Statistical properties of forward selection regression estimators

by

Nicolene Magrietha Thiebaut

Submitted in partial fulfilment of the

requirements for

the degree

MSc

In the Faculty of Natural & Agricultural Sciences

University of Pretoria

Pretoria

April 2011

Promotor: Prof F.E. Steffens

==========

DECLARATION

I*, Nicolene Magrietha Thiebaut *declare that the dissertation, which I hereby submit for the degree MSc at the University of Pretoria, is my own work and has not previously been submitted by me for a degree at this or any other tertiary institution.

***Signature:*_____**

***Date:* _____**

ACKNOWLEDGEMENTS:

I wish to record my sincere thanks and appreciation to the following persons and institutions for their contributions to this dissertation:

The Agriculture Research Council (ARC) for allowing me to further my studies and for offering me a bursary.

Prof. F. E Steffens from the Department of Statistics of the University of Pretoria, prof. Crowther and the other lecturers for their assistance and guidance.

My friend Hartmut, my daughter Nicole, family and other friends for their patience and support.

All the contributions are highly appreciated.

ABSTRACT:

In practice, when one has many candidate variables as explanatory variables in multiple regression, there is always the possibility that variables that are important determinants of the response variable might be omitted from the model, while unimportant variables might be included. Both types of errors are important, and in this dissertation it is attempted to quantify the probabilities of these errors.

A simulation study is reported in this dissertation. Different numbers of variables, i.e. p= 4 to 20 are assumed, and different sample sizes, i.e. n=0.5p, p, 2p, 4p. For each p the underlying model assumes that roughly half of the independent variables are actually correlated with the dependant variable and the other half not. The noise is $\varepsilon \sim N(0, \sigma^2)$, where σ^2, is set fixed. The data was simulated 10000 times for each combination of n and p using known underlying models and ε randomly selected from of a normal distribution.

For this investigation the *full model* and *forward selection regression* are compared. The mean squared error of the estimated coefficient $\hat{\beta}$(p) is determined from the true β of each n and p set. A full discussion, as well as graphs, is presented.

LIST OF CONTENT

Chapter 1

INTRODUCTION

When selecting variables in regression, certain procedures are followed. If one wishes to establish a linear regression equation for a particular response Y (in terms of the predictor variables X_1, X_2, X_3…X_p and furthermore if Z_1, Z_2, Z_3 ….Z_k, were all functions of one or more of the x's as well as the whole set of functions, such as logarithms, inverses etc.), then the following two generally opposed criteria shall be considered:

1) To make the equation useful for prediction purposes, as many predictors as possible should be included – this normally results in a small bias so that reliable fitted values can be determined.

2) To keep the variance of the predictions as small as possible (for a certain p, i.e. the number of variables, and n, i.e. the number of observations) to minimize costs involved in obtaining information on large numbers of predictors and the subsequent monitoring of them, the number of predictors must be as small as possible to be cost effective and easily interpretable.

The different methods for "Selecting the best regression equation" are illustrated, followed by an overview of the full regression model and forward selection. A description of the program, i.e. programmed in SAS IML, and information about the parameters chosen for the study are also given. The final part of this dissertation comprises a summary and conclusion of the study.

Chapter 2

SELECTING THE "BEST" REGRESSION EQUATION.

The practical compromise between the extremes described in 1) and 2) above is normally regarded as the best regression equation. There are no definite procedures for this selection process. Although many procedures have been suggested, the methods commonly used in choosing the *best regression model* are the following[1]:

a) The full model (includes all predictors)
b) The best subset regression using: R^2 (adjusted) and C_p.
c) Forward regression (there are variations on this method)
d) Stepwise regression
e) Backward elimination
f) Variations on previous methods.

These methods should be regarded as exploratory analysis (not providing a definite proof that the selected model is a true reflection of the real life situation).

When using the above methods there can be problems such as:

[1] This discussion refers to Draper and Smith on p.327.

- Possible multicollinearity among the explanatory variables.
- Too few observations
- Several possible models to choose from.

In the following section an overview of the above-mentioned methods, as well as an example is given. The full regression model is used for comparison in this study.

a) The full multiple linear regression model

All the terms are included in the model. In matrix notation, the model is given by:

$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ where:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} 1 & x_{11} \ldots . & x_{1p} \\ 1 & x_{21} \cdots & x_{2p} \\ \vdots & \vdots & \vdots \\ 1 & x_{n1} \cdots & x_{np} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

y is an n x 1 vector of observations, **X** is an n x (p+1) matrix, $\boldsymbol{\beta}$ is a (p+1) x 1 vector of the regression coefficients and $\boldsymbol{\varepsilon}$ is an n x 1 vector of random errors.

The aim is to find the vector of the least-square estimators $\widehat{\boldsymbol{\beta}}$**,** that minimizes:

$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \varepsilon_i^2 = \boldsymbol{\varepsilon}`\boldsymbol{\varepsilon} =$ **(y-$\boldsymbol{X\beta}$)`(y-$\boldsymbol{X\beta}$)**

The least-square estimators must satisfy

$\frac{\partial S}{\partial \beta}|_{\hat{\beta}} = 0$ which simplify to

X`X$\widehat{\boldsymbol{\beta}}$=X`y

which are the least square normal equations.

The least-square estimator of $\boldsymbol{\beta}$ is then:

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}`\boldsymbol{X})^{-1}\boldsymbol{X}`\boldsymbol{y}$$

provided that the inverse matrix $(\boldsymbol{X}`\boldsymbol{X})^{-1}$ exists. For this reason the full linear model cannot be calculated where n < p.

t-test and F-test have been used in this study and are the following:

Test: $\beta_j = 0$ ➔ $t = \frac{(\widehat{\beta}_j)}{\sqrt{\widehat{\mathbf{var}}(\widehat{\beta}_j)}}$ (0 < j < p) and where t $\sim t(n-k-1)$

➔ $F = \frac{(\widehat{\beta}_j)^2}{\widehat{\mathbf{var}}(\widehat{\beta}_j)} = t^2$ where F $\sim F(1, n-k-1)$

where $\text{var}(\widehat{\boldsymbol{\beta}}) = \boldsymbol{\sigma^2}(\boldsymbol{X}`\boldsymbol{X})^{-1}$**,** with $\sigma^2 = \text{var}(y_i)$

$$\widehat{var}(\widehat{\boldsymbol{\beta}}) = \boldsymbol{s^2}(\boldsymbol{X`X})^{-1} \text{ and } (\boldsymbol{s^2}(\boldsymbol{X`X})^{-1})_{j+1,j+1} = \widehat{var}(\widehat{\boldsymbol{\beta}}_j)$$

with $\boldsymbol{s^2}$ the mean-square about regression (MSE)

Example:

For illustration purposes x_1, x_2, x_3 and x_4 are generated each from a normal distribution where $x_i \sim N(0,1)$ n=30 is used, and $y_i \sim N(0,1)$.

The data generated was the following:

y	x0	x1	x2	x3	x4
0.947642	1	0.421586	-0.12949	1.379146	-0.47979
0.778414	1	0.833397	0.07267	0.552539	-0.91775
-1.07774	1	0.266575	-0.75721	0.95104	-1.41633
0.139451	1	-0.50984	-0.15961	0.306478	0.046713
1.274163	1	-0.65201	1.328503	-0.50505	0.971239
0.537372	1	0.364805	-0.76114	0.08087	0.25374
0.431233	1	0.577395	-0.84916	-0.68987	0.635711
-0.05728	1	-0.84625	-0.17513	-0.18889	-0.61391
0.768738	1	-0.65813	1.096219	0.073256	0.176281
1.110896	1	1.026968	-0.25972	0.084936	1.326771
-1.39961	1	-1.77678	-1.17105	1.738608	-0.00173
0.287331	1	0.623366	-0.48989	-0.27325	0.641921
0.889727	1	1.418482	-0.46166	-0.00976	0.982293
0.186761	1	0.004334	1.425703	0.30741	-0.80675
-0.06384	1	0.104088	-0.33449	0.156741	0.651072
0.006442	1	1.170835	-0.57312	1.083025	-1.39766
1.402955	1	-0.66119	0.035194	0.790913	0.494096
1.661528	1	-0.8181	-0.71455	0.965787	1.797969
-0.1518	1	-1.17774	-1.38733	0.264769	-2.49698
1.130318	1	0.301525	1.516795	-0.87938	-0.54283
0.757585	1	-1.20754	0.000857	-2.4172	0.565533
-0.49201	1	-2.3877	0.091767	0.384804	0.455261
2.209267	1	-0.39957	0.050983	-0.08351	-0.02853
0.892306	1	-1.38631	1.980733	1.216849	-0.00048
0.237334	1	-0.59445	0.200447	-1.60794	-0.76354
1.935701	1	1.834607	0.355468	-0.74685	1.479613
0.756728	1	2.242614	0.431939	-0.17352	-0.2563
0.490658	1	2.365015	-0.63934	0.820001	1.261527
0.519579	1	-0.13022	0.802198	-0.03034	0.424951
0.742926	1	-1.29626	0.232773	-0.56458	1.619795

In this example the following output is obtained:

The correlation matrix is:

Correlations

		Y	x1	x2	x3	x4
y	Pearson Correlation	1	.266	.381*	-.279	.459*
	Sig. (2-tailed)		.155	.038	.136	.011
	N	30	30	30	30	30
x1	Pearson Correlation	.266	1	-.105	-.009	.097
	Sig. (2-tailed)	.155		.580	.962	.611
	N	30	30	30	30	30
x2	Pearson Correlation	.381*	-.105	1	-.214	.074
	Sig. (2-tailed)	.038	.580		.256	.699
	N	30	30	30	30	30
x3	Pearson Correlation	-.279	-.009	-.214	1	-.176
	Sig. (2-tailed)	.136	.962	.256		.353
	N	30	30	30	30	30
x4	Pearson Correlation	.459*	.097	.074	-.176	1
	Sig. (2-tailed)	.011	.611	.699	.353	
	N	30	30	30	30	30

*. Correlation is significant at the 0.05 level (2-tailed).

The R^2 value is: 0.42 which gives an indication of a linear trend.

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.647[a]	.418	.325	.64681

The F - probability of the ANOVA is: 0.007, which indicates statistically a significant model.

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	7.519	4	1.880	4.493	.007[a]
	Residual	10.459	25	.418		
	Total	17.978	29			

a. Predictors: (Constant), x4, x2, x1, x3

b. Dependent Variable: y

The coefficients marked are statistically significant at the 5% level.

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	*.529	.121		4.387	.000
	x1	.180	.105	.265	1.717	.098
	x2	*.339	.151	.352	2.238	.034
	x3	-.120	.143	-.133	-.842	.408
	x4	*.306	.124	.384	2.464	.021

a. Dependent Variable: y

The equation of the full model is then:

$y = 0.529 + 0.180x_1 + 0.339x_2 - 0.120x_3 + 0.306x_4$.

The full model thus includes non-significant terms.

b) **The best subset regression using: R^2 (adjusted) and C_p**

- This is a complicated regression technique. It requires the fitting of every possible regression equation that involves Z_0 plus any number of variables Z_1…..Z_r, where Z_0 is the intercept term. In this example the transformation and other combinations of the Z terms will not be fitted.
- The three criteria most often used are the following:
  - The value of R^2 achieved by the least squares fit.
  - The value of the s^2, the residual mean square.
  - The C_p statistic.

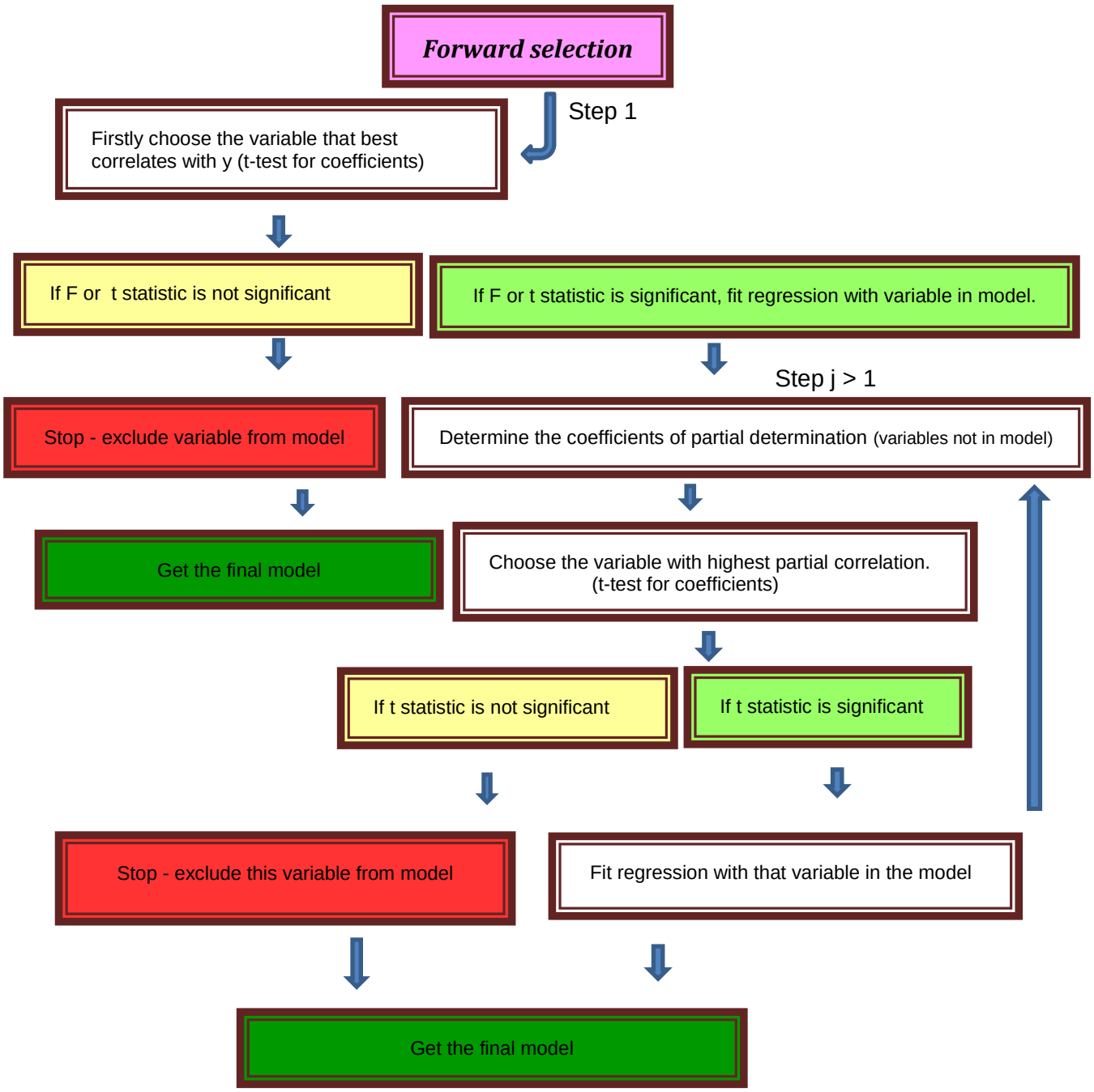
These criteria are related to each other.

All three the above criteria are used to obtain the best regression equation. The fitting of regression equations that involve more predictor variables than are necessary to obtain a satisfactory fit to data is called over-fitting. With these criteria over-fitting can be prevented.

c) Forward selection

- Firstly choose the variable most highly correlated with y;
- Fit the regression and calculate the residuals;
- Choose the variable best correlated with the residuals (if significant);
- Repeat the above sequence until all the variables have been selected, or until there are no more significant correlations with the residuals.

The following flowchart illustrates this selection.



Example continued:

The R^2 =0.332 of y with the variables x_2 and x_4 in the model indicates that there is some linear trend.

Model Summary[c]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.459[a]	.211	.183	.71183	.211	7.480	1	28	.011
2	.576[b]	.332	.283	.66689	.121	4.901	1	27	.035

a. Predictors: (Constant), x4

b. Predictors: (Constant), x4, x2

c. Dependent Variable: y

The regression model with x2 and x4 in the model is significant (p=0.004).

ANOVA[c]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.790	1	3.790	7.480	.011[a]
	Residual	14.188	28	.507		
	Total	17.978	29			
2	Regression	5.969	2	2.985	6.711	.004[b]
	Residual	12.008	27	.445		
	Total	17.978	29			

a. Predictors: (Constant), x4

b. Predictors: (Constant), x4, x2

c. Dependent Variable: y

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.512	.131		3.903	.001
	x4	.366	.134	.459	2.735	.011
2	(Constant)	.506	.123		4.119	.000
	x4	.346	.126	.433	2.748	.011
	x2	.336	.152	.349	2.214	.035

ANOVA[c]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3.790	1	3.790	7.480	.011[a]
	Residual	14.188	28	.507		
	Total	17.978	29			
2	Regression	5.969	2	2.985	6.711	.004[b]
	Residual	12.008	27	.445		
	Total	17.978	29			

a. Predictors: (Constant), x4

b. Predictors: (Constant), x4, x2

a. Dependent Variable: y

The final model with the forward regression is then:

$$y = 0.506 + 0.336x_2 + 0.346x_4$$

(The normality plot and the y-fitted versus residuals plot seem to be more acceptable than those of the full model.)

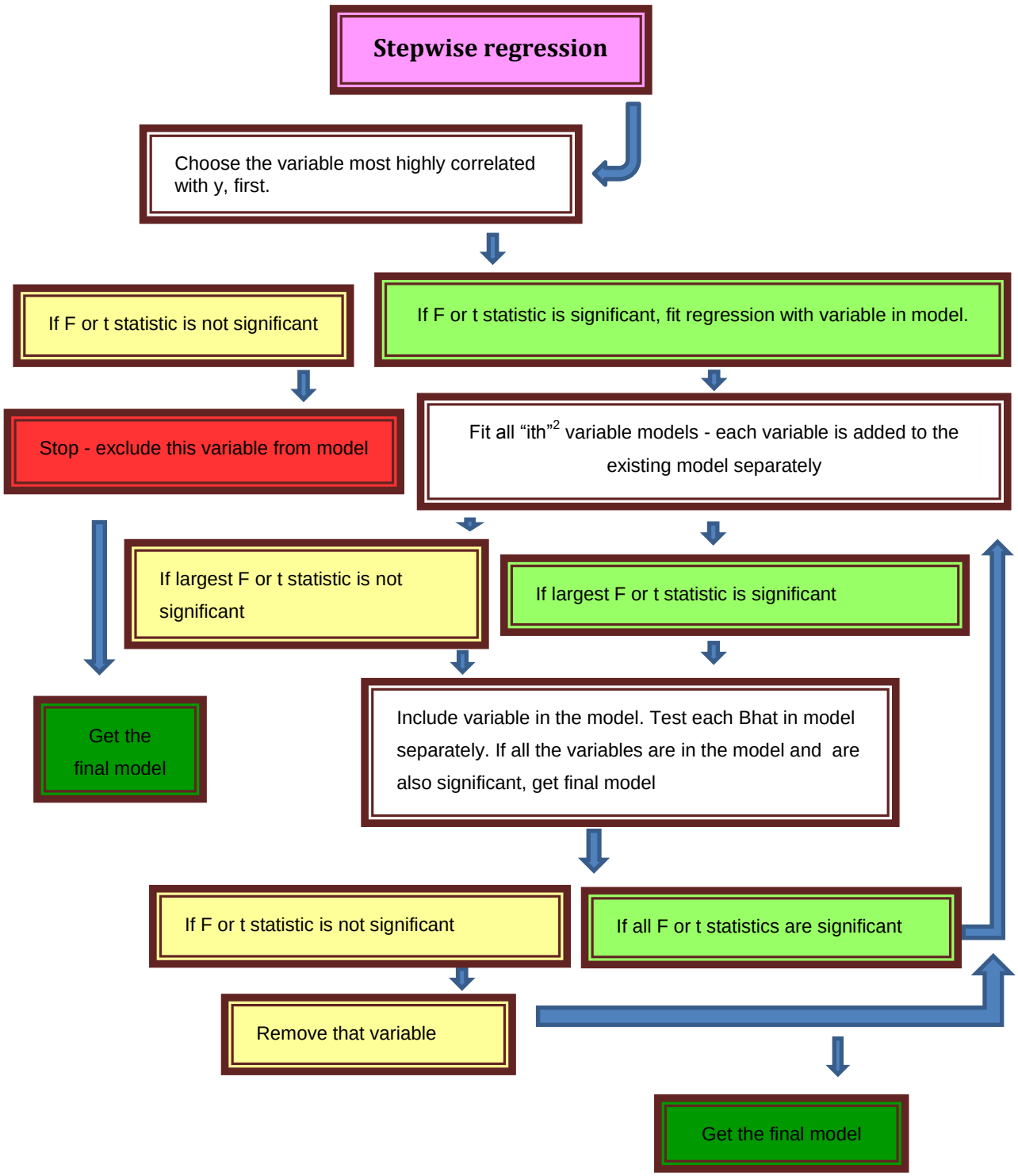
d) Stepwise regression.

Stepwise regression is an improved version of forward regression, which permits re-examination at every step of the variables incorporated in the model in previous steps. A variable that entered at any early stage may become superfluous at a later stage, because of its relationship with the other variables in the model at that particular moment.

The following steps are taken:

- The regression starts in the same manner as in forward selection;
- A variable can be added or removed at any stage - it is removed when the regression coefficient is no longer “significant”;
- The criterion for removing a variable must be stricter than for entering it.

The following flowchart illustrates this selection:



[2] The i 'th variables are fitted. If 2 variables are in the model, each variable is added separately to get a range of 3 variables models, etc.

Example continued:

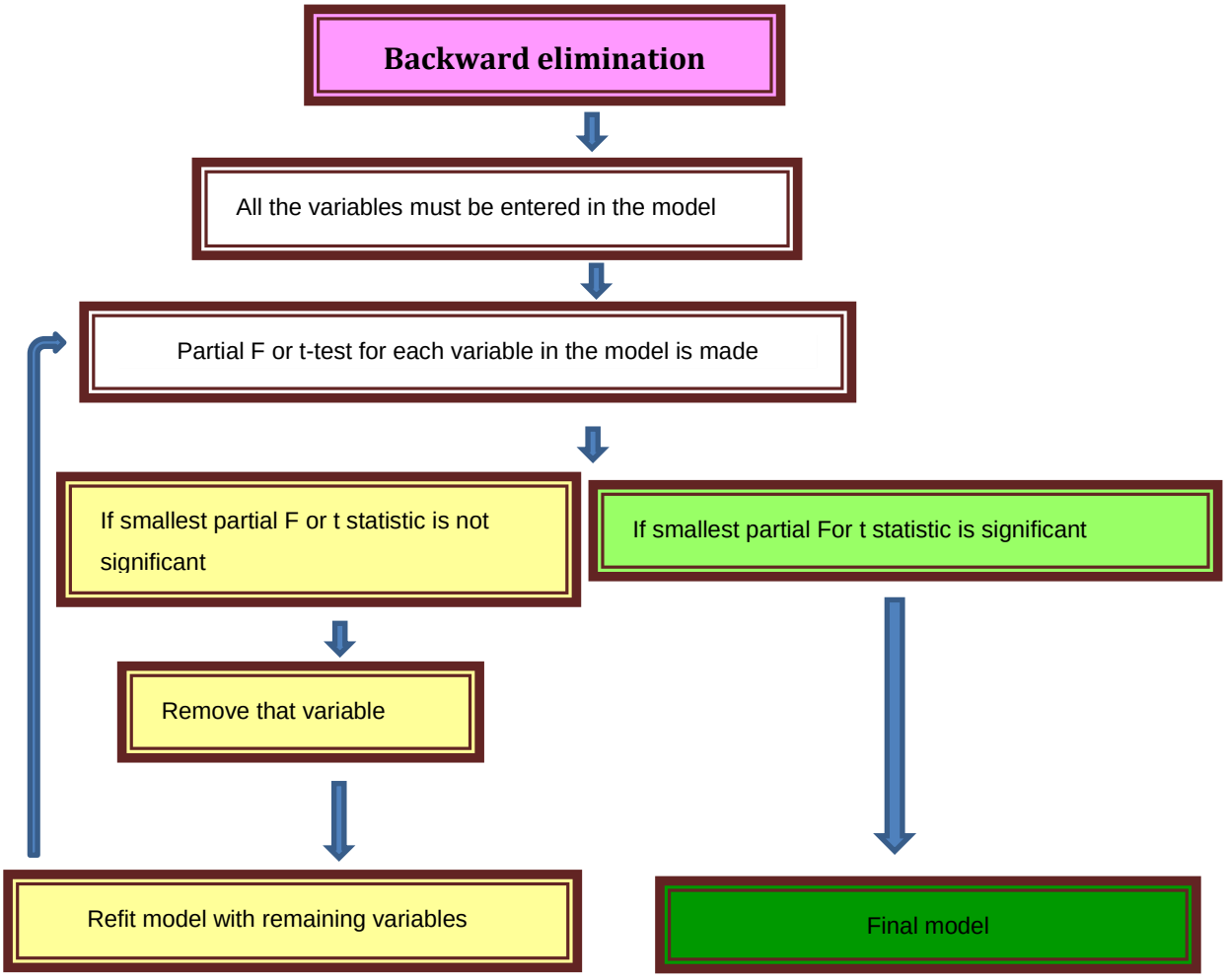
Stepwise regression gives the same results as the forward regression model. X4 and then x1 are included in the model and no variables are removed.

The final model with stepwise regression is: $y = 0.506 + 0.336.x2 + 0.346.x4$.

e) Backward elimination

- Start with the full model, i.e. all the predictor variables must be within the model – this can only be done when n > p, otherwise X'X will be singular;
- Eliminate the least significant variable and recalculate the regression;
- All the remaining regression coefficients must be significant when the selection stops.

The following flowchart illustrates this selection



Example (continued):

There is no significant change in the R^2 value when x_3 is taken out of the model. The final R^2 value is 0.402.

Model Summary[c]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics				
					R Square Change	F Change	df1	df2	Sig. F Change
1	.647[a]	.418	.325	.64681	.418	4.493	4	25	.007
2	.634[b]	.402	.333	.64317	-.016	.709	1	25	.408

a. Predictors: (Constant), x4, x2, x1, x3 b. Predictors: (Constant), x4, x2, x1 c. Dependent Variable: y

The ANOVA with x1, x2 and x4 is statistically significant. (Fprob=0.004)

ANOVA[c]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	7.519	4	1.880	4.493	.007[a]
	Residual	10.459	25	.418		
	Total	17.978	29			
2	Regression	7.222	3	2.407	5.820	.004[b]
	Residual	10.755	26	.414		
	Total	17.978	29			

a. Predictors: (Constant), x4, x2, x1, x3 b. Predictors:(Constant),x4, x2,x1 c. Dependent Variable:

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.529	.121		4.387	.000
	x1	.180	.105	.265	1.717	.098
	x2	.339	.151	.352	2.238	.034
	x3	-.120	.143	-.133	-.842	.408
	x4	.306	.124	.384	2.464	.021
2	(Constant)	.514	.119		4.335	.000
	x1	.181	.104	.267	1.740	.094
	x2	.365	.147	.379	2.477	.020
	x4	.323	.122	.405	2.650	.014

a. Dependent Variable: y

The removal of variables was done if the significant value were > 0.1. That is why x1 is not removed out of the model.

The final model is then:

$y=0.514 + 0.181x_1 + 0.365x_2 + 0.323x_4$.

f) Variations on previous methods.

Variations of the methods above can also be used.

Comment on example: This is not a very good model. The R^2 is around 0.40, which is not good for fitting a linear model. This example was only used for illustrating purposes to show the different methods.

Chapter 3

LITERATURE STUDY

There is no unique method in finding the "best regression equation". In this chapter a few discussions will be made in finding *the best probable model,* taking the previous mentioned multiple regression analysis methods into account, and by focussing on forward selection and some other discussions from the literature.

It is usually assumed that the correct functional specification of the regressors is known (e.g. $1/x_1$, $\ln x_2$) and that no outliers or influential observations are present. If outlier observations were present, it must be removed and the variable selection process must be repeated.

When doing the selection of variables in regression analysis, some important aspects to consider will be discussed. Variable selection procedures should be used by the analyst as methods to explore the structure of the data. There is usually not a single best equation but rather several good ones.

In this study the focus is on the accuracy of a model using forward selection in multiple, linear regression.

A few important concepts used in selecting the best possible linear model, as well as concepts that influence the error in a regression model are discussed.

- ## ASSUMPTIONS ABOUT THE ERROR TERM ARE THE FOLLOWING:

(The assumptions will influence the accuracy of the regression analysis).

1) The residuals must have a zero mean, i.e. $E(\varepsilon_i) = 0$.
2) Homoscedasticity, i.e. $Var(\varepsilon_i) = \sigma^2$ for all i's (constant variance).
3) Normality: for statistical inference (confidence intervals and hypothesis testing), assuming that $\varepsilon_i \sim N(0, \sigma^2)$.
4) Independence: All pairs ε_i and ε_j are (stochastically) independent.

The above-mentioned assumptions are met in this study.

- ## MULTICOLLINEARITY

Another serious problem that may substantially impact the usefulness of a regression model is multicollinearity. This implies near–linear dependence among the regressors. Multicollinearity can have serious effects on the estimates of the regression coefficients and on the general applicability of the estimated model.

A good linear model results when the y (dependent) variables are highly correlated with the x (independent) variables, but with as little as possible multicollinearity among the x variables.

From the example in chapter 2, none of the x variables were significantly correlated, and therefore did not influence the model -see correlation matrix below. X_2 and X_4 were significantly correlated with the dependent variable.

The correlation matrix is:

Correlations

		Y	x1	x2	x3	x4
y	Pearson Correlation	1	.266	.381*	-.279	.459*
	Sig. (2-tailed)		.155	*.038	.136	*.011
	N	30	30	30	30	30
x1	Pearson Correlation	.266	1	-.105	-.009	.097
	Sig. (2-tailed)	.155		.580	.962	.611
	N	30	30	30	30	30
x2	Pearson Correlation	.381*	-.105	1	-.214	.074
	Sig. (2-tailed)	.038	.580		.256	.699
	N	30	30	30	30	30
x3	Pearson Correlation	-.279	-.009	-.214	1	-.176
	Sig. (2-tailed)	.136	.962	.256		.353
	N	30	30	30	30	30
x4	Pearson Correlation	.459*	.097	.074	-.176	1
	Sig. (2-tailed)	.011	.611	.699	.353	
	N	30	30	30	30	30

*. Correlation is significant at the 0.05 level (2-tailed).

Multicollinearity was in this example not a problem, but in many practical cases it may become a problem.

The following collinearity diagnostics are obtained from one of the simulations of this study where n=8 and p=4.

Collinearity Diagnostics[a]

Model	Dimension	Eigenvalue	Condition Index	Variance Proportions				
				(Constant)	x4	x3	x2	x1
1	1	2.338	1.000	.02	.05	.03	.03	.03
	2	1.190	1.402	.35	.00	.05	.04	.03
	3	.906	1.607	.03	.16	.15	.03	.07
	4	.438	2.312	.48	.38	.15	.09	.05
	5	.129	4.260	.12	.41	.61	.80	.81

a. Dependent Variable: y

The condition index is the square root of the ratios of the largest eigenvalue to each successive eigenvalue. A condition index greater than 15 indicates a possible problem and an index greater than 30 suggests a serious problem with collinearity.

All the condition index values are smaller than 15. This implies that there will not be a collinearity problem.

- **THE VARIANCE FOR $\hat{\beta}$**

To obtain the narrowest possible confidence interval for β_j, (and the maximum power in testing the hypothesis about β_j, the variance for β_j must be minimized).

The variance for the estimated $\hat{\beta}_j$, is used for the t-test and F-test and is the following:

$\text{var}(\hat{\beta}) = \sigma^2(X`X)^{-1}$,

$\sigma^2 = \text{var}(y_i)$

$\widehat{var}(\hat{\boldsymbol{\beta}}) = \boldsymbol{s^2}(\boldsymbol{X`X})^{-1}$ and

$(\boldsymbol{s^2}(\boldsymbol{X`X})^{-1})_{i+1,i+1} = \widehat{\boldsymbol{var}}(\hat{\boldsymbol{\beta}}_i)$

with $\boldsymbol{s^2}$ the mean-square about regression (MSE)

- **THE MEAN-SQUARE ERROR FOR $\hat{\beta}$.**

In statistics, the mean square error or MSE of an estimator is one of many ways to quantify the difference between an estimator and the true value of the quantity being estimated. MSE is a risk function, that corresponds to the expected value of the *squared error loss or quadratic loss.*

This is one of the important concepts that is used in the study, i.e. determining the error in multiple forward regression analysis. The mean-squared error is intimately related to the prediction accuracy. The mean-squared error for the full- as well as the forward model is determined for the different values of n and p.

The MSE is the second moment (about the origin) of the error, and thus incorporates both the variance of the estimator and the bias. For an unbiased estimator, the MSE is the variance.

The mean-squared error of the estimator $\hat{\beta}$ in estimating β is[3]:

$$MSE(\hat{\boldsymbol{\beta}}) = E[(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^2] = var(\hat{\boldsymbol{\beta}}) + [E(\hat{\beta}) - \beta]^2$$

$$= var(\hat{\boldsymbol{\beta}}) + [Bias(\hat{\beta}, \beta)]^2$$

$$= (\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})`(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})/p \quad [4] \qquad (1)$$

The red term is the variance, while the blue term is the squared bias. The format of the last part of the equation is used in the program of this study for the full and forward model.

The Gauss-Markov theorem implies that the least squares estimator has the smallest mean-square error for all linear estimators with no bias. However, there may exist a biased estimator with a smaller mean-squared error. Such an estimator would trade a little bias for a larger reduction in variance. Biased estimates are commonly used. Any method that shrinks or sets to zero some of the least square coefficients may result in a biased estimate.

The full linear regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ ….(2) can be written as

$$\boldsymbol{y} = \boldsymbol{X_k\beta_k} + \boldsymbol{X_{p-k}\beta_{p-k}} + \boldsymbol{\varepsilon} \quad \text{….(3)}$$

where the equation (3) is the linear equation of the full model and $\boldsymbol{X_k\beta_k}$ **is** the reduced model.

There are two reasons[5] why one may not be satisfied with the least squares estimates of the full model (eq. 2), with $\hat{\boldsymbol{\beta}} = (\boldsymbol{X`X})^{-1}\boldsymbol{X`y}$, namely:

- The first is prediction accuracy: the least square estimates often have low bias, but large variance. Prediction accuracy can sometimes be improved by shrinking

[3] http://en.wikipedia.org/wiki/Mean_square_error

[4] Formula used in program

[5] Referring to Hastie et al - *The elements of Statistical learning.* p. 57

or setting some coefficients to zero. By doing so, a small amount of bias is sacrificed to reduce the variance of the predicted values, and hence may improve the overall prediction accuracy[5].

Comment: The least square estimates of the parameters in the subset model have a smaller mean square error than the corresponding parameter estimates from the full model, when the deleted variables have regression coefficients that are smaller than the standard errors of their estimates in the full model[6].

- The second reason is interpretation: With a large number of predictors, it is desired to determine a smaller subset that exhibits the strongest effects. In order to get more clarity, one may be willing to sacrifice some of the small details[5].

From a more pragmatic point of view, most models are distortions of the truth, and hence are biased; picking the right model amounts to creating the right balance between bias and variance.

• COEFFICIENT OF PARTIAL DETERMINATION

The coefficient of determination R^2 is a measure of the combined effects of all independent variables $X_1 \ldots X_p$, in the regression model in reducing the total variability. When considering whether or not to add another independent variable to the regression model, a different measure is needed. Such a measure is the coefficient of partial determination, which shows the marginal effect of a single variable associated with the regression model[7].

For the regression model (2) with two independent variables, the coefficient of partial determination between y and x_1, given x_2, already in the model is defined by:

$$r^2_{Y1,2} = \frac{SSE(X_2) - SSE(X_1, X_2)}{SSE(X_2)} \quad [8] \quad (4)$$

where SSE is the error sum of squares.

This formula can be extended to the multivariate case. The square root of the coefficient of partial determination is called the *coefficient of partial correlation*. This and the extended formula are used in the program for the study.

The maximum r is used within each step of the forward regression analysis, determining the next most important variable to be allowed into the model.

---

[6] Montgomery. 2006. *Introduction to linear regression analysis - Fourth Edition*. p. 264.
[7] Wasserman. 1993. *Applied Statistics, 4th Edition*. - on this terminology (Chapter 20).
[8] See program in the appendix how the formula is used

• FORWARD SELECTION VERSUS OTHER SUBSET SELECTION PROCEDURES

Forward regression analysis is a *greedy algorithm* that produces a nested sequence of modelsl[9]. In this sense it might seem sub-optimal compared to best-subset regression. However, there are several reasons why it might be preferred:

- *Computational:* for a large number of variables, one cannot compute the best subset sequence, but can always compute the forward selection (even when p > n).

- *Statistical:* a price is paid in variance for selecting the best subset of each size; forward selection is a more constrained search, and will have lower variance , but perhaps more bias.

Comment: Bendel and Afifi (1977) have shown in a simulation that forward selection with an F-to-enter statistic at nominal $\propto = 0.15$ produces a smaller population mean square residual than other more complex stopping rules.

The next part is also an abstract out of Hastie et al - *The elements of Statistical learning.* - p. 59.

Hastie et al.(2009) have a comparison of four subset-selection techniques on a simulated linear regression analysis dataset. They graphically pictured the mean-squared error of the estimated coefficient $\widehat{\boldsymbol{\beta}}(\boldsymbol{k})$ at each step from the true $\boldsymbol{\beta}$ where k is the number of variables chosen from the subset model ($k \in \{0,1,2 \dots p\}$). He simulated n=300 observations, p=31 variables from a standard Gaussian distribution for 50 runs. 10 of the coefficients for each case were drawn at random from a normal N(0,0.4) distribution, with remaining coefficients equal to zero, and $\boldsymbol{\varepsilon} \sim N(0,6.25)$.

[9] Referring to Hastie et al - *The elements of Statistical learning.* p. 58

The following graph has been obtained:

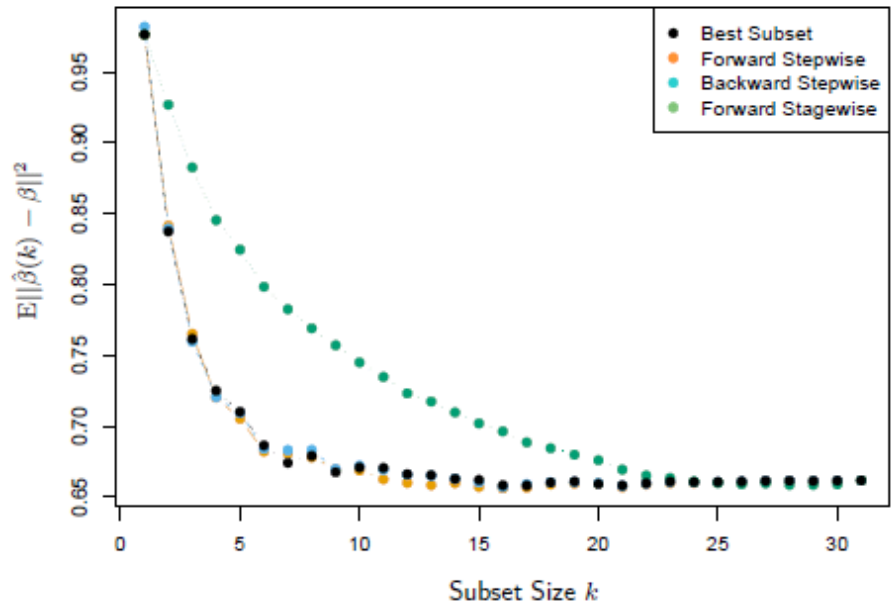


Figure A

Figure A shows the results of a small simulation study to compare best-subset regression with the simple alternative forward and backward selection. There are n=300 observations on p=31 standard Gaussian variables, with pairwise correlations that are all equal to 0.85. For 10 of the variables, the coefficients are drawn at random from a $N(0,0.4)$ distribution, whilst the rest are zero. The noise $\sim N(0,6.25)$ results in a signal-to-noise ratio of 0.64. Results are averages over 50 simulations. In the graph the mean-squared error of the estimated coefficient $\hat{\beta}(k)$ at each step from the true $\boldsymbol{\beta}$ is shown.

The performance of the different subsets is very similar, as is often the case. Included in the figure is forward regression. The forward stagewise regression takes longer to reach minimum error.

On the other hand, backward elimination starts with the full model, and sequentially deletes the predictor that has the least impact on the fit. Backward selection can only be used when n > p, while forward selection can always be used.

Different from this study, Hastie graphed the mean-squared error of the number of subset variables versus the mean-squared error at *each* stage of the selection.

In this study the mean-squared error is plotted against p (number of variables), n (numbers of observations) and k (the number of variables selected) of each n versus p case, discovering the best n versus p ratios as well as the best number of n and p's for the forward selection regression analysis.

Since these different procedures are widely used, knowledge of the distribution of the sample R^2 would be helpful in each case[10]. Berk (1978) has shown, however, that differences between procedures might not be significant in practice.

• OTHER STUDIES ON FORWARD SELECTION

A study by Bendel(1977) slightly differs from this study. He studies the *Comparison of Stopping rules in Forward Stepwise Regression.* In his paper he uses the unconditional mean-square error of prediction as a criterion for comparing stopping rules used with the forward stepwise selection procedure in multivariate normal samples, based on simulations of 48 population correlation matrices. The C_p statistic, "F to enter" ($0.15 < \propto < 0.25$), a rule which minimizes the sample criterion, and one which sequentially tests the equality of the population criterion ($0.25 < \propto < 0.35$) were used. For these rules, the criterion seldom differs by more than 3%, although there are considerable differences between these and some other rules.

For the prediction problem in Bendel's study, the regression equation is used to predict the y value of a randomly chosen element (y_0,$\mathbf{x_0}$) of the population. When $\hat{y}_{(q)}(\mathbf{x}_0)$ is used as an estimator of y_0, the unconditional mean-square error of the prediction is defined as:

$$UMSE_{(q)} = E\{\hat{y}_{(q)}(\mathbf{x}_0) - y_0\}^2$$

where the expectation is taken over all the random variables, including the "new" observation (y_0,$\mathbf{x_0}$) , as well as the independent variables of the regression sample. The criterion UMSE was used by Stein [23] who employed the same basic model as described. Afifi and Elashoff [1] used this criterion to establish relative efficiencies in various missing value techniques.

In another study by Wilkinson (1981), he *simulated upper critical values of the distribution of the sample squared multiple correlation coefficient under forward selection regression when the population multiple correlation is zero and the predictors are mutually independent.*

His simulation involved sampling of a dependent variable and predictors from a standard spherical multi-normal distribution. Sample correlation matrices were generated directly from a standardized Wishart distribution using an algorithm from Odell and Feiveson(1966).

Normal numbers were generated by a subroutine FA03A of the Harwell Subroutine Library (1973). R^2 values were generated by the abbreviated Gauss-Doolittle method with forward selection of the predictors (Draper and Smith 1966, p 178). Forward stepping was continued on each sample until the F-to-enter rule was not met by any variable remaining to be selected. (Barr et al. 1976; Dixon 1979; Nie et al. 1975; denote this stopping rule as F-to-enter). Each regression was computed for at least 500 samples and continued in additional blocks of 100 samples until three successive blocks yielded the same R^2 value to two decimal places for both the 95th and 99th

[10] Wilkinson L. 1981. *Test of Significance in Forward Selection Regression With an F-to-Enter Stopping Rule*

percentage points of the sample distribution. Using this procedure, test runs on known values (when F = 0) yielded the upper tail values correct to two significant digits reported in this article.

The values calculated in Wilkinson's study are shown as follows:

Table A: *Upper Five Percent Points of the Distribution of the Sample Squared Multiple Correlation (× 10^2) Under Forward Selection From m Variables, n Observations, and F - to - enter Value.*

		n − m − 1														
m	F	10	12	14	16	18	20	25	30	35	40	50	60	80	100	150	200
2	2	43	38	33	30	27	24	20	16	14	13	10	8	6	5	3	2
2	3	40	36	31	27	24	22	18	15	13	11	9	7	5	4	3	2
2	4	38	33	29	26	23	21	17	14	12	10	8	7	5	4	3	2
3	2	49	43	39	35	32	29	24	21	18	16	12	10	8	7	4	3
3	3	45	40	36	32	29	26	22	19	17	15	11	9	7	6	4	3
3	4	42	36	33	29	27	25	20	17	15	13	11	9	7	5	4	3
4	2	54	48	44	39	36	33	27	23	20	18	15	12	10	8	5	4
4	3	49	43	39	36	33	30	25	22	19	17	14	11	8	7	5	4
4	4	45	39	35	32	29	27	22	19	17	15	12	10	8	6	5	3
5	2	58	52	47	43	39	36	31	26	23	21	17	14	11	9	6	5
5	3	52	46	42	38	35	32	27	24	21	19	16	13	9	8	5	4
5	4	46	41	38	35	32	29	24	21	18	16	13	11	9	7	5	4
6	2	60	54	50	46	41	39	33	29	25	23	19	16	12	10	7	5
6	3	54	48	44	40	37	34	29	25	22	20	17	14	10	8	6	5
6	4	48	43	39	36	33	30	26	23	20	17	14	12	9	7	5	4
7	2	61	56	51	48	44	41	35	30	27	24	20	17	13	11	7	5
7	3	55	50	46	42	39	36	31	26	23	21	18	15	11	9	7	5
7	4	50	45	41	38	35	32	27	24	21	18	15	13	10	8	6	4
8	2	62	58	53	49	46	43	37	31	28	26	21	18	14	11	8	6
8	3	57	52	47	43	40	37	32	28	24	22	19	16	12	10	7	5
8	4	51	46	42	39	36	33	28	25	22	19	16	14	11	9	7	5
9	2	63	59	54	51	47	44	38	33	30	27	22	19	15	12	9	6
9	3	58	53	49	44	41	38	33	29	25	23	20	16	12	10	7	6
9	4	52	46	43	40	37	34	29	25	23	20	17	14	11	10	7	6
10	2	64	60	55	52	49	46	39	34	31	28	23	20	16	13	10	7
10	3	59	54	50	45	42	39	34	30	26	24	20	17	13	11	8	6
10	4	52	47	44	41	38	35	30	26	24	21	18	15	12	10	8	6
12	2	66	62	57	54	51	48	42	37	33	30	25	22	17	14	10	8
12	3	60	55	52	47	44	41	36	31	28	25	22	19	14	12	9	7
12	4	53	48	45	41	39	36	31	27	25	22	19	16	13	11	9	7
14	2	68	64	60	56	53	50	44	39	35	32	27	24	18	15	11	8
14	3	61	57	53	49	46	43	37	32	29	27	23	20	15	13	10	8
14	4	53	49	46	42	40	37	32	29	26	23	20	17	13	11	9	7
16	2	69	65	61	58	55	53	46	41	37	34	29	25	20	17	12	9
16	3	61	58	54	50	47	44	38	34	31	28	24	21	17	14	11	8
16	4	53	50	46	43	40	38	33	30	27	24	21	18	14	12	10	8
18	2	70	67	63	60	57	55	49	44	40	36	31	27	21	18	13	9
18	3	62	59	55	51	49	46	40	35	32	30	26	23	18	15	12	9
18	4	54	50	46	44	41	38	34	31	28	25	22	19	15	13	11	8
20	2	72	68	64	62	59	56	50	46	42	38	33	28	22	19	14	10
20	3	62	60	56	52	50	47	42	37	34	31	27	24	19	16	12	9
20	4	54	50	46	44	41	39	35	32	29	26	23	20	16	14	11	8

Table B *Upper One Percent Points of the Distribution of the Sample Squared Multiple Correlation (× 10^2) Under Forward Selection From m Variables, n Observations, and F - to - enter Value.*

n - m - 1

n	F	10	12	14	16	18	20	25	30	35	40	50	60	80	100	150	200
2	2	59	53	48	43	40	36	30	26	23	20	17	14	11	9	7	5
2	3	58	52	46	42	38	35	30	25	22	19	16	13	10	8	6	4
2	4	57	49	44	39	36	32	26	22	19	16	13	11	8	7	5	4
3	2	67	60	55	50	46	42	35	30	27	24	20	17	13	11	7	5
3	3	63	58	52	47	43	40	34	29	25	22	19	16	12	10	7	5
3	4	61	54	48	44	40	37	31	26	23	20	16	14	11	9	6	5
4	2	70	64	58	53	49	46	39	34	30	27	23	19	15	12	8	6
4	3	67	62	56	51	47	44	37	32	28	25	21	18	14	11	8	6
4	4	64	58	52	47	43	40	34	29	26	23	19	16	13	11	7	6
5	2	73	67	61	57	52	49	42	37	32	29	25	21	16	13	9	7
5	3	70	65	59	54	50	46	39	34	30	27	23	19	15	12	9	7
5	4	65	60	55	50	46	43	36	31	28	25	20	17	14	12	8	6
6	2	74	69	63	59	55	51	44	39	34	31	26	23	18	14	10	8
6	3	72	67	61	56	51	48	41	36	32	28	24	20	16	13	10	7
6	4	66	61	56	52	48	45	38	33	29	26	22	19	15	13	9	7
7	2	76	70	65	60	56	53	46	40	36	33	28	25	19	15	11	9
7	3	73	68	62	57	53	50	42	37	33	30	25	21	17	14	10	8
7	4	67	62	58	54	49	46	40	35	31	28	23	20	16	14	10	8
8	2	77	72	66	62	58	55	48	42	38	34	29	26	20	16	12	9
8	3	74	69	63	58	54	51	44	39	34	31	26	22	18	15	11	9
8	4	67	63	59	55	50	47	41	36	32	29	24	21	17	15	11	9
9	2	78	73	67	63	60	56	49	43	39	36	31	27	21	17	12	10
9	3	74	69	64	59	56	52	45	40	35	32	27	23	19	16	12	9
9	4	68	63	60	56	51	48	42	37	33	30	25	22	18	16	12	9
10	2	79	74	68	65	61	58	51	45	40	37	32	28	22	18	13	10
10	3	74	69	65	60	57	53	47	41	37	33	28	24	20	17	13	10
10	4	68	64	61	56	52	49	43	38	34	31	26	23	19	17	13	9
12	3	80	75	70	66	63	60	53	48	43	39	34	30	24	20	14	11
12	3	74	70	66	62	58	55	48	43	39	35	30	26	21	18	14	10
12	4	69	65	61	57	53	50	44	40	35	32	27	24	20	18	13	10
14	2	81	76	71	68	65	62	55	50	45	41	36	32	25	21	15	11
14	3	74	70	67	63	60	56	50	45	41	37	31	27	22	19	15	11
14	4	69	65	61	57	54	52	45	41	36	33	28	25	21	19	14	10
16	2	82	77	72	69	66	63	57	52	47	43	38	34	27	22	16	12
16	3	74	70	67	64	61	58	52	47	42	39	33	29	23	20	15	11
16	4	70	66	62	58	55	52	46	42	37	34	29	26	22	20	14	11
18	2	82	78	73	70	67	65	59	54	49	45	39	35	28	23	17	12
18	3	74	70	67	65	62	59	53	48	44	41	35	30	24	21	16	12
18	4	70	65	62	58	55	53	47	43	38	35	30	27	23	20	15	11
20	2	82	78	74	71	68	66	60	55	50	46	41	36	29	24	18	13
20	3	74	70	67	65	62	60	55	50	46	42	36	32	26	22	17	12
20	4	70	66	62	58	55	53	47	43	39	36	31	28	24	21	16	11

The tables (A & B) of simulation results allow graphical interpolation with respect to any of the parameters (F, m: the number of variables selected, and n: number of observations) to within a 0.01 precision in the R^2 values. The use of forward selection with these tables allows the researcher to test the null hypothesis that the population squared multiple correlation is zero when the number of variables in a subset regression has not been specified prior to the data analysis.

Another study on *Forward selection of Explanatory variables* is done by Blanchet et al. (2008):

This paper proposes a new way of using forward selection of explanatory variables in regression or canonical redundancy analysis.

Ecologists are known to sample a large number of environmental variables to try to better understand how and why species and communities are structured. They are often faced with the problem of having too many explanatory variables to perform standard regression or canonical analysis. One method very often used for selecting variables in ecology, is forward selection. It presents the great advantage of being applicable even when the initial data set contains more explanatory variables than sites, which is often the case in ecology.

However, forward selection is known to overestimate the amount of explained variance, which is measured by the coefficient of multiple determination (R^2; Diehr and Hoflin 1974, Rencher and Pun 1980). Another problem with forward selection is a highly inflated Type I error.

Correcting these problems will greatly improve the performance of this very useful method in ecological modelling. To prevent this problem, they proposed a two-step procedure. Firstly, a global test using all explanatory variables is carried out. If and only if, the global test is significant, one can proceed with forward selection. To prevent overestimation of the explained variance, the forward selection has to be carried out with two stopping criteria: (1) the usual alpha significance level and (2) the adjusted coefficient of multiple determination (R^2) calculating using all explanatory variables.
(Adding unimportant variables to an already well-fitted model has practically no impact on the explained variance measured by R_a^2. Thus the use of R_a^2 as an additional stopping criterion is a good choice in forward selection procedure).

When forward selection identifies a variable that brings one or the other criterion over the fixed threshold, that variable is rejected and the procedure is stopped. This improved method is validated by simulations involving univariate and multivariate response data.

The table below is also an abstract out of Blanchet's article.

Table C shows the percentage of simulations in which all the variables are used to create the univariate or multivariate response variable(s), and only those are retained by the forward selection procedure.

Table C

Error terms in response data	Stopping criteria	PCNM (%)	Positive BEM (%)	Negative BEM (%)	Normal (%)	Uniform (%)
Univariate variables						
Standard deviation†	alpha and R_a^2	10.6	10.5	10.5	6.4	6.7
	alpha	0.5	0.4	0.5	0.4	0.5
Standard deviation/4‡	alpha and R_a^2	17	18.4	17.7	14.2	13.7
	alpha	8.3	6.7	7	4.7	4.8
Standard deviation/1000§	alpha and R_a^2	17	16.8	17.2	13.8	13.3
	alpha	8	6.9	7	4.9	4.6
Multivariate variables						
Standard deviation†	alpha and R_a^2	1.2	1.2	1.1	0.8	0.8
	alpha	0.3	0.3	0.3	0.3	0.2
Standard deviation/4‡	alpha and R_a^2	2.4	2.3	2.5	2.7	2.8
	alpha	1.6	2.2	2	1.6	1.5
Standard deviation/1000§	alpha and R_a^2	2.5	2	2	2.5	2.8
	alpha	1.5	1.9	1.7	1.6	1.4

Note: Abbreviations are: PCNM, principal coordinates of neighbor matrices; BEM, binary eigenvector maps; R_a^2, adjusted coefficient of multiple determination (adjusted R^2).
† Error = standard deviation of the deterministic portion of the response variables.
‡ Error = standard deviation 25% that of the deterministic portion.
§ Error = 0.001 times the standard deviation of the deterministic portion.

An ecological example is presented in this article as well.

Whitney (2000) did a study on: *Unsupervised Forward Selection: A method for eliminating Redundant variables.*

An unsupervised learning method is proposed for variable selection and its performance is assessed using three typical QSAR data sets. The aims of this procedure are to generate a subset of descriptors from any given data set in which the relevant variables are relevant, redundancy is eliminated and multicollinearity is reduced. Continuum regression, an algorithm encompassing ordinary least squares regression, regression on principal components, and partial least squares regression, was used to construct models from selected variables. The variable selection routine is shown to produce simple, robust, and easily interpreted models for chosen datasets.

Another study is done by Stodden(2006) where she studied *Model Selection when the number of variables exceeds the number of observations* i.e. ***n < p.***

This classical multivariate linear regression problem assumes p predictor variables X_1, X_2….X_p and a response vector y, each with n observations, and a linear relationship between the two. $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \sim N(0, \sigma^2)$. This thesis found that when n < p, there is a breakdown point for standard model selection schemes, such that model selection only works well below a certain critical complexity level depending on $\frac{n}{p}$. This notation is applied to some standard model selection algorithms (Classical Forward Stepwise, Forward Stepwise with False Discovery Rate thresholding, Lasso, LARS, and Stagewise Othogonal Pursuit) in the case where n <<p.

The notation of the Phase 1 Diagram is borrowed from the signal processing and statistical physics to discover that:

1) The breakdown point is well-defined for random X-models and low noise
2) Increasing noise shifts the breakdown point to lower levels of sparsity, and reduces the model recover ability of the algorithm in a systematic way and
3) Below breakdown, the size of coefficient errors follows the theoretical error distribution for the classical linear model.

Stodden used threshold algorithms for the Classical Forward Stepwise, the Forward Stepwise and the Lasso and LARS methods. She determined that the threshold-to-enter for Classical Stepwise was set at $\sqrt{2\log(p)}$, implying that the absolute value of the t-statistic for variables under consideration for inclusion in the model must be greater than $\sqrt{2\log(p)}$. She used the number of variables, p = 200. The threshold-to-enter would then be 3.25.

Figure B

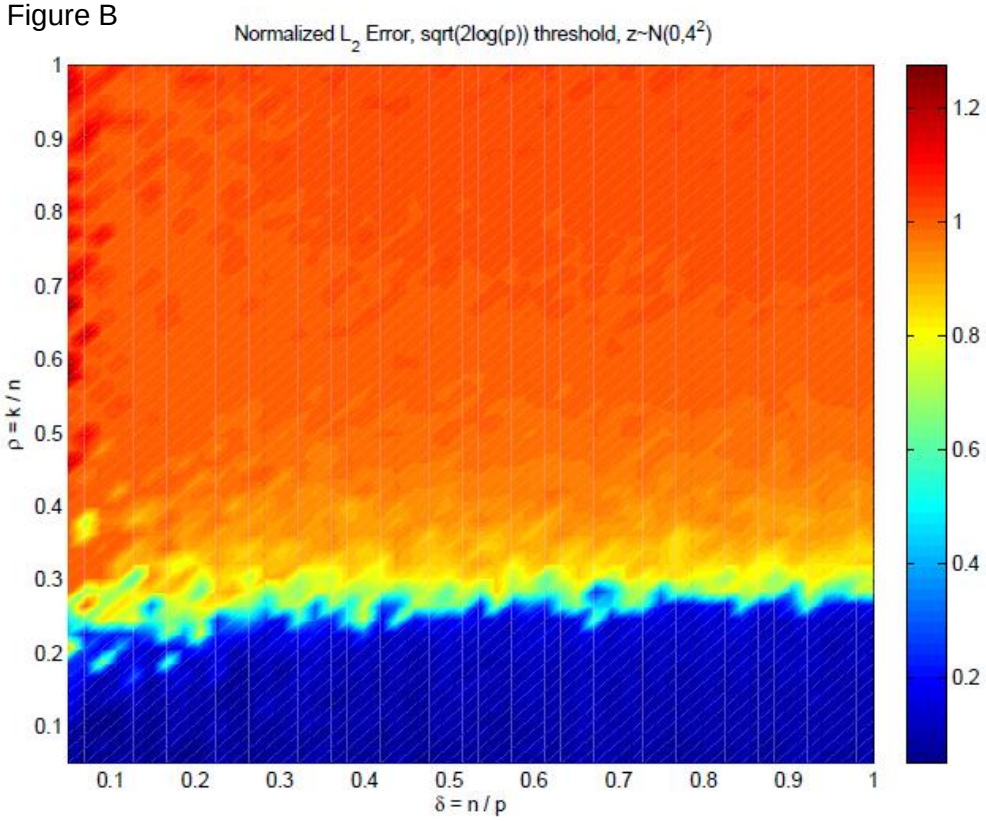


Figure B: This Empirical Phase Diagram for Forward Stepwise illustrates where the underlying sparse model is recovered using the Forward Stepwise Algorithm, with the number of variables, p, fixed at 200 and the noise $\boldsymbol{\varepsilon} \sim N(0,16)$. Variables were greedily added to the model until no remaining t-statistic was greater than $\sqrt{2\log(p)}$ in absolute value. The phase transition is striking: there is a very sharp drop-off below which the algorithm recovers the model with near zero error, and above which the model is unrecoverable. As with the theoretical phase transition diagram in Figure C, along the x-axis the level of underdeterminedness decreases, and along the y-axis the level of

sparsity of the underlying model increases. Each colour indicates a different median normilized l_2 error of the coefficients $\frac{|\hat{\beta}-\beta|^2}{|\beta|^2}$ over 30 realizations

where l_i-norm is defined for a vector x = $[x_1,x_2 \ldots x_n]$:
and where $|x_0|$ = # {non zeros in x}

Using this to define the sparsity solution gives the following problem:

$min_\beta \ |\boldsymbol{\beta}|_0$ s.t.y = $\boldsymbol{X\beta}$

This is intuitively compelling – literally choosing the sparsest solution.

The normalized root mean-square error (nRMSE) = $\frac{|\hat{\beta}-\beta|^2}{|\beta|^2}$

The results are comparable across models built with different problem sizes and noise levels. The median nRMSE over instances at given values for k, n and p have been reported to indicate the algorithm performance.

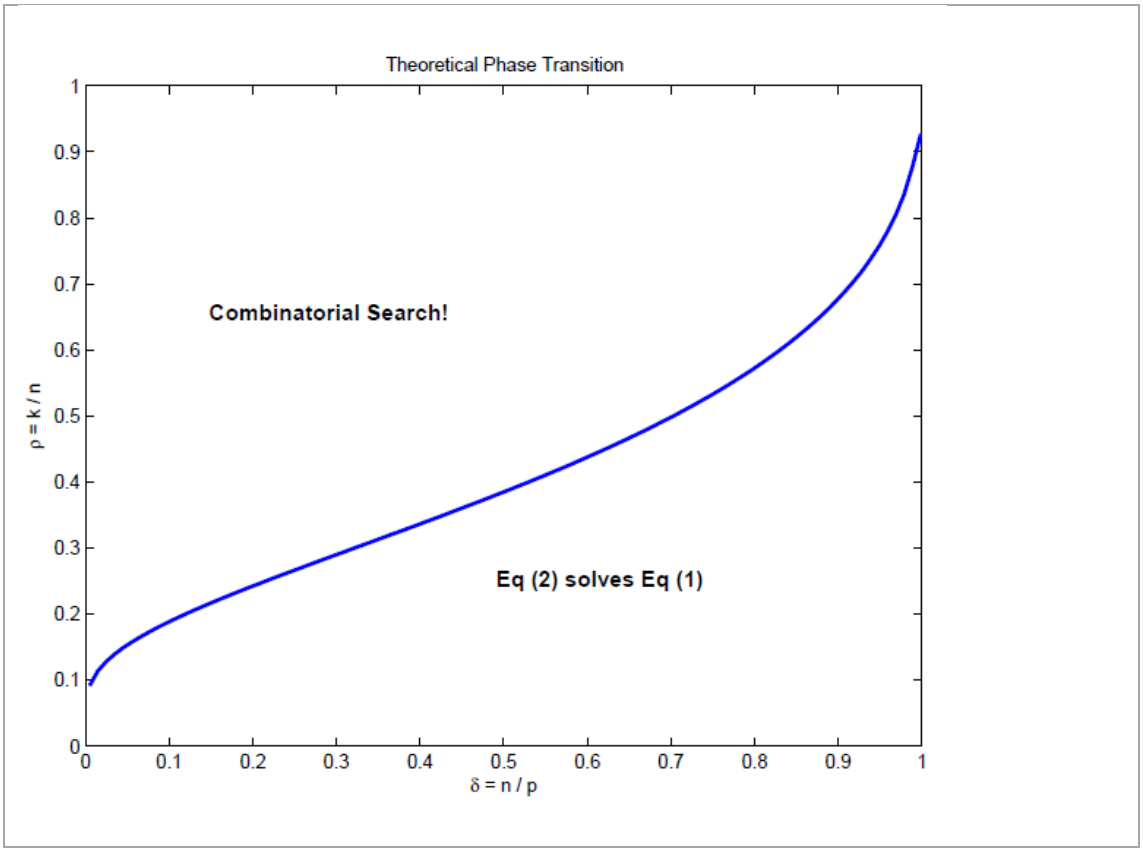


Figure C: Theoretical Phase Transition Diagram: theoretical threshold at which equivalence of the solutions l_1 and l_0 optimization problem breaks down. The curve delineates a phase transition from the lower region where the equivalence holds, to the upper region, where it seems to require combinational search to recover the optimal sparse model. Along the y-axis the level of underdeterminedness decreases, and along the y-axis the level of sparsity of the underlying model increases.

Stodden also determined the nRMSE for Pre-Breakdown Region for the Classical Forward Stepwise, with $\delta = \frac{n}{p}$ = 0.5 (this is also one of the n versus p cases in this study).

Figure D

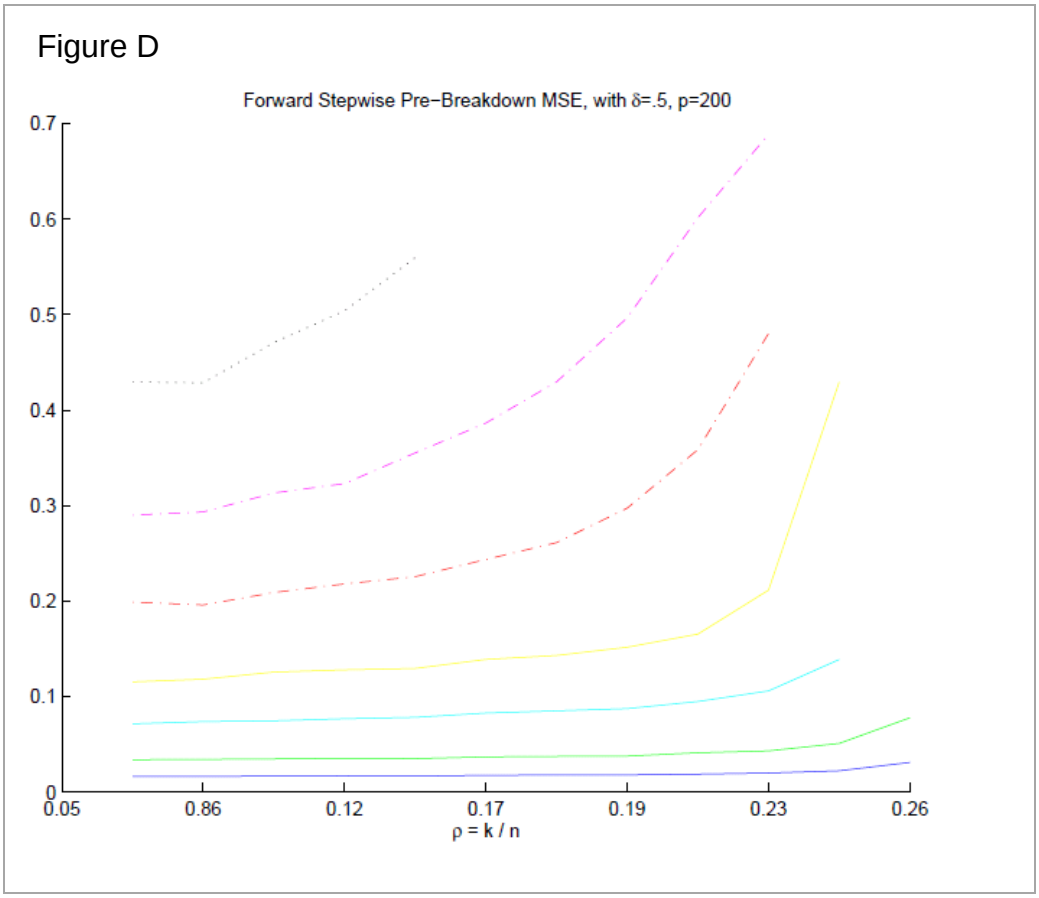


Figure D: nRMSE for Pre-Breakdown Region for the Classical Forward Stepwise; $\delta = \frac{n}{p}$, p = 200. Results for each of $\sigma \in \{0,1,2,4,6,12,16\}$ are displayed. The median relative error is determined over 1000 replications. Although the actual nRMSE rates are low compared to the other algorithms in this study, the breakdown points occur earlier, i.e. at smaller values of ρ.

Stodden mentioned that to expand the analysis beyond $\delta = \frac{n}{p}$ =0.5 would be useful, since the nature of the breakdown in performance, depends on the level of indeterminacy δ.

Extending the experiments to a much larger p would also allow one to draw empirical conclusions about how the level of indeterminacy affects the breakdown point.

• NUMBER OF VARIABLES AND OBSERVATIONS

In this study some attention is paid on the estimating of the regression coefficients in the cases where the number of variables exceeds the number of observations – as studied by Stodden. The accuracy of the model with different n and p values is determined.

Radchenko(2011) on the other hand only focused on variable selection methods in regression situations where the number of predictors (p) is large relative to the number of observations (n). Two commonly applied variable selection approaches are the Lasso, which computes highly shrunk regression coefficients, and Forward Selection, which uses no shrinkage. They propose a new approach, “Forward-Lasso Adaptive Shrinkage” (FLASH), which includes the Lasso and Forward Selection as special cases, and can be used in both the linear regression and the Generalized Linear Model domains.

As with the Lasso and Forward Selection, FLASH iteratively adds one variable to the model in a hierarchical fashion but, unlike these methods, at each step adjusts the level of shrinkage so as to optimize the selection of the next variable. They first present FLASH in the linear regression fitting and show that it can be fitted using a variant of the computationally efficient LARS algorithm. Then they extend FLASH to the GLM domain and demonstrate, through numerous simulations and real world data sets, as well as some theoretical analysis, that FLASH generally outperform many competing approaches.

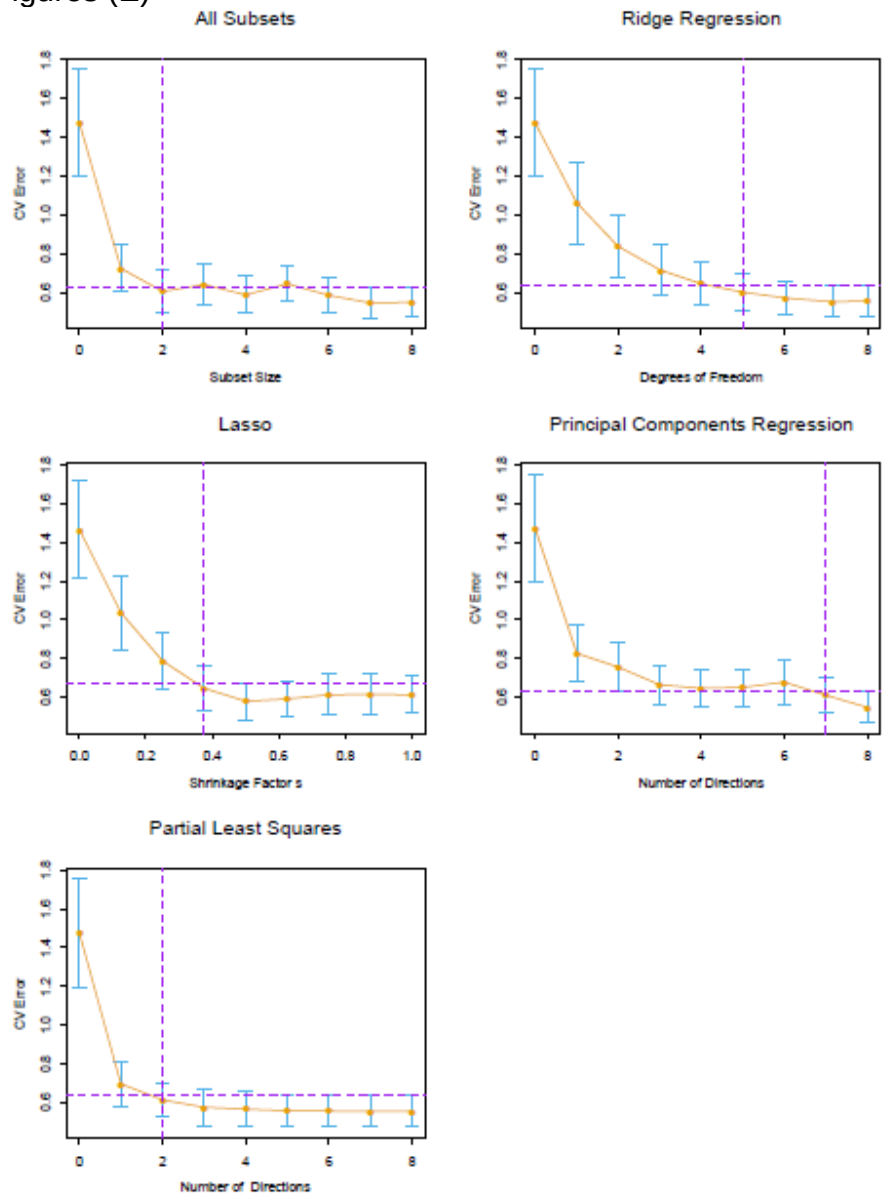
• SHRINKING APPROACHES

Hastie et al.(2009) made a study of the shrinkage methods as well. Subset selection is a method of retaining a subset of the predictors and discarding the rest. It produces a model that is interpretable and has possibly lower prediction error than the full model. However, because it is a discrete process – variables are either retained or discarded – it often exhibits high variance, and so doesn’t reduce the prediction error of the full model. Shrinkage methods are more continuous, and do not suffer as much from high variability.

He compared the different methods, by comparing estimated prediction error curves.

The following graphs, (Figures (E)) show the estimated prediction error curves and their standard errors for the various selection and shrinkage methods. Each curve is plotted as a function of the corresponding complexity parameter for that method. The horizontal axis has been chosen so that the model complexity increases as we move from left to right. The estimates of prediction error and their standard errors were obtained by tenfold cross-validation. The least complex model within one standard error of the best is chosen, indicated by the purple vertical broken lines.

Figures (E)



These different methods are only shown here but this is beyond the scope of this dissertation. It is just shown to take note of.

Chapter 4

DESIGN OF THE SIMULATION STUDY

In this chapter the parameters used in the SAS IML programme are given, and the procedures and steps of the program, is described. The references referred to, is given in chapter 5, "*Steps, following the Statistics of the study*", simulation results followed in chapter 6, conclusions and graphs in chapter 7, as well as the program, in chapter 11 - the appendix.

The forward selection regression model is used for this investigation. The full model is also used for comparison in this study.

The references marked in red referer to the program and / or printout.

In this study, simulations of 10 000 runs (sim=10000)[11] for the chosen n and p values are done for each of the full- and forward multiple regression analysis. Firstly a few modules in SAS IML are defined, followed by the main program. Then the full regression model is done followed by the forward selection model. The last part of the program consists of a few calculations.

The parameters chosen for the study are the following:

The number of variables (p) used in the regression, is chosen to range from 4 to 20 (integer numbers). This is the number of variables to be included in the full model for each case. In the forward selection model, the best k (k < p) variables are chosen. For each of the p = 4…20[12] the values for n, which is the number of observations is chosen to be 0.5p, p, 2p and 4p[13]. In the case where n = 0.5p and n is not an integer, n was rounded up to the next larger integer.

The x values are then generated from a normal distribution[14] where $x_i \sim N(\boldsymbol{\mu_x}, \boldsymbol{\Sigma_X})$ and $\boldsymbol{\mu_x}$={**0**}[15] and

$$\boldsymbol{\Sigma_X} = \begin{pmatrix} 1 & \cdots & 0.5 \\ \vdots & \ddots & \vdots \\ 0.5 & \cdots & 1 \end{pmatrix} = 0.5(I_p + J_p)$$[16] where $I_p = \begin{pmatrix} 1 & \ldots & 0 \\ \vdots & 1 & 0 \\ 0 & \ldots & 1 \end{pmatrix}$ and $J_p = \begin{pmatrix} 1 & \ldots & 1 \\ \vdots & 1 & \vdots \\ 1 & \ldots & 1 \end{pmatrix}$

ε_i are also generated from a normal[17] distribution with ε_i independent ~ N(0,4)[18] distributed. The equation for the linear regression model is $y = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$.

The β_i (where p is even) values are assigned $\beta_i = 1$; for the first half of the total number of x_i's (these are the first $\frac{p}{2}$ values). If the p value (the total number of x's) is an odd

---

[11] The number of runs for each p versus n ratio's named *sim* in the program
[12] p = 4 to 20 - see program
[13] n = 0.5p, p, 2p and 4p – see program
[14] The x variables are generated from a normal distribution
[15] X has mean 0 – see program
[16] This is $\boldsymbol{\Sigma_X}$ **= Cov(x)** names sigma1 in the program and printout
[17] ε_i is generated from a normal distribution
[18] ε_i independent ~ N(0,4) variables – see program

number, then the first $\frac{p+1}{2}$ values were assigned to the $\beta_i = 1$ value and the remaining $\beta_i = 0$.

Thus $\beta = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$ for the first $\frac{p}{2}$ or $\frac{p+1}{2}$ $\beta_i = 1$'s and the rest of $\frac{p}{2}$ $\beta_i's = 0$'s.[19]

That means that half of the number of variables i.e. the first $\frac{p}{2}$ or $\frac{p+1}{2}$, should be included in the model when doing forward selection. In this study we determine the probability of a correct classification as well.

The results and conclusions of this study are mostly illustrated in the graphs presented in Chapter 7.

a. The full model[20]

The full linear regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ is used to determine the least squares estimates of the full model with $\widehat{\boldsymbol{\beta}} = (\boldsymbol{X`X})^{-1}\boldsymbol{X`y}$[21]

The parameters were chosen as above mentioned.

The test for **significance of regression** is a test to determine if there is a **linear relationship**[22] between the response variable (y) and the regressor variables $x_1, x_2 \ldots x_p$. This test is done in the analysis for each simulation and each selected n and p case[23].

This procedure is often thought of as an overall or global test of model adequacy. The appropriate hypotheses are:

$H_0: \beta_1 = \beta_2 = \cdots . \beta_p = 0$
$H_1: \beta_j \neq 0$ (for some j)

Rejection of the null hypothesis implies that at least one of the regressors $x_1, x_2 \ldots x_p$. contributes significantly to the model.

The test procedure is a generalization of the **analysis of variance** used in simple linear regression. The **total sum of squares** (SST)[24] is portioned into the **sum of squares due to regression**, (SSreg)[25] and the **residual sum of squares** (SSE)[26].

Thus SST = SSreg + SSE
MSreg = SSreg /p[27]

---

[19] β_i is assigned the values of 1 and 0 (for even and odd p's) – see program and output
[20] The regression analysis on the full model(a) and (b) for the forward model
[21] Determining $\hat{\beta}$ of the full model – see program
[22] Referring to Montgomery (2006). *Introduction to linear regression analysis* p.80.
[23] See F-statistic and F- probability in program as well as output
[24] See program and printout
[25] The regression sum of squares (SSreg) for the full model (see program and printout)
[26] The residual sum of squares (SSE) for the full model (see program and printout)
[27] Mean-square due to regression – See program and output

MSE = SSE / (n – p – 1)[28]

The R^2 = SSreg/SST[29] is called the **coefficient of determination.**

The full regression model is applied to the simulated data. The F-probability[23] is determined for each simulation and each case (n and p).

The F-probabilities are grouped into the classes p < 0.01, 0.01 < p < 0.05 and p > 0.05 for each simulation, n and p case. The proportion counts of each n and p case (over 10 000 simulations) is determined.[30]

See table 5, graphs 10 and 11 for the results on this study.

The test for the Hypothesis H_0: β_j=0 for each β_j *and*
H_1: $\beta_j \neq 0$ for each β_j is done in this example.

This test should actually be tested against the real β_{j0} values, assigned in the program and not β_j= 0 (for all β_j) as done here.

This test here is actually done for testing the SAS IML program versus the SAS PROC REG program:[31]

The right test for the Hypothesis H_0: $\beta_j=\beta_{j0}$ for each β_j *and*
H_1: $\beta_j \neq \beta_{j0}$ for each β_j for j = 0…p

where the test statistic should be:

$$t_0 = \frac{\hat{\beta}_j - \beta_{j0}}{se(\hat{\beta}_j)}$$

where the null hypothesis H_0: $\beta_j=\beta_{j0}$ for each β_j should be rejected if:

$$|t_0| > t_{\frac{\propto}{2},n-p-1}$$

The mean-squared error is also determined and compared to the mean squared-error for the forward model. Results, on this are seen in table 4 and graphs 3, 5, 8 & 9.

The other calculations follow in chapter 5.

[28] Mean square error – See program and output
[29] R^2 – See program and printout
[30] F-probability (proportion counts) - See program and the results in Chapter 7
[31] See program and printout

b. The Forward selection model[20]

The full linear regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ [32] can be written as $\boldsymbol{y} = \boldsymbol{X_k\beta_k} + \boldsymbol{X_{p-k}\beta_{p-k}} + \varepsilon$ where p is the number of variables and k the numbers of variables selected in the forward selection.

Then, the least-squares estimate of $\boldsymbol{\beta_k}$ is $\widehat{\boldsymbol{\beta_k}} = (\boldsymbol{X_k}`\boldsymbol{X_k})^{-1}\boldsymbol{X_k}`\boldsymbol{y}$ [21]

Firstly the variable maximum correlated to the y-variable is determined.[33]

Test the significance of the variable.[34]

A counter matrix is calculated to determine which variables in- or excluded from the model.[35]

The coefficients of partial *correlation* are determined.[36]

The maximum r is determined.[37]

This process is repeated until there are no significant variables left for inclusion into the model.[38]

The final $\widehat{\boldsymbol{\beta_k}}$ for the forward model is determined.[39]

The mean-squared error of the forward- and full models is determined.[40]

The other calculations which are used for this study be follow in the program, chapter 11 – the appendix.

---

[32] Assigning of model
[33] Determine the variable mostly correlated to the y-variable – See program and output
[34] T-value is determined and tested against the t-statistic
[35] Xin and xout are calculated
[36] See ryx in the program and printout
[37] See program and printout – Maximum r.
[38] Process repeated
[39] Estimate$\widehat{\boldsymbol{\beta_k}}$ - See program and printout
[40] See program and printout

Chapter 5

STEPS, FOLLOWING THE STATISTICS OF THIS STUDY (printout)

This chapter consists of the computer printout for 1 simulation. It actually differs a bit from the program in the appendix. The program in the appendix is for the 10 000 simulation with the different cases of n and p. It is also the full program of the study.

The statistical analysis in IML is compared with the statistics in the SAS (proc reg) printout here. The parameters here are set the same as in chapter 7(the simulation study), but with only 1 simulation here. So it can be compared with SAS (proc reg)

n = 30 and p = 8 is used here.

The SAS IML programme
(See the references on the red letters. It refers back to the footnotes used in Chapter 4)

The SAS System

sim [11]	i
no =	1

sigma1 (x) [16]							
1	0.5	0.5	0.5	0.5	0.5	0.5	0.5
0.5	1	0.5	0.5	0.5	0.5	0.5	0.5
0.5	0.5	1	0.5	0.5	0.5	0.5	0.5
0.5	0.5	0.5	1	0.5	0.5	0.5	0.5
0.5	0.5	0.5	0.5	1	0.5	0.5	0.5
0.5	0.5	0.5	0.5	0.5	1	0.5	0.5
0.5	0.5	0.5	0.5	0.5	0.5	1	0.5
0.5	0.5	0.5	0.5	0.5	0.5	0.5	1

Assigned β: β^{19}
0
1
1
1
1
0
0
0
0

a. ========== REGRESSION FOR THE FULL MODEL[20] ==========

Bhat [21]
-0.219302
0.854587
1.323134
0.8227202
0.3889855
1.4121609
-0.470012
0.7243091
-0.278632

SSreg [25]
449.00154

SSE [26]
297.86037

MSreg [27]
56.125193

MSE [28]
14.183827

SST[24]
746.86191

F[23]
3.9569851

	pp=p [23]
The F-probability is :	0.0049113

	R^2 [29]
The R - square value is:	0.6011841

r correlation matrix									
	y	x1	x2	x3	x4	x5	x6	x7	x8
y	1	0.5767	0.6514	0.4698	0.5708	0.6508	0.3319	0.4342	0.2712
x1	0.576	1	0.6023	0.1606	0.5935	0.5050	0.2886	0.4136	0.1864
x2	0.651	0.6023	1	0.4050	0.5278	0.6079	0.4170	0.4345	0.2714
x3	0.469	0.1606	0.4050	1	0.4822	0.5211	0.3296	0.2439	0.5080
x4	0.570	0.5935	0.5278	0.4822	1	0.5713	0.2278	0.2537	0.2109
x5	0.650	0.5050	0.6079	0.5211	0.5713	1	0.4683	0.3236	0.4455
x6	0.331	0.2886	0.4170	0.3296	0.2278	0.4683	1	0.5543	0.5623
x7	0.434	0.4136	0.4345	0.2439	0.2537	0.3236	0.5543	1	0.2633
x8	0.271	0.1864	0.2714	0.5080	0.2109	0.4455	0.5623	0.2633	1

b [31]	bhat	stderrB	probT
b0	-0.219302	0.7585274	0.7753284
b1	0.854587	0.992829	0.3990919
b2	1.323134	1.1591848	0.2665384
b3	0.8227202	0.9473216	0.3949543
b4	0.3889855	1.0044306	0.7024536
b5	1.4121609	0.9891448	0.1680929
b6	-0.470012	1.0154299	0.6482184
b7	0.7243091	0.7777099	0.3622641
b8	-0.278632	0.993503	0.7818731

b. ========== FORWARD REGRESSION ANALYSIS [20]==========

	xxmax [33]	
keep x	2	in the model

T [34]
4.5438157

	Tprob [34]
tprob=	0.0000964

k
1

Xin	Xout [35]
0	1
1	0
0	1
0	1
0	1
0	1
0	1
0	1

The real β in the model at this stage

bForward
0
1

	xstay
The following x_s stays in the model	2

SSEin
429.88162

Coefficients of partial correlation - ryx [36]						
0.304330	0.2968863	0.3522077	0.4229714	0.0874718	0.2211262	0.129306

[37]	SSEyxmin
The minimum SSE is:	5

[37]	yxmax
The maximum r is:	5

maxryx
0.4229714

	xxmax	37
keep x	5	in the model

t
2.4254713

38 (process repeated)

	tprob
tprob=	0.02225

k = 2

xin	xout
0	1
1	0
0	1
0	1
1	0
0	1
0	1
0	1

bForward
0
1
0

	xstay	
The following x_s stays in the model	2	5

The following variables are not in the model

noutx					
1	3	4	6	7	8

SSEin
352.97373

Ryx					
0.239414	0.1630229	0.231948	0.0444591	0.2059767	0.0308368

	SSEyxmin
The minimum SSE is:	1

	yxmax
The maximum r is:	1

maxryx
0.239414

h
6

38 (process repeated)

t
1.257343

	tprob
tprob =	0.2198067

	xxmax	
x	1	will not be in the model

k
3

	finalxxx	
The following x e will be in the final model	2	5

[39]	bhatFORW
The final estimates values for the forward model are:	-0.094351
	2.3167823
	1.9129184

Xinlast (last counter)							
0	1	0	0	1	0	0	0

MSE_bhattFORW [40]
5.4020743

MSE FOR THE Forward MODEL of the estimated coefficient BHAT at each step from the true B is

MSE_bhatTOTFORW [40]
5.4020743

MsbtotFULL [40]
3.3957896

MSE for the FULL model of the estimated coefficient BHAT at each sim from the true B is

MSE_bhatTOTFULL [40]
3.3957896

dd		
1	0	0

	sim	
The probabilities of the full model with	1	simulations

[30]			no_prob_prob		
<0.01	0.01<=p<=0.05	>0.05	1	0	0

The probabilities of each of the x_s to be in the forward model is:

finalxin_prob							
0	1	0	0	1	0	0	0

DATA

Obs	y	x0	x1	x2	x3	x4	x5	x6	x7	x8
1	-0.7629	1	0.04656	0.62040	-1.26848	-0.27776	-0.53836	0.52450	1.58019	-0.60804
2	-1.6483	1	-1.88630	-0.87413	0.24278	-0.85437	-1.25476	-0.14468	1.35467	-1.86182
3	-11.2217	1	-2.02250	-1.18131	-1.85268	-1.60058	-2.37542	-0.47465	-1.00262	0.05250
4	-3.5763	1	0.85877	-1.09555	-2.85873	-0.07797	-0.93742	-1.14722	0.40687	-1.69449
5	-3.0987	1	0.49500	0.30015	-1.20874	-1.51518	-0.91345	0.67724	0.53144	-0.47134
6	6.2110	1	0.34521	1.68691	1.17978	0.50887	0.09043	-0.05671	0.92642	0.06760
7	3.2436	1	-0.52074	-0.80588	-0.16808	-0.90967	-1.21116	0.43454	0.83004	-0.68337
8	-1.7805	1	-0.24421	0.08733	1.01913	-0.32970	-0.10161	1.32811	0.91512	0.62442
9	-10.6864	1	-1.64873	-1.55292	-0.93073	-1.83826	-1.12564	0.93932	0.70678	0.69050
10	-3.8644	1	-2.06994	-0.73017	-0.56868	-0.69484	-0.56194	0.27507	-0.85876	-0.66961
11	1.8994	1	-0.27581	0.00839	-0.68049	-0.71346	0.77513	-0.71688	-1.11969	-1.07921
12	4.2997	1	1.94553	1.25333	0.63126	1.18097	1.01464	0.71890	1.08941	0.68681
13	-3.1161	1	0.09988	-0.52021	-0.06674	-0.75591	-1.69204	-1.19763	-1.39856	-0.70608
14	-0.7396	1	0.18223	-0.15290	-1.14951	-1.66991	0.36882	-0.41121	0.67547	-0.51172
15	10.3569	1	0.13248	0.78954	1.55271	1.14774	2.10558	2.11764	1.34703	0.77513
16	-7.3901	1	-0.73099	-1.48944	0.30929	-0.69652	-0.85665	-0.95344	0.33151	0.26903
17	-1.9307	1	0.70615	0.11229	1.62247	0.68463	0.83545	0.70869	0.51089	1.25189
18	-4.6856	1	-0.20059	-1.88203	-1.35791	-0.75413	-1.90923	-1.73238	-2.08176	-2.28782
19	-1.4566	1	-0.57142	0.50587	-0.60449	1.61618	0.07156	-1.57343	-0.41659	-1.76394
20	1.0057	1	0.42204	1.27075	0.57853	-0.20164	-0.90917	-0.04812	0.96019	-0.10481
21	-2.9322	1	0.30432	-0.08837	0.51017	-0.28884	-0.12737	-1.47037	0.85729	-1.91170
22	7.2849	1	0.81053	0.26791	1.41117	1.42130	1.82799	0.04652	1.04482	1.13353
23	-3.2717	1	-1.92164	-1.34038	0.15457	-1.34006	-1.38328	-1.59067	-2.04336	-0.36909
24	4.4061	1	0.11616	-0.28016	-0.42876	0.44477	-0.86475	-1.50895	-0.10133	0.40942
25	-1.5366	1	-1.76358	-0.87299	0.50043	-1.54735	0.00474	-1.28116	-1.40051	-0.21098
26	-7.2694	1	-1.47401	-0.36263	-0.09214	-0.93846	0.33981	-0.63277	-1.74654	0.53202
27	-9.2621	1	-1.12125	-0.44434	0.35428	0.46054	-1.13455	-0.78533	-1.99726	-1.47173
28	1.4037	1	-0.21425	0.14567	-0.27417	-0.45418	-0.51584	-0.38450	-1.34642	0.05772
29	1.6243	1	1.58070	0.59476	-0.38736	0.81322	1.27769	1.20860	-0.88698	-0.19220
30	0.8326	1	0.44701	-0.78834	0.96729	1.71914	-0.24969	0.50235	0.24479	0.60695

The SAS System

The CORR Procedure

9 Variables:	y x1 x2 x3 x4 x5 x6 x7 x8

Simple Statistics						
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
Y	30	-1.25540	5.07483	-37.66192	-11.22171	10.35692
x1	30	-0.27245	1.06834	-8.17337	-2.06994	1.94553
x2	30	-0.22728	0.88876	-6.81848	-1.88203	1.68691
x3	30	-0.09546	1.05019	-2.86385	-2.85873	1.62247
x4	30	-0.24871	1.03450	-7.46141	-1.83826	1.71914
x5	30	-0.33168	1.07225	-9.95049	-2.37542	2.10558
x6	30	-0.22095	1.00422	-6.62862	-1.73238	2.11764
x7	30	-0.06958	1.16562	-2.08748	-2.08176	1.58019
x8	30	-0.31468	0.96289	-9.44043	-2.28782	1.25189

Pearson Correlation Coefficients, N = 30
Prob > |r| under H0: Rho=0

	y	x1	x2	x3	x4	x5	x6	x7	x8
y	1.00000	0.57671 0.0009	0.65147 <.0001	0.46981 0.0088	0.57085 0.0010	0.65086 <.0001	0.33199 0.0731	0.43421 0.0165	0.27125 0.1471
x1	0.57671 0.0009	1.00000	0.60233 0.0004	0.16068 0.3963	0.59358 0.0005	0.50504 0.0044	0.28866 0.1219	0.41364 0.0231	0.18641 0.3240
x2	0.65147 <.0001	0.60233 0.0004	1.00000	0.40504 0.0264	0.52789 0.0027	0.60798 0.0004	0.41701 0.0219	0.43459 0.0164	0.27143 0.1468
x3	0.46981 0.0088	0.16068 0.3963	0.40504 0.0264	1.00000	0.48222 0.0070	0.52113 0.0031	0.32964 0.0753	0.24392 0.1940	0.50805 0.0042
x4	0.57085 0.0010	0.59358 0.0005	0.52789 0.0027	0.48222 0.0070	1.00000	0.57133 0.0010	0.22786 0.2259	0.25379 0.1760	0.21098 0.2631
x5	0.65086 <.0001	0.50504 0.0044	0.60798 0.0004	0.52113 0.0031	0.57133 0.0010	1.00000	0.46839 0.0090	0.32360 0.0811	0.44559 0.0136
x6	0.33199 0.0731	0.28866 0.1219	0.41701 0.0219	0.32964 0.0753	0.22786 0.2259	0.46839 0.0090	1.00000	0.55433 0.0015	0.56236 0.0012
x7	0.43421 0.0165	0.41364 0.0231	0.43459 0.0164	0.24392 0.1940	0.25379 0.1760	0.32360 0.0811	0.55433 0.0015	1.00000	0.26333 0.1597
x8	0.27125 0.1471	0.18641 0.3240	0.27143 0.1468	0.50805 0.0042	0.21098 0.2631	0.44559 0.0136	0.56236 0.0012	0.26333 0.1597	1.00000

The SAS System

The REG Procedure - Model: MODEL1
Dependent Variable: y

Number of Observations Read	30

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	449.00154	56.12519	3.96	0.0054
Error	21	297.86037	14.18383		
Corrected Total	29	746.86191			

Root MSE	3.76614	R-Square [29]	0.6012
Dependent Mean	-1.25540	Adj R-Sq	0.4493
Coeff Var	-299.99605		

Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-0.21930	0.75853	-0.29	0.7753
x1	1	0.85459	0.99283	0.86	0.3991
x2	1	1.32313	1.15918	1.14	0.2665
x3	1	0.82272	0.94732	0.87	0.3950
x4	1	0.38899	1.00443	0.39	0.7025
x5	1	1.41216	0.98914	1.43	0.1681
x6	1	-0.47001	1.01543	-0.46	0.6482
x7	1	0.72431	0.77771	0.93	0.3623
x8	1	-0.27863	0.99350	-0.28	0.7819

The SAS System

The REG Procedure
Model: MODEL1
Dependent Variable: y

Number of Observations Read	30

Forward Selection: Step 1
Variable x2 Entered: R-Square = 0.4244 and C(p) = 4.3079

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	1	316.98029	316.98029	20.65	<.0001
Error	28	429.88162	15.35292		
Corrected Total	29	746.86191			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	-0.40992	0.73918	4.72169	0.31	0.5836
x2	3.71992	0.81868	316.98029	20.65	<.0001

Bounds on condition number: 1, 1

Forward Selection: Step 2
Variable x5 Entered: R-Square = 0.5274 and C(p) = 0.8856

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	2	393.88818	196.94409	15.06	<.0001
Error	27	352.97373	13.07310		
Corrected Total	29	746.86191			

Variable	**Parameter Estimate**	**Standard Error**	**Type II SS**	**F Value**	**Pr > F**
Intercept	-0.09435	0.69439	0.24136	0.02	0.8929
x2	2.31678	0.95151	77.50373	5.93	0.0218
x5	1.91292	0.78868	76.90789	5.88	0.0223

Bounds on condition number: 1.5864, 6.3456
No other variable met the 0.0500 significance level for entry into the model.

Summary of Forward Selection							
Step	**Variable Entered**	**Number Vars In**	**Partial R-Square**	**Model R-Square**	**C(p)**	**F Value**	**Pr > F**
1	x2	1	0.4244	0.4244	4.3079	20.65	<.0001
2	x5	2	0.1030	0.5274	0.8856	5.88	0.0223

Note that: $F = t^2$ (See the separate t values from the output in IML)

Chapter 6

SIMULATION RESULTS

The properties of forward regression will be evaluated and compared to the full model in terms of the following:

- The probability that irrelevant variables ($\beta_j = 0$) will be included in the model;
- The probability that relevant variables ($\beta_j = 1$) will be included from the model;
- The mean-squared error of the regression coefficients;
- The relative frequencies of the F – probability in the full model is also illustrated.

Table 1 represents the number of x's included in the model with each simulation. All the cases therefore add up to 10 000, the number of simulations, times the number of variables.

Table 2 represents the total number of b_j's=1 being correctly classified, whereas table 3, represents the total number of b_j's=0 being incorrectly classified.

Histogram graphs of table 2 and 3 are given in chapter 7, graphs 1 and 2.

Table 4 represents the mean-squared error for the full- and forward selection models.

Table 5 represents the F-probability counts of the full model.

npcountxe				Table 1 – The number of x's selected in the model with forward regression																			
case	p	ave_k	n	0xs	1xs	2xs	3xs	4xs	5xs	6xs	7xs	8xs	9xs	10xs	11xse	12xs	13xs	14xs	15xs	16xs	17xs	18xs	19xs	20xs
1	4	0.6	8	4315	5084	550	50	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	4	1.0	16	1385	7411	1131	66	7	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	5	1.1	10	1310	6884	1627	157	22	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	5	1.6	20	84	4852	4476	557	30	1	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	6	0.8	6	3290	5461	1033	192	24	0	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	6	1.2	12	714	6785	2179	294	25	3	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
7	6	1.8	24	17	3464	5519	886	108	6	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
8	7	1.1	7	1491	6255	1828	348	70	8	0	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	7	1.6	14	94	4511	4367	886	128	12	2	0	.	.	.	.	.	.	.	.	.	.	.	.	.
10	7	2.5	28	0	503	5180	3616	630	67	4	0	.	.	.	.	.	.	.	.	.	.	.	.	.
11	8	1.3	8	964	6157	2236	519	100	22	2	0	0	.	.	.	.	.	.	.	.	.	.	.	.
12	8	1.8	16	43	3360	5067	1275	219	34	2	0	0	.	.	.	.	.	.	.	.	.	.	.	.
13	8	2.7	32	0	232	4236	4420	957	139	16	0	0	.	.	.	.	.	.	.	.	.	.	.	.
14	9	1.5	9	340	5237	3299	884	200	37	3	0	0	0	.	.	.	.	.	.	.	.	.	.	.
15	9	2.3	18	5	1159	5087	2969	663	102	14	1	0	0	.	.	.	.	.	.	.	.	.	.	.
16	9	3.4	36	0	7	937	4790	3351	777	129	9	0	0	.	.	.	.	.	.	.	.	.	.	.
17	10	1.7	10	163	4567	3779	1122	284	69	14	2	0	0	0	.	.	.	.	.	.	.	.	.	.
18	10	2.5	20	3	650	4747	3523	864	185	21	7	0	0	0	.	.	.	.	.	.	.	.	.	.
19	10	3.6	40	0	3	522	4149	4033	1091	179	21	1	1	0	.	.	.	.	.	.	.	.	.	.
20	11	1.5	6	873	5319	2472	898	438	0	0	0	0	0	0	0	.	.	.	.	.	.	.	.	.
21	11	2.0	11	57	2930	4521	1789	538	131	26	7	1	0	0	0	.	.	.	.	.	.	.	.	.
22	11	3.1	22	0	131	2615	4464	2157	519	91	17	6	0	0	0	.	.	.	.	.	.	.	.	.
23	11	4.5	44	0	1	49	1155	4289	3307	988	185	24	2	0	0	.	.	.	.	.	.	.	.	.
24	12	1.5	6	755	5124	2591	948	582	0	0	0	0	0	0	0	0	.	.	.	.	.	.	.	.
25	12	2.2	12	31	2315	4563	2152	642	227	56	10	4	0	0	0	0	.	.	.	.	.	.	.	.
26	12	3.3	24	0	55	1890	4535	2557	777	149	29	7	1	0	0	0	.	.	.	.	.	.	.	.
27	12	4.7	48	0	0	14	686	3842	3810	1319	292	35	2	0	0	0	.	.	.	.	.	.	.	.

npcountxe																							
Table 1 – The number of x's selected in the model with forward regression (continued)																							
case	**p**	**ave_k**	**n**	**0xs**	**1xs**	**2xs**	**3xs**	**4xs**	**5xs**	**6xs**	**7xs**	**8xs**	**9xs**	**10xs**	**11xse**	**12xs**	**13xs**	**14xs**	**15xs**	**16xs**	**17xs**	**18xs**	**19xs**	**20xs**
28	13	1.8	7	307	4398	3171	1345	511	268	0	0	0	0	0	0	0	0	.	.	.	.	.	.	.
29	13	2.5	13	6	1227	4344	2879	1075	356	80	26	4	3	0	0	0	0	.	.	.	.	.	.	.
31	13	5.6	52	0	0	1	59	1187	3809	3377	1239	287	36	4	1	0	0	.	.	.	.	.	.	.
32	14	1.9	7	292	4243	3129	1368	610	358	0	0	0	0	0	0	0	0	0	.	.	.	.	.	.
33	14	2.7	14	4	834	3958	3233	1339	454	130	35	9	2	2	0	0	0	0	.	.	.	.	.	.
34	14	4	28	0	6	357	2844	3878	2058	643	164	39	10	1	0	0	0	0	.	.	.	.	.	.
35	14	5.8	56	0	0	0	19	733	3308	3746	1668	442	75	8	1	0	0	0	.	.	.	.	.	.
36	15	2.2	8	79	3178	3665	1783	768	312	215	0	0	0	0	0	0	0	0	0	.	.	.	.	.
37	15	3.1	15	0	331	2972	3763	1881	711	226	82	27	6	0	1	0	0	0	0	.	.	.	.	.
38	15	4.6	30	0	0	69	1347	3496	3112	1390	457	102	22	1	4	0	0	0	0	.	.	.	.	.
39	15	6.7	60	0	0	0	1	86	1081	3423	3348	1552	425	72	11	1	0	0	0	.	.	.	.	.
40	16	2.3	8	70	2983	3592	1825	870	346	314	0	0	0	0	0	0	0	0	0	0	.	.	.	.
41	16	3.3	16	1	217	2488	3734	2217	891	321	96	25	8	2	0	0	0	0	0	0	.	.	.	.
42	16	4.9	32	0	0	24	896	3122	3272	1802	650	183	42	7	0	2	0	0	0	0	.	.	.	.
43	16	6.9	64	0	0	0	0	38	728	2800	3703	1955	634	125	15	2	0	0	0	0	.	.	.	.
44	17	2.6	9	18	1988	3748	2280	1079	503	217	167	0	0	0	0	0	0	0	0	0	0	.	.	.
45	17	3.6	17	0	58	1524	3548	2767	1347	488	197	55	13	2	0	1	0	0	0	0	0	.	.	.
46	17	5.5	34	0	0	7	270	1791	3225	2733	1344	451	129	41	9	0	0	0	0	0	0	.	.	.
47	17	7.9	68	0	0	0	0	0	86	905	2803	3496	1959	614	118	15	4	0	0	0	0	.	.	.
48	18	2.6	9	20	1884	3727	2233	1134	526	270	206	0	0	0	0	0	0	0	0	0	0	0	.	.
49	18	3.8	18	0	29	1178	3342	2984	1524	629	213	72	19	7	3	0	0	0	0	0	0	0	.	.
50	18	5.7	36	0	0	4	170	1339	3115	2889	1635	583	196	57	8	4	0	0	0	0	0	0	.	.
51	18	8.1	72	0	0	0	0	0	50	615	2451	3581	2246	824	199	32	2	0	0	0	0	0	.	.
52	19	2.9	10	6	1179	3395	2676	1440	695	316	168	125	0	0	0	0	0	0	0	0	0	0	0	.
53	19	4.2	19	0	11	597	2672	3158	2016	939	424	129	45	6	2	1	0	0	0	0	0	0	0	.
54	19	6.4	38	0	0	0	28	552	2008	3093	2439	1259	441	139	33	8	0	0	0	0	0	0	0	.
55	19	9	76	0	0	0	0	0	4	63	751	2455	3424	2214	871	183	33	2	0	0	0	0	0	.
56	20	3	10	7	993	3297	2709	1542	728	380	170	174	0	0	0	0	0	0	0	0	0	0	0	0
57	20	4.4	20	0	5	438	2339	3247	2129	1119	459	192	47	13	8	1	2	1	0	0	0	0	0	0
58	20	6.6	40	0	0	0	15	313	1606	3063	2686	1414	627	205	56	12	3	0	0	0	0	0	0	0
59	20	9.3	80	0	0	0	0	0	1	44	512	2016	3287	2683	1085	305	58	8	1	0	0	0	0	0

	Number x_is			βi's=1		Table 2 - This is the x's which is included when it should be included														Total
case	p	ave_k	n	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	TOTAL	prob
1	4	0.6	8	2339	2347			.	.	.	.	.	.	.	.	.	.	.	4686	0.2343
2	4	1	16	4068	4095			.	.	.	.	.	.	.	.	.	.	.	8163	0.4082
3	5	1.1	10	2936	2866	2862			.	.	.	.	.	.	.	.	.	.	8664	0.2888
4	5	1.6	20	4624	4521	4652			.	.	.	.	.	.	.	.	.	.	13797	0.4599
5	6	0.8	6	1860	1797	1852				.	.	.	.	.	.	.	.	.	5509	0.1836
6	6	1.2	12	3173	3074	3094				.	.	.	.	.	.	.	.	.	9341	0.3114
7	6	1.8	24	5017	4977	5019				.	.	.	.	.	.	.	.	.	15013	0.5004
8	7	1.1	7	2041	2048	2103	2006				.	.	.	.	.	.	.	.	8198	0.2050
9	7	1.6	14	3432	3328	3351	3351				.	.	.	.	.	.	.	.	13462	0.3366
10	7	2.5	28	5474	5467	5410	5511				.	.	.	.	.	.	.	.	21862	0.5466
11	8	1.3	8	2213	2174	2218	2153					.	.	.	.	.	.	.	8758	0.2190
12	8	1.8	16	3532	3627	3666	3685					.	.	.	.	.	.	.	14510	0.3628
13	8	2.7	32	5752	5880	5829	5679					.	.	.	.	.	.	.	23140	0.5785
14	9	1.5	9	2209	2312	2237	2259	2310					.	.	.	.	.	.	11327	0.2265
15	9	2.3	18	3886	3868	3852	3857	3804					.	.	.	.	.	.	19267	0.3853
16	9	3.4	36	6252	6176	6138	6114	6146					.	.	.	.	.	.	30826	0.6165
17	10	1.7	10	2317	2393	2398	2380	2426						.	.	.	.	.	11914	0.2383
18	10	2.5	20	3963	4047	4116	4025	4048						.	.	.	.	.	20199	0.4040
19	10	3.6	40	6393	6417	6488	6429	6454						.	.	.	.	.	32181	0.6436
20	11	1.5	6	1572	1579	1652	1529	1650	1565						.	.	.	.	9547	0.1591
21	11	2	11	2411	2448	2499	2548	2504	2469						.	.	.	.	14879	0.2480
22	11	3.1	22	4269	4273	4256	4117	4197	4185						.	.	.	.	25297	0.4216
23	11	4.5	44	6738	6710	6731	6707	6681	6667						.	.	.	.	40234	0.6706
24	12	1.5	6	1606	1596	1564	1562	1545	1553							.	.	.	9426	0.1571
25	12	2.2	12	2490	2598	2549	2623	2503	2678							.	.	.	15441	0.2574
26	12	3.3	24	4451	4428	4350	4340	4404	4474							.	.	.	26447	0.4408
27	12	4.7	48	6998	6983	6974	6949	6889	6974							.	.	.	41767	0.6961
28	13	1.8	7	1711	1714	1591	1710	1750	1695	1651							.	.	11822	0.1689
29	13	2.5	13	2606	2689	2640	2681	2652	2676	2686							.	.	18630	0.2661

	Number x_is			bi's=1		Table 2 - This is the x's which is included when it should be included																		Total
case	p	ave_k	n	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	11	12	13	14	15	16	17	18	19	20	TOTAL	prob
30	13	3.9	26	4575	4600	4623	4536	4544	4608	4532							.	.	.	.	.	.	.	32018	0.457
31	13	5.6	52	7223	7218	7173	7256	7123	7233	7173							.	.	.	.	.	.	.	50399	0.720
32	14	1.9	7	1603	1659	1716	1662	1629	1664	1606								.	.	.	.	.	.	11539	0.165
33	14	2.7	14	2788	2724	2737	2720	2743	2745	2752								.	.	.	.	.	.	19209	0.274
34	14	4	28	4740	4694	4689	4757	4706	4687	4688								.	.	.	.	.	.	32961	0.471
35	14	5.8	56	7484	7458	7409	7444	7426	7420	7473								.	.	.	.	.	.	52114	0.744
36	15	2.2	8	1824	1819	1717	1687	1788	1678	1704	1787								.	.	.	.	.	14004	0.175
37	15	3.1	15	2792	2808	2849	2843	2810	2811	2867	2706								.	.	.	.	.	22486	0.281
38	15	4.6	30	4755	4797	4816	4780	4753	4788	4883	4863								.	.	.	.	.	38435	0.480
39	15	6.7	60	7648	7629	7672	7575	7664	7593	7502	7601								.	.	.	.	.	60884	0.761
40	16	2.3	8	1703	1755	1690	1807	1733	1751	1691	1759									.	.	.	.	13889	0.174
41	16	3.3	16	2908	2941	2939	2867	2846	2903	2892	2991									.	.	.	.	23287	0.291
42	16	4.9	32	5022	4972	5039	5001	4993	5014	4898	5083									.	.	.	.	40022	0.500
43	16	6.9	64	7859	7867	7859	7828	7901	7790	7830	7809									.	.	.	.	62743	0.784
44	17	2.6	9	1819	1917	1855	1884	1800	1751	1826	1811	1838									.	.	.	16501	0.183
45	17	3.6	17	2955	2989	2989	2947	2994	2965	2954	2961	2960									.	.	.	26714	0.297
46	17	5.5	34	5052	5077	5120	5069	5157	5056	5065	5131	5174									.	.	.	45901	0.510
47	17	7.9	68	8041	7976	8023	7907	7999	7979	8055	7987	8057									.	.	.	72024	0.800
48	18	2.6	9	1841	1841	1892	1790	1754	1755	1778	1810	1744										.	.	16205	0.180
49	18	3.8	18	3080	3043	3024	3081	3020	3050	3068	3065	2971										.	.	27402	0.304
50	18	5.7	36	5269	5237	5299	5239	5278	5296	5266	5229	5192										.	.	47305	0.526
51	18	8.1	72	8159	8192	8158	8160	8212	8143	8126	8202	8166										.	.	73518	0.817
52	19	2.9	10	1910	1841	1976	1831	1841	1921	1849	1880	1890	1863										.	18802	0.188
53	19	4.2	19	3095	3041	3129	3097	3072	3116	3086	3048	3075	3131										.	30890	0.309
54	19	6.4	38	5392	5335	5350	5333	5401	5296	5408	5415	5353	5238										.	53521	0.535
55	19	9	76	8219	8321	8252	8287	8209	8292	8310	8302	8331	8354										.	82877	0.829
56	20	3	10	1881	1783	1907	1857	1836	1817	1871	1898	1835	1942											18627	0.186
57	20	4.4	20	3181	3121	3146	3151	3073	3189	3104	3131	3217	3163											31476	0.315
58	20	6.6	40	5490	5525	5416	5546	5546	5504	5519	5469	5478	5526											55019	0.550
59	20	9.3	80	8475	8509	8453	8502	8518	8485	8468	8460	8511	8479											84860	0.849

	Number x_is			bi's=0		Table 3 - This is the x's included where it should not be included														Total
case	p	ave_k	n	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	TOTAL	prob
1	4	0.6	8			829	823	.	.	.	.	.	.	.	.	.	.	.	1652	0.0826
2	4	1	16			862	874	.	.	.	.	.	.	.	.	.	.	.	1736	0.0868
3	5	1.1	10				1017	1016	.	.	.	.	.	.	.	.	.	.	2033	0.1017
4	5	1.6	20				914	889	.	.	.	.	.	.	.	.	.	.	1803	0.0902
5	6	0.8	6				942	890	858	.	.	.	.	.	.	.	.	.	2690	0.0897
6	6	1.2	12				958	922	919	.	.	.	.	.	.	.	.	.	2799	0.0933
7	6	1.8	24				869	870	870	.	.	.	.	.	.	.	.	.	2609	0.0870
8	7	1.1	7					966	1044	1067	.	.	.	.	.	.	.	.	3077	0.1026
9	7	1.6	14					1072	973	980	.	.	.	.	.	.	.	.	3025	0.1008
10	7	2.5	28					891	942	895	.	.	.	.	.	.	.	.	2728	0.0909
11	8	1.3	8					971	976	987	1016	.	.	.	.	.	.	.	3950	0.0988
12	8	1.8	16					1018	935	965	949	.	.	.	.	.	.	.	3867	0.0967
13	8	2.7	32					855	854	839	895	.	.	.	.	.	.	.	3443	0.0861
14	9	1.5	9						1075	1001	1020	1067	.	.	.	.	.	.	4163	0.1041
15	9	2.3	18						1085	1003	1070	1068	.	.	.	.	.	.	4226	0.1057
16	9	3.4	36						865	882	898	906	.	.	.	.	.	.	3551	0.0888
17	10	1.7	10						1018	1039	1059	1050	990	.	.	.	.	.	5156	0.1031
18	10	2.5	20						1042	1015	1013	988	1012	.	.	.	.	.	5070	0.1014
19	10	3.6	40						822	811	822	820	863	.	.	.	.	.	4138	0.0828
20	11	1.5	6							1041	1054	1055	1014	998	.	.	.	.	5162	0.1032
21	11	2	11							1095	1081	1043	1134	1127	.	.	.	.	5480	0.1096
22	11	3.1	22							1077	1100	1101	1063	1051	.	.	.	.	5392	0.1078
23	11	4.5	44							882	872	865	914	921	.	.	.	.	4454	0.0891
24	12	1.5	6							1041	983	1016	986	1000	1026	.	.	.	6052	0.1009
25	12	2.2	12							1080	1102	1135	1143	1046	1091	.	.	.	6597	0.1100
26	12	3.3	24							1022	1031	1073	1061	1072	1009	.	.	.	6268	0.1045
27	12	4.7	48							791	825	832	841	852	852	.	.	.	4993	0.0832
28	13	1.8	7								1066	1039	1040	1046	1029	1117	.	.	6337	0.1056
29	13	2.5	13								1091	1130	1114	1093	1189	1106	.	.	6723	0.1121

	Number x_is			bi's=0			Table 3 - This is the x's included where it should not be included (continued)																	Total
case	p	ave_k	n	x1	2	3	4	5	6	7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	x18	x19	x20	TOTAL	prob
30	13	3.9	26								1104	1102	1040	1077	1106	1053	.	.	.	.	.	.	.	6482	0.108
31	13	5.6	52								876	869	821	881	870	862	.	.	.	.	.	.	.	5179	0.086
32	14	1.9	7								1028	1042	1042	1017	1099	1016	1052	.	.	.	.	.	.	7296	0.104
33	14	2.7	14								1141	1159	1166	1163	1146	1103	1123	.	.	.	.	.	.	8001	0.114
34	14	4	28								1082	1044	1036	1101	1053	1134	1061	.	.	.	.	.	.	7511	0.107
35	14	5.8	56								804	849	815	868	844	838	851	.	.	.	.	.	.	5869	0.084
36	15	2.2	8									1179	1107	1101	1094	1085	1103	1106	.	.	.	.	.	7775	0.111
37	15	3.1	15									1247	1170	1192	1232	1202	1197	1128	.	.	.	.	.	8368	0.120
38	15	4.6	30									1182	1124	1119	1099	1140	1141	1090	.	.	.	.	.	7895	0.113
39	15	6.7	60									825	828	862	864	814	842	901	.	.	.	.	.	5936	0.085
40	16	2.3	8									1123	1107	1121	1091	1063	1115	1100	1127	.	.	.	.	8847	0.111
41	16	3.3	16									1174	1163	1164	1168	1177	1128	1179	1168	.	.	.	.	9321	0.117
42	16	4.9	32									1166	1101	1145	1105	1076	1090	1061	1116	.	.	.	.	8860	0.111
43	16	6.9	64									826	821	806	829	884	840	790	759	.	.	.	.	6555	0.082
44	17	2.6	9										1130	1133	1153	1182	1165	1122	1135	1105	.	.	.	9125	0.114
45	17	3.6	17										1197	1257	1173	1187	1210	1249	1260	1202	.	.	.	9735	0.122
46	17	5.5	34										1124	1208	1162	1149	1166	1116	1186	1185	.	.	.	9296	0.116
47	17	7.9	68										851	812	834	876	834	836	834	849	.	.	.	6726	0.084
48	18	2.6	9										1059	1096	1120	1108	1106	1120	1174	1154	1123	.	.	10060	0.112
49	18	3.8	18										1181	1144	1195	1215	1242	1198	1209	1155	1141	.	.	10680	0.119
50	18	5.7	36										1125	1141	1161	1050	1123	1116	1086	1128	1127	.	.	10057	0.112
51	18	8.1	72										837	811	808	773	782	827	809	814	819	.	.	7280	0.081
52	19	2.9	10											1151	1137	1176	1175	1190	1196	1114	1202	1161	.	10502	0.117
53	19	4.2	19											1271	1249	1264	1254	1268	1194	1247	1211	1218	.	11176	0.124
54	19	6.4	38											1130	1181	1134	1111	1110	1095	1217	1150	1204	.	10332	0.115
55	19	9	76											824	848	829	859	837	881	840	840	850	.	7608	0.085
56	20	3	10											1217	1167	1213	1194	1179	1193	1118	1162	1162	1152	11757	0.118
57	20	4.4	20											1286	1198	1209	1187	1304	1216	1236	1216	1182	1177	12211	0.122
58	20	6.6	40											1167	1123	1113	1159	1146	1056	1188	1100	1109	1131	11292	0.113
59	20	9.3	80											830	808	786	785	835	816	752	793	799	806	8010	0.080

Table 4 – The mean-squared error of $\widehat{\beta}$ from β for 10 000 simulations for different values of n and p for the forward- and full regression models.

Case	p	Average k	n	Mean-squared error for Forward selection	Mean-squared error for the Full model	$\frac{n}{p}$
1	4	0.6	8	1.0960	3.5198	2
2	4	1	16	0.6749	0.7257	4
3	5	1.1	10	1.2283	2.4991	2
4	5	1.6	20	0.7221	0.5689	4
5	6	0.8	6	1.5477	.	1
6	6	1.2	12	0.9600	1.8062	2
7	6	1.8	24	0.5321	0.4650	4
8	7	1.1	7	1.7686	.	1
9	7	1.7	14	1.0385	1.4929	2
10	7	2.4	28	0.5429	0.3976	4
11	8	1.3	8	1.4521	.	1
12	8	1.8	16	0.8402	1.2642	2
13	8	2.7	32	0.4275	0.3446	4
14	9	1.6	9	1.6142	.	1
15	9	2.3	18	0.8876	1.0918	2
16	9	3.4	36	0.4241	0.3036	4
17	10	1.7	10	1.3937	.	1
18	10	2.5	20	0.7532	0.9677	2
19	10	3.6	40	0.3513	0.2727	4
20	11	1.5	6	2.5542	.	0.5
21	11	2	11	1.4632	.	1
22	11	3.1	22	0.7817	0.8492	2
23	11	4.5	44	0.3516	0.2499	4
24	12	1.5	6	2.3981	.	0.5
25	12	2.2	12	1.2885	.	1
26	12	3.3	24	0.6810	0.7669	2
27	12	4.7	48	0.2916	0.2273	4
28	13	1.8	7	2.4179	.	0.5
29	13	2.5	13	1.3679	.	1
30	13	3.9	26	0.7007	0.7015	2
31	13	5.6	52	0.2879	0.2094	4
32	14	1.9	7	2.2781	.	0.5
33	14	2.7	14	1.2219	.	1
34	14	4.1	28	0.6228	0.6405	2
35	14	5.8	56	0.2468	0.1941	4
36	15	2.2	8	2.2874	.	0.5
37	15	3.1	15	1.2847	.	1
38	15	4.7	30	0.6415	0.5978	2
39	15	6.7	60	0.2412	0.1803	4

Table 4 - The means-squared error of $\widehat{\beta}$ from β for 10 000 simulations for different values of n and p for the forward- and full regression models - (continued).

Case	p	Average k	n	Mean-squared error for Forward selection	Mean-squared error for the Full model	$\frac{n}{p}$
40	16	2.2	8	2.2169	.	0.5
41	16	3.3	16	1.1677	.	1
42	16	4.9	32	0.5672	0.5479	2
43	16	6.9	64	0.2085	0.1695	4
44	17	2.5	9	2.2207	.	0.5
45	17	3.6	17	1.2249	.	1
46	17	5.5	34	0.5947	0.5211	2
47	17	7.9	68	0.2025	0.1585	4
48	18	2.6	9	2.1207	.	0.5
49	18	3.8	18	1.1142	.	1
50	18	5.7	36	0.5334	0.4859	2
51	18	8.1	72	0.1781	0.1502	4
52	19	2.9	10	2.1036	.	0.5
53	19	4.2	19	1.1745	.	1
54	19	6.4	38	0.5439	0.4607	2
55	19	9.1	76	0.1738	0.1413	4
56	20	3	10	2.0532	.	0.5
57	20	4.4	20	1.0779	.	1
58	20	6.6	40	0.4945	0.4335	2
59	20	9.3	80	0.1517	0.1346	4

Discussions of this table are given and represented in graphs 3 to 9.

Tables 5 – The proportion F-probability counts of the full linear regression model from the ANOVA table out of 10 000 simulations within the three intervals, p > 0.05, 0.01 < p < 0.05 and p < 0.01. The proportional counts result for the cases where n = 2p and n = 4p.

case	p	n	<0.01	0.01<p<0.05	>0.05
1	4	8	0.0732	0.1399	0.7869
2	4	16	0.2882	0.2705	0.4413
3	5	10	0.1537	0.2481	0.5982
4	5	20	0.739	0.1659	0.0951
6	6	12	0.1862	0.2668	0.547
7	6	24	0.8206	0.1263	0.0531
9	7	14	0.4162	0.3076	0.2762
10	7	28	0.9836	0.0133	0.0031
12	8	16	0.4913	0.2981	0.2106
13	8	32	0.9944	0.0049	0.0007
15	9	18	0.7634	0.175	0.0616
16	9	36	0.9996	0.0004	0
18	10	20	0.8218	0.141	0.0372
19	10	40	1	0	0
22	11	22	0.9517	0.0416	0.0067
23	11	44	1	0	0
26	12	24	0.9727	0.0241	0.0032
27	12	48	1	0	0
30	13	26	0.996	0.0036	0.0004
31	13	52	1	0	0
34	14	28	0.9981	0.0018	0.0001
35	14	56	1	0	0
38	15	30	0.9998	0	0.0002
39	15	60	1	0	0
42	16	32	1	0	0
43	16	64	1	0	0
46	17	34	1	0	0
47	17	68	1	0	0
50	18	36	0.9999	0.0001	0
51	18	72	1	0	0
54	19	38	1	0	0
55	19	76	1	0	0
58	20	40	1	0	0
59	20	80	1	0	0

The full regression model $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ is used to determine the least squares estimates of the full model with $\widehat{\boldsymbol{\beta}} = (\boldsymbol{X`X})^{-1}\boldsymbol{X`y}$[81]. This is applied to the simulated data. The F-probability is determined for each simulation and each case (n and p).

That is, the hypothesis H_0: $\beta_1 = \beta_2 = \cdots . \beta_p = 0$ is being tested and the F_0 statistic is being computed. The H_0 hypothesis is being rejected if:

$$F_0 > F_{\propto,p,n-p-1}$$

The F-probabilities is grouped into the classes $p < 0.01$, $0.01 < p < 0.05$ and $p > 0.05$. The proportion of the counts within the classes $p < 0.01$, $0.01 < p < 0.05$ and $p > 0.05$ for each n and p case (over 10 000 simulations) is determined.

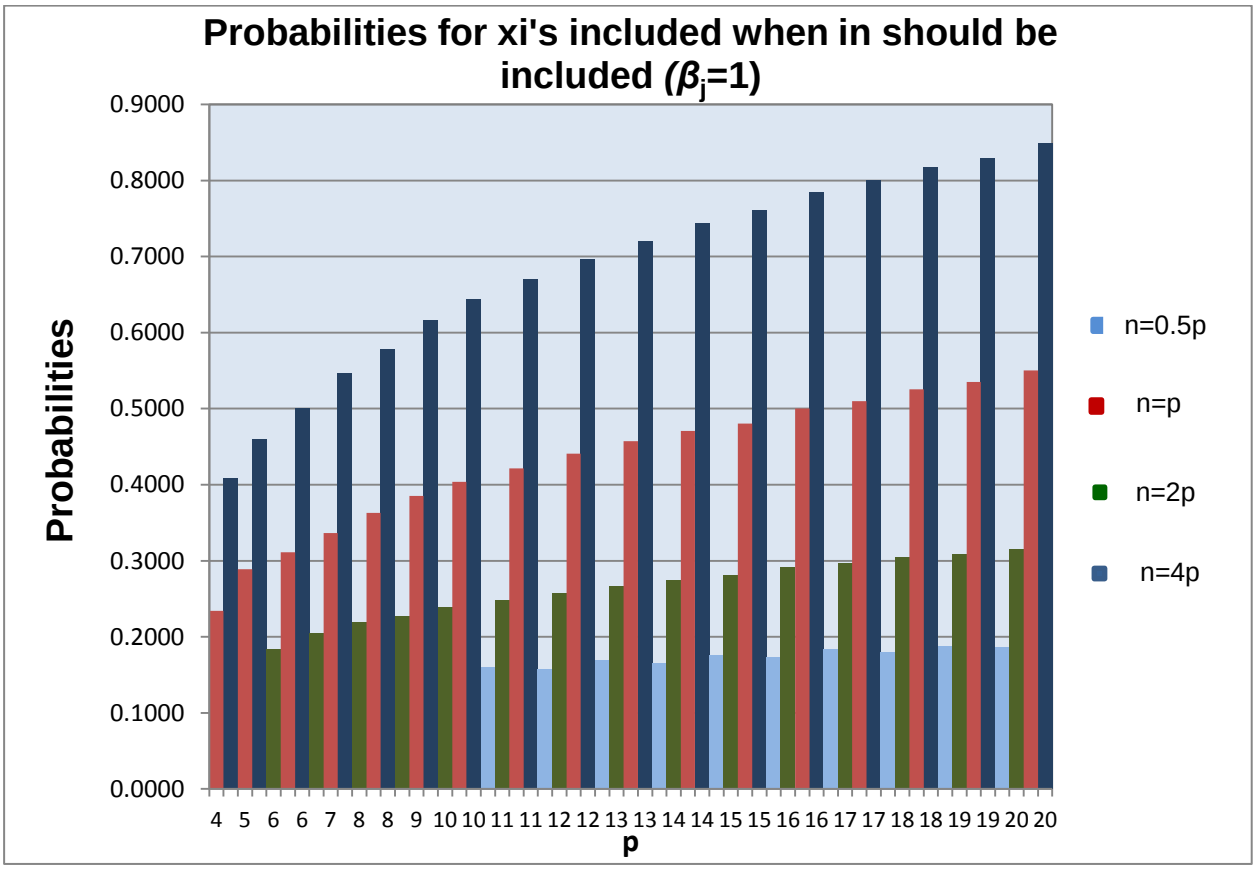
Discussions of the table above are given and represented in graphs 10 to 11.

[81] The estimated *β, $\hat{\beta}$ for the full model*

Chapter 7

CONCLUSIONS AND GRAPHS

Graph 1. This represents the probabilities for x_i's being correctly classified where β_j=1 for the forward selection analysis.



See Table 2 in Chapter 6 for data.

Each of the probability cases (for each n versus p case), is determined by the total number of x_i's being correctly classified (where β_j =1) divided by the total number of simulations, which is multiplied by the number of variables.

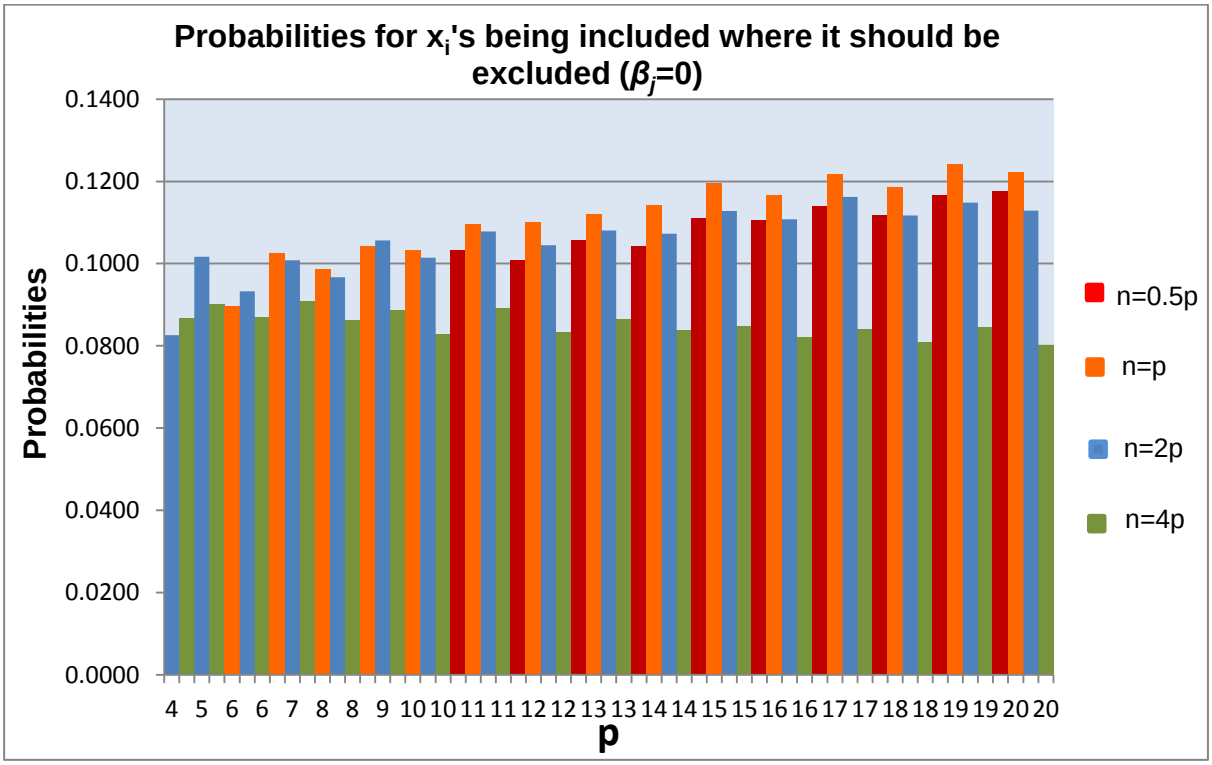
From the above-mentioned graph it is evident that the probabilities (for x_i 's being correctly classified) form 4 distinct groups, i.e. for each group, n=0.5p, p, 2p and 4p they form different probability groups. For a multiple linear forward regression analysis, the best probability for the x_i's to be correctly classified lies within the classification of the n = 4p group.

From the above one may deduce that in the situation presented by the present simulation, the probability for a correct classification is > 40% for n = 4p. For the cases where n = 4p and n > 36 (case 16), the probability of a correct classification > 60%. For the cases where n = 4p and n > 68 (case 47), the probability of a correct classification is more than 80%.

In order to obtain better accuracy in multiple regression it should be evident that n > 4p and n > 36. The probability of a correct classification will then be higher than 60%.

Looking at this graph (with specified conditions) it would not be recommended that n = 0.5p, where n ≤ 20. The probability of a correct classification is smaller than 20%, which is not advisable.

Graph 2. This represents the probabilities for x_i's being misclassified where $\beta_j = 0$.



See Table 3 in Chapter 6 for data.

Each of the probability cases is determined by the total number of x_i's being misclassified (where $\beta_j = 0$) divided by the total number of simulations, which is multiplied by the number of variables.

From the above one may deduce that in the situation presented by the present simulation forms 2 groups for the cases higher than 19 (0 < j < p). This is applicable for the groups, n = 0.5p, p and 2p forming one group and n = 4p forming the other probability group. For a multiple, linear forward regression analysis, the lowest probability for the x_i's to be misclassified lies within the classification of the n = 4p group; especially for the cases higher than 16 (p > 9).

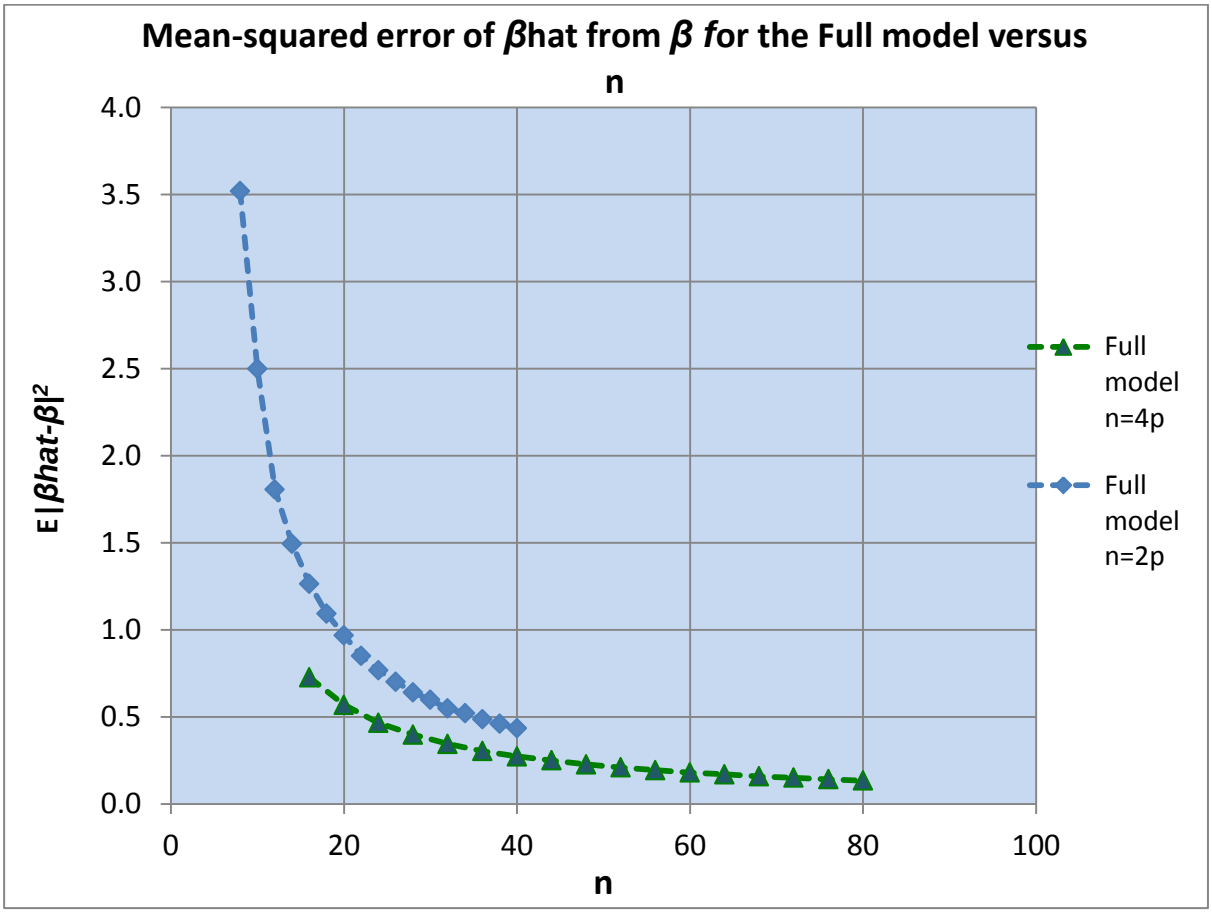
The misclassification for a multiple forward regression for n = 4p is less than 10%. From this graph it is also clear that the misclassification for p > 10 (cases >19) is slightly less than the rest of n = 4p.

To obtain better accuracy in multiple forward regression, it is recommended that n > 4p and n > 40.

Each test was done normally at α = 0.05, but the actual significance level seems to be between 0.08 and 0.12 depending on n and p. This indicates that forward selection is a reliable method to use, but one may have to adjust the significance level.

Different graphs of the mean-squared error of $\hat{\beta}$ is presented below: (See table 4)

Graph 3. The mean-squared error for $\hat{\beta}$ from β for the full regression model versus n for the different values of p.



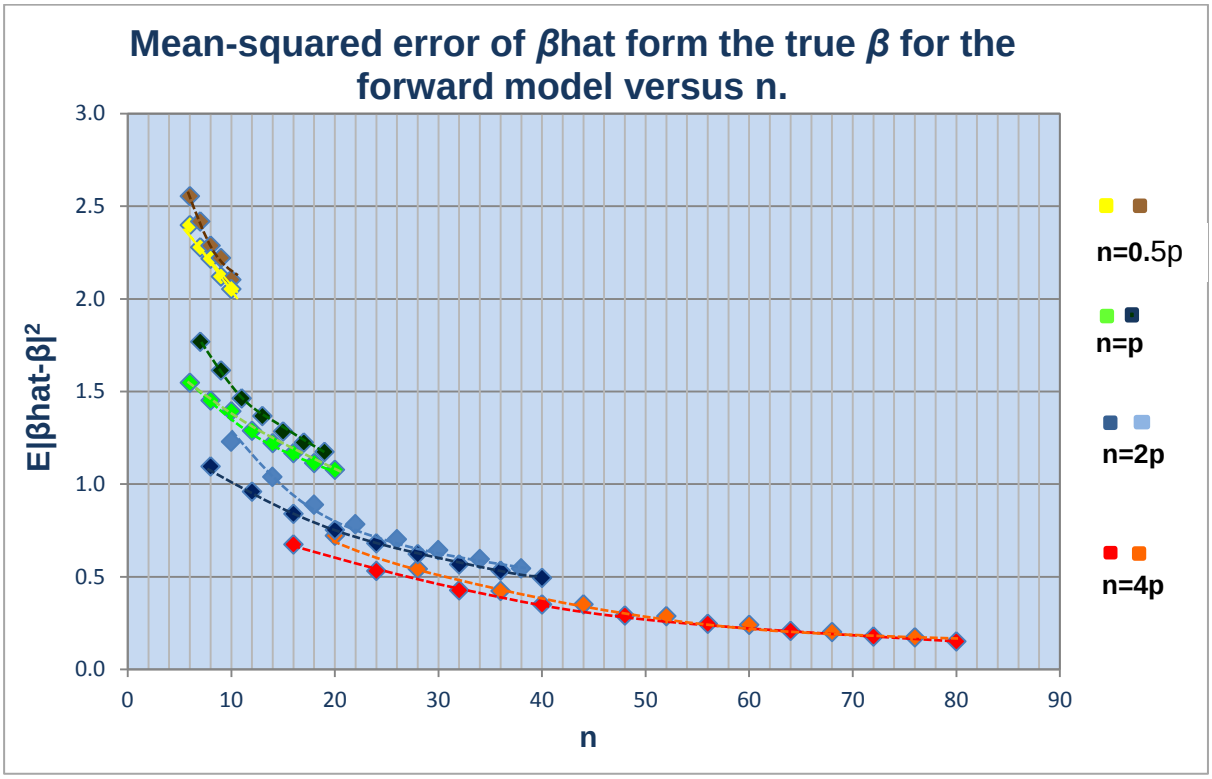
From 10 000 simulations, when n = 2p and n = 4p, the mean-squared error of the estimated coefficient $\hat{\beta}$ is determined for the full regression model. The full regression model can however not be done for the cases n = 0.5p as (X'X) is singular and the inverse does not exist. For n = p an exact fit would be obtained.

From graph 3 it is clear that the mean-squared error of the estimated coefficient $\hat{\beta}$ for the full regression model has an exponential curve when both p and n increase simultaneously. The blue and the green curves decrease, when n=2p and n=4p

increase respectively. There is not much difference in the mean-squared error of $\widehat{\boldsymbol{\beta}}$ between the curves when n = 2p and n = 4p for the same p value.

The multilpe linear regression for the full model is more accurate for higher n values. The mean-squared error for the cases where n = 4p is less and therefore more accurate.
(table 4)

Graph 4. This graph represents the mean-squared error of $\widehat{\boldsymbol{\beta}}$ from $\boldsymbol{\beta}$ for the forward selection model versus n, for different values p.



For n=4p: orange: p is odd.
red: p is even.
For n=2p: light blue: p is odd.
dark blue: p is even.
For n=p: dark green: p is odd.
light green: p is even.
For n=0.5p: brown: p is odd.
yellow: p is even.

For each of the n versus p ratio's the mean-squared error graphs form 4 different trends. For each of the n versus p ratio curves there are two disjoint curves. The reason for this is the following:

$\boldsymbol{\beta}_j$ = 1 is chosen for the first half of the $\boldsymbol{\beta}$ values and the remaining, $\boldsymbol{\beta}$'s = 0 where p is even. This results in one group, whereas the cases where p is an odd number, the first $\frac{p+1}{2}$, $\boldsymbol{\beta}$'s = 1 and the remaining $\boldsymbol{\beta}$'s = 0, results in another group.

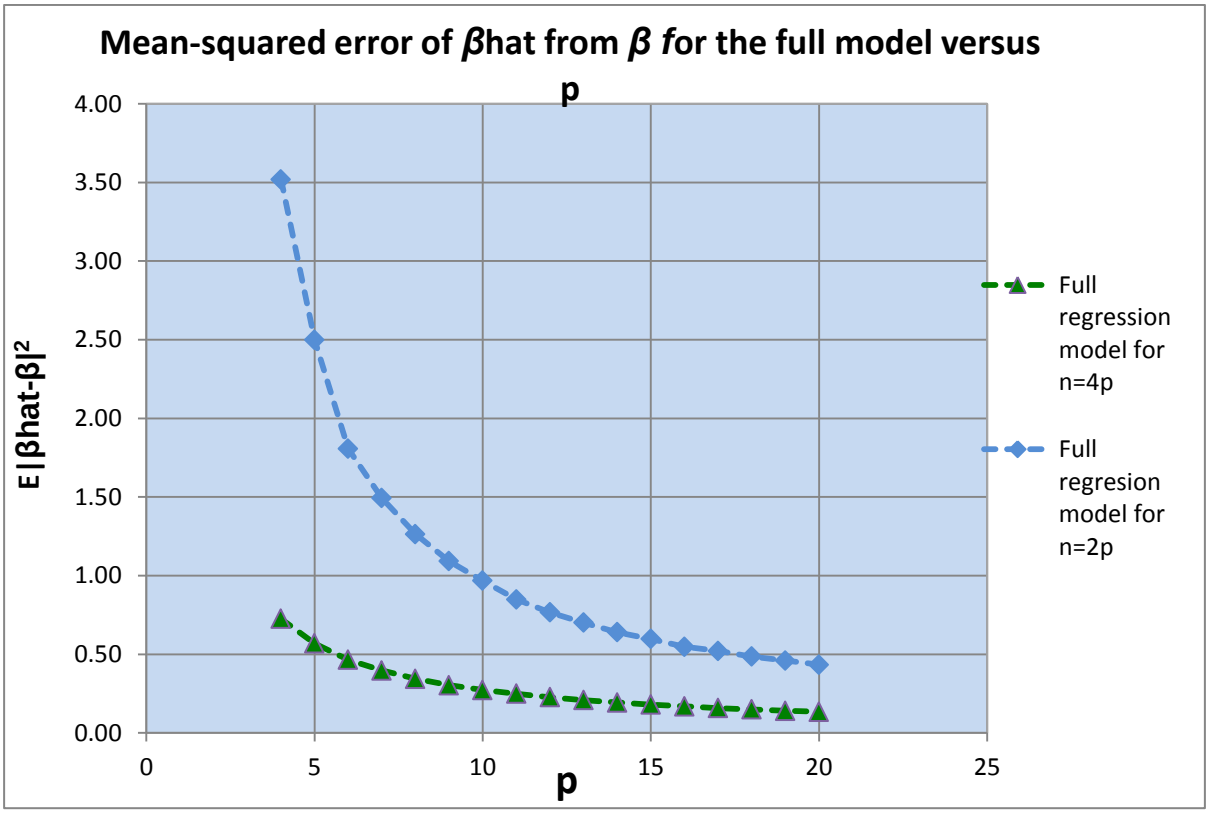
From Graph 4 the minimum sample size (n) for a specified p and the minimum sample size for a certain n versus p ratio (under normal conditions) can be determined.

The sample size, n must not be too small, otherwise the estimator will not be correct. Only a certain number of variables are kept in the model (k) for the forward selection, which is often much less than p - this is shown in graph 7.

In all the cases the mean-squared error decreases as n increases. To obtain a more accurate forward regression model, it is therefore recommended that n = 4p.

It is not advisable that n=0.5p and that n is too small. If this were the case, it is advised that the sample size n be increased, i.e. n must be greater than 10. (table 4)

Graph 5. This graph represents the mean-squared error of $\hat{\beta}$ for the full regression model versus p, for different values of n.

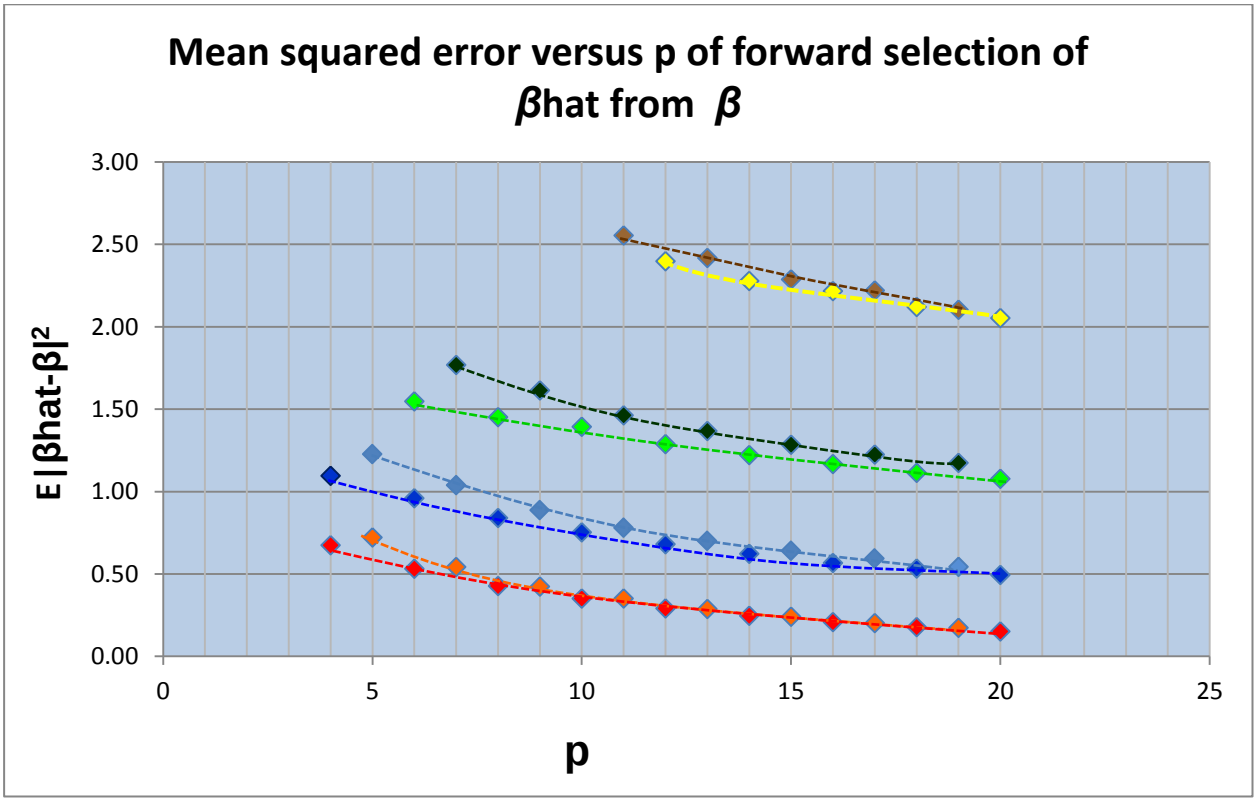


The mean-squared error for the estimated coefficient $\hat{\beta}$ from the true $\boldsymbol{\beta}$ for the full model versus p, where n > p+1 is represented in the graph above.

The graph formed 2 curves, the one for n = 2p and the other one for n = 4p. It forms exponential curves, similar than the plot of mean-squared error for the estimated coefficient $\hat{\beta}$ versus n.

For an accurate forward model n = 4p and p > 5 would be advisable. (table 4)

Graph 6. This graph represents the mean-squared error of $\widehat{\beta}$ for the forward selection model versus p, for different values of n.



The 4 four different n versus p groupings can be shown as:

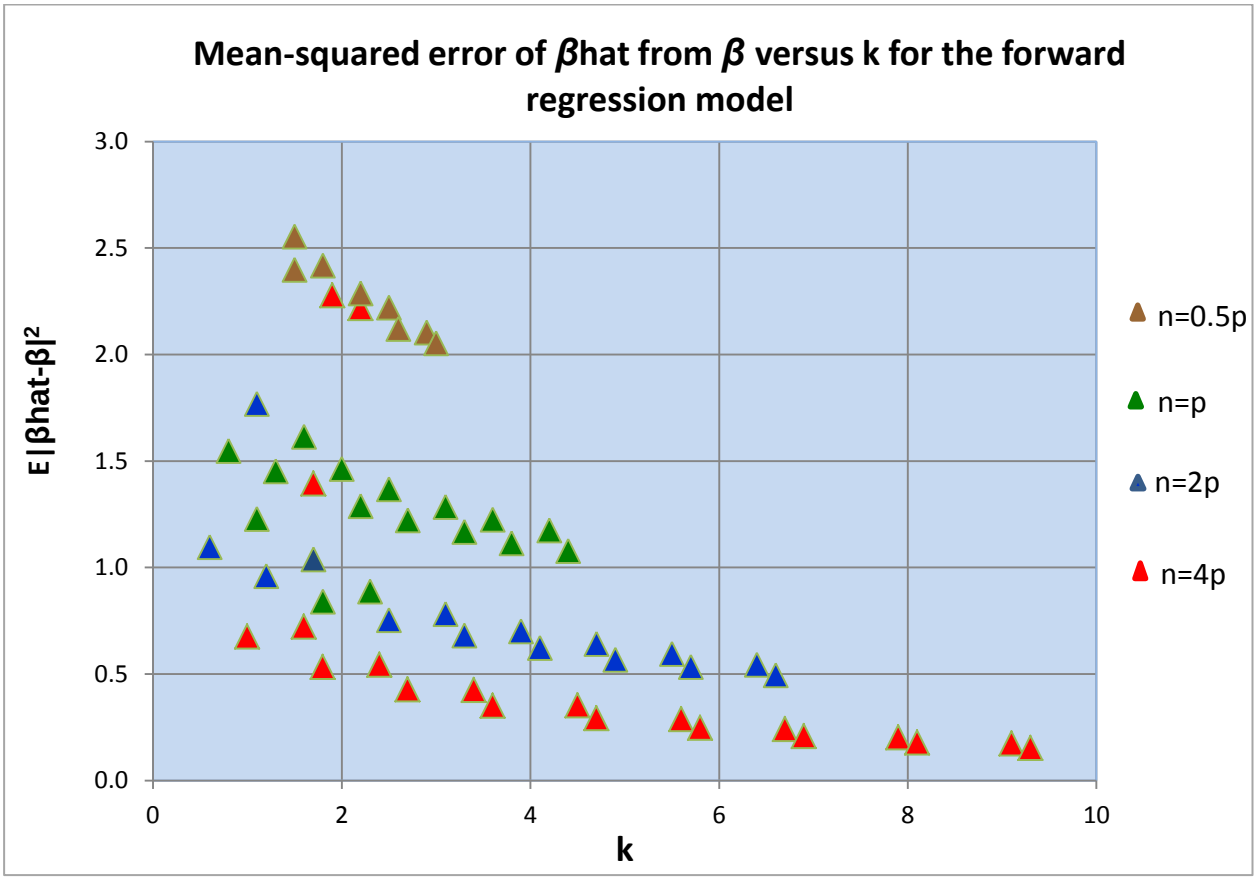
For n=4p: orange: p is odd.
red: p is even.
For n=2p: light blue: p is odd.
dark blue: p is even.
For n=p: dark green: p is odd.
light green: p is even.
For n=0.5p: brown: p is odd.
yellow: p is even.

The mean-squared error is intimately related to prediction accuracy. The lower the mean-squared the more accurate the model and estimates.

The three groups, n = p, n = 2p and n = 4p is grouped together and the group n = 0.5p from a group on its own. It is again advisable that $n \neq 0.5p$ is not used, especially when $p < 20$ (table 4).

The following graph illustrates a different point of view (when compared with Graph 6) when the mean-squared error is plotted against k, the number of variables in the forward selection for different values of n.

Graph 7. The mean-squared error of $\hat{\beta}$ for the forward selection model versus k (the number of variables to be in the model) for different values of n.

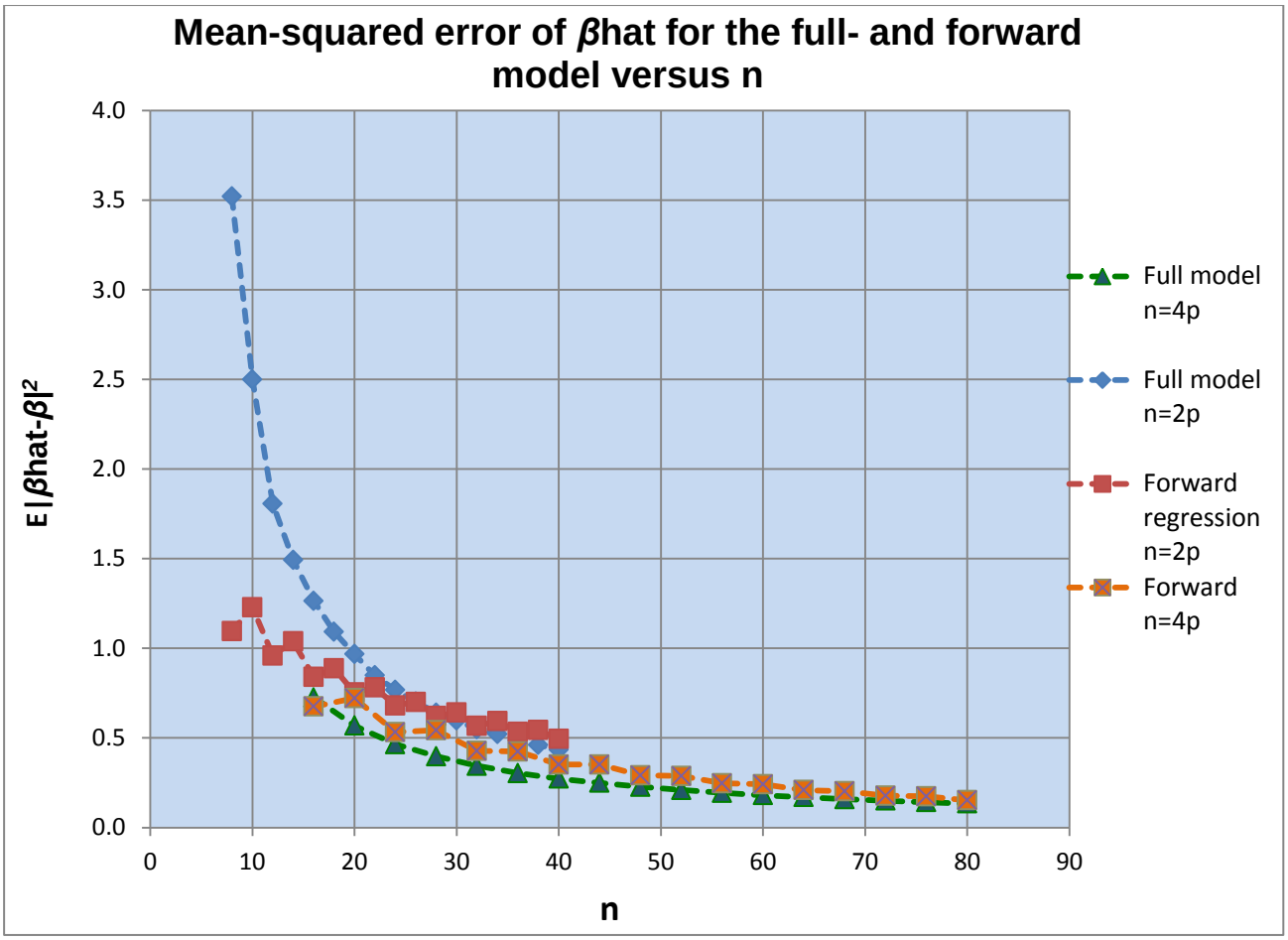


The graph forms similar patterns than the graph of the mean-squared error of $\hat{\beta}$ (graph 6) versus p. The different colour groups represent the groups where n = 0.5p, p, 2p and 4p.

Because k variables are selected out of the p values in forward regression analysis, the outcome of the forward selection model is determined by k, the number of variables included in the model. For a more accurate forward selection, k must be bigger than 4 and n = 2p or 4p. Thus p must be greater than 4 as well.

To obtain a more accurate forward regression model, it is again recommended that n = 4p. (table 4)

Graph 8. The graph represents the differences between the forward- and full model of the mean-squared error of the estimated coefficient, $\widehat{\beta}$. The independent variable here is the sample size (n).



This graph shows the difference in mean-squared error of the full- versus the forward model. The independent variable is here the number of variables (n).

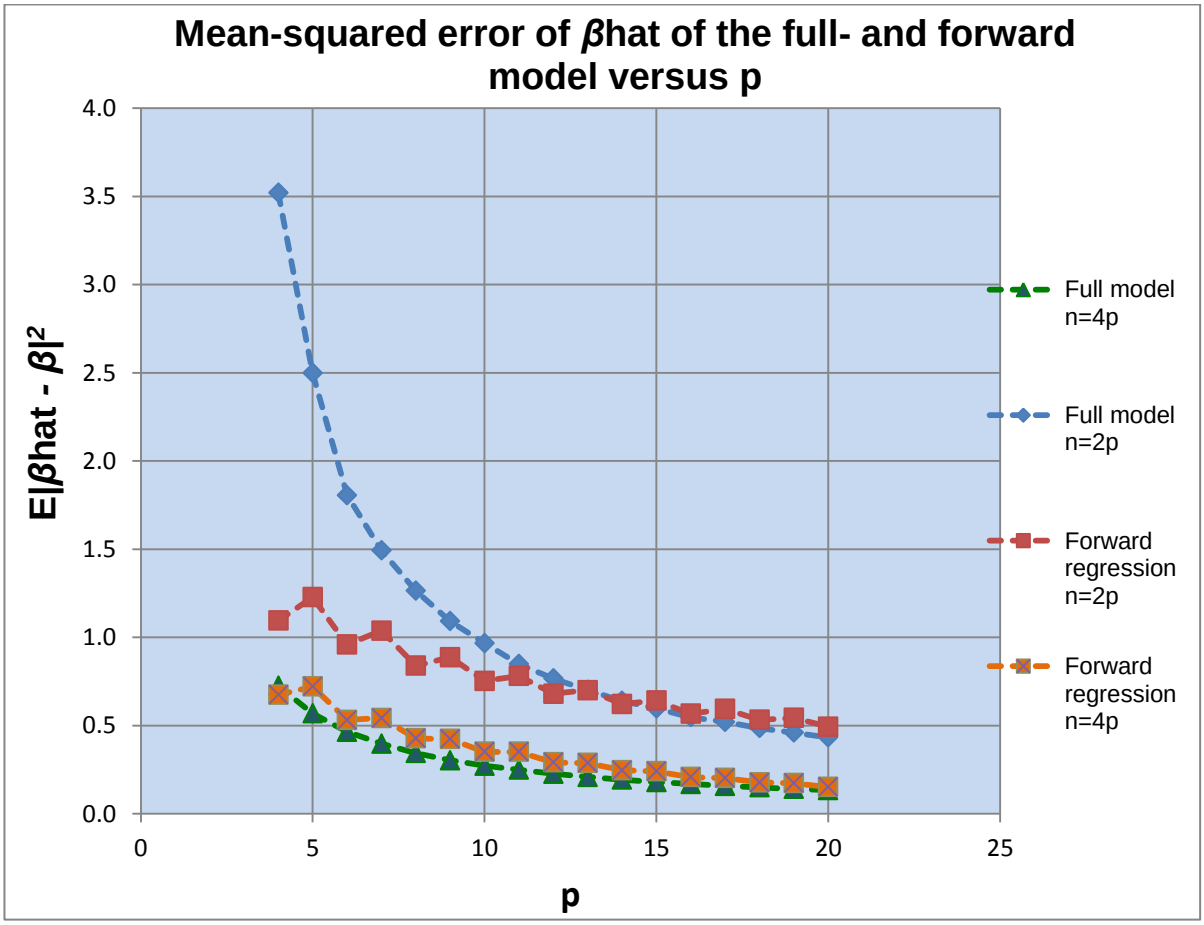
For the cases where n = 2p (comparing the red and blue graph) the forward model has a lower mean-squared error for $\widehat{\boldsymbol{\beta}}$ than the full model, (especially for low n values). The forward model is therefore more accurate than the full model in this case.

For the cases where n = 4p (comparing the orange and green graph) the two graphs do not differ much in mean-squared error values for $\widehat{\boldsymbol{\beta}}$ especially where n > 60.

The full regression model can however not be done for the cases n = 0.5p as (X'X) is singular and the inverse does not exist. For n = p an exact fit would be obtained. In those cases the forward model can only be used.

To obtain better accurate models the full- as well as the forward model is suggested for the case where n = 4p. For the case where n = 2p the forward model is preferred. (table 4)

Graph 9. The graph represents the differences between the forward- and full model of the mean-squared error of the estimated coefficient, $\widehat{\beta}$. The independent variable here is the sample size (p).



This graph shows the difference in mean-squared error of the full- versus the forward model. The independent variable here is the number of variables (p).

The graph shows similar patterns than graph 8. For the cases where n = 2p (comparing the red and blue graph) the forward model has a lower mean-squared error for $\widehat{\boldsymbol{\beta}}$ than the full model, (especially for low p values). The forward model is therefore more accurate than the full model in this case.

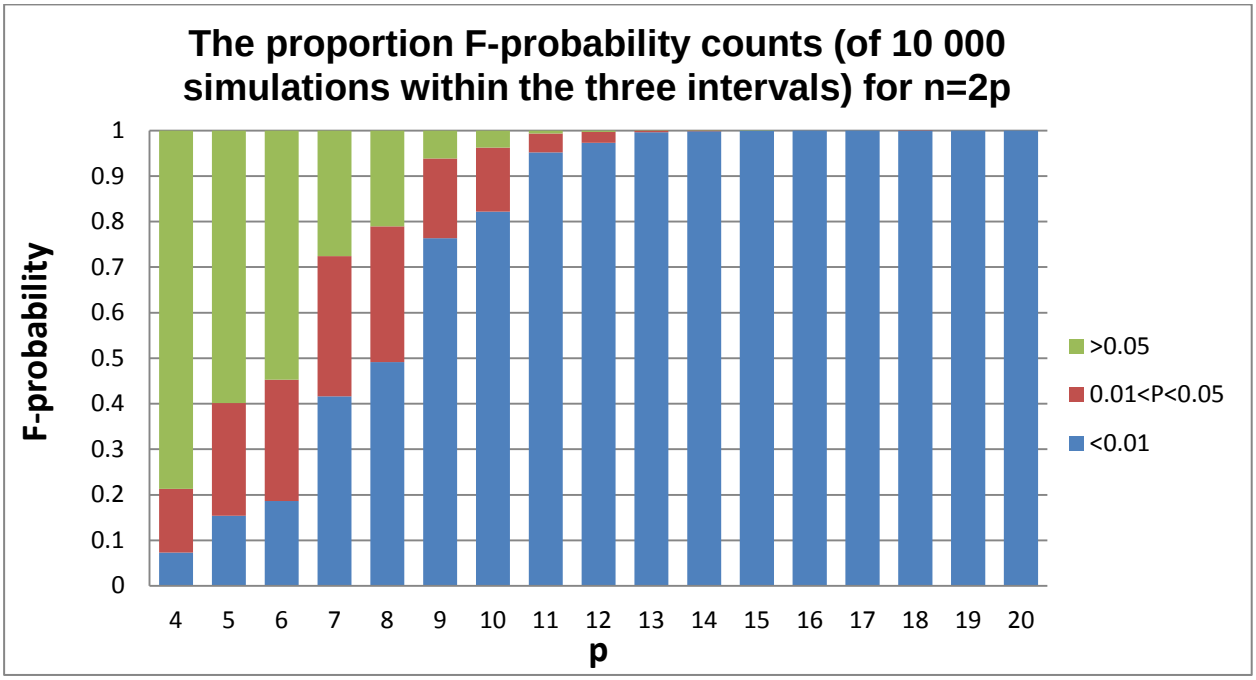
For the cases where n = 4p (comparing the orange and green graph) the two graphs do not differs much in mean-squared error values for $\widehat{\boldsymbol{\beta}}$. Both models is good here.

For the cases where n = 0.5p and n = p, the full model could not be determined, because (X'X) is singular and the inverse does not exist.

To obtain better accurate models the full- as well as the forward model is suggested for the case where n = 4p. In practise it often happened that the most important variables would be chosen, then the forward model is the preferable model, especially when n = 2p.

The following two graphs show the proportion of F-probability counts within the three classes P > 0.05, 0.01 < P < 0.05 and P < 0.01 over the 10 000 simulations of the full model for the cases where n = 2p and n = 4p (for each n and p case).

Graph 10. This graph represents the case where n = 2p .It is shown in green, red and blue respectively.

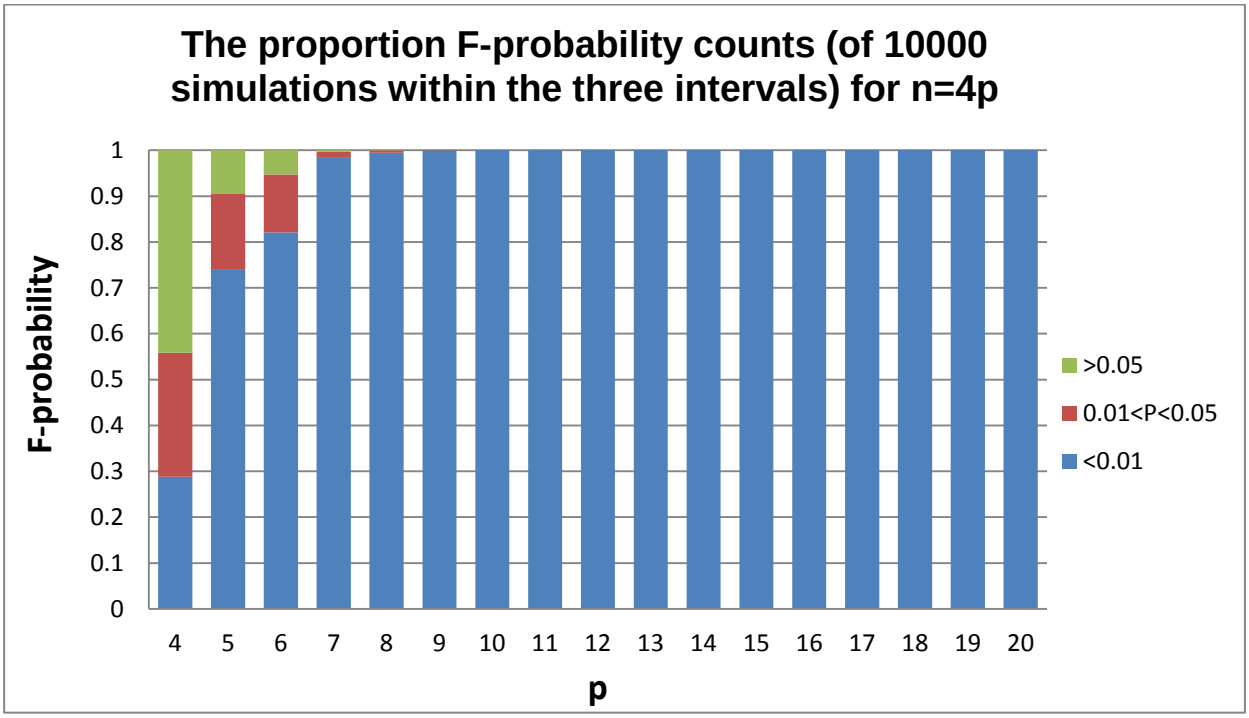


This refers to table 5.

The F-probability for the full model is determined for each simulation and each case (n and p). 10 000 simulations are done for this study. For each simulation the F-probability is grouped into one of the classes, p > 0.05, 0.01 < p < 0.05 and p < 0.01. The average F-probability counts over 10 000 are determined and shown in the graph as a proportion.

Out of this graph it is clear that the multiple regression analysis for the full model is more significant for the larger n (also p) values. It is advisable that n must be larger than 26 for the case where n=2p. The model is significant, as 0.01 level is always achieved.

Graph 11. This graph represents the case where n = 4p. It is shown in green, red and blue respectively.



This refers to table 5.

It is the average F-probability counts for the classes p> 0.05, 0.01<p<0.05 and p<0.01 for n=4p over 10 000 simulations The full regression model can however not be done for the cases n = 0.5p as (X'X) is singular and the inverse does not exist. For n = p an exact fit would be obtained.

The proportion F-probabilities for small n and p values were much higher than of the bigger n and p values in the class (>0.05). Thus, for bigger values of n and p numbers the full model were much better than the, of the small values. For a good (strictly significant) model the case must be higher than 10. Thus n > 28 or p > 7 for n = 4p. The model is then significant at 0.01 level.

Chapter 8

SUMMARY

In practise, when one has many candidate variables as explanatory variables in multiple regression, there is always the possibility that variables that are important determinants of the response variable might be omitted from the model while unimportant variables might be included. Both types of errors are important, and in this paper we attempt to quantify the probabilities of these errors.

The forward model has been addressed to determine the error rate.

The mean-squared error is intimately related to prediction accuracy. The mean-squared error for each n versus p case has an exponential downward trend as n increase (or p) for both the full- and forward models for each of the n versus p groups. The full regression model can however not be done for the cases n = 0.5p as (X'X) is singular and the inverse does not exist. For n = p an exact fit would be obtained.

The mean-squared error of the forward model for the estimated $\boldsymbol{\beta}$, ($\widehat{\boldsymbol{\beta}}$) form the true $\boldsymbol{\beta}$ is lower than that of the full model for the case where n = 2p and more or less the same than the case where n=4p. Looking at this, the forward regression model is a more accurate model than the full model in this case. In the case where n = 4p, both models are more or less the same as prediction accuracy.

A reduction of the full model in many cases is necessary, but reasons for these requirements are outside the scope of this dissertation. In many of the cases the forward selection is preferred over the full model.

The best forward selection analysis is found in the cases where n = 4p and n >36. The probability of a correct classification will then be higher than 60% .

For a multiple, linear forward regression analysis the lowest probability for the x_i's to be misclassified is within the classification of the n = 4p group, especially for the cases higher than 16 (n > 36). Each test was done nominally at $\alpha = 0.05$, but the actual significance level seems to be between 0.08 and 0.12 depending on n and p. This indicates that forward selection is a reliable method to use, but the significance level may have to be adjusted.

Forward regression in all the cases can be done but the accuracy of the model must be kept in mind.

If forward regression analysis is done in cases where n = 0.5p it would be advisable that n > 30. (Thus is not tested in this dissertation). Forward regression must be used very wisely when n=0.5p or other analytical methods like the Lasso shrinkage method must preferably be used.

To conclude, the analysis of an accurate forward regression analysis can be regarded as reliable, provided the sample size is big enough ($n \geq 4p$).

Chapter 9

CONCLUTIONS AND RECOMMENDATIONS ABOUT FORWARD REGRESSION - RECOMMENDATIONS FOR FURTHER STUDY

The analysis of an accurate forward regression analysis can be regarded as reliable, provided the sample size is big enough ($n \geq 4p$) and $p \geq 4$.

The best forward selection analysis is found in the cases where $n = 4p$ and $n > 36$. The probability of a correct classification will then be higher than 60% in study.

This dissertation is only a guideline for using forward selection, keeping the accuracy of the models in mind where the cases are $n = 0.5p$, p , $2p$ and $4p$ with an error$\sim N(0,4)$ and where x is normally distributed with

$$\boldsymbol{\mu_x}=\{\mathbf{0}\} \text{ and } \boldsymbol{\Sigma_X} = \begin{pmatrix} 1 & \cdots & 0.5 \\ \vdots & \ddots & \vdots \\ 0.5 & \cdots & 1 \end{pmatrix}$$

The dissertation could be built upon in a number of ways.

It is suggested that proper guidelines for statisticians and researchers must be published, given guidelines for using forward selection for all the normality conditions of x and for the different noise levels.

It is shown that the different subset selection methods do not differ too much.

In a case where forward selection is not the appropriate method (for a chosen n and p), better guidelines for using other analytical methods must be published.

To further clarify the properties of forward regression, the following additional simulations would be usefull:

- Other values of ρ, rather than 0.5
- Other proportions of variables with $\beta_j = 1$ and $\beta_j = 0$
- Cases where the irrelevant variables ($\beta_j = 0$) are uncorrelated with the relevant variables ($\beta_j = 1$)
- Situations with serious multicollinearity
- Still larger values of p (larger than 20)
- The non-zero regression coefficients β_j not all equal
- Stepwise regression may also be studied in a similar way.

Chapter 10

REFERENCES/ BIOGRAPHY

BAIN LEE & ENGELHARDT, 1987. *Introduction to probability and Mathematical Statistics. B*oston, U.S.A. PWS publishers. p. 305, p. 28, p. 179.

BARR A.J, GOODNOGHT J.H, SALL J.P & HELWIG J.T, *A User's Guide to SAS.* Raleigh, N.C: SAS Instituite.

BENDEL, R. B AND AFIFI A.A. 1977. Comparison of Stopping rules in Forward Stepwise Regression. *J. Ame. Statist. Assoc.* p 72. P.46-53

BERK K. N. 1978. Comparing Subset Procedures. *Technometrics, Vol. 20, p 1-6.*

DIXON W.J. (ed.) 1979. *BMDP: Biomedical Computer Programs.* Berkeley: University of Calafornia Press.

DIEHR G. & D. R HOFLIN. 1974. Approximating the distribution of the sample R^2 in the best subset regressiors. *Technometrics 16:* p. 317-320.

DRAPER NR & SMITH H. 1998. *Applied regression analysis* – Third edition . Canada. Wiley Series in probability Statistics. p. 178, 327 – 345.

HASTIE P, TIBSHIRANI S & FRIEDMAN H. 2009. *The elements of Statistical learning; Datamining, Inference and prediction*; Second edition. Stanford, California. Springer. p. 40 – 60.

HINES WILLIAM W & MONTGOMERY DOUGLAS C. 1980. *Probability and Statistics in Engineering and Management Science - Second Edition*. Canada. Wiley Series in probability Statistics. p.66 – 70, p. 360 - 454.

KLEINBAUM/ KUPPER. 1978. *Applied regression analysis and other multivariable methods* – California. Duxbury Press, A Division of Wadsworth Publishing Company, Inc. p. 228 – 234.

MENDENHALL WILLIAM, MCCLAVE JAMES, 1981. *A second course in Business Statistics: Regression Analysis.* California, Dellen Publishing Company.

MONTGOMERY DOUGLAS C, ELIZABETH A. PERCK. G. GEOFFREY VINING. 2006. *Introduction to linear regression analysis* - Fourth Edition. Canada. Wiley Series in probability Statistics. p. 66 – 70, p.80, 84 – 92, p. 96. p. 264.

ODELL P.L & FEIVESON A.H. 1966. A Numerical Procedure to Generate a Sample Covariance matrix . *J. Amer. Statist. Assoc. 61.* p. 199 – 203.

RADSHENCO PETER 7 JAMES GARETH M. 2011. Improved variable selection with Forward-Lasso adaptive shrinkage. *Annals of Applied Statistics 2011. Vol 5. No. 1*. p. 427 – p. 448.

SAS, Version 9, 2. SAS Institute Inc, SAS Campus Drive, Cary, North Carolina 27513.

http://sparselab.stanford.edu

SPSS – version 19

STODDEN VICTORIA. 2006. *Model Selection with the number of variables exceeds the numbers of observations.* (A dissertation submitted to the department of Statistics of Stanford University for the degree of doctor of Philosophy)

WASSERMAN WILLIAM, JOHN NETER AND G.A. WHITMORE. 1993. *Applied Statistics, 4th Edition*. *Allyn & Bacon.*

WHITNEY D.C, FORD M.G & LIVINGSTONE D.J. 2000 - Unsupervised Forward Selection: A method for eliminating Redundant variables. *J. CHEM. Inf. Comput. Sci, 2000 40(5)* p. 1160-1168.

http://en.wikipedia.org/wiki/Mean_square_error

http://en.wikipedia.org/wiki/Mean_square_error

WILKINSON LELAND AND DALLAL GERALD E. 1981. Test of Significance in Forward Selection Regression With an F-to-Enter Stopping Rule. *Technometrics,* Vol 23, No. 4 NOVEMBER 1981

YOUNGER. 1985. *A first course in linear regression – Second edition. PWS publishers, Duxbury Press* – Boston. p. 428.

Chapter 11

Appendix – Computer program.

The following SAS and SAS IML programme - used in this study.

```
/************      regressionFORWARD.sas     ****************************/
options ls=80 ps=64;

proc iml;

sim=10000;     (11)
dd=j(sim,3,0);
MSE_bhat=j(1,1,.);

*n=30  ;   /*********** This is the n and p for the example ************/
k=0;
*p=8;

/*=========================================================================*/
/*             DEFINING THE MODULES USED IN PROGRAM                        */
/*=========================================================================*/

/*=========================================================================*/
/*    Determine the variable with maximum partial correlation              */
/*=========================================================================*/

Start maxr(p,r,rmax,xxmax);    (33)
   do xxx = 2 to p+1 ;
      if r[1,xxx]=rmax then do;
        xxmax = xxx-1;
      end;
   end;
Finish maxr;


/*=========================================================================*/
/*       Testing the t- statistic for significance                         */
/*=========================================================================*/

Start tprobval(k,n,rmax,xxmax,t,tprob);    (34)
   *print k;
   t=rmax*(sqrt((n-2-k)/(1-rmax**2)));
   *print t;
   tprob =(1-probt(abs(t),n-2-k))*2;
 if tprob <= 0.05 then do;
     *print 'keep x' xxmax ' in the model';
     *print t;
     *print 'tprob=' tprob;
     k=k+1;
 end;
*else do;
*print t;
*print 'tprob =' tprob;
*print 'x' xxmax 'will not be in the model';
*end;
*print k;
Finish tprobval;
```

```
/*=============================================================================*/
/*                Determine the estimated B (BHAT)                             */
/*=============================================================================*/

Start bb2(xFF,n,p,xin2,xxxxin,y,xF,bhatFORW,yhatFORW);
   xFF=j(n,1,1);
   *PRINT xF;

  do g = 1 to p ;
  if xin2[,g] = j(n,1,1) then do;
    xFF = xFF || xxxxin[,g];
  END;
 end;

xF=xFF;
*print xF;  /** This x is the new forward regression analysis x ***/

 bhatFORW = inv(xF`*xF)*xF`*y ;
 yhatFORW = xF*bhatFORW ;
*print bhatFORW;

Finish bb2;

/*=============================================================================*/
/*      Defining the bhatFORWARD to determine the MSE for bhat                 */
/*=============================================================================*/

start bhat_mseFORW(k,xin,p,bhatFORW,bhatFORW2,bhatFORW2_);

jj=1;
do ii=1 to p ;
    if xin[ii,1]=j(1,1,0) then do;
      bhatFORW2=bhatFORW2//j(1,1,0);
    end;
   else do;
      bhatFORW2=bhatFORW2//bhatFORW[1+jj,1];
      jj=jj+1;
   end;
end;

bhatFORW2_= bhatFORW[1,1]//bhatFORW2 ;
free bhatFORW2;
*print bhatFORW2_;

finish;

/*=============================================================================*/
/* Determine the estimated B(BHAT)if no variables are included in model */
/*=============================================================================*/

start bb3(n,y,xFF,xF,bhatFORW,yhatFORW);

 xFF=j(n,1,1);
 xF=xFF;

 /*** This x is the new forward regression analysis x ***/
 bhatFORW = inv(xF`*xF)*xF`*y ;
 yhatFORW = xF*bhatFORW ;
*print yhatFORW, bhatFORW bForw bFF2;

Finish bb3;
```

```
/*=========================================================================*/
/*         Determine the variables in and out of the model                 */
/*=========================================================================*/

Start outnew2x(xin,xout,n,p,xout2,xxxxout,xstay,noutx,xFout);

xstay=loc(xin);
*print 'The following x_s stays in the model' xstay;
noutx=loc(xout);
*print noutx;

 do c = 1 to p ;
  if xout2[,c] = j(n,1,1) then do;
    xFout = xFout || xxxxout[,c];
  END;
 end;

 *print xFout;
 Finish outnew2x;

/*=========================================================================*/
/**      Determine the coefficient of partial correlation    (8&36)      **/
/*=========================================================================*/

Start ryx(y,xF,bhatFORW,p,h,xin,z,SSEin,xFF,BFHat,SSEr,SSErtot,r2yx,ryx);
  SSEin=(y-xF*bhatFORW)`*(y-xF*bhatFORW);
  *print SSEin;

  Do kk=1 to p;
  If xin[kk,1]^=1 then do;
  *print xF;
  xFF=xF;
  xFF=xFF||z[,kk];
  *print xFF;
  BFHat =inv(xFF`*xFF)*xFF`*y;
  SSEr=(y-xFF*BFHat)`*(y-xFF*BFHat);
  SSErtot=SSErtot||SSEr;
  free xFF;

  end;
  end;
  r2yx=j(1,h-1,1)-SSErtot/SSEin;    (8)
  ryx=sqrt(r2yx);
*print ryx , SSErtot;
Finish ryx;

/*=========================================================================*/
/*  Determine the max r or the minimum SSE of the remainder x e (8&36) **/
/*                  Coefficient of partial correlation                     */
/*=========================================================================*/

Start min_SSE(SSErtot,ryx,h,minSSE,noutx,maxryx,SSEyxmin,yxmax) ;
   minSSE = min(SSErtot[1,]);
  *print minSSE;
   maxryx = max(ryx);
  Do m=1 to h;

  if ryx[1,m]=maxryx then do;
    yxmax=noutx[,m];
    SSEyxmin=noutx[,m];
    *print 'The minimum SSE is:' SSEyxmin;
```

```
    *print 'The maximum r is:' yxmax;
    maxryx =ryx[1,m]   ;
    *print maxryx;
  end;

  end;

  Finish min_SSE;

/*=============================================================================*/
/*                              END OF MODULES                                 */
/*                          BEGINNING OF THE PROGRAM                           */
/*=============================================================================*/

/*=============================================================================*/
/*                              The p loop                                     */
/*=============================================================================*/
do p= 4 to 20;     (12)

MSE_bhatFULL=j(1,1,.);


/*=============================================================================*/
/*                              The n loop                                     */
/*=============================================================================*/

DO ggg = 1 TO 4;    /*while (n>k+2);   (13)
n=round(0.25*(2**ggg)*p,1);
*print n k p;

if n>5 then do;


/*=============================================================================*/
/*                              The sim loop    (11)                           */
/*=============================================================================*/
DO i = 1 to sim;


*if n> k+2 then do;
h=p;
rr=j(p+1,p+1,0);

*print " ========= simulation = " i   " p=" p " n=" n " ==================";

      xin=j(p,1,0);      (35)
      xbin=j(p,1,0);
      xout=j(p,1,1);
      xout2=j(n,p,1);
      xin2=j(n,p,0);
      mu1 = j(1,p,0);      (15)
      sigma1=diag(j(p,p,0.5))+j(p,p,0.5);     (16)
      *print sigma1;

   call vnormal(z,mu1,sigma1,n) ;     (14)
   x= j(n,1,1) || z;
   *print x,z;
```

```
/*=========================================================================*/
/*    Generating the e values from a normal distribution with              */
/*          with mean 0 and variance sigma_e                              **/
/*                    Assigning the b values    (19)                      */
/*=========================================================================*/


   b=j(p+1,1,0);
   pb=p/2;
   kb1=(p+1)/2;
   kb2=(p-1)/2;

   if p=4 | p=6 | p=8 | p=10 | p=12 | p=14 | p=16 | p=18 | p=20 then do;
   b=j(1,1,0)//j(pb,1,1)//j(pb,1,0);
   end;

   if p= 5 | p=7 | p=9 | p=11 | p=13 | p=15 | p=17 | p=19 then do;
   b=j(1,1,0)//j(kb1,1,1)//j(kb2,1,0);
   end;

   *print b;
   e=j(n,1,0);
   mu_e={0};    (18)
   sigma_e={2};    (18)
   e=normal(e)#j(n,1,sigma_e)+j(n,1,mu_e);   (17)
   y=x*b + e;
   *print y;

   xy = y||x;
   xx= x[,2:p+1];
   yxx= y || x[,2:p+1];

   ybar=j(n,1,1)`*y/n;
   xbar=j(n,1,1)`*xx/n;
   yxbar=j(n,1,1)`*yxx/n;

  /*=========================================================================*/
  /*                 a.  REGRESSION FOR THE FULL MODEL     (20)              */
  /*=========================================================================*/

   If n > (p+1) then do;

   bhat = inv(x`*x)*x`*y ;    (21)
   yhat = x*bhat ;
   SSreg=(bhat)`*(x`*y)-n*ybar*ybar;   (25)
   SSE=(y-x*bhat)`*(y-x*bhat);         (26)
   SST=y`*y-n*ybar*ybar;               (24)
   MSreg=SSreg/p;                      (27)
   MSE=((y-x*bhat)`*(y-x*bhat))/(n-p-1);   (28)
   F=MSreg/MSE;        (23)                /*  Determine the F-probability */
   pp=1-probf(F,p,n-p);                    /*  Determine the p- value  */
   *print xy;
   R2= SSreg/SST;
                    /* Determine the R2 VALUE    */
   *print bhat, SSreg,SSE, MSreg, MSE,SST, F;
   *print "The F-probability is :" pp;
   *Print "The R - square value is: " R2;
```

```
/*=========================================================================*/
/*Determine the class for the F-prob for each of the n, p and sim cases */
/*=========================================================================*/
(30)

  if pp<= 0.05 then do;
  if pp>=0.01 then do;
     dd[i,1:3]= {0 1 0} ;
     *print "Fprob = 0.05";
  end;
  end;

  If pp < 0.01 then do;
    *Print "Fprob <0.01";
    dd[i,1:3]={1 0 0} ;
  END;

  if pp > 0.05 then do ;
     dd[i,1:3]= {0 0 1} ;
     *print "Fprob > 0.05";
  END;

  /*  Determine the standard error of the B values  */

  stderr=j(p,p,99);  (31)
  stderrB=j(p,1,99);
  invxx=inv(x`*x);
  covb=abs(invxx*MSE);
  mseb=vecdiag(covb);
  mmm=mseb[+,];
  *PRINT covb mseb mmm;
  stderrBB=SQRT(covb);
  stderrB=vecdiag(stderrBB);

  nm3={'b0' 'b1' 'b2' 'b3' 'b4..'};
  nm2={'bhat..'};

  /*  Determinine the t-probabilities of the B values   */

  TB= J(p+1,1,0);
  probT = J(p+1,1,0);

  DO j= 1 TO p+1 ;
     TB[j,1]=abs(bhat[j,1]/stderrB[j,1]);
     probT[j,1]=(1-probt(TB[j,1],n-p-1))*2;
  end;

   *print bhat [rowname=nm3] stderrB probT;

/*=========================================================================*/
/*    Determine the MSE FOR BHAT for the full model for each               */
/*                    of the n, p and sim cases          (40)              */
/*=========================================================================*/

    MSE_bhatFULL=((b-bhat)`*(b-bhat))/p;
      *print MSE_bhatFULL;
    MsbtotFULL= MsbtotFULL //MSE_bhatFULL;
      *print MsbtotFULL;

end;
```

```
else do;
 MsbtotFULL=j(sim,1,.);
 dd[i,1:3]={. . .} ;
end;

*print '    /*** End of regression of full model **/';

/*=========================================================================*/
/*                a.  REGRESSION FOR THE FORWARD MODEL      (20)          */
/*=========================================================================*/


/**a.  Get correlations to determine linear relationships between      */
/*                          all the variables;                         */

    sigma = (yxx-j(n,1,1)*yxbar)`*(yxx-j(n,1,1)*yxbar)/(n-1);
    *print yxbar, sigma;
    stdev_xy = sqrt(vecdiag(sigma));
    *print stdev_xy;
    d = diag(stdev_xy);
    *print d;
    r = inv(d)*sigma*inv(d);
    coryx=r[1,2:p+1];
    rmax =max(coryx);

    nm1={'y' 'x1' 'x2' 'x3' 'x4..'};
    *print 'The correlation matrix is:';
    *print r[colname=nm1] [rowname=nm1];


/** b. Determine which x has maximum correlation with the y-value    ***/

    call maxr(p,r,rmax,xxmax);    (33)
    *print xxmax, rmax;

/*  c.       Test for correlation significance                        **/

    call tprobval(k,n,rmax,xxmax,t,tprob);    (34)

/*d. Choose the best partial correlated variable if significant in b.  */

If tprob < 0.05  then do;    (35)
      xin[xxmax,1]=1 ;
      xout[xxmax,1]=0;
      xin2[1:n,xxmax]=j(n,1,1);
      xout2[1:n,xxmax]=j(n,1,0);
      xxxxout=x[,2:p+1]#xout2;
      xxxxin=x[,2:p+1]#xin2;
    *print xxxxout xxxxin;
    *print xin xout ;

free bForw bF;

call bb2(xFF,n,p,xin2,xxxxin,y,xF,bhatFORW,yhatFORW);

*print yhatFORW, bForw xF;

/* e. Determine the partial correlation matrix   */

call outnew2x(xin,xout,n,p,xout2,xxxxout,xstay,noutx,xFout);
```

```
call ryx(y,xF,bhatFORW,p,h,xin,z,SSEin,xFF,BFHat,SSEr,SSErtot,r2yx,ryx);
      (36)
h=p-1;          /* the variables became 1 less  */

call  min_SSE(SSErtot,ryx,h,minSSE,noutx,maxryx,SSEyxmin,yxmax) ;

/*  f. Test for significance between the residuals and the x variables */

call tprobval(k,n,maxryx,yxmax,t,tprob);

end;

else do;

 finalxx=xin`;
 call bb3(n,y,xFF,xF,bhatFORW,yhatFORW);

end;

free xFout;

/*=============================================================================*/

 /*** g. Repeating the selecting of variables  ****/

do kk=1 to p-2 while((tprob < 0.05) & (n > k+2)) ;

      *print kk k;
      xin[yxmax,1]=1 ;
      xout[yxmax,1]=0;
      xin2[1:n,yxmax]=j(n,1,1);
      xout2[1:n,yxmax]=j(n,1,0);
      xxxxout=x[,2:p+1]#xout2;
      xxxxin=x[,2:p+1]#xin2;
     *print xxxxout xxxxin;

*print xin xout xin2 xout2 ;

call bb2(xFF,n,p,xin2,xxxxin,y,xF,bhatFORW,yhatFORW);

call outnew2x(xin,xout,n,p,xout2,xxxxout,xstay,noutx,xFout);

noutx=loc(xout);

*print 'The following variables are not in the model' noutx;

free SSErtot;

call ryx(y,xF,bhatFORW,p,h,xin,z,SSEin,xFF,BFHat,SSEr,SSErtot,r2yx,ryx);

h=h-1;
*print h ;

/* the variables became 1 less  */

  if h>0 then do ;
      call min_SSE(SSErtot,ryx,h,minSSE,noutx,maxryx,SSEyxmin,yxmax) ;
  end;
```

```
/*  Test for significance between the residuals and the x variables */

call tprobval(k,n,maxryx,yxmax,t,tprob);

free xbin bFstay xFout;

END;

/*===========  Final part  ==========================================*/

If tprob < 0.05 then do;

     *print yxmax;

      xin[yxmax,1]=1 ;
      xout[yxmax,1]=0;
      xin2[1:n,yxmax]=j(n,1,1);
      xout2[1:n,yxmax]=j(n,1,0);
      xxxxout=x[,2:p+1]#xout2;
      xxxxin=x[,2:p+1]#xin2;

     *print xxxxout xxxxin;
     *print xin xout ;

      free bForw;
      call bb2(xFF,n,p,xin2,xxxxin,y,xF,bhatFORW,yhatFORW);

      *print yhatFORW bForw  bhatFORW, xF;
      finalxx=loc(xin);

end;

else do;
 finalxx=xin`;
end;

/*=========================================================================*/
/* Determine the mean squared error of bhat for the separate simulated    */
/*              values of the forward regression model                     */
/*=========================================================================*/


call bhat_mseFORW(k,xin,p,bhatFORW,bhatFORW2,bhatFORW2_);

    MSE_bhatFORW=(b-bhatFORW2_)`*(b-bhatFORW2_)/p;
    MSE_bhattFORW =  MSE_bhattFORW//MSE_bhatFORW;

free xxxx xxxxin;
free h SSErtot SSEin ;

*print 'The following x e will be in the final model' finalxx;

xinlast=xinlast//xin`;

*print xinlast;

k=0;
END;      /*==================== sim do loop ============================**/
```

```
*print "=== Summary for n=" n " and p = " p "  for " sim " simulations ===";

finalxin=j(20,1,.);

*print xinlast;

finalxin_prob=xinlast[+,]/sim;
finalxin=xinlast[+,];
finalamountx=xinlast[,+];
avsfinalamountx=round(finalamountx[+,]/sim,(0.1));

*print "The total of the x_s to be in the model is:" finalxin;
*print 'The probabilities of each of the x_s to be in the forward model is:'
finalxin_prob;
*print "The total amount of x e to be in the models" finalamountx,n,p '
=avs_k';
*print "The average amount of xe over all the simulations in the models is"
avsfinalamountx;
*free bForw bF  ;

countmat=j(sim,p+1,0);

Do sss = 1 to sim;
Do iii = 0 to p;
If finalamountx[sss,1] = iii then do;
  countmat[sss,iii+1] = {1};
  *print countmat;
end;
end;
end;
*free countmat;

/**** Determine the amount of xs to be in the model over all simulations **/

if p<20 then do;
countxe = (j(1,1,p)||j(1,1,avsfinalamountx)||j(1,1,n))||countmat[+,]||j(1,20-
p,.);
npcountxe=npcountxe//countxe;
end;

else do;
countxe = (j(1,1,p)||j(1,1,avsfinalamountx)||j(1,1,n))||countmat[+,];
npcountxe=npcountxe//countxe;
end;

countname={'p','ave_k','n','0xs','1xs','2xs','3xs','4xs','5xs','6xs','7xs','8
xs','9xs','10xs','11xse',
'12xs','13xs','14xs','15xs','16xs','17xs','18xs','19xs','20xs'};

*print countxe[colname=countname];


/******Determine the amount of xi_s to be in all simulations **********/

if p<20 then do;
x_is_np = (j(1,1,p)||j(1,1,avsfinalamountx)||j(1,1,n))||finalxin||j(1,20-
p,.);
amountx_is=amountx_is//x_is_np;
end;

else do;
```

```
x_is_np = (j(1,1,p)||j(1,1,avsfinalamountx)||j(1,1,n))||finalxin;
amountx_is=amountx_is//x_is_np;
end;

countnp={'p','ave_k','n','x1','x2','x3','x4','x5','x6','x7','x8','x9','x10','
x11','x12','x13','x14','x15','x16','x17','x18','x19','x20'};
*print amountx_is[colname=countnp];

/*  The mean-square error for the FORWARD REGRESSION MODEL is:   */

*Print MSE_bhattFORW ;
*Print MsbtotFULL;

MSE_bhatTOTFORW =j(1,1,p)||j(1,1,avsfinalamountx)||j(1,1,n)||
(MSE_bhattFORW[+,]/sim);

*print 'MSE FOR THE Forward MODEL of the estimated coefficient BHAT at each
step from the true B is' MSE_bhatTOTFORW;

/*  The mean-square error for the FULL REGRESSION MODEL is:   */

MSE_bhatTOTFULL= (MsbtotFULL[+,]/sim);

*print 'MSE FOR THE FULL MODEL of the estimated coefficient BHAT at each step
from the true B is p k n' MSE_bhatTOTFULL;

no_prob_prob= dd[+,]/sim;

*print dd;

ddsum = j(1,1,p)||j(1,1,n)||no_prob_prob;

*print "The total number of F probabilities in the full model in the ";
*print "different probability categories ";
*print "<0.01'     '0.01<=p<=0.05'     '>0.05, will be";
*print ddsum ;

Fprob_=Fprob_//ddsum;
*Fprob_prob=Fprob_/sim;

free xinlast MSE_bhattFORW MsbtotFULL xcount ddsum countxe finalamountx
x_is_np;

np_MSE_bhatTOTFULL = np_MSE_bhatTOTFULL//MSE_bhatTOTFULL;
np_MSE_bhatTOTFORW = np_MSE_bhatTOTFORW//MSE_bhatTOTFORW;
mse_bhat= np_MSE_bhatTOTFORW||np_MSE_bhatTOTFULL ;

/*
DO qq = 1 to 59;
if mse_bhat[qq,5] = {0} then do;
   mse_bhat[qq,5] = {.};
end;
end;*/

free MSE_bhatTOTFULL MSE_bhatTOTFORW ;

END;          /*****************************************************/
```

```
end;       /*================= The n do loop ============================*/

END;     /**================   The p do loop  ==========================**/


Print '********    Summary OF simulated data  ***************************';

print 'The total amount of xs to be in the model with' sim 'simulations with
p average k and n is:' ;

print npcountxe[colname=countname];

print 'The total amount of each x_i to be in the model with' sim 'simulations
with p average k and n is:' ;

print amountx_is[colname=countnp];

DO kl = 1 to 59;
if mse_bhat[kl,5] = {0} then do;
   mse_bhat[kl,5] = {.};
end;
end;

mse_bhat2=j(59,6,0);

DO kl = 1 to 59;

if mse_bhat[kl,3] = round(0.5*mse_bhat[kl,1],1) then do;    /***** Labels for
n = 0.5p, p, 2p and 4p ***/
   mse_bhat2[kl,1:6] = mse_bhat[kl,1:5]||j(1,1,0.5);
end;

if mse_bhat[kl,3] = mse_bhat[kl,1] then do;
    mse_bhat2[kl,1:6] = mse_bhat[kl,1:5]||j(1,1,1);
end;

if mse_bhat[kl,3] = 2*mse_bhat[kl,1] then do;
   mse_bhat2[kl,1:6] = mse_bhat[kl,1:5]||j(1,1,2);
end;

if mse_bhat[kl,3] = (4*mse_bhat[kl,1]) then do;
  mse_bhat2[kl,1:6] = mse_bhat[kl,1:5]||j(1,1,4);
end;

end;


print 'Table of p, average k, n and the meansquare of bhat for the forward-
and full model for' sim ' simulations';

print 'p  avs_k  n   MSEBHATFORWARD  MSEBHATFULL n=__p ';

print mse_bhat2;

Print 'The F probabilities of the full model with ' sim 'simulations' ;
print 'with n and p respectively';

print '<0.01'     '0.01<=p<=0.05'    '>0.05  will be'  Fprob_ ;
```

```
cn={'p','avs_k', 'n','MSE_bhatFORW','MSE_bhatFULL','np'};
create msefull from mse_bhat2[colname=cn];
append from mse_bhat2;

run;

 goptions reset=all;
 symbol1 value=dot
        height=2  width=2;

 symbol2 value=star
        height=3  width=2;

 symbol3 value=triangle
        height=3   width=2;
Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the full model";

proc gplot data=msefull ;
 plot MSE_bhatFULL*n=np /vaxis=axis1; ;
run ;

 goptions reset=all;
 symbol1 value=dot
        height=2  width=2;

 symbol2 value=star
        height=3  width=2;

 symbol3 value=triangle
        height=3   width=2;
Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the forward model";

proc gplot data=msefull ;
 plot MSE_bhatFORW*n=np /vaxis=axis1;;
run ;

goptions reset=all;
symbol1 value=dot
        height=2  width=2;

symbol2 value=star
        height=3  width=2;

symbol3 value=triangle
        height=3   width=2;
Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the full model";

proc gplot data=msefull ;
 plot MSE_bhatFULL*p=np /vaxis=axis1;;
run ;
```

```
 goptions reset=all;
 symbol1 value=dot
        height=2  width=2;

 symbol2 value=star
        height=3  width=2;

 symbol3 value=triangle
        height=3   width=2;
Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the forward model";

proc gplot data=msefull ;
 plot MSE_bhatFORW*p=np /vaxis=axis1;;
run ;

goptions reset=all;
 symbol1 value=dot
        height=2  width=2;

 symbol2 value=star
        height=3  width=2;

 symbol3 value=triangle
        height=3   width=2;

Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the full & forward model";

proc gplot data=msefull ;
 plot MSE_bhatFULL*n MSE_bhatFORW*n /overlay vaxis=axis1;
run ;

goptions reset=all;
 symbol1 value=dot
        height=2  width=2;

 symbol2 value=star
        height=3  width=2;

 symbol3 value=triangle
        height=3   width=2;

Axis1 Label=(A=90 c=black  'MSE_bhat') ;
Title1 "The means square error for b hat for the full & forward model";

proc gplot data=msefull ;
 plot MSE_bhatFULL*p MSE_bhatFORW*p /overlay vaxis=axis1;
run ;
```

```
/*=========================================================================*/
/*  TESTING FORWARD REGRESSION IN SAS IML WITH PROC REG IN SAS*/
/*=========================================================================*/

data msefull;
set msefull;
 PROC PRINTTO FILE='e:\msc.prn' ;
 run;  */

/*
colname ={'y' 'x0' 'x1' 'x2' 'x3' 'x4' 'x5' 'x6' 'x7' 'x8' };
create data1 from xy[colname=colname];
append from xy;

colname ={'y' 'x0' 'x1' 'x2' 'x3' 'x4' 'x5' 'x6' 'x7' 'x8' };
create data2 from yxx[colname=colname];
append from yxx;


proc print data=data1;

PROC CORR DATA=data2;

PROC REG data=data1;
  MODEL y
         = x1 x2 x3 x4 x5;
            output out=Data p=yhat  ;

PROC REG data=data1;
  MODEL y
         = x1 x2 x3 x4 x5
             / SELECTION =FORWARD SLENTRY=0.05  ;

proc print data=regression;   **/

 quit;

=========================================================================
```