*constant('pi'))*n*h)*exp(-(1/2)*((x-mu[i])/h)##2));
end;
return (norm_pdf);
finish kde_norm;

start std_norm(x);
y=(1/(sqrt(2*constant('pi'))))*exp(-(1/2)*(x)##2));
return (y);
finish std_norm;

**Theoretical Cauchy Kernel**;
start std_cauchy(x);
y=(1/(constant('pi'))*(1/(1+((x)##2))));
return (y);
finish std_cauchy;

**Generate x's from -n to n (grid)**;
do j=-5 to 5 by 0.01;
if j=-5 then x=j;
else x=x//j;
if j=-5 then kde=kde_norm(j);
else kde=kde//kde_norm(j);
if j=-5 then y=std_norm(j);
else y=y//std_norm(j);
if j=-5 then y_cau=std_cauchy(j);
else y_cau=y_cau//std_cauchy(j);
end;
*xrow=nrow(x);
*fxro=nrow(kde);
*yrow=nrow(y);
x_kde_y_ycau=x||kde||y||y_cau;

*print x_kde_y;
create gaus from x_kde_y_ycau[colname={x kde y y_cau}];
append from x_kde_y_ycau;
quit;
```

```
goptions reset=all i=stepsj ;
axis1 label=(angle=90 'kde');
axis2 label=(angle=90 'y');
axis3 label=('X') ;
symbol1 color=blue width=2 ;
title1
        'Plot of Estimated kernel(kde) on Cauchy distributed data';
/*proc gplot data=gaus;
plot kde*x / vaxis=axis1 haxis=axis3;
run;

goptions reset=title symbol;
symbol1 color=red width=2 ;
title2 'Plot of Standard Normal kernel(y) vs. x';
proc gplot data=gaus;
plot y*x /vaxis=axis2 haxis=axis3;
run;*/

goptions reset=all;
title3
        'Plot of Gaussian Estimated kernel(kde) Overlayed with
        Standard Normal kernel(y) and Theoretical Cauchy Kernel';
proc sgplot data=gaus;
series x=x y=kde;
series x=x y=y;
series x=x y=y_cau;
run;
```

Modelling seismic activity: A Bayesian network approach

Ntombi Mashila 12095576

WST795 Research Report

Submitted in partial fulfilment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr A de Waal, Co-supervisor: Mr MT Loots

Department of Statistics, University of Pretoria

02 November 2015

Abstract

The aim of the research is to investigate the factors that induce seismic activity in the Witwatersrand. In particular we want to find out if the presence of acid mine drainage in underground mining voids has an influence on seismic activity. Acid mine drainage is the flow of acid contaminated water from old mining areas and is notoriously known to pollute soil, dams and underground water. In DuPlessis' spatial assessment of the conditions that induce seismic activity in the Witwatersrand [10], four features were identified which are assumed to trigger seismic activity in the area, these are: the size and distribution of underground mining voids, the groundwater mobility, the rock type of the underground structure and the proximity to fault lines. In this report we set up a Bayesian network to investigate whether there is a relationship between the features assumed to trigger seismic activity and the actual occurrence of seismic activity in the Witwatersrand.

In the application, we make use of the software package Hugin® to model the occurrence of seismic activity in the Witwatersrand. This was done by constructing a Bayesian network of these features and estimating their conditional distributions. We also used the EM learning wizard in Hugin® to estimate the parameters in the Bayesian network.

The results show that only the underground mine voids and the rock type have a significant impact on the occurrence of seismic events. The other two features showed very little to no impact on seismic activity. The findings are also compatible with the results obtained by DuPlessis [10].

Declaration

I, *Ntombi Sandy Mashila,* declare that this essay, submitted in partial fulfilment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Ntombi Sandy Mashila

_____.
Dr A de Waal

_____
Mr MT Loots

_____
Date

Acknowledgements

My sincere appreciation goes out to both my supervisors for their insights and kind encouragement.

Contents

List of Figures

1 Introduction

In August 2014, South Africa experienced an earthquake with a magnitude as high as 5.5 on the Richter scale, as registered by the Council for Geoscience (CGS) in South Africa, making it one of biggest mining related seismic event in the country. Since its epicentre was near a gold mine this has led to increasing interest on the effects of mining activity on seismic events. Consequently, this research report investigates the environmental effects of mining in the Witwatersrand, in particular the impacts of Acid Mine Drainage (AMD) on Seismic activity. Based on the work of DuPlessis [10], four features were identified which are assumed to trigger seismic activity in the Witwatersrand, these are: the size and distribution of underground mining voids, the groundwater mobility, the rock type of the underground structure and the proximity to fault lines. Although other features do exist, these four were chosen because they can be represented by spatial data sets. A large portion of DuPlessis' work was devoted to finding data that is relevant and of good quality. His approach to the problem was to use the weighted overlay tool in ArcGIS to estimate the spatial distribution of the underground mine voids and then compare the estimates to a kernel density model of the locations of historic seismic events using the band statistics tool in ArcGIS (www.arcgis.com). He also carried out spatial correlation statistics to measure the spatial correlation between the features. As a recommendation for future work he suggested that an in-depth statistical analysis be carried out to determine the impacts of each feature on seismic activity, as well as the correlation between the features themselves.
In this report a Bayesian network is used to investigate whether there is a relationship between the features and the occurrence of seismic activity in the Witwatersrand. Specifically, we construct a Bayesian network of these features, then estimate their parameters and conditional distributions. A Bayesian network is a directed acyclic graph that represents the qualitative and quantitative relationships that exist between the random variables in a problem [1, 8, 6]. Bayesian networks are commonly used to perform statistical inference based on data, and are very useful in hypothesis testing and decision-making in cases of uncertainty. This is because a Bayesian network assigns a conditional distribution to each variable. Bayesian networks have the advantage that the modeller is able to incorporate prior assumptions and knowledge about the problem into the model. Experts in various fields have used Bayesian networks in fault diagnosis, software debugging, manufacturing control [5, 12], etc. Because of its popularity, Bayesian network based software packages have been developed to assist in decision-making where uncertainty is a factor. Hugin® (www.hugin.com) is such a package and will be used in the application.

2 Background Theory

2.1 Modelling with Bayesian networks

A Bayesian network is a model that provides a graphical and probabilistic description of the relationships between the most important variables in a system or problem of interest [4, 3]. In other words Bayesian networks model causal dependencies from one variable to another. A causal dependency from A to B implies that when A is in a certain state this has an influence on the state of B.
Each random variable in the model is depicted by a round node and the dependencies between the variables are represented by arrows, as seen in figure 1. A node consists of a set of states that the random variable can assume, as well as an associated conditional probability distribution. The conditional probabilities give the probability of the random variable being in a specific state given the state of the parent random variables. Nodes that are not connected represent random variables that are conditionally independent. Hence conditional independence implies that the model can be broken down into several sub-models, called Markov blankets [3]. The Markov blanket of a node consists of its parents, its children and the parents of its children [16]. It follows that each sub-model (Markov blanket) can be developed separately for each conditional relationship, using information that is available about the process; for instance, historical data, statistical correlations, or expert judgement [3]. A node has no parents when it is not connected to any other nodes in the model except its children, and the variable corresponding to that node can be described statistically by a marginal or unconditional probability distribution [3].
According to Borsuk et al [3] conditional dependence can be expressed as a function of the form:

$$X_i = f(P_i, \varepsilon_i)$$

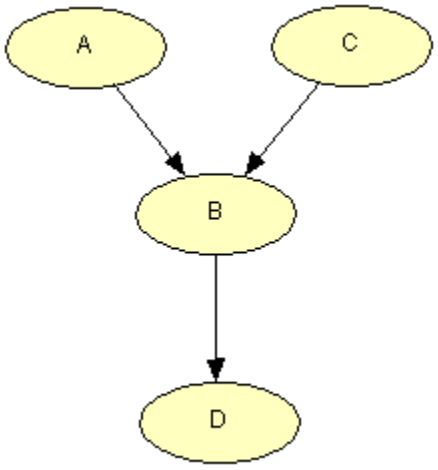


Figure 1: Model to illustrate conditional dependence (independence) between random variables

$i = 1, 2, ..., n$ where X_i is any given node, P_i the parent of that node, and ε_i the error term. The error term is an independent random variable with an arbitrary distribution, it represents information that was omitted or unaccounted for in the model [3]. Nodes that are conditionally independent, will simply be functions of the error term [3], i.e. for $j = 1, 2, .., n$

$$X_j = f(\varepsilon_j).$$

Usage

In general, Bayesian networks are used to form statistical inferences based on data. The inference can be predictive (forward analysis) or diagnostic (backward analysis) [1, 8]. As a result, Bayesian networks can be used to predict the occurrence of certain events by modelling their distributions, or to they can be used to influence management decisions in cases of uncertainty as decision support [13, 8].
A Bayesian network explicitly represents the cause-and-effect assumptions about a problem [4]. It takes into account the effects that different interventions can have on the problem, thus it is useful in implementing "what-if?" analysis [13]. Bayesian network development has a strong focus on stakeholder participation in the development of the model, which can add credibility to the final model that is implemented [13]. This provides a transparent system for exploring the relationships between the variables [1]. Hence, the model is able to capture a common understanding of the problem and can help settle conflicts among the stakeholders, which are the people who may be affected by the decisions being implied [13].

Data sources

Bayesian networks can make use of both discrete and continuous data in the definition of the model [13]. Instances of continuous (qualitative) data that may be considered include surveys, historical or time series data while the discrete (qualitative) type may be expert opinion or stakeholder belief. Other data sources include GIS data, empirical data and subject literature [14, 8].

Model development

Probabilistic networks are models that provide a convenient way of visualizing the probabilistic dependencies between random variables, and hence can allow us to perform techniques such as learning, prediction and diagnosis [5]. This is possible because Probabilistic networks specify the conditional distributions of the

random variables. They have been developed and applied quite extensively in different fields of research including medical diagnosis, artificial intelligence, statistics and forensics. Learning the probabilities is a complicated process since it involves learning the structure of the model and its parameters once the structure is constructed. It is also plagued with the difficulty of dealing with missing values and latent variables.
Bayesian networks are a famous branch of Probabilistic networks. The Bayesian method is a growing area of research which combines prior knowledge about the problem with the information derived from the data, and helps us learn the parameters and structure of the Bayesian network. Learning the structure of the Bayesian network helps us to assess the effect that the variables have on each other, while learning the parameters allows us to determine the conditional probabilities of new instance rather than estimating them directly [5]. The prior information is quantified in terms of the initial conditional probabilities. The conditional probabilities can be given either implicitly as a distribution function or explicitly as tables of values called conditional probability tables, both specify the probability distribution [5]. The parameters of the conditional distribution can be derived easily by using maximum likelihood when the data set is complete, i.e. with no missing values. In the case where the data is incomplete statistical approximations are used instead to find the parameters. A technique that is commonly used in such instances is the EM algorithm. The problem of Bayesian networks reduces to finding the Bayesian network that best represents the probability distribution of the data. This end is the same as finding the structure and the parameters that yield the maximum posterior probability [7]. This process is implemented using a heuristic search technique that is based on a scoring function that evaluates how each structure fits the data [5]. The two common scoring metrics used in Bayesian networks are the Bayesian scoring function and the minimum description length (MDL) function [11].

2.2 Seismic activity in the Witwatersrand

The Witwatersrand basin is a large complex of interconnected mines which was developed as a result of the nature of the gold deposits in the Witwatersrand. It is made up of the eastern basin, the western basin and the central basin, and covers the city of Johannesburg and its surrounding areas. The location of these basins can be seen in figure 13 in the appendix. Although there is no mining taking place at present, seismic activity still occurs in the Witwatersrand. The poor closure of mine infrastructure and the lack of mining land and water rehabilitation has led to significant negative impacts on the environment. When mining operations extended deeper than the water table it was normal for mining voids to fill up with water. Extracting the water to the surface was necessary in order to mine deeper and access the mineral reserves. When most mines closed down in the Witwatersrand the associated operations of extracting mine water from underground voids also ceased. Additional water also came from neighbouring mine voids through underground interconnections, resulting in the flooding of the mine voids. Currently, the basin has linked into one with the water level estimated to be less than 400m below the surface, across the basin [10]. The large volume of water creates underground pressure, which can affect the stability of artificial and natural fractures, causing them to slip and trigger seismic activity[10]. Over time this water also becomes contaminated by sulphur-rich mining wastes within the underground voids. One main consequence of mine void flooding is the formation of Acid Mine Drainage (AMD). In addition, the mineral Pyrite found in rock material oxidises when exposed to oxygen and water, and is known to produce acid in the underground voids. An earth tremor can be generated by the slippage or failure of underground faults that are affected by acid mine drainage, this is referred to as acid water induced seismic activity [10]. The trigger of seismic activity in the Witwatersrand is assumed to be high levels of acid mine drainage in the underground voids, since acid water is known to be more corrosive than uncontaminated water and can result in increased seismic activity. The occurrence of seismic activity in the Witwatersrand is unique in the sense of the size of the Witwatersrand basin, the interconnectedness of the mine voids and the affected basins being in close proximity to densely populated urban centres.
Based on the study by DuPlessis [10], the rock structure of the Witwatersrand basin is made up of three types, namely: igneous, sedimentary and metamorphic rock. Igneous rock is rock that is formed by the cooling and hardening of magma beneath the surface of the earth. The temperature at which the magma cools determines the composition of the rock and the types of minerals that occur in the rock. The main types of igneous rock covered in the study were dolerite, quartzite, basalt, granite, syenite and granodiorite. Most rock structures across the Earth are made up of igneous rock that is covered by layers of sedimentary

rock. Sedimentary rock is a secondary rock structure because it is formed by broken pieces of existing rocks that are compacted together. Sedimentary rock is formed in one of three ways: when small pieces of other rock types become compacted and cemented, or when dissolved minerals are left behind during evaporation, or when debris created by organic processes accumulates. The main types of sedimentary rock covered in the study by DuPlessis [10] were diamictite, shale, sandstone and dolomite. Metamorphic rock is rock that is formed when existing rock is subjected to heat and pressure which alters the physical and chemical properties of the rock. The main types of metamorphic rock considered were quartzite and migmatite.
Groundwater is water that accumulates beneath the earth's surface in the pores and crevices of rocks. The way in which water can move underground is determined by the rocktype and the porosity of the underground structure. Three types of rock porosity were investigated in the study by DuPlessis [10]: intergranular, fractured and karst. An intergranular structures is made out of sedimentary rock, and allows water to move between the grains of soil, sediment and rock. A fractured structure can form from metamorphic or igneous rock. The only way that water passes through this structure is through the fractures that exist in the rock. Karst structures on the other hand are characterised by cave-like features formed when rocks have been chemically weathered by solution.

3 Application

3.1 Data

The seismic events data was extracted from the national seismicity dataset obtained from the Council of Geoscience (www.geoscience.org.za). It contains a record of all seismic events that occurred between 2001 and 2013. This dataset was merged with a dataset obtained from the University of Pretoria Natural Hazard Centre (http://www.up.ac.za/university-of-pretoria-natural-hazard-centre-africa), which contains records of seismic activity from 1966 to 2010. The complete dataset has a total of 1008 cases, since we only included data that coincided with the study area. In addition, the data excludes all seismic events that occurred while the mines were still in operation. The rock type data was extracted from the geological distribution data of the Witwatersrand, provided by Digby Wells Environmental (http://www.digbywells.com/en/). The rocks were then classified into the three main categories: igneous, sedimentary or metamorphic. The ground water mobility data was also derived from the geological distribution data. It contains the three ways in which groundwater moves through the underground rock structure, namely, karst, fractured or intergranular. The size and location of underground mine voids were estimated using the locations of existing mine dumps and shafts, since this information is not made available to the public and belongs to the associated mining companies. A snapshot of the data can be seen from figure 14 in the Appendix.

3.2 Approach

Bayesian methods are an alternative approach to statistical estimation. The Bayesian approach attempts to estimate the parameters, $\boldsymbol{\theta}$, of a distribution F given a random sample from the distribution. It is assumed that the parameters are random variables and will therefore have a distribution. Suppose that A is any event in the sample space Ω and suppose $B_1, B_2.., B_n$is a partition of the sample space, this means that

$$B_i \cap B_j = \oslash, i \neq j$$

and

$$\cup B_i = \Omega, \forall i.$$

The probability of Bi given A is then computed as:

$$P(B_i|A) = \frac{P(A|B_i)P(B)_i}{P(A)} \tag{1}$$

where

$$P(A) = \sum_{i=1}^{n} P(A|B_i)P(B_i).$$

This result is known as Bayes' theorem. Let A be the occurrence of a seismic event in the Witwatersrand basin. For this problem we define the following features:

Feature	Description
B_1	The presence of mine voids in the area
B_2	The presence of fault lines in the area
B_3	Ground water mobility
B_4	The rock type

Table 1: Definition of events

Bayes' theorem allows us to calculate the probability of a seismic event being caused by one of the features described in table 1. Equation 1 can be rewritten as:

$$P(B_i|A) \propto P(A|B_i)P(B_i) \tag{2}$$

In words, Equation 2 states that the posterior probability is proportional to the likelihood times the prior probability. We note that the likelihood $P(A|B_i)$ is the conditional probability of a seismic event given the possible state of nature of the feature B_i. We use the data as described in the previous section to derive a likelihood for the parameters. The EM (expectation-maximization) algorithm is a common iterative method of finding the maximum likelihood estimates of the parameter from the data when we have missing or incomplete data or in the case where the maximum likelihood cannot be solved explicitly [9, 2]. The EM algorithm is a two step procedure that is made up of the following: the E step which computes the expected value of the complete-data log-likelihood in terms of the unknown data, given the current parameter estimates and the observed data; the M step which maximizes the expectation computed in the E step to obtain the new parameter estimates [2]. This procedure is repeated iteratively until we have convergence of the estimates.
The Bayesian network approach when used for prediction, forecasting, etc. can be considered as *classification,* since it involve assigning class labels to the features in a dataset [6]. A naive Bayes classifier is a simple kind of Bayesian network that has the classification node as the parent node of all the other nodes, with no other connections allowed. The naive Bayes classifier is a very effective classifier, especially when the features are not strongly correlated [15]. It is based on the assumption that all the features are independent of each other given the class value. Figure 2 depicts the naive Bayes classifier for this model, represented as a Bayesian network. The class node in this case is whether or not a seismic event occurred.

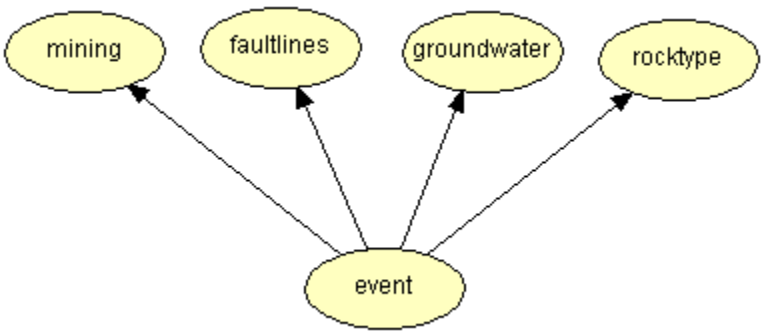


Figure 2: Naive Bayes Model

A naive Bayes classifier learns the parameters of the structure from the training data [6, 11]. To estimate the parameters of the naive Bayes structure in Figure 2, we made us of the built-in EM learning wizard in Hugin®. The learning algorithm in this case comprised of 1008 observations, and performs a total of 100 000 iterations, with a tolerance of 0.0001, which is the default value in Hugin®. Experience tables were used to formulate the prior probabilities of the parameters. They are based on expert judgement and past cases. If it is believed that the present data gives an approximate conditional distribution for the parameters, we initialize the count for the experience table to a higher value, otherwise we set the count to a small value. The count gives the number of observations made so far for a given feature. The initial conditional probabilities can be seen from tables 3, 4, 5, 6 and 7 below.

faultlines | groundwater | rocktype | event | mining

event	yes	no
yes	0.416431	0.227481
no	0.583569	0.772519
Experience	1059	1965

Figure 3: Mining CPT

faultlines | groundwater | rocktype | event | mining

event	yes	no
yes	0.175637	0.210687
no	0.824363	0.789313
Experience	2471	4585

Figure 4: faultlines CPT

faultlines | groundwater | rocktype | event | mining

event	yes	no
Karst	0.268519	0.26087
Fractured	0.188273	0.182276
Intergranular	0.543208	0.556854
Experience	2470.999647	4585

Figure 5: groundwater CPT

faultlines | groundwater | rocktype | event | mining

event	yes	no
igneous	0.411042	0.328881
Alluvium	0.061351	0.026713
metamorphic	0.205522	0.235392
sedimentary	0.322085	0.409014
Experience	2470.999682	4585.000066

Figure 6: rocktype CPT

faultlines	groundwater	rocktype	event	mining
yes	0.350198			
no	0.649802			
Experience	7056			

Figure 7: Event CPT

Based on the locations of historic seismic activity, we want to investigate the causal relationships between the features identified in table 1 and the actual occurrence of seismic activity in the Witwatersrand. Figure 8 gives the marginal probabilities (likelihood estimates) from the EM algorithm.

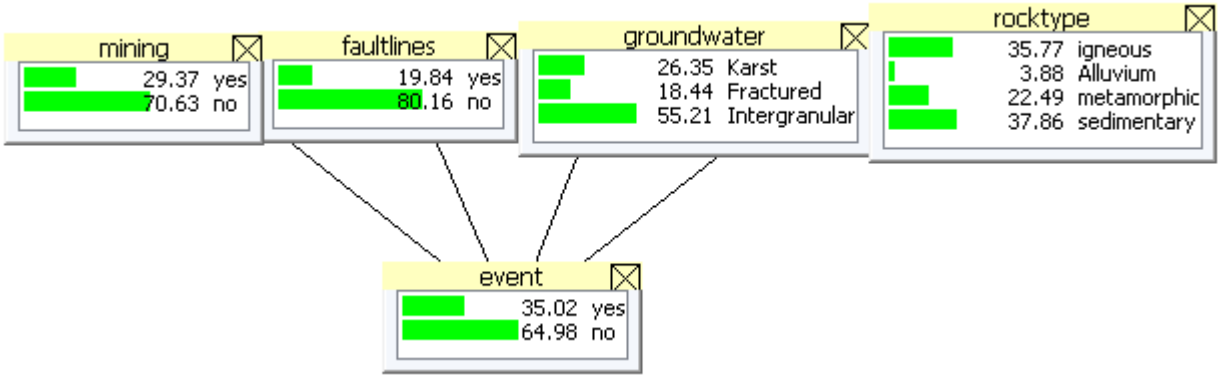


Figure 8: Initial probabilities without inference

4 Results

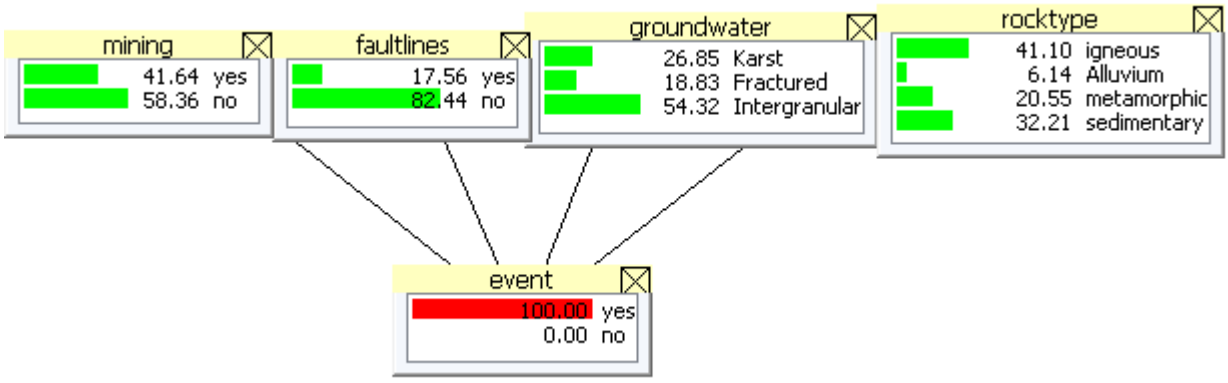


Figure 9: Posterior probabilities

Given that a seismic event occurred, the probability of fault lines in the area decreased from 0.1984 without inference (figure 8) to 0.1756 (figure 9), when the probabilities for all the other features increased. This clearly indicates that seismic activity occurs with little probability when fault lines are in close proximity to a mining area. Furthermore we could infer that fault lines have a negative influence on the occurrence of seismic activity. This also agrees with DuPlessis' finding that locations of fault lines are negatively correlated with the locations of historic seismic activity, which could possibly mean that the proximity to fault lines does not induce seismic activity in the Witwatersrand. We delete fault lines from the model and carry out

a what-if analysis to investigate whether the remaining three features have an effect on the occurrence of seismic activity in the Witwatersrand.
Figure 10 gives a diagnostic form of the Bayesian network. It indicates that mining and rock type do have an influence on seismic activity, and that the underground water mobility has very little influence, since the probabilities remained fairly constant. Figure 11 is the prescriptive form of the Bayesian network. It shows that igneous rock, closely followed by sedimentary rock will most likely influence seismic activity than alluvium and metamorphic rock. Thus, from Figure 12, we can predict that if mining was present in an area, the rock type of the area is Igneous and the groundwater mobility is fractured the probability of a seismic event occurring would be 0.5602.

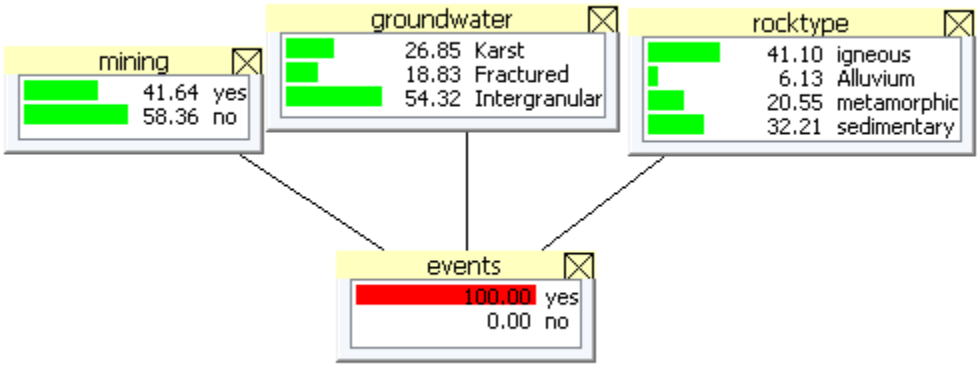


Figure 10: Diagnostic mode

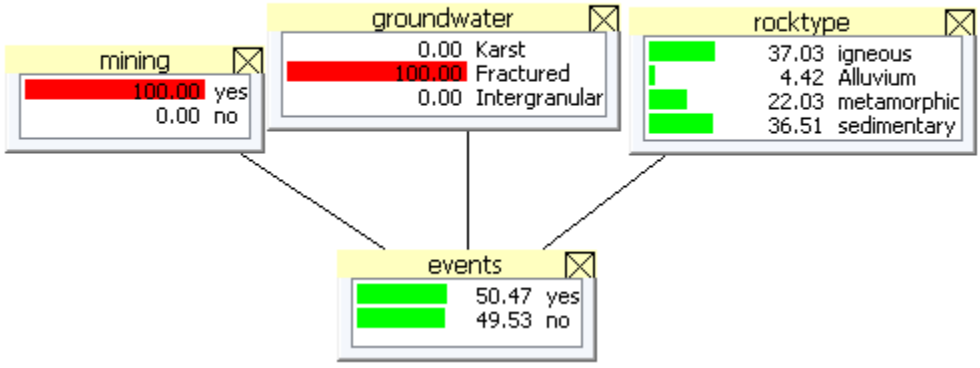


Figure 11: Prescriptive mode

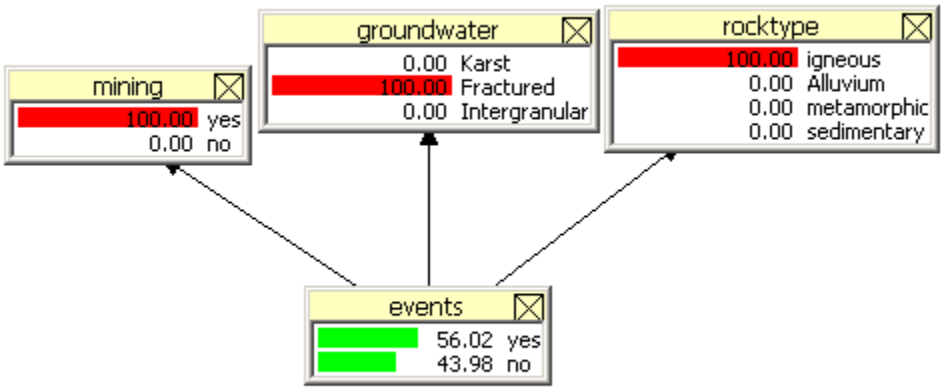


Figure 12: Predictive mode

5 Conclusion

From the results, mining has the most significant impact on seismic activity, and the rock type is the only other feature that has an impact. The mining in this instance refers to the underground voids left behind by mining. This means that underground mine voids do contribute to seismic activity. The findings are also compatible with the results obtained by DuPlessis [10]. Hence we conclude that acid mine drainage could possibly be the leading cause of seismic activity in the Witwatersrand.

In future, expert knowledge can be incorporated into the model to improve its accuracy. This could be done by conducting a workshop with experts in mining and seismology. In addition, the Bayesian network can be extended to include other features not considered in this study, and the assumption of independence can be relaxed to see how the features relate to each other. In other words, the dependencies between the features can be investigated, for example, how acid mine drainage reacts to the different rock types. We can also set up the Bayesian network to answer questions like, "what is the maximum earthquake magnitude that can be expected and if there should be a cause for alarm?"

References

[1] Michelle T Bensi, Armen Der Kiureghian, and Daniel Straub. *A Bayesian network methodology for infrastructure seismic risk assessment and decision support.* Pacific Earthquake Engineering Research Center, 2011.

[2] Jeff A Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126, 1998.

[3] Mark E Borsuk, Craig A Stow, and Kenneth H Reckhow. Integrative environmental prediction using Bayesian networks: A synthesis of models describing estuarine eutrophication. *Integrated Assessment and Decision Support*, 2:102–107, 2002.

[4] Mark E Borsuk, Craig A Stow, and Kenneth H Reckhow. A Bayesian network of eutrophication models for synthesis, prediction, and uncertainty analysis. *Ecological Modelling*, 173(2):219–239, 2004.

[5] Wray L Buntine. A guide to the literature on learning probabilistic networks from data. *Knowledge and Data Engineering, IEEE Transactions on*, 8(2):195–210, 1996.

[6] Jie Cheng and Russell Greiner. Comparing Bayesian network classifiers. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 101–108. Morgan Kaufmann Publishers Inc., 1999.

[7] Gregory F Cooper and Edward Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9(4):309–347, 1992.

[8] A De Waal and T Ritchey. Combining morphological analysis and Bayesian networks for strategic decision support. *ORiON*, 23(2):105–121, 2007.

[9] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (methodological)*, pages 1–38, 1977.

[10] Izak du Plessis. Spatial assessment of the conditions that induce seismic activity in the witwatersrand. Honours research report, University of Pretoria, Faculty of Natural & Agricultural Sciences, Department of Geography, Geoinformatics and Meteorology, 2014.

[11] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

[12] David Heckerman, Abe Mamdani, and Michael P Wellman. Real-world applications of Bayesian networks. *Communications of the ACM*, 38(3):24–26, 1995.

[13] Rebecca A Kelly, Anthony J Jakeman, Olivier Barreteau, Mark E Borsuk, Sondoss ElSawah, Serena H Hamilton, Hans Jørgen Henriksen, Sakari Kuikka, Holger R Maier, and Andrea Emilio Rizzoli. Selecting among five common modelling approaches for integrated environmental assessment and management. *Environmental Modelling & Software*, 47:159–181, 2013.

[14] Hildegarde Koen, JP De Villiers, Gregor Pavlin, Alta de Waal, Patrick de Oude, and Franek Mignet. A framework for inferring predictive distributions of rhino poaching events through causal modelling. In *Information Fusion (FUSION), 2014 17th International Conference on*, pages 1–7. IEEE, 2014.

[15] Pat Langley, Wayne Iba, and Kevin Thompson. An analysis of Bayesian classifiers. In *Association for the Advancement of Artificial Intelligence*, volume 90, pages 223–228, 1992.

[16] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, 2014.

Appendix

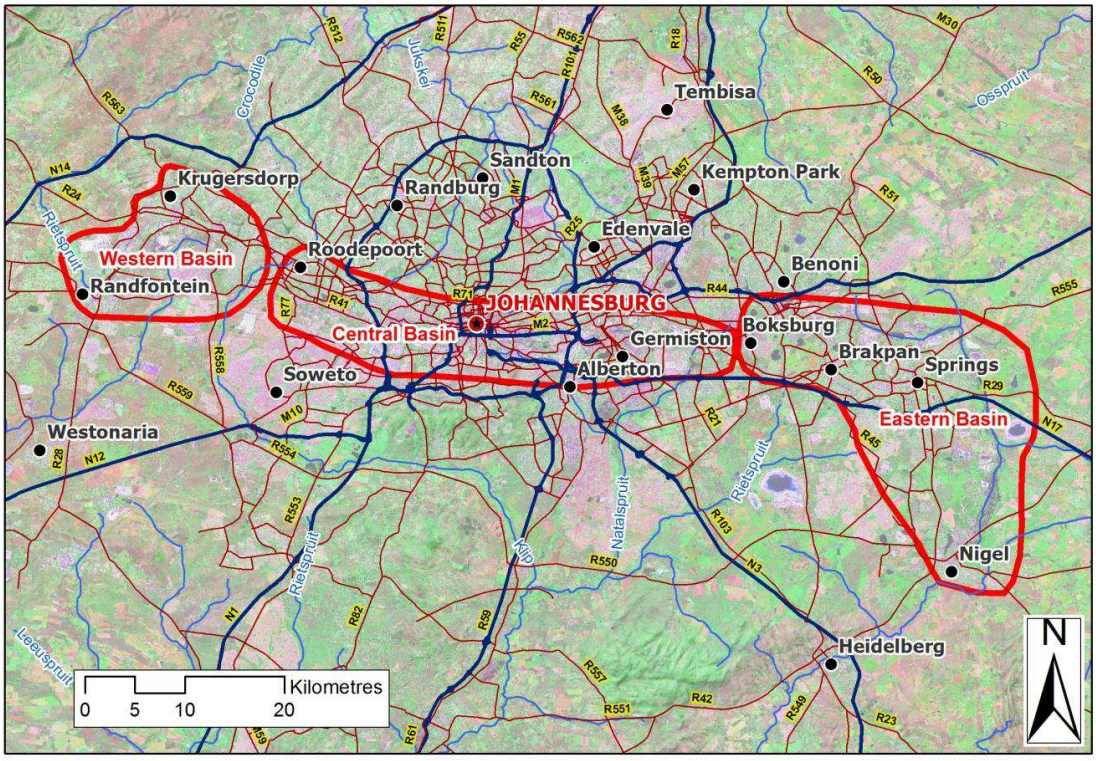


Figure 13: Witwatersrand basin

#	events	mining	faultlines	groundwater	rocktype
0	no	no	no	Intergranular	metamorphic
1	no	no	no	Intergranular	metamorphic
2	no	no	no	Intergranular	metamorphic
3	no	no	no	Intergranular	sedimentary
4	no	no	no	Intergranular	sedimentary
5	no	no	no	Karst	sedimentary
6	no	no	yes	Karst	sedimentary
7	no	no	yes	Karst	sedimentary
8	no	no	no	Karst	sedimentary
9	no	no	no	Karst	sedimentary
10	no	no	no	Karst	sedimentary
11	no	no	no	Karst	sedimentary
12	no	no	no	Karst	metamorphic
13	no	no	no		igneous
14	no	no	no	Intergranular	metamorphic
15	no	no	no	Fractured	sedimentary

Figure 14: Snapshot of the data

Kernel density estimation for bounded distributions

Hodi Patience Matemane 11119064

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: M.T Loots

Department of Statistics, University of Pretoria

02 November 2015 (final draft)

Abstract

When we use kernel density estimation to estimate the density function of a bounded data set, we expect the underlying density function of the data set to also have a boundary condition, however a problem arises where by the estimator sometimes gives values that are not within the bounds of the data set [2]. Different methods have been developed to correct the standard kernel density estimator in cases where the data set of interest is bounded. This research report gives an outline of the different methods that are used to estimate the density function of data that is bounded using kernel density estimation.

Declaration

I, *Hodi Patience Matemane,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Hodi Patience Matemane

_____
M.Theodor Loots

Date
02 November 2015

Contents

List of Figures

List of Tables

1 Introduction

Kernel density estimation (KDE) sometimes called the Parzen-Rosenblatt window method (name given after creators Emanuel Parzen and Murray Rosenblatt) is a distribution-free approach of estimating the probability density function of a random variable, where data is smoothed in order to make logical conclusions about a population using a finite sample [2]. Kernel density estimation makes use of different non-negative functions that integrate to 1 and are 0 mean called kernels and a parameter $h > 0$ called the smoothing parameter (also known as the bandwidth), to estimate the shape of a density function $f(x)$ of a sample that is independent and identically distributed [5]. A problem with kernel density estimation is encountered when we want to estimate the density function of data that comes from a distribution that is bounded, since most of the kernels that are used are defined on the entire real line, therefore the estimator tend to give us values that are meaningless [2]. If we want to estimate the density function of the weight of individuals, for example, kernel density estimation will sometimes give us negative values for weight which is meaningless because an individual can never have negative weight. In this research report we are going to explore some of the methods that can be used to overcome this problem and also to improve the standard kernel density estimation in cases where we have bounded distributions. There is no general correction method that has been found that will always give the best estimator [2]. We are going to look at three different methods, which are: method of ignoring meaningless values, the transformation method and the reflection method. Some methods are better suited for estimating density functions of data sets that are semi-bounded and other methods are suited for estimating density functions of data sets that are bounded on both sides.

2 Kernel density estimation

Kernel density estimation is a distribution-free technique of estimating the density function of a random variable. The technique was first defined in Rosenblatt (1956) and Parzen (1962).

Definition 1. Let $x_1 \dots x_n$ be an independent and identically distributed sample of size n from an unknown distribution $f(x)$, an estimate of the density is given by the function

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} k(\frac{x - X_i}{h})$$

Where the function $k(.)$ is called the kernel function and the parameter $h > 0$ is called the smoothing parameter (or the bandwidth).

2.1 Kernel functions

A kernel function satisfies the condition $\int_a^b k(x)dx = 1$,where a is the lower bound and b is the upper bound of the selected kernel function, this implies that the kernel function is a density function. The kernel function has mean 0, so that for a fixed X_i the kernel function has its center exactly at X_i [2]. There are a large variety of kernel functions that can be used. These are some commonly used kernel functions: Uniform, triangular, biweight, triweight, Epanechikov, normal and many others as shown in Table 1. The performance of the kernel function is measured by the MISE (mean integrated squared error) or the AMISE (Asymptotic MISE) [2]. The Epanechnikov kernel is optimal in a mean square error sense therefore it is widely used [2].

Name of kernel function	Equation
Quadratic or biweight	$K(t) = \frac{15}{16}(1-t^2)^2 \quad \lvert t\rvert \leq 1$
triangular	$K(t) = \frac{35}{32}(1-t^2)^3 \lvert t\rvert \leq 1$
Triangular	$K(t) = (1-\lvert t\rvert) \quad \lvert t\rvert \leq 1$
Gaussian	$K(t) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}t^2) \quad t\epsilon\mathbb{R}$
Epanechnikov	$K(t) = \frac{3}{4}(1-t^2) \quad \lvert t\rvert \leq 1$

Table 1: Some kernel functions in literature

2.2 Bandwidth

The parameter $h > 0$ is called the smoothing parameter (or the bandwidth). The bandwidth controls how wide the probability mass is spread around a point. The selection of the bandwidth is the most crucial part of kernel density estimation since we do not want to under-smooth or over-smooth our estimate, as illustrated in Figures 1 with the relevant SAS code in Appendix 1. The bandwidth controls the roughness or smoothness of the density estimate [5]. The most frequently used method for bandwidth selection is the Silverman's rule of thumb, the bandwidth is calculated as follows

$$h = (\frac{4\sigma^5}{3n})^{\frac{1}{5}}$$

where $\sigma = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2$ is the standard deviation and n is the sample size [5]. Figures 1,2 and 3 below illustrate how choosing different bandwidths has an effect on the smoothness of the density function, Silverman's rule of thumb is the easiest method for calculating h when working with a univariate sample.

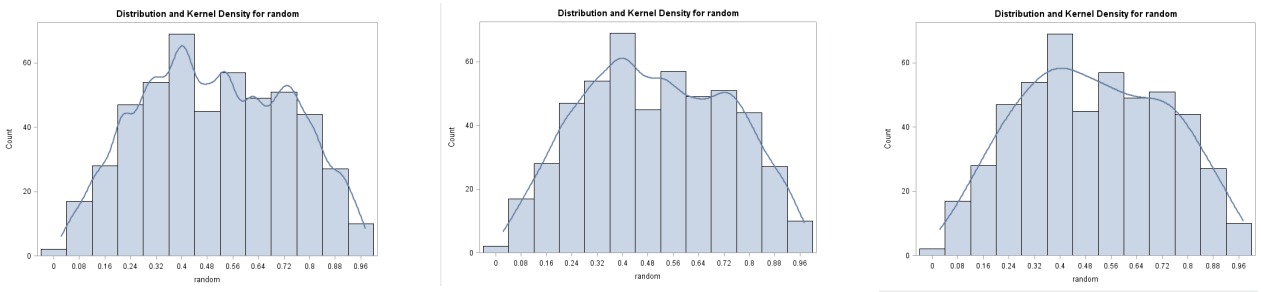


Figure 1: Kernel density function of a data set from a Beta(2,2) distribution using different bandwidths [1]

For more information on bandwidth selection the reader is referred to [Wand and Jones 1994].

3 Boundary correction

The problem with kernel density estimation begins when we want to estimate the density function of data that is bounded, we expect the underlying density to also be bounded. However since the kernels that are used in kernel density estimation are generally unbounded they tend to give values that are meaningless for some estimators, the estimator tends to assign mass to values that are not in the boundary interval, as illustrated in Figure 4 [2]. There is no general correction method that has been found that will always give the best estimator, since the performance of the estimator depends on some properties of the underlying density, for example the shape at the bounds. Thus, for a given bandwidth, there is a fixed information interval for each x and every observation taken from outside this interval has no influence on the estimate. If for example we consider a density function which is continuous on the region $[0, \infty)$ and is zero for negative

values, given a bandwidth h, the interval $[0, h)$ is the boundary interval and the interval $[h, \infty)$ is the interior interval. Since the interior interval is not bounded the primary interest lies in estimating the density function in the boundary interval [2]. A problem arises when a given value for x is smaller than the chosen bandwidth h, unless $x \geq h$ the estimator will not be asymptotically unbiased and will be inconsistent [2].

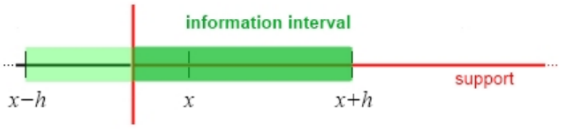


Figure 2: Information interval for bounded densities [2]

When working with a bounded data set the information interval does not follow the boundary condition, hence the estimator will sometimes give values that are not within the bounds. The estimation is based on a reduced amount of information leading to severe bias: the resulting estimate becomes more inaccurate [2]. Different methods are used in achieving a boundary correction. In this research report we are specifically focusing on three methods, which are: the method of ignoring meaningless estimates, the transformation method and the reflection method.

3.1 Ignoring meaningless estimators

If we are working with a bounded data set, we expect the underlying density function to also be bounded. If we are only interested in the general shape of the density and we do not want to capture the exact shape then we can choose to ignore meaningless values (values that are not within the bounds of the density). For example if we want to get a density function of a variable that is bounded on one side i.e density is continuous on the interval $[0, \infty)$, one way of ensuring that $\hat{f}(x)$ is zero for values that are less than zero, is to calculate the estimate for positive values of x ignoring the boundary conditions, and then to set $\hat{f}(x)$ to zero for values of x that are less than zero [3]. The problem with this method is that the estimate obtained will no longer be a density function, meaning that it will no longer integrate to unity ($\int_{-\infty}^{\infty} \hat{f}(x) \neq 1$), however a constant k can be chosen that will make the density integrate to unity [4]. Another drawback with this method is that, points that are near zero will contribute less to $\int_{0}^{\infty} \hat{f}(x)$ than points that are far from zero. Even if we choose a constant k that will make the estimate a density function, the weight of the distribution near zero will be overpowered by the weight of points that are far away from zero [4]. The subsections that follow will look at other approaches that will give better results compared to the method of ignoring meaningless values.

3.2 Transformation method

If we have random variables $X_1, \ldots, X_n$ that are bounded either on one side or on both sides, we could instead work with a set of transformed random variables $Y_j = g(X_i)$ for $i = 1, \ldots.., n$ that are not bounded [3]. The estimator for the density of X can be recovered if the transformation g is invertible. The data can first be transformed using g, then the kernel estimator for $Y = g(X)$ can be calculated using standard kernel density estimation and then inverting the transformation in order to recover the original random variable X using the following formulas:

$$f_y(y) = \frac{f_x(x)}{\mid g'(x) \mid}$$

and

$$f_x(x) = \frac{f_y(y)}{|\ \frac{d}{dx} g^{-1}(y)\ |}$$

For example, if we have a random variable X that is uniform on $[0,1]$, we can use the logit function to map the transformation from $[0,1]$ to $\mathbb{R}$ [3].

$$logit(x) = log(\frac{x}{1-x})$$

The logit function maps [0,1] to $\mathbb{R}$ and it is invertible with inverse :

$$\begin{aligned} logit^{-1}(x) &= logistic(x) \\ &= \frac{e^x}{1+e^x} \end{aligned}$$

Then the random variable X will be recovered using the formula:

$$\begin{aligned} \hat{f}_x(x) &= \frac{\hat{f}_y(y)}{\frac{d}{dy} g^{-1}(y)} \\ &= \frac{\hat{f}_y(logit(x))}{x(1-x)} \end{aligned}$$

We will then get a better estimate that is within the boundary [0,1]. When we work with a random variable that is bounded on one side $[0,\infty)$, we use the transformation function $g(x) = log(x)$, and we get:

$$\hat{f}_x(x) = \frac{\hat{f}_y(log(x))}{x}$$

The transformation method can be used for both semi-bounded densities and for densities that are bounded on both sides.

3.3 Reflection method

Sometime we work with variables that represent physical measures such as time, weight, length and many more, these variables have a natural lower boundary hence we also assume that the underlying true density is also bounded. If we assume that we are working with a density that is bounded only on one side, then we are working with a density that is continuous on $[0,\infty)$ and is 0 for $x < 0$ or a density that is continuous on $(-\infty, 0]$ and is 0 for $x > 0$. We can reflect the data points$X_1, \ldots, X_n$ by the origin and then work with the new random variables $Y_j = \begin{cases} -X_j & j\varepsilon\,\{1,\ldots,n\} \\ X_{2n-j} & j\varepsilon\,\{n+1,\ldots,2n\} \end{cases}$ [2]. This approach does not only generate a twice as large sample but it also yields a sample drawn from a density with unbounded support, therefore the standard kernel density estimation can be applied to the data which is now of size $2n$

$$f^*(x) = \frac{1}{2nh} \sum_{i=1}^{2n} k(\frac{x - Y_i}{h}) \tag{1}$$

Equation (1) is the formula for the standard kernel estimator of size $2n$, hence it integrates to 1 when integrated over the whole real line and the estimate is symmetric around the origin [2]. The natural way to get an estimate with support $[0,\infty)$ that will integrate to 1 is the following:

$$\hat{f}_{refl}(x) = \begin{cases} 2f^*(x) & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2}$$

Equation (2) is referred to as the reflection estimator, and can also be formulated as:

$$\hat{f}_{ref}(x) = \frac{1}{nh}\sum k(\frac{x-X_i}{h}) + k(\frac{x-X_i}{h}) \quad for \;\; x \geq 0$$

In the interior, where $x \geq h$ the reflection estimator becomes equal to the standard kernel density estimator, since if an observation X_i falls into the information interval of an interior point, surely no reflected observations $-X_j$ will be contained in it [2]. The reflection method is better suited for semi-bounded density functions, since we can easily reflect the data points along the origin. The advantage of the reflection method is that the estimator is a density function (hence it integrates to 1 over the whole real line without assigning any mass to the negative axis) [2]. The following proof shows that the reflection method integrates to unity [2].

$$\begin{aligned}
\int_{-\infty}^{\infty} \hat{f}_{ref}(x)dx &= 0 + \int_{0}^{\infty} \frac{1}{nh}\sum_{i=1}^{n}\left(k(\frac{x-X_i}{h}) + k(\frac{x+X_i}{h}\right)dx \\
&= \frac{1}{nh}\sum_{i=1}^{n}\left(\int_{0}^{\infty} k(\frac{x-X_i}{h})dx + \int_{0}^{\infty} k(\frac{x+X_i}{h})dx\right) \\
&= \frac{1}{nh}\sum_{i=1}^{n}\left(\int_{-\frac{X_i}{h}}^{1} k(t)hdt + \int_{\frac{X_i}{h}}^{1} k(t)hdt\right) \\
&= \frac{1}{nh}\sum_{i=1}^{n}\left(\int_{-\frac{X_i}{h}}^{1} k(t)hdt + \int_{\frac{X_i}{h}}^{-1} k(t)hdt\right) \\
&= \frac{1}{nh}\sum_{i=1}^{n} h\int_{-1}^{1} k(t)dt \\
&= 1
\end{aligned}$$

4 Application

Kernel density estimation has a wide variety of applications in practice, the examples includes applications in economics, signal processing and many more fields. Estimating the density function of a given data set makes the data easier to interpret since the general trend of the data can be observed, and to describes the relative likelihoodillustated of a random variable to take on a given value.

4.1 Application of the method of ignoring meaningless values

We want to estimate the probability density function of the marks in percentage of a sample of 100 students for a given module. A random sample of 100 students have been selected, PROC KDE function in SAS can be used to estimate the density function, however PROC KDE has a disadvantage of performing operations in the background that we do not understand, therefore to understand the procedure a lot better we make use of PROC IML as illustrated in the SAS code in Appendix 2. A random sample of 100 students was selected, the Gaussian kernel was used and the bandwidth was calculated using the Silverman's rule of thumb. Figure 5 shows the density function of the sampled data set, the problem is that the estimated density function assigns mass to values outside the boundary condition $[0, 100]$, however we know that the percentage mark that a student obtains may not go below 0 or above 100.

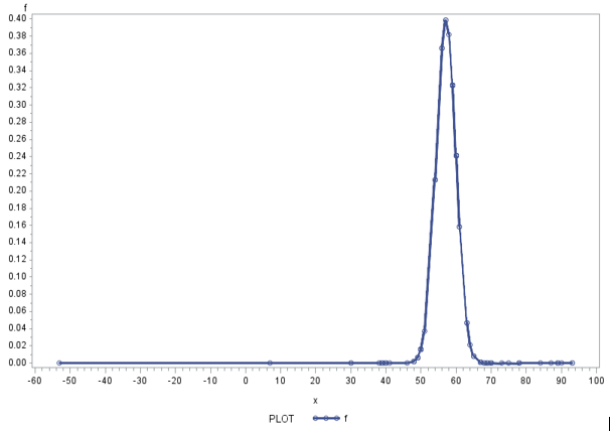


Figure 3: Estimated density function before boundary correction [1]

A better estimate of the density function can be calculated using the method of ignoring meaningless estimates, the same parameters will be used but now we will be working with a refined data set that excludes values that are not within the boundary of interest, Figure 6 shows a better density function with the boundary correction.

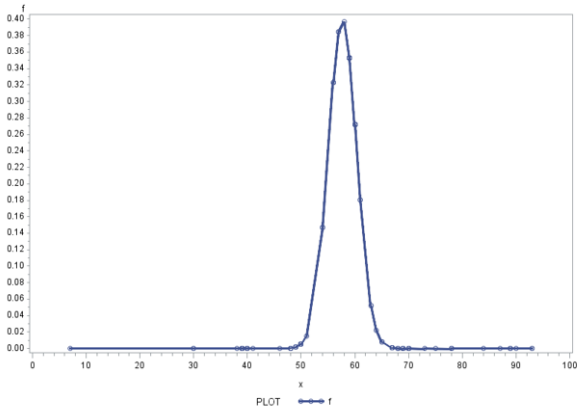


Figure 4: Estimated density function after ignoring the meaningless values [1]

The method of ignoring meaningless values gives a better density function that is easier to interpret and makes more sense.

4.2 Applications of the reflection method

The reflection method is used to estimate the density function of semi-bounded data sets, for example time, weight, temperature and other variables with a natural bound. We are going to illustrate the reflection method by an example, in this example we want to estimate the density function of the eruption lengths (in minutes) of 110 eruptions of Old Faithful geysers, we already know that eruption time cannot be negative hence we expect our function to be bounded by 0. By making use of PROC IML as illustrated in the SAS code in Appendix 3, we estimate the density function of the eruption lengths using kernel density estimation, we again use the Gaussian kernel and Silverman's rule of thumb to calculate the bandwidth. Figure 7 shows the density function of the eruption times before the reflection method was implemented.

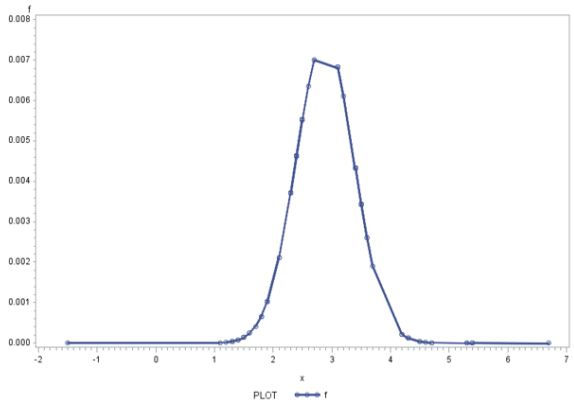


Figure 5: Estimated density function of eruption lengths before boundary correction [1]

A better and more accurate density function can be constructed using the reflection method. The data points are reflected to work with a new sample of 220 observations, however the data points have to be reflected back only on the side where the data is defined to obtain a more accurate density function that is easier to interpret as shown in Figure 8.

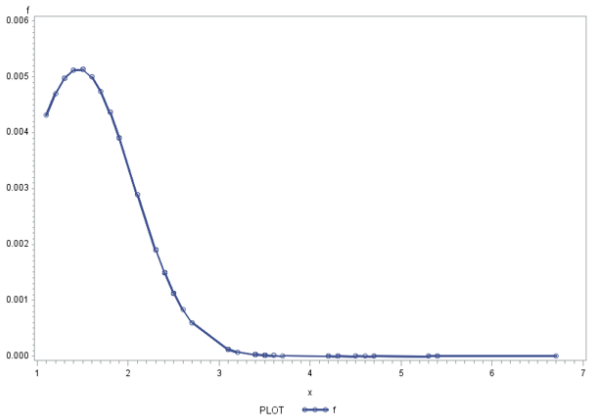


Figure 6: Estimated density function of eruption lengths after implementing the reflection method [1]

5 Conclusion

Kernel density estimation is an easy to use distribution-free approach of estimating the density function of a given data set. The most crucial part of kernel density estimation is the selection of the bandwidth, Silverman's rule of thumb is the widely used method for selecting the bandwidth [5]. A drawback with Kernel density estimation arises when working with a bounded distribution, however boundary correction can be obtained through different approaches. The method of ignoring meaningless estimates can be used if we are only interested in the general shape of the density function, the method entails ignoring all the values outside the boundary condition [4]. The reflection method is applicable only when working with a semi bounded distribution, the reflection method entails reflecting the data first and then working with a new sample of size $2n$ [2]. The transformation method can be used to estimate both semi bounded and bounded distributions, and involves transforming the data set [3]. A disadvantage with some of the methods is that the estimated function is not a true probability density function, however the reflection method has an advantage of generating an estimator that is a true probability density function. Softwares such as SAS can be used to plot the probability density function. PROC IML and PROC KDE in SAS can be used to implement kernel density estimation and for boundary correction.

References

[1] The [Output/code] for this research report was generated SAS software, version [9.3]. Copyright 2015 SAS institute Inc. SAS and all other SAS Institite Inc. product or service names are registered trademarks of SAS Institute Inc., Cary, NC, USA.

[2] M.G Albers. Boundary estimation of densities with bounded support. Master's thesis, Swiss Federal Institute of Technology zurich, 2012.

[3] David Darman. Kernel density estimation for random variables with bounded support. Master's thesis, University of Mayland, College park, 2013.

[4] Bernard W Silverman. *Density Estimation for Statistical and Data Analysis.* Chapman and Hall, 1986.

[5] Matt P Wand and M Chris Jones. *Kernel Smoothing.* Chapman and Hall, 1994.

6 Appendix

6.1 Appendix 1

```
proc iml;
N=500;
call randseed(11119064);
 x = j(N,1);
call randgen(x,"Beta",2,2);
/** create SAS data set from a vector **/
create Patience from x[colname={"random"}];
append from x;
close Patience;
proc print data=Patience;
run;
ods graphics on;
proc kde data=Patience;
   univar random / bwm=0.9 plots=(density histogram histdensity);
run;
ods graphics off;
run;
```

6.2 Appendix 2

```
proc iml;
use marks;
read all into x;
print x; /**Ignoring menaingless values**/
y=x[,1]#(x[,1]<100)#(x[,1]>0);
print y;
N=nrow(y);
min=min(y[,1]);
max=max(y[,1]);
print N min max;
Mean=y[:];
variance=var(y);
stdev=sqrt(variance);
print mean variance stdev;
h1=4*stdev**5;
h2=3*N;
h=(h1/h2)**(0.2);
print h;
do i=1 to N;
pi=3.1415926536;
diff=(y[i,1]-mean);
k=(1/sqrt(2*pi));
constant=(1/(2*(h)));
Wt=k*exp((-constant)*diff##2);
Weight=weight//Wt;
end;
print Weight;
p=nrow(Weight);
print p;
matrix1=y||Weight;
call sort(matrix1,{1});
matrix=matrix1[2:100,1:2];
print matrix;
/** Creating a dataset from a matrix**/
create dataset from matrix[colname={"x" ,"f"}];
append from matrix;
close dataset;
proc print data=dataset;
run;
/**Scatter plot**/
proc sgplot data=dataset;
scatter x=x y=f;
run;
/**Plooting **/
proc gplot data=dataset;
symbol interpol=join width=2 value=circle;
run;
```

6.3 Appendix 3

```
proc iml;
use geyser;
read all into x;
print x;
p=nrow(x);
print p;
/**reflection method**/
y=-x;
c=x//y;
print x y c;
/**reflecting the data back**/
new=c#(c>0);
print new;
j=nrow(new);
print new;
print j;
N=nrow(new);
min=min(new[,1]);
max=max(new[,1]);
print N min max;
Mean=new[:];
variance=var(new);
stdev=sqrt(variance);
print mean variance stdev;
h1=4*stdev**5;
h2=3*N;
h=(h1/h2)**(0.2);
print h;
do i=1 to N;
pi=3.1415926536;
diff=(new[i,1]-mean);
k=(1/sqrt(2*pi));
constant=(1/(2*h**2));
Wt1=k*exp((-constant)*diff##2);
Wt=(1/(N*h))*Wt1;
Weight=weight//Wt;
end;
print Weight;
matrix1=new||Weight;
call sort(matrix1,{1});
matrix=matrix1[111:220,1:2];
print matrix;
/** Creating a dataset from a matrix**/
create dataset from matrix[colname={"x" ,"f"}];
append from matrix;
close dataset;
proc print data=dataset;
run;
/**Scatter plot**/
proc sgplot data=dataset;
 scatter x=x y=f; run;
```

```
/**Plooting **/
proc gplot data=dataset;
symbol interpol=join width=2 value=circle;
run;
```

The hedonic price model

Solly Matlakeng 11315530

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor(s): Dr L Fletcher and Dr E M Louw

Department of Statistics, Univeristy of Pretoria

18 September 2015 (draft 2)

Abstract

The hedonic price model that results from Lancaster's consumer theory and Rosen theoretical model has been applied in various aspects of the housing market. A house as a good can be sold in the market like any other good or service. The price of a house (dependent variable) integrates structural, neighbouring and locational variables (explanatory variables), while environmental factors can also be integrated. Regression model has been used to define the parameters, this will be done after collecting data. The data collected is very large leading to some limitations that will be outlined in the paper. Certain assumptions are made when dealing with the hedonic price model, it operates under the assumption of market equilibrium: the good will be sold to buyers at the highest price they would be prepared to pay and price a seller is prepared to consent with a house being displayed for sale for quite a rational period of time. This research gives a description of the hedonic price model, provides an evaluation on the application of the model, and supplies more literature on the topic. A brief discussion on major empirical issues, inherent limitations and advantages of the hedonic price model is later applied to a practical example.

Declaration

I, *Solly Matlakeng,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_ _
Solly Matlakeng

_ _
Dr L Fletcher and Dr E M Louw

_ _
Date

Acknowledgements

I would like to thank my supervisors Dr L Fletcher and Dr E M Louw for their significant help in writing this research, I am grateful to have you as my supervisors. Sincere gratitude to those who contributed directly and indirectly.

Contents

1 Introduction

The word hedonic is derived from the Greek term hedonikos which means pleasure, defined in economics as the utility derived from buying and using certain goods and services [9]. Research has shown that there is a link between housing sector and wealth of the nation, excessive demand for housing would lead to a major growth in various economic areas. A look into various variables that influence the price of a house is vital because buying a house is an investment decision as well as a consumption decision.

In the housing market two models are utilized to set the price of house, these models are: the monocentric model and the hedonic price model. The monocentric model postulates that the price of a house is merely a function of the workplace . The price of a house shows the relative saving in traveling costs linked with various locations, unlike other consumable goods and services. Housing market is different because it shows the existence of variables of durability, spatial fixity and heterogeneity, thus the hedonic price model is brought to the picture to model the differentiation effectively. The hedonic price model stipulates that goods sold in the market incorporate inherent variables in their package [8], therefore price of one house differs significantly to the other, taking into account the additional units of different variables inherited in one house relative to the other. The process of adding all the implicit prices found after the regression analysis gives the price of a house.

The hedonic price model was developed to get the relationship between variables that are preferred by a consumer and the price of properties. This is due to the fact that the price of a house can be found through an evaluation of the buyers of inherent variables (structural, locational and neighbourhood variables) [4]. The hedonic methods were discovered and applied in price indices before the understanding of their theoretical framework. The first contribution to the hedonic price model theory was made in court back in the year 1941 although there were other informal studies [2].

According to the journal; Haas came up with a hedonic study 15 years before the court and was the one who first published the term hedonic [3].

There are two approaches that add significant contribution to theoretical work on the hedonic price model. These two approaches come from Lancaster's consumer theory and Rosen's model and show how the inherited variables contribute towards the price. The hedonic price model does not postulate joint consumption of goods within one basket, it is also in line with both approaches because Rosen's model is associated with long lasting goods and Lancaster's approach is purposed and good for consumer goods [2].

Furthermore Lancaster's approach postulates a linear relationship between the good's price and the variables that affect it or that it consist of, while Rosen's approach states that unless the consumer can alter the variables by repackaging them, then there exist a nonlinear relationship between the price of the good and the variables [2].

2 The hedonic price model

Statistics is referred to as a science dealing with collection, analysis and interpretation of data. This can be used in many areas to solve problems whereby many variables are being dealt with. In the case of a house because of its nature: fixed location, difference in location and different physical variables, statistics price models have been invented to help in determining the price a house, including regression analysis and the hedonic model [6].

2.1 The regression model

Simple linear regression and multiple regression analysis

Regression analysis is a statistical tool used to determine relationships between variables, normally determining the effect of one or more explanatory variables on the dependent variable, in this case the price of a house. It also determines the statistical significance of the relationships [13]. According to [9] the most basic regression called simple linear model measures the correlation between one dependent variable (Y) and one explanatory variable (X).

To determine such relationships, sufficient current and historical data need to be available. Later in the paper multiple regression will be used to explain whether the explanatory variables add any significant contribution to the price of the house [6].

There are many tangible and intangible explanatory variables to be considered, thus the price of a house can be determined by separating it into different parts. For example, determining the price of a house can be broken down into: neighbourhood, structural and locational.[2].

Separating the variables will also help the investor or the developer that is interested in the variable that mostly affect the price of a house. Previously stated, the price of a house is affected by various variables. Multiple linear regression allows additional variables to be treated individually in the regression with the aim of determining partial regression coefficients. It is vital in quantifying the effect each variable has on a single dependent variable. further, as stated before it can also assist when a developer is only interested in knowing the effect of one of the contributing explanatory variables [10].

Multiple linear regression is capable of incorporating a large number of variables.

2.2 The description of the hedonic price model

2.2.1 Empirical issues

The choice of functional form is the biggest empirical issue concerning the hedonic price model. There exist various basic functional forms that can be used when dealing with the hedonic price model such as linear, semi-log and log-log forms. An inappropriate choice of a functional form may lead to inconsistent estimates. The theory of the hedonic price model gives limited guidance on the matter of choosing a correct functional form even though it has an extensive history [2].

The hedonic price model in theory does not need a division of housing market but in practice there exist different types of divisions due to the fact that the housing markets are not uniform, e.g. the structure of the sub-markets has not been carefully observed when dealing with the hedonic price model even though it is a vital empirical issue [2].

Another issue is the misspecification of the variables, this occurs when irrelevant explanatory variables are taken in account in the model or a case where relevant ones are left out. The problem of misspecification cannot be avoided completely because the model works with implicit prices of the quantities of the variables.

2.2.2 Dependent variable

The hedonic model is an idealistic method of predicting a price of a house (which is our dependent variables) taking into account three main groups of explanatory variables: structural, neighbourhood and locational variables.

2.2.3 Explanatory variables

Physical or structural variables

The price of a house links to the structure of the house. If one house has a desirable feature compared to the next, it will have a higher price as compared to the other. The most important structural variable is the floor space [2]. Buyers are prepared to pay more for space, a house with more rooms and bedrooms will be expensive as compared to the opposite.

In the case where other things stay the same, the price of old houses is less than the initial price due to changes in designs, electrical and mechanical systems that decrease its usefulness. The owner of the house incurs more maintenance and repair costs because the structure of the house deteriorates over the years, leading to a negative relationship between a house age and a house price. According to Li and Brown (1980) the age of a house has an opposite effect on the price of a house. The appreciation is subjected to historical importance or vintage effect of a house [3].

Neighbourhood variables

Neighbourhood variables could be estimated using the hedonic price model even though they cannot explicitly be estimated in the market place. The exclusion of neighbourhood variables in the model will only lead to errors. Studies have shown that higher income households prefer to live in high quality neighbourhoods that are far from central business districts [2].

Neighbourhood variables can be classified into the following categories:

1. Socio-economic variables e.g. social class.
2. Local government or municipal services e.g. schools.
3. Externalities e.g. crime rate.

In the case of the socio-economic variables, research has shown that variables have an effect on the price of the house although there may be other existing elements [2]. Well-off households will likely be sensitive to stay in the neighbourhood that is dominated by working class household owners. They rather buy a house in places that suit their social class.

For local government services the excellence of the public school is considered to have a great effect on a house. School quality is vital to households that have children due to the fact that they want their children to go the best school. This excellence is valued in terms of the pass rate and the school input variables like expenditure per learner [3]. A distance from a house to the hospital also plays a vital role, but can also impact the price of the house negatively depending on the norms of the people.

Externalities like crime negatively impact on the price of the house, people want to live in neighbourhoods that are safe and are not far from the police station. This is true for any individual that is concerned about their safety and that of their family. Externalities are grouped into two groups: positive and negative externalities. Positive externalities or external benefits affect the price of a house positively e.g. pleasant landscape, serenity, quite atmosphere, unpolluted air and the existence of urban forests [3]. Negative externalities could for example be high crime rate or high density housing developments in close proximity.

Locational variables

The location of a house can be viewed in terms of relative and fixed locational variables. The fixed variables are quantified in terms of urban area and relate to some properties from accessibility stand point [2]. These variables are quantified through proxy measures like: racial components, socio-economic economic, pollution levels, etc.

According to the traditional view on location, access to central business districts gives a clear evaluation of accessibility. Transport accessibility is often linked with how easy it is to commute from or to the house. This is done by considering the traveling time, cost of travel, how convenient is it to travel and availability of varies forms of transportation [2]. Properties that are for example close to Gautrain Station or bus station will be considered more costly than those far away from the mentioned.

View is considered as one of the most important things when a consumer buys a house. The consumer prefers locations with good views like lakes or a golf course and are more than prepared to pay more for these views [2]. This variable may not be constant, it differs by types e.g. mountain view, valley view and water view and also by quantity: full view, partial view and poor partial view. These different views add different values to the price of a house; research has shown that there is a strong correlation between floor level and the view due to the fact that higher floors depicts a great view as compared to lower floors [3].

2.3 Advantages and limitations of the hedonic price model

2.3.1 Advantages

The hedonic price model has advantages, the leading one is that only certain information on the price of a house, the components of housing variables and a correct description of the functional relationship are needed [2]. To calculate dependent variable one must estimate the parameters of the hedonic price function. The other advantage of the hedonic price model is versatility, the model can be adopted at ease to take into consideration the numerous interaction between the marketed goods and environmental quality [2]. The method is used to estimate the values of the properties taking into account the choices of consumers. The

last one is that it is easy to gather data on house sales and characteristics of the house that will serve as the explanatory variables for the regression analysis [2].

2.3.2 Limitations

The hedonic price model also has limitations. The amount of data that needs to be gathered is relatively large compared to other models. Applying the model is in actual fact restricted to environmental benefits associated to the price of a house only [1]. The amount of time and expense that will be undertaken to generate an application of the model is affected by the availability and also the accessibility of data. There are also other market limitations; the hedonic price model assumes that consumers have an opportunity to choose the mixture of variables they like [2]. It fails to take into account that the real estate market can be affected by things like taxation rates. Last but not least, the hedonic price model may be difficult to interpret, a high level of statistical knowledge and skills are required.

2.4 Area of application of the hedonic price model

2.4.1 Basic application

The model is mostly applied to the housing market under diverse assumptions made. The first assumption is that the good dealt with is homogeneous . The second assumption is that the market we operating within is under perfect competition, with many buyers and sellers and also buyers being the price takers. The last assumption is that it operates under the assumption of market equilibrium [2]. It is also assumed that data is cross-sectional and therefore excludes the use of time series. The steps taken in applying the hedonic price model are:

Step 1

Collect the data on the house sale for a certain period of time, the preferred period is usually a year [2]. The data comprises of the price of the houses and locations of the houses, house characteristics and neighbourhood characteristics that impact on the price of the house.

Step 2

In this step a regression model is built to define the relationship of the explanatory variables with the price of the house. Each explanatory variable can then be measured to determine whether the predictor is significant or not. Consequently the hedonic price model can be built to determine the price of the house. The results can then be properly interpreted from the regression output.[6].

2.4.2 Specification of the hedonic price model

In section 2.2 dependent and explanatory variables were outlined. Explanatory variables include the structural variables (S), neighbourhood variables (E) and the locational variables (L). Given the above information, the function of the hedonic price model can be formulated as:

$DEPV = B_1 + B_S X_S + B_E X_E + B_L X_L + u$

where:

$DEPV$ is the price value of the house on the market,

X_S is a set of predictor variables of the structural parts,

X_E is a set of predictor variables of the neighbourhood parts,

X_L is a set of predictor variables of the locational parts and

u is a stochastic disturbance term from a classical theoretical regression model.

Other areas of application
Air pollution

We can use the hedonic price model to estimate the effect of air pollution on the value of the property. The coefficients found when building the regression model on air pollution can be used to predict the change in the price of the property Ridker (1967). We can apply this in an area that is close to a dumping site. Based on this alone application of the hedonic price model can be branded under the following categories:

1. Wage-amenity studies- takes into account both the real estate price and wage to determine people inclination to pay for environmental attributes.

2. Housing price- takes into account information on real estate price to determine people inclination to pay for environmental attributes.

3. Wage studies/ value of health risk- takes into account the information on risk premium to determine people inclination to pay in order to avoid health hazards.

Water resources

The model can be used to estimate the value of water resources such as bays, lakes and reservoirs, building of a new harbour, river views and restoration of urban stream. This method can be used to determine the relation between ground water access and land price. An example can be a case of farms, if there is an good abundance of ground water access the land will cost more as compared to that one without an abundance of ground water access [5].

Restaurants

To gain customers and increase sales restaurant must have a proper pricing strategy, the price is a representation for a quality according to consumer's point of view [9]. Studies have shown that it is difficult for a restaurant manager to come up with an effective menu pricing. To eradicate the problem the manager can adopt the hedonic price model to have a better grasp on the pricing factors [11] .

Hospitality industry

The hedonic price model has been adopted by hotels because of different hotel products, services, location and offers. These attributes must be taken into account when determining the correct pricing [1].

2.4.3 Important issues in the application of the hedonic price model

The model is very data rigorous. A large number of observations outlining both the selling price and attributes of the property are required in order to estimate the function of the hedonic price model in a specific market. One of the assumptions made outlined by the hedonic price model is that consumers have perfect information, since it operates in a perfect competition market. If this was not the case the price that they were going to pay for the property was going to vary from sale to sale. Transaction costs are varied and not small in the property market.

Given the current state of the market prices, a household may want to move to a different property with better features and attributes, but if the transaction costs are too high, a household will definitely not move from his or her current place of residence. Therefore the market will stay in the initial equilibrium. Change in demand or supply condition in the housing market does not result in prompt adjustment in the hedonic price model schedule. One problem with the estimation procedure is multicolinearity. Collecting data from more than one market causes severe problems in the hedonic price model analysis e.g. eastern and western suburbs.

All of the listed factors tend to infringe the assumption that we stated before that the property market is in equilibrium [3].

3 A practical application of the hedonic price model

3.1 Problem formulation

This essay investigates the use of Boston housing data in estimating the price of a house. This is done by using 506 census observations [7]. The data shows many problems common to mass appraisal or the hedonic price model despite including numerous vital economic variables.

This essay will allow the use of multiple linear regression analysis to resolve or to correct these issues. The SAS® software will yield the output. Only twelve independent from the Boston housing data (listed in appendix) will be used in estimating the price of a house (dependent variable).

3.2 Descriptive statistics

The Boston housing data were simplified and summarized into a more manageable way, by means of simple statistica and Perason correlation coefficients.

3.2.1 Simple statistics

Simple statistics such as the mean, median and standard deviation of the dependent variable and the 12 independent were obtained using the means procedure in SAS software.Table1 of the appendix shows basic statistics such as the and median means of all 13 variables accompanied by standard deviations, minimum and maximum. The standard deviation in any regards gives a notion of the closeness of the data set to the average value.

3.2.2 Correlation matrix

The correlation matrix is highly informative, as it describes the correlation of linear relationship between two variables. Table2 of the appendix shows a high correlation of 0.9 between two of the independent variables namely RAD and TAX. These two independent variables should be investigated for multicollinearity.

3.3 The hedonic price model

The model building process consisted of three steps.

3.3.1 Step 1: diagnostics for multicollinearity

In this Section the 12 independent variables were included into a regression model and the variance inflation factor (VIF) option was incorporated to help detect multicollinearity. Variance inflation factors measure the degree at which the variance in the dependent variable MEDV is inflated [2]. Looking at the output in Table3 the variance inflation factor of RAD and TAX are very high: 7.44530 and 9.00216 respectively. A variance inflation factor higher than 5 is an indication of serious multicollinearity. Thus milticollinearity was dectected in both both RAD and TAX and it was decided to drop TAX.

3.3.2 Step 2: step-wise selection of significant predictors

In a step-wise regression selection procedure of significant predictors incorporated in a regression model, the selection of significant predictors is done automatically. The next step in this essay was to use a step-wise selection procedure using 11 variables excluding TAX. Output was generated (Table4) showing only nine significant independent variable of predictors. The output is shown in Table5.

3.3.3 Step 3: Hedonic regression model

The final step in the analyses entailed building a regression model using the reg procedure of SAS software on the nine significant predictors identified by the step-wise procedure.

Significance of the model

The regression model has a p-value $<.0001$ which implies that the model is highly significant.

R-squared and Adjusted R-squared

R-squared$= 0.7273$ which implies that approximately 73% of the variation in MEDV is explained by the nine explanatory variables. The Adjusted R-squared is 0.7223, which is almost identical to the R-squared.

Regression equation

$MEDV = B_1 + B_S X_S + B_E X_E + B_L X_L + u$

$= 39.9840 + (3.85RM + 0.04ZN) + (-0.11CRIM - 1.00PTRATIO - 21.38NOX) + (-1.45DIS - 0.55LSTAT + 0.10RAD + 3.14CHAS)$

where

$B_1 = 39.984$

$B_S = a$ compiled coefficient of the structural variable X_S

$B_E =$a compiled coefficient of the neighbourhood variable X_E

$B_L =$a compiled coefficient of the locational variable X_L

Relative importance of predictors

The relative importance of the predictors in the regression equation are evaluated by using standardized coefficients in Table5. The five most important predictors are LSTAT, DIS, NOX, PTRATIO and CRIM.

4 Conclusion

The hedonic price model gives a scientific way of determining the value of the property where other models have failed in integrating the attributes that are pleasurable in determining the value of the property. Outlining the attributes that are important in analyzing can bring the value of the property up to profit the seller. Certain problems will be encountered when building the model due to the fact that the data collected is relatively high. The ultimate price of the property can be affected by negative externalities such as: its closeness to central business district or a busy road [6].

The information provided by this model can be useful to investment fund managers and also real estate owners. The hedonic price model has also been applied in the automobile industry, hospitality industry and restaurants because it explains contributions of various factors to pricing, thus making it simple to tell which attributes are valued more or less and to what degree [2].

References

[1] Adrian O Bull. Pricing a motel's location. *International Journal of Contemporary Hospitality Management*, 6(6):10–15, 1994.

[2] KW Chau and TL Chin. A critical review of literature on the hedonic price model. *International Journal for Housing Science and Its Applications*, 27(2):145–165, 2003.

[3] David E Clark and William E Herrin. Historical preservation districts and home sale prices: evidence from the sacramento housing market. *The Review of regional studies*, 27(1):29–48, 1997.

[4] A Myrick Freeman III. Hedonic prices, property values and measuring environmental benefits: a survey of the issues. *The Scandinavian Journal of Economics*, pages 154–173, 1979.

[5] Haripriya Gundimeda. Hedonic price method–a concept note. *TProject report submitted to South Asian network for economic institution*, 2004.

[6] Matt Monson. Valuation using hedonic pricing models. *Cornell Real Estate Review*, 7(1):10, 2009.

[7] Ronald G Ridker and John A Henning. The determinants of residential property values with special reference to air pollution. *The Review of Economics and Statistics*, pages 246–257, 1967.

[8] Sherwin Rosen. Hedonic prices and implicit markets: product differentiation in pure competition. *The journal of political economy*, pages 34–55, 1974.

[9] Stowe Shoemaker, Mary Dawson, and Wade Johnson. How to increase menu prices without alienating your customers. *International Journal of Contemporary Hospitality Management*, 17(7):553–568, 2005.

[10] Alan O Sykes. An introduction to regression analysis. 1993.

[11] Eun Soon Yim, Suna Lee, and Woo Gon Kim. Determinants of a restaurant average meal price: An application of the hedonic pricing model. *International Journal of Hospitality Management*, 39:11–20, 2014.

Appendix

1. **Description of variables**

Dependent variable			
Variables	**Description**	**Categorical**	**Continuous**
MEDV	median value of owner-occupied homes in USD 1000's		X
Data outlay of explanatory variables			
Structural variables			
Variables	**Description**	**Categorical**	**Continuous**
RM	Average number of rooms per dwelling		X
ZN	Proportion of residential land zoned for lots over 25,000 sq.ft.		X
AGE	Proportion of structure built before 1940		X
INDUS	Proportion of non-retail business acres per town		X
Neighbourhood variables			
Variables	**Description**	**Categorical**	**Continuous**
CRIM	Per capita crime rate by town		X
PTRATIO	Pupil-teacher ratio by town		X
TAX	Full-value house-tax rate per $10,000		X
NOX	Nitric oxides concentration (parts per 10 million)		X
Location variable			
Variables	**Description**	**Categorical**	**Continuous**
DIS	Weighted distances to five Boston employment centres		X
LSTAT	% lower status of the population		X
RAD	Index of accessibility to radial highways		
CHAS	Charlies river dummy variable	X 1= tract bounds river 0= otherwise	

2. **SAS Program**

```
data houses;
input MEDV RM ZN AGE INDUS CRIM PTRATIO TAX NOX DIS LSTAT RAD CHAS;

Datalines;
...
;
```

```
Proc means data=houses median std min max;
var MEDV RM ZN AGE INDUS CRIM PTRATIO TAX NOX DIS LSTAT RAD CHAS;
run;

Proc corr data=houses;
var MEDV RM ZN AGE INDUS CRIM PTRATIO TAX NOX DIS LSTAT RAD CHAS;
run;

Proc reg data=houses;
model MEDV=RM ZN AGE INDUS CRIM PTRATIO TAX NOX DIS LSTAT RAD CHAS /VIF;
run;

Proc reg data=houses;
model MEDV=RM ZN AGE INDUS CRIM PTRATIO NOX DIS LSTAT RAD CHAS / selection=stepwise;
run;

Proc reg data=houses;
 model MEDV=RM ZN CRIM PTRATIO NOX DIS LSTAT RAD CHAS / stb;
run;
```

3. **SAS output**

   (a) **Descriptive statistics** (Table1)

   (b) **Correlation matrix** (Table 2)

   (c) **The hedonic price model**
   Step 1: Diagnostics for multicollinearity (Table 3)

   Step 2: Step-wise selection (Table 4)

   Step 3: Hedonic regression model (Table 5)

The SAS System

Variable	Mean	Median	Std Dev	Minimum	Maximum
MEDV	22.5328063	21.2000000	9.1971041	5.0000000	50.0000000
RM	6.2846344	6.2085000	0.7026171	3.5610000	8.7800000
ZN	11.3636364	0	23.3224530	0	100.0000000
AGE	68.5749012	77.5000000	28.1488614	2.9000000	100.0000000
INDUS	11.1367787	9.6900000	6.8603529	0.4600000	27.7400000
CRIM	3.6135236	0.2565100	8.6015451	0.0063200	88.9762000
PTRATIO	18.4555336	19.0500000	2.1649455	12.6000000	22.0000000
TAX	408.2371542	330.0000000	168.5371161	187.0000000	711.0000000
NOX	0.5546951	0.5380000	0.1158777	0.3850000	0.8710000
DIS	3.7950427	3.2074500	2.1057101	1.1296000	12.1265000
LSTAT	12.6530632	11.3600000	7.1410615	1.7300000	37.9700000
RAD	9.5494071	5.0000000	8.7072594	1.0000000	24.0000000
CHAS	0.0691700	0	0.2539940	0	1.0000000

Table 1: Simple descriptive statistics

Pearson Correlation Coefficients, N = 506
Prob > |r| under H0: Rho=0

	MEDV	RM	ZN	AGE	INDUS	CRIM	PTRATIO	TAX	NOX	DIS	LSTAT	RAD	CHAS
MEDV	1.00000	0.69536	0.36045	-0.37695	-0.48373	-0.38830	-0.50779	-0.46854	-0.42732	0.24993	-0.73766	-0.38163	0.17526
		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001
RM	0.69536	1.00000	0.31199	-0.24026	-0.39168	-0.21925	-0.35550	-0.29205	-0.30219	0.20525	-0.61381	-0.20985	0.09125
	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0402
ZN	0.36045	0.31199	1.00000	-0.56954	-0.53383	-0.20047	-0.39168	-0.31456	-0.51660	0.66441	-0.41299	-0.31195	-0.04270
	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.3378
AGE	-0.37695	-0.24026	-0.56954	1.00000	0.64478	0.35273	0.26152	0.50646	0.73147	-0.74788	0.60234	0.45602	0.08652
	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.0518
INDUS	-0.48373	-0.39168	-0.53383	0.64478	1.00000	0.40658	0.38325	0.72076	0.76365	-0.70803	0.60380	0.59513	0.06294
	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.1575
CRIM	-0.38830	-0.21925	-0.20047	0.35273	0.40658	1.00000	0.28995	0.58276	0.42097	-0.37967	0.45562	0.62551	-0.05589
	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	0.2094
PTRATIO	-0.50779	-0.35550	-0.39168	0.26152	0.38325	0.28995	1.00000	0.46085	0.18893	-0.23247	0.37404	0.46474	-0.12152
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	<.0001	0.0062
TAX	-0.46854	-0.29205	-0.31456	0.50646	0.72076	0.58276	0.46085	1.00000	0.66802	-0.53443	0.54399	0.91023	-0.03559
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	<.0001	0.4244
NOX	-0.42732	-0.30219	-0.51660	0.73147	0.76365	0.42097	0.18893	0.66802	1.00000	-0.76923	0.59088	0.61144	0.09120
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	<.0001	0.0403
DIS	0.24993	0.20525	0.66441	-0.74788	-0.70803	-0.37967	-0.23247	-0.53443	-0.76923	1.00000	-0.49700	-0.49459	-0.09918
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	<.0001	0.0257
LSTAT	-0.73766	-0.61381	-0.41299	0.60234	0.60380	0.45562	0.37404	0.54399	0.59088	-0.49700	1.00000	0.48868	-0.05393
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		<.0001	0.2259
RAD	-0.38163	-0.20985	-0.31195	0.45602	0.59513	0.62551	0.46474	0.91023	0.61144	-0.49459	0.48868	1.00000	-0.00737
	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001	<.0001		0.8687
CHAS	0.17526	0.09125	-0.04270	0.08652	0.06294	-0.05589	-0.12152	-0.03559	0.09120	-0.09918	-0.05393	-0.00737	1.00000
	<.0001	0.0402	0.3378	0.0518	0.1575	0.2094	0.0062	0.4244	0.0403	0.0257	0.2259	0.8687	

Table 2: Correlation matrix

The SAS System 6

The REG Procedure
Model: MODEL1
Dependent Variable: MEDV

Number of Observations Read 506
Number of Observations Used 506

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	12	31367	2613.90638	113.54	<.0001
Error	493	11349	23.02113		
Corrected Total	505	42716			

Root MSE	4.79803	R-Square	0.7343
Dependent Mean	22.53281	Adj R-Sq	0.7278
Coeff Var	21.29355		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Variance Inflation
Intercept	1	41.61727	4.93604	8.43	<.0001	0
RM	1	3.65812	0.42025	8.70	<.0001	1.91253
ZN	1	0.04696	0.01388	3.38	0.0008	2.29846
AGE	1	0.00361	0.01333	0.27	0.7866	3.08823
INDUS	1	0.01347	0.06214	0.22	0.8285	3.98718
CRIM	1	-0.12139	0.03300	-3.68	0.0003	1.76749
PTRATIO	1	-0.93753	0.13221	-7.09	<.0001	1.79706
TAX	1	-0.01268	0.00380	-3.34	0.0009	9.00216
NOX	1	-18.75802	3.85135	-4.87	<.0001	4.36909
DIS	1	-1.49075	0.20162	-7.39	<.0001	3.95404
LSTAT	1	-0.55202	0.05066	-10.90	<.0001	2.87078
RAD	1	0.28940	0.06691	4.33	<.0001	7.44530
CHAS	1	2.83999	0.87001	3.26	0.0012	1.07117

Table 3: Diagnostics for multicollinearity

The REG Procedure
Model: MODEL1
Dependent Variable: MEDV

Stepwise Selection: Step 9

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	31066	3451.72624	146.95	<.0001
Error	496	11651	23.48943		
Corrected Total	505	42716			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	39.98405	4.94523	1535.58247	65.37	<.0001
RM	3.85056	0.41105	2061.29134	87.75	<.0001
ZN	0.03658	0.01357	170.59893	7.26	0.0073
CRIM	-0.11854	0.03330	297.59617	12.67	0.0004
PTRATIO	-1.00175	0.13049	1384.30203	58.93	<.0001
NOX	-21.37566	3.50093	875.68091	37.28	<.0001
DIS	-1.45079	0.18905	1383.28516	58.89	<.0001
LSTAT	-0.55346	0.04797	3126.30726	133.09	<.0001
RAD	0.10457	0.04071	155.00566	6.60	0.0105
CHAS	3.13944	0.86975	306.04911	13.03	0.0003

Bounds on condition number: 3.5382, 185.82

All variables left in the model are significant at the 0.1500 level.

No other variable met the 0.1500 significance level for entry into the model.

Summary of Stepwise Selection

Step	Variable Entered	Variable Removed	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	LSTAT		1	0.5441	0.5441	326.848	601.62	<.0001
2	RM		2	0.0944	0.6386	157.179	131.39	<.0001
3	PTRATIO		3	0.0401	0.6786	86.3359	62.58	<.0001
4	DIS		4	0.0117	0.6903	67.0925	18.90	<.0001
5	NOX		5	0.0178	0.7081	36.7614	30.46	<.0001
6	CHAS		6	0.0077	0.7158	24.7885	13.49	0.0003
7	ZN		7	0.0038	0.7196	19.7905	6.84	0.0092
8	CRIM		8	0.0040	0.7236	14.5160	7.19	0.0076
9	RAD		9	0.0036	0.7273	9.9181	6.60	0.0105

Table 4: Step-wise selection of significant predictors

The SAS System

The REG Procedure
Model: MODEL1
Dependent Variable: MEDV

Number of Observations Read	506
Number of Observations Used	506

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	31066	3451.72624	146.95	<.0001
Error	496	11651	23.48943		
Corrected Total	505	42716			

Root MSE	4.84659	R-Square	0.7273
Dependent Mean	22.53281	Adj R-Sq	0.7223
Coeff Var	21.50904		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Standardized Estimate
Intercept	1	39.98405	4.94523	8.09	<.0001	0
RM	1	3.85056	0.41105	9.37	<.0001	0.29417
ZN	1	0.03658	0.01357	2.69	0.0073	0.09275
CRIM	1	-0.11854	0.03330	-3.56	0.0004	-0.11087
PTRATIO	1	-1.00175	0.13049	-7.68	<.0001	-0.23581
NOX	1	-21.37566	3.50093	-6.11	<.0001	-0.26932
DIS	1	-1.45079	0.18905	-7.67	<.0001	-0.33216
LSTAT	1	-0.55346	0.04797	-11.54	<.0001	-0.42973
RAD	1	0.10457	0.04071	2.57	0.0105	0.09900
CHAS	1	3.13944	0.86975	3.61	0.0003	0.08670

Table 5: Hedonic regression model

The optimal scaling technique for the analysis of multivariate categorical data

Lesedi Matshehla 11029189

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr HF Strydom

Department of Statistics, University of Pretoria

2 November 2015 (final)

Abstract

This paper gives an overview of optimal scaling which is a special case of correspondence analysis to obtain scale values that can be used in subsequent analyses as a quantitative dependent variable [2, 3]. Two methods will be used to do the optimal scaling. The first method maximizes the variance of the total scores between the groups by forming groups that consists of individuals that respond in the same way within the group, but differently between the groups [2].The second method performs optimal scaling in such a way that the total scores vary maximally over individuals [2].

Declaration

I, *Lesedi Matshehla,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics ,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Lesedi Kopeledi Matshehla

_____
Dr HF Strydom

_____
Date

Acknowledgements

I would like to offer my special thanks to Dr HF Strydom for her valuable suggestions during the planning and development of this research report. I appreciate her willingness to give her time so generously.
I would also like to express my appreciation to Dr I Fabris-Rotelli and MT Loots for comments that greatly improved my research report.

Contents

List of Figures

List of Tables

1 Introduction

Relevant information on a population group can be obtained by means of a questionnaire that is completed by the respondents. In most cases the respondents' answers are of a categorical nature. This means that the data obtained by means of a questionnaire can be represented in contingency table form or in a cross-tabulation [3]. As questionnaires in real life research often contain different themes that consists of many items with category choices, the contingency table will have many rows and columns. Therefore, it will be difficult to analyse the contingency table with a simple graphical representation such as a scatter plot since involving many categories would result in a confusion of points and symbols that it will not be easy to see any patterns at all [2]. Therefore, it is important to introduce a statistical technique called correspondence analysis which is suitable for the analysis of contingency tables with large data sets or for data matrices which have non-negative values [2] .
The categorical data obtained by means of a questionnaire can be converted to continuous data by awarding quantitative values to the diverse categories. The technique to obtain these quantitative values is known as optimal scaling, which is equivalent to performing a correspondence analysis on the contingency table [5]. The following table shows the importance of optimal scaling for the analysis of multivariate categorical data. For example, suppose that the variables region and job are coded as shown in the following table.

Region code	Region value	Job code	Job value
1	North	1	Intern
2	South	2	Sales rep
3	East	3	Manager
4	West		

Table 1: Coding scheme for region and job variables

From the table we see that job value is an ordinal categorical variable. The categories of job variable are ordered (i.e. the original categories form a progression from intern to manager), which indeed gives some justification for using the integer codes 1 to 3. Region, on the other hand, is a nominal variable. The categories of the region variable are not ordered. Integer codes 1 to 4 simply represent the four categories; not the order of the categories. Failing any alternative, when categories are not ordered as in region variable, the integer codes (1 to 4) are often used as default values in statistical calculations such as mean and variance [3, 2]. Optimal scaling provides a way of obtaining quantitative values for a multivariate categorical variable based on a specified criterion of optimality [2]. Unlike the original labels (i.e. region codes 1 to 4 and job codes 1 to 3) of the nominal or ordinal variables in the statistical analysis, the scale values determined by an optimal scaling technique define a distance between each pair of categories of a multivariate categorical data set [2].
The optimal scale values are used to calculate a total score which varies maximally between the groups and between the respondents [2]. This total score is then used in subsequent analyses like ANOVA and regression analysis [3]. Before we discuss optimal scaling in detail, it is very important to discuss a statistical technique called correspondence analysis since optimal scaling is a special case of correspondence analysis [3].

2 Correspondence analysis

Correspondence analysis is a technique appropriate for the analysis of contingency tables or for data matrices with non-negative values. It is used to analyse a structure of two-way and higher way contingency tables and it treats the rows and columns of contingency table symmetrically [3, 1]. Correspondence analysis allows us to explore the nature of an association between two or more categorical variables. In fact it is assumed in correspondence analysis that the rows and columns are dependent, and the analysis is concerned with displaying this dependence. [2]

An example of a contingency table generated from the database of the Spanish National Health Survey in 1997 is given in Table 2. A question in this survey is about the judgement that individuals have of their own health, which they can judge to be "very bad", "bad", "regular", "good" or "very good" [2]. Since the respondents' answers to this question are of a categorical nature, it is possible for the data to be presented in contingency table form. The contingency table tabulates these answers with the age groups of the respondents. There are five health categories (columns of Table 2) and seven age groups (rows of Table 2). A total number of 6371 respondents are tabulated and give a symbolic image of how the Spanish nation views its own health at this point in time.

	A	B	C	D	E	F	G
1	Age group	Very good	good	Regular	Bad	Very bad	sum
2	16-24	243	789	167	18	6	1223
3	25-34	220	809	164	35	6	1234
4	35-44	147	658	181	41	8	1035
5	45-54	90	469	236	50	16	861
6	55-64	53	414	306	106	30	909
7	65-74	44	267	284	98	20	713
8	75+	20	136	157	66	17	396
9	sum	817	3542	1495	414	103	6371

Table 2: Cross-tabulation of age group with self-perceived health category [2]

2.1 Matrix required for correspondence analysis

Suppose that the contingency table is described formally by the $I \times J$ matrix $\mathbf{F} = [f_{ij}]$; this contingency table is an $(I \times J)$ rank q matrix of nonnegative values with nonzero row and column sums [3]. We obtain correspondence matrix $\mathbf{S}$ from $\mathbf{F}$ by dividing its entries by their grand total $f_1 = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij}$ [3]:

$$\mathbf{S} = [s_{ij}] = \left[\frac{f_{ij}}{f}\right] \tag{1}$$

Note that all the matrices are denoted by bold uppercase letters, vectors are denoted by bold lowercase letters, and their elements are denoted by italic lowercase letters. The transpose operation of any matrix is denoted by the superscript^t or backtick punctuation mark (‘); the inverse operation of any matrix is denoted

by $^{-1}$. Let $\mathbf{1}$ be a column vector of ones of the appropriate order, let $\mathbf{I}$ represents an identity matrix, let diag(.) be a function that creates a diagonal matrix from a vector [3].
Let

$$\begin{aligned} f_1 &= \sum_{i=1}^{I}\sum_{j=1}^{J} f_{ij} \\ &= \mathbf{1}^t \times \mathbf{F} \times \mathbf{1} \\ \mathbf{S} &= \frac{\mathbf{F}}{f_1} \\ \mathbf{r} &= \mathbf{S} \times \mathbf{1} \\ \mathbf{c} &= \mathbf{S^t} \times \mathbf{1} \\ \mathbf{D_r} &= diag(\mathbf{r}) \\ \mathbf{D_c} &= diag(\mathbf{c}) \\ \mathbf{R_1} &= \mathbf{D_r}^{-1} \times \mathbf{S} \\ \mathbf{C_1} &= \mathbf{D_c}^{-1} \times \mathbf{S}^t \end{aligned}$$

The scalarf_1 is the sum of all elements in the contingency table $\mathbf{F}$. The matrix $\mathbf{S}$ is the correspondence matrix of relative frequencies. The vector $\mathbf{r}$ is the vector of row totals of $\mathbf{S}$ and vector $\mathbf{c}$ is the vector of column totals of $\mathbf{S}$. The matrices $\mathbf{D_r}$ and $\mathbf{D_c}$ indicate the diagonal matrices of vectors $\mathbf{r}$ and $\mathbf{c}$ respectively [3].
The rows of matrix $\mathbf{R_1}$ provides the row profiles and the elements of each row of $\mathbf{R_1}$ sum to one. The columns of matrix $\mathbf{C_1}$ provides the column profiles and the elements of each column of $\mathbf{C_1}$ sum to one.
In correspondence analysis, the column and row coordinates are based on the generalized singular value decomposition of $\mathbf{S}$ [3, 1],

$$\mathbf{S} = \mathbf{A}\mathbf{D_u}\mathbf{B^t} \tag{2}$$

where matrix $\mathbf{A}$ is a $I \times q$ rectangular matrix of left generalized singular vectors, the matrix $\mathbf{B}$ is a $J \times q$ rectangular matrix of right generalized singular vectors and the matrix $\mathbf{D_u}$ is a $q \times q$ diagonal matrix of singular values.
The generalized singular value decomposition of $\mathbf{S–rc^t}$ can be derived from the ordinary singular value decomposition of $\mathbf{D_r}^{-\frac{1}{2}}(\mathbf{S–rc^t})\mathbf{D_c}^{-\frac{1}{2}}$[3, 1]:

$$\begin{aligned} \mathbf{D_r}^{-\frac{1}{2}}(\mathbf{S–rc^t})\mathbf{D_c}^{-\frac{1}{2}} &= \mathbf{U}\mathbf{D_u}\mathbf{V^t} \\ &= (\mathbf{D_r}^{-\frac{1}{2}}\mathbf{A})\mathbf{D_u}(\mathbf{D_c}^{-\frac{1}{2}}\mathbf{B})^{\mathbf{t}} \end{aligned}$$

$$\begin{aligned} \mathbf{S–rc^t} &= \mathbf{D_r}^{\frac{1}{2}}\mathbf{U}\mathbf{D_u}\mathbf{V^t}\mathbf{D_c}^{\frac{1}{2}} \\ &= (\mathbf{D_r}^{\frac{1}{2}}\mathbf{U})\mathbf{D_u}(\mathbf{D_c}^{\frac{1}{2}}\mathbf{V})^t \\ &= \mathbf{A}\mathbf{D_u}\mathbf{B^t} \end{aligned}$$

Hence $\mathbf{A} = \mathbf{D_r}^{\frac{1}{2}}\mathbf{U}$ and $\mathbf{B} = \mathbf{D_c}^{\frac{1}{2}}\mathbf{V}$
The matrices

$$\mathbf{K} = \mathbf{D_r^{-1}AD_u} \tag{3}$$

and

$$\mathbf{L} = \mathbf{D_c^{-1}BD_u} \tag{4}$$

give the default coordinates of rows and columns respectively.
Correspondence analysis plots the first two columns of matrices **K** and **L** to display graphical associations between the row and column categories.

3 Correspondence analysis application

In this section, we examine a data set from the issue of Amstat News published in October 2005 by Luo and Keyes that would be used throughout this paper. The data set gives the results of a survey of ASA members with $6-15$ years of membership [4]. In this survey the questionnaire consisted of various questions and the purpose of one of the questions was to obtain the opinion that respondents have on their primary position. The participants were asked to state whether they agree that their primary position is professionally challenging [4]. All the respondents were asked to rank their agreement with survey items on a scale that includes strongly agree, agree, no opinion, disagree and strongly disagree [4].
Since the respondents' answers to this question are of a categorical nature, it is possible to tabulate the data set in contingency table shown in Table 3. There are five likert scale categories (columns of Table 3) and five employment sector groups (rows of Table 3).
The SAS program CORRESP procedure (Appendix 1) is used to analyze the cross-tabulation in Table 3.

Contingency Table

	Strongly_agree	Agree	No_opinion	Disagree	Strongly_disagree	Sum
Academic (nonstudent)	162	78	8	5	0	253
Business and industry	72	88	5	11	0	176
Federal,state and local government	27	34	5	3	2	71
Private consultant/self-employment	11	15	2	0	0	28
other(retired,students,unemployed)	10	15	5	2	2	34
Sum	282	230	25	21	4	562

Table 3: Cross-tabulation of employment sector and opinion on primary position

3.1 Calculation of the matrices required for correspondence analysis

The correspondence matrix of the contingency table given in Table 4 is defined as the ratio of the elements in the contingency table, **F**, to the sum of all elements in contingency table **F** [1].

- Let $f_1 = \sum_{i=1}^{I} \sum_{j=1}^{J} f_{ij} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix} \times \mathbf{F} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ be the sum of all elements in contingency table $\mathbf{F}$.
- Let $\mathbf{S} = \frac{\mathbf{F}}{f_1}$ be the correspondence matrix of relative frequencies of the contingency table $\mathbf{F}$.
- Let $J(5,1,1)$ be a matrix function that creates 5×1 column vector of ones, that is, $\mathbf{1} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$.

Table 4 shows a correspondence matrix of associated proportions i.e. relative frequencies of the contingency table given in Table 3. This correspondence matrix is calculated using a SAS program (IML procedure) and is given in the appendix.

Employment_sector_groups	Correspondence_matrix				
Academic (nonstudent)	0.2882562	0.13879	0.0142349	0.0088968	0
Business and industry	0.1281139	0.1565836	0.0088968	0.019573	0
Federal,state and local government	0.0480427	0.0604982	0.0088968	0.0053381	0.0035587
Private consultant/self-employment	0.019573	0.0266904	0.0035587	0	0
other(retired,students,unemployed)	0.0177936	0.0266904	0.0088968	0.0035587	0.0035587

Table 4: Correspondence matrix of relative frequencies [2]

3.1.1 Row and column operations

For the operations that follow, we make use of 5×5 correspondence matrix of relative frequencies defined above. The formulas given in section 2.1 will now be applied.

Row and column sums of correspondence matrix S

$$\begin{aligned}\mathbf{r} &= \mathbf{S} \times \mathbf{1} \\ &= \mathbf{S} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 0.4501779 \\ 0.3131673 \\ 0.1263345 \\ 0.0498221 \\ 0.0604982 \end{bmatrix}\end{aligned}$$

$$\begin{aligned}\mathbf{c} &= \mathbf{S^t} \times \mathbf{1} \\ &= \mathbf{S^t} \times \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 0.5017794 \\ 0.4092527 \\ 0.044484 \\ 0.0373665 \\ 0.0071174 \end{bmatrix}\end{aligned}$$

Where row vector **r** and column vector **c** are calculated by the program IML procedure (Appendix).

Row and column diagonals

From previous section we know that a matrix function diag(.) creates a diagonal matrix from a vector. Now we use SAS program (IML procedure) given in the appendix to calculate the diagonal matrix of row vector**r** and column vector**c** denoted by $\mathbf{D_r}$ and $\mathbf{D_c}$ respectively.

$$\begin{aligned}\mathbf{D_r} &= diag(\mathbf{r}) \\ &= \begin{bmatrix} 0.4501779 & 0 & 0 & 0 & 0 \\ 0 & 0.3131673 & 0 & 0 & 0 \\ 0 & 0 & 0.1263345 & 0 & 0 \\ 0 & 0 & 0 & 0.0498221 & 0 \\ 0 & 0 & 0 & 0 & 0.0604982 \end{bmatrix}\end{aligned}$$

$$\begin{aligned}\mathbf{D_c} &= diag(\mathbf{c}) \\ &= \begin{bmatrix} 0.5017794 & 0 & 0 & 0 & 0 \\ 0 & 0.4092527 & 0 & 0 & 0 \\ 0 & 0 & 0.044484 & 0 & 0 \\ 0 & 0 & 0 & 0.0373665 & 0 \\ 0 & 0 & 0 & 0 & 0.0071174 \end{bmatrix}\end{aligned}$$

Row and column profiles

- Let $\mathbf{R_1} = \mathbf{D_r^{-1}} \times \mathbf{S}$

- Let $\mathbf{C_1} = \mathbf{D_c^{-1}} \times \mathbf{S^t}$

where the matrix $\mathbf{R_1}$ provides the row profiles in Table 5 and $\mathbf{C_1}$ provides the column profiles in Table 6. The last row in Table 5, labeled Average, is the profile of the last row [282 230 25 21 4] of Table 3, which contains the sum of the columns of the contingency table without distinguishing between the employment sector groups. We can clearly see from Table 5 that out of a total of 562 ASA members sampled in this survey, 50.178% respondents **strongly agreed** that their primary position is professionally challenging, 40.925% respondents **agreed**, 4.448% respondents had **no opinion**, and so on [4]. Looking again at the elements of Table 5 we can see that the values of all employment sector groups decrease as we go across the table from left to right. This means that most of the respondents **strongly agreed** or **agreed** that their primary position is professionally challenging [2]. But the percentage of their agreement differs. For example, the **Academic (nonstudent)** group has a high percentage in the **strongly agree** category (64.032%) and the **Other (retired, students, unemployed)** group has a lower percentage in the **strongly agree** category (29.412%).
When comparing the profiles we can compare each employment sector group's profile with the average profile, or we can compare one employment sector group's profile with another. For example, when we look at the values in Table 5, we see that the average **Academic (nonstudent)** group has a higher percentage in the **strongly agree** category and is below average for all the remaining four categories. In addition, Table 5 shows that all elements of the row profiles add up to 1, which corresponds with the theory of relative frequencies since we know that all relative frequency values add up to 1.

Employment_sector_groups Col1	Row_profile Col2	Col3	Col4	Col5	Col6	SUM Col7
Academic (nonstudent)	0.6403162	0.3083004	0.0316206	0.0197628	0	1
Business and industry	0.4090909	0.5	0.0284091	0.0625	0	1
Federal,state and local government	0.3802817	0.4788732	0.0704225	0.0422535	0.028169	1
Private consultant/self-employment	0.3928571	0.5357143	0.0714286	0	0	1
other(retired,students,unemployed)	0.2941176	0.4411765	0.1470588	0.0588235	0.0588235	1
AVERAGE	0.5017794	0.4092527	0.044484	0.0373665	0.0071174	1

Table 5: Row profiles of employment sector groups across the likert scale categories, expressed as relative frequencies

Table 6 shows column profiles and the average column profile labeled Average_c. The Average_c is the profile of the last column [253 176 71 28 34]' of Table 3, which contain the sum of each employment sector group; in other words this is the profile of all the likert scale categories put together [2]. By eyeballing the values in Table 6 we realize that out of a total of 562 ASA members sampled in this survey, 45.018% are **Academic (nonstudent)** respondents, 31.317% are **Business and industry** respondents, 12.633% are **Federal, state and local government** respondents, and so on. In addition, of the 282 respondents who strongly agreed, 57.4% are **Academic (nonstudent)** respondents, 25.5% are **Business and industry** respondents, 9.6% are **Federal, state and local government** respondents, and so on. Since the number of respondents who strongly agreed is different in each employment sector group, these values should be compared to those of the average column profile to see if they are higher or lower than the average column profile [2]. For example, 57.4% of the respondents who strongly agreed that their primary position is professionally challenging are **Academic (nonstudent)**, whereas the number of respondents in **Academic (nonstudent)** group is just 45.0% out of a total of 562 ASA members. This implies that there is a high

number of individuals who strongly agreed that their primary position is professionally challenging compared to the average.

Employment_sector_groups Col1	Column_profile Col2	Col3	Col4	Col5	Col6	Average_c Col7
Academic (nonstudent)	0.5744681	0.3391304	0.32	0.2380952	0	0.4501779
Business and industry	0.2553191	0.3826087	0.2	0.5238095	0	0.3131673
Federal,state and local government	0.0957447	0.1478261	0.2	0.1428571	0.5	0.1263345
Private consultant/self-employment	0.0390071	0.0652174	0.08	0	0	0.0498221
other(retired,students,unemployed)	0.035461	0.0652174	0.2	0.0952381	0.5	0.0604982
SUM	1	1	1	1	1	1

Table 6: Column profiles of likert scale categories across the employment sector groups, expressed as relative frequencies

Please refer to the Appendix to see how row and column profiles are computed using a SAS program (IML procedure).

Row and column coordinates

As already discussed above, the row and column coordinates in correspondence analysis are based on the generalized singular value decomposition of the correspondence matrix $\mathbf{S}$ [1]. If we refer to the output of the CORRESP procedure program in Appendix, we see that only the first two columns of the column coordinates and the row coordinates are displayed . This is because correspondence analysis displays graphical associations between row and column categories by plotting the first two columns of the column coordinates and the row coordinates [3, 1].

Formula **3** is used to calculate the row coordinates:

$$\mathbf{K} = \mathbf{D_r^{-1}AD_u} = \begin{bmatrix} -0.2647 & 0.1104 & -0.0063 & 0.0014 & 0.0000 \\ 0.0810 & -0.2431 & -0.0363 & 0.0088 & 0.0000 \\ 0.3340 & 0.0632 & 0.0114 & -0.0599 & 0.0000 \\ 0.1228 & -0.0894 & 0.3162 & 0.0218 & 0.0000 \\ 0.7514 & 0.3785 & -0.0494 & 0.0514 & 0.0000 \end{bmatrix}$$

and formula **4** is used to calculate the column coordinates:

$$\mathbf{L} = \mathbf{D_c^{-1}BD} = \begin{bmatrix} -0.2364 & 0.0947 & 0.0163 & -0.0011 & 0.0000 \\ 0.1657 & -0.1494 & -0.0409 & -0.0098 & 0.0000 \\ 0.5512 & 0.3706 & -0.1127 & 0.0871 & 0.0000 \\ 0.3434 & -0.3056 & 0.3159 & 0.0492 & 0.0000 \\ 1.8884 & 1.2053 & 0.2546 & -0.1657 & 0.0000 \end{bmatrix}$$

3.1.2 Graphical representation of correspondence analysis

The plot in Figure 1 shows the association between the columns (likert scale categories) and the rows (employment sector groups) of the ASA members data set. The proximity of **Academic(nonstudent)** employment sector group and the **strongly agree** category indicates that the **Academic(nonstudent)** group is highly associated with the **strongly agree** category which is clearly the case from the row profile presented in Table 5, i.e. 64.0% of the **Academic(nonstudent)** group **strongly agree** that their primary position is professionally challenging [1]. Similarly, the proximity of the **Other(retired, students, unemployed)** employment sector group to the **no opinion** category and employment sector groups **Private consultant/self-employment** and **Business & industry** to the **agree** category indicates the higher frequency of respondents of those employment sector groups in those likert scale categories [3, 2]. The fact that the **strongly disagree** category is so far away from the employment sector group profiles indicates that no employment sector group is close to this extreme [3]. This is consistent with the contingency table and profiles of ASA members survey which shows that respondents agreed that their primary position is professionally challenging [1].

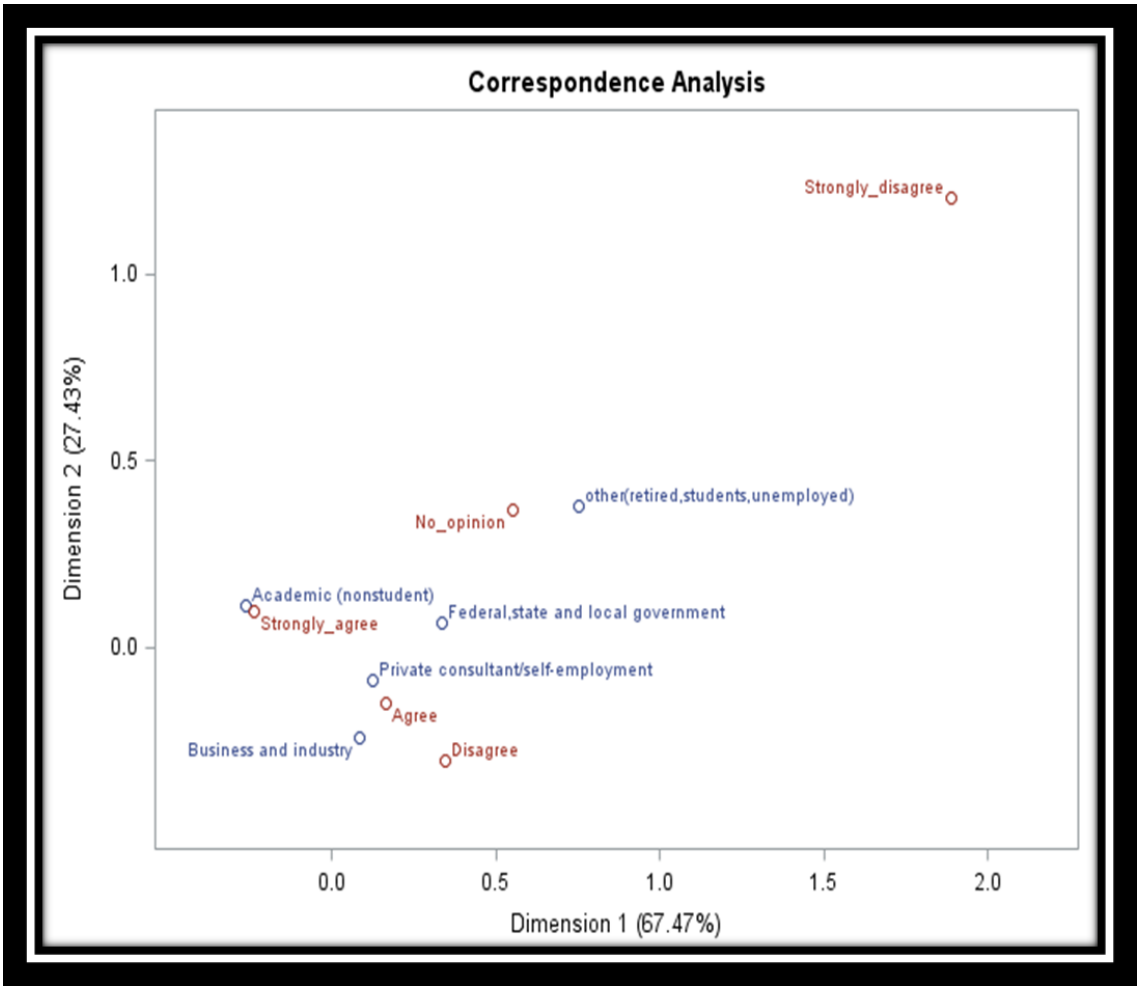


Figure 1: Correspondence analysis of ASA members survey by Lou and Keyes

4 Optimal scalingis then

The categorical data obtained by means of a questionnaire may be converted to quantitative data by awarding quantitative scale values to the various categories [2]. The technique used to obtain these quantitative values is known as optimal scaling, which is equivalent to performing a correspondence analysis on the contingency table [5]. The discussion of the optimal scaling approach will provide additional insight into the properties of correspondence analysis.

4.1 Assigning quantitative variables to the set of categories

Consider again Table 2 where seven age groups and five self-perceived health categories are cross-tabulated. We see that both the rows and columns variables are of a categorical nature. If we want to use self-perceived health categories as a dependent variable in subsequent analyses such as ANOVA and regression analysis, it would be important to have values for each health category [2]. Therefore, it may be true to assume that each of the health categories is exactly one unit apart on such a scale if we use the numbers 1 to 5. Self-perceived health is an ordinal categorical variable, which means it would be appropriate to use the values 1 to 5 [2]. For example, the numerical values may be assigned as follows to the health categories: 5 indicates a **very good** health, 4 indicates a **good** health, and so on down to 1 indicating a **very bad** health.

4.2 Two methods of performing optimal scaling

A total score cab be computed as the sum of scale values obtained by optimal scaling. It is interpreted in different ways due to the two methods applied in doing the optimal scaling. The first method calculates the scale values in such a way that they will have maximum variance between groups. This is done by forming groups so that they consist of individuals that respond very homogenously within the group, but differently between the groups [5]. The second method performs optimal scaling in such a way that the scale values vary maximally over individuals [5]. The total score calculated using either of the two methods can be used as a dependent variable in other statistical analyses like ANOVA and regression analysis.

5 Optimal scaling application

Quantifying a set of likert scale categories
The technique of optimal scaling will now be clarified by the use of examples based on the cross-tabulation in Table 3. In order to use a likert scale category as a variable in a statistical analyses such as ANOVA or regression analysis, or to calculate a statistic on the likert scale variable sush as variance or mean, it would be required to assign quantitative values for each likert scale category [2]. For example, if we assign values 1 to 5 for each likert scale category, it may be true to assume that each of the likert scale category is exactly one unit apart. The likert scale categories are ordered, which indeed make more sense to use the values 1 to 5 [2].
Calculation of overall mean using integer scale values
Firstly, let us reverse the coding of the likert scale categories so that 5 indicates **strongly agree**, 4 indicates **agree**, down to 1 indicating **strongly disagree**. The data set in Table 5 has a total sample of 562 respondents. There are 282 respondents under the **strongly agree** category (code 5), 230 respondents under the **agree** category (code 4), 25 respondents under the **no opinion** category (code 3), 21 respondents under the **disagree** category (code 2), and 4 respondents under the **strongly disagree** category (code 1). If we use these integer codes as scale values, the overall average of likert scale category can be calculated as follows:

$$\begin{aligned} mean &= \frac{282 \times 5 + 230 \times 4 + 25 \times 3 + 21 \times 2 + 4 \times 1}{562} \\ &= 4.36121 \end{aligned}$$

i.e.

$$mean = 0.5017794 \times 5 + 0.4092527 \times 4 + 0.044484 \times 3 + 0.0373665 \times 2 + 0.0071174 \times 1$$
$$= 4.36121$$

where $\frac{282}{562} = 0.501779$, $\frac{230}{562} = 0.4092527$, $\frac{25}{562} = 0.044484$, etc. are the elements of the row profiles, i.e. the elements in the last row of Table 5. The SAS program (IML procedure) is provided in the appendix.

Calculation of group means using integer scale values

Now let us consider a particular employment sector, say **Academic (nonstudent),** we see from the first row of the data set in Table 3 that out of a total of 253 participants in **Academic (nonstudent)** group, there are 163 participants under **strongly agree** category, 78 participants under **agree** category, etc [2]. If we use the integer codes as scale values again, the average likert scale category for **Academic (nonstudent)** group can be calculated as:

$$mean = \frac{162 \times 5 + 78 \times 4 + 8 \times 3 + 5 \times 2 + 0 \times 1}{253} = 4.56917 \tag{5}$$

i.e.

$$mean = 0.6403162 \times 5 + 0.3083004 \times 4 + 0.0316206 \times 3 + 0.0197628 \times 2 + 0 \times 1 = 4.56917 \tag{6}$$

where $\frac{162}{253} = 0.6403162$, $\frac{78}{253} = 0.3083004$, $\frac{8}{253} = 0.0316206$, etc. are the elements of the row profiles for **Academic (nonstudent)** group [2]. We could repeat the above calculation for the other four employment sector groups. The SAS program (IML procedure) is given in the appendix.

Table 7 shows the output of the averages for the five employment sector groups calculated using the integer scale values. Note that the values of the group means are very close to each other and to the overall average of 4.36121. This is because the integer scale values used to calculate the group means are exactly one unit apart from each other. The integer codes 1 to 5 do not determine the exact distance between each pair of likert scale categories [3]. That is why we need optimal scaling technique to determine numerical values which define a distance between each pair of categories [3].

is then

Employment_sector_groups	Group_mean
Academic (nonstudent)	4.56917
Business and industry	4.2556818
Federal,state and local government	4.1408451
Private consultant/self-employment	4.3214286
other(retired,students,unemployed)	3.8529412

Table 7: Averages for each employment sector group

Calculating scores with unknown scale values

The above computations of the overall mean and group means were based on the use of the integer scale values for the likert scale categories. The question is whether there are better scale values that can be used to do the above calculations.

Let us assume that the scale values for the likert scale categories are indicated by the unknown quantities I_1, I_2, I_3, I_4 and I_5 [2]. The possible technique whereby these unknown quantities are determined is known as optimal scaling, which is equivalent to performing a correspondence analysis on the contingency table [5].

Therefore, if we use these unknown quantities as scale values for the likert scale categories, the average across all the respondents would be:

$$mean = 0.5017794 \times I_1 + 0.4092527 \times I_2 + 0.044484 \times I_3 + 0.0373665 \times I_4 + 0.0071174 \times I_5 \qquad (7)$$

while the average for **Academic (nonstudent)** group, for example, would be:

$$mean = 0.6403162 \times I_1 + 0.3083004 \times I_2 + 0.0316206 \times I_3 + 0.0197628 \times I_4 + 0 \times I_5 = 4.56917 \qquad (8)$$

The averages calculated in this way, in terms of these unknown scale values i.e. unknown quantities to be determined by optimal scaling, are known as scores [2]. This implies that (7) is the average score and (8) is the score for the first employment sector group. The score can be formulated in terms of the unknown scale values in the same way for each of the employment sector groups . This implies that we will have five different scores, denoted by s_1, s_2, s_3, s_4 and s_5, for each of the employment sector groups [2].

5.1 Examples of the two methods applied in doing the optimal scaling

First method

As already discussed above, the first method of optimal scaling is concerned with forming groups in such a way that they consist of individuals that respond in the same way within the group, but differently between the groups [5]. Following this method, the scores $(s_1, s_2, ..., s_5)$ calculated by the obtained optimal scale values, will have maximum variance between employment sector groups. Now if one perfoms a correspondence analysis on the contingency table in Table 3, the computer output of the CORRESP procedure in Appendix will cotain the first two columns of the row and column coordintes as well as Inertia and Chi-squared Decomposition table. The table of Chi-squared Decomposition contains the values of principal inertia. The specific part of the principal inertia that is referred to as the first axis is equal to 0.082559 [2]. Therefore, it turns out that the positions of the likert scale categories along the best-fitting correspondence analysis dimension solve this first method for optimal scaling because the maximum variance is equal to the principal inertia (0.082559) on this optimal correspondence analysis dimension. The SAS program used to calculate the optimal scale values is given in the Appendix.

Column_labels	Optimal_scaled_values
STRONGLY_AGREE	-0.822608
AGREE	0.5767095
NO_OPINION	1.9180905
DISAGREE	1.1950482
STRONGLY_DISAGREE	6.5710179

Table 8: Optimal scale values for the likert scale categories

Table 8 shows the optimal scale values, I_1, I_2, I_3, I_4 and I_5, for the likert scale categories which maximize the variance of the employment sector groups. In the case of the integer scale values 5 to 1, the likert scale categories have equal interval of one unit between each pair of categories. The integer scale values (i.e. 5,4,3,2,1)

indicates that **no opinion** (3) is at the middle of scale values [2]. By eyeballing the figures in Table 8, we see that **no opinion** (1.9180905) is not at the middle of the optimal scale values [2]. The proximity of **no opinion** (1.9180905) to the **disagree** (1.1950482) indicates that **no opinion** scale value is much closer to the **disagree** scale value. The scale value of **strongly disagree** (6.5710179) is much further away from all of the other scale values, which is consistent with the results of the correspondence analysis. In addition, these optimal scale values lead to scores for the employment sector groups with maximum variation between the groups [2]. The SAS program is given in the Appendix.

Employment_sector_groups	Group_scores
Academic (nonstudent)	-0.264661
Business and industry	0.0810149
Federal,state and local government	0.3340188
Private consultant/self-employment	0.1227905
other(retired,students,unemployed)	0.7513866

Table 9: Scores of the employment sector groups

Table 9 provides the corresponding scores, $s_1, s_2, ..., s_5$, of the employment sector groups. Note that these values are the same as the values in the first column of row coordinates [1]. When we look clearly at the figures of Table 9, we see that there are large changes in likert scale categories between the employment sector groups **academic(nonstudent)** and **business and industry**, followed by small changes from **business and industry** to **private consultant/ self-employment** employment sector group, and then larger changes between **private consultant/ self-employment** and **other(retired, students, unemployed)** employment sector groups [2]. Looking back to the row profiles in Table 5, we can verify that from **business and industry** to **private consultant/ self-employment** employment sector group there are small changes of profiles in the **strongly agree** and **agree** categories compared to changes between the **academic(nonstudent)** and **business and industry** as well as **private consultant/ self-employment** and **other(retired, students, unemployed)** employment sector groups [2].

Second method

According to second method, optimal scaling is performed over individuals in such a way that the scores of likert scale categories will vary maximally over individuals [5]. This means that optimal scaling technique can also be used to obtain scale values for the employment sector groups which maximize the variance of the likert scale categories [2]. The SAS program (IML procedure) is given in the Appendix.

Employment_sector_groups	Optimal_scaled_values2
Academic (nonstudent)	-0.920927
Business and industry	0.2819034
Federal,state and local government	1.1622678
Private consultant/self-employment	0.4272676
other(retired,students,unemployed)	2.614561

Table 10: Optimal scale values for the employment sector groups

The figures in Table 10 indicates the optimal scale values for the employment sector groups which lead to scores with maximum variation over individuals [2]. As we can see, these scale values define distance between the employment sector groups. For example, we see that there is a small difference between **business and industry** and **private consultant/ self-employment** and a very big difference between **academic(nonstudent)** and **other(retired, students, unemployed)** [2]. The SAS program is given in the Appendix.

Column_labels	individual_scores
STRONGLY_AGREE	-0.236406
AGREE	0.1657379
NO_OPINION	0.5512312
DISAGREE	0.3434394
STRONGLY_DISAGREE	1.8884144

Table 11: Scores of the likert scale categories, i.e. individual scores

Table 11 provides the corresponding scores, $s_1, s_2, ..., s_5$, of the likert scale categories. Note that this values are the same as the values in the first column of column coordinates. From Table 11 it follows that a high score indicates **strongly disagree** over the particular statements in the survey. Avery low score on the other hand indicates that the respondents **strongly agreed** over the particular statements [3, 2].

Analysis of an optimally scaled dependent variable

In the case of multivariate categorical data, the categorical variables can be assigned numerical scale values obtained by a technique known as optimal scaling [2, 3]. The score of the categorical variables is calculated as the sum of optimal scale values and this score is then used as a dependent variable in subsequent analyses like ANOVA, XAID and regression analysis[2].

6 Conclusion

In this report optimal scaling was considered as a technique for transforming multivariate categorical data into continuous variables. The optimal scale values assigned to the categories of multivariate categorical data compute some criterion which lead to maximum variation over the individuals as well as between the groups [2]. Unlike the integer scale values, these optimal scale values define a distance between each pair of categories of multivariate categorical data [2, 3]. The results of correspondence analysis provide more insight in the distance, defined by optimal scale values, between each pair of categories [2].
In the case of multivariate categorical data, the category choices of attitude questions all related to the same theme, can be assigned scaled values obtained by optimal scaling [2]. A total score can be calculated as the sum of scaled values and this total score can be used as a dependent variable in subsequent analyses like ANOVA, XAID and regression analysis [3, 2]. The interpretation of the analyses however is given in terms of the original categorical variables.
All the relevent computer programs and outputs are included in the appendices.

References

[1] Eric J Beh and Rosaria Lombardo. *Correspondence Analysis: Theory, Practice and New Strategies*. John Wiley & Sons, 2014.

[2] Michael Greenacre. *Correspondence Analysis in Practice*. CRC Press, 2007.

[3] Michael J Greenacre. *Theory and Applications of Correspondence Analysis*. Año de, 1984.

[4] Amy Luo and Tim Keyes. Second set of results in from the career track member survey. *Amstat News*, 2:1058–1066, 2005.

[5] SHC Du Toit and C Strasheim. The optimal scaling technique for the analysis of multivariate categorical data. *White Paper*, 00:1–28, 1987.

Appendix

1.CORRESP procedure

```
data lesedi;
input Employment_sector $1-37 Strongly_agree Agree No_opinion Disagree Strongly_disagree;
datalines;
Academic (nonstudent)  162 78 8  5 0
Business and industry   72 88 5 11 0
Federal,state and local government  27 34 5  3 2
Private consultant/self-employment  11 15 2  0 0
other(retired,students,unemployed)  10 15 5  2 2
;
proc print data=lesedi;
run;
ods graphics on;
proc corresp data=lesedi cellchi2 all deviation short plot GREENACRE;
var Strongly_agree Agree No_opinion Disagree Strongly_disagree;
id Employment_sector;
run; used as a dependent variable in existing techniques for the analysis of quantitative data such as A
ods graphics off;
```

CORRESP procedure output

The SAS System

The CORRESP Procedure

Contingency Table

	Strongly_agree	Agree	No_opinion	Disagree	Strongly_disagree	Sum
Academic (nonstudent)	162	78	8	5	0	253
Business and industry	72	88	5	11	0	176
Federal,state and local government	27	34	5	3	2	71
Private consultant/self-employment	11	15	2	0	0	28
other(retired,students,unemployed)	10	15	5	2	2	34
Sum	282	230	25	21	4	562

Row Profiles					
	Strongly_agree	Agree	No_opinion	Disagree	Strongly_disagree
Academic (nonstudent)	0.640316	0.308300	0.031621	0.019763	0.000000
Business and industry	0.409091	0.500000	0.028409	0.062500	0.000000
Federal,state and local government	0.380282	0.478873	0.070423	0.042254	0.028169
Private consultant/self-employment	0.392857	0.535714	0.071429	0.000000	0.000000
other(retired,students,unemployed)	0.294118	0.441176	0.147059	0.058824	0.058824

Column Profiles					
	Strongly_agree	Agree	No_opinion	Disagree	Strongly_disagree
Academic (nonstudent)	0.574468	0.339130	0.320000	0.238095	0.000000
Business and industry	0.255319	0.382609	0.200000	0.523810	0.000000
Federal,state and local government	0.095745	0.147826	0.200000	0.142857	0.500000
Private consultant/self-employment	0.039007	0.065217	0.080000	0.000000	0.000000
other(retired,students,unemployed)	0.035461	0.065217	0.200000	0.095238	0.500000

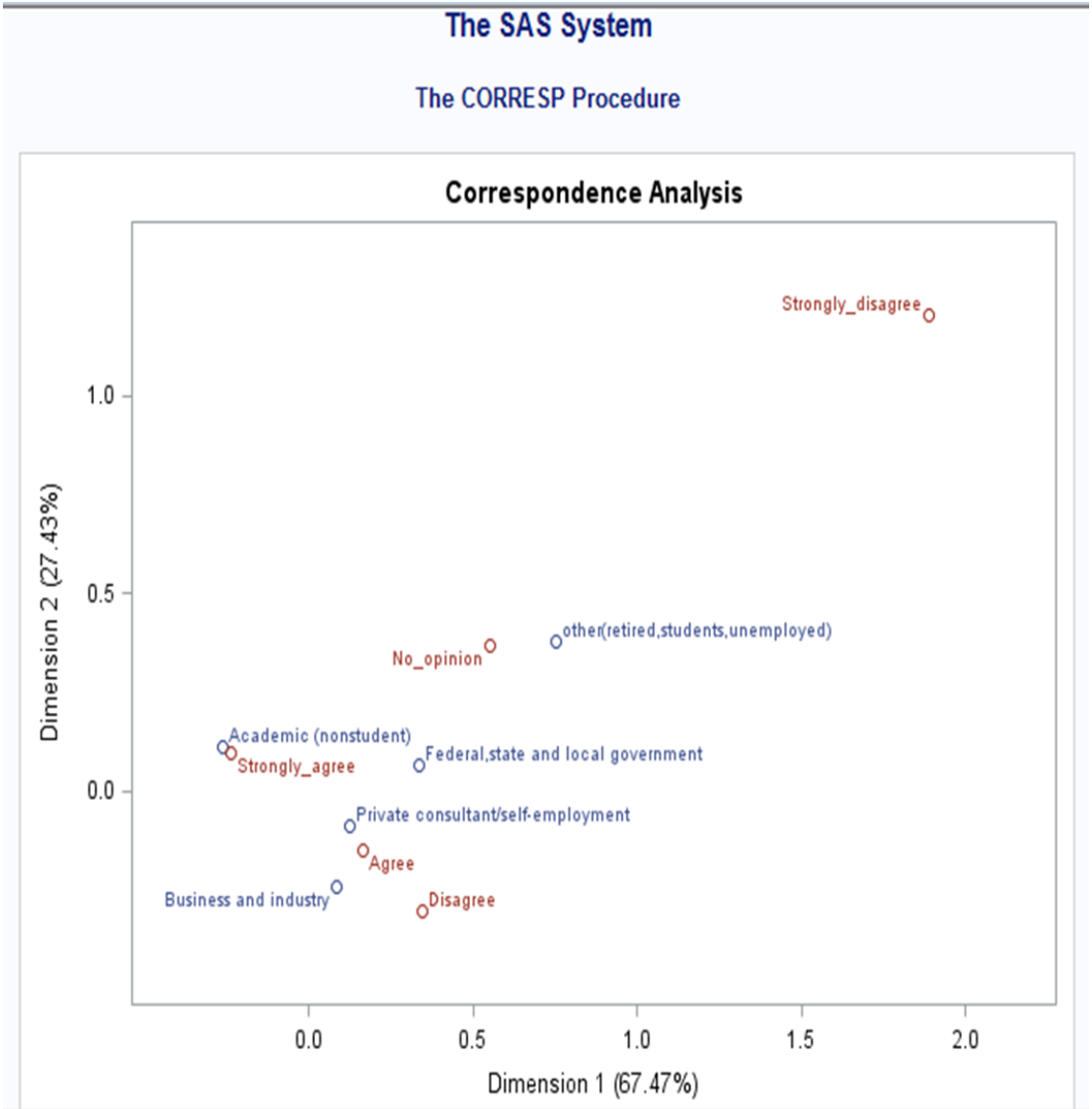
The SAS System

The CORRESP Procedure

Inertia and Chi-Square Decomposition					
Singular Value	Principal Inertia	Chi-Square	Percent	Cumulative Percent	13 26 39 52 65 ----+----+----+----+----+---
0.28739	0.08259	46.4158	67.47	67.47	**************************
0.18323	0.03357	18.8680	27.43	94.90	**********
0.07467	0.00558	3.1335	4.56	99.46	**
0.02574	0.00066	0.3725	0.54	100.00	
Total	0.12240	68.7897	100.00		
Degrees of Freedom = 16					

Row Coordinates		
	Dim1	Dim2
Academic (nonstudent)	-0.2647	0.1104
Business and industry	0.0810	-0.2431
Federal,state and local government	0.3340	0.0632
Private consultant/self-employment	0.1228	-0.0894
other(retired,students,unemployed)	0.7514	0.3785

Column Coordinates		
	Dim1	Dim2
Strongly_agree	-0.2364	0.0947
Agree	0.1657	-0.1494
No_opinion	0.5512	0.3706
Disagree	0.3434	-0.3056
Strongly_disagree	1.8884	1.2053



2. IML procedure

```
data lesedi;
input Employment_sector $1-37 Strongly_agree Agree No_opinion Disagree Strongly_disagree;
datalines;
Academic (nonstudent)   162 78 8  5 0
Business and industry    72 88 5 11 0
Federal,state and local government  27 34 5  3 2
Private consultant/self-employment  11 15 2  0 0
other(retired,students,unemployed)  10 15 5  2 2
;
proc print data=lesedi;
run;
proc iml;
use lesedi;
read all into F;
read all var {Employment_sector} into Y;
```

```
one=J(5,1,1); /*vector of 1s*/
f1=one‘*F*one; /*sum of all elements in contingency table*/
print f1; is then
S= F/f1;/*correspondence matrix*/
Correspondence_matrix=S;
Employment_sector_groups=Y;
print Employment_sector_groups Correspondence_matrix;
/*row and column totals of correspondence matrix*/
r=s*one;
c=s‘*one;
print r;
print c;
/*diagonal matrix of r and c*/
Dr=diag(r);
Dc=diag(c);
print Dr;
print Dc;
Dr1=inv(dr);
dr2=sqrt(Dr1);
Dc1=inv(dc);
dc2=sqrt(Dc1);
print dr2;
print dc2;
/*Row and column profiles*/
R1=inv(Dr)*S;
CT=inv(Dc)*S‘;/*transpose of column profiles*/
C1=CT‘; /*column profiles*/
A={SUM};
Position2=Y//A;
Employment_sector_groups=position2;
print Employment_sector_groups;
/*Sum of elements in contingency table*/
Sumcc=F[+,];/*sum columns of contingency table*/
Crosstab=F//Sumcc;
Sumrc=Crosstab[,+];/*sum rows of contingency table*/
print Employment_sector_groups Crosstab Sumrc;
/*Sum of row and column profiles*/
Average_r=Sumcc[1,]/Sumrc[6,1];
Row_profile=R1//Average_r;
SUM=Row_profile[,+];/*sum rows of matrix R2*/
X={Average};
Position1=Y//X;
Employment_sector_groups=position1;
print Employment_sector_groups;
csum2=C1[,+];/*sum rows of matrix C1*/
sumc2=csum2[+,];/*sum of all values in column vector csum2*/
sum2=csum2/sumc2;
Totalc=sum2[+,];/*sum column of row vector sum2*/
Average_c=Sumrc[,1]/Sumrc[6,1];
Total2=C1[+,];/*sum columns of matrix C2*/
Column_profile=C1//Total2;
print Employment_sector_groups Row_profile SUM;
/*percentages of row profiles*/
```

```
Row_profile_percent=(R1//Average_r)*100;
SUM1=(Row_profile[,+])*100;
print Position1 Row_profile_percent SUM1;
Employment_sector_groups=position2;
print Employment_sector_groups Column_profile Average_c;
/*percentages of column profiles*/
Average_c=(Sumrc[,1]/Sumrc[6,1])*100;
Column_profile_percent=(C1//Total2)*100;
print Position2 Column_profile_percent Average_c;
/*deviation of S from its centre*/
Q=S - r*c`;
print Q; /*RANK s BASIC STRUCTURE*/
O=dr2*Q*dc2; print O;
/*norm of the scaled matric S*/
Norm=trace(inv(Dr)*S*inv(Dc)*S`);
print Norm;
Norm2=one`*((S##2)/(r*c`))*one;
print Norm2;
/*symmetric treatment of rows and columns*/
R1=(inv(Dr)*S)*100;/*percentage of row profile matrix R1*/
Average_r=Sumcc[1,]/Sumrc[6,1]*100;/*percentage of average of row profile matrix*/ Row_profile_ratios=(R
||(R1[,4]/Average_r[1,4])||(R1[,5]/Average_r[1,5]);/*row profile ratios*/
print Row_profile_ratios;
C1=(CT`)*100;/*percentage of column profile matrix C1*/ Column_profile_ratios=(C1[1,]/Average_c[1,1])//(
//(C1[4,]/Average_c[4,1])//(C1[5,]/Average_c[5,1]);/*Column profile ratios*/
print Column_profile_ratios;
/*Association between rows and column*/
F_rc=(F-r*c`)##2;/*(fij-rc`)##2*/
print F_rc;
Ratio=F_rc/(r*c`); /* ((fij-rc`)^2)/(rc`) */
n=562; /* sample size*/
Sum=one`*Ratio*one; /* one is the row vector of 1s */
Chi_squared=n*Sum;
print Chi_squared;
determinant=det(F); /* determinant of contingency table F */
print determinant; /* it is not equal to zero, therefore we conclude that categorical variables are asso
/*rank*/
rank = round(trace(ginv(O)*O));
print rank;
W=O*O`;
CALL EIGEN(eigenvalue1,eigenvector1,W); 2.SAS program of matrix required for correspondence analysis
print eigenvalue1;/*Singular value decomposition of S*/
U=eigenvector1;/*left singular vector*/
print U;
Du=diag(SQRT(eigenvalue1));/*diagonal matrix of singular values*/
print Du;
H=O`*O;
CALL EIGEN(eigenvalue2,eigenvector2,H);
print eigenvalue2;
V=eigenvector2;
Print V;
SVD=U*Du*V`;/*Singular value decomposition of S*/
print SVD;
```

```
A=(Sqrt(Dr))*U;/*rectangular matrix of left generalized singular vector*/
B=(Sqrt(Dc))*V;/*rectangular matrix of right generalized singular vector*/
Column_labels={Strongly_agree, Agree, No_opinion, Disagree, Strongly_disagree};
print Column_labels;
Row_coordinate=(inv(Dr))*A*Du;/*row coordinates*/
Column_coordinate=(inv(Dc))*B*Du;/*column coordinates*/
print y Row_coordinate;
print Column_labels Column_coordinate;
/*calculating overall mean using integer scale values*/
Int_scale = {5,4,3,2,1}; /*integer scale values*/
Participants=crosstab[6,];
Total_sample=Sumrc[6,1]; /*total sample of 562 respondents*/
Overall_mean=(Participants*Int_scale)/Total_sample; /*overall mean of likert scale category*/
print Overall_mean;
RP_total=Row_profile[6,]; /*row profile average*/
Overall_mean2= (RP_total*Int_scale); /*overall mean of likert scale category using row profiles*/
print Overall_mean2;
/*calculating group means using integer scale values*/
Group=F*Int_scale; Group_mean=(Group[1,1]/sumrc[1,1])//(Group[2,1]/sumrc[2,1])//(Group[3,1]/sumrc[3,1]),
//(Group[5,1]/sumrc[5,1]);
Rprofile=R1/100;
Group_mean1=Rprofile*Int_scale;
print Employment_sector_groups Group_mean;
/*calculating column scale values*/ is then
Principal_inertia=SQRT(0.08259);/*Square root of principal inertia*/ Optimal_scaled_values=Column_coordi
print Column_labels Optimal_scaled_values;
/*calculating scores for the employment sector groups*/
Group_scores=Rprofile*Optimal_scaled_values;
Employment_sector_groups=y; print Employment_sector_groups Group_scores;
/*calculating row scale values*/
Optimal_scaled_values2=Row_coordinate[,1]*(1/Principal_inertia);/* calculates row vertex coordinates*/
print Employment_sector_groups Optimal_scaled_values2;
/*calculating total score for individuals*/
Individual_socores=(C1‘/100)*Optimal_scaled_values2;
print Column_labels Individual_socores;
```

IML procedure output

	Employment_sector_groups Col1	Column_profile Col2	Col3	Col4	Col5	Col6	Average_c Col7
ROW1	Academic (nonstudent)	0.5744681	0.3391304	0.32	0.2380952	0	0.4501779
ROW2	Business and industry	0.2553191	0.3826087	0.2	0.5238095	0	0.3131673
ROW3	Federal,state and local government	0.0957447	0.1478261	0.2	0.1428571	0.5	0.1263345
ROW4	Private consultant/self-employment	0.0390071	0.0652174	0.08	0	0	0.0498221
ROW5	other(retired,students,unemployed)	0.035461	0.0652174	0.2	0.0952381	0.5	0.0604982
ROW6	SUM	1	1	1	1	1	1

	Position2 Col1	Column_profile_percent Col2	Col3	Col4	Col5	Col6	Average_c Col7
ROW1	Academic (nonstudent)	57.446809	33.913043	32	23.809524	0	45.017794
ROW2	Business and industry	25.531915	38.26087	20	52.380952	0	31.316726
ROW3	Federal,state and local government	9.5744681	14.782609	20	14.285714	50	12.633452
ROW4	Private consultant/self-employment	3.9007092	6.5217391	8	0	0	4.9822064
ROW5	other(retired,students,unemployed)	3.5460993	6.5217391	20	9.5238095	50	6.0498221
ROW6	SUM	100	100	100	100	100	100

Employment_sector_gr
Academic (nonstudent)
Business and industry
Federal,state and local g
Private consultant/self-em
other(retired,students,une

c
0.5017794
0.4092527
0.044484
0.0373665
0.0071174

Dr				
0.4501779	0	0	0	0
0	0.3131673	0	0	0
0	0	0.1263345	0	0
0	0	0	0.0498221	0
0	0	0	0	0.0604982

Dc				
0.5017794	0	0	0	0
0	0.4092527	0	0	0
0	0	0.044484	0	0
0	0	0	0.0373665	0
0	0	0	0	0.0071174

	Employment_sector_groups Col1	Row_profile Col2	Col3	Col4	Col5	Col6	SUM Col7
ROW1	Academic (nonstudent)	0.6403162	0.3083004	0.0316206	0.0197628	0	1
ROW2	Business and industry	0.4090909	0.5	0.0284091	0.0625	0	1
ROW3	Federal,state and local government	0.3802817	0.4788732	0.0704225	0.0422535	0.028169	1
ROW4	Private consultant/self-employment	0.3928571	0.5357143	0.0714286	0	0	1
ROW5	other(retired,students,unemployed)	0.2941176	0.4411765	0.1470588	0.0588235	0.0588235	1
ROW6	AVERAGE	0.5017794	0.4092527	0.044484	0.0373665	0.0071174	1

	Position1 Col1	Row_profile_percent Col2	Col3	Col4	Col5	Col6	SUM1 Col7
ROW1	Academic (nonstudent)	64.031621	30.83004	3.1620553	1.9762846	0	100
ROW2	Business and industry	40.909091	50	2.8409091	6.25	0	100
ROW3	Federal,state and local government	38.028169	47.887324	7.0422535	4.2253521	2.8169014	100
ROW4	Private consultant/self-employment	39.285714	53.571429	7.1428571	0	0	100
ROW5	other(retired,students,unemployed)	29.411765	44.117647	14.705882	5.8823529	5.8823529	100
ROW6	AVERAGE	50.177936	40.925267	4.4483986	3.7366548	0.7117438	100

	Employment_sector_groups Col1	Column_profile Col2	Col3	Col4	Col5	Col6	Average_c Col7
ROW1	Academic (nonstudent)	0.5744681	0.3391304	0.32	0.2380952	0	0.4501779
ROW2	Business and industry	0.2553191	0.3826087	0.2	0.5238095	0	0.3131673
ROW3	Federal,state and local government	0.0957447	0.1478261	0.2	0.1428571	0.5	0.1263345
ROW4	Private consultant/self-employment	0.0390071	0.0652174	0.08	0	0	0.0498221
ROW5	other(retired,students,unemployed)	0.035461	0.0652174	0.2	0.0952381	0.5	0.0604982
ROW6	SUM	1	1	1	1	1	1

	Position2 Col1	Column_profile_percent Col2	Col3	Col4	Col5	Col6	Average_c Col7
ROW1	Academic (nonstudent)	57.446809	33.913043	32	23.809524	0	45.017794
ROW2	Business and industry	25.531915	38.26087	20	52.380952	0	31.316726
ROW3	Federal,state and local government	9.5744681	14.782609	20	14.285714	50	12.633452
ROW4	Private consultant/self-employment	3.9007092	6.5217391	8	0	0	4.9822064
ROW5	other(retired,students,unemployed)	3.5460993	6.5217391	20	9.5238095	50	6.0498221
ROW6	SUM	100	100	100	100	100	100

Column_labels
STRONGLY_AGREE
AGREE
NO_OPINION
DISAGREE
STRONGLY_DISAGREE

y	Row_coordinate				
Academic (nonstudent)	-0.26466	0.1104297	-0.00631	0.001379	6.791E-10
Business and industry	0.0810147	-0.243141	-0.036288	0.0087934	6.791E-10
Federal,state and local government	0.3340181	0.0632393	0.0114177	-0.059948	6.791E-10
Private consultant/self-employment	0.1227902	-0.089424	0.316188	0.021843	6.791E-10
other(retired,students,unemployed)	0.751385	0.3784712	-0.049436	0.0514163	6.791E-10

Column_labels	Column_coordinate				
STRONGLY_AGREE	-0.236405	0.0946757	0.0162869	-0.00105	6.791E-10
AGREE	0.1657375	-0.149422	-0.04099	-0.009787	6.791E-10
NO_OPINION	0.55123	0.3705601	-0.11269	0.087051	6.791E-10
DISAGREE	0.3434386	-0.305562	0.3158903	0.0492233	6.791E-10
STRONGLY_DISAGREE	1.8884104	1.2053495	0.2545782	-0.165695	6.791E-10

Employment_sector_groups	Group_mean
Academic (nonstudent)	4.56917
Business and industry	4.2556818
Federal,state and local government	4.1408451
Private consultant/self-employment	4.3214286
other(retired,students,unemployed)	3.8529412
SUM	

Column_labels	Optimal_scaled_values
STRONGLY_AGREE	-0.822608
AGREE	0.5767095
NO_OPINION	1.9180905
DISAGREE	1.1950482
STRONGLY_DISAGREE	6.5710179

Employment_sector_groups	Group_scores
Academic (nonstudent)	-0.264661
Business and industry	0.0810149
Federal,state and local government	0.3340188
Private consultant/self-employment	0.1227905
other(retired,students,unemployed)	0.7513866

Employment_sector_groups	Optimal_scaled_values2
Academic (nonstudent)	-0.920927
Business and industry	0.2819034
Federal,state and local government	1.1622678
Private consultant/self-employment	0.4272676
other(retired,students,unemployed)	2.614561

Column_labels	Individual_socores
STRONGLY_AGREE	-0.236406
AGREE	0.1657379
NO_OPINION	0.5512312
DISAGREE	0.3434394
STRONGLY_DISAGREE	1.8884144

Tests for complete spatial randomness

Francois Meintjes 12078302

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr I Fabris-Rotelli

Department of Statistics, University of Pretoria

2 November 2015

Abstract

Spatial statistics is one of the most up and coming areas in statistics which is easier now to consider then years back due to the variety of methods for testing for randomness on some point pattern. In this research report, we explain the main theory and background behind spatial point patterns and discuss the different tests that can be applied to test for spatial randomness. Furthermore, we apply these tests to a certain point pattern obtained from the pulses of the Discrete Pulse Transform and reach a conclusion that our point process is indeed a regular point pattern. Lastly, we will give some conclusion for spatial point patterns in general.

Declaration

I, Francois Meintjes, declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Francois Meintjes

_____
Supervisor: Inger Fabris-Rotelli

_____
Date

Acknowledgements

This work is based on the research supported in part from the National Research Foundation of South Africa for the Grant 90315. Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard.

Contents

List of Figures

1 Introduction

Testing for complete spatial randomness means that for any point pattern, tests can be applied to confirm whether the points are randomly distributed or if there is some pattern between the points. To understand this concept better, we will first define what a point process is and the main types of point processes exist.

A point process is defined as a set of locations that are irregularly distributed within a designated region (most often a quadrant) and it is presumed that the points have been generated by some form of a stochastic mechanism [10].

The three main branches behind point processes consists of [10]:

- Geostatistic point process
- Lattice point processes
- Spatial point processes

In this report, we will focus on spatial point processes. Figure 2 serves as a demonstration of a point process [10].

When testing for complete spatial randomness, three possibilities can occur. Either the data will be regular, random or clustered; where regular implies there exist some pattern, clustered implies some of the points are grouped together and random implies there there exist no pattern or grouping of the points. Figure 3 serves as a visual comparison between the three possibilities that can occur [2]. There are many tests that can be applied to test for complete spatial randomness. We will consider the F, G, K, L and J functions [2].

As with all tests, there usually exist some conditions that must hold or 'problem areas' concerning the test. Concerning point processes, this 'problem area' is the 'edge effect'. Edge effects arise when the quadrant on which the pattern is observed is part of a larger region on which the underlying process operates but is not observed. Luckily, there exist some basic methods to deal with edge effects which will be discussed in the theoretical section of this report. The first method is using buffer zones where buffer zones consists of carrying out all aspects of a statistical analysis after conditioning on the locations of all events which fall within a buffer zone consisting of all events less than some specified distance from the edges of the domain, i.e. using a subset of the domain so that uninformative edges do not occur[8]. Figure 1 illustrates this.

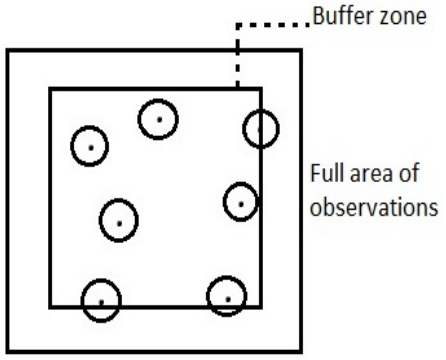


Figure 1: Buffer zones

The second method is explicit adjustments on observed values to take the unobserved values into account. The final method takes into consideration weather the quadrant is rectangular or not. If the quadrant is rectangular, then the quadrant is wrapped onto a torus by identifying opposite edges [8].

Generally, a simple point process is defined for the univariate case, which means that a certain point consists of two variables, say (X, Y), where the variables refer to the position in $\mathbb{R}^2$. A point process can also be defined for the multivariate case, which means it consists of variables $(X, Y, f(X, Y))$ for some discrete function f. Most often the function f takes on two discrete values/event types. No fixed information is included on the function as this function can be of any form. This function is usually known as a covariate,

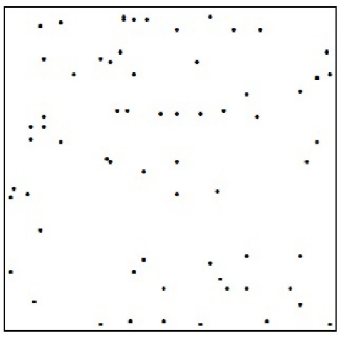
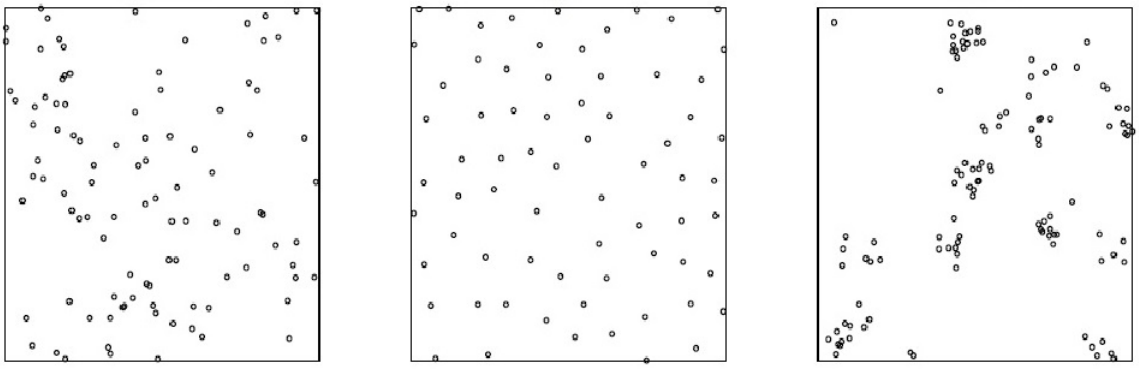


Figure 2: Locations of 65 Japanese black pine saplings in a square of side-length 5.7 meters



(a) Random (b) Regular (c) Clustered

Figure 3: The three possibilities that can occur for spatial point processes

and additional information about the events can be added through it. As an example of the multivariate case, consider a quadrant with certain trees as the events. Just considering the trees will be sufficient as an example of a univariate point process. Adding geographic information, such as height, changes it from a univariate case to a multivariate case where the function $f(x, y)$ is the height of the tree at position (x, y).

This report will introduce the theory of spatial point processes and the functions F, G, K, L and J to measure the nature of a spatial point process. We will focus on the application of this theory on objects extracted from image data.

2 Literature Review

Diggle [8] was one of the first to give any information about spatial point processes through a formal definition as well as a elementary example. From there onwards, articles have been written about spatial point processes and their applications. One such application example is [18] in which Waller and Gotway explain how point processes can be used to analyze everyday problems, such as in this case disease patterns.

In [3] Baddeley et al give a brief explanation of how spatial point processes are defined in the different dimensions as well as how a Poisson process can be applied to a specific spatial point process problem. The Poisson process model can also be seen in [9] where Diggle explains in much detail about how the Poisson process model is defined and all the functions used in this process such as the intensity function. Furthermore, in this article Diggle discusses the maximum likelihood estimator for the function used under the Poisson point process as well as the goodness-of-fit test to assess if the fitted model is a decent model.

In [10], Diggle states a formal hypothesis for testing Complete Spatial Randomness (abbreviated further

in this report as CSR), how this hypothesis is tested using functions such as the F, G, K, L and J functions. In this specific article, Diggle also explains the edge-effects that can occur when testing this hypothesis.

Loh and Stein explain in [12] in great detail what is meant by 'Bootstapping' and what the consequences and results are when applying bootstrapping to a point process. One of the main consequences that follow is how bootstrapping can be used to calculate confidence intervals for the second moment function where as in [15] Sarma et al give a brief explanation on how confidence intervals are used when considering a point process that has the brain as a main center point.

In [7] Vasudevan et al use an example of fixated locations captured by the eyes as a spatial point pattern tested to see whether the locations are completely random or not, discussing each concept such as covariates, intensity and estimations/approximations of the likelihood functions.

An article that gives a good example on how spatial point patterns are used is [17] where Vasudevan et al explain in excellent detail the procedure used for spatial point processes, how the L function is applied to the process, the results that occur and also how a conclusion is made from the observed results.

In [2], Baddeley gives a broad overview concerning point processes but focuses mainly on how the software $R^{©}$[1] can be used to calculate the F, G, K, L and J functions in testing for CSR.

In [4] Baddeley et al give a insightful explanation on how in some point processes, decent models can be fitted though most fitted models gives limited results, further models and tests can be applied to the residuals which can then give more results/information concerning the given point process.

Concerning the G, K, L and J functions, there are three articles which are of most importance. In [13] Ripley was the first to define the K and L functions, hence the name 'Ripley's K and L functions'. In [11] Van Lieshout and Baddeley define the J function and the construction of the function. The J function uses the G function, where the G function was first defined in [3] by Baddeley.

3 Measures of Complete Spatial Randomness

Before we can discuss any measures of CSR, we need to formally define a Poisson process with a constant parameter λ [6]. Consider some recurring event. Let $N(t)$ be the number of such events in the interval $[0, t]$ and assume that the following properties hold (a more technical definition is given in the appendix):

- The probability that an event will occur in a given short interval $[t, t+\Delta t]$ is proportional to the length of the interval, Δt, and does not depend on the position of the interval,
- The occurence of events in nonoverlapping intervals are independent,
- The probability of two or more events occuring in a short interval $[t, t + \Delta t]$ can be considered as negligible.

If the assumptions listed above hold as the length $\Delta t \to 0$, then $N(t)$ will have a Poisson distribution with parameter λt, that is $P_n(t) = P[N(t) = n] = e^{-\lambda t}(\lambda t)^n/n!$. This Poisson process definition is for the homogeneous case and can be extended for the inhomogeneous case. For a completely spatially random homogeneous point process defined on a study area W, usually a quadrant, the number of points,$N(A)$, in a region $A \subseteq W$ is Poisson distributed with parameter λ.

The Poisson process is of fundamental importance to the spatial statistics functions we wil shortly define, as throughout all the following measures, we use the Poisson process concept. To give a better explanation of each measure, we will give a explanation as well as a example using the *spatstat* package in the statistical software $R^{©}$. We consider a homogeneous point process at first. Homogeneous is also referred to as stationarity across a point process domain, that is, invariance under translation. The estimation of the parameter λ is given by

[1] Adrian Baddeley, Rolf Turner, Jorge Mateu, Andrew Bevan (2013). Hybrids of Gibbs Point Process Models and Their Implementation. Journal of Statistical Software, 55(11), 1-43. URL http://www.jstatsoft.org/v55/i11/

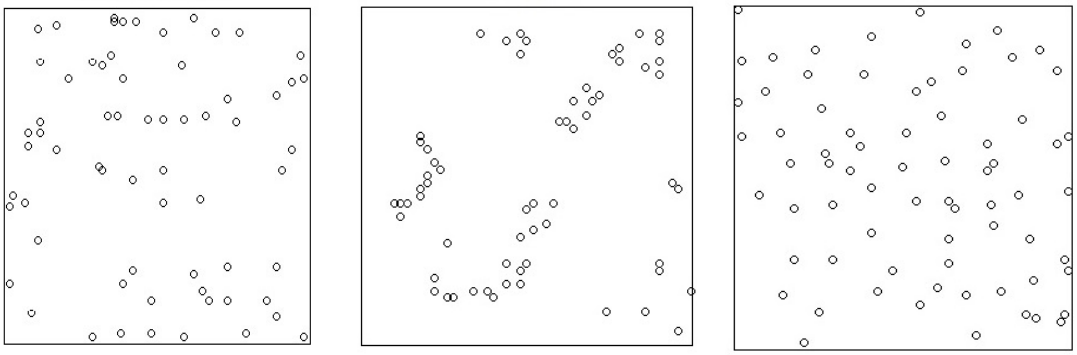
$$\begin{aligned}\hat{\lambda} &= \frac{\text{card}(X)}{\text{area}(W)} \\ &= \frac{\sum_{i=1}^{m} n_i}{\text{area}(W)}\end{aligned}$$

where X is the homogeneous point pattern and W is the study area (quadrant). The study area W is divided into m equally sized quadrants and n_i is the number of events in each of these quadrants [8].

To illustate how the functions are interpreted, we use three data sets from $R^{©}$, namely: *japanesepines*, *redwood* and *swedishpines*. We run the specific function on each of these data sets, and we make a conlusion based on the results. First we plot each data set to see is we can make some conclusion based on the plotted data. We use the following coding in $R^{©}$ to import and plot the data sets:

```
library(spatstat)
data(japanesepines)
plot(japanesepines)
data(redwood)
plot(redwood)
data(swedishpines)
plot(swedishpines)
```

Figure 4 provides the output of the code.



(a) Japanese Pines (b) Redwood (c) Swedish Pines

Figure 4: Plots of the data sets

From the images in Figure 4, we get the impression that the Japanese Pines is random, the Redwood is clustered and the Swedish Pines is regular. We will test this using each of the functions F, G, J, K and L.

3.1 Edge Effects

As discussed in the introduction, a 'problem area' when running some tests on a point pattern is the edge effects. We will briefly discuss the two commonly used methods to deal with edge effects.

The first method discussed is known as the 'border method' or 'reduced sample method' (in the introduction we introduced this concept as 'buffer zones').

Border Method

The main theory behind the border method is that we restrict our observation window, W, so that we only take into consideration the point pattern r and more units away from the border of W and the interior of W [5]: Consider a point process, X (in $\mathbb{R}^2$), illustrated in Figure 5.

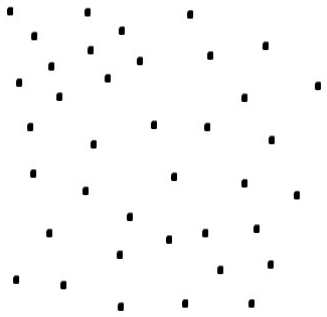


Figure 5: A point process

We observe a window, W, containing some points of X in W and some not (therefore unknown), as in Figure 6.

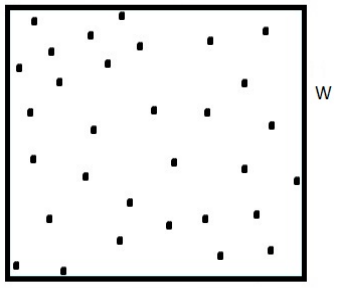


Figure 6: Observed window W

From Figure 6, we observe the problem of edge effects. We do not know (have not observed) the entire neighbouring point process further than the border of W. So what we do to help us deal with this problem, is create a smaller window, W_-, inside of W which is shown in Figure 7, which consists of the points we make use of.

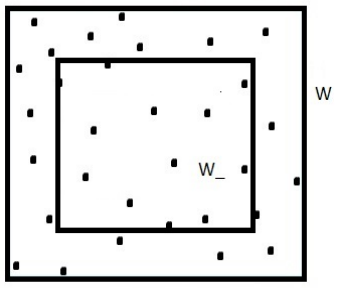


Figure 7: Smaller observation window in W, namely W_-

More detailed theoretical information can be found in [5] by Baddeley. Thise correction adjustment is calculated exactly the same for the inhomogeneous point process. For more information on the inhomogeneous point process and this calculation, see [16] by van Lieshout.

The second method for edge correction is known as Ripley's isotropic correction.

Ripley's Isotropic Correction

This correction is solely based on isotropy where isotropy can also be understood as stationarity under rotations about a fixed point, with homogeneity about any point [14]. Consider some observed point process X inside the window W. This is shown in Figure 8

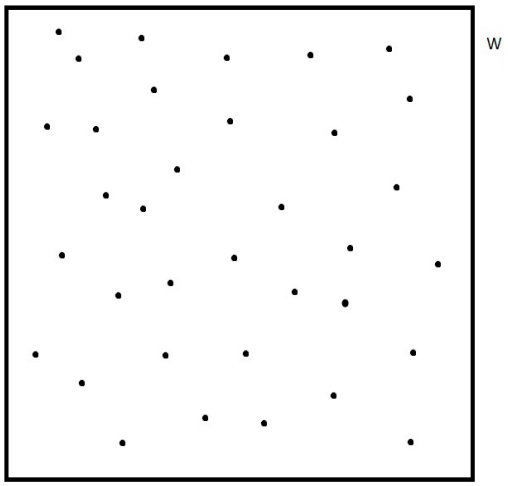


Figure 8: Observed window W

If we then consider a ball centered at a point x (also seen as pair (x, y)) we notice that there might be points outside of W but inside the ball which we do not have infomrtion about. This is shown in Figure 9

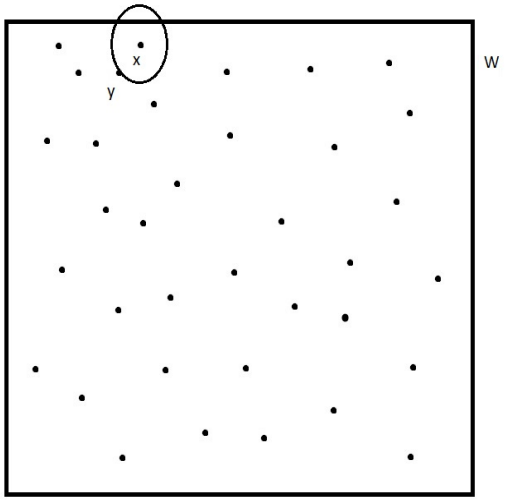


Figure 9: Observed pair (x, y)

Because of this problem, some correction needs to be added. This is known as RIipley's Isotropic Correction.

3.2 The F-Function

Assume X is a stationary point process. Using the cumulative distribution function (also known as the Empty Space Function), the F-function of the empty space distance, developed by Baddeley et al. [3] is defined in [2] as:

$$F(r) = P[d(u, X) \leq r]$$

where $d(u, X) = \min\{\|u - x_i\| : x_i \in X\}$ is the shortest distance from u to the point pattern X, where u is some arbitrary location and r is some distance. Since we assume X is a stationary point process, this definition is valid for all u.

The empirical empty space distance function on a grid of locations u_j, $j = 1, 2, ..., m$ can be estimated by

$$F^*(r) = \frac{1}{m}\sum_j 1_{\{d(u_j, X) \leq r\}}.$$

This estimator for $F(r)$ is negatively biased due to edge effects. A estimator where edge effects are taken into account is given by

$$\hat{F}(r) = \sum_j e(u_j, r) 1_{\{d(u_j, X) \leq r\}}$$

where $e(u_j, r)$ is an edge correction weight designed so that $\hat{F}(r)$ is approximately unbiased [13].

A diffent formulation of this estimator where the edge correction is taken into account is as follows: Let (x_i, e_i) for $i = 1, \ldots, m$ where x_i denote the distance from each of the m sample points to the nearest event in W and e_i denote the distance to the nearest point on the boundary of W. Then this estimator can be defined as

$$\hat{F}(r) = \frac{\#(x_i \leq r \text{ and } e_i > r)}{\#(e_i > r)}$$

where # can be read as 'The number of points such that' [13].

Since we assume the point process is homogeneous (stationary), $d(u_j, X) > r$ will be true if there are no points of X in the disc $b(u, r)$ of radius r centered on u where this statement is a if and only statement.. Consider a homogeneous Poisson process with intensity parameter λ. The number of points that fall in $b(u, r)$ then follows a Poisson distributed with mean μ where

$$\begin{aligned} \mu &= \lambda \text{area}(b(u, r)) \\ &= \lambda \pi r^2 \end{aligned}$$

such that $P[0 \text{ points in the region }] = e^{-\mu} = e^{-\lambda \pi r^2}$. For a homogeneous (stationary) Poisson process with the intensity λ , the empty space distance distribution function is defined as

$$F_{Poi}(r) = 1 - e^{-\hat{\lambda} \pi r^2}.$$

Since we now have a theoretical (null hypothesis) and empirical distribution function, we can compare the two results for a specific point pattern and then make a conclusion on the spatial randomness of the given point pattern. The conclusion is one of the following:

- If $\hat{F}(r) > F_{Poi}(r)$ then it suggests the point pattern is regular, or
- If $\hat{F}(r) < F_{Poi}(r)$ then it suggests the point pattern is a clustered pattern.

Another formulation for the F-function with the border method taken into account, with the methodology as in Section 3.1, is defined as:

$$\hat{F}^{bm}(r) = \frac{|W_- \cap X|_2}{|W_-|_2}$$

where $|.|_2$ is simply the area of the set.

Example 1. First consider the data set *japanesepines*. To run the F-function, we use the following code:

```
library(spatstat)
data(japanesepines)
plot(Fest(japanesepines,correction=c("border","none")))
```

which results in the following image:

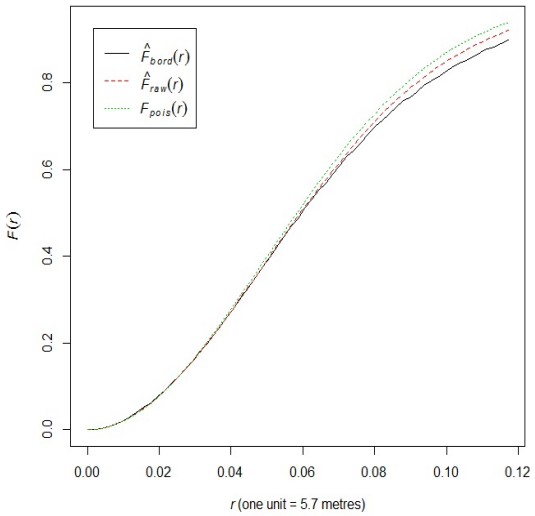


Figure 10: F-function on *japanesepines*

On the x-axis is the parameter, r and on the y-axis is the value for the F-function. $\hat{F}_{bord}(r)$ is the F-function with the border method edge correction, $F_{pois}(r)$ is a theoretical poisson point process and $\hat{F}_{raw}(r)$ is the uncorrected F-function of the data.

From Figure 10, we see that $\hat{F}(r) \approx F_{Poi}(r)$ for both cases. We see that for a small r, the 3 different methods are very close to each other but as r increases (greater than 0.05) we see that $F_{pois}(r)$ becomes bigger. So as r increases the F-function becomes less informative. Thus we can make a conclusion that the Japanese Pines data set is a random point process.

For our second data set, *Redwood*, we run the same code and the output is shown in Figure 11.

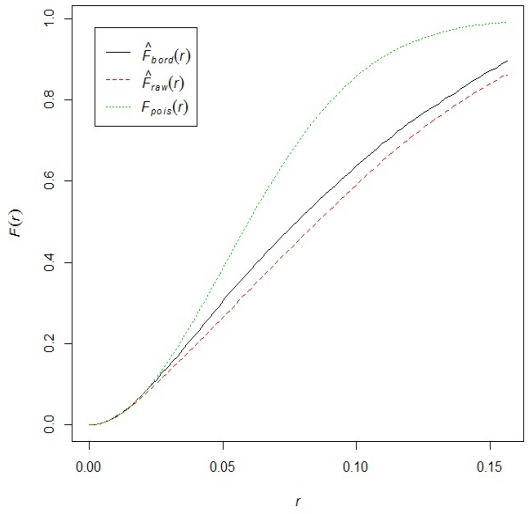


Figure 11: F-function on *redwood*

From Figure 11, we see that for r between 0 and 0.02 all three methods are closely related, but for r greater than 0.03 we see that $\hat{F}(r) < F_{Poi}(r)$ for both cases.Thus for a small r, the graph gives the impression of a random point process but as r increases that conlusion is not valid. Thus we can make a conclusion that the Redwood data set is a clustered point process.

For our third data set, *Swedish Pines*, we run the same code where the output is shown in Figure 12.

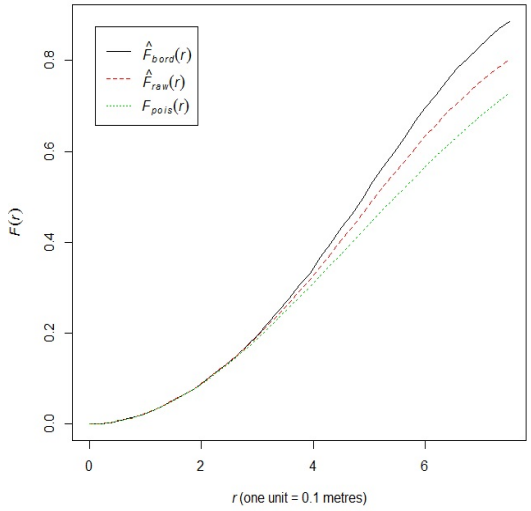


Figure 12: F-function on *swedishpines*

From Figure 12, we see that $\hat{F}(r) > F_{Poi}(r)$ from $r = 3$ onwards for both cases. We also observe the uncorrected graph is the closer to the theoretical process than the border correction. Thus we can make a conclusion that the Swedish Pines data set is a regular point process.

3.3 The G-Function

Assume X is a stationary point process. The G-function, developed by Baddeley [3], uses the cumulative distribution function of the nearest-neighbour distance for a point process and is defined in [2] as:

$$G(r) = P[d(u, X \setminus \{u\}) \leq r | u \in X]$$

where $d(u, X \setminus \{u\}) = \min\{\|u - x_i\| : x_i \in X\}$ is the shortest distance from u to the point pattern X excluding u, where u is some arbitrary location and r is some distance. Since we assume X is a stationary point process, this definition is valid for all u.

The empirical nearest-neighbour distance function can be estimated by

$$G^*(r) = \frac{1}{n(x)} \sum_i 1_{\{t_i \leq r\}}.$$

This estimator for $G(r)$ is negatively biased due to edge effects. A estimator where edge effects are taken into account is given by

$$\hat{G}(r) = \sum_i e(x_i, r) 1_{\{t_i \leq r\}}$$

where $e(x_i, r)$ is an edge correction weight designed so that $\hat{G}(r)$ is approximately unbiased [2].

A different formulation of this estimator where the edge correction is taken into account is as follows: Let (y_i, d_i) for $i = 1, \ldots, m$ where y_i denote the distance from each of the m sample points to the nearest other event in W and d_i denote the distance to the nearest point on the boundary of W. Then this estimator can be defined as

$$\hat{G}(r) \quad = \quad \frac{\#(y_i \leq r \text{ and } d_i > r)}{\#(d_i > r)}$$

where # can be read as 'The number of points such that' [13].

For a homogeneous (stationary) Poisson process with the intensity λ, the nearest-neighbour distance distribution function is defined as

$$G_{Poi}(r) = 1 - e^{-\hat{\lambda}\pi r^2}.$$

As we can see, this is equivalent to the F-function, but the interpretation of the results are different.This fact is only valid for the homogeneous case. This will be made clearer when introducing the J-function.

Since we now have a theoretical (null hypothesis) and empirical distribution function, we can compare the two results for a specific point pattern and then make a conclusion on the spatial randomness of the given point pattern. The conclusion is one of the following:

- If $\hat{G}(r) > G_{Poi}(r)$ then it suggests the point pattern is clustered, or
- If $\hat{G}(r) < G_{Poi}(r)$ then it suggests the point pattern is a regular pattern.

What this means is that if the $\hat{G}(r) > G_{Poi}(r)$ then it means that the inter-distances in the point pattern is shorter than those of the Poisson process, where as $\hat{G}(r) < G_{Poi}(r)$ suggests the opposite. When the empirircal and theoretical are deemed equivalent then the point pattern is said to be completely spatially random and thus further spatial analysis not justified.

Another formulation for the G-function with the border method taken into account, with the methodology as in Section 3.1, is defined as:

$$\hat{G}^{bm}(r) = \frac{\sum_{x \in X \cap W_-} 1_{\{\rho(x, X \setminus \{x\}) \leq r\}}}{X(W_-)}$$

where $\rho(x, W) = \inf\{||x - a|| : a \in W\}$.

Example 2. First consider the data set *japanesepines*. To run the G-function, we use the following code:

```
library(spatstat)
data(japanesepines)
plot(Gest(japanesepines,correction=c("border","none")))
```

This results in the following image:

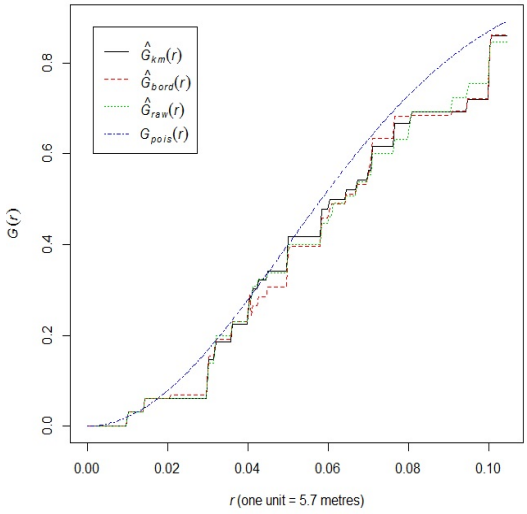


Figure 13: G-function on *japanesepines*

On the x-axis is the parameter, r and on the y-axis is the value for the G-function. $\hat{G}_{km}(r)$ is the G-function with the Kaplan Meier correction, $\hat{G}_{bord}(r)$ is the G-function with the border method correction, $\hat{G}_{raw}(r)$ is the uncorrected G-function and $G_{pois}(r)$ is the theoretical poisson point process.

From Figure 13, we see that the three corrections are close to the theoretical process. Mostly on the graph we notice that the corrections are below the theoretical, but still reseanobly close. Thus we can make a conclusion that the Japanese Pines data set is a random point process.

For our second data set, *Redwood*, we run the same code and illustrate the results in Figure 14.

From this, we see that $\hat{G}(r) > G_{Poi}(r)$ after $r = 0.02$. From 0.10 onwards, the corrections starts to tend back to the theoretical. Thus we can make a conclusion that the Redwood data set is a clustered point process.

For our third data set, *Swedish Pines*, we run the same code. We observe the results in Figure 15.

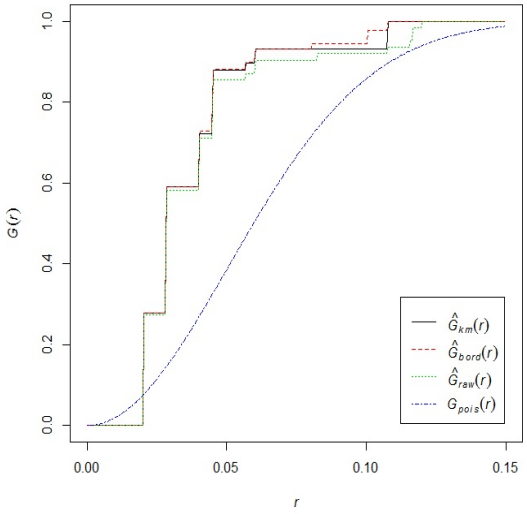


Figure 14: G-function on *redwood*

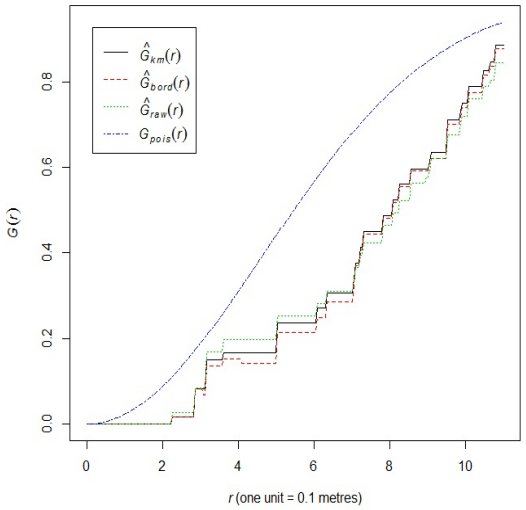


Figure 15: G-function on *swedishpines*

From this, we see that $\hat{G}(r) < G_{Poi}(r)$ for all the different cases, thus we can make a conclusion that the Swedish Pines data set is a regular point process.

3.4 The J-Function

The J-function is a combination of the G-function and the empty space function, F. The J-function, developed by van Lieshout and Baddeley [11], is defined in [2] as:

$$J(r) = \frac{1 - G(r)}{1 - F(r)} \forall r \geq 0$$

such that $F(r) \neq 1$.

For a homogeneous Poisson process $F_{Poi}(r) \equiv G_{Poi}(r)$, so this reduces to

$$J_{Poi}(r) = 1.$$

A estimator where edge effects are taken into account is given by

$$\hat{J}(r) = \frac{1 - \hat{G}(r)}{1 - \hat{F}(r)}$$

where $\hat{G}(r)$ and $\hat{F}(r)$ are as in Sections 3.1 and 3.2.

We can compare the two results for a specific point pattern and then make a conclusion on the spatial randomness of the given point pattern. The conclusion is one of the following:

- If $\hat{J}(r) > J_{Poi}(r)(= 1)$ then it suggest the point pattern is a regular pattern, or
- If $\hat{J}(r) < J_{Poi}(r)(= 1)$ then it suggest the point pattern is clustered.

Another formulation for the J-function with the border method taken into account, with the methodology as in Section 3.1, depends on the formulation for the F-function and G-function.

An appealing property of the J-function is that if you have a superposition of two point processes $X_{\odot} = X_1 \cup X_2$, where X_1 and X_2 are two independent point processes, this superposition has J-function

$$J(r) = \frac{\lambda_1}{\lambda_1 + \lambda_2} J_1(r) + \frac{\lambda_2}{\lambda_1 + \lambda_2} J_2(r)$$

where $J_1(r)$ and $J_2(r)$ are the separate J-functions and λ_1 and λ_2 are the intensities for the separate point processes.

Example 3. First consider the data set *japanesepines*. To run the J-function, we use the following code:

```
library(spatstat)
data(japanesepines)
plot(Jest(japanesepines,correction=c("border","none")))
```

This results in the following image:

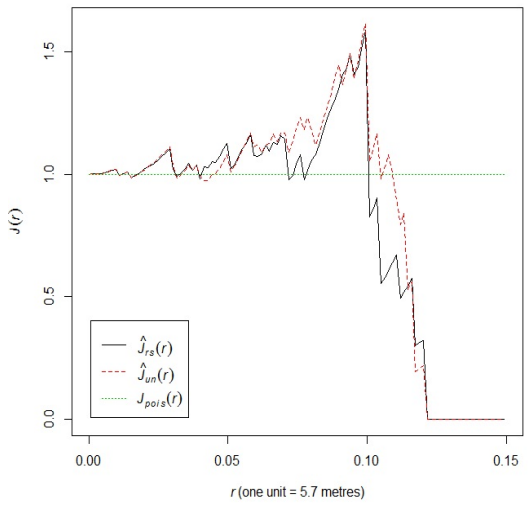


Figure 16: J-function on *japanesepines*

On the x-axis is the parameter, r and on the y-axis is the value for the J-function. $\hat{J}_{rs}(r)$ is the J-function with border method correction, $\hat{J}_{un}(r)$ is the uncorrected J-function and $J_{pois}(r)$ is the theoretical poisson point process.

From Figure 16, we see that $\hat{J}(r) \approx J_{Poi}(r)$ in the beginning for all the different cases. As r increases, the estimates move away from the theoretical process but returns back. After 0.10 all the corrections fall far below the theoretical. Thus we make a conclusion that the Japanese Pines data set is a random point process.

For our second data set, *Redwood*, we run the same code with output shown in Figure 17.

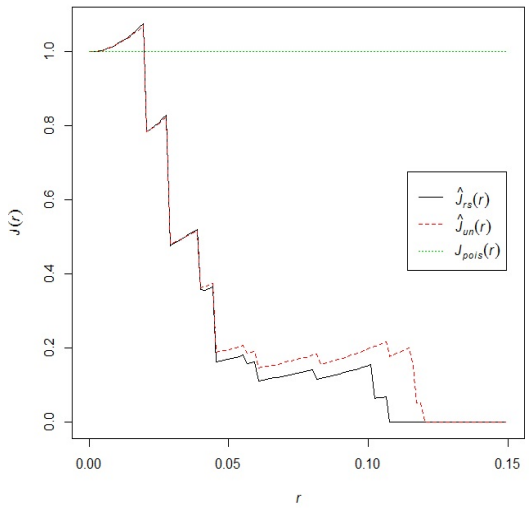


Figure 17: J-function on *redwood*

From this, we see that $\hat{J}(r) < J_{Poi}(r)$ after $r = 0.025$ for all the different corrections. We notice that the estimates falls far below the theoretical process, thus we can make a conclusion that the Redwood data set is a clustered point process.

For our third data set, *Swedish Pines*, we run the same code which results in Figure 18.

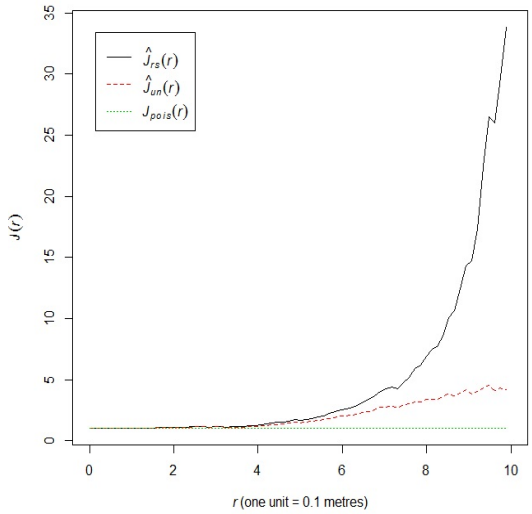


Figure 18: J-function on *swedishpines*

From this, we see that $\hat{J}(r) > J_{Poi}(r)$ for all the different corrections from $r = 4$. We also observe that the uncorrected graph stays reasonably close to the theoretical. Thus we can make a conclusion that the Swedish Pines data set is a regular point process.

3.5 The K-Functon

Assume X is a stationary point process. The K-function is defined such that $\lambda K(r)$ is the expected number of points of the process within a distance r of a typical point of the process. So the K-function, developed by Ripley [13], is defined in [2]as:

$$K(r) = \frac{1}{\lambda} E[N(X \cap b(u, r) \setminus \{u\}) | u \in X].$$

For a homogeneous Poisson process, it is unimportant if u is a point of X since this does not effect the other points of the process. So the expected number of points falling in $b(u, r)$ is $\lambda \pi r^2$. For a homogeneous Poisson process, the K-function is thus defined as:

$$K_{Poi}(r) = \pi r^2$$

which is not dependent on the intensity.

The (renormalized) empirical distribution function of the pairwise distances is of the general form

$$\hat{K}(r) = \frac{1}{\hat{\lambda}^2 \text{area}(W)} \sum_i \sum_{j \neq i} 1_{\{||x_i - x_j|| \leq r\}} e(x_i, x_j, r)$$

where $e(x_i, x_j, r)$ is an edge correction weight designed so that $\hat{K}(r)$ is approximately unbiased [2].

The edge correction formulation in this case depends on the shape of the study area W [8].

- If W is the rectangle $(0, a) \times (0, b)$, we write our observed event x as $x = (x_1, x_2)$. Let $d_1 = \min(x_1, a - x_2)$ and $d_2 = \min(x_2, b - x_2)$ where d_1 and d_2 are the distances to the nearest vertical and horizontal edges of W. There are two cases we need to consider:
  - if $r^2 \leq d_1^2 + d_2^2$ then $e(x, r) = 1 - \pi^{-1}[cos^{-1}\{\frac{\min(d_1, r)}{r}\} + cos^{-1}\{\frac{\min(d_2, r)}{r}\}$
  - if if $r^2 > d_1^2 + d_2^2$ then $e(x, r) = 0.75 - (2\pi)^{-1}[cos^{-1}\{\frac{d_1}{r}\} + cos^{-1}\{\frac{d_2}{r}\}]$.
    - if $r^2 \leq d_1^2 + d_2^2$, note that $e(x, r) = 1$ when $r \leq \min(d_1, d_2)$. These formulations only hold if the values of r is in the range $0 \leq r \leq 0.5\min(a, b)$.
- If W is a disc centered at the origin with radius a, let $R_{rad} = \sqrt{x_1^2 + x_2^2}$ be the distance from x to the centre of the disc (in other words, to the origin). There are two cases we need to consider:
  - if $r^2 \leq a - R_{rad}$ then $e(x, r) = 1$
  - if $r^2 > a - R_{rad}$ then $e(x, r) = 1 - \pi^{-1}[cos^{-1}\{\frac{a^2 - R_{rad} - r^2}{2rR_{rad}}\}]$
    - These formulas holds for the values of r between 0 and a.

Since we now have a theoretical (null hypothesis) and empirical distribution function, we can compare the two results for a specific point pattern and then make a conclusion on the spatial randomness of the given point pattern. The conclusion is one of the following:

- If $\hat{K}(r) > K_{Poi}(r)$ then it suggests the point pattern is clustered, or
- If $\hat{K}(r) < K_{Poi}(r)$ then it suggests the point pattern is a regular pattern.

Another formulation for the K-function with the border method taken into account, with the methodology as in Section 3.1, is defined as:

$$\hat{K}^{bm}(r) = \frac{\sum_{x \in X \cap W_-} X(b(x,r) \backslash \{x\}}{\frac{X(W)}{|W|_2} X(W_-)}$$

where $b(x,r)$ is the ball centered at x with radius r and $|.|_2$ is simply the area of the set.

Example 4. First consider the data set *japanesepines*. To run the K-function, we use the following code:

```
library(spatstat)
data(japanesepines)
plot(Kest(japanesepines,correction=c("border","iso","none")))
```

This results in the following image:

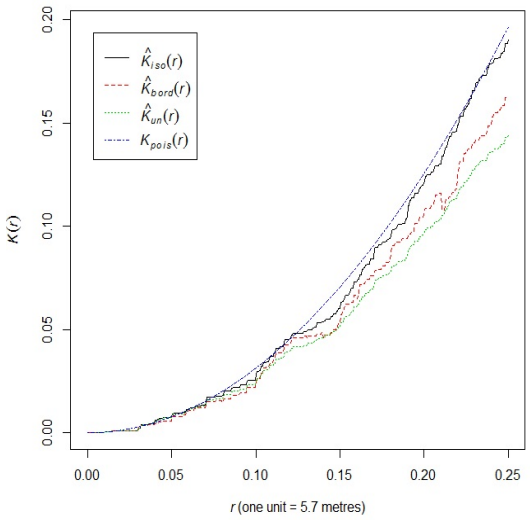


Figure 19: K-function on *japanesepines*

On the x-axis is the parameter, r and on the y-axis is the value for the K-function. $\hat{K}_{iso}(r)$ is the K-function with Ripley's Isotropic correction, $\hat{K}_{bord}(r)$ is the K-function with the border method correction, $\hat{K}_{un}(r)$ is the uncorrected K-function and $K_{pois}(r)$ is the theoretical poisson point process.

From Figure 19, we see that $\hat{K}(r) \approx K_{Poi}(r)$ for all the different corrections. As r becomes larger we observe that the corrections move away from the theoretical process, except for the isotropic correction. Thus we can make a conclusion that the Japanese Pines data set is a random point process.

For our second data set, *Redwood*, we run the same code. This output is shown in Figure 20.

From this, we see that $\hat{K}(r) > K_{Poi}(r)$ for r greater than 0.20 we observe that the estimates approach the theoretical process. Thus we can make a conclusion that the Redwood data set is a clustered point process.

For our third data set, *Swedish Pines*, we run the same code which results in Figure 21.

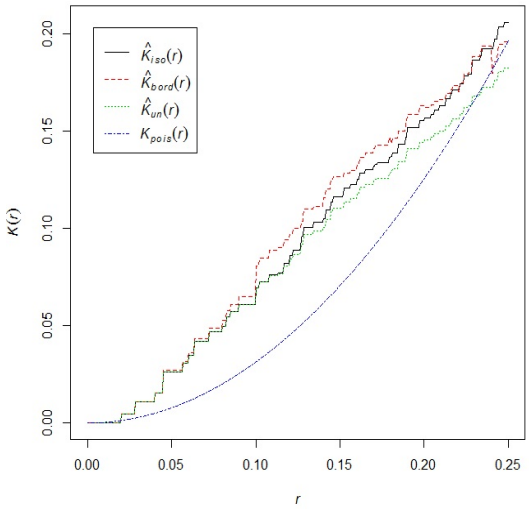


Figure 20: K-function on *redwood*

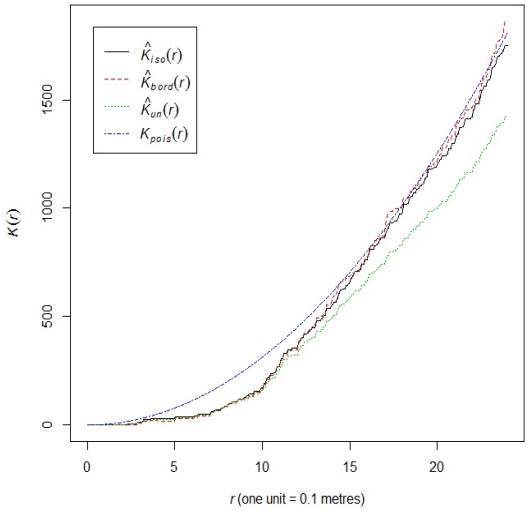


Figure 21: K-function on *swedishpines*

From this, we see that $\hat{K}(r) \lesssim K_{Poi}(r)$, so the graph gives the impression that Swedish Pines is a random point process.

3.6 The L-Function

A commonly used transformation of the K-function is known as the L-function where the L-function, developed by Ripley [13], is defined in [2] as:

$$L(r) = \sqrt{\frac{K(r)}{\pi}}.$$

The reason why this transformation is used is because the K-function is transformed into the straight line $L_{Poi}(r) = r$, where the interpretation of this new graph is more simple than interpreting other graphs. The square root in this formula also helps to stabilize the variance of the estimator.

A estimator where edge effects are taken into account is given by

$$\hat{L}(r) = \sqrt{\frac{\hat{K}(r)}{\pi}}$$

where $\hat{K}(r)$ is as in Section 3.5.

Example 5. First consider the data set *japanesepines*. To run the L-function, we use the following code:

```
library(spatstat)
data(japanesepines)
plot(Lest(japanesepines,correction=c("border","iso","none")))
```

This results in the following image:

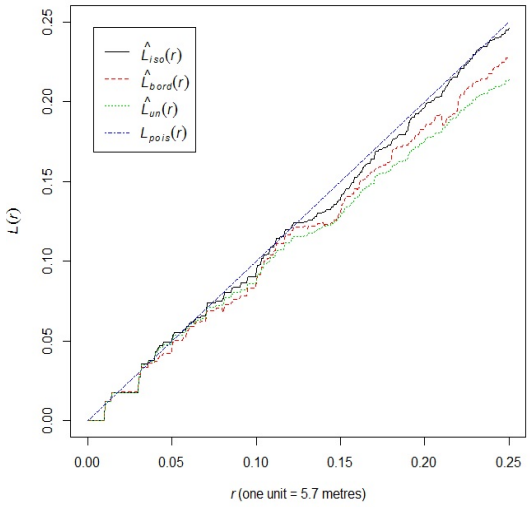


Figure 22: L-function on *japanesepines*

On the x-axis is the parameter, r and on the y-axis is the value for the L-function. The different estimates are the same as with the K-function, which is identified by the subscripts.

From Figure 22, we see that $\hat{L}(r) \approx L_{Poi}(r)$ for all the different estimates. As r becomes larger we observe that the estimates move away from the theoretical process, except for the isotropic correction. Thus we can make a conclusion that the Japanese Pines data set is a random point process.

For our second data set, *Redwood*, we run the same code which results in Figure 23.

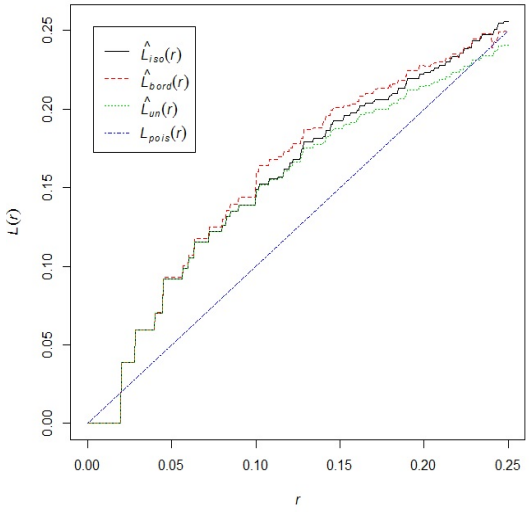


Figure 23: L-function on *redwood*

From this, we see that $\hat{L}(r) > L_{Poi}(r)$. As r increases, the estimates approach the theoretical process. Thus we can make a conclusion that the Redwood data set is a clustered point process.

For our third data set, *Swedish Pines*, we run the same code. This results in Figure 24.

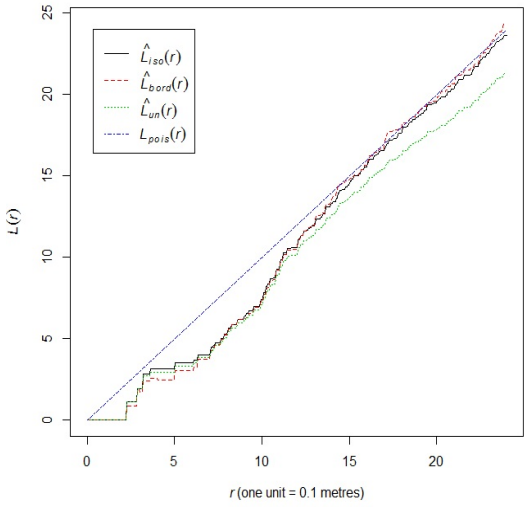


Figure 24: L-function on *swedishpines*

From this, we see that $\hat{L}(r) < L_{Poi}(r)$, but as r increases, the estimates approach the theoretical process.This then gives the impression that Swedish Pines is a random point process.

3.7 Inhomogeneous Spatial Point Process

In a Poisson process, it is not always applicable to assume a constant intensity, as in some processes, the intensity is dependent on some function of time, t. This is denoted by $\lambda(t)$. If we follow the same assumptions as for the homogeneous Poisson process, but with an intensity dependent on time, we get the following results:

If $X(t)$ denotes the number of occurrences is a specified interval, $[0, t]$, then it follows that

$$X(t) \sim POI(\mu(t))$$

where

$$\mu(t) = \int_0^t \lambda(s)ds.$$

To illustate how the functions are interpreted for a inhomogeneous data set, we use the data set *redwoodfull* in $R^{©}$. We run the specific functions on the data set, and we make a conclusion based on the results. First we plot the data set to see if we can make some conclusion based on the plotted data. We use the following coding in $R^{©}$ to import and plot the data set:

```
library(spatstat)
data(redwoodfull)
plot(redwoodfull)
```

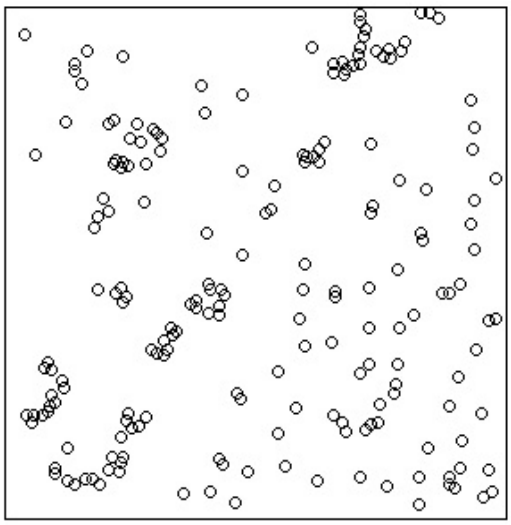


Figure 25: Plot of the inhomogeneous data set

From Figure 25, it is difficult to make a certain conclusion based on this plot since at certain places the points looks clustered and other places it looks regular. So to make an informative conclusion, we run the functions (inhomogeneous) on the data sets.

Example 6. For all the graphs, the parameter r is on the x-axis and the inhomogeneous functions are on the y-axis.

To run the F-function, we use the following code:

```
library(spatstat)
data(redwoodfull)
plot(Finhom(redwoodfull))
```

The output is given in Figure 26.

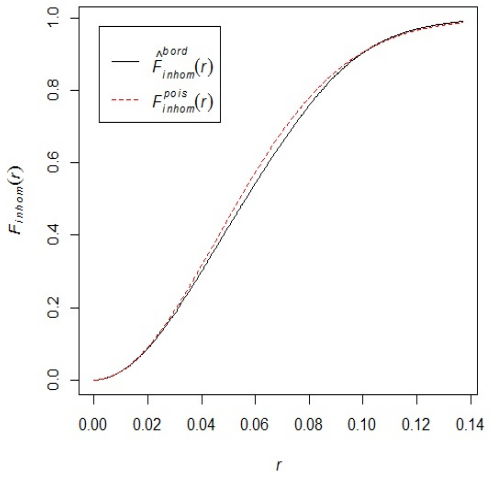


Figure 26: Inhomogeneous F-function

From this, we see that $\hat{F}^{bord}_{inhom}(r) \approx F^{pois}_{inhom}$ where $\hat{F}^{bord}_{inhom}(r)$ is the inhomogeneous F-function with the border method correction and F^{pois}_{inhom} is the theoretical inhomogeneous Poisson process. On the edges we see that the border method correction and the theoretical process is roughly the same but in between we notice that the theoretical is higher.

To run the G-function, we use the following code:

```
library(spatstat)
data(redwoodfull)
plot(Ginhom(redwoodfull))
```

The output is given in Figure 27.

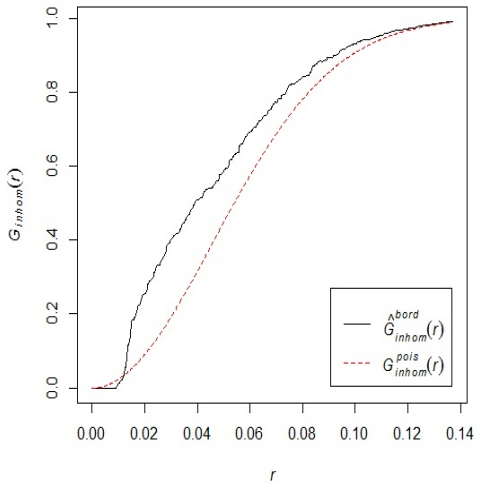


Figure 27: Inhomogeneous G-function

From this, we see that $\hat{G}^{bord}_{inhom}(r) > G^{pois}_{inhom}$ where $\hat{G}^{bord}_{inhom}(r)$ is the inhomogeneous G-function with the border method correction and G^{pois}_{inhom} is the theoretical inhomogeneous Poisson process. For r greater than

0.12 we see that the border method tends to the theoretical process but for less than 0.12 the border correction is higher than the theoretical.

To run the *J*-function, we use the following code:

```
library(spatstat)
data(redwoodfull)
plot(Jinhom(redwoodfull))
plot(density(redwoodfull))
```

The output is given in Figure 28.

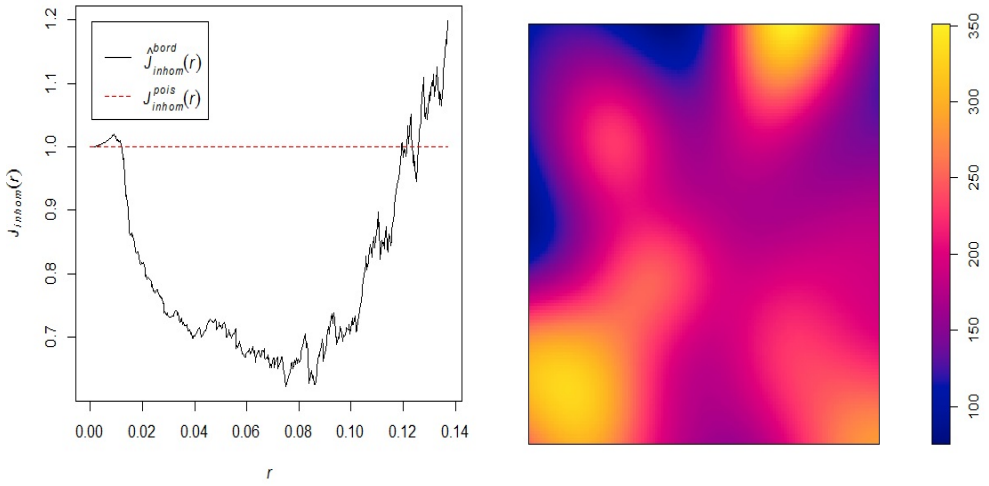


Figure 28: Inhomogeneous *J*-function

From this, we see that $\hat{J}^{bord}_{inhom}(r) > J^{pois}_{inhom}$ in the beginning and end of the graph and $\hat{J}^{bord}_{inhom}(r) < J^{pois}_{inhom}$ in the middle where $\hat{J}^{bord}_{inhom}(r)$ is the inhomogeneous *J*-function with the border method correction and J^{pois}_{inhom} is the theoretical inhomogeneous poisson process. For r less than 0.01 and greater than 0.11 the border correction is above the theoretical process but inbetween the border correction falls far below the the theoretical. We have also added a density mapping of *redwoodfull*.

To run the *K*-function, we use the following code:

```
library(spatstat)
data(redwoodfull)
plot(Kinhom(redwoodfull,correction=c("bord","iso","none")))
```

The output is given in Figure 29.

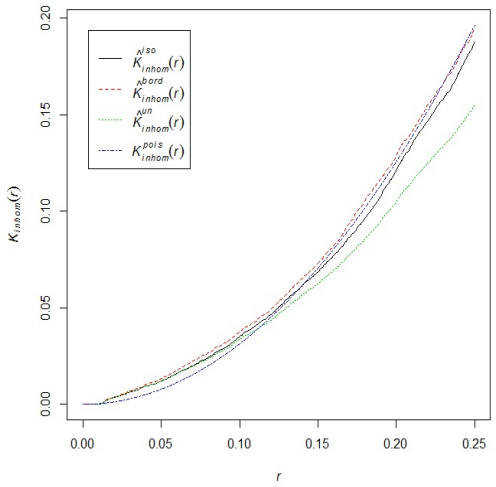


Figure 29: Inhomogeneous K-function

From this we see that $\hat{K}_{inhom}(r) \approx K^{pois}_{inhom}$where $\hat{K}_{inhom}(r)$ is the K-function with the different edge corrections. $\hat{K}^{iso}_{inhom}$ is the inhomogeneous K-function with Ripley's isotropic correction, $\hat{K}^{bord}_{inhom}$is the inhomogeneous K-function with border method correction, $\hat{K}^{pois}_{inhom}$ is the theoretical inhomogeneous Poisson process and $\hat{K}^{un}_{inhom}$ is the uncorrected inhomogeneous$K$-function. We observe that all corrections are reasonable close to the theoretical inhomogeneous poisson process. We also notice that the isotropic correction is the closest to the theoretical process.

To run the L-function, we use the following code:

```
library(spatstat)
data(redwoodfull)
plot(Linhom(redwoodfull,correction=c("bord","iso","none")))
```

The output is given in Figure 30.

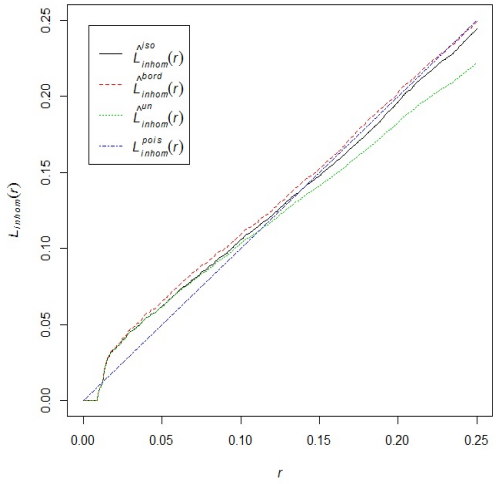


Figure 30: Inhomogeneous L-function

From this we see that $\hat{L}_{inhom}(r) \approx L^{pois}_{inhom}$ where $\hat{L}_{inhom}(r)$ is the L-function with the different edge corrections. The different corrections are identified by the superscripts and is exactly the same as for the K-function. From all the different graphs except for the J-funnction, we get the same results which indicate to us that the data set Redwoodfull is indeed a random inhomogenous poisson process.

4 LULU operators and the Discrete Pulse Transform

As we have explained most of the theory we are going to use, we can move over to our application. We first define the LULU operators as follows [1]. For $f \in A(\mathbb{Z}^2)$, a vector lattice, and $n \in \mathbb{N}$,

$$L_n(f)(x) = \max_{V \in \mathcal{N}_n(x)} \min_{y \in V} f(y), \quad x \in \mathbb{Z}^2$$

$$U_n(f)(x) = \min_{V \in \mathcal{N}_n(x)} \max_{y \in V} f(y), \quad x \in \mathbb{Z}^2$$

where

$$\mathcal{N}_n(x) = \{V \in C : x \in V, \mathrm{card}(V) = n+1\}$$

and where C is a connection defined as follows. Let B be an arbitrary non-empty set. A family C of subsets of B is called a connected class or a **connection** on B if

1. $\emptyset \in C$
2. $\{x\} \in C$ for all $x \in B$
3. for any family $\{C_i\} \subseteq C$ we have $\bigcap_{i \in I} C_i \neq \emptyset \Longrightarrow \bigcup_{i \in I} C_i \in C$

If a set C belongs to a connection C then C is called **connected**. The LULU operators act only on certain types of sets, namely local maximum and minimum sets. A connected subset $V \in \mathbb{Z}^d$ is called a **local maximum set** of $f \in A(\mathbb{Z}^d)$ if

$$\sup_{y \in \mathrm{adj}(V)} f(y) < \inf_{x \in V} f(x).$$

Similarly V is a **local minimum set** if

$$\inf_{y \in \mathrm{adj}(V)} f(y) > \sup_{x \in V} f(x).$$

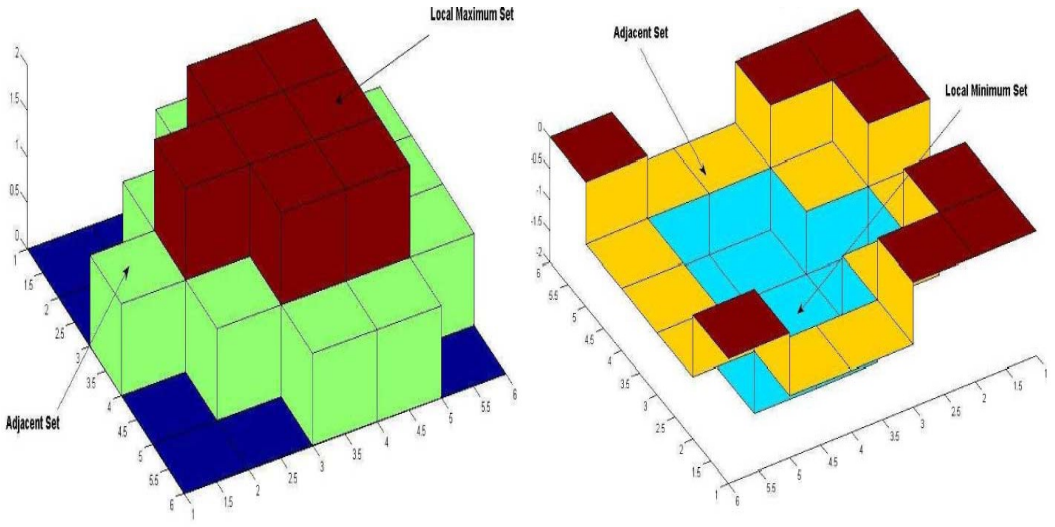


Figure 31: Example of a local maximum and minimum set respectively [1]

Figure 31 illustrates this concept.
The LULU operators act on the local maximum and minimum sets as follows [1]:

- The application of L_n (U_n) removes the local maximum (minimum) sets of size $\leq n$.
- The operator L_n (U_n) does not affect the local minimum (maximum) sets directly in the sense that such sets may be affected only as a result of the removal of local maximum (minimum) sets.
  - No new local maximum (minimum) sets are created if there were no local maximum (minimum) sets.
  - Consider the action of L_n (U_n). This may enlarge existing local maximum (minimum) sets. Joining two or more local maximum (minimum) sets of f into one local maximum (minimum) set of $L_n(f)$ ($U_n(f)$) may also enlarge existing local maximum (minimum) sets.
- $L_n(f) = f$ ($U_n(f) = f$) if f does not have local maximum (minimum) sets of size $\leq n$. Again, this is a if and only statement.

As a result of these actions by the LULU operator any f can be decomposed into a number of pulses ϕ_{ns}:

- The Discrete Pulse Transform (DPT) is obtained via iterative application of the operators L_n, U_n with n increasing from 1 to N.
  - $P_n = L_n \circ U_n$ or $P_n = U_n \circ L_n$
  - $Q_n = P_n \circ P_{n-1} \circ \ldots \circ P_1$
- At each iteration we retain the portions of the image which are filtered out by the application of $P_n(f), n = 1, 2, \ldots, N$ i.e. $(I - P_n)(f) = D_n(f)$, until we obtain $Q_N(f)$, a constant function\image.
- The function f is then decomposed as:

$$f = \sum_{n=1}^{N} D_n(f) = \sum_{n=1}^{N} \sum_{s=1}^{\gamma(n)} \phi_{ns}.$$

4.1 Application

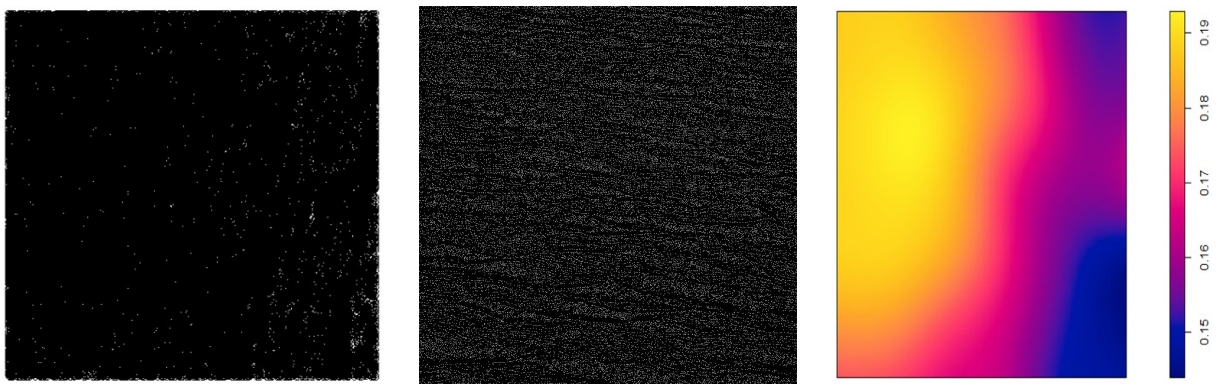
Now that we have discussed all the theoretical aspects, we can move over to the application of the theory to a image. For our image, we use a water image[2]. This image can be seen in Figure 32.

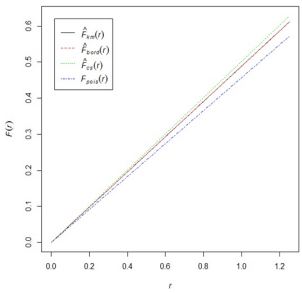
[2]Image was obtained from http://www.ux.uis.no/~tranden/brodatz/D37.gif

Figure 32: Water Image

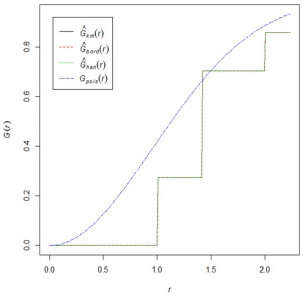
Applying the Discrete Pulse Transformation for ten scales results in the Figures 33 to 42. Take note that included in this output is also the density plot for each of the scales (from scale 1 up to 10). We also apply each of the functions (homogeneous and inhomogeneous) discussed in Section 3.

Scale 1

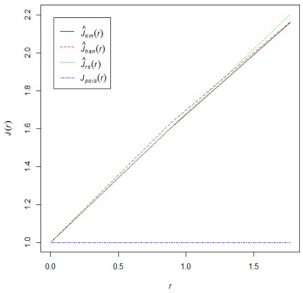




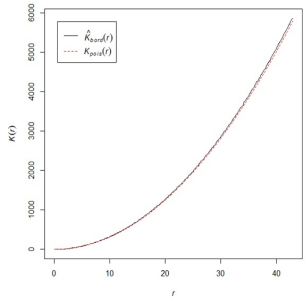
(a) F- Function



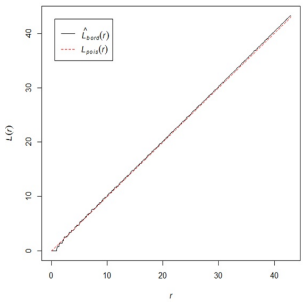
(b) G - Function



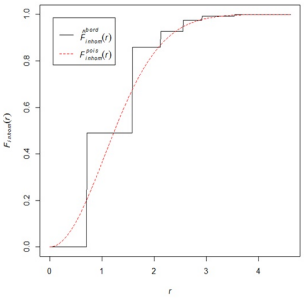
(c) J - Function



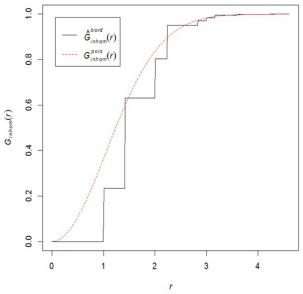
(d) K - Function



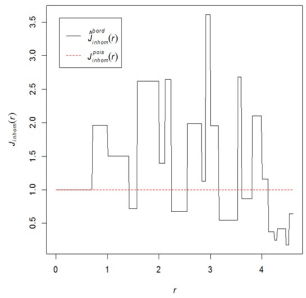
(d) L - Function



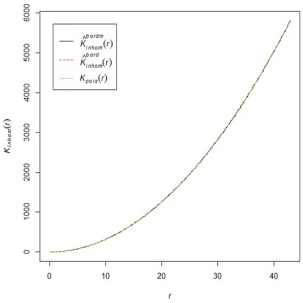
(e) Inh. F - Function



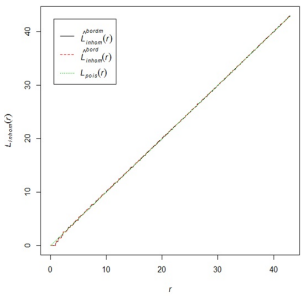
(f) Inh. G - Function



(g) Inh. J - Function



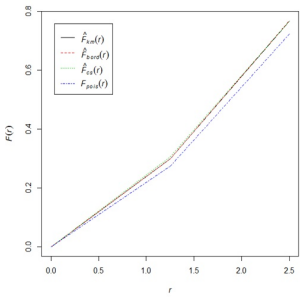
(g) Inh. K - Function



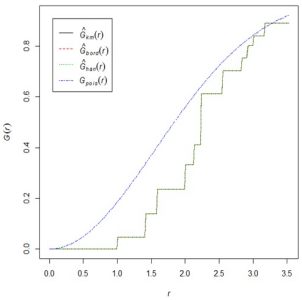
(h) Inh. L - Function

Figure 33: Scale 1

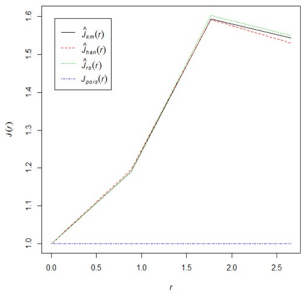
Scale 2



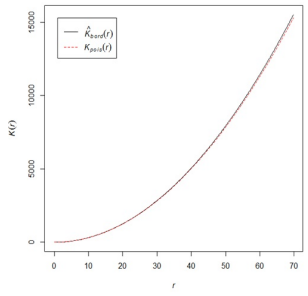
(a) F- Function



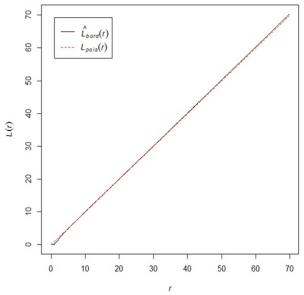
(b) G - Function



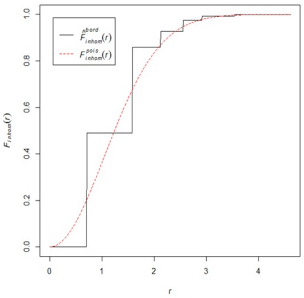
(c) J - Function



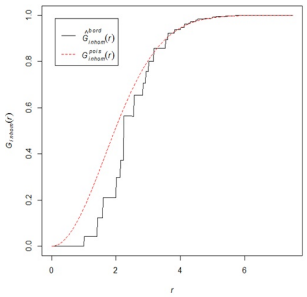
(d) K - Function



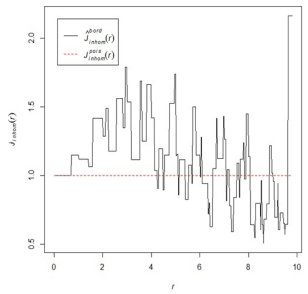
(d) L - Function



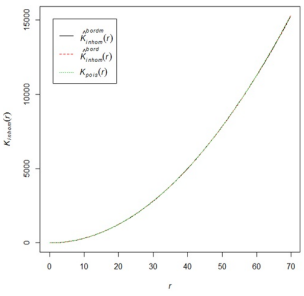
(e) Inh. F - Function



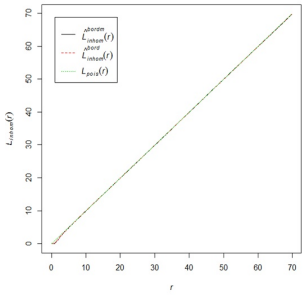
(f) Inh. G - Function



(g) Inh. J - Function



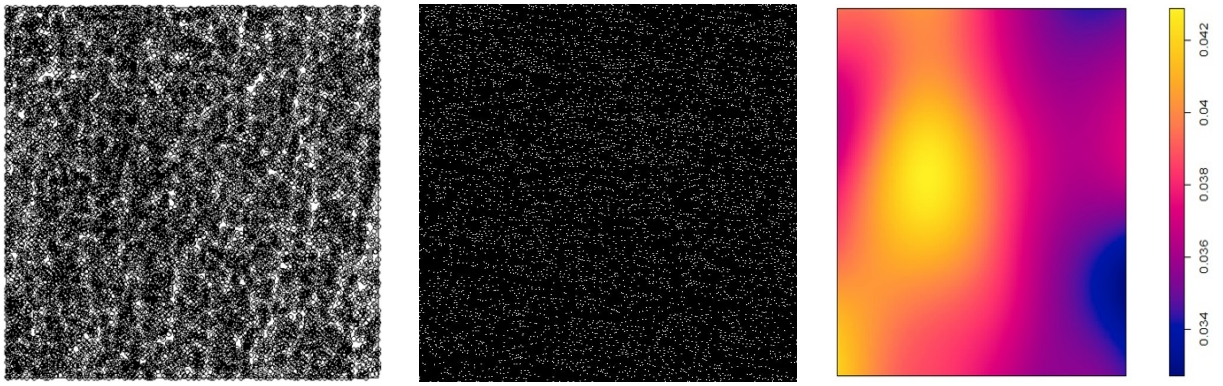
(g) Inh. K - Function

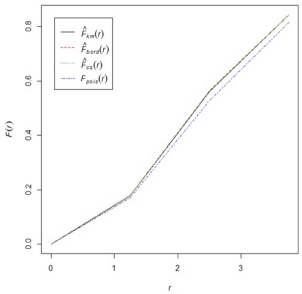


(h) Inh. L - Function

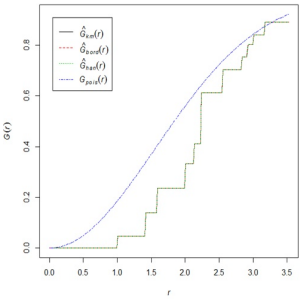
Figure 34: Scale 2

Scale 3

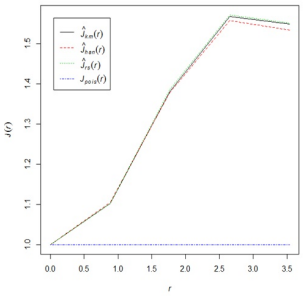




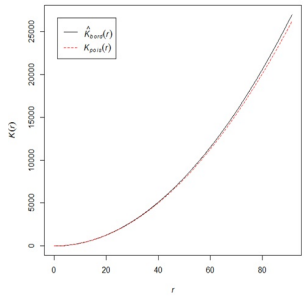
(a) F- Function



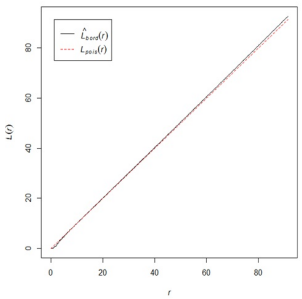
(b) G - Function



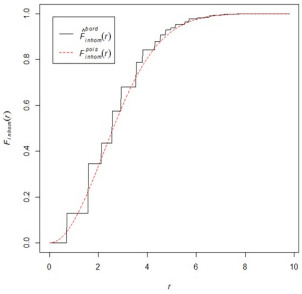
(c) J - Function



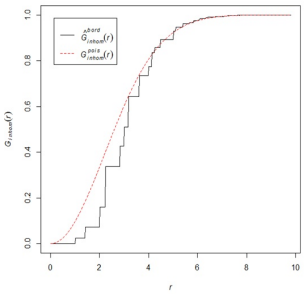
(d) K - Function



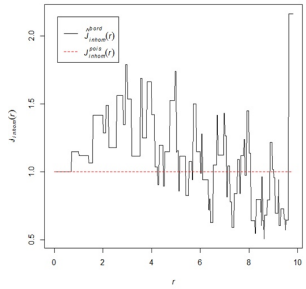
(d) L - Function



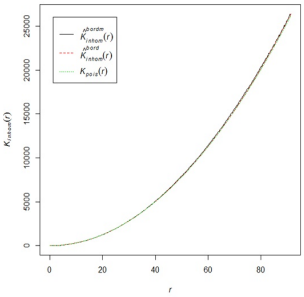
(e) Inh. F - Function



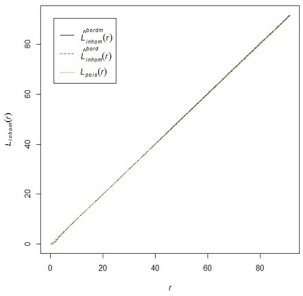
(f) Inh. G - Function



(g) Inh. J - Function



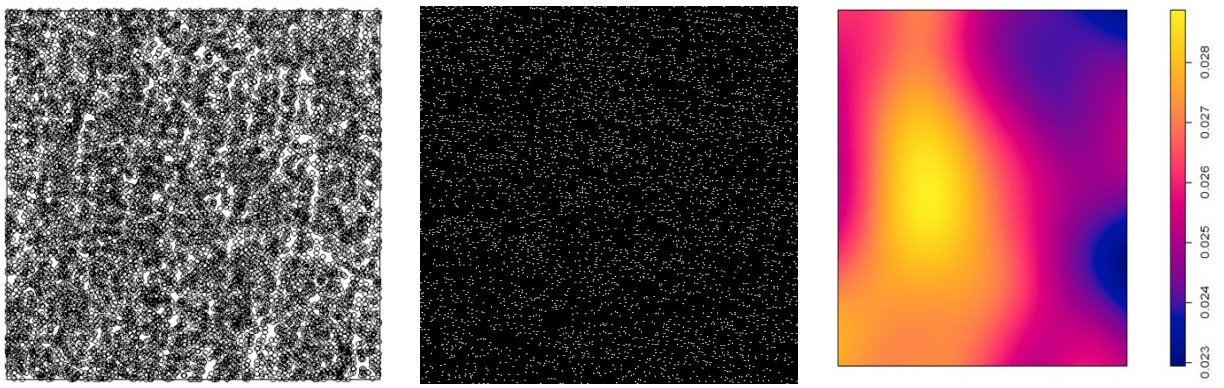
(g) Inh. K - Function

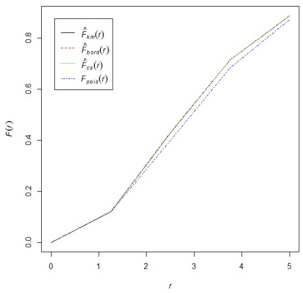


(h) Inh. L - Function

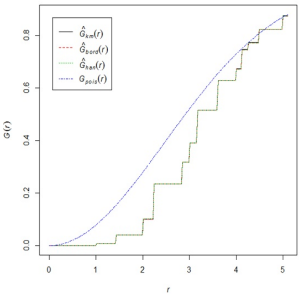
Figure 35: Scale 3

Scale 4

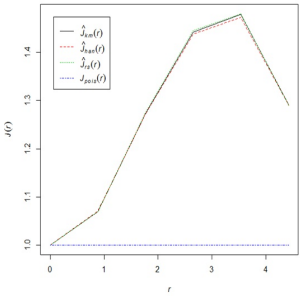




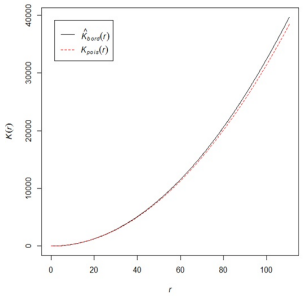
(a) F- Function



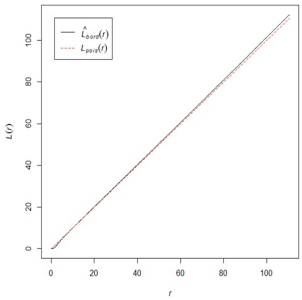
(b) G - Function



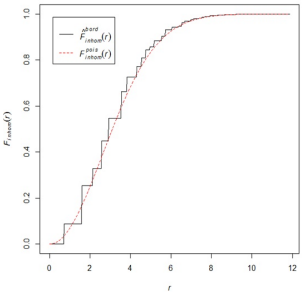
(c) J - Function



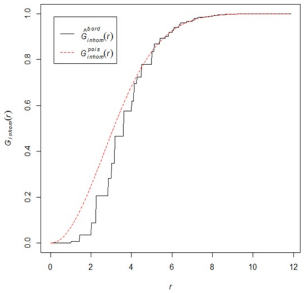
(d) K - Function



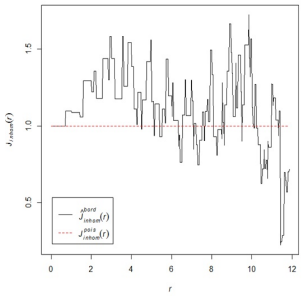
(d) L - Function



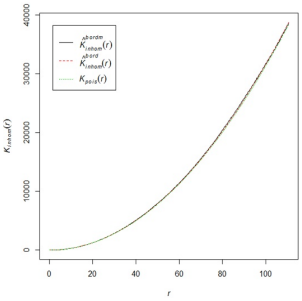
(e) Inh. F - Function



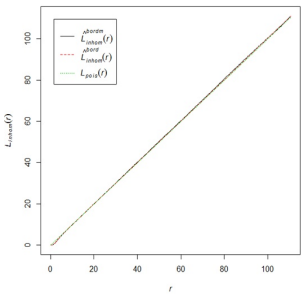
(f) Inh. G - Function



(g) Inh. J - Function



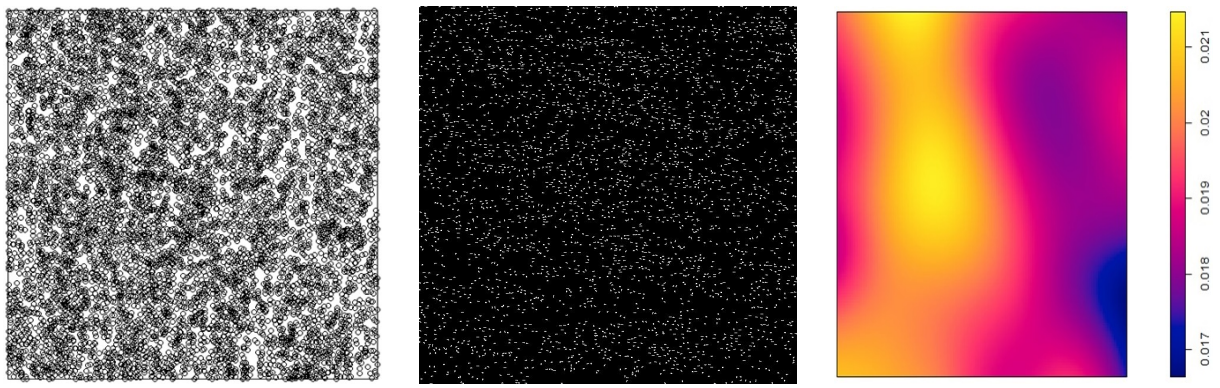
(g) Inh. K - Function

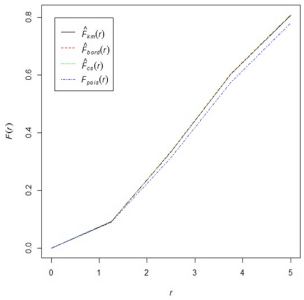


(h) Inh. L - Function

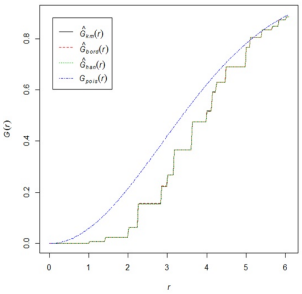
Figure 36: Scale 4

Scale 5

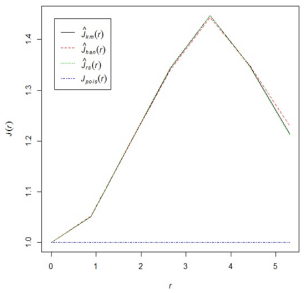




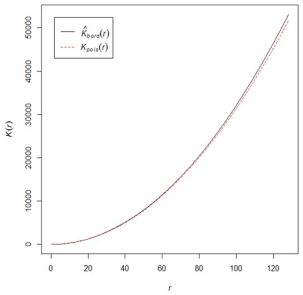
(a) F- Function



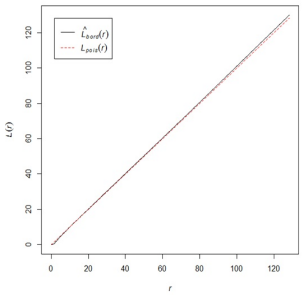
(b) G - Function



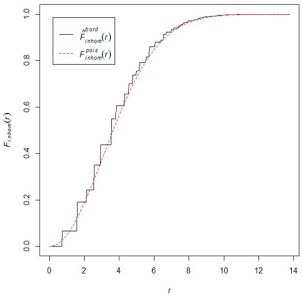
(c) J - Function



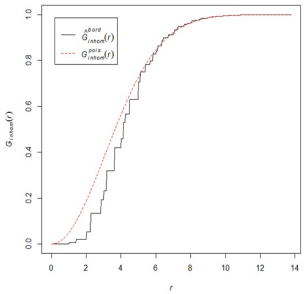
(d) K - Function



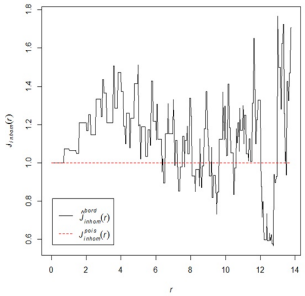
(d) L - Function



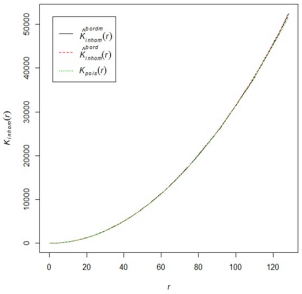
(e) Inh. F - Function



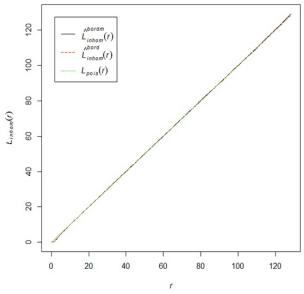
(f) Inh. G - Function



(g) Inh. J - Function



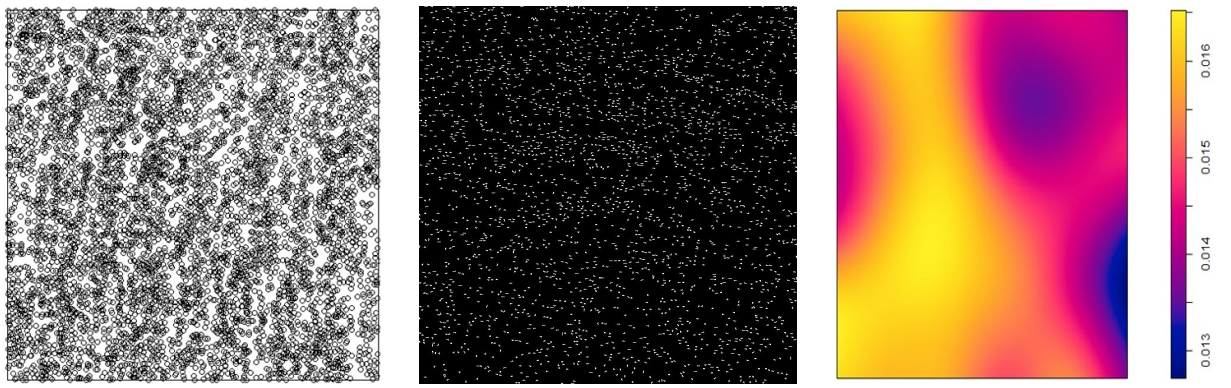
(g) Inh. K - Function

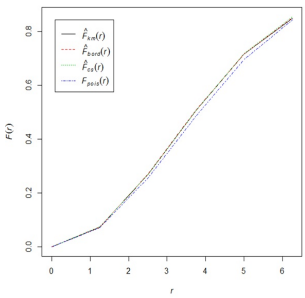


(h) Inh. L - Function

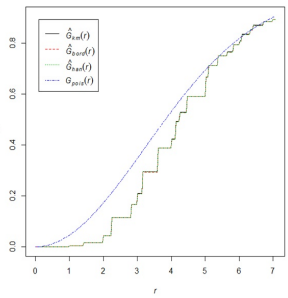
Figure 37: Scale 5

Scale 6

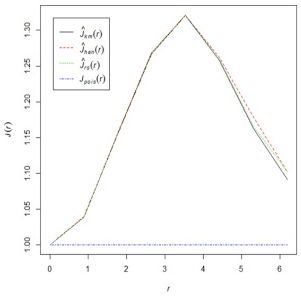




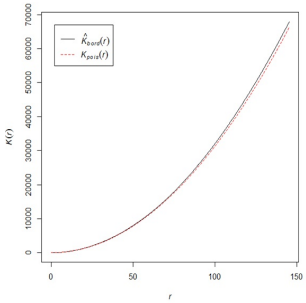
(a) F- Function



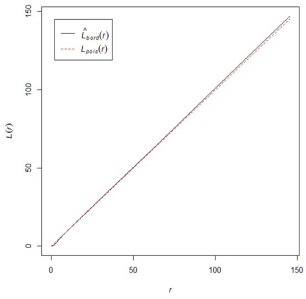
(b) G - Function



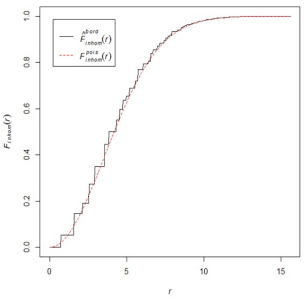
(c) J - Function



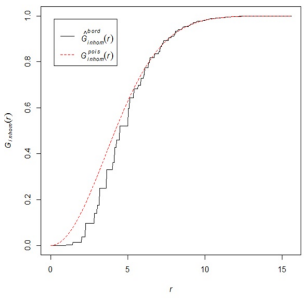
(d) K - Function



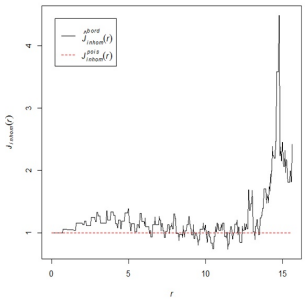
(d) L - Function



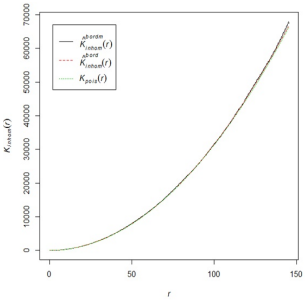
(e) Inh. F - Function



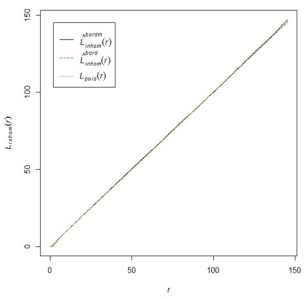
(f) Inh. G - Function



(g) Inh. J - Function



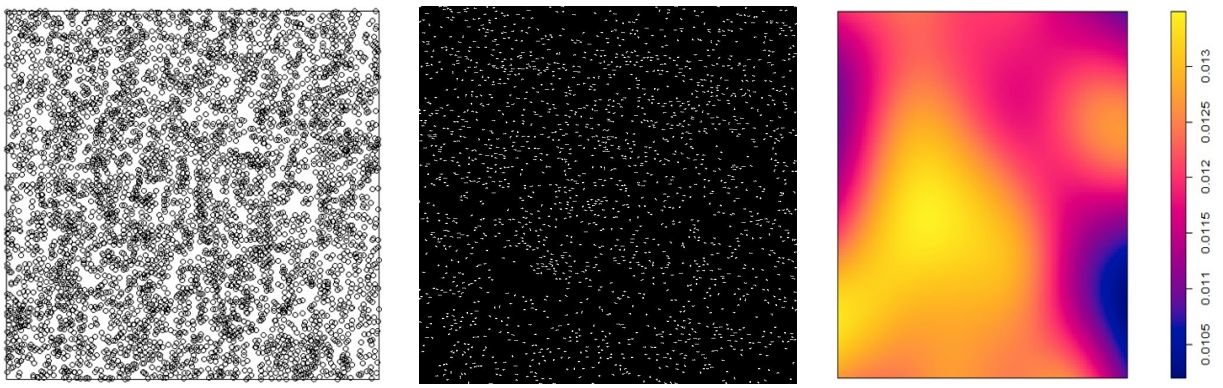
(g) Inh. K - Function

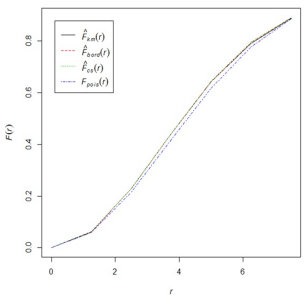


(h) Inh. L - Function

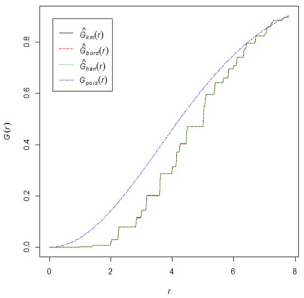
Figure 38: Scale 6

Scale 7

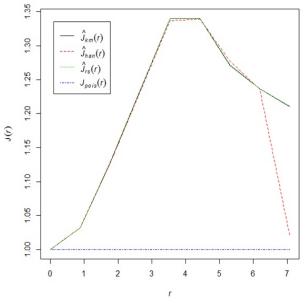




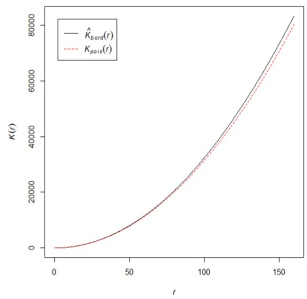
(a) F- Function



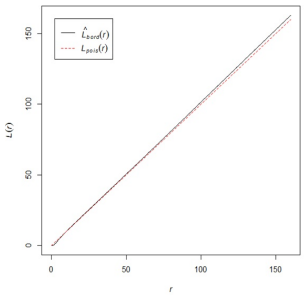
(b) G - Function



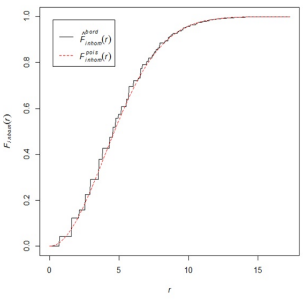
(c) J - Function



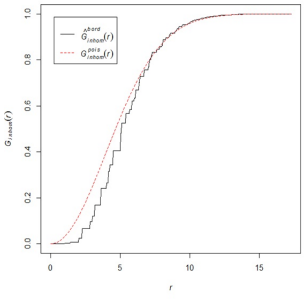
(d) K - Function



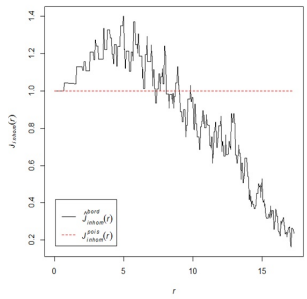
(d) L - Function



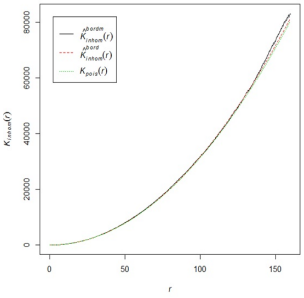
(e) Inh. F - Function



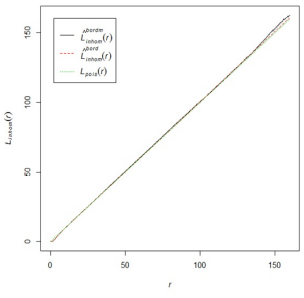
(f) Inh. G - Function



(g) Inh. J - Function



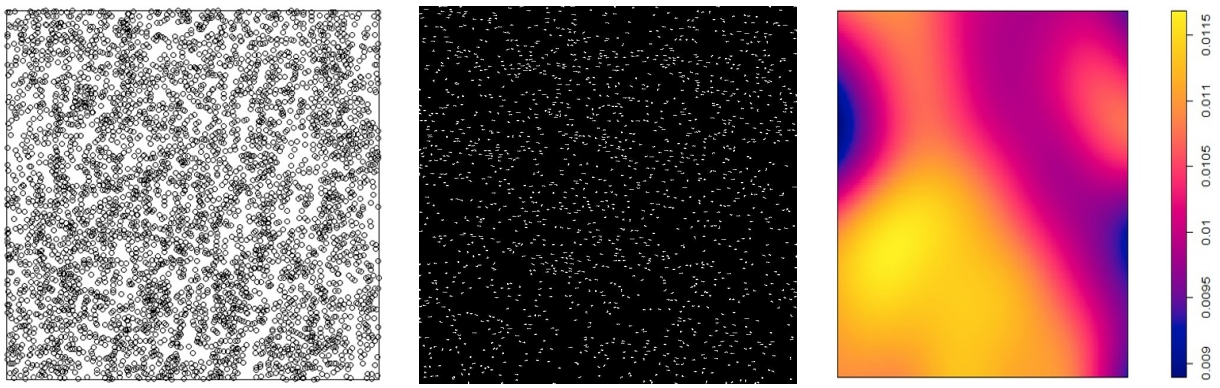
(g) Inh. K - Function

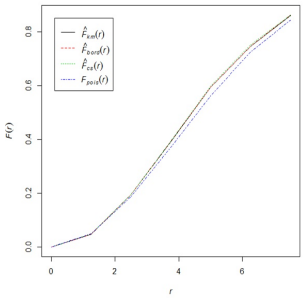


(h) Inh. L - Function

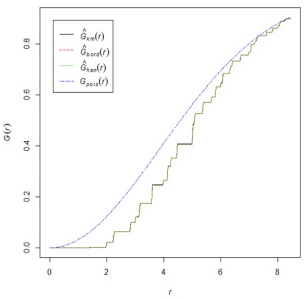
Figure 39: Scale 7

Scale 8

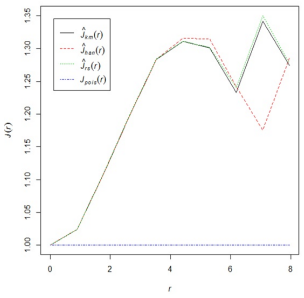




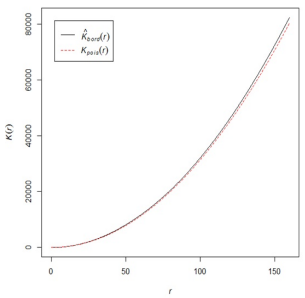
(a) F- Function



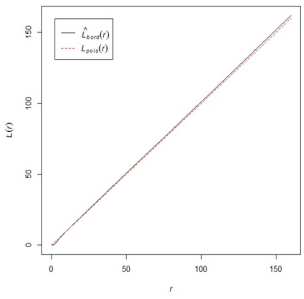
(b) G - Function



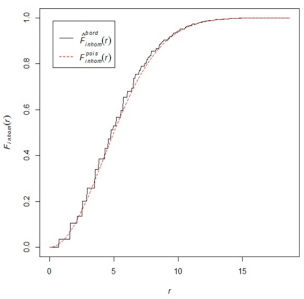
(c) J - Function



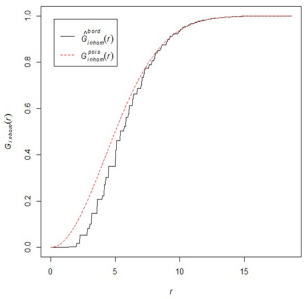
(d) K - Function



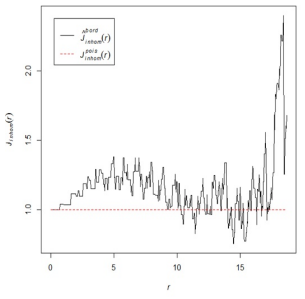
(d) L - Function



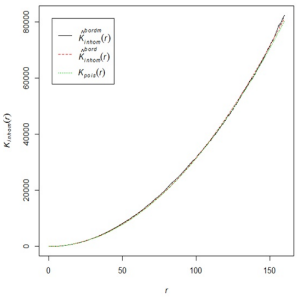
(e) Inh. F - Function



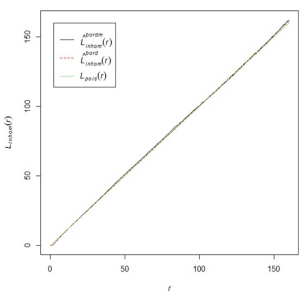
(f) Inh. G - Function



(g) Inh. J - Function



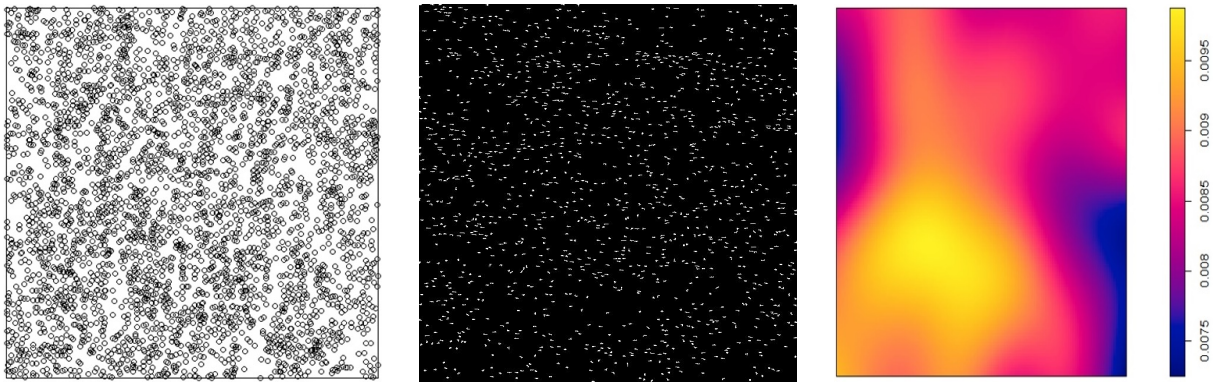
(g) Inh. K - Function

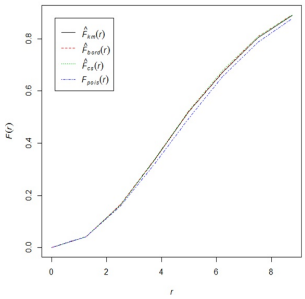


(h) Inh. L - Function

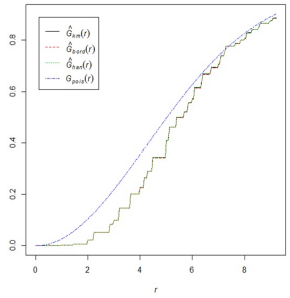
Figure 40: Scale 8

Scale 9

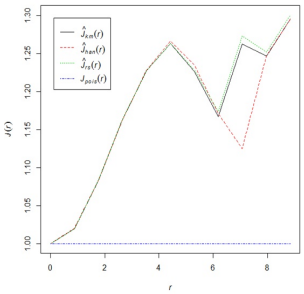




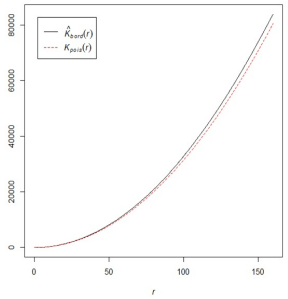
(a) F- Function



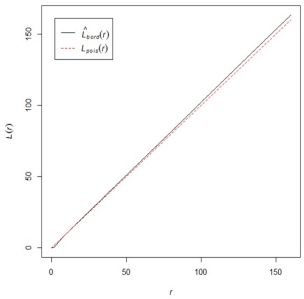
(b) G - Function



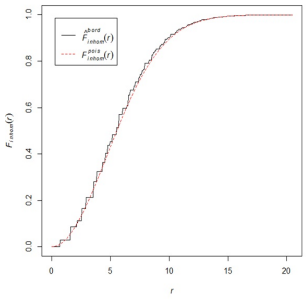
(c) J - Function



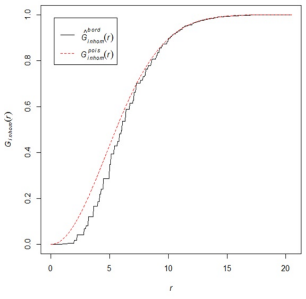
(d) K - Function



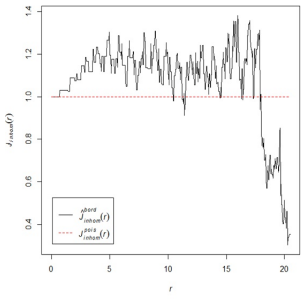
(d) L - Function



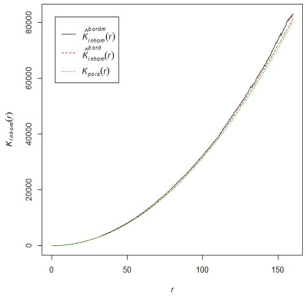
(e) Inh. F - Function



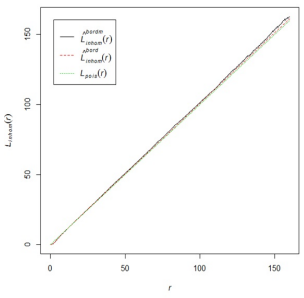
(f) Inh. G - Function



(g) Inh. J - Function



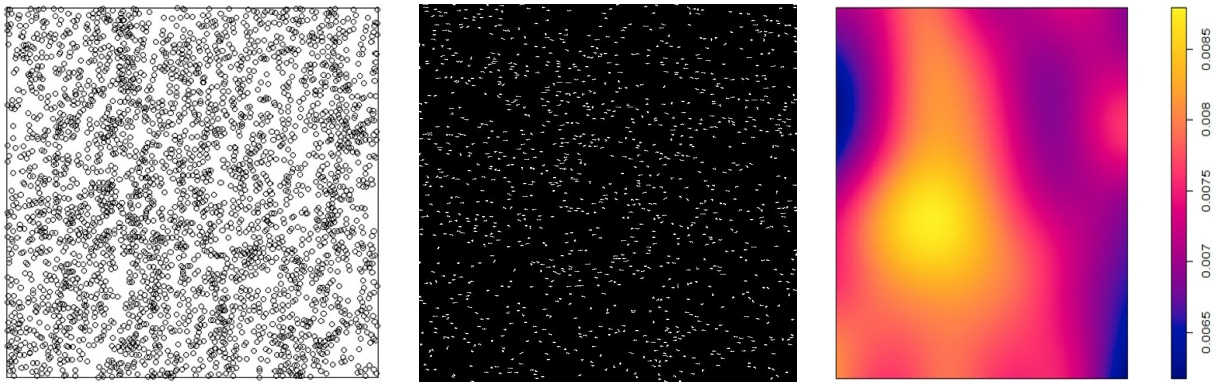
(g) Inh. K - Function

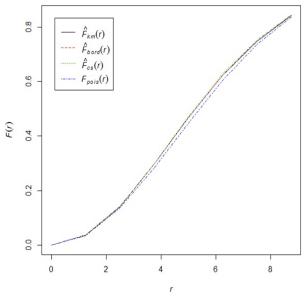


(h) Inh. L - Function

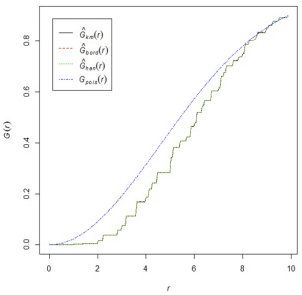
Figure 41: Scale 9

Scale 10

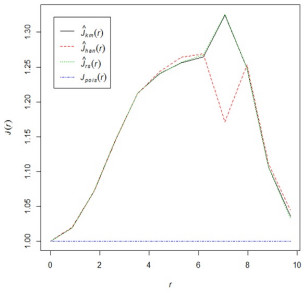




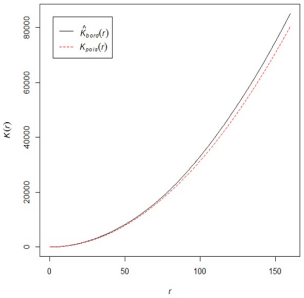
(a) F- Function



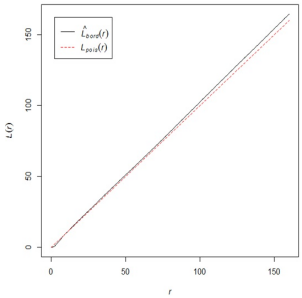
(b) G - Function



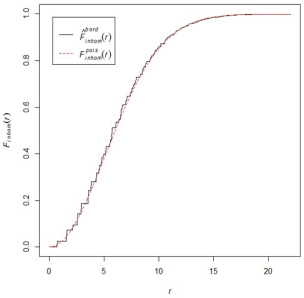
(c) J - Function



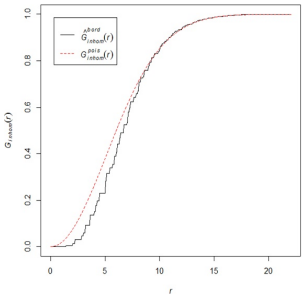
(d) K - Function



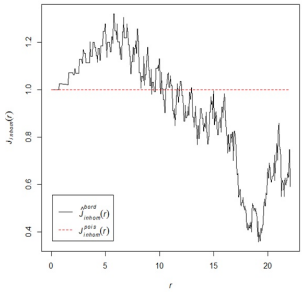
(d) L - Function



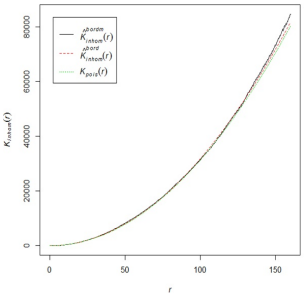
(e) Inh. F - Function



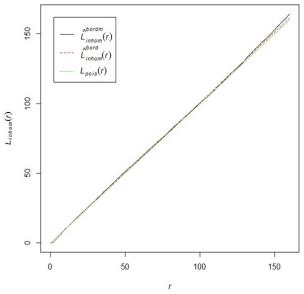
(f) Inh. G - Function



(g) Inh. J - Function



(g) Inh. K - Function



(h) Inh. L - Function

Figure 42: Scale 10

When looking at the images for each of the scales, we can make some conclusions:

- We notice that the inhomogeneous K and J - functions do not give explicit information. The same follows for the homogeneous K and J - function. If one was only to consider these functions, we might have made the conclusion that the points are indeed randomly distributed, but this is not the case. Thus we would conclude that due to the high number of points (in each scale) these functions are not the proper functions to look at.
- For the homogeneous J - function, we make the same conclusion but we observe something different. We notice that, from scale seven onwards, one of the methods for dealing with edge effects becomes very robust and does something different as compared to the other corrections. This specific correction is known as the *Hanisch-type* correction. We also observe that this occurs always between $r = 6$ and $r = 8$. For $r > 8$, this correction method follows then the same pattern as the other corrections. One of the reasons that this correction becomes robust can be due to the fact that this correction is mostly used for circular (disk) study regions.
- Looking at the homogeneous as well as the inhomogeneous F and G - functions, we observe (if information is given) that the points have some form of a regular distribution. This can be seen by looking at each scale's G - function explicitly.
- The inhomogeneous J - function's results are a bit harder to decifer than with the other function. We observe that the vast majority results obtained indicates regularity (as obtained from the F and G functions), but at some instances we also observe some clustering. One of the reasons for these mixed results can be due to the fact that the J - function takes the empty space functions well as the nearest-neighbour distance function into account. What we do observe is that, for $r \leq 10$, the results are mostly regular. In some way, we would expect some clustering to occur due to the fact that as r increases, more points are included in the calculations. So we need to select a threshold for the value of r and only consider values less than this threshold. If we choose our threshold to be $r \approx 10$, we make the conclusion of regularity.

Based on the results mentioned above, we would make a conclusion that at each scale, from one to ten, that the points are regular. Thinking about this logically, this makes intuitive sense. The image we used is a water image with waves. The waves move in a half oval shape, which in some manner is a certain pattern. The results we pick up from our different functions corresponds to this statement. Thus to make a conclusion, we state that the points are indeed regularly distributed.

5 Conclusion

In this report, we have discussed the theory of a point process, using different functions on a point prcess in the manner of testing for Complete Spatial Randomness, and have applied this on a specific image. From this we have observed, and made the conclusion, that our point process actually follows some regular pattern. One of the shortcomings that we encountered (in some manner we wouldn't expect is to be a problem) is that there were many points. Due to this fact, some of our functions applied to the point process were uninformative in some manner. Dealing with this problem, can sometimes results in more problems, due to the fact that is we remove some points or make our study region smaller might result in losing information. A second shortcoming, which was only seen in our results of our application is a threshold value for r. Choosing the correct threshold value can sometimes be difficult due to the fact that there is always the possibility of loosing information. Another shortcoming would be due to the fact that we have only looked at each scale separately, whereas if we considered all the scales combined, we might have obtained different results. A recommendation would be to increasee the number of scales observed, and to consider all the scales combined.

Most images used to test for complete spatial randomness can be divided into different scales by using the DPT. From there onwards, the different tests can then be applied which will then result in a conclusion which might have not been as obvious as looking at the images itself. The reason we use the words 'most images' is due to the fact that not all images can necessarily be divided into scales by using the DPT, but if the DPT can be applied to the images, tests for Complete Spatial Randomness can be applied.

References

[1] R Anguelov and I Fabris-Rotelli. LULU Operators and Discrete Pulse Transform for multidimensional arrays. *Image Processing, IEEE Transactions on*, 19(11):3012–3023, 2010.

[2] A Baddeley. Analysing spatial point patterns in R. Technical report, CSIRO, 2010. Version 4. Available at www. csiro. au/resources/pf16h. html, 2008.

[3] A Baddeley, I Bárány, and R Schneider. *Stochastic Geometry: Lectures given at the CIME Summer School held in Martina Franca, Italy, September 13–18, 2004*, chapter Spatial point processes and their applications, pages 1–75. Springer, 2007.

[4] A Baddeley, R Turner, J Moller, and M Hazelton. Residual analysis for spatial point processes. Technical report, University of Western Australia. Department of Mathematics and Statistics, 2004.

[5] AJ Baddeley. Spatial sampling and censoring. *Stochastic Geometry: Likelihood and Computation*, 2:37–78, 1999.

[6] LJ Bain and M Engelhardt. *Introduction to Probability and Mathematical Statistics*, volume 4. Duxbury Press Belmont, CA, 1992.

[7] S Barthelmé, H Trukenbrod, R Engbert, and F Wichmann. Modeling fixation locations using spatial point processes. *Journal of Vision*, 13(12):1, 2013.

[8] PJ Diggle. *Statistical Analysis of Spatial Point Patterns*. Academic press, 1983.

[9] PJ Diggle. A point process modelling approach to raised incidence of a rare phenomenon in the vicinity of a prespecified point. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, 153(3):349–362, 1990.

[10] P.J Diggle. *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. CRC Press, 2003.

[11] MNM Van Lieshout and AJ Baddeley. A nonparametric measure of spatial interaction in point patterns. *Statistica Neerlandica*, 50(3):344–361, 1996.

[12] JM Loh and ML Stein. Bootstrapping a spatial point process. *Statistica Sinica*, 14(1):69–102, 2004.

[13] BD Ripley. Modelling spatial patterns. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(2):172–212, 1977.

[14] Brian D Ripley. *Statistical Inference for Spatial Processes*. Cambridge University Press, 1991.

[15] SV Sarma, DP Nguyen, G Czanner, S Wirth, MA Wilson, W Suzuki, and EN Brown. Computing confidence intervals for point process models. *Neural Computation*, 23(11):2731–2745, 2011.

[16] MNM Van Lieshout. A J–function for inhomogeneous point processes. *Statistica Neerlandica*, 65(2):183–201, 2011.

[17] K Vasudevan, S Eckel, F Fleischer, V Schmidt, and FA Cook. Statistical analysis of spatial point patterns on deep seismic reflection data: a preliminary test. *Geophysical Journal International*, 171(2):823–840, 2007.

[18] LA Waller and CA Gotway. *Applied Spatial Statistics for Public Health Data*, volume 368. John Wiley & Sons, 2004.

Appendix

Poisson Point Process

We define a Poisson process for the Homogeneous case as follows:

Let $X(t)$ donate the number of occurrences in the interval $[0, t]$ and let $P_n[t] = P[n \text{ occurences in the interval } [0, t]]$. The following properties are important:

1. $X(0) = 0$
2. $P[X(t+h) - X(t) = n | X(s) = m] = P[X(t+h) - X(t) = n]$ for all $0 \leq s \leq t$ and $0 < h$
3. $P[X(t + \Delta t) - X(t) = 1] = \lambda \Delta t + o(\Delta t)$ for some $\lambda > 0$
4. $P[X(t + \Delta t) - X(t) \geq 2] = o(\Delta t)$

Bayesian categorical data analysis on sparse data

Jacob Modiba 10245864

WST795 Research Report

Submitted in partial fulfillment of the degree
BSc(Hons) Mathematical Statistics

Supervisor: Mr M.T Loots

Department of Statistics, University of Pretoria

September 18, 2015

Abstract

This research report focuses on the comparison of frequentist and Bayesian statistics when dealing with categorical data, specifically contingency tables. Categorical data is defined to be data that can be categorized or grouped into non-overlapping categories. There are many methods that can be used to test for association in contingency tables, but in this research report we will discuss the χ^2 and Fisher's exact test. The χ^2 test of independence is the most commonly used method to test for association in contingency tables, but this becomes a problem when many cells in the contingency table have zero or small counts and thus leading to small expected frequencies (sparse data) or when the sample size is too small. In this case Fisher's exact test will be used. We will discuss a Bayesian method that can be used to test for association in contingency table.

Declaration

I, *Jacob Mantjitji Modiba ,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

Insert Student's full name Here

Insert Supervisor(s) name(s) here

Date

Acknowledgements

Add acknowledgements here (not compulsory).

Contents

List of Figures

List of Tables

1 Introduction

Categorical data is data that can be divided into non-overlapping groups or categories so that it can be easily analysed. There are different ways to make categorical data easy to interpret, including frequency tables and contingency tables. In the contingency tables we want to know if the rows and the columns relate, which is to test if one variable can be estimated from another variable (test of independence). The χ^2 test of independence is the most commonly used method to test for association in contingency table, but this becomes a problem when many cells in the contingency table have zero or small counts and thus leading to small expected frequencies (sparse data) or when the sample size is too small. In this case Fisher's exact test will be used. The χ^2 and Fishers exact test uses the p-value under the null hypothesis, in this case the independence hypothesis, but if the test strongly rejects the null hypothesis we do not receive enough information as to what distribution generated the data [2]. We will discuss the Bayesian approach that is not affected by the sample size and small frequencies and compare it with the χ^2 test and Fisher's exact test.

2 Background Theory

2.1 Categorical Data and Categorical Distribution

2.1.1 Categorical Data

A set of data is said to be categorical if its values or observations can be divided into non-overlapping groups or categories, which means that every value should belong to only one category. Analysis of categorical data generally involves the use of data tables (e.g. contingency tables and frequency tables) and graphs (e.g. bar charts and histograms). There are two types categorical of data namely nominal and ordinal, where nominal data has unordered categories and ordinal data has ordered categories. There are various statistical software packages that can be used to analyse categorical data e.g. SAS(*Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.*) and R [8]. When many cells in a contingency table have zero or small counts and thus leading to small frequency, this type of data is called sparse data.

2.1.2 Categorical Distribution

A categorical distribution is a discrete distribution that can take on values 1 to k for a k-way categorical distribution. It is denoted by $Y \sim cat(p_1, p_2, ...p_k)$ with the probability mass function given by $f(y = i) = p_i$. Since a Bernoulli random variable can take on values {0,1}, the Bernoulli distribution can be used to model categorical data.

The binomial distribution is n independent Bernoulli trials with the same probability of success for each trial, say, p. Therefore a binomial distribution can also be used to model categorical data. Since the multinomial distribution is a generalization of the binomial distribution, it can also be used to model categorical data. There are various distribution that can be used to model categorical data, but in this research report we will focus only on the multinomial distribution (hence the binomial distribution). A categorical distribution is simply a multinomial distribution with $n = 1$.

2.2 Frequentist Perspective

Consider a vector of counts $\underline{y} = [y_{11}\, y_{12}....y_{ij}.....y_{nm}]$ from an $n \times m$ contingency table, where y_{ij} represents a count in row i and column j with $i = 1, 2, 3, 4.....n$ and $j = 1, 2, 3, 4.....m$. Let $c_1, c_2, ...c_m$ be marginal columns totals, $r_1, r_2.....r_n$ be marginal row totals and N the grand total. The expected frequency of a cell is given by $E_{ij} = \frac{r_i \times c_i}{N}$.

		Categories					
		1	2	.	.	.	k
Variables	1	y_{11}	y_{12}	.	.	.	y_{1k}
	2	y_{21}	y_{22}	.	.	.	y_{2k}
	.	.	.	.			.
	.	.	.		.		.
	.	.	.			.	.
	n	y_{n1}	y_{n2}	.	.	.	y_{nk}

Table 1: $n \times m$ contingency table

2.2.1 χ^2 Test

H_0 (independence hypothesis) is tested using the following test statistics which was proposed by Professor Karl Pearson,[7]

$$\chi^2 = \sum_{\forall i,j} \sum \frac{(y_{ij} - E_{ij})^2}{E_{ij}}, \tag{1}$$

where χ^2 is compared to χ^2_d on a significance level α where $d = (n-1)(m-1)$ is the degrees of freedom. The null hypothesis is rejected at significant level α if $\chi^2 > \chi^2_v$, that is if the p-value$\leq \alpha$.

2.2.2 Fisher's Exact Test

Professor R.A. Fisher proposed an alternate method defined as follows [3]. Assume that the marginal totals are fixed so that the distribution of the cell counts will be that of a hypergeometric distribution. The probability of the observed table is given by

$$P(outcome) = \frac{r_1!\, r_2!.....r_n!\, c_1\, c_2!.....c_m!}{N!\, y_{11}!\, y_{12!}.....y_{nm}!}. \tag{2}$$

We calculate the probability of each of the possible tables that have the same marginal totals as the observed table using (1), so that the p-value equals the sum of all probabilities that are less than or equal to the probability of the observed table. We then reject the null hypothesis if the p-value is less than the significance level α.

2.3 Bayesian Perspective

2.3.1 Key Elements

Bayesian statistics is a wide field so we will only define the elements that are going to be used. We will not go into much details. For instance there are various types of priors so we will only define a conjugate prior.

Prior distribution - A probability distribution of an unknown parameter, p, before some experiment or test denoted by $f(p)$. The parameters of a prior are called the hyperparameters.

Posterior distribution - The distribution of an parameter, p, given the data set $\underline{X}$. The posterior distribution is of the form

$$\begin{aligned} f(p \mid \underline{X}) &= \frac{f(\underline{X}, p)}{f(\underline{X})} \\ &= \frac{f(\underline{X} \mid p) f(p)}{f(\underline{X})} \end{aligned}$$

Where $f(\underline{X} \mid p)$ is the likelihood , $f(p)$ the prior and $f(\underline{X}) = \int f(\underline{X} \mid p) f(p) dp$ for a continuous distribution and $f(\underline{X}) = \sum_p f(\underline{X} \mid p) f(p) dp$ for a discrete distribution.

Conjugate prior - A prior distribution that leads to a posterior distribution which falls in the same family of distribution as the prior distribution. The reason for using conjugate priors is that they make the calculations of the posterior distribution easier.

2.3.2 Bayes Factor

We will discuss the Bayesian approach that is not affected by the sample size and small frequencies, named the Bayes factor and compare it with the χ^2 test and Fisher's exact test. We define the Bayes factor in favor of the hypothesis, H, provided by some evidence, event or experimental result, E, as the ratio of the final odds to the initial odds, $\frac{O(H|E)}{O(H)}$ [4]. This is equal to $\frac{P(E|H)}{P(E|\bar{H})}$, which is a likelihood ratio only when H and its negation $\bar{H}$ are simple statistical hypothesis [4]. In our context we define the Bayes factor as the ratio of the marginal density of the vector of counts under the alternative hypothesis and marginal density of counts under the null hypothesis $\frac{D(\underline{y}|\bar{H})}{D(\underline{y}|H)}$ where $\underline{y}$ is a vector of counts [1]. We denote it by BF. The full details are given in Section 2.3.3.

2.3.3 Bayesian Method

Consider a vector of counts $\underline{y}_1 = [y_{11}, y_{12}, y_{13}...y_{1k}]$ from the first row of an $n \times k$ contingency table, where y_{1i} represent a count in row 1 and column i with $i = 1, 2, 3, 4, 5.....k$. Let $N = \sum_{i=1}^{n} y_{1i}$ be the total of row 1,

	Categories						
Variables	1	2	.	.	.	k	Total
1	y_{11}	y_{12}	.	.	.	y_{1k}	N

Table 2: $1 \times m$ contingency table

and $\underline{p}_1 = [p_{11}, p_{12}, p_{13}...p_{1k}]$ a vector of probabilities of the cells in the first row of the contingency table. Define the null hypothesis to be $H_0 : p_{11} = p_{12} = ... = p_{1k} = \frac{1}{k}$ and the alternate hypothesis to be H_a : *at least one of the* p_{1j} *are not equal*, therefore,

$$BF = \frac{D(\underline{y} \mid H_a)}{D(\underline{y} \mid H_0)},$$

where $D(\underline{y} \mid H_a)$ and $D(\underline{y} \mid H_0)$ are the marginal density of $\underline{y}$ under the alternative hypothesis and the marginal density of $\underline{y}$ under the null hypothesis respectively [1]. The $y'_{1i}s$ are modeled with a multinomial distribution with parameters $\{p_{1i}\}$ defined to be the cell probabilities.

$$\begin{aligned} D(\underline{y} \mid H_0) &= \frac{N!}{\prod_{i=1}^{k} y_{1i}!} \left(\frac{1}{k}\right)^{y_{11}} \left(\frac{1}{k}\right)^{y_{12}} ... \left(\frac{1}{k}\right)^{y_{1k}} \\ &= \frac{N!}{\prod_{i=1}^{k} y_{1i}!} \left(\frac{1}{k}\right)^{N} \end{aligned}$$

Under the alternative hypothesis that at least one of the p_{1i} are not equal. To find the marginal density in this case we need to choose a suitable prior for $\{p_{1i}\}$, say $g(p)$, since the $\{p_i\}$ are unknown under the

alternative hypothesis. therefore,

$$\begin{aligned} D(\underline{y} \mid H_a) &= \frac{N!}{\prod_{i=1}^{k} y_{1i}!} \int (p_1)^{y_{11}} (p_2)^{y_{12}} .....(p_t)^{y_{1k}} g(p) dp \\ &= \frac{N!}{\prod_{i=1}^{k} y_{1i}!} \int \prod_{i=1}^{k} p_{1i}^{y_{1i}} g(p) dp \end{aligned}$$

Therefore,

$$\begin{aligned} BF &= \frac{D(\underline{y} \mid H_a)}{D(\underline{y} \mid H_0)} \\ &= \frac{\frac{N!}{\prod_{i=1}^{k} y_{1i}!} \int \prod_{i=1}^{k} p_{1i}^{y_{1i}} g(p) dp}{\frac{N!}{\prod_{i=1}^{k} y_{1i}!} \left(\frac{1}{k}\right)^N} \\ &= k^N \int \prod_{i=1}^{k} p_{1i}^{y_{1i}} g(p) dp \end{aligned}$$

Let us consider the problem of calculating the expected value of $p_{1i}\,(i = 1, 2.....k)$ given the observed data $\underline{y}$. This can be done using Johnson's "sufficiency" postulate, which states that the prior for p is a linear combination of priors indexed by a parameter κ such that if we knew values of k, N, y_{1i} and κ, then the knowledge of the other multinomial counts $y_j, j \neq i$ would have no effect on the posterior mean of p_{1i} [1]. Equivalently$E(p_{1i} \mid y_{1i}, k, N, \kappa) = E(p_{1i} \mid y_1, k, N, \kappa)$ [1].

This postulate leads to a posterior mean of p_{1i} to be $E(p_{1i} \mid y_{1i}, k, N, \kappa) = \frac{y_{1i}+\alpha}{N+k\alpha}$ where α depends on k and κ [1]. If we choose the prior distribution for $\underline{p}$ to be that of a symmetric Dirichlet distribution this will lead to a posterior for $\underline{p}$ to be that of a Dirichlet distribution which has the expected value equivalent to $\frac{y_i+\alpha}{N+k\alpha}$. This means that assuming the prior distribution for $\underline{p}$ to be that of a symmetric Dirichlet distribution, is equivalent to using the "sufficiency" postulate, and this also means that the Dirichlet distribution is a conjugate prior for. That is the prior $g(p)$ for $\underline{p}$ is given by $g(p) = \frac{\Gamma(k\alpha)}{[\Gamma(\alpha)]^K} \prod_{i=1}^{k} p_{1i}^{\alpha-1}, \alpha > 0$[1]. Therefore

$$\begin{aligned} BF &= k^N \int \prod_{i=1}^{k} p_{1i}^{y_{1i}} g(p) dp \\ &= k^N \int \prod_{i=1}^{k} p_{1i}^{y_{1i}} \frac{\Gamma(k\alpha)}{[\Gamma(\alpha)]^k} \prod_{i=1}^{k} p_{1i}^{\alpha-1} dp \\ &= k^N \frac{\Gamma(k\alpha)}{[\Gamma(\alpha)]^k} \int \prod_{i=1}^{k} p_{1i}^{(y_{1i}+\alpha)-1} dp \end{aligned}$$

multiply both side of the integral by $\frac{\prod_{i=1}^{k} \Gamma(y_{1i}+\alpha)}{\Gamma(N+k\alpha)}$, we get

$$= k^N \frac{\Gamma(k\alpha) \prod_{i=1}^{k} \Gamma(y_{1i}+\alpha)}{[\Gamma(\alpha)]^k \Gamma(N+k\alpha)} \int \frac{\prod_{i=1}^{k} \Gamma(y_{1i}+\alpha)}{\Gamma(N+k\alpha)} p_{1i}^{(y_{1i}+\alpha)-1} dp$$

where $\int \frac{\prod_{i=1}^{k}\Gamma(y_{1i}+\alpha)}{\Gamma(N+k\alpha)} p_{1i}^{(y_{1i}+\alpha)-1} dp = 1$ since this is the integral of the density function of a symmetric Dirichlet distribution with with parameters $(y_{1i}+\alpha)$, therefore

$$= k^N \frac{\Gamma(k\alpha)\prod_{i=1}^{k}\Gamma(y_{1i}+\alpha)}{[\Gamma(\alpha)]^k\Gamma(N+k\alpha)}, \quad \alpha > 0$$

This formula is equivalent to $\frac{\prod_{i=1}^{k}\prod_{j=1}^{y_{1i}-1}(1+\frac{j}{\alpha})}{\prod_{j=1}^{N-1}(1+\frac{j}{k\alpha})}$, where $\prod_{j=1}^{y_{1i}-1}(1+\frac{j}{\alpha})$ equals 1 when y_{1i} is 0 or 1 [4]. From this we can draw BF as a function of the hyperparameter α, and find a useful test statistic defined to be the maximum of the Bayes factor over α [1] i.e.

$$BF_{max} = max_{\alpha}(BF).$$

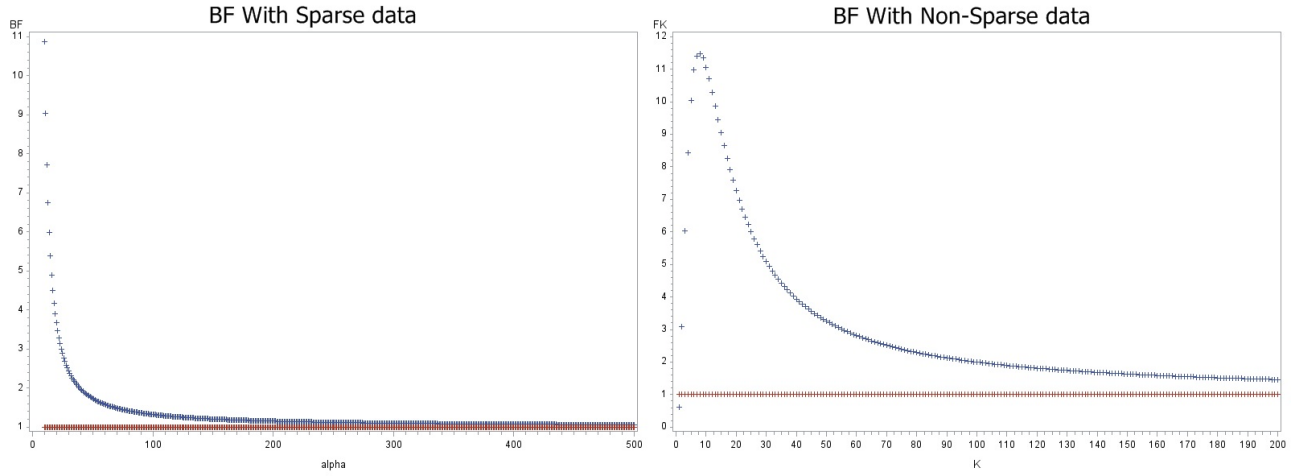


Figure 1: Graphs of BF with sparse and non-sparse data

.

Figure 1 shows the graphs of BF against the hyperparameter α. The red line is the line $BF = 1$ In both graphs BF approaches 1 as α gets larger. $BF > 0$ since $y_i$and α are positive.

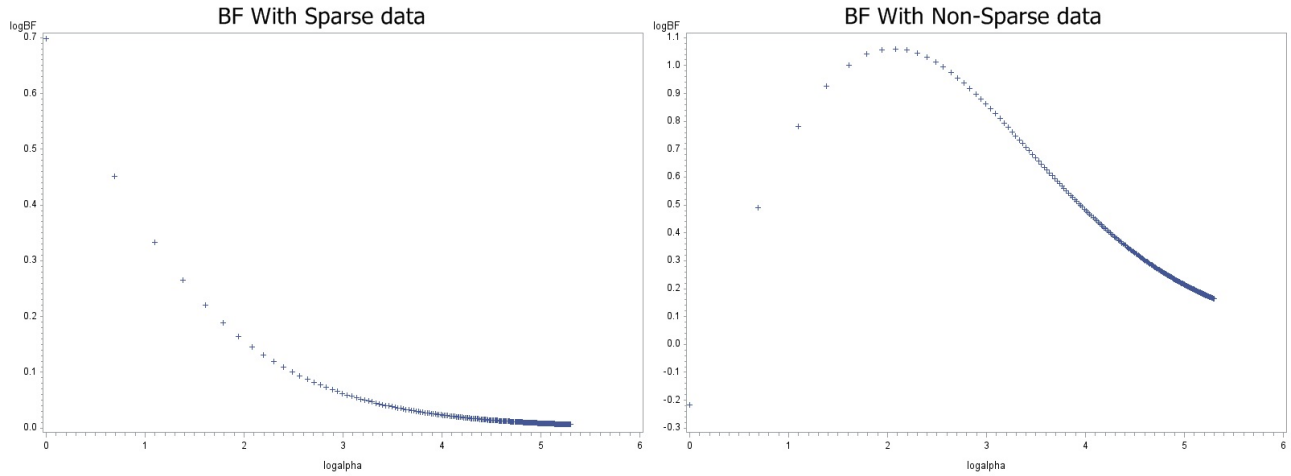


Figure 2: Graphs of BF with sparse and non-sparse data on a log scale

.

Figure 2 shows the graphs of BF against the hyperparameter α on a log scale, i.e $log_{10}(BF)$ against $log(\alpha)$.In both graphs $log_{10}(BF)$ approaches 0 as $log(\alpha)$ gets larger. The data set for sparse data was found in Dr I.J Good's 1967 paper[4], and for non-sparse in the book, *Frontiers of Statistical Decision Making and Bayesian Analysis* [1]

3 Application

	Jan	Feb	March	April	May	June	July	Aug	Sept	Oct	Nov	Dec
Observed count	44	24	35	39	22	25	24	33	30	41	37	46

Table 3: Example

Consider table 3, which record the number of famous writers born in each month. The data was found from, *http://isites.harvard.edu/fs/docs/icb.topic961602.files/notes12.pdf.* We want to test if there is an equal probability of the writers to be born in each month i.e.
$H_0 : P_i = \frac{1}{12} \;\; i = 1, 2, ...12$
H_a : *at least one of the probabilities are not equal*.

3.1 Using χ^2 Test

Using formula (1), the Chi-Square test statistic is found to be $\chi^2 = 22.91$. The critical value for the Chi-square distribution with degrees of freedom equal to $12 - 1 = 11$ on a 5% level of significance is $\chi^2_{11}(0.05) = 19.675138$. Since the test statistic falls inside the rejection region i.e. $\chi^2 = 22.91 > \chi^2_{11}(0.05) = 19.675138$ we reject the null hypothesis H_0. The p-value under the null hypothesis is 0.0181988, therefore on a 5% level of significance we reject the null hypothesis H_0 since p-value=0.0181988 $< \alpha = 0.05$.

3.2 Using Bayes Factor

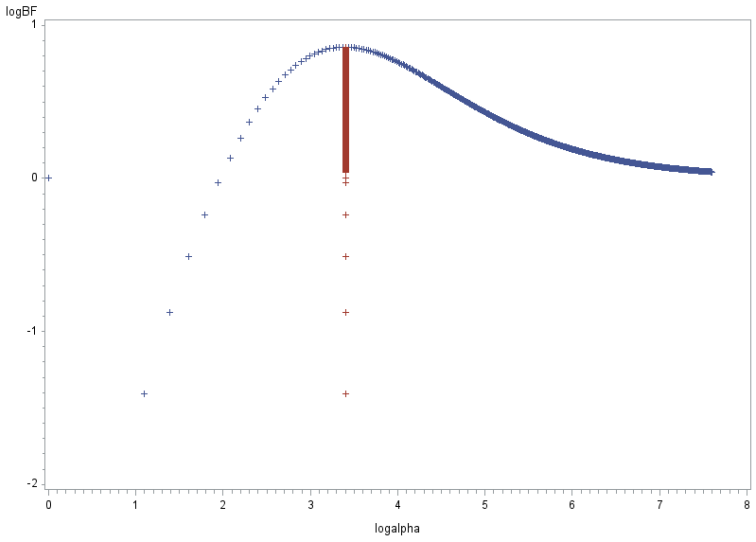


Figure 3: **Left** : BF against alpha, **Right** : BF against alpha on a log scale

.
The graph on the right of figure 3 represent the graph of BF as a function of the hyperparameter α on a log scale. The graph start by increasing until it reaches a maximum point at $log(\alpha) = 3.4011974$ (represent by the red line) and then decreases until it converges to 0. If we assume that the prior probabilities for H_0 and H_a are equal, and the posterior probability of H_0 is represented by the p-value, then the estimate of the Bayes factor is given by $log_{10}(BF_0) = -log_{10}(0.0181988) = 1.739957$. The maximum of BF is at $log_{10}(BF) = 0.85752$. This means that $log_{10}(BF)$ is less than 0.85752 for all α indicating a small but enough evidence against H_0 than as indicated by $log_{10}(BF_0)$.

The [output and data analysis] for this paper was generated using [SAS] software, Version [9.3] of the SAS System for [Unix]. Copyright © [year of copyright] SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA

4 Conclusion

The problem of testing for association in contingency tables can be solved using the χ^2 test, but this becomes a problem when the contingency table has small counts thus leading to small expected frequencies or when the sample size is too small. We introduced a Bayesian statistic that is not affected by small frequencies or sample size. This Bayesian statistic is called the Bayes factor (BF). We tested the null hypothesis of equal cell probabilities assuming the counts follow a multinomial distribution with the cell probabilities as the parameters, and this required us to find a suitable prior for the cell probabilities under the alternate hypothesis.

It was found that the Dirichlet distribution is conjugate prior for the multinomial distribution. The problem is choosing the right hyperparameter the Dirichlet distribution, so we tabulated the values of BF with different values for α. Of this values we used the maximum of the Bayes factor as our test statistic. We can always improve the Bayes factor by finding a suitable prior, say, $g(\alpha)$, for the hyperparameters α of the Dirichlet distribution, and then calculate the new Bayes factor using the formula $\int_0^\infty g(\alpha)BFd\alpha$.

References

[1] Dey D. Müller P. Sun D. Chen, M-H. and K. Ye. *Frontiers of Statistical Decision Making and Bayesian Analysis*. Springer, 2010.

[2] P. Diaconis and B. Efron. Testing for independence in a two-way table: New interpretations of the chi-square statistics. *The Annals of Statistics*, 13(3):845–875, 1985.

[3] R.A Fisher. The logic of inductive inference. *Journal of the Royal Statistical Society*, 98(1):39–82, 1935.

[4] I.J Good. Bayesian significance test for multinomial distribution. *Journal of the Royal Statistical Society. Series B(Methodological)*, 29(3):399–431, 1967.

[5] I.J Good. On the application of symmetric Dirichlet distributions and their mixtures to contingency. *The Annals of Statistics*, 8(6):1198–1218, 1980.

[6] W. E. Johnson and R. B. Braithwaite. *Probability: Deductive and Inductive Problem*. Mind, 1932.

[7] K Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50(302):157–175, 1900.

[8] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.

Appendix

The χ^2 test program

```
proc iml;
use month;
read all into y1;
N1=y1[+,];
t1=nrow(y1);
x1=j(t1,1,0);
do i=1 to t1;
x1[i,1]=((y1[i,1]-(N1/t1))**2);
end;
chi1=((t1/N1)*(x1[+,]))-(t1/N1);
p=1-probchi(chi1,t1-1);
crit=cinv(0.95,t1-1,0);
print chi1 p crit;
run;

proc freq data=month;
tables month / chisq;
run;
```

BF for Sparse Data program

```
proc iml;
use days;
read all into y;
N=y[+,];
t=nrow(y);
x=j(t,1,1);

do k=10 to 500 ; *hyperparameter of the symmetric dirichlet distribution i.e alpha;



         *numerator calculations;

do i=1 to t;                          *for outer product;
             if y[i,]=0 then do;       *condition 1;
              z1=1;
             end;
                   if y[i,]=1 then do;  *condition 2;
                    z1=1;
                   end;
    if (y[i,]>1) then do;
        z=j((y[i,1]-1),1,0);
       do j=1 to (y[i,]-1);                *for inner product;
         z[j,1]=(1+(j/k));
       end;
           x[i,1]=z[#,];
    end;

end;
n=nrow(x);
```

```
E=x[#,1];



              *denominator calculation;

        p=j((N-1),1,0);
        do l=1 to (N-1);
           p[l,1]=(1+(l/(t*k)));
       end;
d=p[#,];

           *bayesian factor and its logarithm;

BF1=(E)/(d);                              *value of the binomial factor;
logBf=log10(BF1);                         *log of the bayesian factor;
logk=log(k);                              *log of the hyperparameter;
BF=BF//(k||logk||BF1||logBf||1);     *matrix containing values of k, logk, BF, logbf and 1;
end;
cn={'alpha' 'logalpha' 'BF' 'logBF' 'one'};
create bfdata from BF[colname=cn];
append from BF;
quit;

proc gplot data=bfdata;
plot (BF one)*alpha/overlay;
run;
```

BF for Non-Sparse Data program

```
proc iml;
use month;
read all into y;
N=y[+,];
t=nrow(y);
x=j(t,1,0);
u=2000;
FK=j(u,1,0);
kv=j(u,1,0);
on=j(u,1,1);
lo=j(u,1,0);
klo=j(u,1,0);
do k=3 to u;
do i=1 to t;
      z=j((y[i,1]-1),1,0);
     do j=1 to ((y[i,1])-(1));
        z[j,1]=(1+(j/k));
     end;
     x[i,1]=z[#,];
end;
q=nrow(x);
E=x[#,];

p=j((N-1),1,0);
```

```
do l=1 to (N-1);
    p[l,1]=(1+(l/(t*k)));
end;
d=p[#,];
FK[k,1]=(E/d);
lo[k,1]=log10(E/d);
kv[k,1]=k;
klo[k,1]=log(k);
FKc=kv||on||FK||lo||klo;
end;
maxiBF=max(FK);
logmaxiBF=max(lo);
maxi=30;
maxl=log(30);
maxBF=j(u,1,max(FK));
maxlogBF=j(u,1,max(lo));
kmax=j(u,1,30);
logkmax=j(u,1,log(30));
FKM=kv||on||FK||lo||klo||maxBF||maxlogBF||kmax||logkmax;
cn={'alpha' 'one' 'BF' 'logBF' 'logalpha' 'maxBF' 'maxlogBF' 'kmax' 'logkmax'};
print maxi maxl maxiBF logmaxiBF;
create fdata from FKM[colname=cn];
append from FKM;
run;
quit;

proc gplot data=fdata;
plot (BF one )*(alpha kmax)/overlay;
plot (logBF )*(logalpha logkmax)/overlay;
run;
```

The analysis of grouped income distributions for the 10% sample of the South African census 2011

Mubatsiri Mukome 11204606

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Dr. G. Crafford

Department of Statistics, University of Pretoria

2 November 2015

Abstract

The aim of this research is to study how income is distributed in South Africa. This is done by considering various factors such as age, level of education, gender and ethnic group. The research is done using the South African census data released by Statistics South Africa in 2011. Methods that can be used to analyse income include the ANOVA, MANOVA and MANCOVA[12]. This is useful when we want to evaluate the income over time. In this paper an odds model will be our main focus. An odds model is used to predict the probability of an individual being in the higher income group compared to the lower income group. The advantage that the odds model provides is that it considers several factors simultaneously in an analysis. This is helpful when studying the effect of different factors on a response variable. The motivation for this topic is mainly to gain insight into how income is distributed according to the variables mentioned. In addition a lot of studies have concentrated on descriptive statistics for example, pie charts and bar graphs whilst inferential statistics are not being implemented. Therefore building an odds model will allow us to see the effect of how various factors affect the level of income. Consequently, this would be helpful to stakeholders such as government, private sector as well as academic and research institutions. This will provide information which is critical for economic growth and assist in the making of cognizant decisions [18].

A brief literature review of similar studies is discussed. Statistical techniques on how one can analyse grouped income will also be given. Theoretical background will be discussed with regards to how we can analyse the data. An application of what has been discussed in theory will then be applied and the statistical inferences will then be drawn. An in-depth analysis of the data along with areas of concern are also explained. A conclusion is then arrived at summarising what was done in the research report. Shortfalls of what was being investigated are given. Areas for future research are suggested in our conclusion.

Declaration

I, *Mubatsiri Mukome,* declare that this essay, submitted in partial fulfilment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Mubatsiri Mukome

_____
Doctor Gretel Crafford

_____
Date

Acknowledgements

I would like to thank my supervisor, Doctor Gretel Crafford for her never give up spirit and time. I am particularly grateful for her wisdom and guidance, which made this quite an experience, that I will cherish for years to come. Her attention to detail and suggestions made this project feasible.

I would also like to acknowledge my cousin for her counsel, which was very enlightening and allowed me to see the research from a different perspective.

Finally, thank you to my parents, sister and immediate family. Their support and encouragement goes beyond what words can explain.

Contents

1 Introduction

According to the 2011 census conducted by Statistics South Africa, StatsSA, a census is defined as a procedure where people are counted after a certain time period in a country. This allows for the collection of information about their demographics, social and economic characteristics [18]. After the information is collected, the process includes the administration, analysis and dissemination of the collected data [18]. The 2011 census was done in order to disseminate statistics on the inhabitants of South Africa. Furthermore, information with regards to social, economic, housing features and the selection of a new sampling frame were also required. In addition to this, the census also wanted to provide a primary base that is essential for the mid-year projections [18]. The research topic to be tackled is going to look at how certain factors affect income by considering the marginal effect of different variables and by considering the different variables simultaneously via the odds model.

2 Literature Review

We shall give an overview of the results concurred on the income distribution from the 2011 census data. Literature that has been studied with regards to the distribution of the income in South Africa will also be given.

2.1 An overview of results of income from the 2011 census

The 2011 census produced a widening income disparity among various ethnic groups [7]. General comments widely emphasized the expansive aperture among those with consistent earnings and those without from a financial perspective. This is despite a narrowing in the fissure but at a slow pace [7].

There was a possibility that individuals did not answer income questions honestly. This was due to the fear that census officials would collude with tax authorities. Thus there was a lack of confidence in the secrecy clause of the census [11]. A comparison of the 2011 census to the South African Revenue Authority showed a 28% difference in incomes between the two when assessing those in the tax paying brackets [4]. Nonetheless this did not detract from the conjectures reached. Despite the irregularities and lack of accuracy, the figures have gone on to be adapted as a reflection of reality, as they have been accepted by the public and statisticians [7].

The person sample data was analysed by making use of a weight variable. This is defined as the product of the person adjustment factor and the inverse of the sampling rate to the relevant population [18].

2.2 Analysis of grouped data

In their 2009 paper Crafford and Crowther presented the following problem where grouped data is common in many disciplines and where continuous variables such as age or income are categorised into class intervals. As a result the usual statistical techniques for continuous response variables can no longer be applied. More often than not researchers are tempted to ignore the underlying continuous distribution of the grouped response variable which results in valuable information being ignored [9]. What was then studied was the case where response variable is only observed in the grouped format.

The main focus of the paper by Crafford and Crowther was to foster basic theoretical concepts and methodology of how to model grouped data. Other distributions such as the Pareto, Weibull could be used to model the continuous nature of the grouped response variable. In her PhD thesis Crafford did a statistical analysis of grouped data with ML estimation procedure of Mathews and Crowther which was used in fitting a continuous distribution to grouped data [10].

2.3 Analysis of income distribution in South Africa

In her thesis of 2007 Malherbe [15] analysed the effect of the grouped income versus the continuous income on income data. This debate was due to various complications that arose when analysing data. For example:

1. People might be reluctant to give their exact income.
2. Also individuals may not know their income to the nearest rand.

This will result in a loss of valuable information and biased results. In order to work with grouped data it was first made to be continuous so that it could be analysed. The following methods were implemented so that continuous data could be obtained from grouped income:

2.3.1 The midpoint method

This method assumes that a person who gives his/her income earns an interval midpoint. Thus traditional methods to analyse the data in grouped format are no longer valid. An upper bound for the top income level does not exist. It was therefore assumed that the midpoint exceeds the lower bound in the last category by 10% [15]. This provided an upper bound. The main downside of this method was the lack of theoretical backing [22]. However it is attractive to use due to the limited knowledge of the statistics required when implementing the midpoint method[21].

2.3.2 Interval Regression

This tries to fit a model to the grouped income dataset by using some well known chosen variables. The model then predicts what income each individual will have based on the variables used to fit the model. Usually dummy variables are required in order to use interval regression. Income can be modelled as follows:

$$lnY_i = b_0 + b_1x_1 + b_2x_2 + b_3x_2^2 + e_i \tag{1}$$

where

Y_i represents income

x_1, x_2 are various factors under consideration

b_1, b_2, b_3 are the regression coefficients

b_0 is the y-intercept

e_i is the standard error

When interval regression is being studied we also have a measure of how well the model fits given by the R^2 statistic [15]. This value is between zero and one where a value of 1 indicates a one hundred percent fit. The formula is given by:

$$R^2 = 1 - (\frac{\sum(X_i - Y_i)^2}{\sum(X_i - \bar{X})^2}) \tag{2}$$

where

X_i the midpoints for the income intervals

Y_i the predicted income

$\bar{X}$ is the mean of the midpoints of the income intervals [15].

2.3.3 Random midpoint

This makes use of the midpoint of income category and then distributes the individuals falling within the income category randomly. We assume the following:

f_i represents the frequency of individuals falling within income category i

x_i represents the midpoint of income category i.

Thus the model used to obtain the midpoint dataset is given by:

$$Y_{ij} = x_i + sign_{ij}U_{ij} \tag{3}$$

where

Y_{ij} the random midpoint income value for income category i and individual j

$sign_{ij}$ the income category i for individual j with

$$sign_{ij} = \begin{cases} +1 & \text{with probability 0.5} \\ -1 & \text{with probability 0.5} \end{cases}$$

where

$U_{ij} \sim Uniform(lowerbound_i, x_i)$

$lowerbound_i$ is the lowerbound of income level i.

What was then inferred by Malherbe was that from an analytical point of view is that income as grouped or continuous data does not make that much of a difference in the results [15]. The same was also conferred by Von Fintel earlier as he came to the conclusion that using either a continuous or grouped variable is equally accurate[21]. Nonetheless two factors will have a big influence on income i.e

1. Size of the income
2. Method used to obtain continuous data set from the grouped income data set.

From a practical point of view the use of grouped income gave the advantage that individuals are more likely to give their income in this form than in the exact amount. The optimal solution when wanting to get as much accurate information with regards to income is to ask individuals to indicate either of the following:

1. Exact income value
2. Indicate which income category they fall in
3. Indicate both of the above

This way the information gathered can be used to obtain better results. Thus income data will be more reliable as individuals are more likely to indicate their correct income bracket [15].

2.4 Inequality in income distribution in South Africa

By analysing income we can then deduce inequality within a population. Inequality looks at differences in the standards of living across a population or region. It refers to any aspect of deprivation [15]. Two types of inequality exist:

1. Relative inequality - this depends on the ratio of individual income to the overall mean. This is used more widely in dealing with the analysis of inequality.
2. Absolute inequality - this refers to the absolute differences in the levels of income.

Measures used to measure inequality include Decile Dispersion Ratio, Percentile Ratio and Gini Coefficient, for example. We discuss a few of them below in further detail.

1. Decile Dispersion Ratio - this is an inequality measure not commonly used. It represents the proportion of average income of the wealthiest ten percent of the population over the average income of the bottom ten percent of the population [8].

2. Percentile Ratio - Inequality can be measured in terms of the ratio between shares of different percentiles, for example between a person at the 85th percentile and a person at the 60th percentile in the distribution. Percentile ratios are useful in understanding the dynamics in different parts of the income distribution. Thus it is not a formal measure of inequality[20]. Percentile ratios are in general not skewed. The drawback is that they do not reveal information about the distribution at points other than the two specific percentiles used in a given ratio [20] .

3. The Gini Coefficient - is the most common measure of income inequality used. It varies between zero(where there is perfect inequality and all the individuals earn the same amount of income) and one(this is when we have imperfect inequality) [20]. When we have a Gini Coefficient of one this means that one person earns all the income and the rest earn nothing. The Gini Coefficient for a population [5] is calculated as follows:

$$G = \frac{1}{2n^2\mu}\sum_i\sum_j |y_i - y_j| \tag{4}$$

where

y_i is the income for the i^{th} person in the population

y_j is the income for the j^{th} person in the population

n is the population size

$\mu = \frac{1}{n}\sum y_i$ is the mean income

3 Background Theory

3.1 Tabulation of data

The frequency gives us an idea of the number of observations under the variable(s) being considered. The frequency tables provide easy access to statistics for testing for association in a contingency or cross tabulation table for example. Chi-square tests can be computed to determine if variables are associated [1]. The statistics provided by the contingency tables include:

•Chi-square tests and measures
•Measures of association
•Odds ratio

A two-way table which indicates whether the row and column variables are dependent or independent should be considered when choosing which measures of association to use [1]. It should however be noted that care should be taken when interpreting measures that are appropriate for one's data [1].

Once the odds from the frequency tables are calculated these are then compared with the odds calculated via the logistic regression or logit model (to be discussed later) and inferences are then made.

3.2 Logit Model and Logistic Regression

Logit model and logistic regression are used to forecast an outcome that is discrete from a grouped data of variables that may be continuous, discrete, dichotomous [13]. The logit i.e the natural logarithm of the odds ratio is the fundamental mathematical concept behind logistic regression [17] . Logistic regression is a more flexible technique when compared to other methods such as discriminant analysis, because they are no

assumptions made about the distributions of the predictor [13]. In addition logistic regression cannot produce negative predicted probabilities. This form of regression is useful when we expect a nonlinear relationship between distribution of the responses on the dependent variable with one or more independent variables.

A simple linear regression equation is given by

$$u = \alpha + \beta X \tag{5}$$

The linear equation then results in the logit or log of the odds which is given by:

$$logit(Y) = \text{natural log(odds)} = ln\left(\frac{\pi}{1-\pi}\right) = \alpha + \beta X \tag{6}$$

- β regression coefficient. This tells us of the relationship between X and the logit of Y
- α is the Y intercept
- u is the linear regression equation
- Y is the outcome of interest
- X is the predictor variable
- $\pi = Probability(Y = \text{ outcome of interest } \mid \text{X=x, a specific value of X }) = \frac{e^{\alpha+\beta X}}{1+e^{\alpha+\beta X}}$
- $e = 2.71828$ is the base of the natural logarithm

We can thus extend the logit of a simple logistic regression to multiple predictors. This then gives the following equation:

$$logit(Y) = ln\left(\frac{\pi}{1-\pi}\right) = \alpha + \sum \beta_j X_j \tag{7}$$

with

- $\pi = Probability(Y = \text{ outcome of interest } \mid X_1 = x_1, ..., X_k = x_k) = \frac{e^{\alpha+\sum \beta_j X_j}}{1+e^{\alpha+\sum \beta_j X_j}}$
- $u = \alpha + \sum \beta_j X_j$ is the linear regression equation
- X_i are the predictors for $i = 1, 2, ...k$

The linear regression equation is the natural logarithm log_e of the probability of being in one group divided by the probability of being in another group. The process that is done for estimating coefficients i.e α and β_j is the maximum likelihood [13].

Logistic regression can be used to fit and compare models. A model that can be fitted by making use of a constant and no predictors typically results in a simple and worst fitting model. A model that has all predictors, interactions and a constant related to the outcome will usually result in a complex and best fitting model [13].

3.2.1 Types of logistic regression

We have the following types of logistic regression which are discussed briefly:

1. Direct Logistic Regression- here all the variables enter the equation simultaneously. When there is no specific hypothesis about the order of importance of the predictor variables, this is an ideal method to utilize. Each predictor is evaluated as if it entered the equation last[13]. One fall back of this type of regression is the interpretation difficulties that arise when the predictors are correlated.

2. Sequential Logistic Regression- this is very similar to sequential multiple regression as order of predictors is specified in the model [13].

3. Statistical (Stepwise) Logistic Regression-here we have a situation where the inclusion and removal of predictors from an equation is based on a statistical criteria. This technique is best seen as a hypothesis or screening generating technique [13].

3.3 The odds model

The ratio that the probability of an event of interest occurs, to the probability that it does not, is called the odds[6]. This is estimated by making use of a ratio which is given as a relationship of the number of times that the event of interest occurs to the number of times that it does not.

We can calculate the odds of an event by making use of the formula

$$o = \frac{p}{1-p} \tag{8}$$

- p is the probability that an event will occur
- $1-p$ is the probability that an event will not occur

One advantage of the odds model is the ability to cater for linear constraints, survey data and the computation of estimated probabilities [16][23].

In order investigate the effect of the independent variables on the dichotomous dependent variable, logistic regression can be implemented. The odds model indices can be used to explain the effect of the independent variables on the dependent variable.

Most statistical analysis distinguish between a response variable or dependent variable i.e the variable we are trying to explain and an explanatory variable(s). Once the odds have been calculated it is now possible to calculate the odds ratio of different categories.

3.3.1 Odds ratio

The odds ratio is defined as the change in the odds of being in one of the categories of outcome when the value of a predictor increases by one unit [13].

Odds ratio can also be calculated by making use of different categories within the response variable. Thus the formula is:

$$\text{odds ratio} = \frac{\text{odds of particular category}}{\text{odds of interest category}} \tag{9}$$

Odds ratio greater than one reflect an increase in the odds of an outcome of one i.e the response category. This is accompanied with a one unit increase in the predictor. Odds ratio less than one reflect a decrease in the odds of that outcome with a one unit change. Odds ratio are calculated to compare different categories with each other.

The use of odds ratio has increased as[6]:

1. An estimate is given along with a confidence interval for the relationship between binary variables, for example consider win or lose variables.

2. Allow for the assessment of effects of other variables on a relationship by making use of logistic regression.

3. Have a distinct and convenient understanding

Once the odds have been calculated, it is now possible to calculate the probabilities that we are interested in by making use of the formula:

$$prob = \frac{o}{1+o} \tag{10}$$

where

o is the odds of the category of interest

4 Application

We are going to make use of the statistical software package called SAS for our data analysis[2]. SAS codes given shall also be explained where applicable. EXCEL shall be used to explain certain output of the data where appropriate.

4.1 Describing the data an overview

The data that will be used in this analysis is the 10% person sample of the South African census 2011.

The metadata file published with the 2011 census provides a description of the data, sampling design and the variables contained in the data. The person data contains variables such as gender, age, race, work status, income. Some of these variables shall be used in this analysis.

4.1.1 The 10% sample Census 2011

The SAS code to obtain initial 10% sample census data of South Africa is given below:

```
data census11;
infile ' C:\Data\Person_10pct_Sample_v1.txt';
input
@16 Age 3.
@19 Gender 1.
@33 PopGrp 1.
@83 Income 2.
@88 Highest_Level_of_Education 2.
@98 P23A_Employ_stat 1.
@99 P23B_Employ_stat 1.
@100 P23C_Employ_stat 1.
;
run;
```

A brief summary of the data that we work with:

- Person data: 4 418 594 observations
- Grouped response variable: Income
- Independent variables:

1. Age
2. Education level
3. Gender
4. Population group

4.2 One way analysis of data

The SAS code to obtain the focus group is given as:

```
data person11;
set  census11;
if Income = 1 then delete;
if Income = . then delete;
if Income = 99 then delete;
if age < 18 then delete;
if age > 65 then delete;
if popgrp = 5 then delete;
if 12<=Highest_Level_of_Education<=18 or 21<=Highest_Level_of_Education<=28;
run;
```

Our focus group is defined below. These factors are run simultaneously in SAS. The result is that there are 601 059 observations from an initial 4 418 594 observations.

Focus Group

1. Income: R1 or more
2. Age: 18-65
3. Education Level: Grade 12 and higher
4. Population group: Black, Coloured, Indian, White

A brief description of how we arrived at the various restrictions of the explanatory variables above is explained below.

4.2.1 Income

Group	Monthly Income
01	**No income**
02	**1 - 400**
03	**401 - 800**
04	**801 - 1 600**
05	**1 601 - 3 200**
06	**3 201 - 6 400**
07	**6 401 - 12 800**
08	**12 801 - 25 600**
09	**25 601 - 51 200**
10	**51 201 - 102 400**
11	**102 401 - 204 800**
12	**204 801 - more**
99	**Unspecified**
.	**Not applicable**

Table 1: Income categories as given by StatsSA

When income was recorded as either 1 (i.e no income), 99 (i.e unspecified) or not applicable we decided to delete these entries. This greatly reduced the sample size of our data. What was noted however was that imputation methods could have been used in filling in the unspecified and not applicable parts of the data. This method involves imputing a replacement value for the missing data. The replacement value is typically taken from another observation that is identical to the nonrespondent on other variables. This is then imputed for the missing value [14]. However, imputation was negated due to if for example, future data analysts want to analyse the data, they would not be able to differentiate between original and the imputed

values. Furthermore, the imputed values may provide good guesses, but they are not real data which could be a real problem when analysing the data.

Income(R)	Frequency	Percent	Cumulative Frequency
1 - 400	24 388	4.06	24 388
401 - 800	27 661	4.60	52 049
801 - 1 600	71 057	11.82	123 106
1 601 - 3 200	99 982	16.63	223 088
3 201 - 6 400	105 919	17.62	329 007
6 401 - 12 800	119 669	19.91	448 676
12 801 - 25 600	94 344	15.70	543 020
25 601 - 51 200	38 864	6.47	581 884
51 201 - 102 400	12 342	2.05	594 226
102 401 - 204 800	4 010	0.67	598 236
204 801 - more	2 823	0.47	601 059

Table 2: One way frequency table for the various income groups

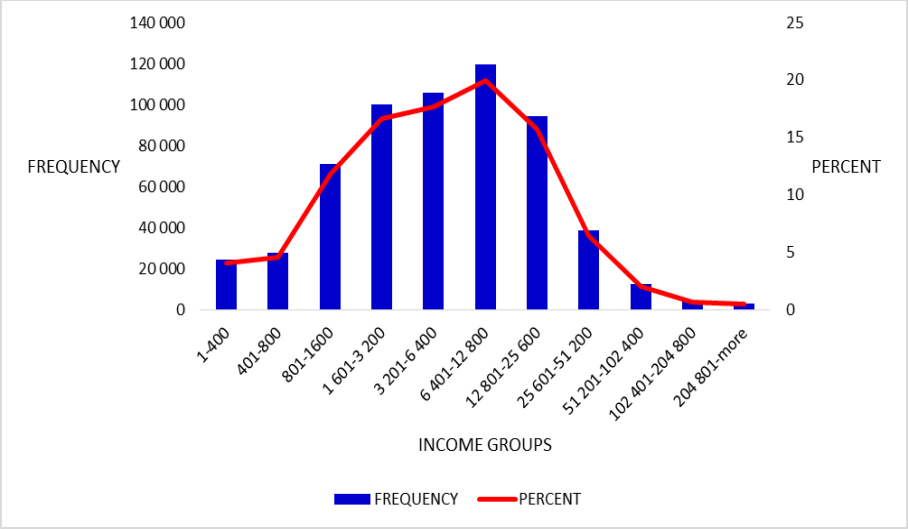


Figure 1: The distribution of grouped income for the focus group

The one way frequency distributions for the various income groups are given in Table 2. A graphical representation of the income groups is then shown in Figure 1.

Classifying income into low and high income

Income was classified as being low if it was below or equal to 12 800 a month and as high if it was greater than or equal to 12 801 a month. The reason behind this was that it was a personal judgment that would be helpful in our analysis. Thus the frequency table is given as follows:

income group	Frequency	Percent	Cumulative Frequency
low	448 676	74.65	448 676
high	152 383	25.35	601 059

Table 3: Frequency table for the classification of income groups

From Table 3, the probability of an individual being in the low income group is about 0.75, whilst the

probability of an individual being in the high income group is approximately 0.25. The odds of an individual to be in the high income group is:

$$\begin{aligned} odds &= \frac{\text{percentage high income}}{\text{percentage low income}} \\ &= \frac{25.35}{74.65} \\ &\simeq \frac{1}{3} \end{aligned}$$

That is, for every 1 person in the high income group, three are in the low income group.
From the odds we can calculate probabilities where:

$$probability = \frac{odds}{1 + odds} = \frac{o}{1 + o}$$

Thus, the probability of being in the high income group is:

$$probability = \frac{\frac{1}{3}}{1 + \frac{1}{3}} = \frac{1}{4}$$

This agrees with the percentage given in Table 3 for the probability to be in the high income group.

4.2.2 Age

Age restrictions when looking at the upper bound were dealt with by considering the Employment Act which does not prescribe an age at which employees must retire [3]. A lot of questions have been raised as to how employers should determine when an employee should retire. One can retire at 55, 60 or 65. Since there is no set retirement age in the labour legislation, the employer can decide the retirement age for his/her employees [3]. In addition, one must also take Section 6 of the Employment Equity Act into consideration when deciding when one should retire[3]. Ages 55 or 60, could have been considered as the cut off points for our sample. This though would have resulted in a smaller sample size as compared to if we considered age 65 as the cut off point. Taking this into consideration it was decided that any persons over the age of 65 should not be considered for our data analysis.

In terms of South African law, children under 18 are legal minors who are not yet fully capable of acting independently without assistance from parents/legal guardians [19]. Consequently, people that were under 18 years of age were not part of the sample, as they were considered to be still minors by law.

Since age is a continuous variable. It was grouped into categorical form in the data so that individuals would then fall into one of the following age-groups:

1. 18-25
2. 26-35
3. 36-45
4. 46-55
5. 56-65

4.2.3 Education level

The level of education one attains is likely to have a massive impact on their income. We considered individuals that had studied from Grade 12 and beyond for our study.

4.2.4 Population group

Individuals were able to specify their population group and in the very rare situations where the subject was not be able to specify their population group they were removed from the sample.

4.3 Cross classification of income

The SAS code to obtain the cross classification of income is given by:

```
proc freq data=finalmod;
tables incomegrp*(agegrp gender popgrp Highest_Level_of_Education ) / chisq expected;
format agegrp agegrp. gender gender. popgrp popgrp. income income. incomegrp incomegrp.
Highest_Level_of_Education Highest_Level_of_Education. ;
run;
```

To investigate the marginal effect of the independent variables on the dependent variable, two way classifications were considered. Our dependent variable is the income group classified as low or high income.

In our analysis we can either look at the low or high income groups in relation to our variables. However, results shall be based on analysis of the effect of various explanatory variables on the high income category. Similar results can be obtained by looking at the low income category as well. Therefore, to obtain results for the low income group you would interchange the resultant numbers in the PROC FORMAT procedure, i.e

```
value incomegrp
1 = 'low'
2 = 'high'
```

Note that the results derived in this section have a marginal effect on income since we are considering one factor at a time.

To investigate the marginal effect of the independent variables, the percentages of the respondents in the high income group are tabulated:

4.3.1 Income vs age-group

age-group	18-25	26-35	36-45	46-55	56-65
% High Income	7.35	19.62	31.49	41.65	37.80
Frequency	7 019	42 172	50 304	36 670	16 218

Table 4: Table showing income vs age-group

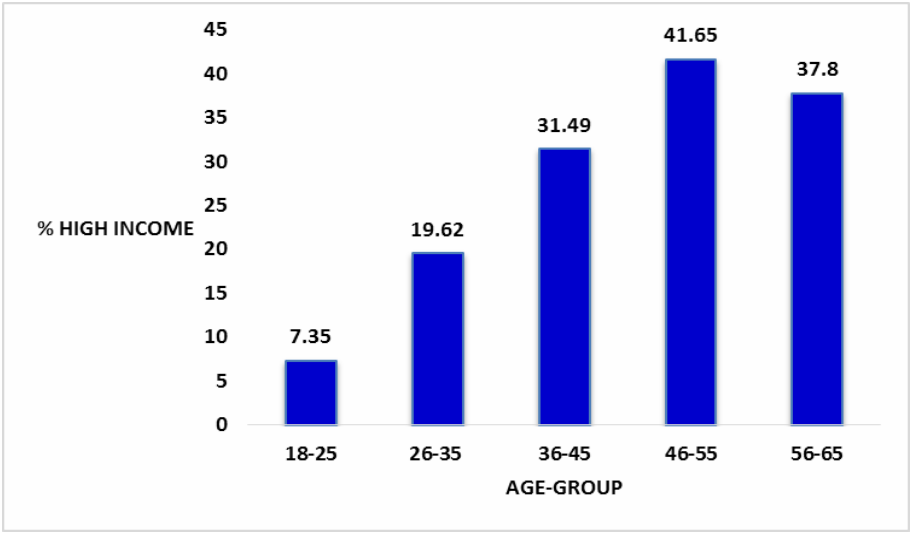


Figure 2: The percentage of high income earners according to age

From Table 4 and Figure 2 above, the percentage of high income earners increase over age. Nonetheless, an anomaly is noted in the age-group 56-65 as the trend decreases from the 46-55 age-group. Therefore, age does have an influence on income as seen by the upward trend in the percentages of high income earners. Hence, we can expect that as people become older the probability of being in the high income group also increases based on the age variable only.

4.3.2 Income vs education level

Education Level	Grade 12	NTC	Certificate	Diploma	Bachelors	Honours	Masters/PhD
% High Income	12.17	32.95	20.41	40.92	58.53	67.04	75.87
Frequency	44 451	8 303	5 506	40 279	30 076	13 463	10 305

Table 5: Table showing income vs education

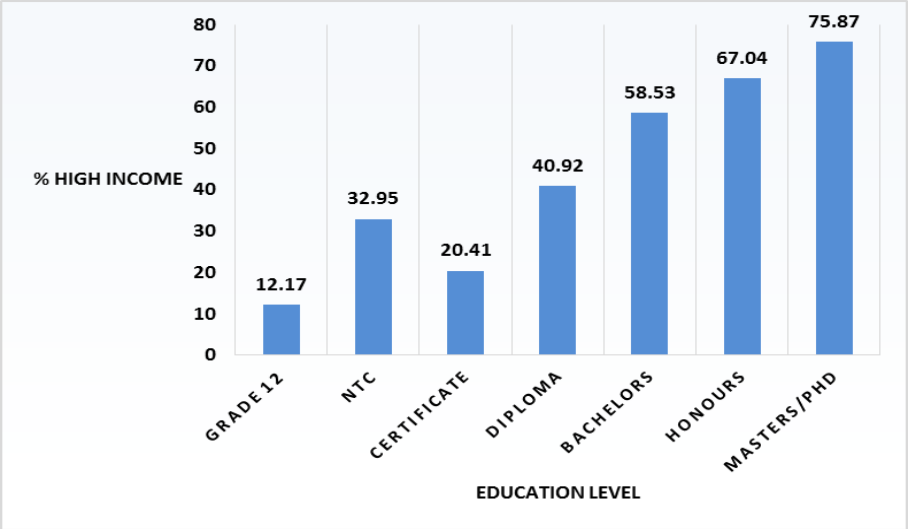


Figure 3: The percentage of high income earners according to education

Table 5 indicates that about 12% of the people with Grade 12 are in the high income category compared to 58% of the people with a Bachelor's degree. A decrease is noted between NTC and Certificate category in Figure 3. This is due to the clustering that has been done where NTC was concerned. It follows that there is a higher percentage of high income individuals in the NTC category than Certificate category. It is interesting to note that 75% of people with a Masters/PhD are classified in the high income group as evidenced in both Table 5 and Figure 3 . An ordinal trend in income over the categories of education is evident. It is clear that education influences the income category that a person is going to belong to.

4.3.3 Income vs gender

Income	Male	Female
% High Income	28.88	21.53
Frequency	90 226	62 157

Table 6: Table showing income vs gender

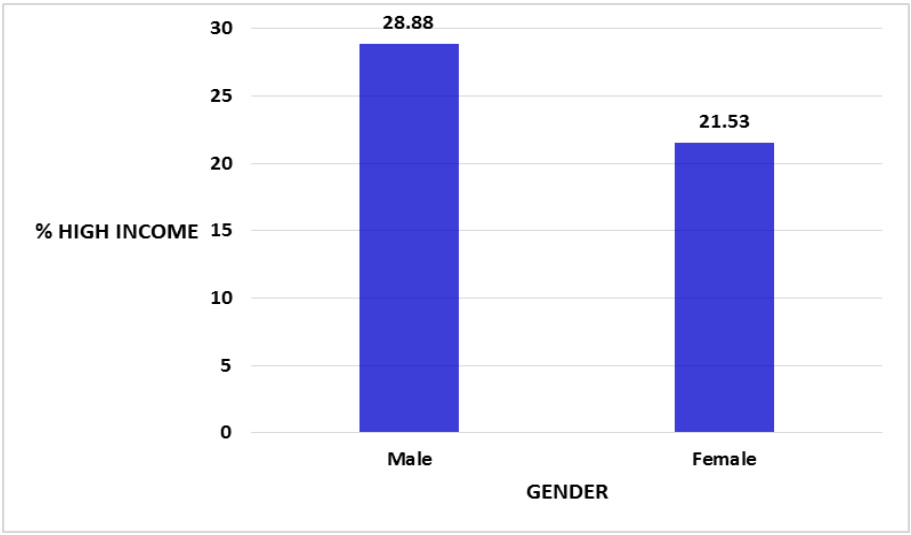


Figure 4: The percentage of high income earners by gender

Our sample constitutes of 90 226 Males i.e 28.88% of the Males whom are high income earners, at the same time, there are 62 157 Females where 21.53% of them are high income earners, as given in Table 6. A significant difference exists between the Male and Female high income earners as noted in the column chart in Figure 4.

Hypothesis testing

Hypothesis testing was done to see if there was a difference in incomes between the Male and Female high income earners.

H_0: No significant difference exists between the Male and Female high income earners

H_A: A significant difference exists between the Male and Female high income earners

Statistic	Value	Prob
Chi-Square	4 281.79	<0.0001

Table 7: Chi Square Results

By making use of the results from Table 7 and testing at a 5% level of significance, we reject the null hypothesis since we have p value < 0.0001 with a value of 4 281.79. Therefore we conclude that significant difference exists between the income of Males and Females. Thus, one can deduce that gender does have an effect on income.

4.3.4 Income vs Population Group

Population Group	Black	Coloured	Indian	White
% High Income	15.42	23.92	35.31	50.02
Frequency	58 157	12 092	11 065	71 069

Table 8: Table showing income vs population group

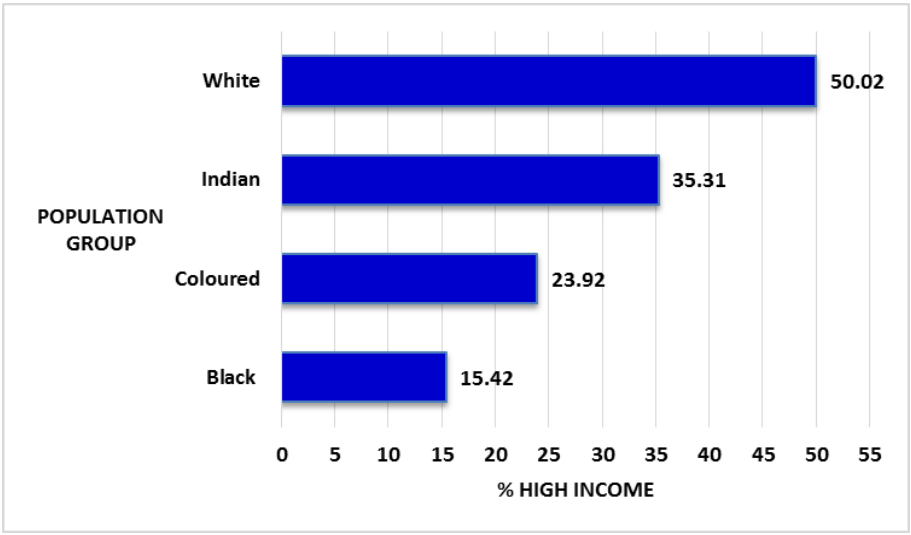


Figure 5: The percentage of high income earners by population group

Table 8 and the bar graph (i.e Figure 5) and show that only 15.42% of the Black people in the sample are classified as being high income earners. The Coloured people have 23.92% categorised as high income earners. There is an 11.39 percentage increase in high income earners from the Coloured people to the Indian people. Approximately 50% of the White people are in the high income category. Therefore, population group has an impact on income as seen by the significant differences in the high income categories of the different population groups.

4.4 Logit Model with one variable being considered

The SAS code which implements the logit model where education is being considered is given below:

```
proc catmod data=finalmod;
model incomegrp = highest_level_of_education / ml nogls oneway;
format incomegrp incomegrp. highest_level_of_education Highest_Level_of_Education.;
Contrast 'overall effect'  intercept    1      / est = exp;
Contrast 'Grade 12'     Highest_Level_of_Education  1 0 0 0 0 0 / est = exp;
Contrast 'NTC'  Highest_Level_of_Education  0 1 0 0 0 0 / est = exp;
Contrast 'Certificate'  Highest_Level_of_Education  0 0 1 0 0 0 / est = exp;
Contrast 'Diploma'  Highest_Level_of_Education  0 0 0 1 0 0 / est = exp;
Contrast 'Bachelors Degree'Highest_Level_of_Education  0 0 0 0 1 0 / est = exp;
Contrast 'Honours Degree'  Highest_Level_of_Education  0 0 0 0 0 1 / est = exp;
Contrast 'Masters / PhD'    Highest_Level_of_Education  -1 -1 -1 -1 -1 -1 / est = exp;
run;
```

The CATMOD procedure performs categorical data modelling on data. CATMOD is short for categorical modelling in SAS. PROC CATMOD fits linear models to functions of response frequencies and can be used for linear modelling, log-linear modelling, logistic regression and repeated measurement analysis. PROC CATMOD uses estimation methods like weighted least squares (WLS), estimation of parameters for a wide range of general linear models. Maximum likelihood (ML) estimation of parameters for log-linear models and the analysis of generalized logits are also used by the CATMOD procedure [1]. The CONTRAST statement provides a mechanism for obtaining custom hypothesis tests[1]. Also, the CONTRAST statement provides parameter estimates for indices. One of the objectives of the CONTRAST statement is hypothesis testing. This is illustrated by considering the education variable. A similar approach can be applied to other variables. Therefore

$$H_0 : \underline{c}^{'}\underline{\lambda} = 0$$

or

$$H_0 : e^{\underline{c}^{'}\underline{\lambda}} = 1$$

where

$\underline{c}$ is a vector of constant values e.g $\underline{c}^{'}$ = (0 0 0 1 0 0)

$\underline{\lambda}$ is a vector of parameters.

However, when modelling the last estimate, our vector $\underline{c}$ would be (-1 -1 -1 -1 -1 -1). This is because the last estimate can be calculated by using the formula

$$\sum \lambda^{edu} = 0$$

Different combinations could be included in the CONTRAST statement. For example, suppose you want to test if the effect of a person with Grade 12 is the same as the effect of a person with a Diploma.

$$H_0 : \lambda_{gr12} = \lambda_{dipl}$$

therefore

$$\lambda_{gr12} - \lambda_{dipl} = 0$$

Our vector $\underline{c}$ in the contrast statement would then be $\underline{c}^{'}$ = (1 0 0 -1 0 0).

To investigate the marginal effect of the independent variable on the dependent variable an odds model is considered. Various models can be considered and the following model will be built when one variable is considered. Education shall be considered in this case.

A simple logit model is formulated as:

$$log(o) = \mu + \lambda^{edu} \tag{11}$$

where we consider education as the only effect on income with

- o is the odds to be in the high income group
- μ is the effect of the overall odds
- λ^{edu} is the effect of education

Note we apply a logit model here since we have a categorical independent variable and categorical dependent variable.

	Index	**Sample Size(n)**
Overall Effect	0.7287	601 059
Education		
Grade 12	0.1900	365 399
NTC	0.6744	25 197
Certificate	0.3519	26 975
Diploma	0.9505	98 435
Bachelor's Degree	1.9365	51 388
Honour's Degree	2.7907	20 083
Masters/PhD	4.3154	13 582

Table 9: Indices value results

The indices given in Table 9 were obtained directly from SAS by specifying the EST = EXP in the SAS code. These were then incorporated into the table above. An ordinal trend in the indices is evident. The odds of being in the higher income group for an individual with a Grade 12 , NTC, Certificate and Diploma

are 81%, 32.56%, 64.81% and 4.95% lower than the overall odds respectively. On the other hand, the odds of being in the high income group for a person with a Bachelor's, Honours and Masters/PhD is 93.65%, 179.07% and 331.54% higher than the overall odds respectively. As a result, education has a huge effect on income as the higher one progresses with regards to education, the higher the odds to be in a better income group.

Calculation of the last estimate

If the EST = EXP is not specified in the SAS code, SAS will only provide the parameter estimates for the logit model. However, care must be taken as the last parameter estimate is not given for an independent variable. Thus to calculate the estimate for Masters/PhD for education in the last category, the formula below is used:

$$\sum \lambda^{edu} = 0 \tag{12}$$

Alternatively

$$e^{\sum \lambda^{edu}} = 1 \tag{13}$$

	Estimates
Education	
Grade 12	-1.6604
NTC	-0.3939
Certificate	-1.0443
Diploma	-0.0508
Bachelor's Degree	0.6609
Honour's Degree	1.0263

Table 10: Estimated values

By using the estimated values in Table 10, it follows that the Masters/PhD estimate is:

$$\begin{aligned} & -\sum \lambda^{edu} \\ = \quad & (-1.6604 - 0.3939 - 1.0443 - 0.0508 + 0.6609 + 1.0263) \\ = \quad & 1.4622 \end{aligned}$$

The estimated values are then used to calculate the indices given earlier by making use of the formula

$$e^{\lambda^{edu}}$$

where λ^{edu} is the parameter estimate for the logit model.
Since there is some difficulty in interpreting the $log(o)$ we take the anti-log to get the following:

$$o = e^{\mu + \lambda^{edu}} = i * i^{edu}$$

with

- i the index of the overall odds
- i^{edu} is the index for various education categories.

Therefore the logit model, models the expected log of the odds (or logit) for a specific category of education. Strictly speaking, μ is the log of the geometric mean odds of various factors incorporated in the model. The more the λ^{edu} parameter deviates from zero, the higher the variation in the odds with respect to the particular factor.

4.4.1 Interpretation of the estimates

Education	Chi-Square	Pr > ChiSq
Grade 12	64 960.90	<0.0001
NTC	1 017.58	<0.0001
Certificate	5 827.36	<0.0001
Diploma	47.70	<0.0001
Bachelor's Degree	5 365.67	<0.0001
Honours Degree	5 690.63	<0.0001
Masters/PhD	6 864.93	<0.0001

Table 11: Chi Square p values for education

According to the estimated values above, there is a strong positive estimate for Honours degree, $\lambda^{hons} = 1.0263$(from Table 10), with a probability < 0.0001(from Table 11). This indicates that the odds to be in the higher income group is significantly higher for people with an Honours degree than the overall odds. There is a negative estimate for Grade 12, $\lambda^{grade12} = -1.6604$ with probability < 0.0001. This tells us that the odds to be in the higher income group differs for people with a grade 12 than the overall odds.

Calculation and interpretation of the indices, odds and probabilities

	Odds	Probability
Overall Effect	0.7287	0.4215
Education		
Grade 12	0.1385	0.1216
NTC	0.4914	0.3296
Certificate	0.2565	0.2041
Diploma	0.6926	0.4092
Bachelor's Degree	1.4111	0.5852
Honour's Degree	2.0336	0.6704
Masters/PhD	3.1446	0.7587

Table 12: Estimated odds and probability results

The geometric mean odds is calculated as:

$$\begin{aligned} & \sqrt[7]{(odds^{gr12} * odds^{NTC} * odds^{cert} * odds^{dipl} * odds^{bach} * odds^{hons} * odss^{Mas/PHD})} \\ = & \sqrt[7]{(0.1385 * 0.4915 * 0.2565 * 0.6926 * 1.411 * 2.0336 * 3.1446)} \\ \simeq & 0.7287 \end{aligned}$$

This is the same as the overall effect given in Table 12. This tells us that, for approximately every 7 people in the high income group, we have 10 in the low income group. This figure is much higher than what is given in the raw data for the observed two way frequency (from Table 3). The reason why we have such a huge difference in the figures is that the odds model places equal weightings for each different category of education.

We illustrate how the indices are calculated, via some examples, as an interpretation has been given earlier. Recall that the indices reveal odds of someone being in a high income group. Consequently, the index for a person with an Honours degree is about 2.7907. This is calculated as:

$$index = e^{estimate} = e^{1.0263} = 2.7907 = i^{Hons}$$

By making use of the indices it is possible to calculate the odds for the different levels of education category. As a result, the odds of being in the high income group for an individual with an Honours degree is

$$odds = i * i^{hons} = 0.7287 * 2.7907 = 2.0335$$

with i being the index for the overall odds.

We shall make use of equation 10 given in the Background Theory to calculate the probability of an individual with an Honours degree to be in the high income category. Thus

$$prob = \frac{o}{1+o} = \frac{2.0335}{3.0335} = 0.6704.$$

The probability of an individual with an Honours degree to be in a high income category is 0.6704. When compared with the cross tabulation output from SAS under income vs education in Table 5, we see that these two probabilities are equal. Thus, a perfect fit is obtained.

Similarly, the index for someone with a Diploma as their highest level of education is about 0.9504. This is calculated as:

$$index = e^{estimate} = e^{-0.0508} = 0.9504 = i^{dipl}$$

Therefore odds to be in the higher income group for a person with a Diploma is 5% lower than the overall geometric mean odds, or alternatively you could say that the odds of being in the high income group decreases by 5%. The odds for an individual being in the high income group with a Diploma is:

$$odds = i * i^{Dipl} = 0.7287 * 0.9504 = 0.6926$$

This will then give us an estimated probability of:

$$prob = \frac{o}{1+o} = \frac{0.6926}{1.6926} = 0.4091$$

The probability of an individual with a Diploma being in the high income category is 0.4091. By evaluating this result with the cross tabulation result given in Table 5 for the Diploma category, we see that these two probabilities are equal. By looking at the odds column in Table 12, it is clear that as the level of education becomes more intense, the odds for a person to be in the high income group increases.

Education	**Low income**	**High Income**	**Odds**
Grade 12	87.83	12.17	0.1386
NTC	67.05	32.95	0.4914
Certificate	79.59	20.41	0.2564
Diploma	59.08	40.92	0.6926
Bachelor	41.47	58.53	1.4113
Honours	32.96	67.04	2.0349
Masters/PhD	24.13	75.87	3.1442

Table 13: Observed odds results

The estimated odds according to the logit model in Table 12 are equal to the observed odds given in Table 13, for all the categories of education level. Thus we have a saturated model. A saturated model is defined as the case where the parameters estimated by logit model equals the number of cells. Therefore we have 7 parameter estimates calculated by SAS and 7 cells.

4.4.2 The odds ratio

This section we look at various odds ratios of an individual to be in a high income category given their level of education in relation to another level of education.

Education	**Odds**	Grade 12	NTC	Certificate	Diploma	Bachelor	Honours	Masters/PhD
Grade 12	0.1385	1.0000	0.2818	0.5400	0.2000	0.0981	0.0681	0.0440
NTC	0.4914	3.5484	1.0000	1.9163	0.7096	0.3483	0.2417	0.1563
Certificate	0.2565	1.8517	0.5218	1.0000	0.3703	0.1817	0.1261	0.0816
Diploma	0.6926	5.0008	1.4093	2.7007	1.0000	0.4908	0.3406	0.2202
Bachelor	1.4111	10.1889	2.8714	5.5025	2.0375	1.0000	0.6939	0.4487
Honours	2.0336	14.6831	4.1379	7.9296	2.9362	1.4411	1.0000	0.6467
Masters/PhD	3.1446	22.7053	6.3993	12.2619	4.5403	2.2284	1.5464	1.0000

Table 14: Odds Ratios Results

The odds ratios for different education categories are given above. The column odds are the numerators whilst the row odds are the denominators. The odds ratio to be in the high income group for a person with a Masters/PhD relative to a person with an NTC in Table 14, is calculated as:

$$\text{odds ratio} = \frac{\text{odds for Masters/PhD}}{\text{odds for NTC}} = \frac{3.1446}{0.4914} = 6.3993$$

This tells us that the odds for someone with a Masters/PhD to be in the high income group is 6.3993 times higher than a person with an NTC. Therefore the odds of an individual with a Masters/PhD to be in the high income group have increased by 539.93%. It is noted that an inverse relationship exists between the upper and lower triangle of the odds ratio table.

4.5 Logit and Logistic Regression Models

The logit and logistic regression models where age is considered are illustrated. The regression equation in terms of coefficients indicates the relative impact of each predictor. The following are shown:

1. Nominal model
2. Ordinal model
3. Linear logistic model
4. Quadratic logistic model

4.5.1 Age as a nominal variable

The SAS code where age-group is considered to be nominal:

```
proc catmod data=finalmod;
model incomegrp = agegrp / ml nogls oneway;
format incomegrp incomegrp. agegrp agegrp.;
run;
```

This model is defined as a saturated model, similar to when education was considered. Therefore,

$$log(o) = \mu + \lambda^{age} \tag{14}$$

where we consider age-group as the only effect on income with

- o is the odds to be in the high income group
- μ is the effect of the overall odds

- λ^{age} is the effect of age-group

The last parameter of the age group variable i.e 55-65 has been omitted from the SAS output. Thus in order to calculate the last parameter the formula:

$$\sum \lambda^{age} = 0$$

is used. Alternatively use the formula:

$$e^{\sum \lambda^{age}} = 1$$

taking the anti log we get

$$o = e^{\mu + \lambda^{age}} = i * i^{age}$$

with

- i the index of the overall odds
- i^{age} is the index for various age-group categories.

	Index	**Sample Size(n)**
Overall Effect	0.3291	601 059
Age		
18-25	0.2411	95 464
26-35	0.7417	214 914
36-45	1.3965	159 739
46-55	2.1690	88 034
56-65	1.8460	42 908

Table 15: Indices value results

The results for the age-group treated as nominal are given in Table 15. An ordinal trend is noted in the index values. Similar interpretations for estimates, indices, odds and probability can be derived for age-group as illustrated when we considered education only.

4.5.2 Age as an ordinal variable

The SAS code for age-group considered as an ordinal variable:

```
data ageord;
set finalmod;
if agegrp =1 then ageordi =-2;
if agegrp =2 then ageordi =-1;
if agegrp =3 then ageordi =0;
if agegrp =4 then ageordi =1;
if agegrp =5 then ageordi =2;
run;
proc catmod data=ageord;
direct ageordi;
model incomegrp = ageordi / ml nogls oneway;
format incomegrp incomegrp.;
run;
```

We would like to model the ordinal trend in the odds over the age categories. This type of modelling makes use of the fact that we are not sure of the class width between each category. Hence we divide our age variable into a specific number of class intervals that sum up to zero. In this case we want 5 class intervals. Therefore we consider integers -2, -1, 0, 1, 2 as representative values. The DIRECT statement is used to model age as an ordinal effect.

Our model is given as:

$$log(o) = \alpha + \beta x \tag{15}$$

where

- x is the integer for each age category
- α is the intercept parameter
- β is the parameter estimate used to estimate the effect of age on income

The fitted probabilities have been calculated by making use of the formula:

$$probability = \frac{o}{1+o}$$

where o is calculated from

$$o = e^{\alpha+\beta x}$$

where the estimates were calculated in SAS.

Hypothesis testing

Hypothesis testing was done to see if the data exhibited an ordinal trend

H_0: Data follows an ordinal trend

H_A: Data does not follow an ordinal trend

Statistic	**Chi-Square**	**Pr > ChiSq**
Likelihood Ratio	7 066.92	<0.0001

Table 16: Likelihood Ratio Results

The results from Table 16 suggest there is a lack of evidence of fit in the model as we have a p-value of < 0.0001 for the likelihood ratio statistic. Therefore, the null hypothesis is rejected. In addition the null hypothesis is rejected due to the large sample size exhibited by the data.

H_0: $\beta = 0$ H_A:$\beta \neq 0$

	Chi-Square	**Pr > ChiSq**
Age	33 425.02	< 0.0001

Table 17: Chi-Square Results

The χ^2 value of 33 425.02 with a p-value < 0.0001 in Table 17 suggest that $\beta \neq 0$, and so the null hypothesis of $\beta = 0$ is rejected.

	Estimate	**Index**
Intercept	-0.9615	0.3823
Beta	0.4937	1.6384

Table 18: Parameter Estimates

Our parameter estimates are $\hat{\alpha} = -0.9615$ and $\hat{\beta} = 0.4937$. This then results in a linear trend in the log odds. The intercept parameter suggests that the overall odds to be in the higher income group for every increase of one age category decreases by a factor of 61.77% i.e the log(odds) decreases by 0.9615. For every increase of one age category the odds of being in the high income group increases by factor of 63.84%. In other words, the log(odds) increase by 0.4937 per age category.

		Observed Probabilities	**Fitted Probabilities**	**Sample Size**
Age	**Ordinal**			
18-25	-2	0.0735	0.1247	95 464
26-35	-1	0.1962	0.1892	214 914
36-45	0	0.3149	0.2766	159 739
46-55	1	0.4165	0.3851	88 034
56-65	2	0.3780	0.5065	42 908

Table 19: Observed and fitted probabilities

The observed probabilities in Table 19 are calculated by using the information from the cross classification of income vs age-group in Table 4. For example, consider the observed probability for the 46-55 age-group. This is calculated as:

$$\text{observed prob} = \frac{\text{number in high income}}{\text{total number in that specific age group}} = \frac{36670}{88034} = 0.4165$$

The fitted probability for the 46-55 age group is then calculated as:

$$prob = \frac{o}{1+o} = \frac{e^{-0.9615+(0.4937*1)}}{1+(e^{-0.9615+(0.4937*1)})} = 0.3851$$

where o is the odds.
The observed and fitted probabilities for other age-groups are calculated in a similar way.

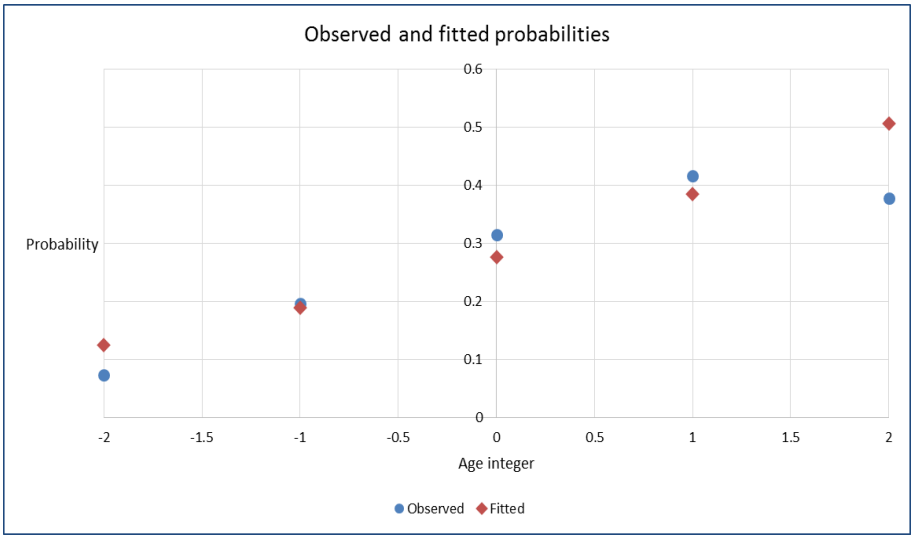


Figure 6: Graph when age is considered as ordinal

From Table 19 and Figure 6, the observed and fitted probabilities are quite close to each other. This suggests that the probability to be in the high income group increases for every age increase of one. As a result, age has an effect on income.

4.5.3 Age with interval class midpoint values

The SAS code for age-group considered for the interval class midpoint method:

```
data agemidpt;
set finalmod;
if agegrp =1 then agemid = 21.5;
if agegrp =2 then agemid =30.5;
if agegrp =3 then agemid =40.5;
if agegrp =4 then agemid =50.5;
if agegrp =5 then agemid =60.5;
run;
proc catmod data=agemidpt;
direct agemid;
model incomegrp = agemid / ml nogls oneway;
format incomegrp incomegrp. ;
run;
```

A linear logistic model is fitted which is given as:

$$log(o) = \alpha + \beta x \tag{16}$$

where

- x is the midpoint of the corresponding class interval of age
- α is the intercept parameter
- β is the parameter estimate used to estimate the effect of age on income

Hypothesis testing

Hypothesis testing was done to see if a linear logistic regression model is applicable on the data.
H_0: Data follows a linear logistic regression model
H_A: Data does not follow a linear logistic regression model

Statistic	Chi-Square	Prob
Likelihood Ratio	7 570.73	<0.0001

Table 20: Likelihood Ratio Results

Table 20 results suggest that there is a lack of fit in the model as we have a χ^2 value of 7 570.73 with a p-value of < 0.0001 for the likelihood ratio statistic. Therefore we reject the null hypothesis.
H_0: $\beta = 0$ H_A: $\beta \neq 0$

	Chi-Square	Pr > ChiSq
Age	33 140	< 0.0001

Table 21: Chi-Square Results

The χ^2 value of 33 140 (in Table 21) suggests that β is not equal to zero since $p < 0.0001$(in Table 21) . It follows that the null hypothesis of $\beta = 0$ is rejected.

	Estimate	Indices
Intercept	-2.9790	0.0508
Beta	0.0497	1.0501

Table 22: Parameter Estimates

Our parameter estimates from Table 22 are $\hat{\alpha} = -2.9790$ and $\hat{\beta} = 0.0497$. This implies that for every increase of one year, the log(odds) increase by 0.0497. Put differently, this tells us that the odds to be in the high income group increases at a rate of 5.01% per year.

		Observed Probabilities	Fitted Probabilities
Age	**Midpoint**		
18-25	21.5	0.0735	0.1289
26-35	30.5	0.1962	0.1880
36-45	40.5	0.3149	0.2765
46-55	50.5	0.4165	0.3848
56-65	60.5	0.3780	0.5070

Table 23: Observed and fitted probabilities

The estimated (i.e fitted) probabilities in Table 23 have been calculated by using the linear logistic model as given by Equation 16, with the estimates calculated in SAS, and by using the midpoints of each age category. As a result, consider for example the fitted probability for the 26-35 age group. This is calculated as:

$$prob = \frac{o}{1+o} = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}} = \frac{e^{-2.979+(0.0497*30.5)}}{1+(e^{-2.979+(0.0497*30.5)})} = 0.1880$$

The fitted probabilities for other age-groups with an interval midpoint are calculated in a similar way.

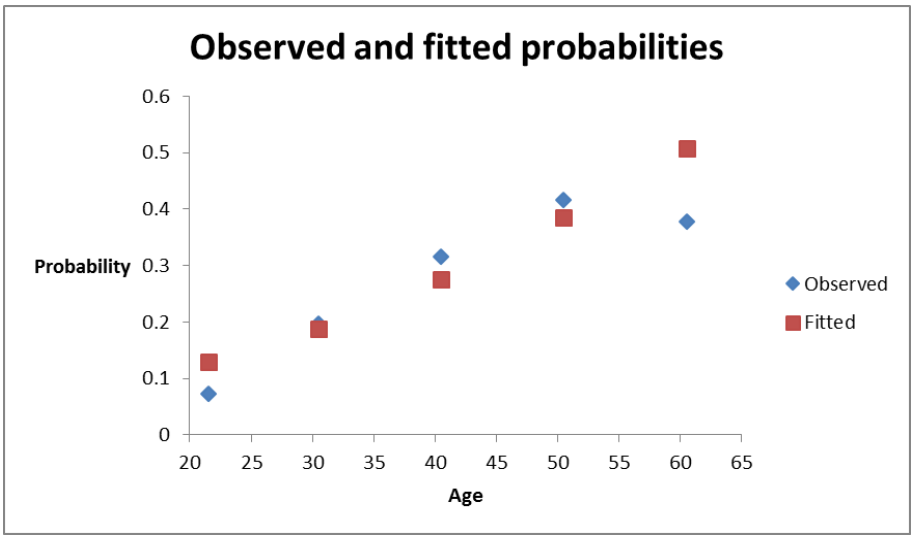


Figure 7: Graph when age is considered as interval midpoints

Table 23 and Figure 7 show that probability to be in the high income group increases, as age increases for both the observed and fitted probabilities. The observed and fitted probabilities are close to each for most of the age categories, with the exception being noted in 56-65 age-group.

4.5.4 Age with a quadratic effect

SAS code for age with the quadratic effect included:

```
data agemidptqd;
set finalmod;
if agegrp =1 then agemid = 21.5;
```

```
if agegrp =2 then agemid =30.5;
if agegrp =3 then agemid =40.5;
if agegrp =4 then agemid =50.5;
if agegrp =5 then agemid =60.5;
agemidsq=agemid*agemid;
run;
proc catmod data=agemidptqd;
direct agemid agemidsq;
model incomegrp = agemid agemidsq / ml nogls oneway;
format incomegrp incomegrp. ;
run;
```

The proposed model is given by:

$$log(o) = \alpha + \beta_1 x + \beta_2 x^2 \tag{17}$$

with

- x is the midpoint of the corresponding class interval of age
- α is the intercept parameter
- β_1, β_2 is the parameter estimate used to estimate the effect of age on income

Hypothesis testing

Hypothesis testing was done on the data to see if age followed a quadratic regression model.
H_0: Data follows a quadratic logistic regression model
H_A: Data does not follow a quadratic logistic regression model

Statistic	Chi-Square	Prob
Likelihood Ratio	285.43	<0.0001

Table 24: Likelihood Ratio Results

Table 24 results advocate for a lack of fit in the model as we have a p-value < 0.0001. Subsequently, the null hypothesis is rejected.

	Estimate	Index
Intercept	-6.2143	0.0020
Beta-1	0.2182	1.2438
Beta-2	-0.00204	0.9980

Table 25: Parameter Estimates

Our parameter estimates are $\hat{\alpha} = -6.2143$, $\hat{\beta_1} = 0.2182$ and $\hat{\beta_2} = -0.00204$. From Table 25, we determine that for every increase of one year, the log(odds) increase by 0.2182. This means that for every increase of one year, the odds of being in the high income group increases by a factor of 24.38%.

		Observed Probabilities	Fitted Probabilities
Age	**Midpoint**		
18-25	21.5	0.0735	0.07828
26-35	30.5	0.1962	0.1889
36-45	40.5	0.3149	0.3267
46-55	50.5	0.4165	0.4019
56-65	60.5	0.3780	0.3822

Table 26: Observed and fitted probabilities

The observed probabilities for each age group have been calculated in exactly the same way as we did when age was considered as an ordinal variable. Hence we have the same results for the observed probabilities.

The estimated (i.e fitted) probabilities have been calculated by using the formula $p = \frac{o}{1+o} = \frac{e^{\alpha+\beta_1 x+\beta_2 x^2}}{1+e^{\alpha+\beta_1 x+\beta_2 x^2}}$. For example, the fitted probability for the 36-45 age-group is calculated as:

$$prob = \frac{o}{1+o} = \frac{e^{\alpha+\beta_1 x+\beta_2 x^2}}{1+e^{\alpha+\beta_1 x+\beta_2 x^2}} = \frac{e^{-6.2143+(0.2182*40.5)-(0.00204*40.5*40.5)}}{1+(e^{-6.2143+(0.2182*40.5)-(0.00204*40.5*40.5)})} = 0.3267$$

The fitted probabilities for other age-groups are calculated in a similar way. Successively, the results as shown in Table 26.

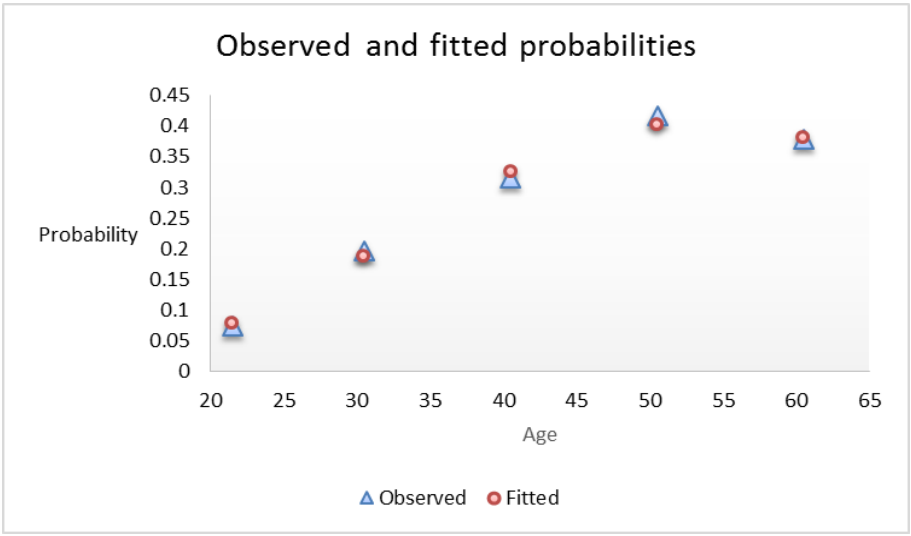


Figure 8: Graph when age is considered with a quadratic effect included

A notable increase in the probability to be in the high income group is noted in Table 26 and Figure 8 as age increases. Figure 8 suggests that age considered as a quadratic effect via the log odds model is a much better model than age considered via the interval class midpoint method (in Figure 7), since the observed and fitted probabilities are much closer to each other.

4.6 Age as a continuous variable

SAS code for when age considered as a continuous variable:

```
data agecont;
set agemod;
if 2 < = income & income <= 7 then incomegrp = 2;
else incomegrp = 1;
```

```
proc freq data = agecont;
tables incomegrp* age; format   incomegrp incomegrp. ;
proc catmod data=agecont;
direct age;
model incomegrp = age / ml nogls oneway;
format   incomegrp incomegrp. ;
run;
```

The frequency procedure was run as we needed to obtain the observed probabilities for plotting the graph for the individual ages. Age was considered as a continuous variable from 18-65.

A suggested model is:

$$log(o) = \alpha + \beta x$$

where

- x is an integer value from 18-65
- α is the intercept parameter
- β is the parameter estimate used to estimate the effect of age on income

	Estimate	**Indices**
Intercept	-3.0228	0.04867
Beta	0.0510	1.0523

Table 27: Parameter Estimates

Our parameter estimates are $\hat{\alpha} = -3.0228$ and $\hat{\beta} = 0.0510$, as given in Table 27. This means that for every increase of one year the log(odds) increases by 0.0510 i.e the odds to be in the high income group increases by a factor of 5.23% every year.

	Number high income	**Total in age**	**Observed Probabilities**	**Fitted Probabilities**
Age				
18	107	2 236	0.0479	0.1086
19	171	4 939	0.0346	0.1137
20	318	7 845	0.0405	0.1189
:	:	:	:	:
:	:	:	:	:
63	1 136	3 692	0.3077	0.5474
64	1 132	3 482	0.3251	0.5600
65	876	3 137	0.2792	0.5725

Table 28: Observed and fitted probabilities

The observed probabilities in Table 28 are calculated as:

$$\text{observed probability} = \frac{\text{number in high income}}{\text{total number in a specific age-group}}$$

For example, the observed probabilities to be in the high income group for an individual aged 20 is:

$$\text{observed probability} = \frac{\text{number in high income}}{\text{total number in a specific age-group}} = \frac{318}{7845} = 0.0405$$

The fitted probability for a person aged 20 to be in the high income group is:

$$prob = \frac{o}{1+o} = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}} = \frac{e^{-3.0228+(0.0510*20)}}{1+(e^{-3.0228+(0.0510*20)})} = 0.1189$$

where o is the odds.
The observed and fitted probabilities for other age-groups are calculated in a similar way.

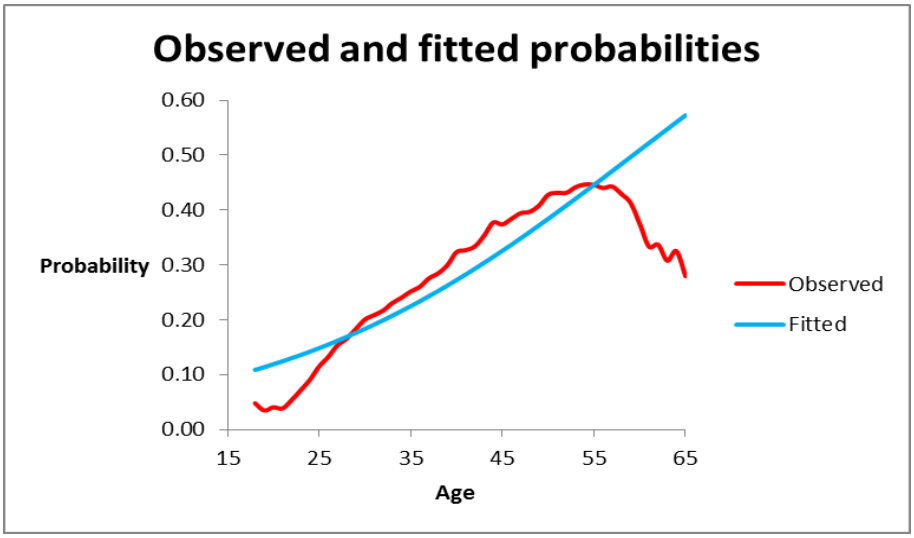


Figure 9: Graph when age is considered as a continuous variable

Erratic patterns in the observed probabilities and fitted probabilities can be inferred from Figure 9. The observed and fitted probabilities are not close to each other in the 18-23 age region and 57-65 age region. However the fitted and observed probabilities are close to each other in the 24-56 age region.

4.7 Logit model for the population group

The SAS code when implementing the logit model with the population group being considered is given below:

```
proc catmod data=finalmod;
model incomegrp = popgrp / ml nogls oneway;
format  popgrp popgrp. income income.  incomegrp incomegrp.;
Contrast 'Black'        popgrp   1  0  0      / est = exp;
Contrast 'Coloured'  popgrp   0  1  0      / est = exp;
Contrast 'Indian'       popgrp   0  0  1     / est = exp;
Contrast 'White'        popgrp  -1 -1 -1     / est = exp;
run;
```

The logit model will also be explained by considering population group variable. The logit model is formulated as:

$$log(o) = \mu + \lambda^{pop} \tag{18}$$

with population as the only effect on income

- o is the odds to be in the high income group
- μ is the effect of the overall odds
- λ^{pop} is the marginal effect of population

	Index	Sample Size(n)
Overall Effect	0.4207	601 059
Population		
Black	0.4335	377 080
Coloured	0.7473	50 556
Indian	1.2976	31 336
White	2.3789	142 087

Table 29: Indices value results

The indices above were obtained directly from SAS.
Calculation of the last estimate

	Estimate
Overall Effect	-0.8659
Population	
Black	-0.8359
Coloured	-0.2913
Indian	0.2605
White	0.8667

Table 30: Estimated values

It follows that, the estimate for the Whites is:

$$\begin{aligned} & -\sum \lambda^{pop} \\ = \quad & -(-0.8359 - 0.2913 - 0.2605) \\ = \quad & 0.8667 \end{aligned}$$

The estimated values are then used to calculate the indices given in Table 29 by making use of the formula:

$$e^{\lambda^{pop}}$$

where λ^{pop} is the estimated value for each population categories
Taking the anti log we get:

$$o = e^{\mu+\lambda^{pop}} = i * i^{pop}$$

with

- i the index of the overall odds
- i^{pop} is the index for various population categories.

4.7.1 Interpretation of the estimates

Population	Chi-Square Value	Pr > Chisq
Black	24329.76	< .0001
Coloured	1163.63	< .0001
Indian	767.65	< .0001
White	23016.64	< .0001

Table 31: Chi Square p values for population

The estimated values inform us that there is a strong positive estimate for the Indian and White people, $\lambda^{indian} = 0.2605$ (from Table 30) and $\lambda^{white} = 0.8667$(from Table 30), both with a probability < 0.0001 (from Table 31). This indicates that the odds to be in the higher income group is significantly higher for Indian and White people than the overall odds. There is a negative estimate for Black and Coloured people, $\lambda^{black} = -0.8359$ and $\lambda^{coloured} = -0.2913$, both with probability < 0.0001. This tells us that the odds of being in the higher income group are quite low for both Black and Coloured people than the overall odds.

Calculation and interpretation of the indices, odds and probabilities

	Odds	**Probability**
Overall Effect	0.4207	0.2961
Population		
Black	0.1824	0.1542
Coloured	0.3144	0.2392
Indian	0.5459	0.3531
White	1.0008	0.5002

Table 32: Estimated odds and Probability results

The geometric mean odds is calculated as:

$$\begin{aligned} & \sqrt[4]{(odds^{black} * odds^{coloured} * odds^{indian} * odds^{white}} \\ \simeq \quad & \sqrt[4]{(0.1824 * 0.3144 * 0.5459 * 1.0008)} \\ \simeq \quad & 0.4207 \end{aligned}$$

This is the same as the overall effect given Table 32. This tells us that for approximately every 4 people in the high income group, we have 10 in the low income group. This figure is higher than what is given in the raw data for the observed two way frequency for population variable (from Table 3) i.e for every 1 individual in the high income group, 3 are in the low income group.

The index for a Black person to be in the high income category is 0.4335 (from Table 29) times higher than the overall odds. This is calculated by:

$$index = e^{estimate} = e^{-0.8359} == i^{black}$$

The the odds to be in the high income group for a Black person is:

$$odds = i * i^{black} = 0.4207 * 0.4335 = 0.1823$$

with i being the index for the overall odds.

The probability of a Black person to be in the high income category is:

$$prob = \frac{o}{1+o} = \frac{0.1835}{1.1835} = 0.1550$$

When compared with the cross tabulation output from SAS under income vs population group from Table 8, we see that these two probabilities are approximately equal.

The odds for the other population groups are calculated in a similar way. From Table 32, the odds to be in the high income group is 0.8176 times lower for a Black person. As we move between the various population groups the odds of being in the high income group increase, as for the Coloured and Indian people it is 0.6856 and 0.4541 lower than the overall odds. The White people have odds of 0.0008 higher than the overall odds of being in the high income group.

4.7.2 The odds ratio

This section looks at various ratios of an individual to be in a high income category given their ethnic group in relation to another ethnic group.

Population	**Odds**	Black	Coloured	Indian	White
Black	0.1823	1.0000	0.5798	0.3339	0.1822
Coloured	0.3144	1.7246	1.0000	0.5759	0.3141
Indian	0.5459	2.9945	1.7363	1.0000	0.5455
White	1.0008	54899	3.1832	1.8333	1.0000

Table 33: Odds Ratios Results

The odds ratios for different population groups are given above. Consider the odds for an Indian to be in the high income group relative to a White person is:

$$\text{odds ratio} = \frac{\text{odds for Indian}}{\text{odds for White}} = \frac{0.5459}{1.0008} = 0.5455$$

Therefore, the odds for an Indian to be in the high income group have decreased by 45.45% when compared to a White person. The other odds ratios in Table 33 are calculated in an analogous manner.

4.8 Logit model without interaction

The SAS code when implementing the logit model with the all the variables considered as independent is given below:

```
proc catmod data=finalmod;
model incomegrp = gender popgrp highest_level_of_education agegrp / ml nogls oneway;
format agegrp agegrp. gender gender. popgrp popgrp. income income. incomegrp incomegrp.
Highest_Level_of_Education Highest_Level_of_Education.;
Contrast 'overall effect' intercept 1 / est = exp;
Contrast 'Male' gender 1 / est = exp;
Contrast 'Female' gender -1 / est = exp;
Contrast 'Black' popgrp 1 0 0 / est = exp;
Contrast 'Coloured' popgrp 0 1 0 / est = exp;
Contrast 'Indian' popgrp 0 0 1 / est = exp;
Contrast 'White' popgrp -1 -1 -1 / est = exp;
Contrast 'Grade 12' Highest_Level_of_Education 1 0 0 0 0 0 / est = exp;
Contrast 'NTC' Highest_Level_of_Education 0 1 0 0 0 0 / est = exp;
Contrast 'Certificate' Highest_Level_of_Education 0 0 1 0 0 0 / est = exp;
Contrast 'Diploma' Highest_Level_of_Education 0 0 0 1 0 0 / est = exp;
Contrast 'Bachelors Degree' Highest_Level_of_Education 0 0 0 0 1 0 / est = exp;
Contrast 'Honours Degree' Highest_Level_of_Education 0 0 0 0 0 1 / est = exp;
Contrast 'Masters / PhD' Highest_Level_of_Education -1 -1 -1 -1 -1 -1 / est = exp;
Contrast '18-25 ' agegrp 1 0 0 0 / est = exp;
Contrast '26-35 ' agegrp 0 1 0 0 / est = exp;
Contrast '36-45' agegrp 0 0 1 0 / est = exp;
Contrast '46-55' agegrp 0 0 0 1 / est = exp;
Contrast '56-65' agegrp -1 -1 -1 -1 / est = exp;
run;
```

The logit model is formulated as:

$$log(o) = \mu + \lambda^{edu} + \lambda^{gen} + \lambda^{pop} + \lambda^{age} \tag{19}$$

where

- o is the odds to be in the high income group
- μ is the effect of the overall odds

- λ^{edu} is the effect of education
- λ^{gen} is the effect of gender
- λ^{pop} is the effect of population group
- λ^{age} is the effect of the age group

4.8.1 Calculation and interpretation of the indices

	Index	**Sample Size(n)**
Overall Effect	0.7359	601 059
Gender		
Male	1.3447	312 396
Female	0.7436	288 663
Population Group		
Black	0.4555	377 080
Coloured	0.9555	50 556
Indian	1.1966	31 336
White	1.9198	142 087
Education Level		
Grade 12	0.2204	365 399
NTC	0.5303	25 197
Certificate	0.4271	26 975
Diploma	0.9957	98 435
Bachelor's Degree	2.0125	51 388
Honours Degree	2.9157	20 083
Masters/PhD	3.4295	13 582
Age		
18-25	0.3071	95 464
26-35	0.9319	214 914
36-45	1.5689	159 739
46-55	1.9766	88 034
56-65	1.1265	42 908

Table 34: Indices value results

The indices reveal the partial effect of the independent variables on the dependent variable, as given in Table 34. A significant difference exists between the 2 gender categories. Accordingly, the partial effect of gender on income is noticeable. The odds to be in the high income group for Males increase by a factor of about 35%, whilst for the Females it is 25.64% as low than the overall odds.

Keeping the other variables constant, the population group indices differ for each ethnic group, thereby suggesting that as we move between different ethnic groups, the partial effect on income is quite significant. As a result, the odds to be in the high income group for Black people decrease by a factor of 54.45%, whilst for the Coloured people, the odds to be in the high income group is 4.45% lower than the overall odds. On the other hand, the odds of being in the high income group for the Indians is about 19.66% higher, and for the White people is almost double than the overall odds.

The indices for education have decreased greatly as compared to when we analysed education on its own(Table 9). An individual with a Grade 12 has odds of 77.96% lower than the overall odds to be in the high income group. The odds for a person with an NTC of being in the high income category is 46.97% lower than the overall odds. A Certificate holder has odds of 57.29% as low than the overall odds of being in the high income group. Whilst the odds for a person with a Diploma to be in the high income group is 0.43% as low than the overall odds. People with either a Bachelor's, Honours or Masters/PhD have odds of 2.0125 times,

2.9157 times and 3.4295 times as high to be in the high income group than the overall odds respectively. The explanation accompanying the reduction in index values for education is that, the effect of other variables in the model is also considered, and education now has a partial effect and not a marginal effect. Thus the effect of education on income has reduced. Furthermore from the results it can be seen that education level explained the most variation in income especially when you look at the Masters/PhD index. However, this result is influenced by a wide range of factors, for example, population group, that is White people could be investing more in their education than other population groups.

The indices in the age-group variable increase until the 46-55 age-group and then decrease in the 56-65 age-group. It can be deduced from the age-group category that the odds for an individual aged between 18-25 to be in the high income group is 3/10 times as high than the overall odds, whereas, the odds for a person aged 26-35 is approximately 9/10 as high than the overall odds. The odds for a person to be in the high income category in the 36-45 age-group is 56.89% higher than the overall odds, for a person aged 46-55 it is 97.66% higher than the overall odds and is 12.65% higher than the overall odds for a person aged between 56-65.

Example 1. Consider the following:

- Male
- Indian
- with a Bachelor's degree
- 26-35 age-group

The estimated odds is:

$$odds = i * i^{gen} * i^{pop} * i^{edu} * i^{agegrp}$$

with

- i the index of the overall odds
- i^{gen} the index for gender
- i^{pop} the index for population group
- i^{edu} the index for the education category
- i^{agegrp} the index for the age-group

It follows that

$$odds = 0.7359 * 1.3447 * 1.1966 * 0.6994 * 0.9319 = 2.2209$$

As a result, the estimated odds for an Indian Male with a Bachelor's degree in the 26-35 age-group to be in the high income group is 2.2209 times higher than the overall odds.

The probability of an Indian Male with a Bachelor's degree in the 26-35 age-group of being in the high income group category is:

$$prob = \frac{o}{1+o} = \frac{2.2209}{3.2209} = 0.6895$$

Example 2. Consider the following:

- Female
- White
- with an Masters/PhD

The estimated odds is then

$$odds = i * i^{gen} * i^{pop} * i^{edu}$$

Therefore

$$odds = 0.7359 * 0.7436 * 1.9198 * 3.4295 = 3.6028$$

Therefore the estimated odds for a White Female with a Masters/PhD degree to be in the high income group is 3.6028 times higher than the overall odds.

The probability of a White Female with a Masters/PhD of being in the high income category is:

$$prob = \frac{o}{1+o} = \frac{3.6028}{4.6028} = 0.7827$$

From Example 2, it can be deduced that it is not necessary to include all the variables in a model when deducing certain analytic results.

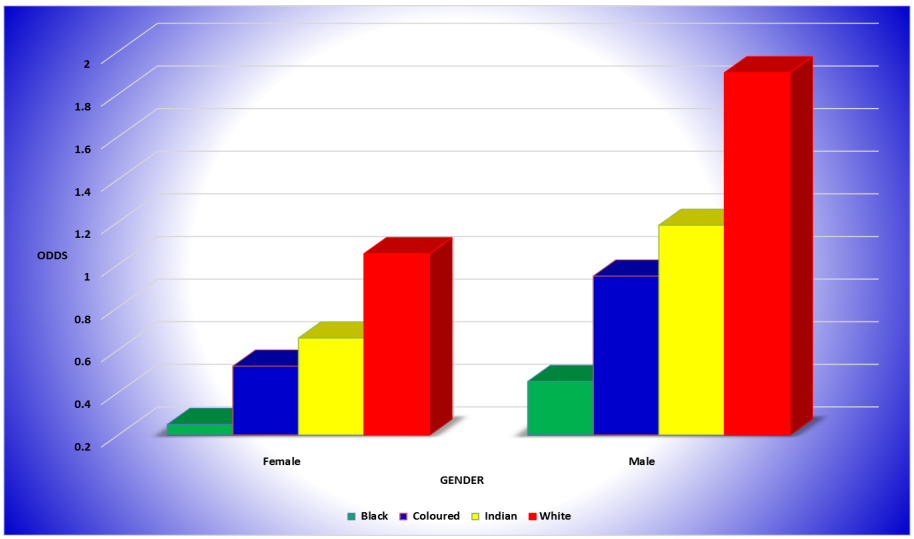


Figure 10: Odds when population and gender are considered

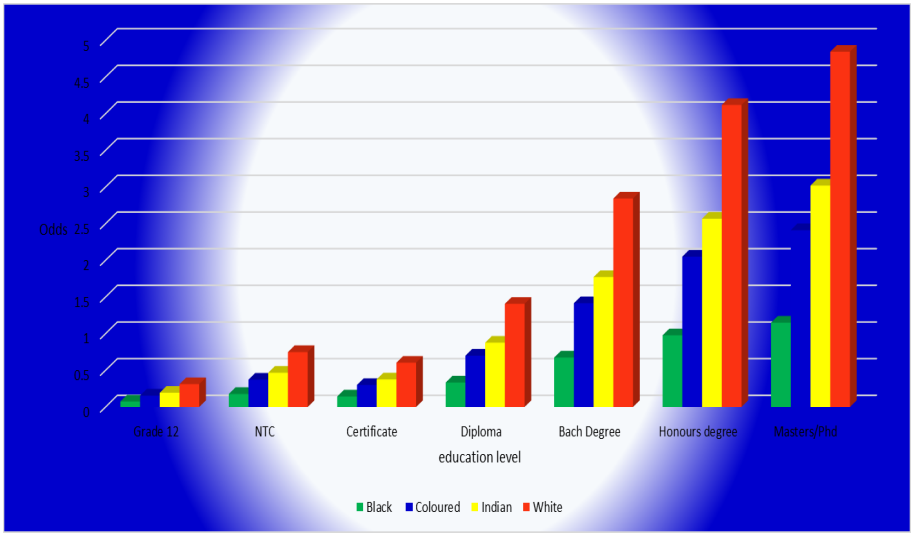


Figure 11: Odds when education and population are considered

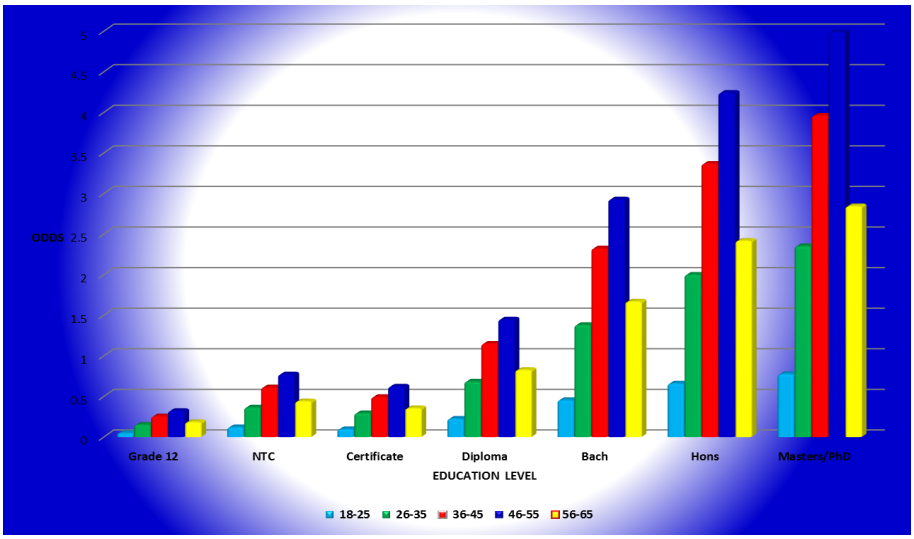


Figure 12: Odds when age-group and education are considered

Since we cannot visualize all the independent variables in one graph, three dimensional graphs of various combinations of the variables are shown above. The graphs shown are for:

- Odds vs Population vs Gender - Figure 10
- Odss vs Education vs Population group - Figure 11
- Odds vs Age-group vs Education - Figure 12

4.8.2 Odds ratio

The odds ratio are calculated by using the formula:

$$\text{odds ratio} = \frac{\text{odds of a particular category}}{\text{odds of interest category}}$$

Variable			
Gender	**Odds**	Male	Female
Male	0.9896	1.0000	1.8085
Female	0.5472	0.5530	1.0000

Table 35: Odds ratios for gender

From Table 35, the odds ratio for Female to be in the high income group relative to a Male is:

$$\text{odds ratio} = \frac{\text{odds Female}}{\text{odds Male}} = \frac{0.5472}{0.9896} = 0.5530$$

Consequently, the odds of a Female to be in the high income group relative to a Male is 44.7% lower. Odds ratio for Male to be in the high income group relative to a Female is:

$$\text{odds ratio} = \frac{\text{odds Male}}{\text{odds Female}} = \frac{0.9896}{0.5472} = 1.8085$$

The odds for a Male to be in the high income group relative to a Female is 80.85% times higher. An inverse relationship exists between the odds of the Male being in the high income group relative to the Female i.e

$$\text{odds ratio} = \frac{\text{odds Female}}{\text{odds Male}} = \frac{0.5472}{0.9896} = \frac{1}{\text{odds ratio Male}} = \frac{1}{1.8085} = 0.5530$$

and vice versa

$$\text{odds ratio} = \frac{\text{odds Female}}{\text{odds Male}} = \frac{0.5472}{0.9896} = \frac{1}{\text{odds ratio Female}} = \frac{1}{0.5530} = 1.8085$$

Variable	**Odds**				
Population group		Black	Coloured	Indian	White
Black	0.3352	1.0000	0.4768	0.3808	0.2373
Coloured	0.7031	2.0974	1.0000	0.7985	0.4978
Indian	0.8806	2.6266	1.2523	1.0000	0.6233
White	1.4127	4.2139	2.0091	1.6043	1.0000

Table 36: Odds ratios for population groups

Variable	**Odds**							
Education Level		Grade 12	NTC	Certificate	Diploma	Bachelor	Honours	Masters/PhD
Grade12	0.1622	1.0000	0.4156	0.5159	0.2213	0.10949	0.0756	0.0643
NTC	0.3902	2.4063	1.0000	1.2415	0.5325	0.2635	0.1819	0.1546
Certificate	0.3143	1.9383	0.8055	1.0000	0.4290	0.2122	0.1465	0.1245
Diploma	0.7327	4.5185	1.8778	2.3312	1.0000	0.4947	0.3415	0.2903
Bachelor	1.4810	9.1331	3.7954	4.7112	2.0213	1.0000	0.6903	0.5868
Honours	2.1456	13.2315	5.4986	6.8264	2.9283	1.4487	1.0000	0.8501
Masters/PhD	2.5237	15.5634	6.4678	8.0295	3.4444	1.7041	1.1762	1.000

Table 37: Odds ratios for education level

Variable	Odds					
Age		18-25	26-35	36-45	46-55	56-65
18-25	0.2260	1.0000	0.3296	0.1958	0.1554	0.2726
26-35	0.6858	3.0344	1.0000	0.5940	0.4715	0.8272
36-45	1.1545	5.1085	1.6835	1.0000	0.7937	1.3926
46-55	1.4546	6.4360	2.1210	1.2599	1.0000	1.7545
56-65	0.8290	3.6682	1.2089	0.7181	0.5700	1.0000

Table 38: Odds ratios for age-groups

The results for the odds ratios for population group, education level and age-group in Tables 36, 37 and 38 are calculated and interpreted similarly as illustrated by the gender variable.

4.9 Income via the random midpoint method

Since income was given as grouped variable, it will be looked at it from a continuous perspective. A midpoint is considered as an upper bound for each income group. The result is that we end up with random income values.

The SAS code used when considering all the independent variables via the random midpoint method is as follows:

```
data incran;
set finalmod;
u =ranuni(0);
if income = 2 then inc = 200.5 + ((200.5-1)*u);
if income = 3 then inc = 600.5 + ((600.5-401)*u);
if income = 4 then inc = 1200.5 + ((1200.5-801)*u);
if income = 5 then inc = 2400.5 + ((2400.5-1601)*u);
if income = 6 then inc = 4800.5 + ((4800.5-3201)*u);
if income = 7 then inc = 9600.5 + ((9600.5-6401)*u);
if income = 8 then inc = 19200.5 + ((19200.5-12801)*u);
if income = 9 then inc = 38400.5 + ((38400.5-25601)*u);
if income = 10 then inc = 76800.5 + ((76800.5-51201)*u);
if income = 11 then inc = 153600.5 + ((153600.5-102401)*u);
if income = 12 then inc =  215041 + ((215041-204801)*u);
run;
proc glm data=incran plots(maxpoints = 1000000);
class agegrp highest_level_of_education popgrp gender;
model inc = agegrp highest_level_of_education popgrp gender / solution;
lsmeans agegrp highest_level_of_education popgrp gender/ pdiff;
format  agegrp agegrp. gender gender. popgrp popgrp.
Highest_Level_of_Education Highest_Level_of_Education. ;
run;
```

The model to be implemented is:

$$Y_i = \alpha + \beta_1 x_{edu} + \beta_2 x_{gen} + \beta_3 x_{pop} + \beta_4 x_{age} \tag{20}$$

where

- $x_{edu}, x_{gen}, x_{pop}$ are categorical variables for education, gender and population
- x_{age} is the midpoint of the various age-group categories
- α is the intercept parameter

- β_1 is the regression coefficient for education
- β_2 is the regression coefficient for gender
- β_3 is the regression coefficient for population
- β_4 is the regression coefficient for age
- Y_i is the income for individual i

The CLASS statement names the classification variables to be used in the model. In this case age, education, population group and gender are labelled in the class statement. The GLM(General Linear Modelling) procedure will then display results that summarize the CLASS variables and their respective levels. The Least Squares Means (LSMEANS) statement computes the effect of each of the listed variables vs the dependent variable. LSMEANS will provide the predicted margins, that is, the partial means will be estimated by applying equal weights to all categories for each explanatory variable, for a sample that is unbalanced. This will enable us to compare the income of the different categories for the various independent variables. PDIFF statement requests the p-values for differences of the LSMEANS be produced. Thus hypothesis testing is being performed to test if significant differences exist between the categories of different variables [1].

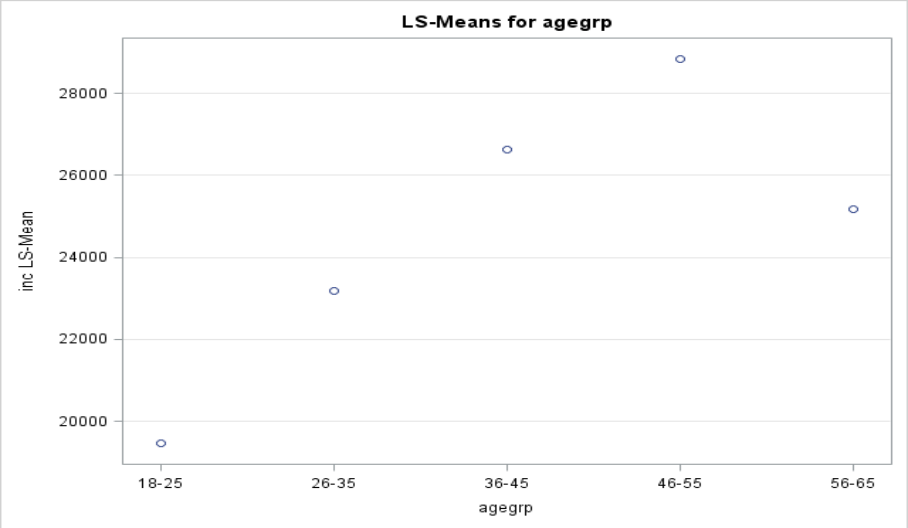


Figure 13: Graph for the LS Means for age group

A quadratic effect can be seen from Figure 13, when income is considered continuous against age. From the graph it can be deduced that the average income for a people aged between 18-25 is about 20 000 a month. The average income increases as age increases steadily with, age-groups 26-35 and 36-45 averaging about 23 500 and 26 000 respectively per month, and with the age group 46-55 receiving the highest mean monthly income, which is pegged at over 28 000 . Income then decreases slightly in the 56-65 age-group , where average income is about 25 000 monthly.

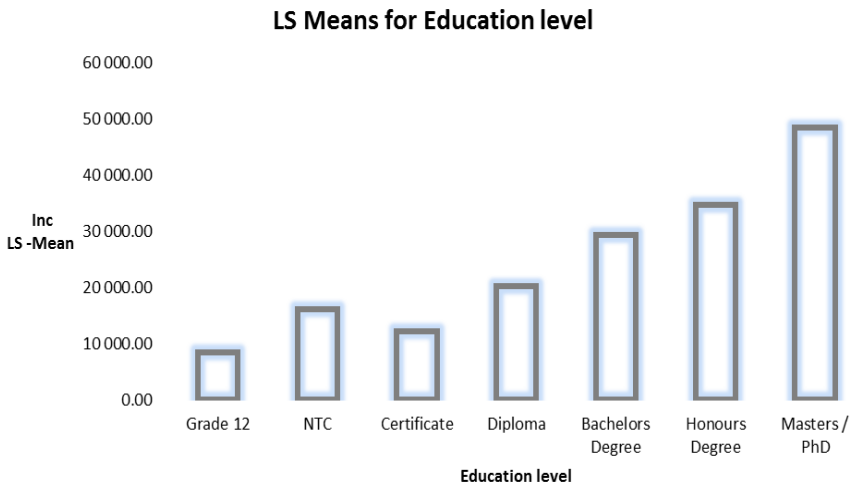


Figure 14: Graph for the LS Means for education level

An upward trend, can be inferred from the different levels of education from Figure 14. Drastic differences in income exist between the different education levels. This is most noticeable from the graph as a person with a Grade 12 earns the least income, with a mean monthly income of just under 10 000. People with a Certificate or NTC earn more less the same, as they earn on average 15 000 a month. Individuals with a Diploma average just over 20 000 a month. A spike in income is then noticed as people with a Bachelor's earn 3 times more than people with a Grade 12, whilst individuals an Honours earn on average 3.5 times more than people with a Grade 12. Masters/PHd graduates earn approximately 3 times more than Diploma or NTC holders.

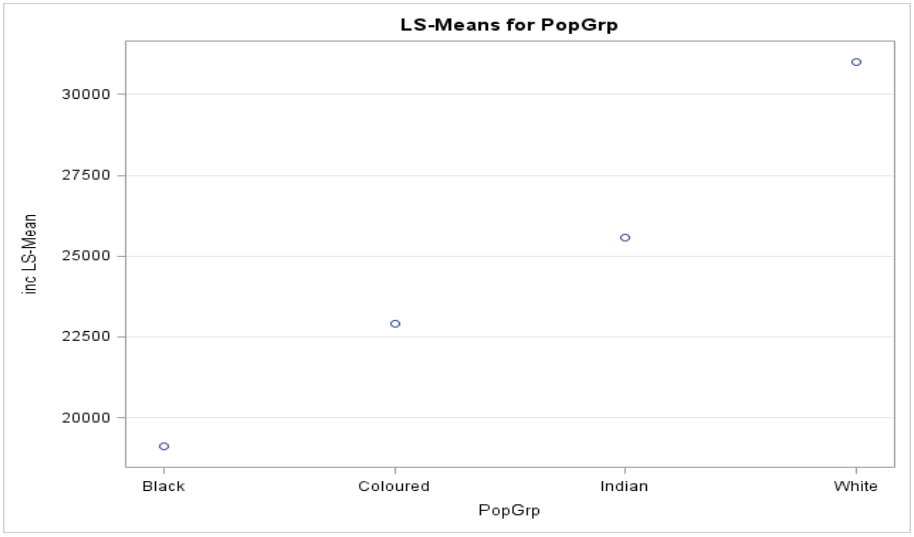


Figure 15: Graph for the LS Means for population

Figure 15 proposes that, significant differences exist between the different population groups. Black people earn the least as they earn a monthly income of 19 000. Coloured people earn on average 18.5% more than their Black counterparts, while Indian people earn 11% more than Coloured people. A sharp rise in income is noticed where White people are concerned, as they earn 24.08% more than Indian people.

Figure 16: Graph for the LS Means for population

The gender of an individual has radical effects on income as Females average approximately 22 300 monthly. This figure is much lower than that of their Male counterparts, who average about 27 000 a month that is 21.076% more, as pointed out in Figure 16.

N.B - the results derived for this section are random, thus if the procedure is to be run again in SAS different results will be obtained. This is due to a specified seed value of zero in the RANUNI function.

5 Conclusion

How income is distributed with regards to various factors is of critical interest to South Africa. It is crucial for relevant stakeholders to know how certain factors influence income so that corrective action, where appropriate, can be taken.

Practical Application was done on the 10% sample of the South African census of 2011. The argument as to why income was given in a grouped format as opposed to exact figures, was to lower the rate of nonresponse. The analysis implored certain restrictions on the data. This then resulted in data that was relevant and practical for analysing grouped income by considering certain factors. How and why these restrictions were put into place was also explained. One-way frequency tables were given, so that we could see how income is distributed by looking various income groups. Income was classified as being low if a person earned 12 000 and below a month, and high if a person earned 12 801 and above per month. This resulted in approximately 75% of our sample being people that earn less than or equal to 12800 a month.

Cross-classification of income against various independent variables was then studied. Notable trends were noticeable in the cross classifications especially where age and education were of concern. The principles underlying the logit and logistic regression model were then illustrated, via examples, with the marginal and partial effect of the independent variable(s) on the dependent variable being studied. Models were built that enabled the calculation of probabilities for individuals to be in a certain income group when certain characteristics pertaining to the individuals were satisfied. These probabilities were calculated via the odds model. The odds model was derived from the index values. Interpretations and how to calculate the indices, odds and probabilities were also explained and illustrated.

From the research it was noted that education has quite an effect on income. This was due to how the odds for an individual to be in the high income group increases as one becomes more educated. Income was then considered as a continuous variable against various independent factors.

Shortfalls of the research included deciding which variables have the largest influence in our model to analyse income. A number of underlying assumptions have to go into the defining the focus group as accurately as possible. A large sample is required for logistic regression. Thus the more explanatory variables

we have, the larger our sample size should be. The disadvantage of attempting to analyse income as a continuous variable is that, the data is not real data, since we made use of the random midpoint method so that estimated values could be derived. This would make it very difficult to predict the exact income that one might earn as the estimates are continuously changing.

Possible areas of future studies could be analysing grouped data by making use of quantile regression models. This is when functional relations between the variables are estimated. The advantage is that we can estimate distribution of income at any quantile. Quantile regression overcomes the following problems which might be problematic for ordinary least squares (OLS):

1. Error terms that are not constant across the distribution. However this violates the homoscedastic property.

2. Not sensitive to extreme outliers, which could distort results significantly in OLS.

Another area of research is to analyse grouped data by making use of discriminant analysis. For this to be implemented would require that our dependent variable be categorical and that our independent variables be continuous. However assumptions about the distribution of the predictor will need to be made. Consequently results from discriminant analysis can then be compared with regression results. This is done to see which methodology analyses grouped data better.

References

[1] The information for this paper was generated from SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Instiute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC , USA.

[2] The code/output and data analysis for the grouped income distribution of the 2011 Census of South Africa for this paper was generated using SAS/STAT software, Version 9.4 of SAS System for Windows. Copyright © [2015] SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute., Cary, NC, USA.

[3] This was extracted from the website of the South African labour guide,website address is http://www.labourguide.co.za/most-recent/1814-when-must-employees-retire.

[4] Adcorp. Jobs down and tax evasion up. Retrieved from http://www.adcorp.co.za/NEws/Pages/DecemberdatareflectSAjobsgrowthof12556decline.aspx, 2013.

[5] John Bibby. Measures of social mobility based on income inequality measures. *Quality and Quantity*, 14(5):619–633, 1980.

[6] Martin Bland and Douglas G Altman. The odds ratio. *British Medical Journal*, 320(7247):1468, 2000.

[7] AJ Christopher. Headlining the release of South Africa's census 2011 results. *South African Geographical Journal*, 96(2):166–179, 2014.

[8] Aline Coudouel, Jesko S Hentschel, and Quentin T Wodon. Poverty measurement and analysis. *A Sourcebook for Poverty Reduction Strategies*, 1:27–74, 2002.

[9] G Crafford and NAS Crowther. Linear models for grouped data: theory and methods. *South African Statistical Journal*, 43(2):151–176, 2009.

[10] Gretel Crafford. *Statistical analysis of Grouped data*. PhD thesis, University of Pretoria, 2007.

[11] Greg J Duncan and Eric Petersen. The long and short of asking questions about income, wealth, and labor supply. *Social Science Research*, 30(2):248–263, 2001.

[12] Heather J Gibson, Matthew Walker, Brijesh Thapa, Kyriaki Kaplanidou, Sue Geldenhuys, and Willie Coetzee. Psychic income and social capital among host nation residents: A pre–post analysis of the 2010 fifa world cup in south africa. *Tourism Management*, 44:113–122, 2014.

[13] Barbara G.Tabachnick and Linda S.Fidell. *Using Multivariate Statistics*. Pearson, 2012.

[14] Sharon Lohr. *Sampling: design and analysis*. Cengage Learning, 2009.

[15] Jeanine Elizabeth Malherbe. An analysis of Income and Poverty in South Africa. Master's thesis, Stellenbosch: University of Stellenbosch, 2007.

[16] Peter McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42(2):109–142, 1980.

[17] Chao-Ying Joanne Peng, Kuk Lida Lee, and Gary M Ingersoll. An introduction to logistic regression analysis and reporting. *The Journal of Educational Research*, 96(1):3–14, 2002.

[18] Stats SA Pretoria. Introduction to census 2011 metadata. The information, products and services from Stats SA which are protected in terms of the Copyright Act, 1978 (Act 98 of 1978).

[19] Ann Strode, Catherine Slack, and Zaynab Essack. Child consent in South African law: Implications for researchers, service providers and policy-makers. *SAMJ: South African Medical Journal*, 100(4):247–249, 2010.

[20] Fiona Tregenna and Mfanafuthi Tsela. Inequality in South Africa: The distribution of income, expenditure and earnings. *Development Southern Africa*, 29(1):35–61, 2012.

[21] Dieter Von Fintel. Earnings bracket obstacles in household surveys-how sharp are the tools in the shed. Department of Economics, Stellenbosch University. Stellenbosch Working Paper Series No. WP08/2006 Publication date: 2006.

[22] Andrew Whiteford. The distribution of income in South Africa. *Human Science Research Council, Pretoria*, 1994.

[23] Richard Williams. Generalized ordered logit/partial proportional odds models for ordinal dependent variables. *Stata Journal*, 6(1):58–82, 2006.

6 Appendix

The PROC FORMAT was invoked into SAS so that the data could be easily read as the various categories of the variables are given in numerical form.

```
proc format;
value gender
1='Male'
2='Female'
;
value popgrp
1='Black'
2='Coloured'
3='Indian'
4='White'
5= 'Other '
;
value income
01 = 'No income'
02 = 'R1 - R400'
03 = 'R401 - R800'
04 = 'R801 - R1 600'
05 = 'R1 601 - R3 200'
06 = 'R3 201 - R6 400'
07 = 'R6 401 - R12 800'
08 = 'R12 801 - R25 600'
09 = 'R25 601 - R51 200'
10 = 'R51 201 - R102 400'
11 = 'R102 401 - R204 800'
12 = 'R204 801 or more'
;
value incomegrp
1 = 'high'
2 = 'low'
;
value agegrp
1 = '18-25'
2 = '26-35'
3 = '36-45'
4 = '46-55'
5 = '56-65'
;
value Highest_Level_of_Education
00 = 'Grade 0'
01 = 'Grade 1 '
02 = 'Grade 2 '
03 = 'Grade 3 '
04 = 'Grade 4 '
05 = 'Grade 5 '
06 = 'Grade 6 '
07 = 'Grade 7 '
08 = 'Grade 8 '
09 = 'Grade 9 '
10 = 'Grade 10 '
```

```
11 = 'Grade 11 '
12 = 'Grade 12 '
13 = 'NTC '
14 = 'NTC '
15 = 'NTC '
16 = 'NTC '
17 = 'NTC '
18 = 'NTC '
21 = 'Certificate'
22 = 'Diploma'
23 = 'Diploma'
24 = 'Diploma'
25 = 'Bach Degree'
26 = 'Bach Degree'
27 = 'Honours degree'
28 = 'Masters / PhD'
29 = 'Other'
98 = 'No schooling'
99 = 'Unspecified '
;
```

Inference: Testing for equality of sample means or variances given directional data

Tichawona Mutoro 11327457

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mr. M.T Loots, Co-supervisor: Ms I.A Maharela

Department of Statistics, University of Pretoria

2 November 2015 (Final Report)

Abstract

This research report is focused on how we can do inference on directional data. That is directional statistics. We are particularly interested in how we can test for equality of sample means, or variance on directional data. In this case we are interested in observations that are not on a straight line but observations that are directional in nature. The purpose of the research is to allow us to be able to compare means and variances of directional data by methods of hypothesis testing, say, given two samples of data, we should be able to test for the equality of means or homogeneity of variance.

Declaration

I, *Tichawona Mutoro,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Tichawona Mutoro

_____
Mr Theodor Loots

_____
Ms Iketle A Maharela

_____
2 November 2015

Contents

1 Introduction

Inference is all about using statistical analysis for decision making purposes. In this research report we intend to use distributions which apply to directional data in order to carry out the calculations for test statistics, confidence intervals, means and standard deviations. The idea behind all this is to be able use hypothesis testing for equality of means or variances.
When dealing with directional data, we are dealing with data that is on angular propagations. This is data of directional movements, angular orientations or displacements. Periodic data can also be transformed to representative angles for example data that is recorded on an hour of the day. [2] [7]
This research considers observations that have a certain direction, rotated or that are axial in nature. Axial data is closely related to circular data and can be converted to circular data by doubling the angles θ to 2θ. When we talk of axial data we talk of observations on the circle for which we consider each direction to be equivalent to the opposite direction such that the angle θ is equivalent to the angle $\theta + 180^o$. In other words, coordinates of opposite angles are identical and therefore θ and $\theta + 180^o$ are identical. As such, the measurement is usually in degrees and sometimes in radians. Observed data can therefore be seen as points on a circle of unit radius or a unit vector. [7] [11] [3]
The examples of distributions applicable to directional data are: Von Mises-Fisher distributions, Uniform distribution, Brownian motion distributions, Kent distributions, Fisher-Watson distributions, Bingham-Mardia distributions, Wood distribution and the projected distribution. These are distributions on spheres, that is, on spherical data. Spherical data is observed in three dimensions. This data can be points on the sphere or ordinary multivariate data. We need to have an understanding of spherical statistical techniques when dealing with such data. Sometimes the observations are not directional in nature but axial,that is, observations are axes and not directions. In such a case we use axial distributions namely; the Watson distribution, Bingham distribution and angular central Gaussian distributions. When observations are axes and not directions, the observed unit vectors $\pm\mathbf{X}$ can not be distinguished and therefore it will not be appropriate to use spherical probability density functions for $\mathbf{X}$ but to consider those on S^{p-1} which are antipodically symmetric, that is; $f(-\mathbf{x}) = f(\mathbf{x})$. S^{p-1} denotes a $(p-1)$-dimensional hypersphere. Directional data is also referred to as circular data and is modeled by circular models namely Lattice distributions, uniform distributions, Von Mises distributions, Cardioid distributions, Projected Normal distributions and Wrapped distributions. [7]

Applications of directional data

We consider the direction of waves in the ocean. In earth sciences the data arises from the spherical surface of the earth. Topics of interest involve geology and palaeomagnetic fields. For studies in biology directional statistics involves the study of animal movement or navigation and also their preferred direction of migration during a certain season. Another good example is meteorology where we look at the studies of wind direction, wind speed, times of the day at which it will thunder and the times of the year when heavy rains pour. Also in medicine it is such that circular data arises from the times of the year at which a disease would cause deaths. In vector cardiology we have vectorial or spherical data, that is, information concerning the electrical activity of the heart, as described in three dimensions. Furthermore, the compass and the clock are used as principal measuring instruments when it comes to circular data.
The compass measures the directions of migrating birds when they vanish from a particular point and it also measures wind direction. The clock can be used for observations on the arrival times of patients at an emergency section of a given hospital. We also consider the spinning of a roulette wheel and observing its position when it stops. This shows that circular observation can therefore be regarded as a direction in the plane. These are just some but not all the examples of directional or circular data applications. [7] [8]

Why we should study this data

Directional data can be useful in observations that are unit vectors in two and three dimensions. This is different from observations on a straight line. Directional statistics can be of significant importance in weather studies or weather report for predictive statistics of wind movement and direction of ocean waves. We can use this data in the study of animal direction to find out whether the directions of these animals follow a particular type of distribution or not. It is also possible to find out if animals make use of clues like the

earth's magnetic fields or the direction of the sun when they tend to head towards these directions. When studying astronomy, the distribution of objects like stars on the celestial sphere calls for the understanding of directional data. These stars can be uniformly distributed around the celestial sphere. [7] [11]

2 Background Theory

In order to carry out tests on the equality of means and variance we need to have an understanding of the summary statistics with respect to the measure of location and dispersion when directional data is given. Summary statistics are important for the calculation of the test statistics to be used in the hypothesis testing.

Understanding of summary statistics given directional (or circular) data

The measure of location is the mean direction. The summary statistics can be constructed appropriately by regarding points on the circle as unit vectors in a plane. We therefore take polar coordinates of the sample mean of these vectors. We regard directions in the plane as points on the circle or unit vectors:
$\mathbf{x}=(\cos\theta,\ \sin\theta)'$ where $\mathbf{x}$ is the unit vector that represents directions in the plane or points on the unit circle and θ is the angle that specifies each circular observation from the initial direction to the point on the circle corresponding to the observation. Thus, the directions can be represented as angles and sometimes as complex numbers but our focus is on angular measurements. [7] [2]

We can consider summary statistics in directional statistics as different from that of observations on the straight line since we are now dealing with directions. For example instead of looking at the mean weight in observations on a straight line we are now taking a look at the mean direction of a certain species of birds, say, in the equatorial forests of Africa.
Since we regard the observations as unit vectors on spheres and on planes, we can have vectors $\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \mathbf{X}_n$ with their corresponding angles $\theta_1, \theta_2, ..., \theta_n$. We then calculate the mean direction as:

$$\bar{\theta} = \tan^{-1}\left(\frac{\bar{S}}{\bar{C}}\right), \text{ if } \bar{C} \geq 0 \qquad (1.1)$$

and

$$\bar{\theta} = \tan^{-1}\left(\frac{\bar{S}}{\bar{C}}\right) + \pi, \text{ if } \bar{C} < 0 \qquad (1.2)$$

where

$$\bar{C} = \frac{1}{n}\sum_{j=1}^{n}\cos\theta_j$$

and

$$\bar{S} = \frac{1}{n}\sum_{j=1}^{n}\sin\theta_j$$

with $-\frac{\pi}{2} \leq \tan^{-1}(\bar{S}/\bar{C}) \leq \frac{\pi}{2}$.
We have that $(\bar{C},\ \bar{S})$ are the cartesian coordinates of the centre of mass where the centre of mass is given by $\bar{\mathbf{x}} = (\mathbf{x}_1 + \mathbf{x}_2 + ... + \mathbf{x}_n)/n$.
It should be noted that the mean direction $\bar{\theta}$ of $\theta_1, \theta_2, ..., \theta_n$ is not equal to $(\theta_1 + \theta_2 + ... + \theta_n)/n$ but is the direction of the resultant observation vector$\mathbf{x}_1 + \mathbf{x}_2 + ... + \mathbf{x}_n$.
The other important measurement is the mean resultant length $\bar{R}$ given by the formula:

$$\bar{R} = \left(\bar{C}^2 + \bar{S}^2\right)^{\frac{1}{2}}$$

and we note that $\bar{\theta}$ is defined as mentioned in equations (1.1) and (1.2) above for $\bar{R} > 0$ and not defined for $\bar{R} = 0$. The resultant length $R = n\bar{R}$. [7]

The measure of concentration of directional data is given by the mean resultant length$\bar{R}$, that is, we can refer to the spread of directional data as concentration therefore we will be considering the hypothesis testing of the concentration parameter. We cannot regard this kind of data as evenly spread around the circle but we will use the mean resultant length to discuss the concetration or level of clustering of circular data. We have that the directions $\theta_1, \theta_2, ..., \theta_n$ are very much clustered for $\bar{R}$ almost equal to 1 and widely dispersed for $\bar{R}$ almost equal to 0. [7]
Circular sample variance V is defined as $V = 1 - \bar{R}$ where $\bar{R}$ is the mean resultant length. $V = 1 - \bar{R}$ implies that circular variance is a function of the mean resultant length $\bar{R}$ and therefore a function of the resultant length since $R = n\bar{R}$. Thus, we can describe dispersion in terms of the resultant length of the unit vectors $\mathbf{x}$.
We consider the von Mises distributions to be very useful and most convenient when doing inference on directional data. The von Mises distribution $M(\mu, \kappa)$ is a continuous distribution and its density function is given as:
$g(\theta; \mu, \kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa \cos(\theta - \mu)}$ with I_0 being the modified bessel function of the first kind and order zero such that $I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} e^{\kappa \cos\theta} d\theta$ which has got the power series expansion given by

$$I_0(\kappa) = \sum_{r=0}^{\infty} \frac{1}{(r!)^2} \left(\frac{\kappa}{2}\right)^{2r}.$$

We have that μ is the parameter that denotes mean direction and κ denotes the concentration parameter. [7]
The mean resultant length $\bar{R}$ which can also be denoted ρ is $A(\kappa)$ where $A(\kappa) = I_1(\kappa)/I_0(\kappa)$ with
$I_1(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} \cos\theta e^{\kappa \cos\theta} d\theta$. The larger the value of κ, the larger the clustering around the mean vector.

Two Sample test for the mean (von Mises)

Suppose that we have two independent random samples $\theta_{11}, \theta_{12}, ..., \theta_{1n_1}$ and $\theta_{21}, \theta_{22}, ..., \theta_{2n_2}$of sizes $n_1$and n_2 distributed like $M(\mu_1, \kappa_1)$ and $M(\mu_2, \kappa_2)$ respectively. Let $\bar{\theta}_1$ and $\bar{\theta}_2$ be the corresponding mean directions in that order with R_1 and R_2 as the respective corresponding resultant lengths. Suppose we take $\bar{\theta}$ to be the mean direction of the combined sample and R to be the corresponding resultant length. The hypothesis testing for the mean direction will be as follows:
$H_0 : \mu_1 = \mu_2$ vs $H_1 : \mu_1 \neq \mu_2$. The concentration parameters are such that $\kappa_1 = \kappa_2 = \kappa$ for some κ unkown. The algorithm for the calculation of the test statistic is as follows

$$\bar{C}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \cos\theta_{ij}$$

$$\bar{S}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \sin\theta_{ij}$$

$$\bar{\theta}_i = \begin{cases} \tan^{-1}(\bar{S}_i/\bar{C}_i) & \bar{C}_i \geq 0 \\ \tan^{-1}(\bar{S}_i/\bar{C}_i) + \pi & \bar{C}_i < 0 \end{cases}$$

$$\bar{R}_i = \left(\bar{C}_i^{\,2} + \bar{S}_i^{\,2}\right)^{\frac{1}{2}}$$

$$R_i = n_i \bar{R}_i$$

We have that $R^2 = R_1^2 + R_2^2 + 2R_1R_2\cos(\bar{\theta}_2 - \bar{\theta}_1)$.

Decision criterion: For $R_1 + R_2 \geq R$ we reject H_0 if $R_1 + R_2 - R$ is large where $R_1 + R_2 - R \sim \chi^2_1$, that is, we reject H_0 if $R_1 + R_2 - R > \chi^2_{1,\frac{\alpha}{2}}$. Note that α is our level of significance of the test. Since we are dealing with a chi-square test, all values of $R_1 + R_2 - R$ must be positive, otherwise the test does not hold for $R_1 + R_2 \leq R$ since $R_1 + R_2 - R$ becomes negative.
Conclusion: If we reject H_0 the conclusion is that the mean direction of the 2 samples differ otherwise the mean direction is the same.

In general, these tests are similar to t tests on straight line observations which are commonly used in the world. The only difference is that we are now dealing with directions, hence the shift in our methods of computing the summary statistics even though the procedure is similar in a way. This is because we will not be able to attain the required results relevant to this type of data if we are to implore the conventional t-tests which are usually used in the real world.

Testing for the equality of concentration parameters (von Mises)

Using the same statistics as calculated above in testing for equality of mean directions, tests can also be performed to compare the dispersion of the data by comparing the concentration of parameters.
The hypothesis is formulated as $H_0 : \kappa_1 = \kappa_2$ vs $H_1 : \kappa_1 \neq \kappa_2$ and κ_1,$\kappa_2$are unknown parameters. [7]
To do this we have 3 cases that we have to deal with:

Case I : $\bar{R} < 0.45$
The test statistic is calculated as follows: Let $z = \frac{2}{\sqrt{3}} \frac{g_1(2\bar{R}_1) - g_1(2\bar{R}_2)}{\{1/(n_1-4)+1/(n_2-4)\}^{\frac{1}{2}}} \sim N(0,1)$ under H_0 and $g_1(2\bar{R}_i) = \sin^{-1}\left(2\sqrt{\frac{3}{8}}\bar{R}_i\right)$.
Thus we reject H_0 if $z > z_{\frac{\alpha}{2}}$ or if $z < -z_{\frac{\alpha}{2}}$ since its is a two-tailed test.

Case II :$0.45 \leq \bar{R} \leq 0.7$
The test statistic is calculated as follows: Let $z = \frac{g_2(\bar{R}_1) - g_2(\bar{R}_2)}{0.893\{1/(n_1-3)+1/(n_2-3)\}^{\frac{1}{2}}} \sim N(0,1)$ under H_0, and $g_2(\bar{R}_i) = \sinh^{-1}\frac{\bar{R}_i - c_1}{c_2}$ where $c_1 = 1.089$ and $c_2 = 0.258$. [7]
Thus we reject H_0 if $z > z_{\frac{\alpha}{2}}$ or if $z < -z_{\frac{\alpha}{2}}$ since its is a two-tailed test.

Case III :$\bar{R} > 0.7$
The test statistics will be calculated as follows: Let $F = \frac{(n_1 - R_1)/(n_1-1)}{(n_2-R_2)/(n_2-1)} \sim F_{n_1-1,n_2-1}$ [7]
Thus we reject H_0 if $F > F_{n_1-1,n_2-1,\frac{\alpha}{2}}$ or $F < F_{n_1-1,n_2-1,1-\frac{\alpha}{2}}$

The restrictions for $\bar{R}$ with $0.45 \leq \bar{R} \leq 0.7$ and $\bar{R} > 0.7$ are chosen such that our tests are based on the variance-stabilising transformations and high-concertration approximations.

By not rejecting the null hypothesis in all 3 cases above we conclude that the level of concentration in the data is the same for both samples otherwise the level of concentration differs. In other words we are saying the data of the two samples is either clustered in the same way or in different ways, that is, both samples can have data that is tightly clustered or widely spread.

3 Application

For the application of the above background theory we are going to generate two random data sets from a von Mises distribution and use our calculations from the background theory.
The null hypothesis is formulated as follows: $H_0 : \mu_1 = \mu_2$ vs $H_1 : \mu_1 \neq \mu_2$
The output for the data and code in appendix is as follows:

Using the calculations in the background theory the results are as follows:
> S_bar1 [1] 0.02891771
> S_bar2 [1] -0.01616344
> C_bar1 [1] 0.8496654
> C_bar2 [1] -0.9136483
> theta_hat1 [1] 0.0340211
> theta_hat2 [1] 3.159282
> R_bar1 [1] 0.8501573
> R_bar2 [1] 0.9137913
> R_1 [1] 42.50787
> R_2 [1] 45.68956
> R [1] 3.262087
> Test_statistic [1] 84.93534
> critical_value [1] 5.023886

To obtain the results above we install the Circular package in R and generate two dtata sets of 50 random numbers from a Von Mises distribution using the rvonmises function. The calculation of the test statistic as illustrated above follows from the explanation given in background theory. We then compare this test statistics to the critical value in order to make a decision on whether to reject or not to reject the null hypothesis of the equality of mean direction. Applying the Watson-Williams test for homogeneity of means on these generated data sets yields the results below. In these results we have the test statistic and the p-value which we can use to make the decision on whether to reject the null hypothesis or not.

Using the Watson-Williams test for homogeneity of means
> #Watson-Williams' two sample test on equality of means
> #two sample test on von Mises generated random numbers
> watson.wheeler.test(list(data1,data2))
Watson-Wheeler test for homogeneity of angles
data: 1 and 2 W = 77.845, df = 2, p-value < 2.2e-16

Watson's Two-Sample Test of Homogeneity
The null hypothesis is formulated as follows: $H_0 : \kappa_1 = \kappa_2$ vs $H_1 : \kappa_1 \neq \kappa_2$
> #Watson's two sample test for homogeneity
> #two sample test on von Mises generated random numbers
> watson.two.test(data1, data2, alpha=0.05)
Watson's Two-Sample Test of Homogeneity
Test Statistic: 2.0191 Level 0.05 Critical Value: 0.187
Reject Null Hypothesis
> watson.two.test(data1, data2)
Watson's Two-Sample Test of Homogeneity
Test Statistic: 2.0191 P-value < 0.001

Watson's two-sample test of homogeneity of concentration parameters is an algorithm that directly carries out the hypothesis testing in R using the generated data sets. This algorithm outputs the test statistic, the critical value, the p-value and it specifies whether the null hypothesis has been rejected or not. As in the above output it can be seen that the null hypothesis of equality of concentration parameters was rejected. As the concentration parameter is a measure of dispersion of data, we have that the null hypothesis of equality of variance is rejected.

4 Conclusion

Since the test statistic of $84.93534 > 5.023886$=critical value, we reject the hypothesis of equal mean directions for the generated samples from a von Mises distribution and conclude that the mean directions are not the same. The rejection of the hypothesis of equality of mean directions is also confirmed using the Watson-Williams test for homogeneity of means since p-value $<$ 2.2e-16 which is less than 0.05, our level of significance of the test. We also we reject the hypothesis of equal concentration parameters for the two data sets since we have a very small p-value of less than 0.001 and we conclude that the concentration parameters differ. We can also reject the hypothesis of homogeneity of concentration parameters using the test statistic since the test statistic $2.0191 > 0.187$=critical value.

We therefore conclude that to carry out hypothesis testing on the equality of mean direction given two directional data samples, the approach should differ from that of testing for equality of observations on a straight line. It is important to find the sines and cosines of the given angular measurements in order to compute the apropriate test statistic. The resultant length of the centre of mass of the observed unit vectors will be used in the calculation of the test statistics and it is distributed like a chi-squared random variable which is different from the t-distributed random variable used as the test statistics for hypothesis on equality of means of observations on a straight line. To test for the equality of variance given directional data, it is sufficient to test for the equality of concentration parameters since they are measures of how the data is dispersed.

References

[1] C Agostinelli and U Lund. R package circular: Circular statistics (version 0.4-3), 2011.

[2] Barry C Arnold and Ashis SenGupta. Recent advances in the analyses of directional data in ecological and environmental sciences. *Environmental and Ecological Statistics*, 13(3):253–256, 2006.

[3] Stuart Coles. Inference for circular distributions and processes. *Statistics and Computing*, 8(2):105–113, 1998.

[4] John C Davis and Robert J Sampson. *Statistics and Data Analysis in Geology*, volume 646. Wiley New York, 1986.

[5] Riccardo Gatto and S Rao Jammalamadaka. Inference for wrapped symmetric α-stable circular models. *Sankhyā: The Indian Journal of Statistics*, 65:333–355, 2003.

[6] S Rao Jammalamadaka and Ashis SenGupta. Predictive inference for directional data. *Statistics & Probability Letters*, 40(3):247–257, 1998.

[7] Kanti V Mardia and Peter E Jupp. *Directional Statistics*, volume 494. John Wiley & Sons, 2009.

[8] KV Mardia. Directional statistics and shape analysis. *Journal of Applied Statistics*, 26:949–957, 1995.

[9] Léopold Simar, Anne Vanhems, and Paul W Wilson. Statistical inference for dea estimators of directional distances. *European Journal of Operational Research*, 220(3):853–864, 2012.

[10] Yoko Watamori. Statistical inference of langevin distribution for directional data. *Hiroshima Mathematical Journal*, 26(1):25–74, 1996.

[11] Geoffrey S Watson. The statistics of orientation data. *The Journal of Geology*, 74:786–797, 1966.

Appendix

R-code and data sets output

```
> #two sample test for equality of means

> #two sample test on von Mises generated random numbers

> data1<-rvonmises(n=50, mu=circular(0), kappa=5)

> data2<-rvonmises(n=50, mu=circular(pi),kappa=4)

> x11<-sin(data1)

> x12<-cos(data1)

> x21<-sin(data2)

> x22<-cos(data2)

> S_bar1<-sum(x11)/50

> S_bar2<-sum(x21)/50

> C_bar1<-sum(x12)/50

> C_bar2<-sum(x22)/50

> theta_hat1<-atan(S_bar1/C_bar1)

> theta_hat2<-atan(S_bar2/C_bar2)+(pi) #since C_bar2<0

> R_bar1<-sqrt(C_bar1**2+S_bar1**2)

> R_bar2<-sqrt(C_bar2**2+S_bar2**2)

> R_1<-50*R_bar1

> R_2<-50*R_bar2 > R<-sqrt(R_1**2+R_2**2+2*R_1*R_2*cos(theta_hat2-theta_hat1))

> Test_statistic<-R_1+R_2-R

> critical_value<-qchisq(0.975, df=1)
```

> #Watson-Williams' two sample test on equality of means
> #two sample test on von Mises generated random numbers
> watson.wheeler.test(list(data1,data2))

R-Code
#Watson's two sample test on homogeneity
#two sample test on von Mises generated random numbers
data1<-rvonmises(n=50, mu=circular(0), kappa=5)
data2<-rvonmises(n=50, mu=circular(pi),kappa=4)
watson.two.test(data1, data2, alpha=0.05)
watson.two.test(data1, data2)

Note that the angular measurement in all these calculation is in radians since R gives radians by default. The R- "circular" package was used. We installed it by downloading the zipped folder "circular_0.4-7" from **https://r-forge.r-project.org/R/?group_id=90** and loading it into R through the option "Load package from a zipped folder" then we took the option "load package" and chose "circular."

output

> data1

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] 6.08627261 6.16567649 5.58965229 6.23352539 0.23544852 0.39538369

[7] 5.67654933 0.74448218 5.94014495 0.84257076 0.60157636 6.26076991

[13] 0.21816610 0.03699071 5.55826696 5.84422309 4.62873570 0.62349536

[19] 6.21602372 0.44522361 0.51400588 6.28263554 0.40008706 0.54473692

[25] 0.50098541 0.83869459 6.15248302 0.77659543 0.08384130 5.78229906

[31] 6.14765961 0.11473173 5.60022970 0.48008201 0.19796098 0.51252250

[37] 0.22194368 6.10867929 5.97833379 6.26817245 5.94254522 0.60035486

[43] 0.57045865 3.60414571 5.91832010 5.85621983 6.18785675 5.94938743

[49] 5.89257358 6.15922325

>

> data2

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] 3.160915 3.656959 3.774494 3.379005 2.357756 3.428432 2.945035 3.085984

[9] 2.619290 3.314650 3.834272 3.458254 2.614611 3.109216 3.300945 3.123586

[17] 2.866910 3.179637 3.527551 2.912405 2.581132 3.084231 3.313567 2.589255

[25] 3.212441 3.143919 2.937113 3.273662 2.972145 2.512002 2.764876 3.161843

[33] 3.097521 2.385885 3.437592 3.360655 3.342413 3.408725 3.066903 4.031711

[41] 2.961790 3.802059 3.612721 3.787726 3.985916 3.056892 3.082269 2.722512

[49] 2.097523 3.421538

> x11

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] -0.195642630 -0.117238566 -0.639258031 -0.049639506 0.233279158

[6] 0.385162309 -0.570106907 0.677591073 -0.336351854 0.746356545

[11] 0.565942795 -0.022413520 0.216439562 0.036982271 -0.663074320

[16] -0.425000305 -0.996503105 0.583876406 -0.067111110 0.430659703

[21] 0.491669431 -0.000549768 0.389498528 0.518193106 0.480290086

[26] 0.743771176 -0.130330469 0.700854992 0.083743106 -0.480203109

[31] -0.135111208 0.114480183 -0.631088477 0.461851919 0.196670539

[36] 0.490377189 0.220126044 -0.173621679 -0.300151545 -0.015012297

[41] -0.334090471 0.564935320 0.540018129 -0.446234332 -0.356823382

[46] -0.414110611 -0.095184243 -0.327633636 -0.380754140 -0.123644825

> x12

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] 0.98067526 0.99310378 0.76899231 0.99876720 0.97240981 0.92284885

[7] 0.82157052 0.73543887 0.94173639 0.66554632 0.82444451 0.99974879

[13] 0.97629602 0.99931592 0.74855357 0.90519321 -0.08355574 0.81184256

[19] 0.99774551 0.90251439 0.87078193 0.99999985 0.92102709 0.85526365

[25] 0.87710970 0.66843432 0.99147061 0.71330378 0.99648738 0.87715733

[31] 0.99083044 0.99342553 0.77571086 0.88695705 0.98046963 0.87151031

[37] 0.97547144 0.98481243 0.95389153 0.99988731 0.94254101 0.82513519

[43] 0.84165338 -0.89491615 0.93417187 0.91022657 0.99545967 0.94480485

[49] 0.92467631 0.99232654

> x21

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] -0.019321094 -0.492853952 -0.591486436 -0.235188603 0.706001647

[6] -0.282922238 0.195294837 0.055579542 0.498877116 -0.172195037

[11] -0.638601140 -0.311395180 0.502926752 0.032371465 -0.158678930

[16] 0.018006057 0.271241408 -0.038034919 -0.376447106 0.227186312

[21] 0.531576377 0.057330052 -0.171127780 0.524678469 -0.070789020

[26] -0.002326351 0.203058132 -0.131686188 0.168637786 0.588813837

[31] 0.367869706 -0.020248932 0.044057281 0.685804009 -0.291696270

[36] -0.217314293 -0.199473029 -0.263966666 0.074620385 -0.777146366

[41] 0.178835474 -0.613485122 -0.453892376 -0.602103528 -0.747521748

[46] 0.084599732 0.059288585 0.406921157 0.864457208 -0.276302871

> x22

Circular Data:

Type = angles

Units = radians

Template = none

Modulo = asis

Zero = 0

Rotation = counter

[1] -0.9998133 -0.8701121 -0.8063149 -0.9719498 -0.7082102 -0.9591429

[7] -0.9807446 -0.9984543 -0.8666727 -0.9850629 -0.7695379 -0.9502805

[13] -0.8643290 -0.9994759 -0.9873302 -0.9998379 -0.9625113 -0.9992764

[19] -0.9264381 -0.9738513 -0.8470104 -0.9983553 -0.9852488 -0.8513005

[25] -0.9974913 -0.9999973 -0.9791667 -0.9912915 -0.9856781 -0.8082687

[31] -0.9298773 -0.9997950 -0.9990290 -0.7277863 -0.9565110 -0.9761017

[37] -0.9799033 -0.9645318 -0.9972120 -0.6293199 -0.9838790 -0.7897063

[43] -0.8910565 -0.7984180 -0.6642373 -0.9964150 -0.9982409 -0.9134633

[49] -0.5027064 -0.9610706

The contribution of clickers to student performance and learning environment

Nomawabo Myeki 12318061

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor(s): Ms F Reyneke, Dr L Fletcher

Department of Statistics, University of Pretoria

2 November 2015 (Final)

Abstract

This paper described the audience response system (clickers) used by first-year undergraduate students enrolled for Statistics (STK 110) in the first semester, at the university of Pretoria in a large group setting. The definition of large group is more than 300 students in a lecture hall. The aim of the research project is to evaluate the role that clickers play in improving the classroom environment (increase participation, active engagement), learning (cooperative, interactive) and assessment (immediate feedback, formative).

How these roles impact the student performance in the course throughout the first semester. An experiment is done to compare the exam scores for the STK 110 in 2014 (without clickers) and 2015 (with clickers). The findings are use to compare and to conclude the if there was an positive impact by using the clickers. A survey questionnaire is used to discuss the student perception of the clicker. The background theory of clicker discussion is offered in the paper. The students and lecturers (instructors) face difficult challenges when the clickers are used in the larger classrooms, these include technological hiccups, the students have to adjust to a new way of learning. These and many of the challenges are discussed fully in the paper. Do findings provide evidence that clickers have a positive impact on student's learning as is measured by the exam scores and the questionnaire responses. This will be discussed in the application chapter later using SAS®.
[1]

[1] The [output/code/data analysis] for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.

Declaration

I, *Nomawabo Myeki,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Insert Student's full name Here

_____
Insert Supervisor(s) name(s) here

_____
Date

Acknowledgments

To God be the glory. To my father Tim Omotoso, thank you for your encouragement. The author acknowledges Dr. L Fletcher and Mrs F Reyneke, research supervisors. STK110 students for contributing to the research. Funding acknowledgment: Department of Statistics.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Overview

A student response system (also known as classroom response system, personal response system, or audience response system) is a hardware and software that allows the instructor to pose question to his/her students via computer or overhead projector [3]. The new system has improved over the past few years, instead of taking only multiple choice or True/False questions, the system can be set to take answers that are sentences to but the characters are limited to 120 characters. The students submit the answers using a hand-handled device (a clicker) that beams a radio-frequency signal to the receiver in the instructor's computer. Then the results are instantly summarized by the software and presented on the display screen, using a histogram showing how many students chose each answer. The peers are unable to identify who chose what answer but the instructors are able to identify each student's device number for testing purposes. With immediate feedback the instructor is able to lead the students into a discussion (peer or classroom discussions) concerning the merits of each answer chosen for the question [3].

Audience response system discourages the students' passiveness that is encouraged by the impersonal classroom environment, auditorium seat setting. In the past student evaluation, students have mentioned that they would like to be more actively involved in the large lecture groups. Audience response system (ARSs) are used to improved the student interaction in the classroom, to get student engaged more in class discussions and peer instruction. The use of ARSs provides immediate feedback to the students [10],[11].

The effectiveness of clicker use in the classroom is as good as the instruction question asked, hence the feedback is also good for the instructor so that they can also improve the instruction questions too. On a day-to-day basis instructors are provided with information on student's learning, this is called the formative use of clickers, where instructor test the student's understanding of the course material. For example Svetlana and Gillian recent review on statistics noted that active learning environment is a big part of integrating the principle of learning, which include practice, feedback to name a few [11].

1.2 Terminology

For this paper it was decided that the terms "audience response system"and "clicker" which is the most popular one amongst students and lecturers will be use interchangeably. The lecturers who present the course material are called the instructors. The term teacher or lecturer is mostly associated with the traditional way of presenting a classroom with the course material, were students are more passive. To adopt the term instructor seemed in many studies to embrace the more active role students take when clickers are introduced [4, 3].

1.3 Literature review

Audience response system technology exists back at least to the 1960s [7]. The early technology used was not wireless infrared but instead radio frequency based, it used transmitter and receiver connected by wires. Authors use different terms for this system depending on the use of the system. In the classrooms the term used is student or classroom response system. Other name include electronic voting system, group response system and other names. There seems to be a general consensus regarding the impact ARSs have on the on student learning [3, 4, 7]. The impact depends largely on the instruction method used. To enable active participation and cooperative learning most instructors teaching methods such as small group discussion (peer discussion) and big group discussion (class wide discussion). This typically improve the learning of students over the methods that are more passive. Even though there is a relatively large number of positive number of reviews some reviews have shown the challenges of ARSs: technical knowledge, less time to cover the course material, etc [11].

The main aim of the study is to evaluate the effectiveness of using clicker in enabling an active learning environment. The second one is describing the student's perceptive on using clicker in classrooms. The questions that the research answers are the following:

1. Is there a difference between the student performance of 2015 (with clickers) and 2014 (without clickers) students.

2. What do students think of the use of clicker system in the classroom.

2 Methods

2.1 Design and statistical methodology

The design is an experiment of two-group comparison with one in 2015 using clickers and another in 2014 using the standard way of teaching but the same course material. This experiment allows us to see if there is an impact in the final mark of students. The independent variable is teaching with clicker strategy. The dependent variable is the student's final mark. The t-test is used to test if the average marks in 2015 are higher (significantly different) than 2014 average marks. The logistic regression is used to build a model to predict whether a student will pass or fail. The chi-squared is used to evaluate if there is a signifiacant association betweeen student performance and the use of clickers (by using year since in 2014 no clickers were used and in 2015 clickers were introduced). Regarding the methodology the qualitative data is based on the student questionnaire that is intended to get student perception of using clickers in the classroom. The questionnaire survey data was collected amongst the students who used clickers is collected and analyzed be excel and SAS.

Figure 1: TurningPoint® clicker system

2.2 Participants

The setting for this experiment is the University of Pretoria's first year STK110 students in South Africa. The course is taught in a fixed auditorium seat seating lecture hall style. This style discourages small group class discussions and makes student to sit passively and the lecturer be the presenter of the course material. In each session, out of 5 the students enrolled are ± 300. The course is mainly taught in English. The student had English at secondary school prior to being accepted to university.

2.3 Procedure

On the first day of class each students was instructed to buy a clicker. It was explained that they will not need to buy a textbook. At the beginning of the course students had trial runs of the the clicker system to familiarize themselves with how it works and its functions, this took two sessions for each group. Four members of the statistics department facilitated the course. Each section is taught the same course material content and all the students write the same clicker tests, tutorials, semester tests and final exam. During each class all the sections answer the same clicker or exercise questions on content the instructor covered in class. A variety of questions are presented on the overhead projector of power point projector and the students use the clicker to answer the questions. A bar or histogram chart displays the responses and a discussion is conducted and misunderstandings are clarified.

To complete the qualitative component of the experiment, the student were asked to complete the questionnaire. All the students who got exam entrance we asked to complete the questionnaire in the same venue where they wrote the exam. They were given 10-15 minutes to complete the questionnaire. Through the questionnaire we want to learn the following:

1. If the clicker made class more and exciting.
2. If the clicker questions in helped student learn.
3. If the student would recommend the instructors to continue using clickers.

The survey was administered at the end of the first semester to assess students responses to using clickers. The questionnaire had three sections. The other sections were about MindTap, Excel and other learning tools used during the semester. The question responses were coded as 0 = "Do not agree to 10 = " Agree 100%".

2.3.1 Anonymity and interaction

Students can respond using their clickers without their peers or instructor knowing the answer they chose. Anonymity gives students an opportunity to be engaged and interact in the classroom learning process without feeling like they are being judged. Literature reports that this is the feature that students enjoy more about the clickers. Participation is increased when clickers are used compared to classrooms that do not use clickers, students that would not otherwise participated now do participate.

2.3.2 Assessment benefits

Feedback: In a tradition classroom an instructor can get feedback in a number of ways, including showing of hands, a student volunteer to share his/her answer, use chalkboard, used stick-note or cards. But these ways have disadvantages. In a large classroom when students show their hands it is difficult to get their sense of understanding and takes time to count the hands. Most of the time student do not think about their answers but copy their peers responses.

Using clickers improves the feedback process, making it reliable by guaranteeing anonymity, quick and summarizes students responses indicating a correct answer to the question. This prevents the students from copying from their peers because the is no gain in doing so. When clickers are used students get the opportunity to think about their responses and later defend them to their peers, and also students learn better from their peers sometimes.

Formative assessment: Instructors use formative assessment to determine students' understanding without grading them also understand the misconceptions and misunderstandings and be able to change the classroom instructions next time. When clickers are used regularly, instructor and students are both offered quick, in real time feedback on how the concepts are being understood. Authors agree that using clickers helps provide meaningful and effective formative assessment [4, 6].

2.3.3 Cooperative and active learning

Studies show that when clickers are used on a regular base, frequent and positive interaction occurs. Beatty mentions that when clickers are used there is a great articulation in students' thinking, they ask more probing questions, their focus is increased, more effective peer discussion and active learning [2, 4]. Research shows that students are more interested and involved when clickers are used [8, 5, 3].

2.4 Challenges of using clickers

The challenges in clickers literature that dominates are: technology, instructor and student challenges (these challenges will be taken from the questionnaire). These challenges encountered are discussed below.

2.4.1 Technology challenges

There are three limitations based on technology that were encountered when the clickers were used. Firstly, the instructors got their clickers the same week that the lectures began, so they did not have enough time to prepare and get use to the system on their own. Secondly, the students were responsible for purchasing their own clicker, some of them did not purchase them and could not participate fully in the class. Students did not bring the clickers consistently and clickers were reported lost. Lastly, the most frustrating technological challenge occurred when the instructors computer did not pick up the signal of the clickers [3]. In some of the sessions during clicker test, the instructor's computer shut down and the system was interrupted so this led to the test being hand written. The old venues in the university like chancellor's building could not be used for clicker sessions because the clicker sytem did not work there.

2.4.2 Instructor-based challenges

The main concern when clickers are introduced is the content coverage [3]. Literature suggest that instructors, and students sometimes believe that less of the content is covered when using clickers however for this experiement this was not a concern. Developing good clicker question is the responsible of the instructor, the system is as good as the questions posed to students [11].The characteristics of good questions are: they focus on a specific learning goal, make students aware of other student's opinions as well, expose confusions and misconceptions and elicit a range of responses.

3 Application

3.1 Data analysis and results

3.1.1 Qualitative data

For the qualitative data, SAS and Excel, were used to assist with data capturing and analysis. A total of 2102 students participated in this experiment. Students from the experimental group were 1226, these are the students who used clickers in 2015. There were 1361 students from the control group, these are students who did not use clickers in 2014. The experimental group provided the qualitative data of this experiment: only 950 questionnaires were used other were invalid because they were incomplete. Questions 1, 3,4 and 5 from the questionnaire were grouped together to analyze if the clicker system contributed in the student learning. Students generally thought clickers were a positive addition combined with the traditional way of lecturing. A few of the students thought clickers were fun and exciting, these students did not experienced any problems with their clickers. Multiple comments centered on "clickers being a bad ideas for writing tests", as one student noted " it [using a clicker] adds extra pressure during the test. I'd prefer not to use it." The primary negative attitude toward clicker was as a result of malfunctioning of the system. This will be discussed further under recommendation. Two topics emerged from the questionnaire data. These topics were: being able to get immediate feedback, active involvement in the classroom and being able to respond anonymously. A substantial number of students preferred using a clicker to answer questions in class rather than raising up their hands. The students liked the fact that no one would know who picked the incorrect

answer saving them from being embarrassed. The students perceived the second topic of being able to get immediate feedback as a great idea to confirm their answers immediately. If more than a half of the students got the question wrong the instructor would explain the course material again. The last topic was active involvement, student perceived being involved in class helpful in their learning process. They said during the use of clicker in classroom they were awake.

3.1.2 Final marks performance (Quantitative assessment)

T-test

year	N	Mean	Std Dev	Std Err
2014	1361	61.8913	15.9965	0.4336
2015	1226	65.2602	15.4011	0.4399

Method	Variances	DF	t Value	Pr > |t|
Pooled	Equal	2585	-5.44	<.0001

Table 1: T-test output

The final marks were significantly different for the two years. In table 1, the p-value $< .0001$ therefore the null hypothesis was rejected. Even though the were technological problems and limited time during lecture times to use clickers, there was a significant increase in the final mean marks of 2015 with clickers compared with 2014 without clickers. In the literature there is conflict results with respect to change on the final marks, from positive change to no change. During the time students were introduced to the clicker system, they were also introduced other learning systems such as the Mind Tap system. Therefore it can not be concluded that that significant difference was sole because of the clickers system. A separate effect of the clicker system on the final exam performance was not done.

Logistic regression

In the logistic regression, explanatory variable year and math mark were both found to be significant in the model. In table 2, Both p-values are $< .0001$, therefore the null hypothesis is rejected. One unit increase in the student's mathematics mark increased the odds of passing (versus failing) by a factor of 1.11, holding term constant. The odds of passing in 2015 were 11% higher than in 2014 when mathematics mark increased by one unit.

Analysis of Maximum Likelihood Estimates						
Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept		1	-5.1772	0.5987	74.7842	<.0001
Term	2014	1	-0.3428	0.0710	23.3208	<.0001
Mathematics		1	0.1053	0.00881	142.8069	<.0001

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
Term 2014 vs 2015	0.504	0.381	0.665
Mathematics	1.111	1.092	1.130

Table 2: The odds table

In table 2 it shows that in 2014 the odds of passing were 0.504 compared to 2015 when clickers were not used, holding math mark constant. In 2014, there is a 50% of students passing the course.

χ^2 test

Table of Term by results

Term Frequency Expected Cell Chi-Square Percent	Fail	Pass	Total
2014	184 142 12.426 7.12	1176 1218 1.4486 45.48	1360 52.59
2015	86 128 13.784 3.33	1140 1098 1.6069 44.08	1226 47.41
Total	270 10.44	2316 89.56	2586 100.00

Statistics for Table of Term by results

Statistic	DF	Value	Prob
Chi-Square	1	29.2650	<.0001

Table 3: The FREQ Procedure (Pass or Fail)

In table 3, $\chi^2 = 29.2650$ with 1 degrees of freedom, p-value $< .0001$. Therefore there is an association between year and results. In 2015, 1098 students were expected to pass but only 1140 students passed and 128 students were expected to fail but only 86 students failed. In 2014, 1218 students were expected to pass but 1176 students passed and only 142 students were expected to fail but 184 students failed.

Table of year by mark1

year	mark1				
Frequency Expected Cell Chi-Square Percent	A	B	C	D	Total
2014	184 142 12.426 7.12	613 592.7 0.6953 23.70	207 230.35 2.3666 8.00	356 394.96 3.8427 13.77	1360 52.59
2015	86 128 13.784 3.33	514 534.3 0.7713 19.88	231 207.65 2.6252 8.93	395 356.04 4.2626 15.27	1226 47.41
Total	270 10.44	1127 43.58	438 16.94	751 29.04	2586 100.00

Statistics for Table of year by mark1

Statistic	DF	Value	Prob
Chi-Square	3	40.7732	<.0001

Table 4: The FREQ Procedure

The final marks were divided into 4 categories A (less than 50%), B (between 50 and 64), C (between 65 and 74) and D (75+). The $\chi^2 = 40.7732$, p-value $< .0001$ with 3 degrees of freedom therefore there is an association between year and categorised final mark. The number of students who got distinctions increased from 38% in 2014 to 43% in 2015. The number of students that obtain less than 50% final mark for STK110 dropped from 12% in 2014 to 5.6% in 2015. In 2015 the overall number of students who failed dropped from 13.05% in 2014 to 7.05% in 2015.

3.1.3 Active involvement

The data from this experiment is not enough to conclude that clickers were the reason for the positive effect on the final performance of student because the instructor adopted also other systems. Also the data suggest that the outcome variable may not be just the final marks. The qualitative data supports active involvement and peer interaction in the classroom. The degree of active involvement and engagement was not measured in this experiment. Literature has referred to clickers as entertainment toys. Previous studies have assessed involvement to items that encourage class attendance, preparing before class, enjoying class and having fun. Rhem (2007) notes that the department needs to get a sense of what constitutes as the best pedagogies to stimulate students involvement in Stk110 comparing that with how often they participate. Given the fact this experiment focused on a large classroom size and structure, the main goal was to increase interaction among peers and instructor. The data from the focus group supported the increased involvement and anonymity encouraged learning. Rhem (2007) suggests that assessing the department and student perceptions of active engagement through surveys done at the beginning and end of the semester, he believes that this will provide a compelling information for the department to draw pedagogical choices.

3.1.4 Shortfalls

A number of limitations existed that may have influenced the student perception of the clicker system and the outcome of the final performance. Firstly, from the focus group data the comments were mainly stating that the students do not want to use the clicker for the tests instead for class engagement and this might have been the main contributor to the negative perception of clickers. Secondly, the clickers were not used for during every class time. Increasing the frequency use of clickers during class times, as well finding different ways

of using the clickers during the semester may have impacted the student's perception of clickers. Thirdly, from the data of the focus group, a substantial number of student admitted to having problems with the technology, such as clickers not being picked up by the TurningPoint® software. The students said that they were frustrated by the malfunctioning of the clicker technology. By the end of the semester the department instructors managed to resolve the majority of the technological problems. The other concern that students raised was the cost they had to incur in order to obtain a clicker, they felt that the clicker was too expensive hence not all could obtain one.

4 Conclusion

There was a significant difference between the two group in exam performance but we can not conclude that this improving was because of the introduction of the clicker technology, because the department had other technologies introduced simultaneously with clicker technology. If the clicker technology did contribute to the improvement of exam performance, the sole impact it had could not be tested. It can be concluded though that the technology intervention employed by the department in Stk110 improve the exam results in year 2015. The students seemed to think that the clickers increased interaction in the classroom compared to their other courses that they are enrolled in with no clickers. The benefit of immediate feedback was also identified by the students as a instrument to understanding difficult concepts. I recommend that the use of clickers be continued. Technology based classrooms are becoming a norm at most intstitutions of higher learning, clicker being the new advanced in the education environment. The department needs to make informative decisions concerning the use of clicker. The students perspective on how clickers could used in class should be taken into consideration. These considerations can include whether clickers should be used for tests, formative or summative or all of them. The preferences of students changes and vary from student to student, the department must be able to change their strategy after every test if the one employed before does not work or give desirable results.

References

[1] R. Anguelov and I. Fabris-Rotelli. LULU operators and discrete pulse transform for multidimensional arrays. *Image Processing, IEEE Transactions on*, 19(3):3012–3023, 2010.

[2] Ian D Beatty, William J Gerace, William J Leonard, and Robert J Dufresne. Designing effective questions for classroom response system teaching. *American Journal of Physics*, 74(1):31–39, 2006.

[3] Derek Bruff. *Teaching with classroom response systems: Creating active learning environments.* John Wiley & Sons, 2009.

[4] Jane E Caldwell. Clickers in the large classroom: Current research and best-practice tips. *CBE-Life Sciences Education*, 6(1):9–20, 2007.

[5] Angel Hoekstra. Vibrant student voices: Exploring effects of the use of clickers in large college courses. *Learning, Media and Technology*, 33(4):329–341, 2008.

[6] Robin H Kay and Ann LeSage. Examining the benefits and challenges of using audience response systems: A review of the literature. *Computers & Education*, 53(3):819–827, 2009.

[7] Agnes Kukulska-Hulme and John Traxler. *Mobile learning: A handbook for educators and trainers.* Psychology Press, 2005.

[8] Margie Martyn. Clickers in the classroom: An active learning approach. *Educause quarterly*, 30(2):71, 2007.

[9] Cheryl Moredich and Ellen Moore. Engaging students through the use of classroom response systems. *Nurse Educator*, 32(3):113–116, 2007.

[10] Keng Siau, Hong Sheng, and Fiona Fui-Hoon Nah. Use of a classroom response system to enhance classroom interactivity. *Education, IEEE Transactions on*, 49(3):398–403, 2006.

[11] April R Trees and Michele H Jackson. The learning environment in clicker classrooms: student processes of learning and involvement in large university-level courses using student response systems. *Learning, Media and Technology*, 32(1):21–40, 2007.

Appendix

```
*************************************************************************************
                                        The Qualitative data SAS Program
*************************************************************************************
PROC IMPORT OUT= SASUSER.STATS
            DATAFILE= "e:\question.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="question$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;
proc chart data=SASUSER.STATS;
*by learn;
vbar learn exciting techissue tests/ midpoints= 0 to 10 by 1 type=percent;
run;
*************************************************************************************

The Quantitative Data-exam performance
*************************************************************************************
PROC IMPORT OUT= SASUSER.marks
            DATAFILE= "e:\marks.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="marks$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;

PROC TTEST;
 CLASS term;
 VAR mark;
 RUN;
/*proc univariate ;
class term;
run;
*************************************************************************************
                                        The Output of TTest
*************************************************************************************
The program the Frequency of groups marks                          24

                                The UNIVARIATE Procedure
                                Variable:  mark  (mark)
                                     Term = 2014

                                      Moments

                N                        1361     Sum Weights              1361
```

Mean	61.8912564	Sum Observations	84234
Std Deviation	15.9964588	Variance	255.886696
Skewness	−0.0315535	Kurtosis	−0.4339345
Uncorrected SS	5561354	Corrected SS	348005.906
Coeff Variation	25.8460722	Std Error Mean	0.43360551

Basic Statistical Measures

Location		Variability	
Mean	61.89126	Std Deviation	15.99646
Median	61.00000	Variance	255.88670
Mode	50.00000	Range	81.00000
		Interquartile Range	25.00000

Tests for Location: Mu0=0

Test	−Statistic−		———p Value———	
Student's t	t	142.7363	Pr > |t|	<.0001
Sign	M	680.5	Pr >= |M|	<.0001
Signed Rank	S	463420.5	Pr >= |S|	<.0001

Quantiles (Definition 5)

Quantile	Estimate
100% Max	99
99%	95
95%	89
90%	83
75% Q3	75
50% Median	61
25% Q1	50
10%	38
5%	35
1%	27
0% Min	18

The program the Frequency of groups marks

25

The UNIVARIATE Procedure
Variable: mark (mark)
Term = 2014

Extreme Observations

———Lowest———		———Highest———	
Value	Obs	Value	Obs

18	2209	98	1585
22	2379	98	1771
22	2095	98	2236
22	2061	99	2203
22	2032	99	2358

The program the Frequency of groups marks

26

The UNIVARIATE Procedure
Variable: mark (mark)
Term = 2015

Moments

N	1226	Sum Weights	1226
Mean	65.2601958	Sum Observations	80009
Std Deviation	15.4011396	Variance	237.1951
Skewness	−0.1475548	Kurtosis	−0.2179959
Uncorrected SS	5511967	Corrected SS	290563.998
Coeff Variation	23.5995914	Std Error Mean	0.43985306

Basic Statistical Measures

Location		Variability	
Mean	65.26020	Std Deviation	15.40114
Median	65.00000	Variance	237.19510
Mode	50.00000	Range	87.00000
		Interquartile Range	22.00000

Tests for Location: Mu0=0

Test	−Statistic−		———p Value———	
Student's t	t	148.3682	Pr > |t|	<.0001
Sign	M	613	Pr >= |M|	<.0001
Signed Rank	S	376075.5	Pr >= |S|	<.0001

Quantiles (Definition 5)

Quantile	Estimate
100% Max	99
99%	97
95%	91
90%	86
75% Q3	76
50% Median	65
25% Q1	54

10%	50
5%	37
1%	28
0% Min	12

The program the Frequency of groups marks

27

The UNIVARIATE Procedure
Variable: mark (mark)
Term = 2015

Extreme Observations

——Lowest——		——Highest——	
Value	Obs	Value	Obs
12	950	98	15
22	185	98	384
24	836	98	424
24	216	99	265
25	972	99	845

The program the Frequency of groups marks

29

The TTEST Procedure

Variable: mark (mark)

Term	N	Mean	Std Dev	Std Err	Minimum	Maximum
2014	1361	61.8913	15.9965	0.4336	18.0000	99.0000
2015	1226	65.2602	15.4011	0.4399	12.0000	99.0000
Diff (1−2)		−3.3689	15.7172	0.6189		

Term	Method	Mean	95% CL Mean		Std Dev	95% CL Std Dev	
2014		61.8913	61.0406	62.7419	15.9965	15.4173	16.6212
2015		65.2602	64.3972	66.1231	15.4011	14.8148	16.0362
Diff (1−2)	Pooled	−3.3689	−4.5825	−2.1554	15.7172	15.3002	16.1576
Diff (1−2)	Satterthwaite	−3.3689	−4.5801	−2.1578			

Method	Variances	DF	t Value	Pr > |t|
Pooled	Equal	2585	−5.44	<.0001

```
                    Satterthwaite     Unequal       2573.6       -5.45       <.0001

                                        Equality of Variances

                                Method       Num DF     Den DF     F Value     Pr > F

                                Folded F       1360       1225        1.08     0.1740
********************************************************************************************************
                                        Chi-squared test program
********************************************************************************************************
proc format;

   value RspFmt 1='Pass'
                0='Fail';

title1 'The program the Frequency of groups marks';
PROC IMPORT OUT= SASUSER.marks
            DATAFILE= "e:\marks.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="marks$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;RUN;
data b;
set SASUSER.marks;

run;

proc freq data=a;
format results RspFmt. ;

table term*results /cellchi2 chisq expected norow nocol  ;
run;

********************************************************************************************************
                                        chi-squared output
********************************************************************************************************
                                            The SAS System           12

                                              The FREQ Procedure

                                           Table of Term by results

                                Term              results

                                Frequency
                                Expected
                                Cell Chi-Square
                                Percent           Fail     Pass      Total

                                             2014      184     1176     1360
```

		142	1218	
		12.426	1.4486	
		7.12	45.48	52.59
	2015	86	1140	1226
		128	1098	
		13.784	1.6069	
		3.33	44.08	47.41
Total		270	2316	2586
		10.44	89.56	100.00

Statistics for Table of Term by results

Statistic	DF	Value	Prob
Chi-Square	1	29.2650	<.0001
Likelihood Ratio Chi-Square	1	30.0087	<.0001
Continuity Adj. Chi-Square	1	28.5724	<.0001
Mantel-Haenszel Chi-Square	1	29.2537	<.0001
Phi Coefficient		0.1064	
Contingency Coefficient		0.1058	
Cramer's V		0.1064	

Fisher's Exact Test

Cell (1,1) Frequency (F)	184
Left-sided Pr <= F	1.0000
Right-sided Pr >= F	<.0001
Table Probability (P)	<.0001
Two-sided Pr <= P	<.0001

Sample Size = 2586

```
*******************************************************************************
                          Grouped Data-frequenct Tables
*******************************************************************************
OPTION NODATE;
title1 'The program the Frequency of groups marks';
PROC IMPORT OUT= SASUSER.marks
            DATAFILE= "e:\marks.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="marks$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;RUN;
data b;
set SASUSER.marks;
```

```
                *DATA SASUSER.marks;
if mark<50 then mark1="LES THAN 50";
if 50<=MARK>=64 then mark1="BTWEEN 50 AND 64";
if mark>65 and mark<74 then mark1="BTWEEN 65 AND 74";
if mark>75 then mark1="DISTINCTION";
run;

proc freq data=b;
tABLES TERM*MARK1/cellchi2 chisq expected norow nocol;
RUN;
********************************************************************************
                                        The Output chi-squares test
********************************************************************************
                                    The SAS System          11

                                    The FREQ Procedure

                                  Table of year by mark1

year            mark1

Frequency
Expected
Cell Chi-Square
Percent         A       B       C       D         Total

          2014     184     613     207     356     1360
                   142   592.7  230.35  394.96
                12.426  0.6953  2.3666  3.8427
                  7.12   23.70    8.00   13.77    52.59

          2015      86     514     231     395     1226
                   128   534.3  207.65  356.04
                13.784  0.7713  2.6252  4.2626
                  3.33   19.88    8.93   15.27    47.41

Total              270    1127     438     751     2586
                 10.44   43.58   16.94   29.04   100.00


                  Statistics for Table of year by mark1

    Statistic                       DF       Value      Prob

    Chi-Square                       3     40.7732    <.0001
    Likelihood Ratio Chi-Square      3     41.4988    <.0001
    Mantel-Haenszel Chi-Square       1     31.2475    <.0001
    Phi Coefficient                        0.1256
    Contingency Coefficient                0.1246
    Cramer's V                             0.1256

                          Sample Size = 2586
```

```
*******************************************************************************************
                                                    Logostic regression SAS program
*******************************************************************************************
PROC IMPORT OUT= SASUSER.STATS
            DATAFILE= "E:\stats.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="stats$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;
proc logistic data=SASUSER.STATS DESCENDING;
 class  term ;
  model results  =term mathematics  / lackfit;
run ;
*******************************************************************************************
                                    Logistic Procedure Outputs
*******************************************************************************************
                              The program the Frequency of groups marks
10

                                        The LOGISTIC Procedure

                                          Model Information

                        Data Set                          SASUSER.STATS
                        Response Variable                 results                 results
                        Number of Response Levels         2
                        Model                             binary logit
                        Optimization Technique            Fisher's scoring


                                 Number of Observations Read        2587
                                 Number of Observations Used        2586


                                        Response Profile

                                 Ordered                          Total
                                   Value      results      Frequency

                                       1            1           2315
                                       2            0            271

                                    Probability modeled is results=1.

NOTE: 1 observation was deleted due to missing values for the response or explanatory varia


                                        Class Level Information
```

Class	Value	Design Variables
Term	2014	1
	2015	−1

Model Convergence Status

Convergence criterion (GCONV=1E−8) satisfied.

Model Fit Statistics

Criterion	Intercept Only	Intercept and Covariates
AIC	1737.169	1531.778
SC	1743.027	1549.352
−2 Log L	1735.169	1525.778

The program the Frequency of groups marks

11

The LOGISTIC Procedure

Testing Global Null Hypothesis: BETA=0

Test	Chi−Square	DF	Pr > ChiSq
Likelihood Ratio	209.3906	2	<.0001
Score	189.3936	2	<.0001
Wald	165.9567	2	<.0001

Type 3 Analysis of Effects

Effect	DF	Wald Chi−Square	Pr > ChiSq
Term	1	23.3208	<.0001
Mathematics	1	142.8069	<.0001

Analysis of Maximum Likelihood Estimates

Parameter		DF	Estimate	Standard Error	Wald Chi−Square	Pr > ChiSq
Intercept		1	−5.1772	0.5987	74.7842	<.0001
Term	2014	1	−0.3428	0.0710	23.3208	<.0001

	Mathematics	1	0.1053	0.00881	142.8069
<.0001					

Odds Ratio Estimates

Effect		Point Estimate	95% Wald Confidence Limits	
Term	2014 vs 2015	0.504	0.381	0.665
Mathematics		1.111	1.092	1.130

Association of Predicted Probabilities and Observed Responses

Percent Concordant	74.7	Somers' D	0.510
Percent Discordant	23.8	Gamma	0.518
Percent Tied	1.5	Tau-a	0.096
Pairs	627365	c	0.755

The program the Frequency of groups marks

12

The LOGISTIC Procedure

Partition for the Hosmer and Lemeshow Test

		results = 1		results = 0	
Group	Total	Observed	Expected	Observed	Expected
1	261	180	180.77	81	80.23
2	254	203	203.15	51	50.85
3	249	209	211.51	40	37.49
4	287	258	253.96	29	33.04
5	248	229	226.34	19	21.66
6	274	254	256.08	20	17.92
7	244	232	232.15	12	11.85
8	251	240	242.19	11	8.81
9	267	262	260.89	5	6.11
10	251	248	247.97	3	3.03

Hosmer and Lemeshow Goodness-of-Fit Test

Chi-Square	DF	Pr > ChiSq
2.1593	8	0.9757

Using linear programming for allocation problems in SAS

Mbongeni M Mzila 11132664

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisors: Dr I Fabris-Rotelli, Dr P.J Van Staden and Ms M Venter

Department of Statistics, Univerisity of Pretoria

02 November 2015 (Final)

Abstract

Different types of industries are faced with the problem of properly allocating input resources to processes in order to get an optimal output that maximises or minimizes the expected return. SAS (Statistical Analysis System) provides different kinds of tools that help in solving this problem. The aim of this research is to provide a knowledge on how the Linear Programming (LP) Procedure in SAS solves this problem using the simplex method. A theoretical background to linear programming will be provided and the Linear Programming (LP) Procedure applied in an Allocation problem done in order to understand the theory behind the procudure.

Declaration

I, *Mbongeni M Mzila,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Mbongeni Mzila

_____
Dr I Fabris-Rotelli *Dr P van Staden* *Ms M Venter*

_____
Date

Contents

List of Figures

List of Tables

1 Introduction

Linear programming is a division of optimization theory which deals with problems of minimization or maximization of linear functions on sets defined by systems of linear equalities and/or inequalities. Linear programming originated in the 30-40s of the twentieth century under the influence of technical and economic problems. Thanks to the works of J. von Neumann, L.V Kantorovich, G. Dantzig, and many other well known mathematicians linear programming became an independent branch of mathematics and continues its development today.[19] The main aim of optimization is to minimize or maximize a specific function with constrained variables.

Optimization problems are divided into various types depending on the sets of values that the variables are restricted to (real, integer or binary or a combination) and the nature of the functional form of the constraints and objectives (linear, quadratic or nonlinear). An algorithm determines the optimal values for the decision variables so that the objective is either maximized or minimized. The optimal values that are assigned to decision variables are on or between allowable bounds and the constraints are obeyed.

When the constraints in an optimization problem are linear and the objective is either linear or quadratic, the optimisation problem can be encapsulated in SAS© data sets and then solved using an appropriate SAS/OR (Operations Research) procedure, in this research namely LP Procedure. The LP Procedure solves linear programming problems that are submitted in a SAS© data set that uses a mathematical programming system (MPS) format.

1.1 Literature Review

Many literature publications discuss about what linear programming is and how to apply it to different situations. Taha [18] describes the differences in different types of programming namely linear programming and integer programming (where variables take on integer variables). In [16], the following types of programming are discussed: dynamic programming where the main model is split into more tractable sub problems, network programming where the model is designed as a network and nonlinear programming where the model has nonlinear functions.

The following is a list of some of the application areas in which optimization-based decision support systems have been used: [2]

- Product-mix problems : finding a combination of products that generate the greatest return when a number of products contend for limited resources. [21]
- Blending problems : allocating a mix of components for a product to minimize costs while achieving minimum standards. [12, 20]
- Time-staged problems : models structured such that they iterate as a function of time. Typical examples include production and inventory models. In each period, a mathematical formula is as follows: production *add* inventory *less* current demand *equaling* to inventor brought over to the next period. [12, 6]
- Scheduling problems : assigning of people to tasks, places, or times so as to optimize the performance and preferences of people while making sure the demands of the schedule are met [12]. In sport it involves assigning teams to play each other and venues to the different games [14].
- Multiple objective problems : objectives are solved sequentially according to the priority order due to multiple conflicting objectives. [17, 5, 10]
- Capital budgeting and project selection problems : finding the set of projects or project that produce maximum return. [13]
- Location problems : seeking appropriate areas that meet distribution needs at minimum cost. [15, 3]

- Cutting stock problems : finding the portion of raw material that minimizes waste and fulfills demand. [8, 9]

Other applications have developed since the the increase of usage of computers and technology including the use of computers to monitor and improve the physical strength of athletes through analysis of previous records [4]. The major challenge in sport is mainly of finding dates and venues in which various games will be played. Examples of sport where application of scheduling of tournaments is involved include: baseball, football, hockey, cricket and basketball. These scheduling problems have been resolved by various approaches (including approximate and exact approaches) namely integer, hybrid, metaheuristics and constraint programming methods. [14]

In most literature an Excel Solver as well as AMPL(an algebraic modelling language also known as A Mathematical Programming Language) are used to solve the allocation problem but in this report the use of LP Procedure will be explored.

2 Background Theory

2.1 Basic components and Properties of the Linear Programming model

In an LP model, there are three basic elements that are crucial in the definition of the model namely:

1. Decision variables : $x_1, x_2, ..., x_n$.
2. Objective function : the goal equation seeking to optimize, a function for example $f(x_1, x_2, ..., x_n) = \sum_{j=1}^{n} c_j x_j$ for some constants c_j, $j = 1, 2, ..., n$.
3. Constraints : rules that the solution must meet in order for it to be valid for example $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ for each i and some constants a_{ij} and b_j

The above thus help in obtaining the optimum solution by breaking the problem in such a manner that is managable to quantify.

Since the model is linear, it must satisfy the following three properties:

1. Proportionality : each decision variable should be directly proportional to the value of the variable in the objective function and the constraints.
2. Additivity : the direct sum of the individual contributions of each variable should be equal to the total contribution of all variables in the objective function and the constraints.
3. Certainty : the coefficients of the objective function and the constraints are either known or specified by probabilistic distributions.

General models can be formally written as follows:

a) Parameters

The following is a list of the parameters used in the model :
n : number of input variables, indexed by $j = 1, ..., n$
m : number of constraints, indexed by $i = 1, ..., m$
c_j : coefficients in the objective function.
b_j : righthand limit of the constraint equations.
a_{ij} : coefficients in the constraints.

b) Variables

x_j : decision/input variables where $j = 1, ..., n$

c) Model

Either maximize or minimize $\sum_{j=1}^{n} c_j x_j$

subject to $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ for $i = 1, ..., m$

In order to understand the theory and formulation of the LP Procedure we start with a two variable example taken from the book by Taha [18] and then a general formulation of a n-variable case with an example.

2.2 A Two Variable Example

It must be stated that in real life situations a two-variable scenario is highly improbable hence this example will used as a starting point for the building up of the theory. Below is an example of a company that produces two types of paint (interior and exterior) using two raw materials named R1 and R2.

The problem

The Table 1 gives the data of the amounts produced and the profit made from each type of paint.

	Tons of raw material needed per ton of paint		
	Exterior paint	Interior paint	Maximum daily availability (tons)
Raw material R1	6	4	24
Raw material R2	1	2	6
Profit per ton (R1000)	5	4	

Table 1: Basic data of the two variable example

A survey made shows that the interior paint's daily demand cannot exceed that for exterior paint by more than one and that the maximum daily demand for interior paint is two tons.

2.2.1 Definition of variables

Since we are dealing with a two-variable case we define two variables x_1 and x_2 namely:

x_1: daily of exterior paint produced in tons
x_2: daily of interior paint produced in tons
$f(x_1, x_2)$: total daily profit (in R'000)

2.2.2 Constraints

- Usage of raw material R1 by both paints :

$$(6x_1 + 4x_2) \text{ tons/day}$$

- Usage of raw material R2 by both paints :

$$(\ x_1 + 2x_2) \text{ tons/day}$$

- Usage of a raw material by both paints must be *lessthanorequalto* the maximum raw material available i.e.

$$6x_1 + 4x_2 \leq 24$$
$$x_1 + 2x_2 \leq 6$$

- Interior paint produced cannot exceed that for exterior paint produced by more than one :

$$x_2 - x_1 \leq 1$$

- Maximum daily demand for interior paint is two :

$$x_2 \leq 2$$

- The total tons must be non-negative :

$$x_1,\, x_2 \geq 0$$

2.2.3 The Objective Function

The objective of this company is to determine the best product mix of exterior and interior paints that maximizes the total daily profit. Written in mathematical form, we want to maximize:

$$f(x_1, x_2) = 5x_1 + 4x_2$$

2.2.4 Summary of problem

We want to maximize

$$f(x_1, x_2) = 5x_1 + 4x_2$$

subject to :

$$6x_1 + 4x_2 \leq 24$$
$$x_1 + 2x_2 \leq 6$$
$$x_2 - x_1 \leq 1$$
$$x_2 \leq 2$$
$$x_1, x_2 \geq 0$$

The next step is to use a software package to find the feasible solution, as we will show in the application section.

2.3 A n-Variable example (Resource Allocation Problem)

The following example taken from [11] presents a real life situation with many variables which help to simplify the understanding of the general formulation. We will look at a profit maximization problem that involves allocating scarce resources among a few machines.

The problem

In this particular example the scarce resource is the time each machine is available for production. The requirement for each machine (in hours per unit) are shown in Table 2 for each product produced. The letter M represents the Machine of concern. The problem is to determine the weekly production with the aim of maximizing profits.

Machine	Product 1	Product 2	Product 3	Product 4	Product 5	Number of machines
M1	1.2	1.3	0.7	0.0	0.5	4
M2	0.7	2.2	1.6	0.5	1.0	5
M3	0.9	0.7	1.3	1.0	0.8	3
M4	1.4	2.8	0.5	1.2	0.6	7
Unit Profit	18	25	10	12	15	

Table 2: Machine data and processing requirements

2.3.1 Definition of variables

We define the following variables:
x_j : quantity of product j produced $j = 1, 2, ..., 5$
$f(x_1, ..., x_5)$: weekly profit obtained

2.3.2 Constraints

The available number of hours on each machine is 40 times the number of machines and hence the total hours available for the 4 machines is 160 hours. The constraints on each machine is given by:

$$\begin{aligned}
&\text{M1}: 1.2x_1 + 1.3x_2 + 0.7x_3 + 0.0x_4 + 0.5x_5 \leq 160 \\
&\text{M2}: 0.7x_1 + 2.2x_2 + 1.6x_3 + 0.5x_4 + 1.0x_5 \leq 160 \\
&\text{M3}: 0.9x_1 + 0.7x_2 + 1.3x_3 + 1.0x_4 + 0.8x_5 \leq 160 \\
&\text{M4}: 1.4x_1 + 2.8x_2 + 0.5x_3 + 1.2x4 + 0.6x_5 \leq 160 \\
&\text{where } x_j \geq 0 \text{ for } j = 1, ..., 5
\end{aligned}$$

2.3.3 The Objective Function

The profit maximization criterion is given by:

$$f(x_1, ..., x_5) = 18x_1 + 25x_2 + 10x_3 + 12x_4 + 15x_5$$

2.3.4 Summary of problem

We want to maximize

$$f(x_1, ..., x_5) = 18x_1 + 25x_2 + 10x_3 + 12x_4 + 15x_5$$

subject to :

$$\begin{aligned}
&1.2x_1 + 1.3x_2 + 0.7x_3 + 0.0x_4 + 0.5x_5 \leq 160 \\
&0.7x_1 + 2.2x_2 + 1.6x_3 + 0.5x_4 + 1.0x_5 \leq 160 \\
&0.9x_1 + 0.7x_2 + 1.3x_3 + 1.0x_4 + 0.8x_5 \leq 160 \\
&1.4x_1 + 2.8x_2 + 0.5x_3 + 1.2x4 + 0.6x_5 \leq 160 \\
&x_j \geq 0 \text{ for } j = 1, ..., 5
\end{aligned}$$

Once again, the software package will be used to find the feasible solution as shown in the Proc LP section.

2.4 Methods of optimization

An optimization algorithm is used to determine the solution to a given problem whereby the algorithm is executed repeatedly by comparing various solutions until an optimum solution is found. There are two types of algorithms namely:

1. Deterministic algorithms : which uses specific rules for moving from one solution to another.
2. Stochastic algorithms : with probabilistic translation rules for moving from one solution to another.

We limit our scope of research to linear programming in SAS/OR software which provides a wide range of procedures that can be used to solve various types of problems. Data describing the models are supplied in a form appropriate for the particular type of problem, and a specific type of optimization algorithms is used to make use of the special characteristics and structures of the problems that they solve. We will focus on proc LP and the simplex method for solving linear programming problems.

The LP procedure, with the help of interactive features, provides various options and solution strategies that enable the user to produce various kinds of intermediate and final solution. Iterations can be stoped at any intermediate stage to view current results and if required, one can change the options or strategies used then execution of procedure is resumed. The procedure makes use of a two-phase revised simplex

method, which employs the Bartels-Golub update of the LU (Lower-Upper) decomposed basis matrix(basis is decomposed into upper (U) and lower (L) triangular factors) to pivot between feasible solutions[1].

PROC LP goes through two phases in sloving an LP problem. In phase 1, it finds a basic feasible solution using the Bartels-Golub update method while in phase 2 it finds an optimal solution using the simplex method. The procedure handles unrestricted variables, lower-bounded variables, upper-bounded variables and ranges on constraints and when no explicit lower bounds are specified, the procedure assumes that the variables are bounded below by zero.

Integer programs are solved sequencially by the branch-and-bound technique. This sequence can be shown using a tree. Each node of the tree is identified with a linear program derived from the problem, on the path leading to the root of the tree. A problem with an active node is chosen then attempts to solve it using the dual simplex algorithm. If a problem is infeasible, it is dropped but if it can be solved and does not have an integer solution (i.e. a solution were all variables are considered as integers), it then defines two new problems. The new problems each contain the parent problems, all of the constraints including the appropriate additional one. Branching continues until either there are no active nodes or an integer solution is found. [7]

2.4.1 The Simplex Method

In the development of the simplex method, two requirements are imposed on the constraints namely:

1. Constraints are equations with nonnegative right-hand side (except for nonnegativity variables).
2. All variables are nonnegative.

   These requirements are put in order to standardize the simplex method calculations but various software packages do accept nonnegative right-hand sides, inequality constraints and unrestricted variables. There are two approaches used to explain and understand the iterative nature of the simplex method. These include explanation using the graph method or the use of algebra to find the optimal solution. We use an example taken and modified from [18] to illustrate these two approaches.

Suppose we want to maximize the objective function given by

$$f(x_1, x_2) = 2x_1 + 3x_2$$

subject to:

$$2x_1 + x_2 \leq 4$$
$$x_1 + 2x_2 \leq 5$$
$$x_1, x_2 \geq 0$$

The solution space is then given by the following graph extracted and edited from[18]:

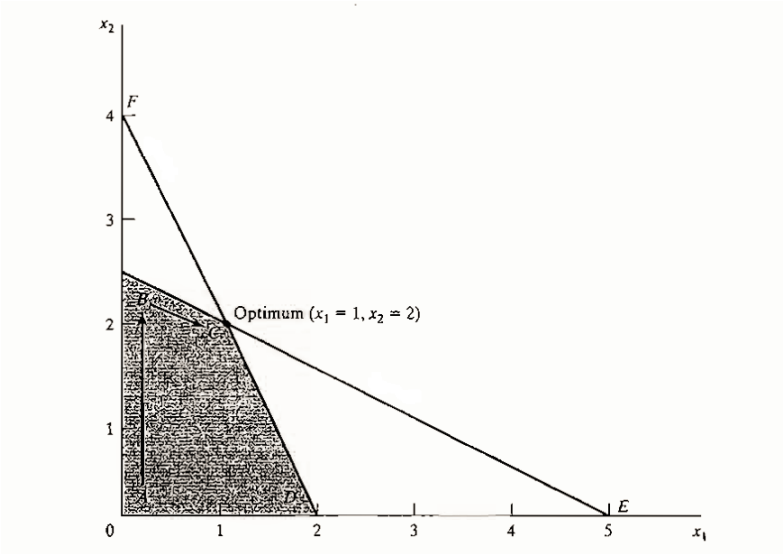


Figure 1: Solution space of the example.

The method starts the investigation of the optimal solution at the origin where $x_1 = x_2 = 0$. The design is such that the variable with the largest rate of improvement in $f(x_1, x_2)$ is increased first followed by the other variable. In this example the value of $f(x_1, x_2)$ will increase first by two for each unit increase in x_1 followed by three for each unit increase in x_2, so that the rate of improvement in the value of $f(x_1, x_2)$ is 2 for x_1 and 3 for x_2. We then choose to increase x_2 first and move from point A ($x_1 = x_2 = 0$) to point B as shown in the graph above. The simplex method will then increase the value of $x_1$at point B to reach the corner at point C, the optimum solution. The path of the simplex algorithm is therefore defined as from A to B then to C with each corner along the path is associated with an iteration. The simplex method moves alongside the edges of the solution space and cannot move directly from A to C cutting across the solution space.

The algebraic approach requires defining variables s_1 and s_2 that transform the inequalities of the objective function into equalities. The points A, B and C in Figure 1 are therefore represented by their basic and non-basic variables as shown in Table 3 below:

Corner Point	Basic variables	Nonbasic (zero) variables
A	s_1, s_2	x_1, x_2
B	s_1, x_2	x_1, s_2
C	x_1, x_2	s_1, s_2

Table 3: Algebraic solution table

From A to B, non-basic x_2 at A becomes basic at B and basic s_2 at A becomes non-basic at B. x_2 is the entering variable(because it enters the basic solution while s_2 is the leaving variable because it leaves the basic solution. In a similar way, at point B, x_1 enters and s_1 leaves, hence leading to the point C.

2.4.2 Computational details of the simplex method

We show through an example how the simplex iteration works. The rules for determining the entering and leaving variables for stopping the computations when the optimum solution has been reached is discussed. We refer back to our two-variable example (Section 2.2) of the company that produces paint (extracted and modified from [18]). Recall that we want to maximize $f(x_1, x_2) = 5x_1 + 4x_2$ subject to:

$$6x_1 + 4x_2 \leq 24$$
$$x_1 + 2x_2 \leq 6$$
$$x_2 - x_1 \leq 1$$
$$x_2 \leq 2$$
$$x_1, x_2 \geq 0$$

We define variables $s_1, ...,s_4$ (known as slack variables) that are used to transform the inequalities of the objective function into equalities and then the above can be re-written as follows:

Maximize:

$$f(x_1, x_2, s_1, ..., s_4) = 5x_1 + 4x_2 + 0s_1 + 0s_2 + 0s_3 + 0s_4$$

subject to :

$$6x_1 + 4x_2 + s_1 = 24$$
$$x_1 + 2x_2 + s_2 = 6$$
$$x_2 - x_1 + s_3 = 1$$
$$x_2 + s_4 = 2$$

$x_1, x_2, s_1, s_2, s_3, s_4 \geq 0$ with the variables $s_1, ..., s_4$ taken as the slack or inactive variables associated with the respective constraints. The starting simplex table can be summarised as follows in Table 4:

Basic	$f(.)$	x_1	x_2	s_1	s_2	s_3	s_4	Solution [1]	
$f(.)$	1	-5	-4	0	0	0	0	0	$f(.) - row$
s_1	0	6	4	1	0	0	0	24	$s_1 - row$
s_2	0	1	2	0	1	0	0	6	$s_2 - row$
s_3	0	-1	1	0	0	1	0	1	$s_3 - row$
s_4	0	0	1	0	0	0	1	2	$s_4 - row$

Table 4: Starting simplex table

The design of the table specifies the set of basic and non-basic variables as well as provides the solution associated with the starting iteration. As we have shown earlier, the simplex iterations start at the origin, i.e. $(x_1, x_2) = (0, 0)$, with the associated set of nonbasic and basic variables defined as:

- Nonbasic(zero) variables: (x_1, x_2)
- Basic variables: (s_1, s_2, s_3, s_4)

Substituting the nonbasic variables $(x_1, x_2) = (0, 0)$, noting the special 0-1 arrangement of the coefficients of $f(.)$ and the basic variables (s_1, s_2, s_3, s_4) in the table, we obtain the following (that is, a corner point solution when $(x_1, x_2) = (0, 0)$):

$$f(.) = 0$$
$$s_1 = 24$$
$$s_2 = 6$$
$$s_3 = 1$$
$$s_4 = 2$$

This information is shown in Table 4 by listing the basic variables in the left most 'Basic' column and their values (initial values) in the right most Solution column. The table therefore defines the current corner by specifying its basic variables, their values as well as the corresponding value of the objective function, $f(.)$ and the nonbasic variables (those not listed in the Basic column) are always equal to zero. The objective

[1] Coefficients of the right hand side of the constraints.

function $f(.) = 5x_1 + 4x_2$ shows that the solution can be improved by increasing x_1 or x_2. Using the argument presented in Section 2.4.1, x_1 is selected as the entering variable since it has the most positive coefficient. Similarly, since the simplex table expresses the objective function as $f(.) = -5x_1 - 4x_2 = 0$, the entering variable will correspond to the variable with the most negative coefficient in the objective equation. This is referred to as the Optimality Condition.

In order to find the the leaving variable from the simplex table, determine the minimum non-negative ratios of the right-hand side of the equations (Solution column) to the corresponding constraint coefficients under the entering variable, x_1, as shown in table 5 below.

Basic	Entering x_1	Solution	Ratio (intercept)
s_1	6	24	$x_1 = \frac{24}{6} = 4$ (minimum)
s_2	1	6	$x_1 = \frac{6}{1} = 6$
s_3	-1	1	$x_1 = \frac{1}{-1} = -1$ (ignore)
s_4	0	2	$x_1 = \frac{2}{0} = \infty$ (ignore)
Conclusion: x_1 enters and $s_1$leaves			

Table 5: Ratios for determining entering and leaving variables

In summary, we conclude that x_1 is the entering variable because it has the most positive coeffient in the objective function and s_1 is the leaving variable because it has the minimum non-negative ratio or intercept as shown in Table 5 above.

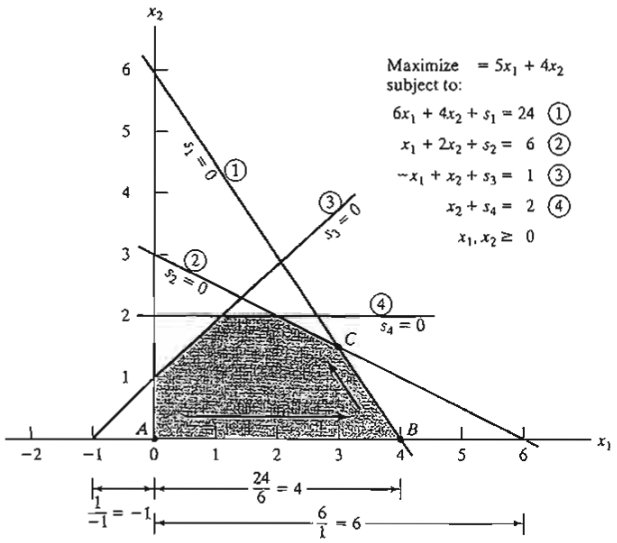


Figure 2: Graphical interpretation of the simplex method ratios

The minimum nonnegative ratio automatically identifies the current basic variable s_1 as the leaving variable and assigns the entering variable x_1 the new value of 4. The graph in Figure 2 shows that the computed ratios are the intercepts of the constraints with the entering variable (x_1) axis. It can be seen that the value of x_1 must be increased to 4 at point B, which is the smallest nonnegative intercept with the x_1-axis. An increase beyond B to $x_1 = 6$ is infeasible. At this point, the current basic variable s_1 associated with constraint 1 assumes a zero value and becomes the leaving variable. This is refered to as the feasibility condition because it guarantees the feasibility of the new solution. The new solution point B is determined by exchanging the entering variable x_1 and the leaving variable s_1 in the simplex table to produce the following sets of nonbasic and basic variables:

- Nonbasic (zero) variables at B: (s_1, x_2)
- Basic variables at B: (x_1, s_2, s_3, s_4)

The exchanging process is based on the Gauss-Jordan row operations which identifies the entering variable column as the Pivot Column and the leaving variable row as the Pivot Row with the intersection of the pivot column and the pivot row is called the Pivot Element. In Table 4 above, the pivot column is then the one labelled x_1 and the pivot row is labelled s_1.

The Gauss-Jordan computations needed to produce the new basic solution include two types.

1. Pivot row
   a. Replace the leaving variable in the basic column with the entering variable.
   b. New pivot row equal to Current pivot row divided by pivot element.
2. All other rows, including $f(.)$

   New Row = (Current row) − (Its pivot column coefficient) × (New pivot row)

These computations are applied to the rest of Table 4 as follows:

1. Replace s_1 in the Basic column with x_1:

   New x_1-row = Current s_1-row $\div 6$
   $= (0\ 6\ 4\ 1\ 0\ 0\ 0\ 24\ )$
   $=(0\ 1\ \frac{2}{3}\ \frac{1}{6}\ 0\ 0\ 0\ 4)$
2. New $f(.)$-row = Current $f(.)$-row $-(-5)\times$ New x_1-row
   $= (1\ -5\ -4\ 0\ 0\ 0\ 0\ 0\ ) - (-5)\times\ (0\ 1\ \frac{2}{3}\ \frac{1}{6}\ 0\ 0\ 0\ 4\ )$
   $= (1\ 0\ -\frac{2}{3}\ \frac{5}{6}\ 0\ 0\ 0\ 20\ )$
3. New s_2-row = Current s_2-row$-(1)\times$ New x_1-row
   $= (0\ 1\ 2\ 0\ 1\ 0\ 0\ 6\ ) - (1) \times (0\ 1\ \frac{2}{3}\ \frac{1}{6}\ 0\ 0\ 0\ 4\ )$
   $= (0\ 0\ \frac{4}{3}\ -\frac{1}{6}\ 1\ 0\ 0\ 2\ )$
4. New s_3-row = Current s_3-row $-(-1)\times$ New x_1-row
   $= (0\ -1\ 1\ 0\ 0\ 1\ 0\ 1\ ) - (-1) \times (0\ 1\ \frac{2}{3}\ \frac{1}{6}\ 0\ 0\ 0\ 4\ )$
   $= (0\ 0\ \frac{5}{3}\ \frac{1}{6}\ 0\ 1\ 0\ 5\ )$
5. New s_4-row = Current s_4-row $-(0)\times$ New x_1-row
   $= (0\ 0\ 1\ 0\ 0\ 0\ 1\ 2\ ) - (0) \times\ (0\ 1\ \frac{2}{3}\ \frac{1}{6}\ 0\ 0\ 0\ 4\ )$
   $= (0\ 0\ 1\ 0\ 0\ 0\ 1\ 2\ )$

The new basic solution is therefore given by $(x_1, s_2, s_3, s_4,)$ and the new table becomes:

				⇓					
	Basic	$f(.)$	x_1	x_2	s_1	s_2	s_3	s_4	Solution
	$f(.)$	1	0	$-\frac{2}{3}$	$\frac{5}{6}$	0	0	0	20
	x_1	0	1	$\frac{2}{3}$	$\frac{1}{6}$	0	0	0	4
⇐	s_2	0	0	$\frac{4}{3}$	$-\frac{1}{6}$	1	0	0	2
	s_3	0	0	$\frac{5}{3}$	$\frac{1}{6}$	0	1	0	5
	s_4	0	0	1	0	0	0	1	2

Table 6: New simplex solution table

The new table has the same properties as the starting table. Setting the new nonbasic variables x_2 and s_2 to zero, the Solution column automatically yields the new basic solution $(x_1 = 4, s_2 = 2, s_3 = 5, s_4 = 2)$. The corresponding new objective value is $f(.) = 20$, which is consistent with:

$$\begin{aligned}\text{New } f(.) &= \text{Old } f(.) + \text{New } x_1\text{-value} \times \text{its objective coefficient} \\ &= 0 + 4 \times 5 = 20\end{aligned}$$

In Table 6 above, the optimality condition shows that x_2 is the entering variable. The feasibility condition produces the following table:

Basic	Entering x_2	Solution	Ratio
x_1	$\frac{2}{3}$	4	$x_2 = 4 \div \frac{2}{3} = 6$
s_2	$\frac{4}{3}$	2	$x_2 = 2 \div \frac{4}{3} = 1.5$ (minimum)
s_3	$\frac{5}{3}$	5	$x_2 = 5 \div \frac{5}{3} = 3$
s_4	1	2	$x_2 = 2 \div 1 = 2$

Table 7: Feasible solution table

Therefore, s_2 leaves the basic solution and new value of x_2 is 1.5. The corresponding increase in $f(.)$ is $\frac{2}{3}x_1 = \frac{2}{3} \times 1.5 = 1$, which yields new $f(.) = 20 + 1 = 21$.

Replacing s_2 in the Basic column with entering x_2, the following Gauss-Jordan row operations are applied:

1. New pivot x_2-row = Current s_2-row $\div \frac{4}{3}$
2. New $f(.)$-row = Current $f(.)$-row $-(-\frac{2}{3})\times$ New x_2-row
3. New x_1-row = Current x_1-row $-(\frac{2}{3})\times$ New x_2-row
4. New s_3-row = Current s_3-row $-(\frac{5}{3})\times$ New x_2-row
5. New s_4-row= Current s_4-row $-(1)\times$ New x_2-row

The computations above produce the following table:

Basic	$f(.)$	x_1	x_2	s_1	s_2	s_3	s_4	Solution
$f(.)$	1	0	0	$\frac{3}{4}$	$\frac{1}{2}$	0	0	21
x_1	0	1	0	$\frac{1}{4}$	$-\frac{1}{2}$	0	0	3
x_2	0	0	1	$-\frac{1}{8}$	$\frac{3}{4}$	0	1	$\frac{3}{2}$
s_3	0	0	0	$\frac{3}{8}$	$-\frac{5}{4}$	1	0	$\frac{5}{2}$
s_4	0	0	0	$\frac{1}{8}$	$-\frac{3}{4}$	0	1	$\frac{1}{2}$

Table 8: Optimal solution of simplex method

Based on the optimality condition, the $f(.)$-row coefficients associated with the nonbasic variables, s_1 and s_2, are non-negative and thus Table 8 is optimal. The optimal values of the variables in the Basic column are given in the right-hand-side Solution column and the optimum solution can be interpreted as follows:

Decision variable	Optimum value	Recommendation
x_1	3	Produce 3 tons of exterior paint daily
x_2	$\frac{3}{2}$	Produce 1.5 tons of interior paint daily
$f(x_1, x_2)$	21	Daily profit is $R21000$

Table 9: Interpretation of the optimal solution

The solution also gives the status of the resources. If the model uses a resource completely then the resource is defined as scarce otherwise the resource is abundant. This information is obtained from the optimum table, Table 8, by checking the value of the slack/inactive variable $(s_1, ..., s_4)$ associated with the constraint representing the resource. A slack value of zero shows that the resource is used completely and is classified as scarce otherwise a positive slack shows that the resource is abundant. Recall that when determining the entering and leaving variables we made s_1 and s_2 to be equal to zero since they were leaving variables. The values of s_3 and s_4 are found by reading from the right-hand-side solution column of Table 8. The following table then summaries the classification of the constraints of the model:

Resource	Slack	Status
Raw Material, R1	$s_1 = 0$	Scarce
Raw material, R2	$s_2 = 0$	Scarce
Market limit	$s_3 = \frac{5}{2}$	Abundant
Demand limit	$s_3 = \frac{1}{2}$	Abundant

Table 10: Classification of resources

From the above we see how the derivation of the optimal solution is tedious for a 2-variable case. We therefore introduce a more simplified way of finding an optimal solution. This is through the use of proc LP in SAS©which solves LPs (using a general simplex algorithm) and mixed integer programming problems.

2.4.3 Other optimization procedures

Although we focus on proc LP, SAS/OR software has other optimization procedures namely [7]:

- PROC NETFLOW - which solves network optimization problems, which are LPs with a dominant network (node and arc) structure. These are specialized simplex methods.
- PROC INTPOINT - an additional procedure focused on the use of the interior-point algorithm in solving linear programs and network optimization problems.
- PROC NLP - which solves nonlinear programming problems, quadratic programming problems and least-squares problems. There are different available techniques which include:
  - Trust-region method
  - Newton-Raphson method with line search
  - Quadratic active set technique
  - Quasi-Newton methods
  - Double-dogleg method
  - Conjugate gradient methods
  - Newton-Raphson method with ridging
  - Nelder-Mead simplex method

- Hybrid quasi-Newton methods
- Levenberg-Marquardt method

- PROC TRANS - which solves transportation problems, were items must be moved from a set of "supply" locations to a set of "demand" locations at lowest cost.

2.4.4 The Simplex algorithm

We summarize the simplex method in a general case and also give a summary of the computational details. The following conditions must hold.

1. Optimality condition: The entering variable in a maximization (minimization) problem is the nonbasic variable having the most negative (positive) coefficient in the $f(x_1, x_2)$-row. The optimum solution is reached at the iteration where all the $f(x_1, x_2)$-row coefficients of the nonbasic variables are nonnegative (nonpositive).
2. Feasibility condition: The leaving variable is the basic variable associated with the smallest nonnegative ratio (with strictly positive denominator).

Gauss-Jordan row operations:

1. Pivot row:

   a : Replace the leaving variable in the Basic column with the entering variable.

   b : New pivot row = Current pivot row/Pivot element

2. All other rows, including $f(x_1, x_2)$ New row = (Current row) − (pivot column coefficient) × (New pivot row)

The steps of the simplex algorithm are:

Step 1: Determine a starting basic feasible solution.

Step 2: Select an entering variable using the optimality condition. Stop if there is no entering variable; the last solution is optimal. Else, go to step 3.

Step 3: Select a leaving variable using the feasibility condition.

Step 4: Determine the new basic solution by using the appropriate Gauss-Jordan computations. Go to step 2.

The simplex table also gives additional information that includes:

- Post-optimal analysis, which deals with finding a new optimal solution when the data of the model are changed.
- Sensitivity analysis, which deals with determining the conditions that will keep the current solution unchanged.

2.5 Proc LP

2.5.1 SAS Code of Two examples and output

A Two Variable Example

We refer back to the two variable example in Section 2.2. As mentioned in the last paragraph of Section 2.2, we are going to use PROC LP as our software package to find the feasible solution. The following is the code for the example:

```
data;
format _row_ $7. ;
input _row_ $ x1 x2 _type_ $ _rhs_;
datalines;
object 5 4 max .
c1 6 4 le 24
c2 1 2 le 6
c3 -1 1 le 1
c4 . 2 UPPERBD .
;
proc lp;
run;
```

The SAS program above is divided into two parts which are the data step and the PROC LP procedure step. The data step is used for storing data in a data set that can be used by the LP procedure to solve the problem. The linear program is built using the DATA step and the model is saved in SAS as sparse input format for PROC LP. The _RHS_, _ROW_, and _TYPE_variables are special variables in PROC LP for the right-hand-side, row variable, variable, and type variable [7]. Table 11 gives a description of the above code in relation to the example given in Section 2.2.

Code line	Explanation	Relation to the Example
data;	This line specifies the name of the data set. In this example the data set name has not been specified therefore proc lp will use the recent formulated data set to carry out the analysis.	data paint;
format _row_ $7. ;	The $ sign specifies that the variable is made of characters. The line then formats the variable to character type variable and also specifies the length or number of characters that the variable i.e. the variable _row_ should have.	
input _row_ $ x1 x2 _type_ $ _rhs_;	Defines the decision variables for the objective function and the right hand side of the constraints. The _row_ variable contains names of constraints and objective function (names of rows), the _type_ variable contains the type of each observation and _rhs_ contains right hand side constants for the constraints.	The decision variables are x_1 and x_2.
datalines;	Specifies the constraints for the objective function and opens a platform for entering data to the data set.	
object 5 4 max .	This is the objective function of the problem. The max specifies that we are maximising the function and the dot shows that it does not have a right hand side value.	$f(x_1, x_2) = 5x_1 + 4x_2$
c1 6 4 le 24	Coeffients for the less than or equal to inequality (le) constrained row.	$6x_1 + 4x_2 \leq 24$
c2 1 2 le 6	Coeffients for the less than or equal to inequality (le) constrained row.	$x_1 + 2x_2 \leq 6$
c3 -1 1 le 1	Coeffients for the less than or equal to inequality (le) constrained row.	$x_2 - x_1 \leq 1$
c4 . 2 UPPERBD .	Identifies the upper bound on the decision variable corresponding to the position in which the numeric value is.	$x_2 \leq 2$
;	Closes the platform for entering data to the data set	
proc lp;	Solves the problem using the LP Procedure.	
run;	Submits the code so that the analysis can be made.	

Table 11: Description of code for Section 2.2

When running the above code in SAS, four summaries are displayed that include the problem summary, the variable summary, the solution summary,and the constraint summary. The outputs of interest are the variable summary and the solution summary (refer to appendix for the complete output and Section 2.5.2 for a complete description of the output).

Table 12 displays the solution summary for the above SAS code which indicates whether or not an optimal solution was found. In this example, the procedure executes successfully (with an optimal solution), with 21 as the value of the objective function. Also included in this section of output is the number of phase 1 (Bartels-Golub update method) and phase 2 iterations (simplex method), number of variables used in the initial basic feasible solution and time used to solve the problem. The other details of the output will be explained in Section 2.5.2.

The LP Procedure

Solution Summary	
Terminated Successfully	
Objective Value	21
Phase 1 Iterations	0
Phase 2 Iterations	2
Phase 3 Iterations	0
Integer Iterations	0
Integer Solutions	0
Initial Basic Feasible Variables	5
Time Used (seconds)	0
Number of Inversions	3
Epsilon	1.00E-08
Infinity	1.797693E308
Maximum Phase 1 Iterations	100
Maximum Phase 2 Iterations	100
Maximum Phase 3 Iterations	99999999
Maximum Integer Iterations	100
Time Limit (seconds)	120

Table 12: Solution Summary for example in Section 2.2

The SAS System

The LP Procedure

Variable Summary

Col	Variable Name	Status	Type	Price	Activity	Reduced Cost
1	x1	BASIC	NON-NEG	5	3	0
2	x2	BASIC	UPPERBD	4	1.5	0
3	c1		SLACK	0	0	-0.75
4	c2		SLACK	0	0	-0.5
5	c3	BASIC	SLACK	0	2.5	0

Table 13: Variable summary output for example in section 2.2

The Variable Summary in Table 13 gives the value of the structural variables at optimality and tells how to produce the two types of paint in order to maximize the total daily profit. The activity column gives the optimal values, which in this example are three units of interior paint and 1.5 units of exterior paint to be produced. The reduced cost associated with each nonbasic variable (variables without the term "Basic" under the status column) is the marginal value of that variable if it is brought into the basis. This

is the same as the description provided in Section 2.4.2 in respect to entering and leaving variables. This means that the objective function value would increase by the reduced cost of a nonbasic variable if the variable's value increases by one or decrease by the reduced cost of a nonbasic variable if that variable's value decreases by one. Basic variables have a zero reduced cost as also mentioned in Section 2.4.2. At optimality, for a maximization problem as is in this example, nonbasic variables that are not at an upper bound have nonpositive reduced costs. Nonbasic variables at upper bounds have nonnegative reduced costs, showing that increasing the upper bound (if the reduced cost is not zero) does not decrease the objective.

A n-Variable example (Resource Allocation problem)

We also refer back to the resourse allocation example in Section 2.3. As mentioned in the last paragraph of Section 2.3, we are going to use PROC LP as our software package to find the feasible solution. The following is the SAS code for the example:

```
data;
format _row_ $7. ;
input _row_ $ p1 p2 p3 p4 p5 _type_ $ _rhs_;
datalines;
object 18 25 10 12 15 max .
m1 1.2 1.3 0.7 0 0.5 le 160
m2 0.7 2.2 1.6 0.5 1 le 200
m3 0.9 0.7 1.3 1 0.8 le 120
m4 1.4 2.8 0.5 1.2 0.6 le 280
;
proc lp;
run;
```

Table 14 gives a description of the above code in relation to the example given in Section 2.3. The explanations are similar to those found in example given above. In Table 14, the decision variables $x_1, ..., x_5$ are the exact same the variables as $p_1, ..., p_5$ that are written in the code.

As in the previous section, the above code produces four summary tables that can be used to analyse and conclude on the problem. We focus on the solution summary and the variable summary as these provide the relevant information for our purpose. For this example, the process terminated successfully with an objective value of 2988.73 with four phase 2 iterations as shown in Table 15 below.

Code line	Explanation	Relation to the Example
data;	This line specifies the name of the data set. In this example the data set name has not been specified therefore so proc lp will use the recent formulated data set to carry out the analysis.	data allocation;
format _row_ $7. ;	The $ sign specifies that the variable is made of characters. The line then formats the variable to character type variable and also specifies the length or number of characters that the variable i.e. the variable _row_ should have.	
input _row_ $ p1 p2 p3 p4 p5 _type_ $ _rhs_;	Defines the decision variables for the objective function and the right hand side of the constraints. The _row_ variable contains names of constraints and objective function (names of rows), the _type_ variable contains the type of each observation and _rhs_ contains right hand side constants for the constraints.	The decision variables are $x_1, ...,$ x_5.
datalines;	Specifies the constraints for the objective function and opens a platform for entering data to the data set.	
object 18 25 10 12 15 max .	This is the objective function of the problem. The max specifies that we are maximising the function and the dot shows that it does not have a right hand side value.	$f(x_1, ..., x_5) =$ $18x_1 + 25x_2 + 10x_3 + 12x_4 + 15x_5$
m1 1.2 1.3 0.7 0 0.5 le 160	Coeffients for the less than or equal to inequality (le) constrained row.	$1.2x_1 + 1.3x_2 + 0.7x_3 + 0.0x_4 +$ $0.5x_5 \leq 160$
m2 0.7 2.2 1.6 0.5 1 le 200	Coeffients for the less than or equal to inequality (le) constrained row.	$0.7x_1 + 2.2x_2 + 1.6x_3 + 0.5x_4 +$ $1.0x_5 \leq 160$
m3 0.9 0.7 1.3 1 0.8 le 120	Coeffients for the less than or equal to inequality (le) constrained row.	$0.9x_1 + 0.7x_2 + 1.3x_3 + 1.0x_4 +$ $0.8x_5 \leq 160$
m4 1.4 2.8 0.5 1.2 0.6 le 280	Coeffients for the less than or equal to inequality (le) constrained row.	$1.4x_1 + 2.8x_2 + 0.5x_3 + 1.2x4 +$ $0.6x_5 \leq 160$
;	Closes the platform for entering data to the data set	
proc lp;	Solves the problem using the LP Procedure.	
run;	Submits the code so that the analysis can be made.	

Table 14: Description of code for Section 2.3

The SAS System

The LP Procedure

Solution Summary	
Terminated Successfully	
Objective Value	2988.73
Phase 1 Iterations	0
Phase 2 Iterations	4
Phase 3 Iterations	0
Integer Iterations	0
Integer Solutions	0
Initial Basic Feasible Variables	6
Time Used (seconds)	0
Number of Inversions	3
Epsilon	1.00E-08
Infinity	1.797693E308
Maximum Phase 1 Iterations	100
Maximum Phase 2 Iterations	100
Maximum Phase 3 Iterations	1E+08
Maximum Integer Iterations	100
Time Limit (seconds)	120

Table 15: Solution summary for example in Section 2.3

Table 16 below shows how much should be allocated to resourses given the available constraints. The reduced cost of P3 indicates that if this variable were increased from 0 to 1 the objective value (or profit) will decrease by 13.53. When a nonbasic variable changes, the basic variables change so that the equations defining the solution remain satisfied. The reduced costs are derivatives that indicate the rate of change. The activity column shows that in order to maximize the profit 59 units of product 1, 63 units of product 2, zero units of product 3, 11 units of product 4 and 15 units of product 5 need to be produced. Jensen et al [11] used the Math Programming add-in of Excel (solved the problem with the Jensen LP add-in) and got the same results as the ones found with PROC LP.

The SAS System

The LP Procedure

Variable Summary

Col	Variable Name	Status	Type	Price	Activity	Reduced Cost
1	p1	BASIC	NON-NEG	18	58.9614	0
2	p2	BASIC	NON-NEG	25	62.6346	0
3	p3		NON-NEG	10	0	-13.53
4	p4	BASIC	NON-NEG	12	10.5763	0
5	p5	BASIC	NON-NEG	15	15.6428	0
6	m1		SLACK	0	0	-4.8195
7	m2		SLACK	0	0	-5.2016
8	m3		SLACK	0	0	-8.9635
9	m4		SLACK	0	0	-0.3631

Table 16: Variable Summary for example in Section 2.3

2.5.2 General PROC LP SAS Code and Output

SAS code

As mentioned above, a data set needs to be created first for it to be used by PROC LP, usually using dense format where the model is written in the way that its formulated. In the DATA step, the variable names are the decision/input variables, the rows are the constraints, and the coefficients are given as the values for the structural variables. The same model can be specified in the sparse format which enables to omit the zero coefficients in the description of the linear program.

The LP procedure has some key words that are reserved for specific purposes. Table 17 explains the function of each key variable and also gives the data format that it can be used in.

Variable name	Description	Data Format
COEF	Variables that contain coefficients	sparse
COL	Variable that contains column names	sparse
ROW	Variable that contains names of constraints and objective functions (names of rows) for the dense format	sparse
_ID_	Alternative for the ROW statement	
RANGE	Variable (column) that contains the range constant for the dense format for range analysis	sparse
RHS	Variables (columns) that contains right hand side constants for the dense format	sparse
RHSSEN	Variables (columns) that define right hand side change vectors for the dense format for sensitivity analysis	sparse
TYPE	Variable that contains the type of each observation	
VAR	Structural (decision) variables	dense

Table 17: Variable Type List [7]

For the dense format, a model's row names appear as character values in a SAS data set. For the sparse format, both the row and the column names of the model appear as character values in the data set. When

referring to these names in the problem definition statement or other LP statements, you must use single or double quotes around them.

SAS Output

When the SAS code is run, i.e. submited, there are four summary tables that are produced. Tables 18 through to 21 describe what each of the four summary tables entail (taken from SAS help [7]):

Problem Summary	
Item	**Description**
Type of optimization and the name of the objective row (as identified by the ID or ROW variable)	Max OBJ
Name of the SAS variable that contains the right-hand-side constants	_RHS_
Name of the SAS variable that contains the type keywords	_TYPE_
Density of the coefficient matrix after the slack and surplus variables have been appended	Density - The ratio of the number of nonzero elements to the number of total elements
Number of each type of variable in the mathematical program	
Number of each type of constraint in the mathematical program	

Table 18: Problem Summary Output

The Solution Summary	
Item	**Description**
Termination status of the procedure	Indicates if the procedure terminated succesfully or not
Objective value of the current solution	The optimal value of the objective function
Number of phase 1 iterations that were completed	Number of Bartels-Golub update method iterations
Number of phase 2 iterations that were completed	Number of simplex method iterations
Number of phase 3 iterations that were completed	
Number of integer iterations that were completed	
Number of integer feasible solutions that were found	
Number of initial basic feasible variables identified	
Time used in solving the problem excluding reading the data and displaying the solution	
Number of inversions of the basis matrix	
current value of several of the options	

Table 19: Solution Summary Outout

<table>
<tr><th colspan="2">Variable Summary</th></tr>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Col</td><td>Column number associated with each structural or logical variable in the problem</td></tr>
<tr><td>Variable Name</td><td>Name of each structural or logical variable in the problem i.e. the name of the constraint. If no ID variable is specified, the procedure names the logical variable _OBSn_, where is n the observation that describes the constraint.</td></tr>
<tr><td>Status</td><td>Variable's status in the current solution. The status can be BASIC, DEGEN, ALTER, blank, LOWBD, or UPPBD.
<table>
<tr><td>BASIC</td><td>if variable is basic</td></tr>
<tr><td>DEGEN</td><td>basic variable whose activity is at its input lower bound</td></tr>
<tr><td>ALTER</td><td>a nonbasic variable that can be brought into the basis to define an alternate optimal solution</td></tr>
<tr><td>blank</td><td>a nonbasic variable at its default lower bound 0</td></tr>
<tr><td>LOWBD</td><td>a nonbasic variable at its lower bound</td></tr>
<tr><td>UPPBD</td><td>a nonbasic variable at its upper bound.</td></tr>
</table></td></tr>
<tr><td>Type</td><td>Type of a variable (non-negative, binary, slack or other value restriction).</td></tr>
<tr><td>Price</td><td>Value of the objective coefficient associated with each variable</td></tr>
<tr><td>Activity</td><td>Activity of the variable in the current solution</td></tr>
<tr><td>Reduced Cost</td><td>Variable's reduced cost in the current solution</td></tr>
</table>

Table 20: Variable Summary Output

<table>
<tr><th colspan="2">Constraint Summary</th></tr>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Row</td><td>Constraint row number and its ID</td></tr>
<tr><td>Constraint Name</td><td>Kinds of constraints include:
<table>
<tr><td>OBJECTIVE</td><td>defines the objective function</td></tr>
<tr><td>LE</td><td>less than or equal to, constrained row</td></tr>
<tr><td>EQ</td><td>equality constrained row</td></tr>
<tr><td>GE</td><td>greater than or equal to, constrained row</td></tr>
<tr><td>RANGELE</td><td>range constrained row that has 'less than or equal to' variables</td></tr>
<tr><td>RANGEEQ</td><td>range constrained row that has 'equal to' variables</td></tr>
<tr><td>RANGEGE</td><td>range constrained row that has 'greater than or equal to' variables</td></tr>
<tr><td>FREE row</td><td>nonbinding constraint</td></tr>
</table></td></tr>
<tr><td>Type</td><td>number of the slack or surplus variable associated with the constraint row</td></tr>
<tr><td>S/S Col RHS</td><td>value of the right-hand-side constant associated with the constraint row</td></tr>
<tr><td>Activity</td><td>current activity of the row (excluding logical variables)</td></tr>
<tr><td>Dual Activity</td><td>current activity of the dual variable (shadow price) associated with the constraint row</td></tr>
</table>

Table 21: Constraint Summary Output

3 Application - The Allocation Problem

To illustrate the use of the the LP procedure in an allocation set up, we focus on an assignment problem were there are four machines that can produce any of six grades of cloth and there are five customers that demand different amounts of each grade of cloth. The return from supplying a customer with a demanded grade depends on the machine on which the cloth was made and also the machine capacity depends both upon the specific machine used and the grade of cloth made. We want to maximize the objection function thereby maximizing the return from selling the cloth. Futhermore, we illustrate the use of the sparse input format for storing data in SAS. We apply the above theory to the example taken from SAS Help and Documentation.[7]

3.1 The Problem

Definition of variables

We define the following variables

i - customer.
j - grade of cloth.
k - machine.
x_{ijk} - amount of cloth of grade j made on machine k for customer i.
r_{ijk} - return from selling one unit of grade j cloth made on machine k to customer i.
d_{ij}- demand for grade j cloth by customer i.
c_{jk}- number of units of machine k required to produce one unit of grade j cloth.

a_k - number of units of machine k available.

Objective function and Constraints

We want to maximize $\sum_{ijk} r_{ijk}x_{ijk}$

subject to: $\sum_{k} x_{ijk} = d_{ij}$ for all i and j

$\sum_{ij} c_{jk}x_{ijk} \leq a_k$ for all k

$x_{ijk} \geq 0$ for all i, j and k

In the above functions, our objective is to msximize the return that we get from selling each grade of cloth to different customers. The total amount of cloth of grade j made on each machine k for customer i for all machines must be equal to the demand for grade j cloth by customer i. The number of machine units produced by each machine should not exceed the number of units available for machine k and all amounts of cloth produced is positive.

3.2 Data Formation

In order for Proc LP to formulate a solution, the data should be stored in a way that the procedure can use and that is in the sparse format form. The data used is first stored in three data sets which will then be converted to the sparse format. The following code shows how the objective function is built in a linear program, with the other contraints built in a similar way found the appendix to form a complete model.

Algorithm 1 Generating the Objective function

```
/* generate the objective function */
_type_='MAX';
_row_='OBJ';
do k=1 to nmach;
  do i=1 to ncust;
    link readobj; /* read the objective coefficient data */
      do j=1 to ngrade;
        if grade{j}^=. then do;
        _col_='X'||put(i,1.)||put(j,1.)||put(k,1.); _coef_=grade{j};
        output;
      end;
    end;
  end;
end;
```

Table 22 below shows a preview of how the objective will look when the linear program has been formulated and the rest of the data is stored in a similar way.

type	_row_	_col_	_coef_
MAX	OBJ	X111	102
MAX	OBJ	X121	140
MAX	OBJ	X131	105
MAX	OBJ	X141	105
MAX	OBJ	X151	125
MAX	OBJ	X161	148
MAX	OBJ	X211	115
MAX	OBJ	X221	133
MAX	OBJ	X231	118
MAX	OBJ	X241	118
MAX	OBJ	X251	143
MAX	OBJ	X261	166
MAX	OBJ	X311	70

Table 22: Sparse format for the Objective function

3.3 Solution

Algorithm 2 provides the program for solving the model and saves the solution to a dataset called Primal.

Algorithm 2 Proc LP procedure

```
proc lp data=model sparsedata noprint primalout=primal;
run;
```

The program terminated successfully with an objective value of 871426. The solution summary and problem summary in the appendix give the output produces by the LP procedure.

The solution is then tabulated so as to produce a comprehensive report on the allocation of the cloths to different machines as per the demand of customers. Table 23 shows the solution of the allocation problem using PROC TABULATE (code provided in the appendix). For example, customer 1 gets 100 units of grade 1 made from machine 4, 100 units of grade 2 from machine 1, 150 units of grade 3 from machine 1, 150 units of grade 4 from machine 1, 175 units og grade 5 from machine 1 and 250 units of grade 6 from machine 1.

An Assignment Problem

		grade 1	2	3	4	5	6
		amount Sum	amount Sum	amount Sum	amount Sum	amount Sum	amount Sum
machine	**customer**						
1	**1**	.	100	150	150	175	250
	2	.	.	300	.	.	.
	3	.	.	256.72	210.31	.	.
	4	.	.	750	.	.	.
	5	.	92.27	.	.	.	.
2	**3**	.	.	143.28	.	340	.
	5	.	.	300	.	.	.
3	**2**	.	.	.	275	310	325
	3	.	.	.	289.69	.	.
	4	.	.	.	750	.	.
	5	.	.	.	.	210	360
4	**1**	100	.	.	.	.	.
	2	300	125	.	.	.	.
	3	400	.	.	.	.	.
	4	250	.	.	.	.	.
	5	.	507.73	.	.	.	.

Table 23: Solution of the Allocation Problem

4 Conclusion

The LP procedure is a very useful tool in solving linear programs, integer programs, and mixed-integer programs. It also performs parametric programming, range analysis, and reports on solution sensitivity to changes in the right-hand-side constants and price coefficients. It has been shown that the procedure works very well utilising the simplex method in solving the problem.

The short side of the LP procedure is that it can only solve linear problems. SAS also provides other packages that can help solve both linear and non linear problems. These include the OPTMODEL procedure and the procedures stated in section 2.4.3.

More research can still be done in exploring the limits of the LP procedure in SAS especially if there are more constraints. Also the study of other procedures can help enhance solving problems of any kind easily and faster.

References

[1] Richard H Bartels. A stabilization of the simplex method. *Numerische Mathematik*, 16(5):414–434, 1971.

[2] Jörg Becker, editor. *Proceedings of the University Alliance Executive Directors Workshop - ECIS 2001*, number 75, 2001.

[3] James F Campbell. Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72(2):387–405, 1994.

[4] Han Can, Ma Lu, and Luying Gan. The research on application of information technology in sports stadiums. *Physics Procedia*, 22:604–609, 2011.

[5] Abraham Charnes and William Wager Cooper. Goal programming and multiple objective optimizations: Part 1. *European Journal of Operational Research*, 1(1):39–54, 1977.

[6] George B Dantzig. Linear programming. *Operations Research*, 50(1):42–47, 2002.

[7] Ruth Farnsworth. Sas help guide for version 9.3. 2013.

[8] Paul C Gilmore and Ralph E Gomory. A linear programming approach to the cutting-stock problem. *Operations Research*, 9(6):849–859, 1961.

[9] Robert W Haessler and Paul E Sweeney. Cutting stock problems and solution procedures. *European Journal of Operational Research*, 54(2):141–150, 1991.

[10] James P Ignizio. *Linear Programming in Single-& Multiple-Objective Systems*. Prentice Hall, 1982.

[11] Paul A Jensen and Jonathan F Bard. *Operations Research Models and Methods*. John Wiley & Sons Incorporated, 2003.

[12] Lynwood A Johnson and Douglas C Montgomery. *Operations Research in Production Planning, Scheduling, and Inventory Control*, volume 6. Wiley New York, 1974.

[13] Jin Woo Lee and Soung Hie Kim. Using analytic network process and goal programming for interdependent information system project selection. *Computers & Operations Research*, 27(4):367–382, 2000.

[14] Celso C Ribeiro. Sports scheduling: Problems and applications. *International Transactions in Operational Research*, 19(1-2):201–226, 2012.

[15] Darko Skorin-Kapov, Jadranka Skorin-Kapov, and Morton O'Kelly. Tight linear programming relaxations of uncapacitated p-hub median problems. *European Journal of Operational Research*, 94(3):582–593, 1996.

[16] Moshe Sniedovich. *Dynamic Programming: Foundations and Principles*. CRC press, 2010.

[17] Ralph E Steuer. *Multiple Criteria Optimization: Theory, Computation, and Applications*. Wiley, 1986.

[18] Hamdy A Taha. *Operations Research: an Introduction*. Pearson/Prentice Hall, 2007.

[19] F.P Vasilyev and A Yu. Ivanitskiy. *In-Depth Analysis of Linear Programming*. Kluwer Academic Publishers, The Netherlands, 2001.

[20] Thomas L Wright and Patrick C Doherty. A linear programming and least squares computer method for solving petrologic mixing problems. *Geological Society of America Bulletin*, 81(7):1995–2008, 1970.

[21] Waheed Babatunde Yahya, Muhammed Kabir Garba, Samuel Oluwasuyi Ige, and Adekunle Ezekiel Adeyosoye. Profit maximization in a product mix company using linear programming. *European Journal of Business and Management*, 4(17):126–131, 2012.

5 Appendix

5.1 Example Section 2.2

SAS code for example in section 2.2

```
data;
    format _row_ $7. ;
    input _row_ $ x1 x2 _type_ $ _rhs_;
    datalines;
    object 5 4 max .
    c1 6 4 le 24
    c2 1 2 le 6
    c3 -1 1 le 1
    c4 . 2 UPPERBD .
    ;
    proc lp;
    run;
```

SAS output for example in section 2.2

The SAS System

The LP Procedure

Problem Summary	
Objective Function	Max object
Rhs Variable	_rhs_
Type Variable	_type_
Problem Density (%)	60
Variables	Number
Non-negative	1
Upper Bounded	1
Slack	3
Total	5
Constraints	Number
LE	3
Objective	1
Total	4

The SAS System

The LP Procedure

Solution Summary	
Terminated Successfully	
Objective Value	21
Phase 1 Iterations	0
Phase 2 Iterations	2
Phase 3 Iterations	0
Integer Iterations	0
Integer Solutions	0
Initial Basic Feasible Variables	5
Time Used (seconds)	0
Number of Inversions	3
Epsilon	1.00E-08
Infinity	1.797693E308
Maximum Phase 1 Iterations	100
Maximum Phase 2 Iterations	100
Maximum Phase 3 Iterations	99999999
Maximum Integer Iterations	100
Time Limit (seconds)	120

The SAS System

The LP Procedure

Variable Summary

Col	Variable Name	Status	Type	Price	Activity	Reduced Cost
1	x1	BASIC	NON-NEG	5	3	0
2	x2	BASIC	UPPERBD	4	1.5	0
3	c1		SLACK	0	0	-0.75
4	c2		SLACK	0	0	-0.5
5	c3	BASIC	SLACK	0	2.5	0

The SAS System

The LP Procedure

Constraint Summary

Row	Constraint Name	Type	S/S Col	Rhs	Activity	Dual Activity
1	object	OBJECTVE	.	0	21	.
2	c1	LE	3	24	24	0.75
3	c2	LE	4	6	6	0.5
4	c3	LE	5	1	-1.5	0

5.2 Example Section 2.3

SAS code

```
data;
    format _row_ $7. ;
    input _row_ $ p1 p2 p3 p4 p5 _type_ $ _rhs_;
    datalines;
    object 18 25 10 12 15 max .
    m1 1.2 1.3 0.7 0 0.5 le 160
    m2 0.7 2.2 1.6 0.5 1 le 200
    m3 0.9 0.7 1.3 1 0.8 le 120
    m4 1.4 2.8 0.5 1.2 0.6 le 280
    ;
    proc lp;
    run;
```

SAS output

Table 24: Add caption

The SAS System

The LP Procedure

Problem Summary	
Objective Function	Max object
Rhs Variable	_rhs_
Type Variable	_type_
Problem Density (%)	63.89
Variables	Number
Non-negative	5
Slack	4
Total	9
Constraints	Number
LE	4
Objective	1
Total	5

The SAS System

The LP Procedure

Solution Summary	
Terminated Successfully	
Objective Value	2988.73
Phase 1 Iterations	0
Phase 2 Iterations	4
Phase 3 Iterations	0
Integer Iterations	0
Integer Solutions	0
Initial Basic Feasible Variables	6
Time Used (seconds)	0
Number of Inversions	3
Epsilon	1.00E-08
Infinity	1.797693E308
Maximum Phase 1 Iterations	100
Maximum Phase 2 Iterations	100
Maximum Phase 3 Iterations	1E+08
Maximum Integer Iterations	100
Time Limit (seconds)	120

The SAS System

The LP Procedure

Variable Summary

Col	Variable Name	Status	Type	Price	Activity	Reduced Cost
1	p1	BASIC	NON-NEG	18	58.9614	0
2	p2	BASIC	NON-NEG	25	62.6346	0
3	p3		NON-NEG	10	0	-13.53
4	p4	BASIC	NON-NEG	12	10.5763	0
5	p5	BASIC	NON-NEG	15	15.6428	0
6	m1		SLACK	0	0	-4.8195
7	m2		SLACK	0	0	-5.2016
8	m3		SLACK	0	0	-8.9635
9	m4		SLACK	0	0	-0.3631

The LP Procedure

Constraint Summary

Row	Constraint Name	Type	S/S Col	Rhs	Activity	Dual Activity
1	object	OBJECTVE	.	0	2988.73	.
2	m1	LE	6	160	160	4.81951
3	m2	LE	7	200	200	5.2016
4	m3	LE	8	120	120	8.96348
5	m4	LE	9	280	280	0.3631

5.3 Application Section 3

Data for Generating constraints

Algorithm 3 Datasets for the Constraints

```
title 'An Assignment Problem';
data object;
input machine customer grade1 grade2 grade3 grade4 grade5 grade6;
datalines;
1 1 102 140 105 105 125 148
1 2 115 133 118 118 143 166
1 3 70 108 83 83 88 86
1 4 79 117 87 87 107 105
1 5 77 115 90 90 105 148
2 1 123 150 125 124 154 .
2 2 130 157 132 131 166 .
2 3 103 130 115 114 129 .
2 4 101 128 108 107 137 .
2 5 118 145 130 129 154 .
3 1 83 . . 97 122 147
3 2 119 . . 133 163 180
3 3 67 . . 91 101 101
3 4 85 . . 104 129 129
3 5 90 . . 114 134 179
4 1 108 121 79 . 112 132
4 2 121 132 92 . 130 150
4 3 78 91 59 . 77 72
4 4 100 113 76 . 109 104
4 5 96 109 77 . 105 145
;

data demand;
input customer grade1 grade2 grade3 grade4 grade5 grade6;
datalines;
1 100 100 150 150 175 250
2 300 125 300 275 310 325
3 400 0 400 500 340 0
4 250 0 750 750 0 0
5 0 600 300 0 210 360
;

data resource;
input machine grade1 grade2 grade3 grade4 grade5 grade6 avail;
datalines;
1 .250 .275 .300 .350 .310 .295 744
2 .300 .300 .305 .315 .320 . 244
3 .350 . . .320 .315 .300 790
4 .280 .275 .260 . .250 .295 672
;
```

Linear Programming Model

Algorithm 4 Building Linear programming Constraints

```
/* build the linear programming model */
data model;
array grade{6} grade1-grade6;
length _type_ $ 8 _row_ $ 8 _col_ $ 8;
keep _type_ _row_ _col_ _coef_;
ncust=5; nmach=4; ngrade=6;
/* generate the objective function */
_type_='MAX'; _row_='OBJ';
do k=1 to nmach; do i=1 to ncust;
link readobj; /* read the objective coefficient data */
do j=1 to ngrade;
if grade{j}^=. then do;
_col_='X'||put(i,1.)||put(j,1.)||put(k,1.);
_coef_=grade{j};
output;
end;
end; end; end;
/* generate the demand constraints */
do i=1 to ncust;
link readdmd; /* read the demand data */
do j=1 to ngrade;
if grade{j}^=. then do;
_type_='EQ';
_row_='DEMAND'||put(i,1.)||put(j,1.);
_col_='_RHS_';
_coef_=grade{j};
output;
_type_=' ';
do k=1 to nmach;
_col_='X'||put(i,1.)||put(j,1.)||put(k,1.);
_coef_=1.0; output;
end; end; end; end;
/* generate the machine constraints */
do k=1 to nmach;
link readres; /* read the machine data */
_type_='LE';
_row_='MACHINE'||put(k,1.);
_col_='_RHS_';
_coef_=avail;
output;
_type_=' ';
do i=1 to ncust;
do j=1 to ngrade;
if grade{j}^=. then do;
_col_='X'||put(i,1.)||put(j,1.)||put(k,1.);
_coef_=grade{j};
output; end;
end; end; end;
readobj: set object;
return;
readdmd: set demand;
return;
readres: set resource;
return;
run;
```

SAS Output - Application

An Assignment Problem

The LP Procedure

Problem Summary	
Objective Function	Max OBJ
Rhs Variable	_RHS_
Type Variable	_type_
Problem Density (%)	5.31
Variables	Number
Non-negative	120
Slack	4
Total	124
Constraints	Number
LE	4
EQ	30
Objective	1
Total	35

Table 25: Problem Summary

An Assignment Problem

Solution Summary	
Terminated Successfully	
Objective Value	871426
Phase 1 Iterations	0
Phase 2 Iterations	40
Phase 3 Iterations	0
Integer Iterations	0
Integer Solutions	0
Initial Basic Feasible Variables	36
Time Used (seconds)	0
Number of Inversions	3
Epsilon	1.00E-08
Infinity	1.797693E308
Maximum Phase 1 Iterations	100
Maximum Phase 2 Iterations	100
Maximum Phase 3 Iterations	999999
Maximum Integer Iterations	100
Time Limit (seconds)	120

Table 26: Solution Summary

Tabulating the Solution

Algorithm 5 Tabulating the Solution

```
data solution;
set primal;
keep customer grade machine amount;
if substr(_var_,1,1)='X' then do;
if _value_^=0 then do;
customer = substr(_var_,2,1);
grade = substr(_var_,3,1);
machine = substr(_var_,4,1);
amount = _value_;
output;
end; end;
run;
proc tabulate data=solution;
class customer grade machine;
var amount;
table (machine*customer), (grade*amount);
run;
```

Subjective Bayesian analysis of the univariate normal model

Priyanka Nagar 12023109

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mrs J van Niekerk

Department of Statistics, University of Pretoria

2 November 2015

Abstract

The normal model is widely used in modern statistical modeling and hence the estimation of the parameters are very important. This study produces subjective Bayesian estimators under a normal-inverse gamma prior and a normal-gamma prior and LINEX loss function. It is shown that the normal-gamma prior results in estimators with less error than the well-known inverse gamma prior as well as the MLE's with a simulation study. The analytical expressions of the estimators are used instead of the MCMC sampling.

Declaration

I, *Priyanka Nagar,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Priyanka Nagar

_____
Janet van Niekerk

_____
Date

Acknowledgments

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF.

Contents

List of Figures

List of Tables

1 Introduction

One of the most commonly used distributions in statistics is the normal distribution . Bayesian analysis is frequently used to estimate the parameters of the normal distribution under the assumption of different loss functions as discussed in Murphy [6], Zellner [12], Samaniego [9] and Hoque et al [5].

Bayesian analysis can be subjective or objective depending on the choice of the prior. An objective prior is when there is very little prior information available or when a distribution is chosen such that all possible values of the parameter are equally likely. An advantage of an objective prior is that sometimes the results acquired is the same as that of the frequentists methods. Comparisons between the Bayesian and frequentist approaches to estimations are discussed by Samaniego [9] . A subjective prior is when previous information is used to make a decision regarding which prior distribution should be used in a particular study. Subjective priors are more beneficial to objective priors in that they include additional information about the study into the Bayesian analysis. Press [7] explores subjective and objective Bayesian statistics and its application. The prior density function is usually obtained from previous sampling information. The posterior distribution is the result of applying Bayes' theorem and is the conditional distribution of the given sample data. The posterior distribution is proportional to the product of the prior distribution and the joint probability density functions of the sample data. The joint probability density functions of the sample data can be replaced by the likelihood function where the likelihood function is the joint probability density functions of the sample data. From Bayes' theorem, the posterior and marginal distributions of the unknown parameters can be found. Using the Bayesian approach, estimators for the unknown parameters can be derived by choosing an appropriate loss function.

When a parameter $\hat{\theta}$ is used as an estimator for an unknown parameter θ, the loss incurred is measured by a loss function. An estimator which minimizes the expected value of the loss function with respect to the posterior distribution is obtained. The expected value of a loss function is defined as the risk function. The most popular choice of loss functions is the quadratic loss function. This loss function is symmetric and is a popular choice as the Bayes' estimator under the quadratic loss function is the expected value of the posterior distribution which makes the computations simpler. The zero/one loss function is another symmetric function used when testing if the unknown parameter is within a predetermined interval. The Bayes' estimator in this case is the mode of the posterior distribution. Other symmetric loss functions include the linear loss function and the absolute error loss function, where the Bayes' estimate is the median of the posterior distribution under the absolute error loss. An asymmetric loss function is the Linear Exponential (LINEX) loss function which was introduced by Varian [11]. The LINEX loss function differentiates between under-estimation and over-estimation.This LINEX loss function is further discussed in section 2.

In this study, the Bayesian approach will be used with the assumption that we have a random sample from a normal distribution with unknown mean and unknown variance. Bayesian estimators for these unknown parameters will be derived using two different joint priors under the LINEX loss function, namely, the normal-inverse gamma prior and the normal-gamma prior. These estimators are then evaluated in section 5.

2 LINEX Loss Function

The simplicity of the squared-error loss function makes it a popular choice. The squared-error loss is symmetric and does not differentiate between under-estimation and over-estimation. However, this is not appropriate in cases when the effects of over-estimation or under-estimation is more severe. A more appropriate asymmetric loss function was introduced by Varian (1975) [11] called the Linear Exponential loss function (LINEX). The LINEX loss allows for over-estimation and under-estimation by assigning unequal weights through the introduction of a shape parameter. In manufacturing, for example, the over-estimation of the average shelf life of a product for consumer information would have more severe consequences than under-estimation. The LINEX loss function used to estimate a parameter θ is defined as

$$L(\Delta) = b\left\{exp(a\Delta) - a\Delta - 1\right\} \text{ for } a \neq 0 \text{ and } b > 0$$

The $\Delta = (\theta * -\theta)$ is called the estimation error. The parameter a is the shape parameter and b is the scale parameter of $L(\Delta)$. For $a > 0$, the effect of over-estimation is more severe than under-estimation and for $a < 0$ the effect of under-estimation is more severe than over-estimation. The skewness of the loss function is determined by the weight of a. It comprises of a linear and exponential part. The loss function increases exponentially on one side of $\Delta = 0$ and increases linearly on the other side. If $a > 0$ then for $\Delta > 0$ the loss function declines exponentially and for $\Delta < 0$ the loss function grows linearly.

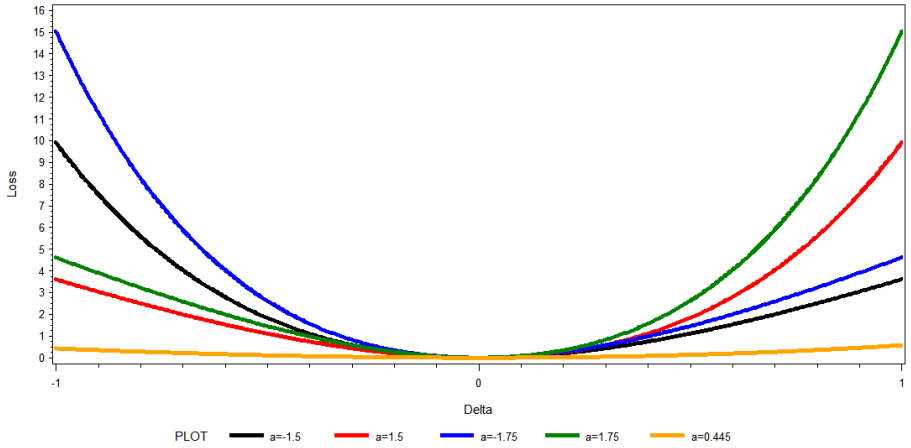


Figure 1: LINEX loss function for constant b

From the above graph, it can be seen that the LINEX loss function decreases exponentially and increases linearly for $a < 0$. The slope of the loss function becomes steeper as the value of alpha becomes smaller. For $a > 0$ the loss function decreases linearly and increases exponentially. The slope of the loss function becomes steeper as the value of alpha becomes larger. As $a \to 0$, the $\exp(a\Delta) \to 1$ and therefore the loss function $\mathrm{L}(\Delta) \to 0$. Also for very small a the LINEX loss is approximately equal to the squared-error loss.

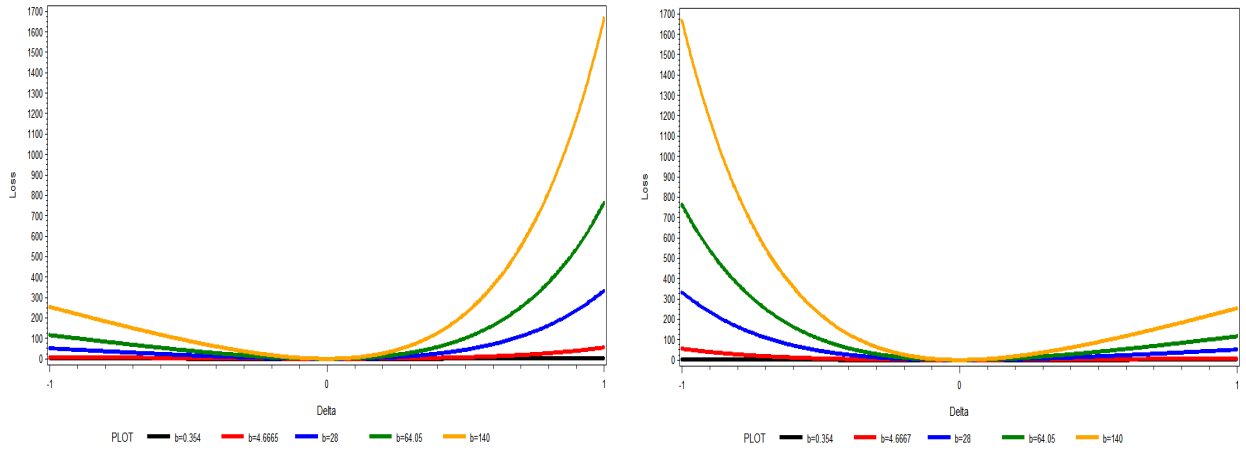


Figure 2: LINEX Loss function for constant $a > 0$ and $a < 0$

As b increases the loss function magnifies, i.e. the larger the b the greater the loss for a specific Δ. As $b \to 0$ the loss function $\mathrm{L}(\Delta) \to 0$.

3 Normal-inverse gamma prior

3.1 Likelihood

Let $X_i \sim N(\mu, \sigma^2)$ (A.1) for all i = 1, 2, 3,...,n. Therefor the density function of $x|\mu, \sigma^2$ is given by

$$f(\underline{x}|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\} \quad ,\sigma^2 > 0 \tag{1}$$

Thus, the likelihood function (A.8) is

$$\begin{aligned} L(f(\underline{x}|\mu,\sigma^2)) &= \prod_{i=1}^{n} f(x_i|\mu,\sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}\right\} \\ &= \prod_{i=1}^{n} (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}\right\} \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{\sigma^2}\right\} \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2}{\sigma^2}\right)\right\} \end{aligned}$$

3.2 Priors

Assume that $\mu|\sigma^2 \sim N(\mu_0, \sigma^2)$ (A.1) then,

$$\begin{aligned} p(\mu|\sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \\ &= (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \quad ,-\infty < \mu < \infty \end{aligned}$$

Assume that $\sigma^2 \sim IG(\alpha, \beta)$ (A.2) then,

$$p(\sigma^2) = \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\sigma^2\right)^{-\alpha-1} exp\left\{-\frac{\beta}{\sigma^2}\right\} \qquad ,\sigma^2 > 0$$

Joint priors

The joint probability density function (A.9) of μ and σ^2 is

$$\begin{aligned} p(\mu,\sigma^2) &= p(\mu|\sigma^2)p(\sigma^2) \\ &= (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \times \frac{\beta^\alpha}{\Gamma(\alpha)} (\sigma^2)^{-\alpha-1} exp\left\{-\frac{\beta}{\sigma^2}\right\} \end{aligned}$$

$$p(\mu,\sigma^2) = (2\pi)^{-\frac{1}{2}} (\sigma^2)^{-\alpha-\frac{3}{2}} \times \frac{\beta^\alpha}{\Gamma(\alpha)} \times exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \frac{\beta}{\sigma^2}\right\} \tag{2}$$

3.3 Joint posterior distribution

By using (A.10) the posterior density function $q(\mu, \sigma^2|\underline{x})$ is given by

$$\begin{aligned} q(\mu,\sigma^2|\underline{x}) &\propto (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2}{\sigma^2}\right)\right\} \times (2\pi)^{-\frac{1}{2}}(\sigma^2)^{-\alpha-\frac{3}{2}} \\ &\quad \times\frac{\beta^\alpha}{\Gamma(\alpha)} \times exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \frac{\beta}{\sigma^2}\right\} \end{aligned}$$

Hence,

$$q(\mu,\sigma^2|\underline{x}) = W \times \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{\sigma^2}\right) - \frac{\beta}{\sigma^2}\right\} \tag{3}$$

with W the normalizing constant such that $\int_{-\infty}^{\infty}\int_0^{\infty} q(\mu,\sigma^2|\underline{x})d\sigma^2 d\mu = 1$

3.4 Marginal posterior distributions

Marginal posterior distribution of μ

The marginal distribution of μ is obtained from equation (3) as follows

$$\begin{aligned} q(\mu|\underline{x}) &= \int_0^{\infty} q(\mu,\sigma^2|\underline{x})d\sigma^2 \\ &\propto \int_0^{\infty} \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{\sigma^2}\right) - \frac{\beta}{\sigma^2}\right\} d\sigma^2 \\ &= \int_0^{\infty} \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2\sigma^2}\left[\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2\right) + (\mu-\mu_0)^2\right] - \frac{\beta}{\sigma^2}\right\} d\sigma^2 \\ &= \int_0^{\infty} \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{\frac{-\frac{1}{2}\left[\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2\right) + (\mu-\mu_0)^2\right] - \beta}{\sigma^2}\right\} d\sigma^2 \end{aligned}$$

Since σ^2 follows a an IG $\left(\alpha + \frac{n-1}{2}, \frac{1}{2}\left[\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2\right) + (\mu-\mu_0)^2\right] + \beta\right)$ (A.2),

$$\begin{aligned} q(\mu|\underline{x}) &\propto \frac{\Gamma(\alpha+\frac{n-1}{2})}{\left(\frac{1}{2}\left[\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2\right) + (\mu-\mu_0)^2\right] + \beta\right)^{\alpha+\frac{n-1}{2}}} \\ &= \frac{\Gamma(\alpha+\frac{n-1}{2})}{\left(\frac{1}{2}\left[\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2\right) + (\mu-\mu_0)^2\right] + \beta\right)^{\alpha+\frac{n-1}{2}}} \\ &\propto \frac{\Gamma(\alpha+\frac{n-1}{2})}{\left[\frac{n+1}{2}\left\{\mu - \left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\right\}^2 - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1} + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} + \beta\right]^{\alpha+\frac{n-1}{2}}} \\ &\propto \left[\frac{n+1}{2}\left\{\mu - \left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\right\}^2 - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1} + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} + \beta\right]^{-\alpha-\frac{n-1}{2}} \end{aligned}$$

$$
\begin{aligned}
&= \left[\frac{(2\alpha+n-2)(n+1)\left(\mu-\left[\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right]\right)^2}{2(2\alpha+n-2)}+\left(\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right]^{-\alpha-\frac{n-1}{2}} \\
&= \left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}^{-\alpha-\frac{n-1}{2}} \\
&\quad \left[\frac{(2\alpha+n-2)(n+1)\left(\mu-\left[\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right]\right)^2}{2(2\alpha+n-2)\left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}}+1\right]^{-\alpha-\frac{n-1}{2}} \\
&= \left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}^{-\alpha-\frac{n-1}{2}} \\
&\quad \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} \\
&\propto \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} \\
&\sim t\left(2\alpha+n-2,\frac{\sum_{i=1}^n x_i+\mu_0}{n+1},\frac{\left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}}{\left(\alpha+\frac{n}{2}-1\right)(n+1)}\right)
\end{aligned}
$$

Hence, the marginal distribution of μ follows a non-central t distribution (A.4) with $(2\alpha+n-2)$ degrees of freedom and non centrality parameter $\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)$ and $\left(\frac{\left\{\beta+\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right\}}{\left(\alpha+\frac{n}{2}-1\right)(n+1)}\right)$. See theorem (A.11) for the marginal distribution.

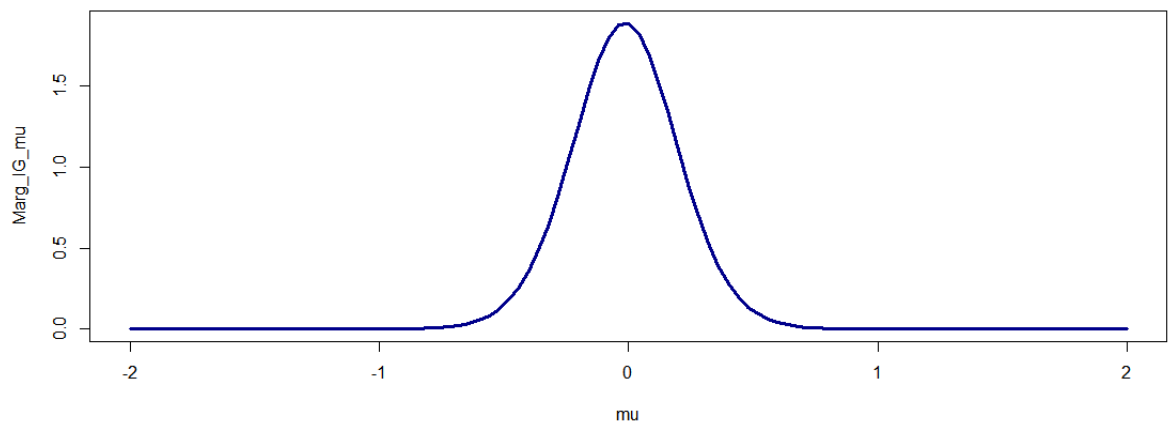


Figure 3: Marginal distribution of μ for the inverse gamma prior with $\alpha = 4$, $\beta = 3$, $n = 20$ and $\mu_0 = \bar{x}$.

Marginal posterior distribution of σ^2

The marginal distribution of σ^2 is obtained from equation (3) as follows

$$
\begin{aligned}
q(\sigma^2|\underline{x}) &= \int_{-\infty}^{\infty} q(\mu, \sigma^2|\underline{x}) d\mu \\
&\propto \int_{-\infty}^{\infty} (\sigma^2)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2}{\sigma^2}\right) - \frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \frac{\beta}{\sigma^2}\right\} d\mu \\
&= \int_{-\infty}^{\infty} (\sigma^2)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{\beta}{\sigma^2}\right\} exp\left\{-\frac{1}{2\sigma^2}\left(\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + \mu^2 - 2\mu\mu_0 + \mu_0^2\right)\right\} d\mu \\
&= (\sigma^2)^{-\alpha-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{\beta}{\sigma^2}\right\} \int_{-\infty}^{\infty} exp\left\{-\frac{1}{2\sigma^2}\left(\mu^2(n+1) - 2\mu\left(\sum_{i=1}^n x_i + \mu_0\right) + \sum_{i=1}^n x_i^2 + \mu_0^2\right)\right\} d\mu \\
&= (\sigma^2)^{-\alpha-\frac{n}{2}-\frac{3}{2}} \times exp\left\{-\frac{\beta}{\sigma^2} - \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2\sigma^2}\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{n+1}{2\sigma^2}\left[\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2 - \left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2\right]\right\} d\mu \\
&= (\sigma^2)^{-\alpha-\frac{n}{2}-\frac{3}{2}} \times exp\left\{-\frac{\beta}{\sigma^2} - \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2\sigma^2} + \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)\sigma^2}\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{1}{2}\left(\frac{\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2}{\frac{\sigma^2}{n+1}}\right)\right\} d\mu
\end{aligned}
$$

Since μ follows a $N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$ distribution (A.1),

$$
\begin{aligned}
q(\sigma^2|\underline{x}) &= \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-\frac{3}{2}} \times exp\left\{-\frac{\beta}{\sigma^2} - \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2\sigma^2} + \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)\sigma^2}\right\} \times \left(\frac{2\pi\sigma^2}{n+1}\right)^{\frac{1}{2}} \\
&\propto \left(\sigma^2\right)^{-\alpha-\frac{n}{2}-1} \times exp\left\{-\frac{\beta}{\sigma^2} - \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2\sigma^2} + \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)\sigma^2}\right\} \\
&= \left(\sigma^2\right)^{-(\alpha+\frac{n}{2})-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} \\
&\sim \text{IG}\left(\alpha+\frac{n}{2}, \beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)
\end{aligned}
$$

Hence, the marginal distribution of σ^2 follows a inverse gamma (A.2) distribution with parameters $\left(\alpha+\frac{n}{2}\right)$ and $\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)$ respectively. See theorem (A.11) for the marginal distribution.

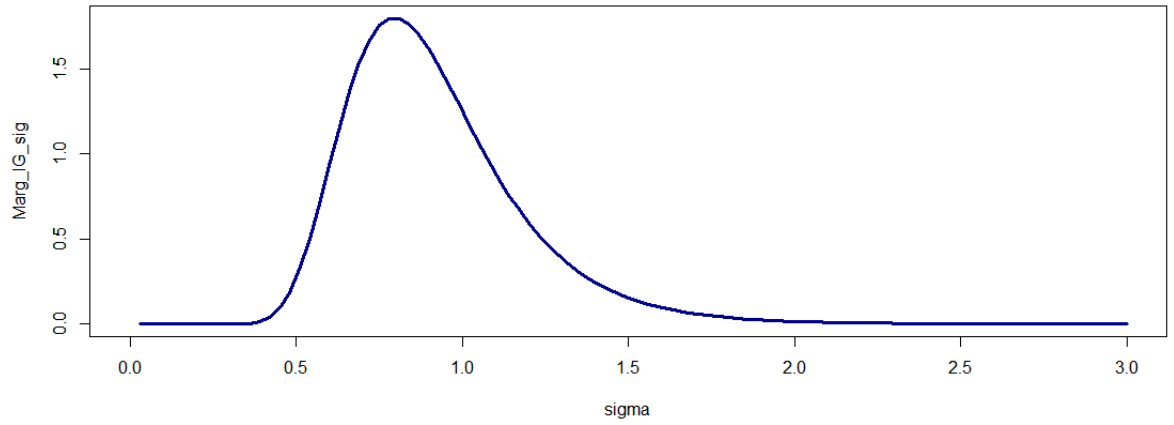


Figure 4: Marginal distribution of σ^2 for the inverse gamma prior with $\alpha = 4$, $\beta = 3$, $n = 20$ and $\mu_0 = \bar{x}$.

3.5 Risk function and estimators

3.5.1 Estimator of μ

Loss function

Using the LINEX loss function (A.12) where γ is the estimator for μ

$$L(\mu,\gamma) = b\left[exp\left\{a\left(\gamma-\mu\right)\right\} - a\left(\gamma-\mu\right) - 1\right]$$

where $b > 0$ and $a \neq 0$.

Risk function

The risk function (A.13) is the expected value of the loss function with respect to the marginal distribution of μ

$$R(\mu,\gamma) = \mathbb{E}_\mu[L(\mu,\gamma)]$$

$$
\begin{aligned}
&= \int_0^{\infty} L(\mu,\gamma) q(\mu|\underline{x}) d\mu \\
&= \int_{-\infty}^{\infty} b\left[exp\left\{a\left(\gamma-\mu\right)\right\}-a\left(\gamma-\mu\right)-1\right] \times \frac{\Gamma\left(\frac{2\alpha+n-1}{2}\right)}{\Gamma\left(\frac{2\alpha+n-2}{2}\right)} \\
&\quad \times \frac{1}{\sqrt{\left((2\alpha+n-2)\pi\left(\frac{\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{\left(\alpha+\frac{n}{2}-1\right)(n+1)}\right)\right)}} \\
&\quad \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu \\
&\quad \text{Let } R_2 = \frac{\Gamma\left(\frac{2\alpha+n-1}{2}\right)}{\Gamma\left(\frac{2\alpha+n-2}{2}\right)}\left((2\alpha+n-2)\pi\left(\frac{\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{\left(\alpha+\frac{n}{2}-1\right)(n+1)}\right)\right)^{-\frac{1}{2}} \text{ then,} \\
R(\mu,\gamma) &= \int_{-\infty}^{\infty} R_2 b\left[exp\left\{a\left(\gamma-\mu\right)\right\}-a\left(\gamma-\mu\right)-1\right] \times \\
&\quad \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu \\
&= \int_{-\infty}^{\infty} R_2 b exp\left\{a\left(\gamma-\mu\right)\right\} \times \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu \\
&\quad - \int_{-\infty}^{\infty} R_2 b a\left(\gamma-\mu\right) \times \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu \\
&\quad - \int_{-\infty}^{\infty} R_2 b \times \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu \\
&= b exp\left\{a\gamma\right\} \int_{-\infty}^{\infty} R_2 exp\left\{-a\mu\right\} \times \left[\frac{\left(\mu-\left[\frac{\sum_{i=1}^{n} x_i+\mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2}-\frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}}+1\right]^{-\alpha-\frac{n-1}{2}} d\mu
\end{aligned}
$$

$$
\begin{aligned}
&-ba\gamma \int_{-\infty}^{\infty} R_2 \times \left[\frac{\left(\mu - \left[\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta + \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}} + 1 \right]^{-\alpha - \frac{n-1}{2}} d\mu \\
&+ba \int_{-\infty}^{\infty} \mu R_2 \times \left[\frac{\left(\mu - \left[\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta + \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}} + 1 \right]^{-\alpha - \frac{n-1}{2}} d\mu \\
&-b \int_{-\infty}^{\infty} R_2 \times \left[\frac{\left(\mu - \left[\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right]\right)^2}{(2\alpha+n-2)\frac{2\left\{\beta + \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{n+1}\right\}}{(2\alpha+n-2)(n+1)}} + 1 \right]^{-\alpha - \frac{n-1}{2}} d\mu \\
= \quad & bexp\left\{a\gamma\right\} M_\mu(-a) - ba\gamma + ba\mathbb{E}_\mu[\mu|\underline{x}] - b
\end{aligned}
$$

where $M_\mu(-a)$ is the moment generating function (A.7) of $\mu|\underline{x}$ which follows a

$$
t\left(2\alpha+n-2, \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\left\{\beta + \frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{n+1}\right\}}{\left(\alpha + \frac{n}{2} - 1\right)(n+1)}\right)
$$

distribution in the point $(-a)$ and $\mathbb{E}_\mu(\mu|\underline{x})$ is the expected value (A.6) of μ .

Estimator

By differentiating $R(\mu, \gamma)$ and setting the derivative equal to zero, an estimator $\hat{\gamma}$ for μ can be obtained such that $\hat{\gamma}$ minimizes $R(\mu\gamma)$.

$$
\frac{d}{d\gamma}R(\mu,\gamma) = baexp\{a\gamma\}M_\mu(-a) - ba
$$

Setting the $\frac{d}{d\gamma}R(\mu,\gamma) = 0$ and solving for γ.

$$
\begin{aligned}
\frac{d}{d\theta}R(\mu,\gamma) &= 0 \\
ba &= baexp\left\{a\hat{\gamma}\right\} M_\mu(-a) \\
1 &= exp\left\{a\hat{\gamma}\right\} M_\mu(-a) \\
exp\left\{a\hat{\gamma}\right\} &= \frac{1}{M_\mu(-a)} \\
a\hat{\gamma} &= ln\left(\frac{1}{M_\mu(-a)}\right) \\
a\hat{\gamma} &= ln\left(M_\mu(-a)\right)^{-1} \\
\hat{\gamma} &= -\frac{1}{a} ln\left(M_\mu(-a)\right)
\end{aligned}
$$

Therefore the estimator for μ is given by

$$
\hat{\mu} = ln\left[M_\mu(-a)\right]^{-\frac{1}{a}} \tag{4}
$$

3.5.2 Estimator of σ^2

Loss function

Using the LINEX loss function (A.12) where θ is the estimator for σ^2

$$L(\sigma^2,\theta) = b\left[exp\left\{a\left(\theta-\sigma^2\right)\right\} - a\left(\theta-\sigma^2\right) - 1\right]$$

where $b > 0$ and $a \neq 0$.

Risk function

The risk function (A.13) is the expected value (A.6) of the loss function with respect to the marginal distribution of σ^2

$$\begin{aligned}
R(\sigma^2,\theta) &= \mathbb{E}_{\sigma^2}[L(\sigma^2,\theta)] \\
&= \int_0^\infty L(\sigma^2,\theta)q(\sigma^2|\underline{x})d\sigma^2 \\
&= \int_0^\infty b\left[exp\left\{a\left(\theta-\sigma^2\right)\right\} - a\left(\theta-\sigma^2\right) - 1\right] \times \frac{\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)^{\alpha+\frac{n}{2}}}{\Gamma\left(\alpha+\frac{n}{2}\right)} \\
&\quad \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad \text{Let } R_1 = \frac{\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)^{\alpha+\frac{n}{2}}}{\Gamma\left(\alpha+\frac{n}{2}\right)} \text{ then,} \\
&= \int_0^\infty R_1 b exp\left\{a\left(\theta-\sigma^2\right)\right\}\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty R_1 ba\left(\theta-\sigma^2\right)\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty R_1 b\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&= \int_0^\infty R_1 b exp\left\{a\theta\right\}\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} exp\left\{-a\sigma^2 - \frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty R_1 ba\theta\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad + \int_0^\infty R_1 ba\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty R_1 b\left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2
\end{aligned}$$

$$
\begin{aligned}
&= R_1 b exp\{a\theta\} \int_0^\infty \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} exp\left\{-a\sigma^2 - \frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad -ba\theta \int_0^\infty R_1 \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad +R_1 ba \int_0^\infty \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 \\
&\quad -b \int_0^\infty R_1 \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)-1} \times exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2
\end{aligned}
$$

Using the Bessel function of the third kind (A.5) with $v_0 = -\left(\alpha + \frac{n}{2}\right)$, we get

$$
\begin{aligned}
R(\sigma^2,\theta) &= R_1 b exp\{a\theta\} \times 2 \times \left(\frac{\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)}{a}\right)^{\frac{v_0}{2}} \\
&\quad K_{v_0}\left(2\sqrt{a\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)}\right) - ba\theta \\
&\quad +R_1 ba \int_0^\infty \left(\sigma^2\right)^{-\left(\alpha+\frac{n}{2}\right)} exp\left\{-\frac{1}{\sigma^2}\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)\right\} d\sigma^2 - b
\end{aligned}
$$

Since $\sigma^2 \sim IG\left(\alpha + \frac{n}{2} - 1, \beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)$ we get,

$$
\begin{aligned}
R(\sigma^2,\theta) &= R_1 b exp\{a\theta\} \times 2 \times \left(\frac{\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)}{a}\right)^{\frac{v_0}{2}} \\
&\quad K_{v_0}\left(2\sqrt{a\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)}\right) - ba\theta \\
&\quad +R_1 ba \times \frac{\Gamma\left(\alpha + \frac{n}{2} - 1\right)}{\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right)^{\alpha+\frac{n}{2}-1}} - b
\end{aligned}
$$

.

By using the Bessel function of the third kind (A.5), certain conditions need to be placed on the parameters. For this $R(\sigma^2,\theta)$ to hold, $a > 0$, and $\left(\beta + \frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{n+1}\right) > 0$.

Estimator

By differentiating $R(\sigma^2,\theta)$ and setting the derivative equal to zero, an estimator $\hat{\theta}$ for σ^2 can be obtained such that $\hat{\theta}$ minimizes $R(\sigma^2,\theta)$.

$$\frac{d}{d\theta}R(\sigma^2,\theta) = baR_1 exp\left\{a\theta\right\} \times 2 \times \left(\frac{\left(\beta + \frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right)}{a}\right)^{\frac{v_0}{2}}$$
$$K_{v_0}\left(2\sqrt{a\left(\beta + \frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right)}\right) - ba$$

Setting the $\frac{d}{d\theta}R(\sigma^2,\theta) = 0$ and solving for θ.

Let $\delta = 2 \times \left(\frac{\left(\beta+\frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right)}{a}\right)^{\frac{v_0}{2}} K_{v_0}\left(2\sqrt{a\left(\beta + \frac{\sum_{i=1}^{n} x_i^2+\mu_0^2}{2} - \frac{1}{2}\frac{\left(\sum_{i=1}^{n} x_i+\mu_0\right)^2}{n+1}\right)}\right)$

$$\begin{aligned}
\frac{d}{d\theta}R(\sigma^2,\theta) &= 0 \\
ba &= baR_1 exp\left\{a\hat{\theta}\right\} \times \delta \\
1 &= R_1 exp\left\{a\hat{\theta}\right\} \times \delta \\
exp\left\{a\hat{\theta}\right\} &= \left[R_1 \times \delta\right]^{-1} \\
a\hat{\theta} &= ln\left[R_1 \times \delta\right]^{-1} \\
\hat{\theta} &= -\frac{1}{a} ln\left[R_1 \times \delta\right]
\end{aligned}$$

Therefore the estimator for σ^2 is given by

$$\hat{\sigma^2} = ln\left[R_1 \times \delta\right]^{-\frac{1}{a}} \tag{5}$$

4 Normal-gamma prior

4.1 Likelihood

Let $X_i \sim N(\mu, \sigma^2)$ (A.1) for all i = 1, 2, 3,...,n. Therefor the density function of $x|\mu,\sigma^2$ is given by

$$f(\underline{x}|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\} \quad , \sigma^2 > 0 \tag{6}$$

Thus, the likelihood function (A.8) is

$$\begin{aligned}
L(f(\underline{x}|\mu,\sigma^2)) &= \prod_{i=1}^{n} f(x_i|\mu,\sigma^2) \\
&= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}\right\} \\
&= \prod_{i=1}^{n} (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}\right\} \\
&= (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\frac{(x_i-\mu)^2}{\sigma^2}\right\}
\end{aligned}$$

$$L(f(\underline{x}|\mu,\sigma^2)) = (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2}{\sigma^2}\right)\right\} \tag{7}$$

4.2 Priors

Assume that $\mu|\sigma^{2\sim}$ N(μ_0, σ^2) (A.1) then,

$$\begin{aligned} p(\mu|\sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \\ &= (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \qquad ,-\infty<\mu<\infty \end{aligned}$$

Assume that$\sigma^{2\sim}$ G(θ, γ) (A.3) then,

$$p(\sigma^2) = \frac{\gamma^\theta}{\Gamma(\theta)}(\sigma^2)^{\theta-1} exp(-\gamma\sigma^2) \qquad ,\sigma^2>0$$

Joint priors

The joint probability density function (A.9) of μ and σ^2 is

$$\begin{aligned} p(\mu,\sigma^2) &= p(\mu|\sigma^2)p(\sigma^2) \\ &= (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2}\right\} \times \frac{\gamma^\theta}{\Gamma(\theta)}(\sigma^2)^{\theta-1} exp(-\gamma\sigma^2) \\ &= (2\pi)^{-\frac{1}{2}}\frac{\gamma^\theta}{\Gamma(\theta)} \times (\sigma^2)^{\theta-\frac{3}{2}} \times exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \gamma\sigma^2\right\} \end{aligned}$$

$$p(\mu,\sigma^2) = (2\pi)^{-\frac{1}{2}}\frac{\gamma^\theta}{\Gamma(\theta)} \times (\sigma^2)^{\theta-\frac{3}{2}} \times exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \gamma\sigma^2\right\} \tag{8}$$

4.3 Joint posterior distribution

By using (A.10) the posterior density function $q(\mu, \sigma^2|\underline{x})$ is given by

$$\begin{aligned} q(\mu,\sigma^2|\underline{x}) \propto\ & (2\pi\sigma^2)^{-\frac{n}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2}{\sigma^2}\right)\right\} \times (2\pi)^{-\frac{1}{2}}\frac{\gamma^\theta}{\Gamma(\theta)} \\ & \times(\sigma^2)^{\theta-\frac{3}{2}} \times exp\left\{-\frac{1}{2}\frac{(\mu-\mu_0)^2}{\sigma^2} - \gamma\sigma^2\right\} \end{aligned}$$

Hence,

$$q(\mu,\sigma^2|\underline{x}) = J \times (\sigma^2)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{\sigma^2}\right) - \gamma\sigma^2\right\} \tag{9}$$

with J the normalizing constant such that $\int_{-\infty}^{\infty}\int_0^{\infty} q(\mu,\sigma^2|\underline{x})d\sigma^2 d\mu = 1$

4.4 Marginal posterior distributions

Marginal posterior distribution of μ

The marginal distribution of μ is obtained from (9) as follows

$$
\begin{aligned}
q(\mu|\underline{x}) &= \int_0^\infty q(\mu,\sigma^2|\underline{x})d\sigma^2 \\
&\propto \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{\sigma^2}\right) - \gamma\sigma^2\right\} d\sigma^2 \\
&\propto \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1} exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right) - \gamma\sigma^2\right\} d\sigma^2
\end{aligned}
$$

Using the Bessel function of the third kind (A.5) with $v_2 = \theta - \frac{n}{2} - \frac{1}{2}$, we get

$$
\begin{aligned}
q(\mu|\underline{x}) \propto\ & 2\times\left(\frac{\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}{\gamma}\right)^{\frac{v_2}{2}} \\
& K_{v_2}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}\right)
\end{aligned}
$$

For $q(\mu|\underline{x})$ to be a valid density function, $\int_{-\infty}^{\infty} q(\mu|\underline{x})d\mu = 1$. Therefore,

$$
\begin{aligned}
1 &= \int_{-\infty}^{\infty} C_1\times\left(\frac{\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}{\gamma}\right)^{\frac{v_2}{2}} \\
&\quad K_{v_2}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}\right) d\mu \\
\therefore C_1^{-1} &= \int_{-\infty}^{\infty}\int_0^{\infty} (\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1} exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right) - \gamma\sigma^2\right\} d\sigma^2 d\mu \\
C_1^{-1} &= \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1} exp\left\{-\gamma\sigma^2\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + \mu^2 - 2\mu\mu_0 + \mu_0^2}{2}\right)\right\} d\sigma^2 d\mu \\
C_1^{-1} &= \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2}\right)\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2 + \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)\sigma^2}\right\} d\sigma^2 d\mu \\
C_1^{-1} &= \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2\right\} d\sigma^2 d\mu
\end{aligned}
$$

Since μfollows a $N\left(\frac{\sum_{i=1}^{n} x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$(A.1). Hence,

$$C_1^{-1} = \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} \int_0^{\infty} (\sigma^2)^{\theta - \frac{n}{2} - 1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times d\sigma^2$$

Now, using the Bessel function of the third kind (A.5) with $v_1 = \theta - \frac{n}{2}$, we get,

$$C_1^{-1} = \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} \times 2 \times \left(\frac{\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}}{\gamma}\right)^{\frac{v_1}{2}} K_{v_1}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}\right)}\right)$$

$$\therefore C_1 = \frac{\left(\frac{2\pi}{n+1}\right)^{-\frac{1}{2}}}{2 \times \left(\frac{\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}}{\gamma}\right)^{\frac{v_1}{2}} K_{v_1}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}\right)}\right)}$$

Therefore, the marginal distribution of μ is

$$q(\mu|\underline{x}) = C_1\left(\frac{\left(\frac{\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}{\gamma}\right)^{\frac{v_2}{2}} K_{v_2}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}\right) \tag{10}$$

By using the Bessel function of the third kind (A.5), certain conditions need to be placed on the parameters. For this $q(\mu|\underline{x})$ to hold, $\gamma > 0$,$\left(\frac{\sum_{i=1}^{n} x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^{n} x_i + \mu_0\right)^2}{2(n+1)}\right) > 0$ and $\left(\frac{\sum_{i=1}^{n} x_i^2 - 2\mu\sum_{i=1}^{n} x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right) > 0$. See theorem (A.11) for the marginal distribution.

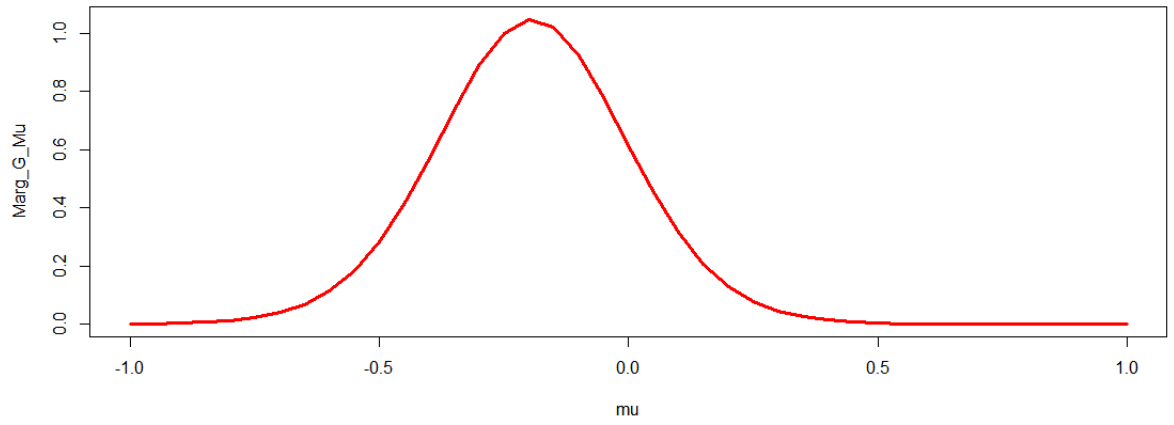


Figure 5: Marginal distribution of μ for the gamma prior with $\theta = 2$, $\gamma = 2$, $n = 20$ and $\mu_0 = \bar{x}$.

Marginal posterior distribution of σ^2

The marginal distribution of σ^2 is obtained from (9) as follows

$$
\begin{aligned}
q(\sigma^2|\underline{x}) &= \int_{-\infty}^{\infty} q(\mu,\sigma^2|\underline{x})d\mu \\
&\propto \int_{-\infty}^{\infty} (\sigma^2)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{\sigma^2}\right) - \gamma\sigma^2\right\} d\mu \\
&= (\sigma^2)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{\gamma\sigma^2\right\} \int_{-\infty}^{\infty} exp\left\{-\frac{1}{2}\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + \mu^2 - 2\mu\mu_0 + \mu_0^2}{\sigma^2}\right)\right\} d\mu \\
&= (\sigma^2)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{\gamma\sigma^2\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{1}{2\sigma^2}\left(\mu^2(n+1) - 2\mu\left(\sum_{i=1}^n x_i + \mu_0\right) + \sum_{i=1}^n x_i^2 + \mu_0^2\right)\right\} d\mu \\
&= J\left(\sigma^2\right)^{\theta-\frac{n}{2}-\frac{3}{2}} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times \\
&\quad \int_{-\infty}^{\infty} exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2\right\} d\mu \\
&\quad \text{Since } \mu \text{follows a } N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right) \text{(A.1). Hence} \\
q(\sigma^2|\underline{x}) &\propto \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\}
\end{aligned}
$$

For $q(\sigma^2|\underline{x})$ to be a valid density function, $\int_0^\infty q(\sigma^2|\underline{x})d\sigma^2 = 1$. Therefore,

$$
\begin{aligned}
1 &= \int_0^{\infty} C_2 \times (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
\therefore C_2^{-1} &= \int_0^{\infty} (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2
\end{aligned}
$$

Now, using the Bessel function of the third kind (A.5) with $v_1 = \theta - \frac{n}{2}$, we get,

$$
\begin{aligned}
C_2^{-1} &= 2\left(\frac{\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}}{\gamma}\right)^{\frac{v_1}{2}} K_{v_1}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)}\right) \\
\therefore C_2 &= \left[2\left(\frac{\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}}{\gamma}\right)^{\frac{v_1}{2}} K_{v_1}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)}\right)\right]^{-1}
\end{aligned}
$$

Therefore, the marginal distribution of σ^2 is

$$q(\sigma^2|\underline{x}) = C_2 \times (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} \tag{11}$$

By using the Bessel function of the third kind (A.5), certain conditions need to be placed on the parameters. For this $q(\sigma^2|\underline{x})$ to hold, $\gamma > 0$ and $\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right) > 0$. See theorem (A.11) for the marginal distribution.

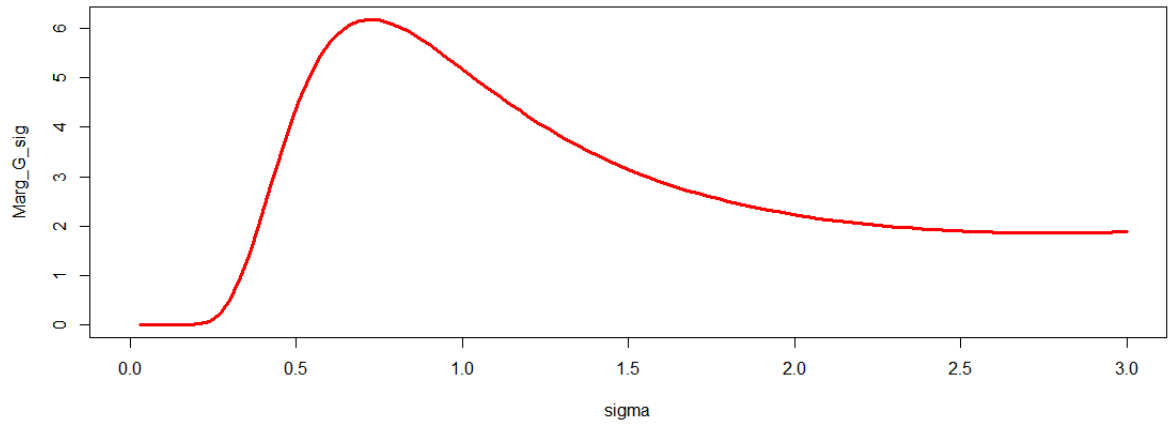


Figure 6: Marginal distribution of σ^2 for the gamma prior with $\theta = 2$, $\gamma = 2$, $n = 20$ and $\mu_0 = \bar{x}$.

4.5 Risk function and estimators

4.5.1 Estimator of μ

Loss function

Using the LINEX loss function (A.12) where β is the estimator for μ

$$L(\mu, \beta) = b\left[exp\left\{a\left(\beta-\mu\right)\right\} - a\left(\beta-\mu\right) - 1\right]$$

where $b > 0$ and $a \neq 0$.

Risk function

The risk function (A.13) is the expected value (A.6) of the loss function with respect to the marginal distribution of μ

$$\text{Let } K = \left(\frac{\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}{\gamma}\right)^{\frac{v_2}{2}} K_{v_2}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2 - 2\mu\sum_{i=1}^n x_i + n\mu^2 + (\mu-\mu_0)^2}{2}\right)}\right), \text{ then}$$

$$\begin{aligned} R(\mu,\beta) &= \mathbb{E}_\mu[L(\mu,\beta)] \\ &= \int_{-\infty}^{\infty} L(\mu,\beta) q(\mu|\underline{x}) d\mu \\ &= \int_{-\infty}^{\infty} b\left[exp\left\{a\left(\beta-\mu\right)\right\} - a\left(\beta-\mu\right) - 1\right] \times C_1 \times K \, d\mu \end{aligned}$$

$$
\begin{aligned}
& -\int_{-\infty}^{\infty} ba\left(\beta-\mu\right)C_1\times K\,d\mu-\int_{-\infty}^{\infty} bC_1\times K\,d\mu \\
= & \int_{-\infty}^{\infty} bexp\left\{a\beta\right\}exp\left\{-a\mu\right\}C_1\int_{0}^{\infty}(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}\times \\
& exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n}x_i^2-2\mu\sum_{i=1}^{n}x_i+n\mu^2+(\mu-\mu_0)^2}{2}\right)-\gamma\sigma^2\right\}d\sigma^2 d\mu \\
& -\int_{-\infty}^{\infty} ba\beta C_1\times K\,d\mu+\int_{-\infty}^{\infty} ba\mu C_1\times K\,d\mu-\int_{-\infty}^{\infty} bC_1\times K\,d\mu \\
= & \int_{0}^{\infty} bC_1 exp\left\{a\beta\right\}exp\left\{-\gamma\sigma^2\right\}(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}\int_{-\infty}^{\infty} exp\{-a\mu\}\times \\
& exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n}x_i^2-2\mu\sum_{i=1}^{n}x_i+n\mu^2+\mu^2-2\mu\mu_0+\mu_0^2}{2}\right)\right\}d\mu d\sigma^2 \\
& -ba\beta\int_{-\infty}^{\infty} C_1\times K\,d\mu+\int_{-\infty}^{\infty} ba\mu C_1\int_{0}^{\infty}(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}\times \\
& exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n}x_i^2-2\mu\sum_{i=1}^{n}x_i+n\mu^2+(\mu-\mu_0)^2}{2}\right)-\gamma\sigma^2\right\}d\sigma^2 d\mu \\
& -b\int_{-\infty}^{\infty} C_1\times K\,d\mu \\
= & \int_{0}^{\infty} bC_1 exp\left\{a\beta\right\}exp\left\{-\gamma\sigma^2\right\}(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}\int_{-\infty}^{\infty} exp\{-a\mu\}\times \\
& exp\left\{-\frac{1}{2\sigma^2}\left(\mu^2(n+1)-2\mu\left(\sum_{i=1}^{n}x_i+\mu_0\right)+\sum_{i=1}^{n}x_i^2+\mu_0^2\right)\right\}d\mu d\sigma^2 \\
& -ba\beta+\int_{0}^{\infty} baC_1(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}exp\left\{-\gamma\sigma^2\right\}\int_{-\infty}^{\infty}\mu\times \\
& exp\left\{-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n}x_i^2-2\mu\sum_{i=1}^{n}x_i+n\mu^2+\mu^2-2\mu\mu_0+\mu_0^2}{2}\right)\right\}d\mu d\sigma^2-b \\
= & \int_{0}^{\infty} bC_1 exp\left\{a\beta\right\}exp\left\{-\gamma\sigma^2-\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2\sigma^2}\right\}(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}\int_{-\infty}^{\infty} exp\{-a\mu\}\times \\
& exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu-\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)^2+\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)\sigma^2}\right\}d\mu d\sigma^2 \\
& -ba\beta+\int_{0}^{\infty} baC_1(\sigma^2)^{\theta-\frac{n}{2}-\frac{1}{2}-1}exp\left\{-\gamma\sigma^2\right\}\int_{-\infty}^{\infty}\mu\times \\
& exp\left\{-\frac{1}{2\sigma^2}\left(\mu^2(n+1)-2\mu\left(\sum_{i=1}^{n}x_i+\mu_0\right)+\sum_{i=1}^{n}x_i^2+\mu_0^2\right\}d\mu d\sigma^2-b\right.
\end{aligned}
$$

$$= \int_0^\infty \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} bC_1 exp\left\{a\beta\right\} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} (\sigma^2)^{\theta - \frac{n}{2} - 1}$$

$$\times \int_{-\infty}^\infty exp\{-a\mu\} \left(\frac{2\pi}{n+1}\right)^{-\frac{1}{2}} (\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2\right\} d\mu d\sigma^2$$

$$-ba\beta + \int_0^\infty baC_1 (\sigma^2)^{\theta - \frac{n}{2} - \frac{1}{2} - 1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2}\right)\right\} \int_{-\infty}^\infty \mu \times$$

$$exp\left\{-\frac{n+1}{2\sigma^2}\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2 + \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)\sigma^2}\right\} d\mu d\sigma^2 - b$$

Since μfollows a $N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$(A.1), we get

$$R(\mu,\beta) = \int_0^\infty \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} bC_1 exp\left\{a\beta\right\} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\}$$

$$(\sigma^2)^{\theta - \frac{n}{2} - 1} \times M_\mu\left(-a\right) d\sigma^2 - ba\beta$$

$$+ \int_0^\infty baC_1 \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} (\sigma^2)^{\theta - \frac{n}{2} - 1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times$$

$$\int_{-\infty}^\infty \mu \left(\frac{2\pi\sigma^2}{n+1}\right)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{\left(\mu - \frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)^2}{\frac{\sigma^2}{n+1}}\right\} d\mu d\sigma^2 - b$$

where $M_\mu(-a)$is the moment-generating (A.7) function of $\mu \sim N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$(A.1)

Since μfollows a $N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$(A.1)

$$R(\mu,\beta) = \int_0^\infty \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} bC_1 exp\left\{a\beta\right\} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times$$

$$(\sigma^2)^{\theta - \frac{n}{2} - 1} exp\left\{-a\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right) + \frac{1}{2}\left(\frac{\sigma^2}{n+1}\right)k^2\right\} d\sigma^2 - ba\beta$$

$$+ \int_0^\infty baC_1 \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} (\sigma^2)^{\theta - \frac{n}{2} - 1} exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} \times$$

$$\mathbb{E}_\mu[\mu|\underline{x}] d\sigma^2 - b$$

where $\mathbb{E}_\mu[\mu|\underline{x}]$is the expected value (A.6) of $\mu \sim N\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}, \frac{\sigma^2}{n+1}\right)$(A.1)

$$R(\mu,\beta) = \int_0^\infty \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} bC_1 exp\left\{a\beta - a\left(\frac{\sum_{i=1}^n x_i + \mu_0}{n+1}\right)\right\} \times$$

$$exp\left\{-\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2 + \mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i + \mu_0\right)^2}{2(n+1)}\right)\right\} (\sigma^2)^{\theta - \frac{n}{2} - 1} \times$$

$$exp\left\{\frac{1}{2}\left(\frac{\sigma^2}{n+1}\right)a^2\right\} d\sigma^2 - ba\beta$$

$$+\int_0^\infty baC_1\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}(\sigma^2)^{\theta-\frac{n}{2}-1}exp\left\{-\gamma\sigma^2-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\}\times$$
$$\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)d\sigma^2-b$$

$$R(\mu,\beta) = \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}bC_1exp\left\{a\beta-a\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\right\}\times$$
$$\int_0^\infty(\sigma^2)^{\theta-\frac{n}{2}-1}exp\left\{-\sigma^2\left(\gamma-\frac{a^2}{2(n+1)}\right)-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\}d\sigma^2$$
$$-ba\beta+baC_1\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\int_0^\infty(\sigma^2)^{\theta-\frac{n}{2}-1}\times$$
$$exp\left\{-\gamma\sigma^2-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\}d\sigma^2-b$$

Using the Bessel function of the third kind (A.5) with $v_1=\theta-\frac{n}{2}$, we get

$$R(\mu,\beta) = \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}bC_1exp\left\{a\beta-a\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\right\}\times 2\times\left(\frac{\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}}{\gamma-\frac{a^2}{2(n+1)}}\right)^{\frac{v_1}{2}}$$
$$K_{v_1}\left(2\sqrt{\left(\gamma-\frac{a^2}{2(n+1)}\right)\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)$$
$$-ba\beta+baC_1\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\times 2\times\left(\frac{\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)\sigma^2}}{\gamma}\right)^{\frac{v_1}{2}}$$
$$K_{v_1}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)\sigma^2}\right)}\right)-b$$

By using the Bessel function of the third kind (A.5), certain conditions need to be placed on the parameters. For this $R(\mu,\beta)$ to hold, $\gamma>0$, $\left(\gamma-\frac{a^2}{2(n+1)}\right)>0$ and
$\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)>0$.

Estimator

By differentiating $R(\mu,\beta)$ and setting the derivative equal to zero a estimator $\hat{\beta}$ of μ can be obtained such that $\hat{\beta}$ minimizes $R(\mu,\beta)$.

$$\frac{d}{d\beta}R(\mu,\beta) = ab\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}}C_1exp\left\{a\beta-a\left(\frac{\sum_{i=1}^n x_i+\mu_0}{n+1}\right)\right\}\times 2\times\left(\frac{\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}}{\gamma-\frac{a^2}{2(n+1)}}\right)^{\frac{v_1}{2}}$$
$$K_{v_1}\left(2\sqrt{\left(\gamma-\frac{a^2}{2(n+1)}\right)\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)$$

$$-bk$$

Setting the $\frac{d}{d\beta}R(\mu,\beta)=0$ and solving for β.

Let $\kappa = 2\times\left(\frac{\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{(\sum_{i=1}^{n}x_i+\mu_0)^2}{2(n+1)}}{\gamma-\frac{a^2}{2(n+1)}}\right)^{\frac{v_1}{2}} K_{v_1}\left(2\sqrt{\left(\gamma-\frac{a^2}{2(n+1)}\right)\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)$

$$\begin{aligned}
\frac{d}{d\beta}R(\mu,\hat{\beta}) &= 0\\
ba &= ab\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{a\hat{\beta}\right\} exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\\
1 &= \left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{a\hat{\beta}\right\} exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\\
exp\left\{a\hat{\beta}\right\} &= \left[\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\right]^{-1}\\
a\hat{\beta} &= ln\left[\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\right]^{-1}\\
\hat{\beta} &= -\frac{1}{a}ln\left[\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\right]
\end{aligned}$$

Therefore the estimator of μ is given by

$$\hat{\mu} = ln\left[\left(\frac{2\pi}{n+1}\right)^{\frac{1}{2}} C_1 exp\left\{-a\left(\frac{\sum_{i=1}^{n}x_i+\mu_0}{n+1}\right)\right\}\times\kappa\right]^{-\frac{1}{a}} \tag{12}$$

4.5.2 Estimator of σ^2

Loss function

Using the LINEX loss function (A.12) where α is the estimator for σ^2

$$L(\sigma^2,\alpha) = b\left[exp\left\{a\left(\alpha-\sigma^2\right)\right\}-a\left(\alpha-\sigma^2\right)-1\right]$$

where $b>0$ and $a\neq 0$.

Risk function

The risk function (A.13) is the expected value (A.6) of the loss function with respect to the marginal distribution of σ^2

$$\begin{aligned}
R(\sigma^2,\alpha) &= \mathbb{E}_{\sigma^2}[L(\sigma^2,\alpha)]\\
&= \int_0^\infty L(\sigma^2,\alpha)q(\sigma^2|\underline{x})d\sigma^2\\
&= \int_0^\infty b\left[exp\left\{a\left(\alpha-\sigma^2\right)\right\}-a\left(\alpha-\sigma^2\right)-1\right]\times C_2\times(\sigma^2)^{\theta-\frac{n}{2}-1}\times\\
&\quad exp\left\{\gamma\sigma^2-\frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)\right\}d\sigma^2
\end{aligned}$$

$$
\begin{aligned}
&= \int_0^\infty b exp\left\{a\alpha - a\sigma^2\right\} \times C_2(\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty ba(\alpha-\sigma^2) \times C_2(\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty b \times C_2 \times (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&= bC_2 exp\left\{a\alpha\right\} \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{-\sigma^2(\gamma+a) - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty ba\alpha \times C_2(\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&\quad + \int_0^\infty ba\sigma^2 \times C_2(\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 \\
&\quad - \int_0^\infty b \times C_2(\sigma^2)^{\theta-\frac{n}{2}-1} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2
\end{aligned}
$$

Using the Bessel function of the third kind (A.5) with $v_1 = \theta - \frac{n}{2}$, we get

$$
\begin{aligned}
R(\sigma^2,\alpha) &= bC_2 exp\left\{a\alpha\right\} \times 2 \left(\frac{\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}{\gamma+a}\right)^{\frac{v_1}{2}} \\
&\quad K_{v_1}\left(2\sqrt{(\gamma+a)\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}\right) \\
&\quad -ba\alpha + baC_2 \int_0^\infty (\sigma^2)^{\theta-\frac{n}{2}} exp\left\{\gamma\sigma^2 - \frac{1}{\sigma^2}\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)\right\} d\sigma^2 - b
\end{aligned}
$$

Using the Bessel function of the third kind (A.5) with $v_3 = \theta - \frac{n}{2} + 1$, we get

$$
\begin{aligned}
R(\sigma^2,\alpha) &= bC_2 exp\left\{a\alpha\right\} \times 2 \times \left(\frac{\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}{\gamma+a}\right)^{\frac{v_1}{2}} \\
&\quad K_{v_1}\left(2\sqrt{(\gamma+a)\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}\right) \\
&\quad -ba\alpha + baC_2 \times 2 \times \left(\frac{\left(\frac{\sum_{i=1}^n x_i^2+\mu_0^2}{2} - \frac{\left(\sum_{i=1}^n x_i+\mu_0\right)^2}{2(n+1)}\right)}{\gamma}\right)^{\frac{v_3}{2}}
\end{aligned}
$$

$$K_{v_3}\left(2\sqrt{\gamma\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)-b$$

By using the Bessel function of the third kind (A.5), certain conditions need to be placed on the parameters. For this $R(\sigma^2,\theta)$ to hold, $\gamma+a>0$, $\gamma>0$ and
$\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)>0$.

Estimator

By differentiating $R(\sigma^2,\alpha)$ and setting the derivative equal to zero a estimator $\hat{\alpha}$ of σ^2 can be obtained such that $\hat{\alpha}$ minimizes $R(\sigma^2,\alpha)$.

$$\begin{aligned}\frac{d}{d\alpha}R(\sigma^2,\alpha) &= baC_2exp\{a\alpha\}\times 2\times\left(\frac{\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}{\gamma+a}\right)^{\frac{v_1}{2}}\\ &\quad K_{v_1}\left(2\sqrt{(\gamma+a)\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)\\ &\quad -ba\end{aligned}$$

Setting the $\frac{d}{d\alpha}R(\sigma^2,\alpha)=0$ and solving for α.
Let $\tau=2\times\left(\frac{\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}{\gamma+a}\right)^{\frac{v_1}{2}}K_{v_1}\left(2\sqrt{(\gamma+a)\left(\frac{\sum_{i=1}^{n}x_i^2+\mu_0^2}{2}-\frac{\left(\sum_{i=1}^{n}x_i+\mu_0\right)^2}{2(n+1)}\right)}\right)$ then,

$$\begin{aligned}\frac{d}{d\alpha}R(\sigma^2,\alpha) &= 0\\ ba &= baC_2exp\{a\hat{\alpha}\}\times\tau\\ 1 &= C_2exp\{a\hat{\alpha}\}\times\tau\\ exp\{a\hat{\alpha}\} &= \frac{1}{C_2\times\tau}\\ exp\{a\hat{\alpha}\} &= [C_2\times\tau]^{-1}\\ a\hat{\alpha} &= ln\,[C_2\times\tau]^{-1}\\ \hat{\alpha} &= -\frac{1}{a}ln\,[C_2\times\tau]\end{aligned}$$

Therefore the estimator of σ^2 is given by

$$\hat{\sigma^2}=ln\,[C_2\times\tau]^{-\frac{1}{a}} \tag{13}$$

5 Simulation study

A simulation study based on a random sample of size 20 from a normal distribution, with parameters $\mu=0$ and $\sigma^2=1$, was performed. Estimates for the parameters μ and σ^2 were then calculated for the two priors, normal - inverse gamma ($\mu|\sigma^2\sim N(\bar{x},\sigma^2)$ and $\sigma^2\sim IG(4,3)$) and normal - gamma ($\mu|\sigma^2\sim N(\bar{x},\sigma^2)$ and $\sigma^2\sim G(2,2)$) using the equations (4) and (5) in section 3 and (12) and (13) in section 4. Since the moment generating function of the Student's t-distribution in the point $-a$ is unknown in equation (4), an unbiased

estimator [10] for the expression was used in the computations. Note that the prior parameters are chosen such that the expected prior value is equal to the true parameter value. The following results were obtained:

	Estimates		True Value	
	$\hat{\mu}$	$\hat{\sigma^2}$	Mean	Variance
Bayesian estimates using a normal-inverse gamma prior	-0.53998594	0.81737401	0	1
Bayesian estimates using a normal-gamma prior	0.00567354	1.01012104	0	1
Frequentist Estimates	0.0253235	1.0340205	0	1

Table 1: Results for the estimates of μ and σ^2 using the two priors with a LINEX loss function with parameters $a = 1$ and $b = 1$.

From table 1, the estimates for σ^2 for both priors yield results close to the true parameter value and are, thus, good estimates. However, the gamma prior estimates for μ and σ^2 provides better estimates than the inverse gamma prior since the estimated values are closer to the true parameter values. Thus, the normal-gamma-prior would be a more suitable choice of prior. The results of the frequentist approach yield estimates close to that of the Bayesian approach.

6 Conclusion

This study focused on subjective Bayesian analysis using the univariate normal distribution as the underlying distribution. It was assumed that both the mean and variance parameters of the normal distribution are unknown. The normal-inverse gamma joint prior and the normal-gamma joint prior were considered and the loss function was assumed to be a LINEX loss function. The posterior distributions, marginal distributions, risk functions and estimators were derived. The superiority of the normal-gamma prior is evident from simulation study.

References

[1] L.J. Bain and M. Engelhardt. *Introduction to Probability and Mathematical Statistics*. Duxbury Classic. Duxbury/Thomson Learning, 1992.

[2] H Bateman and A Erdelyi. *Higher Transcendental Functions. Bessel Functions, Parabolic Cylinder Functions, Orthogonal Polynomials*. McGraw-Hill New York, 1953.

[3] J.K. Ghosh, M. Delampady, and T. Samanta. *An Introduction to Bayesian Analysis: Theory and Methods*. Springer Science & Business Media, 2007.

[4] M.H.J. Gruber. *Regression Estimators: A Comparative Study*. JHU Press, 2010.

[5] Z. Hoque, J. Wesolowski, and S. Khan. Observed data based estimator and both observed data and prior information based estimator under asymmetric losses. *American Journal of Mathematical and Management Sciences*, 27(1-2):93–110, 2007.

[6] Kevin P Murphy. Conjugate bayesian analysis of the gaussian distribution. 2007.

[7] S.J. Press. *Subjective and Objective Bayesian Statistics: Principles, Models and Applications*, volume 590. John Wiley & Sons, 2009.

[8] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.

[9] F.J. Samaniego. *A Comparison of the Bayesian and Frequentist Approaches to Estimation*. Springer Science & Business Media, 2010.

[10] Janet Van Niekerk and Andriette Bekker. Bayesian estimators of the location parameter of the normal distribution with unknown variance. In *South African Statistical Journal Proceedings: Proceedings of the 54th Annual Conference of the South African Statistical Association: Congress 1*, pages 10–17, 2012.

[11] H.R. Varian. *A Bayesian Approach to Real Estate Assessment*, pages 195–208. North Holland: Amsterdam, 1975.

[12] A. Zellner. Bayesian estimation and prediction using asymmetric loss functions. *Journal of the American Statistical Association*, 81(394):446–451, 1986.

Glossary

A.1 [1] A random variable X follows a normal distribution with mean μ and variance σ^2 if it has the probability density function

$$f(x) = (2\pi\sigma^2)^{-\frac{1}{2}} exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}$$

for $-\infty < x < \infty$, where $-\infty < \mu < \infty$ and $\sigma^2 > 0$. This is denoted by X $\sim$N(μ, σ^2).
The moment generating function of X $\sim$N(μ, σ^2) is

$$M_x(t) = exp\left\{\mu t + \frac{\sigma^2 t^2}{2}\right\}$$

A.2 [3] A random variable X follows a Inverse Gamma distribution with parameters α and β if it has the probability density function

$$f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} exp\left(-\frac{\beta}{x}\right)$$

where $x > 0$. This is denoted by X $\sim$IG(α, β).

A.3 [3] A random variable X follows a Gamma distribution with parameters α and λ if it has the probability density function

$$f(x) = \frac{\lambda^\alpha}{\Gamma(\alpha)} x^{\alpha-1} exp\left(-\lambda x\right)$$

where $x > 0$, $alpha > 0$ and $\lambda > 0$. This is denoted by X $\sim$G(α, λ).

A.4 [1] A random variable X follows a non-central t distribution with v degrees of freedom and non-centrality parameter δ if it has the probability density function

$$f(x) = \frac{\Gamma\left(\frac{v+1}{2}\right)}{\Gamma(\frac{v}{2})} \frac{1}{\sigma\sqrt{v\pi}} \left[1 + \frac{(x-\delta)^2}{v\sigma^2}\right]^{\frac{-(v+1)}{2}}$$

where $-\infty < x < \infty$, $v > 0$,$-\infty < \delta < \infty$ and $\sigma^2 > 0$ .This is denoted by X $\sim t\ (v, \delta, \sigma^2)$.
If X$\sim t\ (v, \delta, \sigma^2)$ distribution then the expected value of X is δ if $v > 1$.

A.5 [2] Let $K_v(\cdot)$ be the Bessel Function of the third kind then for Re$(\beta) > 0$, Re$(\gamma) > 0$

$$\int_0^\infty x^{v-1} exp\left(-\frac{\beta}{x} - \gamma x\right) dx = 2\left(\frac{\beta}{\gamma}\right)^{\frac{v}{2}} K_v\left(2\sqrt{\beta\gamma}\right)$$

A.6 [1] If X is a continuous random variable with probability density function $f(x)$ then the expected value of X is defined by

$$\mathbb{E}(x) = \int_{-\infty}^{\infty} x f(x) dx$$

A.7 [1] If X is a random variable then the expected value

$$M_x(t) = \mathbb{E}(e^{tX}) = \int_{-\infty}^{\infty} e^{tx} f(x)dx$$

is called the moment generating function of X.

A.8 [1] The joint density function of n random variables $X_1, ..., X_n$ evaluated at $x_1, ..., x_n$, say $f(x_1, ..., x_n; \theta)$ is referred to as a likelihood function. For fixed $x_1, ..., x_n$ the likelihood function is a function of θ and is denoted by $L(\theta)$. If $X_1, ..., X_n$ represent a sample from $f(x; \theta)$, then

$$L(\theta) = f(x_1; \theta)...f(x_n; \theta) = \prod_{i=1}^{n} f(x_i; \theta)$$

A.9 [1] If X_1 and X_2 are random variables with joint probability density function $f(x_1, x_2)$, then the conditional probability density function of X_2 given $X_1 = x_1$ is defined to be

$$f(x_2|x_1) = \frac{f(x_1, x_2)}{f(x_2)}$$

for values of x_1 such that $f(x_1) > 0$.

A.10 [1] The conditional density of θ given the sample observations $x = (x_1, ..., x_n)$ is called the posterior density and is given by

$$f_{\theta|x}(\theta) = \frac{f(x_1, ..., x_n|\theta)p(\theta)}{\int f(x_1, ..., x_n|\theta)p(\theta)d\theta}$$

where $p(\theta)$ is the prior density for the parameter θ.

A.11 [1] If the pair (X_1, X_2) of continuous random variables has the joint probability density function $f(x_1, x_2)$, then the marginal probability density function of $X_1$is

$$f_1(x_1) = \int_{-\infty}^{\infty} f(x_1, x_2)dx_2$$

A.12 [4] Let $\hat{\theta}$ be an estimator of a parameter θ. The LINEX loss function is defined by

$$L(\theta, \hat{\theta}) = b\left[e^{a(\hat{\theta}-\theta)} - a(\hat{\theta} - \theta) - 1\right]$$

where $a \neq 0$ and $b > 0$.

A.13 [1] The risk function is defined to be the expected loss,

$$R(\theta, \hat{\theta}) = \mathbb{E}_\theta[L(\theta, \hat{\theta})]$$

Appendix

The LINEX loss function graphs for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA

The SAS software code used for the LINEX loss function:

```
/* ---------------------------------------- */
        /* Effect of a, constant b */
/* ---------------------------------------- */
data linex_b;
b = 5;
a1 = -1.5;
a2 = 1.5;
a3 = -1.75;
a4 = 1.75;
a5 = 0.445;

do delta = -1 to 1 by 0.005;
        L1 = b*(exp(a1*delta) - a1*delta -1);
        L2 = b*(exp(a2*delta) - a2*delta -1);
        L3 = b*(exp(a3*delta) - a3*delta -1);
        L4 = b*(exp(a4*delta) - a4*delta -1);
        L5 = b*(exp(a5*delta) - a5*delta -1);
        output;
end;
run;

goptions reset=all;
axis1 label = (angle=90 'Loss');
axis2 label = ('Delta');
legend1 value=('a=-1.5' 'a=1.5' 'a=-1.75' 'a=1.75' 'a=0.445');
symbol1 color=black i=join w = 5;
symbol2 color=red i=join w = 5;
symbol3 color=blue i=join w = 5;
symbol4 color=green i=join w = 5;
symbol5 color=orange i=join w = 5;
title1 'Linex loss function for constant b';
proc gplot data=linex_b;
plot (L1 L2 L3 L4 L5)*delta /overlay legend=legend1 vaxis=axis1 haxis=axis2 ;
run;

/* ---------------------------------------- */
        /* Effect of b, a > 0 */
/* ---------------------------------------- */
data linex_a1;
a = 2.75;
b1 = 0.354;
b2 = 4.6665;
b3 = 28;
b4 = 64.05;
b5 = 140;
```

```
do delta = -1 to 1 by 0.005;
        L1 = b1*(exp(a*delta) - a*delta -1);
        L2 = b2*(exp(a*delta) - a*delta -1);
        L3 = b3*(exp(a*delta) - a*delta -1);
        L4 = b4*(exp(a*delta) - a*delta -1);
        L5 = b5*(exp(a*delta) - a*delta -1);
        output;
end;
run;

goptions reset=all;
axis1 label = (angle=90 'Loss');
axis2 label = ('Delta');
legend1 value=('b=0.354' 'b=4.6665' 'b=28' 'b=64.05' 'b=140');
symbol1 color=black i=join w = 5;
symbol2 color=red i=join w = 5;
symbol3 color=blue i=join w = 5;
symbol4 color=green i=join w = 5;
symbol5 color=orange i=join w = 5;
title1 'Linex loss function for a > 0';
proc gplot data=linex_a1;
plot (L1 L2 L3 L4 L5)*delta /overlay legend=legend1 vaxis=axis1 haxis=axis2 ;
run;

/* ---------------------------------- */
        /* Effect of b, a < 0 */
/* ---------------------------------- */
data linex_a2;
a = -2.75;
b1 = 0.354;
b2 = 4.6665;
b3 = 28;
b4 = 64.05;
b5 = 140;

do delta = -1 to 1 by 0.005;
        L1 = b1*(exp(a*delta) - a*delta -1);
        L2 = b2*(exp(a*delta) - a*delta -1);
        L3 = b3*(exp(a*delta) - a*delta -1);
        L4 = b4*(exp(a*delta) - a*delta -1);
        L5 = b5*(exp(a*delta) - a*delta -1);
        output;
end;
run;

goptions reset=all;
axis1 label = (angle=90 'Loss');
axis2 label = ('Delta');
legend1 value=('b=0.354' 'b=4.6667' 'b=28' 'b=64.05' 'b=140');
symbol1 color=black i=join w = 5;
symbol2 color=red i=join w = 5;
symbol3 color=blue i=join w = 5;
```

```
symbol4 color=green i=join w = 5;
symbol5 color=orange i=join w = 5;
title1 'Linex loss function for a < 0';
proc gplot data=linex_a2;
plot (L1 L2 L3 L4 L5)*delta /overlay legend=legend1 vaxis=axis1 haxis=axis2 ;
run;
```

The Marginal distribution and application was performed in R [8].

Marginal density and application for parameter μ using the inverse gamma prior:

```
n = 20
mu = 0
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
alpha = 4
beta = 3
b = 1
a = 1

b1 = beta + (0.5)*(sum_x2 + mu0^2) - 0.5*((sum_x+mu0)^2/(n+1))
df = 2*alpha + n -2
delta = (sum_x + mu0)/(n+1)
par = b1/((df/2)*(n+1))

gam1 = gamma((df+1)/2)
gam2 = gamma(df/2)
pow = (-df-1)/2

Marg_IG_mu <- function(mu1){(gam1/gam2)*((par*df*pi)^(-0.5))*(1+((mu1-delta)^2)
                                                              /(df*par))^pow}
plot(Marg_IG_mu,from=-2,to=2,add=FALSE,col="dark blue",lwd=3,type="l",
     xlab="mu",ylab=NULL,xlim=NULL)

#Estimate of mu for inverse gamma prior
n = 20
mu = 0
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
alpha = 4
beta = 3
b = 1
```

```
a = 1

u = alpha + (n/2) - (1/2)
w = alpha + (n/2) - 1
b1 = beta + (0.5)*(sum_x2 + mu0^2) - 0.5*((sum_x+mu0)^2/(n+1))
gam1 = gamma(u)
gam2 = gamma(w)
R2 = (gam1/gam2)*(2*w*pi*(b1/(w*(n+1))))^(-1/2)

mgf = (1/n)*sum(exp((x-x_bar/sqrt((sample_var*(n-1))/n))*(-a)))

mu_hat_IG = (-1/a)*log(mgf, base = exp(1))
```

Marginal density and application for parameter σ^2 using the inverse gamma prior:

```
n = 20
mu = 0
sigma = 1
x = rnorm(n, mean=mu, sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
alpha = 4
beta = 3
b = 1
a = 1

par1 = alpha + (n/2)
par2 = beta + (sum_x2+mu0^2)/2 - ((sum_x+mu0)^2)/(2*(n+1))

Marg_IG_sig <- function(sigma1){1/gamma(par1)*(par2^par1)*sigma1^(-par1-1)
                                *exp(-par2/sigma1)}
plot(Marg_IG_sig, from=0, to=3, add=FALSE, col='dark blue', lwd=3, type="l",
     xlab="sigma", ylab=NULL, xlim=NULL)

#Estimate of sigma for inverse gamma prior
n = 20
mu = 0
sigma = 1
x = rnorm(n, mean=mu, sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
alpha = 4
beta = 3
b = 1
a = 1
```

```
v0 = -alpha - n/2
b1 = beta + (sum_x2+mu0^2)/2 - ((sum_x+mu0)^2)/(2*(n+1))
d = alpha + n/2
R1 = ((b1)^d)/gamma(d)

bes1 = 2*sqrt(a*b1)
Bessel_1=besselK(bes1, v0, expon.scaled = FALSE)
delta = Bessel_1*2*(b1/a)^(v0/2)

theta = R1*delta
sigma_hat_est = (-1/a)*log(theta,base = exp(1))
```

Marginal density and application for parameter μ using the gamma prior:

```
mu <- seq(-1,1,by=0.05)
n = 20
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
theta = 2
gam_par = 2
b = 1
a = 1

c0 = (2*pi/(n+1))^(-0.5)
v1 = theta - n/2
v2 = theta - (n/2) - (1/2)
b1 = ((sum_x2 + mu0^2)/2)-((sum_x+mu0)^2/(2*(n+1)))
b2 = (0.5)*(sum_x2 - 2*mu*sum_x + n*mu^2 + (mu - mu0)^2)
bes1 = 2*sqrt(b1*gam_par)
Bessel_1=besselK(bes1, v1, expon.scaled = FALSE)
bes2 = 2*sqrt(b2*gam_par)
Bessel_2 = besselK(bes2, v2, expon.scaled = FALSE)

C1 = c0/(2*(b1/gam_par)^(v1/2)*Bessel_1)

Marg_G_Mu= C1*(b2/gam_par)^(v2/2)*Bessel_2



plot(mu,Marg_G_Mu,col="red",lwd=3,type="l",xlab="mu",ylab=NULL,xlim=NULL)

#Estimate of mu for gamma prior
n = 20
mu = 0
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
```

```
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
theta = 2
gam_par = 2
b = 1
a = 1

c0p = (2*pi/(n+1))^(0.5)
c0m = (2*pi/(n+1))^(-0.5)

v1 = theta - n/2
b1 = ((sum_x2 + mu0^2)/2)-((sum_x+mu0)^2/(2*(n+1)))
b2 = gam_par - (a^2/(2*(n+1)))
bes1 = 2*sqrt(b1*gam_par)
Bessel_1=besselK(bes1, v1, expon.scaled = FALSE)
bes2 = 2*sqrt(b1*b2)
Bessel_2 = besselK(bes2, v1, expon.scaled = FALSE)

C1 = c0m/(2*(b1/gam_par)^(v1/2)*Bessel_1)
e = exp(-a*((sum_x+mu0)/(n+1)))
k = 2*(b1/b2)^(v1/2)*Bessel_2

beta = c0p*C1*e*k

mu_hat_est = (-1/a)*log(beta,base = exp(1))
```

Marginal density and application for parameter σ^2 using the gamma prior:

```
n = 20
mu = 0
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
theta = 2
gam_par = 2
b = 1
a = 1

v1 = theta - n/2
b1 = ((sum_x2 + mu0^2)/2)-((sum_x+mu0)^2/(2*(n+1)))
b2 = gam_par + a
bes1 = 2*sqrt(b1*gam_par)
Bessel_1=besselK(bes1, v1, expon.scaled = FALSE)
C2 = 1/(2*(b1/gam_par)^(v1/2)*Bessel_1)

Marg_G_sig <- function(sigma1){C2*(sigma1^(v1+1))*exp((gam_par*sigma1)
                                                      - (b1/sigma1))}
```

```
plot(Marg_G_sig,from=0,to=3,col = "red",lwd=3,add=FALSE,type="l",
     xlab="sigma",ylab=NULL,xlim=NULL)

#Estimate of sigma for gamma prior
n = 20
mu = 0
sigma = 1
x = rnorm(n,mean=mu,sd=sqrt(sigma))
x2 = x^2
sum_x = sum(x)
sum_x2 = sum(x2)
sample_var= var(x)
mu0 = mean(x)
x_bar = mean(x)
theta = 2
gam_par = 2
b = 1
a = 1

v1 = theta - n/2
b1 = ((sum_x2 + mu0^2)/2)-((sum_x+mu0)^2/(2*(n+1)))
b2 = gam_par + a
bes1 = 2*sqrt(b1*gam_par)
Bessel_1=besselK(bes1, v1, expon.scaled = FALSE)

C2 = 1/(2*(b1/gam_par)^(v1/2)*Bessel_1)

bes2 = 2*sqrt(b1*b2)
Bessel_2 = besselK(bes2, v1, expon.scaled = FALSE)

tau = 2*(b1/b2)^(v1/2)*Bessel_2

alpha =C2*tau
sigma_hat_est = (-1/a)*log(alpha,base = exp(1))
```

STAR time series models in finance and economics

Ané Neethling 11138948

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor: Dr. PJ van Staden

Department of Statistics, University of Pretoria

2 November 2015

Abstract

This report discusses various developments and extensions of the smooth transition autoregressive (STAR) time series model which is an extension of the threshold autoregressive (TAR) model. The focus will be on the representation of the STAR model associated with its different transition functions, as well as methods for various hypothesis tests, especially on tests against nonlinearity. Note that this report will assume only an AR(1) for simplicity in the theory discussion. Model specification, estimation and evaluation will be discussed accordingly, with the application aim to fit an LSTAR model to a financial/economic time series.

Nonlinear time series models have become more popular over the past years, especially in the economic and finance environment. Many times in the economic environment a change from one regime to another occurs which causes for a change in the economic behaviour. These regimes refer to the upswings and downswings in the economy. The STAR model is a nonlinear time series model which allows for a smooth transition between two regimes. For descriptive, evaluation and forecasting purposes, it is necessary to model the relevant (nonlinear) time series to a STAR model with a specified transition function. The various transition functions (first-order logistic, second-order logistic and exponential function) are discussed and mathematically illustrated in the theory discussion of this report.

This report will educate the reader exactly what the STAR model is and how it is used. It will explain the modelling cycle in the STAR framework, guiding the reader step by step with an empirical example on how to model data to a specified STAR model and evaluate the resulting model. The STAR model can then also be used for forecasting purposes (which will not be covered in this report).

It is important that research on STAR models is enhanced since we live in a world where everything is influenced by the economy and vice versa. Changes in regimes happen on a regular basis and the economic-behaviour adjusts accordingly. Thus, the demand for STAR modelling in the economic and finance environment is increasing steadily.

Declaration

I, *Ané Neethling,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_ _
Ané Neethling

_ _
Dr. Paul J. van Staden

_ _
Date

Acknowledgements

This research was supported by a South African Research Chair Initiative (SARChI) bursary awarded to the Department of Statistics at the University of Pretoria, as well as a Bureau for Statistical and Survey Methodology (STATOMET) bursary.

The guidance, hospitality and enthusiasm provided by supervisor Dr. Paul J. van Staden, University of Pretoria, Department of Statistics are greatly acknowledged and appreciated, as well as the support from a number of special friends and family. It motivated me all the way through.

Contents

List of Figures

List of Tables

1 Introduction

Interest in nonlinear time series models grew steadily since the past couple of years. Models which allow for regime-switching behaviour have been used on a more regular basis, especially in the application of finance or economic time series. One of such regime-switching models is the smooth transition autoregressive (STAR) time series model, which will be introduced and illustrated in this report. The aim of this report is to illustrate and apply the modelling cycle for STAR models.

In summary, STAR models can be observed at $t = 1-p,\ 1-(p-1), \ldots, -1, 0, 1, \ldots, T-1, T$ and are defined as

$$\begin{aligned} y_t &= \left(\phi_{1,0} + \phi_{1,1} y_{t-1} + \cdots + \phi_{1,p} y_{t-p}\right)\left(1 - G\left(s_t; \gamma, c\right)\right) \\ &\quad + \left(\phi_{2,0} + \phi_{2,1} y_{t-1} + \cdots + \phi_{2,p} y_{t-p}\right) G\left(s_t; \gamma, c\right) + \varepsilon_t \qquad for\ t = 1, \ldots, T \end{aligned} \tag{1}$$

or alternatively

$$y_t = \phi_1^{'} x_t \left[1 - G\left(s_t, \gamma, c\right)\right] + \phi_2^{'} x_t G\left(s_t, \gamma, c\right) + \varepsilon_t \tag{2}$$

where $x_t = \left(1, \tilde{x}_t^{'}\right)^{'}$ with $\tilde{x}_t = \left(y_{t-1}, \ldots, y_{t-p}\right)^{'}$ and $\phi_i = \left(\phi_{i,0}, \phi_{i,1}, \ldots, \phi_{i,p}\right)^{'}$ for $i = 1, 2$. It is assumed that the ε_i's are a martingale difference sequence related to the past observations in the specific time series up until $t = t-1$, denoted as $\Omega_{t-1} = \left\{y_{t-1}, y_{t-2}, \ldots, y_{1-(p-1)}, y_{1-p}\right\}$, thus $E\left(\varepsilon_t | \Omega_{t-1}\right) = 0$. It is also assumed that ε_t has a constant variance of σ^2.

What exactly is the STAR model and why are STAR models used? By looking at Eq. 1, one needs to realize that the STAR model is an extension of the TAR model, which is an extension of the AR model - thus, the STAR model can be defined as some type of autoregressive (AR) time series model. The AR model is a linear time series model which describes certain time-varying processes in various environments such as nature, economics, etc. The dependent Y_t (output) variable depends on its own previous values, that is Y_t depends on Y_{t-i}. The threshold autoregressive (TAR) model, discussed in [2], is used in nonlinear time series, when two or more regimes occur. According to [8] economic theory often includes the idea that there will be a change in economic behaviour if a specific variable changes/moves to another level. STAR models will allow for a higher level of flexibility in the model parameters, which in turn allows for a smooth transition between the two regimes. Teräsvirta [16] argues that the logistic STAR (LSTAR) model is a special case of the single-threshold TAR model, whereas the exponential STAR (ESTAR) model is a special case of the double-threshold TAR model.

Three different choices of transition functions are associated with STAR models, namely the logistic function, the exponential function and the second-order logistic function. These three transition functions result in different regime-switching behaviours. This report will give special attention to the logistic function (resulting in the LSTAR model).

The objective of [4] was to define and represent a specific extension of the TAR model, namely the STAR model. After representing the general STAR model, extensions of the STAR model were discussed in detail, namely LSTAR, ESTAR, etc. van Dijk [4] also explained the modelling cycle in the STAR framework and then used an application to illustrate this modelling cycle (US unemployment with STAR models). Several other applications on STAR models have proven the relevance of modelling certain transition functions in some specific environment. For example, the LSTAR model is more appropriate for modelling business cycles, where the regimes are associated with expansions and contractions in the business cycle. The ESTAR model, on the other hand, has especially been used in the application of real exchange rates. It is however important to use the correct transition function, thus the correct type of STAR model. In order to master this, one needs to consider the hypothesis testing for linearity and misspecification in smooth transition models, which was another focus point in [4].

2 Representation of the STAR model and its various transition functions

In the following sections we will discuss the STAR model considering properties of the AR(1) time series model. This will provide for simplification in order to assist the reader gaining a much better understanding of what exactly the STAR model is and how it is applied. Note that it is however straightforward to extend the theory discussed below to higher order AR(p) models.

The basic smooth transition autoregressive model for a specific time series y_t, where $t = 1-p, 1-(p-1), \ldots, -1, 0, 1, \ldots, T-1, T$, can be defined by the mathematical Eq. 1 introduced in Section 1. If one specifically considers the STAR model as an extension of the AR(1) model, Eq. 1 reduces to

$$y_t = (\phi_{1,0} + \phi_{1,1} y_{t-1})(1 - G(s_t; \gamma, c)) + (\phi_{2,0} + \phi_{2,1} y_{t-1}) G(s_t; \gamma, c) + \varepsilon_t \qquad for\, t = 1, \ldots, T. \tag{3}$$

$G(s_t; \gamma, c)$ is known as the transition function - a continuous function between the extreme values 0 and 1. When the transition function is equal to either one of the two extreme values, that is 0 and 1, it operates to create a smooth transition between two regimes . In this case, the STAR model can be interpreted as a regime-switching model. If the transition function is equal to different values between 0 and 1 for each regime respectively, the STAR model accepts for a "continuum" of regimes. In this report, however, we will only consider the first case, where the STAR model operates as a regime-switching model. Several popular choices exist for the STAR model's transition function, however, only three choices will be considered in this report.

The transition variable s_t can embrace several characteristics:

- Teräsvirta [16] assumes that s_t is a lagged endogenous variable, hence $s_t = y_{t-d}$, where $d > 0$.
- Other references [4] suggest that this assumption should not be made after all, hence s_t can also be used as some exogenous variable or as some function of lagged endogenous variables.
- Also, s_t can be considered as some linear time trend, thus $s_t = t$, resulting in a model with smoothly changing parameters (refer back to [12]).

One of the popular choices for the transition function would be the first-order logistic function

$$G(s_t; \gamma, c) = [1 + \exp\{-\gamma(s_t - c)\}]^{-1} \qquad for\, \gamma > 0 \tag{4}$$

where the obtained model is called the logistic STAR (LSTAR) model. In Eq. 4 c is the parameter that can be interpreted as the point at which the change occurs from the one regime to the other. Parameter γ influence the level of smoothness of the transition between the two regimes. As the parameter γ increases, the change between the two regimes occurs more frequently. From Eq. 3 and 4, when $\gamma = 0 \Rightarrow G(s_t; \gamma, c) = \frac{1}{2}$, then the LSTAR model transforms into a linear AR(1) model. The two regimes in the LSTAR model correspond to small and large values of the transition variable s_t, relative to c. This is especially useful for modelling business cycle asymmetry, where the regimes in the LSTAR model relate to various contractions and expansions in the economy.

Another popular choice for the transition function would be the exponential function

$$G(s_t; \gamma, c) = 1 - \exp\left\{-\gamma(s_t - c)^2\right\} \qquad for\, \gamma > 0. \tag{5}$$

The use of the exponential function leads to the exponential STAR (ESTAR) model and owns the characteristic that $G(s_t; \gamma, c) \to 1$ as both $s_t \to -\infty$ and $s_t \to \infty$, whereas $G(s_t; \gamma, c) = 0$ when $s_t = c$. As either $\gamma \to 0$ or $\gamma \to \infty$, the exponential function in Eq. 5 will transform into a linear model and the ESTAR model does not hold a self-existing threshold autoregressive (SETAR) model as a special case. If this is unfavorable, one can instead make use of the second-order logistic function

$$G(s_t; \gamma, c) = [1 + \exp\{-\gamma(s_t - c_1)(s_t - c_2)\}]^{-1} \qquad where\, c_1 \leq c_2,\ \gamma > 0. \tag{6}$$

From Eq. 6, as $\gamma \to 0$, the model will become linear. However, if $\gamma \to \infty$ and $c_1 \neq c_2$, the transition function will tend to 1 for $s_t < c_1$ and $s_t > c_2$, and 0 otherwise.

Finally, the transition functions in Eq. 4 and 6 result from the n-order (general) logistic function

$$G\left(s_t;\gamma,c\right)=\left[1+\exp\left\{-\gamma\prod_{i=1}^{n}\left(s_t-c_i\right)\right\}\right]^{-1}\qquad where\, c_1\leq c_2\leq\cdots\leq c_n,\ \gamma>0. \tag{7}$$

This general logistic function can be used to find several changes between only the two regimes.

3 Hypothesis testing in the context of STAR models

Before considering the modelling cycle for STAR models, one needs to be familiar with a number of hypothesis tests, and be able to apply them, in order to select an appropriate and relevant STAR model for a specific time series. Hypothesis tests in this section include tests of linearity against STAR, LSTAR and ESTAR nonlinearity. Tests for misspecification of STAR models include tests for the absence of error autocorrelation, no remaining nonlinearity as well as tests for constancy of parameters which will be introduced in the following section.

3.1 Testing linearity against STAR

The very first step in building STAR models, is to test linearity against STAR nonlinearity, that is to test whether the fitted model is nonlinear. The null hypothesis of linearity is defined so that the two autoregressive parameters in the two regimes of the model is equal to one another, that is $H_0:\phi_1=\phi_2$, while the alternative hypothesis can be defined as $H_a:\ \phi_{1,j}\neq\phi_{2,j}$ for at least one $j\in\{0,1\}$. Take note that we are discussing the STAR model as an extension of an AR(1) time series model, hence j will take on values of only 0 and 1. For a general STAR time series model as an extension from an AR(p) model, see detailed discussion in [4].

Another way to express the null hypothesis for linearity will be $H_0^{'}:\ \gamma=0$. If this alternative null hypothesis ($H_0^{'}$) is accepted, it will result in a linear model for any one of the three transition functions introduced previously. In the case of $H_0^{'}$ being used, the unknown parameters will be c, the location parameter, and the two autoregressive parameters in the two regimes, that is ϕ_1 and ϕ_2.

The presence of unknown parameters makes it problematic for testing linearity against STAR alternatives under the null hypothesis. A possible solution suggested by [13] is to replace the transition function by a proper Taylor series approximation. The resultant equation does not have the problem of unknown parameters anymore. Consequently, linearity can be tested by using the Lagrange Multiplier (LM) statistic, with a χ^2-distribution. This approach contains two key advantages. Firstly, there is no need to estimate the model under the alternative hypothesis, and secondly, one has usual asymptotic theory available which can be used to obtain critical values for the test statistics.

3.1.1 Testing linearity against LSTAR

Note that the LSTAR model can be rewritten from Eq. 2 and 4 as

$$y_t=\phi_1^{'}x_t+\left(\phi_2-\phi_1\right)^{'}x_tG\left(s_t;\gamma,c\right)+\varepsilon_t \tag{8}$$

and it can be assumed that $\{\varepsilon_t\}\sim n.i.d.\left(0,\sigma^2\right)$. In case it is wished to develop a linearity test against LSTAR nonlinearity (using Eq. 8), it is advised to approximate the transition function in Eq. 4 with a first-order Taylor approximation around $\gamma=0$. The auxiliary regression equation follows as a result

$$y_t=\beta_0^{'}x_t+\beta_1^{'}x_ts_t+e_t \tag{9}$$

with $\beta_i=\left(\beta_{i,0},\beta_{i,1}\right)^{'}$, $i=0,1$ and $e_t=\varepsilon_t+\left(\phi_2-\phi_1\right)^{'}x_tR_1\left(s_t,\gamma,c\right)$, where $R_1\left(s_t,\gamma,c\right)$ is what is left of the Taylor expansion. Under the null hypothesis $R_1\left(s_t,\gamma,c\right)\equiv0$, thus $e_t=\varepsilon_t$. Making use of the auxiliary regression Eq. 9, one can use a second alternative for a null hypothesis, such that $H_0^{''}:\ \beta_1=0$, to test for linearity against LSTAR nonlinearity. Notice when $\beta_1=0$, Eq. 9 reduces to a linear model. The test statistic is defined as $LM_1\sim\chi^2\left(p+1\right)$ under the null hypothesis for linearity, where $p=1$ (referring back to the AR(1) model).

Note that when $s_t = y_{t-d}$ for $1 \leq d \leq p$, it is best to remove $\beta_{1,0}s_t$ from Eq. 9 in order to avoid multi-collinearity. This should be done for all tests discussed below. When this is the case for the transition variable (s_t), the test statistic LM_1 is powerless when only the intercept changes across the two regimes, that is when $\phi_{1,0} \neq \phi_{2,0}$ but $\phi_{1,1} = \phi_{2,1}$. One can estimate the transition function by using a third-order Taylor approximation in order to achieve a test which is powerless against this alternative. By doing this, the resulting auxiliary regression equation follows

$$y_t = \beta_0^{'} x_t + \beta_1^{'} x_t s_t + \beta_2^{'} x_t s_t^2 + \beta_3^{'} x_t s_t^3 + e_t \tag{10}$$

with $e_t = \varepsilon_t + (\phi_2 - \phi_1)^{'} x_t R_3 (s_t, \gamma, c)$, and $\beta_{0,0}$ and $\beta_i = 1, 2, 3$ are functions of the parameters ϕ_1, ϕ_2, γ and c. The consequent null hypothesis is $H_0^{''} : \beta_1 = \beta_2 = \beta_3 = 0$ with test statistic $LM_3 \sim \chi^2 (3(p+1))$ under the null hypothesis for linearity.

Testing algorithm for LSTAR

Note that when working with small samples, it will be best to use the F version of the LM test statistics, since it has more accurate size properties than what the χ^2 version has. Both these versions can be calculated using two linear regressions as illustrated below.

Suppose the LM_3 test statistic is to be calculated under the null hypothesis of linearity. Note that the LM_3 test statistic is based on the auxiliary regression Eq. 10. The following steps described below is needed for calculations;

Step 1 Using the time series data at hand, fit the model under the null hypothesis of linearity by regressing the dependent variable, y_t, onto the independent variable, x_t. Determine the residuals, $\hat{\varepsilon}_t$, as well as the sum of the squared residuals, $SSR_0 = \sum_{t=1}^{T} \hat{\varepsilon}_t^2$.

Step 2 Fit the auxiliary regression Eq. 10 of y_t on x_t as well as $x_t s_t^i$, for $i = 1, 2, 3$. Calculate the residuals, $\hat{e}_t$, as well as the sum of the squared residuals, $SSR_1 = \sum_{t=1}^{T} \hat{e}_t^2$.

Step 3 Finally, using the results obtained from step 1 and 2, one can calculate the χ^2 version of the LM_3 test statistic using

$$LM_3 = \frac{T(SSR_0 - SSR_1)}{SSR_0}, \tag{11}$$

while the F version can be calculated using

$$LM_3 = \frac{(SSR_0 - SSR_1)/3(p+1)}{SSR_1/[T - 4(p+1)]}. \tag{12}$$

From Eq. 12, it is clear that the F version of the test statistic follows a F-distribution with $3(p+1)$ and $T - 4(p+1)$ degrees of freedom. Since we are using an AR(1) time series model, Eq. 12 reduces to

$$LM_3 = \frac{(SSR_0 - SSR_1)/6}{SSR_1/[T - 8]}. \tag{13}$$

To determine the LM_1 test statistic under the null hypothesis of linearity for the LSTAR model, one can follow the same approach as for the LM_3 test statistic discussed above.

3.1.2 Testing linearity against ESTAR

When testing linearity against an ESTAR alternative, it is suggested to use the auxiliary equation

$$y_t = \beta_0^{'} x_t + \beta_1^{'} x_t s_t + \beta_2^{'} x_t s_t^2 + e_t \tag{14}$$

with $e_t = \varepsilon_t + (\phi_2 - \phi_1)^{'} x_t R_2 (s_t, \gamma, c)$, which results from Eq. 2 and 5/6. The corresponding null hypothesis is defined as $H_0 : \beta_1 = \beta_2 = 0$, with test statistic $LM_2 \sim \chi^2 (2(p+1))$.

Instead of using a first-order Taylor approximation, another test can be performed by using a second-order Taylor approximation. The resultant auxiliary regression equation follows:

$$y_t = \beta_0^{'} x_t + \beta_1^{'} x_t s_t + \beta_2^{'} x_t s_t^2 + \beta_3^{'} x_t s_t^3 + \beta_4^{'} x_t s_t^4 + e_t. \tag{15}$$

The null hypothesis that follows is defined as $H_0^{'} = \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$, with the corresponding test statistic $LM_4 \sim \chi^2 (4(p+1))$. Again take note that $p = 1$, since we are working with an AR(1) time series model. Note that there is a trade-off between the additional auxiliary variables and the level of the null hypothesis when deciding between the tests based on Eq. 14 and 15. Neither one of these tests are more powerful than the other.

The calculation of LM-test statistics for ESTAR models (using the auxiliary regression Eq. 15) can be done using the same algorithm as was done for LSTAR models in Section 3.1.2.

3.2 Misspecification tests of STAR models

One should thoroughly evaluate a fitted STAR model, including performing a number of misspecification tests, before it can be accepted as a satisfactory model. Some of these misspecification tests include testing for no error autocorrelation, no remaining nonlinearity, as well as tests for the constancy of parameters. Some of these tests, however, relate to extensions of the STAR model [4] which are not discussed in this report.

In the case of multiple regime STAR (MRSTAR) models, testing the hypothesis for no remaining nonlinearity is of concern. These models allow for more than two regimes, however, this report focuses only on a two regime STAR model, thus the hypothesis tests for no remaining nonlinearity will not be discussed in any further detail.

Testing for the constancy of parameters is relevant when working with time varying STAR (TVSTAR) models. Again, this is not covered in this report. Note that a TVSTAR model is relevant when one of the transition variables in a multiple regime STAR model is set to be equal to time. The reader is referred back to [4] for a full discussion on this section.

There are several computational aspects which need to be considered when applying misspecification tests. For a full discussion the reader is referred to [5]. More specifically, one problem arises when $\hat{\gamma}_1$ is very large so that the transition from the one regime to the other occurs rapidly under the null hypothesis. Another problem considered by [4] is that the residuals $\hat{\varepsilon}_t$ from the fitted two-regime STAR model are seldom orthogonal to the slope matrix. This problem occurs as a consequence when the fitted model is not appropriate for the observed time series.

4 Modelling cycle in the STAR framework

Teräsvirta [16] follows a "specific-to-general" approach (recommended by [7]) when modelling STAR models. This approach suggests that one starts with a simple model, working towards more involved models only if the analytical tests show that the maintained model is poor. The following steps are required for modelling a STAR model:

Step 1 Use a suitable model selection criterion to specify a linear AR model of order p for the time series at hand. To explain the theory in this section, we will be considering only an AR(1) model.

Step 2 Test the null hypothesis of linearity against the alternative of STAR nonlinearity, as discussed in Section 3.1. If nonlinearity is concluded, specify the proper transition variable, s_t, as well as a proper form of the transition function, $G(s_t; \gamma, c)$.

The LM_3 test statistic has power against both the LSTAR and ESTAR alternatives. This implies that the transition variable, s_t, can be specified before the form of the transition function is specified. The LM_3 test statistic can be calculated for a number of transition variables $s_{1t}, \ldots, s_{mt}$. The transition variable with the smallest p-value is then chosen. The reason for choosing the one with the smallest p-value is that we want the test to have greatest probability that the correct transition variable is applied. Teräsvirta [16] has shown that this approach is very effective in practice.

After specifying the appropriate transition variable, one needs to select the appropriate form of the transition function. The choices may, however, be limited between the logistic function (Eq. 4), exponential function (Eq. 5) or the second-order logistic function (Eq. 6). Different methods for choosing the appropriate transition function have been proposed in the past, however, recently it became easier to estimate numerous LSTAR as well as ESTAR models, and then select the appropriate model only at the evaluation stage (step 4) of the modelling cycle.

Step 3 Estimate the parameters for the chosen STAR model. Note that in this report we consider only the two-regime STAR model, however, the discussion on estimating the parameters can be applied to MRSTAR and TVSTAR models as well.

The parameters for the STAR model (Eq. 2) are estimated using a fairly simple method of nonlinear least squares (NLS). Thus, the parameters $\theta = \left(\phi_1^{'}, \phi_2^{'}, \gamma, c\right)^{'}$ can be estimated by

$$\hat{\theta} = \arg\min_{\theta} Q_T(\theta) = \arg\min_{\theta} \sum_{t=1}^{T} \left(y_t - F(x_t;\theta)\right)^2 \tag{16}$$

with $F(x_t;\theta) = \phi_1^{'} x_t (1 - G(s_t;\gamma,c)) + \phi_2^{'} x_t G(s_t;\gamma,c)$. Since we have assumed that the ε_t's are normally distributed, NLS can be considered as the maximum likelihood.

Another suggestion that makes the estimation problem a whole lot easier, known as concentrating the sum of squares function, was made by [11]. When the parameters γ and c are known and constant, the STAR model becomes linear in the autoregressive parameters ϕ_1 and ϕ_2. In this case ordinary least squares can be used to estimate the parameters $\phi = \left(\phi_1^{'}, \phi_2^{'}\right)^{'}$, hence

$$\hat{\phi}(\gamma, c) = \left(\sum_{t=1}^{T} x_t(\gamma,c)\, x_t(\gamma,c)^{'}\right)^{-1} \left(\sum_{t=1}^{T} x_t(\gamma,c)\, y_t\right) \tag{17}$$

where $x_t(\gamma,c) = \left[x_t^{'}(1 - G(s_t;\gamma,c)), x_t^{'} G(s_t;\gamma,c)\right]^{'}$ and $\phi(\gamma,c)$ implies that the estimate of the parameter ϕ is conditional upon γ and c. In other words, the sum of squares function $Q_T(\theta)$ can be concentrated with repect to ϕ_1 and ϕ_2 such that

$$Q_T(\gamma,c) = \sum_{t=1}^{T} \left(y_t - \phi(\gamma,c)^{'} x_t(\gamma,c)\right)^2. \tag{18}$$

This result reduces the complexity of the NLS estimation problem significantly.

Take note that it is very difficult to get an accurate result when estimating the parameter γ, when this parameter is in fact large. Large adjustments in γ actually has a very small impact on the transition function. Consequently, the estimate of γ may appear insignificant when looking at its t-statistic (see [4] for a more detailed discussion).

Step 4 By making use of analytical tests and impulse response analysis, one has to evaluate the resultant model. When evaluating the STAR model, one should apply misspecification tests such as those introduced in Section 3.2. Should at least one of the null hypotheses be rejected, one has to reconsider the specification of the model.

Methods for evaluation of the STAR model include local/sliced spectra and impulse response analysis. These two methods, however, will not be discussed in this report. The reader can follow a full discussion on these methods in [4].

Step 5 If needed, improve the model.

Step 6 Finally, use the resultant model for future purposes, such as forecasting.

5 Application: Modelling SA inflation rate with LSTAR model

As introduced initially, STAR models are mainly used (and was shown to be successful) to describe and analyse the behaviour of various macroeconomic time series. On macroeconomic level the main objectives that serve as criterion to judge the state of the economy are price stability, economic growth, full employment, balance of payments stability and equitable distribution of income. In this report the LSTAR model has been mainly discussed, hence we will illustrate the modelling cycle for the LSTAR model specifically in the following section. The LSTAR model was specifically shown to be more appropriate for the analysis of business cycles, where the regimes refer to the downswings and upswings in the economy respectively.

The following application is based on a time series of the seasonally unadjusted inflation rate for South Africa from January 1969 to July 2015, on a monthly basis, which have been obtained from *Statistics South Africa*[1]. Inflation is a yardstick of the general price level over time. By observing the inflation rate, and fitting the LSTAR model specifically to the time series, we can analyse the behaviour of the economy and get some indication of the business cycles. The software RStudio [14], with packages tsDyn [3] and TSA [1], has been used for the application of LSTAR modelling in this report (all relevant R-code is included in Section 7). An LSTAR model will be fitted using specific built-in functions of the software used. Afterwards, a comprehensive evaluation of the time series as well as the fitted model will be made. Finally, the regimes, which refer to the various business cycles, will be illustrated and analysed using a graph.

When analysing Figure 1, one can clearly notice business cycle irregularity. One should also notice that an increase in the inflation rate occurs rapidly during recessions while a decrease in the inflation rate occurs more gradually during expansions. This irregularity can be explained by various external factors which influence the inflation rate, such as the price of oil, demand and supply, climate and many more. External factors such as these all yield some contribution/impact on the general price level - some greater than others.

Since South Africa is a large importer of oil, any change in the oil price has a significant impact on the South African inflation rate. As can be noticed from Figure 1, in 1979 the Irarian revolution led to a sudden increase in the oil price[2]. Increases in the oil price usually have a negative impact on economic growth overall since production costs increase, hence supply decreases. Also, the increase in the oil price, leads to an increase in food as well. This caused the South African inflation rate to increase dramatically and becoming very volatile afterwards. The extremely high and unstable inflation rate during the 1970's and 1980's can also be explained by the unstable economic environment in South Africa as a result of the sanctions implemented in the 1960's in protest of the South African apartheid system. This implementation reached its peak in the mid-1980's [15]. There are various other explanations for the high volatility in the inflation rate as well.

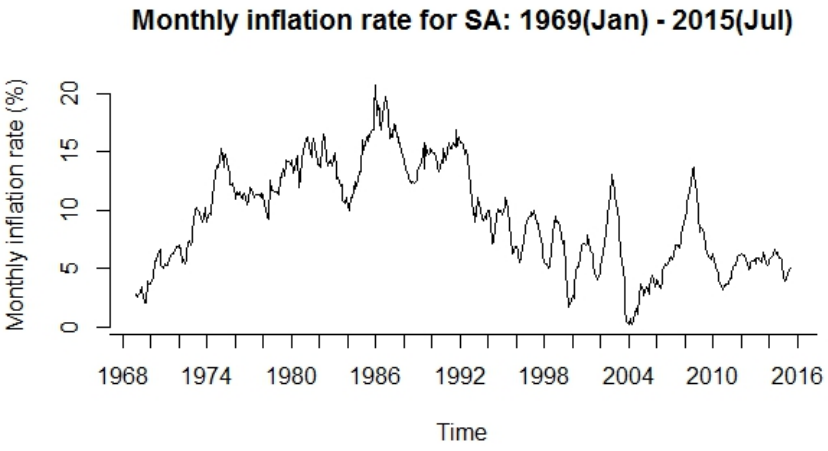


Figure 1: Seasonally unadjusted monthly inflation rate for South Africa, January 1969 to July 2015.

[1] http://www.statssa.gov.za/publications/P0141/CPIHistory.pdf?

[2] http://www.nytimes.com/2008/03/03/business/worldbusiness/03cnd-oil.html?hp&_r=0

Before testing for nonlinearity, the sample autocorrelation as well as sample partial autocorrelation function for the time series should be evaluated in order to define a possible linear AR(p) model. So far in this report we have assumed only an AR(1) model for simplicity purposes. However, after analysing Figure 2 and Figure 3 (which illustrates the sample autocorrelation function and sample partial autocorrelation function respectively), an AR(2) model should be considered since the sample partial autocorrelations at lag 1 and 2 in Figure 3 are not close to 0, thus they fall outside the control limits. When we evaluate the sample autocorrelation coefficients for the monthly inflation rate time series in Figure 2, it is clear that we have significant autocorrelations for all lags up until lag 10, which makes sence since an AR(p) model is applicable and the lagged time series observations have an influence on the current time seires value.

Note that in practice it is important to determine to most appropriate linear AR model (that is with the appropriate order p) for the observed time series. This can be done by comparing the Akaike information criterion (AIC) and the Bayesian infromation criterion (BIC) values for the various AR model. The model which has the smallest AIC or BIC is the appropriate AR model to use.

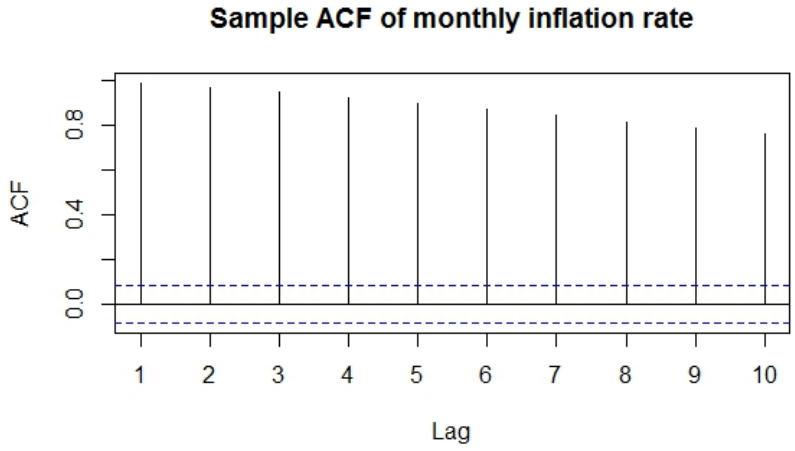


Figure 2: Sample autocorrelation function of the monthly inflation rate.

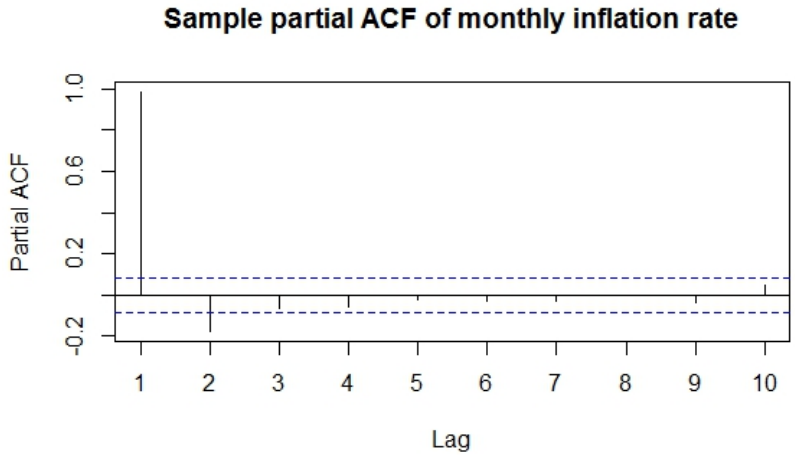


Figure 3: Sample partial autocorrelation function of the monthly inflation rate.

Once an appropriate linear AR model has been specified, a test against nonlinearity should be done. In order to fit some STAR model, the chosen AR(p) model should be nonlinear. To test for nonlinearity, the Keenan test [10], Tsay test [18] as well as the Lagrange multiplier (LM)-type test proposed by [16] were done in RStudio which all concluded nonlinearity when using an AR(2) model. Referring to Table 1, the p-values for all three these LM-type tests show that we conclude nonlinearity at a 1% level of significance for an AR(2) model.

Method	*AR(2)*
Keenan test	0.00940
Tsay test	< 0.001
LM-type test	< 0.001

Table 1: P-values from tests done to test against nonlinearity using AR(2): Keenan's test [10], Tsay's test [18] and Lagrange multiplier (LM)-type linearity test considered by Teräsvirta [16]. All three p-values reject linearity confidently at a 1% level of significance.

Since nonlinearity has been concluded using an AR(2) model, we can then estimate the parameters for the chosen LSTAR model. Note that in practice the decision between LSTAR and ESTAR can be made in the evaluation stage of the modelling cycle, since the two transition functions will estimate approximately the same parameters. Using the *function* LSTAR [6] [17] [11] in RStudio, one can estimate both the linear as well as the nonlinear parameters with their associated standard errors and p-values. These results are summarised in Table 2. RStudio estimates the parameters using the method of concentrated least squares. The nonlinear parameters γ and c were estimated using a grid search since they have not been specified in the *function* LSTAR. Note that almost all the parameters are highly significant, except for the constants for both regime 1 and 2, which is significant at a 10% and 5% level of significance respectively, whereas γ, the smoothing parameter, is not significant at all.

Coefficient	*Estimate*	*Standard error*	*t value*	*p-value*
Regime 1: Linear parameters				
$\phi_{1,0}$	0.15324	0.08432	1.81730	0.06917
$\phi_{1,1}$	1.38426	0.05379	25.73310	< 0.001
$\phi_{1,2}$	-0.40120	0.05277	-7.60230	<0.001
Regime 2: Linear parameters				
$\phi_{2,0}$	1.21158	0.49222	2.46150	0.01384
$\phi_{2,1}$	-0.53519	0.08564	-6.24930	< 0.001
$\phi_{2,2}$	0.45888	0.08106	5.66110	< 0.001
Nonlinear parameters				
γ	100.0001	168.34779	0.59400	0.55251
c	12.43943	0.08529	145.8520	< 0.001

Table 2: LSTAR model parameter estimates.

The estimated LSTAR model can be written as follows:

$$\hat{y}_t = (0.15 + 1.384y - 0.4y_{t-2})\,(1 - G\,(s_t; 100, 12.44)) + (1.21 - 0.54y_{t-1} + 0.46y_{t-2})\,G\,(s_t; 100, 12.44)\,.$$

Once the estimation stage has been completed, one need to evaluate the estimated model. Analysing the residuals gives us an idea of the adequacy of the model as we want the residuals to have a constant variance and indicating no pattern over the observed time period. The ideal is to have uncorrelated or independent residuals - there should be no presence of significant autocorrelation between the residuals. If we plot the residuals against time, we want it to have a similar "pattern" than that of white noise, thus independent and random. Observing Figure 4 one can argue that the residuals are in fact relatively independent and random, thus no clear pattern can be observed in the time plot of the residuals. Although the residuals have a mean

of 0, it appears however, that the residuals do not have a constant variance - the variance is quite large up until 2010, where after the variance decreases quite drastically, suggesting that the model estimated more accurately after 2010. This presence of heteroscedasticity, unfortunately, can be problematic and is not ideal.

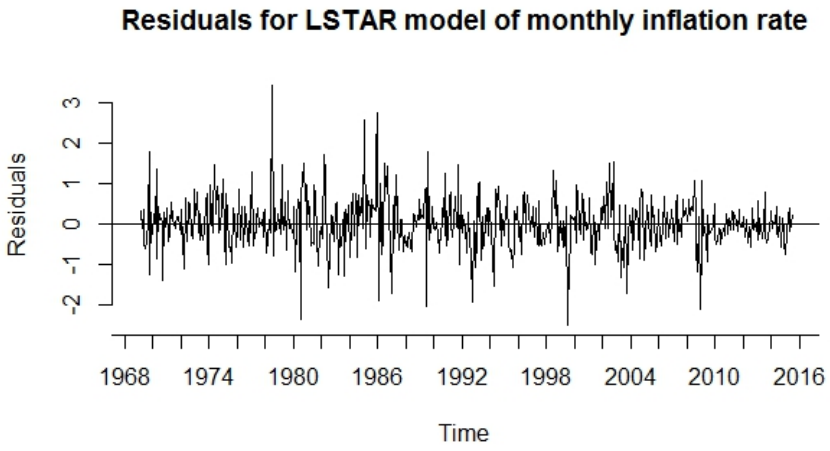


Figure 4: Residuals from fitted LSTAR accross time.

The sample autocorrelation function for the residuals from the fitted LSTAR model is illustrated in Figure 5. Note that the blue dashed lines illustrate the confidence intervals. The sample autocorrelations plot inside the confidence intervals for all lags - that is for lag 1 to 10 the sample autocorrelation coefficients are all close to 0, indicating that the residuals are uncorrelated. The sample partial autocorrelation function up to lag 10 in Figure 6 shows insignificant partial autocorrelation for the residuals as well.

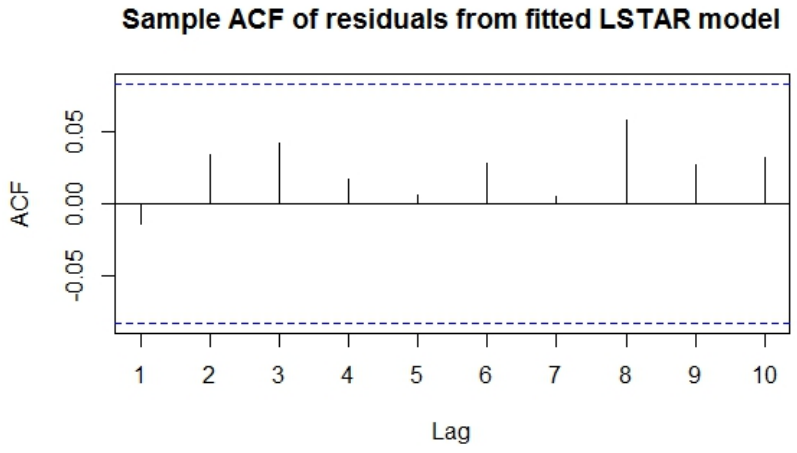


Figure 5: Sample autocorrelation function of the residuals from fitted LSTAR model.

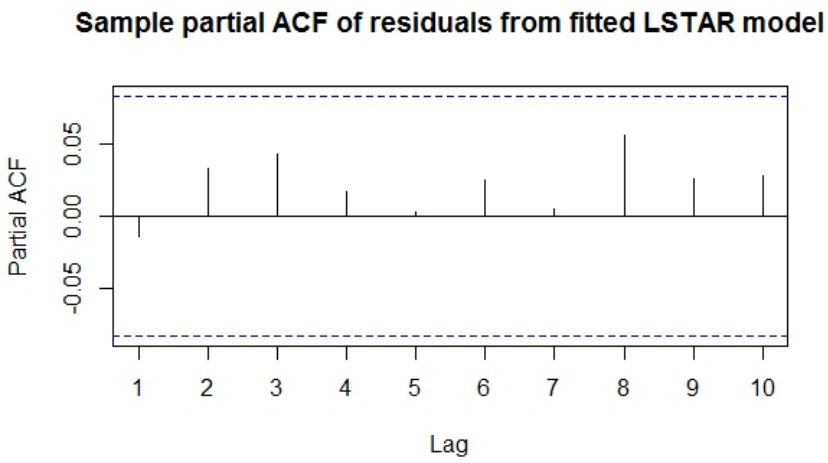


Figure 6: Sample partial autocorrelation function of the residuals from fitted LSTAR model.

Finally, one can use the fitted LSTAR model to analyse the economic behaviour and business cycles regarding the general price level. In Figure 7 the transition function (red line), predicted values (dotted black line) as well as the observed seasonally unadjusted South African monthly inflation rate (blue line) are plotted over time. The low regime indicates an expansion in the economy, whereas the high regime indicates a contraction. If one compares these two regimes to the inflation troughs and peaks, it is noticable that the regimes act as some lagging indicator of the business cycle the economy finds itself into. For example, (approximately) in 1975, 1987, 1992 and 2003, respectively, the South African inflation rate reached a trough (a turning point when the inflation rate started improving). Shortly after, the low regime followed, indicating an expansion in the economy. On the other hand, (approximately) in 1974, 1984 and 2002, respectively, the inflation rate reached a peak (a turning point when the inflation rate started deteriorating). Shortly after, the high regime followed, indicating a contraction in the economy.

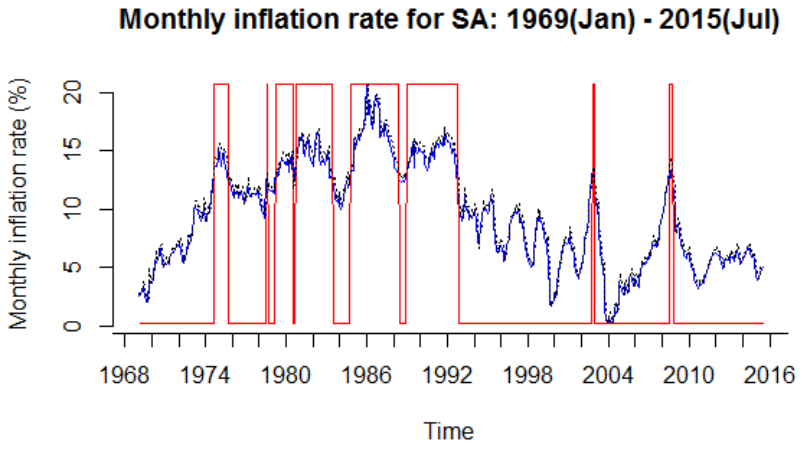


Figure 7: Regimes, fitted and observed monthly inflation rate against time.

It is clear that the LSTAR model is a relatively well fitted model and becomes very relevant in the analysis of macroeconomic time series, as well as business cycles, and can eventually also be used for forecasting purposes. It is especially useful since in real life one would seldom have access to linear time series.

6 Conclusion

This report discussed the STAR model comprehensively, yet simple by building all theory onto an AR(1) model. The reader will notice that the order of the linear AR model (which the STAR model is just an extension of) can be increased in a relatively simple manner. The STAR model with its three well known transition function was discussed and illustrated broadly. The various tests against nonlinearity have been introduced and the testing algorithm was shown specifically for the LSTAR model (which is similar for the ESTAR model) to guide the reader through the calculation process. Misspecification test have also been introduced, although this report did not discuss them in any detail. These misspecification tests are applied mainly to multiple regime STAR (MRSTAR) models as well as time varying STAR (TVSTAR) models. Thus, the misspecification tests as well as the MRSTAR and TVSTAR models can be investigated for future which can possibly yieald in more accurate results for the analysis of STAR models and in the forecasting stages.

The reader has been guided through the entire modelling cycle in the STAR framework. The various steps have been discussed in depth, however still as simple and straightforward as possible so that the reader with little mathematics background would be able to understand and apply the modelling cycle.

To simplify the theory discussion even more, and to really explain the role of STAR models, finally an empirical example was done using an actual time series of the seasonally unadjusted monthly inflation rate for South Africa over a time period of almost 47 years. In this application we have also tried to keep the modelling process as simple as possible. The software RStudio has been used with packages tsDyn [3] as well as TSA [1]. There are several other software programs that can be used for STAR modelling such as standard econometric packages, however some of these standard software do not cover all steps in the STAR modelling cycle. A collection of general GAUSS programmes, written by Stefan Lundbergh[3], covers the entire modelling cycle for STAR models [4].

So far STAR models have been mainly used in the application and analysis of macroeconomic time series. Therefore, other sectors such as the finance and marketing sectors create a great opportunity for future research. Future research in the properties of vector STAR models also need to be extended. Combining smooth transitions in panel data models creates another exciting opportunity for future research. Johansen [9] has been the first attempt in this area.

[3] http://ideas.uqam.ca/ideas/data/Softwares/bocbocodeG111201.html

References

[1] K S Chan and B Ripley. *TSA: Time Series Analysis. R package version 1.01*, 2012.

[2] J D Cryer and K S Chan. *Time Series Analysis With Applications in R*. Springer, 2008.

[3] A F Di Narzo, J L Aznarte, and M Stigler. *tsDyn: Time series analysis based on dynamical systems theory. R package version 0.6-0*, 2008.

[4] D van Dijk, T Teräsvirta, and P H Franses. Smooth transition autoregressive models - a survey of recent developments. *Econometric Reviews*, 21(1):1–47, 2002.

[5] Ø Eitrheim and T Teräsvirta. Testing the adequacy of smooth transition autoregressive models. *Journal of Econometrics*, 74(1):59–75, 1996.

[6] P H Franses and D Van Dijk. *Non-linear Time Series Models in Empirical Finance*. Cambridge University Press, 2000.

[7] C W J Granger. Strategies for modelling nonlinear time-series relationships. *Economic Record*, 69(3):233–238, 1993.

[8] C W J Granger and T Teräsvirta. *Modelling Nonlinear Economic Relationships*. Oxford University Press, 1993.

[9] K Johansen. Nonlinear wage responses to internal and external factors. Technical report, Norwegian University of Science and Technology, 1999.

[10] D MR Keenan. A Tukey nonadditivity-type test for time series nonlinearity. *Biometrika*, 72(1):39–44, 1985.

[11] S Leybourne, P Newbold, and D Vougas. Unit roots and smooth transitions. *Journal of Time Series Analysis*, 19(1):83–97, 1998.

[12] C F J Lin and T Teräsvirta. Testing the constancy of regression parameters against continuous structural change. *Journal of Econometrics*, 62(2):211–228, 1994.

[13] R Luukkonen, P Saikkonen, and T Teräsvirta. Testing linearity against smooth transition autoregressive models. *Biometrika*, 75(3):491–499, 1988.

[14] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.

[15] S Rose and H Rose. Boycott of Israel? It worked for South Africa. *Nature*, 417(6886):221–221, 2002.

[16] T Teräsvirta. Specification, estimation, and evaluation of smooth transition autoregressive models. *Journal of the American Statistical Association*, 89(425):208–218, 1994.

[17] H Tong. *Non-linear Time Series: A Dynamical System Approach*. Oxford University Press, 1990.

[18] R S Tsay. Nonlinearity tests for time series. *Biometrika*, 73(2):461–466, 1986.

7 Appendix

The code for the application in Section 5, using RStudio [14], is as follows:

```
library("tsDyn", lib.loc="C:/Program Files/R/R-3.2.2/library")
library("TSA", lib.loc="C:/Program Files/R/R-3.2.2/library")

#Constructing time seires for monthly data
data.minf = read.table("C:\\Users\\Ané\\Desktop\\STK795Data\\
    monthly_inflation.txt")
minf.ts = ts(data.minf, start = c(1969,1), end = c(2015,7), frequency=12)

#Testing for nonlinearity (for both order=1 & order=2)
Keenan.test(minf.ts, order = 1)
Tsay.test(minf.ts, order = 1)
Keenan.test(minf.ts, order = 2)
Tsay.test(minf.ts, order = 2)
starmod = star(minf.ts, m=1)
starmod = star(minf.ts, m=2)

#Estimating parameters (linear as well as nonlinear)
summary(starmod)
lstarmod = lstar(minf.ts, m=2)
summary(lstarmod)

#Evaluating the fitted model
#Time series
plot(minf.ts, main='Monthly inflation rate for SA: 1969(Jan) - 2015(Jul)',
    ylab='Monthly inflation rate (%)', xlab='Time', type='l', axes = FALSE)
axis(side = 1, at = seq(0, 2025, by=2))
axis(side = 2)
#Residuals
plot(residuals(lstarmod), main='Residuals for LSTAR model of monthly inflation
    rate', ylab='Residuals', xlab='Time', type='l', axes = FALSE)
axis(side = 1, at = seq(0, 2025, by=2))
axis(side = 2)
abline(h=0)
#Autocorrelations
acf(ts(minf.ts, freq=1), lag.max = 10, xaxp=c(0,10,10), main='Sample ACF of
    monthly inflation rate')
lstarmod2 = lstar(ts(minf.ts, freq=1), m=2)
acf(residuals(lstarmod2), na.action = na.pass, main='Sample ACF of residuals
    from fitted LSTAR model', lag.max = 10, xaxp=c(0,10,10))
#Partial autocorrelations
pacf(ts(minf.ts, freq=1), lag.max = 10, xaxp=c(0,10,10), main='Sample partial
    ACF of monthly inflation rate')
pacf(residuals(lstarmod2), na.action = na.pass, main='Sample partial ACF of
    residuals from fitted LSTAR model', lag.max = 10, xaxp=c(0,10,10))

#Regimes
regime(lstarmod)
plot(regime(lstarmod))
plot(minf.ts, main='Monthly inflation rate for SA: 1969(Jan) - 2015(Jul)',
    ylab='Monthly inflation rate (%)', xlab='Time', type='l', col='blue',
```

```
          axes = FALSE)
axis(side = 1, at = seq(0, 2025, by=2))
axis(side = 2)
par(new=TRUE)
plot(regime(lstarmod), col='red', axes = FALSE, ylab='', xlab='')
par(new=TRUE)
plot(fitted(lstarmod), axes = FALSE, lty=3, ylab='', xlab='')
```

Credit scoring using non-parametric kernel density estimation

Estian Rademeyer 12014894

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor(s): Dr C.M. van der Walt

Department of Statistics, University of Pretoria

2 November 2015

Abstract

This paper investigates the performance of one-class and two-class classification on the Australian and German credit scoring data sets by extending one-class parametric Gaussian and non-parametric Parzen classifiers to two-class classifiers with Bayes' rule. Furthermore, the performance of Parzen classficiation with Silverman and Minimum Leave-one-out Entropy (MLE) Gaussian kernel bandwidth estimation is also investigated.

Declaration

I, *Estian Rademeyer,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____

Estian Rademeyer

_____

Dr C.M. van der Walt

_____

Date

Acknowledgments

A special thanks to Dr. Van der Walt for his guidance and infinite wisdom. I would furthermore like to thank the University of Pretoria's statistics department, as well as my family, for giving me the opportunity to do this research.

Contents

List of Figures

List of Tables

Listings

1 Introduction

The 2007-2008 financial crisis is the perfect example to emphasize the vital importance of credit scoring. The Federal Funds rate was lowered from 6.5% to 1% by the Federal Reserve during the period of May 2000 to June 2003. This is the lowest it has been for 45 years. To make things even worse the Securities Exchange Commission reduced the capital requirement for five banks (in October 2004) [17]. This made credit seem extremely cheap, something it surely isn't. Not only did bankers grant credit to those that wouldn't be able to repay it at higher interest rates, but they took on too many debtors and passed it on to other financial institutions.

The first domino started to fall when the rates were increased during the period June 2004 to June 2006. Suddenly the U.S Home Construction Index fell and borrowers could not repay their debt [17]. Although many factors contributed to the financial crisis, it was poor credit scoring that essentially led to majour lenders filing for bankruptcy.

It is clear that a fine balance needs to exist between the granting of too much or too little credit. As demonstrated so elegantly by the 2007-2008 financial crisis, the granting of too much credit can lead to borrowers defaulting on their debt and thus driving the lender to bankruptcy. However, granting too little credit can lead to poor business performance and will ultimately also force the lender to default. This is where credit scoring plays a vital role.

In previous years, the decision to grant credit was made by a specialist in the field on a case-by-case basis. As the demand for credit increased, this process was computerized by means of credit scoring. Credit scoring is a process used to determine the likelihood that borrowers would repay their debt. Credit scoring can be subdivided into applicational scoring and behavioral scoring. Application scoring awards borrowers points at the time of the application based on a few relevant characteristics such as income, age and profession. Behavioral scoring on the other hand, scores current borrowers based on their recent transactions. Depending on the type of scoring system, a credit score is typically a number between 300 and 850 or 501 and 990 [8]. The score divides borrowers into two classes. Borrowers that achieve a score above a predetermined level, or so called 'cut-off level', would be considered to have a low probability of default or to be credit worthy. However, borrowers that do not achieve a score exceeding the 'cut-off level' would be considered to have a high probability of default.

Modeling credit risk could be complicated by the low default portfolio problem. The latter occurs when an imbalance between the two classes exists. This imbalance may arise due to two possible reasons. The first is that the proportion of the sample, as well as the population of the one class differs from the proportion of the other class. The second is that the proportion of one class in the population differs from the proportion in the sample. This typically occurs when modeling credit risk, due to a small proportion of defaulters. This shortage of data (number of defaulters) results in the calculated probability of defaults producing a distorted image of the behavior of defaults.

When modeling credit risk, there is a certain level of uncertainty with regard to whether to use non-parametric or parametric distribution estimation. For this reason this paper will compare non-parametric kernel density estimation and a parametric Gaussian density estimation on a predetermined data set.

One of the key elements to successful non-parametric kernel density estimation is the selection of the bandwidth parameter. The bandwidth has a strong influence on the resulting estimate. The mean integrated squared error (MISE) is often used to select the optimum bandwidth estimator. The MISE can not be used directly since the formula for the MISE contains the density function we wish to estimate using kernel density estimation (this can explicitly be seen in the univariate proof of Silverman's Rule of Rhumb in the appendix). This problem is solved by using cross validation and plug-in selectors. This paper will only consider plug-in selectors; in particular it will make use of Silverman's Rule of Thumb and the Minimum Leave-One-Out Entropy (MLE) method.

The paper will also evaluate the performance of one-class compared to two-class classification on low default portfolios. Although a paper by Kennedy compared one-class and two-class classifiers, it only utilised the Gaussian and Parzen classifiers as one-class classifiers where the majority class was modeled [11]. It concluded that one-class classification outperforms two-class classification when the proportion of defaulters is very low, typically when defaulters are less then 1% of sample. This paper will therefore aim to use Bayes' rule to extend the Gaussian and Parzen classifiers to two class classification by modeling both class-

conditional pdf's and accounting for class imbalances with class priors, the negative effect of imbalanced data on the performance of two class classification would be reduced. Note that one-class classification methods only take into account the distribution of past borrowers that repayed their debt. Two-class classification methods on the other hand take into account the distribution of borrowers that repayed their debt as well as those that defaulted on their debt.

2 Literature Review

As mentioned above, the low default portfolio problem drastically complicates the modeling of credit risk. Models addressing the low default probability problem can either be statistical models based on historical data [7], or systems with parameters estimated by financial experts [23]. The statistical models can be subdivided into two groups: Duration models and classification models. Duration models require large data sets and focus on the time to default [15]. These models however do not directly provide an estimate of the probability of default. Classification models use methods such as discriminant analysis and logic regression [3], decision trees [19], non-parametric statistics and neural networks [22]. Other methods that have been used include support vector machines [10], genetic algorithms [16] and ant colony optimization [14].

There are two main types of classification methods: Discriminative and generative (also known as model based) classification. Discriminative classifiers model $P(y|\mathbf{x})$, the posterior class probabilities. Note that $P(x)$ isn't modeled and therefore assumptions with regard to unlabeled samples are required. Discriminative classifiers fit a decision boundary between two classes in such a way that the error rate, or cost of error is minimized. K-Nearest Neighbor, Auto Encoders, Neural Networks, Support Vector Machines, Decision Trees and k-Means are a few examples of discriminative classification.

The k-Nearest Neighbor classifier receives multidimensional vectors each with a class label, as training sets. A constant k is specified by the user. Unlabeled vectors are then classified by assigning the class that occurs most often among the k nearest neighbors to the unlabeled vector. In general, the Euclidean distance is used to measure the distance between the unlabeled vector and the training sets. Other methods such as the Hamming distance may also be used. Henley and Hand investigated the optimal k and distance for the evaluation of credit risk. They found that the k-Nearest Neighbor classifier is fairly insensitive to these parameters [9].

A Neural Network model is essentially a set of mathematical functions that receive an input vector say $\mathbf{x}$ and produces an output vector, say $\mathbf{o}$. The relationship that exists between $\mathbf{x}$ and $\mathbf{o}$ depends on the set of functions used in the model. Neural Networks should have a minimum of three layers: The input, output and a hidden layer [3]. A network generally has three parameter types: The interconnection pattern that connects different layers, the learning process that updates the weights of the interconnections, and the function that transforms the weighted input into output.

Support Vector Machines aim to find an optimal hyperplane through the maximization of Lagrange multipliers of the training vectors. An optimal hyperplane results in the minimum number of training errors. The Support Vector Machine transforms the training sets into a higher dimensional space in order to generate a flexible boundary. The classification is determined by the distance between the object that needs to be classified and the boundary of the threshold.

Generative classifiers calculate the joint distribution of the underlying class. Based on this joint distribution it can generate labeled instances. Decision theory is applied in the case of unlabeled instances. In the case of classification, determining the full underlying distribution is unnecessary and might even result in lower performance. If the cost of error is independent of the joint distribution, then generative classification outperforms discriminative classification [4].

Generative classifiers can further be subdivided into non-parametric and parametric classifiers. Examples of non-parametric generative classifiers include Mixture of Gaussian and non-parametric kernel density estimation, whereas Gaussian and Naïve Bayes are examples of parametric generative classifiers, since a parametric function is imposed on the class-conditional density function.

The linear combination of k Gaussian distributions is known as the mixture of Gaussian model. The training set is subdivided into k clusters. Each cluster is modeled by a single Gaussian distribution. The

superposition is then given by:

$$f(\mathbf{z}) = \sum_{i=1}^{k} \alpha_i e^{-(\mathbf{z}-\mu_i)^T \Sigma_i^{-1} (\mathbf{z}-\mu_i)}$$

with Σ the covariance matrix, μ the mean and α_i the mixture weight. The classification is determined by the threshold on the density. It is important to take into account that this method will result in a large variance if the amount of data is insufficient.

As the name suggests, non-parametric kernel density estimation (also known as Parzen density estimation) uses a predetermined kernel to estimate the density function.

For the univariate case, let $D = \{x_1, x_2, ..., x_n\}$ be a sample of n independent identically distributed random variables from an unknown distribution $p(\cdot)$. The non-parametric kernel density estimate is then given by:

$$\begin{aligned} \hat{p}(x) &= \frac{1}{n} \sum_{i=1}^{n} K_h(x - x_i) \\ &= \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \end{aligned}$$

where $h > 0$ is the bandwidth (the smoothing parameter), $K(\cdot)$ is the kernel and $K_h = \frac{1}{h} K\left(\frac{x}{h}\right)$. Note that the kernel integrates to one, has a mean of zero and is non-negative. In this paper we will assume that K is a Gaussian kernel function, thus:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$$

Considering the multivariate case, let $D = \{\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_n}\}$ be a sample of n, d-dimensional random vectors from an unknown distribution $p(\cdot)$. Then the multivariate non-parametric kernel density estimate is given by:

$$\begin{aligned} \hat{p}(\mathbf{x}) &= \frac{1}{n} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{x}_i) \\ &= \frac{1}{n} |\mathbf{H}|^{-\frac{1}{2}} \sum_{i=1}^{n} K\left((\mathbf{H}^{-\frac{1}{2}})(\mathbf{x} - \mathbf{x}_i)\right) \end{aligned}$$

where $\mathbf{H}$ is a positive definite, symmetric $d \times d$ bandwidth matrix, $K(\cdot)$ is the kernel function and $\mathbf{x}$ and $\mathbf{x}_i$ are d dimensional vectors. Since we assume K to be a Gaussian kernel it follows that:

$$K(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}} e^{-\frac{1}{2}\mathbf{x}^T \mathbf{x}}$$

This paper will use Silverman's Rule of Thumb in both the univariate and multivariate case for the approximation to the bandwidth. Silverman suggested using $h \approx 1.06 \hat{\sigma} n^{-\frac{1}{5}}$ (see section 6.1 for the proof) for the univariate case, where $\hat{\sigma}$ is the standard deviation of the elements and n is the number of elements. He also suggested using $\sqrt{\mathbf{H}_{ii}} = \left(\frac{4}{d+2}\right)^{\frac{1}{d+4}} n^{\frac{-1}{d+4}} \hat{\sigma}_i$ for the multivariate case, where n is the number of row vectors and $\hat{\sigma}_i$ is the standard deviation of the $i-th$ row vector. A paper by Van der Walt and Barnard compares traditional bandwidth estimators for kernel density estimation. One of the conclusions they draw is that in general Silverman's rule of thumb performs consistently well across numerous data sets investigated [21].

It is important to realize that when there are large variance in the density, this method will not be able to adopt to such variations. This problem can be addressed through the use of variable kernel bandwidths. Breiman investigated the use of bandwidth variation where the data is scattered [1]. This paper will only focus on the difference between parametric (Gaussian) estimation and non-parametric kernel density estimation with fixed bandwidths.

As mentioned in the introduction, over and above Silverman's rule of thumb this paper will make use of the Minimum Leave-One-Out Entropy (MLE) estimator. One of the key properties of the MLE estimator is that it is feasible for higher-dimensional estimations unlike some conventional bandwidth estimators. This method uses the principle that minimization of the sample entropy and maximization of the log-likelihood function is equivalent. The MLE's equation is similar to that of the Maximum Leave-One-Out Likelihood's (MLL) equation, only differing in the fact that both the numerator and the denominator of the MLE equation is normalized. This leads to the reduction in the effect of data points that fall in dense regions and an increase in the effect of data points that fall in the lower dense regions, on the estimated bandwidth. Therefore although MLL and MLE estimators result in similar performance, the MLE estimator outperformed the MLL estimator for all data sets investigated. Another advantage of the MLE estimator is that it can estimate unique bandwidths for the outliers, which often require larger bandwidths[20]. The MLE bandwidth estimate for a diagonal bandwidth matrix is given by:

$$\mathbf{H}_{k(d,d)} = \frac{\sum_{i=1}^{N} \frac{K_{\mathbf{H}_k}(\mathbf{X}_i - \mathbf{X}_k|\mathbf{H}_k)(x_{id}-x_{kd})^2}{p_{\mathbf{H}(-i)}(\mathbf{X}_i)})}{\sum_{i=1}^{N} \frac{K_{\mathbf{H}_k}(\mathbf{X}_i - \mathbf{X}_k|\mathbf{H}_k)}{p_{\mathbf{H}(-i)}(\mathbf{X}_i)}}$$

where $\mathbf{H}_{k(d,d)}$ is the bandwidth for the kernel fitted over the k^{th} data point in dimension d. See section 6.2 for the proof.

Gaussian estimation assumes that the data is normally distributed. However, if the data is not normally distributed, the model may result in a large bias. In the multivariate case, the Mahalanobis distance for a point $\mathbf{z}$ is given by:

$$f(\mathbf{z}) = (\mathbf{z} - \mu)^T \Sigma^{-1} (\mathbf{z} - \mu)$$

where Σ is the covariance matrix and μ is the mean of the data. Finally the classification is made by comparing the distance to the threshold.

Naive Bayes classifiers typically model the class-conditional density functions with a Gaussian distribution with a diagonal covariance matrix. The features are thus assumed to be uncorrelated. Let $\mathbf{x} = (x_1, ..., x_n)$ be a vector of n features. Then a Naive Bayes probability model assigns probabilities $P(C_k|x_1, ..., x_n)$ to each of the k possible classes (C_k denotes the k-th class). Using Bayes' theorem it can be simplified to:

$$P(C_k|\mathbf{x}) = \frac{P(C_k)P(\mathbf{X}|C_k)}{P(\mathbf{X})}$$

Assuming every feature is indipendent of the other features, this expression can simplify to:

$$P(C_k|x_1, ..., x_n) = \frac{1}{P(\mathbf{X})} P(C_k) \prod_{i=1}^{n} P(X_i|C_k)$$

A maximum a posteriori decision rule needs to be added to form a Naive Bayes classifier. The function that labels the class $\hat{y} = C_k$ for a k is then given by:

$$\hat{y} = \underset{k \in \{1,...,K\}}{\arg\max} P(C_k) \prod_{i=1}^{n} P(X_i|C_k)$$

K-fold cross-validation is done by splitting the data set into k folds of equal size. The classes (in this case defaulters and non-defaulters) in the data set are determined and an equal ratio, of class one to class

two, is divided into each fold. A training set is set up such that it consists out of all the combinations of $k-1$ of the k folds. For each combination the remaining set is used as the testing set. Cross-validation ensures that the data is independent. It also ensures that the error on the testing set represent the model performance. In order to reduce variability each round of cross-validation must be repeated multiple times. K-fold cross-validation is illustrated in Figure 1

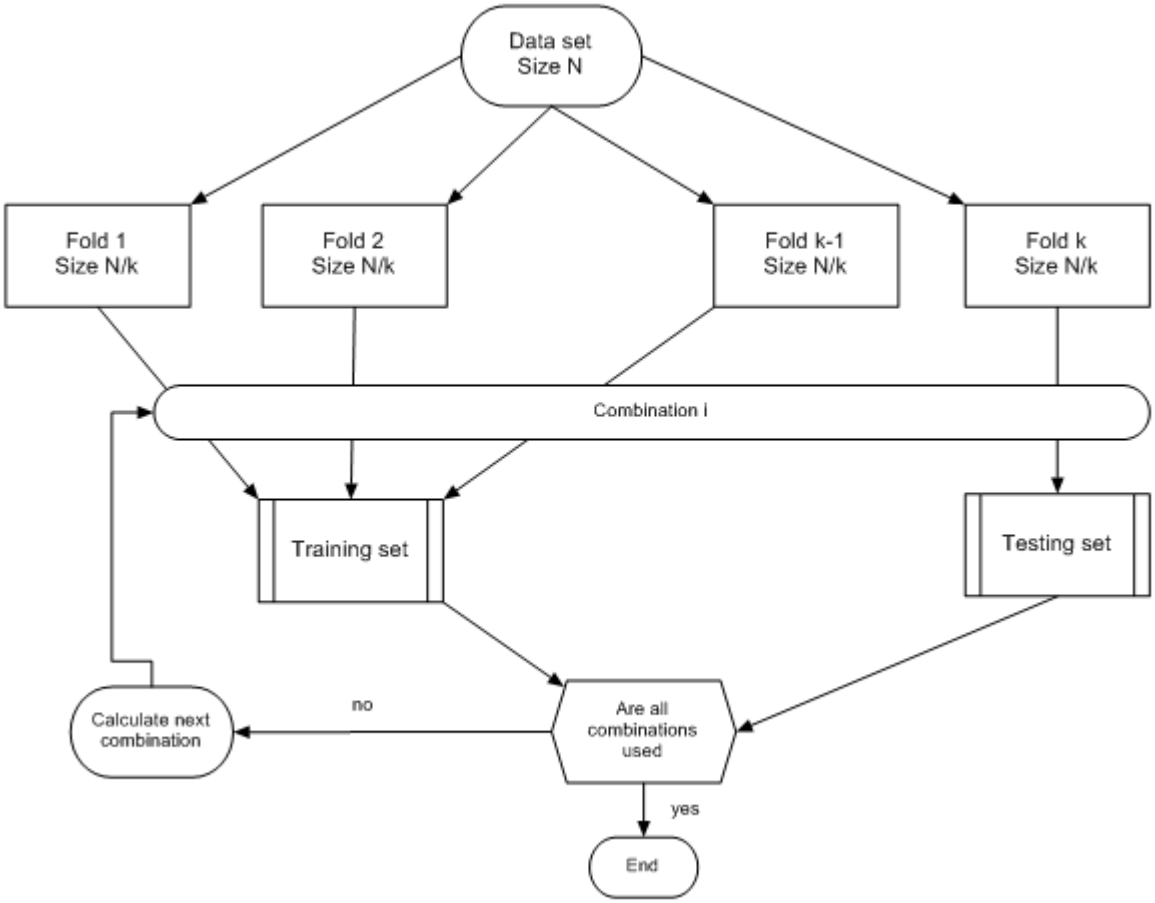


Figure 1: k-fold cross-validation

It is important to understand the difference between one-class and two-class classification. Two-class classification divides the training set (as described above) into a training set for class one and another for class two. These training sets are used respectively to train the particular classifier for each class. Once the classifier trained on each of the classes respectively it uses the testing set to calculate two sets of likelihoods, one for each class. The way in which the likelihoods are calculated depends on the particular classification method and have already been discussed for various methods. The likelihood scores for each entry in the testing set is compared. The class label of the largest likelihood is assigned to the entry.

One class-classification on the other hand only uses one training set to train the particular classifier. Likelihoods are calculated for each instance in the data set using the particular classification method. Based on a threshold the class label is assigned to each instance; any instance with a likelihood larger than the threshold is assigned one class label, while any instance with a likelihood smaller than the threshold is assigned the other class label.

A paper by Desai concludes that neural networks outperforms traditional techniques such as logic regression and linear discriminant analysis. It also states that neural networks and genetic algorithms have the ability to capture non-linear relationships as well as classifying borrowers into three groups instead of two [2].

Based on the specified data set Huang concludes that support vector machine based models, perform similar to that of genetic programming, backpropagation neural networks and C4.5. He goes further to recommend a hybrid version of support vector machines [10]. An interesting paper by Stiglitz and Weiss states that credit risk is affected by the interest rate charged by banks [18].

Kennedy evaluates the effectiveness of using one class classification to address the low default probability problem. This is done by comparing the performance of one class classification to that of two class classification with regard to the low default probability problem. Gaussian, Mixture of Gaussian, Parzen Density Estimation, Naïve Parzen, k-Nearest Neighbour, Support Vector Domain Description, k-Means and Auto-encoders are all estimation methods used to compare one and two class classifiers. Nine different data sets are used. The study evaluates the performance by means of the harmonic mean and the H measure. Kennedy shows that as defaulters are gradually removed, the performance of the two class classifiers (measured using H measure) deteriorates drastically. It also shows that since there aren't any non-defaulters removed, the performance of the one class classifiers remains fixed throughout. Kennedy recommends that with a proportion of 1% or less, one class classification should be used. It is suggested that one class classification might be a solution to the low default probability problem. Kennedy states that if a population drift occurs that the performance of two class classifiers will deteriorate [11].

In 1963 Bayes published a paper entitled "An Essay Towards Solving a Problem in the Doctrine of Chances". The paper contains the theorem of elementary probability theory that led to "Bayes' rule" as it is known today. It was Laplace that suggested that the posterior distribution is proportional to the likelihood [5]. Today Bayes' rule is known and used as: The posterior distribution is proportional to the prior times the likelihood. That, in mathematical terms is:

$$f(\theta|\underline{X}) \propto f(\underline{X}|\theta)f(\theta)$$

It is important to take into account the assumptions concerning Bayesian estimation: Data is viewed in a probabilistic fashion and prior information exists and contributes to the estimation. Bayesian modeling is done by implementing a probability model for the unknown parameter, updating the knowledge about this parameter by conditioning the probability model and finally evaluating the appropriateness of the model.

It was Neyman and Pearson that helped to develop the techniques that are today known as frequentest methods [5]. Take note that the frequentest approach assumes that a frequency exists, the parameters are fixed and the study is repeatable. The class priors for Bayes' rule can be estimated by simply counting the frequencies of the two classes.

The German data set used in this paper is a multivariate data set. It contains a thousand instances with twenty attributes. It was donated in 1994 by Professor Dr. Hans Hofmann of the University of Hamburg. The original data set contains categorical as well as integer values. A short description of each attribute follows in Table: 1 [12]:

Attribute number	**Feature**	**Possible Values**
1	Status of Checking Account	In terms of Deutsche Mark(DM) A11-... < 0DM A12-0 ≤ ... < 200DM A13-... ≥ 200 DM A14-No Checking Account
2	Duration	Integer Value (Months)
3	Credit History	A30 - Credits paid back duly A31 - Credits paid back duly at this bank A32 - Existing credits paid back dully to date A33 - Previous delay in payment A34 - Other credits existing
4	Purpose	A40 - New car A41 - Used car A42 - Furniture/ Equipment A43 - Radio/ Television

		A44 - Domestic appliances A45 - Repairs A46 - Vacation A47 - Education A48 - Retraining A49 - Business A410 - Other
5	Credit Amount	Numerical value rounded to the nearest hundred. Given in hundreds.
6	Savings Account or Bonds	A61 -< 100 DM A62 - 100 $\leq$ ... < 500 DM A63 - 500 $\leq$ ... < 1000 DM A64 - ... $\geq$ 1000 DM A65 - Unknown
7	Present Employment	A71 - Unemployed A72 -< 1 year A73 - 1 $\leq$ ... < 4 years A74 - 4 $\leq$ ... < 7 years A75 - $\geq$ 7 years
8	Installment Rate	Percentage of disposable income
9	Personal Status and Sex	A91 - Male: divorced/ separated A92 - Female: divorced/ separated/ married A93 - Male: single A94 - Male: married/widowed A95 - Female: Single
10	Other Debtors	A101 - None A102 - Co-applicant A103 - Guarantor
11	Present Residence since	Numerical value
12	Property	A121 - Real estate A122 - Building society/ life insurance A123 - Car or other not in attribute 6 A124 - No known property
13	Age	Numerical value (years)
14	Other Installment plans	A141 - Bank A142 - Stores A143 - None
15	Housing	A151 - Rent A152 - Own A153 - For free
16	Existing Credits at this Bank	Numerical value
17	Job	A171 - Unemployed/ unskilled non-resident A172 - Unskilled resident A173 - Skilled employee A174 - Management/ self-employed/ highly qualified employee/ officer
18	Number of dependence	Numerical value
19	Telephone	A191 - None A192 - Registered to debtor
20	Foreign Worker	A201 - Yes A202 - No

Table 1: German data set

In order to do apply the necessary algorithms and calculations to the data, the data set must be free of any categorical variables. Therefore we will use the modified version "german.data-numeric" of the data set. The modifications were made by Strathclyde University. They replaced all categorical variables with indicator functions and integer values. The modified data set has twenty-four attributes.

This paper will also use the Australian Credit Approval data set. The data set contains data relating to credit card applications. The data set consists of six-hundred-and-ninety instances and fourteen attributes. The data set also contains a few missing values. Even though the source of the data set is confidential it can be obtained from UCI [12]. A description of the attributes are given in Table 2.

Attribute number	Variable type	Possible values
A1	Categorical	0,1
A2	Continues	
A3	Continues	
A4	Categorical	1,2,3
A5	Categorical	1,2,3,4,5,6,7,8,9,10,11,12,13,14
A6	Categorical	1,2,3,4,5,6,7,8,9
A7	Continues	
A8	Categorical	0,1
A9	Categorical	0,1
A10	Continues	
A11	Categorical	0,1
A12	Categorical	1,2,3
A13	Continues	
A14	Continues	
A15	Class attribute	1,2

Table 2: Australian data set

Since the classifying methods all need a complete set of data and the Australian data set have a few missing values, imputation is implemented. Imputation is used to assign values to the missing entries. The data set is divided into two classes based on the class variable. The averages of the values of the non-missing entries in each class is calculated for each individual attribute. These values are substituted for each of the missing continuous values in the corresponding class and attribute. This method is known as cell mean imputation. The same is done for the missing categorical variables with the mode instead of the mean. An additional categorical variable should be added to indicate whether any attribute for the entry has been measured or imputed. It is assumed that the missing entries are missing completely at random [13].

3 Experimental Design

The aim of the study, as mentioned before, is to not only evaluate the performance of one-class versus two-class classifiers at different default ratios, but also to evaluate the performance of parametric versus non-parametric classifiers. In order to do this a series of experiments are conducted.

3.1 Preliminaries

Receiver operating characteristic (ROC) curves use confusions matrices to plot the true positive rate (TPR) against the false positive rate (FPR) for various thresholds. In the case of a perfect classification or prediction by the classifier a point would appear with the coordinate (0,1), indicating 100% sensitivity and 0% false positives. A classification or prediction that is completely at random on the other hand would result in a point appearing on the diagonal or line of no-discrimination.

The area under a ROC curve represents the accuracy of the particular classifier. For the following experiments the areas under the curves are calculated using the trapezoidal rule. An area of 0.5 implies that

the classifier is ineffective and has the same probability of making the correct classification as when a class is chosen at random. On the other hand an area of 1 implies that every class is predicted correctly.

Over and above the use of the AUC, the harmonic mean is used to evaluate the performance of the classifiers. The harmonic mean measures the performance of a classifier at a fixed threshold. The harmonic mean uses the confusion matrices to calculate the sensitivity as well as the specificity. This measures the quality of the classifiers. Specificity is given by

$$\frac{TN}{TN+FP}$$

and sensitivity is given by

$$\frac{TP}{TP+FN}$$

Finally

$$Harmonic\ Mean = \frac{2 \times Sensitivity \times Specificity}{Sensitivity + Specificity}$$

The hit rate of a classifier is simply the average of the number of classes correctly classified. In order to determine whether a classifier is performing acceptable, the hit rate should be compared to a benchmark. The proportional chance criterion can be calculated as $C_{PRO} = q_1^2 + q_2^2$, where q_1 and q_2 are the proportions of the different classes respectively. In general the hit rate of a classifier should be a quarter greater than the proportional chance criterion. It should be mentioned that this method only gives a rough indication of performance and becomes redundant for larger class imbalances.

A confusion matrix or contingency table consists of columns that represent the instances of the predicted classes, whereas the rows represent the instances of the actual classes. See Table 3.1 for an example of a confusion matrix. From the confusion matrix the false positive rate (FPR) can be calculated as

$$FPR = \frac{FP}{FP+TN}$$

and the true positive rate can be calculated as

$$TPR = \frac{TP}{TP+FN}$$

	C_1'	C_2'
C_1	True Positive (TP)	False Negative (FN)
C_2	False Positive (FP)	True Negative (TN)

Table 3: 2×2 Confusion Matrix

For all experiments that follow the one-class and two-class classifiers that are used include: Naive Bayse, Gaussian, Kernel Density Estimation using Silverman's rule of Thumb and Kernel Density Estimation using MLE classifiers. These classifiers are implemented on the German as well as the Australian data sets for all of the following experiments.

3.2 One-class vs. two-class classification for a fixed default rate

The aim of this experiment is to compare the performance of one-class versus two-class classifiers for a fixed ratio of defaulters. This is done by graphing the receiver operating characteristic (ROC) curves for the one-class as well as the two-class classifiers on the same set of axis, for each of the classifying techniques.

Ten-fold cross-validation is implemented and the likelihoods for each classifier are calculated as described in the Literature Review. These likelihoods of the two-class classifiers are multiplied by the corresponding

frequentest priors to form new likelihood scores. These new likelihoods are then normalized. The normalized likelihoods are used to determine the predicted classes and hence to set up the respective confusion matrices. The confusion matrices are used to draw the required ROC curves for the corresponding two-class classifiers.

The likelihoods of the one-class classifiers are ordered from small to large. The respective thresholds are set equal to the midpoints of the consecutive likelihoods. The final threshold is found by taking an arbitrary number larger than the largest threshold. Any likelihood larger than the threshold is classified as class two, while any likelihood less than the threshold is classified as class one. Once the classifications are made a confusion matrix can be constructed for each of the thresholds.

Once the vectors relating to the ROC curves have been calculated the area under each curve can be calculated. From the AUC values a comparison can be made between the performance of the different classifying techniques as well as between the performance one-class and two-class classifiers.

3.3 One-class vs. two-class classification for varied class imbalances using the areas under ROC curves

This experiment is designed to probe the effect of the class imbalance on the performance of both one and two-class classifiers.

The same procedure is followed as set out in section 3.2 with one vital exception. Instead of only calculating the areas under the ROC curves for a fixed ratio of defaulters to non-defaulters, the ratio is varied. The ratio is varied by removing a fixed number of defaulters from the data set. Once the first set of defaulters have been removed the areas under the ROC curves are calculated and the next set of defaulters are removed. This process is repeated until the desired class imbalance is reached. The process can be viewed in Figure 2. The AUC values is used to compare the performance of the different one-class classifiers, as well as that of the two-class classifiers.

It is important to notice that as the ratio is varied the size of the data set is also varied.

3.4 One-class vs. two-class classification for varied class imbalances using harmonic means

Since the AUC method of evaluating performance depends on the probability function of the likelihoods of the costs, which in turn depends on the distributions of the actual scores of the classifier, the AUC method isn't considered to be robust [11]. Therefore this experiment investigates the performance of both one and two-class classifiers for various class imbalances using the harmonic mean.

Once ten-fold cross-validation is implemented the likelihood scores for the one-class and two-class classifiers can be calculated as described in the Literature review.

For the two-class classifiers the likelihood scores are multiplied by the corresponding frequentest priors. The predicted classes are determined and a confusion matrix is set up using the predicted and actual classes. The confusion matrices are used to determine the harmonic mean values.

For the one-class classifiers the likelihood scores are determined. Classification is done for various thresholds as described in section 3.2. The optimum thresholds are determined by determining the thresholds that leads to the maximum accuracy for each individual classifier. The relevant confusion matrices are used to determine the harmonic mean.

This process is repeated for multiple class imbalances. The different ratios of defaulters to non-defaulters are obtained as described in section 3.3. This process is set out in Figure 2.The harmonic means of the different classes for the different class imbalances are compared. It is once again important to note that the size of the data set is varied as the class imbalance is varied.

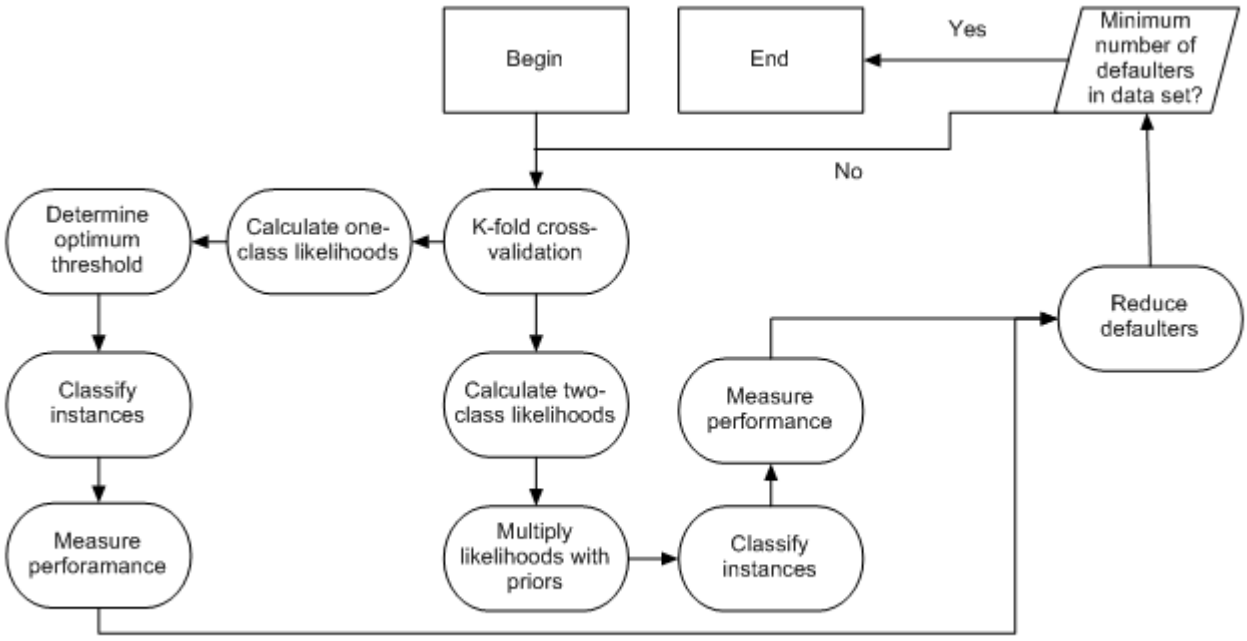


Figure 2: Process used for classification done with varied class imbalances

3.5 One-class vs. two-class classification for varied class imbalances with equal priors

In theory the exact frequentest priors for the two-class classifiers and the exact optimum thresholds for the one-class classifiers can be calculated. In practice it may be required that these values be deduced from historical data or be recommended by an expert in the particular field. This experiment therefore aims to evaluate the effect of frequentest priors that aren't optimal, but realistic in a real world setting, on the performance of the classifiers.

This is done by using equal priors in the calculations to determine the AUC, hit rate and the harmonic mean of the classifiers. These results are then compared to the results where the true priors are known. In other words the procedure set out in section 3.4 is repeated with the changed component of equal priors.

4 Results

4.1 One-class vs. two-class classification for a fixed default rate

4.1.1 Australian data set

For the fixed class imbalance it is clear that the two-class classifiers outperform the one-class classifiers, as seen in Figures 3, 4 and 5. These figures also emphasize the importance PCA plays in the classification performance of the MLE classifier. If the data is only z-scored the one-class MLE classifier's ROC curve is close to the line of no discrimination. This is confirmed by considering the areas for the one-class classifiers summarized in Table 4. The table also indicates that the two-class Silverman classifier performs best regardless whether PCA with all features, PCA with only features explaining 95% of the variance is kept, or only z-scoring is applied to the data. The two-class MLE classifier performs second best for the fixed default ratio.

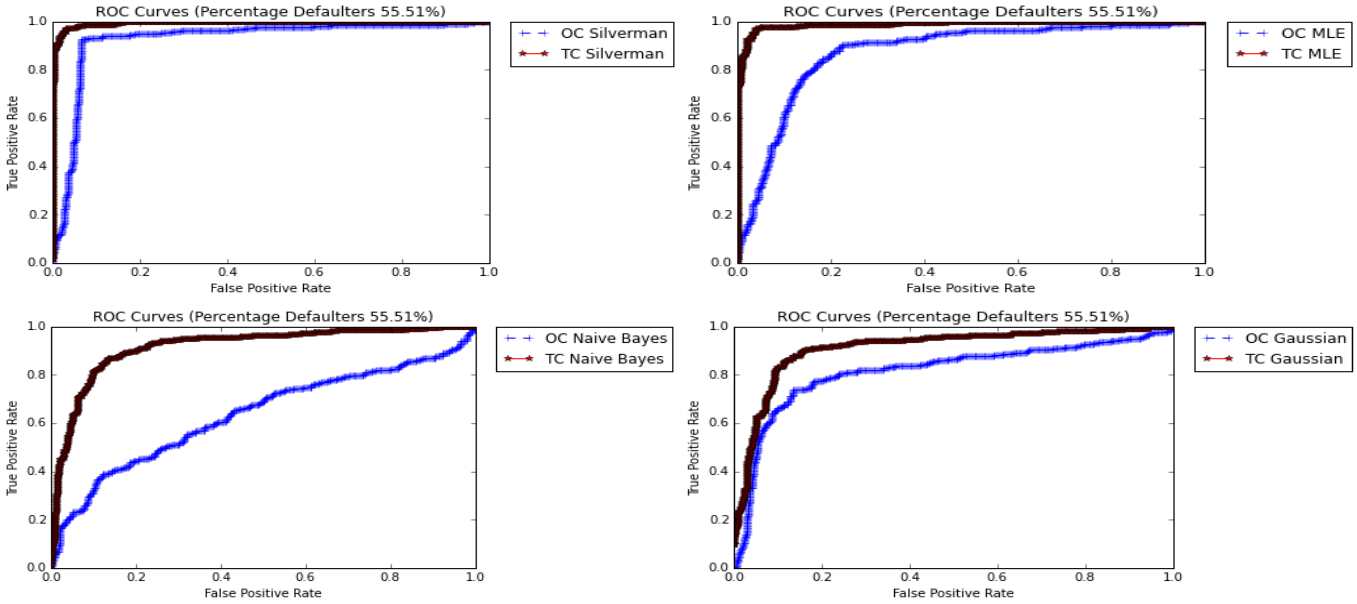


Figure 3: Australian data: ROC Curves (PCA 95%)

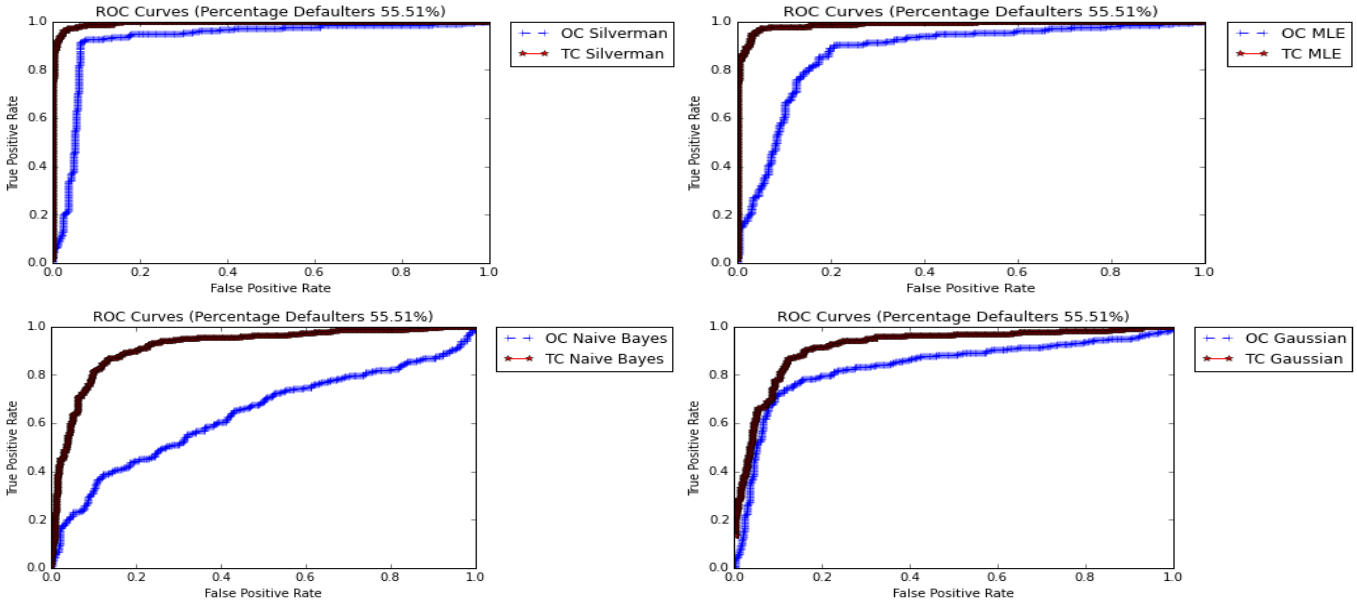


Figure 4: Australian data: ROC Curves (PCA)

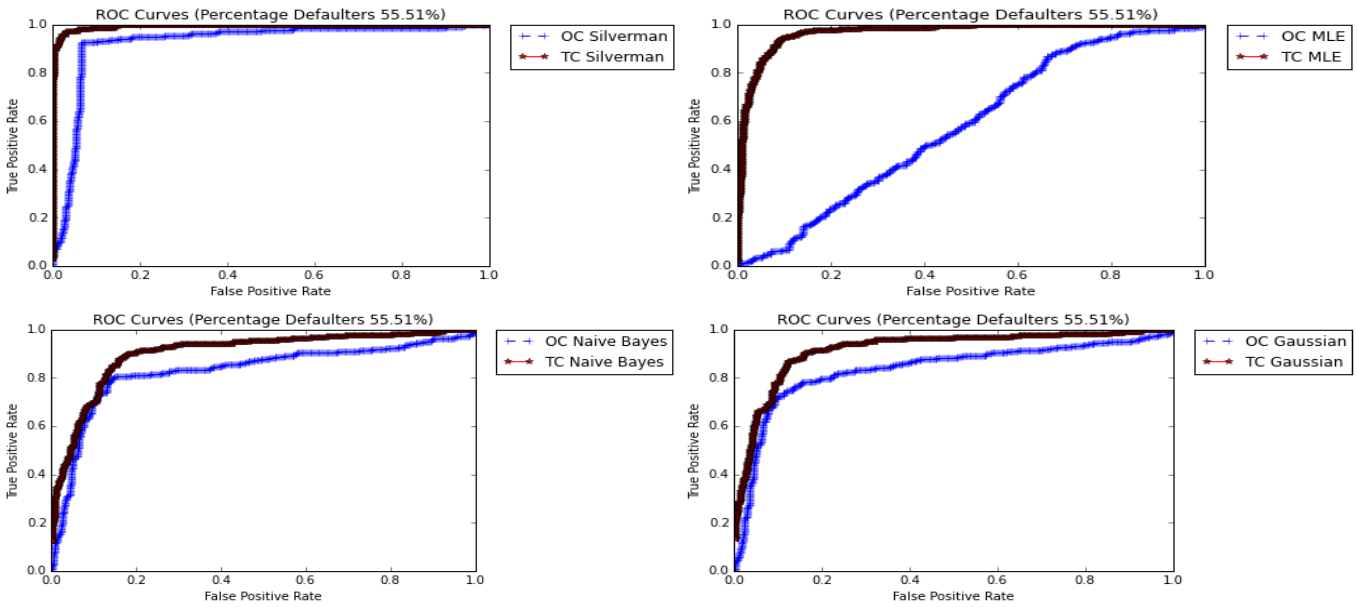


Figure 5: Australian data: ROC Curves (Z-Score)

	PCA 95%		PCA		Z-Score	
	One-Class	Two-Class	One-Class	Two-Class	One-Class	Two-Class
Silverman	**0.9283**	**0.9943**	**0.9269**	**0.9945**	**0.9270**	**0.9942**
MLE	0.8742	0.9882	0.8769	0.9891	0.5825	0.9694
Naive Bayes	0.6366	0.9174	0.6337	0.9157	0.8297	0.9023
Gaussian	0.8170	0.9112	0.8365	0.9167	0.8365	0.9167

Table 4: Australian data: Area Under Curves (55.51% Defaulters)

4.1.2 German data set

Figures 4, 6 and 8 indicate that the two-class classifiers outperform the one-class classifiers. The Naive Bayes and Gaussian one-class classifiers have ROC curves that are close to the line of no discrimination. Table 5 confirms this, indicating that the areas under the one-class ROC curves exceeds that of the line of no discrimination by a small margin in each case. It is interesting to note that the area under the ROC curve of the two-class Silverman classifier with PCA applied is the same as that of the two-class MLE classifier with the data only z-scored. These two classifiers perform the best at this particular default ratio.

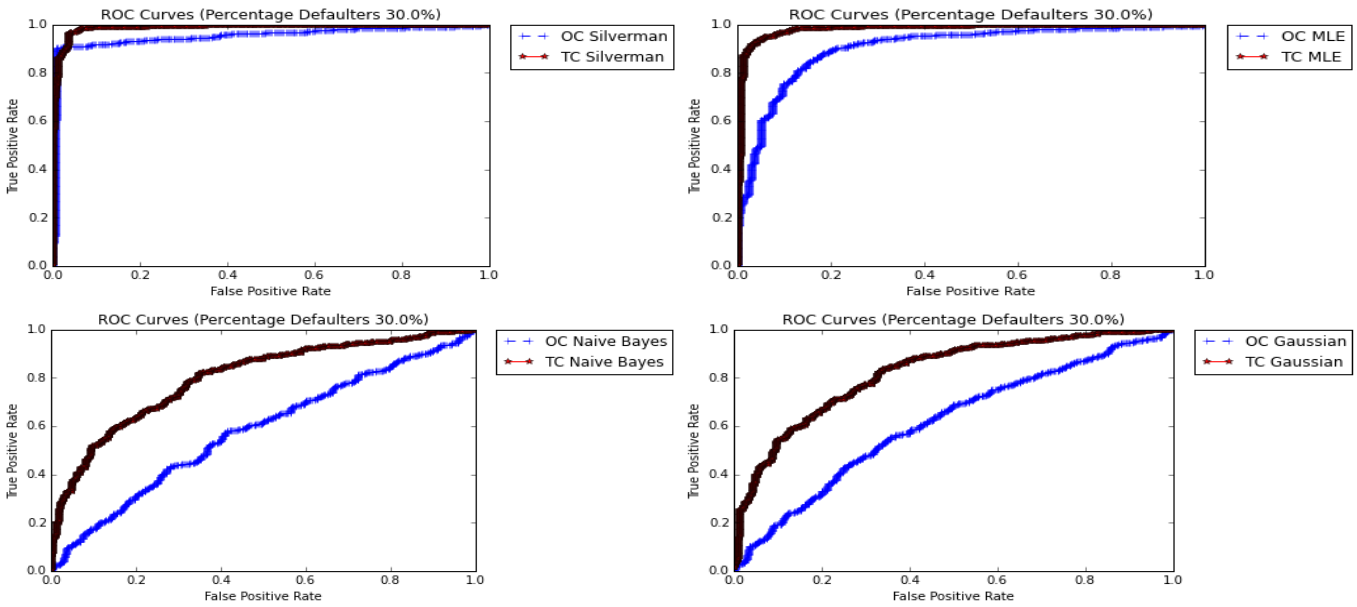


Figure 6: German data: ROC Curves (PCA 95%)

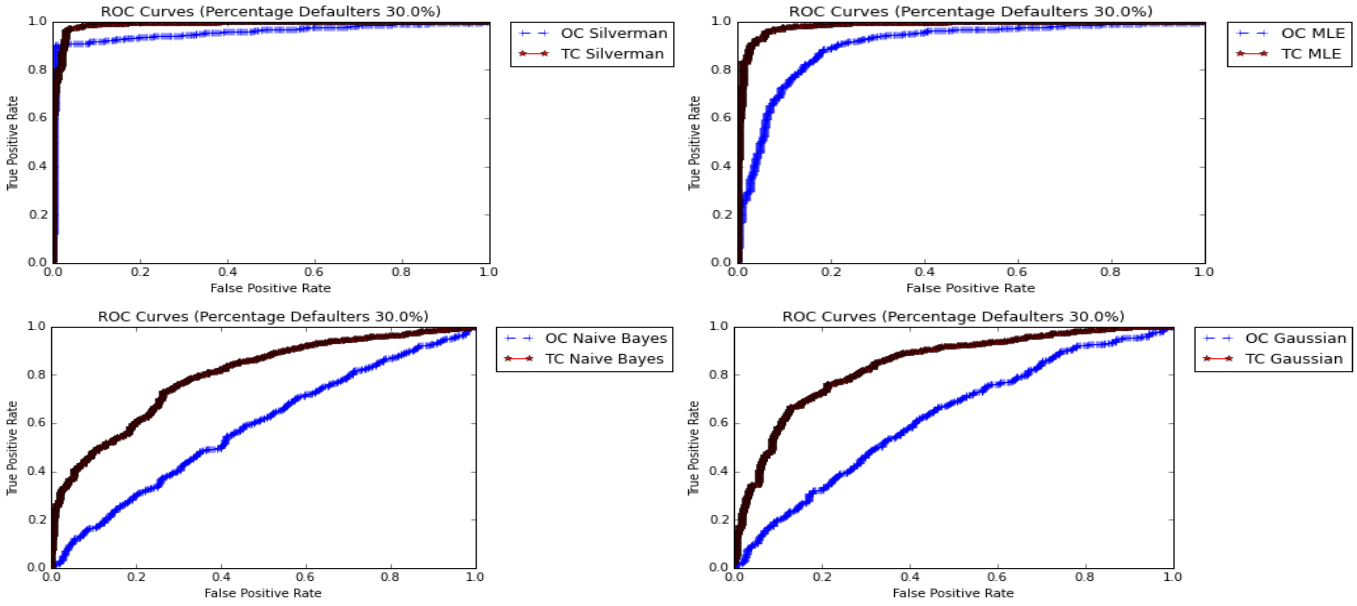


Figure 7: German data: ROC Curves (PCA)

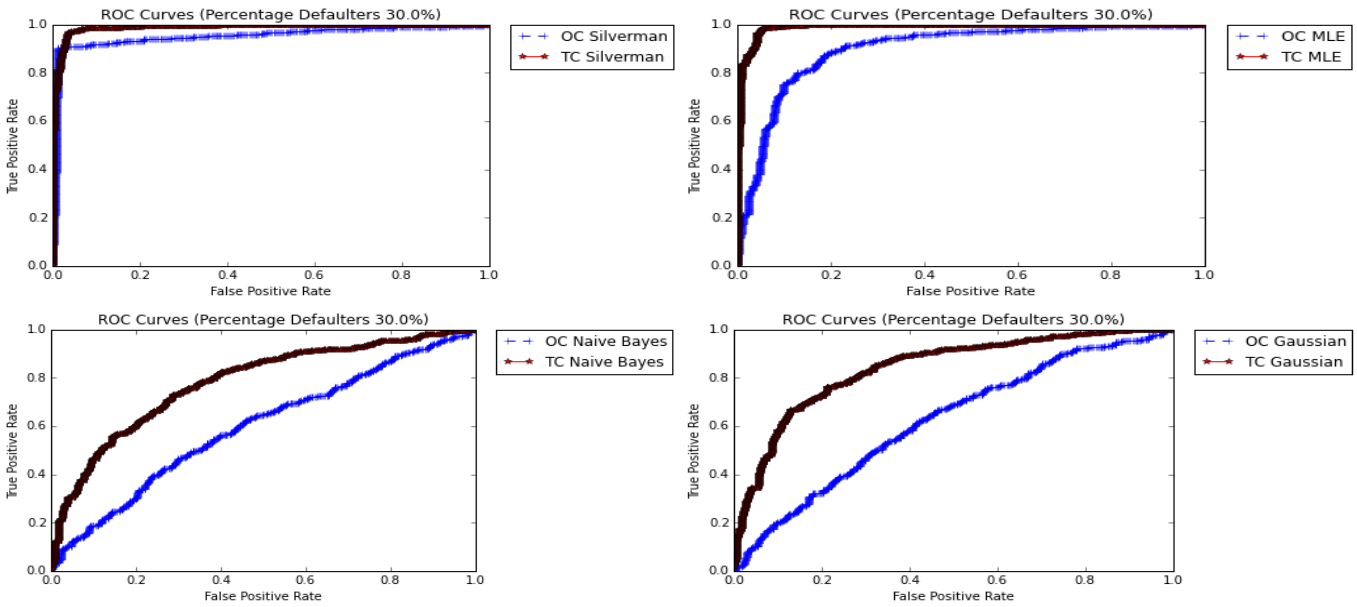


Figure 8: German data: ROC Curves (Z-Score)

	PCA 95%		PCA		Z-Score	
	One-Class	Two-Class	One-Class	Two-Class	One-Class	Two-Class
Silverman	**0.9530**	**0.9896**	**0.9568**	**0.9900**	**0.9527**	0.9893
MLE	0.9061	0.9849	0.9045	0.9847	0.9003	**0.9900**
Naive Bayes	0.5822	0.8005	0.5837	0.7968	0.5969	0.7813
Gaussian	0.6150	0.8219	0.6248	0.8405	0.6248	0.8405

Table 5: German data: Area Under Curves (30% Defaulters)

4.2 One-class vs. two-class classification for varied class imbalances using areas under the ROC curves

4.2.1 Australian data set

Evaluating the performance of the classifiers using the areas under the ROC curves for the data set with PCA applied to it, as seen in Figure 9, a few observations can be made. First of all, the AUC's for the non-parametric two-class classifiers are the largest for all tested class imbalances. These classifiers have areas above 0.95 regardless of the class imbalance.

The AUC's for the two-class parametric classifiers are similar, with the two-class Naive Bayes and Gaussian classifiers increasing in performance as the class imbalance is increased. For the larger class imbalances the two-class Naive Bayes classifier performs better than the two-class Gaussian classifier. At an imbalance of 6.97% the performance of the two-class Gaussian classifier increase to such an extend that it surpasses that of the two-class Naive Bayes classifier. The logistic regression classifier performs similar to the two-class Naive Bayes classifier, slightly outperforming it for lower levels of defaulters.

All of the one-class classifiers have a noticeable spike in performance occurring between default ratios 11.43% and 6.97%. The one-class Naive Bayes classifier has a significant upwards trend. The non-parametric one-class classifiers performs better than the parametric one-class classifiers, with Silverman's one-class classifier performing similar to the parametric two-class classifiers.

Comparing Figures 9 and 10 it is quite clear that the removal of features, that do not contribute to 95% of the variance, have very little to no effect on the calculated AUC's for various class imbalances. The one-class

Gaussian classifier does perform better with a small margin, when all features are kept.

By considering Figure 11 the effect of the lack of PCA can be observed on the AUC's of both the one-class and two-class MLE classifiers. Applying PCA to the data results in the features, in the transformed feature space, being orthogonal. The data is thus uncorrelated; explaining the higher performance of the one-class, as well as of the two-class MLE classifiers for the PCA data.

By applying PCA on the data the AUC's for the one-class Naive Bayes classifier are reduced in comparison to when the data is only z-scored.

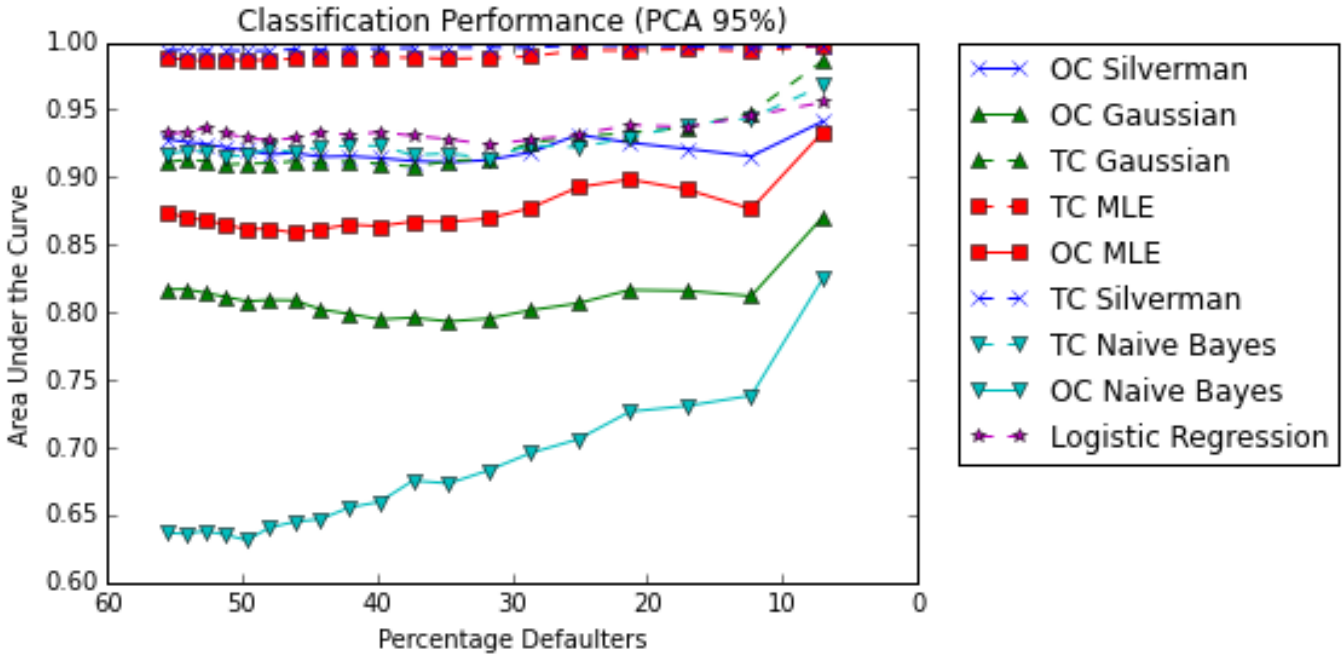


Figure 9: Australian data: Area Under the Curve (PCA 95%)

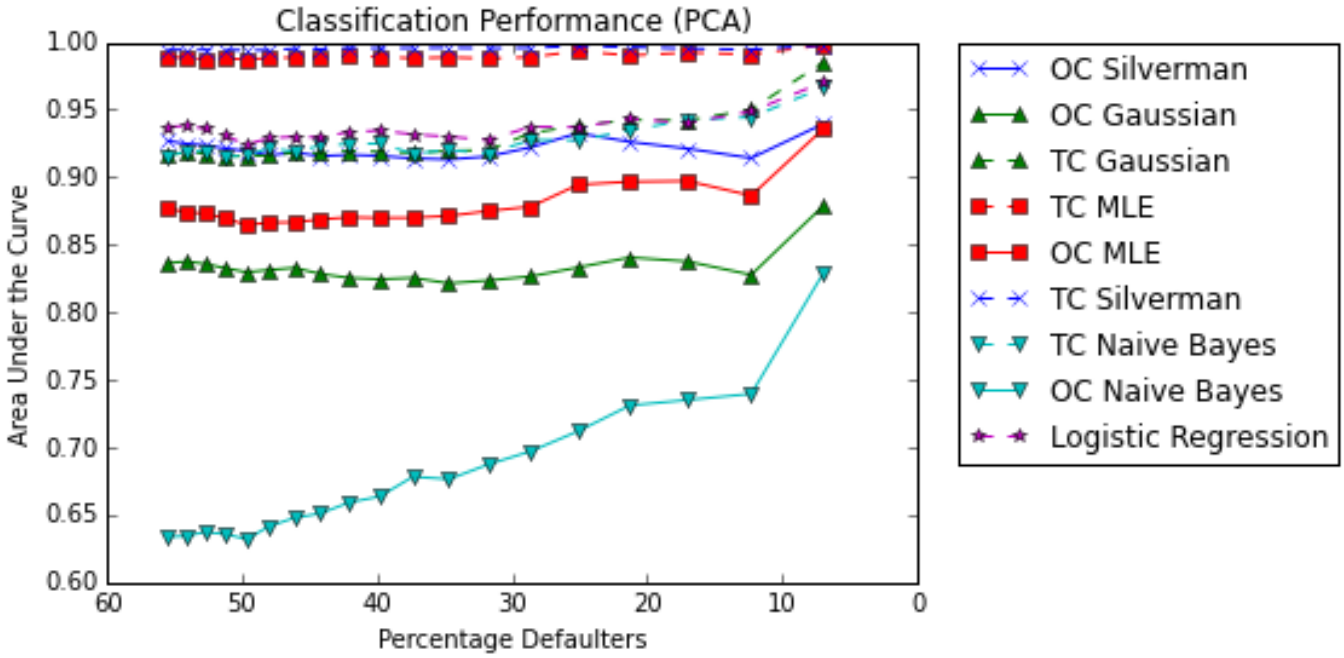


Figure 10: Australian data: Area Under the Curve (PCA)

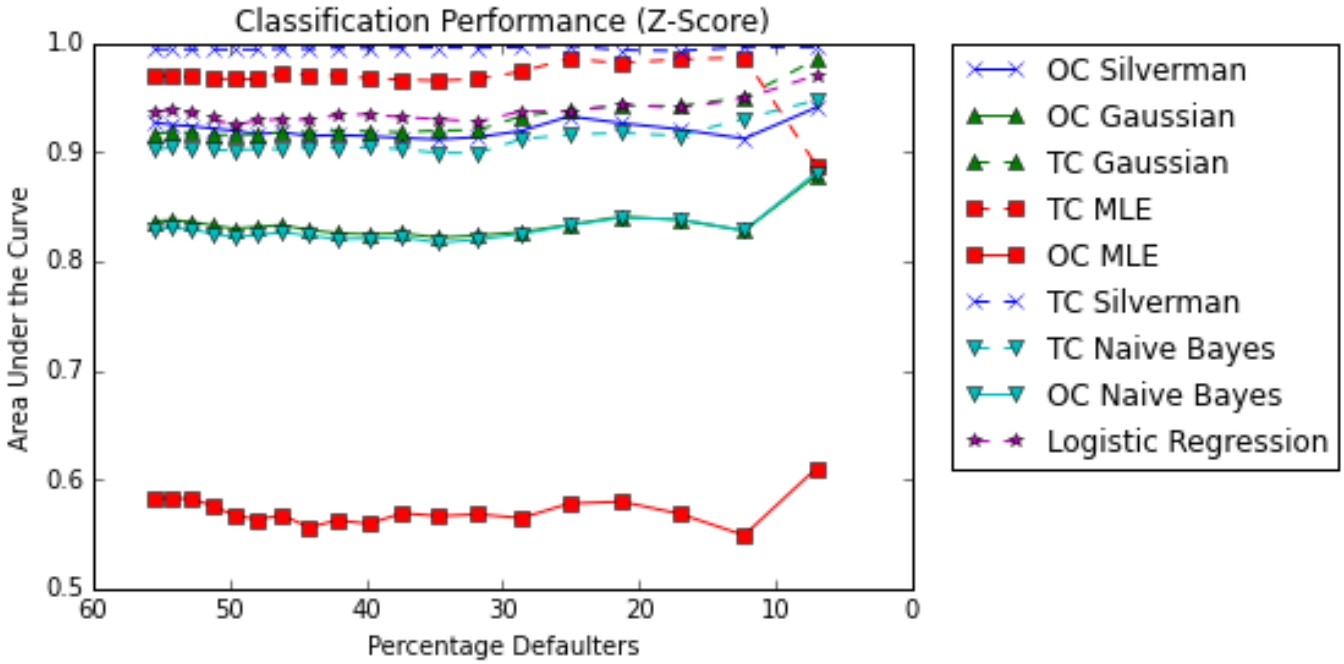


Figure 11: Australian data: Area Under the Curve (Z-Score)

4.2.2 German data set

First and foremost it is clear that the two-class non-parametric classifiers outperform all other classifiers over all class imbalances. All of the two-class classifiers outperform the one-class classifiers. The one-class Silverman classifier outperforms not only all of the other one-class classifiers, but also the two-class parametric classifiers. The AUC's for this classifier remain fairly constant at 0.95 over all class imbalances. The one-class MLE classifier outperforms the parametric two-class classifiers up to a point of 10.26% defaulters, at which point the two-class Gaussian classifier starts to exceed it in performance.

The two-class Gaussian classifier increases in performance as the percentage of defaulters is decreased. It outperforms the logistic regression, as well as the two-class Naive Bayes classifiers as expected. The Naive Bayes classifier assumes independence and only takes into account the variances of each feature, whereas the Gaussian classifier not only takes the variances, but also the covariances of the features into account.

The AUC's of the two-class Naive Bayes classifier increase slightly but remains fairly unchanged at 0.8. The performance of the logistic regression classifier is similar to that of the two-class Naive Bayes classifier. Up to a point of 20.45% defaulters the logistic regression classifier performs slightly better, after which the two-class Naive Bayes classifier performs slightly better.

Both the one-class Naive Bayes and Gaussian classifiers decrease in performance as the class imbalance increase. As expected the one-class Gaussian classifier still outperforms the one-class Naive Bayes classifier. The AUC's remain between 0.65 and 0.55 for all class imbalances. This can be observed in Figure 12.

The performance of the classifiers when PCA is applied to the data, with all features kept, is similar to that when only the features explaining 95% of the variance is kept. However in the case where all features are kept the performance of the one and two-class Gaussian as well as the one-class Naive Bayes classifier is slightly better. See Figure 13. Since the Gaussian classifier use a covariance matrix, by not using all the features and hence not all the variance and covariance components its classification accuracy is reduced. A similar argument can be made for the one-class Naive Bayes classifier.

By comparing Figures 12 and 14 a few differences can be observed. The two-class MLE classifier has a smaller AUC for a ratio of 5.41% defaulters if PCA isn't applied to the data. The two-class Naive Bayes classifier performs somewhat worse up to the point of 10.26% defaulters, after which the classifier performs similarly for both cases.

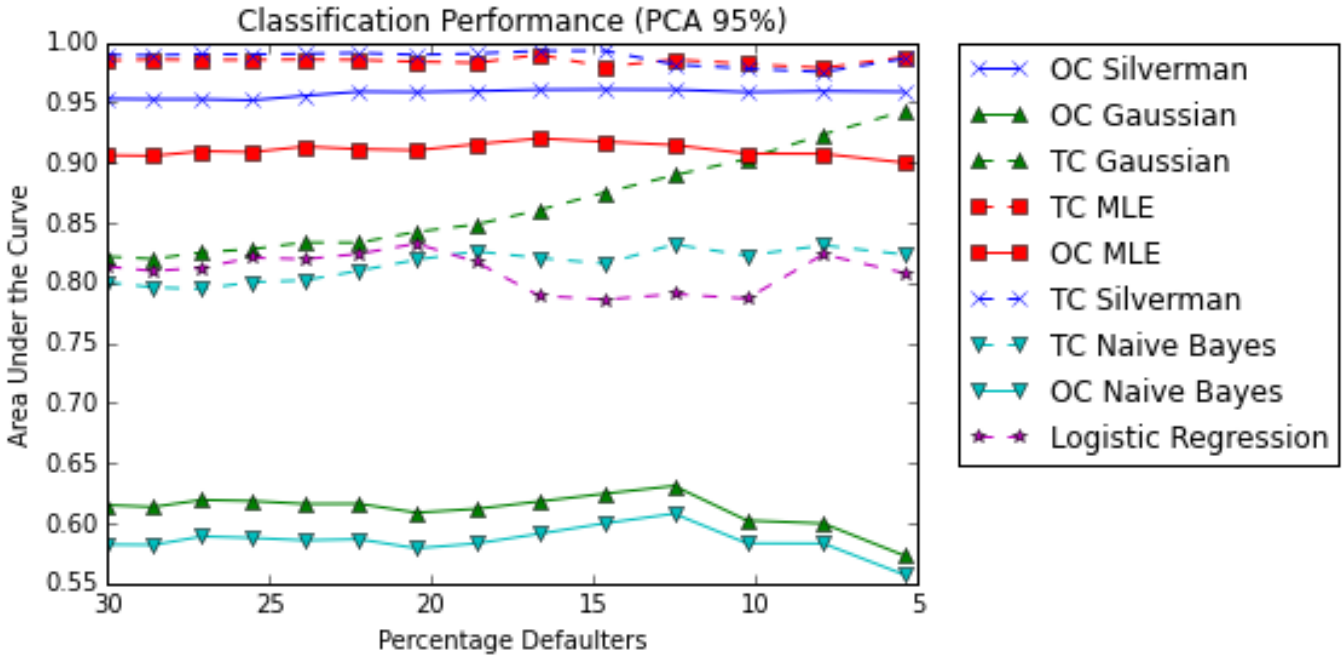


Figure 12: German data: Area under Curve (PCA 95%)

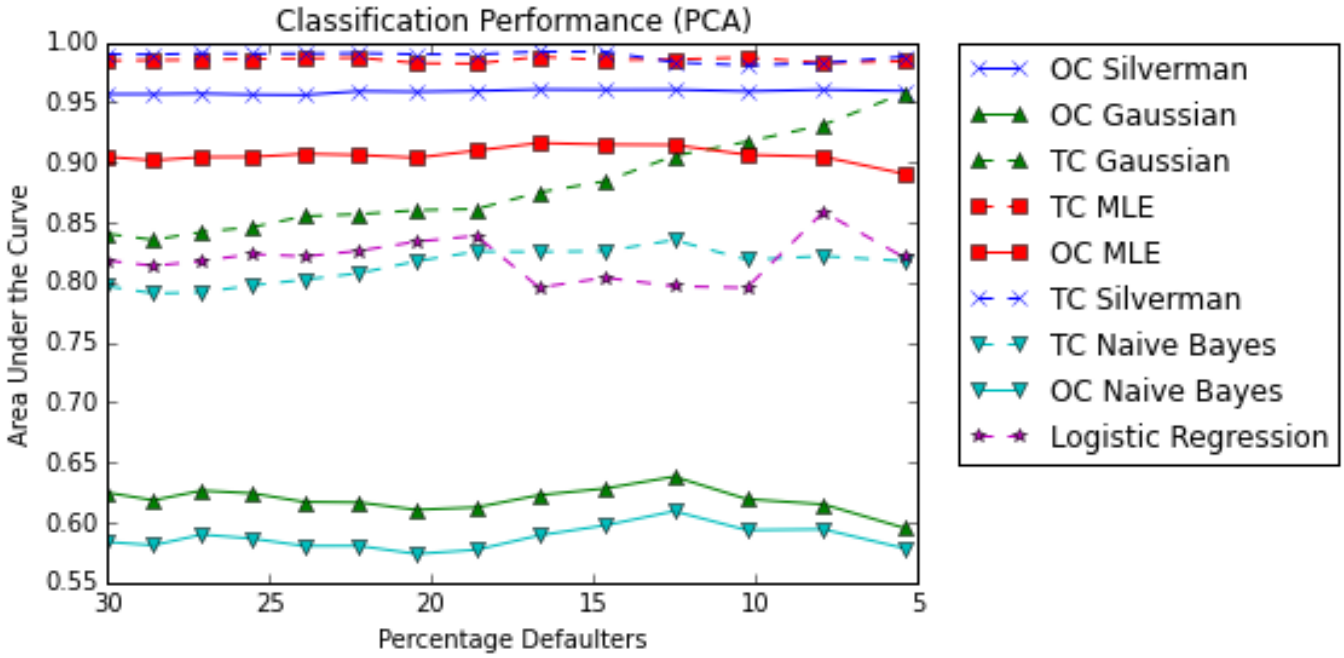


Figure 13: German data: Area under Curve (PCA)

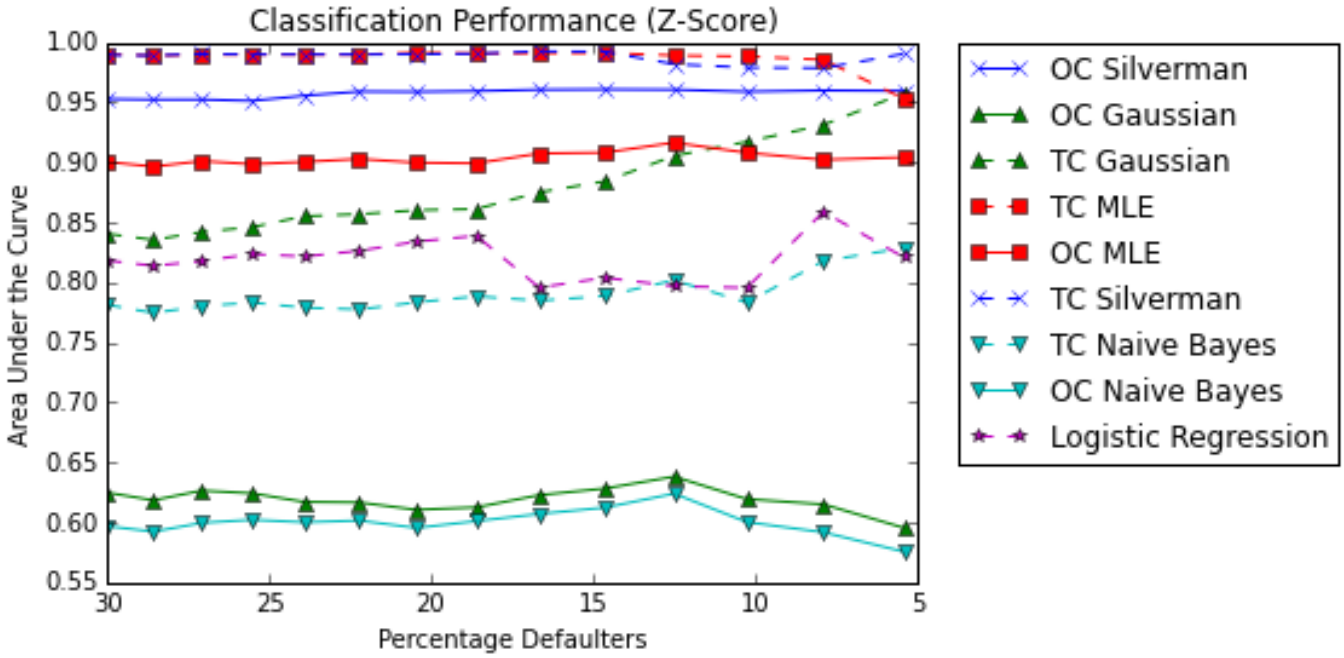


Figure 14: German data: Area under Curve (Z-Score)

4.3 One-class vs. two-class classification for varied class imbalances using hit rates

The series indicated as "Chance Criterion" in the following figures serves as a benchmark and is based on the proportional chance criterion. The series "Chance Criterion" indicates one and a quarter times the proportional chance criterion.

4.3.1 Australian data set

Figure 15 shows that for the data on which PCA is performed with the features explaining 95% of the variance kept, the hit rates of the non-parametric two-class classifiers are the highest. The two-class MLE classifier exceeds the two-class Silverman classifier for greater class imbalances. The two-class Naive Bayes and Gaussian classifiers have upward trends, with the Naive Bayes classifier exceeding the Gaussian classifier in performance. All of the two-class classifiers have higher hit rates than the one-class classifiers up to a class imbalance of 6.97% defaulters. The one-class classifiers all fall below the benchmark. The hit rate of the one-class classifiers is an increasing function of the class imbalance, with values similar to the proportion of the class containing the most instances.

Fairly similar conclusions can be made for the case where all the features are kept and not just those explaining 95% of the variance. See Figure 16.

Figure 17 once again emphasize the importance of PCA for the two-class MLE classifier. In comparison to the PCA data the hit rate of the two-class MLE classifier is much lower for the z-scored data. The two-class Silverman classifier has a lower hit rate for smaller class imbalances, however it increases as the as the class imbalance increase. For a default ratio of 6.97% the two-class Silverman classifier performs better for the z-scored data compared to the PCA data.

The overall performance of the two-class Naive Bayes classifier is adversely affected by only z-scoring the data. This contradicts the observation made in section 4.2. The contradiction can be explained by the fact that the hit rate only incorporates the instances correctly classified, whereas the AUC takes into account all instances.

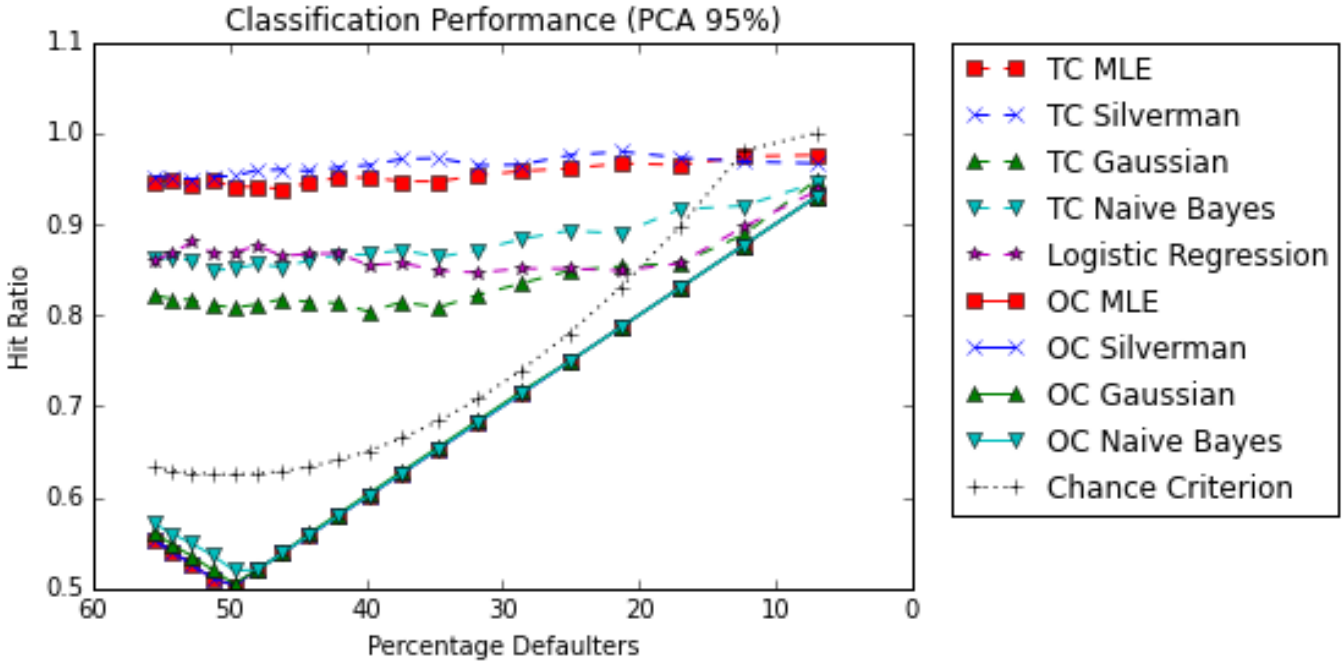


Figure 15: Australian data: Hit Ratio (PCA 95%)

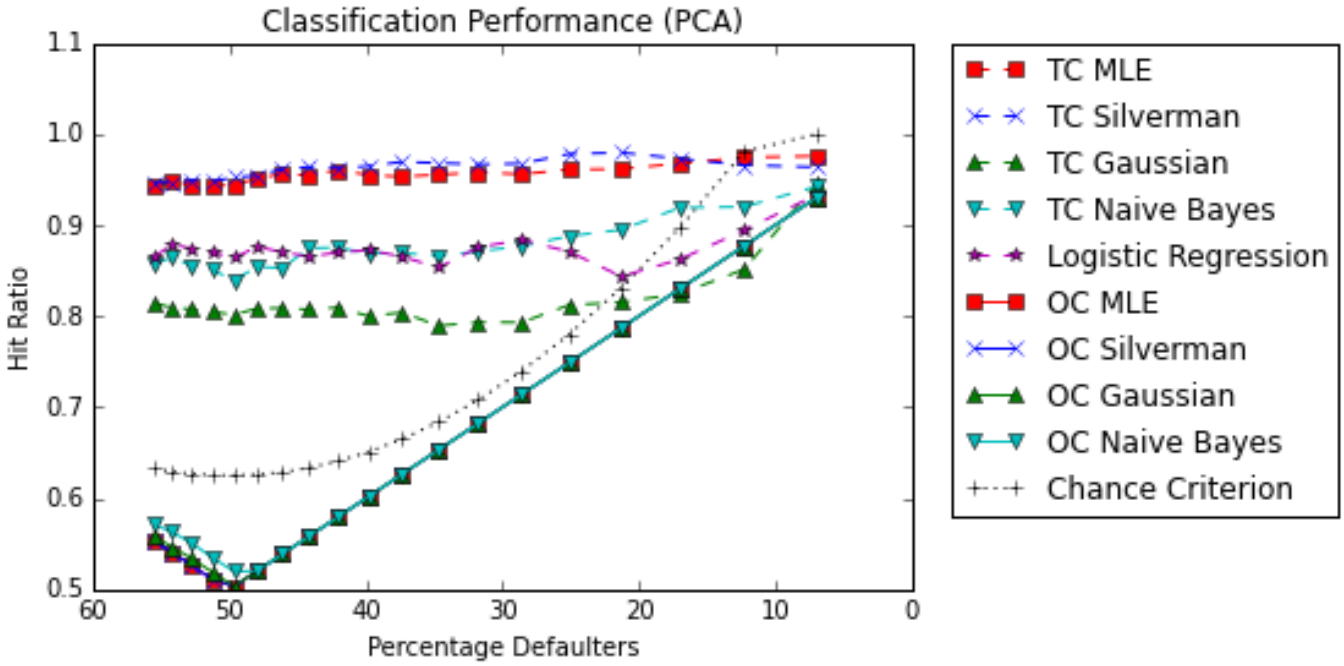


Figure 16: Australian data: Hit Ratio (PCA)

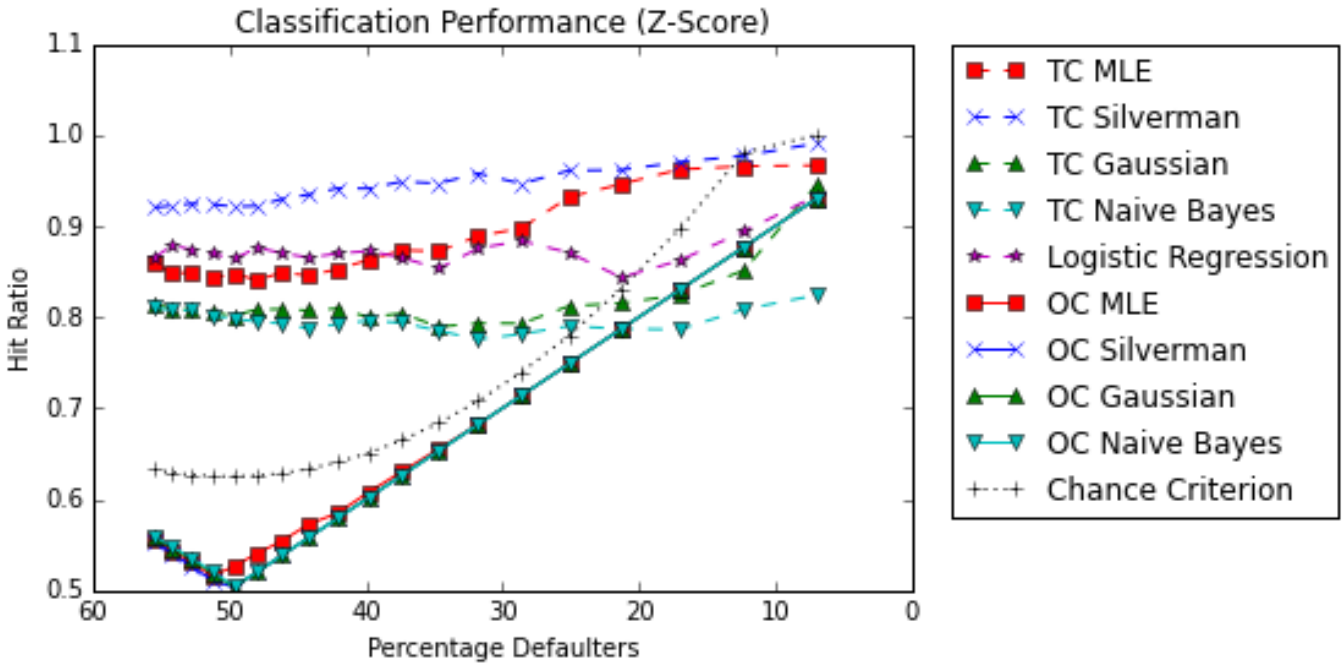


Figure 17: Australian data: Hit Ratio (Z-score)

4.3.2 German data set

The one-class classifiers never exceed the benchmark and can be considered to under perform, regardless of the class imbalance. The hit rate of the one-class classifiers are similar to the percentage of non-defaulters.

The logistic regression classifier only performs acceptable for default ratios of 30% and 28.57% of defaulters. It follows a similar upward trend as that of the one-class classifiers. The parametric two-class classifiers have an acceptable hit rate up to a class imbalance of 23.91% of defaulters after which it falls below the benchmark.

The hit rate of the one-class classifiers at any given default ratio is the ratio of the non-defaulting class. Therefore as the class imbalance is increased and the proportion of the non-defaulters increase, the hit ratio of the one-class classifiers increase.

The two-class Silverman classifier slightly outperforms the two-class MLE classifier for most of the class imbalances. The hit ratio of the two-class MLE classifier increases for 7.89% and 5.41% defaulters such that it exceeds that of Silverman. From a ratio of 12.5% defaulters both non-parametric two-class classifiers fall below the benchmark.

Comparing Figures 18 and 19 it can be seen that by only maintaining the features explaining 95% of the variance have a very small impact on the hit rate of the classifiers. In fact the hit rate of the two-class Gaussian classifier is slightly higher if all the features are kept.

Figure 20 indicates that by not performing principal component analysis and only z-scoring the data that the hit rate of the two-class MLE is higher. On the other hand, the hit rate of the two-class Naive Bayes classifier is drastically lower. From a default ratio of 22.22% it is lower than the hit ratio of the one-class classifiers.

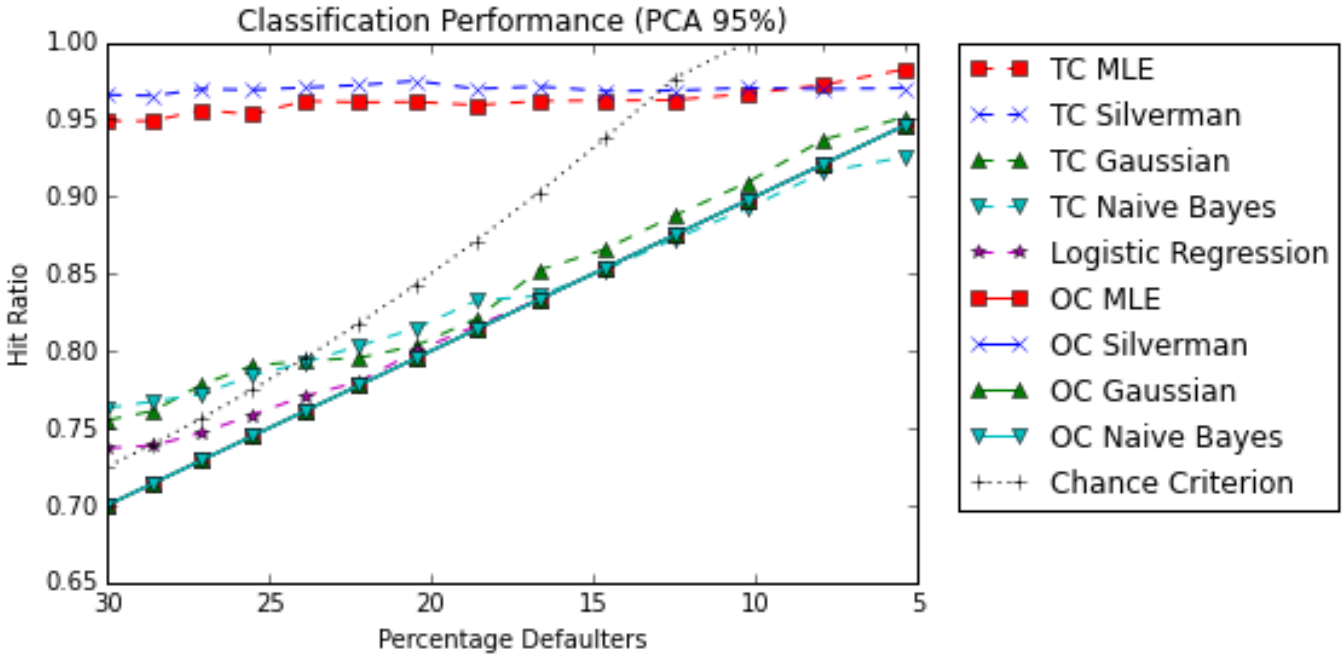


Figure 18: German data: Hit Ratio (PCA 95%)

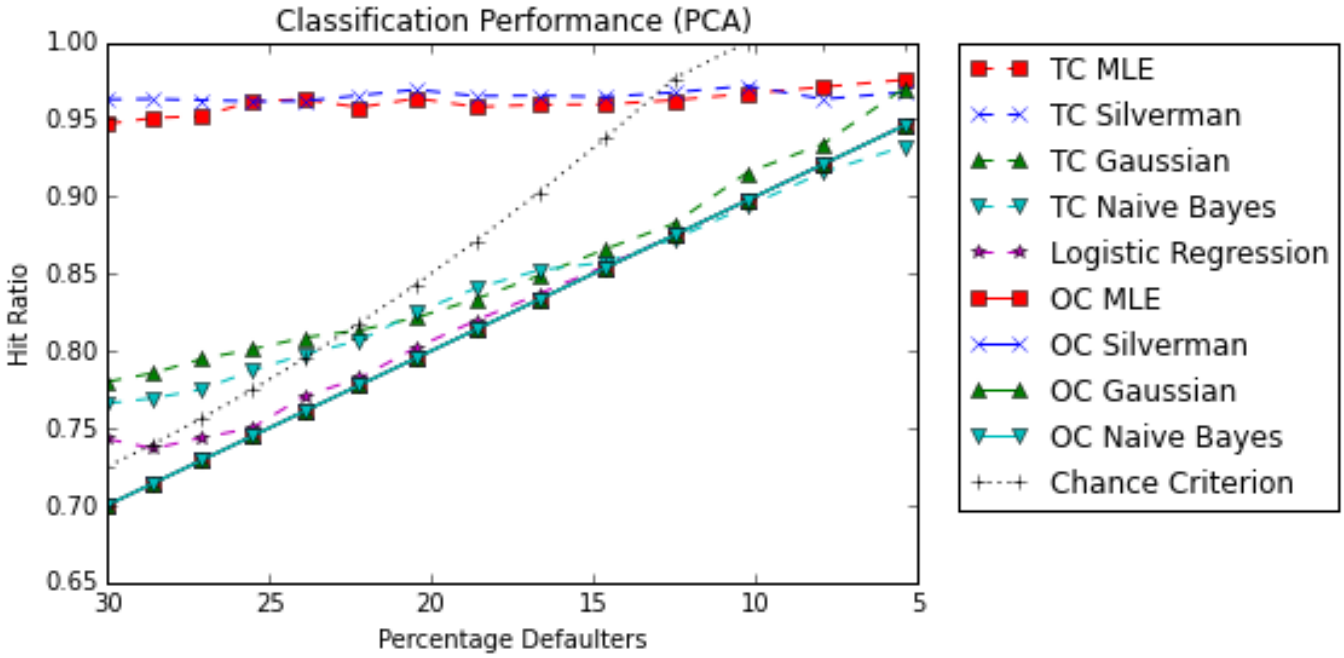


Figure 19: German data: Hit Ratio (PCA)

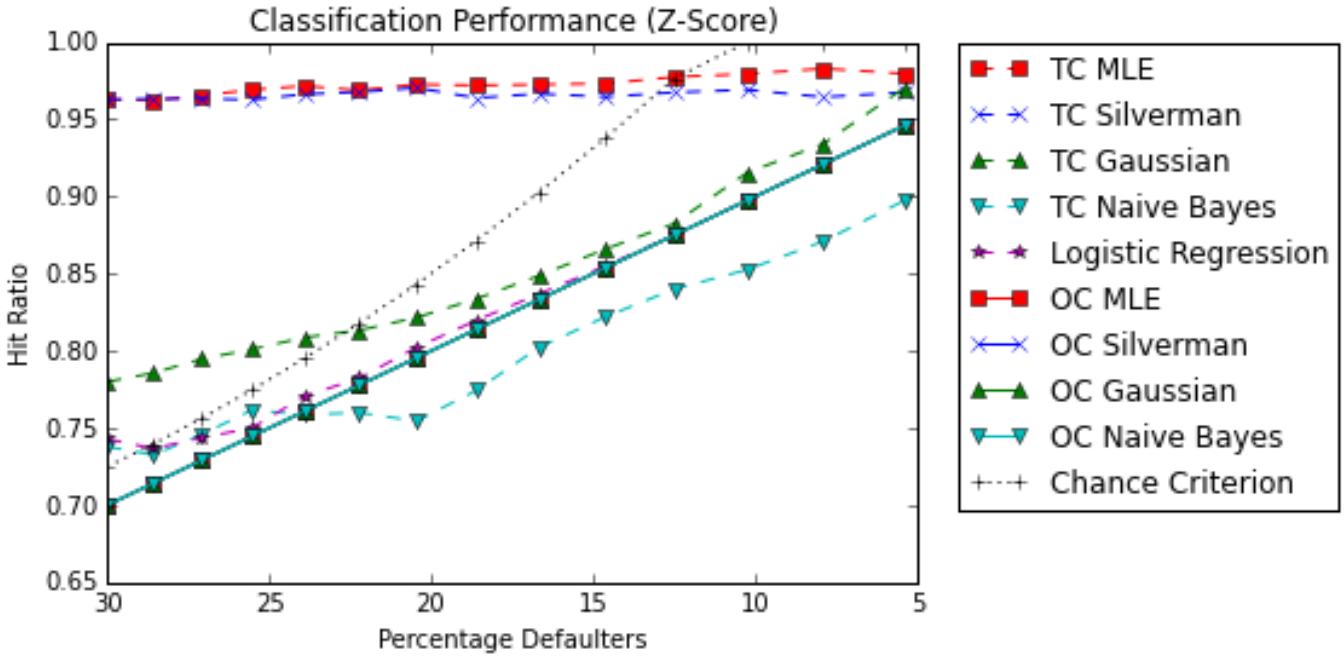


Figure 20: German data: Hit Ratio (Z-Score)

4.4 One-class vs. two-class classification for varied class imbalances using harmonic means

4.4.1 Australian data set

For the data with PCA applied to it and only the most significant features kept, the harmonic mean of the two-class non-parametric classifiers are the highest for small class imbalances. As the class imbalance is increased the harmonic mean of the non-parametric two-class classifiers start to decrease. The rate at which the harmonic mean of the two-class MLE decrease is much lower than that of the two-class Silverman classifier. This is due to the change in the sensitivities and specificities of the classifiers. Although the sensitivity of the two-class MLE classifier increase from 0.9055 at a ratio of 55.51% defaulters to 0.9934 at 6.97% defaulters, the specificity decrease from 0.9791 to 0.7391. The sensitivity of the two-class Silverman classifier increases from 0.9088 at 55.51% defaulters to 1 at 6.97% defaulters, whereas the specificity decreases from 0.9869 to 0.5217. The larger drop in the specificity of the Silverman classifier results in the MLE outperforming the Siverman classifier for larger class imbalances.

The harmonic mean two-class Gaussian and Naive Bayes classifiers increase as the class imbalance is increased, to such an extent that they outperform the non-parametric classifiers for a class imbalance of 6.97% defaulters. The increase in the harmonic mean of the Gaussian classifier is caused by a drastic increase in the sensitivity and a slight increase in the specificity. The sensitivity of the two-class Naive Bayes classifier also increases, but the specificity decreases slightly. Therefore the two-class Gaussian classifier outperforms the one-class Naive Bayes classifier for large class imbalances.

The harmonic mean of the logistic regression classifier deteriorates very quickly as the class imbalance is increased. This is caused by the deterioration of the specificity, as the ratio of defaulters is decreased.

All of the one-class classifiers, except the one-class Naive Bayes classifier, have harmonic means of zero over all the default ratios. Up to a class imbalance of about 50% the sensitivities of the classifiers are high. Once this class imbalance is exceeded, the sensitivities of the one-class classifiers decrease to a value close to zero and the specificities increase to a value close to one or one itself. The higher harmonic mean values of the one-class Naive Bayes classifier compared to the other one-class classifiers, for the first five class imbalances, is caused by a higher specificity.

The harmonic mean of the parametric two-class classifiers remain similar regardless whether all features are kept or only those features explaining 95% of the variance. The harmonic means of the non-parametric classifiers are slightly lower in comparison for the PCA data. The rate at which the logistic regression classifier deteriorates is also higher for the PCA data. See Figure 22.

If the data is only z-scored, as in Figure 23, then the harmonic means of the two-class MLE classifier are lower compared to when PCA is applied to the data. The harmonic mean of the two-class MLE classifier drastically decrease over the interval 17.03% to 6.97% defaulters. This is caused by a much lower specificity for the z-scored data in comparison to the PCA data. The harmonic mean of the two-class Silverman classifier doesn't decrease for the large class imbalances if the data is only z-scored. Overall the specificity and sensitivity of the classifier increase as the class imbalance increase.

Over the interval 49.67% to 37.35% defaulters a peak in the one-class MLE classifier can be observed in Figure 23. This peak can be explained by a spike in the specificity.

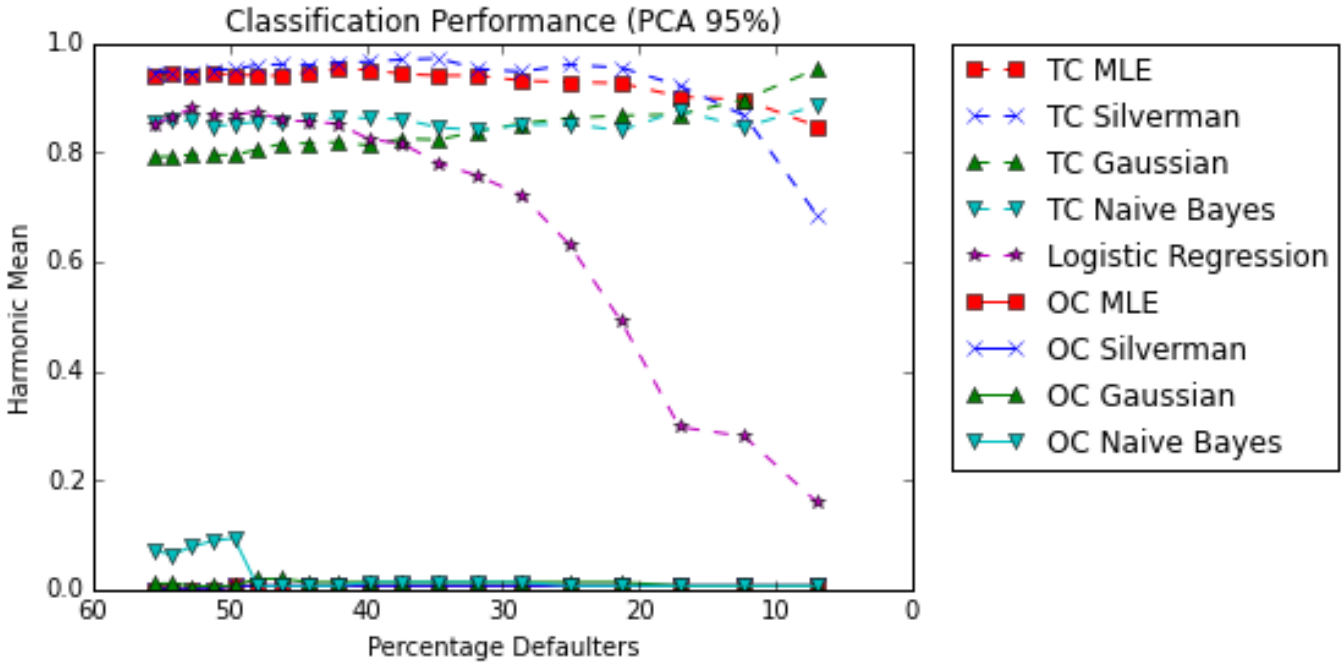


Figure 21: Australian data: Harmonic Mean (PCA 95%)

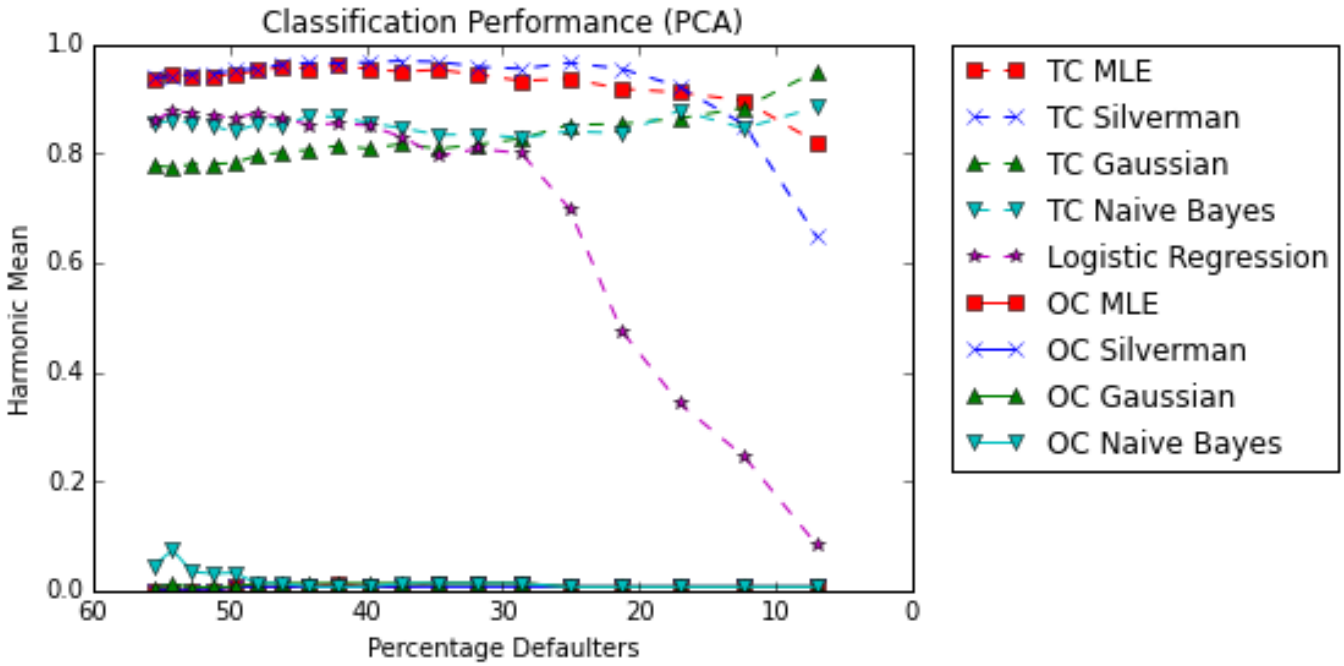


Figure 22: Australian data: Harmonic Mean (PCA)

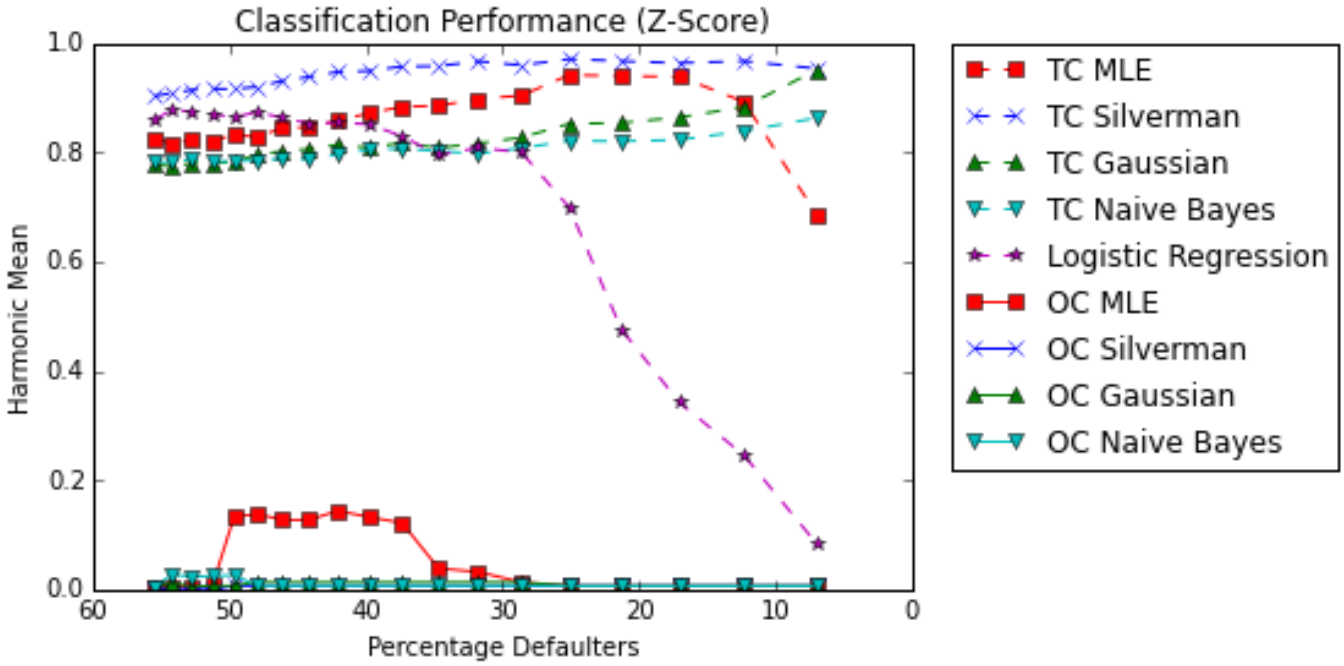


Figure 23: Australian data: Harmonic Mean (Z-Score)

	Sensitivity		Specificity	
	55.51% Defaulters	6.97% Defaulters	55.51% Defaulters	6.97% Defaulters
Two-class Silverman	0.9088	**1**	**0.9869**	0.5217
Two-class MLE	0.9055	0.9934	0.9791	0.7391
Two-class Gaussian	0.6906	0.9479	0.9269	0.9565
Two-class Naive Bayes	0.8208	0.9544	0.8956	0.8261
Logistic Regression	0.8046	**1**	0.9034	0.0870
One-class Silverman	**1**	0.00326	0	**1**
One-class MLE	**1**	0.00326	0	**1**
One-class Gaussian	0.9837	0.00326	0.0052	**1**
One-class Naive Bayes	0.9218	0.00326	0.0366	**1**

Table 6: Australian data: Sensitivity and Specificity (PCA 95%)

4.4.2 German data set

The harmonic mean of the two-class MLE classifier deteriorates slightly at a consistent rate. However, it seems to stabilize and remain constant for a class imbalance of 7.89% and 5.41% defaulters. The deterioration is caused by a significant drop in the specificity of the classifier. The sensitivity on the other hand is higher for greater class imbalances as seen in Table 7.

The two-class Silverman classifier acts in a similar fashion but instead of stabilizing, like the MLE, at an imbalance of 7.89% its rate of deterioration increases. This results in it having a lower harmonic mean than the two-class Gaussian classifier for a ratio of 5.41% defaulters. This is due to a large decrease in the specificity of the two-class Silverman classifier. Although the sensitivity increases, it doesn't increase as much as that of the two-class MLE classifier.

The two-class Gaussian classifier remains fairly constant with a sudden upward spike occurring at 5.41% defaulters. This spike is explained by an increase, from the previous default ratio, of 0.1833 in the specificity

of the classifier.

The two-class Naive Bayes classifier shows an overall deterioration in the harmonic mean as the class imbalance is increased, with slight fluctuations occurring at larger class imbalances. Even though the sensitivity of the classifier is higher than that of the Gaussian classifier, its specificity is considerably lower than that of the Gaussian classifier.

The harmonic mean of the logistic regression classifier is much lower than all of the two-class classifiers. It deteriorates quite quickly, resulting in a harmonic mean of zero before a class imbalance 15% defaulters is reached. The sensitivity of the logistic regression classifier is very high, but its specificity is very low.

All of the one-class classifiers have an harmonic mean of zero for all the class imbalances investigated. This is due to a very low level of sensitivity in comparison to the level of specificity, indicating the proportion of positive instances correctly classified is very small. In other words the proportion of non-defaulters correctly classified as non-defaulting is very low. See Table 7.

Maintaining all the features results in lower harmonic means at greater class imbalances for the non-parametric classifiers, as seen by comparing Figures 21 and 22. It results in more consistency for the two-class Gaussian classifier and a lower overall rate of deterioration of the harmonic mean for the logistic regression classifier. It can also be observed that the removal of features results in an overall greater deterioration in the harmonic mean for the two-class Naive Bayes classifier.

Figure 23 shows by only z-scoring the data, the overall performance, in terms of the harmonic mean, of the two-class Naive Bayes classifier is better, whereas that of the two-class non-parametric classifiers are considerably worse for greater class imbalances.

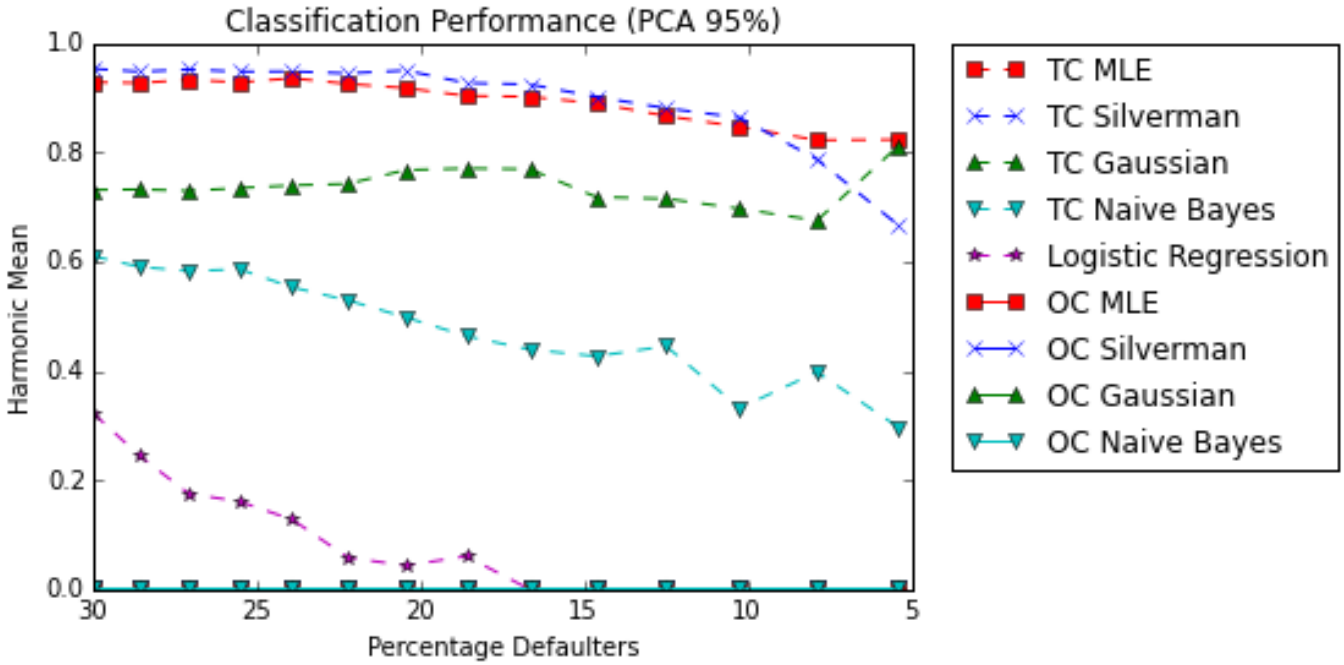


Figure 24: German data: Harmonic Mean (PCA 95%)

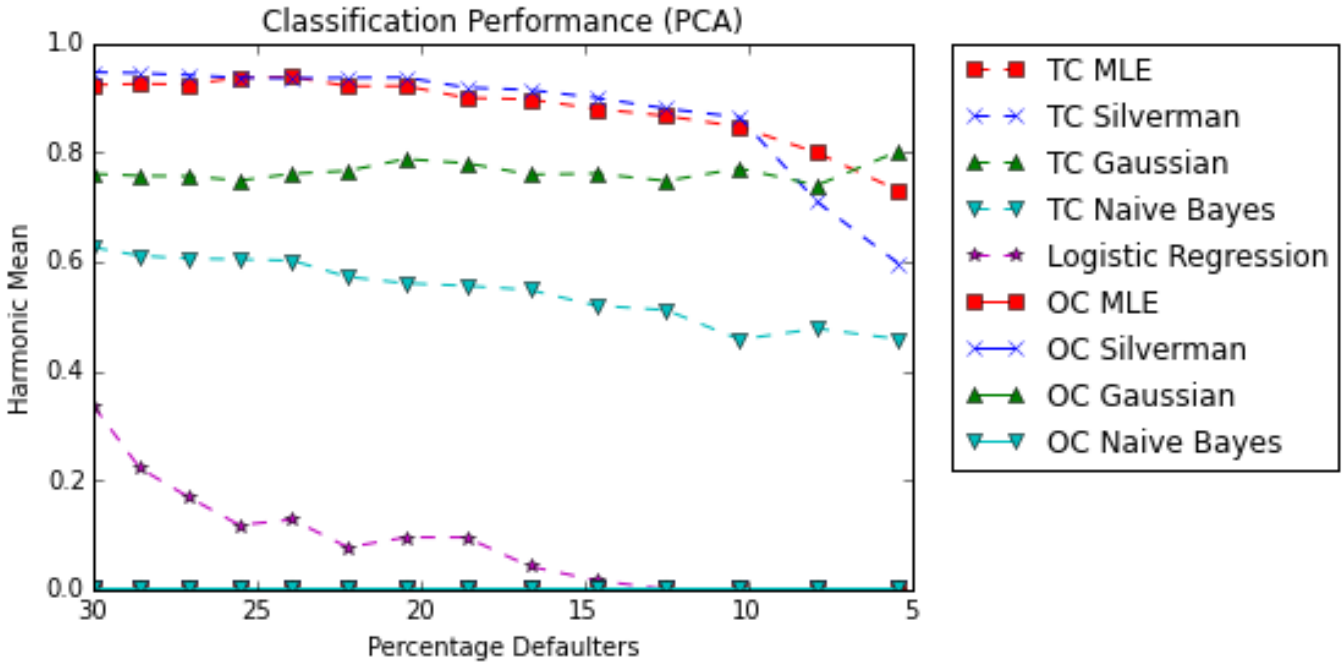


Figure 25: German data: Harmonic Mean (PCA)

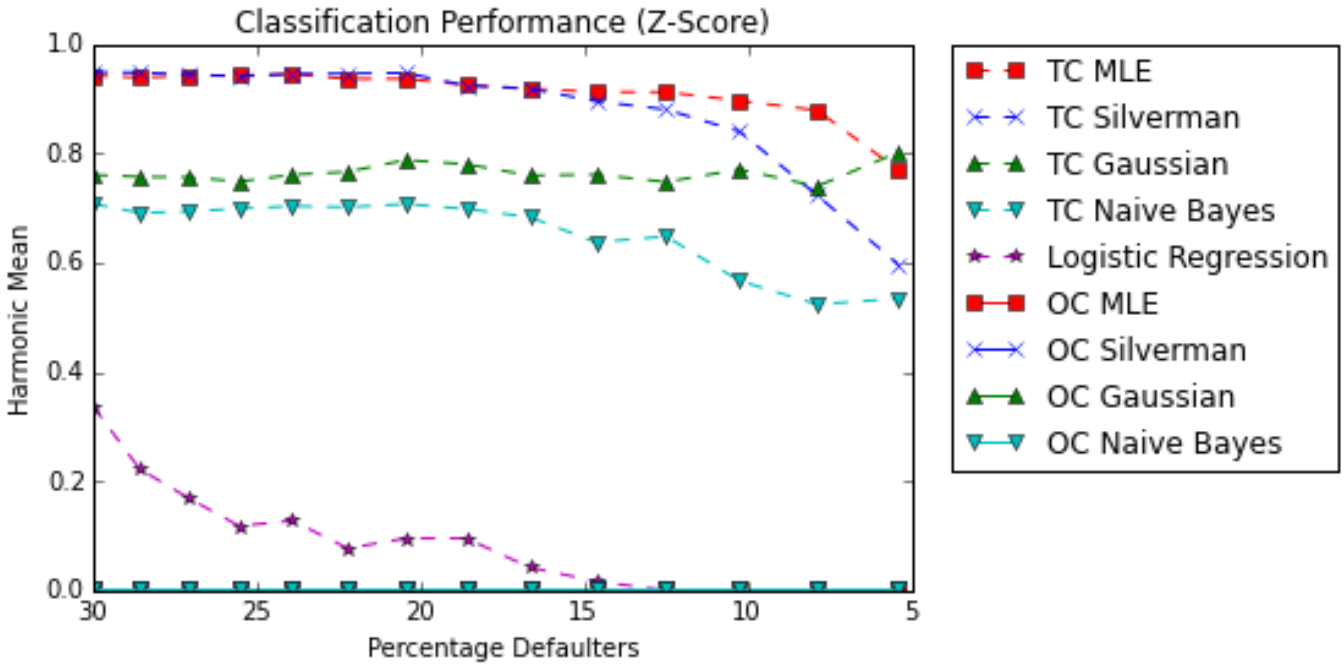


Figure 26: German data: Harmonic Mean (Z-Score)

	Sensitivity		Specificity	
	30% Defaulters	5.41% Defaulters	30% Defaulters	5.41% Defaulters
Two-class Silverman	**0.9843**	0.9971	0.9233	0.5000
Two-class MLE	0.9771	0.9986	0.8833	0.7000
Two-class Gaussian	0.7857	0.9657	0.6833	0.7000
Two-class Naive Bayes	0.8914	0.9686	0.4633	0.1750
Logistic Regression	0.9700	**1**	0.1933	0.0000
One-class Silverman	0.0014	0.0014	1.0000	**1**
One-class MLE	0.0014	0.0014	0.9933	**1**
One-class Gaussian	0.0014	0.0014	**1**	**1**
One-class Naive Bayes	0.0014	0.0014	**1**	**1**

Table 7: German data: Sensitivity and Specificity (PCA 95%)

4.5 One-class vs. two-class classification for varied class imbalances with equal priors

4.5.1 Australian data

By comparing Figures 9 and 27 the conclusion can be made that the frequentest priors have no effect on the areas under the ROC curves for any of the classifiers. This is confirmed by comparing the actual AUC values of all the classifiers with varied priors to those with equal priors.

The frequentest priors do however affect the hit rate of the classifiers. This can be observed in the comparison of Figures 15 and 28. It has a stabilizing effect on the hit rate of the two-class Gaussian classifier. The priors also result in a higher hit rate for the two-class Naive Bayes classifier. The frequentest priors have a small adverse effect on the hit rate of the two-class non-parametric classifiers.

The harmonic means of the classifiers are also affected by the priors. Comparing Figure 21 with 29 the most striking difference is the harmonic mean of the logistic regression classifier. The use of frequentest priors have a large adverse effect on the harmonic mean of the logistic regression classifier. Even though its harmonic mean decreases it doesn't decrease as rapidly, or as much as in the case where frequentest priors are used. The harmonic mean of the two-class MLE classifier increase instead of decrease for the interval 17.03% to 6.97% defaulters. The harmonic mean of the last seven class imbalances is also larger if equal, instead of frequentest priors are used. The frequentest priors results in a higher sensitivity for the classifiers at class imbalances with less than 50% defaulters.

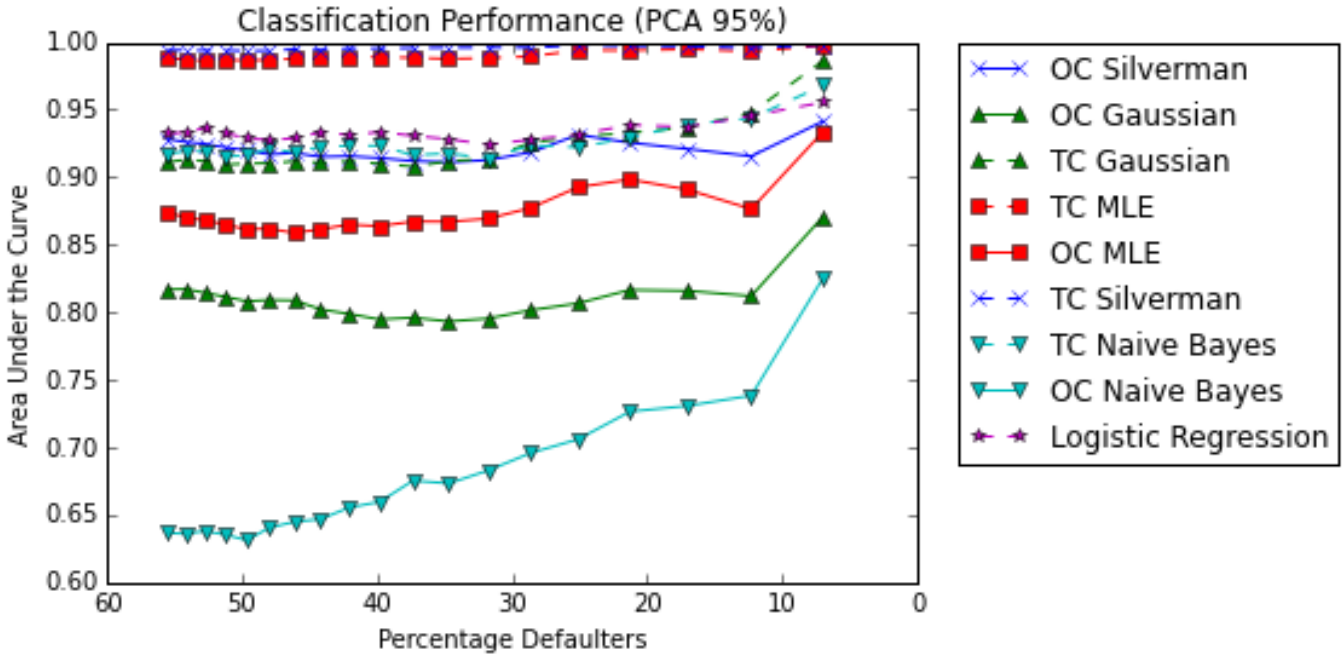


Figure 27: Australian data: Area Under the Curve (PCA 95%)

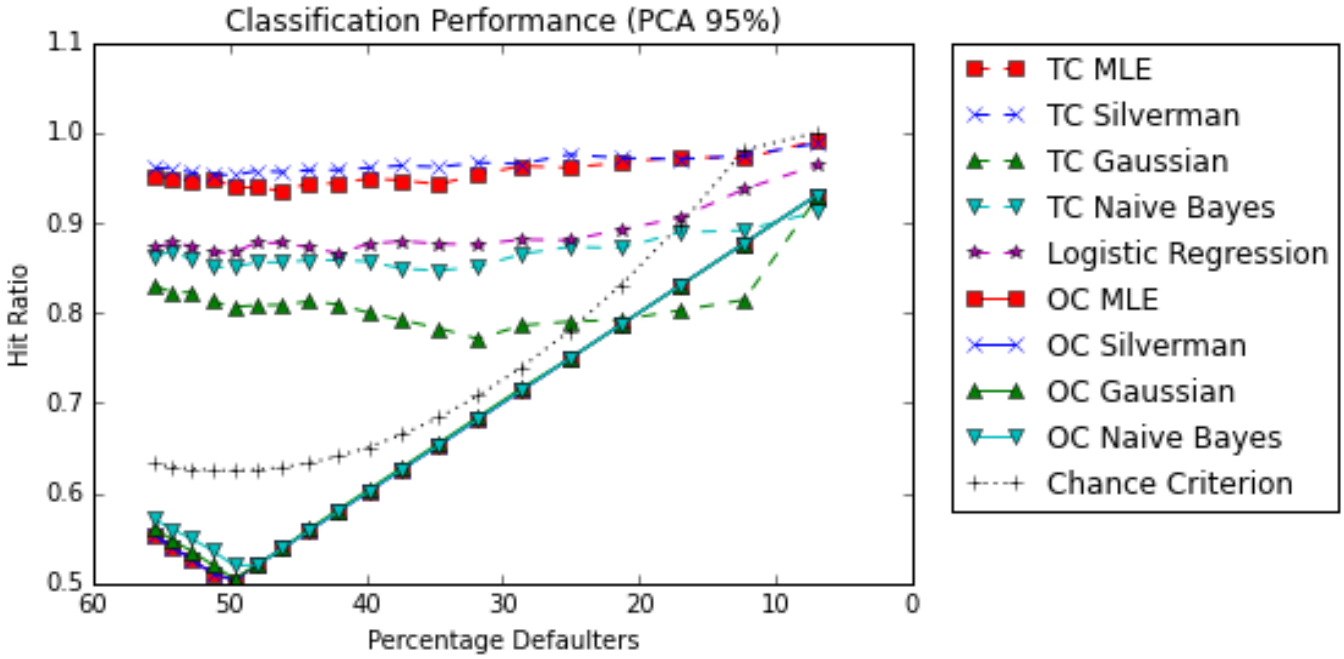


Figure 28: Australian data: Hit Rate (PCA 95%)

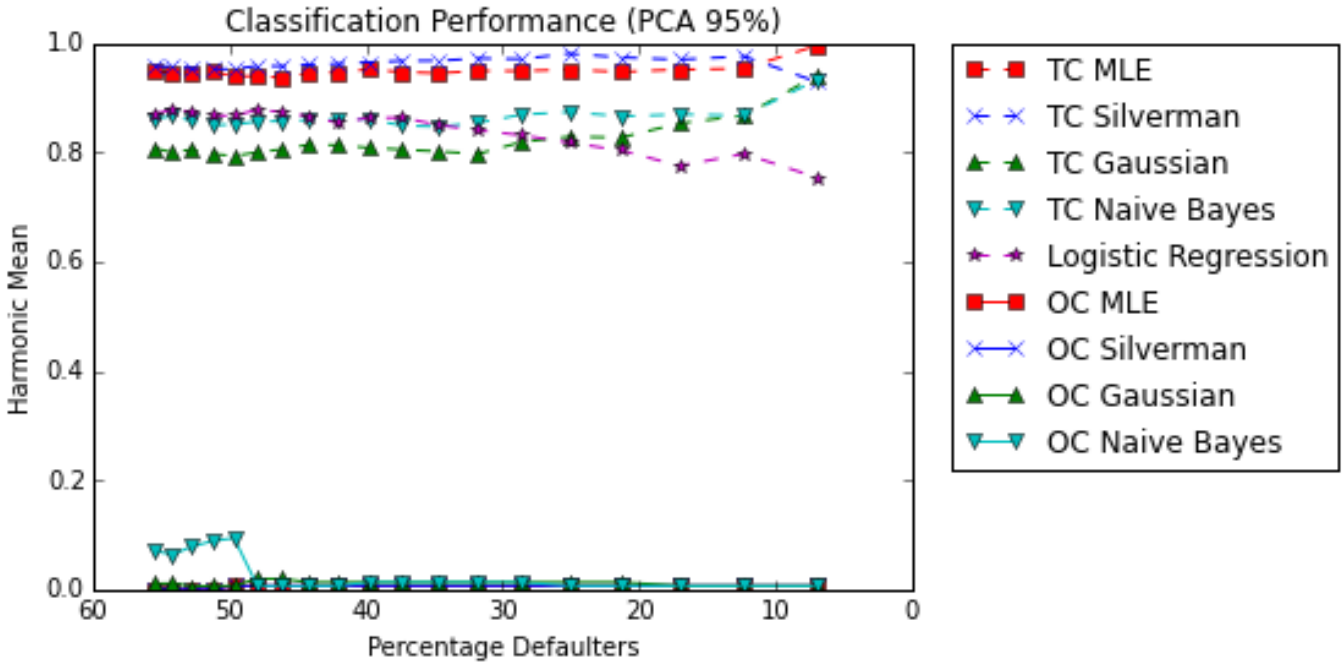


Figure 29: Australian data: Harmonic Mean (PCA 95%)

4.5.2 German data

As in the case of the Australian data set the AUC is unaffected by the frequentest priors. Figure 12 and 30 are thus identical.

The hit rates of the two-class Gaussian and Naive Bayes classifiers are increased by the use of frequentest priors. The hit rates of these classifiers fall below that of the one-class classifiers if equal instead of frequentest priors are used. The hit rate of the Gaussian classifier is below the benchmark regardless of the class imbalance. These observations are made from Figure 31. The comparison of Figures 18 and 31 indicates that the frequentest priors have an adverse effect on the hit rate of the logistic regression classifier. It also shows that effect of the frequentest priors on the two-class Silverman and MLE classifiers are very small.

The harmonic means of all the classifiers are adversely affected by the frequentest priors. The frequentest priors result in a larger sensitivity value for each of the classifiers for every class imbalance. However, the specificity of the classifiers decrease when frequentest, instead of equal priors, are used. It is important to notice that even when equal priors are used, the harmonic means of the non-parametric classifiers decrease as the class imbalance increase. The rate at which the harmonic mean of the two-class Silverman classifier decrease, for large class imbalances, is higher than that of the two-class MLE classifier. See Figure 32.

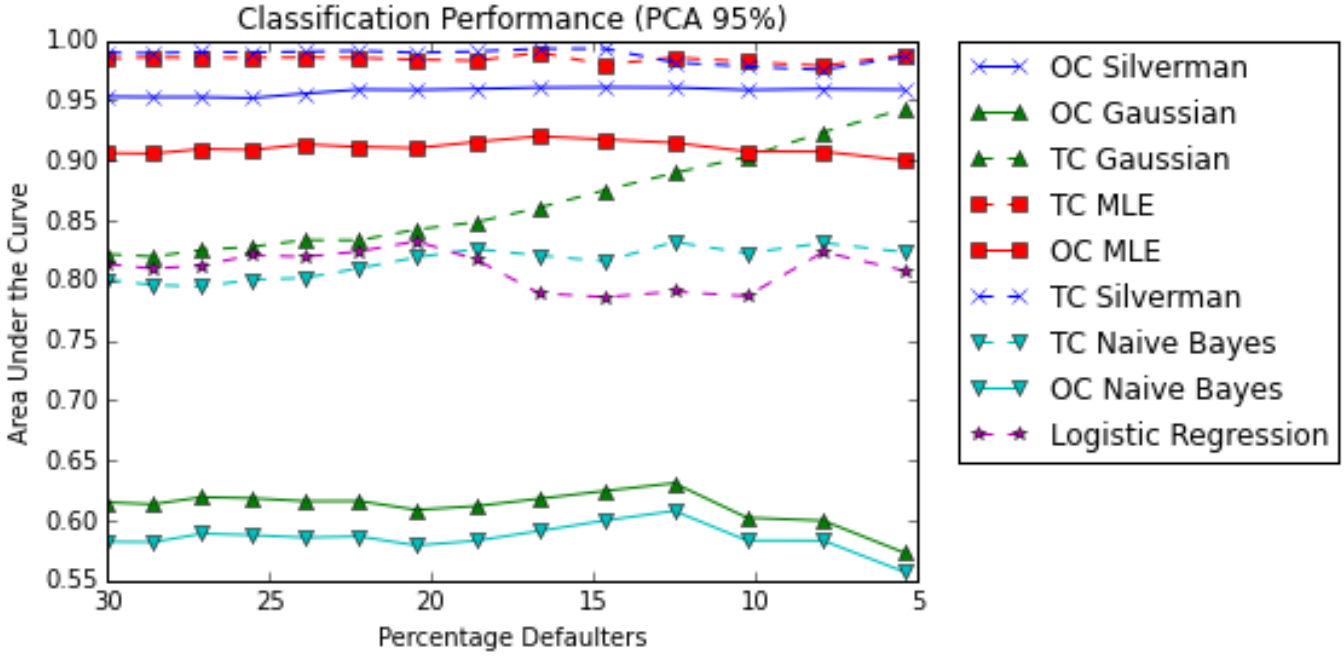


Figure 30: German data: Area Under Curve (PCA 95%)

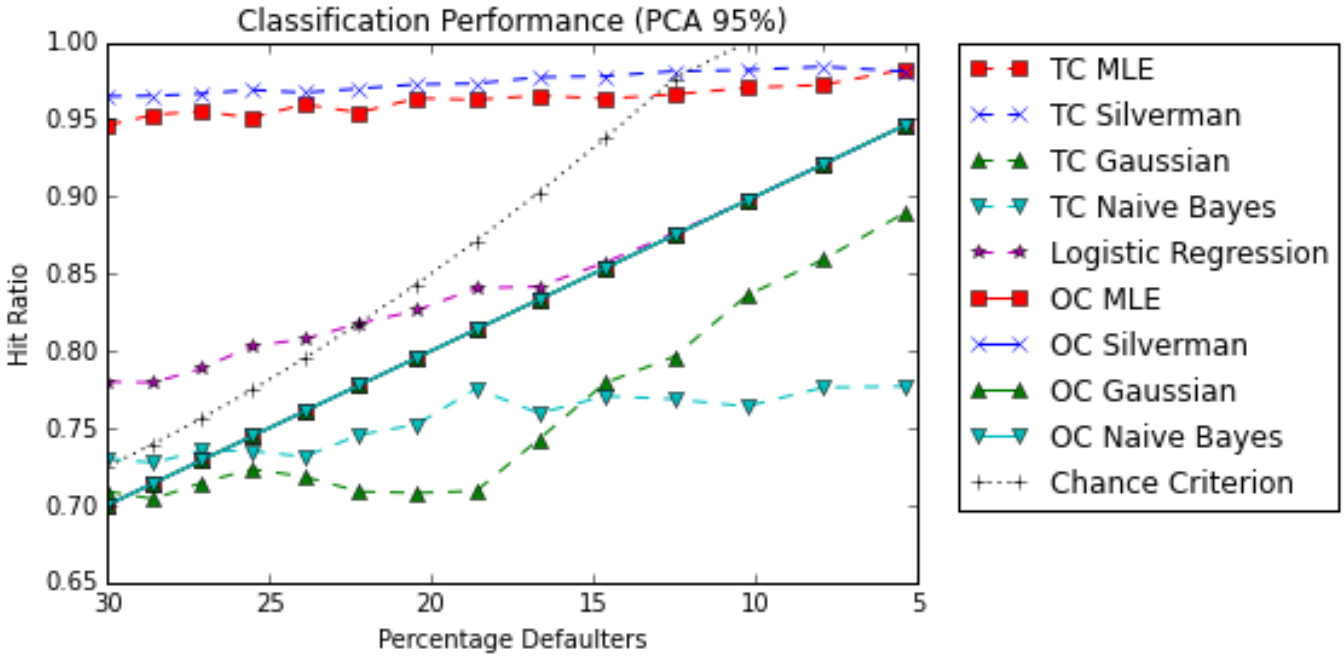


Figure 31: German data: Hit Rate (PCA 95%)

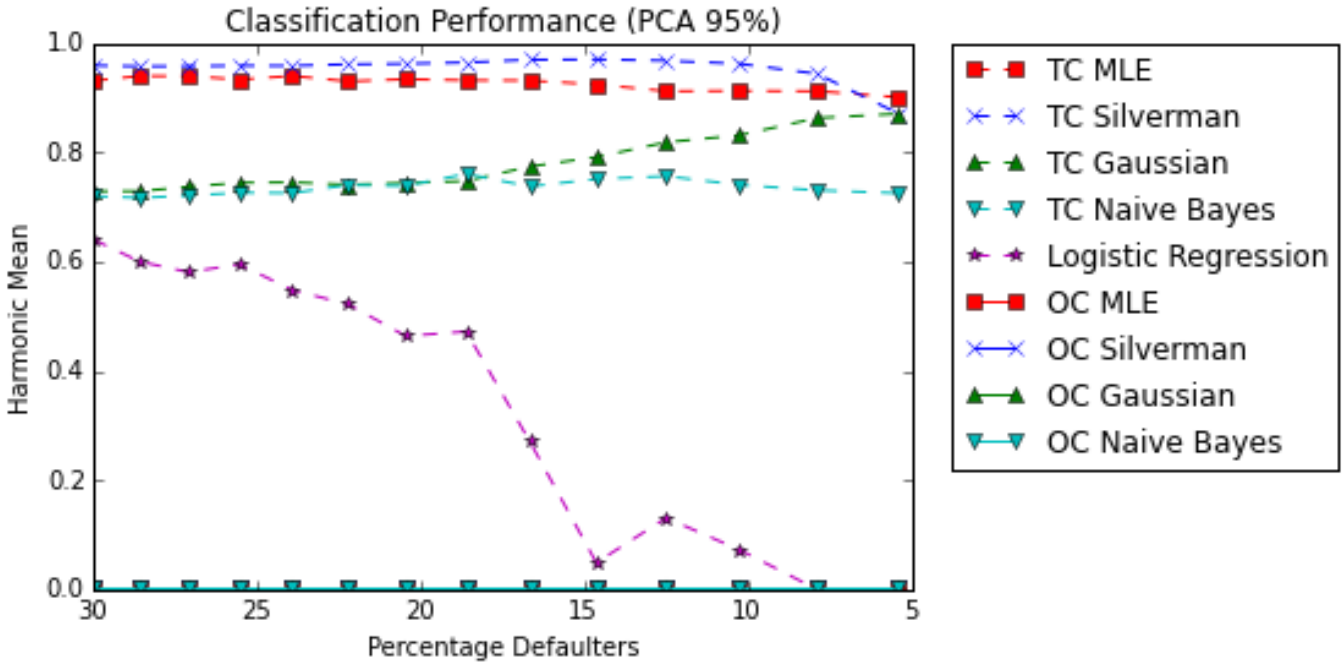


Figure 32: German data: Harmonic Mean (PCA 95%)

5 Conclusion

The aim of this paper is to evaluate the performance of both parametric and non-parametric classifiers on credit scoring data. It furthermore investigats the effect of frequentest priors, used in both parametric and non-parametric two-class classifiers, on the performance of these classifiers.

The non-parametric two-class classifiers outperforms all of the one-class classifiers, regardless of the class imbalance and method used to measure performance. The performance, measured in terms of the harmonic mean, of the two-class MLE classifier is more robust than that of the two-class Silverman classifier, given that PCA is applied to the data.

Even though the non-parametric one-class classifiers show promising results in terms of the AUC, the harmonic means of these classifiers contradict this. Depending on the class imbalance, either the sensitivity, or the specificity of these classifiers are very low. This goes to show that at any given default ratio, either the proportion of instances correctly classified as non-defaulting, or the proportion of instances correctly classified as defaulting is very low. This is also true for the parametric one-class classifiers. The weak performance of the one-class classifiers is confirmed by the hit ratios of these classifiers. Regardless of the class imbalance, the hit ratios of these classifiers fall below the benchmark. The optimal hit ratio that the one-class classifiers can achieve, is the proportion of instances in the largest class.

The priors have no effect on the AUC of the two-class classifiers. It does, however, affect the sensitivity and specificity and thus the harmonic mean of the classifiers. For any tested default ratio less than 50% the sensitivity is larger and specificity smaller when frequentest, instead of equal priors are used. If correctly classifying non-defaulters are of more importance than correctly classifying defaulters, the use of frequentest priors might be worth considering.

In this study multivariate techniques were used to estimate the bandwidths of the non-parametric kernel density estimators, and therefore the minimum default ratio is limited. The performance of these classifiers should be evaluated at smaller default ratios. In order to do this data sets with a larger proportion of observations compared to features are needed.

During all experiments the cost of misclassification was ignored. In practice the cost of wrongfully classifying an instance might be of more importance than the performance of the classifier. A paper by Hand

suggest the use of H-measures instead of the AUC to measure the performance of classifiers, since the H-measure unlike the AUC takes the cost of misclassification into account [6]. Future work should therefore also incorporate the cost of misclassification.

References

[1] L. Breiman, W. Meisel, and E. Purcell. Variable kernel estimates of multivariate densities. *Technometrics*, 19(2):135–144, 1977.

[2] V.S. Desai, D.G. Conway, J.N. Crook, and G.A. Overstreet. Credit-scoring models in the credit-union environment using neural networks and genetic algorithms. *IMA Journal of Management Mathematics*, 8(4):323–346, 1997.

[3] V.S. Desai, J.N. Crook, and G.A. Overstreet. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95(1):24–37, 1996.

[4] C. Drummond. Discriminative vs. generative classifiers for cost sensitive learning. In *The Proceedings of the Nineteenth Canadian Conference on Artificial Intellignce*, Canada, June 2006.

[5] S.E. Fienberg et al. When did Bayesian inference become "Bayesian"? *Bayesian Analysis*, 1(1):1–40, 2006.

[6] David J Hand. Measuring classifier performance: a coherent alternative to the area under the roc curve. *Machine learning*, 77(1):103–123, 2009.

[7] D.J. Hand and W.E. Henley. Statistical classification methods in consumer credit scoring: a review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 160(3):523–541, 1997.

[8] S. Harbour. Investopedia: what do credit score ranges mean? http://www.investopedia.com/articles/personal-finance/081514/what-do-credit-score-ranges-mean.asp, August 2014. Accessed: 2015-02-28.

[9] W.E. Henley and D. J. Hand. A k-nearest-neighbour classifier for assessing consumer credit risk. *The Statistician*, pages 77–95, 1996.

[10] C.-L. Huang, M.-C. Chen, and C.-J. Wang. Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*, 33(4):847–856, 2007.

[11] K. Kennedy, B. Mac Namee, and S.J. Delany. Using semi-supervised classifiers for credit scoring. *Journal of the Operational Research Society*, 64(4):513–529, 2012.

[12] M. Lichman. UCI machine learning repository, 2013.

[13] S.L. Lohr. *Sampling: Design and Analysis*. Richard Stratton, 2nd edition, 2010.

[14] D. Martens, T. Van Gestel, M. De Backer, R. Haesen, J. Vanthienen, and B. Baesens. Credit rating prediction using ant colony optimization. *Journal of the Operational Research Society*, 61(4):561–573, 2010.

[15] L. Medema, R.H. Koning, and R. Lensink. A practical approach to validating a pd model. *Journal of Banking & Finance*, 33(4):701–708, 2009.

[16] C.-S. Ong, J.-J. Huang, and G.-H. Tzeng. Building credit scoring models using genetic programming. *Expert Systems with Applications*, 29(1):41–47, 2005.

[17] M. Singh. Investopedia: the 2007-08 financial crisis in review. http://www.investopedia.com/articles/economics/09/financial-crisis-review.asp, 2015. Accessed: 2015-02-28.

[18] J.E. Stiglitz and A. Weiss. Credit rationing in markets with imperfect information. *The American Economic Review*, pages 393–410, 1981.

[19] L.C. Thomas. A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International Journal of Forecasting*, 16(2):149–172, 2000.

[20] C.M. Van der Walt. *Maximum-likelihood kernel density estimation in high-dimensional feature spaces*. PhD thesis, North-West University, 2014.

[21] C.M. van der Walt and E. Barnard. Kernel bandwidth estimation for non-parametric density estimation: a comparative study. In *Proceedings of the Twenty-Fourth Annual Symposium of the Pattern Recognition Association of South Africa*, Johannesburg, South-Africa, November 2013.

[22] D. West. Neural network credit scoring models. *Computers & Operations Research*, 27(11):1131–1152, 2000.

[23] L. Yu, S. Wang, and K.K. Lai. An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: the case of credit scoring. *European Journal of Operational Research*, 195(3):942–959, 2009.

6 Appendix

6.1 Proof: Silverman's univariate rule of thumb

Consider the $AMISE$ of an unknow function $\hat{f}_k$:

$$AMISE(\hat{f}_k) = \frac{1}{nh}\|K\|_2^2 + \frac{h^4}{4}\{\mu_2(K)\}^2\|f''\|_2^2$$

By differentiating $AMISE(\hat{f}_k)$ towards h we obtain:

$$\frac{d}{dh}AMISE(\hat{f}_k) = -\frac{1}{nh^2}\|K\|_2^2 + h^3\{\mu_2(K)\}^2\|f''\|_2^2$$

Setting this equal to 0 and solving h we get:

$$h = \left(\frac{\|K\|_2^2}{n\{\mu_2(K)\}^2\|\hat{f}_k\|_2^2}\right)^{\frac{1}{5}} \tag{1}$$

where K is the kernel being used, f is an unknown density function and $\mu_2(K) = \int x^2K(x)dx$.
Assume that f belongs to the family of normal distributions with mean μ and variance σ^2. Then

$$\|f''\|_2^2 = \left[\left\{\int_{-\infty}^{\infty}|f''(x)|^2dx\right\}^{\frac{1}{2}}\right]^2 \qquad \text{Since } \|j\|_2 = \left\{\int_a^b |j(x)|^2dg(x)\right\}^{\frac{1}{2}} \text{ with } g(x) = x$$

$$= \int_{-\infty}^{\infty}|f''(x)|^2dx \tag{2}$$

The pdf of f is given by

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

So

$$f'(x) = -\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}\left(\frac{1}{\sigma}\right)\left(\frac{x-\mu}{\sigma}\right)$$

$$= -\frac{1}{\sqrt{2\pi}\sigma^2}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}\left(\frac{x-\mu}{\sigma}\right)$$

and

$$f''(x) = \frac{1}{\sqrt{2\pi}\sigma^3}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}\left(\frac{x-\mu}{\sigma}\right)^2 - \frac{1}{\sqrt{2\pi}\sigma^3}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$= \frac{1}{\sqrt{2\pi}\sigma^3}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}\left[\left(\frac{x-\mu}{\sigma}\right)^2 - 1\right] \tag{3}$$

Now note that the standard normal pdf $p(z)$ is given by

$$p(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2}$$

so

$$p'(z) = -\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} z$$

and

$$p''(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2}(z^2 - 1) \tag{4}$$

Consider the transformation $z = \frac{x-\mu}{\sigma}$. Equation (3) becomes

$$f''(z) = \frac{1}{\sqrt{2\pi}\sigma^3} e^{-\frac{1}{2}z^2} \left[z^2 - 1\right]$$

taking equation (4) into account we see that

$$f''(z) = \frac{1}{\sigma^3} p''(z) \tag{5}$$

Substituting equation (5) back into equation (2)

$$\begin{aligned}
\|f''\|_2^2 &= \int_{-\infty}^{\infty} |f''(x)|^2 dx \\
&= \int_{-\infty}^{\infty} |\frac{1}{\sigma^3} p''(z)|^2 \sigma dz && \text{since } dx = \sigma dz \\
&= \sigma^{-5} \int_{-\infty}^{\infty} \{p''(z)\}^2 dz \\
&= \frac{\sigma^{-5}}{2\pi} \int_{-\infty}^{\infty} e^{-z^2}(z^2 - 1)^2 dz \\
&= \frac{\sigma^{-5}}{2\pi} \left[\int_{-\infty}^{\infty} e^{-z^2} z^4 dz - 2\int_{-\infty}^{\infty} e^{-z^2} z^2 dz + \int_{-\infty}^{\infty} e^{-z^2} dz\right] && (6)
\end{aligned}$$

Consider $\int_{-\infty}^{\infty} e^{-ax^2} dx$:

Let $I = \int_{-\infty}^{\infty} e^{-ax^2} dx$ and let $I = \int_{-\infty}^{\infty} e^{-ay^2} dy$ then

$$\begin{aligned}
I^2 &= \int_{-\infty}^{\infty} e^{-ax^2} dx \int_{-\infty}^{\infty} e^{-ay^2} dy \\
&= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-a(x^2+y^2)} dx dy \\
&= \int_0^{2\pi} \int_0^{\infty} e^{-ar^2} r dr d\theta \qquad \text{since } x = r\cos(\theta) \text{ and } y = r\sin(\theta) \\
&= \left[-\frac{1}{2} e^{-ar^2} \right]_0^{\infty} [\theta]_0^{2\pi} \\
&= (\frac{1}{2a})(2\pi) \\
&= \frac{\pi}{a}
\end{aligned}$$

so

$$I = \frac{\sqrt{\pi}}{\sqrt{a}} \tag{7}$$

Letting $a = 1$ in equation (7) we see that

$$\int_{-\infty}^{\infty} e^{-z^2} dz = \sqrt{\pi} \tag{8}$$

$\int_{-\infty}^{\infty} e^{-z^2} z^2 dz$ is solved by applying Feynman's trick.

$$\begin{aligned}
\int_{-\infty}^{\infty} e^{-z^2} z^2 dz &= -\frac{d}{da} \sqrt{\frac{\pi}{a}} |_{a=1} \\
&= \frac{\sqrt{\pi}}{2} a^{-\frac{3}{2}} |_{a=1} \\
&= \frac{\sqrt{\pi}}{2}
\end{aligned} \tag{9}$$

Applying the same techique to solve $\int_{-\infty}^{\infty} e^{-z^2} z^4 dz$

$$\begin{aligned}
\int_{-\infty}^{\infty} e^{-z^2} z^4 dz &= \frac{d^2}{da^2} \sqrt{\frac{\pi}{a}} |_{a=1} \\
&= -\frac{d}{da} \frac{\sqrt{\pi}}{2} a^{-\frac{3}{2}} |_{a=1} \\
&= \frac{3\sqrt{\pi}}{4} a^{-\frac{5}{2}} |_{a=1} \\
&= \frac{3\sqrt{\pi}}{4}
\end{aligned} \tag{10}$$

Substituting equations (8),(9) and (10) back into equation (6) we get:

$$\begin{aligned}
\|f''\| &= \frac{\sigma^{-5}}{2} \left[\int_{-\infty}^{\infty} e^{-z^2} z^4 dz - 2 \int_{-\infty}^{\infty} e^{-z^2} z^2 dz + \int_{-\infty}^{\infty} e^{-z^2} dz \right] \\
&= \frac{\sigma^{-5}}{2} \left[\frac{3\sqrt{\pi}}{4} - 2 \left(\frac{\sqrt{\pi}}{2} \right) + \sqrt{\pi} \right] \\
&= \sigma^{-5} \frac{3}{8\sqrt{\pi}}
\end{aligned} \tag{11}$$

However since σ is unknown it is estimated by $\hat{\sigma} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$
Assumeing the kernel is the Gaussian kernel function: $p(z) = \frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}z^2}$.
Then

$$\begin{aligned}
\mu_2(K) &= \mu_2(p(z)) \\
&= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} z^2 dz \\
&= E(Z^2) \\
&= Var(Z) + E[Z]^2 \\
&= 1 + 0^2 && \text{Since } Z \sim \text{N(0,1)} \\
&= 1 && (12)
\end{aligned}$$

and

$$\begin{aligned}
\|p\|_2^2 &= \left[\left\{ \int_{-\infty}^{\infty} |p(z)|^2 dz \right\}^{\frac{1}{2}} \right]^2 \\
&= \int_{-\infty}^{\infty} \frac{1}{2\pi} e^{-z^2} dz \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-z^2} dz \\
&= \left(\frac{1}{2\pi} \right) \left(\sqrt{\pi} \right) && \text{from equation (8)} \\
&= \frac{1}{2\sqrt{\pi}} && (13)
\end{aligned}$$

Finally we can calculate $\hat{h}$:

$$\begin{aligned}
\hat{h} &= \left(\frac{\|p\|_2^2}{\|\hat{f}''\|_2^2 \mu_2^2(p) n} \right)^{\frac{1}{5}} && \text{from equation (1)} \\
&= \left(\frac{\frac{1}{2\sqrt{\pi}}}{\left(\hat{\sigma}^{-5} \frac{3}{8\sqrt{\pi}} \right) (1)^2 n} \right)^{\frac{1}{5}} && \text{from equation (11), (12), and (13)} \\
&= \left(\frac{4}{3n} \hat{\sigma^5} \right)^{\frac{1}{5}} \\
&\approx 1.06 \hat{\sigma} n^{-\frac{1}{5}}
\end{aligned}$$

This concludes the proof for the univariate case of silverman's rule of thumb.

6.2 Proof: MLE of multivariate Gaussian kernel with a diagonal bandwidth matrix

The MLE estimator is derived using the LOUT ML, which is given by $l_{H(-i)}(X) = \sum_{i=1}^{N} \ln[p_{H(-i)}(X_i)]$. Therefore the derivation is as follows:

$$
\begin{aligned}
l_{H(-i)}(X) &= \sum_{i=1}^{N} \ln[p_{H(-i)}(\mathbf{X}_i)] \\
&= \sum_{i=1}^{N} \ln\left[\frac{1}{N-1}\sum_{j\neq i}^{N} K_{H_j}(\mathbf{X}_i - \mathbf{X}_j|H_j)\right] \\
&= \sum_{i=1}^{N} \ln\left[\frac{1}{N-1}\sum_{j\neq i}^{N}\prod_{p=1}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \text{ Since } H \text{ is a diagonal matrix } H_{j(p,p)} = h_{jp}^2 \\
&= \sum_{i=1}^{N} \ln\left[\frac{1}{N-1}\sum_{j\neq i}^{N} K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \\
&= \sum_{i=1}^{N} \ln\left[\frac{1}{N-1}\sum_{j\neq i}^{N}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \qquad (14)
\end{aligned}
$$

Equation (14) follows from the fact that $K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right) = \frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}$
By taking the partial derivative of equation (14) we obtain:

$$
\begin{aligned}
\frac{\partial}{\partial h_{jd}} l_{H(-i)}(X) &= \frac{\partial}{\partial h_{jd}}\sum_{i=1}^{N}\ln\left[\frac{1}{N-1}\sum_{j\neq i}^{N}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \\
&= \sum_{i=1}^{N}\frac{1}{\frac{1}{N-1}\sum_{j\neq i}^{N}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)} \\
&\times \sum_{i=1}^{N}\left[\frac{1}{N-1}\left(\frac{\partial}{\partial h_{jd}}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \qquad (15)
\end{aligned}
$$

This is true since if the derivative of the function is convergent, the derivative can be taken into the sumation.

Now consider $\frac{\partial}{\partial h_{jd}}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)$

$$
\begin{aligned}
\frac{\partial}{\partial h_{jd}}\left(\frac{1}{\sqrt{2\pi}h_{jd}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right) &= -\left(\frac{1}{h_{jd}^2}\right)\left(\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right) \\
&+ \left(\frac{1}{h_{jd}}\right)\left(\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)^2}\right)\left(\frac{x_{id}-x_{jd}}{h_{j}d}\right)\left(\frac{x_{id}-x_{jd}}{h_{jd}^2}\right) \\
&= \left(\frac{1}{h_{jd}}\right)\left[\frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{h_{jd}^2} - K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\right] \qquad (16)
\end{aligned}
$$

Substituting equation (16) into equation (15) we obtain:

$$
\begin{aligned}
&\frac{\partial}{\partial h_{jd}} l_{H(-i)}(X) \\
&= \sum_{i=1}^{N} \frac{1}{\frac{1}{N-1}\sum_{j\neq i}^{N} K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)} \\
&\times \sum_{i=1}^{N}\left[\frac{1}{N-1}\left(\left(\frac{1}{h_{jd}}\right)\left[\frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{h_{jd}^2} - K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\right]\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \\
&= \sum_{i=1}^{N} \frac{1}{\frac{1}{N-1}\sum_{j\neq i}^{N} K_{H_j}(\mathbf{X}_i - \mathbf{X}_j|H_j)} \\
&\times \sum_{i=1}^{N}\left[\frac{1}{N-1}\left(\left(\frac{1}{h_{jd}}\right)\left[\frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{h_{jd}^2} - K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\right]\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right)\right] \\
&= \sum_{i=1}^{N} \frac{1}{p_{H(-i)}(\mathbf{X}_i)}
\end{aligned}
$$

$$
\begin{aligned}
&\times \sum_{i=1}^{N}\left[\frac{1}{N-1}\left(\left(\frac{1}{h_{jd}}\right)\left[\frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{h_{jd}^2} - K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)\right]\right)\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right] \\
&= \frac{1}{(N-1)h_{jd}} \sum_{i=1}^{N}\left(\left[\frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{h_{jd}^2 p_{H(-i)}(\mathbf{X}_i)} - \frac{K_{h_{jd}}\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)}{p_{H(-i)}(\mathbf{X}_i)}\right]\prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)\right) \qquad (17)
\end{aligned}
$$

Setting the derivative in equation (17) equal to zero and solving for h_{jd} we get:

$$
h_{jd}^2 = \frac{\sum_{i=1}^{N} \frac{K\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)(x_{id}-x_{jd})^2}{p_{H(-i)}(\mathbf{X}_i)} \prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)}{\sum_{i=1}^{N} \frac{K\left(\frac{x_{id}-x_{jd}}{h_{jd}}\right)}{p_{H(-i)}(\mathbf{X}_i)} \prod_{p\neq d}^{N} K_{h_{jp}}\left(\frac{x_{ip}-x_{jp}}{h_{jp}}\right)}
$$

Therefore

$$
H_{j(d,d)} = \frac{\sum_{i=1}^{N} \frac{K_{\mathbf{H}_j}(x_i - x_k|\mathbf{H}_j)(x_{id}-x_{jd})^2}{p_{H(-i)}(\mathbf{X}_i)}}{\sum_{i=1}^{N} \frac{K_{\mathbf{H}_j}(x_i - x_k|\mathbf{H}_j)(x_{id}-x_{jd})^2}{p_{H(-i)}(\mathbf{X}_i)}}
$$

6.3 Python code

Listing 1: Australian data module

```
import numpy as np
import matplotlib.pyplot as plt
from fixed_module import FIXEDRATIO
from fold_module import FOLDS
from sklearn import metrics

def TF_pos(Confusion):
```

```
    T_pos = np.divide(Confusion[0,0,:],Confusion[0,0,:]+Confusion[0,1,:],dtype=float)
    F_pos = np.divide(Confusion[1,0,:],Confusion[1,0,:]+Confusion[1,1,:],dtype=float)
    auc = metrics.auc(F_pos,T_pos,reorder=True)
    return [T_pos, F_pos], auc

def decrease_c2(data,remove,col,c2):
    count =0
    i = 0
    while (count < remove) and (i-count < len(data)):
        if float(data[i-count,col]) == float(c2):
            data = np.delete(data,(i-count),axis = 0)
            count = count +1
        i = i+1
    return data

def graph(x,y,title,xlabel,ylabel,hit):
    plt.title(title)
    plt.xlabel(xlabel)
    plt.ylabel(ylabel)
    plt.gca().invert_xaxis()
    plt.plot(x,y[:,0],'--rs',label="TC MLE")
    plt.plot(x,y[:,1],'--bx', label="TC Silverman")
    plt.plot(x,y[:,2],'--g^',label="TC Gaussian")
    plt.plot(x,y[:,3],'--cv', label="TC Naive Bayes")
    plt.plot(x,y[:,4],'--m*', label="Logistic Regression")
    plt.plot(x,y[:,5], '-rs',label="OC MLE")
    plt.plot(x,y[:,6],'-bx',label="OC Silverman")
    plt.plot(x,y[:,7],'-g^',label="OC Gaussian")
    plt.plot(x,y[:,8],'-cv',label="OC Naive Bayes")
    if hit == True:
        plt.plot(x,y[:,9],':k+',label="Chance Criterion")
    plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)
    return None

def graph_roc(c1_TF,c2_TF,ratio,legend):
    plt.title('ROC Curves (Percentage Defaulters ' + str(round(ratio*100,2)) + '%)')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    l1 = "OC " + legend
    l2 = "TC " + legend
    plt.plot(c1_TF[1],c1_TF[0],'--+b', label=l1)
    plt.plot(c2_TF[1],c2_TF[0],'-*r',label = l2)
    plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)
    return None


aus = np.genfromtxt('Australian Credit Data.txt', dtype = float, delimiter = '')
#aus = aus[:150,:]
class_col = 14
N = len(aus)
n_folds = 10
c2 = 0 #Creditcard Rejected
c1 = 1
```

```
d = aus.shape[1]
remove = 20
interval = 19
pca = 0.95

obj = FIXEDRATIO(class_col,c1,c2,n_folds)
fold_obj = FOLDS(class_col,c1,d,n_folds)
c_size = fold_obj.Class_size(aus)
#Initialise arrays
accuracy = np.zeros((interval,10))
harmonics = np.zeros((interval,9))
H_measures = np.zeros((interval,9))
auc_1s = np.zeros(interval)
auc_2s = np.zeros(interval)
auc_1g = np.zeros(interval)
auc_2g = np.zeros(interval)
auc_2MLE = np.zeros(interval)
auc_1MLE = np.zeros(interval)
auc_2NB = np.zeros(interval)
auc_1NB = np.zeros(interval)
auc_log = np.zeros(interval)
ratio = np.zeros(interval)
plot = 0
for j in range(0,interval):
    print j
    ratio[j] = c_size[1]/float(c_size[1]+c_size[0])
    accuracy[j,:], harmonics[j,:], H_measures[j,:], S_1c, Gaus_1c, Gaus_2c,\
    MLE_1c, logit, MLE_2c, S_2c, NB_2c, NB_1c = obj.single_r(aus,pca)
    aus = decrease_c2(aus,remove,class_col,c2)
    c_size[1] = c_size[1] - remove
    #Calculate true and false positives and area under the curves
    S1, auc_1s[j] = TF_pos(S_1c)
    S2, auc_2s[j] = TF_pos(S_2c)
    G1, auc_1g[j] = TF_pos(Gaus_1c)
    G2, auc_2g[j] = TF_pos(Gaus_2c)
    MLE1, auc_1MLE[j] = TF_pos(MLE_1c)
    MLE2, auc_2MLE[j] = TF_pos(MLE_2c)
    NB2, auc_2NB[j] = TF_pos(NB_2c)
    NB1, auc_1NB[j] = TF_pos(NB_1c)
    Log2, auc_log[j] = TF_pos(logit)
    plt.figure(plot)
    graph_roc(S1,S2,ratio[j],'Silverman')
    plot = plot+1
    plt.figure(plot)
    graph_roc(G1,G2,ratio[j],'Gaussian')
    plot = plot+1
    plt.figure(plot)
    graph_roc(MLE1,MLE2,ratio[j],'MLE')
    plot = plot+1
    plt.figure(plot)
    graph_roc(NB1,NB2,ratio[j],'Naive Bayes')
    plot = plot+1
```

```
ratio = ratio*100
if pca == 0:
    title = 'Classification Performance (Z-Score)'
else:
    if pca > 1:
       title = 'Classification Performance (PCA)'
    else:
        title = 'Classification Performance (PCA ' + str(int(pca*100)) + '%)'

#plot results
plt.figure(plot)
plt.title(title)
plt.xlabel('Percentage Defaulters')
plt.ylabel('Area Under the Curve')
plt.gca().invert_xaxis()
plt.plot(ratio, auc_1s, '-bx', label = "OC Silverman")
plt.plot(ratio,auc_1g,'-g^', label = "OC Gaussian")
plt.plot(ratio,auc_2g,'--g^',label = "TC Gaussian")
plt.plot(ratio,auc_2MLE,'--rs',label = "TC MLE")
plt.plot(ratio,auc_1MLE,'-rs',label = "OC MLE")
plt.plot(ratio,auc_2s,'--bx',label = "TC Silverman")
plt.plot(ratio,auc_2NB,'--cv',label = "TC Naive Bayes")
plt.plot(ratio,auc_1NB,'-cv',label = "OC Naive Bayes")
plt.plot(ratio,auc_log,'--m*',label = "Logistic Regression")
plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)

plt.figure(plot+1)
graph(ratio,accuracy,title,'Percentage Defaulters','Hit Ratio',True)
plt.figure(plot+2)
graph(ratio,harmonics,title,'Percentage Defaulters','Harmonic Mean',False)
plt.figure(plot+3)
graph(ratio,H_measures,title,'Percentage Defaulters','H-Measure',False)
plt.show()
np.savetxt('auc_Gaus'+title+'.txt',np.transpose([auc_1g, auc_2g]), delimiter = ' ')
np.savetxt('auc_Silverman'+title+'.txt',np.transpose([auc_1s, auc_2s]), delimiter = ' ')
np.savetxt('auc_Naive'+title+'.txt',np.transpose([auc_1NB, auc_2NB]), delimiter = ' ')
np.savetxt('auc_MLE'+title+'.txt',np.transpose([auc_1MLE, auc_2MLE]), delimiter = ' ')

del obj
del fold_obj
```

Listing 2: German data module

```
# -*- coding: utf-8 -*-
"""
Created on Sat Jun 27 18:50:43 2015

@author: Estian
"""

import numpy as np
import matplotlib.pyplot as plt
from fixed_module import FIXEDRATIO
from fold_module import FOLDS
```

```
from sklearn import metrics

def TF_pos(Confusion):
    T_pos = np.divide(Confusion[0,0,:],Confusion[0,0,:]+Confusion[0,1,:],dtype=float)
    F_pos = np.divide(Confusion[1,0,:],Confusion[1,0,:]+Confusion[1,1,:],dtype=float)
    auc = metrics.auc(F_pos,T_pos,reorder=True)
    return [T_pos, F_pos], auc

def decrease_c2(data,remove,col,c2):
    count =0
    i = 0
    while (count < remove) and (i-count < len(data)):
        if float(data[i-count,col]) == float(c2):
            data = np.delete(data,(i-count),axis = 0)
            count = count +1
        i = i+1
    return data

def graph(x,y,title,xlabel,ylabel,hit):
    plt.title(title)
    plt.xlabel(xlabel)
    plt.ylabel(ylabel)
    plt.gca().invert_xaxis()
    plt.plot(x,y[:,0],'--rs',label="TC MLE")
    plt.plot(x,y[:,1],'--bx', label="TC Silverman")
    plt.plot(x,y[:,2],'--g^',label="TC Gaussian")
    plt.plot(x,y[:,3],'--cv', label="TC Naive Bayes")
    plt.plot(x,y[:,4],'--m*', label="Logistic Regression")
    plt.plot(x,y[:,5], '-rs',label="OC MLE")
    plt.plot(x,y[:,6],'-bx',label="OC Silverman")
    plt.plot(x,y[:,7],'-g^',label="OC Gaussian")
    plt.plot(x,y[:,8],'-cv',label="OC Naive Bayes")
    if hit == True:
        plt.plot(x,y[:,9],':k+',label="Chance Criterion")
    plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)
    return None

def graph_roc(c1_TF,c2_TF,ratio,legend):
    plt.title('ROC Curves (Percentage Defaulters ' + str(round(ratio*100,2)) + '%)')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    l1 = "OC " + legend
    l2 = "TC " + legend
    plt.plot(c1_TF[1],c1_TF[0],'--+b', label=l1)
    plt.plot(c2_TF[1],c2_TF[0],'-*r',label = l2)
    plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)
    return None


german = np.genfromtxt('German Credit Data.txt', dtype = float, delimiter = '')
#german = german[:150,:]
class_col = 24
N = len(german)
```

```
n_folds = 10
c2 = 2 #Default class
c1 = 1
d = german.shape[1]
remove = 20
interval = 14
pca = 0.95


obj = FIXEDRATIO(class_col,c1,c2,n_folds)
fold_obj = FOLDS(class_col,c1,d,n_folds)
c_size = fold_obj.Class_size(german)
#Initialise arrays
accuracy = np.zeros((interval,10))
harmonics = np.zeros((interval,9))
H_measures = np.zeros((interval,9))
auc_1s = np.zeros(interval)
auc_2s = np.zeros(interval)
auc_1g = np.zeros(interval)
auc_2g = np.zeros(interval)
auc_2MLE = np.zeros(interval)
auc_1MLE = np.zeros(interval)
auc_2NB = np.zeros(interval)
auc_1NB = np.zeros(interval)
auc_log = np.zeros(interval)
ratio = np.zeros(interval)
plot = 0
for j in range(0,interval):
    print j
    ratio[j] = c_size[1]/float(c_size[1]+c_size[0])
    accuracy[j,:], harmonics[j,:], H_measures[j,:], S_1c, Gaus_1c, Gaus_2c,\
    MLE_1c, logit, MLE_2c, S_2c, NB_2c, NB_1c = obj.single_r(german,pca)
    german = decrease_c2(german,remove,class_col,c2)
    c_size[1] = c_size[1] - remove
    #Calculate true and false positives and area under the curves
    S1, auc_1s[j] = TF_pos(S_1c)
    S2, auc_2s[j] = TF_pos(S_2c)
    G1, auc_1g[j] = TF_pos(Gaus_1c)
    G2, auc_2g[j] = TF_pos(Gaus_2c)
    MLE1, auc_1MLE[j] = TF_pos(MLE_1c)
    MLE2, auc_2MLE[j] = TF_pos(MLE_2c)
    NB2, auc_2NB[j] = TF_pos(NB_2c)
    NB1, auc_1NB[j] = TF_pos(NB_1c)
    Log2, auc_log[j] = TF_pos(logit)
    plt.figure(plot)
    graph_roc(S1,S2,ratio[j],'Silverman')
    plot = plot+1
    plt.figure(plot)
    graph_roc(G1,G2,ratio[j],'Gaus')
    plot = plot+1
    plt.figure(plot)
    graph_roc(MLE1,MLE2,ratio[j],'MLE')
```

```
    plot = plot+1
    plt.figure(plot)
    graph_roc(NB1,NB2,ratio[j],'Naive Bayes')
    plot = plot+1

ratio = ratio*100
if pca == 0:
    title = 'Classification Performance (Z-Score)'
else:
    if pca > 1:
       title = 'Classification Performance (PCA)'
    else:
        title = 'Classification Performance (PCA ' + str(int(pca*100)) + '%)'

#plot results
plt.figure(plot)
plt.title(title)
plt.xlabel('Percentage Defaulters')
plt.ylabel('Area Under the Curve')
plt.gca().invert_xaxis()
plt.plot(ratio, auc_1s, '-bx', label = "OC Silverman")
plt.plot(ratio,auc_1g,'-g^', label = "OC Gaussian")
plt.plot(ratio,auc_2g,'--g^',label = "TC Gaussian")
plt.plot(ratio,auc_2MLE,'--rs',label = "TC MLE")
plt.plot(ratio,auc_1MLE,'-rs',label = "OC MLE")
plt.plot(ratio,auc_2s,'--bx',label = "TC Silverman")
plt.plot(ratio,auc_2NB,'--cv',label = "TC Naive Bayes")
plt.plot(ratio,auc_1NB,'-cv',label = "OC Naive Bayes")
plt.plot(ratio,auc_log,'--m*',label = "Logistic Regression")
plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0.)

plt.figure(plot+1)
graph(ratio,accuracy,title,'Percentage Defaulters','Hit Ratio',True)
plt.figure(plot+2)
graph(ratio,harmonics,title,'Percentage Defaulters','Harmonic Mean',False)
plt.figure(plot+3)
graph(ratio,H_measures,title,'Percentage Defaulters','H-Measure',False)
plt.show()
np.savetxt('auc_Gaus'+title+'.txt',np.transpose([auc_1g, auc_2g]), delimiter = ' ')
np.savetxt('auc_Silverman'+title+'.txt',np.transpose([auc_1s, auc_2s]), delimiter = ' ')
np.savetxt('auc_Naive'+title+'.txt',np.transpose([auc_1NB, auc_2NB]), delimiter = ' ')
np.savetxt('auc_MLE'+title+'.txt',np.transpose([auc_1MLE, auc_2MLE]), delimiter = ' ')
del obj
del fold_obj
```

Listing 3: Prior module

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jun 25 12:02:43 2015

@author: Estian
"""
from fold_module import FOLDS
```

```
class PRIOR:
    def __init__(self,c_col,c1_bool,d,n_folds,N):
        self._c_col = c_col
        self._c1_bool = c1_bool
        self._d = d
        self._n_folds = n_folds
        self._N = N
        self._in_fold = 0

    #Calculate frequentist prior
    def f_prior(self,x):
        obj = FOLDS(self._c_col,self._c1_bool,self._d,self._n_folds)
        n_c1, n_c2 = obj.Class_size(x)
        total = n_c1 + n_c2
        prior_c1 = float(n_c1)/(total)
        prior_c2 = float(n_c2)/(total)
        del obj
        return prior_c1, prior_c2
```

Listing 4: Single default ratio module

```
# -*- coding: utf-8 -*-
"""
Created on Wed Jun 24 13:37:17 2015

@author: Estian
"""
import numpy as np
from fold_module import FOLDS
from classify_module import CLASSIFIERS
from priors_module import PRIOR
from sklearn.decomposition import PCA

class FIXEDRATIO:
    def __init__(self,c_col, c1_bool, c2_bool,n_folds):
        self._n_folds = n_folds
        self._c_col = c_col
        self._c1_bool = c1_bool
        self._c2_bool = c2_bool

    #applies principal component analysis
    def P_C_A(self,x,persent):
        pca = PCA(n_components= persent)
        x_r = pca.fit(x).transform(x)
        d = pca.n_components_
        return x_r, d

    #Normalize data
    def z_score(self,x):
        sigma = np.std(x,axis=0)
        mean = np.mean(x,axis=0)
        z = (x-mean)/sigma
        return z
```

```
def confusion(self,confusion,title,f_p_c1):
    sensitivity = confusion[0,0]/(confusion[0,0]+confusion[0,1])
    specificity = confusion[1,1]/(confusion[1,1]+confusion[1,0])
    f_handle = file(title,'a')
    np.savetxt(f_handle,[sensitivity,specificity],delimiter=',',\
    header='Matrix '+ str(f_p_c1),fmt='%.6e')
    f_handle.close
    return None


def single_r(self,data, pca_p):
    d = data.shape[1]
    classes = data[:,self._c_col]
    classes = np.matrix(classes)
    data = data[:,:self._c_col]
    data = self.z_score(data)
    if pca_p <> 0:
        data, new_d = self.P_C_A(data, pca_p)
        n_pca = d-new_d-1
    else:
        n_pca = 0
    data = np.concatenate((data,np.transpose(classes)),axis=1)
    N = data.shape[0]
    d = data.shape[1]
    P = np.zeros((N/self._n_folds,2,self._n_folds))
    K = np.zeros((N/self._n_folds,3,self._n_folds))
    One_c_K = np.zeros((N/self._n_folds,2,self._n_folds))
    One_c_MLE = np.zeros((N/self._n_folds,2,self._n_folds))
    G_1c_prob = np.zeros((N/self._n_folds,2,self._n_folds))
    G_prob = np.zeros((N/self._n_folds,2,self._n_folds))
    NB_2c_prob = np.zeros((N/self._n_folds,2,self._n_folds))
    NB_1c_prob = np.zeros((N/self._n_folds,2,self._n_folds))
    log = np.zeros((N/self._n_folds,2,self._n_folds))
    fold_obj = FOLDS(self._c_col-n_pca,self._c1_bool,d,self._n_folds)
    class_obj = CLASSIFIERS(self._c1_bool,self._c2_bool)
    X_F = fold_obj.Fold(data,N)
    prior_obj = PRIOR(self._c_col-n_pca,self._c1_bool,d,self._n_folds,N)
    f_p_c1, f_p_c2 = prior_obj.f_prior(data)#calculate the
    #frequentist prior values
    prop = f_p_c1**2 + f_p_c2**2
    prop = prop + 0.25*prop
    if prop > 1:
        prop = 1
    for j in range(0,self._n_folds):
        X_tr_c1, X_tr_c2, X_te_c1, X_te_c2 = fold_obj.Train(j,X_F)
        # Remove class dimension
        Y_tr_c1 = X_tr_c1[:,:self._c_col-n_pca]
        Y_tr_c2 = X_tr_c2[:,:self._c_col-n_pca]
        Y_te_c1 = X_te_c1[:,:self._c_col-n_pca]
        Y_te_c2 = X_te_c2[:,:self._c_col-n_pca]
        X_te = np.concatenate((X_te_c1,X_te_c2),axis=0)
        X_tr = np.concatenate((X_tr_c1,X_tr_c2),axis=0)
```

```
Y_te = np.concatenate((Y_te_c1,Y_te_c2), axis=0)
Y_tr = np.concatenate((Y_tr_c1,Y_tr_c2),axis=0)
#Calculate standard deviations
Sigma_c1 = np.std(Y_tr_c1,axis=0,ddof=Y_tr_c1.shape[1])
Sigma_c2 = np.std(Y_tr_c2,axis=0,ddof=Y_tr_c1.shape[1])
#Calculata Silverman bandwidths
N_c1 = Y_tr_c1.shape[0]
N_c2 = Y_tr_c2.shape[0]
N_var = Y_tr_c1.shape[1]
H_c1 = class_obj.Silverman(Sigma_c1,N_c1, N_var)
H_c2 = class_obj.Silverman(Sigma_c2, N_c2, N_var)
#Calculate p(x|H_silverman)*prior for two class
K[:,0,j] = class_obj.Sum(H_c1,Y_tr_c1,Y_te)#*f_p_c1
K[:,1,j] = class_obj.Sum(H_c2,Y_tr_c2,Y_te)#*f_p_c2
K[:,2,j] = X_te[:,self._c_col-n_pca]
#Calculate bandwidth using MLE
H_mle_c1= class_obj.MLE2(Sigma_c1,Y_tr_c1)
H_mle_c2 = class_obj.MLE2(Sigma_c2,Y_tr_c2)
P[:,0,j] = class_obj.Sum_MLE(H_mle_c1,Y_tr_c1,Y_te)#*f_p_c1
P[:,1,j] = class_obj.Sum_MLE(H_mle_c2,Y_tr_c2,Y_te)#*f_p_c2
############ two class Gaussian ######################################
G_prob[:,0,j], G_prob[:,1,j] = \
class_obj.Gaussian_2c(Y_tr_c1,Y_tr_c2,Y_te)
G_prob[:,0,j] = G_prob[:,0,j]#*f_p_c1
G_prob[:,1,j] = G_prob[:,1,j]#*f_p_c2
############## two class Naive Bayes ##################################
NB_2c_prob[:,0,j], NB_2c_prob[:,1,j] = \
class_obj.Naive_Bayes_2c(Y_tr_c1,Y_tr_c2,Y_te)
NB_2c_prob[:,0,j] = NB_2c_prob[:,0,j]#*f_p_c1
NB_2c_prob[:,1,j] = NB_2c_prob[:,1,j]#*f_p_c2
########### one class Silverman#######################################
Sigma_one = np.std(Y_tr,axis=0,ddof=N_var)
One_c_H = class_obj.Silverman(Sigma_one,N,Y_tr.shape[1])
One_c_K[:,0,j] = class_obj.Sum(One_c_H,Y_tr_c1,Y_te)
One_c_K[:,1,j] = X_te[:,N_var]
############ One class MLE#############################################
One_MLE_H = class_obj.MLE2(Sigma_one,Y_tr_c1)
One_c_MLE[:,0,j] = class_obj.Sum_MLE(One_MLE_H,Y_tr_c1,Y_te)
One_c_MLE[:,1,j] = X_te[:,N_var]
############ one class Gaussian ######################################
G_1c_prob[:,0,j] = class_obj.Gaussian_1c(Y_tr_c1,Y_te)
G_1c_prob[:,1,j] = X_te[:,N_var]
############ One class Naive Bayes ###################################
NB_1c_prob[:,0,j] = class_obj.Naive_Bayes_1c(Y_tr_c1,Y_te)
NB_1c_prob[:,1,j] = X_te[:,N_var]
############ Logistic regression #####################################
log[:,0,j], log[:,1,j] = class_obj.logit(Y_tr,\
X_tr[:,self._c_col-n_pca],Y_te, X_te[:,self._c_col-n_pca])
log[:,0,j] = log[:,0,j]#*f_p_c1
log[:,1,j] = log[:,1,j]#*f_p_c2
if j == 0:
    K_s = K[:,:,0]
    P_mle = P[:,:,0]
```

```
            P_mle_1c = One_c_MLE[:,:,0]
            P_s_1c = One_c_K[:,:,0]
            P_gaus = G_prob[:,:,0]
            P_gaus_1c = G_1c_prob[:,:,0]
            P_NB_2c = NB_2c_prob[:,:,0]
            P_NB_1c = NB_1c_prob[:,:,0]
            P_log = log[:,:,0]
        else:
            K_s = np.concatenate((K_s,K[:,:,j]), axis=0)
            P_mle = np.concatenate((P_mle,P[:,:,j]), axis=0)
            P_mle_1c = np.concatenate((P_mle_1c,One_c_MLE[:,:,j]), axis=0)
            P_s_1c = np.concatenate((P_s_1c,One_c_K[:,:,j]),axis=0)
            P_gaus = np.concatenate((P_gaus,G_prob[:,:,j]),axis=0)
            P_gaus_1c = np.concatenate((P_gaus_1c,G_1c_prob[:,:,j]),axis=0)
            P_NB_2c = np.concatenate((P_NB_2c,NB_2c_prob[:,:,j]),axis=0)
            P_NB_1c = np.concatenate((P_NB_1c,NB_1c_prob[:,:,j]),axis=0)
            P_log = np.concatenate((P_log,log[:,:,j]),axis=0)
    true_c = K_s[:,2]
    #Calculate values for MLE
    MLE_class, MLE_accuracy = class_obj.classify(true_c,P_mle)
    Confusion_mle = class_obj.confusion(true_c,MLE_class)
    harmonic_mle = class_obj.harmonic(Confusion_mle)
    H_measure_mle = class_obj.H_measure(MLE_accuracy)
    P_MLE_ROC = np.zeros((N,2))
    P_MLE_ROC[:,0] = np.divide(P_mle[:,0],P_mle[:,0] + P_mle[:,1])
    P_MLE_ROC[:,1] = true_c
    MLE_result_2, MLE_2_class, true_mle_2 = class_obj.One_Class(P_MLE_ROC)
    #Calculate values for Silverman
    KDE_class, KDE_accuracy = class_obj.classify(true_c, K_s)
    Confusion_s = class_obj.confusion(true_c, KDE_class)
    harmonic_s = class_obj.harmonic(Confusion_s)
    H_measure_s = class_obj.H_measure(KDE_accuracy)
    P_S_ROC = np.zeros((N,2))
    P_S_ROC[:,0] = np.divide(K_s[:,0],K_s[:,0]+K_s[:,1])
    P_S_ROC[:,1] = true_c
    S_result_2, S_2_class, true_2s = class_obj.One_Class(P_S_ROC)
    #Classify using 2 class gaussian
    G_class, G_accuracy = class_obj.classify(true_c,P_gaus)
    Confusion_g = class_obj.confusion(true_c,G_class)
    harmonic_g = class_obj.harmonic(Confusion_g)
    H_measure_g = class_obj.H_measure(G_accuracy)
    P_g_ROC = np.zeros((N,2))
    P_g_ROC[:,0] = np.divide(P_gaus[:,0],P_gaus[:,0] + P_gaus[:,1])
    P_g_ROC[:,1] = true_c
    Gaus_2c_result, Gaus_2c_class, true_2g = class_obj.One_Class(P_g_ROC)
    #calculate values for two class Naive Bayes
    NB2_class,NB_accuracy = class_obj.classify(true_c,P_NB_2c)
    Confusion_NB = class_obj.confusion(true_c,NB2_class)
    harmonic_NB = class_obj.harmonic(Confusion_NB)
    H_measure_NB = class_obj.H_measure(NB_accuracy)
    P_NB_ROC = np.zeros((N,2))
    P_NB_ROC[:,0] = np.divide(P_NB_2c[:,0],P_NB_2c[:,0]+P_NB_2c[:,1])
    P_NB_ROC[:,1] = true_c
```

```
NB_2c_result, NB_2c_class, true_NB = class_obj.One_Class(P_NB_ROC)
#Calculate values for logistic regression
Log_class, log_accuracy = class_obj.classify(true_c,P_log)
Confusion_log = class_obj.confusion(true_c,Log_class)
harmonic_log = class_obj.harmonic(Confusion_log)
H_measure_log = class_obj.H_measure(log_accuracy)
P_log_ROC = np.zeros((N,2))
P_log_ROC[:,0] = np.divide(P_log[:,0],P_log[:,0]+P_log[:,1])
P_log_ROC[:,1] = true_c
Log_res, Log_2c, true_log = class_obj.One_Class(P_log_ROC)
#Classify using one class silverman
S_1c_result, S_1c_class, true_1s = class_obj.One_Class(P_s_1c)
#Classify using one class gaussian
Gaus_1c_result, Gaus_1c_class, true_g = class_obj.One_Class(P_gaus_1c)
#Classify using one class Naive Bayes
NB_1c_result, NB_1c_class, true_NB1 = class_obj.One_Class(P_NB_1c)
#Classify using one class MLE
MLE_1c_result, MLE_1c_class, true_MLE = class_obj.One_Class(P_mle_1c)
n_thresh = P_s_1c.shape[0]
Confusion_s_1c = np.zeros((2,2,n_thresh))
Confusion_gaus_1c = np.zeros((2,2,n_thresh))
Confusion_gaus_2c = np.zeros((2,2,n_thresh))
Confusion_MLE_2c = np.zeros((2,2,n_thresh))
Confusion_S_2c = np.zeros((2,2,n_thresh))
Confusion_NB_2c = np.zeros((2,2,n_thresh))
Confusion_NB_1c = np.zeros((2,2,n_thresh))
Confusion_MLE_1c = np.zeros((2,2,n_thresh))
Confusion_Log_2c = np.zeros((2,2,n_thresh))
#Set up ROC curve vectors
for j in range(0,n_thresh):
    Confusion_s_1c[:,:,j] = \
    class_obj.confusion(true_1s,S_1c_class[j,:])
    Confusion_gaus_1c[:,:,j] = \
    class_obj.confusion(true_g,Gaus_1c_class[j,:])
    Confusion_gaus_2c[:,:,j] = \
    class_obj.confusion(true_2g,Gaus_2c_class[j,:])
    Confusion_MLE_2c[:,:,j] = \
    class_obj.confusion(true_mle_2,MLE_2_class[j,:])
    Confusion_S_2c[:,:,j] = \
    class_obj.confusion(true_2s, S_2_class[j,:])
    Confusion_NB_2c[:,:,j] = \
    class_obj.confusion(true_NB, NB_2c_class[j,:])
    Confusion_NB_1c[:,:,j] = \
    class_obj.confusion(true_NB1, NB_1c_class[j,:])
    Confusion_MLE_1c[:,:,j] = \
    class_obj.confusion(true_MLE, MLE_1c_class[j,:])
    Confusion_Log_2c[:,:,j] = \
    class_obj.confusion(true_log,Log_2c[j,:])
#Find the optimal threshold
S_opt = max(S_1c_result, key=lambda x:x[0])
S_index = np.where(S_1c_result==S_opt)[0][0]
Gaus_opt = max(Gaus_1c_result, key=lambda x:x[0])
G_index = np.where(Gaus_1c_result == Gaus_opt)[0][0]
```

```
        NB_opt = max(NB_1c_result, key=lambda x:x[0])
        NB_index = np.where(NB_1c_result==NB_opt)[0][0]
        MLE_opt = max(MLE_1c_result, key=lambda x:x[0])
        MLE_index = np.where(MLE_1c_result==MLE_opt)[0][0]
        #Calculate the harmonic means
        harmonic_S1 = class_obj.harmonic(Confusion_s_1c[:,:,S_index])
        harmonic_g1 = class_obj.harmonic(Confusion_gaus_1c[:,:,G_index])
        harmonic_NB1 = class_obj.harmonic(Confusion_NB_1c[:,:,NB_index])
        harmonic_MLE1 = class_obj.harmonic(Confusion_MLE_1c[:,:,MLE_index])
        #Calculate the H-measures
        H_measure_s1 = class_obj.H_measure(S_opt[0])
        H_measure_g1 = class_obj.H_measure(Gaus_opt[0])
        H_measure_NB1 = class_obj.H_measure(NB_opt[0])
        H_measure_MLE1 = class_obj.H_measure(MLE_opt[0])

        self.confusion(Confusion_s,'Silv2.txt',f_p_c1)
        self.confusion(Confusion_mle,'MLE2.txt',f_p_c1)
        self.confusion(Confusion_g,'Gauss2.txt',f_p_c1)
        self.confusion(Confusion_NB,'NB2.txt',f_p_c1)
        self.confusion(Confusion_MLE_1c[:,:,MLE_index],'MLE1.txt',f_p_c1)
        self.confusion(Confusion_NB_1c[:,:,NB_index],'NB1.txt',f_p_c1)
        self.confusion(Confusion_gaus_1c[:,:,G_index],'Gauss1.txt',f_p_c1)
        self.confusion(Confusion_log,'Log.txt',f_p_c1)
        self.confusion(Confusion_s_1c[:,:,S_index],'Silv1.txt',f_p_c1)

        del fold_obj
        del class_obj
        del prior_obj
        return([MLE_accuracy, KDE_accuracy, G_accuracy, NB_accuracy,
                log_accuracy,
                MLE_opt[0], S_opt[0], Gaus_opt[0], NB_opt[0],prop],
               [harmonic_mle, harmonic_s, harmonic_g, harmonic_NB, harmonic_log,
                harmonic_MLE1, harmonic_S1, harmonic_g1, harmonic_NB1],
                [H_measure_mle, H_measure_s, H_measure_g, H_measure_NB, H_measure_log,
                 H_measure_MLE1, H_measure_s1, H_measure_g1, H_measure_NB1],
                Confusion_s_1c,Confusion_gaus_1c, Confusion_gaus_2c, Confusion_MLE_1c,
                Confusion_Log_2c,
                Confusion_MLE_2c, Confusion_S_2c, Confusion_NB_2c, Confusion_NB_1c)
```

Listing 5: Classifiers module

```
# -*- coding: utf-8 -*-
"""
Created on Tue Jun 23 21:18:29 2015

@author: Estian
"""
import numpy as np
from scipy.stats import beta
from scipy.stats import multivariate_normal
#from sklearn import svm
from sklearn.linear_model import LogisticRegressionCV

class CLASSIFIERS:
```

```
    def __init__(self, c1_bool, c2_bool):
        self._c1_bool = c1_bool
        self._c2_bool = c2_bool

    # Calculate Silverman's bandwidth
    def Silverman(self,Sigma, N,d):
        H = np.zeros((d,d))
        H = np.mat(H)
        a = (4/(float(d)+2))**(1/(float(d)+4))
        for i in range(0,d):
            H[i,i] = a*Sigma[i]*N**(-1/(float(d)+4))
            assert(Sigma[i] != 0)
        return H

#    # Calculate MLE bandwidth
#    def MLE(self,Sigma,x, dist):
#        d = x.shape[1]
#        N = len(x)
#        Hk = np.zeros((d,d,N))#initialise bandwidth
#        num = np.zeros((N,d))
#        den = np.zeros((N,d))
#        kernel = 0
#        H= self.Silverman(Sigma,N,d)#initial bandwidth
#        H = H**2
#        p_lout = self.pH(H,x)#calculate LOUT
#        for dim in range(0,d):
#            for k in range(0,N):
#                for i in range(0,N):
#                    if i != k:
#                        kernel = multivariate_normal.pdf(x[i,:],x[k,:],H)
#                        diff = (x[i,dim]-x[k,dim])**2
#                        num[k,dim] = (kernel*diff)/float(p_lout[i]) + num[k,dim]
#                        den[k,dim] = kernel/float(p_lout[i]) + den[k,dim]
#                Hk[dim,dim,k] = num[k,dim]/den[k,dim]
#                #in_dist = self.reg(x[k,dim],np.transpose(x[:,dim]))
#                if (Hk[dim,dim,k] < dist[k,dim]):
#                    Hk[dim,dim,k] = dist[k,dim]
#        return np.sqrt(Hk)

    def MLE2(self,Sigma,x):
        d = x.shape[1]
        N = len(x)
        Hk = np.zeros((d,d,N))#initialise bandwidth
        num = np.zeros((N,d))
        den = np.zeros((N,d))
        kernel = 0
        H= self.Silverman(Sigma,N,d)#initial bandwidth
        H = H**2
        p_lout = self.pH(H,x)#calculate LOUT
        x1 = np.tile(x,(N,1))
        x2 = np.repeat(x,N,axis=0)
        diff = x1-x2
```

```
            p_lout = np.tile(p_lout,N)
            p_lout = np.transpose(np.matrix(p_lout))
            for i in range(0,N):
                diff = np.delete(diff,(i*N),axis=0)
                p_lout = np.delete(p_lout,(i*N),axis=0)
            power = np.zeros(len(diff))
            f = diff*np.linalg.inv(H)
            for i in range(0,len(diff)):
                power[i] = np.dot(f[i,:],np.transpose(diff)[:,i])
            del f
            a = 1/(np.sqrt((2*np.pi)**d*np.linalg.det(H)))
            kernel = a*np.power(np.e,-0.5*power)
            del power #free memory
            kernel = np.matrix(kernel)
            kernel = np.transpose(kernel)
            diff_2 = diff**2
            del diff
            mult = np.multiply(kernel,diff_2)
            num = np.divide(mult,p_lout)
            del mult
            den = np.divide(kernel,p_lout)
            del kernel
            for i in range(0,N):
                sl = slice(i*(N-1),(i+1)*(N-1))
                temp_d = np.sum(den[sl])
                temp_d = np.tile(temp_d,d)
                temp_n = np.sum(num[sl,:], axis=0)
                div = np.divide(temp_n,temp_d)
                Min = np.min(np.ma.masked_equal(diff_2[sl,:],0.0,copy=False),axis=0)
                Max = np.maximum(div,Min)
                Hk[:,:,i] = np.diagflat(Max)
            del diff_2
            del den
            del num
            return np.sqrt(Hk)

        #LOUT expression
        def pH(self,H,x):
            H = np.asmatrix(H)
            N = len(x)
            p_lout = np.zeros(N)
            for i in range(0,N):
                for j in range(0,N):
                    if j != i:
                        h_k = multivariate_normal.pdf(x[i,:],x[j,:],H) #Hj?
                        p_lout[i] = p_lout[i] + h_k
                p_lout[i] = p_lout[i]/float(N-1)
                assert(p_lout[i] != 0)
            return p_lout

#       #Regulirasation
#       def reg(self,point,vector):
#           dist = (vector-point)**2
```

```
#         if np.array_equal(dist, np.zeros(len(vector))):
#            Min = 0
#         else:
#             Min = np.min(dist[np.nonzero(dist)])#Calculate smallest distance not 0
#         return Min #error at dim 14 i=1 k=0

    #Calculate the sum of kernels for Silverman
    def Sum(self,H,X_tr,X_te):
        N_tr = X_tr.shape[0]
        in_te = X_te.shape[0]
        kernel = np.zeros(in_te)
        for j in range(0,in_te):
            for i in range(0,N_tr):
                kernel[j] = multivariate_normal.pdf(X_tr[i,:],X_te[j,:],H) + kernel[j]
        return kernel/N_tr

    #Calculate the sum of kernels for MLE
    def Sum_MLE(self,H, X_tr, X_te):
        N_tr = X_tr.shape[0]
        in_te = X_te.shape[0]
        kernel = np.zeros(in_te)
        for j in range(0,in_te):
            for i in range(0,N_tr):
                kernel[j] = multivariate_normal.pdf(X_tr[i,:],X_te[j,:],H[:,:,i]) +\
                kernel[j]
        return kernel/float(N_tr)

    #Classify using a support vetor machine
#    def SVM(self,x):
#        d = x.shape[1]-1
#        #10^-1 - 10^5 gamma=1/d
#        clf = svm.SVC(C=0.5, gamma=0, kernel='linear',probability=True)
#        a,b = x[:,:d], x[:,d]
#        clf.fit(a,b)
#        prob = clf.predict_proba(x[:,:d])
#        return prob

    #Calculate the one class classifications
    def One_Class(self, likelihood):
        n = likelihood.shape[0]
        sort = sorted(likelihood,key=lambda x: x[0],reverse=False)
        sort = np.asarray(sort)
        Class = np.zeros((n,n))
        threshold = np.zeros(n)
        result = np.zeros((n,2))
        for j in range(0,n):
            if j != n-1:
                threshold[j] = (sort[j,0]+sort[j+1,0])/2
            else:
                threshold[j] = sort[j,0]+0.1
            count = 0
            for i in range(0,n):
                if sort[i,0] > threshold[j]:
```

```
                Class[i,j] = self._c2_bool
            else:
                Class[i,j] = self._c1_bool
            if Class[i,j] == sort[i,1]:
                count = count+1
        accuracy = float(count)/n
        result[j,0] = accuracy
        result[j,1] = threshold[j]
        true = sort[:,1]
    return result, Class, true

#Calculate two class gaussian classes
def Gaussian_2c(self,c1_tr,c2_tr,Y_te):
    n = Y_te.shape[0]
    prob_c1 = np.zeros(n)
    prob_c2 = np.zeros(n)
    m1 = np.mean(c1_tr,axis=0)
    cov1 = np.cov(c1_tr,rowvar=0)
    m2 = np.mean(c2_tr,axis=0)
    cov2 = np.cov(c2_tr,rowvar=0)
    for j in range(0,n):
        prob_c1[j] = multivariate_normal.pdf(Y_te[j,:],m1,cov1)
        prob_c2[j] = multivariate_normal.pdf(Y_te[j,:],m2,cov2)
    return prob_c1, prob_c2

#Calculate one class gaussian classes
def Gaussian_1c(self,Y_tr,Y_te):
    n = Y_te.shape[0]
    prob = np.zeros(n)
    m = np.mean(Y_tr,axis=0)
    cov = np.cov(Y_tr,rowvar=0)
    for j in range(0,n):
        prob[j] = multivariate_normal.pdf(Y_te[j,:],m,cov)
    return prob

def Naive_Bayes_2c(self,c1_tr,c2_tr,Y_te):
    n = Y_te.shape[0]
    prob_c1 = np.zeros(n)
    prob_c2 = np.zeros(n)
    m1 = np.mean(c1_tr,axis=0)
    m2 = np.mean(c2_tr,axis=0)
    cov1 = np.diag(np.var(c1_tr,axis=0))
    cov2 = np.diag(np.var(c2_tr,axis=0))
    for j in range(0,n):
        prob_c1[j] = multivariate_normal.pdf(Y_te[j,:],m1,cov1)
        prob_c2[j] = multivariate_normal.pdf(Y_te[j,:],m2,cov2)
    return prob_c1, prob_c2

#Calculate one class Naive Bayes classes
def Naive_Bayes_1c(self,Y_tr,Y_te):
    n = Y_te.shape[0]
    prob = np.zeros(n)
    m = np.mean(Y_tr,axis=0)
```

```
    cov = np.diag(np.var(Y_tr, axis=0))
    for j in range(0,n):
        prob[j] = multivariate_normal.pdf(Y_te[j,:],m,cov)
    return prob

#Logistic regression
def logit(self,Y_tr,Y_tr_class,Y_te,Y_te_class):
    model = LogisticRegressionCV(refit=True)
    model = model.fit(Y_tr,Y_tr_class)
    prob = model.predict_proba(Y_te)
    return prob[:,1], prob[:,0]

#Classify the observations
def classify(self,true, estimated):
    N = true.shape[0]
    count =0
    result = np.zeros(N)
    for i in range(0,N):
        if estimated[i,0] > estimated[i,1]:
            result[i] = self._c1_bool
        else:
            result[i] = self._c2_bool
        if float(result[i]) == float(true[i]):
            count = count+1
    accuracy = float(count)/N
    return result, accuracy

#Calculates the confusion matrix (contingency table)
def confusion(self,true, predicted):
    N = true.shape[0]
    matrix = np.zeros((2,2))
    for i in range(0,N):
        if true[i] == 1:
            if predicted[i] == true[i]:
                matrix[0,0] = matrix[0,0] +1#true positive
            else:
                matrix[0,1] = matrix[0,1] +1#false positive
        else:
            if predicted[i] == true[i]:
                matrix[1,1] = matrix[1,1] +1
            else:
                matrix[1,0] = matrix[1,0] +1
    return matrix

#Calculates Harmonic mean
def harmonic(self,confusion):
    sensitivity = confusion[0,0]/(confusion[0,0]+confusion[0,1])
    specificity = confusion[1,1]/(confusion[1,1]+confusion[1,0])
    h_mean = float(2*sensitivity*specificity)/(sensitivity+specificity)
    return h_mean

#Calculates the H-measure
def H_measure(self,x):
```

```
        a=2
        b=2
        value = beta.cdf(x,a,b)
        return value
```

Listing 6: Cross-validation module

```
# -*- coding: utf-8 -*-
"""
Created on Mon Jun 22 16:03:23 2015

@author: Estian
"""
import numpy as np
import math

class FOLDS:
    def __init__(self, c_col, c1_bool, d, n_folds):
        self._c_col = c_col
        self._c1_bool = c1_bool
        self._d = d
        self._n_folds = n_folds
        self._in_fold = 0

    #Determine the size of each class
    def Class_size(self,x):
        count1 = 0
        count2 = 0
        N = x.shape[0]
        for i in range(0,N):
            if x[i,self._c_col] == self._c1_bool:
                count1 = count1+1
            else: count2 = count2 +1
        return [count1, count2]

    # divide data into classes
    def Class(self,size_c,x):
        count1 =0 #counter for class1
        count2 =0 #counter for class2
        x_c1 = np.zeros((size_c[0],self._d))
        x_c2 = np.zeros((size_c[1],self._d))
        for i in range(0,x.shape[0]):
            if x[i,self._c_col] == self._c1_bool:
                x_c1[count1] = x[i,:]
                count1 = count1 +1
            else:
               x_c2[count2] = x[i,:]
               count2 = count2 +1
        return x_c1, x_c2

    #Set up the folds
    def Fold(self,x,N):
        #Set up the classes
        self._in_fold = math.ceil(N/self._n_folds)
```

```
    size_c = FOLDS.Class_size(self,x)
    x_c1, x_c2 = FOLDS.Class(self,size_c,x)
    x_c = np.concatenate((x_c1,x_c2))
    x_F = np.zeros((self._in_fold,self._d,self._n_folds))
    for i in range(0,self._n_folds):
        count = 0
        for j in range(i,N,self._n_folds):
            x_F[count,:,i] = x_c[j,:] #order is: row, col, folds
            count = count +1
    return x_F

#Train the data
def Train(self,f_num,x_F):
    X_te = np.zeros((self._in_fold,self._d))
    temp = np.zeros((self._in_fold*(self._n_folds-1),self._d,self._n_folds))
    X_te = x_F[:,:,f_num]
    if f_num != 0:
        temp = np.concatenate((x_F[:,:,:(f_num-1)],x_F[:,:,f_num:]),axis=2)
    else:
        temp = x_F[:,:,(f_num+1):]
    for j in range(0,self._n_folds-1):
        if j == 0:
            X_tr = temp[:,:,j]
        else:
            X_tr = np.concatenate((X_tr,temp[:,:,j]),axis=0)
    size_te = self.Class_size(X_te)
    size_tr = self.Class_size(X_tr)
    X_te_c1, X_te_c2 = self.Class(size_te,X_te)
    X_tr_c1, X_tr_c2 = self.Class(size_tr,X_tr)
    return X_tr_c1, X_tr_c2, X_te_c1, X_te_c2
```

Spatial modelling of peak ground acceleration in the Witwatersrand Basin

Hayley Reynolds 12044700

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisor: Mr T Loots, Co-supervisors: Prof. A Kijko, Ms A Smit

Department of Statistics, University of Pretoria

2 November 2015

Abstract

Spatial statistics involves data whose location plays a significant role in the characteristics of the observations. These observations, which are subject to random influence, have an additional variable, location, which tells the reader exactly where the observation occurred. Geostatistics is most well-known for its application of spatial interpolation in geosciences; predicting values at specific locations for which no observations have been recorded. Emphasis is placed specifically on the spatial interpolation method known as Kriging which calculates estimates and develops graphs to provide more insight into what can be expected at a location based on the values of neighbouring observations. Peak ground acceleration (PGA) is defined as the maximum acceleration amplitude measure of ground motion vibrations of an earthquake. This report uses spatial interpolation to generate a continuous spatial seismic hazard map for South Africa. Following the steps of the Kriging process resulted in a smooth contour plot of point measurements of estimated PGA. From these plots, PGA is expected to be high in the Western Cape, KwaZulu-Natal and the area known as the Witwatersrand Basin. Further research can be done to determine why this is so.

Declaration

I, Hayley Reynolds, declare that this essay, submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics, at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Hayley Reynolds

_____
Mr T Loots, Prof A Kijko, Ms A Smit

_____
Date

Acknowledgements

To my supervisors, Ms Ansie Smit, Mr Theodor Loots, and Professor Andrzej Kijko; your guidance and support has been invaluable.
A very special thank you to Ms Victoria Rautenbach of the Department of Geography, Geoinformatics and Meteorology at the University of Pretoria; your assistance with the graphics is much appreciated.
Acknowledgments must also be made to the University of Pretoria Natural Hazard Centre for the use of their estimated peak ground acceleration data in the application of this report as well as the Council for Geoscience for their observations of seismic events in the Witwatersrand Basin from 2000 to 2014.

Contents

List of Figures

List of Tables

1 Introduction

Waldo Tobler, a professor of Geography and Statistics at the University of California, suggested the first law of geography: "Everything is related to everything else, but near things are more related to each other" [37]. This is the driving force behind spatial statistics.

Geostatistics uses adapted methods of regression in an effort to describe the spatial continuity of natural phenomena [17]. Georges Matheron's estimation techniques (established in the *Theory of Regionalised Variables*) [27] are also utilised in geostatistics by being applied to observations which are influenced by position as well as by observations made nearby [7] i.e. spatial statistics.
Data used in geostatistics is measured in a space where the domain D is restricted and is most likely to be dependent. To be specific, spatial dependence implies that points found nearer to one another will have more attributes in common than points found further apart. The data will then be modelled and used to predict possible values at locations for which no data have been recorded. Other characteristics of the data which are also important are stationarity, ergodicity and isotrophy.
Modelling spatial independence involves fitting what is known as a semi-variogram $\gamma(\mathbf{h})$, where $\mathbf{h}$ is the distance between two observations. A semi-variogram is a graphical representation of the expected differences between pairs of samples with a related location [7]. It explains variation between observations. A semi-variogram is fitted to the data set by assigning weights to each observation i.e. a weighted non-linear least squares fit. The semi-variogram will be fitted using the SAS/GRAPH® software's capabilities; specifically the procedure PROC VARIOGRAM. The data will be analysed and the most efficient model will be used to fit the semi-variogram. In fact, an integration of multiple models is generally used in practice [7], which are known as nested models. The model is used to estimate values at locations for which no observations have been recorded. This process of spatial interpolation [35] aims to find a linear, unbiased, best predictor [14].
Danie Krige, a mining engineer from South Africa [1], described a method whereby optimum weights are assigned to data based on their relevance to what is being estimated as a way to find the best linear unbiased predictor [21]. Krige's contribution to spatial statistics smooths the data and improves the accuracy of estimation and prediction [40] and, for this reason, his procedure for spatial interpolation is known as Kriging [26]. Kriging will be executed through the SAS® procedure PROC KRIGE2D. As there are multiple types of Kriging, such as Ordinary Kriging, Co-Kriging and Log Normal Kriging, deciding which method to use is dependent on the characteristics of the data.
To summarise; given spatial data, the accuracy of an estimator can be assessed provided there is a sufficient model for the semi-variogram. From this semi-variogram a minimum variance, linear, unbiased estimator can be produced through Kriging [7].

2 Theory

The theory, concepts and notation of spatial statistics are developed extensively in the *Handbook of Spatial Statistics*, Isobel Clark's *Practical Geostatistics* as well as Ansie Smit's Masters dissertation entitled *Interpolation in stationary spatial and spatial-temporal data sets.* The theory and processes involved in building a semi-variogram model and spatial interpolation will be discussed further.

2.1 Geostatistics

Geostatistics is the statistical study of natural phenomena [18] which are spatially correlated [17] and in geostatistical studies the main aim is to interpolate the spatial distribution of a particular event [34]. The notion of geostatistics came about after D.G. Krige had established methods for processing true ore-grade distribution from samples [20] and then Georges Matheron went on to publish *Theory of Regionalised Variables* [27]. Spatial statistics has since been applied to many different fields such as meteorology and agronomy [38, 2].
It was soon found that these estimation techniques can be applied to any observations made nearby [7]. Spatial statistics was then developed under three branches; continuous spatial variation (the focus of this report), discrete spatial variation and spatial point patterns [14].
The easiest way to describe geostatistics is through a graph. Suppose the point A in Figure 1 is to be estimated, given samples 1 to 5. It seems logical that more importance would be placed on sample 1, by assigning it a higher weight, than on sample 5, due to the proximity of sample 1 to the point A. Therefore, it can be said that the relationship between the point being estimated and any sample is dependent on the geometric placing of the samples [7].

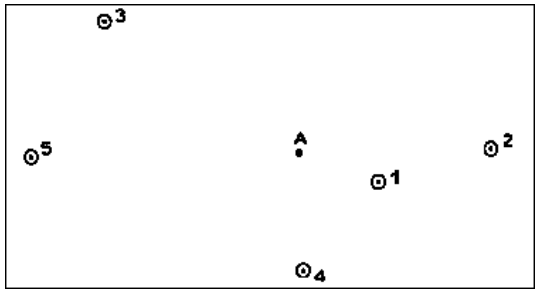


Figure 1: A hypothetical spatial statistical sample.

2.2 General Linear Model

A spatial co-ordinate will be defined here as the y (latitude) and x (longitude) co-ordinates of the i^{th} spatial location $\mathbf{u}_i = (x_i, y_i)$. This makes sense if the world map is placed onto an $y-x$ axis. Put more simply, there is a set of observations which are made up of a latitude variable and a longitude variable describing where the observations took place.
The set of locations for spatial analysis is:

$$\{\mathbf{u}_i : \mathbf{u}_i \in D; i = 1, ..., n\}. \tag{1}$$

For the univariate spatial case, let $Z(\mathbf{u}_i)$ represent a single measurement of a characteristic at location $\mathbf{u}_i$. Then $Z(\mathbf{u}_1)Z(\mathbf{u}_2)...Z(\mathbf{u}_n)$ is the set of dependent, stochastic variables, measured at n locations.

The general linear spatial model for prediction [29] can be expressed as:

$$Z(\mathbf{u}) = \mu(\mathbf{u}) + S(\mathbf{u}) + e(\mathbf{u}), \tag{2}$$

where:

- $\mu(\mathbf{u})$ = the deterministic trend,
- $S(\mathbf{u})$ = spatially dependent term $E[S(\mathbf{u})] = 0$,
- $e(\mathbf{u})$ = spatially independent term $E[S(\mathbf{u})] = 0$.

Listed below are a few properties of the data which are crucial in determining best linear unbiased estimators:

Stationarity

Stationarity is a set of assumptions regarding data distribution which allows for parameter estimation based on a standard set of properties [34]. Three types of stationarity are important for the Kriging procedure; strict, second-order and intrinsic stationarity, as this allows for a unique set of Kriging parameters to be determined.
Strict stationarity is the strongest form of stationarity and is defined as:

$$F\left(Z\left(\mathbf{u}_1\right), ..., Z\left(\mathbf{u}_n\right)\right) = F\left(Z(\mathbf{u}_1 + \mathbf{h}), ..., Z\left(\mathbf{u}_n + \mathbf{h}\right)\right) \forall \mathbf{h}. \tag{3}$$

The joint probability of n variables is unaffected by a shift $\mathbf{h}$, where $\mathbf{h}$ has been defined as the distance between two observations.
Second-order stationarity has two assumptions:

$$E[Z(\mathbf{u})] = \mu, \tag{4}$$

$$cov(Z(\mathbf{u}_i), Z(\mathbf{u}_i + \mathbf{h})) = C(\mathbf{h}). \tag{5}$$

This implies that the data satisfies the assumptions that the mean is constant regardless of position (equation (4)) and the covariance between two observations depends on the size of the distance between them and not the positioning (equation (5)).
Intrinsic stationarity has two requirements for all shifts, $\mathbf{h}$:

$$E[Z(\mathbf{u}) - Z(\mathbf{u} + \mathbf{h})] = 0,$$

$$var(Z(\mathbf{u} + \mathbf{h}) - Z(\mathbf{u})) = 2\gamma(\mathbf{h}).$$

$\gamma(\mathbf{h})$ is the notation used for a semi-variogram, explained in detail in Section 2.3.
Intrinsic stationarity allows for analysis of data whose variance increases as the spatial lag (distance between observations) increases. It is also important to note that strict stationarity implies second-order stationarity [9] which, in turn, implies intrinsic stationarity.

Ergodicity

Ergodicity and stationarity are closely related [40]. If $\bar{Z}$ is the constant sample mean and $Z(\mathbf{u})$ a random variable, then $\bar{Z} = E[Z(\mathbf{u})]$. This property allows spatial averages to be used for the entire space of data [9].

Isotropy

In an isotropic field, the variation of $Z(\mathbf{u})$ is the same in every direction. In other words, the covariance function $C(\mathbf{h})$ depends only on the length of the distance vector $\mathbf{h}$ [1]. If the variation of $Z(\mathbf{u})$ does not exhibit the same behaviour in every direction, then the field is anisotropic [18]. There are two types of anisotropy, namely geometric and zonal. Anisotropy is discussed in more detail in Margaret Armstrong's *Basic Linear Geostatistics* [1] and its interpretation in complex cases was investigated in *Interpolation of concentration measurements by Kriging using flow coordinates* [31].

2.3 Spatial Modelling

Spatial statistical dependence models are built from semi-variograms [10] and a semi-variogram is described as a graphical representation of the expected difference between pairs of observations at given locations [7]. Another way of defining a semi-variogram is to say it provides a description of the pattern and scale of spatial variability [11].
Before going into the details of the semi-variogram, it is necessary to define the covariance and correlation function for univariate spatial data.

Covariance

The population covariance is defined as:

$$\begin{aligned} C(\mathbf{h}) &= cov(Z(\mathbf{u}_i), Z(\mathbf{u}_i + \mathbf{h})) \\ &= E[(Z(\mathbf{u}_i) - E[Z(\mathbf{u}_i)])(Z(\mathbf{u}_i + \mathbf{h}) - E[Z(\mathbf{u_i} + \mathbf{h})])]. \end{aligned} \tag{6}$$

If $A = \{(\mathbf{u}_i, \mathbf{u}_j) | \mathbf{u}_i - \mathbf{u}_j = \mathbf{h}\}$ and N_A is the number of elements in the set A, then the sample covariance is defined as:

$$C(\mathbf{h}) = \frac{1}{N_A} \sum [Z(\mathbf{u}_i) - \frac{1}{N_A} \sum Z(\mathbf{u}_i)][Z(\mathbf{u}_i + \mathbf{h}) - \frac{1}{N_A} \sum Z(\mathbf{u}_i + \mathbf{h})]. \tag{7}$$

This makes sense, since as the observations get further and further apart, the relationships will decrease and if the observations have a lag distance of zero, then the covariance will simply be the variance.
The semi-variogram and covariance functions standardise the local mean of the data and give the relationship:

$$\gamma(\mathbf{h}) = C(\mathbf{0}) - C(\mathbf{h}), \tag{8}$$

which is graphically depicted in Figure 2.
This is based on intrinsic stationarity and is derived as follows [3]:

$$\begin{aligned} 2\gamma(\mathbf{h}) &= var[Z(\mathbf{u}_i + \mathbf{h}) - Z(\mathbf{u}_i)] \\ &= var[Z(\mathbf{u}_i + \mathbf{h})] + var[Z(\mathbf{u}_i)] - 2cov[Z(\mathbf{u}_i + \mathbf{h}), Z(\mathbf{u}_i)] \\ &= C(\mathbf{0}) + C(\mathbf{0}) - 2C(\mathbf{h}) \\ \gamma(\mathbf{h}) &= C(\mathbf{0}) - C(\mathbf{h}). \end{aligned} \tag{9}$$

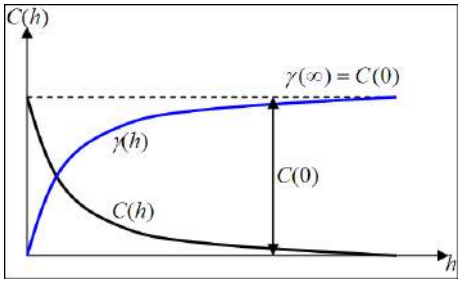


Figure 2: Graphical representation of the relationship between the covariance function and a semi-variogram. [34]

Some properties of the covariance function are [32]:

- $C(\mathbf{h}) \xrightarrow[\infty]{\mathbf{h}} 0$,
- $C(\mathbf{0}) \geq 0$,
- $C(\mathbf{h}) = C(-\mathbf{h})$ i.e. even function; unchanged after rotation about y axis.

Similarly, some properties of the semi-variance function [32]:

- $\gamma(\mathbf{0}) = 0$,
- $\gamma(\mathbf{h}) = \gamma(-\mathbf{h}) \geq 0$,
- $\gamma(\mathbf{h}) \xrightarrow[\infty]{\mathbf{h}} C(\mathbf{0})$.

Correlation

The population correlation is:

$$\rho(\mathbf{h}) = \frac{cov(Z(\mathbf{u}_i), Z(\mathbf{u}_i + \mathbf{h})}{\sqrt{var(Z(\mathbf{u}_i)) \cdot var(Z(\mathbf{u}_i + \mathbf{h}))}}. \tag{10}$$

While the experimental correlation is:

$$r(\mathbf{h}) = \frac{C(\mathbf{h})}{\sqrt{\frac{1}{N_A}\sum[Z(\mathbf{u}_i) - \frac{1}{N_A}\sum Z(\mathbf{u}_i)]^2 \cdot \frac{1}{N_A}\sum[Z(\mathbf{u}_i + \mathbf{h}) - \frac{1}{N_A}\sum Z(\mathbf{u}_i + \mathbf{h})]^2}}. \tag{11}$$

The semi-variogram represents spatial dependence [25] and the experimental semi-variogram can be estimated as:

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2n}\sum_{i=1}^{n}[Z(\mathbf{u}_i) - Z(\mathbf{u}_i + \mathbf{h})]^2. \tag{12}$$

There are many specific models which can be used to develop the semi-variogram more efficiently, it is generally a combination of a few different models which produces the best semi-variogram. McBratney and Webster [28] provide a review of some of the most widely used models for semi-variograms of which the most prevalent will be discussed here.

2.3.1 Spherical Model

Consider two observations made at one location ($\mathbf{h} = 0$), it is expected that there will be no difference in value for the two observations (since they have occurred at the exact same position), implying that the graph will cut through the origin. Now, if there is a small shift away from one observation to another, then it is expected that there will be a small difference in values. Ideally, as the distance between observations increases, so the value of the observations will become independent [7] and the semi-variogram will "level out" with the value of γ becoming somewhat constant. This shape is considered to be the ideal depiction of a semi-variogram and is more commonly known as the Spherical model. It is seen to be, to geostatistics what the normal distribution is, to statistics [7].
As can be seen in Figure 3, the Spherical model reaches a point where it stops increasing and remains constant. The distance $\mathbf{h}$ at which this is achieved is known as the range [17] and is denoted by a, while the semi-variance value at which the range is obtained is known as the sill [40] and will be denoted in this report by s.
The form of the Spherical model is:

$$\gamma(\mathbf{h}) = \begin{cases} s\left[\frac{3}{2}\left(\frac{[\![\mathbf{h}]\!]}{a}\right) - \frac{1}{2}\left(\frac{[\![\mathbf{h}]\!]}{a}\right)^3\right] & [\![\mathbf{h}]\!] \leq a \\ s & [\![\mathbf{h}]\!] > a \end{cases} \tag{13}$$

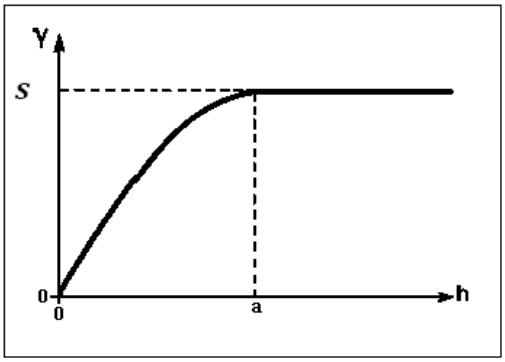


Figure 3: Graphical representation of the Spherical model. [7]

2.3.2 Exponential Model

The Exponential model is also used quite often when developing the semi-variogram and is described by:

$$\gamma(\mathbf{h}) = s[1 - e^{-\frac{[\![\mathbf{h}]\!]}{a}}]. \tag{14}$$

The effective range (r_ε) of the Exponential model is approximately 5% of the covariance at $\mathbf{h} = \mathbf{0}$ [6].

2.3.3 Gaussian Model

The Gaussian model of semi-variance is also frequently used. Much like the Exponential model, it approaches the sill asymptotically.
The Gaussian model is defined as:

$$\gamma(\mathbf{h}) = s[1 - e^{-([\![\mathbf{h}]\!]/a)^2}]. \tag{15}$$

A comparison of these three models is shown in Figure 4. It is clear that the Exponential model rises at a slower rate than the Spherical model and the Gaussian model rises the slowest of all three models, implying that as the distance between observations increases, the difference in variation of observations increases at a lower rate. While the Spherical model reaches its sill value, the Exponential and Gaussian models approach their sills asymptotically.

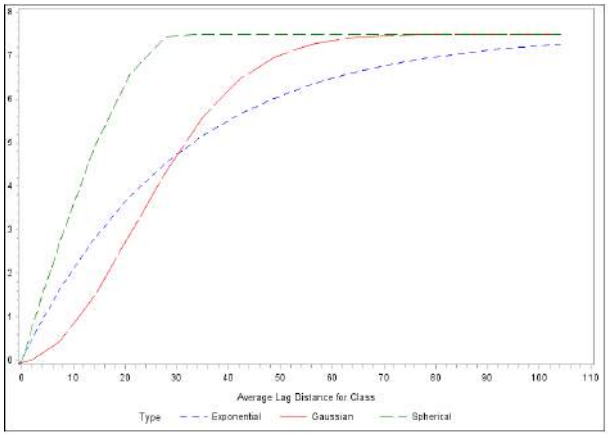


Figure 4: Graphical representation of the comparison of the Spherical, Exponential, Gaussian semi-variogram models.

2.3.4 Nested Models

It is often found that none of the aforementioned models fit the data efficiently on their own. In these situations, it is permissible to use what is known as a nested model. This is used when sources of variability occur simultaneously and for all $\mathbf{h}$ [18]. For example, if there exists a spatial process $Z(\mathbf{u})$ which contains correlation structures of both the Spherical and the Exponential models, then the nested model will be a combination of Equations 13 and 14, with the appropriate equation coming into play over the applicable range. The graphical representation of the nested semi-variogram is depicted in Figure 5.

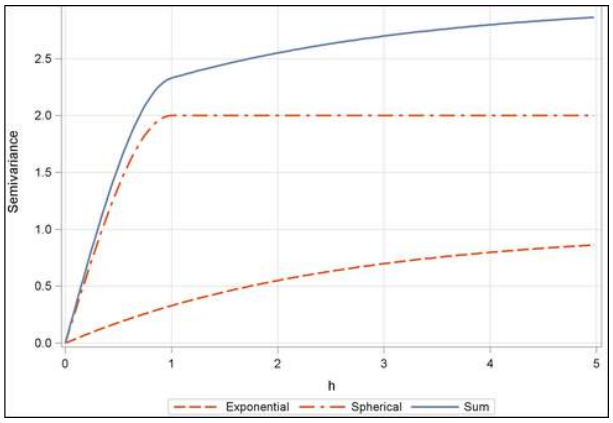


Figure 5: Example of a nested semi-variogram comprised of the Exponential and Spherical models (SAS®).

2.4 Spatial Interpolation

As was previously said, the main aim of geostatistical studies is to perform spatial interpolation in the most effective way. There are typically two forms of interpolation: point and areal (space). Point interpolation is generally applied to contour mapping which is why it will be used in this report. Furthermore, point interpolation consists of exact and approximate methods [22], as presented in Table 1 below.

Point Interpolation Methods	
Exact	**Approximate**
Kriging	Power series trend models
Distance-weighting	Fourier models
Spline interpolation	Distance-weighted least squares
Interpolating polynomials	Least-squares fitting with splines

Table 1: Examples of exact and approximate point interpolation methods.

Kriging is essentially a case of optimal linear prediction applied to a random field [36], whereby an attempt is made to estimate the value at a location for which no observations have been recorded. Specifically, points found closer to the location where the estimate is being made will be given a larger weighting than points found further away, which is similar to the Inverse Distance Weighting (IDW) function [33]. The biggest difference is that IDW uses Euclidean distances to determine weights while Kriging uses spatial dependent weights [34].
Originally, Kriging produced a linear predictor which implies that the estimated weights are linear combinations of the sample values. However, in recent developments, methods of optimal nonlinear spatial prediction have been developed and introduced into geostatistics [8]. This report will focus on linear Kriging.
There are many different types of Kriging; the choice of which method to use depends on the characteristics of the data as well as the desired spatial model [23]. Table 2 gives a few types of linear Kriging, their relative characteristics and a brief description of when the application of each model is appropriate [34, 23, 5, 4, 12].

Some advantages of Kriging [5]:

- Helps compensate for data clustering by treating a cluster as a single point.
- Gives error estimation as well as point estimation; allows for stochastic simulation of possible $Z(\mathbf{u})$.

Type	Characteristics	Application
Simple	Strict model assumptions	Data with a constant mean across entire domain
Ordinary	Known local mean	Data with a local stationary semi-variogram (covariance model)
Universal	Variable mean	Data with a strong trend which can be modelled with simple functions
Indicator	Estimating distribution as opposed to mean	Data with a categorical variable
Factorial	Several scales of variation	Multivariate data with co-regionalised variables
Log normal	Conservation of Log Normality	Log Normally distributed data
Co-Kriging	More than one attribute	Defined correlation between attributes

Table 2: Types of linear Kriging.

Ordinary Kriging
Ordinary Kriging is the most robust method of Kriging and is used most often [40]. It incorporates the semi-variogram model to generate a set of weighting factors which provide a minimum error.
The spatial ordinary Kriging estimator is:

$$\hat{Z}(\mathbf{u}_0) = \sum_{i=1}^{n} \lambda_i Z(\mathbf{u}_i). \tag{16}$$

That is, the estimator is the linear weighted average of the available measurements. Figure 6 shows the basics behind estimating the sample used in Figure 1. The shaded block is the area to be estimated. This will be done by giving each sample value a weighting (sample 1 getting the highest weight since it is in the block and sample 4 the next highest weight as it is second closest to the point A) and then calculating the estimator using equation (16).

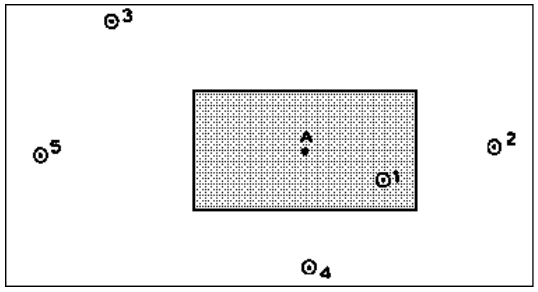


Figure 6: Kriging estimation area.

The method of ordinary Kriging has the following assumptions [34]:

1. The mean is a known constant μ,
2. the variogram and covariance are known,
3. $\sum_{i=1}^{n} \lambda_i = 1$,
4. the data is intrinsic stationary.

The third assumption ensures that the estimate is unbiased.
Suppose the data is second-order stationary (see Section 2.2), then the estimation error is:

$$\hat{Z}(\mathbf{u}_0) - Z(\mathbf{u}_0), \tag{17}$$

with:

$$\begin{aligned}\sigma^2(\mathbf{u}_0) &= var\left(\hat{Z}(\mathbf{u}_0) - Z(\mathbf{u}_0)\right) \\ &= var(\hat{Z}(\mathbf{u}_0)) - 2cov(\hat{Z}(\mathbf{u}_0), Z(\mathbf{u}_0)) + var(Z(\mathbf{u}_0)) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i\lambda_j C(\mathbf{u}_i - \mathbf{u}_j) - 2\sum_{i=1}^{j} \lambda_i C(\mathbf{u}_i - \mathbf{u}_0) + C(\mathbf{u}_0 - \mathbf{u}_0)\end{aligned} \tag{18}$$

minimised subject to $\sum_{i=1}^{n} \lambda_i = 1$ [15].
To represent the estimation variance in equation (18) in terms of the semi-variogram, the relationship in equation 8 will be used to get [34]:

$$\begin{aligned}\sigma^2(\mathbf{u}_0) &= \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i\lambda_j \left[C(\mathbf{u}_0 - \mathbf{u}_0) - \gamma(\mathbf{u}_i - \mathbf{u}_j)\right] - 2\sum_{i=1}^{n} \lambda_i \left[C(\mathbf{u}_0 - \mathbf{u}_0) - \gamma(\mathbf{u}_i - \mathbf{u}_0)\right] + C(\mathbf{u}_0 - \mathbf{u}_0) \\ &= -\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i\lambda_j \gamma(\mathbf{u}_i - \mathbf{u}_j) + 2\sum_{i=1}^{n} \lambda_i \gamma(\mathbf{u}_i - \mathbf{u}_0).\end{aligned} \tag{19}$$

In order to minimise equation (19) for equation (16), the weights will be subject to assumption 3, which will hold true for the variogram in equation (12) and the Lagrange multipliers will be used since a maximum or minimum of a function is required [34].
At the point $\mathbf{u}_0$, the variance to be minimised is:

$$E\left[\sum_{i=1}^{n} \lambda_i Z(\mathbf{u}_i) - Z(\mathbf{u}_0)\right]^2 - 2\varphi\left[\sum_{i=1}^{n} \lambda_i - 1\right], \tag{20}$$

where φ is the Lagrange multiplier (which ensures assumption 3 is met) resulting in:

$$-\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i\lambda_j \gamma(\mathbf{u}_i - \mathbf{u}_j) + 2\sum_{i=1}^{n} \lambda_i \gamma(\mathbf{u}_0 - \mathbf{u}_i) - 2\varphi\left(\sum_{i=1}^{n} \lambda_i - 1\right). \tag{21}$$

After differentiation of equation (21) with respect to λ_i and setting the derivative equal to zero:

$$-\sum_{j=1}^{n} \lambda_j \gamma(\mathbf{u}_i - \mathbf{u}_j) + \gamma(\mathbf{u}_0 - \mathbf{u}_i) - \varphi = 0. \tag{22}$$

The spatial ordinary Kriging system can then be expressed in terms of the semi-variogram [16] :

$$\begin{cases} \sum_{j=1}^{n} \lambda_j \gamma(\mathbf{u}_i - \mathbf{u}_j) - \varphi = \gamma(\mathbf{u}_i - \mathbf{u}_0) \quad i = 1, 2, ..., n \\ \sum_{j=1}^{n} \lambda_j = 1 \end{cases} \tag{23}$$

The λ_j's are then substituted into equation (17) and the spatial ordinary Kriging estimate for location $\mathbf{u}_0$ is determined.

Matrix Notation
Equation (23) can be rewritten as:

$$\begin{bmatrix} \gamma_{11} & \cdots & \gamma_{1n} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \gamma_{n1} & \cdots & \gamma_{nn} & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ \varphi \end{bmatrix} = \begin{bmatrix} \gamma_{01} \\ \vdots \\ \gamma_{0n} \\ 1 \end{bmatrix}, \tag{24}$$

where γ_{ij} is the semi-variogram between the $i^{th}$and j^{th} locations, or:

$$\mathbf{\Gamma}\lambda = \gamma_{\mathbf{0}}. \tag{25}$$

If equation (25) is partitioned as follows:

$$\begin{bmatrix} \mathbf{\Gamma} & \mathbf{1} \\ \mathbf{1}` & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \varphi \end{bmatrix} = \begin{bmatrix} \gamma_{\mathbf{0}} \\ 1 \end{bmatrix}, \tag{26}$$

then:

$$\begin{aligned} \lambda &= \mathbf{\Gamma}^{-1}\gamma_{\mathbf{0}} \\ &= \left(\gamma_{\mathbf{0}} + \mathbf{1}\frac{\mathbf{1} - \mathbf{1}\mathbf{\Gamma}^{-1}\gamma_{\mathbf{0}}}{\mathbf{1}\mathbf{\Gamma}^{-1}\gamma_{\mathbf{0}}}\right)` \mathbf{\Gamma}^{-1} \end{aligned} \tag{27}$$

and:

$$\varphi = -\frac{(1 - \mathbf{1}\mathbf{\Gamma}^{-1}\gamma_{\mathbf{0}}}{\mathbf{1}\mathbf{\Gamma}^{-1}\mathbf{1}`}. \tag{28}$$

Equation (8) can then be used to express the ordinary Kriging system in terms of the covariance:

$$\begin{cases} \sum_{j=1}^{n} \lambda_j C(\mathbf{u}_i - \mathbf{u}_j) + \varphi = C(\mathbf{u}_i - \mathbf{u}_0) \quad i = 1, 2, ..., n \\ \sum_{j=1}^{n} \lambda_j = 1 \end{cases} \tag{29}$$

Deciding whether to use the system represented by the semi-variogram or the covariance depends entirely on what the reader's preference is and which tools will best describe the data.

3 Application

Peak ground acceleration (PGA) is a seismic hazard, since seismic hazard is defined as a natural phenomenon as a result of an earthquake, such as ground movement or a fault rupture [39]. In seismology there is also seismic risk estimation, which is the calculation of the possible effects future earthquakes can have on a community as well as the probability of such an event taking place [19].
In this research report, the data used is the estimated PGA for South Africa with a 10% probability of being exceeded at least once in a 50 year period. For the remainder of the report it will simply be referred to as PGA. The estimates are of a spatial nature [10] since they contain predictions with longitude and latitude co-ordinates describing their location. In order to calculate seismic risk for South Africa, it is necessary to have seismic hazard and therefore as much information as possible about PGA in the country. The spatial interpolation method of Kriging is optimal because a smooth contour plot can be created from the PGA estimates, which is the easiest to interpret in this case.
It is important to note that the data that is available for estimated PGA is very well-populated, which is not generally the case. It is therefore necessary to perform Kriging on all the data as well as on a much smaller portion to obtain results of a more realistic event. This will demonstrate the effectiveness of the Kriging procedure if both outcomes are very similar.
It can be seen in Figure 7 that the Witwatersrand Basin has more seismic events of a greater magnitude than the rest of the country, therefore this area will also be of greater interest and will be investigated more closely.
The SAS code used throughout the application can be found in Appendix A and the output in Appendix B.

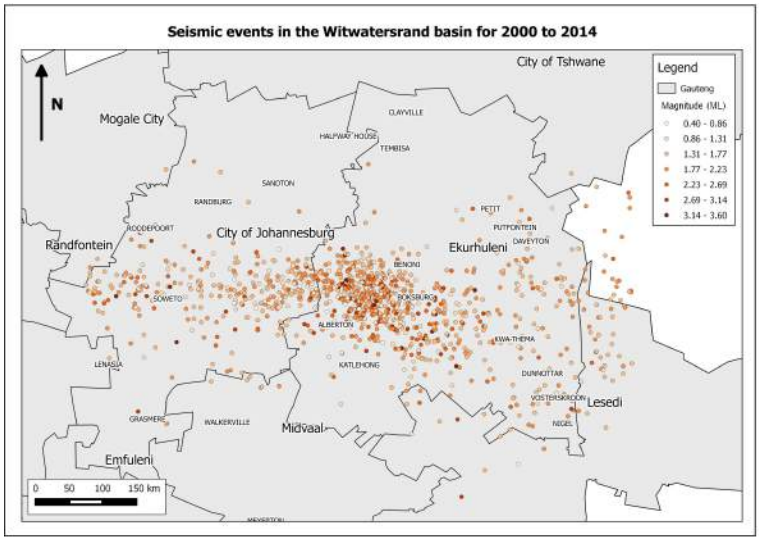


Figure 7: Plot of the recorded seismic events for the Witwatersrand Basin from 2000 to 2014.

3.1 Spatial Modelling

A large data set with 4161 estimated observations was used with variables: longitude (x), latitude (y) and PGA which is measured in terms of gravitational acceleration ($g = 9.8m/s$) . The co-ordinates have an incremental value of 0.25°, i.e. there is an estimated PGA values every 0.25° on the map.
Figure 8 shows how the PGA estimates are distributed. A circle represents an estimate and the shading of the circle indicates the magnitude of the PGA. It is evident from the scale on the right hand side that there are 3 prominent areas with a higher PGA than the rest of the data.

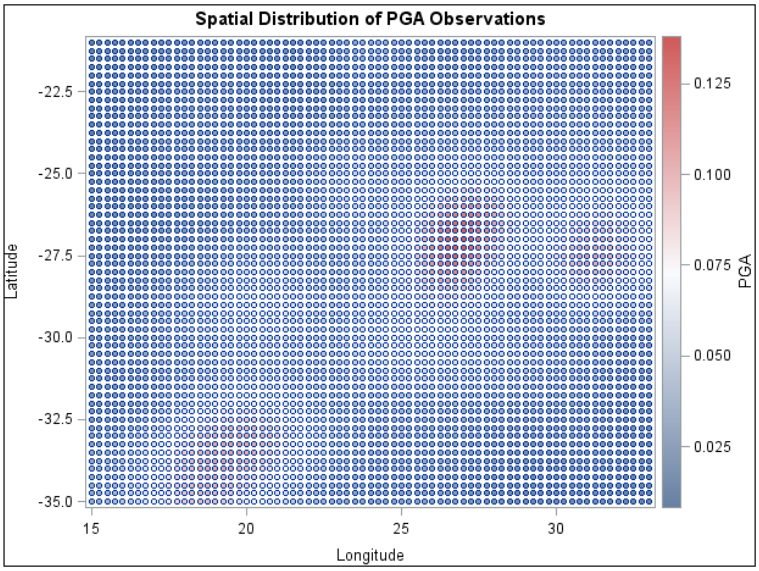


Figure 8: The spatial distribution of the peak ground acceleration for South Africa.

As has been discussed, it is necessary to first build a semi-variogram model. This can be done through the SAS® procedure, PROC VARIOGRAM.

3.1.1 Spatial Data Analysis

Before a model can be built, it is necessary to do some investigation into the nature of the data. The data is first examined to see if any surface trend exists, that is to say the data is being inspected for stationarity. It was discovered that the data used has no surface trend, therefore the process continues. Had there been a surface trend, certain calculations would have been performed to extract the trend. For more information regarding surface trend analysis see [24].
Next, it is necessary to select an appropriate lag distance and maximum lag. That is selecting the maximum distance an observation can be away from a location, in order to have an influence on the value observed at the location (max lag) and the lag distance classes (lag distance/lagd). With the PGA data set, it was found that a maximum lag distance of 0.76° and a lag distance of 15° was ideal for the model.

3.1.2 Autocorrelation Analysis

Autocorrelation analysis is then performed. This is to assess whether or not the observations are in fact related to one another. The Moran scatterplot seen in Figure 9, provides a visual representation of the spatial associations in the neighbourhood around each observation. It is evident that, due to the clustering of the observations around the line $y = x$, the observations of PGA exhibit a strong positive spatial association. This means observations found within 0.76° of one another will have similar values of PGA.

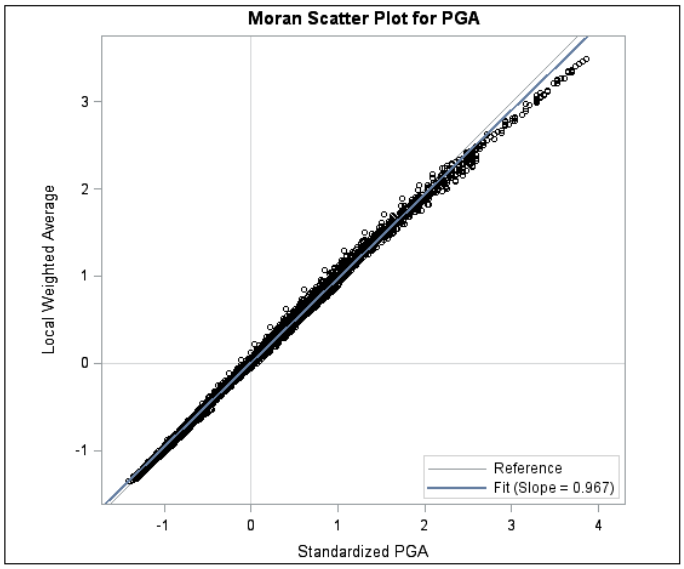


Figure 9: Analysis of autocorrelation between observations of peak ground acceleration.

3.1.3 Theoretical Semi-variogram Model Fitting

As was previously mentioned, there are multiple possibilities for the underlying structure of the semi-variogram model. It is therefore necessary to evaluate which model will be the best fit. Through analysis, it was found that the best models for the data set are the Gaussian and Exponential model. Each model was then fitted to see which one would produce the optimum result. As is clear from Figures 10 and 11, the Gaussian model fits the data in the best way, following the empirical curve quite closely.

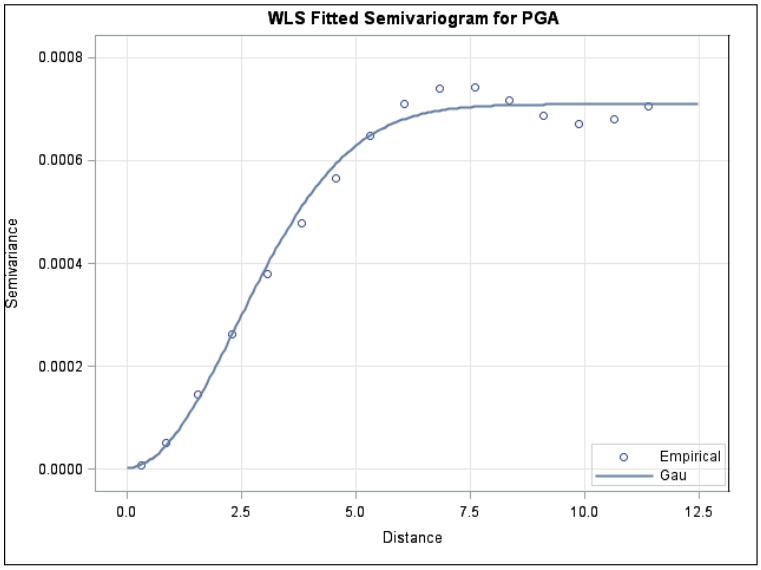


Figure 10: A Gaussian weighted least squares (WLS) fit of the peak ground acceleration observations.

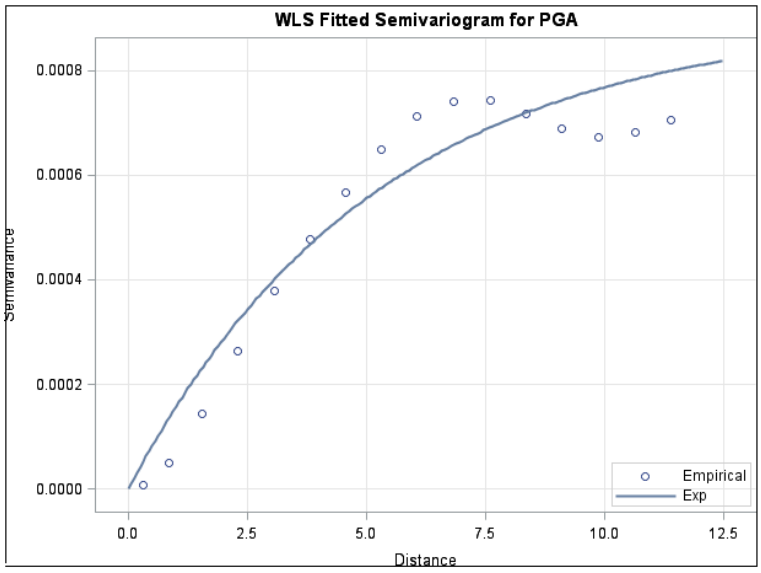


Figure 11: An Exponential weighted least squares (WLS) fit of the peak ground acceleration observations.

The information obtained for the Gaussian model is then exported to a file in SAS®to be used in the Kriging procedure.

3.2 Spatial Interpolation

A contour plot is used to identify overall trends in two dimensional univariate data [17] where an attribute, in this case the PGA, is plotted against the location (x, y) by using a pattern of coloured lines [34]. The PROC KRIGE2D procedure in SAS® produces contour plots of the Kriging estimates which will be developed from the model built in the previous step.

Local analysis is performed, i.e. the estimated value at a location is a function of the existing values around the same location [13]. This is because the local spatial behaviour differs over the space. To explain this more logically - an event occurring in the Northern Province may be very different to an event occurring in the Western Cape, due to the distance between the provinces.

A grid is chosen (in this case with increments of 0.1°) for the interpolation to give a smooth contour plot of estimations. 25 521 points were estimated by using neighbours within a 3° radius, with a maximum number of 20 neighbours used in the Kriging calculations. For easier interpretation, 1° is roughly 111 kilometres.

Figure 12 is the contour plot of the Kriging estimates established by PROC KRIGE 2D. The interpretation of a contour map involves viewing different levels on the map. A measurement found on either side of the contour line is either higher or lower than the value expressed on the line. For example, in Figure 12, considering the red area almost in the centre of the plot; the outermost contour line has a value of 0.08, implying that on that line it is estimated that the PGA will be 0.08. Anything outside of that line will be less than 0.08. This is evident by the change in colour (from red, being high PGA, to blue, lower PGA) and because the next contour line outside of the 0.08 line is 0.06. Similarly, a point found within the circle created by the contour line with the value 0.08, will have an estimated value higher than 0.08. The contour plot is essentially a plot of the Kriging estimates over a map of South Africa. The overlaying of the map of South Africa onto the contour plot was executed through QGIS® [30], and can be seen in Figure 13.

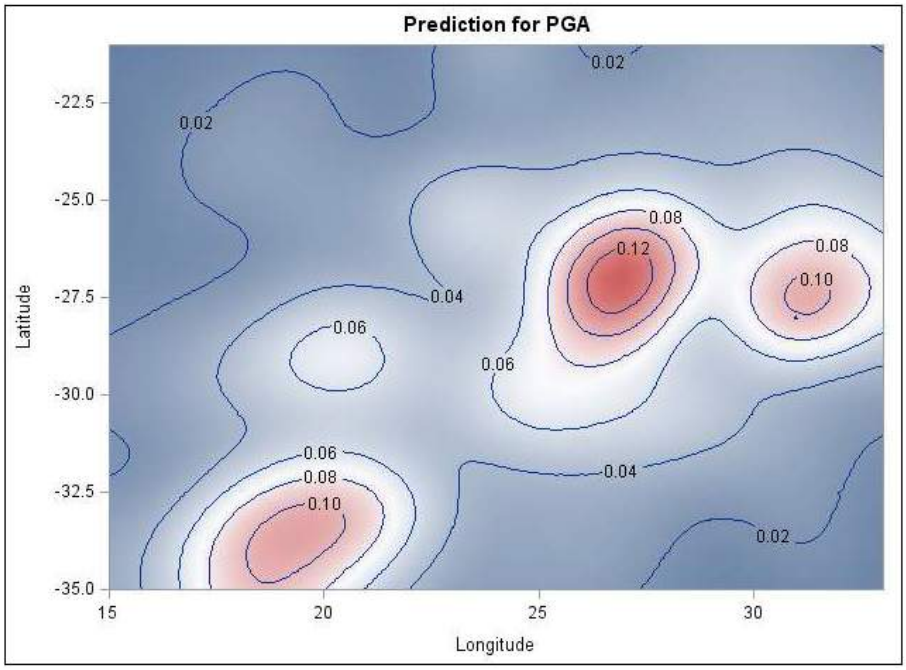


Figure 12: Contour plot of peak ground acceleration Kriging estimates.

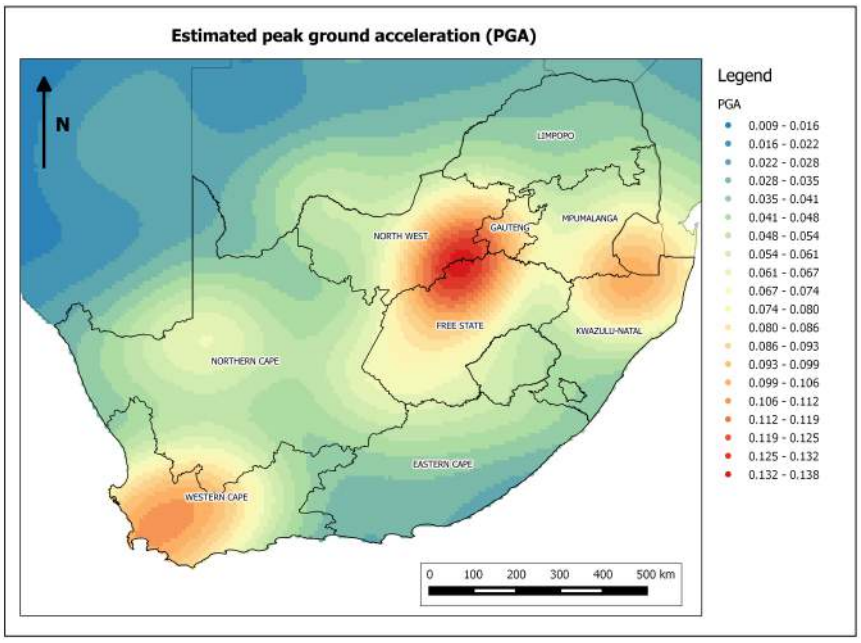


Figure 13: Estimated peak ground acceleration for South Africa.

Since the amount of PGA estimates used in the Kriging procedure is extensive, an investigation is done into what would happen to the semi-variogram model and Kriging estimates should the amount of available data reduce to a more realistic 40% of the original data.
All the steps involved in spatial modelling are repeated on the new data set and it is found that the lag distance, max lag and the model used remain the same. Figures 14, 15, 16 and 17 show the results of this application on 40% of the data. It is clear that these results are very similar to the results obtained from using the entire data set (if not the same). This shows the efficiency of the procedure even when the amount of data available is limited.

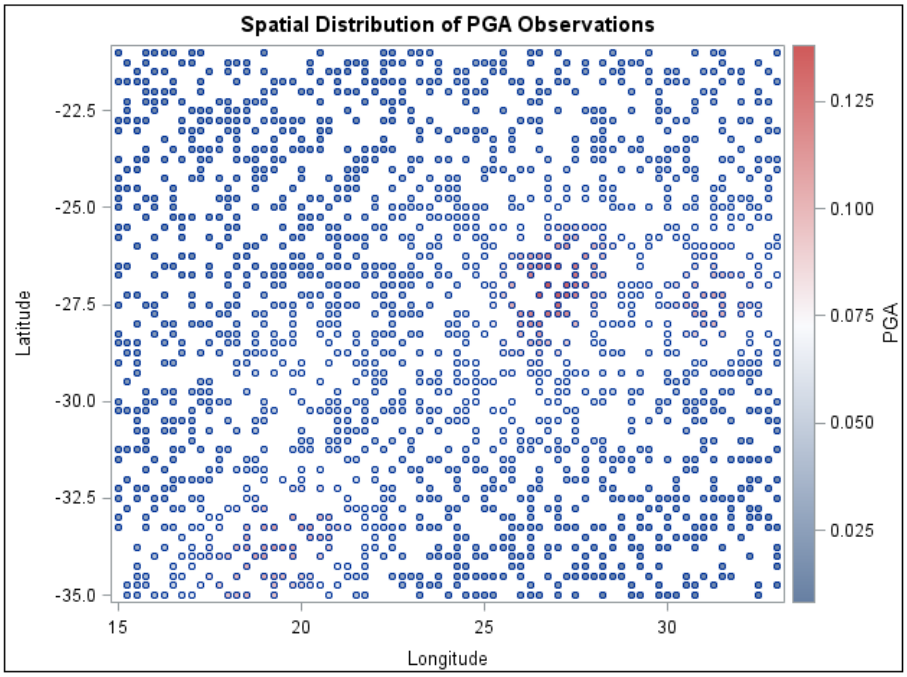


Figure 14: Spatial distribution of estimated peak ground acceleration based on 40% of the data.

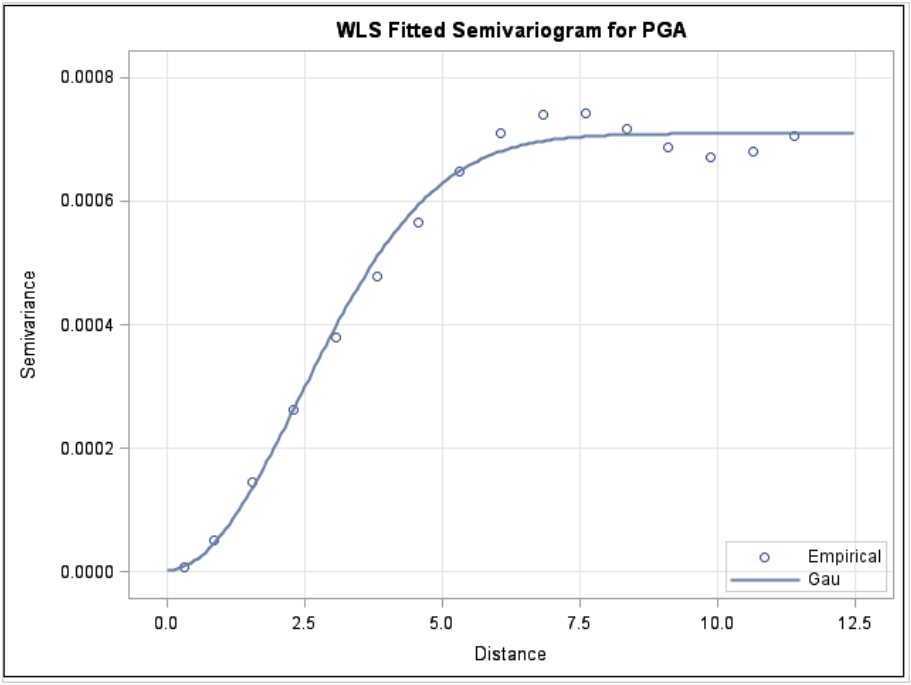


Figure 15: Gaussian weighted least squares fitted semi-variogram for peak ground acceleration based on 40% of the data.

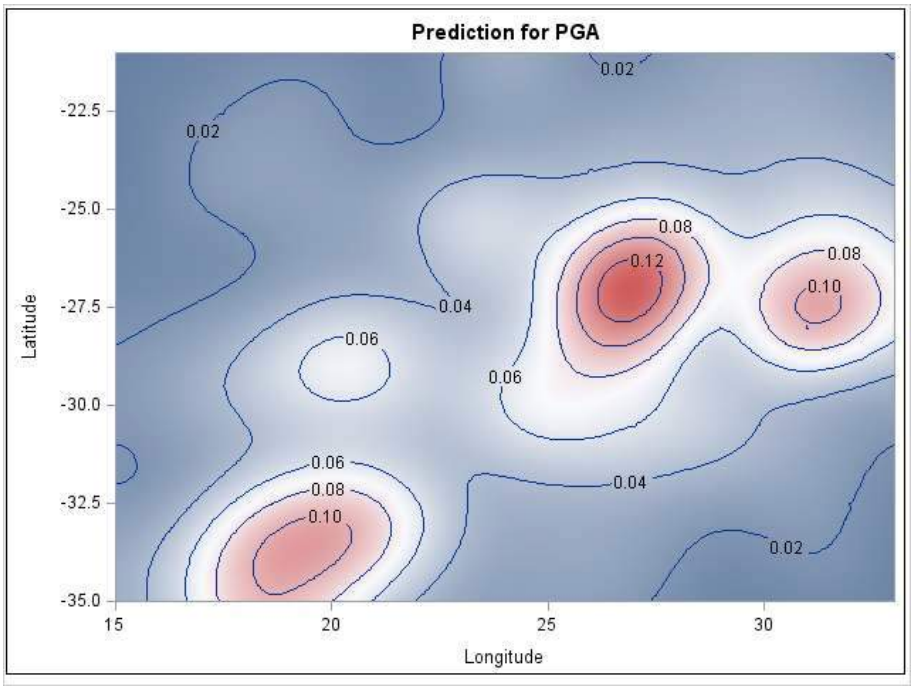


Figure 16: Contour plot of peak ground acceleration Kriging estimates based on 40% of the data.

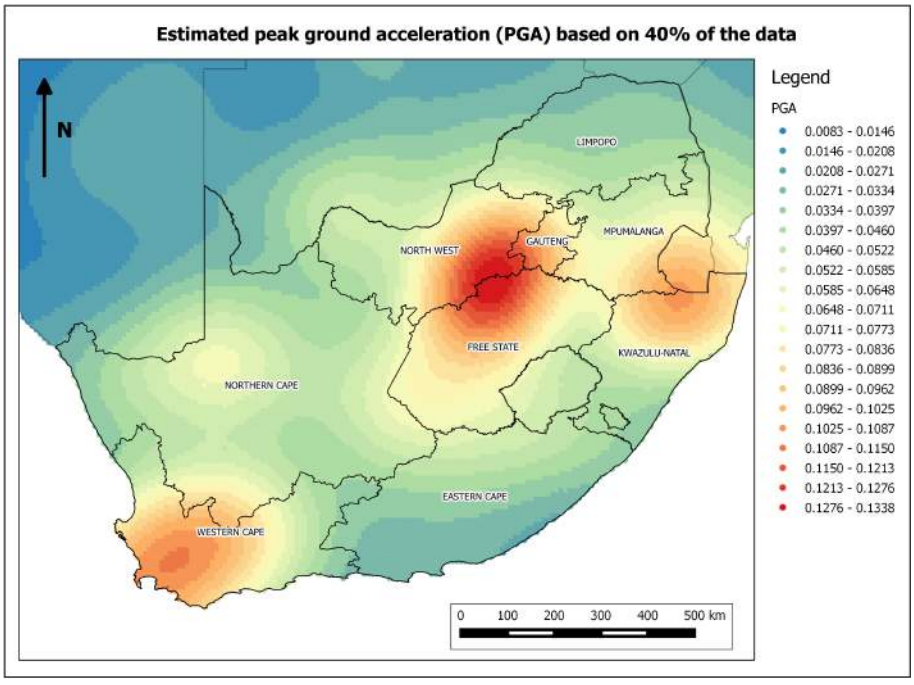


Figure 17: Estimated peak ground acceleration based on 40% of the data for South Africa.

It was decided to investigate specifically the Witwatersrand Basin's PGA, as it is an area in South Africa with a large amount of seismic activity relative to the rest of the country and, as can be seen by the map in Figure 13, has a high estimated PGA. The process for spatial modelling and spatial interpolation remains the same as before, except the lag distance changes to 0.33°. Figures 18, 19, 20 and 21 show, once again, the results of the spatial modelling and spatial prediction of PGA over the Witwatersrand basin, based on 40% of the data.
In Figure 19, it is evident that the semi-variogram doesn't seem to have an optimum fit. Although the SAS® procedure found the Gaussian fit to be the best, it will most probably be seen upon further investigation, that a nested model would be a better fit. However, it is evident in Figures 20 and 21, that the chosen semi-variogram still produces an efficient contour plot, very similar to the one produced when using all the available PGA estimates.

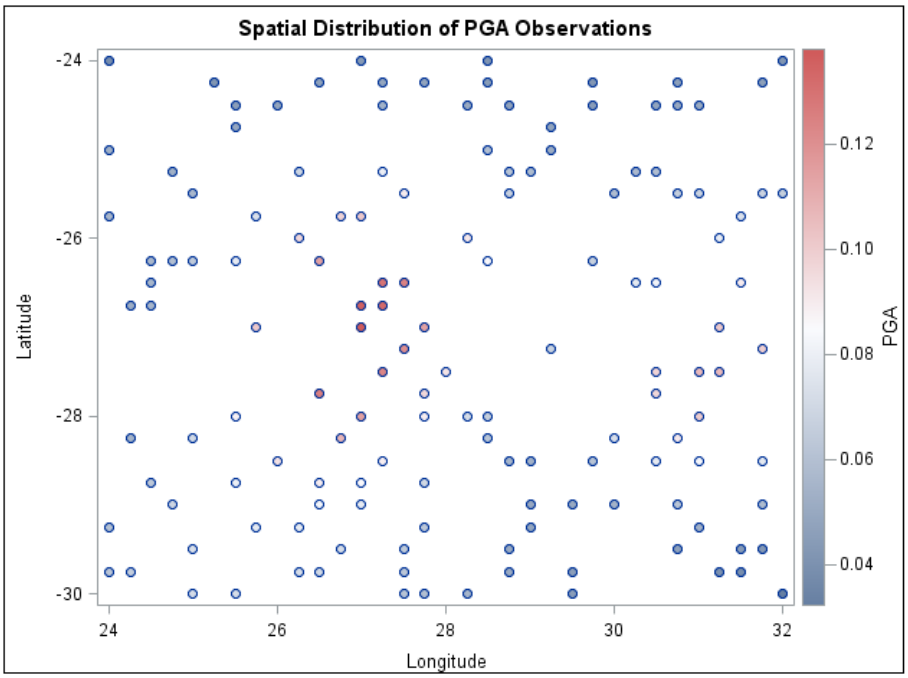


Figure 18: Spatial distribution of estimated peak ground acceleration for the Witwatersrand Basin.

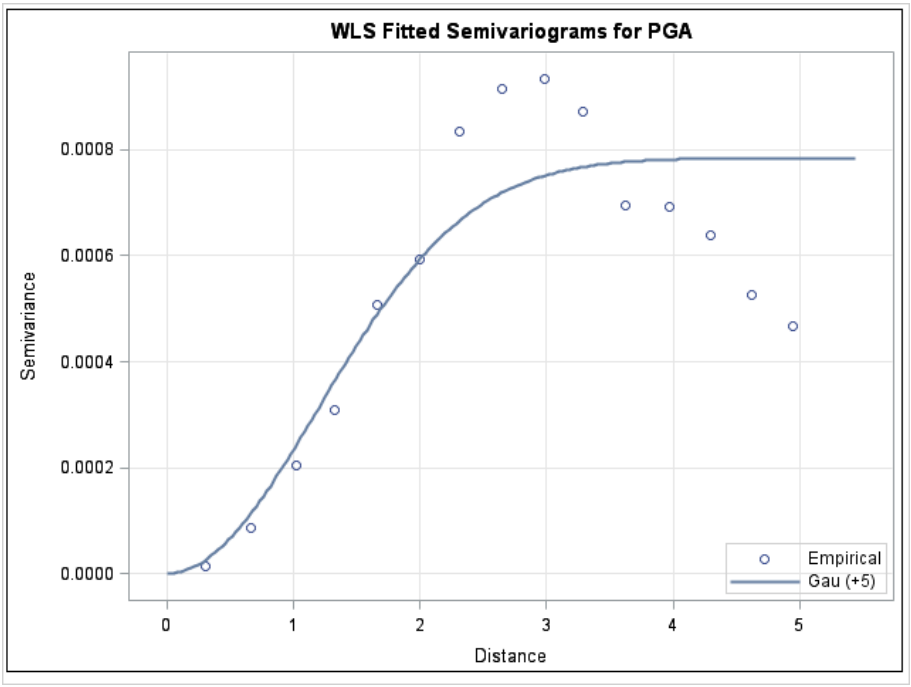


Figure 19: Gaussian weighted least squares fitted semi-variogram for estimated peak ground acceleration for the Witwatersrand Basin.

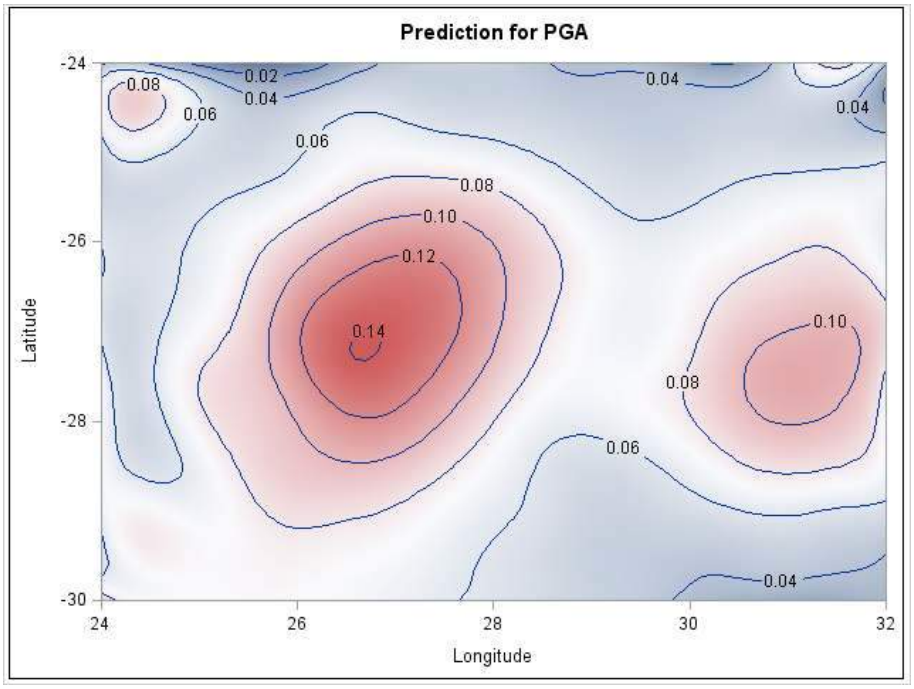


Figure 20: Contour plot of peak ground acceleration Kriging estimates for the Witwatersrand Basin.

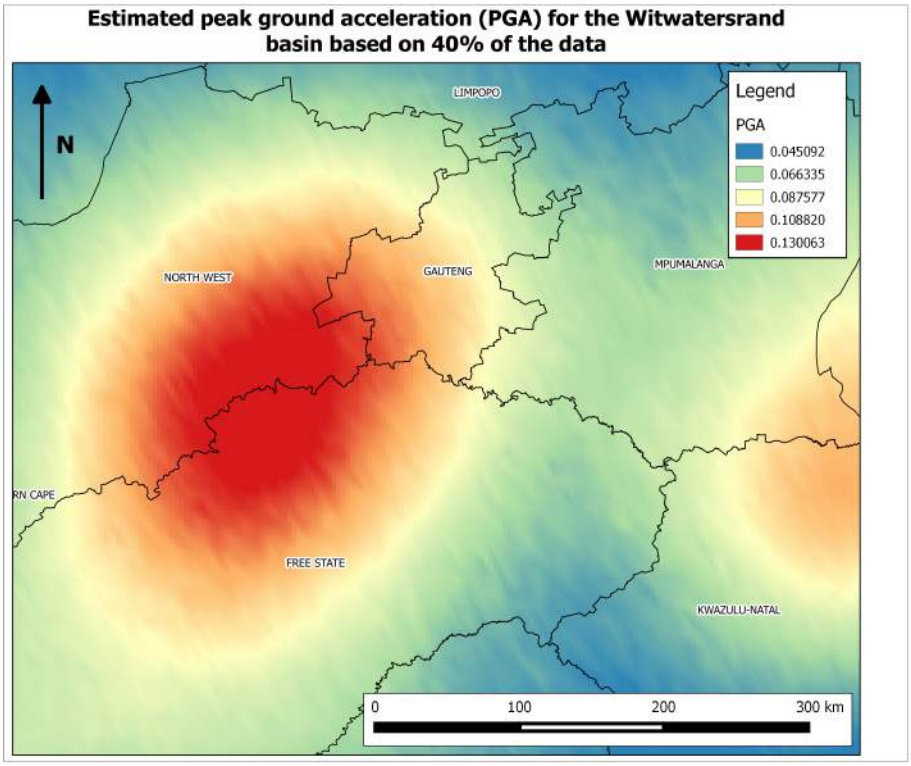


Figure 21: Estimated peak ground acceleration for the Witwatersrand Basin.

4 Conclusion

This research report investigated spatial statistics and its application in estimating the peak ground acceleration for South Africa. What was found, was that spatial statistics is a very broad field with multiple applications in many different domains. While the preparation and many processes involved in the Kriging procedure may be tedious and time-consuming, they are not done in vain. This is due to the efficiency of the kriging procedure, which produces adequate results, even when working with a limited amount of data. A contour plot of the estimated PGA for South Africa with a 10% probability of being exceeded at least once in a 50 year period, was established with an efficient Gaussian model. These estimations can be used in calculations of seismic risk, which is an interaction of seismic hazard and vulnerability.

It is noted that a more accurate model may have been obtained had the possibility of a nested semi-variogram model, comprised of the Gaussian and Exponential models, been investigated. An even more accurate result may be obtained if the time at which an event occurs is considered i.e. spatial-temporal data.
It was obvious that the estimated PGA in the Witwatersrand Basin was high, which would logically be due to an increased occurrence of seismic events; which could be attributed to the vast mining history of the area. Possible future research could include investigating the correlation between the seismic activity in the area and the mining influence such as rock type, acid water, and underground voids. This would require multivariate spatial analysis and co-kriging since this report dealt only in the univariate domain. All of these prospects have created opportunity for more research into spatial statistics.

References

[1] M Armstrong. *Basic Linear Geostatistics*. Springer Science & Business Media, 1998.

[2] M Ashraf, JC Loftis, and KG Hubbard. Application of geostatistics to evaluate partial weather station networks. *Agricultural and Forest Meteorology*, 84(3):255–271, 1997.

[3] S Banerjee, BP Carlin, and AE Gelfand. *Hierarchical Modeling and Analysis for Spatial Data*. Crc Press, 2014.

[4] A Benamghar and J Jaime. Factorial kriging of a geochemical dataset for heavy-metal spatial-variability characterization. *Environmental Earth Sciences*, 71(7):3161–3170, 2014.

[5] G Bohling. Kriging. *C&PE 940*, 1(1):1–20, October 2005.

[6] G Christakos. *Random Field Models in Earth Sciences*. Courier Corporation, 2012.

[7] I Clark. *Practical Geostatistics*, volume 3. Applied Science Publishers London, 1979.

[8] N Cressie. The Origins of Kriging. *Mathematical Geology*, 22(3):239–252, 1990.

[9] N Cressie and NA Cassie. *Statistics for Spatial Data*, volume 900. Wiley New York, 1993.

[10] N Cressie and CK Wikle. *Statistics for Spatio-Temporal Data*. John Wiley & Sons, 2011.

[11] PJ Curran. The semivariogram in remote sensing: An introduction. *Remote Sensing of Environment*, 24(3):493–507, 1988.

[12] PA Dowd. Lognormal kriging: The general case. *Mathematical Geology*, 14(5):475–499, 1982.

[13] AS Fotheringham and C Brunsdon. Local forms of spatial analysis. *Geographical Analysis*, 31(4):340–358, 1999.

[14] AE Gelfand, P Diggle, P Guttorp, and M Fuentes. *Handbook of Spatial Statistics*. CRC Press, 2010.

[15] H Gilgen. *Univariate Time Series in Geosciences*. Springer, 2006.

[16] P Goovaerts. *Geostatistics for Natural Resources Evaluation*. Oxford University Press, 1997.

[17] EH Isaaks and RM Srivastava. *Applied Geostatistics*, volume 2. Oxford University Press New York, 1989.

[18] AG Journel and CJ Huijbregts. *Mining Geostatistics*. Academic press, 1978.

[19] A Kijko, SJP Retief, and G Graham. Seismic hazard and risk assessment for Tulbagh, South Africa: Part ii–Assessment of seismic risk. *Natural Hazards*, 30(1):25–41, 2003.

[20] DG Krige. *A Statistical Approach to Some Mine Valuation and Allied Problems on the Witwatersrand: By DG Krige*. PhD thesis, University of the Witwatersrand, 1951.

[21] DG Krige. *Lognormal-de Wijsian Geostatistics for Ore Evaluation*. South African Institute of Mining and Metallurgy Johannesburg, 1981.

[22] NS Lam. Spatial interpolation methods: A review. *The American Cartographer*, 10(2):129–150, 1983.

[23] AS Lefohn, HP Knudsen, and DS Shadwick. Using ordinary Kriging to estimate the seasonal w126, and n100 24-h concentrations for the year 2000 and 2003. *ASL & Associates*, 111(1):2–3, October 2005.

[24] P Legendre and MJ Fortin. Spatial pattern and ecological analysis. *Vegetatio*, 80(2):107–138, 1989.

[25] CD Lloyd and PM Atkinson. Archaeology and geostatistics. *Journal of Archaeological Science*, 31(2):151–165, 2004.

[26] G Matheron. Principles of Geostatistics. *Economic Geology*, 58(8):1246–1266, 1963.

[27] G Matheron. *The Theory of Regionalized Variables and its Applications*, volume 5. Ecole nationale supérieure des mines de Paris, 1971.

[28] AB McBratney and R Webster. Choosing functions for semi-variograms of soil properties and fitting them to sampling estimates. *Journal of Soil Science*, 37(4):617–639, 1986.

[29] B Minasny and AB McBratney. Spatial prediction of soil properties using eblup with the matérn covariance function. *Geoderma*, 140(4):324–336, 2007.

[30] QGIS Development Team. *QGIS Geographic Information System*. Open Source Geospatial Foundation, 2009.

[31] M Rivest, D Marcotte, and P Pasquier. Interpolation of concentration measurements by Kriging using flow coordinates. In *Geostatistics Oslo 2012*, pages 519–530. Springer, 2012.

[32] O Schabenberger and CA Gotway. *Statistical Methods for Spatial Data Analysis*. CRC press, 2004.

[33] D Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference*, pages 517–524. ACM, 1968.

[34] A Smit. Interpolation in stationary spatial and spatial-temporal datasets. Master's thesis, University of Pretoria, 2010.

[35] A Stein, F van der Meer, and B Gorte. *Spatial Statistics for Remote Sensing*, volume 1. Springer Science & Business Media, 1999.

[36] ML Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer Science & Business Media, 2012.

[37] WR Tobler. A computer movie simulating urban growth in the Detroit region. *Economic Geography*, 46:234–240, 1970.

[38] BB Trangmar, RS Yost, and G Uehara. Application of geostatistics to spatial studies of soil properties. *Advances in Agronomy*, 38(1):45–94, 1985.

[39] Z Wang. Seismic hazard vs. seismic risk. *Seismological Research Letters*, 80(5):673–674, 2009.

[40] R Webster and MA Oliver. *Geostatistics for Environmental Scientists*. John Wiley & Sons, 2007.

Appendix A

```
title1 'Spatial Prediction of Peak Ground Acceleration';

ods graphics on;

****Theoretical Semivariogram Fitting****;
****Exponential****;
proc variogram data=sasuser.PGA;
   store out=SemivarStoreExponential / label='PGA Exponential WLS Fit';
   compute lagd=0.76 maxlag=15;
   coordinates xc=Longitude yc=Latitude;
   model form=exp cl / covb;
   var PGA;
run;
title2 "Fitting Gaussian Model";
proc variogram data=sasuser.PGA;
   store out=SemivarStoreGaussian / label='PGA Exponential WLS Fit';
   compute lagd=0.76 maxlag=15;
   coordinates xc=Longitude yc=Latitude;
   model form=gau cl / covb;
   var PGA;
run;

****Kriging****;
title3 'Kriging based on the Gaussian Semi-variogram';
proc krige2d data=Sasuser.PGA plots(only)=
   (pred(fill=pred line=pred obs=linegrad));
   restore in=SemivarStoreGaussian;
   coordinates xc=Longitude yc=Latitude;
   predict var=PGA radius = 3; *Radius makes it local;
   SelModel: model storeselect;
   grid x= 15 to 33 by 0.1 y=-35 to -21 by 0.1;
run;

****40%****;
proc surveyselect data=Sasuser.PGA
   method=srs n=1664 /*Roughly 40% of 4161*/ out=Sasuser.PGA_40;
run;
title1"Variogram for 40% of the data";
proc variogram data=sasuser.PGA_40 plot=pairs(mid);
   compute novariogram nhc=30;
   coordinates xc=Longitude yc=Latitude;
   var PGA;
run;

****Gaussian Model****;
proc variogram data=sasuser.PGA_40;
   store out=SemivarStoreGaussian_40 /
        label='40% of Data - PGA Exponential WLS Fit';
   compute lagd=0.76 maxlag=15;
   coordinates xc=Longitude yc=Latitude;
  model form=gau cl / covb;
```

```
   var PGA;
run;
title1 "Kriging 40% of the data";
proc krige2d data=Sasuser.PGA_40 plots(only)=
   (pred(fill=pred line=pred)); /*Predictions*/
   restore in=SemivarStoreGaussian_40;
   coordinates xc=Longitude yc=Latitude;
   predict var=PGA radius = 3; *Radius makes it local;
   SelModel: model storeselect;
   grid x= 15 to 33 by 0.1 y=-35 to -21 by 0.1;
run;
title1 "Developing the variogram based on all the data";
data Sasuser.PGA_Basin;
        set Sasuser.PGA;
        if Latitude > -24 | Latitude < -30 then delete;
        if Longitude > 32 | Longitude < 24 then delete;
run;
 ****Checking lags****;
proc variogram data=sasuser.PGA_Basin plot=pairs(mid);
   compute novariogram nhc=30;
   coordinates xc=Longitude yc=Latitude;
   var PGA;
run;

proc variogram data=sasuser.PGA_Basin plots(only)=semivar;
   compute lagd=0.33 maxlag=15 ;
   coordinates xc=Longitude yc=Latitude;
   var PGA;
run;

****Computing empirical semivariogram with
95% confidence limits for classical semivariance;
proc variogram data=sasuser.PGA_Basin outv=sasuser.SCM_Basin;
   compute lagd=0.33 maxlag=15 cl robust;
   coordinates xc=Longitude yc=Latitude;
   var PGA;
run;

proc variogram data=sasuser.PGA_Basin;
   store out=SemivarStoreGaussian_Basin /
        label='PGA_Basin Exponential WLS Fit';
   compute lagd=0.33 maxlag=15;
   coordinates xc=Longitude yc=Latitude;
  model form=auto(mlist=(exp,gau,mat) nest=1 to 2);
   var PGA;
run;

title1 "Kriging based on all the data";

proc krige2d data=Sasuser.PGA_Basin plots(only)=
                          (pred(fill=pred line=pred));
   restore in=SemivarStoreGaussian_Basin;
   coordinates xc=Longitude yc=Latitude;
```

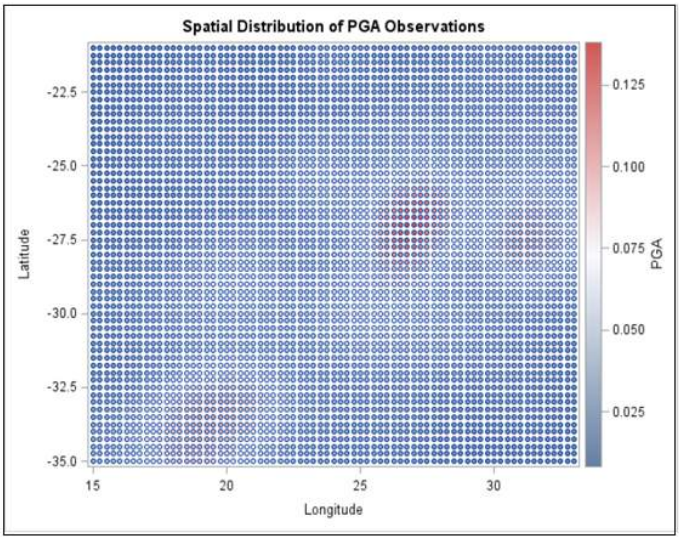
```
   predict var=PGA;
   SelModel: model storeselect;
   grid x= 24 to 32 by 0.1 y=-30 to -24 by 0.1;
run;


****Selecting 40% of the data****;
*title1 "Randomly selecting 40% of the data";
proc surveyselect data=Sasuser.PGA_Basin
   method=srs n=143 /*Roughly 40% of 357*/
        out=Sasuser.PGA__Basin_40;
run;


title1"Developing Variogram based on  for 40% of the data";
proc variogram data=Sasuser.PGA__Basin_40 plot=pairs(mid);
   compute novariogram nhc=30;
   coordinates xc=Longitude yc=Latitude;
   var PGA;
run;

****Gaussian Model****;
proc variogram data=Sasuser.PGA__Basin_40;
   store out=SemivarStoreGaussian_Basin_40 /
        label='40% of Data - PGA Exponential WLS Fit';
   compute lagd=0.33 maxlag=15;
   coordinates xc=Longitude yc=Latitude;
  model form=auto(mlist=(exp,gau,mat) nest=1 to 2);
   var PGA;
run;
title1 "Kriging based on 40% of the data";
proc krige2d data=Sasuser.PGA__Basin_40 plots(only)=
        (pred(fill=pred line=pred)
        pred(fill=se line=se obs=linegrad)); /*Predictions*/
   restore in=SemivarStoreGaussian_Basin_40;
   coordinates xc=Longitude yc=Latitude;
   predict var=PGA ;
   SelModel: model storeselect;
   grid x= 24 to 32 by 0.1 y=-30 to -24 by 0.1;
run;

ods graphics off;
```

Appendix B

Spatial Prediction of Peak Ground Acceleration

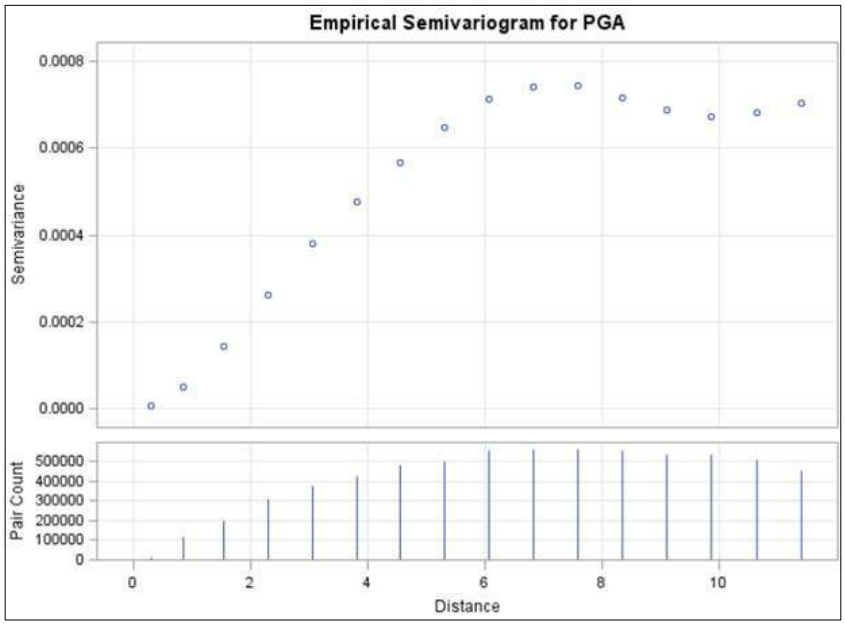
The VARIOGRAM Procedure
Dependent Variable: PGA

Number of Observations Read	4161
Number of Observations Used	4161



Empirical Semivariogram

Lag Class	Pair Count	Average Distance	Semivariance
0	16256	0.301	6.383E-6
1	116498	0.851	0.000050
2	197788	1.548	0.00014
3	308450	2.289	0.00026
4	376856	3.061	0.00038
5	421950	3.813	0.00048
6	479608	4.563	0.00057
7	499008	5.307	0.00065
8	556332	6.064	0.00071
9	559450	6.835	0.00074
10	561608	7.594	0.00074
11	556762	8.351	0.00072
12	530620	9.098	0.00069
13	534300	9.860	0.00067
14	503178	10.632	0.00068
15	455100	11.397	0.00070



Spatial Prediction of Peak Ground Acceleration

The VARIOGRAM Procedure
Dependent Variable: PGA
Angle: Omnidirectional
Current Model: Gaussian

Semivariogram Model Fitting	
Name	Gaussian
Label	Gau
Output Item Store	WORK.SEMIVARSTOREGAUSSIAN
Item Store Label	PGA Exponential WLS Fit

Model Information	
Parameter	**Initial Value**
Nugget	0
Scale	0.000686
Range	5.6987

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	3
Lower Boundaries	3
Upper Boundaries	0
Starting Values From	PROC

Spatial Prediction of Peak Ground Acceleration

The VARIOGRAM Procedure
Dependent Variable: PGA
Angle: Omnidirectional
Current Model: Gaussian

Dual Quasi-Newton Optimization

Dual Broyden - Fletcher - Goldfarb - Shanno Update (DBFGS)

Hessian Computed by Finite Differences (Using Analytic Gradient)

Optimization Results			
Iterations	20	Function Calls	134
Gradient Calls	0	Active Constraints	0
Objective Function	7091.4340073	Max Abs Gradient Element	90.777322811
Slope of Search Direction	-8.350799E-7		

Convergence criterion (GCONV=1E-8) satisfied.

Note: At least one element of the gradient is greater than 1e-3.

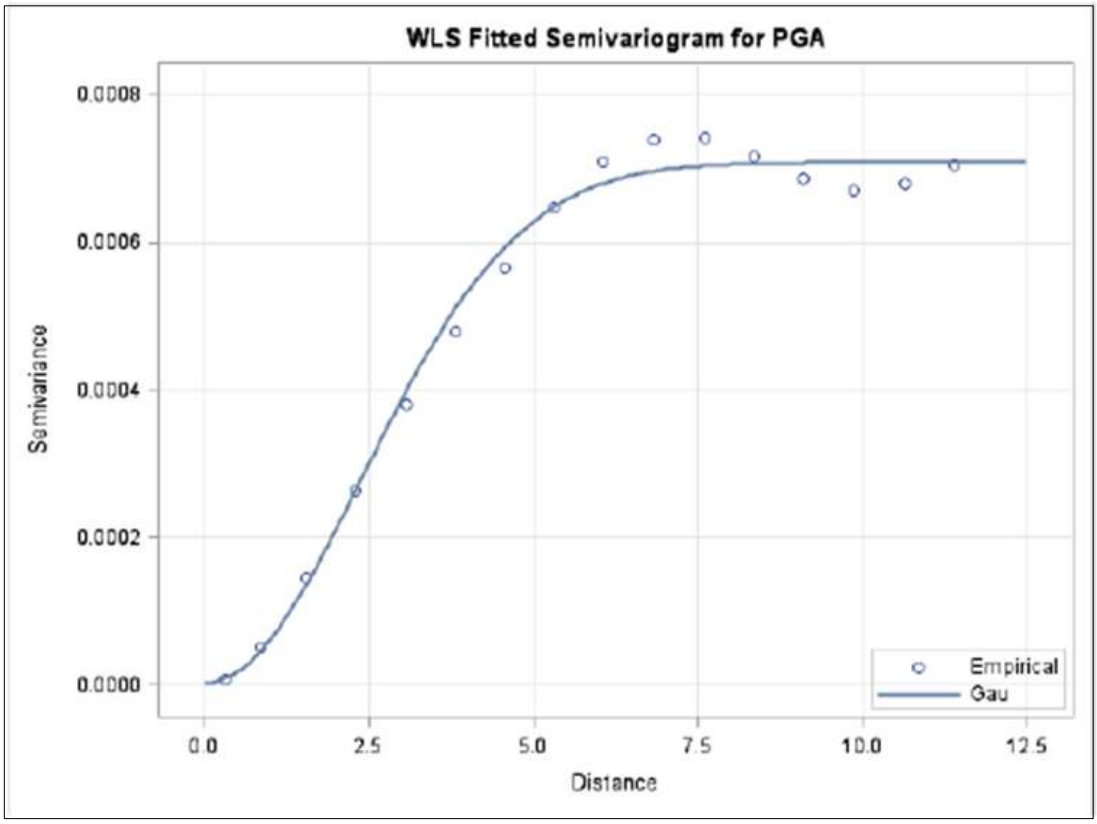
Parameter Estimates								
			Approximate 95% Confidence Limits					
Parameter	Estimate	Approx Std Error	Lower	Upper	DF	t Value	Approx Pr > |t|	Gradient
Nugget	2.398E-6	0	2.398E-6	2.398E-6	13	.	.	-90.4051
Scale	0.000707	3.688E-7	0.000706	0.000708	13	1917.40	<.0001	90.77732
Range	3.3896	0.003624	3.3817	3.3974	13	935.41	<.0001	-0.00829

Approximate Covariance Matrix			
Parameter	Nugget	Scale	Range
Nugget	0.0000	0.0000	0.0000
Scale	0.0000	0.0000	0.0000
Range	0.0000	0.0000	0.0000

Fit Summary		
Model	Weighted SSE	AIC
Gau	7091.4	103.50487

Spatial Prediction of Peak Ground Acceleration

The VARIOGRAM Procedure
Dependent Variable: PGA



Spatial Prediction of Peak Ground Acceleration

The KRIGE2D Procedure

Correlation Model Item Store Information	
Input Item Store	WORK.SEMIVARSTOREGAUSSIAN
Item Store Label	PGA Exponential WLS Fit
Data Set Created From	SASUSER.PGA
By-group Information	No By-groups Present
Created By	PROC VARIOGRAM
Date Created	01NOV15:12:12:30

Spatial Prediction of Peak Ground Acceleration

The KRIGE2D Procedure
Dependent Variable: PGA

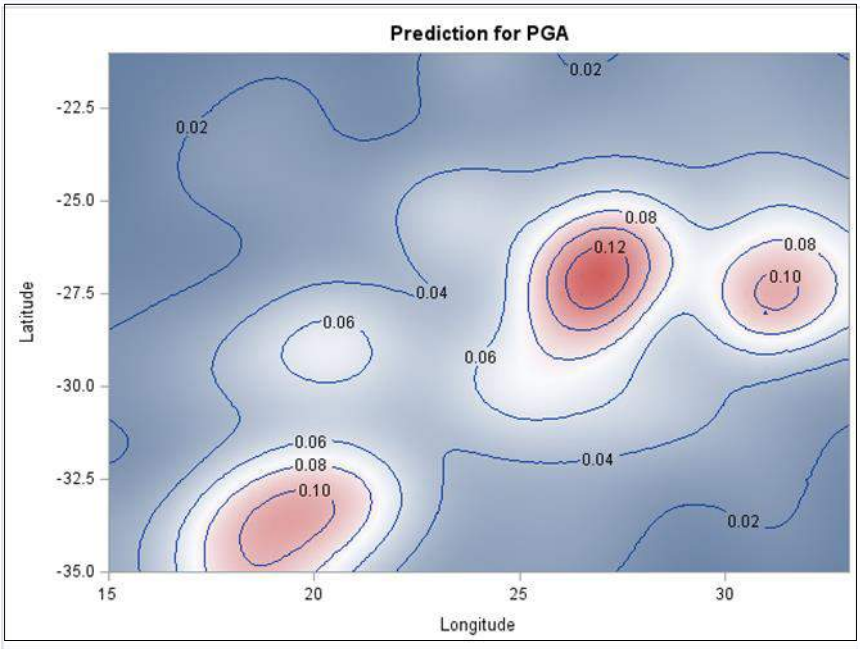
Number of Observations Read	4161
Number of Observations Used	4161

Kriging Information	
Prediction Grid Points	25521
Type of Analysis	Local
Neighborhood Search Radius	3
Minimum Neighbors	20
Maximum Neighbors	All Within Radius

Spatial Prediction of Peak Ground Acceleration

The KRIGE2D Procedure
Dependent Variable: PGA
Prediction: Pred1, Model: SelModel

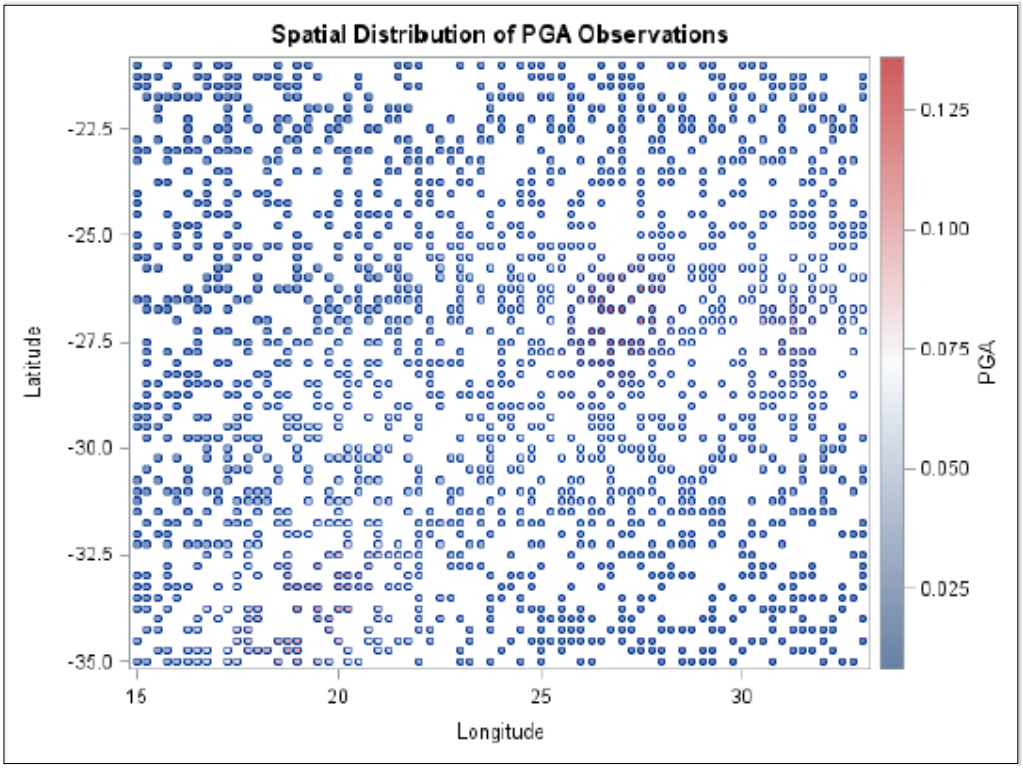
Covariance Model Information	
Type	Gaussian
Sill	0.0007071
Range	3.3895593
Effective Range	5.8708889
Nugget Effect	2.3978E-6



Variogram for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA

Number of Observations Read	1664
Number of Observations Used	1664



Pairs Information	
Number of Lags	31
Lag Distance	0.76
Maximum Data Distance in Longitude	18.00
Maximum Data Distance in Latitude	14.00
Maximum Data Distance	22.80

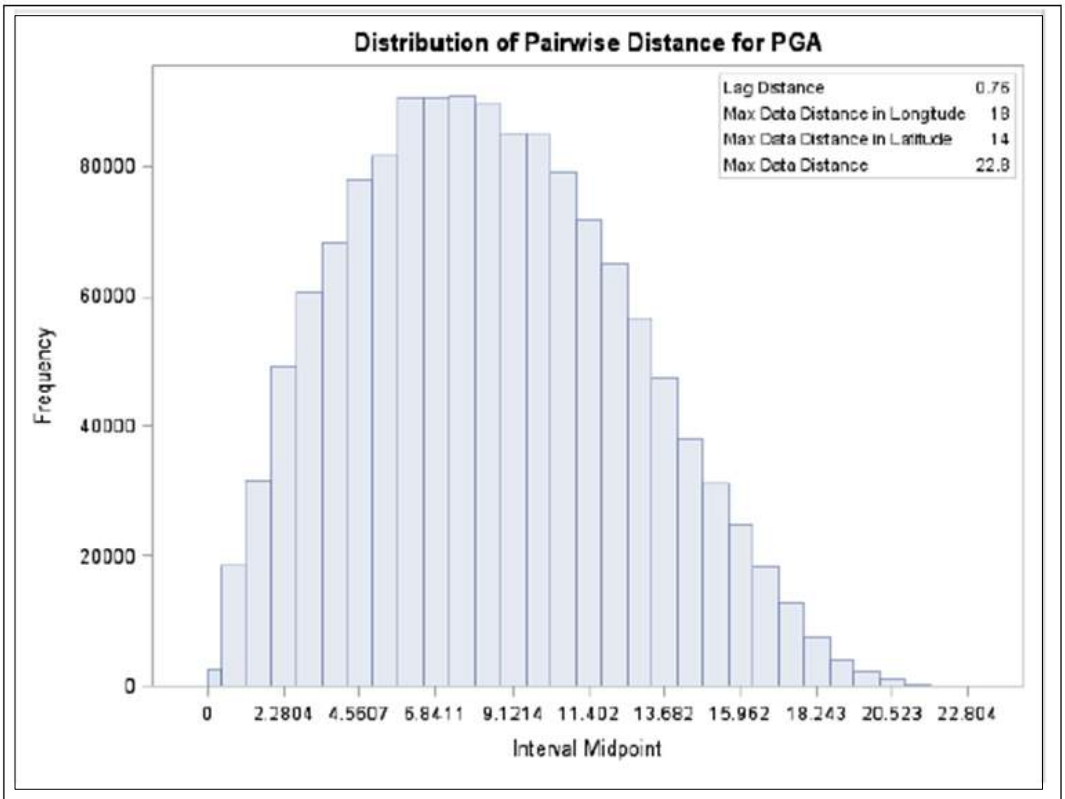
Variogram for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA
Angle: Omnidirectional
Current Model: Gaussian

Semivariogram Model Fitting	
Name	Gaussian
Label	Gau
Output Item Store	WORK.SEMIVARSTOREGAUSSIAN_40
Item Store Label	40% of Data - PGA Exponential WLS Fit

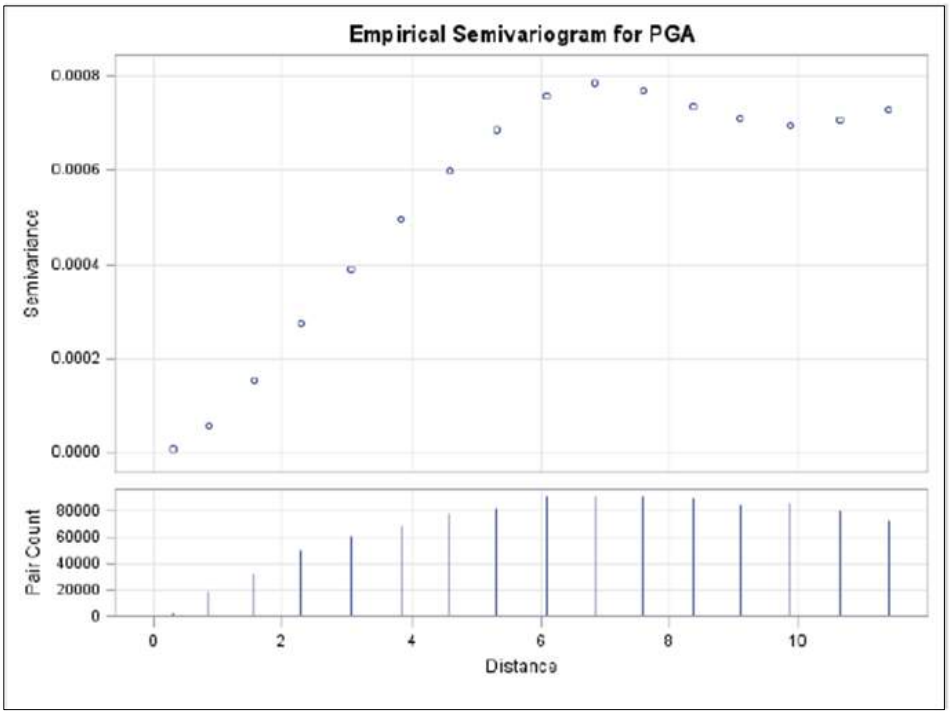
Model Information	
Parameter	Initial Value
Nugget	0
Scale	0.000709
Range	5.6980

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	3
Lower Boundaries	3
Upper Boundaries	0
Starting Values From	PROC



Variogram for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA



Dual Quasi-Newton Optimization

Dual Broyden - Fletcher - Goldfarb - Shanno Update (DBFGS)

Hessian Computed by Finite Differences (Using Analytic Gradient)

Optimization Results			
Iterations	18	Function Calls	93
Gradient Calls	0	Active Constraints	0
Objective Function	1515.3972556	Max Abs Gradient Element	172.65992943
Slope of Search Direction	-6.101074E-7		

Convergence criterion (GCONV=1E-8) satisfied.

Note: At least one element of the gradient is greater than 1e-3.

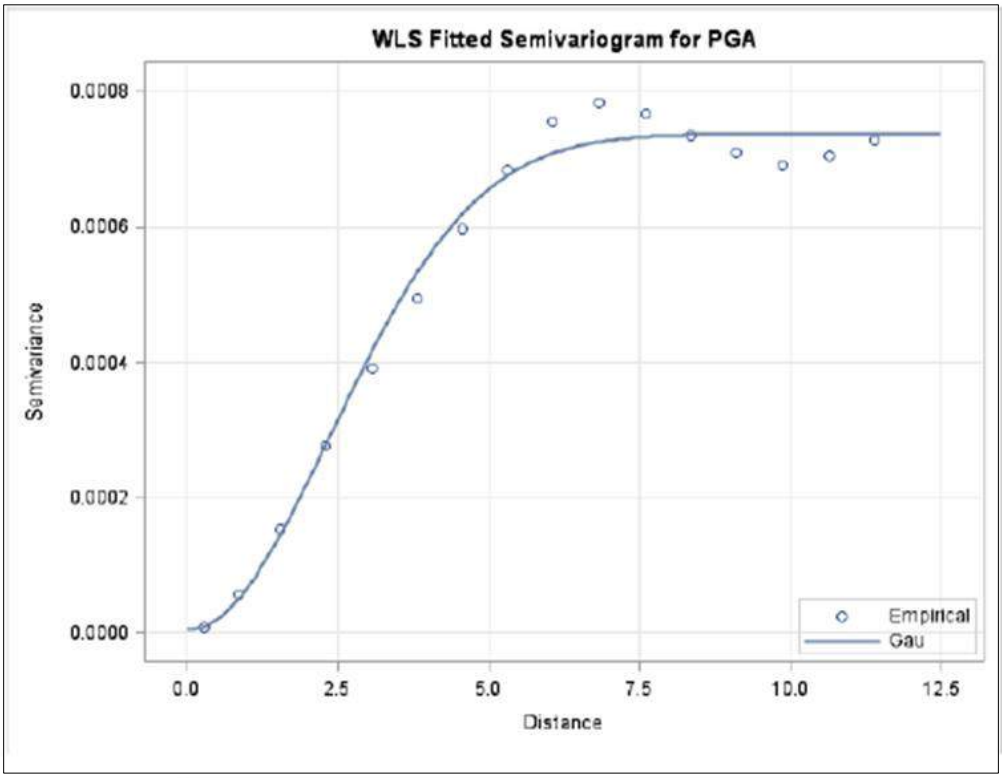
Parameter Estimates								
			Approximate 95% Confidence Limits					
Parameter	Estimate	Approx Std Error	Lower	Upper	DF	t Value	Approx Pr > |t|	Gradient
Nugget	3.833E-6	0	3.833E-6	3.833E-6	13	.	.	172.6599
Scale	0.000735	9.731E-7	0.000733	0.000737	13	755.06	<.0001	45.84643
Range	3.3761	0.009410	3.3558	3.3964	13	358.79	<.0001	-0.00806

Approximate Covariance Matrix			
Parameter	Nugget	Scale	Range
Nugget	0.0000	0.0000	0.0000
Scale	0.0000	0.0000	0.0000
Range	0.0000	0.0000	0.0001

Fit Summary		
Model	Weighted SSE	AIC
Gau	1515.4	78.81351

Variogram for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA



Kriging 40% of the data

The KRIGE2D Procedure

Correlation Model Item Store Information	
Input Item Store	WORK.SEMIVARSTOREGAUSSIAN_40
Item Store Label	40% of Data - PGA Exponential WLS Fit
Data Set Created From	SASUSER.PGA_40
By-group Information	No By-groups Present
Created By	PROC VARIOGRAM
Date Created	01NOV15:12:23:10

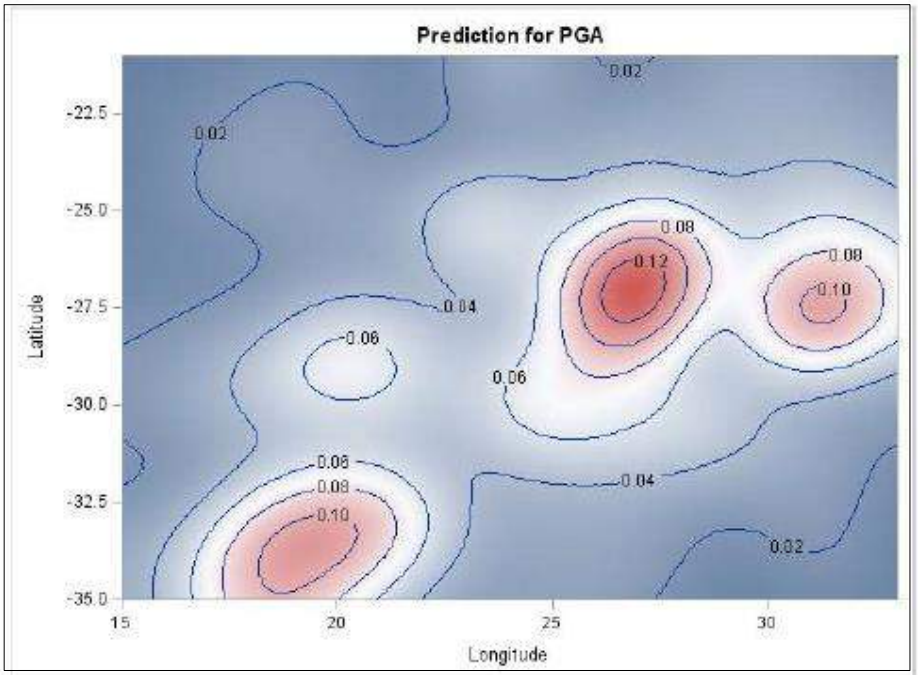
Number of Observations Read	1664
Number of Observations Used	1664

Kriging Information	
Prediction Grid Points	25521
Type of Analysis	Local
Neighborhood Search Radius	3
Minimum Neighbors	20
Maximum Neighbors	All Within Radius

Kriging 40% of the data

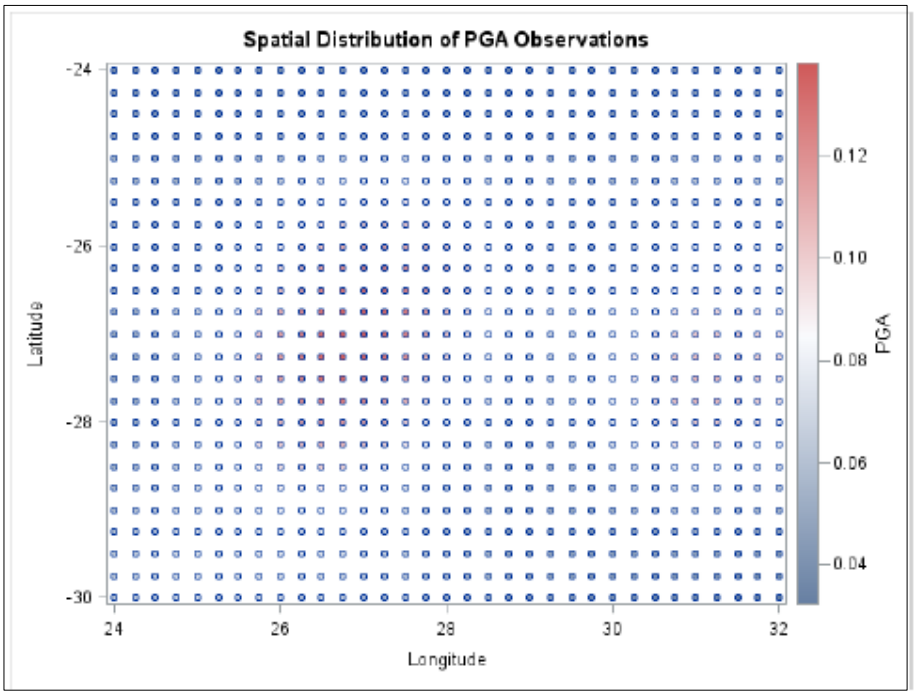
The KRIGE2D Procedure
Dependent Variable: PGA
Prediction: Pred1, Model: SelModel

Covariance Model Information	
Type	Gaussian
Sill	0.0007347
Range	3.3761192
Effective Range	5.8476099
Nugget Effect	3.8326E-6

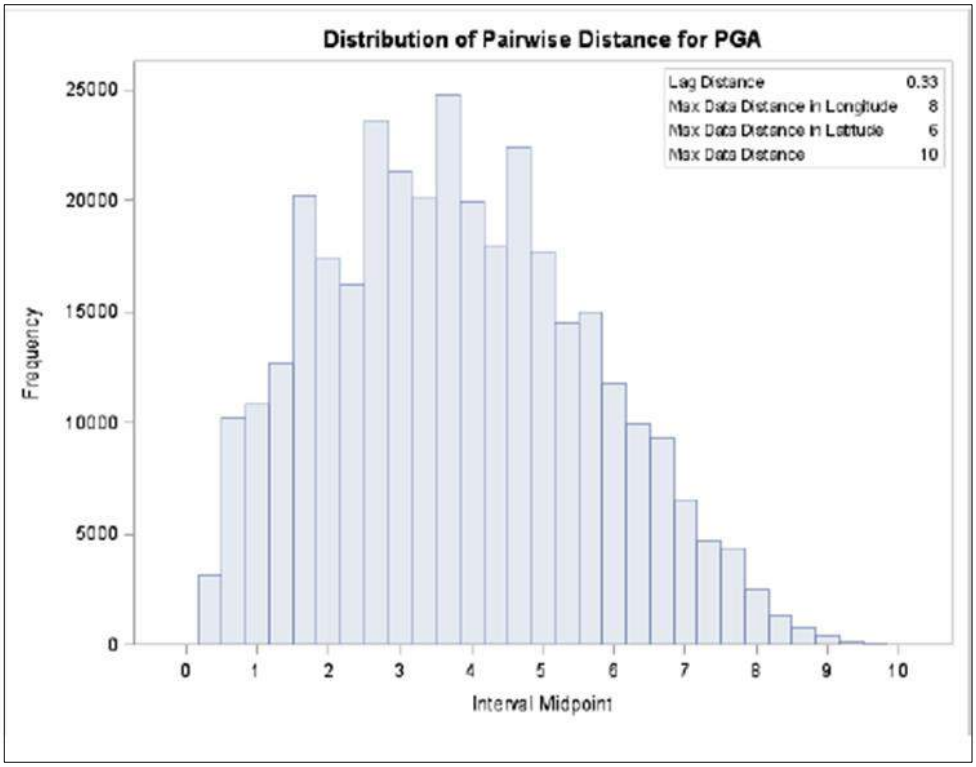


The VARIOGRAM Procedure
Dependent Variable: PGA

Number of Observations Read	825
Number of Observations Used	825



Pairs Information	
Number of Lags	31
Lag Distance	0.33
Maximum Data Distance in Longitude	8.00
Maximum Data Distance in Latitude	6.00
Maximum Data Distance	10.00



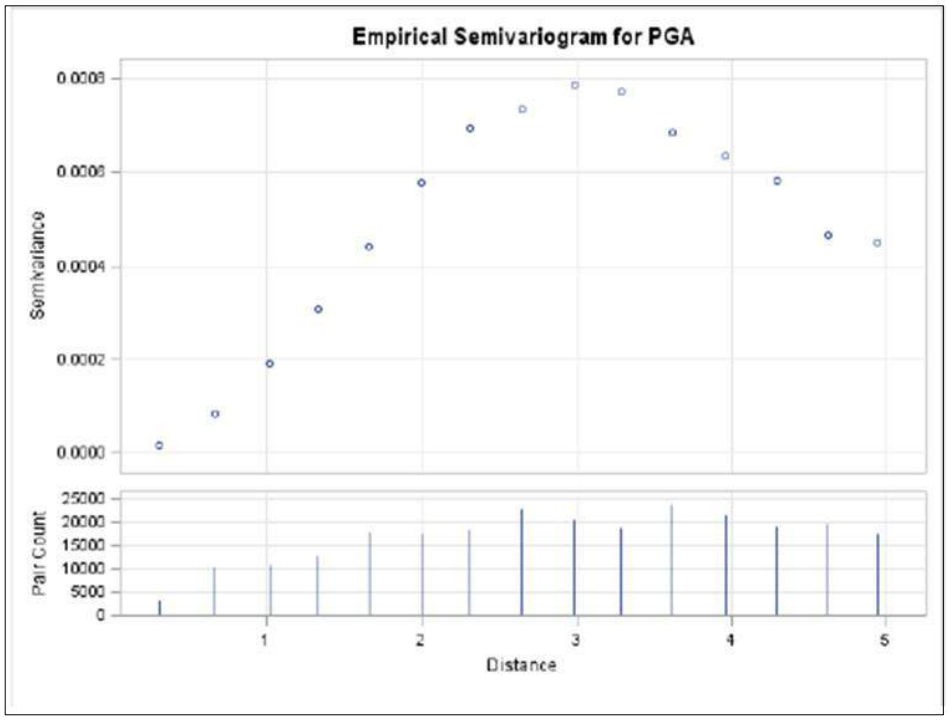
The VARIOGRAM Procedure
Dependent Variable: PGA

Number of Observations Read	825
Number of Observations Used	825

Empirical Semivariogram

Lag Class	Pair Count	Average Distance	Semivariance
0	0	.	.
1	3128	0.301	0.000017
2	10244	0.663	0.000083
3	10854	1.019	0.00019
4	12698	1.330	0.00031
5	17898	1.658	0.00044
6	17640	1.994	0.00058
7	18284	2.310	0.00069
8	22758	2.645	0.00074
9	20524	2.983	0.00079
10	18852	3.289	0.00077
11	23736	3.619	0.00068
12	21630	3.966	0.00064
13	19202	4.295	0.00058
14	19754	4.622	0.00047
15	17484	4.946	0.00045

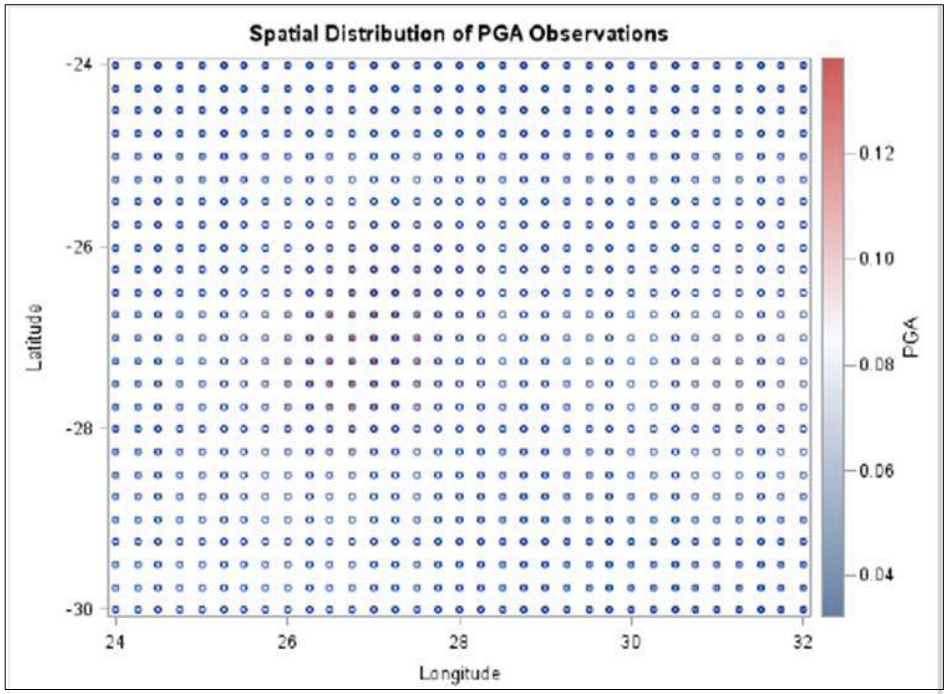
The VARIOGRAM Procedure
Dependent Variable: PGA



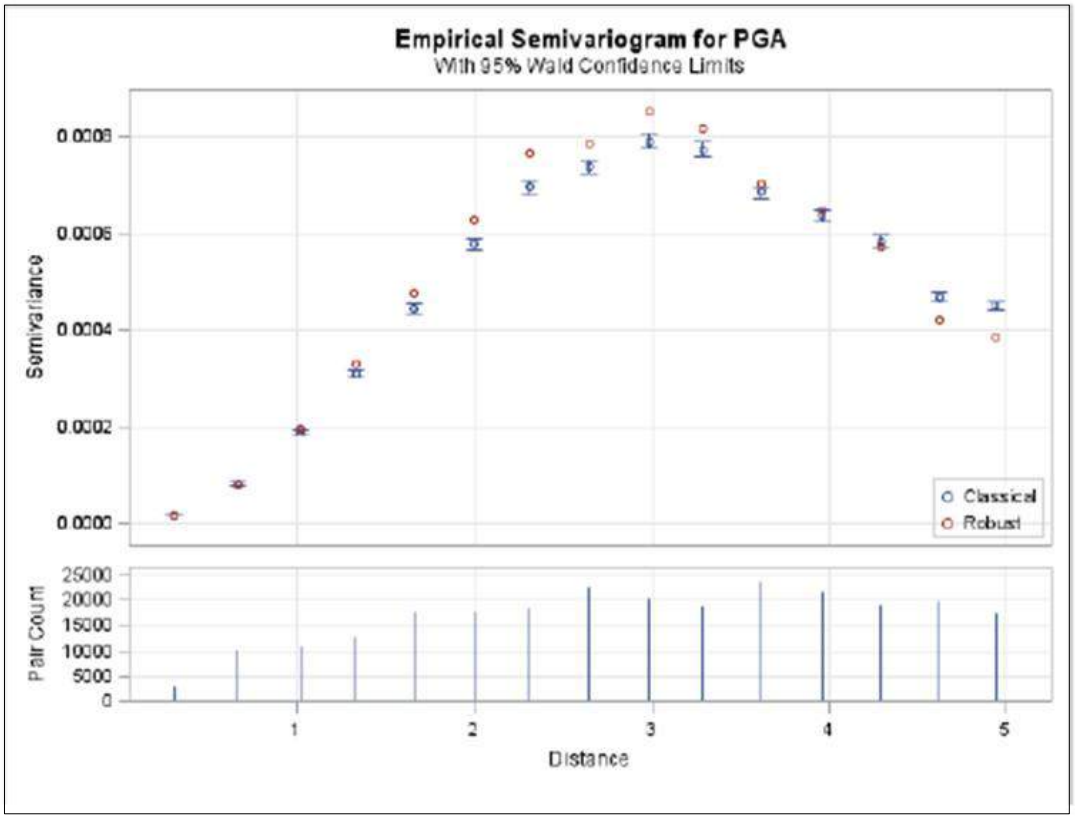
Developing the variogram based on all the data

The VARIOGRAM Procedure
Dependent Variable: PGA

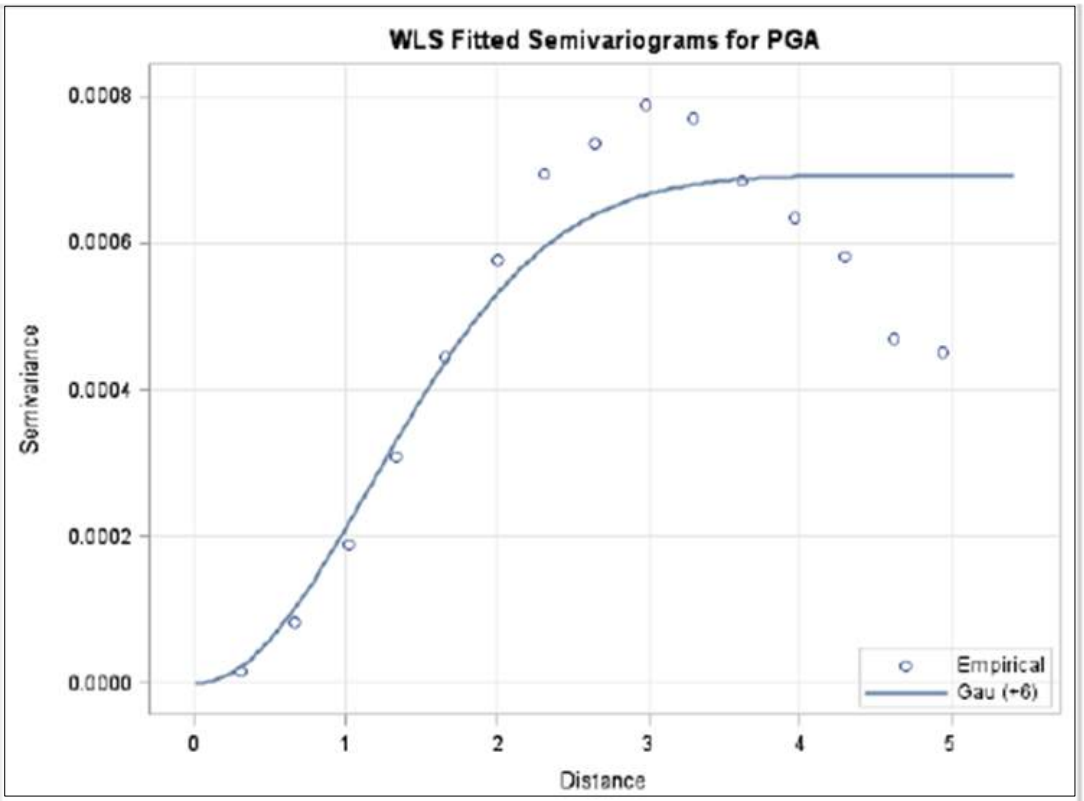
Number of Observations Read	825
Number of Observations Used	825



Empirical Semivariogram							
			Semivariance				
Lag Class	Pair Count	Average Distance	Robust	Classical	Standard Error	95% Confidence Limits	
0	0	.	.	.	.	.	.
1	3128	0.301	0.000016	0.000017	4.319E-7	0.000016	0.000018
2	10244	0.663	0.000081	0.000083	1.159E-6	0.000081	0.000085
3	10854	1.019	0.00020	0.00019	2.56E-6	0.00018	0.00019
4	12698	1.330	0.00033	0.00031	3.877E-6	0.00030	0.00032
5	17898	1.658	0.00047	0.00044	4.688E-6	0.00043	0.00045
6	17640	1.994	0.00063	0.00058	6.152E-6	0.00057	0.00059
7	18284	2.310	0.00076	0.00069	7.253E-6	0.00068	0.00071
8	22758	2.645	0.00078	0.00074	6.892E-6	0.00072	0.00075
9	20524	2.983	0.00085	0.00079	7.783E-6	0.00077	0.00080
10	18852	3.289	0.00082	0.00077	7.941E-6	0.00076	0.00079
11	23736	3.619	0.00070	0.00068	6.281E-6	0.00067	0.00070
12	21630	3.966	0.00065	0.00064	6.107E-6	0.00062	0.00065
13	19202	4.295	0.00057	0.00058	5.944E-6	0.00057	0.00059
14	19754	4.622	0.00042	0.00047	4.712E-6	0.00046	0.00048
15	17484	4.946	0.00038	0.00045	4.813E-6	0.00044	0.00046



The VARIOGRAM Procedure
Dependent Variable: PGA



Kriging based on all the data

The KRIGE2D Procedure

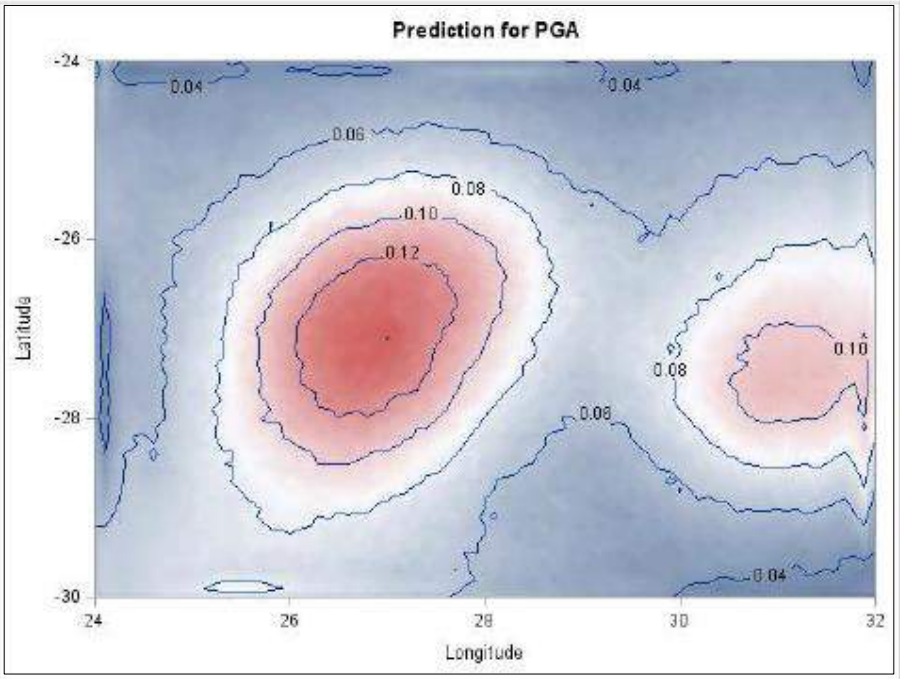
Correlation Model Item Store Information	
Input Item Store	WORK.SEMIVARSTOREGAUSSIAN_BASIN
Item Store Label	PGA_Basin Exponential WLS Fit
Data Set Created From	SASUSER.PGA_BASIN
By-group Information	No By-groups Present
Created By	PROC VARIOGRAM
Date Created	01NOV15:12:24:41

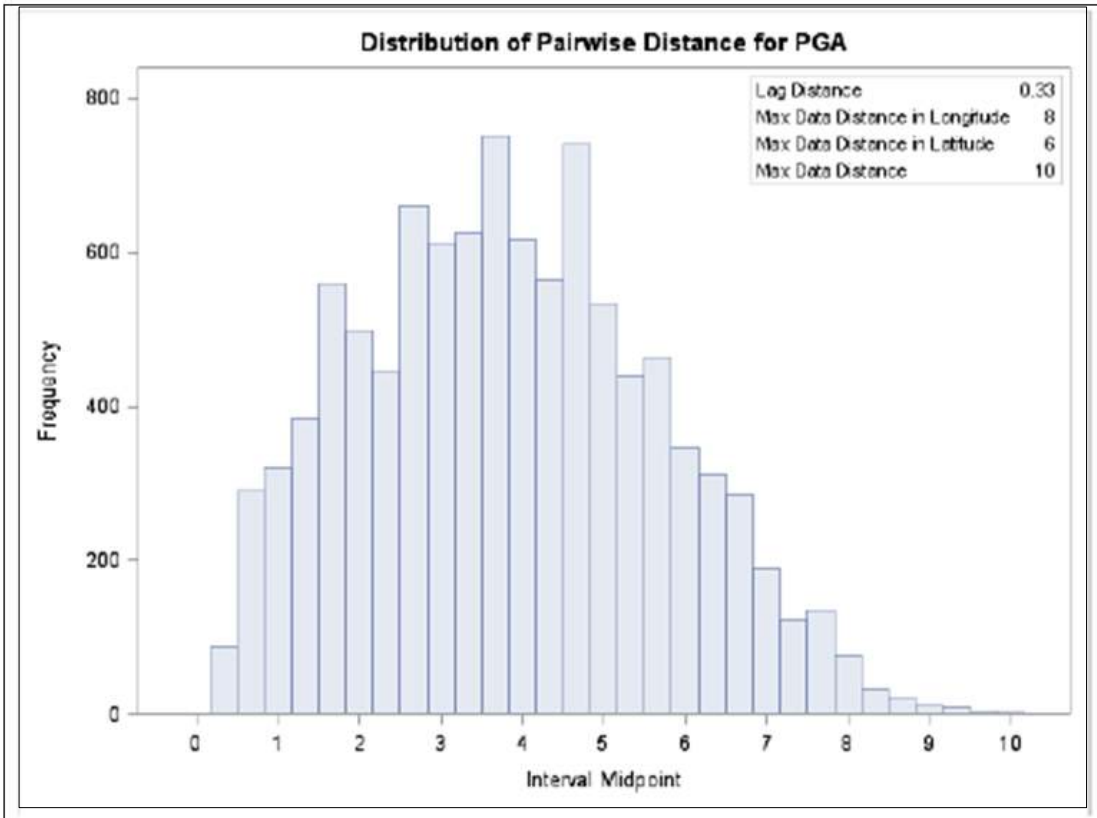
Number of Observations Read	825
Number of Observations Used	825

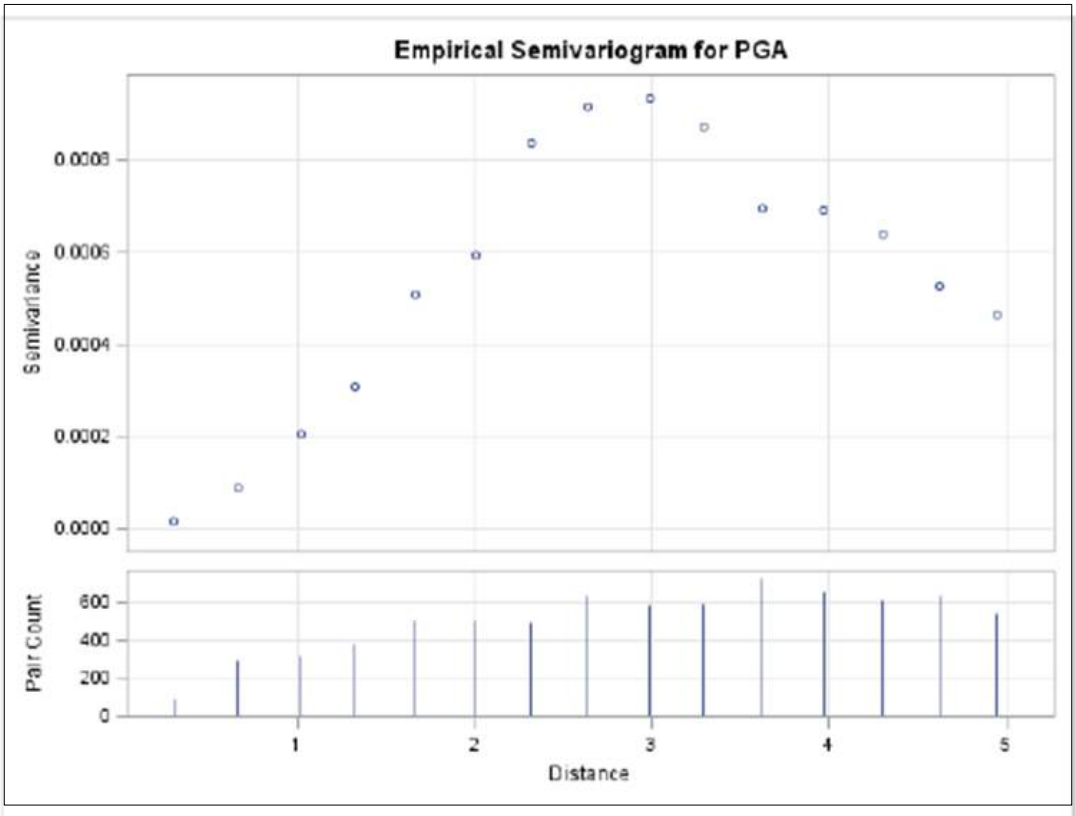
Kriging Information	
Prediction Grid Points	4941
Type of Analysis	Global

The KRIGE2D Procedure
Dependent Variable: PGA
Prediction: Pred1, Model: SelModel

Covariance Model Information	
Type	Gaussian
Sill	0.0006929
Range	1.6552782
Effective Range	2.8670259
Nugget Effect	0




Distribution of Pairwise Distance for PGA
Lag Distance 0.33
Max Data Distance in Longitude 8
Max Data Distance in Latitude 6
Max Data Distance 10
Frequency
800
600
400
200
0
0 1 2 3 4 5 6 7 8 9 10
Interval Midpoint


Empirical Semivariogram for PGA
Semivariance
0.0008
0.0006
0.0004
0.0002
0.0000
Pair Count
600
400
200
0
1 2 3 4 5
Distance

Developing Variogram based on for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA
Angle: Omnidirectional

Semivariogram Model Fitting	
Model	Selection from 12 form combinations
Output Item Store	WORK.SEMIVARSTOREGAUSSIAN_BASIN_40
Item Store Label	40% of Data - PGA Exponential WLS Fit

Fit Summary

Class	Model	Weighted SSE	AIC	Notes
1	Gau	184.74205	43.66365	
	Gau-Exp	184.74206	47.66366	Questionable fit
	Exp-Gau	184.74206	47.66366	Questionable fit
	Gau-Gau	184.74206	47.66366	
	Exp	366.08269	53.92214	
	Exp-Exp	366.09447	57.92262	

Semivariogram Model Fitting	
Name	Gaussian
Label	Gau

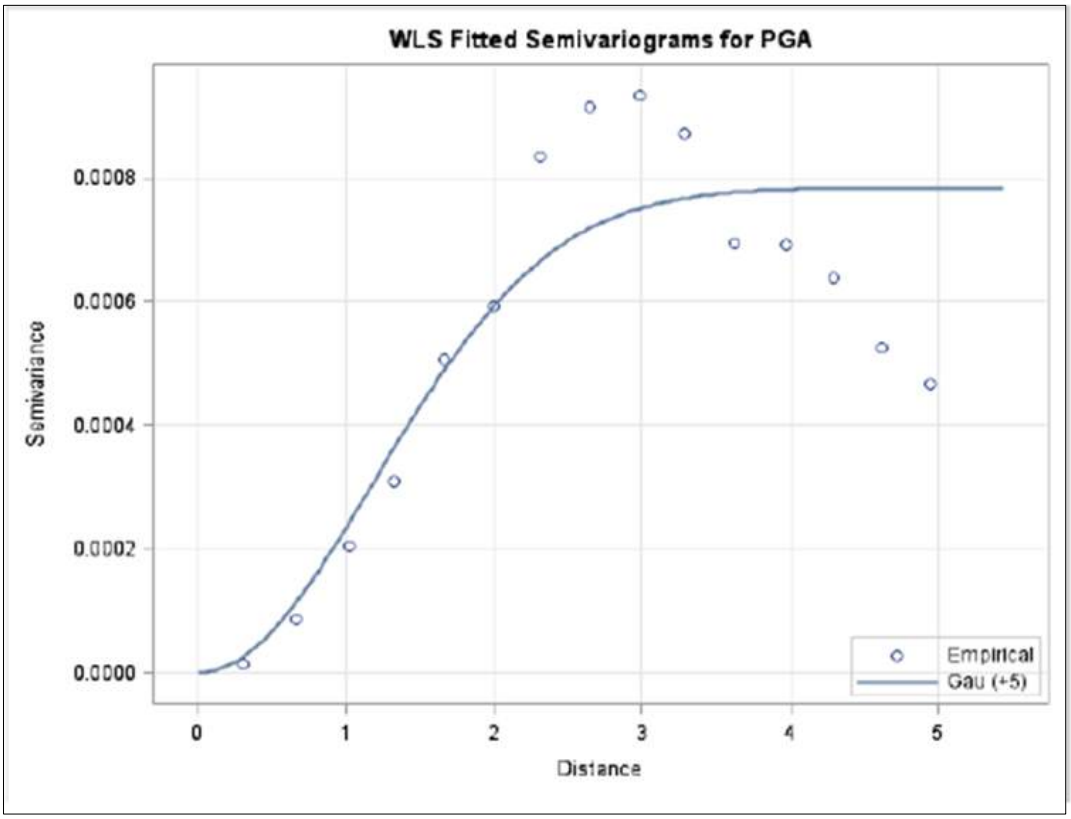
Model Information

Parameter	Initial Value
Nugget	0
Scale	0.000544
Range	2.4726

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	3
Lower Boundaries	3
Upper Boundaries	0
Starting Values From	PROC

Developing Variogram based on for 40% of the data

The VARIOGRAM Procedure
Dependent Variable: PGA



Kriging based on 40% of the data

The KRIGE2D Procedure

Correlation Model Item Store Information	
Input Item Store	WORK.SEMIVARSTOREGAUSSIAN_BASIN_40
Item Store Label	40% of Data - PGA Exponential WLS Fit
Data Set Created From	SASUSER.PGA__BASIN_40
By-group Information	No By-groups Present
Created By	PROC VARIOGRAM
Date Created	01NOV15:12:25:24

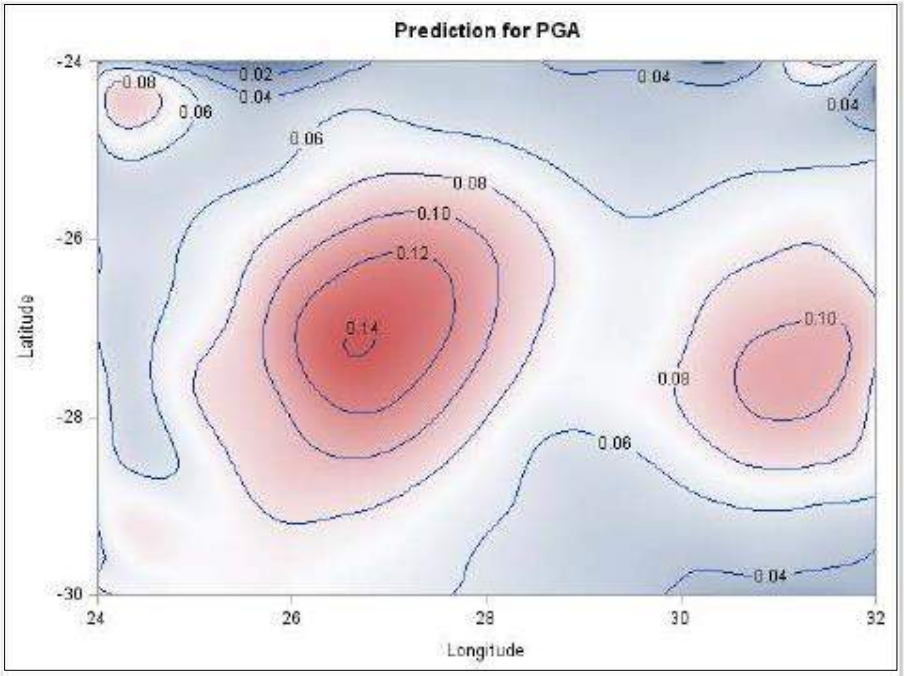
Number of Observations Read	143
Number of Observations Used	143

Kriging Information	
Prediction Grid Points	4941
Type of Analysis	Global

Kriging based on 40% of the data

The KRIGE2D Procedure
Dependent Variable: PGA
Prediction: Pred1, Model: SelModel

Covariance Model Information	
Type	Gaussian
Sill	0.0007847
Range	1.6792818
Effective Range	2.9086014
Nugget Effect	0



Mixtures of gamma distributions to model the signal-to-noise ratio of wireless channels

Brett Rowland 12032906

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisors: Mr. J. T. Ferreira, Prof. A. Bekker

Department of Statistics, University of Pretoria

2 November 2015

Abstract

In the current digital realm, modeling digital communication and wireless channels and investigating the performance thereof is of high importance. A variety of models are available to model wireless channels and some key characteristics thereof, however, some of the characteristics and performance measures associated with these models have clumsy analytical expressions and are cumbersome to compute. In this study, the mixture gamma ($M\mathcal{G}$) distribution is considered as a approximating model for the signal-to-noise (SNR) ratio of some specific composite wireless channels. A numerical simulation and performance analysis is carried out to identify the accuracy and suitability of the proposed $M\mathcal{G}$ models as an approximation of the SNR distributions of the Nakagami-lognormal (NL) and Generalised K (K_G) channels, and the advantages of the use of the $M\mathcal{G}$ distribution is highlighted.

Keywords: composite fading, gamma shadowing, Gaussian-Quadrature approximation, K_G channel, lognormal shadowing, moment matching method, multipath fading, Nakagami-lognormal channel, Nakagami-m channel, Rayleigh channel, Rician channel, shadowing

Declaration

I, *Brett William Rowland,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_ _

Brett William Rowland

_ _

Mr. J. T. Ferreira

_ _

Prof. A. Bekker

_ _

Date

Dedication

This research report is dedicated to my grandmother, and to the memory of my grandfather, to whom I have made a promise to continue in my pursuit of excellence through hard-work and determination.

Acknowledgments

A special feeling of gratitude to my loving parents, Tim and Jeanne Rowland, whose words of encouragement and support have made this work possible. Their unconditional love motivates and enables me to navigate through both personal and academic obstacles.

I would like to express my deepest appreciation to my supervisors, Mr. J. T. Ferreira and Prof. A. Bekker, both who display a spirit of academic adventure and admirable respect for the discipline. Without their guidance and assistance, this research report would not have been possible.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background

In wireless channels, a transmitted signal interacts in a complicated way as it propagates through the medium between the transmitter and the receiver [2]. Fading, an important component of wireless communication, is defined as deviation in the gradual loss in intensity of signal passing through certain propagation media. The propagation media may include air, water, foliage and concrete, and describes all mediums through which the electromagnetic wave travels [19].

Two main types of fading exist; multipath fading and shadowing. Multipath fading is used to describe the rapid fluctuations around the mean level of the radio signal over short periods of time or short distances, and is the collective effect of reflection, diffraction, scattering and absorption of a radiated electromagnetic wave as illustrated in Figure 1 [2, 5, 20]. Signal components may therefore follow multiple paths from the transmitter and arrive at the receiver at varying times. The resulting signal may vary significantly in amplitude and phase [5]. Waves arriving in-phase reinforce each other and produce a stronger signal. Conversely, waves arriving out-of-phase cause destructive interference and result in a fading signal [2]. Shadowing is caused by the general terrain, large buildings and vegetation, and refers to the slow variations in the local mean of the received signal strength [2, 5].

Modelling composite fading channels, where multipath fading and shadowing are modelled jointly, is essential for the performance analysis of wireless systems [2]. Multipath fading is usually modelled using either the Rayleigh, Rician or Nakagami-m distribution, whereas shadowing is generally modelled using a lognormal distribution which is supported by empirical measurements [21]. Consequently, a composite fading environment classically consists of multipath fading superimposed on lognormal shadowing [5]. The Nakagami-lognormal (NL) channel is an example of a composite fading channel which superimposes Nakagami-m multipath fading on lognormal shadowing [21]. In statistical terms; this constitutes a type of compound probability density function (PDF) involving the Nakagami-m and lognormal distribution used to model the multipath fading and shadowing jointly. However, lognormal-based composite fading models do not lead to closed form expressions and are therefore not suitable to further analytical derivations of performance metrics [2, 3]. To somewhat overcome these difficulties we use the result in [7] that the gamma PDF can be used as an alternative to lognormal PDF, and so the gamma PDF is used to describe large scale shadowing [21]. This leads to simpler composite models *i.e.* the K and Generalised-K (K_G) distributions, where the gamma distribution replaces the lognormal distribution in the Rayleigh-lognormal and NL channels respectively [20].

In essence, a composite channel has a multipath component and a shadowing component, each of which have specified distribution that when modelled jointly result in a compound distribution which inherits characteristics from its component distributions.

Table 1 summarises these common composite fading models in use and highlights the multipath and shadowing components of each.

Composite Channel	Multipath Component	Shadowing Component
Suzuki	Rayleigh	Lognormal
NL	Nakagami-m	Lognormal
K	Rayleigh	Gamma
K_G	Nakagami-m	Gamma

Table 1: Common models in use for composite fading in wireless channels [3]

Representations of the signal-to-noise ratio (SNR) distributions, in particular the SNR distributions of the NL and K_G channels, as a mixture gamma distribution gives an analytical and computational advantage due to the availability and ease-of-use of the gamma distribution.

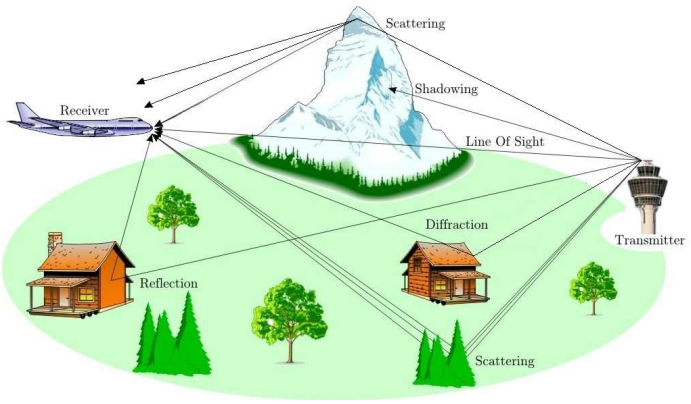


Figure 1: Wireless signals reflecting off smooth surfaces, scattering on rough surfaces, diffracting around sharp edges and transmitting through some objects [19]

1.2 Literature Review

1.2.1 Previous use of Mixture Gamma ($M\mathcal{G}$) distribution

This section briefly describes the broad use of $M\mathcal{G}$ in various fields.

Target Recognition

High-resolution radar range profiles are used for classification of a target ships. The uncertainty of orientation to the target along with the varying strength of the radar signal transmitted can be taken into account in the mixture model environment. The adoption of this model has resulted in a marked improvement in long-range statistical profiling of categories of ships [25].

Probabilistic Drought Classification

The Standard Precipitation Index (SPI), is a mathematical tool developed to monitor and classify drought events. A $M\mathcal{G}$ model allows a suitable distribution for the SPI to be determined whilst allowing for modeling uncertainties in probabilistic drought classification [13].

Site Rate Heterogeneity

The relative probability of substitution rates between nucleotide and amino acid sites is characteristic of molecular sequence evolution and helps reveal the evolutionary dynamics of a particular gene [26]. Among-site variation refers to the phenomenon that some sites within a sequence of a gene may be more likely to undergo substitutions than other sites. The $M\mathcal{G}$ model can better describe among-site variation especially where there is a multimodal distribution of rates [14].

Estimating Income Distributions

The $M\mathcal{G}$ model with two and three components can be used to estimate income distributions. Usually, a parametric approach is preferred to model the data as it allows certain inferences concerning inequality and poverty to be made. However, this approach usually suffers from a lack of flexibility. This shortcoming is mitigated by the use of the $M\mathcal{G}$ model which capitalises on both its flexibility and the advantages of parametric estimation [8].

Introvascular Ultrasound Imaging

Medical ultrasound imaging uses the propagation of a pulse as a mechanism to form images of internal organs. The pulse undergoes scattering which can cause delays and interference in the signal. The signal is thus affected by a granular pattern of noise. Distributions have been proposed to model the characterisation of the noise and the $M\mathcal{G}$ parameters and coefficients are useful in describing the granular noise. A $M\mathcal{G}$ model can provide the probability of each pixel belonging to a particular tissue class allowing the most probable edges between tissues to be detected [22, 23].

1.2.2 Multipath fading channels

Theory, simulation and measurement of wireless multipath fading channels [15]

The study aims to quantify the multipath propagation effects of both fixed and mobile wireless channels. Theory focuses on the multipath effects on signal transmission and touches on the relevant metrics used to assess and describe the fading of multipath channels. The objective was to, using MATLAB, practically demonstrate and measure the effects multipath fading. The Matlab simulations were successfully implemented and illustrate a method of how a signal fading effect can be measured.

1.2.3 Shadowing channels

Why is Shadow Fading Lognormal? [18]

In this article, an additive cluster-based model for shadowing is proposed. The article offers a more plausible explanation for the choice of the lognormal distribution as a shadowing distribution when compared to traditional approaches. It is demonstrated that under certain assumptions and conditions (i.e. the central limit theorem), that shadowing will be approximately lognormally distributed. This conclusion is supported with measurement results.

1.2.4 Composite channels

Mobile Communication Systems in the Presence of Fading/Shadowing, Noise and Interference [6]

This article investigates the effects of composite fading. Important statistical metrics for the SNR ratio are studied whilst the theory provides background used to analyse outage probability of composite fading channels. The analysis is presented with numerically evaluated results which clearly show the value of the study in the framework of composite fading networks.

Human body shadowing in cellular device-to-device channels [9]

Cellular devices are exposed to an increased risk of shadowing from the immediate surroundings and the device users themselves. In this article, the shadowed $\kappa - \mu$ model is proposed, which can characterise a shadowed multipath fading environment. The $\kappa - \mu$ model is shown to provide a good fit to empirical data, whilst highlighting interesting characteristics of the received signal.

1.3 Objective and aims

1. Identify multipath fading and shadowing channels;
2. Explore the $M\mathcal{G}$ as an alternative representation of composite fading channels;
3. Consider the outage probability of investigated models in theoretically and with the $M\mathcal{G}$ representation;
4. Implement moment matching and Gaussian-Quadrature approximation to approximate composite fading channels with a $M\mathcal{G}$ representation;
5. Investigate Kullback-Leibler divergence and Mean Square Error as a measure of accuracy of an approximation.

1.4 Outline of study

The rest of this project is organised as follows: Section 2.1 and Section 2.2 briefly describe the SNR distributions of common multipath fading channels and shadowing distributions respectively. In Section 2.3, two composite fading models are investigated. Section 3 introduces the $M\mathcal{G}$ wireless channel and provides a detailed description of how the $M\mathcal{G}$ approximates the channels given in Section 2.3. The moment matching method used to approximate the SNR distribution of the K_G channel with a $M\mathcal{G}$ is implemented Section 4. In Section 5, a $M\mathcal{G}$ distribution is fitted to the SNR PDFs of the two composite channels defined in Section 2.3 using Gaussian-Quadrature approximation. In Section 6, outage probability is defined as a performance metric, and the outage probabilities of the theoretical and approximating $M\mathcal{G}$ channels are plotted. Thereafter, outage probabilities are plotted against the fading and shadowing parameters of the composite channels. Finally some conlcusions are reached and future prospects discussed in Section 7 and Section 8 respectively.

2 Characteristics and Modelling of Channels

A fading channel is a statistical characterisation of the variation of the envelope of the received signal over time. The statistical modelling of fading channels will depend on the environment in which the signal is being propagated [21]. In this section three common multipath fading channels, each of which are useful in particular environments, and the related SNR distribution of each, will be reviewed. Shadowing channels will then be introduced and thereafter two composite fading models will be investigated.

A good measure of the quality of a signal is the signal-to-noise ratio (SNR). This is the ratio of true signal amplitude (average amplitude or peak height) to standard deviation of noise. The instantaneous SNR is denoted γ where average SNR is denoted $\overline{\gamma}$ which is given by $\overline{\gamma} = \mathbb{E}[\gamma] = \int_0^\infty x f_\gamma(x) dx$ where $f_\gamma(x)$ is the distribution of the SNR of the channel under review [21]. We note that each channel possesses an expression for SNR, denoted γ, which is a random variable with PDF $f_\gamma(x)$. We note that the severity of multipath fading or shadowing is directly proportional to the spread of the SNR distribution of the channel under investigation.

2.1 Multipath Fading Channels

In this section three well-known models of multipath fading will be reviewed.

Rayleigh Channel

The Rayleigh channel is used to model worst-case scenario multipath fading with no dominant signal component [19, 21]. This means that there no direct signal path from transmitter to receiver or no dominant reflected component. This will occur when all the multipath components have approximately the same amplitude and hence why $\overline{\gamma}$ is of interest. The Rayleigh channel is useful for modelling radio performance of a signal in built-up areas where the signal is reflected many times and may follow many non-dominant paths from transmitter to receiver [19]. The SNR PDF of the Rayleigh channel is an exponential distribution with PDF given in [21] by

$$f_\gamma(x) = \frac{1}{\overline{\gamma}} \exp\left(-\frac{x}{\overline{\gamma}}\right), \; x > 0 \tag{1}$$

where $\overline{\gamma} > 0$ is as defined above and is denoted $\gamma \sim EXP(\frac{1}{\overline{\gamma}})$.

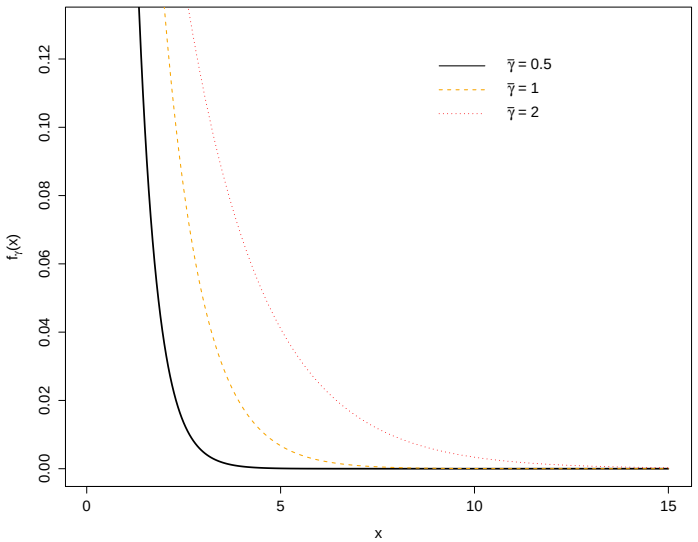


Figure 2: The SNR PDF of the Rayleigh channel (1) with varying $\overline{\gamma}$

Figure 2 shows that as $\overline{\gamma}$ increases, the PDF shifts to the right. This illustrates that as the average SNR of the channel increases, the probability of attaining a larger instantaneous SNR over the channel increases, which is an indication of an improvement in the instantaneous SNR. The Rayleigh channel does not have a fading parameter as this channel's SNR distribution depends only on the average SNR of the channel.

Rician (Nakagami-n) Channel

This model is relevant in an environment which has a dominant line-of-sight signal and may have many weaker components, as in the case of indoor propagation or satellite channels [21]. The SNR distribution of the Rician channel is known as the Nakagami-n distribution with PDF

$$f_\gamma(x) = \frac{(1+n^2)e^{-n^2}}{\overline{\gamma}} \exp\left[-\frac{(1+n^2)x}{\overline{\gamma}}\right] I_0\left(2n\sqrt{\frac{(1+n^2)x}{\overline{\gamma}}}\right), \; x > 0 \tag{2}$$

for $\overline{\gamma} > 0$ and where $I_0(.)$ is the zero order modified Bessel function of the first kind as defined in Section 10.3 and the Nakagami-n fading parameter is $n \in [0, \infty)$ which is inversely proportional to multipath fading severity [20]. n^2 is the ratio between the power in the direct path to the power in other scattered paths [16]. The Rician channel is a generalisation of the Rayleigh channel. Therefore, the Rayleigh channel can be used to describe environments where there may or may not be a dominant line-of-sight signal.

Special Cases:

1. For $n = 0$, we have $I_0(0) = 1$ (see Section 10.3) and so the the Nakagami-n SNR PDF (2) collapses to the Rayleigh SNR PDF (1); and
2. For $n = \infty$, the channel experiences no fading.

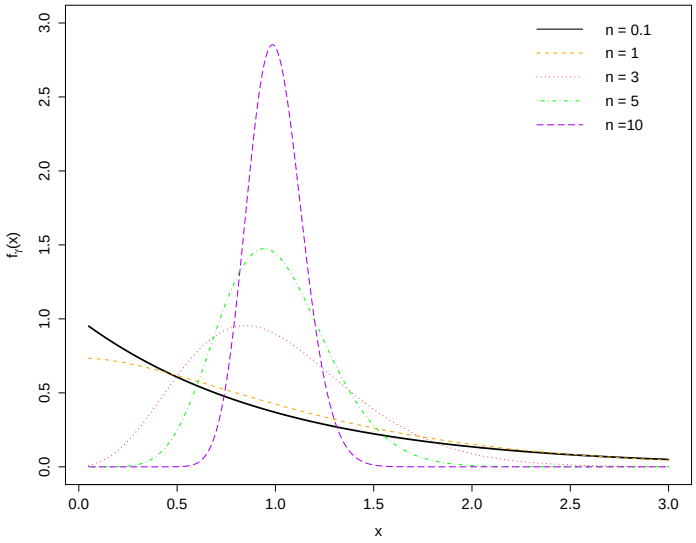


Figure 3: The SNR PDF of the Rician channel (2) with arbitrary $\overline{\gamma} = 1$ and varying n

Figure 3 shows that as n increases; the spread of the PDF (2) decreases and thus; fading decreases.

Nakagami-m Channel

This model fits indoor mobile and land mobile propagation well [19]. The SNR distribution of the Nakagami-m channel is a gamma distribution with PDF given by

$$f_\gamma(x) = \frac{m^m x^{m-1}}{\overline{\gamma}^m \Gamma(m)} \exp(-\frac{mx}{\overline{\gamma}}),\ x > 0 \tag{3}$$

for $\overline{\gamma} > 0$ and where $\Gamma(.)$ is the gamma function as defined in Section 10.1 and Nakagami-m fading parameter is $m \in [\frac{1}{2}, \infty)$ which is inversely proportional to multipath fading severity [20] and is denoted $\gamma \sim GAM(m, \frac{m}{\overline{\gamma}})$. The Nakagami-m model is also a generalisation of the Rayleigh model in the case when the dominant component is zero i.e. when $m = 1$.

The PDF of the SNR distribution of a Nakagami-m channel is sometimes used to approximate the PDF of the SNR distribution of a Rician channel. Matching the first and second moments of the Rician and Nakagami PDFs gives the relationship between m in the Nakagami-m channel and n in the Rician channel and is given in [16] as:

$$m = \frac{(1+n^2)^2}{1+2n^2}, \quad n \geq 0.$$

Special Cases:

1. For $m = 1$, Nakagami-m channel reduces to the Rayleigh channel; and
2. For $m < 0.5$, Nakagami-m channel experiences fading worse than that of the Rayleigh channel; and
3. For $m = \infty$, the channel experiences no fading.

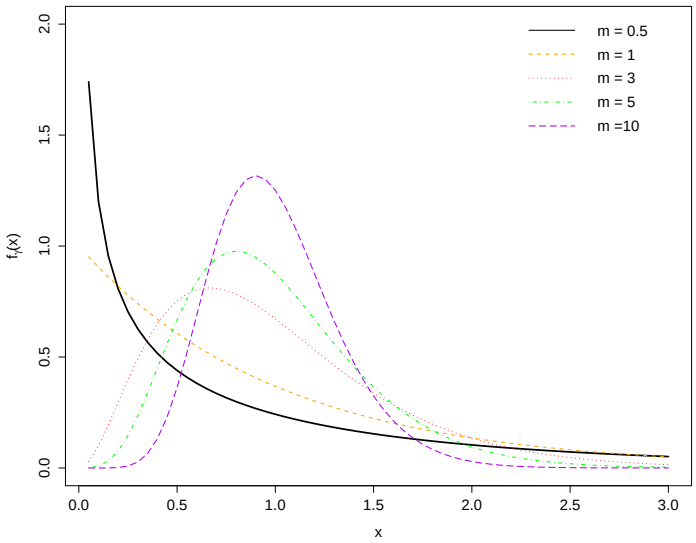


Figure 4: The SNR PDF of the Nakagami-m channel (3) with arbitrary $\overline{\gamma} = 1$ and varying m

Figure 4 shows that as m increases; the spread of the PDF (3) decreases and thus; fading decreases.

2.2 Shadowing Channels

In this section two well known models of shadowing will be reviewed.

Lognormal Shadowing

There is empirical evidence supporting the statement that the lognormal channel is suitable in modelling large scale shadowing [21]. The SNR distribution of the lognormal-shadowed channel is a lognormal distribution with PDF given by

$$g_\gamma(x) = \frac{1}{\sqrt{2\pi}\lambda x} \exp(-\frac{(\ln x - u)^2}{2\lambda^2}),\ x > 0 \tag{4}$$

where $\mu \in \mathbb{R}$ and $\lambda > 0$ are the mean and standard deviation of the lognormal distribution respectively. In this case λ is the shadowing parameter, which is directly proportional to the spread of the SNR distribution of the lognormal channel, which is directly proportional to shadowing severity.

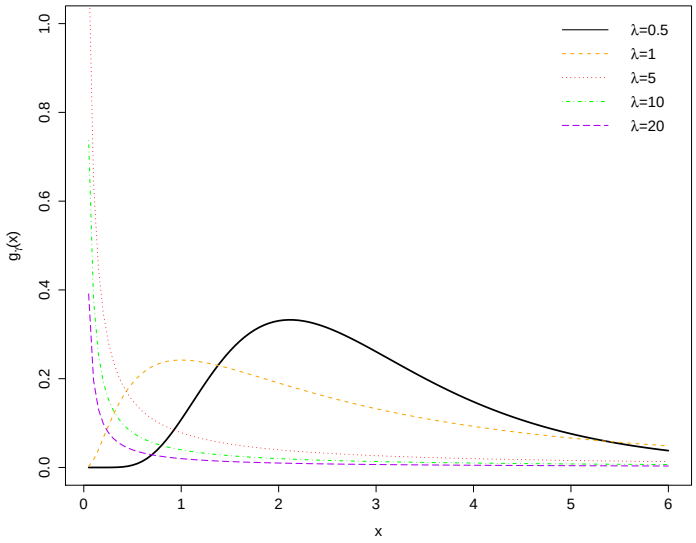


Figure 5: The SNR PDF of a lognormal-shadowed channel (4) with arbitrary $\mu = 1$ and varying λ

Figure 5 shows that as λ increases; the spread of the PDF (4) increases and thus; shadowing increases.

Gamma Shadowing

The use of the gamma distribution for the large scale shadowing leads to closed-form results in the composite fading models. The SNR distribution of the gamma-shadowed channel is gamma distributed with PDF given by (3) for $\overline{\gamma} > 0$ and the shadowing parameter $m > 0$ which is inversely proportional to the the spread of the SNR distribution of the gamma channel, which is directly proportional to shadowing fading severity and is denoted $\gamma \sim GAM(m, \frac{m}{\overline{\gamma}})$.

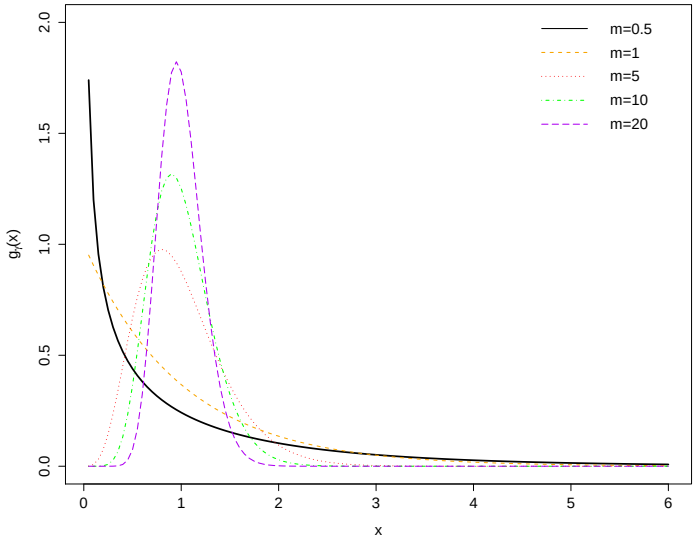


Figure 6: The SNR PDF of a gamma-shadowed channel (3) with arbitrary $\overline{\gamma} = 1$ and varying m

Figure 6 shows that as m increases; the spread of the PDF (3) decreases and thus; shadowing decreases.

2.3 Composite Fading Models

A composite fading environment consists of multipath fading superimposed on a shadowing distribution [21]. This environment necessitates the use of an instantaneous composite signal as it does not average out multipath fading over a given distance [21]. Congested urban environments and slow moving pedestrians are good examples of such an environment [21].

Nakagami-lognormal Channel

The SNR distribution of the NL channel follows a composite gamma-lognormal distribution with the following PDF:

$$h_\gamma(x) = \int_0^\infty \frac{x^{m-1}\exp[-\frac{mx}{\rho y}]}{\Gamma(m)} \left(\frac{m}{\rho y}\right)^m \frac{\exp[-\frac{(\ln y-\mu)^2}{2\lambda^2}]}{\sqrt{2\pi}\lambda y} dy,\ x>0 \tag{5}$$

where ρ is the unfaded SNR (also known as SNR at the transmitter), $\mu \in \mathbb{R}$ and $\lambda > 0$ are the mean and standard deviation of the lognormal distribution respectively and $m \in [\frac{1}{2}, \infty)$ is the Nakagami-m fading parameter [4, 21]. We can write $\gamma \sim X|Y$ where $X \sim GAM(m, \frac{\rho y}{m})$ and $Y \sim LN(\mu, \lambda)$.

The severity of multipath-fading is inversely proportional to m and the severity of shadowing is directly proportional to λ [4].

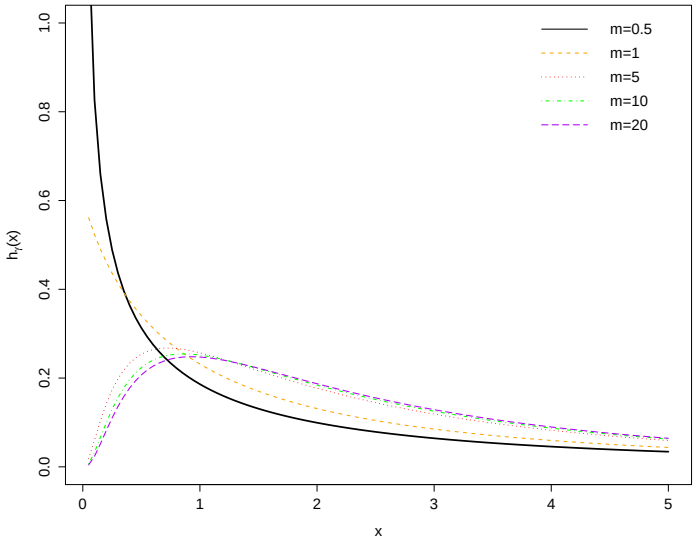


Figure 7: The SNR PDF of the NL channel (5) with arbitrary $\lambda = 1$, $\rho = 1$ and varying m

Figure 7 shows that as m increases; the spread of the PDF (5) decreases and thus; multipath fading component decreases implying that composite fading decreases.

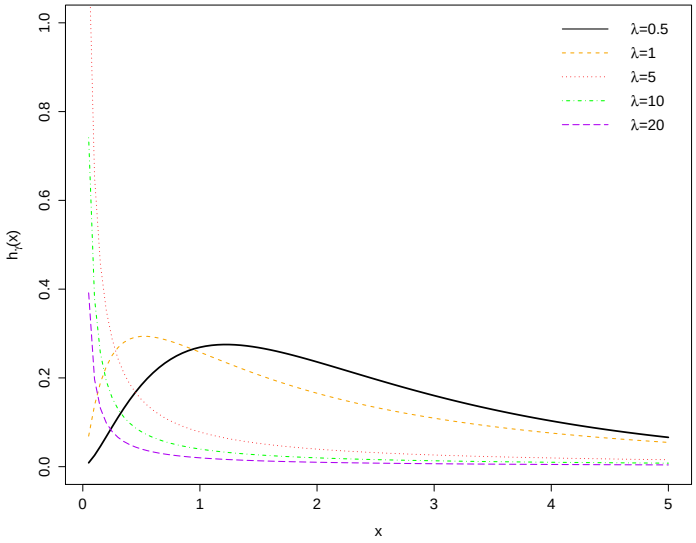


Figure 8: The SNR PDF of the NL channel (5) with arbitrary $m = 2.7$, $\rho = 1$ and varying λ

Figure 8 shows that as λ increases; the spread of the PDF (5) increases and thus; shadowing component increases implying that composite fading increases.

$\mathbf{K}_G$ Channel

The SNR distribution of the K_G channel follows a composite gamma-gamma distribution with the following PDF:

$$h_\gamma(x) = \frac{\lambda^m x^{m-1}}{\Gamma(m)\Gamma(k)} \int_0^\infty e^{-t} g(t) dt, \; x > 0 \tag{6}$$

for $\overline{\gamma} > 0$ where $g(t) = t^{\alpha-1} e^{\frac{-\lambda x}{t}}$, $\lambda = \frac{km}{\overline{\gamma}}$ and $\alpha = k - m$. In this model, $k \in [\frac{1}{2}, \infty)$ and $m > 0$ are the fading parameters, representing multipath fading and shadowing effects respectively. We can write $\gamma \sim X|Y$ where $X \sim GAM(m, \frac{m}{\overline{\gamma}})$ and $Y \sim GAM(k, \frac{k}{\overline{\gamma}})$. The severity of multipath-fading and shadowing is inversely proportional to k and m respectively [4] which is illustrated in Figure 9 and Figure 10 respectively.

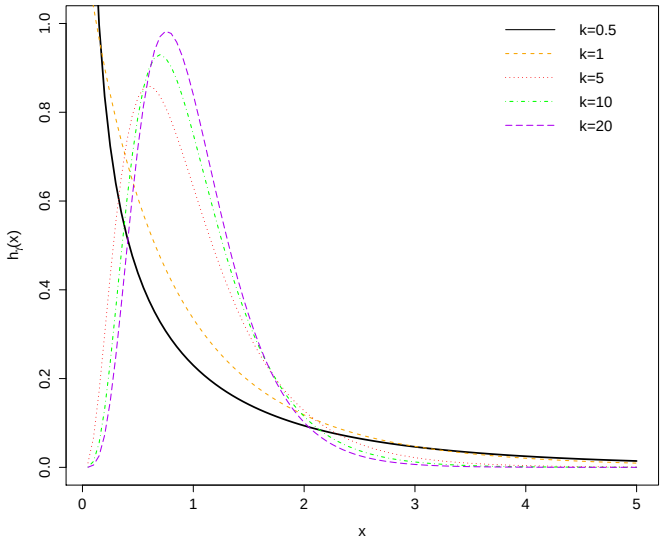


Figure 9: The SNR PDF of the K_G Channel (6) with arbitrary $\overline{\gamma} = 1$, $m = 6$ and varying k

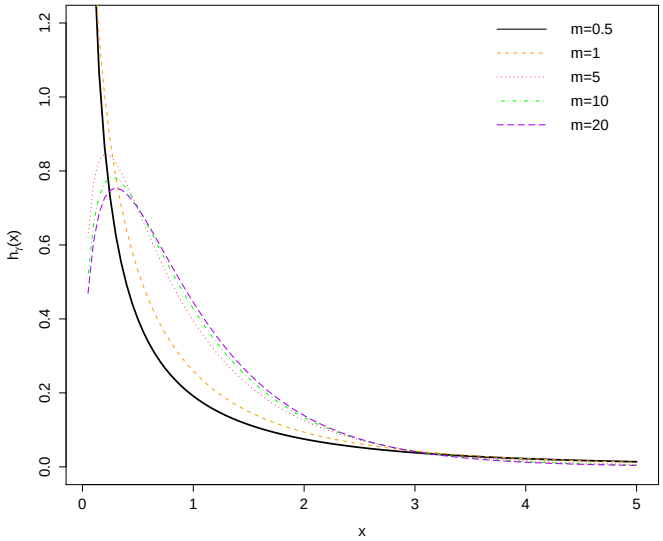


Figure 10: The SNR PDF of the K_G Channel (6) with arbitrary $\overline{\gamma} = 1$, $k = 1.5$ and varying m

3 The $M\mathcal{G}$ Wireless Channel

Note that instantaneous SNR is denoted γ as in Section 2. The following result from [10] is crucial for developing $M\mathcal{G}$ wireless channel representations, and is stated here without proof.

Result 1

Any function $f(x)$ where $x \in (0, \infty)$ and $\lim_{x\to\infty} f(x) \to 0$ can be written as $f(x) = \lim_{u\to\infty} S_u(x)$,
where $S_u(x) = e^{-ux} \sum_{k=0}^{\infty} \frac{(uk)^k}{k!} f(\frac{k}{u})$ for $u > 0$.
$S_u(x)$ is then a linear combination of gamma PDFs.

Therefore the following $M\mathcal{G}$ distribution is proposed as an improvement to model the SNR of any wireless channel:

$$f_\gamma(x) = \sum_{i=1}^{N} w_i f_i(x) = \sum_{i=1}^{N} \alpha_i x^{\beta_i - 1} \exp(-\zeta_i x), \ x > 0 \tag{7}$$

where $f_i(x) = \frac{\zeta_i^{\beta_i} x^{\beta_i - 1} e^{-\zeta_i x}}{\Gamma(\beta_i)}$ and $w_i = \frac{\alpha_i \Gamma(\beta_i)}{\zeta_i^{\beta_i}}$, N is the number of components in the $M\mathcal{G}$ distribution, and α_i, β_i, and ζ_i are the parameters of the i^{th} gamma component.

This result provides the rationale for using the $M\mathcal{G}$ distribution to represent any wireless SNR model [4]. $M\mathcal{G}$ models can be used to generalise the models in Table 1 into a form with mathematically tractable expressions of the PDF, cumulative density function (CDF), moment generating function (MGF) and moments and are therefore also lead to convenient expressions of performance metrics [4]. $M\mathcal{G}$ models are useful in approximating composite fading channels and can also be used to model most small scale fading channels whilst ensuring high accuracy by adjusting the parameters of each component of the $M\mathcal{G}$ distribution [4]. Although the K and K_G models have closed form probability density functions, they include special functions which lead to mathematical and numerical complications when deriving performance metrics [4], however, as will be investigated in Section 6, the $M\mathcal{G}$ can be used to approximate these fading channels and thereafter, facilitate rapid calculation of performance metrics.

3.1 Statistical properties of the $M\mathcal{G}$ distribution

In this section important statistical properties of the $M\mathcal{G}$ distribution are derived.

Cumulative Distribution Function

Suppose γ is a random variable that follows a $M\mathcal{G}$ distribution with PDF (7). Then the CDF of γ is given by

$$F_\gamma(x) = \int_0^x f_\gamma(s)ds = \sum_{i=1}^{N} \alpha_i \zeta_i^{-\beta_i} \gamma(\beta_i, \zeta_i x), \ x > 0 \tag{8}$$

where $\alpha_i > 0$, $\beta_i > 0$, $\zeta_i > 0$ and $\gamma(\beta_i, \zeta_i x)$ is the lower incomplete gamma function as defined in Section 10.2.

Proof. Consider, from (7)

$$\begin{aligned} F_\gamma(x) &= \int_0^x f_\gamma(s)ds \\ &= \int_0^x \sum_{i=1}^{N} \alpha_i s^{\beta_i - 1} e^{-\zeta_i s} ds \\ &= \sum_{i=1}^{N} \alpha_i \int_0^x s^{\beta_i - 1} e^{-\zeta_i s} ds \end{aligned}$$

Now let $\zeta_i s = t$ which has Jacobian $\frac{ds}{dt} = \zeta_i^{-1}$, therefore

$$\begin{aligned} F_\gamma(x) &= \sum_{i=1}^{N} \alpha_i \int_0^{x\zeta_i} \left(\frac{t}{\zeta_i}\right)^{\beta_i - 1} e^{-t} \zeta_i^{-1} dt \\ &= \sum_{i=1}^{N} \alpha_i \int_0^{x\zeta_i} t^{\beta_i - 1} e^{-t} \zeta_i^{-(\beta_i - 1) - 1} dt \\ &= \sum_{i=1}^{N} \alpha_i \int_0^{x\zeta_i} t^{\beta_i - 1} e^{-t} \zeta_i^{-\beta_i} dt \\ &= \sum_{i=1}^{N} \alpha_i \zeta_i^{-\beta_i} \int_0^{x\zeta_i} t^{\beta_i - 1} e^{-t} dt \\ &= \sum_{i=1}^{N} \alpha_i \zeta_i^{-\beta_i} \gamma(\beta_i, \zeta_i x) \end{aligned}$$

by using Result 10.2 in Appendix, where $\alpha_i > 0$, $\beta_i > 0$, $\zeta_i > 0$ and $\gamma(\beta_i, \zeta_i x)$ is the lower incomplete gamma function. ■

Moment Generating Function

Suppose γ is a random variable that follows a $M\mathcal{G}$ distribution with PDF (7). Then the MGF of γ is given by

$$M_\gamma(s) = \sum_{i=1}^{N} \frac{\alpha_i \Gamma(\beta_i)}{(s + \zeta_i)^{\beta_i}} \tag{9}$$

where $\beta_i > 0$, $\zeta_i > 0$ and s an integer.

Proof. Consider, from (7)

$$\begin{aligned} M_\gamma(s) &= \mathbb{E}[e^{sx}] \\ &= \int_0^{\infty} e^{sx} f_\gamma(x) dx \\ &= \int_0^{\infty} e^{sx} \sum_{i=1}^{N} \alpha_i x^{\beta_i - 1} e^{-\zeta_i x} dx \\ &= \sum_{i=1}^{N} \alpha_i \int_0^{\infty} e^{sx} x^{\beta_i - 1} e^{-\zeta_i x} dx \\ &= \sum_{i=1}^{N} \alpha_i \int_0^{\infty} x^{\beta_i - 1} e^{-x(\zeta_i + s)} dx \\ &= \sum_{i=1}^{N} \alpha_i \frac{\Gamma(\beta_i)}{(\zeta_i + s)^{\beta_i}} \end{aligned}$$

where $\beta_i > 0$, $\zeta_i > 0$ and s an integer, by using Result 10.1 in Appendix.

■

Moments

Suppose γ is a random variable that follows a $M\mathcal{G}$ distribution with PDF (7). Then the moments of γ are

$$m_\gamma(r) = \sum_{i=1}^{N} \alpha_i \Gamma(\beta_i + r) \zeta_i^{-(\beta_i + r)} \tag{10}$$

where $\beta_i > 0$, $\zeta_i > 0$ and r an integer.

Proof. Consider, from (7)

$$\begin{aligned} m_\gamma(r) &= \mathbb{E}[x^r] \\ &= \int_0^\infty x^r f_\gamma(x) dx \\ &= \int_0^\infty x^r \sum_{i=1}^{N} \alpha_i x^{\beta_i - 1} e^{-\zeta_i x} dx \\ &= \sum_{i=1}^{N} \alpha_i \int_0^\infty x^{r+\beta_i - 1} e^{-\zeta_i x} dx \\ &= \sum_{i=1}^{N} \alpha_i \Gamma(\beta_i + r) \zeta_i^{-(\beta_i + r)} \end{aligned}$$

where $\beta_i > 0$, $\zeta_i > 0$ and r an integer, by using Result 10.1 in Appendix.

■

3.2 $M\mathcal{G}$ used to approximate the SNR distributions of composite channels

Gaussian-Quadrature sums will be used to approximate the integrals in the PDF of the SNR distribution of the NL (5) and K_G (6) channel and thereafter manipulated into the form of a $M\mathcal{G}$ channel. Specifically Gaussian-Hermite and Gaussian-Laguerre quadrature sums will be used to approximate the NL and K_G channels respectively. Once the NL and K_G channels are in the form of a $M\mathcal{G}$ channel, the original PDF and the accuracy of the approximation can be investigated.

$M\mathcal{G}$ used to approximate the SNR distribution of the Nakagami-lognormal channel (5)

Using a substitution, (5) can be written as:

$$h_\gamma(x) = \frac{x^{m-1}}{\sqrt{\pi}\Gamma(m)} \left(\frac{m}{p}\right)^m \int_{-\infty}^{\infty} e^{-t^2} g(t) dt, \ x > 0 \tag{11}$$

where $g(t) = e^{-m(\sqrt{2}\lambda t + \mu)} e^{-\frac{m}{p} e^{-(\sqrt{2}\lambda t + \mu)} x}$ [4].
In this section it is shown how (11) can be written in the form of (7) using Gaussian-Quadrature approximation. Equation (11) can be approximated by using a Gaussian-Hermite quadrature sum where w_i and t_i are the respective

Gaussian-Hermite weights and nodes associated with the i^{th} gamma component which can be found in the Appendix.

$$\begin{aligned}
h_\gamma(x) &\approx \frac{x^{m-1}}{\sqrt{\pi}\Gamma(m)}\left(\frac{m}{p}\right)^m \sum_{i=1}^{n} w_i g(t_i) \\
&= \sum_{i=1}^{n} \frac{x^{m-1}}{\sqrt{\pi}\Gamma(m)}\left(\frac{m}{p}\right)^m w_i g(t_i) \\
&= \sum_{i=1}^{n} \frac{x^{m-1}}{\sqrt{\pi}\Gamma(m)}\left(\frac{m}{p}\right)^m w_i e^{-m(\sqrt{2}\lambda t_i+\mu)} e^{-\frac{m}{p}e^{-(\sqrt{2}\lambda t_i+\mu)x}} \\
&= \sum_{i=1}^{n} \frac{x^{m-1}}{\sqrt{\pi}\Gamma(m)}\left(\frac{m}{p}\right)^m w_i e^{-m(\sqrt{2}\lambda t_i+\mu)} e^{-\xi_i x} \text{ where } \xi_i = \frac{m}{p}e^{-(\sqrt{2}\lambda t_i+\mu)} \\
&= \sum_{i=1}^{n} \left(\frac{m}{p}\right)^m \frac{w_i e^{-m(\sqrt{2}\lambda t_i+\mu)}}{\sqrt{\pi}\Gamma(m)} x^{m-1} e^{-\xi_i x} \\
&= \sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i x} \text{ , where } \theta_i = \left(\frac{m}{p}\right)^m \frac{w_i e^{-m(\sqrt{2}\lambda t_i+\mu)}}{\sqrt{\pi}\Gamma(m)} \\
&= \sum_{i=1}^{n} \theta_i x^{\beta_i-1} e^{-\xi_i x} \text{ , where } \beta_i = m \quad .
\end{aligned}$$

To ensure f_γ is a valid PDF, we find the normalising constant k:

$$h_\gamma(x) = k\sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i x};$$

and thus

$$\begin{aligned}
1 &= \int_0^\infty f_\gamma(x)dx \\
&= k\int_0^\infty \sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i x} dx \\
&= k\sum_{i=1}^{n} \theta_i \int_0^\infty x^{m-1} e^{-\xi_i x} dx \\
&= k\sum_{i=1}^{n} \theta_i \Gamma(m) \xi_i^{-m}.
\end{aligned}$$

Therefore

$$k = \frac{1}{\sum_{i=1}^{n} \theta_i \Gamma(m) \xi_i^{-m}}$$

which results in

$$h_\gamma(x) = \sum_{i=1}^{N} \alpha_i x^{\beta_i-1} e^{-\xi_i x} \tag{12}$$

where $\alpha_i = \frac{\theta_i}{\sum_{j=1}^{N} \theta_i \Gamma(m) \xi_i^{-m}}$, $\theta_i = \left(\frac{m}{p}\right)^m \frac{w_i e^{-m(\sqrt{2}\lambda t_i+\mu)}}{\sqrt{\pi}\Gamma(m)}$, $\xi_i = -\frac{m}{p}e^{-(\sqrt{2}\lambda t_i+\mu)}$ and $\beta_i = m$.

$M\mathcal{G}$ used to approximate the SNR distribution of the $\mathbf{K}_G$ channel

In this section it is shown how Equation (6) can be written in the form of (7) using Gaussian-Quadrature approximation.

Equation (6) can be approximated by using a Gaussian-Laguerre quadrature sum where w_i and t_i are respective the Gaussian-Laguerre weights and nodes associated with the i^{th} gamma component which can be found in the Appendix.

$$\begin{aligned}
h_\gamma(x) &\approx \frac{\lambda^m x^{m-1}}{\Gamma(m)\Gamma(k)} \sum_{i=1}^{n} w_i g\left(t_i\right) \\
&= \sum_{i=1}^{n} \frac{\lambda^m x^{m-1}}{\Gamma(m)\Gamma(k)} w_i g\left(t_i\right) \\
&= \sum_{i=1}^{n} \frac{\lambda^m x^{m-1}}{\Gamma(m)\Gamma(k)} w_i t_i^{\alpha-1} e^{\frac{-\lambda x}{t_i}} \\
&= \sum_{i=1}^{n} \frac{\lambda^m w_i t_i^{\alpha-1}}{\Gamma(m)\Gamma(k)} x^{m-1} e^{-\xi_i x} \text{ where } \xi_i = \frac{\lambda}{t_i} \\
&= \sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i} \text{ where } \theta_i = \frac{\lambda^m w_i t_i^{\alpha-1}}{\Gamma(m)\Gamma(k)} \\
&= \sum_{i=1}^{n} \theta_i x^{\beta_i-1} e^{-\xi_i} \text{ where } \beta_i = m \quad .
\end{aligned}$$

To ensure f_γ is a valid PDF, we find the normalising constant k which is derived as before:

$$h_\gamma(x) = k \sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i x};$$

and thus

$$\begin{aligned}
1 &= \int_0^\infty f_\gamma(x) dx \\
&= k \int_0^\infty \sum_{i=1}^{n} \theta_i x^{m-1} e^{-\xi_i x} dx \\
&= k \sum_{i=1}^{n} \theta_i \int_0^\infty x^{m-1} e^{-\xi_i x} dx \\
&= k \sum_{i=1}^{n} \theta_i \Gamma(m) \xi_i^{-m}
\end{aligned}$$

and thus

$$k = \frac{1}{\sum_{i=1}^{n} \theta_i \Gamma(m) \xi_i^{-m}}$$

which results in

$$h_\gamma(x) = \sum_{i=1}^{N} \alpha_i x^{\beta_i-1} e^{-\xi_i x} \tag{13}$$

where $\alpha_i = \frac{\theta_i}{\sum_{j=1}^{N} \theta_i \Gamma(m) \xi_i^{-m}}$, $\theta_i = \frac{\lambda^m w_i t_i^{\alpha-1}}{\Gamma(m)\Gamma(k)}$, $\xi_i = \frac{\lambda}{t_i}$ and $\beta_i = m$.

3.3 Determining N in the $M\mathcal{G}$ distribution

N, the number of the components in the $M\mathcal{G}$ distribution, can be selected as the minimum value such that:

1. The first r moments of the two distributions match to the nearest integer value. This method is referred to as moment-matching and is discussed and implemented in Section 4 [4]; or
2. Mean Square Error $(MSE) = \mathbb{E}\left[(f_{Ext}(x) - f_{App}(x))^2\right]$ is minimum where f_{Ext} is the theoretical PDF and f_{App} is the approximating PDF; or
3. Kullback-Leibler Divergence $(\mathcal{D}_{KL}) = \int_{-\infty}^{\infty} f_{Ext}(x) \log \frac{f_{Ext}(x)}{f_{App}(x)} dx$ is a minimum (See Section 10.5).

Note: In order to calculate the MSE and $\mathcal{D}_{KL}$ we use the PDF values calculated in small increments to evaluate the expressions for MSE and $\mathcal{D}_{KL}$.

4 Moment Matching Method

The moment matching method described in Section 3.3 is implemented in this section. In this project we only consider the K_G channel.

The parameters in the $M\mathcal{G}$ distribution in (7) can be determined by matching the first r moments of the the standard gamma distribution and SNR distribution of the K_G channel of to the nearest integer value. The SNR distribution of the NL and the K_G channels will be approximated by a mixture gamma where $N = 1$ to demonstrate the method.

4.1 K_G Channel

The n^{th} raw moment of the SNR distribution of the K_G channel (6) is given in [2] as

$$m_\gamma^1(n) = \frac{\Gamma(k+n)\Gamma(m+n)}{\Gamma(k)\Gamma(m)} \left(\frac{\Omega_0}{km}\right)^n \tag{14}$$

where Ω_0 denotes the average local mean power [2].

If $X \sim GAM(k, \theta)$ with PDF as in Appendix (10.4) then the n^{th} raw moment of X is given by

$$m_\gamma^2(n) = \frac{\Gamma(k+n)\theta^n}{\Gamma(k)}. \tag{15}$$

A gamma distribution can be generally be used effectively when matching the lower order moments [24]. The SNR distribution of the K_G channel will be dominated by one of the two gamma distributions that make up (6), for large values of m or k [11].

It is therefore appropriate to use a single gamma distribution to approximate the K_G channel in this case. Matching different pairs of moments generated by equation (14) and (15) yield different shape and scale parameters, denoted k and θ respectively, of the approximating gamma distribution.

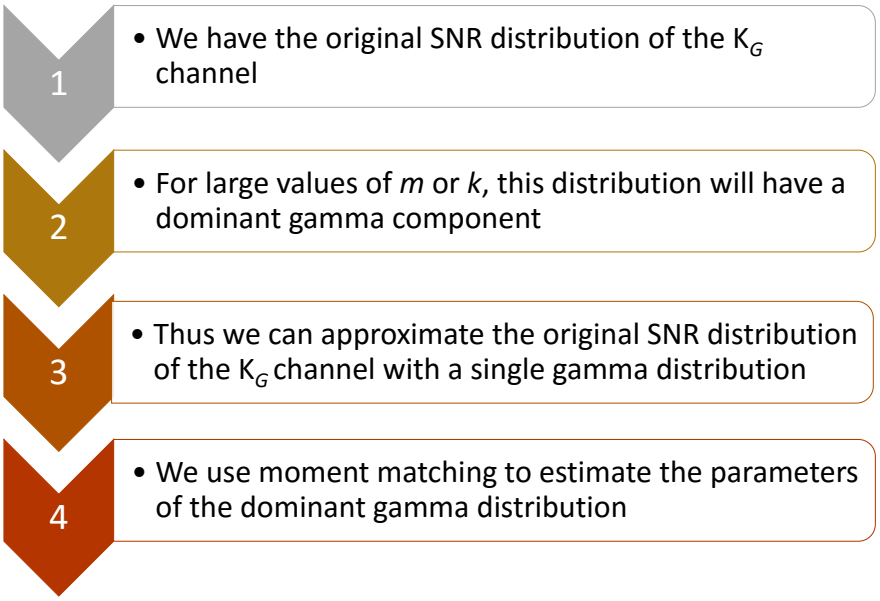


Figure 11: The rationale behind the moment matching method

We match the n^{th} moments of (6) and (28) by using expressions (14) and (15).

For the 1^{st} positive moments we equate:

$$\begin{aligned} m_\gamma^1(1) &= m_\gamma^2(1) \\ \frac{\Gamma(k+1)\Gamma(m+1)}{\Gamma(m)\Gamma(k)}\left(\frac{\Omega_0}{mk}\right) &= \frac{\Gamma(k+1)\theta}{\Gamma(k)} \\ k\theta &= \Omega_0. \end{aligned}$$

For the 2^{nd} positive moments we equate:

$$\begin{aligned} m_\gamma^1(2) &= m_\gamma^2(2) \\ \frac{\Gamma(k+2)\Gamma(m+2)}{\Gamma(m)\Gamma(k)}\left(\frac{\Omega_0}{mk}\right)^2 &= \frac{\Gamma(k+2)\theta^2}{\Gamma(k)} \\ \frac{(k+1)(m+1)}{mk}\Omega_0^2 &= k(k+1)\theta^2. \end{aligned}$$

For the 3^{rd} positive moments we equate:

$$\begin{aligned} m_\gamma^1(3) &= m_\gamma^2(3) \\ \frac{\Gamma(k+3)\Gamma(m+3)}{\Gamma(m)\Gamma(k)}\left(\frac{\Omega_0}{mk}\right)^3 &= \frac{\Gamma(k+2)\theta^3}{\Gamma(k)} \\ \frac{(k+2)(m+2)}{mk}\frac{(k+1)(m+1)}{mk}\Omega_0^3 &= k(k+1)(k+2)\theta^3. \end{aligned}$$

For the 1^{st} negative moments we equate:

$$\begin{aligned} m_\gamma^1(-1) &= m_\gamma^2(-1) \\ \frac{\Gamma(k-1)\Gamma(m-1)}{\Gamma(m)\Gamma(k)}\left(\frac{\Omega_0}{mk}\right)^{-1} &= \frac{\Gamma(k-1)\theta^{-1}}{\Gamma(k)} \\ \frac{mk}{(m-1)(k-1)}\left(\frac{mk}{\Omega_0}\right) &= \frac{1}{(k-1)\theta} \\ (k-1)\theta &= \frac{(m-1)(k-1)}{mk}\Omega_0. \end{aligned}$$

For the 2^{nd} negative moments we equate:

$$\begin{aligned} m_\gamma^1(-2) &= m_\gamma^2(-2) \\ \frac{\Gamma(k-1)\Gamma(m-1)}{\Gamma(m)\Gamma(k)}\left(\frac{\Omega_0}{mk}\right)^{-2} &= \frac{\Gamma(k-1)\theta^{-2}}{\Gamma(k)} \\ \frac{mk}{(m-2)(k-2)}\frac{mk}{(m-1)(k-1)}\left(\frac{mk}{\Omega_0}\right)^{-2} &= \frac{1}{(k-2)(k-1)\theta^2} \\ (k-2)(k-1)\theta^2 &= \frac{(m-2)(k-2)}{mk}\frac{(m-1)(k-1)}{mk}\Omega_0^2. \end{aligned}$$

We then have for the positive moments:

$$k\theta = \Omega_0 \tag{16}$$

$$k(k+1)\theta^2 = K_1\Omega_0^2 \tag{17}$$

$$k(k+1)(k+2)\theta^3 = K_2K_1\Omega_0^3 \tag{18}$$

and for the negative moments:

$$\begin{aligned} (k-1)\theta &= K_{-1}\Omega_0 \quad (19)\\ k(k-2)(k-1)\theta^2 &= K_{-1}K_{-2}\Omega_0^2 \quad (20)\end{aligned}$$

where $K_s = \frac{(m-s)(k-s)}{mk}$ for $s = 1, 2, 3...$ and Ω_0 denotes the average local mean power.

The notation $k_{i,j}$ and $\theta_{i,j}$ refers to the shape and scale parameters of the approximating gamma distribution that are obtained by matching the i^{th} and j^{th} moments generated by equation (14) and (15) respectively.

Matching the 1^{st} and 2^{nd} positive moments of (6) and (28):
We use (16) and (17) and solve

$$\begin{aligned} k_{1,2}\theta_{1,2} &= \Omega_0 \quad (21)\\ k(k+1)(k+2)\theta^3 &= K_2K_1\Omega_0^3 \quad (22)\end{aligned}$$

simultaneously for $k_{1,2}$ and $\theta_{1,2}$ and we obtain

$$\begin{aligned} k_{1,2} &= \frac{1}{1-K_1}\\ \theta_{1,2} &= \Omega_0(K_1-1).\end{aligned}$$

Matching the 1^{st} and 3^{rd} positive moments of (6) and (28):
We use (16) and (18) and we solve

$$\begin{aligned} k_{1,3}\theta_{1,3} &= \Omega_0 \quad (23)\\ k_{1,3}(k_{1,3}+1)(k_{1,3}+2)\theta_{1,3}^3 &= K_2K_1\Omega_0^3 \quad (24)\end{aligned}$$

simultaneously for $k_{1,3}$ and $\theta_{1,3}$ and we obtain

$$\begin{aligned} k_{1,3} &= \frac{4}{(-3+\sqrt{9+8(K_1K_2-1)})}\\ \theta_{1,3} &= \frac{(-3+\sqrt{9+8(K_1K_2-1)}\Omega_0}{4}\end{aligned}$$

Matching the 1^{st} positive and 2^{nd} negative moments of (6) and (28):
We use (16) and (20) and we solve

$$\begin{aligned} k_{1,-2}\theta_{1,-2} &= \Omega_0 \quad (25)\\ k_{1,-2}(k_{1,-2}-2)(k_{1,-2}-1)\theta_{1,-2}^2 &= K_{-1}K_{-2}\Omega_0^2 \quad (26)\end{aligned}$$

simultaneously for $k_{1,-2}$ and $\theta_{1,-2}$ and we obtain

$$\begin{aligned} k_{1,-2} &= \frac{4}{(3-\sqrt{9+8(K_{-1}K_{-2}-1)})}\\ \theta_{1,-2} &= \frac{(3-\sqrt{9+8(K_{-1}K_{-2}-1)}\Omega_0}{4}\end{aligned}$$

Thus, by matching these moments, expressions for parameters of a gamma distribution can be obtained which accurately approximates the SNR PDF of the K_G channel. To determine which specific combination of moments to match, this project suggests that one first considers all possible combinations of moments and chooses the combination resulting in the lowest MSE or $\mathcal{D}_{KL}$ as defined in Section (3.3).

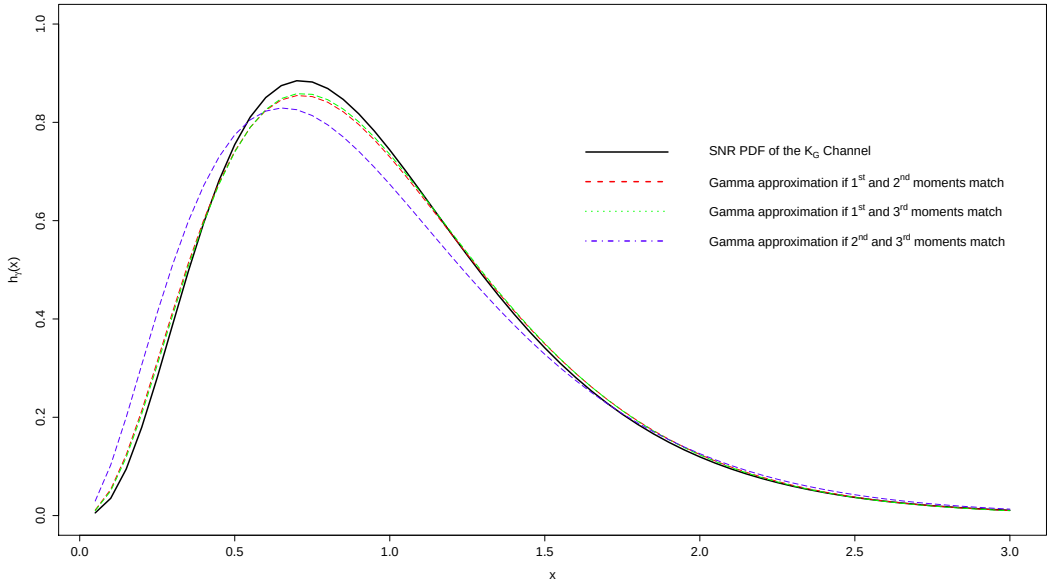


Figure 12: SNR PDF of the K_G channel (6) and approximating gamma distribution using the moment matching method with positive moments

In Figure 12 the PDF of the SNR distribution of the K_G channel with arbitrary parameters $m = 40$, $k = 4$ and $\Omega = 1$ is plotted. Matching the 1^{st} and 2^{nd} positive moments of (6) and (28), results in $k_{1,2} = 3.5556$ and $\theta_{1,2} = 0.28125$. Matching the 1^{st} and 3^{rd} positive moments of (6) and (28) respectively, results in $k_{1,3} = 3.50723$ and $\theta_{1,3} = 0.28513$. Matching the 2^{nd} and 3^{rd} positive moments of (6) and (28), results in $k_{2,3} = 3.018883$ and $\theta_{2,3} = 0.3249684$.

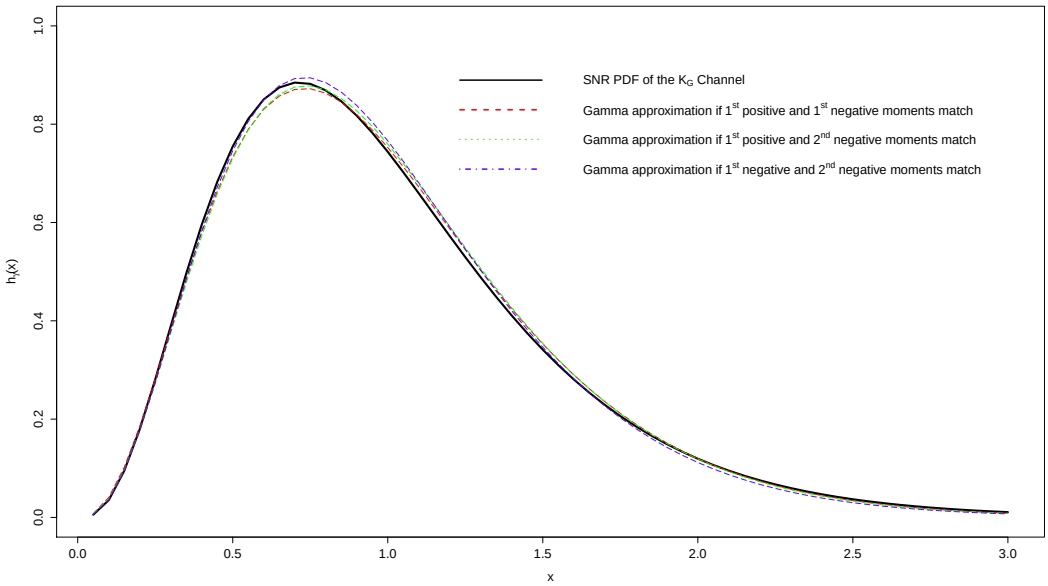


Figure 13: SNR PDF of the K_G channel (6) and approximating gamma distribution using the moment matching method with positive and negative moments

In Figure 13 the K_G channel with arbitrary parameters $m = 40$, $k = 4$ and $\Omega = 1$ is plotted. Matching the 1^{st} positive and 1^{st} negative moment of (6) and (28), results in $k_{1,-1} = 3.787523$ and $\theta_{1,-1} = 0.26875$. Matching the 1^{st} positive and 2^{nd} negative moment of (6) and (28), results in $k_{1,-2} = 3.787523$ and $\theta_{1,-2} = 0.2640248$. Matching the 1^{st} negative and 2^{nd} negative moment of (6) and (28), results in $k_{-1,-2} = 3.853659$ and $\theta_{-1,-2} = 0.25625$.

5 Fitting a Mixture Gamma ($M\mathcal{G}$) distribution to Composite Fading Channels

In this section, the $M\mathcal{G}$ distribution is used to approximate the SNR distribution of the NL channel and K_G channel using Gaussian-Quadrature approximation. To determined the accuracy of this approximation, the MSE and $\mathcal{D}_{KL}$ is calculated for the varying number of components of the $M\mathcal{G}$ distribution (N).

5.1 Nakagami-lognormal Channel

A simulation was performed in R with arbitrary $m = 2.7$, $\mu = 2$, $\lambda = 1$ and $\rho = 1$ to approximate (5) by (12) using Gaussian-Hermite weights, w_i, and nodes, t_i, as tabulated in Appendix.

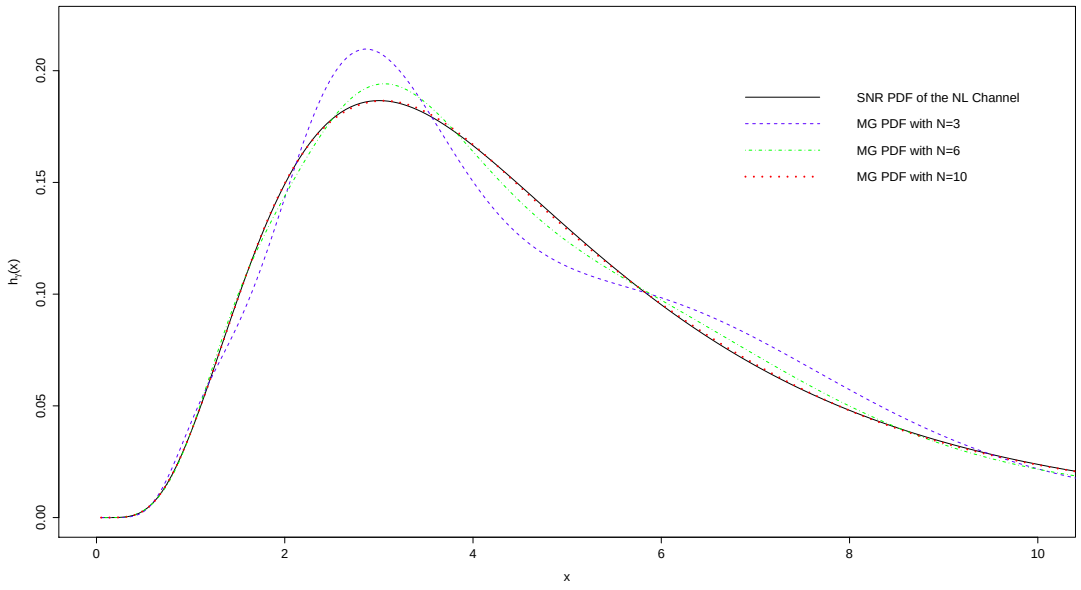
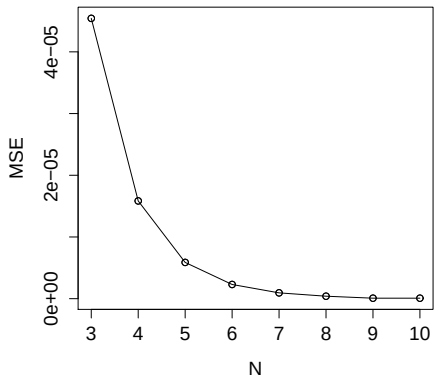


Figure 14: SNR PDF for the NL channel (5) and approximating $M\mathcal{G}$ distributions with varying components (N)

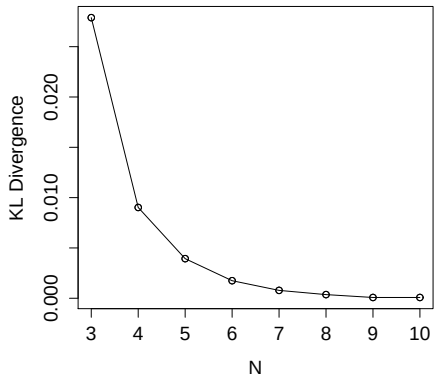
Mean Square Error (MSE)



N	MSE
3	4.542910e-05
4	1.584063e-05
5	5.875951e-06
6	2.301013e-06
7	9.421266e-07
8	4.000819e-07
9	7.899671e-08
10	7.899671e-08

MSE calculated between the SNR PDF of the NL channel (5) and the approximating $M\mathcal{G}$ distributions with varying components (N).

Kullback-Leibler divergence ($\mathcal{D}_{KL}$)



N	$\mathcal{D}_{KL}$
3	2.788494e-02
4	9.022269e-03
5	3.935075e-03
6	1.744588e-03
7	7.820785e-04
8	3.644176e-04
9	7.744053e-05
10	7.744053e-05

$\mathcal{D}_{KL}$ calculated between the SNR PDF of the NL channel (5) and the approximating $M\mathcal{G}$ distributions with varying components (N).

Observation

The most noticeable improvement in the $M\mathcal{G}$ approximation occurs as the number of components goes from $N = 3$ to $N = 4$. Thereafter, a marginally smaller improvement in the approximation is noted as N increases. To decide on the number of components necessary in the approximating $M\mathcal{G}$ distribution, N can be chosen such that the MSE or $\mathcal{D}_{KL}$ is below a specified value.

5.2 The K_G Channel

A simulation was performed in R with arbitrary $m = 2.5$, $k = 3$, $\lambda = \frac{km}{\overline{\gamma}}, \overline{\gamma} = 7.5$, $\alpha = k - m$, $\beta = m$ to approximate (11) by (13) using w_i and t_i as tabulated in Appendix.

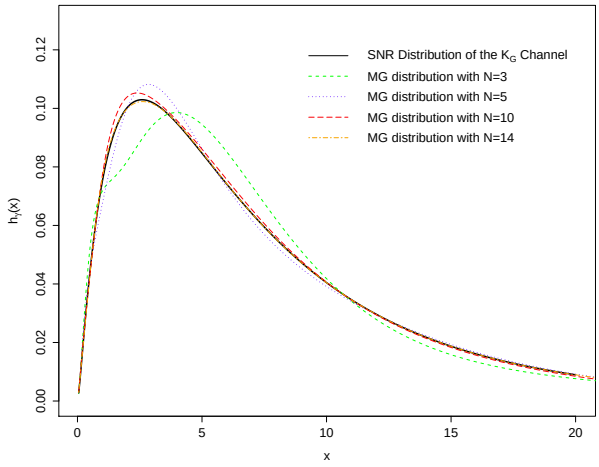
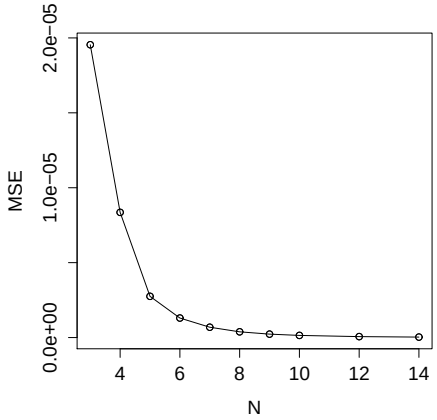


Figure 15: SNR PDF of the K_G channel (6) and approximating $M\mathcal{G}$ distributions with varying components (N)

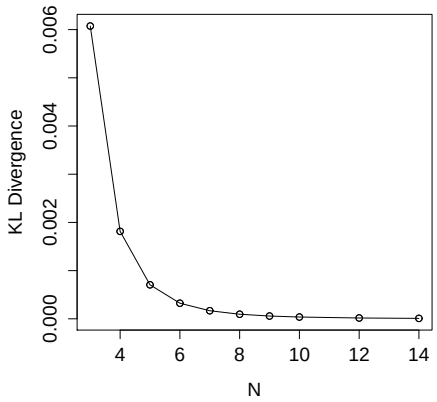
Mean Square Error (MSE)



N	MSE
3	1.954360e-05
5	2.752026e-06
8	3.827075e-07
10	1.434362e-07
12	6.321724e-08
14	3.128635e-08

MSE calculated between the SNR PDF of the K_G channel (6) and the approximating $M\mathcal{G}$ distributions with varying components (N).

Kullback-Leibler divergence ($\mathcal{D}_{KL}$)



N	$\mathcal{D}_{KL}$
3	6.073781e-03
5	7.054555e-04
8	9.517005e-05
10	3.715264e-05
12	1.736801e-05
14	9.187473e-06

$\mathcal{D}_{KL}$ calculated between the SNR PDF of the K_G channel (6) and the approximating $M\mathcal{G}$ distributions with varying components (N).

Observation

The most noticeable improvement in the $M\mathcal{G}$ approximation occurs, again, as the number of components goes from $N = 3$ to $N = 4$. Thereafter, a marginally smaller improvement in the approximation is noted as N increases. To decide on the number of components necessary in the approximating $M\mathcal{G}$ distribution, N can be chosen such that the MSE or $\mathcal{D}_{KL}$ is below a specified value.

6 Outage Probability of Composite Fading Channels and Approximating $M\mathcal{G}$ distribution

This section is concerned with calculating the outage probability of the theoretical and the approximating $M\mathcal{G}$ SNR distributions. Firstly, the concept of outage probability is defined. Secondly, theoretical outage probabilities are plotted against unfaded SNR (which is the SNR achieved in an fading-free environment) for varying threshold SNRs where threshold SNR refers to the minimum SNR such that a signal can still be transmitted. Thirdly, theoretical outage probabilities are plotted with the approximating $M\mathcal{G}$ representation outage probabilities for varying threshold SNR values to illustrate the use of the $M\mathcal{G}$ channel for performance analysis.

Outage Probability

The outage probability is a performance metric of a wireless channel and is defined as the probability P_{out}, that is, the probability that γ, the instantaneous SNR, falls below a specified threshold, γ^{th}, resulting in a signal 'dropping' [4].

$$P_{out} = \int_0^{\gamma^{th}} f_\gamma(s)ds = F_\gamma(\gamma^{th}) \tag{27}$$

For the purposes of this project, we consider only the outage probability for the NL channel.

6.1 Nakagami-lognormal Channel

Theoretical outage probabilities

For varying threshold SNRs, the CDF of the SNR distribution of the NL channel is calculated with a given unfaded SNR. This value is then plotted against the unfaded SNR.

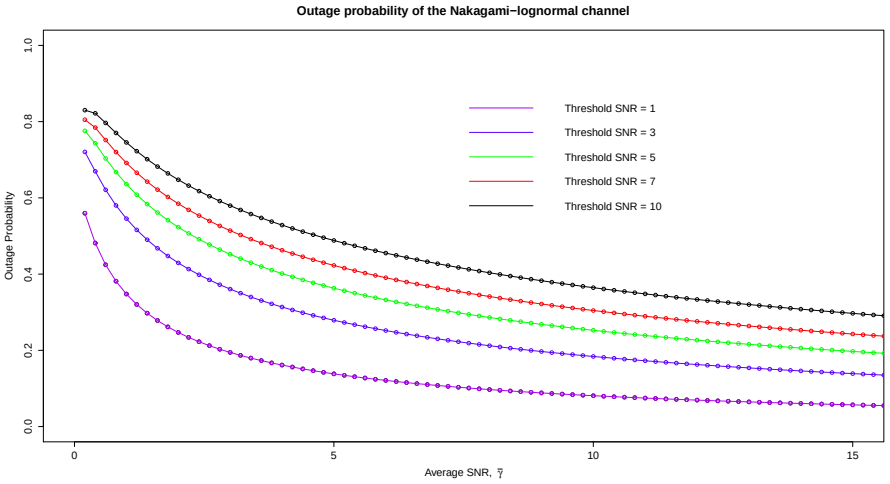


Figure 16: Outage probability of the SNR distribution of the NL channel with $m = 2$, $\mu = 1$, $\lambda = 2$ for different threshold SNRs

Observations:

- As the unfaded SNR increases, the outage probability decreases since the quality of the signal is improving
- Given an unfaded SNR, if the threshold SNR increases, the outage probability of the channel increases since it is more likely the SNR drops below a higher threshold SNR
- As the unfaded SNR becomes relatively large in comparison to the parameters chosen, the outage probabilities of the channels with different thresholds converge.

Comparison of theoretical outage probability and $M\mathcal{G}$ representation, $N = 5$

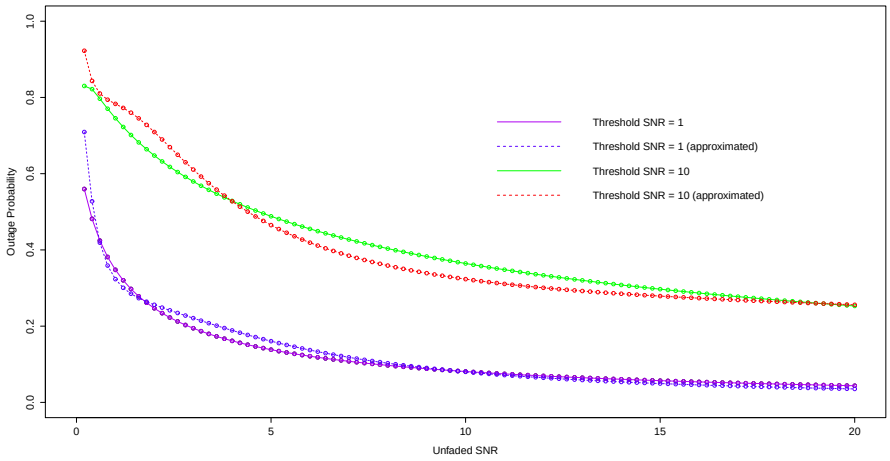


Figure 17: Outage probability of SNR distribution of the NL channel ($m = 2$, $\mu = 1$, $\lambda = 2$) and $M\mathcal{G}$ approximation with $N = 5$

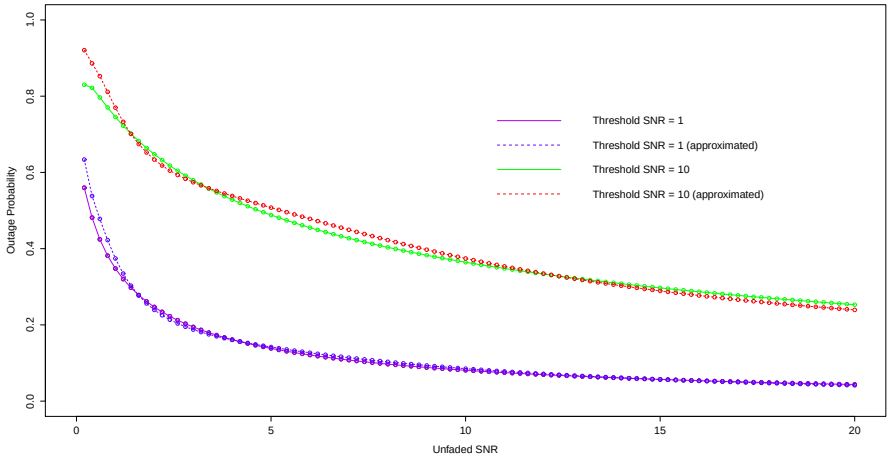


Figure 18: Outage probability of SNR distribution of the NL channel ($m = 2$, $\mu = 1$, $\lambda = 2$) and $M\mathcal{G}$ approximation with $N = 8$

Observations:

- The above two figures compares the outage probability of the theoretical NL channel and the approximating $M\mathcal{G}$ with $N = 5$ and $N = 8$ respectively;
- Accuracy of the approximation of outage probability of the NL channel increases as N increases; and
- The approximation performs badly at lower levels of unfaded SNR.

Outage probability given a change in fading parameter (m)

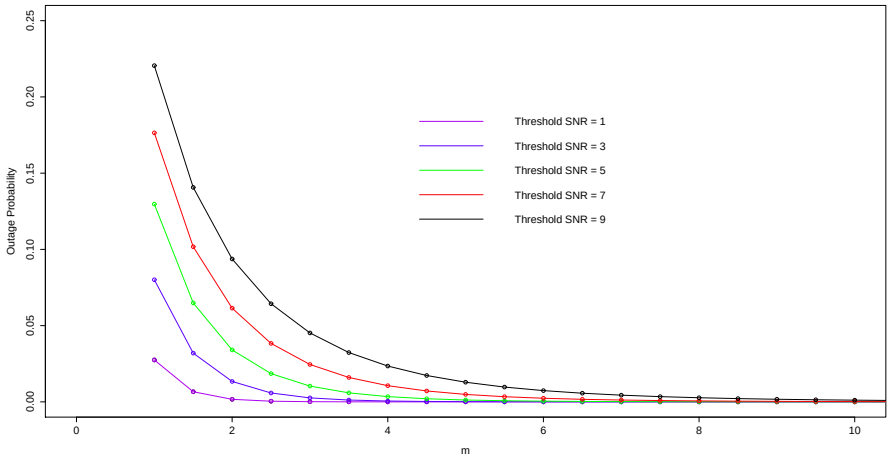


Figure 19: Outage probability of the SNR distribution of the NL channel against multipath-fading parameter m ($\rho = 5$, $\mu = 2$, $\lambda = 0.25$) for varying threshold SNRs

Observations:

- The multipath fading severity (i.e. to what degree the channel experiences fading) is inversely proportional to the parameter m;
- The outage probabilities are higher the smaller m is and decreases the larger m becomes;
- Given multipath fading parameter m , if the threshold SNR increases, the outage probability of the channel increases since it is more likely the SNR drops below a higher threshold SNR; and
- The outage probabilities begin to converge at larger values of m, where multipath fading becomes insignificant.

Outage probability given a change in shadowing parameter (λ)

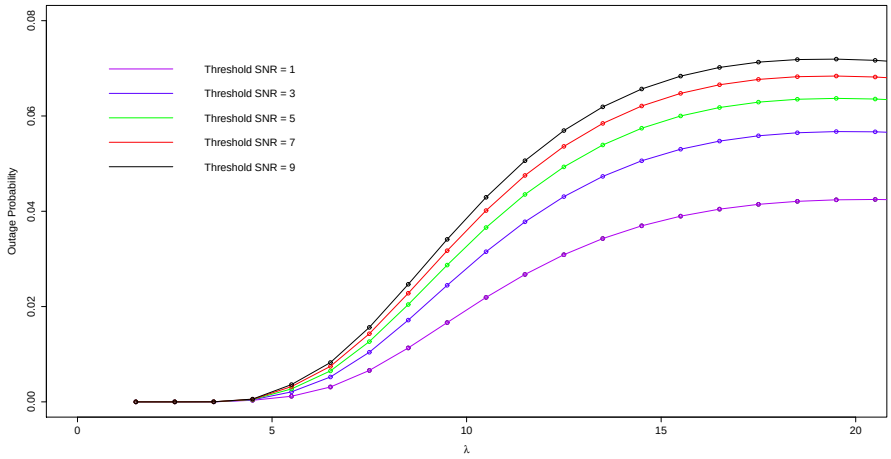


Figure 20: Outage probability of the SNR distribution of the NL channel against shadowing parameter λ ($\rho = 1$, $\mu = 20$, $m = 0.5$) for varying threshold SNRs

Observations:

- When the shadowing parameter, λ, is less than the mean of the shadowing distribution , μ, the outage probability increases with λ;
- The shadowing fading severity is directly proportional to the parameter λ;
- The outage probabilities are lower the smaller λ is; and increase the larger λ becomes;
- Given shadowing parameter λ, if the threshold SNR increases, the outage probability of the channel increases since it is more likely the SNR drops below a higher threshold SNR; and
- The outage probabilities begin to converge at smaller values of λ, where shadowing becomes insignificant.

7 Conclusion

In this study we investigated multipath and shadowing channels and provided the SNR PDFs of different channels. We then considered composite channels, where multipath fading and shadowing were modelled jointly.

The $M\mathcal{G}$ distribution to model the SNR of wireless was then introduced. It was shown that the SNR PDFs of the NL and K_G channels can be approximated as a $M\mathcal{G}$, where parameters of the $M\mathcal{G}$ are obtained using Gaussian-Quadrature approximation or by the matching of moments. As has been demonstrated, the $M\mathcal{G}$ representation offers high accuracy, as measured by MSE and $\mathcal{D}_{KL}$, and offers a closed form expression which facilitates the calculation of channel performance metrics.

The outage probability of the NL channel was studied and it was shown that the $M\mathcal{G}$ representation of this channel can be used to approximate outage probabilities to any degree of accuracy by increasing the number of components in the $M\mathcal{G}$ distribution.

8 Future work

Possible future areas of study may include the representation of other composite fading channels in the form of $M\mathcal{G}$ and other performance metrics such as average channel capacity (ACC), average bit error rate (ABER) andsSymbol error rate (SER) in $M\mathcal{G}$ representation. A possible extension would be to consider the representation of wireless channels as mixtures of non-central gamma distributions. Multivariate fading extensions, including multivariate Nakagami models, can also be explored.

References

[1] A. Abramowitz and I. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Department of Commerce: United States of America, 1972.

[2] S. Al-Ahmadi. *Composite fading channel modeling and information capacity of distributed antenna architectures in cellular networks*. PhD thesis, Carleton University, 2010.

[3] S. Al-Ahmadi. The gamma-gamma signal fading model: A survey. *IEEE Antennas and Propagation Magazine*, 56(5):245–260, 2014.

[4] S. Atapattu, C. Tellambura, and H. Jiang. A mixture gamma distribution to model the SNR of wireless channels. *IEEE Transactions on Wireless Communications*, 10(12):4193–4203, 2011.

[5] H. Bidgoli. *Handbook of Information Security, Key Concepts, Infrastructure, Standards, and Protocols*. John Wiley and Sons, 2006.

[6] P.S. Bithas. Mobile communication systems in the presence of fading/shadowing, noise and interference. *IEEE Transactions on Communications*, 2015.

[7] U. Charash. *A Study of Multipath Reception with Unknown Delays*. PhD thesis, University of California, 1962.

[8] D. Chotikapanich. *Economic Studies in Equality, Social Exclusion and Well-Being*. Springer, 2008.

[9] S. Cotton. Human body shadowing in cellular device-to-device communications: Channel modeling using the shadowed kappa - mu fading model. *IEEE Journal on Selected Areas in Communicationse*, 33(1):111–119, 2015.

[10] R. DeVore and G. Lorentz. *Constructive Approximation*. Springer Science & Business Media, 1993.

[11] I. Kostic. Analytical approach to performance analysis for channel subject to shadowing and fading. *IEEE Proceedings-Communications*, 152(6):821–827, 2005.

[12] S. Kullback. *Information Theory and Statistics*. PhD thesis, George Washington University, 1967.

[13] G. Mallya, S. Tripathi, and R. Govindaraju. Probabilistic drought classification using gamma mixture models. *Journal of Hydrology*, 2014.

[14] I. Mayrose. A gamma mixture model better accounts for among site rate heterogeneity. *Bioinformatics- Oxford Journal*, 21(2):151–158, 2005.

[15] K. Mella. *Theory, Simulation and Measurement of Wireless Multipath Fading Channels*. PhD thesis, Norwegian University of Science and Technology, 2007.

[16] M. Natarajan. *Cognitive Radio Technology Applications for Wireless and Mobile Ad Hoc Networks*. IGI Global, 2013.

[17] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. New York: McGraw-Hill, 1984.

[18] J. Salo. Why is shadow fading lognormal? In *International Symposium on Wireless Personal Multimedia Communications*, pages 522–526, Available: lib.tkk.fi/Diss/2006/isbn951228247X/article8.pdf, 2006.

[19] S. Sanja. Modelling and characterization of wireless channels in harsh environments. Master's thesis, Marladalen University, 2011.

[20] B. Selim. Modeling and analysis of wireless channels via the mixture of Gaussian distribution. Submited for publication, Available: http://arxiv.org/abs/1503.00877, 2015.

[21] M. Simon and M. Alouini. *Digital Communication Over Fading Channels*. John Wiley & Sons, 2005.

[22] G. Sánchez-Ferrero. Gamma mixture classifier for plaque detection in intravascular ultrasonic images. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 61(1):44–61, 2014.

[23] G. Sánchez-Ferrero. *Multi-Modality Atherosclerosis Imaging and Diagnosis*. Springer New York, 2014.

[24] M. D. Springer. *The Algebra of Random Variables*. Wiley, 1979.

[25] A. R. Webb. Gamma mixture models for target recognition. *Pattern Recognition*, 33(12):2045–2054, 2000.

[26] Z. Yang. Approximate methods for estimating the pattern of nucleotide substitution and the variation of substitution rates among sites. *Institute of Molecular Evolutionary Genetics and Department of Biology, Pennsylvania State University*, 13(5):650–659, 1996.

9 Appendix: Gaussian-Quadrature Approximation Weights and Nodes

9.1 Gaussian-Hermite weights (w_i) and nodes (t_i) for varying number of $M\mathcal{G}$ components (N) [1]

Table 2: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 3$ $M\mathcal{G}$ components

i	t_i	w_i
1	-1.224744871391589049099	0.29540897515091933788 3
2	0	1.181635900603677351532
3	1.224744871391589049099	0.29540897515091933788 3

Table 3: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 4$ $M\mathcal{G}$ components

i	t_i	w_i
1	-1.650680123885784555883	0.08131283544724517714303
2	-0.524647623275290317884 1	0.80491409000551283650 6
3	0.524647623275290317884 1	0.80491409000551283650 6
4	1.650680123885784555883	0.08131283544724517714 3

Table 4: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 5$ $M\mathcal{G}$ components

i	t_i	w_i
1	-2.020182870456085632929	0.019953242059045913207 74
2	-0.958572464613818507112 8	0.39361932315224115982 8
3	0	0.945308720482941881226
4	0.958572464613818507112 8	0.39361932315224115982 85
5	2.020182870456085632929	0.019953242059045913207 74

Table 5: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 6$ $M\mathcal{G}$ components

i	t_i	w_i
1	-2.350604973674492222834	0.004530009905508845640 9
2	-1.335849074013696949715	0.157067320322856643916 3
3	-0.436077411927616508679 2	0.724629595224392524092
4	0.436077411927616508679	0.724629595224392524092
5	1.335849074013696949715	0.157067320322856643916 3
6	2.350604973674492222834	0.004530009905508845640 86

Table 6: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 7$ $M\mathcal{G}$ components

i	t_i	w_i
1	-2.651961356835233492447	9.717812450995191541 5E-4
2	-1.673551628767471445032	0.054515582819127030592 2
3	-0.816287882858964663039	0.425607252610127800520 3
4	0	0.810264617556807326765
5	0.816287882858964663039	0.425607252610127800520 3
6	1.673551628767471445032	0.054515582819127030592 2
7	2.651961356835233492447	9.717812450995191541 5E-4

Table 7: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 8$ $M\mathcal{G}$ components

i	t_i	w_i
1	-2.930637420257244019224	1.9960407221136761920 6E-4
2	-1.981656756695842925855	0.017077983007413475456 2
3	-1.157193712446780194721	0.207802325814891879543 3
4	-0.381186990207322116854 7	0.66114701255824129103
5	0.381186990207322116854 7	0.661147012558241299103 04
6	1.157193712446780194721	0.207802325814891879543
7	1.981656756695842925855	0.017077983007413475456 2
8	2.930637420257244019224	1.9960407221136761920 6E-4

Table 8: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 9$ $M\mathcal{G}$ components

i	t_i	w_i
1	-3.19099320178152760723	3.9606977263264381904 6E-5
2	-2.266580584531843111802	0.004943624275536947217 22
3	-1.468553289216667931667	0.088474527394376573288
4	-0.723551018752837573323	0.432651559002555750199 8
5	0	0.720235215606050957124
6	0.723551018752837573322 6	0.432651559002555750199 8
7	1.468553289216667931667	0.088474527394376573288
8	2.266580584531843111802	0.004943624275536947217 22
9	3.19099320178152760723	3.9606977263264381904 6E-5

Table 9: Gaussian-Hermite weights (w_i) and nodes (t_i) for $N = 10$ $M\mathcal{G}$ components

i	t_i	w_i
1	-3.436159118837737603327	7.6404328552326206292E-6
2	-2.532731674232789796409	0.001343645746781232692201
3	-1.756683649299881773451	0.033874394455481063136 2
4	-1.036610829789513654177	0.240138611082314686416 5
5	-0.342901327223704608789	0.610862633735325798784
6	0.342901327223704608789 2	0.610862633735325798784
7	1.036610829789513654177	0.240138611082314686417
8	1.756683649299881773451	0.033874394455481063136 2
9	2.532731674232789796409	0.001343645746781232692201
10	3.436159118837737603327	7.6404328552326206292E-6

9.2 Gaussian-Laguerre weights (w_i) and nodes (t_i) for varying number of $M\mathcal{G}$ components (N) [1]

Table 10: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 3$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.4157745567834790833115	0.7110930099291730154 5
2	2.294280360279041719822	0.278517733569240848801
3	6.289945082937479196866	0.010389256501586135749

Table 11: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 4$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.3225476896193923118	0.603154104341633601636
2	1.745761101158346575687	0.357418692437799686641
3	4.536620296921127983279	0.038887908515005384272 44
4	9.395070912301133129234	5.392947055613274501 04E-4

Table 12: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 5$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.2635603197181409102031	0.5217556105828086524759
2	1.413403059106516792218	0.39866681108317592745
3	3.596425771040722081223	0.07594244968170759539
4	7.085810005858837556922	0.003611758679922048 4545
5	12.64080084427578265943	2.336997238577622789 1E-5

Table 13: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 6$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.222846604179260689464	0.458964673949963593568
2	1.188932101672623030743	0.417000830772120994113
3	2.992736326059314077691	0.113373382074044975738 7
4	5.77514356910451050184	0.010399197453149074898 91
5	9.837467418382589917716	2.610172028149320594 79E-4
6	15.98287398060170178255	8.985479064296212388 25E-7

Table 14: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 7$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.1930436765603624138382	0.40931895170127390213
2	1.026664895339191950345	0.421831277861719779929 3
3	2.567876744950746206908	0.147126348657505278395 4
4	4.900353084526484568102	0.020633514468716939865 71
5	8.182153444562860791082	0.001074010143280745522 13
6	12.73418029179781375801	1.586546434856420126 87E-5
7	19.39572786226254031171	3.170315478995580562 27E-8

Table 15: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 8$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.170279632051009997889	0.369188589341637529 9206
2	0.903701776799379912186	0.418786780814342956077
3	2.251086629866130689307	0.175794986637171805 6997
4	4.266700170287658793649	0.033343492261215651 5221
5	7.04590540239346569728	0.002794536235225672 52494
6	10.75851601018099522406	9.07650877335821310424E-5
7	15.74067864127800457803	8.485746716272531544 9E-7
8	22.8631317368892641057	1.048001174871510381615E-9

Table 16: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 9$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.152322227731808247428	0.336126421797962519673
2	0.807220022742255847741	0.411213980423984387309 1
3	2.005135155619347122983	0.199287525370885580860 6
4	3.783473973331232991675	0.047460562765651599262 1
5	6.204956777876612606974	0.005599626610794583177
6	9.37298525168757620181	3.052497670932105663 05E-4
7	13.4662369110920935711	6.592123026075352392 26E-6
8	18.833597788991696614 15	4.110769330349548442 9E-8
9	26.374071890927376796 14	3.290874030350707576 47E-11

Table 17: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 10$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.137793470540492430 8308	0.308441115765020141 55
2	0.729454549503170498 16	0.401119929155273551 516
3	1.808342901740316048233	0.21806828761180942159
4	3.401433697854899514483	0.062087456098677747 3929
5	5.552496140063803632418	0.009501516975181100554
6	8.330152746764496700239	7.530083885875387754 6E-4
7	11.84378583790006556492	2.825923349599565567 4E-5
8	16.27925783137810209953	4.249313984962686372 59E-7
9	21.99658581198076195128	1.839564823979630780922E-9
10	29.92069701227389155991	9.911827219609008558 38E-13

Table 18: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 11$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.1257964421879675226 76	0.284933212894200605056
2	0.665418255839227841678	0.389720889527849377938
3	1.647150545872169309587	0.23278183184899133394
4	3.091138143035254953302	0.076564435461966864009
5	5.02928440157983321237	0.014393282767350695 0919
6	7.509887863806616819411	0.001518880846484873 06985
7	10.60595099954696778056	8.513122435471922597 2E-5
8	14.43161375806418553532	2.292403879574504078 58E-6
9	19.17885740321467864782	2.486353702767795873 73E-8
10	25.21770933967756110409	7.712626933691320470 28E-11
11	33.49719284717553727319	2.883775868323623861 6E-14

Table 19: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 12$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.115722117358020675267	0.26473137105544319035
2	0.611757484515130665392	0.377759275873137982024
3	1.512610269776418786782	0.244082011319877564 2549
4	2.833751337743507228627	0.090449222211680930728
5	4.599227639418348461	0.020102381154634096 5227
6	6.844525453115177347754	0.002663973541865315 88105
7	9.621316842456867043912	2.032315926629993921 21E-4
8	13.00605499330634772035	8.365055856819798745 3E-6
9	17.11685518746225572818	1.668493876540910261 17E-7
10	22.151090379397005669 92	1.342391030515004145 52E-9
11	28.48796725098400031257	3.061601635035020781 42E-12
12	37.099121044466920336 64	8.148077467426241682 5E-16

Table 20: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 13$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.1071423884722523106485	0.2471887084299626213462
2	0.5661318990404018534 06	0.3656888229005219453067
3	1.3985643364510197179 28	0.2525624200576585023568
4	2.61659710840641129808	0.103470758024183705114
5	4.2388459290170332793 7	0.026432754415561615 7782
6	6.2922562711400737803 94	0.004220396040254752 76555
7	8.8150019411869780473 3	4.118817704727347748 92E-4
8	11.861403588811242576 22	2.351547398155323868 83E-5
9	15.510762037703752781 85	7.317311620249099104E-7
10	19.8846356638802283332	1.108841625703980679 79E-8
11	25.1852638646777580843	6.770826692205898840 6E-11
12	31.800386301947268371 37	1.159979959905076060 95E-13
13	40.72300866926557956 59	2.245093203892758415 99E-17

Table 21: Gaussian-Laguerre weights (w_i) and nodes (t_i) for $N = 14$ $M\mathcal{G}$ components

i	t_i	w_i
1	0.099747507032597574574	0.23181557714486497784
2	0.526857648851902896405	0.3537846915975431518
3	1.300629121251496481708	0.25873461024542808599
4	2.430801078730844636 17	0.115482893556923210087
5	3.932102822932188821 31	0.033192092159337360039
6	5.825536218301708419339	0.006192869437006610 2168
7	8.140240141565145030 06	7.398903778673859424E-4
8	10.91649950736601884081	5.490719466841698378 6E-5
9	14.210805011161288683 11	2.409585764085377496 76E-6
10	18.104892220218098412 55	5.801543981676495180 9E-8
11	22.723381628269624823 23	6.819314692484974119 62E-10
12	28.272981723248205695 42	3.221207751894847939 81E-12
13	35.149443660592426582 86	4.221352440516587351 6E-15
14	44.366081711117423041 63	6.052375022289188808 4E-19

10 Results

10.1 Gamma function

The gamma function given by Equation 6.1.1 in [1] is

$$\int_0^x x^{t-1}e^{-xs}dx = \frac{\Gamma(t)}{s^t} \ \forall s,t \text{ such that } \mathbb{R}(t) > 0$$

where $\mathbb{R}(t)$ denotes the real part of t.

10.2 Lower incomplete gamma function

The lower incomplete gamma function given by Equation 6.5.2 in [1] is

$$\gamma(a,x) \equiv \int_0^x t^{\alpha-1}e^{-t}dt \ \forall a \text{ such that } \mathbb{R}(a) > 0$$

where $\mathbb{R}(t)$ denotes the real part of t.

10.3 Zero-order modified Bessel function of the first kind

The zero-order modified Bessel function of the 1st kind is given by

$$I_0(x) \equiv \sum_{m=0}^{\infty} \frac{1}{m!\Gamma(m+1)} \left(\frac{x}{2}\right)^{2m} \text{where } \Gamma(.) \text{ is the gamma function.}$$

as given by Equation 9.6.10 in [1].
Note that $I_0(0) \equiv 1$.

10.4 Gamma distribution

If $X \sim GAM(k,\theta)$ then the PDF is defined in [17] as

$$f(x) = \frac{1}{\Gamma(\alpha)\theta^k}x^{k-1}e^{-\frac{x}{\theta}} \text{ for } x > 0 \tag{28}$$

where $k > 0$ and $\theta > 0$ are the shape and scale parameters respectively.

10.5 Kullback-Leibler divergence ($\mathcal{D}_{KL}$)

$\mathcal{D}_{KL}(f_{Ext} \parallel f_{App}) = \int_{-\infty}^{\infty} f_{Ext}(x)\log\frac{f_{Ext}(x)}{f_{App}(x)}dx$ is also known as relative entropy. It is a non-symmetric measure of the difference between the two probability distributions, f_{Ext} and f_{App}, which denotes the exact density and approximate density respectively. It measures the information lost when f_{App} is used to approximate f_{Ext} [12].

Properties:

1. $\mathcal{D}_{KL}(f_{Ext} \parallel f_{App}) \geq 0$ with equality $\iff f_{Ext} = f_{App}$ almost everywhere [12].
2. $\mathcal{D}_{KL}(f_{Ext} \parallel f_{App}) \neq \mathcal{D}_{KL}(f_{App} \parallel f_{Ext})$ (not symmetric) [12].

11 Code

11.1 Moment matching (only positive moments)

```
a = 0.05
b = 3
i = 0
omega = 1
#change
m =40#can be any number >= 0.5
k= 4
lambda = k*m
alpha_const = k-m

#SNR Distribution of the Kg Channel
matrix1 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
integrand <- function(t){((x^(m-1)*lambda**(m))/(gamma(m)*gamma(k))*
(exp(-t))*(t**(alpha_const-1))*exp(-(lambda*x)/t))}
It <- (integrate(integrand, lower = 0, upper = Inf))
f_x <- It$value
matrix1[i,1] <- x
matrix1[i,2]<-f_x
}
vec = matrix1[,2]
df = data.frame(matrix1)

#Gamma approximation if 1st positive and 3rd positive moments matched
K1 = ((m+1)*(k+1))/(m*k)
K2 = ((m+2)*(k+2))/(m*k)
gamma_k = 4/(-3+sqrt(9+8*(K1*K2-1)))
gamma_theta = ((-3+sqrt(9+8*(K2*K1-1)))*omega)/4

matrix2 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1 matrix2[i,1] <- x
matrix2[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}

#Gamma approximation if 1st positive and 2nd positive moments matched
K1 = ((m+1)*(k+1))/(m*k)
K2 = ((m+2)*(k+2))/(m*k)
gamma_k = 1/(K1-1)
gamma_theta = omega*(K1-1)
```

```
matrix3 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix3[i,1] <- x
matrix3[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}

#Gamma Approximation if 2nd positive and 3rd positive moments matched
K1 = ((m+1)*(k+1))/(m*k)
K2 = ((m+2)*(k+2))/(m*k)
gamma_k = ((-((K2**2)/K1)+4)+sqrt(((K2**2)/K1)**2+(8*((K2**2)/K1))))/2*(K2**2/K1-1)
gamma_theta = omega*sqrt(K1/(gamma_k**2+gamma_k))

matrix4 <- matrix(data=NA, nrow = 799, ncol=2)
for
(x in seq(from=a, to=b, by=0.05))
{ i = i+1 matrix4[i,1] <- x
matrix4[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}

#Plot Graph
plot(data.frame(matrix1),type="l", xlab="x", ylab="f(x)", ylim=c(0,1), lwd=2.5)
lines(data.frame(matrix2),lty=5,col="red",lwd=1.5)
lines(data.frame(matrix3),lty=5,col="green",lwd=1.5)
lines(data.frame(matrix4),lty=5,col="blue",lwd=1.5)
legend(
1.5,0.8,
c(expression(
'K'[G]*" Model "),
expression(1^{st}~ 'and'~ 2^{nd}~'Moments Matched'),
expression(1^{st}~ 'and'~ 3^{rd}~'Moments Matched'),
expression(2^{nd}~ 'and'~ 3^{rd}~'Moments Matched')),
lty=c(1,2,3,4), lwd=c(2.5,2.5,2.5,2.5),
col=c("black","red","green", "blue"),bty="n")
```

11.2 Moment matching (positive and negative moments)

```
a = 0.05
b = 3
i = 0
omega = 1

#change
m =40#can be any number >= 0.5
k= 4
lambda = k*m
alpha_const = k-m

matrix1 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
integrand <- function(t){((x^(m-1)*lambda**(m))/(gamma(m)*gamma(k))*
(exp(-t))*(t**(alpha_const-1))*exp(-(lambda*x)/t))}
It <- (integrate(integrand, lower = 0, upper = Inf))
f_x <- It$value
matrix1[i,1] <- x
matrix1[i,2]<-f_x
}

vec = matrix1[,2]
df = data.frame(matrix1)

#Gamma approximation if 1st positive and 1st negative moments matched
K_1 = ((m-1)*(k-1))/(m*k)
gamma_k = 1/(1-K_1)
gamma_theta = (1-K_1)*omega print(gamma_k)

matrix2 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1 matrix2[i,1] <- x
matrix2[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}
```

```
#Gamma approximation if 1st positive and 2nd negative moments matched
K_1 = ((m-1)*(k-1))/(m*k)
K_2 = ((m-2)*(k-2))/(m*k)
gamma_k = 4/(3-sqrt(9+8*(K_1*K_2-1)))
gamma_theta =((3-sqrt(9+8*(K_1*K_2-1)))*omega)/4
matrix3 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix3[i,1] <- x
matrix3[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}

#Gamma approximation if 1st negative and 2nd negative moments matched
K_1 = ((m-1)*(k-1))/(m*k)
K_2 = ((m-2)*(k-2))/(m*k)
gamma_theta =(1/m+1/k-3/(m*k))*omega
gamma_k = (K_1*omega/gamma_theta)+1

matrix4 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{ i = i+1 matrix4[i,1] <- x
matrix4[i,2]<- dgamma(x, shape=gamma_k, scale = gamma_theta, log = FALSE)
}

#Plot Graph
plot(data.frame(matrix1),type="l", xlab="x", ylab="f(x)", ylim=c(0,1), lwd=2.5)
lines(data.frame(matrix2),lty=5,col="red",lwd=1.5)
lines(data.frame(matrix3),lty=5,col="green",lwd=1.5)
lines(data.frame(matrix4),lty=5,col="blue",lwd=1.5)
legend( 1.5,0.8,
c(expression('K'[G]*" Model "),
expression(1^{st}~ 'Positive and'~ 1^{st}~'Negative Moments Matched'),
expression(1^{st}~ 'Positive and'~ 2^{nd}~'Negative Moments Matched'),
expression(1^{st}~ 'Negative and'~ 2^{nd}~'Negative Moments Matched')),
lty=c(1,2,3,4), lwd=c(2.5,2.5,2.5,2.5),
col=c("black","red","green", "blue"),bty="n")
```

11.3 Nakagami-lognormal channel SNR distribution with $M\mathcal{G}$ fit for $N = 3$ and $N = 10$

```
a = 0.05
b = 39.95
i = 0
#change m = 2.7 #can be any number >= 0.5
mu= 2
lambda = 1
rho = 1
beta=m

matrix1 <- matrix(data=NA, nrow = 799, ncol=3)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
integrand <- function(y){((x^(m-1)*exp(-(m*x)/(rho*y)))/gamma(m))*(m/(rho*y))^m*
(exp(-((log(y, base = exp(1)))-mu)^2/(2*lambda^2)))/(sqrt(2*pi)*lambda*y)}
It <- (integrate(integrand, lower = 0, upper = Inf))
f_x <- It$value
matrix1[i,1] <- x
matrix1[i,2]<-f_x
matrix1[i,3]<-0.05*matrix1[i,2]
}

matrix1cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0
matrix1cdf[1,2]<-matrix1[1,3]
for(x in seq(from=a, to=b, by=0.05))
{
i=i+1
matrix1cdf[i,1]<- x
matrix1cdf[i+1,2]<- matrix1cdf[i,2]+matrix1[i+1,3]
}
matrix1 <- matrix1[,-3] matrix1 <- matrix1[,-3]

library(zipfR)
##############################################
N=3

t1 =  -1.22474487139
t2=    0
t3=    1.22474487139
t = c(t1,t2,t3)

w1 = 0.295408975151
w2 = 1.1816359006
w3 = 0.295408975151
w = c(w1,w2,w3)
```

```
theta <- matrix(data=NA, nrow = N, ncol=1)
xi  <- matrix(data=NA, nrow = N, ncol=1)
alpha <- matrix(data=NA, nrow = N, ncol=1)

for(i in seq(from=1, to=N))
{
theta[i,1] <- ((m/rho)**(m))*(w[i]*exp(-m*(sqrt(2)*lambda*t[i]+mu)))/(sqrt(pi)*gamma(m))
}

for(i in seq(from=1, to=N))
{
xi[i,1] = (m/rho)*exp(-(sqrt(2)*lambda*t[i]+mu))
}

for(i in seq(from=1, to=N))
{
alpha[i,1] = theta[i]/((theta[1]*gamma(beta)*xi[1]**(-beta))+
(theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta)))
}

matrix3 <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0

for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix3[i,1] <- x
matrix3[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+
alpha[2]*(x**(beta-1))*exp(-xi[2]*x)+
alpha[3]*(x**(beta-1))*exp(-xi[3]*x)
}

matrix3 <- na.omit(matrix3)
matrix3cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0

for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix3cdf[i,1] <- x
matrix3cdf[i,2] <- alpha[1]*xi[1]**(-beta)*
Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)
}
```

```
##############################################
N=10
t1 =  -3.43615911884
t2=   -2.53273167423
t3=   -1.7566836493
t4 =  -1.03661082979
t5 =  -0.342901327224
t6=    0.342901327224
t7=    1.03661082979
t8=    1.7566836493
t9=    2.53273167423
t10=   3.43615911884
t = c(t1,t2,t3,t4,t5,t6,t7,t8,t9,t10)

w1 = 7.64043285523E-006
w2 = 0.00134364574678
w3 = 0.0338743944555
w4 = 0.240138611082
w5=  0.610862633735
w6=  0.610862633735
w7 = 0.240138611082
w8 = 0.0338743944555
w9=  0.00134364574678
w10= 7.64043285523E-006
w = c(w1,w2,w3,w4,w5,w6,w7,w8,w9,w10)

theta <- matrix(data=NA, nrow = N, ncol=1)
xi  <-   matrix(data=NA, nrow = N, ncol=1)
alpha <- matrix(data=NA, nrow = N, ncol=1)

for(i in seq(from=1, to=N))
{
theta[i,1] <- ((m/rho)**(m))*(w[i]*exp(-m*(sqrt(2)*lambda*t[i]+mu)))/(sqrt(pi)*gamma(m))
}

for(i in seq(from=1, to=N))
{
xi[i,1] = (m/rho)*exp(-(sqrt(2)*lambda*t[i]+mu))
}

for(i in seq(from=1, to=N))
{
alpha[i,1] = theta[i]/((theta[1]*gamma(beta)*xi[1]**(-beta))+
(theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta))+
(theta[4]*gamma(beta)*xi[4]**(-beta))+
(theta[5]*gamma(beta)*xi[5]**(-beta))+
(theta[6]*gamma(beta)*xi[6]**(-beta))+
(theta[7]*gamma(beta)*xi[7]**(-beta))+
(theta[8]*gamma(beta)*xi[8]**(-beta))+
(theta[9]*gamma(beta)*xi[9]**(-beta))+
(theta[10]*gamma(beta)*xi[10]**(-beta)))
}

a = 0.05
b = 39.95
```

```
i = 0

matrix10 <- matrix(data=NA, nrow = 799, ncol=2)

for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix10[i,1] <- x
matrix10[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+
alpha[2]*(x**(beta-1))*exp(-xi[2]*x) +
alpha[3]*(x**(beta-1))*exp(-xi[3]*x) +
alpha[4]*(x**(beta-1))*exp(-xi[4]*x) +
alpha[5]*(x**(beta-1))*exp(-xi[5]*x) +
alpha[6]*(x**(beta-1))*exp(-xi[6]*x) +
alpha[7]*(x**(beta-1))*exp(-xi[7]*x) +
alpha[8]*(x**(beta-1))*exp(-xi[8]*x) +
alpha[9]*(x**(beta-1))*exp(-xi[9]*x)+
alpha[10]*(x**(beta-1))*exp(-xi[10]*x)
}

matrix10 <- na.omit(matrix10)
matrix10cdf <- matrix(data=NA, nrow = 799, ncol=2)

a = 0.05
b = 39.95
i = 0
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix10cdf[i,1] <- x
matrix10cdf[i,2] <- alpha[1]*xi[1]**(-beta)*Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)+
alpha[4]*xi[4]**(-beta)*Igamma(beta, xi[4]*x, lower=TRUE, log=FALSE)+
alpha[5]*xi[5]**(-beta)*Igamma(beta, xi[5]*x, lower=TRUE, log=FALSE)+
alpha[6]*xi[6]**(-beta)*Igamma(beta, xi[6]*x, lower=TRUE, log=FALSE)+
alpha[7]*xi[7]**(-beta)*Igamma(beta, xi[7]*x, lower=TRUE, log=FALSE)+
alpha[8]*xi[8]**(-beta)*Igamma(beta, xi[8]*x, lower=TRUE, log=FALSE)+
alpha[9]*xi[9]**(-beta)*Igamma(beta, xi[9]*x, lower=TRUE, log=FALSE)+
alpha[10]*xi[10]**(-beta)*Igamma(beta, xi[10]*x, lower=TRUE, log=FALSE)
}
```

```
#Plot Graph (Approximation and Fit PDF)
par(mar=c(4,4.5,2,1))
plot(data.frame(matrix1), type="l", xlab="x", ylab=expression('h'[gamma]*"(x)"),
ylim=c(0,0.13),lwd=2.5)
lines(data.frame(matrix3),lty=2,col="purple",lwd=1.5)
lines(data.frame(matrix4),lty=2,col="blue",lwd=1.5)
lines(data.frame(matrix6),lty=4,col="green",lwd=1.5)
lines(data.frame(matrix10),lty=3,col="red",lwd=3.5)
legend(8.5,0.125,
c("SNR Distribution of the NL Channel",
"MG Distribution with N=3",
"MG Distribution with N=4",
"MG Distribution with N=6",
"MG Distribution with N=10"),lty=c(1,5,2,4,3),lwd=c(1.5,1.5,1.5,1.5,2.5),
col=c("black","purple","blue","green", "red"),
bty="n")


#Plot Graph (Approximation and Fit CDF)
par(mar=c(4,4.5,2,1))
plot(data.frame(matrix1cdf),type="l", xlab="x", ylab=expression('H'[gamma]*"(x)"),
ylim=c(0,1),lwd=2.5 )
lines(data.frame(matrix3cdf),lty=2,col="purple",lwd=1.5)
lines(data.frame(matrix4cdf),lty=2,col="blue",lwd=1.5)
lines(data.frame(matrix6cdf),lty=4,col="green",lwd=1.5)
lines(data.frame(matrix10cdf),lty=3,col="red",lwd=3.5)
legend(20,0.6,
c("SNR CDF of the NL Channel",
"MG CDF with N=3",
"MG CDF with N=4",
"MG CDF with N=6",
"MG CDF with N=10"),
lty=c(1,5,2,4,3), lwd=c(1.5,1.5,1.5,1.5,2.5),
col=c("black","purple","blue","green", "red"),
bty="n")
```

11.4 K_G channel SNR distribution with $M\mathcal{G}$ fit for $N = 3$ and $N = 14$

```
a = 0.05
b = 39.95
i = 0
#change
m = 2.5 #can be any number >= 0.5
k= 3
averageSNR = 7.5
lambda = k*m/averageSNR
alpha_const = k-m beta = m

matrix1 <- matrix(data=NA, nrow = 799, ncol=3)
for(x in seq(from=a, to=b, by=0.05)) shape
{
i = i+1
integrand <- function(t){((x^(m-1)*lambda**(m))/(gamma(m)*gamma(k))*
(exp(-t))*(t**(alpha_const-1))*exp(-(lambda*x)/t))}
It <- (integrate(integrand, lower = 0, upper = Inf))
f_x <- It$value
matrix1[i,1] <- x
matrix1[i,2]<-f_x
matrix1[i,3]<-0.05*matrix1[i,2]
}
vec = matrix1[,2]
df = data.frame(matrix1)
matrix1 <- na.omit(matrix1)

matrix1cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0
matrix1cdf[1,2]<-matrix1[1,3]
for(x in seq(from=a, to=b, by=0.05))
{
i=i+1
matrix1cdf[i,1]<- x
matrix1cdf[i+1,2]<- matrix1cdf[i,2]+matrix1[i+1,3]
}

library(zipfR)
##############################################
N=3
t1 =  0.4157745567834790833115
t2=   2.294280360279041719822
t3=   6.289945082937479196866
t = c(t1,t2,t3)

w1 = 0.71109300992917301545
w2 = 0.278517733569240848801
w3 = 0.010389256501586135749
w = c(w1,w2,w3)

theta <- matrix(data=NA, nrow = N, ncol=1)
xi  <- matrix(data=NA, nrow = N, ncol=1)
alpha <- matrix(data=NA, nrow = N, ncol=1)
```

```
for(i in seq(from=1, to=N))
{
theta[i,1] <- ((lambda)**(m))*(w[i]*t[i]**(alpha_const-1))/(gamma(m)*gamma(k))
}

for(i in seq(from=1, to=N))
{
xi[i,1] = lambda/t[i]
}

for(i in seq(from=1, to=N))
{
alpha[i,1] = theta[i]/((theta[1]*gamma(beta)*xi[1]**(-beta))+
(theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta)))
}
a = 0.05
b = 39.95
i = 0

matrix3 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{   i = i+1
matrix3[i,1] <- x
matrix3[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+
alpha[2]*(x**(beta-1))*exp(-xi[2]*x) +
alpha[3]*(x**(beta-1))*exp(-xi[3]*x)
}

matrix3 <- na.omit(matrix3)
matrix3cdf <- matrix(data=NA, nrow = 799, ncol=2)

a = 0.05
b = 39.95
i = 0

for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix3cdf[i,1] <- x
matrix3cdf[i,2] <- alpha[1]*xi[1]**(-beta)*Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)
}
```

```
############################################
N= 14
t1 =  0.099747507032597574574
t2=   0.52685764885190289 6405
t3=   1.300629121251496481708
t4=   2.43080107873084463617
t5=   3.932102822293218882131
t6=   5.825536218301708419339
t7=   8.14024014156514503006
t8=   10.91649950736601884081
t9=   14.21080501116128868311
t10=  18.10489222021809841255
t11=  22.72338162826962482323
t12=  28.27298172324820569542
t13=  35.14944366059242658286
t14=  44.36608171111742304163
t = c(t1,t2,t3,t4,t5,t6,t7,t8,t9,t10,t11,t12,t13,t14)

w1 = 0.23181557714486497784
w2 = 0.3537846915975431518
w3 = 0.25873461024542808599
w4=  0.115482893556923210087
w5=  0.033192092159337360039
w6=  0.0061928694370066102168
w7=  7.398903778673859424E-4
w8=  5.490719466841698 3786E-5
w9=  2.409585764085377 49676E-6
w10= 5.801543981676495 1809E-8
w11= 6.819314692484974 11962E-10
w12= 3.221207751894847 93981E-12
w13= 4.221352440516587 3516E-15
w14= 6.052375022289188 8084E-19
w = c(w1,w2,w3,w4,w5,w6,w7,w8,w9,w10,w11,w12,w13,w14)

theta <- matrix(data=NA, nrow = N, ncol=1)
xi  <- matrix(data=NA, nrow = N, ncol=1)
alpha <- matrix(data=NA, nrow = N, ncol=1)

for(i in seq(from=1, to=N))
{
theta[i,1] <- ((lambda)**(m))*(w[i]*t[i]**(alpha_const-1))/(gamma(m)*gamma(k))
}

for(i in seq(from=1, to=N))
{
xi[i,1] = lambda/t[i]
}
```

```
for(i in seq(from=1, to=N))
{
alpha[i,1] = theta[i]/((theta[1]*gamma(beta)*xi[1]**(-beta))+
(theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta))+
(theta[4]*gamma(beta)*xi[4]**(-beta))+
(theta[5]*gamma(beta)*xi[5]**(-beta))+
(theta[6]*gamma(beta)*xi[6]**(-beta))+
(theta[7]*gamma(beta)*xi[7]**(-beta))+
(theta[8]*gamma(beta)*xi[8]**(-beta))+
(theta[9]*gamma(beta)*xi[9]**(-beta))+
(theta[10]*gamma(beta)*xi[10]**(-beta))+
(theta[11]*gamma(beta)*xi[11]**(-beta))+
(theta[12]*gamma(beta)*xi[12]**(-beta))+
(theta[13]*gamma(beta)*xi[13]**(-beta))+
(theta[14]*gamma(beta)*xi[14]**(-beta)))
}

a = 0.05
b = 39.95
i = 0

matrix14 <- matrix(data=NA, nrow = 799, ncol=2)

for(x in seq(from=a, to=b, by=0.05))
{
  i = i+1
matrix14[i,1] <- x
matrix14[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+
alpha[2]*(x**(beta-1))*exp(-xi[2]*x) +
alpha[3]*(x**(beta-1))*exp(-xi[3]*x) +
alpha[4]*(x**(beta-1))*exp(-xi[4]*x) +
alpha[5]*(x**(beta-1))*exp(-xi[5]*x)+
alpha[6]*(x**(beta-1))*exp(-xi[6]*x)+
alpha[7]*(x**(beta-1))*exp(-xi[7]*x)+
alpha[8]*(x**(beta-1))*exp(-xi[8]*x)+
alpha[9]*(x**(beta-1))*exp(-xi[9]*x)+
alpha[10]*(x**(beta-1))*exp(-xi[10]*x)+
alpha[11]*(x**(beta-1))*exp(-xi[11]*x)+
alpha[12]*(x**(beta-1))*exp(-xi[12]*x)+
alpha[13]*(x**(beta-1))*exp(-xi[13]*x)+
alpha[14]*(x**(beta-1))*exp(-xi[14]*x)
}

matrix14 <- na.omit(matrix14)
matrix14cdf <- matrix(data=NA, nrow = 799, ncol=2)

a = 0.05
b = 39.95
i = 0
```

```
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix14cdf[i,1] <- x
matrix14cdf[i,2] <- alpha[1]*xi[1]**(-beta)*Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)+
alpha[4]*xi[4]**(-beta)*Igamma(beta, xi[4]*x, lower=TRUE, log=FALSE)+
alpha[5]*xi[5]**(-beta)*Igamma(beta, xi[5]*x, lower=TRUE, log=FALSE)+
alpha[6]*xi[6]**(-beta)*Igamma(beta, xi[6]*x, lower=TRUE, log=FALSE)+
alpha[7]*xi[7]**(-beta)*Igamma(beta, xi[7]*x, lower=TRUE, log=FALSE)+
alpha[8]*xi[8]**(-beta)*Igamma(beta, xi[8]*x, lower=TRUE, log=FALSE)+
alpha[9]*xi[9]**(-beta)*Igamma(beta, xi[9]*x, lower=TRUE, log=FALSE)+
alpha[10]*xi[10]**(-beta)*Igamma(beta, xi[10]*x, lower=TRUE, log=FALSE)+
alpha[11]*xi[11]**(-beta)*Igamma(beta, xi[11]*x, lower=TRUE, log=FALSE)+
alpha[12]*xi[12]**(-beta)*Igamma(beta, xi[12]*x, lower=TRUE, log=FALSE)+
alpha[13]*xi[13]**(-beta)*Igamma(beta, xi[13]*x, lower=TRUE, log=FALSE)+
alpha[14]*xi[14]**(-beta)*Igamma(beta, xi[14]*x, lower=TRUE, log=FALSE)
}

#Plot Graph (Approximation and Fit PDF)
par(mar=c(4,4.5,2,1))
plot(data.frame(matrix1), type="l", xlab="x", ylab=expression('h'[gamma]*"(x)"),
ylim=c(0,0.13),lwd=2.5)
lines(data.frame(matrix3),lty=5, lwd = 2.5,col="green")
lines(data.frame(matrix5),lty=2,lwd = 2.5, col="orange")
lines(data.frame(matrix10),lty=4, lwd = 2.5, col="purple")
lines(data.frame(matrix14),lty=3, lwd = 3.5, col="red")
legend(8.5,0.125,
c("SNR Distribution of the Kg Channel",
"MG Distribution with N=3",
"MG Distribution with N=5",
"MG Distribution with N=10",
"MG Distribution with N=14"),
lty=c(1,5,2,4,3), lwd=c(2.5,2.5,2.5,2.5,3.5),
col=c("black","green","orange", "purple", "red"),bty="n")

#Plot Graph (Approximation and Fit CDF)
par(mar=c(4,4.5,2,1))
plot(data.frame(matrix1cdf),type="l", xlab="x", ylab=expression('H'[gamma]*"(x)"),
ylim=c(0,1),lwd=2.5 )
lines(data.frame(matrix3cdf),lty=2,col="purple",lwd=1.5)
lines(data.frame(matrix5cdf),lty=2,col="blue",lwd=1.5)
lines(data.frame(matrix10cdf),lty=4,col="green",lwd=1.5)
lines(data.frame(matrix14cdf),lty=3,col="red",lwd=3.5)
legend(20,0.6,
c("SNR CDF of the NL Channel","MG CDF with N=3",
"MG CDF with N=4",
"MG CDF with N=6",
"MG CDF with N=10"),
lty=c(1,5,2,4,3), lwd=c(1.5,1.5,1.5,1.5,2.5),
col=c("black","purple","blue","green", "red"),
bty="n")
```

11.5 MSE and $\mathcal{D}_{\mathcal{KL}}$ calculations

```
#MSE Calculations MSE.plugin = function(pdf1, pdf2)
{
SSE <- matrix(data=NA, nrow = 799, ncol=1)
for(i in seq(from=1, to=799))
{
SSE[i,1] <- (pdf1[i,2]-pdf2[i,2])**2
}
MSE = mean(SSE,na.rm = TRUE)
return(MSE)
}

MSE3  <- MSE.plugin(matrix1, matrix3)
MSE4  <- MSE.plugin(matrix1, matrix4)
MSE5  <- MSE.plugin(matrix1, matrix5)
MSE6  <- MSE.plugin(matrix1, matrix6)
MSE7  <- MSE.plugin(matrix1, matrix7)
MSE8  <- MSE.plugin(matrix1, matrix8)
MSE9  <- MSE.plugin(matrix1, matrix9)
MSE10 <- MSE.plugin(matrix1, matrix10)
N <- c(3, 4, 5, 6, 7, 8, 9, 10)
MSE <- c(MSE3, MSE4, MSE5, MSE6, MSE7, MSE8, MSE9, MSE10)
plot(N, MSE, xlab="N", ylab="MSE");
lines(N, MSE, type="o")

#KL Calculations
KL.plugin = function(pdf1, pdf2)
{
pdf1 = pdf1/sum(pdf1) # ensure that that 'PDF' sums
pdf2 = pdf2/sum(pdf2)
LR = ifelse(pdf1 > 0, log(pdf1/pdf2), 0)
KL = sum(pdf1*LR)
return(KL)
}

KL3 <- KL.plugin(matrix1[,2], matrix3[,2])
KL4 <-KL.plugin(matrix1[,2], matrix4[,2])
KL5 <- KL.plugin(matrix1[,2], matrix5[,2])
KL6 <-KL.plugin(matrix1[,2], matrix6[,2])
KL7<- KL.plugin(matrix1[,2], matrix7[,2])
KL8 <-KL.plugin(matrix1[,2], matrix8[,2])
KL9 <- KL.plugin(matrix1[,2], matrix9[,2])
KL10 <-KL.plugin(matrix1[,2], matrix10[,2])
N <- c(3, 4, 5, 6, 7, 8, 9, 10)
KL <- c(KL3, KL4, KL5, KL6, KL7, KL8, KL9, KL10)
plot(N, KL, , xlab="N", ylab="KL Divergence");
lines(N, KL, type="o")
```

11.6 Outage probabilities of the Nakagami-lognormal channel and MG representation

```
## Nakagami-lognormal Channel Outage Probability
library(zipfR)
#Theoretical outage probability

matrix.plugin=function(threshold, rho)
{
a = 0.05
b = 39.95
i = 0

#change
m = 2 #can be any number >= 0.5
mu= 1
lambda =2
matrix1 <- matrix(data=NA, nrow = 799, ncol=3)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
integrand <- function(y){((x^(m-1)*exp(-(m*x)/(rho*y)))/gamma(m))*(m/(rho*y))^m*

It <- (integrate(integrand, lower = 0, upper = Inf))
f_x <- It$value
matrix1[i,1] <- x
matrix1[i,2]<-f_x
matrix1[i,3]<-0.05*matrix1[i,2]
}

matrix1cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b= 39.95
i = 0

matrix1cdf[1,2]<-matrix1[1,3]
x=a
for(i in seq(from=1, to=798, by=1))
{
matrix1cdf[i,1]<- x
matrix1cdf[i+1,2]<- matrix1cdf[i,2]+matrix1[i+1,3]
x <- x+0.05
}
return(matrix1cdf[threshold,1:2])
}

rows=100
rho_increment=0.2

rho=0
outageprob1 <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob1[i,1]<- rho
outageprob1[i,2]<- matrix.plugin(20,rho)[2] #threshold SNR = 1
}
```

```
rho=0
outageprob2 <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob2[i,1]<- rho
outageprob2[i,2]<- matrix.plugin(60,rho)[2] #threshold SNR = 3
}

rho=0
outageprob3 <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob3[i,1]<- rho
outageprob3[i,2]<- matrix.plugin(100,rho)[2] #threshold SNR = 5
}

rho=0
outageprob4 <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob4[i,1]<- rho
outageprob4[i,2]<- matrix.plugin(140,rho)[2] #threshold SNR = 7
}

rho=0
outageprob5 <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob5[i,1]<- rho
outageprob5[i,2]<- matrix.plugin(200,rho)[2] #threshold SNR = 9
}

################################################
#Plot Theoretical
plot(outageprob1[,1], outageprob1[,2],
xlab=expression("Average SNR, "~bar(gamma)), ylab="Outage Probability", ylim=c(0,1), xlim=c(0,15));
lines(outageprob1[,1], outageprob1[,2], type="o", col="purple")
lines(outageprob2[,1], outageprob2[,2], type="o", col = "blue")
lines(outageprob3[,1], outageprob3[,2], type="o", col="green")
lines(outageprob4[,1], outageprob4[,2], type="o", col="red")
lines(outageprob5[,1], outageprob5[,2], type="o", col="black")
legend(7,0.9,c("Threshold SNR = 1", "Threshold SNR = 3", "Threshold SNR = 5",
"Threshold SNR = 7", "Threshold SNR = 10"),
lty=c(1,1,1,1,1),
lwd=c(1.5,1.5,1.5,1.5,1.5),
col=c("purple","blue","green","red", "black"), bty="n")
```

```
##############################################
#MG Approximation for N=5

matrix.plugin5=function(threshold, rho)
{
a = 0.05
i = 0
#change
m = 2 #can be any number >= 0.5
mu= 1
lambda = 2

N=5
t1 =  -2.02018287046
t2=   -0.958572464614
t3=    0
t4 =   0.958572464614
t5 =   2.02018287046
t = c(t1,t2,t3,t4,t5)

w1 = 0.019953242059
w2 = 0.393619323152
w3 = 0.945308720483
w4 = 0.393619323152
w5=  0.019953242059
w = c(w1,w2,w3,w4,w5)

theta <- matrix(data=NA, nrow = N, ncol=1)
xi    <- matrix(data=NA, nrow = N, ncol=1)
alpha    <- matrix(data=NA, nrow = N, ncol=1)
for(i in seq(from=1, to=N))
{
theta[i,1] <- ((m/rho)**(m))*(w[i]*exp(-m*(sqrt(2)*lambda*t[i]+mu)))/(sqrt(pi)*gamma(m))
}
for(i in seq(from=1, to=N))
{
xi[i,1] = (m/rho)*exp(-(sqrt(2)*lambda*t[i]+mu))
}
for(i in seq(from=1, to=N))
{
alpha[i,1]=theta[i]/((theta[1]*gamma(beta)*xi[1]**(-beta))+(theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta))+(theta[4]*gamma(beta)*xi[4]**(-beta))+
(theta[5]*gamma(beta)*xi[5]**(-beta)))
}

a = 0.05
b = 39.95
i = 0
```

```
matrix5 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix5[i,1] <- x
matrix5[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+ alpha[2]*(x**(beta-1))*exp(-xi[2]*x)+
alpha[3]*(x**(beta-1))*exp(-xi[3]*x) + alpha[4]*(x**(beta-1))*exp(-xi[4]*x)+
alpha[5]*(x**(beta-1))*exp(-xi[5]*x)
}

matrix5cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix5cdf[i,1] <- x
matrix5cdf[i,2] <- alpha[1]*xi[1]**(-beta)*Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)+
alpha[4]*xi[4]**(-beta)*Igamma(beta, xi[4]*x, lower=TRUE, log=FALSE)+
alpha[5]*xi[5]**(-beta)*Igamma(beta, xi[5]*x, lower=TRUE, log=FALSE)
}
return(matrix5cdf[threshold,1:2])
}

rho=0
outageprob1mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob1mg[i,1]<- rho
outageprob1mg[i,2]<- matrix.plugin5(20,rho)[2]
}

rho=0 outageprob2mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob2mg[i,1]<- rho
outageprob2mg[i,2]<- matrix.plugin5(60,rho)[2]
}

rho=0 outageprob3mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob3mg[i,1]<- rho
outageprob3mg[i,2]<- matrix.plugin5(100,rho)[2]
}
```

```
rho=0
outageprob4mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob4mg[i,1]<- rho
outageprob4mg[i,2]<- matrix.plugin5(140,rho)[2]
}

rho=0
outageprob5mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob5mg[i,1]<- rho
outageprob5mg[i,2]<- matrix.plugin5(200,rho)[2]
}

#########################
#MG Approximation for N=8

matrix.plugin8=function(threshold, rho)
{
a = 0.05
b = 39.95
i = 0

#change
m = 2 #can be any number >= 0.5
mu= 1
lambda = 2
beta=m
N=8

t1 =  -2.93063742026
t2=    -1.9816567567
t3=   -1.15719371245
t4 = - 0.381186990207
t5 = 0.381186990207
t6= 1.15719371245
t7= 1.9816567567
t8=2.93063742026
t = c(t1,t2,t3,t4,t5,t6,t7,t8)

w1 = 0.000199604072211
w2 = 0.0170779830074
w3 = 0.207802325815
w4 = 0.661147012558
w5=  0.661147012558
w6=  0.207802325815
w7 = 0.0170779830074
w8 = 0.000199604072211
w = c(w1,w2,w3,w4,w5,w6,w7,w8)

theta <- matrix(data=NA, nrow = N, ncol=1)
xi  <- matrix(data=NA, nrow = N, ncol=1)
```

```
alpha <- matrix(data=NA, nrow = N, ncol=1)
for(i in seq(from=1, to=N))
{
theta[i,1] <- ((m/rho)**(m))*(w[i]*exp(-m*(sqrt(2)*lambda*t[i]+mu)))/(sqrt(pi)*gamma(m))
}
for(i in seq(from=1, to=N))
{
xi[i,1] = (m/rho)*exp(-(sqrt(2)*lambda*t[i]+mu))
}
for(i in seq(from=1, to=N))
{
alpha[i,1] = theta[i]/
((theta[1]*gamma(beta)*xi[1]**(-beta))+ (theta[2]*gamma(beta)*xi[2]**(-beta))+
(theta[3]*gamma(beta)*xi[3]**(-beta)) + (theta[4]*gamma(beta)*xi[4]**(-beta))+
(theta[5]*gamma(beta)*xi[5]**(-beta))+(theta[6]*gamma(beta)*xi[6]**(-beta))+
(theta[7]*gamma(beta)*xi[7]**(-beta))+(theta[8]*gamma(beta)*xi[8]**(-beta)))
}

a = 0.05
b = 39.95
i = 0
matrix8 <- matrix(data=NA, nrow = 799, ncol=2)
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix8[i,1] <- x
matrix8[i,2]<- alpha[1]*(x**(beta-1))*exp(-xi[1]*x)+ alpha[2]*(x**(beta-1))*exp(-xi[2]*x)
+ alpha[3]*(x**(beta-1))*exp(-xi[3]*x) +
alpha[4]*(x**(beta-1))*exp(-xi[4]*x) +
alpha[5]*(x**(beta-1))*exp(-xi[5]*x) +
alpha[6]*(x**(beta-1))*exp(-xi[6]*x) +
alpha[7]*(x**(beta-1))*exp(-xi[7]*x) +
alpha[8]*(x**(beta-1))*exp(-xi[8]*x)
}      matrix8 <- na.omit(matrix8)

matrix8cdf <- matrix(data=NA, nrow = 799, ncol=2)
a = 0.05
b = 39.95
i = 0
for(x in seq(from=a, to=b, by=0.05))
{
i = i+1
matrix8cdf[i,1] <- x
matrix8cdf[i,2] <-
alpha[1]*xi[1]**(-beta)*Igamma(beta, xi[1]*x, lower=TRUE, log=FALSE)+
alpha[2]*xi[2]**(-beta)*Igamma(beta, xi[2]*x, lower=TRUE, log=FALSE)+
alpha[3]*xi[3]**(-beta)*Igamma(beta, xi[3]*x, lower=TRUE, log=FALSE)+
alpha[4]*xi[4]**(-beta)*Igamma(beta, xi[4]*x, lower=TRUE, log=FALSE)+
alpha[5]*xi[5]**(-beta)*Igamma(beta, xi[5]*x, lower=TRUE, log=FALSE)+
alpha[6]*xi[6]**(-beta)*Igamma(beta, xi[6]*x, lower=TRUE, log=FALSE)+
alpha[7]*xi[7]**(-beta)*Igamma(beta, xi[7]*x, lower=TRUE, log=FALSE)+
alpha[8]*xi[8]**(-beta)*Igamma(beta, xi[8]*x, lower=TRUE, log=FALSE)
}
return(matrix8cdf[threshold,1:2])
}
```

```
rows=100
rho_increment=0.2

rho=0
outageprob1mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob1mg[i,1]<- rho
outageprob1mg[i,2]<- matrix.plugin8(20,rho)[2]
}

rho=0
outageprob5mg <- matrix(data=NA, nrow = rows, ncol=2)
for(i in seq(from=1, to=100, by=1))
{
rho=rho+rho_increment
outageprob5mg[i,1]<- rho
outageprob5mg[i,2]<- matrix.plugin8(200,rho)[2]
}

#Plot Theoretical Outage Probabilities
plot(outageprob1[,1], outageprob1[,2], xlab="Average SNR", ylab="Outage Probability",
ylim=c(0,1), xlim=c(0,20));
lines(outageprob1[,1], outageprob1[,2], type="o", col="purple")
lines(outageprob2[,1], outageprob2[,2], type="o", col = "blue")
lines(outageprob3[,1], outageprob3[,2], type="o", col="green")
lines(outageprob4[,1], outageprob4[,2], type="o", col="red")
lines(outageprob5[,1], outageprob5[,2], type="o", col="black")
legend(8,0.8,
c("Threshold SNR = 1",
"Threshold SNR = 3",
"Threshold SNR = 5",
"Threshold SNR = 7",
"Threshold SNR = 10"),lty=c(1,1,1,1,1), lwd=c(1.5,1.5,1.5,1.5,1.5),
col=c("purple","blue","green","red", "black"), bty="n")

#Plot Theoretical and MG Channel Outage Probablities
plot(outageprob1[,1], outageprob1[,2], xlab="Unfaded SNR", ylab="Outage Probability",
ylim=c(0,1), xlim=c(0,20));
lines(outageprob1[,1], outageprob1[,2], type="o", col="purple")
lines(outageprob1mg[,1], outageprob1mg[,2], type="o", col="blue", lty=2)
lines(outageprob5[,1], outageprob5[,2], type="o", col = "green")
lines(outageprob5mg[,1], outageprob5mg[,2], type="o", col="red", lty=2)
legend(10,0.8
c("Threshold SNR = 1",
"Threshold SNR = 1 (approximated)",
"Threshold SNR = 10",
"Threshold SNR = 10 (approximated)"),lty=c(1,2,1,2), lwd=c(1.5,1.5,1.5,1.5),
col=c("purple","blue","green","red"), bty="n")
```

Statistical robotics

Prenil Sewmohan, 12064620

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisors: Dr. FHJ Kanfer, Dr. I Fabris-Rotelli and Mr. S Millard

Department of Statistics, University of Pretoria

2 November 2015

Abstract

This report outlines the key concepts in robotics with respect to statistical theory. It focuses on the importance of stochastic and statistical methods in robot programing, processing and perception. The premise is that integrating statistical methods into robotics programming results in robots which have a higher degree of intelligence. There are various different opinions on what constitutes intelligence in robotics. The Florida Institute for Human and Machine Cognition define artificial intelligence as "the ability of a system to act appropriately in an uncertain environmentl". It will be with a similar criteria for intelligence that this paper assesses the role of statistical programming in robotics. This will be done with specific reference to state estimation techniques, using information filters and the localization problem. The aim is to set out the basic terminology and theory behind programming a robot statistically, while also programming a robot to track a moving object. Then finally to grab data from the completion of this task and analyse it with the tools and theory previously examined in an attempt to practically illustrate the theory by improving the initial task.

Declaration

I, *Prenil Sewmohan,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Prenil Sewmohan

_____
Dr. FHJ Kanfer, Dr. I Fabris-Rotelli, Mr. S Millard

_____
Date

Acknowledgements

This work is based on the research supported in part from the National Research Foundation of South Africa for the Grant 90315. Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard.

Contents

1 Introduction

The use and application of robotics within numerous industries and manufacturing is one of the main driving forces behind further invention and improvement within the field of robotics. This implies that, although we can program robots to do very simple repetitive tasks inside a controlled environment, the ideal to which all robotics should be moving towards is that of robots capable of navigating dynamic and complex environments. While also completing tasks in these ambiguous environments and learning from their own experience.

While it is theoretically possible to deterministically program every possible decision which the programmer can imagine into the robot's programming, it is impossible to account for every conceivable eventuality. Furthermore, all programs share a common set of constraints around which every programmer must work. One of the most important constraints is presented not by the complexity of code that the programmer can conceive but rather by the computational strain which that code presents to the machine processing it. It becomes clear that deterministic programs face a trade-off between being flexible and adaptable, and trying to keep the complexity of their code as low as possible to comply with the constraints imposed by the available technology.

Additionally, a robot operated in any real world environment will have to deal with the large amounts of inherent uncertainty due to changing conditions, moving objects and its own interactions with objects in that environment. A deterministically programmed robot will not be able to cope with this kind of uncertainty. It wouldn't recognize that it is making a mistake if it does. On the other hand a robot programmed stochastically will have the ability to deal with the inherent uncertainty and account for each eventuality. This ability to adapt to uncertain environments will also allow the robot to learn from its environment and correct its own mistakes.

Furthermore, a robot cannot directly perceive its environment whether it is due to not having enough sensors, inadequate sensors or just the latent noise present in all sensors. Hence, it will need to possess the ability to estimate its environment from whatever sensor data which it is able to gather whilst also accounting for the noise present. A deterministic robot is incapable of this kind of estimation since by default it must assume that its information is perfect. Stochastic programming will enable the robot to filter out the noise and estimate its true environment much more reliably than its deterministic counterparts.

This sets the stage for robots which use stochastic and statistical methods in their programming. By definition, no statistical program can be theoretically 100% accurate due to them being based on statistical theory pertaining to state evolution. This being said, statistical programming in general will be less complex than an equivalent deterministic program. Robots programmed in this way are able to inherently deal with the uncertainty that characterizes real life environments. That is; the uncertainty in its position, uncertainty from its sensors, surrounding noise and that resulting from dynamic environments. This gives the robot the ability to make inferences on and learn about its environment through interaction with that environment.

The Bayes Filter Algorithm provides an effective and efficient way to compute the state of a robot, however, due to its complexity in all but very strict conditions it must be estimated to avoid unnecessary complexity. Two methods of approximation are to use Gaussian filters or nonparametric filters. In this paper we aim to focus on a specific type of Gaussian filter, called the Kalman filter, which is a technique for filtering and prediction in linear Gaussian systems. It works on the assumption of continuous states and represents the robot's beliefs through moment parametrization.

A robot's perception of its environment is an important part of robotics. In particular, it is useful to be able to differentiate between different colours in an environment. This is useful for detection of objects in that environment. Colours are digitally represented in the red, green, blue (RGB) format [16] or the alpha, red, green, blue (ARGB) format [22]. In the RGB format colours are stored in memory using three 8-bit numbers. This gives us a range from 0 to 255 for each number. These numbers represent the amount each of the primary red, green or blue components present in the specific colour being defined. For example, black would be represented by red, green and blue components all zero. The second format (the 32-bit ARGB format) is one of the most common formats for storing this colour. In the 32-bit ARGB format each component of the vector has 8 bits allocated to it. Alpha can range from 0 to 1 and the colour components range from 0 to 255. Using these two methods the colour of any given pixel in an image is then defined by a vector of three or four values respectively. Both contain red, green and blue components in the vector to define colour. The ARGB format has an alpha channel as well. This alpha channel defines the transparency

of the pixel to its background. If we were to look for a specific colour then we could easily filter the image by only showing pixels with colour vectors where each vector component is defined to be within a specified range. Furthermore, all colours within a more constrained range can then be redefined to have the colour represented by the mean vector of that constrained range.

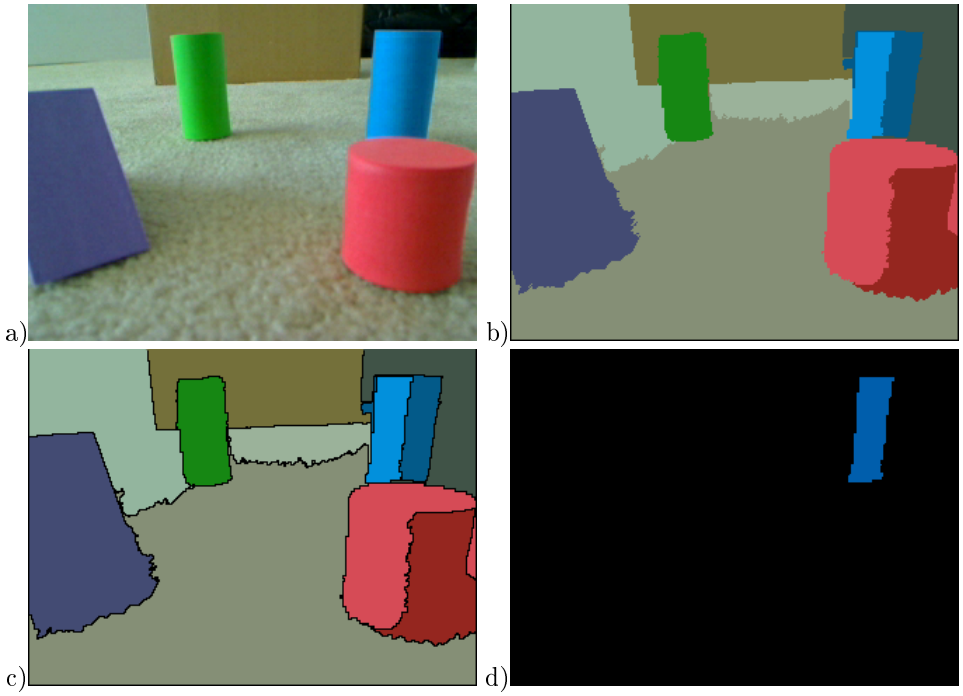


Figure 1: a) Gives the original picture. b) Picture after colours have been averaged. c) Same as b) with different blobs separated by borders d) Colour filter for just blue

This makes it easy to define what are called blobs. A blob is a region in an image of pixels with a similar colour [11]. These blobs can be defined by identifying regions which have colours that all fall into some predefined range. All pixels within that region become blob. The wider the acceptable region of colour for each group, the less accurate the detection of smaller regions becomes. The more narrow the regions, the higher accuracy with respect to identifying small, specific regions and the higher the risk of identifying one singular region as being composed of more than one blob. The blobs can then be further constrained by specifying the maximum or minimum size which a blob can be defined to. This will result in a number of objects which can be identified from the original image and can then be processed to feed into the robot's data about its environment and for example, any obstacles that are in its proximity. This process is represented in Figure 1. In a live feed a blob can be defined and then tracked as it moves through the frames.

Closely related to the idea of identifying objects in the environment, is the need to identify motion and track where and how objects move. There are various different ways to accomplish this. Some of these methods are background subtraction, Gaussian background subtraction and applying the Kalman filter either to data from some kind of background subtraction or from a sensor which is capable of measuring the coordinates of the moving object [14][25]. The Kalman filter uses a mixture normal distribution to estimate the distribution of some characteristic of the data. The advantage of using the Kalman filter is that it allows the robot to estimate the location of the moving object even if it cannot be observed for some period of time. This fact very nicely illustrates the advantages of using statistical programming [9].

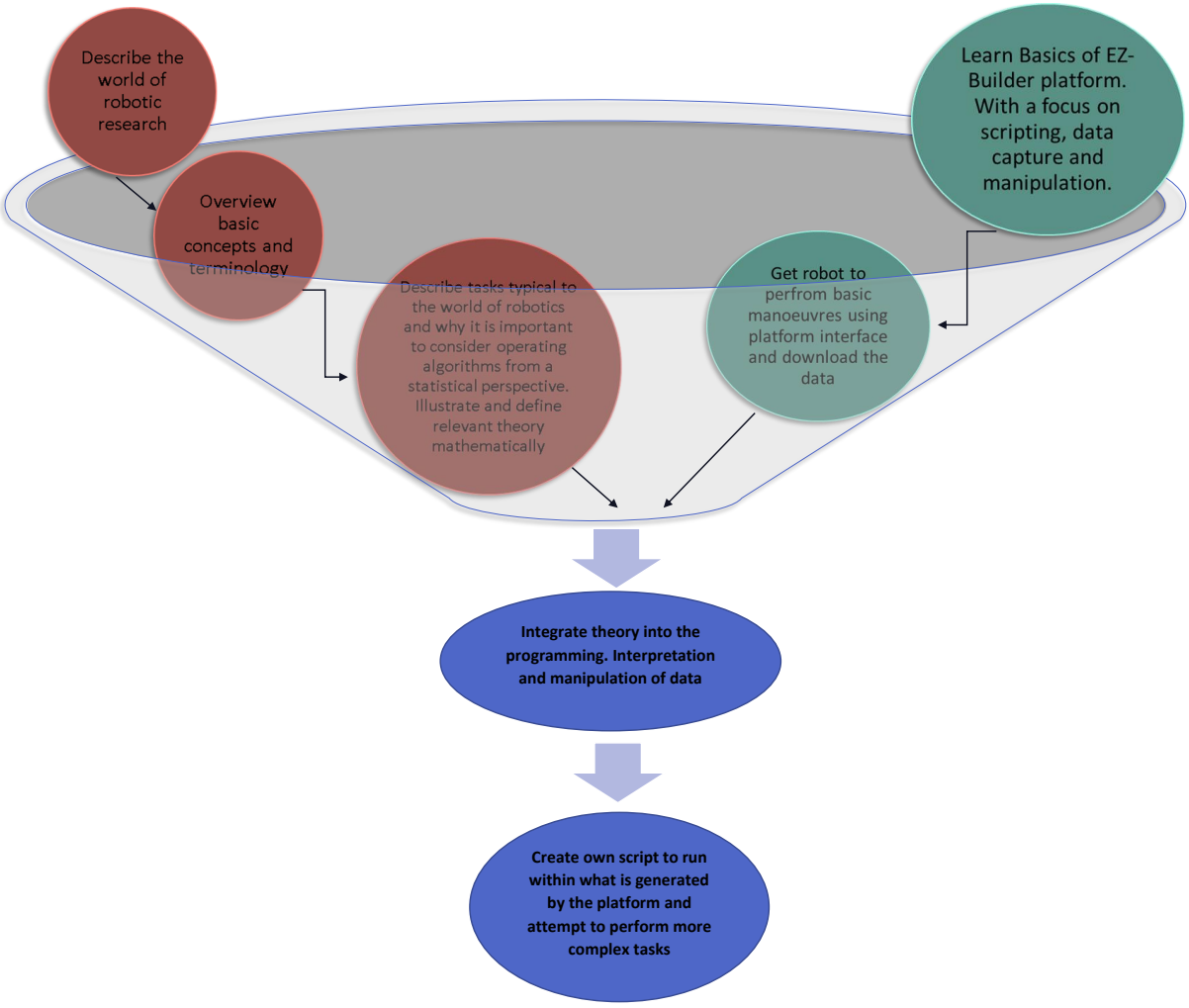


Figure 2: Method of approach

This research essay will have a general focus on the basics of probabilistic robotics with two parallel streams of investigation as illustrated in Figure 2. The first stream deals with the basic terminology and concepts of statistical robotics; leading towards the theory and importance of statistics in robotics and illustrating this mathematically. The second stream focuses on the practical implementation of the theory, working in the EZ-Builder platform[1] and using an EZ robot (with help from the online community forum[2]) to practically demonstrate the theory. It will focus on the practical programming as well as downloading and analysing real time data. The aim thereafter is to merge the two streams and try to implement the theory into the practical coding and script supplied by the EZ-Builder in order to get the robot to track a moving object using the Kalman filter. The addition of more complex tasks and algorithms will be contingent on succeeding in these preliminary goals as well as on time constraints.

[1] More information about this at http://www.ez-robot.com
[2] http://www.ez-robot.com/Community/Forum/

2 Literature Review

2.1 Basic Terminology and Bayesian Filters

Thrun, in [27], deals with the programming methodology for probabilistic robots. It contends that stochastic algorithms scale-up better than others in dealing with real world situations. This notion is defended by the fact that most interesting environments and most environments with real applications for advanced robotics inherently contain some uncertainty. This uncertainty is compounded by robot sensors being imperfect and so picking up large amounts of noise or just simply not being able to perceive the environment in the way required (even in a certain environment). This means that the robot must rely instead on inferring the information it needs from what data it can gather. This uncertainty is dealt with by programming the robot to interact with and perceive its environment in terms of probability densities. This affords the robot the ability to react to changes and uncertainty, even recovering from a failure. This makes the program quite robust. However, there are drawbacks to this robustness and impressive capability. Firstly, the fact that computing entire probability densities makes the robots less efficient and slower because the integrals involved are very expensive to compute. Secondly, some degree of approximation is necessary since the robots need to work in a continuous environment and computing an infinite probability distribution is not feasible computationally. That being said, research is continually leading to more efficient algorithms and new technology increases in computing power at a steady pace.

In [2], Baum et al. set out the mathematical basis for the estimation of states and measurements under the assumption of finite Markov chains. This assumption serves to simplify the models so as to reduce the amount of memory required and the computational times involved. Practical situations rarely conform to the constraints of Markov chains completely however it is reasonable to suggest that the assumption is valid enough for the model to be useful. This is one of the key assumptions which underpins the theory behind the Bayesian filter at the heart statistical robotics.

Cassandra et al. in [4] explore the use of the Bayesian framework for modelling a robots belief about its environment in addition to sub-optimal control strategies given the Bayesian belief state. It contends that in certain cases (moving through a door for example) the full pose of the robot is not always a necessity and the robot would be able to move through the door simply if it knew that it was in the area of a door-like structure. Essentially, we need only consider the relevant information for a particular decision and not all available information. This results in a more coarse-grained probability model, saving time and resources on computation. They then attempt to present an optimal control strategy for a robot using discrete models. The models presented however have a high degree of dependence on the world being static and can cope only with transitions to transient states, not absorbent ones.

Bayesian filters, due to their complexity, have to be approximated. One method for accomplishing this is to use a Gaussian filter. One specific type of Gaussian filter that has been used widely in the field is the Kalman filter. A very good explanation of this model is provided by Faragher in [9] by way of a derivation.

2.1.1 Bayesian networks

Bayesian networks are explained by Ghahramani in [12]. They graphically represent conditional independence relations between a number of random variables. These graphical representations consist of nodes which represent variables and the nodes are connected by arcs. A node which receives and arc from another node is conditionally dependent on that node. The node which the arc comes from is called the parent and the node receiving the arc is the child. The descendants of a node are its children and all of their children and so on. Each node is then conditionally independent of nodes which are not descendent from it given its parents.

Dynamic Bayesian networks are used in statistical robotics for computing what is called the robot's state. The state of a robot is a vector containing all the information which the robot has stored about its environment and itself. The initial state is represented by a particular probability distribution and is then updated as new information becomes available and the new distribution of the state is given by a mixture model. A dynamic Bayesian network is a Bayesian network applied to time series data. Dynamic Bayesian networks are a class of Bayesian networks which hidden Markov models are part of.

Hidden Markov models are used widely for analysing time series data. They are also notably used in most speech recognition technology as well as computer vision. Essentially, they represent probability distributions

over a number of observations. Hidden Markov models can be adjusted to also allow for an input variable. In this case the conditional distribution of a sequence of outputs variables given some sequence of inputs [13]. The Kalman filter (also known as the linear-Gaussian state-space model falls into the same subset of dynamic Bayesian models as the hidden Markov model. The Kalman filter is essentially a hidden Markov model with a continuous state variable.

2.2 Colour recognition

The human visual system is thought to use three different colour components to produce the sensation of colour. Digital graphics take after this idea and create very large ranges of colours using just three primaries - red, green and blue. The various "versions" of the three primaries used influence what is called the colour space (the collection of all possible colours which can be created from the three primaries used). A similar space is created using cyan, magenta and yellow as primaries. This is explained by Joblove in [16].

Most commonly, colours are defined in terms of their RGB (red, green and blue) components. However, when working with digital images, it is useful to add a matte element to these three dimensional colour vectors. In their paper, [22], Porter and Duff show the importance of using the RGBA (red, green, blue and alpha) scheme in synthetic pictures. They call this matte element the alpha channel. It controls the extent to which each pixel in an object mixes (the mixing factor) with the colour of the background behind it. An alpha channel value of 1 corresponds to full coverage by the pixel of its background and a value of 0 corresponds to no coverage. This method was pioneered by Edwin Catmull in his 1978 paper [5].

In [29] Voorhees deals with a method for grouping small compact and elongated linear images into objects called blobs or bars in a greyscale environment. Blobs are generally defined to be areas which are lighter or darker than their backgrounds. Blobs are fairly insensitive to shadows and other large scale changes in illuminations. Positive values indicate regions in the image which are relatively darker than their surroundings.

Blobs can also be used with colour. In [11] Gavilan et al. define blobs as 4D objects in the scale space and colour blobs as $3+n$ degree objects where n corresponds to the degree of the colour space. So for RGBA objects the colour blob will be 7D. They then use the amount of relation between objects in the environment to define blobs within the picture.

2.3 Image recognition

One of the big goals of robotics is, essentially, to endow robots with "senses". One such sense is sight. Lowe, in [19], describes a computer vision system which recognizes a three-dimensional object from a viewpoint which is unknown to the computer system in single greyscale images. It accomplishes this without trying to reconstruct the depth information in the image. Three other mechanisms are used to move from the 2D image to 3D object detection: perceptual organization forms groups of pixels in the image which will likely remain unchanged over a number of different perspectives of the image, the items in the image are ranked probabilistically to decrease the number of items that must be searched through when matching models and lastly a process of spatial correspondence solves for viewpoints which are unknown and also finds model parameters. This method is quite robust when dealing with missing data and occlusion (when, due to set up or sensor properties, some property which you want to observe is rendered unobservable).

[26] deals with object recognition using colour instead of the more widely used shape algorithms with the aim of faster processing. He demonstrates that colour histograms of objects with many colours can be used to form an efficient cue for indexing into database of models. It also shows that they are able to differentiate amongst a large number of objects. Histogram Intersection is used to match histograms of the images and the models to deal with identification of an object in a known location. Furthermore histogram backprojection is utilized to locate a known object in "crowded" images. In a similar manner, Lee et al. in their 1994 paper [17] present an automatic recognition method for car license plate using colour detection.

One of the biggest problems to overcome when giving a robot any kind of sensors is how to get the robot to focus on relevant information from the sensors and ignore the background noise which will be picked up as well.. Astola et al. [1] explain that one way of filtering image noise out is through the use of the median filter. This filter is nonlinear and works by moving a window over an area which contains some signal or property and then finding the median of the values inside this window and returns that value as the output.

In [18] Lee presents an algorithm for digital image noise filtering. Extending Lee's local statistics method by using local gradient information, is used to do this. Most other noise filtering techniques require extensive modelling and result in images which have a significant loss in contrast. Using this method noise along the edges is reduced, thereby maintaining contrast.

2.4 Motion-based recognition

Motion tracking or motion-based recognition uses a series of images and from these, attempts to identify an object or the movement of that object based on the changes in the images. This is a very important part of computer vision since, just as it is with human perception, objects in the field of vision which are moving are more interesting than those which are not. They are more interesting in the sense that a moving object contains more information about the object itself and the environment which it is in than a static object would. Furthermore, static objects can always be examined at a later point in time whereas a moving object may require an immediate response. This is the same reason for which human vision also focuses more on moving objects [6].

Generally information about the object or its motion can be used to build models which can be used in the process of identifying the motion or the object. The idea is that if one is able to track a moving object then this implies that they have been able to achieve recognition of the object. Motion-based recognition can be thought of as consisting of two parts. In the first part the images must be used to determine what kind of representation will be used for the objects or motion. This forms part of the creation of a model for the motion or for the object which we want to track. Once this is done we then, in the second step, need to match an input from the images with this model. Objects with input which doesn't match the model will then not be recognized [6].

Various methods for tracking motion are available. There is usually some kind of background subtraction used [20]. Some notable methods are gray level background subtraction [15], modelling colour variations in pixels [30], Gaussian mixture background subtraction [25] and the Kalman filter.

2.5 Localization and Mapping

In [21] Olson focuses on one of the main problems in robotics, namely, localization. The robot must be able to determine where it is in the environment accurately, if it is to operate effectively in that environment. Using maximum likelihood, a map generated by the robot in its current position (its local map) is compared to one generated beforehand (the global map) in an attempt to maximize the degree of agreement between the two. This method of using maximum likelihood results in a likelihood surface over the possible positions of the robot which can be used to derive a probability distribution for use in Markov localization. Roumeliotis, in [23], deals with the problem of localization where Bayesian hypothesis testing and Kalman filtering are combined so that both Markov localization and pose tracking can be used in one localization algorithm. The robot tracks its pose continuously through different areas while also monitoring landmarks on the map. This approach overcomes the problem of the Kalman filter not being able to represent multi-modal distributions.

The problem of simultaneous localization of a robot in its environment and building of a map of that environment (SLAM) is well known. It deals with a robot which can start in an unknown position, in an unknown environment and construct a map of the unknown environment as it moves through that environment, while simultaneously finding its own location on the map. In [8], Dissanayake et al. prove that this problem can be solved and that the estimated map will converge monotonically to a relative map. The uncertainties also converge to zero. It shows that the lower bound to the absolute accuracy of the map and robot location is the initial uncertainty of the robots position. The problem is that as the number of objects in the environment increases, the computational complexity becomes very large. Two ways of dealing with this complexity are presented in [7] and [**?**] by using a transformation and bounded approximation respectively.

In [10] Thrun deals with the problem of localization by making use of the robots own actuators (tools that the robot has with which to manipulate its environment) instead of passively determining the robots position. That is, assuming that the robot cannot move or effect its environment during localization. Markov localization provides a criteria for setting the robot's direction of motion as well as determining which direction its sensors are pointing for efficient localization. This is especially effective when the environment contains a

relatively low number or landmarks which would unambiguously determine the robots position. Actions are chosen so that they maximize the expected decrease in uncertainty (contribute the least amount of entropy). This entropy calculation however is computationally complex and so presents a problem when the localization is scaled up to larger environments.

Another interesting addition to solving the problem of localization is presented by Roy et al. in [24]. In this paper a method which generates trajectories for the robot to move along takes into account information about the environment and the density of people in the environment. Even the smallest errors in dead reckoning could lead to very large errors over time. It is clear then, for a robot to localize itself it must incorporate information from its environment. Dynamic obstacles also pose a problem for a robot trying to determine where it is.

The approach used here is based on the concept that when there is a lack of ability to accurately position themselves, ships sail close to the coast, allowing them to determine their position more accurately. Here coastal lines are generated so that they contain just enough information about the environment along their path to allow for accurate localization. The probability of getting lost is minimized by traveling along the path with the highest information density in the environment. This requires a map of information to be generated and then later a path trajectory to be planned.

3 Background Theory

3.1 Basic terminology

A robot consists of a processing unit which acts as the robot's brain and carries out all computations. In addition to this, a memory chip is added to allow it to store information. Built around this processor will be several different kinds of sensors to gather information from its environment which are then processed by the processor. If necessary, data gathered from the sensors can be stored in the memory. Examples of sensors would be: cameras, laser or ultrasonic measurement tools and sound recognition devices. There will also be a variety of actuators which the robot can use to interact with and change the environment around it or to move the robot itself, for example, arms, legs and gripping devices. Most robots will also have some means of mobility, be it legs or wheels. Figure 3 is an example of a robot with legs, which we used[3].

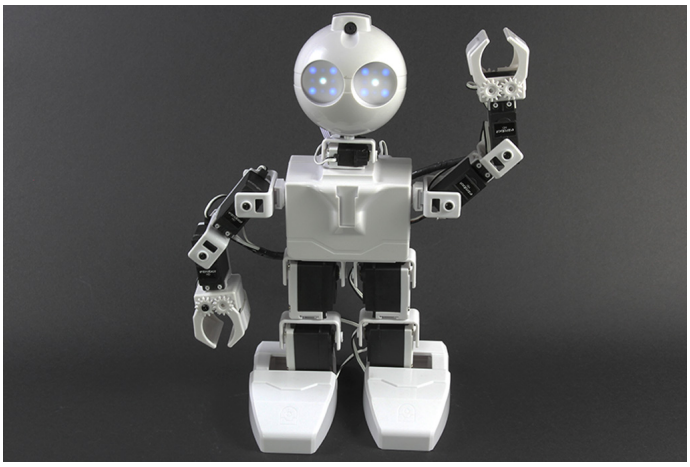


Figure 3: Robot with legs

The state (denote by x_t the state at time t) of a robot is a collection of all aspects of the robot and the environment it is in which could impact its future state. Variables in the state generally include:

[3]Image from: http://www.ez-robot.com/Shop/AccessoriesDetails.aspx?prevCat=101&productNumber=31

- The pose. This is the robot's location and orientation relative to a global map of the environment, this could include three dimensional coordinates as well as angular orientation, velocity and direction of movement.
- Landmarks which are distinct, stationary features of the environment.

Robots interact with their environments in two distinct ways:

1. Getting information about the state from measurements (sensor data)
2. Influencing its environment through its actuators (commands given to the robot to carry out).

The first interaction specifies information about the momentary state of the environment and the second about the change of state in the environment. Generally measurements increase the robots knowledge of its environment. While motion and control inputs decrease its knowledge due to the stochastic manner in which the robot necessarily perceives its environment. Now, x_t is called a complete state if it is the best predictor of the future. This implies knowledge of past states, measures (of the environment though sensors, denoted by z_t) and controls (commands given to the robot at time t, denoted by u_t) so that no additional information would be needed to predict the future state of the robot. It therefore follows that theoretically, the state of the robot at time t is represented by a Markov Chain. However since it is impossible to specify a complete state in practice we must instead deal with a subset of all state variables, called an incomplete state. Theoretically however, we still assume that the properties of a Markov chain are satisfied in some simpler applications.

3.2 State-space models

This section serves to from the foundations for the next section about state estimation. As mentioned above, a dynamic Bayesian network is a Bayesian network which is applied to time series data. One such dynamic Bayesian network is called the hidden Markov model. In these models it is assumed that a past event may affect a future event but that future events do not affect the past. Hence all arks in the graphical representation of this type of Bayesian model always flow forwards in time. Now if we assumed that the sequence of variables $\{Z_1, Z_2, ..., Z_T\}$ form a first order Markov process (where a variable at time t is affected only by the variable at time $t-1$) then, $P(Z_{1:T}) = P(Z_1)P(Z_2|Z_1)...P(Z_T|Z_{T-1})$. This frame work may be adjusted to allow for observations (Z_t) to be dependent on some true state variable (X_t) which is hidden and the sequence of which forms a Markov process. This adjustment creates what is called the hidden Markov model.

Hence, we assume that we have a sequence of observation vectors $\{Z_1, Z_2, ..., Z_T\}$ and that these observations are generated by a hidden variable X_t which denotes the true state of the system. If these state vectors are assumed to form a Markov jump process then we have the following [13].

$$P(X_{1:T}, Z_{1:T}) = P(X_1)P(Y_1|X_1)\prod_{t=2}^{T} P(X_t|X_{t-1})P(Z_t|X_t)$$

This is a specific factorization of the joint probability and so can be represented by a Bayesian network. This specific factorization is identical to that of the hidden Markov model. Here $P(X_t|X_{t-1})$ is called the state-space transition probability and $P(Z_t|X_t)$ is called the observation probability. These imply that the X_t and Z_t can both be broken down into stochastic and non-stochastic parts: $X_t = f(X_{t-1}) + v_t$ and $Z_t = g(X_t) + u_t$. This is an important property to note for later sections on the Kalman filter. It is important because if both X_t and Z_t are linear and time-invariant and if we assume that the relevant noise terms (v_t and u_t) are normally distributed then, the model becomes a linear-Gaussian state-space model. So the model becomes:

$$\begin{aligned} X_t &= f(X_{t-1}) + v_t = AX_{t-1} + v_t \\ Z_t &= g(Z_{t-1}) + u_t = CX_t + u_t \end{aligned}$$

for some matrices A and C. Where, using the appropriate terminology, A is the state transition matrix and C is the transformation matrix. If we allow for inputs (U_t) into the system then we have $X_t = AX_{t-1}BU_t$ (if we again assume linearity and Gaussian errors) where B is the input matrix. The assumptions made are not trivial and we will return to them at a later point.

A very big drawback of hidden Markov models is that, in some cases they are required to model all possible states of a number of objects of interest. All these objects will also have state vectors which may be large. This results in a large amount of computational strain [28][13].

3.2.1 Bayesian inference

The *a priori* information required for inference in the framework described above will be in the form of a prior probability distribution of the models structure and parameters (all arcs in the model prior to the current time). New data coming in is then used to update the model via the likelihood function of the new data to form a posterior distribution.

3.3 State estimation

State estimation is a fundamental and important problem at the core of stochastic robots. It only possible for the robot to know its initial state with certainty (even then this is not always the case) due to noise in the the data as well as the inherent uncertainty within any real world environment. It is, therefore, imperative that a robot be able to stochastically estimate its state and revise this estimation as more data becomes available. If the robot deterministically evaluated its state based on its previous assumed state then it would be extremely vulnerable to any kind of remodeled uncertainty within the environment and after making an error would be unable to correct itself, something which a stochastic robot is capable of. It is also very hard for a deterministic program to react to a situation where the sequence of events that it expects to happen don't occur as it assumes they do.

Recursive state estimation is the act of estimating the state of a robot from a sequence of observations from its sensor data. However, due to technological and economic constrains a robot's sensors will generally only be able to obtain parts of the information needed and this information will inherently contain a degree of background noise.

The state $x_t$is generated stochastically from the previous state x_{t-1}. It follows that x_{t-1} must specify the probability distribution from which x_t is generated. There are two important probabilities which need to be defined:

- The state transition probabilities (how environmental states change over time as a function of controls) given by: $p(x_t|x_{0:t-1},\, z_{1,t-1},\, u_{1:t})$ and
- The measurement probabilities (probabilistic law used to generate measurements from the environment) given by $p(z_t|x_{0:t},\, z_{1,t-1},\, u_{1:t})$

where the subscript $a : b$ represents the collection of all variables to which the subscript is attached from time a to time b.

In this case we will assume that the robot's state forms a Markov process. Since we assume a Markov process, x_{t-1}will be a complete and sufficient statistic for all that happens up to time $t-1$ (including all previous controls and measurements). Hence x_{t-1}will contain the information from $x_{0:t-1}, z_{1,t-1}$ and $u_{1:t-1}$ but not from u_t since the state at time t exists independently of any control given to the robot at that same time. Similarly, $x_t$will contain the information from $x_{0:t}, z_{1,t}$and $u_{1:t}$Using this information, the probabilities above can be simplified to:

$$p(x_t|x_{0:t-1}, z_{1,t-1}, u_{1:t}) = p(x_t|x_{t-1}, u_t)$$

and

$$p(z_t|x_{0:t},\, z_{1,t-1},\, u_{1:t}) = p(z_t|x_t).$$

This setup, where the state at time t is determined in part by the control at time t and the measurement at time t depends on the state, is an example of a dynamic Bayesian network. Specifically, it is a hidden Markov model. This dynamic Bayes network is illustrated graphically in Figure 4.

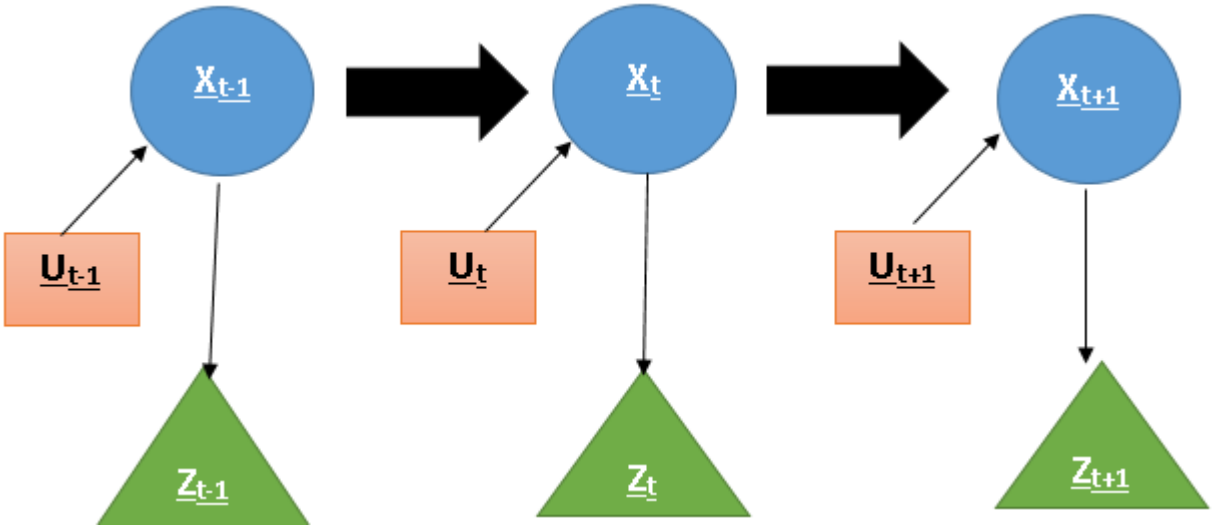


Figure 4: Dynamic Bayes network where x_t, u_t and z_t are the state, control and measurement at time t

This leads onto the concept of belief distributions. A belief is defined as the internal knowledge the robot has of the state of its environment. A robot's belief can never reflect the true reality since its beliefs are inferred from its sensors (the true state can never be measured directly). Belief distributions give the probability distributions of each of its possible beliefs about its state with respect to the state which it is actually in. The belief is given by $bel(x_t) = p(x_t|\, z_{1:t}, u_{1:t})$, the probability distribution of the state given all past control and measurements. Note that this works on the assumption that the measurement at time t, z_t, is known. However, and after control u_t but before we have measured z_t, the probability distribution represents a prediction. This predicted belief is given by: $bel^*(x_t) = p(x_t|z_{1:t-1}, u_{1:t})$. The idea is for the robot to start with a prediction it generates of what it thinks the outcome of its control will be on the environment and once the control is carried out and the new measurement taken, this prediction is updated in what is called the measurement update step to become the belief distribution at time t [27][28].

3.3.1 Bayes Filter Algorithm

All of the ground work thus far brings us to a cornerstone of probabilistic programming; Bayes Filters. It is a recursive algorithm for the calculation of beliefs. As such some form or approximation of a Bayesian Filter is generally present in a probabilistic robot. The general skeleton of the algorithm is given as follows:

$$\text{Bayes\_filter\_algorithm}(bel(x_{t-1}, u_t, z_t)):$$

$$\begin{aligned}
&\text{for all } x_t \text{ do} \\
&\quad bel^*(x_t) \quad = \quad \int P(x_t|u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1} \\
&\quad bel(x_t) \quad = \quad \eta\, P(z_t|x_t)\, bel^*(x_t) \\
&\text{end for} \\
&\text{return} \quad bel(x_t)
\end{aligned}$$

where $\eta = \frac{1}{P(z_t|z_{1:t-1}, u_{1:t})}$.

This is as a result of Bayes rule which states that: $P(x|y,z) = \frac{P(y|x,y)P(x|z)}{P(y|z)}$. Applying this to our problem, it is clear that

$$\begin{aligned} bel(x_t) = P(x_t|z_{1:t}, u_{1:t}) &= \frac{P(z_t|x_t)P(x_t|z_{1:t-1}, u_{1:t})}{P(z_t|z_{1:t-1}, u_{1:t})} \\ bel(x_t) &= \eta.P(z_t|x_t).bel^*(x_t). \end{aligned}$$

It follows from this that what this algorithm effectively does is first, to predict the distribution of the state ($bel^*(x_t)$) at time t, before any measurement is taken. This is done using conditional probabilities. The second step is to include the measurement, z_t, at time t once it has bee taken to find a distribution for the actual belief, $bel(x_t)$. This is done by treating $bel(x_t)$ as the posterior distribution of the prior distribution, $P(z_t|z_{1:t-1}, u_{1:t}) = P(z_t|x_t)$ and the likelihood function, $bel^*(x_t) = p(x_t|z_{1:t-1,}u_{1:t})$. Bayes rule is then used to find this posterior, giving the distribution of the robots state at time t. This is then repeated for all states.

For this algorithm to be used an initial state x_0 is required:

- If this initial state is known with probability 1 then $bel(x_0)$ is initialized with a distribution assigning a probability of 1 to the value of $x_0$and assigning a probability of 0 elsewhere.
- If the initial state is unknown then a uniform distribution over the domain of x_0, is assumed.

While this algorithm is a good starting point, in practice, due to the complexity of computing the integral to obtain a posterior distribution it generally has to be approximated. The Bayes filter has been found to be quite robust to approximation, however the affects of approximation are not well understood as is the case with the effects of violating the Markov assumption and using incomplete states [27]. A few such approximations of the Bayes Filter are the Kalman Filter and it variants.

3.4 The Kalman Filter Algorithm

This topic has already been touched on in the preceding sections. We now define this concept directly and in full detail since it forms the basis for much of the rest of this paper. The Kalman filter is a linear Gaussian filter, meaning that the posterior created is in the form of a Gaussian distribution. Beliefs are represented by multivariate normal distributions and so results in a uni-modal distribution with a single maximum at the true state and, for a good approximation, a small spread. Gaussian filters however are bad for global estimation problems since these problems comprise many distinct hypotheses and each of these forms its own mode in the posterior distribution making computation and interpretation quite difficult. These can be parametrized via moments parametrization using the standard $\underline{\mu}$ (mean) and Σ (covariance matrix) or via conical parametrization. The Kalman filter is numerically efficient, minimizes mean square error in parameters when noise can be assumed to be normally distributed. In the cases where normality can't be assumed it gives the best linear estimate, if the mean and variance of the noise are known. A Bayes filter posterior is Gaussian if, in addition to fulfilling the Markov assumption [28]:

1. The state transition probability must be a linear function of its arguments with added Gaussian noise. A linear Gaussian system is given by: $x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$ where $\varepsilon_t \sim N(0, R_t)$ is an error term (or noise term), B_t and A_t are $n \times n$ and $n \times m$ constant matrices respectively. The matrix A_t is the transition matrix , it applies the effect of each state parameter at time $t-1$ to the state at time t. The matrix B_t is the control input matrix, it applies the effect of each control input on the state at time t[**?**]. So when this x_t is considered as a multivariate normal distribution the result is a posterior distribution with a mean $A_t x_{t-1} + B_t u_t$ and variance R_t

2. The measurement probability also needs to be a linear with Gaussian noise; $z_t = C_t x_t + \delta_t$ where $\delta_t \sim N(0, Q_t)$ is the parameter for measurement noise. Where C_t is a transformation matrix. It transforms the state vector into the domain of the measurements. So the measurement probability becomes:
$$p(z_t|x_t) = |2\pi Q_t|^{-\frac{1}{2}} e^{-\frac{1}{2}(z_t - C_t x_t)' Q_t^{-1} (z_t - C_t x_t)}$$

3. The internal belief must be such that $bel(x_0) \sim N(\mu_0, \Sigma_0)$, where

$$bel(x_0) \quad = \quad p(x_0) = |2\pi\Sigma_0|^{-\frac{1}{2}} exp(-\frac{1}{2}(x_0 - \mu_0)'\Sigma_0^{-1}(x_0 - \mu_0)).$$

These three conditions imply that the belief at time t, $bel(x_t)$ is Gaussian at any point in time, t.

The Kalman filter operates by updating assumptions using new data. This relies on Bayes' Rule: $f(\theta) \propto \pi(\theta)L(\theta; x)$. This says that the posterior distribution of the state of the robot will be proportional to the product of the prior distribution of the state and the likelihood of observing the measurements which were observed, given the current state.

The Kalman filter (named after Rudolf E. Kálmán)[9] is used specifically for filtering and prediction in linear Gaussian systems with continuous states. Moments parametrization is used here; at time t the state is represented by the state mean μ_t and the state covariance Σ_t. It is an algorithm that allows exact inference in a linear dynamical system where all latent variables are continuous and all observed variables have a Gaussian distribution[9]. For the one-dimensional case with Gaussian noise it has been shown to be the optimal estimator of the true state vector of the robot. The standard form has two steps; prediction and a measurement update step.

The Kalman filter algorithm is given by:

$$\begin{aligned}
&\text{Algorithm Kalman filter}(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t): \\
&\qquad \mu_t^* \quad = \quad A_t\mu_{t-1} + B_t u_t \\
&\qquad \Sigma_t^* \quad = \quad A_t\Sigma_{t-1}A_t^T + R_t \\
&\qquad \mu_t \quad = \quad \mu_t^* + K_t(z_t - C_t\mu_t^*) \\
&\qquad \Sigma_t \quad = \quad (I - K_t C_t)\Sigma_t^* \\
&\text{return} \quad \mu_t, \Sigma_t
\end{aligned}$$

where $K_t = \frac{\Sigma_t^* C_t}{C_t\Sigma_t^* C_t' + Q_t}$ is the Kalman gain and is the degree to which measurements are incorporated into the new state estimates, it is chosen such that the covariance matrix (Σ_t) is minimised, and $C_t\mu_t$ is the expected value of the measurement at time t. $z_t = C_t x_t + \delta_t$and $\delta_t \sim N(0, Q_t)$. The Kalman filter is also computationally efficient.

One of the largest benefits of the Kalman filter is that it it very robust to uncertainty because of the fact the it can operate in the absence of a measurement and just perform the prediction step to estimate the new state using the most recent information available to it. So the prediction and measurement update can function independently of one another. This is demonstrated by the example in the following section.

The disadvantages of the Kalman filter are a result of the assumptions that it makes. It assumes both linearity in its model as well as normally distributed error terms. When the error terms are not normally distributed and especially in the case where the error increases over time, the Kalman filter begins to stray from the true model. Since linearity is assumed, it means that the Kalman filter can only accurately estimate systems which develop linearly. It is possible to combat this second drawback, to some extent, if measurements can be made at regular intervals where the intervals are very small. This will give a type of local linear approximation to the function.

3.4.1 Kalman filter example

To illustrate the workings of the Kalman filter in one-dimension consider estimating the distance of an object where the object can accelerate or move at a constant velocity. First the transition and input control matrices need to be defined. In order to do this we consider the manner in which position, velocity and acceleration are related.

$$p_t = p_0 + v_0 t + \tfrac{1}{2}at^2$$

$$v_t = v_0 + at$$

where p_t is displacement at time t, v_t is the velocity and a_t is acceleration at time t. In this example, however, we consider a constant rate of acceleration to preserve linearity and so $a_t = a$. Here t is the amount of time past since the first observations but we assume a Markov chain and so need only estimate these values using the ones preceding them directly. Modifying the equations above:

$$p_t = p_{t-1} + v_{t-1}dt + \tfrac{1}{2}adt^2$$

$$v_t = v_{t-1} + adt$$

Here dt represents the amount of time past between the values at time t and those at the previous time step, $t-1$. Here we assume uniform intervals of measurement, half a second apart each time for 200 seconds so $dt = 0.5$. So if the state consists of position and velocity then they will be updated as follows:

$$\begin{bmatrix} p_t \\ v_t \end{bmatrix} = \begin{bmatrix} 1 & dt \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{t-1} \\ v_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}dt^2 \\ t \end{bmatrix} a$$

$$\begin{bmatrix} p_t \\ v_t \end{bmatrix} = A_t \begin{bmatrix} p_{t-1} \\ v_{t-1} \end{bmatrix} + B_t a.$$

This gives us the process:$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$ with an added error term ε_t. As an initial guess here we will assume that the object being tracked will have started at a distance of 0 and have velocity 0 too, hence $\begin{bmatrix} x_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. We, furthermore, assume that process noise is 0.09^2, $Q_t = 36$ and $a = u_t = 2$.

So we then if we assume that all variability in the process is attributed to its acceleration, we have that:

$$R_t = 0.09^2 \begin{bmatrix} \frac{dt^4}{4} & \frac{dt^3}{2} \\ \frac{dt^3}{2} & dt^2 \end{bmatrix}, \text{ since:}$$

$$var(p_t) = var(p_{t-1} + v_{t-1}dt + \tfrac{dt^2}{2}a_t) = \tfrac{dt^4}{4}var(a_t) = 0.09^2\tfrac{t^4}{4}$$

$$var(v_t) = var(v_{t-1} + a_t dt) = dt^2 var(a_t) = dt^2 0.09^2$$

$$cov(p_{t-1} + v_{t-1}dt + \tfrac{dt^2}{2}a_t, v_{t-1} + a_t dt) = \tfrac{dt^3}{2}0.09^2$$

Now applying the Kalman filter to this problem:

Prediction step:

$$\begin{aligned} \mu_t^* &= A_t\mu_{t-1} + B_t u_t \\ \Sigma_t^* &= A_t\Sigma_{t-1}A_t^T + R_t \end{aligned}$$

So for the first time step:

$$\begin{aligned} \mu_1^* &= A_1\mu_0 + B_1 u_1 \\ \Sigma_1^* &= A_1\Sigma_0 A_1^T + R_1 \end{aligned}$$

$$\begin{aligned} \mu_1^* &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \\ 1 \end{bmatrix} 2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \\ \Sigma_1^* &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0.5 & 1 \end{bmatrix} + 0.09^2 \begin{bmatrix} 0.015625 & 0.0625 \\ 0.0625 & 0.25 \end{bmatrix} = \begin{bmatrix} 0.001265625 & 0.00050625 \\ 0.00050625 & 0.002025 \end{bmatrix} \end{aligned}$$

Now, the measurement update step:

Assume that the first measurement the robot takes tells it that $z_1 = 11.9565$ is the position of the object being tracked and that the robot can only measure distance, not velocity or acceleration. Since this is the case, it also follows that, the transformation matrix is given by $C_t = \begin{bmatrix} 1 & 0 \end{bmatrix}$ because only the position component of the state vector will be affected by the measurements and the same units of measurement are assumed to have been used in both cases. Then we have:

$$
\begin{aligned}
K_1 &= \frac{\Sigma_1^* C_1^T}{C_1 \Sigma_1^* C_1^T + Q_1} = \begin{bmatrix} 0.001265625 & 0.00050625 \\ 0.00050625 & 0.002025 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} (\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0.001265625 & 0.00050625 \\ 0.00050625 & 0.002025 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 36)^{-1} \\
&= \begin{bmatrix} 0.001265625 \\ 0.00050625 \end{bmatrix} 0.0277768 \\
&= \begin{bmatrix} 0.000035155 \\ 0.000014062 \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\mu_t &= \mu_1^* + K_t(z_t - C_t \mu_1^*) \\
\Sigma_t &= (I - K_t C_t) \Sigma_t^*
\end{aligned}
$$

$$
\begin{aligned}
\mu_1 &= \mu_1^* + K_1(z_1 - C_1 \mu_1^*) \\
\Sigma_1 &= (I - K_1 C_1) \Sigma_1^*
\end{aligned}
$$

$$
\begin{aligned}
\mu_1 &= \begin{bmatrix} 1 \\ 2 \end{bmatrix} + \begin{bmatrix} 0.000035155 \\ 0.000014062 \end{bmatrix} (11.9565 - \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix}) = \begin{bmatrix} 1.000385176 \\ 2.00015407 \end{bmatrix} \\
\Sigma_1 &= (\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.000035155 \\ 0.000014062 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}) \begin{bmatrix} 0.001265625 & 0.00050625 \\ 0.00050625 & 0.002025 \end{bmatrix} = \begin{bmatrix} 0.0012656 & 0.0005062 \\ 0.0005062 & 0.002025 \end{bmatrix}
\end{aligned}
$$

This process is then repeated continually and although the filter starts fairly far from the correct position, it corrects itself very quickly as shown by the graph of a simulation of this example in Figure 5.

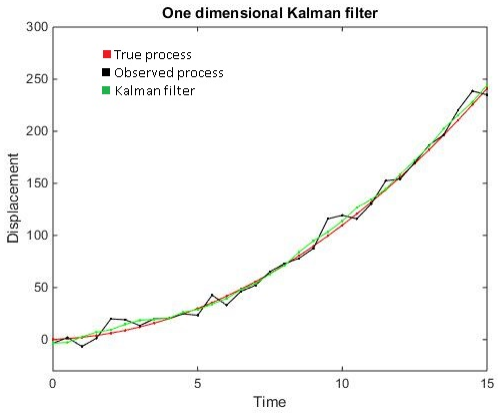


Figure 5: Graph of the one dimensional Kalman filter from MATLAB for the illustration example discussed

Here the actual displacement of the object is represented by the red line, the black line connects the noise observations of distance that the robot receives and the green represents the Kalman filter estimates. The code for generating this graph can be found in the appendix.

To demonstrate how Bayes' Rule is used here refer to Figure 6, where the normal distributions of the observed and predicted values are combined to form a posterior distribution for the objects position. In Figure 6 a graphical representation of the update step is given. The predicted distribution of the state x_t is first found at time t. After that an observation is obtained at time t. This observation has a distribution given by its likelihood function of occurring. The posterior distribution for the state is then found by applying Bayes rule. Here the posterior distribution is seen to have much more variability than the predicted distribution.

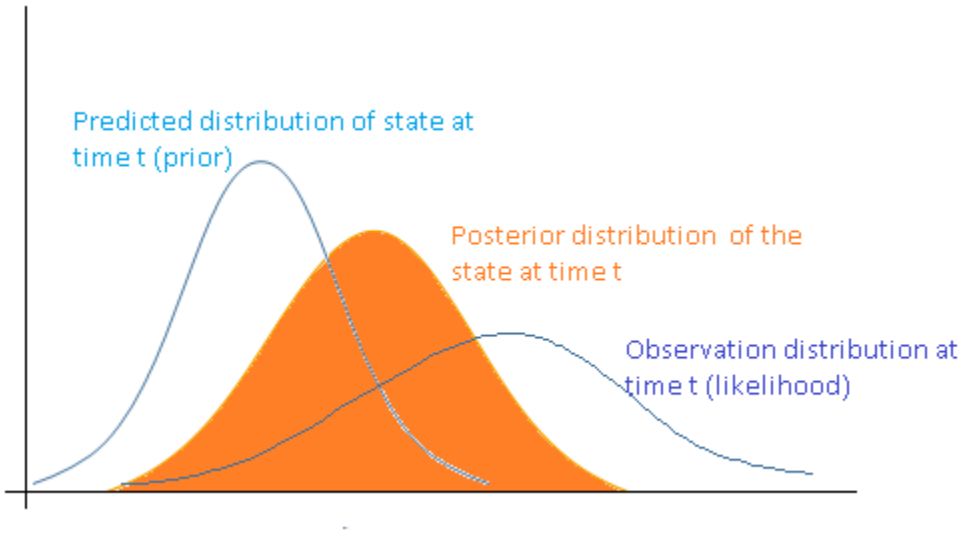


Figure 6: Bayes' rule illustrated in Kalman filter

This is usually the case since, there is generally a larger error in measurement than error in the process. In the amount by which the prediction is weighted against the likelihood when the posterior is formed is determined by the Kalman gain (K_t). It is a function of both the process and measurement covariances since these determine the amount of trust which can be placed in the predicted distribution and likelihood respectively. This code is found in Appendix 5. It was edited and adapted from Student Dave's Tutorials[4].

3.4.2 How the updated distribution is found

The statistics behind these normal plots are as follows.

- Prediction:

$bel^*(x_t) = \int p(x_t|x_{t-1},u_t).bel(x_{t-1})dx_{t-1}$where $p(x_t|x_{t-1},u_t) \sim N(x_t; A_t x_{t-1} + B_t u_t, R_t)$ and $bel(x_{t-1}) \sim N(x_{t-1}; \mu_{t-1}, \Sigma_{t-1})$

$$\begin{aligned} bel^*(x_t) &= \eta \int e^{-\frac{1}{2}(x_t - A_t x_{t-1} - B_t u_t)' R_t^{-1} (x_t - A_t x_{t-1} - B_t u_t)} e^{-\frac{1}{2}(x_{t-1} - \mu_{t-1})' R_t^{-1} (x_{t-1} - \mu_{t-1})} dx_{t-1} \\ \text{Let } L_t &= \frac{1}{2}(x_t - A_t x_{t-1} - B_t u_t)' R_t^{-1} (x_t - A_t x_{t-1} - B_t u_t) + \frac{1}{2}(x_{t-1} - \mu_{t-1})' R_t^{-1} (x_{t-1} - \mu_{t-1}) \\ \text{Then } bel^*(x_t) &= \eta \int e^{-L_t} dx_{t-1} \end{aligned}$$

Now setting the first derivative of $L_t = 0$ gives us the mean:

$$x_{t-1} = \Psi_t(\quad A_t' R_t^{-1} A_t (x_{t-1} - B_t u_t) + \Sigma_{t-1}^{-1} \mu_{t-1})$$

Now $det(2\pi\Psi)^{-\frac{1}{2}} e^{-L_t(x_{t-1}, x_t)}$ is a valid PDF and that we must then therefore have that:

$$\begin{aligned} \int_{-\infty}^{\infty} det(2\pi\Psi)^{-\frac{1}{2}} e^{-L_t(x_{t-1}, x_t)} dx_{t-1} &= 1 \\ \int_{-\infty}^{\infty} e^{-L_t(x_{t-1}, x_t)} dx_{t-1} &= det(2\pi\Psi)^{\frac{1}{2}} \end{aligned}$$

[4] http://studentdavestutorials.weebly.com/kalman-filter-with-matlab-code.html

Hence the value of the integral is independent of the value of x_t. This implies that the integral is constant and so can be absorbed into the constant η to create a new constant function η^*. Then we get that:

$$bel^*(x_t) \quad = \quad \eta^* e^{-L_t(x_t)}$$

Now:

$$\begin{aligned} L_t(x_t) \quad = \quad & \frac{1}{2}(x_t - B_t u_t)' R_t^{-1}(x_t - B_t u_t) + \frac{1}{2}\mu'_{t-1}\Sigma_{t-1}^{-1}\mu_{t-1} - \frac{1}{2}[A'_t R_t^{-1}(x_t - B_t u_t) \\ & + \Sigma_{t-1}^{-1}\mu_{t-1}]'(A'_t R_t^{-1} A_t + \Sigma_{t-1}^{-1})^{-1}[A'_t R_t^{-1}(x_t - B_t u_t) + \Sigma_{t-1}^{-1}\mu_{t-1}] \end{aligned}$$

From this we also see that $L_t(x_t)$ is quadratic in x_t. This implies that $bel^*(x_t)$ is normally distributed. Setting $\frac{\partial L_t(x_t)}{\partial x_t} = 0$ to obtain the minimum and solve for the mean:

$$x_t \quad = \quad B_t u_t + A_t \mu_{t-1} \text{ is the mean of } bel^*(x_t)$$

Now:
$\frac{\partial^2 L_t(x_t)}{\partial x_t^2} = (A_t \Sigma_{t-1}^{-1} A'_t + R_t)^{-1}$gives the curvature of $L_t(x_t)$ and so the inverse of this must give us the covariance of $bel^*(x_t)$. We now have the mean and covariance of the normal distribution obtained from the prediction step:

$$\begin{aligned} \mu_t^* \quad &= \quad B_t u_t + A_t \mu_{t-1} \\ \Sigma_t^* \quad &= \quad A_t \Sigma_{t-1}^{-1} A'_t + R_t \end{aligned}$$

- Measurement update:

$$bel(x_t) \quad = \quad \eta p(z_t|x_t) bel^*(x_t)$$

where $p(z_t|x_t) \sim N(z_t; C_t x_t, Q_t)$ and $bel^*(x_t) \sim N(x_t; \mu_t^*, \Sigma_t^*)$. Then: $bel(x_t) = \eta e^{\frac{1}{2}(z_t - C_t x_t)' Q_t^{-1}(z_t - C_t x_t) + \frac{1}{2}(x_t - \mu_t^*)' \Sigma_t^{*-1}(x_t - \mu_t^*}$
The exponent is a function which is quadratic in x_t, say J_t. This implies that $bel(x_t)$ is a Gaussian function. To get the parameters of this new normal distribution we now follow the same steps as above.
$\frac{\partial J}{\partial x_t} = -C'_t Q_t^{-1}(z_t - C_t x_t) + \Sigma_t^{*-1}(x_t - \mu_t^*)$, minimising this function will result in the mean of $bel(x_t)$:

$$\begin{aligned} \text{Let } K_t \quad &= \quad C_t Q_t^{-1}(C'_t Q_t^{-1} C_t + \Sigma_t^{*-1})^{-1} \\ \text{Hence} \mu_t \quad &= \quad \mu_t^* + K_t(z_t - C_t \mu_t^*) \end{aligned}$$

$$\begin{aligned} \Sigma_t \quad &= \quad (C'_t Q_t^{-1} C_t + \Sigma_t^{*-1})^{-1} \\ \text{Furthermore } K_t \quad &= \quad \Sigma_t C'_t Q_t^{-1} \\ &= \quad \Sigma_t C'_t Q_t^{-1}(C_t \Sigma_t^* C'_t + Q_t)(C_t \Sigma_t^* C'_t + Q_t)^{-1} \\ &= \quad \Sigma_t^* C'_t (C_t \Sigma_t^* C'_t + Q_t)^{-1} \\ \text{From this definition: } \Sigma_t \quad &= \quad (C'_t Q_t^{-1} C_t + \Sigma_t^{*-1})^{-1} \\ &= \quad [I - K_t C_t]\Sigma_t^* \end{aligned}$$

3.4.3 Additional notes on the Kalman filter

It must be noted that because the Kalman filter relies on the assumption of a Markov process, it is susceptible to small localized variations in the process as it is observed. For example, suppose at present we observe an extreme measurement which should come from the one of the tails of the true distribution of the process. This extreme observation will still cause the Kalman filter to momentarily shift in its own direction away from the true process. However, provided that the following observations are not also too extreme, the Kalman filter will correct itself just as quickly.

It is also important to note that in the case where the processes assumed variance is much larger than the variance of the observations that this causes the Kalman gain to be very close to 1 since $K_t = \frac{\Sigma_t^* C_t}{C_t \Sigma_t^* C_t' + Q_t}$. This is especially the case where the process variance increases over time. The consequence in a situation such as this is that the Kalman filter will tend to follow the variance in the observable process and give less weight to initial internal model that it is given about how to process should progress. In some cases this is desirable as that large amounts of variation from the initial idea could be indicative of have made the wrong assumptions initially.

3.5 Gaussian Mixture Model Background Subtraction

Another method for tracking moving objects is to use Gaussian mixture model [25]. This done by using a fixed number, K, of Gaussian distributions. Each uniform object or surface is represented by one of $k = \{1, 2, ...K\}$states where K is an assumed constant number of surfaces. Some states are part of the background of the image and others the foreground. Here the background is made up of all static objects in the image frames of the video and the foreground will be made up of objects which are moving. In some motorway surveillance systems three Gaussian distributions are used; one to model the cars, one for their shadows and one for the road [3].This model has two parameters α and T. α is a learning constant and T is the fraction of the data which should be explained by the background. The rate of adaption of the model is controlled by a global parameter $\alpha \in (0, 1)$. Small values of α result in a model which takes too long to converge and large values result in the model being too sensitive. As each new frame arrives the parameters of the Gaussian distributions are updated and are then evaluated to determine which are part of the background. It works as follows.

Each pixel is characterized using the RGB colour space and then the probability of observing those particular values is calculated as: $P(X_t) = \sum_{i=1}^{K} \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$ where X_t is the pixel value and $\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu_{i,t})' \Sigma_{i,t}^{-1} (X_t - \mu_{i,t})}$ is a Gaussian probability density function with mean $\mu_{i,t}$ and standard deviation $\Sigma_{i,t}$. It is assumed that the colour components are independent and have equal variances. So $\Sigma_{i,t} = \sigma_{i,t}^2 I$. K is the number of distributions, $\omega_{i,t}$ is the weight of the i^{th} Gaussian at time t with mean $\mu_{i,t}$ and standard deviation $\Sigma_{i,t}$. Hence the probability of each pixel value is made up of the mixture of K Gaussian distributions.K is determined by computational restrictions as well as the modality required to model the background. The weights, means and covariances can be initialized using expectation-maximization algorithms [3].

When initialization is done, the first foreground detection is carried out. The parameters of the K Gaussian distributions are then updated. In [25], the ratio $r_j = \frac{\omega_j}{\sigma_j}$ is used as a criterion for ordering the K Gaussians. The idea being that a background pixel will have a higher weight but lower variance than a moving object since it should be more persistent in the image. The first B Gaussians exceeding a certain threshold value, T, are then taken to be part of the background distribution where: $B = argmin_b(\sum_{i=1}^{b} \omega_{i,t} < T)$. The remaining distributions are assumed to be part of the foreground.

The next frame is received at time $t+1$ and a match test is conducted on each pixel. A pixel is assumed to match a certain Gaussian distribution if the Mahalanobis distance: $\sqrt{(X_{t+1} - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_{t+1} - \mu_{i,t})} < k\sigma_{i,t}$ where k is a constant threshold which is generally taken to be 2.5.

Now either a match will be found with one the K Gaussians or there will not be a match. If the pixel matches with one of the K Gaussians which happens to be a background Gaussian then, the pixel is classified as being in the background otherwise, it is in the foreground. If no match is found the pixel is also classified as part of the foreground. We now have what is called a binary mask. The parameters then need to be

updated before the next foreground detection is performed. For components that match the update is done as follows.

$$\begin{aligned}
\omega_{i,t+1} &= (1-\alpha)\omega_{i,t} + \alpha \\
\mu_{i,t+1} &= (1-\rho)\mu_{i,t} + \rho X_{t+1} \\
\sigma^2_{i,t+1} &= (1-\rho)\sigma^2_{i,t} + \rho(X_{t+1} - \mu_{i,t+1})(X_{t+1} - \mu_{i,t+1})^T \\
\rho &= \alpha\eta(X_{t+1}, \mu_i, \Sigma_i)
\end{aligned}$$

For unmatched components the updates of mean and variance are the same but the weights are updated by $\omega_{j,t+1} = (1-\alpha)\omega_{j,t}$.

If no components match with any of the K Gaussian distributions then the least probable distribution k is replaced by a new one with parameters:

$$\begin{aligned}
\omega_{i,t+1} &= \text{some low prior weight} \\
\mu_{i,t+1} &= X_{t+1} \\
\sigma^2_{i,t} &= \text{some large initial variance}
\end{aligned}$$

This process is then repeated. This process is good at tracking, however it is not robust to occlusion or missing values and so is out performed by the Kalman filter in such situations.

3.6 The Location Problem

Another fundamental problem in the world of robotics is the location problem: locating the robot relative to a map of the area it which it is operating. This may be regarded as a coordinate transformation problem. One where the global coordinate system must be translated to the internal coordinate system of the robot [28]. However the catch is that the robots pose can't be sensed directly and so must be inferred from noisy sensor data. There are various levels to this problem and they can be broken up into the following categories:

Local versus global localization: The type of information available initially and at run time define the type of localization problem. The availability and type of information distinguishes three different localization problems. These in order of increasing difficulty are :

1. Position tracking : The initial pose is assumed to be known. Localization is accomplished through accounting for the noise created by the motion of the robot. The effect of this noise is usually very small. This is classified as a local problem since the uncertainty which needs to be dealt with is "local" in that it is confined to a relatively small region near the robot's pose.

2. Global localization: Here the initial pose of the robot is unknown. So the robot may be placed in a familiar environment but not be told where in that environment it is. Here the amount of error cannot be assumed to be small since it is, in-fact, unbounded if the robot has no idea where it is at all. The robot does however know that it doesn't know where in the environment it is (it knows that there is a very large amount of uncertainty in its estimation of its position).

3. Kidnapped robot problem: Here a robot is assumed to be "kidnapped" while operating and replaced in a different location in the same environment. One of the main difficulties here is that the robot may in-fact think that it knows where it is when it actually doesn't.

Static versus dynamic environments: In a dynamic environment, objects other that the robot have the ability to change their location or configuration over time. In a static environment the only variable quantity is the robot's pose. So the only object that can move is the robot itself.

Passive versus active approaches: In a passive approach the localization module in the robot only observes the environment and the robot's motion is not aimed at helping the robot localize itself. Active

algorithms control the robot and its motion to minimize localization error.

In localization a map is defined as a list of objects, $m = \{m_1, ..., m_N\}$, in the environment and their corresponding properties. These objects are points of interest in the map (landmarks) such as obstacles, walls and people. Two important ways to go about mapping an environment are:
Location-based maps: This is a volumetric approach, it gives a label to any location in the environment. Hence they contain information about objects in the environment as well as about the absence of objects in the environment.
Feature-based maps: These only specify the shape of the environment at specific locations (where objects are located).

3.6.1 Markov Localization Algorithm

This algorithm is a variant of the basic Bayes filter where the map (m) is incorporated into the state transition (motion) model as well as the measurement model. These become $p(x_t|x_{t-1}, u_t, m)$ and $p(z_t|x_t, m)$ respectively. It is capable of addressing the global localization problem as well as the kidnapped robot problem in a static environment.

4 Application

4.1 Colour Identification with EZ - Robot

Using the EZ-Robot interface and one of the robots, the Revolution JD, colour identification was explored. The built-in Multi-Color Tracking algorithm is activated and used to make the robot point in the direction of a specified colour when it is seen. This can be done using the graphical user interface supplied by the EZ-Builder platform or by using the code in Appendix 5. The colour tracking algorithm in the platform works by looping through all the pixels in the image and searching for pixels in the colour range specified in the code[5]. In the EZ-Builder platform the ARGB32 platform is used. Once all the specific coloured pixels are identified they are extracted from the image and the AForge Blob function[6] is used to identify groups of the specific coloured pixels of a certain size or within a specified range of size. The result is a list of locations of where there are chunks of the specific colour in the image. The code used to do this can be found in the appendix.

4.2 Two Dimensional Kalman Filter on Video Frames

4.2.1 Introductory example

As an introduction to this section we first apply the Kalman filter to a very simple scenario in order to demonstrate the underlying principal of the more complex example which follows, involving analysis on video frames. This code is in Appendix 5.

In this model we model the position of an object in two dimensional space. We assume that the object has a constant velocity in both the x and y directions. Only the x and y positions at each time are observed not the corresponding velocities. Hence we have both linearity and the data are generated to have Gaussian error terms.

The model is therefore given by:

[5] http://www.ez-robot.com/Community/Forum/Thread?threadId=7505
[6] http://www.aforgenet.com/framework/features/blobs_processing.html

$$
\begin{aligned}
X_t &= A_t X_{t-1} + \varepsilon_t \\
&= \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} X_{t-1} + \varepsilon_t
\end{aligned}
$$

and

$$
\begin{aligned}
Z_t &= C_t X_t + \delta_t \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} X_t + \delta_t
\end{aligned}
$$

where $X_t = \begin{pmatrix} x_t \\ y_t \\ \dot{x}_t \\ \dot{y}_t \end{pmatrix} = \begin{pmatrix} x \text{ displacement} \\ y \text{ displacement} \\ x \text{ velocity} \\ y \text{ velocity} \end{pmatrix}$ and δ_t and ε_t and Gaussian error terms. There are no control inputs here since we assume a constant velocity and no external forces acting on the object. The exact distribution of the measurement and process error is not important. This is because when the model is implemented, we assume that this information is unknown and that the magnitude of the error must be inferred or found by trail and error to give the best results. However, it is assumed that the covariance matrices for the process and measurements have the following form: $R_t \propto \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ and $Q_t \propto \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. That is, we assume that the displacements and velocities in each direction are independent processes.

In Figure 7 the theoretical, observed and real-world values are plotted on the same axes.

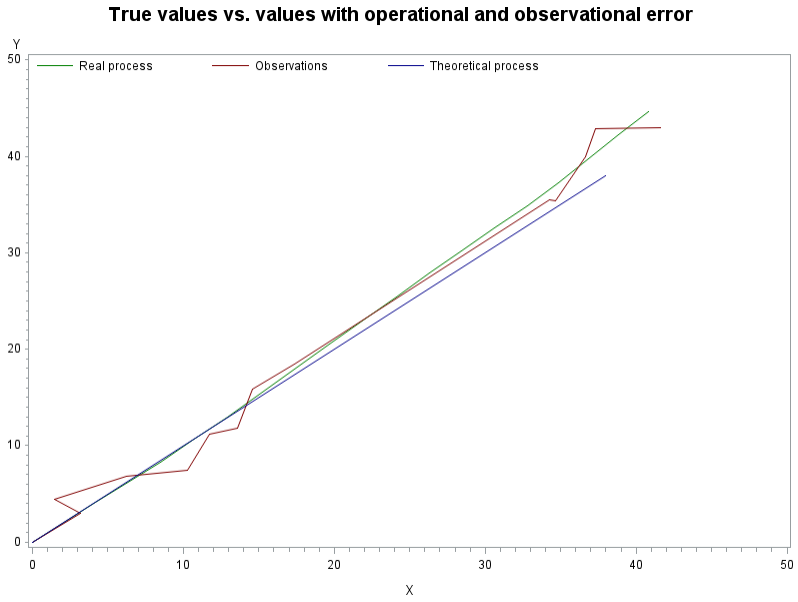


Figure 7: Graph and theoretical, observed and real values of the model in the simple SAS IML example

Applying the Kalman filter to this data, we obtain the graph in Figure 8.

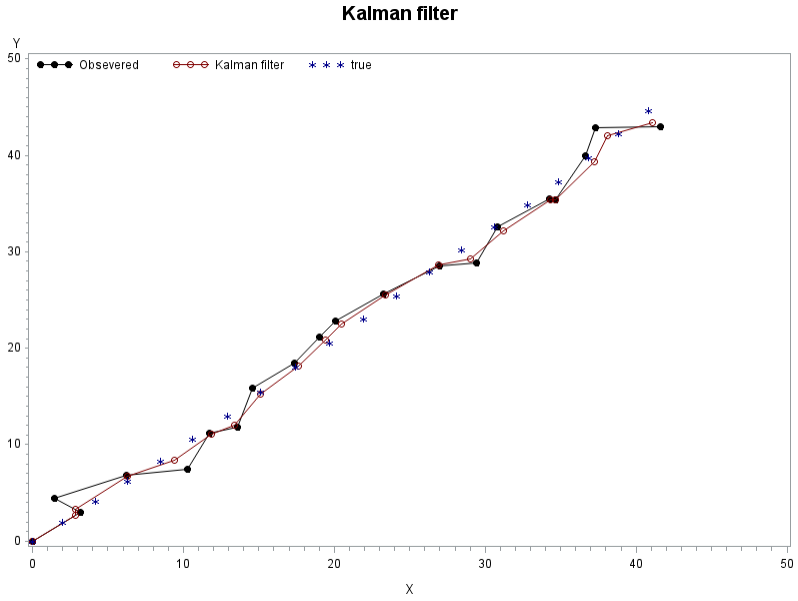


Figure 8: Kalman filter applied to the SAS IML example data

4.2.2 Missing data points

Next we set some of the observed data points to be missing values. Here the 10^{th} to the 15^{th} observations are set to be missing values. Now we applying the Kalman filter again in Figure 9.

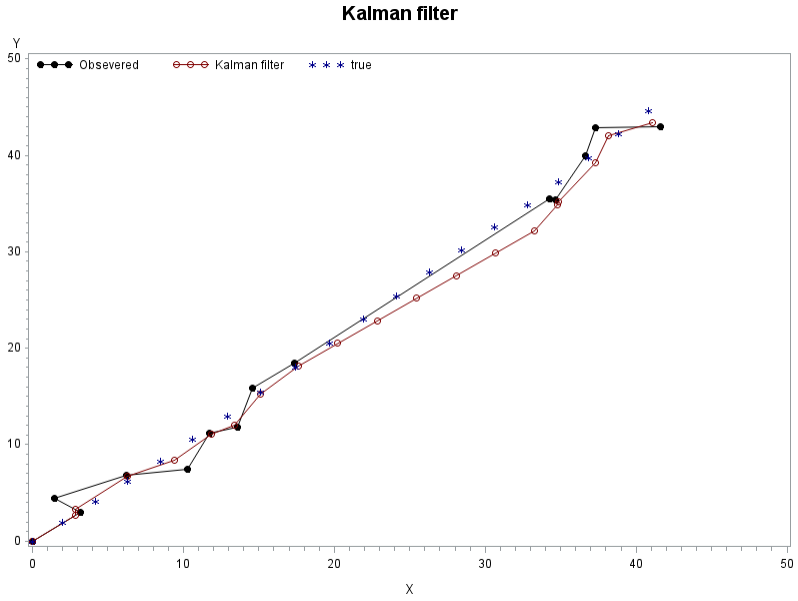


Figure 9: Kalman filter for SAS IML example with missing observations

Here we see that even though there are no longer any observations, the Kalman filter is still able to use its knowledge of the last known observation to keep predicting what will happen until it receives a new observation. Most deterministic programming would simply take the position on the graph as x progresses as being estimated by the previous observed position. This becomes a problem when there is no last observed position or when the last observed position was very long ago.

It is evident here that the large amount of variation in the observable process causes the Kalman filter to adjust when there are a number of observations which seem to display some sort of trend due to random variations in the same direction consecutively. This is the second idea which was discussed in the section titled additional notes on the Kalman filter above.

4.2.3 Application to video frames

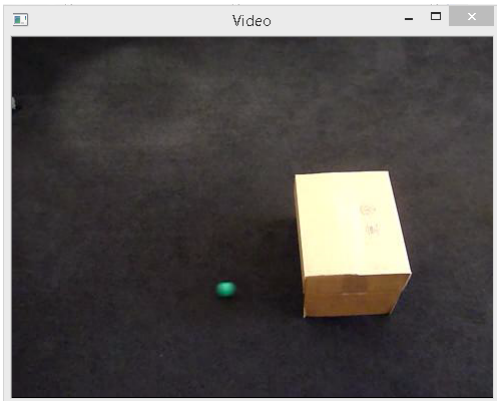


Figure 10: Screen shot of video used

In this section a 2D Kalman filter is applied to a video of a ball being rolled across the screen. The ball passes under a box in the process. Both the Kalman filter and tracking using Gaussian mixture model background subtraction are used to graphically illustrate to the difference in performance. This video was obtained from a MathWorks tutorial[7].

The Gaussian background subtraction first applied to the original video frames. This isolates the movement of the ball since it is the only mobile object in the video. It does this (as explained above) by comparing the pixels in successive frames by assigning each of the a Gaussian probability distribution and then checking if the change in the colour of the pixel between the frames is significant with respect to the distribution assigned to it. The result is that only the pixels in each frame which change significantly are highlighted. When these frames are put back together the result is that we observe the ball move across the screen and vanish when it moves under the box and then reappear again when it rolls out from the other side. These frames are then used to obtain the coordinates of the ball on the screen at certain intervals.

The Kalman filter then takes those coordinates as measurement values and predicts the position of the ball over its whole trajectory. Since it doesn't require measurements to form a prediction however, it is capable of predicting the position of the ball even while it is under the box and fills in the gap left by the background subtraction model.

A screen shot of the video setup is given in Figure 10. The two methods were implemented using Python. The code used here can be found in the appendix. The results are shown in Figure 11. Where the blue dots represent the state with the Kalman filter output and the red crosses are the state measurements outputted by the Gaussian mixture background subtraction method. From this it is apparent that while the background subtraction method only makes use of the immediate information with in the current frame, the Kalman filter is capable of using all information from previous frames too. The Kalman filter uses this previous information to predict the location of the ball when no measurements are found within the data because the ball moves under the box and so cannot be seen by the camera. The code for this example is found in Appendix 5.

This clearly illustrates the utility of the Kalman filter (and indeed Bayesian state estimation methods in general) when applied to situations where there is uncertainty or missing information. They have the ability to draw on past information to make inferences about the future state of a system. However, because this inference is done in a stochastic manner, it means that the system doesn't just estimate a single future

[7] MathWorks tutorial:http://www.mathworks.com/help/vision/examples/using-kalman-filter-for-object-tracking.html

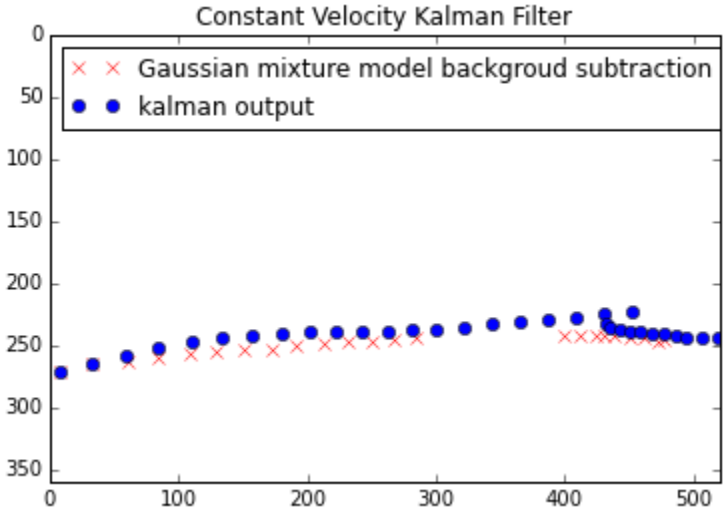


Figure 11: Kalman filter vs. Gaussian mixture background subtraction tacking

outcome but a number of future outcomes. All of these outcomes have a likelihood attached to them and these outcomes and their likelihoods change according to what is actually observed. In this way, all possibilities are accounted for and considered instead of only considering the most likely outcome at each step and discarding the rest. This also allows for more versatility in cases where an unexpected event (the object disappearing under a box) occur.

5 Conclusion

The primary aim of this research was to investigate the world of robotics from a statistical point of view. To that end, some of the main challenges in robotics were examined to supplement the idea that more "intelligent" robotics are made possible using a stochastic approach to how a robot perceives and interacts with its environment. Intelligent, in this case, implying that a robot which is programmed probabilistically is more capable of making adjustments, working in the presence of uncertainty or missing data and to some extent, capable of learning from it experience as compared to a robot which has been programmed with a set of deterministic rules. It was found that Bayesian filters can successfully implemented to process the kind of data and information needed to implement this kind of stochastic approach to robotics. The basic idea is that the robot will have a vector stored in its memory with information about its environment which is pertinent for it to perform its desired function. The bigger this vector, the more accurate the robot will be. However, the obvious problems with lack of memory and computational power which constrain the size of this so-called state vector. The state of the robot is populated with whatever information is available and as the robot operates over time it will gather more data about its environment. A Bayesian filter operates in the background, it assigns a specific probability distribution to the robot's current state and a likelihood to the new incoming data. The probability distribution of the state is treated as a prior distribution and Bayes' rule is applied to this and the likelihood to give a resulting prior distribution for the robot's new state. The idea is that the use of a probability distribution which is continually updated by new information will allow for uncertainty, adaptability and a certain degree of learning from the environment which a robot which is deterministically programmed cannot be capable of.

The focus then shifted specifically to the Kalman filter, one of the most popular and widely used Bayesian filters. The Kalman filter is applied widely for use in the filtering noise in digital images, object tracking and image retrieval [18]. Here its capability was demonstrated by applying the methodology to object tracking in the field of computer vision. It was demonstrated that while a normal Gaussian background subtraction could be used ot track the motion of a ball, when the ball in no longer visible the robot will stop tracking it.

However, when applying a Kalman filter to the data from the background subtraction, the filter is capable of predicting the balls motion even when it cannot be seen. Although this method will be computationally more taxing, the idea was to demonstrate the adaptability and "intelligence" of this approach in that while the ball cannot be seen, the robot will still "know" where the ball is.

The drawbacks of this methodology which have been touched on in the preceding paragraphs are twofold. Firstly, the calculation of the posterior distribution is computationally taxing and in many cases the integrals which need to be solved are prohibitively complicated. This has always been a problem with regards to the implementation of Bayes' rule. The second issue is that of memory; for a robot to run all of its systems using this kind of approach (not just its camera as demonstrated here) will require the robot to store large amounts of information. If the goal of this kind of probabilistic programming is some form of "intelligent" robot then, while robots programmed in this way can certainly be shown to be more intelligent in some ways, for the robot to become increasingly intelligent and aware (of its own state and its surroundings) will require ever increasing amounts of information and processing power. On a more positive note however, memory capacity and computational power limits remain on an upwards trend and only years ago some of the applications discussed here would have been impossible.

In the future it would be interesting to investigate the rate of technological advancement required for an increases level of ability. Closely related to this is the question of exactly what information is relevant to what kind of tasks and if it is possible to determine a base number of variables required to perform these specific tasks to a reasonable degree of accuracy.

References

[1] Jaakko Astola, Petri Haavisto, and Yrjo Neuvo. Vector median filters. *Proceedings of the IEEE*, 78(4):678–689, 1990.

[2] Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 12 1966.

[3] Thierry Bouwmans, Fida El Baf, and Bertrand Vachon. Background modeling using mixture of Gaussians for foreground detection-a survey. *Recent Patents on Computer Science*, 1(3):219–237, 2008.

[4] Anthony R Cassandra, Leslie Pack Kaelbling, and James A Kurien. Acting under uncertainty: discrete Bayesian models for mobile-robot navigation. In *Intelligent Robots and Systems' 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 2, pages 963–972. IEEE, 1996.

[5] Edwin Catmull. A hidden-surface algorithm with anti-aliasing. In *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '78, pages 6–11, New York, NY, USA, 1978. association for computing machinery.

[6] Claudette Cedras and Mubarak Shah. Motion-based recognition a survey. *Image and Vision Computing*, 13(2):129–155, 1995.

[7] Michael Csorba and Hugh F Durrant-Whyte. New approach to map building using relative position estimates. In *Navigation and Control Technologies for Unmanned Systems II*, volume 3087, pages 115–125, 1997.

[8] MWM Gamini Dissanayake, Paul Newman, Steve Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, 2001.

[9] Ramsey Faragher. Understanding the basis of the Kalman filter via a simple and intuitive derivation. *IEEE Signal Processing Magazine*, 29(5):128–132, 2012.

[10] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Active Markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(3):195–207, 1998.

[11] David Gavilan Ruiz, Hiroki Takahashi, and Masayuki Nakajima. Image categorization using color blobs in a mobile environment. In *Computer Graphics Forum*, volume 22, pages 427–432. Wiley Online Library, 2003.

[12] Zoubin Ghahramani. Learning dynamic bayesian networks. In Lee Giles and Marco Gori, editors, *Adaptive Processing of Sequences and Data Structures*, volume 1387 of *Lecture Notes in Computer Science*, pages 168–197. Springer Berlin Heidelberg, 1998.

[13] Zoubin Ghahramani. An introduction to hidden markov models and bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(01):9–42, 2001.

[14] Nasser H Ali and Ghassan M Hassan. Kalman filter tracking. *International Journal of Computer Applications*, 89(9):15–18, 2014.

[15] Ismail Haritaoglu, David Harwood, and Larry S Davis. W 4: Real-time surveillance of people and their activities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):809–830, 2000.

[16] George H Joblove and Donald Greenberg. Color spaces for computer graphics. In *ACM Siggraph Computer Graphics*, volume 12, pages 20–25. association for computing machinery, 1978.

[17] Eun Ryung Lee, Pyeoung Kee Kim, and Hang Joon Kim. Automatic recognition of a car license plate using color image processing. In *Image Processing, 1994.IEEE International Conference*, volume 2, pages 301–305. IEEE, 1994.

[18] Jong-Sen Lee. Refined filtering of image noise using local statistics. *Computer Graphics and Image Processing*, 15(4):380–389, 1981.

[19] David G Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355–395, 1987.

[20] Stephen J McKenna, Sumer Jabri, Zoran Duric, Azriel Rosenfeld, and Harry Wechsler. Tracking groups of people. *Computer Vision and Image Understanding*, 80(1):42–56, 2000.

[21] Clark F Olson. Probabilistic self-localization for mobile robots. *Robotics and Automation, IEEE Transactions on*, 16(1):55–66, 2000.

[22] Thomas Porter and Tom Duff. Compositing digital images. In *ACM Siggraph Computer Graphics*, volume 18, pages 253–259. association for computing machinery, 1984.

[23] Stergios I Roumeliotis and George A Bekey. Bayesian estimation and Kalman filtering: A unified framework for mobile robot localization. In *IEEE International Conference on robotics and automation, 2000.*, volume 3, pages 2985–2992. IEEE, 2000.

[24] Nicholas Roy, Wolfram Burgard, Dieter Fox, and Sebastian Thrun. Coastal navigation-mobile robot navigation with uncertainty in dynamic environments. In *International Conference on robotics and automation, 1999.*, volume 1, pages 35–40. IEEE, 1999.

[25] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE, 1999.

[26] Michael J Swain and Dana H Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[27] Sebastian Thrun. Probabilistic algorithms in robotics. *AI Magazine*, 21(4):93, 2000.

[28] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT press, 2005.

[29] Harry Voorhees and Tomaso Poggio. Computing texture boundaries from images. *Nature*, 333(6171):364–367, 1988.

[30] Christopher Richard Wren, Ali Azarbayejani, Trevor Darrell, and Alex Paul Pentland. Pfinder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):780–785, 1997.

Appendix

SAS IML code for simple Kalman filter example:

```
*Kalman filter in IML; *data generation;
proc iml;
n=20;
y=J(n,1,.); yproc=J(n,1,.);  yobs=J(n,1,.);
y[1,]=0;     yproc[1,]=0; yobs[1,]=0;
x=J(n,1,.); xproc=J(n,1,.);  xobs=J(n,1,.);
x[1,]=0;     xproc[1,]=0; xobs[1,]=0;
Vx=J(n,1,.);    Vxproc=J(n,1,.);
Vx[1,]=2;   Vxproc[1,]=2;
Vy=J(n,1,.);    Vyproc=J(n,1,.);
Vy[1,]=2;   Vyproc[1,]=2;
e1=0; e11=0;
e2=0; e22=0;

do t=2 to n;

call rannor(1,e1);
e1=e1*.09;
call rannor(2,e11);
e11=e11*.09;
call rannor(6748,e2);
e2=e2*1.05;
call rannor(3,e22);
e22=e22*1.05;

*true process;
x[t,]=x[t-1,]+Vx[t-1,];
y[t,]=y[t-1,]+Vy[t-1,];
Vx[t,]=Vx[t-1,];
Vy[t,]=Vy[t-1,];

*Obeservable process;
xproc[t,]=xproc[t-1,]+Vxproc[t-1,];
yproc[t,]=yproc[t-1,]+Vyproc[t-1,];
Vxproc[t,]=Vxproc[t-1,]+e1;
Vyproc[t,]=Vyproc[t-1,]+e11;

*observed;
xobs[t,]=xproc[t,]+e2;
yobs[t,]=yproc[t,]+e22;
end;

*xobs[10:15,]=.;
*yobs[10:15,]=.;

dat=x||y||Vx||Vy||xproc||yproc||Vxproc||Vyproc||xobs||yobs;
create kalman_data var{x y Vx Vy xproc yproc Vxproc Vyproc xobs yobs};
append from dat;
```

```
quit;

goptions reset=all;
title1 "True values vs. values with operational and observational error";
symbol1 interpol=join
value=none
color=green;
symbol2 interpol=join
value=none
color=maroon ;
symbol3 interpol=join
value=none
color=darkblue ;
axis1 label=("X");
axis2 label=("Y");
legend1 label=none
value=('Real process' 'Observations' 'Theoretical process')
position=(top left inside)
mode=share;
run;

proc gplot data=kalman_data;
plot yproc*xproc yobs*xobs y*x / overlay legend=legend1 haxis=axis1 vaxis=axis2;
run;

*Kalman filter;
proc iml;
use Kalman_data;
read all var {x y Vx Vy xproc yproc Vxproc Vyproc xobs yobs};

n=20;
Rt=I(4)*15;
Qt=7*I(2);
At={1 0 1 0,  0 1 0 1,  0 0 1 0,  0 0 0 1};
Bt=0;
ut=0;
zt=xobs||yobs||Vxproc||Vyproc;
ct={1 0 0 0,  0 1 0 0};
mustar=J(4,n,.);
mustar[,1]={0,0,2,2};
mut=J(4,n,.);
mut[,1]={0,0,2,2};
sigmat=I(4);

do i=2 to n;

*prediction step;
mustar[,i]=At*mut[,i-1]+Bt*ut;
sigmastar=At*sigmat*At`+Rt;

*update step;
Kt=(sigmastar*ct`)*inv(ct*sigmastar*ct`+Qt);
if zt[i,1:2]=. then do;
```

```
mut[,i]=mustar[,i];
end;
else do;
mut[,i]=mustar[,i]+Kt*(ct*t(zt[i,])-ct*mustar[,i]);
end;
sigmat=(1-Kt*ct)*sigmastar;
end;

xk=T(mut[1,]);
yk=T(mut[2,]);

dat2=xk||yk||xobs||yobs||xproc||yproc;
create plot var{xk yk xobs yobs xproc yproc};
append from dat2;

quit;

goptions reset=all;
title1 "Kalman filter";
symbol1 interpol=join
value=dot
color=black;
symbol2 interpol=join
value=circle
color=maroon ; symbol3
interpol=none
value=star
color=darkblue ;
axis1 label=("X");
axis2 label=("Y");
legend1 label=none
value=('Obsevered' 'Kalman filter' 'true')
position=(top left inside)
mode=share;
run;

proc gplot data=plot; plot yobs*xobs yk*xk yproc*xproc/ overlay legend=legend1 haxis=axis1 vaxis=axis2 ;
run;

quit;
```

Matlab code for 1D Kalman filter:

```
duration = 200;
dt = .5;
At = [1 dt; 0 1] ;
Bt = [dt^2/2; dt];
Ct = [1 0];
ut = 2;
mu= [0; 0];
mu_star = mu;
Rt = 0.09^2;
Qt = 36;
```

```
Sigmat = Rt * [dt^4/4 dt^3/2; dt^3/2 dt^2];
sigma_star = Sigmat;
x = [];
Vx = [];
measured_position = [];
figure(2);clf
figure(1);clf
for t = 0 : dt: duration
error1 = Rt^0.5 * [(dt^2/2)*randn; dt*randn];
mu= At * mu+ Bt * ut + error1;
    Obs_error = Qt * randn;
z = Ct * mu+ Obs_error;
x = [x; mu(1)];
measured_position = [measured_position; z];
Vx = [Vx; mu(2)];
    plot(0:dt:t, x, '-r.')
plot(0:dt:t, measured_position, '-k.')
axis([0 15 -30 300])
hold on
end plot(0:dt:t, smooth(measured_position), '-g.')
x_estimate = [];
Vx_estimate = [];
mu= [0; 0];
sigma_estimate = sigma_star;
sigma_mag_estimate = [];
Pred_state = [];
Pred_cov = [];
for t = 1:length(x)
% Prediction step
mu_star = At * mu_star + Bt * ut;
Pred_state = [Pred_state; mu_star(1)] ;
sigma_star = At * sigma_star * At' + Sigmat;
Pred_cov = [Pred_cov; sigma_star] ;
    % Kalman Gain
K = sigma_star*Ct'*inv(Ct*sigma_star*Ct'+Qt);
% Update step
   mu_star = mu_star + K * (measured_position(t) - Ct * mu_star);
sigma_star =  (eye(2)-K*Ct)*sigma_star;
x_estimate = [x_estimate; mu_star(1)];
Vx_estimate = [Vx_estimate; mu_star(2)];
sigma_mag_estimate = [sigma_mag_estimate; sigma_star(1)];
end
title('One dimensional Kalman filter')
xlabel('Time')
ylabel('Displacement')
figure(2);
tt = 0 : dt : duration;
plot(tt,x,'-r.',tt,measured_position,'-k.', tt,x_estimate,'-g.');
axis([0 15 -30 300])
```

Code for EZ-Robot colour tracking:

```
#Identify colour and point at that colour
:loop

#reset servo positions
ControlCommand("Auto Position", AutoPositionFrameJump, "Calibrate")
sleep(100)
ControlCommand("RGB Animator", AutoPositionAction, "spin")
servospeed(d0, 4)
servospeed(d1, 4)

#Define colour variable and start the camera
$CameraObjectColor = ("Blue")
ControlCommand("Camera", CameraStart)

SayWait("let me see")

#Enable colour tacking on the camera
ControlCommand("Camera", CameraMultiColorTrackingEnable)

#Define reaction to seeing or not seeing the colour
IF ($CameraObjectColor != "Blue")
  SayWait(" No I do not see the color")
ENDIF

IF ($CameraObjectColor = "Blue")
  SayWait("I see an object colored" + $CameraObjectColor)
#Say the camera coordinates of the object
  SayWait("It is in" +$CameraVerticalQuadrant +"and" +
$CameraHorizontalQuadrant)
ENDIF

#Move arms to react to the position of the object
#Object is on the left side
IF ($CameraHorizontalQuadrant = "Left")
  Servo(D4, 5)
  IF ($CameraVerticalQuadrant = "Bottom")
    Servo(D3, 85)
  ENDIF
  IF ($CameraVerticalQuadrant ="Top")
    Servo(D3, 114)
  ENDIF
  IF ($CameraVerticalQuadrant = "Middle")
    Servo(D3, 90)
  ENDIF
ENDIF

#Object is on the right side
IF ($CameraHorizontalQuadrant = "Right")
  Servo(D7, 5)
  IF ($CameraVerticalQuadrant = "Bottom")
    Servo(D2, 85)
```

```
  ENDIF
  IF ($CameraVerticalQuadrant = "Top")
    Servo(D2, 114)
  ENDIF

#Object is in the middle of the horizontal plane
#Move both arms instead of just the left or right
one when object is in the middle
  IF ($CameraVerticalQuadrant = "Middle")
    Servo(D2, 90)
  ENDIF
ENDIF
IF ($CameraHorizontalQuadrant = "Middle")
  Servo(D4, 5)
  Servo(D7, 5)
  IF ($CameraVerticalQuadrant = "Bottom")
    Servo(D2, 85)
    Servo(D3, 85)
  ENDIF
  IF ($CameraVerticalQuadrant = "Top")
    Servo(D2, 114)
    Servo(D3, 114)
  ENDIF
  IF ($CameraVerticalQuadrant = "Middle")
    Servo(D2, 90)
    Servo(D3, 90)
  ENDIF
ENDIF

#Stop camera
ControlCommand("Camera", CameraStop)
goto (loop)
```

Code Python Kalman filter and Gaussian mixture background subtraction:

```
# -*-Attempt 2-*-
"""
Created on Wed Jul 01 09:27:22 2015
@author: Prenil Sewmohan
"""

import cv2
import numpy as np
import matplotlib.pyplot as plt

file="singleball.avi"
capture=cv2.VideoCapture(file)
print "\t Width: ",capture.get(cv2.cv.CV_CAP_PROP_FRAME_WIDTH)
print "\t Height: ",capture.get(cv2.cv.CV_CAP_PROP_FRAME_HEIGHT)
print "\t FourCC: ",capture.get(cv2.cv.CV_CAP_PROP_FOURCC)
print "\t Framerate: ",capture.get(cv2.cv.CV_CAP_PROP_FPS)
```

```
numframes=capture.get(7)
print "\t Number of Frames: ",numframes

count=0
history = 10
nGauss = 3
bgThresh = 0.6
noise = 20
bgs = cv2.BackgroundSubtractorMOG(history,nGauss,bgThresh,noise)

plt.figure()
plt.hold(True)
plt.axis([0,480,360,0])

measuredTrack=np.zeros((numframes,2))-1
while count<numframes:
count+=1
img2 = capture.read()[1]
cv2.imshow("Video",img2)
foremat=bgs.apply(img2)
cv2.waitKey(100)
foremat=bgs.apply(img2)
ret,thresh = cv2.threshold(foremat,127,255,0)
contours, hierarchy = cv2.findContours(thresh,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)
if len(contours) > 0:
m= np.mean(contours[0],axis=0)
measuredTrack[count-1,:]=m[0]
plt.plot(m[0,0],m[0,1],'ob')
cv2.imshow('Foreground',foremat)
cv2.waitKey(80)
capture.release()
print measuredTrack np.save("ballTrajectory", measuredTrack)
plt.show()


# -*- coding: utf-8 -*-
"""
Created on Wed Jul 01 12:14:50 2015
@author: Prenil Sewmohan
"""
import cv2
import numpy as np
from pykalman import KalmanFilter
from matplotlib import pyplot as plt

Measured=np.load("ballTrajectory.npy")
while True:
if Measured[0,0]==-1.:
Measured=np.delete(Measured,0,0)
else:
break
if cv2.waitKey(1) == 27:
break
```

```
numMeas=Measured.shape[0]

MarkedMeasure=np.ma.masked_less(Measured,0)

Transition_Matrix=[[1,0,1,0],[0,1,0,1],[0,0,1,0],[0,0,0,1]]
Observation_Matrix=[[1,0,0,0],[0,1,0,0]]

xinit=MarkedMeasure[0,0]
yinit=MarkedMeasure[0,1]
vxinit=MarkedMeasure[1,0]-MarkedMeasure[0,0]
vyinit=MarkedMeasure[1,1]-MarkedMeasure[0,1]
initstate=[xinit,yinit,vxinit,vyinit]
initcovariance=1.0e-3*np.eye(4)
transistionCov=1.0e-4*np.eye(4)
observationCov=1.0e-1*np.eye(2)
kf=KalmanFilter(transition_matrices=Transition_Matrix,
observation_matrices =Observation_Matrix,
initial_state_mean=initstate,
initial_state_covariance=initcovariance,
transition_covariance=transistionCov,
observation_covariance=observationCov)
filtered_state_means,filtered_state_covariances)=kf.filter(MarkedMeasure)
plt.plot(MarkedMeasure[:,0],MarkedMeasure[:,1],'xr',
label='Gaussian mixture model background subtraction')
plt.axis([0,520,360,0])
plt.hold(True)
plt.plot(filtered_state_means[:,0],filtered_state_means[:,1],'ob',
label='kalman output')
plt.legend(loc=2)
plt.title("Constant Velocity Kalman Filter")
plt.show()
```

Generalizations of the logistic distribution

Anika Wessels 11027569

WST795 Research Report

Submitted in partial fulfillment of the degree BSc(Hons) Mathematical Statistics

Supervisors: Dr. Paul J. van Staden and Ms. B. Omachar

Department of Statistics, University of Pretoria

November 2, 2015

Abstract

In this report we look at various generalizations of the logistic distribution proposed in the literature. These include the density based skew-logistic, the quantile based skew-logistic, a reparametrization of the log-logistic distribution and four generalized logistic distributions (GLOs) labelled Type I to Type IV. We present the basic properties of these distributions, in particular, the mean and variance, and the skewness and kurtosis moment-ratios. Their flexibility in terms of distributional shape is compared via moment-ratio diagrams.

Declaration

I, *Anika Wessels,* declare that this essay, submitted in partial fulfillment of the degree *BSc(Hons) Mathematical Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Anika Wessels

_____
Dr. Paul J. van Staden and Ms. B. Omachar

_____
Date

Acknowledgements

I have taken effort in this research. However, it would not have been possible without the kind support and help of the individuals who guided and assisted me throughout.

Ms. B. Omachar and Dr. Paul J. van Staden are only two of those people I would like to extend my sincere thanks to. I am highly indebted to them for their guidance and constant supervision as well as for providing me with the necessary information and also for their support in completing my research.

My thanks and appreciations also go to my colleagues and peers for their friendship, encouragement and for willingly helping me out with their abilities.

Contents

List of Figures

List of Tables

1 Introduction

The logistic distribution is a continuous symmetric distribution with a unimodel bell-shaped density curve which resembles that of the normal distribution (see Figure 1) but with much heavier tails and higher kurtosis. The value of the kurtosis moment-ratio of the normal distribution is 3, while this value is 4.2 for the logistic distribution due to its relatively longer tails.

For a detailed discussion on the logistic distribution's theoretical properties and its applications, see Johnson et al. [11] and Balakrishnan [3]. This distribution is used in various growth models, see for example Pearl and Reed [13], for the reason that the hazard function is proportional to the cumulative distribution function (cdf), $F(x)$. Also, the logistic distribution is used with logistic regression which estimates the outcome of a dichotomous response variable [10]. It is a highly versatile distribution in the sense that it can be generalized in various ways. The idea behind generalization is to create more flexible distributions with respect to distributional shape, through the inclusion of some parameters.

In this report different generalizations of the logistic distribution will be studied. These include:

- a reparametrization of the log-logistic distribution
- the quantile-based skew logistic
- the density-based skew logistic
- four univariate types of generalizations: Type I, Type II, Type III and Type IV.

Closed form expressions (if they exist) for either the probability density function (pdf), $f(x)$, or the quantile (inverse cumulative distribution) function, $Q(p)$, can be used to obtain moments (raw and central) as well as moment-ratios. In particular, the skewness and kurtosis moment-ratios can be used to compare the shape properties of these generalized distributions. The moment-ratio diagram, which is a plot of the kurtosis against the skewness, is a graphical tool we will use for the comparison among these generalizations.

A detailed description of the various above-mentioned generalizations of the logistic distribution is given in this report. Section 2 gives an overview on the background of the different generalizations. We define these generalizations in full throughout Section 3 and describe their distributional properties. Section 4 presents the moment-ratio diagrams– the graphical tool used for comparing the different generalizations in terms of skewness and kurtosis. A conclusion is given in Section 5.

2 Background Theory

Different generalizations of the logistic distribution have been proposed in the literature. There are four types of univariate logistic generalizations namely, Type I, Type II, Type III and Type IV which are presented by Johnson et al. [11]. The properties of these distributions were summarized by Balakrishnan and Nevzorov [4], and they can be regarded as special cases of a general family proposed by Perks [14]. The aforementioned distributions have a shape parameter(s) incorporated, resulting in a positively skewed, negatively skewed or symmetric distribution depending on the value of the shape parameter(s). In all four cases the distributions coincide with the logistic distribution when the shape parameter(s) is set equal to one. From these four generalizations, only the Type III GLO is symmetrically distributed about zero, which can be used as an approximation to other symmetric distributions such as the Student's t distribution.

The density-based skew logistic distribution proposed by Wahed and Ali [17], is based on the skewness methodology by Azzalini [2]. The skewness parameter, λ, allows for a greater degree of flexibility in terms of its shape. The quantile-based skew logistic distribution, originally presented by Gilchrist [5] and analyzed by Van Staden and King [16], is defined through its quantile function, since closed form expressions for the cdf and pdf do not exist. Hosking [8] introduced another generalization of the logistic distribution. This generalization, as indicated by Hosking and Wallis [9], is a reparametrization of the log-logistic distribution by Ahmad et al. [1].

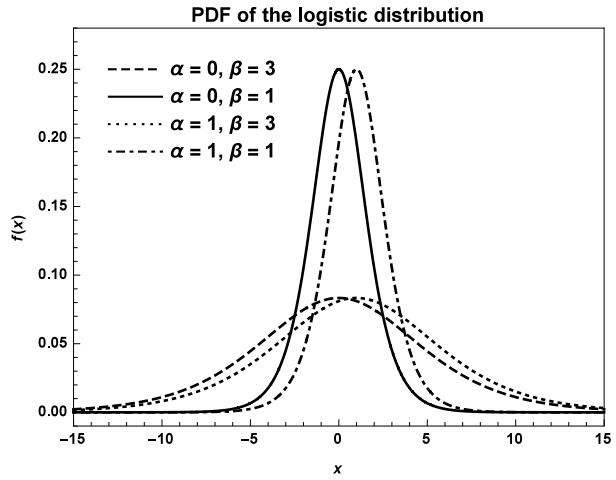


Figure 1: Probability density curves of the logistic distribution for different values of the location parameter α and scale parameter β

3 Definitions and properties

The generalizations are defined in this section. The mean and variance will be denoted as μ and σ^2 respectively, whereas the skewness and kurtosis moment-ratios will be indicated by α_3 and α_4 respectively. For application purposes the location parameter α is set equal to 0 and the scale parameter β is set equal to 1, without loss of generality.

3.1 Logistic Distribution

Before considering the different generalizations of the logistic distribution, the logistic distribution and its properties are briefly presented.

Definition 1. The random variable X has the logistic distribution, denoted $X \sim Lo(\alpha, \beta^2)$, if its cdf is given by

$$F(x) = \frac{1}{1 + e^{-\frac{(x-\alpha)}{\beta}}}\ , \ \ -\infty < x < \infty,\ -\infty < \alpha < \infty,\ \beta > 0$$

and pdf (see Figure 1 for different combinations of α and β) is

$$f(x) = \frac{e^{-\frac{(x-\alpha)}{\beta}}}{\beta(1 + e^{-\frac{(x-\alpha)}{\beta}})^2}$$

and quantile function given by $Q(p) = \alpha + \beta \log(\frac{p}{1-p})\ , \quad 0 < p < 1\ .$

Theorem 2. *The mean, variance, skewness moment-ratio and kurtosis moment-ratio of a random variable* $X \sim Lo(\alpha, \beta^2)$ *is*

$$\mu = \alpha$$
$$\sigma^2 = \frac{\pi^2\beta^2}{3}$$
$$\alpha_3 = 0$$
$$\alpha_4 = \frac{21}{5}$$

■

From the expressions given above (see Balakrishnan and Nevzorov [4]), α is the location parameter while β controls the spread of the distribution which is clear in Figure 1.

3.2 Skew Logistic Distribution

We distinguish between two types of skew logistic distributions: the density-based skew logistic (SLDB) and the quantile-based skew logistic (SLQB), with shape parameters λ and δ respectively.

Density-based skew logistic distribution (SLDB)

Wahed and Ali [17] proposed the density-based skew logistic distribution by introducing a new parameter to control the skewness. This is based on the skewness methodology by Azzalini [2].

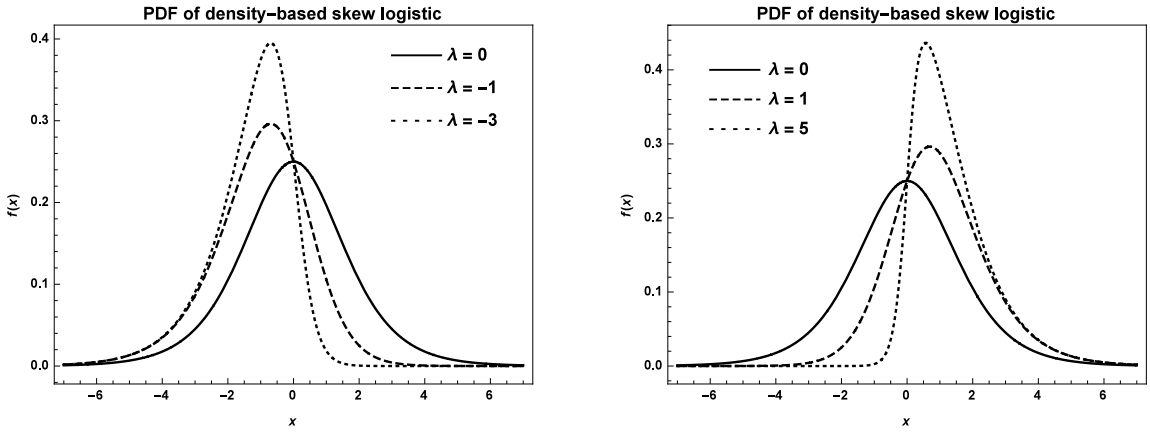
It is said that a random variable X has Azzalini's skew distribution if its pdf is of the form

$$f_X(x) = 2g(x)G(\lambda x);\ -\infty < x < \infty,\ \lambda \in \mathbb{R} \tag{1}$$

where $g(x)$ and $G(x)$ are the pdf and cdf of the symmetric distribution respectively.
The SLDB was further studied in detail by Gupta et al. [6] and Nadarajah [12].

Definition 3. Let X be a real-valued random variable. X is said to have a density-based skew logistic distribution if its pdf is given by

$$f(x) = \frac{2e^{-\frac{x-\alpha}{\beta}}}{\beta(1+e^{-\frac{x-\alpha}{\beta}})^2(1+e^{-\frac{\lambda x-\alpha}{\beta}})},\ -\infty < x < \infty,\ \lambda \in \mathbb{R}.$$



(a) (b)

Figure 2: Probability density curves of the density-based skew logistic distribution for different values of the shape parameter λ

Referring to Figure 2, (a) shows that the SLDB is negatively skewed when $\lambda < 0$, and (b) shows that it is positively skewed when $\lambda > 0$.

No closed form expressions are available for the moments of the SLDB. Gupta et al. [6] presented formulae for calculating the first four moments,

$$E[X] = \alpha + 2\beta A_1\,,$$

$$E[X^2] = \frac{1}{3}(\pi\beta)^2\,,$$

$$E[X^3] = 2\beta^3 A_3\,,$$

$$E[X^4] = \frac{7}{15}(\pi\beta)^4\,,$$

where $A_j = \int_0^\infty \frac{(log(z))^j}{(1+z)^2(1+z^{-\lambda})}dz\,,\ j = 1, 3\,.$

Theorem 4. *The mean and variance of a random variable X having a density-based skew logistic distribution are given by*

$$\mu = \alpha + 2\beta A_1$$

and

$$\sigma^2 = \beta^2\left(\frac{\pi^2}{3} - 4A_1^2\right)\,,$$

and the first two moment-ratios are

$$\alpha_3 = \frac{2\beta^3}{\sigma^3}\left(A_3 - 3\pi^2 A_1 + 8A_1^3\right)$$

and

$$\alpha_4 = \frac{\beta^4}{\sigma^4}\left(\frac{7}{15}\pi^4 - 16A_1A_3 + 8\pi^2A_1^2 - 48A_1^4\right)\,.$$

Proof. Using the raw moments by Gupta et al. [6] , the central moments are calculated. The second central moment is the variance, the third central moment multiplied by $\frac{1}{\sigma^3}$ is the skewness moment-ratio and the fourth central moment multiplied by $\frac{1}{\sigma^4}$ is the kurtosis moment-ratio. ■

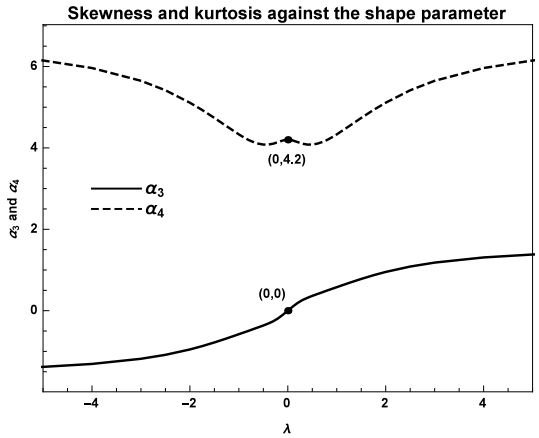


Figure 3: The skewness and kurtosis moment-ratios against the shape parameter for the density-based skew logistic distribution

Table 1 presents values for α_3 and α_4 for selected values of $\lambda \geq 0$. For $\lambda = 0$, in effect, for the logistic distribution, $\alpha_3 = 0$ and $\alpha_4 = 4.2$. For $\lambda < 0$, the skewness values are the same as those for $\lambda > 0$ with a change in sign. The kurtosis values for $\lambda < 0$ are the same as those for $\lambda > 0$.

Using the expressions in Theorem 4, values for α_3 and α_4 in Table 1 could be obtained for only certain values of λ by numerical integration with Mathematica 10.0 [15]. For all other values of λ, the values of α_3 and α_4 were estimated as the averages of the sample skewness and kurtosis moment-ratios of 20 simulated samples of size 50 000 each. These samples were simulated using the methodology proposed by Gupta and Kundu [7].

λ	α_3	α_4	Method
0	0.0000	4.2000	logistic distribution
0.1	0.0994	4.1832	estimated
0.2	0.1898	4.1432	estimated
0.3	0.2617	4.1029	estimated
0.4	0.3160	4.0821	estimated
0.5	0.3636	4.0820	estimated
0.6	0.4078	4.1036	estimated
0.7	0.4500	4.1429	estimated
0.8	0.4947	4.1935	estimated
0.9	0.5360	4.2593	estimated
1	0.5772	4.3327	numerical integration
1.25	0.6820	4.5325	estimated
1.5	0.7803	4.7370	estimated
1.75	0.8713	4.9324	estimated
2	0.9502	5.1116	numerical integration
2.5	1.0847	5.4163	estimated
3	1.1791	5.6506	numerical integration
4	1.3057	5.9653	numerical integration
5	1.3808	6.1481	estimated
6	1.4229	6.2681	numerical integration
7	1.4539	6.3391	estimated
8	1.4731	6.3904	estimated
9	1.4871	6.4277	estimated
10	1.4951	6.4606	numerical integration
100	1.5419	6.5762	estimated
1000	1.5423	6.5775	estimated

Table 1: The skewness and the kurtosis values of the density-based skew logistic for different values of the shape parameter λ

Quantile-based skew logistic distribution (SLQB)

The quantile-based skew logistic distribution introduced by Gilchrist [5] is defined through its quantile function since no closed form expressions for the pdf and cdf exist.

Definition 5. The random variable X has the quantile-based skew logistic distribution, denoted $X \sim SLQB(\alpha, \beta, \delta)$, if its quantile function is defined as

$$Q(p) = \alpha + \beta[(1-\delta)\log(p) - \delta\log(1-p)]\,,\ 0 < \delta < 1,\ 0 < p < 1\,, \tag{2}$$

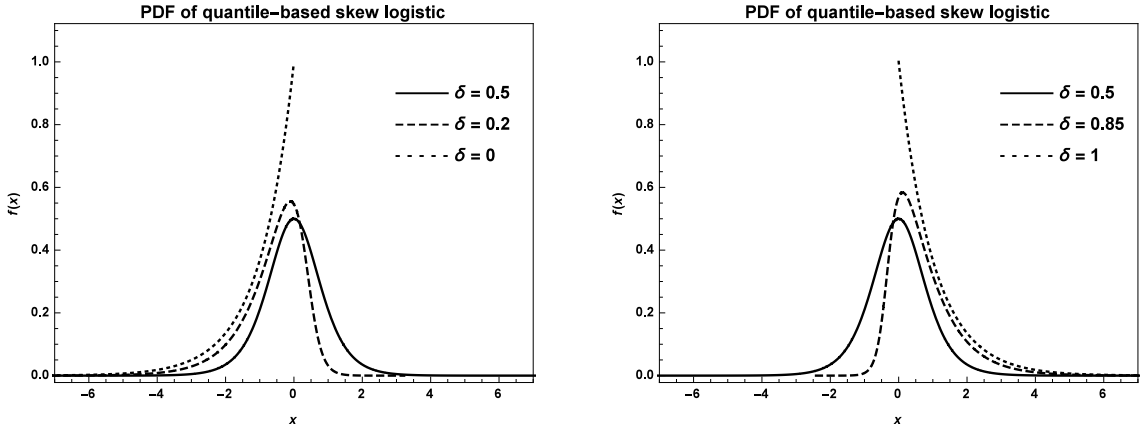
the quantile density function given as

$$q(p) = \beta\left(\frac{1-\delta}{p} + \frac{\delta}{1-p}\right)\,,\ 0 < p < 1 \tag{3}$$

and the density quantile function as

$$f_p(p) = \frac{p(1-p)}{\beta(\delta p + (1-\delta)(1-p))}\,,\ 0 < p < 1\,. \tag{4}$$

Remark 6. The equation in (3) is obtained through $\frac{dQ(p)}{dp}$ and equation (4) is simply $\frac{1}{q(p)}$.



(a) (b)

Figure 4: Probability density curves of the quantile-based skew logistic distribution for different values of the shape parameter δ

In Figure 4, (a) shows that this distribution is negatively skewed for $0 < \delta < 0.5$, (b) shows that it is positively skewed for $0.5 < \delta < 1$ and reduces to the logistic distribution when $\delta = 0.5$.

Theorem 7. *The mean, variance, the skewness moment-ratio and the kurtosis moment-ratio for* $X \sim SLQB(\alpha, \beta, \delta)$ *are given respectively by*

$$\mu = \alpha + \beta(2\delta - 1)\,,$$

$$\sigma^2 = \beta^2[(2\delta-1)^2 + \frac{\pi^2}{3}\omega]\,,$$

$$\alpha_3 = \frac{\beta^3}{\sigma^3}[2(2\delta-1)(1-\omega(4-3\zeta(3)))]\,,$$

and

$$\alpha_4 = \frac{\beta^4}{\sigma^4}[9 + \omega(2((2\delta-1)^2\pi^2 - 4) + (9\omega - 4)(16 - \frac{\pi^4}{15}))]\ ,$$

where $\omega = \delta(1-\delta)$ *and* $\zeta(a) = \sum_{j=1}^{\infty}\frac{1}{j^a}$ *the Riemann zeta function.*

Proof. The results given above were derived by Van Staden and King [16]. ■

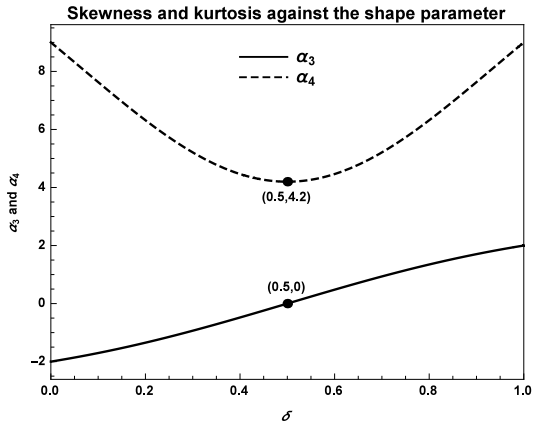


Figure 5: The skewness and kurtosis moment-ratios against the shape parameter for the quantile-based skew logistic distribution

As can be seen from Figure 5, symmetry is obtained when $\delta = 0.5$. The quantile-based skew logistic distribution has a kurtosis moment-ratio equal to 4.2 at the point of symmetry (same as the standard logistic distribution). This distribution is positively skewed for $\delta > 0.5$ and negatively skewed for $\delta < 0.5$.

3.3 Hosking's GLO

Hosking [8] proposed another generalization of the logistic distribution, henceforth referred to as Hosking's GLO, denoted by GLO_H. The logistic distribution is the special case where $k = 0$.

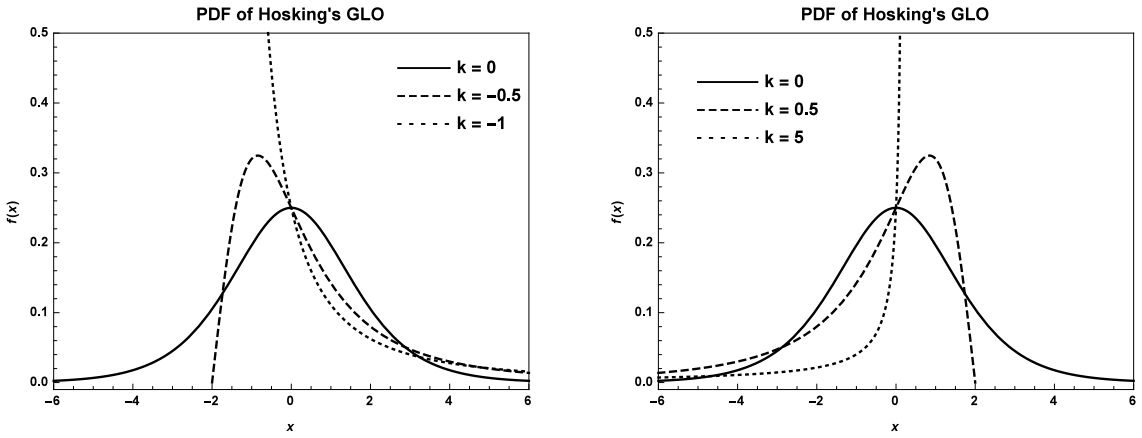
Definition 8. The random variable X has Hosking's GLO distribution, with shape parameter k, if the cdf is given by

$$F(x) = \frac{1}{(1+e^{-y})} \ ,$$

where $y = \begin{cases} -\frac{1}{k}\log\left[1-\frac{k(x-\alpha)}{\beta}\right] & k \neq 0 \\ \frac{x-\alpha}{\beta} & k = 0 \end{cases}$ and pdf $f(x) = \frac{e^{-(1-k)y}}{\beta(1+e^{-y})^2}$

and quantile function is defined as

$$Q(p) = \begin{cases} \alpha + \frac{\beta}{k}\left[1-\left\{\frac{1-p}{p}\right\}^k\right] & k \neq 0 \\ \alpha - \beta\log\left\{\frac{1-p}{p}\right\} & k = 0 \end{cases} \ , \ 0 < p < 1 \, .$$



(a) (b)

Figure 6: Probability density curves of Hosking's generalized logistic distribution for different values of the shape parameter k

As can be seen in Figure 6 (a) and (b), the pdf of Hosking's GLO is J-shaped when $|k| \geq 1$.

Theorem 9. *The conventional moments (see Hosking [8]) of* $X \sim GLO_H(\alpha, \beta, k)$ *are*

$$\mu = \alpha + \beta \frac{(1 - g_1)}{k},$$

$$\sigma^2 = \frac{\beta^2 (g_2 - g_1^2)}{k^2},$$

$$\alpha_3 = \frac{(-signk)(g_3 - 3g_2 g_1 + 2g_1^3)}{(g_2 - g_1^2)^{\frac{3}{2}}}$$

and

$$\alpha_4 = \frac{(g_4 - 4g_3 g_1 + 6g_2 g_1^2 - 3g_1^4)}{(g_2 - g_1^2)^2},$$

where $g_r = \Gamma(1 - rk)\Gamma(1 + rk)$. ■

Remark 10. The r^{th} order moment exists if $|k| < \frac{1}{r}$.

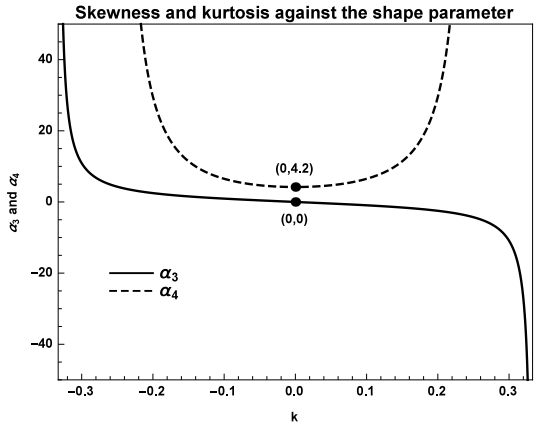


Figure 7: The skewness and kurtosis moment-ratios against the shape parameter for Hosking's generalized logistic distribution

Figure 7 shows that Hosking's GLO has point of symmetry equivalent to the logistic distribution when $k = 0$. It is clear that this distribution is negatively skewed for positive values of k and positively skewed for negative values of k.

3.4 Type I GLO

The first of the four univariate generalizations studied by Balakrishnan and Nevzorov [4], and Johnson et al. [11] is the Type I GLO, denoted GLO_I. The Type I distribution is negatively skewed when the shape parameter $\theta \in (0, 1)$ and positively skewed when $\theta > 1$. If $\theta = 1$, the Type I GLO reduces to the logistic distribution (see Figure 8) .

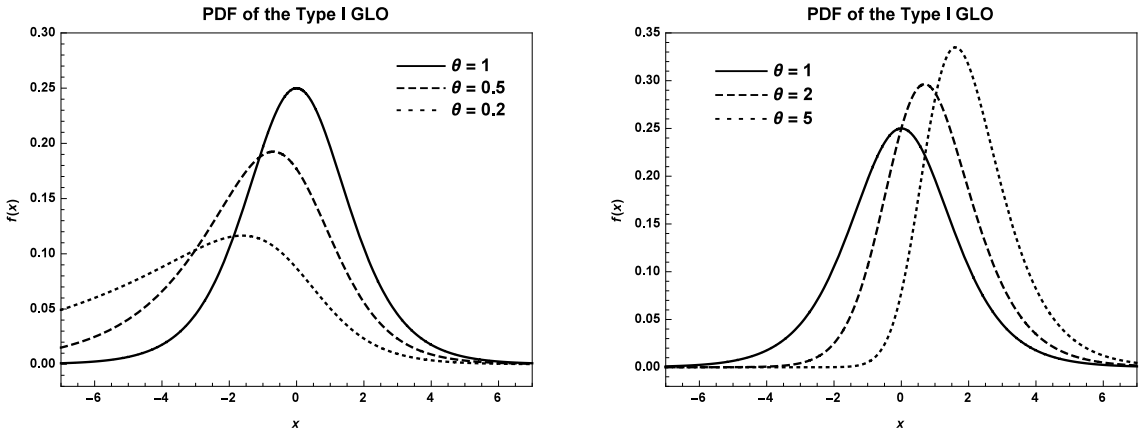
Definition 11. Let X be a random variable with a Type I generalized logistic distribution. The pdf of X is given by

$$f(x) = \frac{\theta\beta e^{-\frac{(x-\alpha)}{\beta}}}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{\theta+1}}\,, \quad -\infty < x < \infty,\ -\infty < \alpha < \infty,\ \beta > 0,\ \theta > 0\,,$$

the corresponding cdf is

$$F(x) = \frac{1}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{\theta}}$$

and quantile function is expressed as $Q(p) = -\log(-1 + p^{-\frac{1}{\theta}})\ , 0 < p < 1$.



(a) (b)

Figure 8: Probability density curves of the Type I generalized logistic distribution for different values of the shape parameter θ

Theorem 12. *The mean and variance of $X \sim GLO_I(\alpha, \beta, \theta)$ are given by*

$$\mu = \psi(\theta) - \psi(1) \text{ and } \sigma^2 = \psi'(\theta) + \psi'(1)$$

respectively, where $\psi(x) = \frac{d}{dx}\ln\Gamma(x)$ is the digamma function and $\psi^r(x) = \frac{d^r}{dx^r}\psi(x)$ the r^{th} derivative of the digamma function.

The skewness moment-ratio is given by $\alpha_3 = \frac{\psi''(\theta)-\psi''(1)}{(\psi'(\theta)+\psi'(1))^{3/2}}$ and the kurtosis moment-ratio by $\alpha_4 = 3 + \frac{\psi'''(\theta)+\psi'''(1)}{(\psi'(\theta)+\psi'(1))^2}$. ■

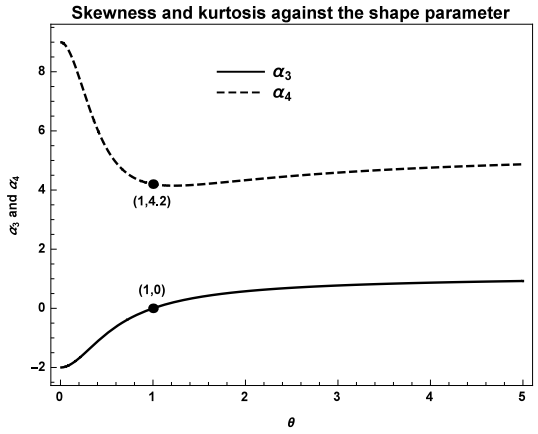


Figure 9: The skewness and kurtosis moment-ratios against the shape parameter for the Type I generalized logistic distribution

The point of symmetry for the Type I GLO is where the shape parameter is equal to 1 i.e. when $\theta = 1$. This is the point where this distribution reduces to the logistic distribution and therefore has a kurtosis of 4.2.(see Figure 9). The kurtosis tends to 5.4 as θ increases. Also evident from Figure 9 is the positive skewness when $\theta > 1$ which reaches a maximum of 1.13955 as the shape parameter increases. These limiting values were calculated by using the function *limit* in Mathematica 10.0 [15].

3.5 Type II GLO

The Type II GLO is related to Type I given above. If the random variable X has a Type I distribution then $-X$ has a Type II distribution. The latter has a positively skewed distribution when the shape parameter $h \in (0, 1)$ and is negatively skewed when $h > 1$.

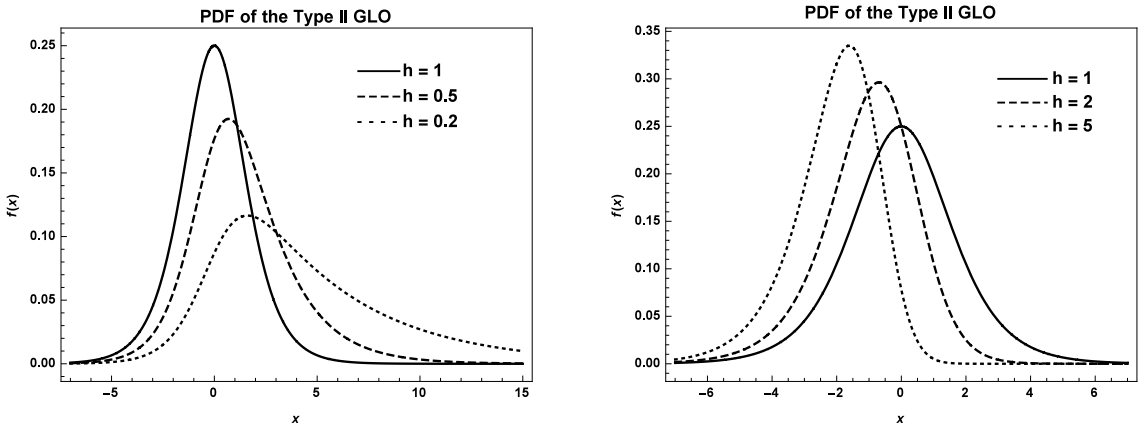
Definition 13. If the random variable X has a Type II generalized logistic distribution, then its pdf must be given by

$$f(x) = \frac{h\beta e^{-h\frac{(x-\alpha)}{\beta}}}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{h+1}}\ ,\ \ -\infty < x < \infty,\ -\infty < \alpha < \infty,\ \beta > 0,\ h > 0\ ,$$

the cdf given by

$$F(x) = 1 - \frac{\beta e^{-h\frac{(x-\alpha)}{\beta}}}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{h}}\ ,$$

and the quantile function $Q(p) = \log\left[-1 + \frac{1}{(1-p)^{1/h}}\right]\ , 0 < p < 1.$



(a) (b)

Figure 10: Probability density curves of the Type II generalized logistic distribution for different values of the shape parameter h

Remark 14. From the fact that the Type II GLO is the negative of the Type I GLO, the conventional moments follow accordingly.

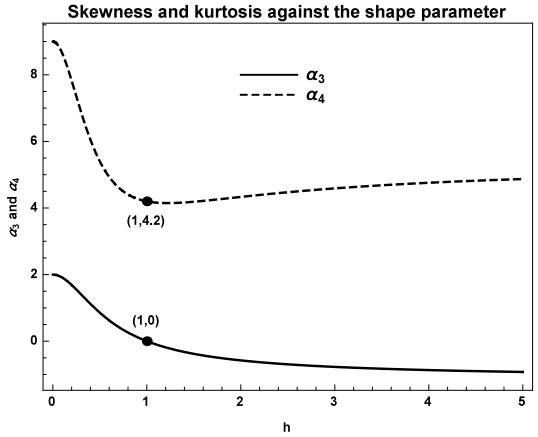


Figure 11: The skewness and kurtosis moment-ratios against the shape parameter for the Type II generalized logistic distribution

By comparing the skewness moment-ratio diagram illustrated in Figure 11 to that in Figure 9, Remark 14 is justified.

3.6 Type III GLO

The Type III GLO is the only distribution of the four univariate generalizations which is symmetric about zero, which is evident from the skewness moment-ratio curve in Figure 12. Therefore, the mean is zero and this applies to all the moments of odd order. The variance is $2\psi'(\kappa)$ and the kurtosis moment-ratio is given by $\alpha_4 = 3 + \frac{\psi'''(\kappa)}{2(\psi'(\kappa))^2}$ indicating that the Type III GLO has heavier tails (and therefore longer tails) than the normal distribution. For large values of the shape parameter κ, $\sqrt{2/\kappa}X$ will approximately be standard normal.

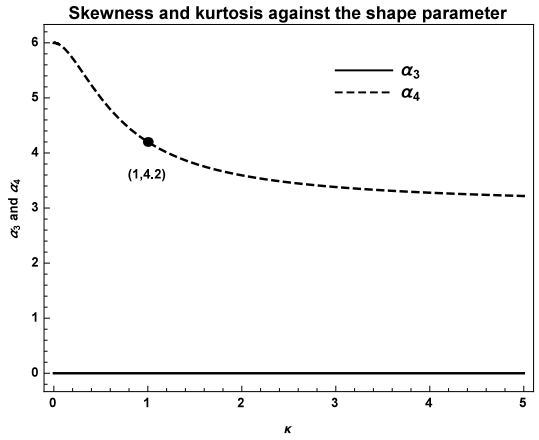


Figure 12: The skewness and kurtosis moment-ratios against the shape parameter for the Type III generalized logistic distribution

The kurtosis moment-ratio reaches a minimum of 3 as κ tends to infinity and a maximum of 6 as κ decreases to zero (see Figure 12) . These limiting values were calculated by using the function *limit* in Mathematica 10.0 [15]

Definition 15. The pdf of a Type III GLO random variable is

$$f(x) = \frac{1}{B(\kappa,\kappa)} \frac{\beta e^{-\kappa\frac{(x-\alpha)}{\beta}}}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{2\kappa}}\,, \quad -\infty < x < \infty,\ -\infty < \alpha < \infty,\ \beta > 0,\ \kappa > 0\,,$$

where $B(\kappa,\kappa) = \frac{(\Gamma(\kappa))^2}{\Gamma(2\kappa)}$.

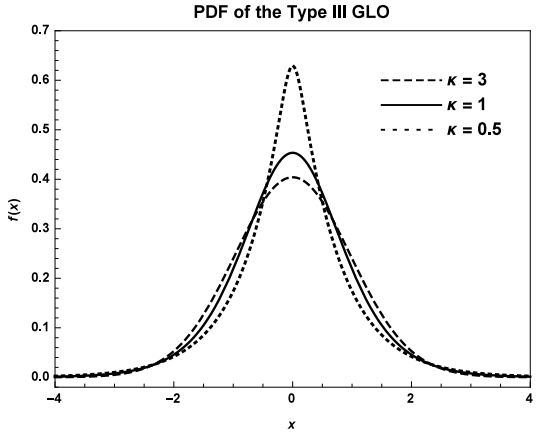


Figure 13: Probability density curves of the Type III generalized logistic distribution for different values of the shape parameter κ

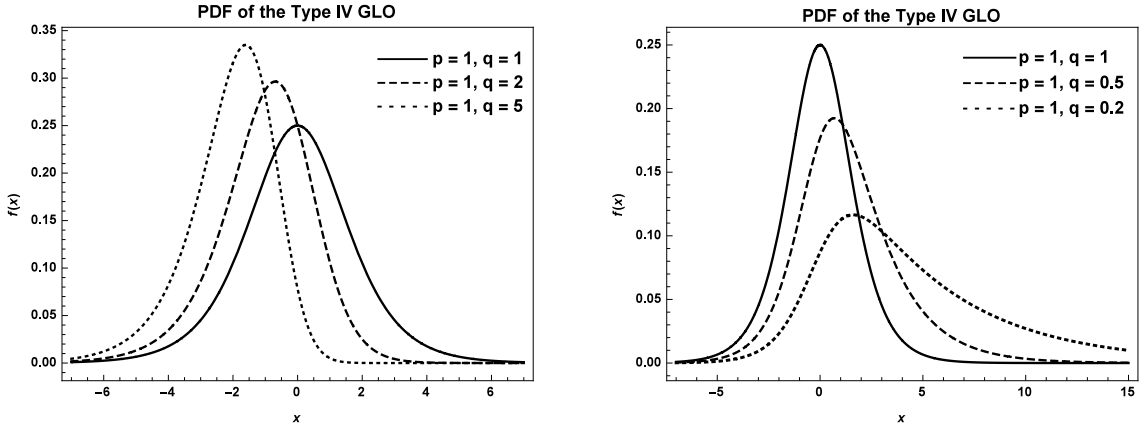
3.7 Type IV GLO

All the above-mentioned types of generalizations are special cases of the Type IV GLO, denoted GLO_{IV}, which has two shape parameters, p and q . Type I is obtained by setting $p = \theta$ and $q = 1$ .The same follows for the Type II which can also be regarded as the negative of the Type I. Type III is obtained by setting $p = q = \kappa$. The Type IV reduces to the logistic whenever $p = q = 1$ (Figure 14) .

Definition 16. For a random variable X to have the Type IV generalized logistic distribution its pdf must be given as

$$f(x) = \frac{1}{B(p,q)} \frac{\beta e^{-q\frac{(x-\alpha)}{\beta}}}{(1+e^{-\frac{(x-\alpha)}{\beta}})^{p+q}}\,, \;\; -\infty < x < \infty,\; -\infty < \alpha < \infty,\; \beta > 0,\; p > 0,\; q > 0\;,$$

where $B(p,q) = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)}$.



(a) (b)

Figure 14: Probability density curves of the Type IV generalized logistic distribution for different values of the shape parameters p and q

Theorem 17. *If $X \sim GLO_{IV}(\alpha, \beta, p, q)$, then*

$$\mu = \psi(p) - \psi(q)\,,$$

$$\sigma^2 = \psi'(p) + \psi'(q)\,,$$

$$\alpha_3 = \frac{\psi''(p) - \psi''(q)}{(\psi'(p) + \psi'(q))^{3/2}}$$

and

$$\alpha_4 = 3 + \frac{\psi'''(p) + \psi'''(q)}{(\psi'(p) + \psi'(q))^2}\,.$$

■

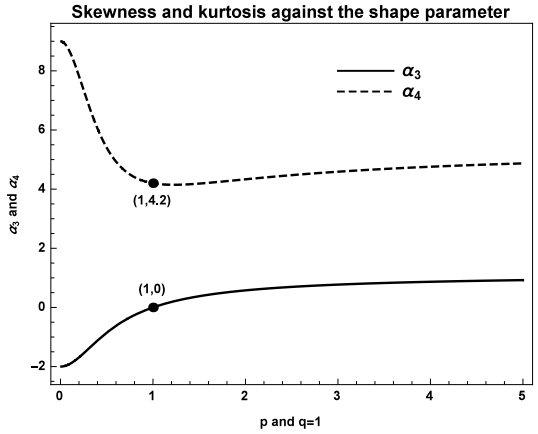


Figure 15: The skewness and kurtosis moment-ratios against the shape parameters for the Type IV generalized logistic distribution.

Fixing the value of $q = 1$, while increasing the value of p , we see that symmetry is obtained when $p = 1$ (see Figure 15). If $p > 1$, the Type IV GLO is positively skewed. A kurtosis of 4.2 is obtained when both parameters are equal to 1, since the Type IV GLO then reduces to the logistic distribution.

4 Moment-ratio diagrams

The generalizations are distinct in terms of their shape parameters and how these parameters affect the flexibility of the generalizations. Due to this fact, comparison of the different generalizations via their probability distributions is insufficient. A graphical tool used to compare these distributions in terms of their skewness moment-ratios and kurtosis moment-ratios is called a moment-ratio diagram.
Generalizations which consist of one shape parameter is a curve in the plotting region (Figure 16 (a), (b), (c), (d), (e) and (f)), while those with two shape parameters cover a region of possible combinations of α_3 and α_4 values (Figure 16 (g)).

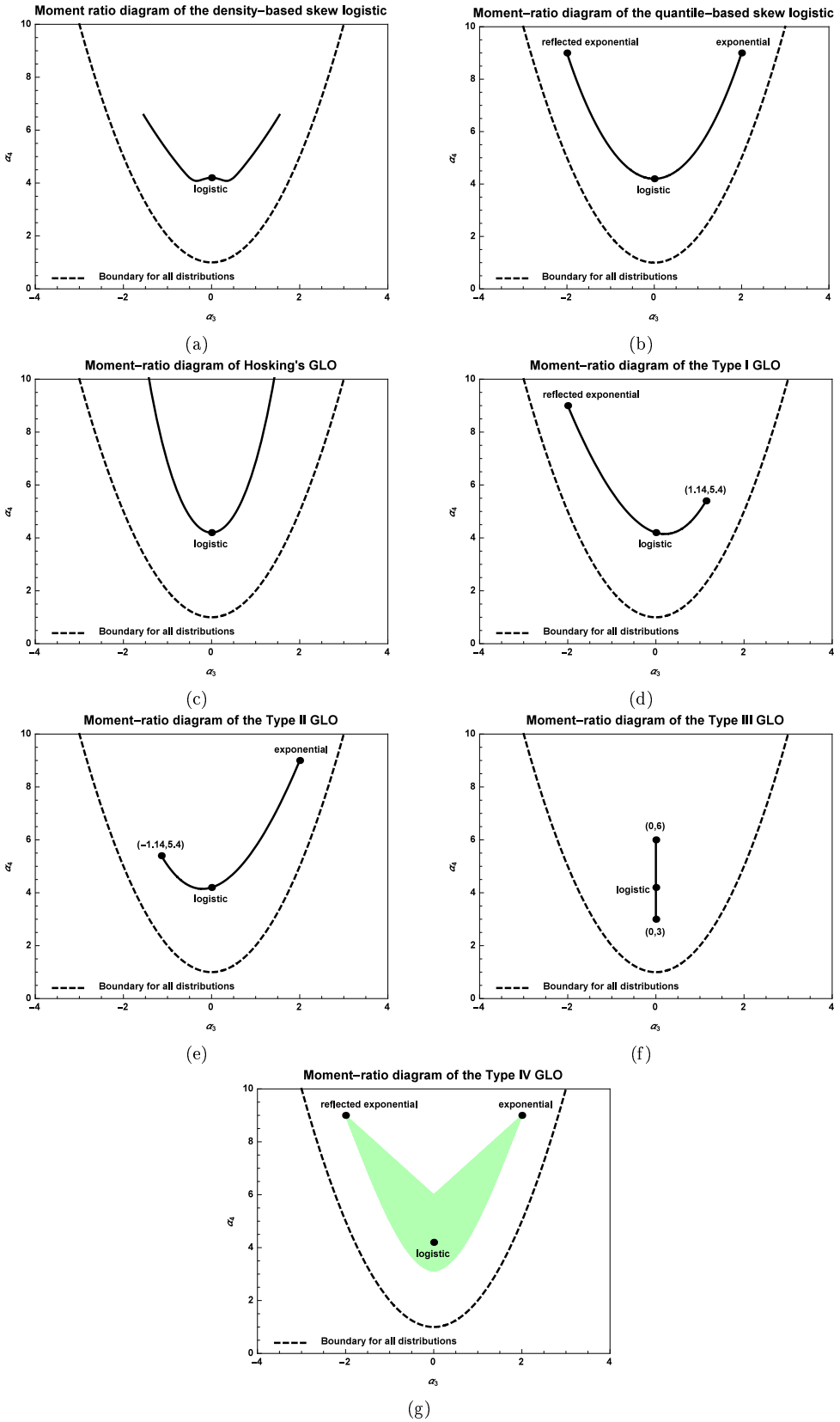


Figure 16: Moment-ratio diagrams of the different generalized logistic distributions

5 Conclusion

The aim of the report was to compare the flexibility and shape properties of various generalizations of the logistic distribution using the skewness and kurtosis moment-ratios for each of the generalizations. The moment-ratio diagram which is a plot of the kurtosis moment-ratio against the skewness moment-ratio, illustrated in Figure 17 below, is used for the comparison. It is evident that the Type IV GLO which covers the region shaded green is highly flexible with respect to shape as compared to the other generalizations. This is as a result of the two shape parameters that are included in this generalization. The boundaries for this region is the exponential and reflected exponential as indicated on the diagram.

Table 2 below summarizes the distributional shape of the various generalizations of the logistic distribution with regards to their shape parameters.

Distribution	**Negatively skewed**	**Symmetric**	**Positively skewed**
Density-based skew logistic	$\lambda < 0$	$\lambda = 0$	$\lambda > 0$
Quantile-based skew logistic	$\delta < 0.5$	$\delta = 0.5$	$\delta > 0.5$
Hosking's GLO	$k > 0$	$k = 0$	$k < 0$
Type I GLO	$\theta < 1$	$\theta = 1$	$\theta > 1$
Type II GLO	$h > 1$	$h = 1$	$h < 1$
Type III GLO	*Always symmetric*		
Type IV GLO	$p < q$	$p = q$	$p > q$

Table 2: The distributional shape as a result of different values of the shape parameter(s)

It is evident from Table 2 that since the Type I and Type II GLO are reflective distributions of each other, the distributional shape of the pdfs are reflective when their shape parameters are equal to 1. The density-based skew logistic and Hosking's GLO is symmetric when the corresponding shape parameter is equal to zero. Hosking's GLO behaves opposite to the value of the shape parameter i.e. negatively skewed for positive values of the shape parameter and positively skewed for negative values of the shape parameter. The Type I and Type II GLOs is symmetrically distributed when the respective shape parameter is equal to 1. The quantile-based GLO is symmetric for the shape parameter equal to 0.5. For the Type IV GLO, symmetry is obtained when the shape parameters are equal and in particular reduces to the logistic distribution when both shape parameters are equal to 1.

For a selected level of skewness, Hosking's GLO obtain a higher level of kurtosis compared to the other generalizations (Figure 17). Figure 18 zoomed in on Figure 17 for more clarity.

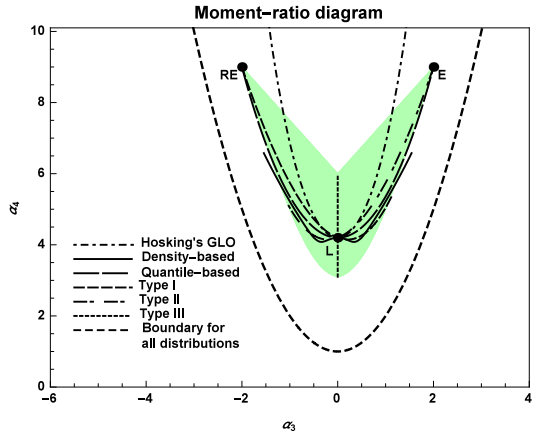
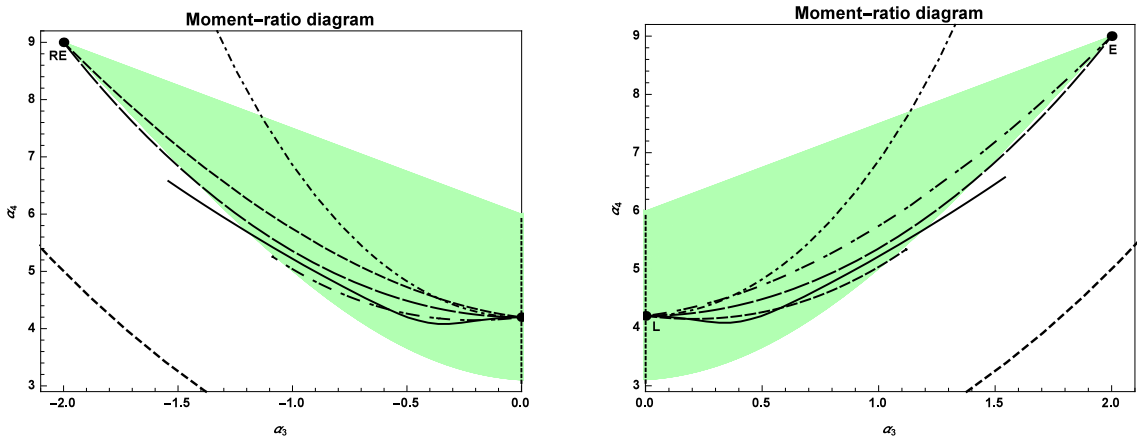


Figure 17: Moment-ratio diagram of all the generalizations: the exponential, reflected exponential and the logistic distributions are indicated by E, RE and L respectively.



(a) (b)

Figure 18: Moment-ratio diagram (zoomed in)

In this report six of the seven generalizations possessed one shape parameter whereas only one, the Type IV, possessed two shape parameters. Future research could compare other generalizations with more than one shape parameter and/or other families of distributions with the logistic as a special or limiting case.

References

[1] M.I. Ahmad, C.D. Sinclair, and A. Werritty. Log-logistic flood frequency analysis. *Journal of Hydrology*, 98(3):205–224, 1988.

[2] A. Azzalini. A class of distributions which includes the normal ones. *Scandinavian Journal of Statistics*, 12(2):171–178, 1985.

[3] N. Balakrishnan. *Handbook of the Logistic Distribution*. Marcel Dekker, Inc., New York, 1992.

[4] N. Balakrishnan and V.B. Nevzorov. *A Primer on Statistical Distributions*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2004.

[5] W. Gilchrist. *Statistical Modelling with Quantile Functions*. CRC Press, 2000.

[6] A.K. Gupta, F. Chang, and W. Huang. Some skew-symmetric models. *Random Operators and Stochastic Equations*, 10(2):133–140, 2002.

[7] R.D. Gupta and D. Kundu. Generalized logistic distributions. *Journal of Applied Statistical Science*, 18(1):51–66, 2010.

[8] J.R.M. Hosking. The theory of probability weighted moments. Technical report, RC12210, IBM Research, Yorktown Heights, 1986.

[9] J.R.M. Hosking and J.R. Wallis. *Regional Frequency Analysis: An Approach Based on L-moments*. Cambridge University Press, United Kingdom, 1997.

[10] D.W. Hosmer Jr and S. Lemeshow. *Applied Logistic Regression*. John Wiley & Sons, Inc., New York, 2004.

[11] N.L. Johnson, S. Kotz, and N. Balakrishnan. *Continuous Univariate Distributions*, volume 2. John Wiley & Sons, Inc., New York, 1995.

[12] S. Nadarajah. The skew logistic distribution. *AStA Advances in Statistical Analysis*, 93(2):187–203, 2009.

[13] R. Pearl and L.J. Reed. On the rate of growth of the population of the United States since 1790 and its mathematical representation. *Proceedings of the National Academy of Sciences of the United States of America*, 6(6):275, 1920.

[14] W.F. Perks. On some experiments in the graduation of mortality statistics. *Journal of the Institute of Actuaries*, 11:12–57, 1932.

[15] Wolfram Research. Mathematica 10.0, Champaign, Illinois, 2014.

[16] P.J. van Staden and R.A.R. King. The quantile-based skew logistic distribution. *Statistics & Probability Letters*, 96:109–116, 2015.

[17] A. Wahed and M.M. Ali. The skew-logistic distribution. *Journal of Statistical Research*, 35:71–80, 2001.

A mathematical model of trends, conformity and non-conformity

Keunyoung Yoo 12084795

STK795 Research Report

Submitted in partial fulfillment of the degree BCom(Hons) Statistics

Supervisor(s): Dr A de Waal, Co-supervisor(s): Mr M. T. Loots

Department of Statistics, University of Pretoria

2 November 2015

Abstract

In this paper a model of predicting trend that incorporates information delay is investigated as opposed to a Markov chain approach of trend prediction (which does not take information delay into account). This paper will also explain why and how the new model can give us more insight to the problem and possible applications of the model will also be discussed.

Declaration

I, *Keunyoung Yoo,* declare that this essay, submitted in partial fulfillment of the degree *BCom(Hons) Statistics,* at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

_____
Keunyoung Yoo

_____
Dr A de Waal, Mr M.T. Loots

_____
Date

Contents

List of Figures

List of Tables

1 Introduction

Consider the following three factors: first, we live in a world where, if examined carefully, many things are explained with binary (two) states: wins and losses; likes and dislikes; purchasing or not purchasing [1], etc. Second, as human beings our choices (and therefore our outcome) are influenced by our own thoughts and thoughts of those who are around us - our family, friends and the broader society in which we live. And third, one person may have a nature that is very susceptible to suggestions and follow those who are around while another person may always choose to simply do what no one else does. By bringing these three ideas together, we are able to explain how a person will make binary choices that will lead to binary outcomes - initially based on his/her own preferences but later on by taking others' opinion into consideration. This behavior can be described by a statistical process, which is a "mathematical model that evolves over time in a probabilistic manner" [6]. We will then add another factor, a delay, and examine how this changes the behavior of the individuals. By combining choices, influence and delay, we can describe this process as the "Hipster Effect," where we find "synchronization in random systems"[1]. The topic of this paper is in the realm of Statistical Physics, which explains the phenomena of physics by making use of statistical techniques. Within the field of physics there are numerous worlds that vary in size, from subatomic to astronomic. As such, describing a bigger world by using components from the smaller world could sometimes provide a good approximation. In the book written by Huang, it is said that statistical methods "provide a bridge between the microscopic and the macroscopic world" and the work done in this paper is to see how individuals (microscopic) behave due to the majority (macroscopic) behaviors [5]. However, despite the origin of the topic, it is not only limited to application in natural science but can also be used in neuroscience as well as finance [8]. In the application section, we will take a look at real data from the U.S. stock market and compare it with the simulated data, an empirical study that is not yet been explored. We will also briefly look at the extension of the model, when 3 or more choices are available.

2 Background Theory - Model

The basic setup of the problem is as follows: we have individuals who are either defined as "hipsters" (who deviate from the overall consensus) and "mainstream" (who follow the overall consensus). These individuals will have two choices available, where for this explanation we use iPhones and Samsung. Thus if a hipster individual, who possesses an iPhone, notices that most people own Samsung, he will be happy and keep his iPhone whereas if he possessed a Samsung phone, he will immediately go and get an iPhone. The opposite will apply for a Mainstream individual so a Mainstream with an iPhone and a consensus of Samsung will induce the individual to switch to Samsung and vice versa. For some people, family may have a big influence on making a decision while for others, friends may have a bigger influence. Therefore each individual assigns a different weight onto the choices of those who surround the individual and when it is added up, this will be the consensus that the individual will consider.

2.1 Mathematical setup

What has been explained above verbally can be rewritten mathematically in order to solve the problem. The variables have been defined in line with Touboul's work [8]. We have n number of individuals across t time periods, giving us an $n \times t$ matrix to work with. Each cell in the matrix will contain the state (or choice) or the individual at time t (discrete). Therefore numbers in a row tell us how an individual's choices have changed over time whereas numbers in a column tell us what the choices of all individuals at time t was, namely:

$$\text{Nature of individual } (\varepsilon) \begin{cases} Hipster & : -1 \\ Mainstream & : 1 \end{cases}$$

[1] Personal correspondence with the author of [8], Jonathan Touboul

$$\text{State at t}(s(t)): \begin{cases} iPhone & : -1 \\ Samsung & : 1 \end{cases}$$

We assume that ε and $s(1)$ are determined prior to the simulation and fixed throughout the simulation. ε is a column vector of dimension $n \times 1$ and $s(t)$ is also a $n \times 1$ column vector, since it only looks at the states of all individuals at time t. Both variables can be randomly generated from a symmetric distribution with a mean of 0 and classified to 1 if the number is positive and -1 if the number is negative. Next we define the influence matrixJ_{ij}, an $n \times n$ matrix with positive values which is also determined prior to the simulation. It is not symmetric since two people do not necessarily influence each other with the same magnitude. The main diagonals are 0, since a person's decision does not influence himself/herself.
With J_{ij} and $s(t)$, we can calculate the consensus, or the mean-field trend, that individual i perceives at time t according to the following formula:

$$m_i(t) = \frac{1}{n} \sum_j J_{ij} s_j(t)$$

This results in $m_i(t)$ being a $n \times 1$ column vector, where every element will either be a positive or a negative number. This represents whether the individual views the the aggregate phones in use to be either Samsung or iPhone. Therefore the mean-field trend is simply the weighted average of all the choices that are seen around the individual i.
By multiplying ε_i, $m_i(t)$ and $s_i(t)$ elementwise, we can determine x, a new variable which we will use to define the new state:

$$x_i = \varepsilon_i m_i(t) \left| s_i(t) \right|$$

Bringing this equation back to our story, if a hipster individual feels that the general consensus is Samsung and he owns an iPhone at the time,

$$x_i = (-1)(+) \left| (-1) \right| = -1$$

and he will continue to own his iPhone. Conversely, a Mainstream individual feeling the general consensus of Samsung and owning an iPhone will take on the following values:

$$x_i = (1)(+) \left| (-1) \right| = 1$$

and we can see that the Mainstream individual will now switch to Samsung. What matters is the sign of the answers and not the magnitude of the answer value. Here, 1 was used only to keep the illustration simple. This is similar to the calculations performed using Markov chains since the next state of any individual depends on the current state of all individuals [6]. Table 1 summarizes the possible outcomes for different values of ε, m and s.

$x_i = \varepsilon_i m_i(t) \left\| s_i(t) \right\|$	MFT			
	Samsung(+)		iPhone (-)	
	Hipster (-1)	Mainstream (1)	Hipster (-1)	Mainstream (1)
iPhone (-1)	-1	1	1	-1
Samsung (1)	-1	1	1	-1

Table 1: Different outcomes of x_i depending on different values of ε, m and s

2.2 Sigmoid function and Poisson process

Thus far in our discussion there has been no randomness in determining the new state, x_i, everything was deterministic with predetermined values for ε, $s(t)$, J_{ij}. Thus the convergence of consensus, if it were to happen, would mature immediately and show a clear cyclical pattern. This is no different from saying that

everyone has access to perfect information, which is a rather unrealistic assumption. In order to bring it closer to our world, we add what is referred to as the noise [8]. The idea is that if the noise level was too high and prevented the information from getting through to the individual, he will keep his current state. However, if an individual was able to perceive the information beyond the level of noise from the environment, he will adjust his new state accordingly. This is achieved by making use of a sigmoid function, where we insert the newly obtained x_i and compare it to a random level of noise from the uniform distribution. For now, we let the sigmoid function take the form of $\varphi(x) = \frac{1+\tanh(\beta x)}{2}$ where $\varphi(x)$ is a rate parameter (typically denoted as λ) in an inhomogeneous Poisson process. Here, the Poisson process would translate to whether the individual will either continue to use one type of phone or switch to the other type of phone in the next state [8]. If a Mainstream individual's choice differs from the perceived consensus, he is more likely to change his state in the next time period and this is synonymous with having a higher rate of switch in a Poisson process. The β is used to change the "sharpness of the rate function" [8] and this means only a sufficiently high value of β ensures that an individual is aware of the surroundings beyond the noise level and will make informed decisions. Gladwell calls this "The tipping point" and attributes the sudden rise in sale of Hush Puppies products in the mid 90s to such phenomenon [4]- the β had reached a point high enough that would start a trend. We will later try the logistic and Cauchy distributions as the sigmoid functions and see how they differ from the plain $\tanh(\beta x)$.

2.3 Adding Delay

Although the very simple model that was explained above can do a fairly good job in laying the foundation of how this idea of hipster effect works, it can be made more realistic by adding just one more factor. The assumption until now is that essentially, everyone in this system has perfect information at any given time,t. Therefore an individual can exactly calculate the consensus and determine whether he/she should keep or change the state for the next time period, $t+1$. In reality, this is highly unlikely as there will be many factors, such as distance and communication medium, that will hinder information from being transferred effectively and instantaneously [2]. Therefore, in order to make the assumption slightly more practical, we now assume that individuals perceive information after a bit of delay, which we will label as τ. This will only change our mean-field trend to:

$$m_i(t) = \frac{1}{n}\sum_j J_{ij}s_j(t-\tau_{ij})$$

where the value τ_{ij} is a is randomly generated positive integer. Generating it from a Poisson distribution is the easiest way, since the numbers will be discrete. Hence it will most likely be different for each individual, so that individual i will see the consensus after some slight variation of time τ has passed. Empirical estimation of τ_{ij} for large n will not be deal with in this paper, as the scope can be very broad and complex.

3 Application

3.1 Pseudo code

The following pseudo code explains the necessary steps needed to simulate the results needed for this experiment and [9] served as the foundation. Algorithm 1 deals with creating spaces and inserting values into the spaces to perform the calculations needed in Algorithm 2.

Algorithm 1 Initialization/Input

Require: Specify number of individuals (n): scalar
Require: Specify number of time period (t): scalar
Require: Specify hipster fraction and initial fraction of iPhone owners ($0 <$ fraction < 1): scalar
Require: Specify nature of individuals (ε) and initial state (S_1): n by 1 column vector
for each vectors i=1 to n **do**
Use Uniform distribution to generate n random numbers and insert each value into the ith row of ε and (S_1)
Positive numbers are changed into 1 and negative numbers are changed into -1
end for
Require: Specify influence matrix (J) and delay matrix (D): n by n matrix
$J_{i,j}$=influence of j on i, $D_{i,j}$=time taken for i to be aware of j's decision
for i= 1 to n **do**
for j=1 to n **do**
Use Normal distribution for J and Poisson distribution for D to generate n^2 random numbers and insert each value into the ith row and jth column of J and D
Remove main diagonals since influence comes from other individuals and there is no delay in knowing what the individual himself/herself has chosen
end for
end for

Algorithm 2 uses the values from the initialization part to create column vector of new choices.

Algorithm 2 Loop/Process

Require: Construct the delayed state vector (*sdelay*): n by 1 vector
Determine whether enough time has passed for individual i to be aware of individual j's state at time t
for i=1 to n **do**
If $t \leq D_{i,j}$ then $sdelay(i)$=$s_1(i)$
Else $sdelay(i)$=$s_{t-D}(i)$
end for
Require: Calculate $m_i(t)$ using $J, sdelay$
Require: Construct the new state vector (s_{new})
for i=1 to n **do**
Require: Determine $s_{new}(i)$ using $\varepsilon, m_i(t)$ and *sdelay*
Substitute values of $\varepsilon, m_i(t)$ and *sdelay* into a sigmoid function. This will generate a number, *Switchprob*, between 0 and 1.
Use a standard normal distribution to generate *noise*, numbers between 0 and 1. This represents the amount of noise that are present between the individuals. If it is significantly high, individuals will not be aware of others' states and will not change states
Therefore, if $Switchprob > noise$ then $s_{new}(i) = 1$
Else $s_{new}(i)$ = -1
end for
Repeat the above t times and horizontally concatenate s_{new} every time to the previous states to create the resultant matrix

Algorithm 3 is very simple, thanks to the heatmap function of SAS/IML® software, University Edition for Windows.[2]

[2]The output for this paper was generated using SAS software. Copyright, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.

Algorithm 3 Output

Display results in a grid where vertical axis is n, horizontal axis is t and the value at every n, t indicates the state, either -1 (brown) or 1 (white)

3.2 Graphical output and interpretation

In all the following figures the vertical axis represents the ith individual and the horizontal axis represents the time periods. The different colours at a given x, y coordinate represents either one of the two decisions made by ith individual at time t. Firstly we take a look at the instance where there are a few hipsters and many Mainstream. In Figure 1, it is clear that Mainstream will soon find their consensus and have no incentive to deviate from it. Likewise, if the majority of the population continuously tends to a particular option, hipsters will always want to choose the less selected option. Once a consensus is formed, everyone is happy with the decision made and it barely changes over time.

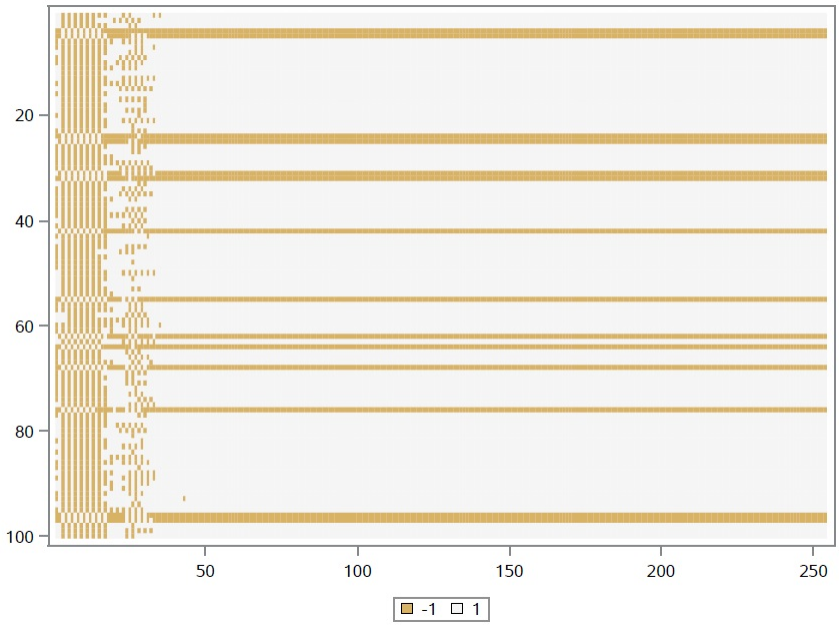


Figure 1: Hipster fraction=10%

Secondly, in we look at the situation where there are more or less equal number of hipsters and Mainstream in figure 2. Under such a circumstance, it will be difficult to form a solid consensus since whatever the Mainstream want to stick with, hipsters will abandon. This will result in no clear consensus throughout the whole time period and choices look rather random.

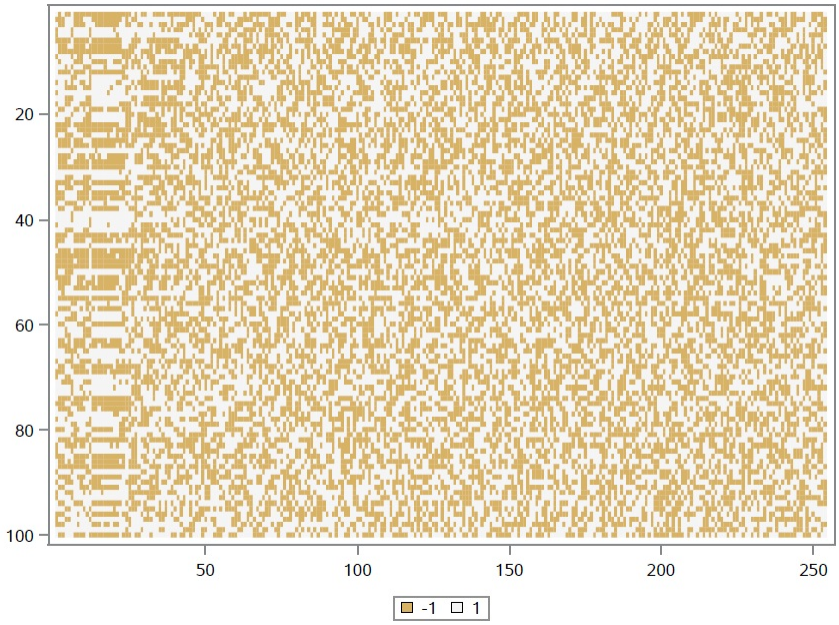


Figure 2: Hipster fraction=50%

Lastly, we look at the interesting case, where the majority of the population comprises of hipsters in figure 3. Until the information regarding others' decisions is obtained, the choices made by individuals are random. However, once they start becoming aware of what the others are doing, they behave accordingly. Since this population is mostly made up of hipsters, they end up walking away from their prior decision after realising that they have been doing the same thing as those around them (the duration is determined by the delay parameter). Since there are only two choices available in this scenario, the hipsters end up cyclically switching their decisions from one state to another. For a practical example of such behavior, think of share trading at a stock exchange - once a particular share's market price starts rising many traders will want to sell that share which they own in order to make a profit. When a substantial amount of shares are available on the market, with excess supply the share price will fall and some individuals will start purchasing this share, hoping that its market price will once again rise. With many such buyers the demand for the share will rise and so will the price, taking us to the stage where they will be sold again.

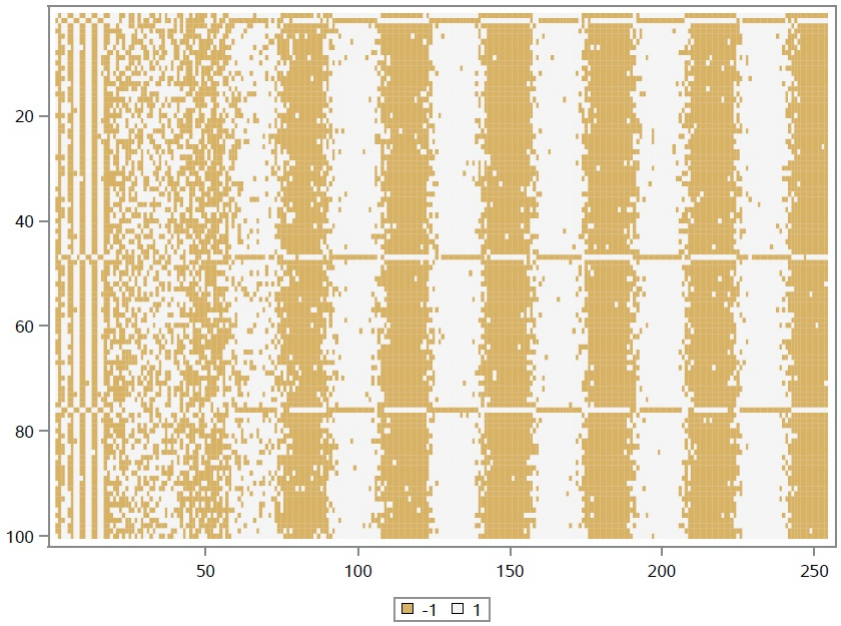


Figure 3: Hipster fraction=80%, $\beta = 1$

3.3 Changing the sigmoid function

The results in Section 3.2 were produced using the hyperbolic tangent function as the sigmoid. It is, however, not the only sigmoid function that exists. The logistic distribution is a prime example of modeling qualitative variables and it is also defined for all real values of x with y values between 0 and 1. Since any CDF of a continuous probability distribution defined for all x values will do, the Cauchy distribution has also been chosen. Because of the way the CDF is defined, logistic and Cauchy needed a higher value of β in order to show the synchronization than for $\tanh(\beta x)$ but it is clear that $\tanh(\beta x)$ is not the only function that can serve as the sigmoid. This allows us a number of ways to compare the simulated model with a real data set.

Figure 4 was generated using the R software [7]. As we can see in figure 4, the CDF of logistic distribution and the $\tanh(\beta x)$ function are quite similar to each other, where the only difference being the steepness between the two functions - logistic function slightly takes longer to reach the asymptote.

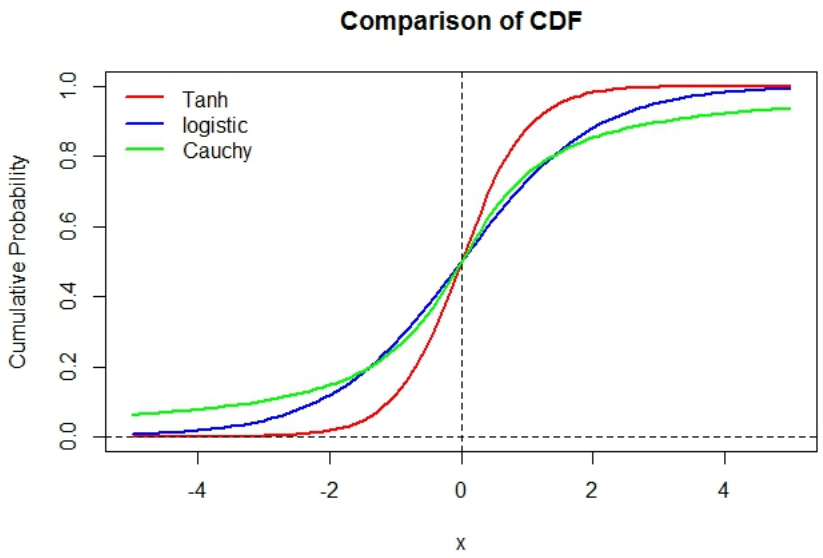


Figure 4: Comparison of different sigmoid functions

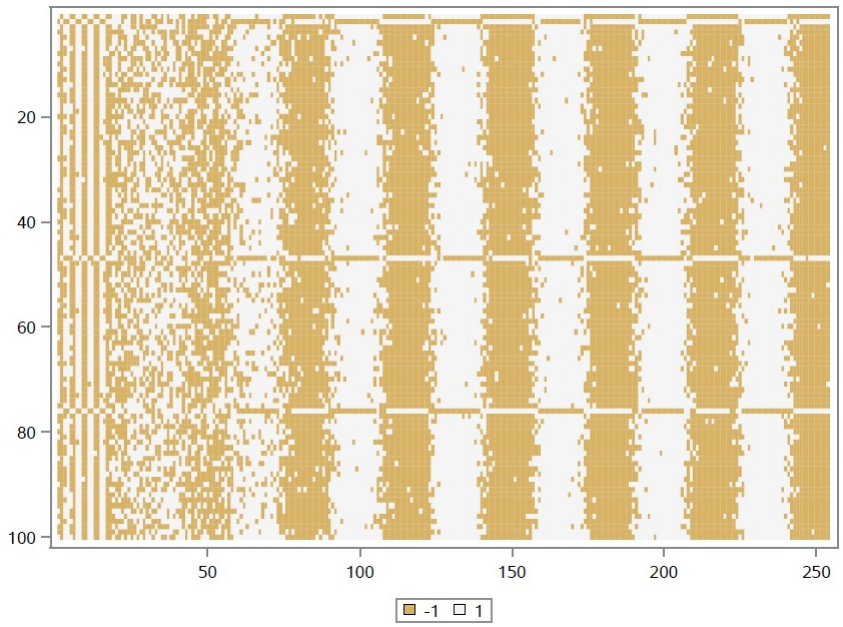


Figure 5: Logistic distribution, β=8

Hence in figure 4, the graph looks almost identical to figure 3.

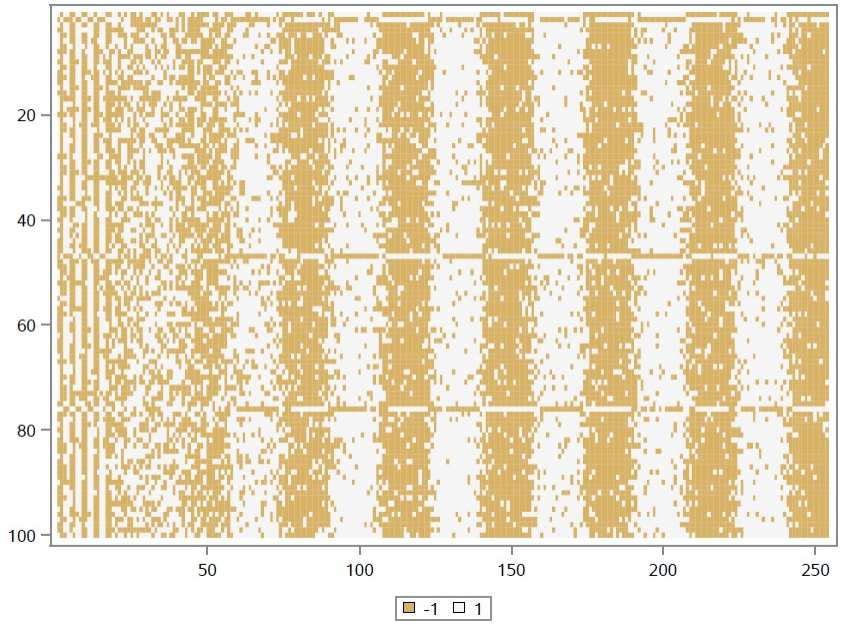


Figure 6: Cauchy distribution, β=8

Cauchy distribution, on the other hand, takes noticeably longer to reach the asymptote compared to the other two functions. The effect is that it has more people who are having trouble to adjust themselves correctly to the consensus in the beginning, which can be seen in figure 6. In the limit as t tends to infinity, however, it will closely resemble figures 3 and 5.

3.4 Comparison with real data

Figures 7 and 8 have been generated using data from Google Finance. [3] The share price of 100 companies with large market capitalization were taken from June 17 2007 and 255 weekly and daily values were extracted. The second difference from one day to another was calculated for a smoother change and the value 1 indicates a rise in share price compared to the previous time period whereas -1 indicates a fall from the previous time period. Although there is no way of telling whether the shares are owned by hipsters or Mainstream, the assumption is that the shareholders want to make profit from trading shares and would want to act against the market, hence they would need to behave in a hipster-like manner.

[3] Accessed on 21 June 2015. https://docs.google.com/spreadsheets/d/1rWodknu-TYlySFvb5p0HwrwMasewZOuGC67VdBEF6oc/edit#gid=0

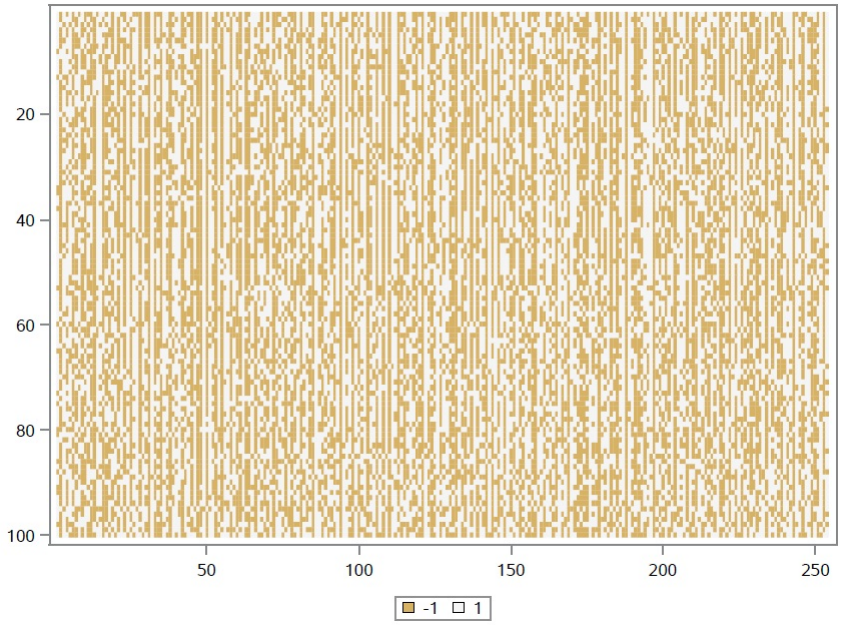


Figure 7: Large cap daily price movements

Figures 7 and 8 appears to have characteristics from both figure 2 and figure 3. Overall, there is a frequent switch between a rise and fall in price which closely resembles figure 2, but upon closer inspection we can see that when there is a rise or a fall in price from the previous day, it tends to happen for most firms on a given day. This is similar to figure 3, only with a much shorter delay of, for example, 2 or one. It could be that modern communication technologies make information exchange almost instantaneous on a daily basis. Figures 7 and 8 could possibly look different if the unit of time were to be smaller, say an interval of minute or seconds.

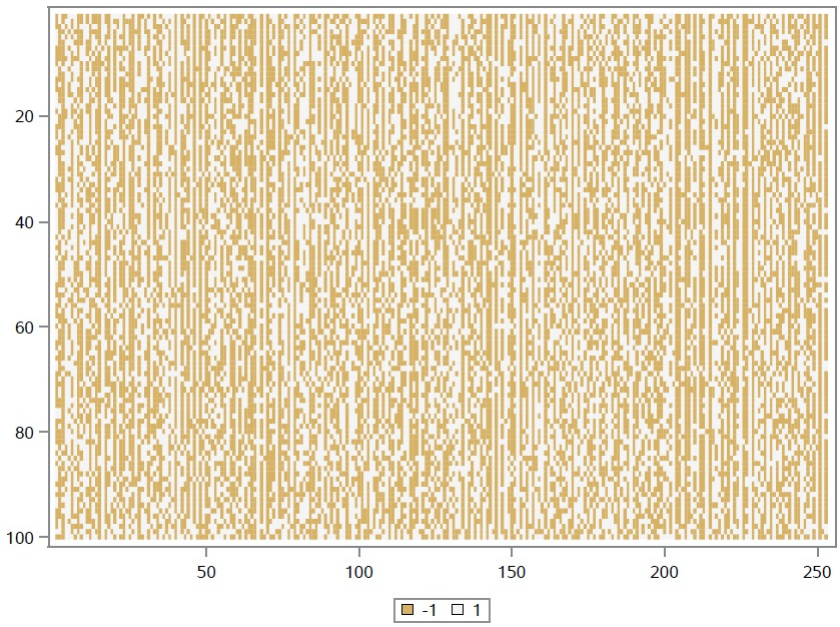


Figure 8: Large cap weekly price movements

With the simulated model, which had a delay value of 15, the duration of choices could clearly be seen. In the real data it is difficult to observe continuous rise or fall in share prices for more than 5 periods. However, if we look at Figure 8, between the time period of 150 and 200 there are less brown blocks than the time just after 200. Although we cannot see a clear cut synchronization as with the simulated model, there are some patterns that can be seen that are comparable to the simulated model, albeit with much more noise than the simulation. An important note to make here is that selecting companies based on other criteria, such as sectors and profitability, could lead to a more similar sample and thus a better output. Also, just as the economic cycles differ in length from time to time, adjusting the delay value could give an output that better resembles the real data. In order to see how closely the simulated data can resemble the real data, an attempt has been made in finding the minimum sum of squared differences between the real data and simulated data with different value of β. Every simulated data matrix, consisting of either -1 or 1, was compared with the real data matrix, which also contains -1 or 1 by counting the number of 1 for every time period and the sum of squared differences were measured. This simple approach did not yield a clear cut answer, as the minimum sum of squares seemed to be a local minimum instead of a global minimum. This indicated that parametrization is much more involved than simple heuristics.

3.5 L different states

So far we have only looked at the problem when there are only 2 outcomes available. In this case the computation and classification is relatively simple, since the principle is to classify all values above a certain level of threshold to one group and the rest into another group. It would be interesting to see what would happen if the individuals can choose from more than 2 outcomes (iPhone, Samsung and now Windows phone) but a new question arises: where do we place the second threshold? As a matter of fact, if the categories cannot be ranked, threshold will not be of any use. A simple way of overcoming this problem is through clustering algorithms. We let L denote the number of different states that are available. By following the k nearest-neighbour approach [3], where $k = n$ (we assume that every individual influences everyone else), we can model the behaviours of our hipsters and Mainstream when there are, say, 3 options available to them. For simplicity we will not consider influence and delay to play a role. The idea is that at every time period, every individual will count the frequency of each choice and hipsters will choose the least frequent

state, whereas Mainstream will choose the most frequent state. In the following miniature simulations with $n = 20$ and $t = 50$, we once again vary the hipster fraction to see what happens.

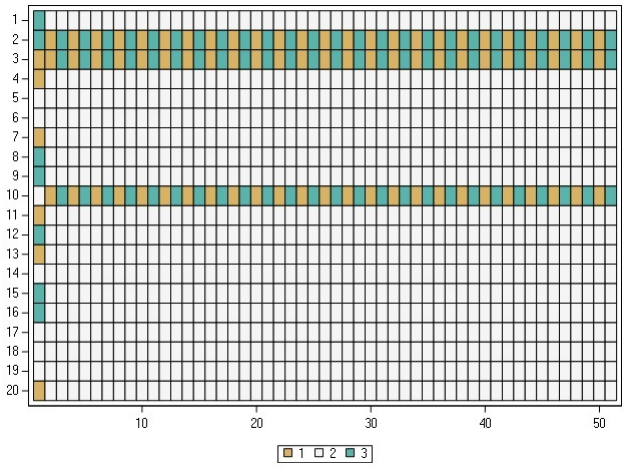


Figure 9: $L = 3$, Hipster fraction=15%

In figure 9 a tie occurred between groups 2 and 3, and group 2 was randomly selected as the majority group. As expected, Mainstream all chose group 2 (the most frequent state) and hipsters chose group 1 (the least frequent state) at $t = 2$. In the next time period where $t = 3$, Mainstream will be content with their choice of group 2, since it remains the majority. Now an interesting event takes place - since no individual chose group 3, it becomes the least selected group which is then the hipsters' preferred choice at the next time period and this pattern repeats itself indefinitely. Basically, hipsters take Mainstream decision as a given, eliminate it from their available options and choose to alternate between the remaining 2 groups.

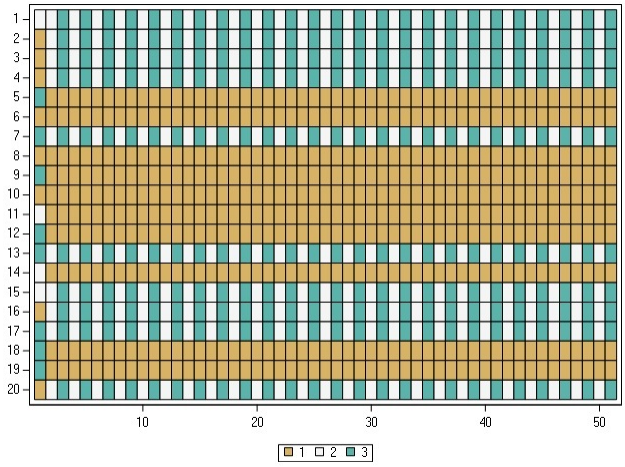


Figure 10: $L = 3$, Hipster fraction=50%

Even when the hipster fraction increases to 50% in Figure 10, it does not differ too much from Figure 9. It is possible that with a much larger n and delay, we can see a more random pattern in the behaviours of hipsters and Mainstream.

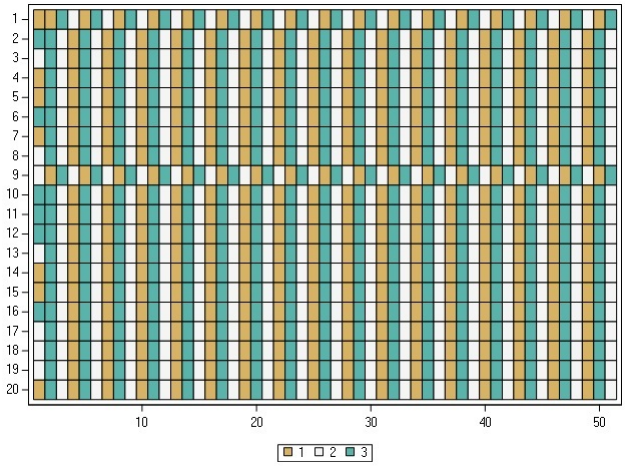


Figure 11: $L = 3$, Hipster fraction=90%

The 3 different states case presents us with yet another interesting result. In figure 11, the majority of the population comprises of hipsters. They see that group 3 was the least frequent state at $t = 1$ and choose group 3 at $t = 2$. The Mainstream decided that the majority is group 1 and chose group 1 at $t = 2$. Now the hipsters, seeing as their choices created the majority, turn to the least frequent group, which happened to be no one's choice, group 2. The Mainstream will then follow the majority, which is group 3, but they are falling behind the hipsters. Again this pattern is repeated indefinitely and with hipsters choosing the least frequent and Mainstream choosing the most frequent states.

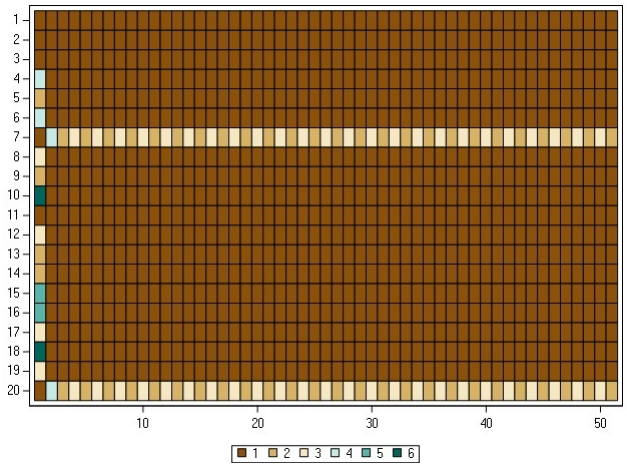


Figure 12: $L = 6$, Hipster fraction=15%

At this stage one may wonder what would happen for $L > 3$. Figures 12 and 13 have been produced to answer that question, where $L = 6$. The initial state contains all 6 categories and in Figures 12 and 13 the hipsters and Mainstream once again behave in their usual manner. We can see that in the long run they only choose among 3 different states as time changes and they behave similarly to their counterparts, Figure 9 and figure 11 respectively. It is unclear whether the other 3 groups are not resurfacing due to the crudeness of the model or because they are actually forgotten by the population. The reasons attributable to the disappearance of the remaining 3 groups is a topic for further study.

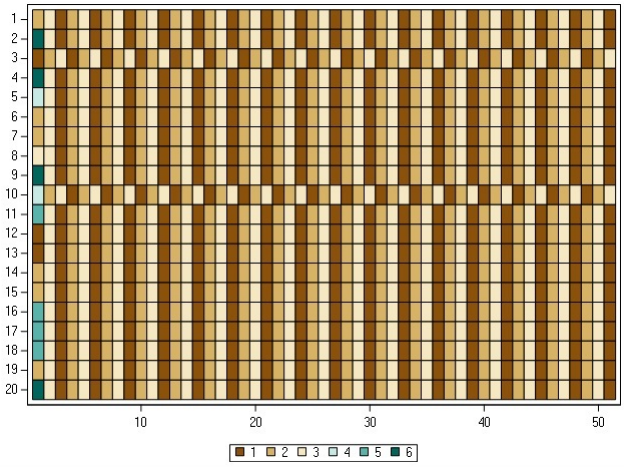


Figure 13: $L = 6$, Hipster fraction=90%

4 Conclusion

In this paper we looked at how this theoretical "Hipster Effect" is developed and used it to see the different outcomes under different parameters of hipster fraction and delay. The sigmoid functions have also been changed and we saw that with a large enough β value, they all show a cyclical pattern when the majority of the population consists of hipsters. A heuristic approach was taken in investigating the effect of the hipster fraction and β values. A comparison of theory with real data was done and saw that there are still more factors that need to be considered for the simulation to adequately model the real data. Finally, an extension of the binary model to L number of states using clustering approach have been discussed. A more detailed study of dynamic delay as a function of time, changing the influence matrix realistically so that few individuals (such as politicians and celebrities) influence people more than others as well as a mathematical technique to estimate the beta value that results in the optimal model are but few things that can further be investigated in this new and interesting field.

References

[1] Christopher Chatfield and Gerald J Goodhardt. The beta-binomial model for consumer purchasing behaviour. In *Mathematical Models in Marketing*, pages 53–57. Springer, 1976.

[2] Nikhil Chopra and Mark W Spong. Output synchronization of nonlinear systems with time delay in communication. In *Decision and Control, 2006 45th IEEE Conference on*, pages 4986–4992. IEEE, 2006.

[3] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin, 2001.

[4] Malcolm Gladwell. *The tipping point: How little things can make a big difference*. Little Brown, 2006.

[5] Kerson Huang. *Introduction to Statistical Physics*. CRC Press, 2001.

[6] John G Kemeny and J Laurie Snell. *Markov Chains*. Springer, 1976.

[7] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015.

[8] Jonathan Touboul. The hipster effect: When anticonformists all look the same. *arXiv preprint arXiv:1410.8001*, 2014.

[9] Jake Vanderplas. The hipster effect: An ipython interactive exploration, Nov 2014. https://jakevdp.github.io/blog/2014/11/11/the-hipster-effect-interactive/.