UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Maximum likelihood estimation procedures for categorical data

by

René Ehlers

Submitted in partial fulfilment
of the requirements for the degree
Magister Scientiae (Mathematical Statistics)

in the
Faculty of Natural and Agricultural Sciences
University of Pretoria

July 2002

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Professor N.A.S. Crowther, for his guidance and motivation. I also wish to thank my parents for their continuous support and for making this study possible.

ABSTRACT

Maximum likelihood estimation procedures for categorical data

by

René Ehlers

Supervisor: Professor N.A.S. Crowther
Department of Statistics
University of Pretoria

There are a large number of maximum likelihood estimation procedures for categorical data available for scientific application. In this dissertation the most commonly used methods, namely the Newton-Raphson, Fisher scoring and EM algorithms are compared with a maximum likelihood estimation procedure under constraints. An exposition of the theory and application of the methods are given.

Chapter 1 gives a brief overview of the exponential family, the generalized linear model and measures of goodness of fit.

In Chapter 2 the theory of the Newton-Raphson, Fisher scoring and EM algorithms and the method of maximum likelihood estimation under constraints is discussed.
The Newton-Raphson algorithm is an iterative procedure which is employed for solving non-linear equations. It makes use of the vector of first order partial derivatives and matrix of second order partial derivatives of the function to be maximized. The Fisher scoring algorithm is similar to the Newton-Raphson algorithm, the distinction being that Fisher scoring uses the expected value of the matrix of second order partial derivatives with respect to the parameters in the model.
In the broad class of models referred to as generalized linear models the observations come from an exponential family and a function of their expectation is written as a linear model using a link function. Agresti (1990) shows that when a canonical link function is used the Newton-Raphson and Fisher scoring algorithms are identical.
The EM algorithm is a very general iterative algorithm for ML estimation in incomplete data problems and is described in detail by Dempster, Laird and Rubin (1977). The algorithm makes use of the interdependence between the missing data and the parameters to be estimated. The missing data are filled in based on an initial estimate of the parameters (the E-step). The parameters are then re-estimated based on the observed data and the filled in data (the M-step). The process iterates between the two steps until the estimates converge.
Matthews (1995) presents a maximum likelihood estimation procedure for the mean of the exponential family subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, where $\mathbf{g}$ is a vector valued function of $\boldsymbol{\mu}$. If $\mathbf{Y}$ is a random vector with probability function belonging to the exponential family with $E(\mathbf{Y}) = \boldsymbol{\mu}$, then the ML estimate of $\boldsymbol{\mu}$ subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, is given by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - (\mathbf{G}_\mu \mathbf{V})' \left(\mathbf{G}_\mathbf{y} \mathbf{V} \mathbf{G}_\mu'\right)^{-1} g(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|)$$

where $\mathbf{g}(\boldsymbol{\mu})$ is a continuous vector valued function of $\boldsymbol{\mu}$ for which the first order partial derivatives exist, $\mathbf{G}_\mu = \dfrac{\partial \mathbf{g}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}$, $\mathbf{G}_\mathbf{y} = \dfrac{\partial \mathbf{g}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}|_{\mu=\mathbf{y}}$ and $\mathbf{V}$ is the covariance matrix which could be known or could be some function of $\boldsymbol{\mu}$, say $\mathbf{V}_\mu$. This result implies that the ML estimate must be obtained iteratively.
Comparative examples of all the above procedures are given in Chapter 2.

In Chapter 3 ML estimation of parameters for loglinear and logistic regression models is discussed. The results obtained by using the method under constraints are the same as those obtained by using the Newton-Raphson algorithm.

In Chapter 4 different patterns of symmetry in squared contingency tables are discussed and illustrated with an example from Agresti (1990). Results obtained are the same as the special cases considered in literature.

In Chapter 5 the method of ML estimation under constraints is used to determine ML estimates of cell probabilities in an incomplete contingency table for any loglinear model. It is assumed that the data are missing at random (MAR) and that the missing data mechanism is ignorable. It is shown that results are asymptotically the same as those obtained with the EM algorithm, the advantage being that the method under constraints is computationally less intensive.

CONTENTS

1 INTRODUCTION

There are a large number of maximum likelihood estimation procedures for categorical data available for scientific application. In this dissertation the most commonly used methods are compared with a maximum likelihood estimation procedure under constraints and an exposition of the theory and application of the methods are given.

The more generally used methods of maximum likelihood estimation for categorical data includes the Newton-Raphson and Fisher scoring algorithms for complete data and the EM algorithm for incomplete data. The Newton-Raphson algorithm is an iterative procedure which is employed for solving non-linear equations. It makes use of the vector of first order partial derivatives and matrix of second order partial derivatives of the function to be maximized. The Fisher scoring algorithm is similar to the Newton-Raphson algorithm, the distinction being that Fisher scoring uses the expected value of the second derivative with respect to the parameters in the model.
In the broad class of models referred to as generalized linear models the observations come from an exponential family and a function of their expectation is written as a linear model using a link function. Agresti (1990) shows that when a canonical link function is used the Newton-Raphson and Fisher scoring algorithms are identical.

The EM algorithm can be used for maximum likelihood estimation in incomplete contingency tables. The algorithm makes use of the interdependence between the missing data and the parameters to be estimated. The missing data are filled in based on an initial estimate of the parameters (the E-step). The parameters are then re-estimated based on the observed data and the filled in data (the M-step). The process iterates between the two steps until the estimates converge. The EM algorithm is specifically applied to the exponential family to determine ML estimates in incomplete contingency tables when the missing data mechanism is ignorable. Little and Rubin (1987) describes and uses the EM algorithm to determine the ML estimates of cell probabilities for loglinear models.

Matthews (1995) presents a maximum likelihood estimation procedure for the mean of the exponential family subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, where $\mathbf{g}$ is a vector valued function of $\boldsymbol{\mu}$.
For the loglinear model and logistic regression the results obtained from this method are the same as those obtained from the Newton-Raphson algorithm.
The analysis of patterns of symmetry in squared contingency tables are considered by using ML estimation under contraints and a program is given which can be used for any squared contingency table. Results obtained are the same as the special cases considered in literature.
The method is also further developed to determine maximum likelihood estimates for loglinear models when the contingency table is incomplete and the missing data mechanism is ignorable. This also illustrates the elegance with which the method of ML estimation under contraints can be applied.
The method under constraints is conceptually comprehensive, logically clear and at the same time computationally less intensive than the EM and other algorithms.

1 INTRODUCTION

There are a large number of maximum likelihood estimation procedures for categorical data available for scientific application. In this dissertation the most commonly used methods are compared with a maximum likelihood estimation procedure under constraints and an exposition of the theory and application of the methods are given.

The more generally used methods of maximum likelihood estimation for categorical data includes the Newton-Raphson and Fisher scoring algorithms for complete data and the EM algorithm for incomplete data. The Newton-Raphson algorithm is an iterative procedure which is employed for solving non-linear equations. It makes use of the vector of first order partial derivatives and matrix of second order partial derivatives of the function to be maximized. The Fisher scoring algorithm is similar to the Newton-Raphson algorithm, the distinction being that Fisher scoring uses the expected value of the second derivative with respect to the parameters in the model.
In the broad class of models referred to as generalized linear models the observations come from an exponential family and a function of their expectation is written as a linear model using a link function. Agresti (1990) shows that when a canonical link function is used the Newton-Raphson and Fisher scoring algorithms are identical.

The EM algorithm can be used for maximum likelihood estimation in incomplete contingency tables. The algorithm makes use of the interdependence between the missing data and the parameters to be estimated. The missing data are filled in based on an initial estimate of the parameters (the E-step). The parameters are then re-estimated based on the observed data and the filled in data (the M-step). The process iterates between the two steps until the estimates converge. The EM algorithm is specifically applied to the exponential family to determine ML estimates in incomplete contingency tables when the missing data mechanism is ignorable. Little and Rubin (1987) describes and uses the EM algorithm to determine the ML estimates of cell probabilities for loglinear models.

Matthews (1995) presents a maximum likelihood estimation procedure for the mean of the exponential family subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, where $\mathbf{g}$ is a vector valued function of $\boldsymbol{\mu}$.
For the loglinear model and logistic regression the results obtained from this method are the same as those obtained from the Newton-Raphson algorithm.
The analysis of patterns of symmetry in squared contingency tables are considered by using ML estimation under contraints and a program is given which can be used for any squared contingency table. Results obtained are the same as the special cases considered in literature.
The method is also further developed to determine maximum likelihood estimates for loglinear models when the contingency table is incomplete and the missing data mechanism is ignorable. This also illustrates the elegance with which the method of ML estimation under contraints can be applied.
The method under constraints is conceptually comprehensive, logically clear and at the same time computationally less intensive than the EM and other algorithms.

1.1 THE EXPONENTIAL FAMILY

Let $\mathbf{Y}$ be a $p \times 1$ random vector and $\boldsymbol{\theta}$ a $p \times 1$ vector of parameters. Barndorff-Nielsen (1978) defines the exponential family by

$$p(\mathbf{y}, \boldsymbol{\theta}) = b(\mathbf{y}) \exp\left[\mathbf{y}'\boldsymbol{\theta} - \kappa(\boldsymbol{\theta})\right], \quad \mathbf{y} \in \Re^p, \quad \boldsymbol{\theta} \in \aleph \tag{1}$$

where $\kappa(\boldsymbol{\theta})$ is referred to as the cumulant generating function and $\aleph$ is the natural parameter space for the canonical parameter $\boldsymbol{\theta}$.

The moment generating function of the exponential family is given by

$$\begin{aligned}
\mathbf{M}_{\mathbf{Y}}(\mathbf{t}) &= E\left[e^{\mathbf{t}'\mathbf{Y}}\right] \\
&= \int \cdots \int b(\mathbf{y}) \exp\left[\mathbf{y}'(\boldsymbol{\theta}+\mathbf{t}) - \kappa(\boldsymbol{\theta})\right] d\mathbf{y} \\
&= \exp\left[-\kappa(\boldsymbol{\theta})\right] \int \cdots \int b(\mathbf{y}) \exp\left[\mathbf{y}'(\boldsymbol{\theta}+\mathbf{t})\right] d\mathbf{y} \\
&= \exp\left[-\kappa(\boldsymbol{\theta})\right] \exp\left[\kappa(\boldsymbol{\theta}+\mathbf{t})\right] \int \cdots \int b(\mathbf{y}) \exp\left[\mathbf{y}'(\boldsymbol{\theta}+\mathbf{t}) - \kappa(\boldsymbol{\theta}+\mathbf{t})\right] d\mathbf{y} \\
&= \exp\left[-\kappa(\boldsymbol{\theta})\right] \exp\left[\kappa(\boldsymbol{\theta}+\mathbf{t})\right].
\end{aligned}$$

From this the cumulant generating function can be derived.

$$\begin{aligned}
\log \mathbf{M}_{\mathbf{Y}}(\mathbf{t}) &= \kappa(\boldsymbol{\theta}+\mathbf{t}) - \kappa(\boldsymbol{\theta}) \\
&= \kappa(\boldsymbol{\theta}) + \left[\frac{\partial}{\partial \boldsymbol{\theta}}\kappa(\boldsymbol{\theta})\right]' \mathbf{t} + \tfrac{1}{2}\mathbf{t}' \left[\frac{\partial}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}'}\kappa(\boldsymbol{\theta})\right] \mathbf{t} + r(\mathbf{t}) - \kappa(\boldsymbol{\theta}) \\
&= \left[\frac{\partial}{\partial \boldsymbol{\theta}}\kappa(\boldsymbol{\theta})\right]' \mathbf{t} + \tfrac{1}{2}\mathbf{t}' \left[\frac{\partial}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}'}\kappa(\boldsymbol{\theta})\right] \mathbf{t} + r(\mathbf{t}).
\end{aligned}$$

The mean vector and covariance matrix of $\mathbf{Y}$ are given by

$$E(\mathbf{Y}) = \frac{\partial}{\partial \boldsymbol{\theta}}\kappa(\boldsymbol{\theta}) = \boldsymbol{\mu} \quad \text{and} \quad \operatorname{Cov}(\mathbf{Y}) = \frac{\partial^2}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}'}\kappa(\boldsymbol{\theta}) = \mathbf{V}.$$

EXAMPLE 1.1
The Poisson distribution as a member of the exponential family.

Let Y_i, $i = 1, 2, \ldots, p$ be independent Poisson random variables with $E(Y_i) = \mu_i$. The joint probability function of $\mathbf{Y}' = (Y_1, Y_2, \ldots, Y_p)$ is

$$f_{\mathbf{Y}}(\mathbf{y}|\boldsymbol{\mu}) = \frac{\exp\left(-\sum \mu_i\right) \prod \mu_i^{y_i}}{\prod y_i!} = \exp\left[\sum y_i \log \mu_i - \sum \mu_i\right] \exp\left[-\sum \log y_i!\right]$$

which is a member of the exponential family since it has the form

$$p(\mathbf{y}, \boldsymbol{\theta}) = b(\mathbf{y}) \exp\left[\mathbf{y}'\boldsymbol{\theta} - \kappa(\boldsymbol{\theta})\right]$$

with $b(\mathbf{y}) = \exp\left[-\sum \log y_i!\right]$
$\boldsymbol{\theta}$ a $p \times 1$ vector with $\theta_i = \log \mu_i$, that is $\mu_i = e^{\theta_i}$
$\kappa(\boldsymbol{\theta}) = \sum \mu_i = \sum \exp(\theta_i)$.

The mean vector of $\mathbf{Y}$ is given by

$$\begin{aligned}
E(\mathbf{Y}) &= \frac{\partial}{\partial \boldsymbol{\theta}}\kappa(\boldsymbol{\theta}) \\
&= \begin{pmatrix} \frac{\partial \kappa(\boldsymbol{\theta})}{\partial \theta_1} \\ \frac{\partial \kappa(\boldsymbol{\theta})}{\partial \theta_2} \\ \vdots \\ \frac{\partial \kappa(\boldsymbol{\theta})}{\partial \theta_p} \end{pmatrix} = \begin{pmatrix} e^{\theta_1} \\ e^{\theta_2} \\ \vdots \\ e^{\theta_p} \end{pmatrix} \\
&= \boldsymbol{\mu}.
\end{aligned}$$

The covariance matrix of $\mathbf{Y}$ is

$$\begin{aligned}
\text{Cov}(\mathbf{Y}) &= \frac{\partial^2}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}'}\kappa(\boldsymbol{\theta}) \\
&= \begin{pmatrix} \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_1\partial\theta_1} & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_1\partial\theta_2} & \cdots & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_1\partial\theta_p} \\ \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_2\partial\theta_1} & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_2\partial\theta_2} & \cdots & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_2\partial\theta_p} \\ \vdots & \vdots & & \vdots \\ \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_p\partial\theta_1} & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_p\partial\theta_2} & \cdots & \frac{\partial^2\kappa(\boldsymbol{\theta})}{\partial\theta_p\partial\theta_p} \end{pmatrix} = \begin{pmatrix} e^{\theta_1} & 0 & \cdots & 0 \\ 0 & e^{\theta_2} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & e^{\theta_p} \end{pmatrix} \\
&= \text{Diag}(\boldsymbol{\mu}).
\end{aligned}$$

1.2 COMPONENTS OF A GENERALIZED LINEAR MODEL

Suppose that $\mathbf{Y} : p \times 1$ is a random vector and that the joint probability function is a member of the natural exponential family with $E(\mathbf{Y}) = \boldsymbol{\mu}$. Let $\boldsymbol{\theta}$ be a $p \times 1$ vector of natural parameters.
A generalized linear model (GLM) consists of the following three components:

1. *The random component.*
   The random component, $\mathbf{Y}' = (Y_1, Y_2, \ldots, Y_p)$, refers to the vector with response variables from a distribution in the natural exponential family. That is, the joint probability function is of the form given in (1).

2. *The systematic component.*
   The systematic component relates parameters $\{\eta_i\}$ to the explanatory variables using a linear predictor
   $$\eta_i = \sum_j \beta_j x_{ij} \quad i = 1, 2, \ldots, p.$$
   In matrix form
   $$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}$$
   where $\boldsymbol{\eta} : p \times 1$, $\boldsymbol{\beta} : m \times 1$ are model parameters, and $\mathbf{X} : p \times m$ is the design matrix consisting of the values of the explanatory variables for the p observations.

3. *The link between the random and systematic components.*
   The link function h, connects the expected values of the random component, μ_i, to the linear predictor by
   $$h(\mu_i) = \eta_i$$
   where h is a monotonic differentiable function.
   A GLM links μ_i to the explanatory variables through the equation
   $$h(\mu_i) = \eta_i = \sum_j \beta_j x_{ij} \quad i = 1, 2, \ldots, p.$$
   The link function that transforms μ_i, to the natural parameter θ_i, is called the canonical link, for which
   $$h(\mu_i) = \eta_i = \theta_i = \sum_j \beta_j x_{ij}.$$

EXAMPLE 1.2
The components of a GLM for a loglinear model.

Suppose the elements of $\mathbf{Y} : 3 \times 1$ are independent Poisson random variables with parameter vector $\boldsymbol{\mu}$. The model to be fitted is $\mu_i = \alpha\gamma^{i-1}$ or, as a loglinear model

$$\log \mu_i = \log \alpha + (i-1) \log \gamma.$$

The generalized linear model is

$$\log \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}.$$

The three components of the GLM are:

1. The random component $\mathbf{Y}$.
   In Example 1.1 it was shown that the joint probability function of $\mathbf{Y}$ is a member of the natural exponential family.

2. The systematic component
   $$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$
   with $\boldsymbol{\beta}' = (\beta_1, \beta_2)$ where $\beta_1 = \log \alpha$ and $\beta_2 = \log \gamma$.

3. The link function, which is also a canonical link for this example, is given by
   $$\eta_i = h(\mu_i) = \log \mu_i = \theta_i = \sum_j \beta_j x_{ij}.$$

1.3 MEASURES OF GOODNESS OF FIT

Suppose that $\{\widehat{\mu}_i\}$ are the estimated frequencies for the contingency table on fitting an appropriate model to the data The following statistics can be used to test the goodness of fit of a model:

- The Pearson Chi-squared Statistic
  $$\chi^2 = \sum_{i=1}^{p} \frac{(\mu_i - \widehat{\mu}_i)^2}{\widehat{\mu}_i}.$$

- The Deviance
  A saturated GLM has as many parameters as observations, giving a perfect fit. In a saturated model all variation is consigned to the systematic component. For a given unsaturated model the ratio
  $$-2 \log \left(\frac{\text{maximum likelihood under model}}{\text{maximum likelihood under saturated model}} \right)$$
  describes lack of fit.
  The deviance, as defined by Nelder and Wedderburn (1972), is given by
  $$D = 2\left[L(\widehat{\boldsymbol{\mu}}, \mathbf{y}) - L(\mathbf{y}, \mathbf{y})\right]$$
  where $L(\widehat{\boldsymbol{\mu}}, \mathbf{y})$ is the log-likelihood maximized over some vector of parameters and $L(\mathbf{y}, \mathbf{y})$ is the maximum likelihood achievable in the saturated model.
  As an example consider the form of the deviance for the Poisson distribution.
  Let $Y_1, Y_2, \ldots, Y_n$ be n independent Poisson random variables with $E(Y_i) = \mu_i$. The log-likelihood function is
  $$\begin{aligned} L(\boldsymbol{\mu}, \mathbf{y}) &= \log \left[\prod \frac{e^{\mu_i} \mu_i^{y_i}}{y_i!} \right] \\ &= \sum y_i \log \widehat{\mu}_i - \sum \widehat{\mu}_i - \sum \log y_i! \end{aligned}$$

The deviance for a model with fitted values $\widehat{\mu}_i$ is

$$\begin{aligned} D &= 2\left[\sum y_i \log \widehat{\mu}_i - \sum \widehat{\mu}_i - \sum \log y_i! - \{\sum y_i \log y_i - \sum y_i - \sum \log y_i!\}\right] \\ &= 2\left[\sum y_i \log \frac{\widehat{\mu}_i}{y_i} + \sum (y_i - \widehat{\mu}_i)\right]. \end{aligned}$$

- The Wald Statistic
  If the model under consideration is formulated in terms of the constraints $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$ and $\mathbf{G} = \frac{\partial \mathbf{g}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}|_{\mu=\mathbf{y}}$ then the Wald statistic is

$$W = \mathbf{g}'(\mathbf{y}) \left(\mathbf{G}_{\mathbf{y}} \mathbf{V}_{\mathbf{y}} \mathbf{G}_{\mathbf{y}}'\right)^{-1} \mathbf{g}(\mathbf{y}).$$

2 MAXIMUM LIKELIHOOD ESTIMATION PROCEDURES

This chapter outlines the theory of the Newton-Raphson, Fisher-Scoring and EM algorithms as procedures for maximum likelihood estimation. The EM algorithm is specifically applied to the exponential family to determine ML estimates for incomplete data when the missing data mechanism is ignorable. A maximum likelihood estimation procedure for the mean of the exponential family, subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, is also discussed.

2.1 THE NEWTON-RAPHSON ALGORITHM

The Newton-Raphson method is an iterative procedure to determine the value $\widehat{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$ that maximizes a function $g(\boldsymbol{\beta})$.
Let $\boldsymbol{\beta}^{(r)}$ be the rth approximation of $\widehat{\boldsymbol{\beta}}$ where $r = 0, 1, 2, \ldots$. As described in Agresti (1990), the method requires an initial guess, $\boldsymbol{\beta}^{(0)}$, for the value that maximizes the function. At step r in the iterative process the function $g(\boldsymbol{\beta})$ is approximated by the terms up to the second order in the Taylor series expansion of $g(\boldsymbol{\beta})$ around $\boldsymbol{\beta}^{(r)}$, that is

$$Q^{(r)}(\boldsymbol{\beta}) = g\left(\boldsymbol{\beta}^{(r)}\right) + \mathbf{q}^{(r)\prime}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(r)}\right) + \tfrac{1}{2}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(r)}\right)' \mathbf{H}^{(r)}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(r)}\right) + o\left(\left\|\boldsymbol{\beta} - \boldsymbol{\beta}^{(r)}\right\|\right) \tag{2}$$

where $\mathbf{H}$ is the matrix having elements $\dfrac{\partial^2 g(\boldsymbol{\beta})}{\partial\beta_h \partial\beta_k}$, $\mathbf{q}$ is the vector having elements $\dfrac{\partial g(\boldsymbol{\beta})}{\partial\beta_k}$, and $\mathbf{H}^{(r)}$ and $\mathbf{q}^{(r)}$ are $\mathbf{H}$ and $\mathbf{q}$ evaluated at $\boldsymbol{\beta} = \boldsymbol{\beta}^{(r)}$.
The next approximation of $\widehat{\boldsymbol{\beta}}$ is in the location of the maximum value of (2).
Solving $\dfrac{\partial Q^{(r)}(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = \mathbf{q}^{(r)} + \mathbf{H}^{(r)}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(r)}\right) = \mathbf{0}$ for $\boldsymbol{\beta}$ yields the next approximation of $\widehat{\boldsymbol{\beta}}$,

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} - \left(\mathbf{H}^{(r)}\right)^{-1}\mathbf{q}^{(r)} \tag{3}$$

assuming $\mathbf{H}^{(r)}$ is nonsingular.
Iteration continues until convergence is attained.

EXAMPLE 2.1
Determining ML estimates using the Newton-Raphson algorithm.

The number of accidents per thousand per age group in a certain factory is given in Table 2.1.

TABLE 2.1: Accidents per 1000 per age group.

Age group	I	II	III
Number of accidents	80	15	5

Suppose the elements of $\mathbf{Y} : 3 \times 1$, the number of accidents for each category, are independent Poisson random variables with parameter vector $\boldsymbol{\mu}$ The observed vector is $\mathbf{y}' = (80, 15, 5)$. The model under consideration is $\mu_i = \alpha\gamma^{i-1}$ for $i = 1, 2, 3$. The likelihood function is given by

$$\begin{aligned} l(\boldsymbol{\mu}|\mathbf{y}) &= \frac{\exp\left(-\sum\mu_i\right)\prod\mu_i^{y_i}}{\prod y_i!} \\ &= \frac{\exp(-\alpha)\left(1+\gamma+\gamma^2\right)\alpha^{(y_1+y_2+y_3)}\gamma^{(y_2+2y_3)}}{\prod y_i!}. \end{aligned}$$

The value, $\widehat{\boldsymbol{\beta}}' = (\widehat{\alpha}, \widehat{\gamma})$, that maximizes l will also maximize the log-likelihood function

$$L(\boldsymbol{\beta}|\mathbf{y}) = (-\alpha)\left(1+\gamma+\gamma^2\right) + (y_1+y_2+y_3)\log(\alpha) + (y_2+2y_3)\log(\gamma) - \sum\log(y_i!)$$

and is determined iteratively with the expression

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} - \left(\mathbf{H}^{(r)}\right)^{-1} \mathbf{q}^{(r)} \tag{4}$$

where $\boldsymbol{\beta}^{(r)}$ is the rth approximation of $\widehat{\boldsymbol{\beta}}$, and $\mathbf{q}^{(r)}$ and $\mathbf{H}^{(r)}$ are $\mathbf{q}$ and $\mathbf{H}$ evaluated at $\boldsymbol{\beta} = \boldsymbol{\beta}^{(r)}$ with

$$\mathbf{q} = \frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \begin{pmatrix} \frac{\partial L(\boldsymbol{\beta})}{\partial \alpha} \\ \frac{\partial L(\boldsymbol{\beta})}{\partial \gamma} \end{pmatrix} = \begin{pmatrix} -\left(1+\gamma+\gamma^2\right) + \frac{y_1+y_2+y_3}{\alpha} \\ -\alpha(1+2\gamma) + \frac{y_2+2y_3}{\gamma} \end{pmatrix} \tag{5}$$

$$\mathbf{H} = \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}'} = \begin{pmatrix} \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \alpha^2} & \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \alpha \partial \gamma} \\ \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \gamma \partial \alpha} & \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \gamma^2} \end{pmatrix} = \begin{pmatrix} -\frac{(y_1+y_2+y_3)}{\alpha^2} & -(1+2\gamma) \\ -(1+2\gamma) & -2\alpha - \frac{(y_2+2y_3)}{\gamma^2} \end{pmatrix}. \tag{6}$$

From the model to be fitted $\alpha = \mu_1$ and $\gamma = \dfrac{\mu_2}{\alpha} = \dfrac{\mu_2}{\mu_1}$. If the observed data is used as an initial estimate of $\boldsymbol{\mu}$ the first approximation of $\widehat{\boldsymbol{\beta}}$ is

$$\boldsymbol{\beta}^{(0)} = \begin{pmatrix} \alpha^{(0)} \\ \gamma^{(0)} \end{pmatrix} = \begin{pmatrix} 80 \\ 0.1875 \end{pmatrix}$$

and is used to determine $\mathbf{q}^{(0)}$ and $\mathbf{H}^{(0)}$. Substituting $\boldsymbol{\beta}^{(0)}$, $\mathbf{q}^{(0)}$ and $\mathbf{H}^{(0)}$ into (4) gives

$$\boldsymbol{\beta}^{(1)} = \boldsymbol{\beta}^{(0)} - \left(\mathbf{H}^{(0)}\right)^{-1} \mathbf{q}^{(0)}.$$

This is used to determine $\mathbf{q}^{(1)}$ and $\mathbf{H}^{(1)}$.
The process continues until convergence is attained. Table 2.2 shows $\boldsymbol{\beta}^{(r)}$ at different steps of the algorithm.

TABLE 2.2: Values of $\boldsymbol{\beta}^{(r)}$ at different steps of the Newton-Raphson algorithm.

r	$\alpha^{(r)}$	$\gamma^{(r)}$
0	80	0.1875
1	79.294919	0.2153986
2	78.829748	0.2200938
3	78.821827	0.2201973
4	78.821823	0.2201973

The value $\widehat{\boldsymbol{\beta}}$ that maximizes the log-likelihood function is

$$\widehat{\boldsymbol{\beta}} = \begin{pmatrix} \widehat{\alpha} \\ \widehat{\gamma} \end{pmatrix} = \begin{pmatrix} 78.821823 \\ 0.2201973 \end{pmatrix}.$$

Substituting this into the model to be fitted, $\mu_i = \alpha\gamma^{i-1}$, gives

$$\widehat{\boldsymbol{\mu}} = \begin{pmatrix} \widehat{\mu}_1 \\ \widehat{\mu}_2 \\ \widehat{\mu}_3 \end{pmatrix} = \begin{pmatrix} \widehat{\alpha} \\ \widehat{\alpha}\widehat{\gamma} \\ \widehat{\alpha}\widehat{\gamma}^2 \end{pmatrix} = \begin{pmatrix} 78.821823 \\ 17.356354 \\ 3.8218228 \end{pmatrix}.$$

The program is given in the Appendix.

EXAMPLE 2.2
Determining ML estimates for a loglinear model using the Newton-Raphson algorithm.

Consider the model in Example 1.2 and Example 2.1. The log-likelihood function is

$$L(\boldsymbol{\mu}|\mathbf{y}) = \sum_i y_i \log \mu_i - \sum_i \mu_i - \sum_i \log y_i!. \tag{7}$$

In Example 1.2 the model $\mu_i = \alpha\gamma^{i-1}$ was written as the generalized linear model

$$\log \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$$

with $\beta_1 = \log\alpha$ and $\beta_2 = \log\gamma$, and $\mathbf{X}$ the design matrix.
Using the fact that $\log\mu_i = \sum_j \beta_j x_{ij}$ and $\mu_i = \exp\left(\sum_j \beta_j x_{ij}\right)$ the log-likelihood function in (7) can be written as a function of the elements of $\boldsymbol{\beta}$. That is

$$L(\boldsymbol{\beta}|\mathbf{y}) = \sum_i y_i \sum_j \beta_j x_{ij} - \sum_i \exp\left(\sum_j \beta_j x_{ij}\right) - \sum_i \log y_i!. \tag{8}$$

The value of $\widehat{\boldsymbol{\beta}}$ that maximizes $L(\boldsymbol{\beta}|\mathbf{y})$ can be found iteratively with

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} - \left(\mathbf{H}^{(r)}\right)^{-1}\mathbf{q}^{(r)} \tag{9}$$

where $\mathbf{q}$ is the vector with elements the first order partial derivatives

$$q_k = \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_k} = -\sum_i x_{ik} \exp\left(\sum_j \beta_j x_{ij}\right) + \sum_i y_i x_{ik}$$

and $\mathbf{H}$ is the matrix of second order partial derivatives having elements

$$h_{hk} = \frac{\partial^2 L(\boldsymbol{\beta})}{\partial\beta_h \partial\beta_k} = -\sum_i x_{ih}x_{ik} \exp\left(\sum_j \beta_j x_{ij}\right) = -\sum_i x_{ih}x_{ik}\mu_i.$$

From this

$$\mathbf{q}^{(r)} = \mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right) \tag{10}$$

$$\mathbf{H}^{(r)} = -\mathbf{X}'\text{diag}\left(\boldsymbol{\mu}^{(r)}\right)\mathbf{X} \tag{11}$$

with $\boldsymbol{\mu}^{(r)} = \exp\left(\mathbf{X}\boldsymbol{\beta}^{(r)}\right)$ the rth approximation of $\widehat{\boldsymbol{\mu}}$, $(r = 0, 1, 2, \ldots)$.
Substituting (10) and (11) into (9) gives

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left[\mathbf{X}'\text{diag}\left(\boldsymbol{\mu}^{(r)}\right)\mathbf{X}\right]^{-1}\mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right). \tag{12}$$

From the model to be fitted $\alpha = \mu_1$ and $\gamma = \dfrac{\mu_2}{\alpha} = \dfrac{\mu_2}{\mu_1}$. Using the observed data as an initial estimate of $\boldsymbol{\mu}$, the approximation of $\widehat{\boldsymbol{\beta}}$ at $r = 0$ is

$$\boldsymbol{\beta}^{(0)} = \begin{pmatrix} \log\alpha^{(0)} \\ \log\gamma^{(0)} \end{pmatrix} = \begin{pmatrix} 1.90309 \\ -0.72700 \end{pmatrix}.$$

This is used to determine $\boldsymbol{\mu}^{(0)} = \exp\left(\mathbf{X}\boldsymbol{\beta}^{(0)}\right)$. Substituting $\boldsymbol{\beta}^{(0)}$ and $\boldsymbol{\mu}^{(0)}$ in (12) gives the next approximation for $\widehat{\boldsymbol{\beta}}$,

$$\boldsymbol{\beta}^{(1)} = \boldsymbol{\beta}^{(0)} + \left[\mathbf{X}'\text{diag}\left(\boldsymbol{\mu}^{(0)}\right)\mathbf{X}\right]^{-1}\mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(0)}\right)$$

which is used to determine $\boldsymbol{\mu}^{(1)}$.
The process continues until convergence is attained and the value $\widehat{\boldsymbol{\beta}}$ that maximizes the log-likelihood function in (8) is

$$\widehat{\boldsymbol{\beta}} = \begin{pmatrix} \widehat{\beta}_1 \\ \widehat{\beta}_2 \end{pmatrix} = \begin{pmatrix} \log\widehat{\alpha} \\ \log\widehat{\gamma} \end{pmatrix} = \begin{pmatrix} 4.3671899 \\ -1.513231 \end{pmatrix}.$$

Substituting this into the model, $\mu_i = \alpha\gamma^{i-1}$, gives

$$\widehat{\boldsymbol{\mu}} = \exp\left(\mathbf{X}\widehat{\boldsymbol{\beta}}\right) = \begin{pmatrix} \widehat{\mu}_1 \\ \widehat{\mu}_2 \\ \widehat{\mu}_3 \end{pmatrix} = \begin{pmatrix} 78.821823 \\ 17.356354 \\ 3.8218228 \end{pmatrix}$$

This is the same result as obtained in Example 2.1.
The program is given in the Appendix.

2.2 THE FISHER SCORING ALGORITHM

The Fisher scoring algorithm is similar to the Newton-Raphson algorithm, the distinction being that Fisher scoring uses the information matrix. The information matrix is the negative expected value of the second order derivitave matrix of the function to be maximized. The Newton-Raphson algorithm uses the observed value of the second order derivitave matrix. The formula for Fisher scoring is

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left(\mathbf{Inf}^{(r)}\right)^{-1} \mathbf{q}^{(r)}$$

where $\mathbf{Inf}^{(r)}$ is the rth approximation for the estimated information matrix. The information matrix, $\mathbf{Inf}$, is the negative expected value of the matrix of second order partial derivatives of the log-likelihood and has elements $\text{Inf}_{hk} = -E\left(\frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_h \partial \beta_k}\right)$.

EXAMPLE 2.3
Determining ML estimates using the Fisher scoring algorithm.

Suppose the elements of $\mathbf{Y} : 3 \times 1$ are independent Poisson random variables with parameter vector $\boldsymbol{\mu}$ and observed vector $\mathbf{y}' = (80, 15, 5)$. The model to be fitted is $\mu_i = \alpha\gamma^{i-1}$. In Example 2.1 the Newton-Raphson algorithm was used to find the ML estimates.
The equation used in the iterative procedure is

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left(\mathbf{Inf}^{(r)}\right)^{-1} \mathbf{q}^{(r)}$$

where $\mathbf{Inf}^{(r)}$ is

$$\mathbf{Inf} = -E\left[\frac{\partial^2 L(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}'}\right] = \begin{pmatrix} -E\left(\frac{\partial^2 L(\boldsymbol{\beta})}{\partial\alpha^2}\right) & -E\left(\frac{\partial^2 L(\boldsymbol{\beta})}{\partial\alpha\partial\gamma}\right) \\ -E\left(\frac{\partial^2 L(\boldsymbol{\beta})}{\partial\gamma\partial\alpha}\right) & -E\left(\frac{\partial^2 L(\boldsymbol{\beta})}{\partial\gamma^2}\right) \end{pmatrix} = \begin{pmatrix} E\left(\frac{(y_1+y_2+y_3)}{\alpha}\right) & -(1+2\gamma) \\ -(1+2\gamma) & E\left(2\alpha + \frac{(y_2+2y_3)}{\gamma^2}\right) \end{pmatrix}$$

evaluated at $\boldsymbol{\beta}^{(r)}$.
Table 2.3 gives the values of $\boldsymbol{\beta}^{(r)}$ at different steps of the Fisher scoring algorithm.

TABLE 2.3: Values of $\boldsymbol{\beta}^{(r)}$ at different steps of the Fisher scoring algorithm.

r	$\alpha^{(r)}$	$\gamma^{(r)}$
0	80	0.1875
1	79.294919	0.2153986
2	78.820871	0.2201953
3	78.821823	0.2201973

This is the same result as obtained in Example 2.1 with the Newton-Raphson algorithm.
The program is given in the Appendix.

EXAMPLE 2.4
Determining ML estimates for a loglinear model using the Fisher scoring algorithm.

This example uses the model and data in Example 2.2 where the ML estimates for the GLM were found iteratively with the Newton-Raphson algorithm given by the equation

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left[\mathbf{X}'\text{diag}\left(\boldsymbol{\mu}^{(r)}\right)\mathbf{X}\right]^{-1}\mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right).$$

Since

$$\mathbf{H}^{(r)} = -\mathbf{X}'\text{diag}\left(\boldsymbol{\mu}^{(r)}\right)\mathbf{X}$$

is not a function of the observed data $\mathbf{y}$, the observed and expected second derivative matrices are the same. Thus

$$\mathbf{Inf} = -\mathbf{H}.$$

This happens for all GLMs that use a canonical link function. The Newton-Raphson and Fisher scoring algorithms are identical in such cases.

2.3 IGNORABLE MISSING DATA MECHANISM

The EM algorithm can be used to determine maximum likelihood estimates for incomplete data. Before presenting the theory of the EM algorithm, it is necessary to define an ignorable missing data mechanism.

Suppose the data of interest is denoted by $\mathbf{Y} = (Y_{ij}) : n \times p$ matrix of n observations measured for p variables. The data is assumed to be generated by a model with probability function $f(\mathbf{y}|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is the vector of unknown parameters. In the case of incomplete data let $\mathbf{Y}' = (\mathbf{Y}'_{obs}, \mathbf{Y}'_{mis})$ where $\mathbf{Y}_{obs}$ represents the observed part of $\mathbf{Y}$ and $\mathbf{Y}_{mis}$ denotes the missing values. The joint probability function of $\mathbf{Y}_{obs}$ and $\mathbf{Y}_{mis}$ is given by $f(\mathbf{y}|\boldsymbol{\theta}) = f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta})$.
An indicator random variable is included in the model which indicates whether each component of $\mathbf{Y}$ is observed or missing. Define a response indicator $\mathbf{R} = (R_{ij})$ such that

$$R_{ij} = \begin{cases} 1, & y_{ij} \text{ observed}, \\ 0, & y_{ij} \text{ missing}. \end{cases}$$

The joint probability function of $\mathbf{R}$ and $\mathbf{Y}$ can be written as

$$f(\mathbf{y}, \mathbf{r}|\boldsymbol{\theta}, \boldsymbol{\psi}) = f(\mathbf{y}|\boldsymbol{\theta}) f(\mathbf{r}|\mathbf{y}, \boldsymbol{\psi}) \tag{13}$$

where $f(\mathbf{r}|\mathbf{y}, \boldsymbol{\psi})$ is the distribution of the missing data mechanism. This mechanism depends on $\mathbf{Y}$ and some unknown vector of parameters $\boldsymbol{\psi}$. In the case where the distribution of the missing data mechanism does not depend on the missing values $\mathbf{Y}_{mis}$, the data is said to be missing at random (MAR) and

$$f(\mathbf{r}|\mathbf{y}_{obs}, \mathbf{y}_{mis}, \boldsymbol{\psi}) = f(\mathbf{r}|\mathbf{y}_{obs}, \boldsymbol{\psi}). \tag{14}$$

MAR requires only that the missing values behave like a random sample within subclasses defined by the observed data. If the missing data values are a random sample of all data values the data is said to be missing completely at random (MCAR).
The observed data consist of the values of the variables $(\mathbf{Y}_{obs}, \mathbf{R})$ and its probability function is obtained by integrating out the missing data $\mathbf{Y}_{mis}$:

$$f(\mathbf{y}_{obs}, \mathbf{r}|\boldsymbol{\theta}, \boldsymbol{\psi}) = \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta}) f(\mathbf{r}|\mathbf{y}_{obs}, \mathbf{y}_{mis}, \boldsymbol{\psi}) \, d\mathbf{y}_{mis}. \tag{15}$$

The likelihood of $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ is proportional to (15), that is

$$l(\boldsymbol{\theta}, \boldsymbol{\psi}|\mathbf{y}_{obs}, \mathbf{r}) \propto f(\mathbf{y}_{obs}, \mathbf{r}|\boldsymbol{\theta}, \boldsymbol{\psi}). \tag{16}$$

If the data is missing at random, that is if (14) holds, the probability function of the observed data, given in (15), can be written as

$$\begin{aligned} f(\mathbf{y}_{obs}, \mathbf{r}|\boldsymbol{\theta}, \boldsymbol{\psi}) &= \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta}) f(\mathbf{r}|\mathbf{y}_{obs}, \boldsymbol{\psi}) \, d\mathbf{y}_{mis} \\ &= f(\mathbf{r}|\mathbf{y}_{obs}, \boldsymbol{\psi}) \times \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta}) \, d\mathbf{y}_{mis} \\ &= f(\mathbf{r}|\mathbf{y}_{obs}, \boldsymbol{\psi}) f(\mathbf{y}_{obs}|\boldsymbol{\theta}). \end{aligned} \tag{17}$$

The likelihood of the observed data under MAR can thus be factored into two pieces, one pertaining to the parameter of interest $\boldsymbol{\theta}$, and the other to $\boldsymbol{\psi}$. The parameters $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ are distinct if the joint parameter space of $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ is the product of the parameter space of $\boldsymbol{\theta}$ and the parameter space of $\boldsymbol{\psi}$. If both MAR and distinctness hold, the missing data mechanism is said to be ignorable (Little and Rubin, 1987) and likelihood based inferences about $\boldsymbol{\theta}$ will be unaffected by $\boldsymbol{\psi}$ or $f(\mathbf{r}|\mathbf{y}_{obs}, \boldsymbol{\psi})$.
From equation (17) it follows that

$$f(\mathbf{y}_{obs}, \mathbf{r}|\boldsymbol{\theta}, \boldsymbol{\psi}) \propto f(\mathbf{y}_{obs}|\boldsymbol{\theta})$$

and thus

$$l(\boldsymbol{\theta}, \boldsymbol{\psi}|\mathbf{y}_{obs}, \mathbf{r}) \propto l(\boldsymbol{\theta}|\mathbf{y}_{obs})$$

which means that all relevant statistical information about the parameters is contained in the observed data likelihood, $l(\boldsymbol{\theta}|\mathbf{y}_{obs})$.

EXAMPLE 2.5
Incomplete univariate data with an ignorable missing data mechanism.

Let $\mathbf{Y} : n \times 1$ denote a vector of n independent identically distributed random variables. Let $\mathbf{Y}' = (\mathbf{Y}'_{obs}, \mathbf{Y}'_{mis})$ with $\mathbf{Y}'_{obs} = (Y_1, Y_2, \ldots, Y_m)$ and $\mathbf{Y}'_{mis} = (Y_{m+1}, Y_{m+2}, \ldots, Y_n)$. That is, m units are observed and $n-m$ are missing. Let $\mathbf{R}' = (R_1, R_2, \ldots, R_n)$ denote the response indicators, where $R_i = 1$ if y_i is observed and $R_i = 0$ if y_i is missing. Suppose that each unit is observed with probability ψ. The missing data mechanism is

$$f(\mathbf{r}|\mathbf{y},\psi) = \prod_{i=1}^{n} \psi^{r_i} (1-\psi)^{1-r_i} = \psi^m (1-\psi)^{n-m}$$

and since it does not depend on $\mathbf{Y}_{mis}$ the data is MAR. If $\boldsymbol{\theta}$ and ψ are distinct, inferences about $\boldsymbol{\theta}$ can be based on the observed data likelihood

$$\begin{aligned} l(\boldsymbol{\theta}|\mathbf{y}_{obs}) &= \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta})\, d\mathbf{y}_{mis} \\ &= \int \cdots \int \prod_{i=1}^{m} f(y_i|\boldsymbol{\theta}) \prod_{i=m+1}^{n} f(y_i|\boldsymbol{\theta})\, dy_{m+1} \cdots dy_n. \\ &= \prod_{i=1}^{m} f(y_i|\boldsymbol{\theta}) \end{aligned}$$

which is a complete data likelihood based on the reduced sample $(Y_1, Y_2, \ldots, Y_m)'$.

EXAMPLE 2.6
Bivariate data with one variable subject to nonresponse if the missing data mechanism is ignorable.

Consider a dataset with variables Y_1 and Y_2 where Y_1 is observed for units $1, 2, \ldots, n$ and Y_2 is observed only for units $1, 2, \ldots, m < n$. The missing data will be MAR if the probability that Y_2 is missing does not depend on Y_2, although it may possibly depend on Y_1. Let y_{i1} and y_{i2} denote the values of Y_1 and Y_2, respectively, for unit i. Since

$$f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta}) = f(\mathbf{y}_{obs}|\boldsymbol{\theta})\, f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})$$

the observed data likelihood can be written as

$$\begin{aligned} l(\boldsymbol{\theta}|\mathbf{y}_{obs}) &= \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta})\, d\mathbf{y}_{mis} \\ &= \int f(\mathbf{y}_{obs}|\boldsymbol{\theta})\, f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})\, d\mathbf{y}_{mis} \\ &= \int \prod_{i=1}^{m} f(y_{i1}, y_{i2}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} f(y_{i1}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} f(y_{i2}|y_{i1}, \boldsymbol{\theta})\, d\mathbf{y}_{mis} \\ &= \prod_{i=1}^{m} f(y_{i1}, y_{i2}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} f(y_{i1}|\boldsymbol{\theta}) \int \prod_{i=m+1}^{n} f(y_{i2}|y_{i1}, \boldsymbol{\theta})\, d\mathbf{y}_{mis} \\ &= \prod_{i=1}^{m} f(y_{i1}, y_{i2}|\boldsymbol{\theta}) \prod_{i=m+1}^{n} f(y_{i1}|\boldsymbol{\theta}). \end{aligned}$$

This is the product of the joint likelihood for Y_1 and Y_2 where Y_1 and Y_2 are both observed, and the likelihood of Y_1 where only Y_1 is observed.

2.4 THE EM ALGORITHM

2.4.1 Theory of the EM Algorithm

Assuming that the ignorability assumption is correct, all relevant statistical information about the parameters is contained in the observed data likelihood, $l(\boldsymbol{\theta}|\mathbf{y}_{obs})$. The EM algorithm uses the interdependence that exists between the missing data $\mathbf{Y}_{mis}$ and the parameters $\boldsymbol{\theta}$. An initial estimate of $\boldsymbol{\theta}$ is obtained from the observed data $\mathbf{Y}_{obs}$. The missing data is filled in based on this initial estimate of $\boldsymbol{\theta}$ and $\boldsymbol{\theta}$ is then re-estimated based on $\mathbf{Y}_{obs}$ and the filled in $\mathbf{Y}_{mis}$. The process iterates until the estimates converge.
Suppose the density function of the complete data $\mathbf{y}$ is given by $f(\mathbf{y}|\boldsymbol{\theta})$ where $\boldsymbol{\theta}$ is the unknown parameter. Let $\mathbf{Y}' = (\mathbf{Y}'_{obs}, \mathbf{Y}'_{mis})$ where $\mathbf{Y}_{obs}$ represents the observed part of $\mathbf{Y}$ and $\mathbf{Y}_{mis}$ denotes the missing values. The distribution of the complete data can be factored as

$$f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta}) = f(\mathbf{y}_{obs}|\boldsymbol{\theta})\, f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta}). \tag{18}$$

The objective is to maximize the likelihood function for the observed data, that is maximize

$$l(\boldsymbol{\theta}|\mathbf{y}_{obs}) \propto \int f(\mathbf{y}_{obs}, \mathbf{y}_{mis}|\boldsymbol{\theta})\, d\mathbf{y}_{mis}$$

with respect to $\boldsymbol{\theta}$ or, alternatively, to maximize the log-likelihood

$$L(\boldsymbol{\theta}|\mathbf{y}_{obs}) = \log\left[l(\boldsymbol{\theta}|\mathbf{y}_{obs})\right].$$

The log-likelihood that corresponds to (18) is

$$L(\boldsymbol{\theta}|\mathbf{y}_{obs}, \mathbf{y}_{mis}) = L(\boldsymbol{\theta}|\mathbf{y}_{obs}) + \log\left[f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})\right]$$

and can be written as

$$L(\boldsymbol{\theta}|\mathbf{y}_{obs}) = L(\boldsymbol{\theta}|\mathbf{y}_{obs}, \mathbf{y}_{mis}) - \log\left[f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})\right] \tag{19}$$

where $L(\boldsymbol{\theta}|\mathbf{y}_{obs})$ is the observed log-likelihood to be maximized, $L(\boldsymbol{\theta}|\mathbf{y}_{obs}, \mathbf{y}_{mis})$ is the complete data log-likelihood and $\log\left[f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})\right]$ is the missing part of the complete data log-likelihood.
The expectation of both sides of (19) over the distribution of the missing data $\mathbf{Y}_{mis}$, given $\mathbf{Y}_{obs}$ and a current estimate of $\boldsymbol{\theta}$, say $\boldsymbol{\theta}^{(r)}$ is

$$L(\boldsymbol{\theta}|\mathbf{y}_{obs}) = Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right) - H\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right) \tag{20}$$

where

$$Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right) = \int \left[L(\boldsymbol{\theta}|\mathbf{y}_{obs}, \mathbf{y}_{mis})\right] f\left(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta}^{(r)}\right) d\mathbf{y}_{mis} \tag{21}$$

and

$$H\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right) = \int \left\{\log\left[f(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta})\right]\right\} f\left(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta}^{(r)}\right) d\mathbf{y}_{mis}. \tag{22}$$

From Jensen's inequality (Rao 1972)

$$H\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right) \leq H\left(\boldsymbol{\theta}^{(r)}|\boldsymbol{\theta}^{(r)}\right) \tag{23}$$

and therefore maximization of $L(\boldsymbol{\theta}|\mathbf{y}_{obs})$ is equivalent to maximization of $Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right)$ with respect to $\boldsymbol{\theta}$.
Each step of the EM algorithm consists of an E-step (expectation step) and an M-step (maximization step):

- In the E-step the function $Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right)$ is calculated by averaging the complete data log-likelihood $L(\boldsymbol{\theta}|\mathbf{y})$ over $f\left(\mathbf{y}_{mis}|\mathbf{y}_{obs}, \boldsymbol{\theta}^{(r)}\right)$.
- In the M-step $\boldsymbol{\theta}^{(r+1)}$ is found by maximizing $Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right)$. That is $Q\left(\boldsymbol{\theta}^{(r+1)}|\boldsymbol{\theta}^{(r)}\right) \geq Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}\right)$ for all $\boldsymbol{\theta}$.

2.4.2 The EM Algorithm for exponential families

Little and Rubin (1987) presents a simple characterization of the EM algorithm when $f(\mathbf{y}|\boldsymbol{\theta})$ has the form for the regular exponential family defined by

$$f(\mathbf{y}|\boldsymbol{\theta}) = b(\mathbf{y}) \exp\left(\mathbf{s}(\mathbf{y})' \boldsymbol{\theta}\right) / a(\boldsymbol{\theta}) \tag{24}$$

where $\boldsymbol{\theta}$ is the parameter vector and $\mathbf{s}(\mathbf{Y})$ is the vector of complete data sufficient statistics. For regular exponential families the complete data MLE can be found as a solution to the likelihood equations

$$E(\mathbf{s}(\mathbf{Y})|\boldsymbol{\theta}) = \mathbf{s} \tag{25}$$

where $\mathbf{s}$ is the realized value of the vector $\mathbf{s}(\mathbf{Y})$.
Suppose $\boldsymbol{\theta}^{(r)}$ denotes the current value $\boldsymbol{\theta}$ after r cycles of the algorithm. The next cycle can be described in two steps, as follows:

- E-step: Estimate the complete data sufficient statistics $\mathbf{s}(\mathbf{Y})$ by finding

$$\mathbf{s}^{(r)} = E\left(\mathbf{s}(\mathbf{Y})|\mathbf{Y}_{obs}, \boldsymbol{\theta}^{(r)}\right). \tag{26}$$

- M-step: The M-step determines the new estimate $\boldsymbol{\theta}^{(r+1)}$ of $\boldsymbol{\theta}$ as the solution of the equations

$$E(\mathbf{s}(\mathbf{Y})|\boldsymbol{\theta}) = \mathbf{s}^{(r)} \tag{27}$$

  which are the likelihood equations for the complete data with $\mathbf{s}(\mathbf{Y})$ replaced by $\mathbf{s}^{(r)}$ as obtained in the E-step in (26).

EXAMPLE 2.7
Incomplete univariate normal data. EM algorithm for the regular exponential family.

Suppose Y_i, $i = 1, 2, \ldots, n$ are independent identically distributed random variables from a $N(\mu, \sigma^2)$ distribution. Let $\boldsymbol{\theta}' = (\mu, \sigma^2)$. The log-likelihood function for the complete data is

$$\begin{aligned} L(\boldsymbol{\theta}|\mathbf{y}) &= -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu)^2 \\ &= -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\left[\sum_{i=1}^{n} y_i^2 - 2\mu\sum_{i=1}^{n} y_i + n\mu^2\right] \end{aligned}$$

which is linear in the sufficient statistics $\mathbf{s}(\mathbf{Y}) = (s_1(\mathbf{Y}), s_2(\mathbf{Y})) = \left(\sum_{i=1}^{n} Y_i, \sum_{i=1}^{n} Y_i^2\right)$.
With no missing data the ML estimates of μ and σ^2 are

$$\begin{aligned} \widehat{\mu} &= \frac{1}{n}\sum_{i=1}^{n} y_i \\ \widehat{\sigma}^2 &= \frac{\sum_{i=1}^{n} y_i^2}{n} - \left(\frac{\sum_{i=1}^{n} y_i}{n}\right)^2 \end{aligned}$$

Suppose now that only the first m components of the data vector $\mathbf{Y}$ are observed and that the data are missing at random (MAR).

The E-step of the EM algorithm calculates

$$s_1^{(r)} = E\left(s_1(\mathbf{Y})|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right) = E\left(\sum_{i=1}^{n} Y_i|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right) = \sum_{i=1}^{m} y_i + (n-m)\,\mu^{(r)}$$

$$s_2^{(r)} = E\left(s_2(\mathbf{Y})|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right) = \sum_{i=1}^{m} y_i^2 + (n-m)\left[\left(\mu^{(r)}\right)^2 + \sigma^{2(r)}\right]$$

for current estimates $\boldsymbol{\theta}^{(r)} = \left(\mu^{(r)}, \sigma^{2(r)}\right)$ of the parameters. In the M-step the expectations of the sufficient statistics calculated in the E-step are substituted in the expressions for the ML estimates giving

$$\begin{aligned}\mu^{(r+1)} &= \frac{1}{n}E\left(\sum_{i=1}^{n} Y_i|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right)\\ &= \frac{1}{n}\left[\sum_{i=1}^{m} y_i + (n-m)\,\mu^{(r)}\right]\end{aligned}$$

and

$$\begin{aligned}\sigma^{2(r+1)} &= \frac{1}{n}E\left(\sum_{i=1}^{n} Y_i^2|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right) - \left(\mu^{(r+1)}\right)^2\\ &= \frac{1}{n}\left[\sum_{i=1}^{m} y_i^2 + (n-m)\left[\left(\mu^{(r)}\right)^2 + \sigma^{2(r)}\right]\right] - \left(\mu^{(r+1)}\right)^2.\end{aligned}$$

Numerical Example

Suppose Y_i, $i = 1,2,\ldots,10$ are independent identically distributed random variables from a $N(12,9)$ distribution and that Y_i are observed for $i = 1,2,\ldots,6$ and missing for $i = 7,\ldots,10$. The 6 observed values are 12.893, 7.012, 12.165, 12.274, 14.657 and 8.644.

The initial values of $\mu^{(0)} = 10$ and $\sigma^{2(0)} = 10$ were chosen arbitrarily. Table 2.4 displays the results at different steps of the algorithm until convergence. The results are the same as the mean and variance for the six observed data points, that is

$$\begin{aligned}\widehat{\mu} &= \frac{1}{6}\sum_{i=1}^{6} y_i\\ \widehat{\sigma}^2 &= \frac{\sum_{i=1}^{6} y_i^2}{6} - \widehat{\mu}^2\end{aligned}$$

TABLE 2.4: Iterations of the EM algorithm for incomplete univariate normal data, $n = 10$ and $m = 6$.

	M-Step		E-Step	
r	$\mu^{(r)}$	$\sigma^{2(r)}$	$E\left(\sum_{i=1}^{n} Y_i\|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right)$	$E\left(\sum_{i=1}^{n} Y_i^2\|\mathbf{Y}_{obs},\boldsymbol{\theta}^{(r)}\right)$
0	10	10	107.645	1243.582
1	10.765	8.4884	110.703	1301.015
2	11.070	7.550	111.926	1323.988
3	11.193	7.124	112.415	1333.178
4	11.242	6.945	112.611	1336.853
5	11.261	6.873	112.689	1338.324
6	11.269	6.843	112.721	1338.912
7	11.272	6.831	112.733	1339.147
8	11.273	6.827	112.738	1339.241
9	11.274	6.825	112.740	1339.279
10	11.274	6.824	112.741	1339.294
11	11.274	6.824	112.741	1339.300
∞	11.274	6.824	112.741	1339.300

EXAMPLE 2.8

EM algorithm for data from a multinomial distribution.

This example, discussed by Dempster, Laird and Rubin (1977) gives the data in which 197 animals are distributed multinomially into five categories. The complete data, $\mathbf{Y}' = (Y_1, Y_2, Y_3, Y_4, Y_5)$, are the counts for each category and the cell probabilities in this model are given as

$$\pi' = \left(\tfrac{1}{2}, \tfrac{1}{4}p, \tfrac{1}{4}(1-p), \tfrac{1}{4}(1-p), \tfrac{1}{4}p\right) \text{ for some } 0 \leq p \leq 1.$$

For the complete data the density function is

$$f(\mathbf{y}|p) = \frac{(y_1+y_2+y_3+y_4+y_5)!}{y_1!y_2!y_3!y_4!y_5!}\left(\tfrac{1}{2}\right)^{y_1}\left(\tfrac{1}{4}p\right)^{y_2}\left(\tfrac{1}{4}-\tfrac{1}{4}p\right)^{y_3}\left(\tfrac{1}{4}-\tfrac{1}{4}p\right)^{y_4}\left(\tfrac{1}{4}p\right)^{y_5}.$$

The ML estimate of p for the complete data is given by

$$\widehat{p} = \frac{y_2+y_5}{y_2+y_3+y_4+y_5}. \tag{28}$$

The kernel of the complete data log-likelihood is

$$L(p|\mathbf{y}) = y_1 \log \tfrac{1}{2} + (y_2+y_5)\log \tfrac{1}{4}p + (y_3+y_4)\log\left(\tfrac{1}{4}-\tfrac{1}{4}p\right)$$

and the counts are the sufficient statistics.
The observed data is $\mathbf{y}'_{obs} = (y_1+y_2, y_3, y_4, y_5) = (125, 18, 20, 34)$. Only the total of Y_1 and Y_2 is observed. In the E-step the conditional expectations of the sufficient statistics, Y_i, $i = 2,3,4,5$, given the observed values and a current estimate of p, are calculated. At step r $(r = 0, 1, 2, \ldots)$

$$\begin{aligned}
E(Y_2|\mathbf{Y}_{obs}, p^{(r)}) &= 125\frac{\frac{1}{4}p^{(r)}}{\frac{1}{2}+\frac{1}{4}p^{(r)}} \\
E(Y_3|\mathbf{Y}_{obs}, p^{(r)}) &= 18 \\
E(Y_4|\mathbf{Y}_{obs}, p^{(r)}) &= 20 \\
E(Y_5|\mathbf{Y}_{obs}, p^{(r)}) &= 34.
\end{aligned}$$

In the M-step the conditional expectations of Y_i as calculated in the E-step are substituted in expression (28) giving the next estimate of $\widehat{p}$ in the iterative process

$$\begin{aligned}
p^{(r+1)} &= \frac{E(Y_2|\mathbf{Y}_{obs}, p^{(r)}) + 34}{E(Y_2|\mathbf{Y}_{obs}, p^{(r)}) + 18 + 20 + 34} \\
&= \frac{125\frac{\frac{1}{4}p^{(r)}}{\frac{1}{2}+\frac{1}{4}p^{(r)}} + 34}{125\frac{\frac{1}{4}p^{(r)}}{\frac{1}{2}+\frac{1}{4}p^{(r)}} + 18 + 20 + 34}.
\end{aligned}$$

The process iterates between the E-step and the M-step until convergence is attained.
Table 2.5 shows that, starting from $p^{(0)} = 0.5$, the EM algorithm converges after seven steps.

TABLE 2.5: Iterations of the EM algorithm.

	M-step	E-step
r	$p^{(r)}$	$E(Y_2\|\mathbf{Y}_{obs}, p^{(r)})$
0	0.5	25
1	0.608247	29.15020
2	0.624321	29.73727
3	0.626489	29.82589
4	0.626777	29.82634
5	0.626816	29.82773
6	0.626821	29.82792
7	0.626821	29.82794
∞	0.626821	29.82794

2.5 A MAXIMUM LIKELIHOOD ESTIMATION PROCEDURE WHEN MODELLING IN TERMS OF CONSTRAINTS

Proposition 1
Suppose $\mathbf{Y}$ is a random vector with probability function belonging to the exponential family and with $E(\mathbf{Y}) = \boldsymbol{\mu}$. Matthews (1995) derives a ML estimate of $\boldsymbol{\mu}$ subject to the constraints $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, as

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - (\mathbf{G}_\mu \mathbf{V})' \left(\mathbf{G}_\mathbf{y} \mathbf{V} \mathbf{G}_\mu'\right)^{-1} g(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|) \tag{29}$$

where $\mathbf{g}(\boldsymbol{\mu})$ is a continuous vector valued function of $\boldsymbol{\mu}$ for which the first order partial derivatives exist, $\mathbf{G}_\mu = \dfrac{\partial \mathbf{g}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}$, $\mathbf{G}_\mathbf{y} = \dfrac{\partial \mathbf{g}(\boldsymbol{\mu})}{\partial \boldsymbol{\mu}}|_{\mu=\mathbf{y}}$ and $\mathbf{V}$ is the covariance matrix which could be known or could be some function of $\boldsymbol{\mu}$, say $\mathbf{V}_\mu$. This result implies that the ML estimate must be obtained iteratively.

Matthews (1995) gives the following proof of this result.

Proof:
Let $\boldsymbol{\gamma}$ be a vector of Lagrange multipliers. To find the ML estimate of $\boldsymbol{\mu}$ subject to the constraints $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$, we maximize

$$\frac{\partial}{\partial \boldsymbol{\mu}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) = \ln b(\mathbf{y}) + \mathbf{y}'\boldsymbol{\theta} - \kappa(\boldsymbol{\theta}) + \boldsymbol{\gamma}' \mathbf{g}(\boldsymbol{\mu}(\boldsymbol{\theta})).$$

Hence we find

$$\frac{\partial}{\partial \boldsymbol{\mu}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) = \frac{\partial}{\partial \boldsymbol{\theta}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) \left[\frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\mu}}\right].$$

Since we set $\dfrac{\partial}{\partial \boldsymbol{\mu}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) = \mathbf{0}$ for a maximum, and since $\left[\dfrac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\mu}}\right]$ is invertible for a regular exponential family, we need further only consider $\dfrac{\partial}{\partial \boldsymbol{\theta}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma})$.
Thus

$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\theta}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) &= \mathbf{y} - \frac{\partial}{\partial \boldsymbol{\theta}} \kappa(\boldsymbol{\theta}) + \frac{\partial}{\partial \boldsymbol{\theta}} \{\boldsymbol{\gamma}' \mathbf{g}(\boldsymbol{\mu}(\boldsymbol{\theta}))\} \\
&= \mathbf{y} - \boldsymbol{\mu} + \left\{\frac{\partial}{\partial \boldsymbol{\mu}} \mathbf{g}(\boldsymbol{\mu}(\boldsymbol{\theta})) \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]\right\}' \boldsymbol{\gamma} \\
&= \mathbf{y} - \boldsymbol{\mu} + \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu' \boldsymbol{\gamma}.
\end{aligned}$$

Setting $\dfrac{\partial}{\partial \boldsymbol{\theta}} \omega(\mathbf{y}; \boldsymbol{\theta}; \boldsymbol{\gamma}) = \mathbf{0}$, we get

$$\boldsymbol{\mu} = \mathbf{y} + \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu' \boldsymbol{\gamma}. \tag{30}$$

Using the linear Taylor expansion of $\mathbf{g}(\boldsymbol{\mu})$ about $\mathbf{y}$, we get

$$\begin{aligned}
\mathbf{g}(\boldsymbol{\mu}) &= \mathbf{g}\left(\mathbf{y} + \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu' \boldsymbol{\gamma}\right) \\
&= \mathbf{g}(\mathbf{y}) + \mathbf{G}_y \left(\mathbf{y} + \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu' \boldsymbol{\gamma} - \mathbf{y}\right) + o(\|\mathbf{y} - \boldsymbol{\mu}\|) \\
&= \mathbf{g}(\mathbf{y}) + \mathbf{G}_y \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu' \boldsymbol{\gamma} + o(\|\mathbf{y} - \boldsymbol{\mu}\|).
\end{aligned}$$

Setting $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$ and solving for $\boldsymbol{\gamma}$, gives

$$\boldsymbol{\gamma} = -\left(\mathbf{G}_y \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}_\mu'\right)^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

Substituting γ in (30) we get

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \left(\mathbf{G}_{\boldsymbol{\mu}} \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]\right)' \left(\mathbf{G}_{\mathbf{y}} \left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' \mathbf{G}'_{\boldsymbol{\mu}}\right)^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

Now

$$\frac{\partial \mu_i}{\partial \theta_j} = \frac{\partial}{\partial \theta_j}\left\{\frac{\partial \kappa(\boldsymbol{\theta})}{\partial \theta_i}\right\} = \frac{\partial^2 \kappa(\boldsymbol{\theta})}{\partial \theta_j \partial \theta_i}.$$

Hence

$$\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}} = \left[\frac{\partial \mu_i}{\partial \theta_j}\right] = \frac{\partial^2 \kappa(\boldsymbol{\theta})}{\partial \theta_j \partial \theta_i}$$

and

$$\left[\frac{\partial \boldsymbol{\mu}}{\partial \boldsymbol{\theta}}\right]' = \left[\frac{\partial^2 \kappa(\boldsymbol{\theta})}{\partial \theta_j \partial \theta_i}\right] = \mathbf{V}.$$

Therefore

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - (\mathbf{G}_{\boldsymbol{\mu}} \mathbf{V})' (\mathbf{G}_{\mathbf{y}} \mathbf{V} \mathbf{G}'_{\boldsymbol{\mu}})^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|)$$

which is the required result.

The iterative procedure

The process is a double iteration over $\mathbf{y}$ and $\boldsymbol{\mu}$. Let $\boldsymbol{\mu}^{(i,j)}$ denote the (i,j)th approximation obtained for the ML estimate $\widehat{\boldsymbol{\mu}}_c$ of $\boldsymbol{\mu}$, where i $(i = 0, 1, 2, \ldots)$ refers to iteration over $\boldsymbol{\mu}$, and j $(j = 0, 1, 2, \ldots)$ refers to iteration over $\mathbf{y}$. Note that $j = 0$ at the start of every iteration over $\mathbf{y}$.

The initial value for $\boldsymbol{\mu}$ is $\boldsymbol{\mu}^{(0,0)} = \mathbf{y}$, the vector of observed values. Iteration then takes place over $\mathbf{y}$ and the value of $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{V}_{\boldsymbol{\mu}}$ is kept constant at $\boldsymbol{\mu}^{(0,0)} = \mathbf{y}$. The first approximation of $\widehat{\boldsymbol{\mu}}_c$ is given by

$$\boldsymbol{\mu}^{(0,1)} = \mathbf{y} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,0)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{G}_{\mathbf{y}} \mathbf{V}_{\boldsymbol{\mu}^{(0,0)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,0)}}\right)^{-1} \mathbf{g}(\mathbf{y}).$$

If convergence over $\mathbf{y}$ is not attained at this step, $\mathbf{y}$ is replaced by $\boldsymbol{\mu}^{(0,1)}$ to obtain the next approximation of $\widehat{\boldsymbol{\mu}}_c$, whilst the estimated value for $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{V}_{\boldsymbol{\mu}}$ is kept constant at $\boldsymbol{\mu}^{(0,0)} = \mathbf{y}$. Thus,

$$\boldsymbol{\mu}^{(0,2)} = \boldsymbol{\mu}^{(0,1)} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,0)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,1)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,0)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,0)}}\right)^{-1} \mathbf{g}\left(\boldsymbol{\mu}^{(0,1)}\right).$$

This is repeated until convergence over $\mathbf{y}$ is attained, say at $j = k$.

The value at convergence, $\boldsymbol{\mu}^{(0,k)}$, is used as the next estimate for $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{V}_{\boldsymbol{\mu}}$. The procedure again iterates over $\mathbf{y}$, starting with the vector of observed values, $\mathbf{y}$, and keeping the estimated value for $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{V}_{\boldsymbol{\mu}}$ constant at $\boldsymbol{\mu}^{(0,k)}$. That is

$$\boldsymbol{\mu}^{(1,1)} = \mathbf{y} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,k)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,k)}}\right)' \left(\mathbf{G}_{\mathbf{y}} \mathbf{V}_{\boldsymbol{\mu}^{(0,k)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,k)}}\right)^{-1} \mathbf{g}(\mathbf{y}).$$

If convergence over $\mathbf{y}$ is not obtained at this step, the next approximation of $\widehat{\boldsymbol{\mu}}_c$ is

$$\boldsymbol{\mu}^{(1,2)} = \boldsymbol{\mu}^{(1,1)} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,k)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,k)}}\right)' \left(\mathbf{G}_{\boldsymbol{\mu}^{(1,1)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,k)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,k)}}\right)^{-1} \mathbf{g}\left(\boldsymbol{\mu}^{(1,1)}\right).$$

At convergence the iteration over $\mathbf{y}$ yields the next estimate for $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{V}_{\boldsymbol{\mu}}$. The process continues until convergence over $\boldsymbol{\mu}$ is attained.

In certain cases the iterative procedure simplifies to an iteration only over $\mathbf{y}$ or only over $\boldsymbol{\mu}$.

- If $\mathbf{g}$ is a linear function of $\boldsymbol{\mu}$, say $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{A}\boldsymbol{\mu}$ then $\mathbf{G}_{\mu} = \mathbf{A} = \mathbf{G}_{\mathbf{y}}$ and

$$\boldsymbol{\mu}^{(0,1)} = \mathbf{y} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{y}. \tag{31}$$

For the iteration over $\mathbf{y}$ convergence is immediately attained since substitution of $\boldsymbol{\mu}^{(0,1)}$ into $\mathbf{y}$ in equation (31) gives

$$\begin{aligned}
& \boldsymbol{\mu}^{(0,1)} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}'_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A}\boldsymbol{\mu}^{(0,1)} \\
= \; & \mathbf{y} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{y} - \\
& \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}'_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A} \left[\mathbf{y} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{y}\right] \\
= \; & \mathbf{y} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\right)' \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{y} \\
= \; & \boldsymbol{\mu}^{(0,1)}.
\end{aligned}$$

The process simplifies to iteration only over $\boldsymbol{\mu}$ with $\mathbf{y}$ remaining constant.
At step $i+1 \quad (i = 0, 1, 2, \ldots)$ the approximation of $\widehat{\boldsymbol{\mu}}_c$ is given by

$$\boldsymbol{\mu}^{(i+1)} = \mathbf{y} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\mu}^{(i)}}\right)' \left(\mathbf{A}\mathbf{V}'_{\boldsymbol{\mu}^{(i)}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{y}$$

with $\boldsymbol{\mu}^{(0)} = \mathbf{y}$. The process converges to the ML estimate $\widehat{\boldsymbol{\mu}}_c$.

- Let $\mathbf{D}_{\mu}$ be a diagonal matrix with the elements of $\boldsymbol{\mu}' = (\mu_1, \mu_2, \ldots, \mu_p)$ on the principal diagonal and $\mathbf{V} = \mathbf{D}_{\mu}$. Suppose $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{A}\log(\boldsymbol{\mu})$. Then

$$\begin{aligned}
\mathbf{G}_{\mu} &= \frac{\partial}{\partial \boldsymbol{\mu}} \mathbf{A}\log(\boldsymbol{\mu}) = \mathbf{A}\mathbf{D}_{\mu}^{-1} \\
\mathbf{G}_{\mathbf{y}} &= \mathbf{A}\mathbf{D}_{\mathbf{y}}^{-1}
\end{aligned}$$

and

$$\begin{aligned}
\widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - \left(\mathbf{G}_{\mu}\mathbf{V}_{\mu}\right)' \left(\mathbf{G}_{\mathbf{y}}\mathbf{V}_{\mu}\mathbf{G}'_{\mu}\right)^{-1} \mathbf{A}\log(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right) \\
&= \mathbf{y} - \left(\mathbf{A}\mathbf{D}_{\mu}^{-1}\mathbf{D}_{\mu}\right)' \left(\mathbf{A}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{D}_{\mu}\mathbf{D}_{\mu}^{-1}\mathbf{A}'\right)^{-1} \mathbf{A}\log(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right) \\
&= \mathbf{y} - \mathbf{A}'\left(\mathbf{A}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{A}'\right)^{-1} \mathbf{A}\log(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right).
\end{aligned}$$

Iteration is only over $\mathbf{y}$. At step $j+1 \quad (j = 0, 1, 2, \ldots)$ the approximation of $\widehat{\boldsymbol{\mu}}_c$ is given by

$$\boldsymbol{\mu}^{(j+1)} = \boldsymbol{\mu}^{(j)} - \mathbf{A}'\left(\mathbf{A}\mathbf{D}_{\boldsymbol{\mu}^{(j)}}^{-1}\mathbf{A}'\right)^{-1} \mathbf{A}\log\left(\boldsymbol{\mu}^{(j)}\right)$$

with $\boldsymbol{\mu}^{(0)} = \mathbf{y}$. The process converges to the ML estimate $\widehat{\boldsymbol{\mu}}_c$.

Proposition 2
The asymptotic covariance matrix of $\widehat{\boldsymbol{\mu}}_c$ is given by

$$\boldsymbol{\Sigma}_c = \mathbf{V}_{\mu} - \left(\mathbf{G}_{\mu}\mathbf{V}_{\mu}\right)' \left(\mathbf{G}_{\mu}\mathbf{V}_{\mu}\mathbf{G}'_{\mu}\right)^{-1} \mathbf{G}_{\mu}\mathbf{V}_{\mu}.$$

with the MLE obtained by replacing $\boldsymbol{\mu}$ with $\widehat{\boldsymbol{\mu}}_c$.

EXAMPLE 2.9

Determining ML estimates under constraints with iteration over $\mathbf{y}$ *and* $\boldsymbol{\mu}$.

The number of accidents per thousand per age group in a certain factory is given in Table 2.6.

TABLE 2.6: Accidents per 1000 per age group.

Age group	I	II	III
Number of accidents	80	15	5

The model under consideration is $\mu_i = \alpha\gamma^{i-1}$ for $i = 1, 2, 3$, and independent Poisson sampling is assumed.
This model implies the constraint

$$\mathbf{g}(\boldsymbol{\mu}) = \mu_1\mu_3 - \mu_2^2 = 0.$$

In this case

$\mathbf{V}_\mu = \mathbf{D}_\mu$

$\mathbf{G}_\mu = (\ \mu_3,\ \ -2\mu_2,\ \ \mu_1\ )$

$\mathbf{G}_\mathbf{y} = (\ y_3,\ \ -2y_2,\ \ y_1\ )$

$\mathbf{G}_\mu\mathbf{D}_\mu = (\ \mu_1\mu_3,\ \ -2\mu_2^2,\ \ \mu_1\mu_3\ )$

$\mathbf{G}_\mathbf{y}\mathbf{D}_\mu\mathbf{G}'_\mu = (y_1 + y_3)\,\mu_1\mu_3 + 4y_2\mu_2^2.$

The ML estimate of $\boldsymbol{\mu}$ is found iteratively from

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - (\mathbf{G}_\mu\mathbf{D}_\mu)'\left(\mathbf{G}_\mathbf{y}\mathbf{D}_\mu\mathbf{G}'_\mu\right)^{-1}\mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|). \tag{32}$$

Iteration is over $\mathbf{y}$ and $\boldsymbol{\mu}$. The process converges after eight steps.

Table 2.7 gives the approximation of $\widehat{\boldsymbol{\mu}}_c$ at different steps of the iterative procedure.These are the same results as obtained by the Newton-Raphson and Fisher scoring algorithms (see Examples 2.1, 2.2 and 2.3).

TABLE 2.7: Approximation of $\widehat{\boldsymbol{\mu}}_c$ at different steps of the iterative procedure.

i	$\mu_1^{(i,j)}$	$\mu_2^{(i,j)}$	$\mu_3^{(i,j)}$	j	$\mu_1^{(i,j)}$	$\mu_2^{(i,j)}$	$\mu_3^{(i,j)}$
0	80	15	5	0	80	15	5
				1	78.526316	16.657895	3.5263158
				2	78.531142	16.652465	3.5311418
1	78.531142	78.531142	3.5311418	0	80	15	5
				1	78.793103	17.413793	3.7931034
				2	78.821807	17.356387	3.8218065
				3	78.821823	17.356354	3.8218228
2	78.821823	17.356354	3.8218228	0	80	15	5
				1	78.793103	17.413793	3.7931034
				2	78.821807	17.356387	3.8218065
				3	78.821823	17.356354	3.8218228

Description of the procedure:

- Both $\mathbf{y}$ and $\boldsymbol{\mu}$ in equation (32) are initially estimated by the observed data, that is $\mathbf{y} = \boldsymbol{\mu}^{(0,0)}$. The first approximation of $\widehat{\boldsymbol{\mu}}_c$ is given by

$$\boldsymbol{\mu}^{(0,1)} = \mathbf{y} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{D}_{\boldsymbol{\mu}^{(0,0)}}\right)'\left(\mathbf{G}_\mathbf{y}\mathbf{D}_{\boldsymbol{\mu}^{(0,0)}}\mathbf{G}'_{\boldsymbol{\mu}^{(0,0)}}\right)^{-1}\mathbf{g}(\mathbf{y}).$$

  The process iterates over $\mathbf{y}$ until convergence is attained at $(i, j) = (0, 2)$. At this stage the approximation of $\widehat{\boldsymbol{\mu}}_c$ is

$$\boldsymbol{\mu}^{(0,2)} = \begin{pmatrix} 78.531142 \\ 16.652465 \\ 3.5311418 \end{pmatrix}$$

  This becomes the next estimate of $\boldsymbol{\mu}$ in $\mathbf{G}_\mu$ and $\mathbf{D}_\mu$.

- The process again iterates over $\mathbf{y}$ with the initial value of $\mathbf{y} = \begin{pmatrix} 80 \\ 15 \\ 5 \end{pmatrix}$, the vector of observed data.
  For $(i, j) = (1, 0)$

$$\boldsymbol{\mu}^{(1,0)} = \mathbf{y} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,2)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,2)}}\right)' \left(\mathbf{G}_{\mathbf{y}} \mathbf{V}_{\boldsymbol{\mu}^{(0,2)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,2)}}\right)^{-1} \mathbf{g}(\mathbf{y})$$

  and for $(i, j) = (1, 1)$

$$\boldsymbol{\mu}^{(1,1)} = \boldsymbol{\mu}^{(1,0)} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(0,2)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,2)}}\right)' \left(\mathbf{G}_{\boldsymbol{\mu}^{(1,0)}} \mathbf{V}_{\boldsymbol{\mu}^{(0,2)}} \mathbf{G}'_{\boldsymbol{\mu}^{(0,2)}}\right)^{-1} \mathbf{g}\left(\boldsymbol{\mu}^{(1,0)}\right).$$

  Convergence is attained at $(i, j) = (1, 3)$. The vector $\boldsymbol{\mu}^{(1,3)}$ becomes the next estimate of $\boldsymbol{\mu}$ in $\mathbf{G}_{\boldsymbol{\mu}}$ and $\mathbf{D}_{\boldsymbol{\mu}}$.

- The process again iterates over $\mathbf{y}$ with the initial value of $\mathbf{y}$ the vector of observed data.This iteration over $\mathbf{y}$ converges at $(i, j) = (2, 3)$ and at this stage

$$\boldsymbol{\mu}^{(2,3)} = \boldsymbol{\mu}^{(2,2)} - \left(\mathbf{G}_{\boldsymbol{\mu}^{(1,3)}} \mathbf{V}_{\boldsymbol{\mu}^{(1,3)}}\right)' \left(\mathbf{G}_{\boldsymbol{\mu}^{(2,2)}} \mathbf{V}_{\boldsymbol{\mu}^{(1,3)}} \mathbf{G}'_{\boldsymbol{\mu}^{(1,3)}}\right)^{-1} \mathbf{g}\left(\boldsymbol{\mu}^{(2,2)}\right).$$

  Since $\boldsymbol{\mu}^{(2,3)} = \boldsymbol{\mu}^{(1,3)}$ convergence over $\boldsymbol{\mu}$ is also attained at this step and the process stops.

The program is given in the Appendix.

EXAMPLE 2.10

Determining ML estimates under constraints with iteration over $\mathbf{y}$.

Consider the same data as in Example 2.9 but using the constraint

$$\mathbf{g}(\boldsymbol{\mu}) = \log(\mu_1\mu_3) - 2\log(\mu_2) = 0$$

In this case

$\mathbf{V} = \mathbf{D}_\mu$

$\mathbf{G}_\mu = \left(\frac{1}{\mu_1}, \quad \frac{-2}{\mu_2}, \quad \frac{1}{\mu_3} \right)$

$\mathbf{G}_\mathbf{y} = \left(\frac{1}{y_1}, \quad \frac{-2}{y_2}, \quad \frac{1}{y_3} \right)$

$\mathbf{G}_\mu\mathbf{D}_\mu = \left(1, \quad -2, \quad 1 \right)$

$\mathbf{G}_\mathbf{y}\mathbf{D}_\mu\mathbf{G}'_\mu = \frac{1}{y_1} + \frac{4}{y_2} + \frac{1}{y_3}.$

The ML estimate of $\boldsymbol{\mu}$ is found iteratively from

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - (\mathbf{G}_\mu\mathbf{D}_\mu)' \left(\mathbf{G}_\mathbf{y}\mathbf{D}_\mu\mathbf{G}'_\mu\right)^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|) \\ &= \mathbf{y} - \begin{pmatrix} 1 \\ -2 \\ 1 \end{pmatrix} \frac{\log(y_1y_3) - 2\log(y_2)}{\frac{1}{y_1} + \frac{4}{y_2} + \frac{1}{y_3}} + o(\|\mathbf{y} - \boldsymbol{\mu}\|). \end{aligned}$$

Iteration is only over $\mathbf{y}$.

Table 2.8 gives the estimates of $\widehat{\boldsymbol{\mu}}_c$ at different steps of the iterative procedure.

TABLE 2.8: Approximation of $\widehat{\boldsymbol{\mu}}_c$ at different steps of the iterative procedure.

	Approximation of $\widehat{\boldsymbol{\mu}}_c$ by $\boldsymbol{\mu}^{(r)}$		
r	$\mu_1^{(r)}$	$\mu_2^{(r)}$	$\mu_3^{(r)}$
0	80	15	5
1	78.79924	17.40152	3.79924
2	78.821801	17.356397	3.8218013
3	78.821823	17.356354	3.8218228

Alternatively, the constraint can also be set up in terms of the GLM given in Example 1.2. The model is

$$\log\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$$

with $\boldsymbol{\beta}' = (\beta_1, \beta_2)$ where $\beta_1 = \log\alpha$ and $\beta_2 = \log\gamma$, and $\mathbf{X}$ the design matrix given in Example 1.2. Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})\mathbf{X}'$. The model can be written in terms of the implied constraints as

$$\mathbf{g}(\boldsymbol{\mu}) = [\mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})\mathbf{X}']\log\boldsymbol{\mu} = \mathbf{P}\log\boldsymbol{\mu} = \mathbf{0}.$$

The ML estimate for $\boldsymbol{\mu}$ subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$ is found iteratively from

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - (\mathbf{G}_\mu\mathbf{V}_\mu)' \left(\mathbf{G}_\mathbf{y}\mathbf{V}_\mu\mathbf{G}'_\mu\right)^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|)$$

with $\mathbf{V}_\mu = \mathbf{D}_\mu$

$\mathbf{G}_\mu = \mathbf{P}\mathbf{D}_\mu^{-1}$

$\mathbf{G}_\mathbf{y} = \mathbf{P}\mathbf{D}_\mathbf{y}^{-1}$

$\mathbf{G}_\mu\mathbf{V} = \mathbf{P}$

$\mathbf{G}_\mathbf{y}\mathbf{V}\mathbf{G}'_\mu = \mathbf{P}\mathbf{D}_\mathbf{y}^{-1}\mathbf{P}.$

Hence, the estimation procedure is

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{P}\left(\mathbf{P}\mathbf{D}_\mathbf{y}^{-1}\mathbf{P}\right)^{-1}\mathbf{P}\log\mathbf{y} + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

Iteration is only over $\mathbf{y}$. The estimates of $\widehat{\boldsymbol{\mu}}_c$ at different steps of the iterative procedure is exactly the same as given in Table 2.8. The programs with these two restrictions are given in the Appendix.

EXAMPLE 2.11

Determination of maximum likelihood estimates under constraints. An example for incomplete data.

Example 2.8 gives data in which 197 animals are distributed multinomially into five categories.
The complete data, $\mathbf{Y}' = (Y_1, Y_2, Y_3, Y_4, Y_5)$, are the counts for each category and the cell probabilities in this model are given as

$$\boldsymbol{\pi}' = \left(\tfrac{1}{2}, \tfrac{1}{4}p, \tfrac{1}{4}(1-p), \tfrac{1}{4}(1-p), \tfrac{1}{4}p\right) \text{ for some } 0 \leq p \leq 1.$$

The random vector of complete data is $\mathbf{Y}'=(Y_1, Y_2, Y_3, Y_4, Y_5)$ and the random vector of observed data is $\mathbf{Y}'_{obs} = (Y_1 + Y_2, Y_3, Y_4, Y_5)$ where only the sum of Y_1 and Y_2 is observed. The observed data is $\mathbf{y}'_{obs} = (125, 18, 20, 34)$.
The distributions of $\mathbf{Y}$ and $\mathbf{Y}_{obs}$ are both multinomial and can be written as

$$\mathbf{Y} \sim Mult(n, \boldsymbol{\pi})$$

with

$$\begin{aligned} \boldsymbol{\pi}' &= (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5) \\ &= \left(\tfrac{1}{2}, \tfrac{1}{4}p, \tfrac{1}{4}(1-p), \tfrac{1}{4}(1-p), \tfrac{1}{4}p\right) \text{ for some } 0 \leq p \leq 1 \end{aligned}$$

and

$$\mathbf{Y}_{obs} \sim Mult(n, \boldsymbol{\pi}_{obs})$$

with

$$\begin{aligned} \boldsymbol{\pi}'_{obs} &= (\pi_1 + \pi_2, \pi_3, \pi_4, \pi_5) \\ &= \left(\tfrac{1}{2} + \tfrac{1}{4}p, \tfrac{1}{4}(1-p), \tfrac{1}{4}(1-p), \tfrac{1}{4}p\right) \text{ for some } 0 \leq p \leq 1. \end{aligned} \tag{33}$$

The ML estimate of p must be obtained from the observed data, $\mathbf{Y}_{obs}$. For the multinomial distribution

$$E(\mathbf{Y}_{obs}) = n\boldsymbol{\pi}_{obs} = \boldsymbol{\mu}_{obs}.$$

From the cell probabilities given in (33) the constraint $\mathbf{g}(\boldsymbol{\mu}_{obs}) = \mathbf{0}$ can be written as

$$\mathbf{g}(\boldsymbol{\mu}_{obs}) = \mathbf{X}\boldsymbol{\mu}_{obs} = \begin{pmatrix} 1 & -1 & -1 & -3 \\ 0 & 1 & -1 & 0 \end{pmatrix} \boldsymbol{\mu}_{obs}$$

where $\boldsymbol{\mu}'$ is the vector of expected cell counts.
The ML estimate, $\widehat{\boldsymbol{\mu}}_{obs,c}$, of the expected cell counts $\boldsymbol{\mu}_{obs}$ are obtained by solving

$$\widehat{\boldsymbol{\mu}}_{obs,c} = \mathbf{y}_{obs} - \left(\mathbf{G}_{\boldsymbol{\mu}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\right)' \left(\mathbf{G}_{\mathbf{y}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\mathbf{G}'_{\boldsymbol{\mu}_{obs}}\right)^{-1} \mathbf{g}(\mathbf{y}_{obs}) + o(\|\mathbf{y}_{obs} - \boldsymbol{\mu}_{obs}\|)$$

where $\mathbf{V}_{\boldsymbol{\mu}_{obs}} = \text{Diag}(\mathbf{y}_{obs}) - \frac{1}{n}\mathbf{y}_{obs}\mathbf{y}'_{obs}$
$\mathbf{G}_{\boldsymbol{\mu}_{obs}} = \mathbf{X} = \mathbf{G}_{\mathbf{y}_{obs}}$
$\mathbf{g}(\mathbf{y}_{obs}) = \mathbf{X}\mathbf{y}_{obs}$.
Since $\mathbf{g}(\boldsymbol{\mu}_{obs})$ is a linear function of $\boldsymbol{\mu}_{obs}$ iteration is only over $\boldsymbol{\mu}_{obs}$.
The ML estimate of p is then determined from $\widehat{\boldsymbol{\mu}}_{obs,c}$ by

$$\widehat{p} = 4\frac{\widehat{\mu}_{obs,4}}{n}.$$

The process converges after 4 steps and $\widehat{\boldsymbol{\mu}}_{obs,c} = \begin{pmatrix} 129.37096 \\ 18.379041 \\ 18.379041 \\ 30.870959 \end{pmatrix}$ giving $\widehat{p} = 0.6268215$. This is the same result as obtained with the EM algorithm in Example 2.8.
The program is given in the Appendix.

3 CATEGORICAL DATA ANALYSIS

Maximum likelihood estimation procedures for loglinear and logistic regression models are discussed in this chapter.

3.1 LOGLINEAR ANALYSIS

3.1.1 The Model

Consider a completely classified contingency table and arrange the observed frequencies into a vector $\mathbf{y}'=(y_1, y_2, y_3, \ldots, y_p)$. The expected cell frequencies are given by $\boldsymbol{\mu}'=(\mu_1, \mu_2, \mu_3, \ldots, \mu_p)$. A Poisson sampling scheme is assumed.
For independent Poisson sampling the joint probability function of Y_i, $i = 1, 2, \ldots, p$ is

$$\begin{aligned} f_{\mathbf{Y}}(\mathbf{y}|\boldsymbol{\mu}) &= \prod_{i=1}^{p} \frac{\exp^{-\mu_i}\mu_i^{y_i}}{y_i!} \\ &= \exp\left[\sum y_i \log \mu_i - \sum \mu_i\right] \exp\left[-\sum \log y_i!\right] \end{aligned} \tag{34}$$

which is a member of the exponential family since it has the form

$$p(\mathbf{y}, \boldsymbol{\theta}) = b(\mathbf{y}) \exp\left[\mathbf{y}'\boldsymbol{\theta} - \kappa(\boldsymbol{\theta})\right]$$

with $b(\mathbf{y}) = \exp\left[-\sum \log y_i!\right]$
$\boldsymbol{\theta}$ a 4×1 vector of natural parameters with $\theta_i = \log \mu_i$, that is $\mu_i = \exp(\theta_i)$
$\kappa(\boldsymbol{\theta}) = \sum \mu_i = \sum \exp(\theta_i)$.

The expected value of Y_i is

$$\begin{aligned} E(Y_i) &= \frac{\partial}{\partial \theta_i}\kappa(\boldsymbol{\theta}) \\ &= e^{\theta_i} \\ &= \mu_i \end{aligned}$$

and the covariance of Y_i, Y_j is

$$\begin{aligned} \mathrm{Cov}(Y_i, Y_j) &= \frac{\partial^2}{\partial \theta_i \partial \theta_j}\kappa(\boldsymbol{\theta}) \\ &= \begin{cases} e^{\theta_i} & \text{if } i = j \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

Thus $E(\mathbf{Y}) = \boldsymbol{\mu}$ and $\mathrm{Cov}(\mathbf{Y}) = \mathrm{Diag}(\boldsymbol{\mu})$.

In the case of a 2×2 contingency table with two categorical variables A and B, the model to be fitted, written as a loglinear model is

$$\begin{aligned} \log \mu_1 &= \alpha + \lambda_1^A + \lambda_1^B + \lambda_{11}^{AB} \\ \log \mu_2 &= \alpha + \lambda_1^A - \lambda_1^B - \lambda_{11}^{AB} \\ \log \mu_3 &= \alpha - \lambda_1^A + \lambda_1^B - \lambda_{11}^{AB} \\ \log \mu_4 &= \alpha - \lambda_1^A - \lambda_1^B + \lambda_{11}^{AB} \end{aligned}$$

The generalized linear model is

$$\log \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}.$$

The three components of the GLM are:

1. The random component $\mathbf{Y}$.

2. The systematic component

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ \lambda_1^A \\ \lambda_1^B \\ \lambda_{11}^{AB} \end{pmatrix}$$

where $\mathbf{X}$ is the design matrix and $\boldsymbol{\beta}' = \left(\alpha, \lambda_1^A, \lambda_1^B, \lambda_{11}^{AB}\right)$ the vector with model parameters.

3. The link function is also a canonical link and is given by

$$\eta_i = h(\mu_i) = \log \mu_i = \theta_i = \sum_j \beta_j x_{ij}. \tag{35}$$

3.1.2 Newton-Raphson algorithm for ML estimation

From equation (34) the log-likelihood function for independent Poisson sampling is

$$L(\boldsymbol{\mu}|\mathbf{y}) = \sum_i y_i \log \mu_i - \sum_i \mu_i - \sum_i \log y_i!. \tag{36}$$

In equation (35) $\log \mu_i$ was written as $\log \mu_i = \sum_j \beta_j x_{ij}$. By substituting $\mu_i = \exp\left(\sum_j \beta_j x_{ij}\right)$ into the log-likelihood function in (36), the log-likelihood can be written as a function of the elements of $\boldsymbol{\beta}$. That is

$$L(\boldsymbol{\beta}|\mathbf{y}) = \sum_i y_i \sum_j \beta_j x_{ij} - \sum_i \exp\left(\sum_j \beta_j x_{ij}\right) - \sum_i \log y_i!. \tag{37}$$

The value of $\widehat{\boldsymbol{\beta}}$ that maximizes $L(\boldsymbol{\beta}|\mathbf{y})$ can be found iteratively with the Newton-Raphson algorithm

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} - \left(\mathbf{H}^{(r)}\right)^{-1} \mathbf{q}^{(r)} \tag{38}$$

where $\boldsymbol{\beta}^{(r)}$ is the rth approximation of $\widehat{\boldsymbol{\beta}}$, $r = 0, 1, 2, \ldots$ and $\mathbf{q}^{(r)}$ and $\mathbf{H}^{(r)}$ are $\mathbf{q}$ and $\mathbf{H}$ evaluated at $\boldsymbol{\beta}^{(r)}$. From Section 2.1, $\mathbf{q}$ is the vector with elements the first order partial derivatives

$$q_k = \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_k} = -\sum_i x_{ik} \exp\left(\sum_j \beta_j x_{ij}\right) + \sum_i y_i x_{ik}$$

and $\mathbf{H}$ is the matrix of second order partial derivatives having elements

$$h_{hk} = \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_h \partial \beta_k} = -\sum_i x_{ih} x_{ik} \exp\left(\sum_j \beta_j x_{ij}\right) = -\sum_i x_{ih} x_{ik} \mu_i.$$

Hence,

$$\mathbf{q}^{(r)} = \mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right) \tag{39}$$

$$\mathbf{H}^{(r)} = -\mathbf{X}' \text{diag}\left(\boldsymbol{\mu}^{(r)}\right) \mathbf{X} \tag{40}$$

with $\boldsymbol{\mu}^{(r)} = \exp\left(\mathbf{X}\boldsymbol{\beta}^{(r)}\right)$ the rth approximation of $\widehat{\boldsymbol{\mu}}$, $(r = 0, 1, 2, \ldots)$.
Substituting equations (39) and (40) into equation (38) gives

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left[\mathbf{X}' \text{diag}\left(\boldsymbol{\mu}^{(r)}\right) \mathbf{X}\right]^{-1} \mathbf{X}'\left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right). \tag{41}$$

The algorithm requires an initial guess, $\boldsymbol{\beta}^{(0)}$, for the values that maximizes the function $L(\boldsymbol{\beta}|\mathbf{y})$. The ML estimates of the parameters in the saturated model are used as the initial estimates and are given by

$$\boldsymbol{\beta}^{(0)} = (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}' \log \mathbf{y}.$$

The asymptotic covariance matrix of $\widehat{\boldsymbol{\beta}}$ is

$$\text{Cov}\left(\widehat{\boldsymbol{\beta}}\right) = \left[\mathbf{X}'\text{diag}\left(\widehat{\boldsymbol{\mu}}\right)\mathbf{X}\right]^{-1} = -\widehat{\mathbf{H}}^{-1}.$$

A canonical link function was used in the GLM in which case the observed and expected second derivative matrices are identical. Hence, the Fisher scoring and Newton-Raphson algorithms are identical.

3.1.3 Maximum likelihood estimation under constraints

This procedure is also discussed by Crowther and Matthews (1995).
The saturated loglinear model can be written as

$$\log \boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta} \tag{42}$$

where $\boldsymbol{\mu}' = \left(\mu_1, \mu_2, \mu_3, \ldots, \mu_p\right)$ is the vector with expected cell frequencies for the model, $\mathbf{X} : p \times p$ is the design matrix and $\boldsymbol{\beta} : p \times 1$ is the vector of parameters for the saturated loglinear model. The ML estimate of $\boldsymbol{\beta}$ for the saturated model is

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\log \mathbf{y}.$$

For a lower order model certain elements of $\boldsymbol{\beta}$ will be equal to zero.
Let $\mathbf{C}$ be a matrix specifying the elements of $\boldsymbol{\beta}$ which are set equal to zero. The hypothesis that certain elements of $\boldsymbol{\beta}$ are zero, can be written as the constraint

$$\begin{aligned} \mathbf{g}\left(\boldsymbol{\mu}\right) &= \mathbf{C}\boldsymbol{\beta} \\ &= \mathbf{C}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\log \boldsymbol{\mu} \\ &= \mathbf{A}_C'\log \boldsymbol{\mu} \\ &= \mathbf{0}. \end{aligned} \tag{43}$$

The ML estimate of $\boldsymbol{\mu}$ subject to the constraint $\mathbf{g}\left(\boldsymbol{\mu}\right) = \mathbf{A}_C'\log \boldsymbol{\mu} = \mathbf{0}$ is given by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \left(\mathbf{G}_\mu \mathbf{V}_\mu\right)'\left(\mathbf{G}_\mathbf{y}\mathbf{V}_\mu\mathbf{G}_\mu'\right)^{-1}\mathbf{g}\left(\mathbf{y}\right) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right)$$

where $\mathbf{G}_\mu = \dfrac{\partial}{\partial \boldsymbol{\mu}}\mathbf{g}\left(\boldsymbol{\mu}\right) = \mathbf{A}_C'\mathbf{D}_\mu^{-1}$ and $\mathbf{V}_\mu = \mathbf{D}_\mu$.
Thus

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - \left(\mathbf{A}_C'\mathbf{D}_\mu^{-1}\mathbf{D}_\mu\right)'\left(\mathbf{A}_C'\mathbf{D}_\mathbf{y}^{-1}\mathbf{D}_\mu\mathbf{D}_\mu^{-1}\mathbf{A}_C\right)^{-1}\mathbf{g}\left(\mathbf{y}\right) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right) \\ &= \mathbf{y} - \mathbf{A}_C\left(\mathbf{A}_C'\mathbf{D}_\mathbf{y}^{-1}\mathbf{A}_C\right)^{-1}\mathbf{g}\left(\mathbf{y}\right) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right). \end{aligned} \tag{44}$$

The ML estimate for $\widehat{\boldsymbol{\mu}}_c$ is obtained by iterating over $\mathbf{y}$ and the asymptotic covariance matrix of $\widehat{\boldsymbol{\mu}}_c$ is

$$\widehat{\boldsymbol{\Sigma}}_c = \mathbf{D}_{\widehat{\boldsymbol{\mu}}_c} - \mathbf{A}_C\left(\mathbf{A}_C'\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\mathbf{A}_C\right)^{-1}\mathbf{A}_C'.$$

The ML estimate for the vector of cell probabilities is

$$\widehat{\mathbf{p}}_c = \frac{\widehat{\boldsymbol{\mu}}_c}{n}$$

where n is the number of observations.
The ML estimates for the parameters in the loglinear model are given by

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\log \widehat{\boldsymbol{\mu}}_c.$$

The covariance matrix $\widehat{\boldsymbol{\beta}}$ is

$$\text{Cov}\left(\widehat{\boldsymbol{\beta}}\right) = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\text{Cov}\left[\log \widehat{\boldsymbol{\mu}}_c\right]\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}.$$

The "delta method" is used to determine the asymptotic covariance matrix

$$\begin{aligned} est\left[\text{Cov}\left(\log\widehat{\boldsymbol{\mu}}_c\right)\right] &= \left(\frac{\partial \log\widehat{\boldsymbol{\mu}}_c}{\partial\widehat{\boldsymbol{\mu}}_c}\right)\widehat{\boldsymbol{\Sigma}}_c\left(\frac{\partial \log\widehat{\boldsymbol{\mu}}_c}{\partial\widehat{\boldsymbol{\mu}}_c}\right)' \\ &= \mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\widehat{\boldsymbol{\Sigma}}_c\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}. \end{aligned}$$

Hence, the estimated covariance matrix for $\widehat{\boldsymbol{\beta}}$ is

$$est\left[\text{Cov}\left(\widehat{\boldsymbol{\beta}}\right)\right] = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\left[\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\widehat{\boldsymbol{\Sigma}}_c\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\right]\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}.$$

EXAMPLE 3.1

Maximum likelihood estimation for a loglinear model.

Pugh (1983) designed a study to examine the disposition of jurors to base their judgments of defendants ("guilty" or "not guilty") on the alleged behavior of a rape victim. Pugh's study varied the degree to which the juror could assign fault to the victim ("low" or "high") and the presentation of the victim as someone with "high moral character", "low moral character" or "neutral". The data are given in Table 3.1.

TABLE 3.1: Data from Pugh (1983).

		Moral (M)		
Verdict (V)	Fault (F)	High	Neutral	Low
Guilty	Low	42	79	32
	High	23	65	17
Not Guilty	Low	4	12	8
	High	11	41	24

The saturated model, $\log\left(\mu_{ijk}\right) = \alpha + \lambda_i^M + \lambda_j^V + \lambda_k^F + \lambda_{ij}^{MV} + \lambda_{ik}^{MF} + \lambda_{jk}^{VF} + \lambda_{ijk}^{MVF}$, can be written as

$$\log\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$$

$$= \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & -1 & 1 & 0 & -1 & 0 & -1 & -1 & 0 \\ 1 & 1 & 0 & -1 & 1 & -1 & 0 & 1 & 0 & -1 & -1 & 0 \\ 1 & 1 & 0 & -1 & -1 & -1 & 0 & -1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & -1 & 0 & 1 & 0 & -1 & -1 & 0 & -1 \\ 1 & 0 & 1 & -1 & 1 & 0 & -1 & 0 & 1 & -1 & 0 & -1 \\ 1 & 0 & 1 & -1 & -1 & 0 & -1 & 0 & -1 & 1 & 0 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \lambda_1^M \\ \lambda_2^M \\ \lambda_1^V \\ \lambda_1^F \\ \lambda_{11}^{MV} \\ \lambda_{21}^{MV} \\ \lambda_{11}^{MF} \\ \lambda_{21}^{MF} \\ \lambda_{11}^{VF} \\ \lambda_{111}^{MVF} \\ \lambda_{211}^{MVF} \end{pmatrix}$$

Consider in this example the reduced model $\log(\mu_{ijk}) = \alpha + \lambda_i^M + \lambda_j^V + \lambda_k^F + \lambda_{ij}^{MV} + \lambda_{jk}^{VF}$ which contains only the interaction terms between Verdict and Fault and between Verdict and Moral.

Results from the Proc Catmod procedure in SAS

The program and output obtained from the PROC CATMOD procedure in SAS are given in the Appendix. The results are summarized in Table 3.2.

TABLE 3.2: Results from SAS: Proc Catmod.

Maximum Likelihood Estimates		
Variable	Par Estimate	Standard Error
λ_1^M	−0.4221	0.1062
λ_2^M	0.6067	0.0811
λ_1^V	0.5520	0.0734
λ_1^F	−0.1941	0.0666
λ_{11}^{MV}	0.2512	0.1062
λ_{21}^{MV}	0.0178	0.0811
λ_{11}^{VF}	0.3823	0.0666

Model Fitting Information	
Likelihood Ratio	2.81
Pearson Chi-Square	2.80

Obtaining the ML estimates by using the Newton-Raphson algorithm

The ML estimates are obtained iteratively with equation (41),

$$\boldsymbol{\beta}_u^{(r+1)} = \boldsymbol{\beta}_u^{(r)} + \left[\mathbf{X}_u' \text{diag}\left(\boldsymbol{\mu}^{(r)}\right) \mathbf{X}_u\right]^{-1} \mathbf{X}_u' \left(\mathbf{y} - \boldsymbol{\mu}^{(r)}\right)$$

where the matrix $\mathbf{X}_u$ is a submatrix of the design matrix, $\mathbf{X}$, of the saturated model and $\boldsymbol{\beta}_u$ is the parameter vector of the reduced model. The model is

$$\log \boldsymbol{\mu} = \mathbf{X}_u \boldsymbol{\beta}_u = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & -1 & 1 & 0 & -1 \\ 1 & 1 & 0 & -1 & 1 & -1 & 0 & -1 \\ 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & -1 & 0 & 1 & -1 \\ 1 & 0 & 1 & -1 & 1 & 0 & -1 & -1 \\ 1 & 0 & 1 & -1 & -1 & 0 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ \lambda_1^M \\ \lambda_2^M \\ \lambda_1^V \\ \lambda_1^F \\ \lambda_{11}^{MV} \\ \lambda_{21}^{MV} \\ \lambda_{11}^{VF} \end{pmatrix}.$$

The ML estimates of the parameters for the saturated model are used as an initial guess of $\widehat{\boldsymbol{\beta}}_u$ and are given by

$$\boldsymbol{\beta}_u^{(0)} = \left(\mathbf{X}_u' \mathbf{X}_u\right)^{-1} \mathbf{X}_u' \log \mathbf{y}.$$

The covariance matrix of $\widehat{\boldsymbol{\beta}}_u$ is

$$\text{Cov}\left(\widehat{\boldsymbol{\beta}}_u\right) = \left[\mathbf{X}_u' \text{diag}\left(\widehat{\boldsymbol{\mu}}\right) \mathbf{X}_u\right]^{-1}.$$

The results obtained are the same as in Table 3.2. The program is given in the Appendix.

Obtaining the ML estimates under constraints

For the model $\log(\mu_{ijk}) = \alpha + \lambda_i^M + \lambda_j^V + \lambda_k^F + \lambda_{ij}^{MV} + \lambda_{jk}^{VF}$, the ML estimate of $\boldsymbol{\mu}$ subject to the constraint

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{C}\boldsymbol{\beta} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{\beta} = \mathbf{0}$$

can be determined iteratively with equation (44),

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{A}_C \left(\mathbf{A}_C' \mathbf{D}_{\mathbf{y}}^{-1} \mathbf{A}_C\right)^{-1} \mathbf{g}(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|)$$

where $\mathbf{A}_C' = \mathbf{C}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$.

Furthermore

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \log \widehat{\boldsymbol{\mu}}_c$$

and

$$est\left[\mathrm{Cov}\left(\widehat{\boldsymbol{\beta}}\right)\right] = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\left[\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\widehat{\boldsymbol{\Sigma}}_c\mathbf{D}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\right]\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}.$$

The Wald statistic is 2.79 and the other results obtained are the same as in Table 3.2. The program is given in the Appendix.

3.2 LOGISTIC REGRESSION

3.2.1 The Model

Let Y_i, $i = 1, 2, \ldots, p$ be independent random variables with $Y_i \sim bi(n_i, \pi_i)$. The frequency distribution for the p independent binomial distributions is given in Table 3.3.

TABLE 3.3: Frequency distribution of p independent binomial distributions.

	Subgroups			
	1	2	$\cdots$	p
Successes	y_1	y_2	$\cdots$	y_p
Failures	$n_1 - y_1$	$n_2 - y_2$	$\cdots$	$n_p - y_p$

Suppose that m covariates, $X_1, X_2, \ldots, X_m$, are observed and that at occasion i, $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im})$ and y_i is the number of successes in the n_i trials, $i = 1, 2, \ldots, p$. Let $\boldsymbol{\pi}' = (\pi_1, \pi_2, \ldots, \pi_p)$ be the vector with probabilities of a success within each subgroup and $\mathbf{n}' = (n_1, n_2, \ldots, n_p)$ the vector indicating the number of trials within each subgroup.
The joint probability function of $Y_1, Y_2, \ldots, Y_p$ is

$$\begin{aligned} f(\mathbf{y}|\boldsymbol{\pi}) &= \prod_{i=1}^{p} P(Y_i = y_i) \\ &= \prod_{i=1}^{p} \binom{n_i}{y_i} \pi_i^{y_i} (1-\pi_i)^{n_i - y_i} \\ &= \exp\left[\log \prod_{i=1}^{p} \binom{n_i}{y_i} + \log \prod_{i=1}^{p} \pi_i^{y_i} + \log \prod_{i=1}^{p} (1-\pi_i)^{(n_i - y_i)}\right] \\ &= \exp\left[\sum_{i=1}^{p} \log \binom{n_i}{y_i} + \sum_{i=1}^{p} y_i \log \pi_i + \sum_{i=1}^{p} (n_i - y_i) \log(1-\pi_i)\right] \\ &= \exp\left[\sum_{i=1}^{p} \log \binom{n_i}{y_i} + \sum_{i=1}^{p} y_i \log \frac{\pi_i}{1-\pi_i} + \sum_{i=1}^{p} n_i \log(1-\pi_i)\right] \end{aligned} \tag{45}$$

which is a member of the natural exp he form

$$p(\mathbf{y}, \boldsymbol{\theta}) = b(\mathbf{y}) \exp[\mathbf{y}'\boldsymbol{\theta} - \kappa(\boldsymbol{\theta})]$$

where

$b(\mathbf{y}) = \prod_{i=1}^{p} \binom{n_i}{y_i}$

$\boldsymbol{\theta}$ a $p \times 1$ vector with natural parameters $\theta_i = \log \frac{\pi_i}{1-\pi_i}$, that is $\pi_i = \frac{e^{\theta_i}}{1+e^{\theta_i}}$

$\kappa(\boldsymbol{\theta}) = -\sum_{i=1}^{p} n_i \log(1-\pi_i) = -\sum_{i=1}^{p} n_i \log\left(\frac{1}{1+e^{\theta_i}}\right) = \sum_{i=1}^{p} n_i \log\left(1+e^{\theta_i}\right).$

For the exponential class

$$\begin{aligned} E(Y_i) &= \frac{\partial}{\partial \theta_i}\kappa(\boldsymbol{\theta}) \\ &= n_i \frac{e^{\theta_i}}{1+e^{\theta_i}} \\ &= n_i \pi_i = \mu_i \end{aligned}$$

and

$$\begin{aligned} \mathrm{Cov}(Y_i, Y_j) &= \frac{\partial^2}{\partial \theta_j \partial \theta_i}\kappa(\boldsymbol{\theta}) \\ &= \begin{cases} n_i\pi_i(1-\pi_i) & \text{if } i = j \\ 0 \quad \textit{otherwise.} \end{cases} \end{aligned} \tag{46}$$

Thus, $\mathrm{E}(\mathbf{Y}) = \boldsymbol{\mu}$ and $\mathrm{Cov}(\mathbf{Y}) = \mathbf{V}_\mu$ =diag$[n_i\pi_i(1-\pi_i)]$.
The logistic regression model is written as $\ell_\mu = \mathbf{X}\boldsymbol{\beta}$ with

$$\ell_{i,\mu} = \log \frac{\pi_i}{1-\pi_i} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_m x_{im} = \sum_{j=0}^{m} \beta_j x_{ij}.$$

The three components for the GLM are:

- The random component $\mathbf{Y}$, the vector of successes.
- The systematic component which relates the linear predictor to a set of explanatory variables,

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1m} \\ 1 & x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{p1} & x_{p2} & \cdots & x_{pm} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_m \end{pmatrix}.$$

- The link function which links $\mu_i = E(Y_i)$ to η_i,

$$\eta_i = \log \frac{\pi_i}{1-\pi_i} = \log \frac{n_i\pi_i}{n_i - n_i\pi_i} = \log \frac{\mu_i}{n_i - \mu_i} = h(\mu_i).$$

The function h is a canonical link since $h(\mu_i) = \theta_i = \log \frac{\pi_i}{1-\pi_i}$.

3.2.2 Newton-Raphson algorithm for ML estimation

From equation (45) the log-likelihood function for the logistic regression model is

$$L(\boldsymbol{\pi}|\mathbf{y}) = \sum_{i=1}^{p} \log \binom{n_i}{y_i} + \sum_{i=1}^{p} y_i \log \frac{\pi_i}{1-\pi_i} + \sum_{i=1}^{p} n_i \log(1-\pi_i).$$

Since $\log\left(\frac{\pi_i}{1-\pi_i}\right) = \sum_{j=0}^{m} \beta_j x_{ij}$,
and $\log(1-\pi_i) = -\log\left[1+\exp\left(\sum_{j=0}^{m} \beta_j x_{ij}\right)\right]$
the log-likelihood function in terms of $\boldsymbol{\beta}$ is given by

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{p} \log \binom{n_i}{y_i} + \sum_{i=1}^{p} y_i \sum_{j=0}^{m} \beta_j x_{ij} - \sum_{i=1}^{p} n_i \log\left[1+\exp\left(\sum_{j=0}^{m} \beta_j x_{ij}\right)\right].$$

The value $\widehat{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$ that maximizes $L(\boldsymbol{\beta})$ can be determined with the Newton-Raphson algorithm. At step $r+1$ $(r=0,1,2,\ldots)$ in the iterative process the approximation of $\widehat{\boldsymbol{\beta}}$ is given by

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} - \left(\mathbf{H}^{(r)}\right)^{-1} \mathbf{q}^{(r)} \tag{47}$$

where $\mathbf{q}$ is the vector having elements $\frac{\partial L(\boldsymbol{\beta})}{\partial \beta_k}$, $\mathbf{H}$ is the matrix having elements $\frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_h \partial \beta_k}$, and $\mathbf{q}^{(r)}$ and $\mathbf{H}^{(r)}$ are $\mathbf{q}$ and $\mathbf{H}$ evaluated at $\boldsymbol{\beta} = \boldsymbol{\beta}^{(r)}$.
The elements of $\mathbf{q}^{(r)}$ can be written as

$$\begin{aligned} q_k^{(r)} &= \frac{\partial L(\boldsymbol{\beta})}{\partial \beta_k}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(r)}} \\ &= \sum_{i=1}^{p} y_i x_{ik} + \sum_{i=1}^{p} n_i x_{ik} \left[\frac{\exp\left(\sum_{j=0}^{m} \beta_j^{(r)} x_{ij}\right)}{1+\exp\left(\sum_{j=0}^{m} \beta_j^{(r)} x_{ij}\right)}\right] \\ &= \sum_{i=1}^{p} x_{ik}\left(y_i + n_i \pi_i^{(r)}\right) \end{aligned}$$

and the elements of $\mathbf{H}^{(r)}$ as

$$\begin{aligned} h_{hk}^{(r)} &= \frac{\partial^2 L(\boldsymbol{\beta})}{\partial \beta_h \partial \beta_k}\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(r)}} \\ &= -\sum_{i=1}^{p} x_{ih} x_{ik} n_i \frac{\exp\left(\sum_{j=0}^{m} \beta_j^{(r)} x_{ij}\right)}{\left[1+\exp\left(\sum_{j=0}^{m} \beta_j^{(r)} x_{ij}\right)\right]^2} \\ &= -\sum_{i=1}^{p} x_{ih} x_{ik} n_i \pi_i^{(r)}\left(1-\pi_i^{(r)}\right). \end{aligned}$$

Thus

$$\mathbf{q}^{(r)} = \mathbf{X}'\left(\mathbf{y} - \mathbf{n}'\boldsymbol{\pi}^{(r)}\right) \tag{48}$$

and

$$\mathbf{H}^{(r)} = -\mathbf{X}'\mathrm{Diag}\left[n_i \pi_i^{(r)}\left(1-\pi_i^{(r)}\right)\right]\mathbf{X}. \tag{49}$$

Substituting (48) and (49) into (47) gives

$$\boldsymbol{\beta}^{(r+1)} = \boldsymbol{\beta}^{(r)} + \left\{\mathbf{X}'\text{Diag}\left[n_i\pi_i^{(r)}\left(1-\pi_i^{(r)}\right)\right]\mathbf{X}\right\}^{-1}\mathbf{X}'\left(\mathbf{y}-\mathbf{n}'\boldsymbol{\pi}^{(r)}\right) \tag{50}$$

where

$$\pi_i^{(r)} = \frac{\exp\left(\sum_{j=0}^{m}\beta_j^{(r)}x_{ij}\right)}{1+\exp\left(\sum_{j=0}^{m}\beta_j^{(r)}x_{ij}\right)}. \tag{51}$$

The algorithm requires an initial guess for $\widehat{\boldsymbol{\beta}}$, which is

$$\boldsymbol{\beta}^{(0)} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\boldsymbol{\ell}$$

where $\boldsymbol{\ell}$ is calculated from the observed data and has elements $\ell_i = \log\frac{\frac{y_i}{n_i}}{1-\frac{y_i}{n_i}}$.
For $r > 0$ the iterative process proceeds by using equations (50) and (51).
The estimated asymptotic covariance matrix of $\widehat{\boldsymbol{\beta}}$ is a by-product of the Newton-Raphson algorithm,

$$\text{Cov}\left(\widehat{\boldsymbol{\beta}}\right) = \left\{\mathbf{X}'\text{Diag}\left[n_i\widehat{\pi}_i\left(1-\widehat{\pi}_i\right)\right]\mathbf{X}\right\}^{-1} = -\widehat{\mathbf{H}}^{-1} \tag{52}$$

where $\widehat{\pi}_i$ is the value of $\pi_i^{(r)}$ on convergence.
A canonical link function was used in the GLM in which case the observed and expected second derivative matrices are identical. The Fisher scoring algorithm is identical to the Newton-Raphson algorithm.

3.2.3 Maximum likelihood estimation under constraints

Maximum likelihood estimation for the logistic regression model, using constraints is discussed by Crowther and Matthews (1998).
The logistic regression model can be written as $\boldsymbol{\ell}_\mu = \mathbf{X}\boldsymbol{\beta}$ as discussed in section 3.2.1. The elements of $\boldsymbol{\ell}_\mu$ written as a function of μ_i is

$$\ell_{i,\mu} = \log\frac{\pi_i}{1-\pi_i} = \log\frac{\mu_i}{n_i-\mu_i}.$$

Let $\mathbf{P} = \mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)\mathbf{X}'$ be the projection matrix of the error space. From this the constraint for a logistic regression model as a function of $\boldsymbol{\mu}$ is

$$\mathbf{g}\left(\boldsymbol{\mu}\right) = \mathbf{P}\boldsymbol{\ell}_\mu = \mathbf{P}\mathbf{X}\boldsymbol{\beta} = \mathbf{0}.$$

The ML estimate for $\boldsymbol{\mu}$ is found iteratively with

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \left(\mathbf{G}_\mu\mathbf{V}_\mu\right)'\left(\mathbf{G}_\mathbf{y}\mathbf{V}_\mu\mathbf{G}_\mu'\right)^{-1}\mathbf{g}\left(\mathbf{y}\right) + o\left(\|\mathbf{y}-\boldsymbol{\mu}\|\right) \tag{53}$$

where $\mathbf{G}_\mu = \frac{\partial\mathbf{g}\left(\boldsymbol{\mu}\right)}{\partial\boldsymbol{\mu}} = \mathbf{P}\mathbf{V}_\mu^{-1}$ since $\frac{\partial\ell_{i,\mu}}{\partial\mu_i} = \frac{1}{n_i\pi_i\left(1-\pi_i\right)}$ and $\mathbf{V}_\mu$ =diag$[n_i\pi_i\left(1-\pi_i\right)]$. Furthermore, $\mathbf{G}_\mathbf{y} = \frac{\partial\mathbf{g}\left(\boldsymbol{\mu}\right)}{\partial\boldsymbol{\mu}}|_{\mu=\mathbf{y}} = \mathbf{P}\mathbf{V}_\mathbf{y}^{-1}$ and $\mathbf{g}\left(\mathbf{y}\right) = \mathbf{P}\boldsymbol{\ell}_\mathbf{y}$ where $\boldsymbol{\ell}_\mathbf{y}$ has elements $\ell_{i,y} = \log\frac{y_i}{n_i-y_i}$. Substituting this into (53) gives

$$\begin{aligned}\widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - \left(\mathbf{P}\mathbf{V}_\mu^{-1}\mathbf{V}_\mu\right)'\left(\mathbf{P}\mathbf{V}_\mathbf{y}^{-1}\mathbf{V}_\mu\mathbf{V}_\mu^{-1}\mathbf{P}\right)^{-1}\mathbf{P}\boldsymbol{\ell}_\mathbf{y} + o\left(\|\mathbf{y}-\boldsymbol{\mu}\|\right)\\ &= \mathbf{y} - \mathbf{P}\left(\mathbf{P}\mathbf{V}_\mathbf{y}^{-1}\mathbf{P}\right)^{-1}\mathbf{P}\boldsymbol{\ell}_\mathbf{y} + o\left(\|\mathbf{y}-\boldsymbol{\mu}\|\right).\end{aligned}$$

Iteration takes place over $\mathbf{y}$.
The asymptotic covariance matrix of $\widehat{\boldsymbol{\mu}}_c$ is

$$\widehat{\boldsymbol{\Sigma}}_c = \mathbf{V}_{\widehat{\boldsymbol{\mu}}_c} - \mathbf{P}\left(\mathbf{P}\mathbf{V}_{\widehat{\boldsymbol{\mu}}_c}^{-1}\mathbf{P}\right)^{-1}\mathbf{P}.$$

The ML estimates for the parameters in the model are given by

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\ell_{\widehat{\mu}_c}$$

where $\ell_{\widehat{\mu}_c}$ is the vector of logits at convergence.
The asymptotic covariance matrix of $\widehat{\boldsymbol{\beta}}$ is

$$\text{cov}\left(\widehat{\boldsymbol{\beta}}\right) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\text{cov}\left(\ell_{\widehat{\mu}_c}, \ell'_{\widehat{\mu}_c}\right)\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}.$$

From the "delta method",

$$\begin{aligned} est\left[\text{cov}\left(\ell_{\widehat{\mu}_c}, \ell'_{\widehat{\mu}_c}\right)\right] &= \left(\frac{\partial \ell_{\widehat{\mu}_c}}{\partial \widehat{\mu}_c}\right)\widehat{\boldsymbol{\Sigma}}_c\left(\frac{\partial \ell_{\widehat{\mu}_c}}{\partial \widehat{\mu}_c}\right)' \\ &= \mathbf{V}_{\widehat{\mu}_c}^{-1}\widehat{\boldsymbol{\Sigma}}_c\mathbf{V}_{\widehat{\mu}_c}^{-1} \end{aligned}$$

and hence, the estimated covariance matrix for $\widehat{\boldsymbol{\beta}}$ is

$$est\left[\text{cov}\left(\widehat{\boldsymbol{\beta}}\right)\right] = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\left[\mathbf{V}_{\widehat{\mu}_c}^{-1}\widehat{\boldsymbol{\Sigma}}_c\mathbf{V}_{\widehat{\mu}_c}^{-1}\right]\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}.$$

EXAMPLE 3.2
Maximum likelihood estimation for a logistic regression model with a continuous covariate.

The data in Table 3.4, taken from Agresti (1990), was reported by Cornfield (1962) for a sample of male residents of Framingham, Massachusetts, aged 40-59, classified into 8 subgroups according to blood pressure . During a six-year follow-up period, they were classified according to whether they developed coronary heart disease. This is the response variable. The explanatory variable in the model is the value, x_i, which represents the blood pressure in subgroup i, $i = 1, 2, \ldots, 8$.

TABLE 3.4: Cross-Classification of Framingham Men by Blood Pressure and Heart Disease.

		Heart Disease	
Blood Pressure	x_i	Present (y_i)	Absent ($n_i - y_i$)
< 117	111.5	3	153
$117 - 126$	121.5	17	235
$127 - 136$	131.5	12	272
$137 - 146$	141.5	16	255
$147 - 156$	151.5	12	127
$157 - 166$	161.5	8	77
$167 - 186$	176.5	16	83
> 186	191.5	8	35

The model to be fitted is $\ell_{i,\mu} = \log\frac{\pi_i}{1-\pi_i} = \beta_0 + \beta_1 x_i$ which can be written as

$$\ell_{\boldsymbol{\mu}} = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & 111.5 \\ 1 & 121.5 \\ \vdots & \vdots \\ 1 & 191.5 \end{pmatrix}\begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}. \tag{54}$$

Results from the Proc Logistic and Proc Genmod procedures in SAS
The programs and output obtained from the PROC LOGISTIC and PROC GENMOD procedures in SAS are given in the Appendix. The results are summarized in Table 3.5.

TABLE 3.5: Results from SAS: Proc Logistic and Proc Genmod.

Maximum Likelihood Estimates		
Variable	Parameter Estimate	Standard Error
Intercept	−6.0820	0.7243
Blood Pressure	0.0243	0.00484

Model Fitting Information	
Pearson Chi-Square	6.2899
Deviance	5.9092

Obtaining the ML estimates by using the Newton-Raphson algorithm.
The ML estimate of $\boldsymbol{\beta}$ is found iteratively with equation (50) and the covariance matrix is given by equation (52).
The same results as in Table 3.5 are obtained. The program is given in the Appendix.

Obtaining the ML estimates under constraints
The ML estimate for $\boldsymbol{\mu}$ subject to the constraint $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{P}\ell_{\mu} = \mathbf{PX}\boldsymbol{\beta} = \mathbf{0}$ is found iteratively with the equation

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{P}\left(\mathbf{P}\mathbf{V}_{\mathbf{y}}^{-1}\mathbf{P}\right)^{-1}\mathbf{P}\ell_{\mathbf{y}} + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right)$$

where $\ell_{\mathbf{y}} = (\ell_{1,y}, \ell_{2,y}, \ldots, \ell_{p,y})$, $\ell_{i,y} = \log\dfrac{y_i}{n_i - y_i}$ for $i = 1, 2, 3 \ldots p$ and $\mathbf{P} = \mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)\mathbf{X}'$.
Iteration takes place only over $\mathbf{y}$.
The maximum likelihood estimates for the parameters are given by

$$\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\ell_{\widehat{\boldsymbol{\mu}}_c}$$

where $\ell_{\widehat{\boldsymbol{\mu}}_c}$ is the vector of logits at convergence.
The asymptotic covariance matrix of $\widehat{\boldsymbol{\beta}}$ is

$$\mathrm{Cov}\left(\widehat{\boldsymbol{\beta}}\right) = \left\{\mathbf{X}'\mathrm{Diag}\left[n_i\widehat{\pi}_i\left(1 - \widehat{\pi}_i\right)\right]\mathbf{X}\right\}^{-1}.$$

The same results as in Table 3.5 are obtained. The program is given in the Appendix.

EXAMPLE 3.3

Maximum likelihood estimation for a logistic regression model with a categorical covariate (logit model).

Pugh (1983) designed a study to examine the disposition of jurors to base their judgments of defendants on the alleged behavior of a rape victim. Pugh's study varied the degree to which the juror could assign fault to the victim ("low" or "high"). It also varied the presentation of the victim as someone with "high moral character", "low moral character" or "neutral". The response variable is the judgment of the defendant as "guilty" or "not guilty" by the jurors. The data are given in Table 3.6.

TABLE 3.6: Data from Pugh (1983).

		Moral (M)		
Verdict (V)	Fault (F)	High	Neutral	Low
Guilty	Low	42	79	32
	High	23	65	17
Not Guilty	Low	4	12	8
	High	11	41	24

The model to be fitted is $\ell_{i,\mu} = \log \frac{\pi_i}{1-\pi_i} = \alpha + \lambda_1^M x_{i1} + \lambda_2^M x_{i2} + \lambda_1^F x_{i3}$ where

$x_{i1} = 1$ and $x_{i2} = 0$ if Moral = High,
$x_{i1} = 0$ and $x_{i2} = 1$ if Moral = Neutral,
$x_{i1} = -1$ and $x_{i2} = -1$ if Moral = Low,
$x_{i3} = 1$ if Fault = Low,
$x_{i3} = -1$ if Fault = High.

This model assumes no interaction between moral and fault but it can be extended to include the interaction.

The model can be written as the logit model

$$\ell_\mu = \mathbf{X}\boldsymbol{\beta} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & 0 & -1 \\ 1 & 0 & 1 & -1 \\ 1 & -1 & -1 & -1 \end{pmatrix} \begin{pmatrix} \alpha \\ \lambda_1^M \\ \lambda_2^M \\ \lambda_1^F \end{pmatrix}.$$

Programs similar to those in Example 3.2 are given in the Appendix and the results are summarized in Table 3.7.

TABLE 3.7: Results for Example 3.3.

Maximum Likelihood Estimates		
Variable	Parameter Estimate	Standard Error
Intercept	1.0783	0.1469
Moral High	0.4553	0.2226
Moral Neutral	0.1210	0.1717
Fault Low	0.7739	0.1355

Model Fitting Information	
Pearson Chi-Square	0.2552
Deviance	0.2554

4 SYMMETRY MODELS FOR SQUARE CONTINGENCY TABLES WITH ORDERED CATEGORIES

Contingency tables are considered where the same variable with ordered categories is measured for both members of a matched pair. Responses are summarized in a two-way table in which both classifications have the same categories. One of the matters of interest in the analysis of square tables is the pattern of symmetry that may be exhibited by the cell probabilities in terms of their location relative to the main diagonal of the table. These models are discussed in more detail by Agresti (1984), Agresti (1990), Matthews (1995) and Tomizawa (1990).

4.1 SYMMETRY MODEL (S)

Consider an $I \times I$ contingency table with categorical variable $C = \{1, 2, \ldots, I\}$. A Poisson sampling procedure is assumed. Let Y_{ij} be the count in cell (i, j), y_{ij} the observed value of Y_{ij} and $n = \sum\sum y_{ij}$ the total counts. The counts can be arranged in a vector $\mathbf{Y}' = (Y_{11}, Y_{12}, \ldots, Y_{II})$ with $E(\mathbf{Y}) = \boldsymbol{\mu}$, the vector of expected counts. Let π_{ij} denote the probability that an observation falls in cell (i, j).
There is symmetry if

$$\pi_{ij} = \pi_{ji} \quad \text{for} \quad i \neq j.$$

Thus, if

$$\log\left(\mu_{ij}/\mu_{ji}\right) = \log \mu_{ij} - \log \mu_{ji} = 0 \quad \text{for} \quad i < j.$$

This can also be written as the constraint

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{C} \log \boldsymbol{\mu} = \mathbf{0}$$

where, in the case of a 4×4 table the matrix $\mathbf{C}$ is given by

μ_{11}	μ_{12}	μ_{13}	μ_{14}	μ_{21}	μ_{22}	μ_{23}	μ_{24}	μ_{31}	μ_{32}	μ_{33}	μ_{34}	μ_{41}	μ_{42}	μ_{43}	μ_{44}
0	1	0	0	−1	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	−1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	−1	0	0	0
0	0	0	0	0	0	1	0	0	−1	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	−1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	−1	0

(55)

Furthermore

$$\mathbf{G}_{\mu} = \frac{\partial}{\partial \boldsymbol{\mu}} \mathbf{g}(\boldsymbol{\mu}) = \mathbf{C}\mathbf{D}_{\mu}^{-1}.$$

The ML estimate for the vector with expected frequencies is given by

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - \left(\mathbf{G}_{\mu}\mathbf{V}_{\mu}\right)' \left(\mathbf{G}_{\mathbf{y}}\mathbf{V}_{\mu}\mathbf{G}_{\mu}'\right)^{-1} \mathbf{g}(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right) \\ &= \mathbf{y} - \mathbf{C}'\left(\mathbf{C}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{C}'\right)\mathbf{C} \log(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right). \end{aligned}$$

The degrees of freedom for the likelihood ratio statistic is $I(I-1)/2$.

4.2 CONDITIONAL SYMMETRY (CS)

The conditional symmetry model is defined as

$$\pi_{ij} = \begin{cases} \tau\psi_{ij} & \text{when} \quad i < j \\ \psi_{ij} & \text{when} \quad i \geq j, \end{cases}$$

where $\psi_{ij} = \psi_{ji}$.
This is similar to

$$\log\left(\mu_{ij}/\mu_{ji}\right) = \log\tau \quad \text{for} \quad i < j$$

or

$$\log\mu_{ij} - \log\mu_{ji} = \log\tau \quad \text{for} \quad i < j.$$

This model can be formulated as $\mathbf{g}(\boldsymbol{\mu}) = \mathbf{0}$. Consider a 4×4 table with

$$\boldsymbol{\mu}' = \left(\mu_{11}, \mu_{12}, \mu_{13}, \mu_{14}, \mu_{21}, \mu_{22}, \mu_{23}, \mu_{24}, \mu_{31}, \mu_{32}, \mu_{33}, \mu_{34}, \mu_{41}, \mu_{42}, \mu_{43}, \mu_{44}\right).$$

Then

$$\mathbf{C}\log(\boldsymbol{\mu}) = \mathbf{X}\log\tau$$

where $\mathbf{C}$ is the matrix given in (55) and $\mathbf{X}' = (1,1,1,1,1,1) = \mathbf{1}_6'$.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'$. The constraint for the model is

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC}\log(\boldsymbol{\mu}) = \mathbf{K}\log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
Furthermore

$$\mathbf{G}_\mu = \frac{\partial}{\partial\boldsymbol{\mu}}\mathbf{g}(\boldsymbol{\mu}) = \mathbf{KD}_\mu^{-1}.$$

The ML estimate for the vector with expected frequencies is obtained iteratively with

$$\begin{aligned} \widehat{\boldsymbol{\mu}}_c &= \mathbf{y} - \left(\mathbf{G}_\mu\mathbf{V}_\mu\right)'\left(\mathbf{G}_\mathbf{y}\mathbf{V}_\mu\mathbf{G}_\mu'\right)^{-1}\mathbf{g}(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right) \\ &= \mathbf{y} - \mathbf{K}'\left(\mathbf{KD}_\mathbf{y}^{-1}\mathbf{K}'\right)\mathbf{K}\log(\mathbf{y}) + o\left(\|\mathbf{y} - \boldsymbol{\mu}\|\right). \end{aligned}$$

The ML estimate for τ is obtained by

$$\widehat{\tau} = \exp\left[\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{C}\log\left(\widehat{\boldsymbol{\mu}}_c\right)\right].$$

The degrees of freedom for the likelihood ratio statistic is $(I+1)(I-2)/2$.

4.3 DIAGONALS-PARAMETER SYMMETRY (DPS)

Goodman (1979) defines the diagonals-parameter symmetry model as

$$\pi_{ij} = \begin{cases} \delta_{j-i}\psi_{ij} & \text{when} \quad i < j, \\ \psi_{ij} & \text{when} \quad i \geq j, \end{cases}$$

where $\psi_{ij} = \psi_{ji}$.
Consider a 4×4 table. The model can be written as

$$\mathbf{C}\log(\boldsymbol{\mu}) = \mathbf{X}\log\boldsymbol{\delta}$$

where $\mathbf{C}$ is the matrix given in (55),

$$\mathbf{X} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

and $\boldsymbol{\delta}' = (\delta_1, \delta_2, \delta_3)$.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The constraint for the model is

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC}\log(\boldsymbol{\mu}) = \mathbf{K}\log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
The ML estimates for the expected frequencies are obtained iteratively by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{K}'\left(\mathbf{K}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{K}'\right)\mathbf{K}\log(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

The ML estimate for $\boldsymbol{\delta}$ is obtained by

$$\widehat{\boldsymbol{\delta}} = \exp\left[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{C}\log(\widehat{\boldsymbol{\mu}}_c)\right].$$

The degrees of freedom for the likelihood ratio statistic is $(I-1)(I-2)/2$.

4.4 LINEAR DIAGONALS-PARAMETER SYMMETRY (LDPS)

The linear diagonals-parameter symmetry model is defined as

$$\pi_{ij} = \begin{cases} \rho^{j-i}\psi_{ij} & \text{when} \quad i < j, \\ \psi_{ij} & \text{when} \quad i \geq j, \end{cases}$$

where $\psi_{ij} = \psi_{ji}$.
Consider a 4×4 table. The model can be written as

$$\mathbf{C}\log(\boldsymbol{\mu}) = \mathbf{X}\log\rho$$

where $\mathbf{C}$ is the matrix given in (55) and $\mathbf{X}' = (1, 2, 3, 1, 2, 1)$.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The constraint for the model is

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC}\log(\boldsymbol{\mu}) = \mathbf{K}\log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
The ML estimates for the expected frequencies are obtained iteratively by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{K}'\left(\mathbf{K}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{K}'\right)\mathbf{K}\log(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

The ML estimate for ρ is obtained by

$$\widehat{\rho} = \exp\left[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{C}\log(\widehat{\boldsymbol{\mu}}_c)\right].$$

The degrees of freedom for the likelihood ratio statistic is $(I+1)(I-2)/2$.

4.5 ANOTHER LINEAR DIAGONALS-PARAMETER SYMMETRY MODEL (ALDPS)

Another linear diagonals-parameter symmetry model (ALDPS) is defined by Tomizawa (1990) as

$$\pi_{ij} = \begin{cases} \rho^{I-(j-i)}\psi_{ij} & \text{when} \quad i < j, \\ \psi_{ij} & \text{when} \quad i \geq j, \end{cases}$$

where $\psi_{ij} = \psi_{ji}$.
Consider a 4×4 table. The model can be written as

$$\mathbf{C}\log(\boldsymbol{\mu}) = \mathbf{X}\log\rho$$

where $\mathbf{C}$ is the matrix given in (55) and $\mathbf{X}' = (3, 2, 1, 3, 2, 3)$.

Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The const

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC}\log(\boldsymbol{\mu}) = \mathbf{K}\log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
The ML estimates for the expected frequencies are obtained iteratively by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{K}'\left(\mathbf{K}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{K}'\right)\mathbf{K}\log(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

The ML estimate for ρ is obtained by

$$\widehat{\rho} = \exp\left[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{C}\log(\widehat{\boldsymbol{\mu}}_c)\right].$$

The degrees of freedom for the likelihood ratio statistic is $(I+1)(I-2)/2$.

4.6 2-RATIOS-PARAMETER SYMMETRY (2RPS)

The 2-ratios-parameter symmetry model is defined by Tomizawa (1990) as

$$\pi_{ij} = \begin{cases} \phi\theta^{j-i-1}\psi_{ij} & \text{when} \quad i < j, \\ \psi_{ij} & \text{when} \quad i \geq j, \end{cases}$$

where $\psi_{ij} = \psi_{ji}$.
Consider a 4×4 table. The model can be written as

$$\mathbf{C}\log(\boldsymbol{\mu}) = \mathbf{X}\log\boldsymbol{\zeta}$$

where $\mathbf{C}$ is the matrix given in (55),

$$\mathbf{X} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}$$

and $\boldsymbol{\zeta}' = (\phi, \theta)$.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The constraint for the model is

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC}\log(\boldsymbol{\mu}) = \mathbf{K}\log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
The ML estimates for the expected frequencies are obtained iteratively by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{K}'\left(\mathbf{K}\mathbf{D}_{\mathbf{y}}^{-1}\mathbf{K}'\right)\mathbf{K}\log(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

The ML estimate for $\boldsymbol{\zeta}$ is

$$\widehat{\boldsymbol{\zeta}} = \exp\left[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{C}\log(\widehat{\boldsymbol{\mu}}_c)\right].$$

The degrees of freedom for the likelihood ratio statistic is $\left(I^2 - I - 4\right)/2$.

4.7 QUASI SYMMETRY (QS)

Quasi symmetry is defined as

$$\pi_{ij} = \alpha_i \beta_j \psi_{ij} \qquad \text{for all} \quad i, j,$$

where $\psi_{ij} = \psi_{ji}$.
Consider a 4×4 table. The model can be written as

$$\mathbf{C} \log(\boldsymbol{\mu}) = \mathbf{X} \log \boldsymbol{\theta}$$

where $\mathbf{C}$ is the matrix given in (55),

$$\mathbf{X} = \begin{pmatrix} 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 \end{pmatrix}$$

and $\boldsymbol{\theta}' = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. The constraint for the model is

$$\mathbf{g}(\boldsymbol{\mu}) = \mathbf{PC} \log(\boldsymbol{\mu}) = \mathbf{K} \log(\boldsymbol{\mu}) = \mathbf{0}$$

where $\mathbf{K} = \mathbf{PC}$.
The ML estimates for the expected frequencies are obtained iteratively by

$$\widehat{\boldsymbol{\mu}}_c = \mathbf{y} - \mathbf{K}' \left(\mathbf{K} \mathbf{D}_{\mathbf{y}}^{-1} \mathbf{K}'\right) \mathbf{K} \log(\mathbf{y}) + o(\|\mathbf{y} - \boldsymbol{\mu}\|).$$

The ML estimate for $\boldsymbol{\theta}$ is obtained by

$$\widehat{\boldsymbol{\theta}} = \exp\left[(\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{C} \log(\widehat{\boldsymbol{\mu}}_c)\right].$$

The degrees of freedom for the likelihood ratio statistic is $(I-1)(I-2)/2$.

4.8 EXAMPLE

Table 4.1, taken from Agresti (1984) and also discussed by Tomizawa (1990) is the father's and son's occupational mobility data in Britain. The table relates father's and son's occupational status category. The symmetry models discussed in this chapter were fitted to the data. Table 4.2 gives the expected cell frequencies for each model, Table 4.3 gives the goodness of fit statistics, and Table 4.4 gives the ML estimates for the model parameters.

TABLE 4.1: Occupational Status for British Father-Son Pairs.

	Son's Status					
Father's Status	1	2	3	4	5	Total
1	50	45	8	18	8	129
2	28	174	84	154	55	495
3	11	78	110	223	96	518
4	14	150	185	714	447	1510
5	3	42	72	320	411	848
Total	106	489	459	1429	1017	

TABLE 4.2: Occupational Status for British Father-Son Pairs.

	Son's Status					
Father's Status	1	2	3	4	5	Total
1	50	45	8	18	8	129
	–	(36.5)[a]	(9.5)	(16.0)	(5.5)	
	–	(40.7)[b]	(10.6)	(17.8)	(6.1)	
	–	(41.4)[c]	(10.0)	(18.1)	(8.0)	
	–	(38.9)[d]	(10.7)	(19.1)	(6.9)	
	–	(41.1)[e]	(10.4)	(17.0)	(5.7)	
	–	(41.0)[f]	(10.5)	(17.3)	(5.8)	
	–	(42.2)[g]	(10.7)	(18.8)	(7.3)	
2	28	174	84	154	55	495
	(36.5)	–	(81.0)	(152.0)	(48.5)	
	(32.3)	–	(90.3)	(169.5)	(54.1)	
	(31.6)	–	(91.8)	(159.7)	(54.9)	
	(34.1)	–	(86.3)	(172.0)	(58.0)	
	(31.9)	–	(91.2)	(166.4)	(51.6)	
	(32.0)	–	(90.9)	(167.6)	(52.5)	
	(30.8)	–	(78.3)	(155.2)	(56.8)	
3	11	78	110	223	96	518
	(9.5)	(81.0)	–	(204.0)	(84.0)	
	(8.4)	(71.7)	–	(227.5)	(93.7)	
	(9.0)	(70.2)	–	(213.2)	(88.3)	
	(8.3)	(76.7)	–	(217.5)	(95.0)	
	(8.6)	(70.8)	–	(229.7)	(91.9)	
	(8.5)	(71.1)	–	(229.0)	(92.6)	
	(8.3)	(83.7)	–	(215.0)	(101.0)	
4	14	150	185	714	447	1510
	(16.0)	(152.0)	(204.0)	–	(383.5)	
	(14.2)	(134.5)	(180.5)	–	(427.7)	
	(13.9)	(144.3)	(176.8)	–	(434.6)	
	(12.9)	(132.0)	(190.5)	–	(408.9)	
	(15.0)	(137.6)	(178.3)	–	(431.8)	
	(14.7)	(136.4)	(179.0)	–	(430.6)	
	(13.2)	(148.8)	(193.0)	–	(441.0)	
5	3	42	72	320	411	848
	(5.5)	(48.5)	(84.0)	(383.5)	–	
	(4.9)	(42.9)	(74.3)	(339.3)	–	
	(3.0)	(42.1)	(79.7)	(332.4)	–	
	(4.1)	(39.0)	(73.0)	(358.2)	–	
	(5.3)	(45.4)	(76.1)	(335.2)	–	
	(5.2)	(44.5)	(75.4)	(336.4)	–	
	(3.7)	(40.2)	(67.0)	(326.0)	–	
Total	106	489	459	1429	1017	

[a]Estimated expected frequencies for symmetry model (S).
[b]Estimated expected frequencies for conditional symmetry model (CS).
[c]Estimated expected frequencies for diagonals–parameter symmetry model (DPS).
[d]Estimated expected frequencies for linear diagonals–parameter symmetry model (LDPS).
[e]Estimated expected frequencies for another linear diagonals–parameter symmetry model (ALDPS).
[f]Estimated expected frequencies for 2-ratio–parameter symmetry model (2RPS).
[g]Estimated expected frequencies for quasi symmetry model (QS).

TABLE 4.3: Goodness of Fit statistics.

Model	df	χ^2	G^2	AIC^+
S	10	37.22	37.46	17.46
CS	9	10.30	10.35	−7.65
DPS	6	6.44	6.44	−5.56
LDPS	9	17.09	17.13	−0.87
ALDPS	9	10.05	10.13	−7.87
2RPS	8	9.96	10.02	−5.98
QS	6	4.67	4.66	−7.34

TABLE 4.4: ML Estimates of model parameters.

Model	Parameters	ML Estimates
S	–	–
CS	τ	$\widehat{\tau} = 1.26$
DPS	$\boldsymbol{\delta}' = (\delta_1, \delta_2, \delta_3, \delta_4)$	$\widehat{\delta}_1 = 1.31, \quad \widehat{\delta}_2 = 1.11, \quad \widehat{\delta}_3 = 1.30, \quad \widehat{\delta}_4 = 2.67$
LDPS	ρ	$\widehat{\rho} = 1.14$
ALDPS	ρ	$\widehat{\rho} = 1.07$
2RPS	$\boldsymbol{\zeta}' = (\phi, \theta)$	$\widehat{\phi} = 1.28, \quad \widehat{\theta} = 0.96$
QS	$\boldsymbol{\theta}' = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$	$\widehat{\alpha}_1 = 1.17, \quad \widehat{\alpha}_2 = 1.00 \quad \widehat{\alpha}_3 = 1.03 \quad \widehat{\alpha}_4 = 0.98$ $\widehat{\alpha}_5 = 0.84 \quad \widehat{\beta}_1 = 0.85 \quad \widehat{\beta}_2 = 1.00 \quad \widehat{\beta}_3 = 0.97$ $\widehat{\beta}_4 = 1.02 \quad \widehat{\beta}_5 = 1.19$

Discussion of results

Tomizawa (1990) selected the best model by using the modified AIC defined as

$$AIC^+ = G^2 - 2\,(df)\,.$$

The best fitting model is the one with the smallest AIC^+, which in this example is the ALDPS and CS models.

For the ALDPS model $\widehat{\rho} = 1.065$. Thus, the proportion of father-son pairs for which the son had a k grades higher status category than the father, for $k = 1, 2, 3, 4$, is estimated to be $(1.065)^{5-k}$ times higher than the proportion in which the father had the k grades higher status category.

For the CS model $\widehat{\tau} = 1.26$ which means that for each pair of categories, (i, j) and (j, i), the proportion of father-son pairs for which the son had the higher status is estimated to be 1.26 times higher than the proportion in which the father had the higher status.

The program for this example is given in the appendix and can be used for any square contingency table with ordered categories.

5 INCOMPLETE CONTINGENCY TABLES

An incomplete contingency table is a contingency table where information on one or more of the categorical variables is missing. It is assumed that the data are MAR and the missing data mechanism is ignorable. This chapter discusses ML estimation of cell probabilities in an incomplete contingency table by using all the observed data - including data where information on one or more of the categorical variables is missing. Lipsitz, Parzen and Molenberghs (1998) uses the Poisson generalized linear model to obtain ML estimates of cell probabilities for the saturated loglinear model whilst Little and Rubin (1987) describes and uses the EM algorithm to determine the ML estimates of cell probabilities for any loglinear model. Maximum likelihood estimation under constraints is also discussed in this chapter as a method to determine the ML estimates of cell probabilities. The advantage of this method is that it is less computational intensive compared to the more generally used EM algorithm. It also illustrates the elegance with which the method of ML estimation under constraints can be applied.

5.1 ML ESTIMATION IN INCOMPLETE CONTINGENCY TABLES

Consider an $I \times J$ contingency table with categorical variables $C_1 = \{1, 2, \ldots, I\}$ and $C_2 = \{1, 2, \ldots, J\}$. A multinomial sampling procedure is assumed. Let Y_{ij} be the count in cell (i, j), y_{ij} the observed value of Y_{ij} and $n = \sum\sum y_{ij}$ the total counts. The counts in each cell can be arranged to form the complete data vector $\mathbf{Y}' = (Y_{11}, Y_{12}, \ldots, Y_{IJ})$ with $E(\mathbf{Y}) = \boldsymbol{\mu}$, the vector of expected counts.
If information on one or both of the categories is missing the contingency table is said to be incomplete. The data to be classified in the contingency table can be split into two parts namely:

- the fully classified cases where information on all the categories is available and,
- the partially classified cases where information on some of the categories is missing.

It is assumed that the data are MAR and the missing data mechanism is ignorable.
In this section the saturated model is considered and the EM algorithm and ML estimation under constraints are described and illustrated as methods which uses both the fully and partially classified cases to determine the ML estimates of the cell probabilities.

5.1.1 The EM Algorithm

Multinomial Sampling
If the probability that an observation falls in cell (i, j) is π_{ij}, where $\pi_{ij} \geq 0$ and $\sum\sum \pi_{ij} = 1$ then the complete data $\mathbf{Y}$ have a multinomial distribution,

$$\mathbf{Y} \sim Mult(n; \pi_{11}, \pi_{12}, \ldots, \pi_{IJ})$$

with probability function

$$f(\mathbf{y}|\boldsymbol{\pi}) = \frac{n!}{\prod\prod y_{ij}!} \pi_{11}^{y_{11}} \pi_{12}^{y_{12}} \cdots \pi_{IJ}^{y_{IJ}} \tag{56}$$

where $\boldsymbol{\pi}' = (\pi_{11}, \pi_{12}, \ldots, \pi_{IJ})$.
The kernel of the complete data log-likelihood is

$$L(\boldsymbol{\pi}|\mathbf{y}) = y_{11} \log \pi_{11} + y_{12} \log \pi_{12} + \cdots + y_{IJ} \log \pi_{IJ}.$$

The cell counts, Y_{ij}, are the sufficient statistics and the MLE of π_{ij} is

$$\widehat{\pi}_{ij} = \frac{y_{ij}}{n}.$$

Product Multinomial Sampling

Let $Y_{i+} = \sum_j Y_{ij}$ be the total counts in row i and $\pi_{i+} = \sum_j \pi_{ij}$ the probability that an element falls in row i. If the Y_{i+} elements of row i are independent, each having a probability distribution $\frac{\pi_{ij}}{\pi_{i+}}$, $j = 1, 2, \ldots, J$, then, given the row total Y_{i+} and the vector of cell probabilities $\boldsymbol{\pi}$, the elements of row i have a multinomial distribution

$$Y_{i1}, Y_{i2}, \cdots, Y_{iJ} | Y_{i+}, \boldsymbol{\pi} \sim Mult\left(y_{i+}; \frac{\pi_{i1}}{\pi_{i+}}, \frac{\pi_{i2}}{\pi_{i+}}, \cdots, \frac{\pi_{iJ}}{\pi_{i+}}\right) \tag{57}$$

and $E\left(Y_{ij} | Y_{i+} = y_{i+}\right) = y_{i+}\left(\frac{\pi_{ij}}{\pi_{i+}}\right)$.

When samples from different rows are independent, the joint probability function for the entire data set is the product of I multinomial probability functions,

$$f\left(\mathbf{y} | \boldsymbol{\pi}, y_{1+}, y_{2+}, \ldots, y_{I+}\right) = \prod_{i=1}^{I} \left[\frac{y_{i+}!}{y_{i1}! y_{i2}! \cdots y_{iJ}!} \left(\frac{\pi_{i1}}{\pi_{i+}}\right)^{y_{i1}} \left(\frac{\pi_{i2}}{\pi_{i+}}\right)^{y_{i2}} \cdots \left(\frac{\pi_{iJ}}{\pi_{i+}}\right)^{y_{iJ}}\right].$$

Similarly, if the column totals are fixed then the elements of column j will have a multinomial distribution

$$Y_{1j}, Y_{2j}, \cdots, Y_{Ij} | Y_{+j}, \boldsymbol{\pi} \sim Mult\left(y_{+j}; \frac{\pi_{1j}}{\pi_{+j}}, \frac{\pi_{2j}}{\pi_{+j}}, \cdots, \frac{\pi_{Ij}}{\pi_{+j}}\right) \tag{58}$$

with $E\left(Y_{ij} | Y_{+j} = y_{+j}\right) = y_{+j}\left(\frac{\pi_{ij}}{\pi_{+j}}\right)$.

EM algorithm to determine the ML estimates of the cell probabilities in an incomplete $I \times J$ contingency table: data missing on both categories

If missing values occur on both C_1 and C_2, the observed data can be partitioned into three parts denoted by A, B and C respectively, where A includes units having both C_1 and C_2 observed, B includes those having only C_1 observed and C includes those where only C_2 was observed. In part A observations are fully classified and in B and C only partially. The three parts of the sample are displayed in Table 5.1. The objective is to determine the ML estimates of cell probabilities in the $I \times J$ table by using the fully and partially classified data.

TABLE 5.1 (a), (b) and (c): Classification of sample units in an incomplete $I \times J$ contingency table.

Sample part A

(a) Both variables observed

	$C_2 = 1$	$C_2 = 2$	$\cdots$	$C_2 = J$	
$C_1 = 1$	y_{11}^A	y_{12}^A	$\cdots$	y_{1J}^A	y_{1+}^A
$C_1 = 2$	y_{21}^A	y_{22}^A	$\cdots$	y_{2J}^A	y_{2+}^A
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
$C_1 = I$	y_{I1}^A	y_{I2}^A	$\cdots$	y_{IJ}^A	y_{I+}^A
	y_{+1}^A	y_{+2}^A	$\cdots$	y_{+J}^A	

Sample part B

(b) C_2 missing

$C_1 = 1$	y_{1+}^B
$C_1 = 2$	y_{2+}^B
$\vdots$	$\vdots$
$C_1 = I$	y_{I+}^B

Sample part C

(c) C_1 missing

$C_2 = 1$	$C_2 = 2$	$\cdots$	$C_2 = J$
y_{+1}^C	y_{+2}^C	$\cdots$	y_{+J}^C

Assume that the data are MAR and the missing data mechanism is ignorable. Let $\mathbf{Y}^{A\prime} = \left(Y_{11}^A, Y_{12}^A, \ldots, Y_{IJ}^A\right)$, $\mathbf{Y}^{B\prime} = \left(Y_{1+}^B, Y_{2+}^B, \ldots, Y_{I+}^B\right)$ and $\mathbf{Y}^{C\prime} = \left(Y_{+1}^C, Y_{+2}^C, \ldots, Y_{+J}^C\right)$ be the random vectors with counts for sample parts A, B and C respectively. Since C_2 is missing in sample part B, the counts observed are totals across C_2. Hence, compared to sample part A, row totals are observed in sample part B and column totals in sample part C. The observed data are

$$\left\{Y_{ij}^A, Y_{i+}^B, Y_{+j}^C : i = 1, 2, \ldots, I;\ j = 1, 2, \ldots, J\right\}.$$

Let $\mathbf{Y}_{obs}' = \left(\mathbf{Y}^{A\prime}, \mathbf{Y}^{B\prime}, \mathbf{Y}^{C\prime}\right)$ be the observed data vector, $\mathbf{Y}' = (Y_{11}, Y_{12}, \ldots, Y_{IJ})$ the complete data vector and $\boldsymbol{\pi}' = (\pi_{11}, \pi_{12}, \ldots, \pi_{IJ})$ the vector of cell probabilities for which the ML estimates must be determined.

Each complete data count, Y_{ij}, can be expressed as the sum of contributions from each of the three sample parts, that is $Y_{ij} = Y_{ij}^A + Y_{ij}^B + Y_{ij}^C$. For sample part B totals across C_2 are observed, that is Y_{i+}^B, whilst the individual cell counts, Y_{ij}^B, are missing. It follows from (57) that the predictive distribution of the missing data in part B given $\mathbf{Y}_{obs}$ and $\boldsymbol{\pi}$ is a product multinomial,

$$Y_{i1}^B, Y_{i2}^B, \cdots, Y_{iJ}^B | Y_{i+}^B, \boldsymbol{\pi} \sim Mult\left(y_{i+}^B; \frac{\pi_{i1}}{\pi_{i+}}, \frac{\pi_{i2}}{\pi_{i+}}, \cdots, \frac{\pi_{iJ}}{\pi_{i+}}\right) \tag{59}$$

with $E\left(Y_{ij}^B | Y_{i+}^B = y_{i+}^B, \boldsymbol{\pi}\right) = y_{i+}^B \left(\frac{\pi_{ij}}{\pi_{i+}}\right)$.

For part C only the totals across C_1 are observed, that is Y_{+j}^C. From (58) the predictive distribution of the missing data in sample part C given $\mathbf{Y}_{obs}$ and $\boldsymbol{\pi}$ is a product multinomial given by

$$Y_{1j}^C, Y_{2j}^C, \cdots, Y_{Ij}^C | Y_{+j}^C, \boldsymbol{\pi} \sim Mult\left(y_{+j}^C; \frac{\pi_{1j}}{\pi_{+j}}, \frac{\pi_{2j}}{\pi_{+j}}, \cdots, \frac{\pi_{Ij}}{\pi_{+j}}\right) \tag{60}$$

with $E\left(Y_{ij}^C | Y_{+j}^C = y_{+j}^C, \boldsymbol{\pi}\right) = y_{+j}^C \left(\frac{\pi_{ij}}{\pi_{+j}}\right)$.

Thus, $E\left(Y_{ij} | \mathbf{Y}_{obs}, \boldsymbol{\pi}\right) = E\left(Y_{ij}^A + Y_{ij}^B + Y_{ij}^C | \mathbf{Y}_{obs}, \boldsymbol{\pi}\right) = y_{ij}^A + y_{i+}^B \left(\frac{\pi_{ij}}{\pi_{i+}}\right) + y_{+j}^C \left(\frac{\pi_{ij}}{\pi_{+j}}\right)$.

The distribution of the complete data belong to the regular exponential family with sufficient statistics the cell counts, Y_{ij}. In the E-step of the EM algorithm $E\left(Y_{ij} | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right)$ is calculated where $\boldsymbol{\pi}^{(r)}$, $r = 0, 1, 2, \ldots$, is the rth estimate of $\widehat{\boldsymbol{\pi}}$. From (59) and (60)

$$\begin{aligned} E\left(Y_{ij} | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right) &= E\left(Y_{ij}^A + Y_{ij}^B + Y_{ij}^C | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right) \\ &= y_{ij}^A + E\left(Y_{ij}^B | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right) + E\left(Y_{ij}^C | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right) \\ &= y_{ij}^A + y_{i+}^B \left(\frac{\pi_{ij}^{(r)}}{\pi_{i+}^{(r)}}\right) + y_{+j}^C \left(\frac{\pi_{ij}^{(r)}}{\pi_{+j}^{(r)}}\right). \end{aligned} \tag{61}$$

In the M-step $\boldsymbol{\pi}^{(r+1)}$ is calculated by substituting the results from the E-step into the expression of the MLE of $\boldsymbol{\pi}$ for the complete data. That is,

$$\begin{aligned} \pi_{ij}^{(r+1)} &= \frac{1}{n} E\left(Y_{ij} | \mathbf{Y}_{obs}, \boldsymbol{\pi}^{(r)}\right) \\ &= \frac{1}{n}\left[y_{ij}^A + y_{i+}^B \left(\frac{\pi_{ij}^{(r)}}{\pi_{i+}^{(r)}}\right) + y_{+j}^C \left(\frac{\pi_{ij}^{(r)}}{\pi_{+j}^{(r)}}\right)\right] \end{aligned} \tag{62}$$

The process iterates between (61) and (62) until convergence is attained.

5.1.2 ML Estimation under constraints

The data from parts A, B and C in Table 5.1 can be considered as three independent multinomial samples. Let $n^A = \sum\sum y_{ij}^A$, $n^B = \sum y_{i+}^B$ and $n^C = \sum y_{+j}^C$ be the total counts in sample parts A, B and C respectively.

Let $\mathbf{p}^A = \frac{1}{n^A}\,\mathbf{y}^A$, $\mathbf{p}^B = \frac{1}{n^B}\,\mathbf{y}^B$ and $\mathbf{p}^C = \frac{1}{n^C}\,\mathbf{y}^C$ be the proportions in each sample part and $\mathbf{p}'_{obs} = \left(\mathbf{p}^{A\prime}, \mathbf{p}^{B\prime}, \mathbf{p}^{C\prime}\right)$ with $E\left(\mathbf{p}'_{obs}\right) = \boldsymbol{\pi}'_{obs} = \left(\boldsymbol{\pi}^{A\prime}, \boldsymbol{\pi}^{B\prime}, \boldsymbol{\pi}^{C\prime}\right)$.

For the saturated model the maximum likelihood estimates of $\boldsymbol{\pi}_{obs}$ can be determined under the constraints

$$\pi_{i+}^A - \pi_{i+}^B = 0 \quad \text{for } i = 1, 2, \ldots, I \tag{63}$$

and

$$\pi_{+j}^A - \pi_{+j}^C = 0 \quad \text{for } j = 1, 2, \ldots, J. \tag{64}$$

Hence, the constraint can be written as $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$ where

$$\mathbf{A} : (I+J) \times (IJ + I + J) = \begin{pmatrix} \mathbf{I}_I \otimes \mathbf{1}'_J & \\ \mathbf{1}'_I \otimes \mathbf{I}_J & -\mathbf{I}_{I+J} \end{pmatrix}$$

and where $\mathbf{1}'_J$ and $\mathbf{1}'_I$ indicates $1 \times J$ and $1 \times I$ row vectors respectively with all values equal to 1.

The ML estimate of the vector of cell probabilities, under the constraint $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$ is given by

$$\widehat{\boldsymbol{\pi}}_{obs,c} = \left(\widehat{\boldsymbol{\pi}}_c^{A\prime}, \widehat{\boldsymbol{\pi}}_c^{B\prime}, \widehat{\boldsymbol{\pi}}_c^{C\prime}\right)' = \mathbf{p}_{obs} - \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\pi}_{obs}}\right)' \left(\mathbf{A}\mathbf{V}_{\boldsymbol{\pi}_{obs}}\mathbf{A}'\right)^{-1} \mathbf{A}\mathbf{p}_{obs} \tag{65}$$

where

$$\begin{aligned} \mathbf{V}_{\boldsymbol{\pi}_{obs}} &= \begin{pmatrix} \mathrm{Cov}\left(\boldsymbol{\pi}^A\right) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathrm{Cov}\left(\boldsymbol{\pi}^B\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathrm{Cov}\left(\boldsymbol{\pi}^C\right) \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{n^A}\left(D_{\boldsymbol{\pi}^A} - \boldsymbol{\pi}^A\boldsymbol{\pi}^{A\prime}\right) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{n^B}\left(D_{\boldsymbol{\pi}^B} - \boldsymbol{\pi}^B\boldsymbol{\pi}^{B\prime}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{1}{n^C}\left(D_{\boldsymbol{\pi}^C} - \boldsymbol{\pi}^C\boldsymbol{\pi}^{C\prime}\right) \end{pmatrix}. \end{aligned} \tag{66}$$

Since the constraint, $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$, is linear in $\boldsymbol{\pi}_{obs}$ iteration is only over $\boldsymbol{\pi}_{obs}$.

The ML estimates of cell probabilities in the $I \times J$ table are given by the elements of $\widehat{\boldsymbol{\pi}}^A$ in $\widehat{\boldsymbol{\pi}}_{obs,c}$.

EXAMPLE 5.1

Determining the ML estimates of cell probabilities in an incomplete contingency table by using the EM algorithm.

Consider the data in Table 5.2 from Schafer (1997) obtained through the National Crime Survey conducted by the U.S. Bureau of the Census. Housing unit occupants were interviewed to determine whether they had been victimized by crime in the preceding six-month period. Six months later the units were visited again to determine whether the occupants had been victimized in the intervening months.

TABLE 5.2: Victimization status from the National Crime Survey.

	Second Visit		
First Visit	Crime-free	Victims	Missing
Crime-free	392	55	33
Victims	76	38	9
Missing	31	7	

Following the notation in 5.1.1,
$\mathbf{y}'_{obs} = (\mathbf{y}^{A\prime}, \mathbf{y}^{B\prime}, \mathbf{y}^{C\prime})$ where
$\mathbf{y}^{A\prime} = \{y_{ij} : i,j = 1,2\} = (392, 55, 76, 38)$
$\mathbf{y}^{B\prime} = \{y^B_{i+} : i = 1,2\} = (33, 9)$
$\mathbf{y}^{C\prime} = \{y^C_{+j} : j = 1,2\} = (31, 7)$.
The fully classified data, $\mathbf{y}^A$, were used to determine a starting value for the algorithm, $\boldsymbol{\pi}^{(0)\prime} = \frac{1}{561}(392, 55, 76, 38) \approx (0.70, 0.10, 0.13, 0.07)$. From (62) the first estimate of $\widehat{\pi}_{11}$ is

$$\begin{aligned}
\pi^{(1)}_{11} &= \frac{1}{n}\left[y^A_{11} + y^B_{1+}\left(\frac{\pi^{(0)}_{11}}{\pi^{(0)}_{1+}}\right) + y^C_{+j}\left(\frac{\pi^{(0)}_{11}}{\pi^{(0)}_{+1}}\right)\right] \\
&= \frac{1}{641}\left[392 + 33\left(\frac{0.70}{0.80}\right) + 31\left(\frac{0.70}{0.83}\right)\right] \\
&= 0.6974.
\end{aligned}$$

Similarly, the first estimates of $\widehat{\pi}_{12}$, $\widehat{\pi}_{21}$ and $\widehat{\pi}_{22}$ are

$$\begin{aligned}
\pi^{(1)}_{12} &= \frac{1}{641}\left[55 + 33\left(\frac{0.10}{0.80}\right) + 7\left(\frac{0.10}{0.17}\right)\right] = 0.0987 \\
\pi^{(1)}_{21} &= \frac{1}{641}\left[76 + 9\left(\frac{0.13}{0.20}\right) + 31\left(\frac{0.13}{0.83}\right)\right] = 0.1353 \\
\pi^{(1)}_{22} &= \frac{1}{641}\left[38 + 9\left(\frac{0.07}{0.20}\right) + 7\left(\frac{0.07}{0.17}\right)\right] = 0.0687.
\end{aligned}$$

This gives $\boldsymbol{\pi}^{(1)\prime} = (0.6974, 0.0987, 0.1353, 0.0687)$ which is used to calculate the second estimate for $\widehat{\boldsymbol{\pi}}$. The process continues until convergence is attained. Table 5.3 shows the values at different steps of the algorithm.

TABLE 5.3: Iterations of the EM algorithm.

r	$\pi^{(r)}_{11}$	$\pi^{(r)}_{12}$	$\pi^{(r)}_{21}$	$\pi^{(r)}_{22}$
0	0.7000	0.1000	0.1300	0.0700
1	0.6974	0.0987	0.1353	0.0687
2	0.6972	0.0986	0.1357	0.0685
3	0.6971	0.0986	0.1358	0.0685
∞	0.6971	0.0986	0.1358	0.0685

EXAMPLE 5.2

Determining the ML estimates of cell probabilities in an incomplete contingency table under constraints.

Consider the data in Example 5.1.

$\mathbf{y}'_{obs} = (\mathbf{y}^{A\prime}, \mathbf{y}^{B\prime}, \mathbf{y}^{C\prime})$ where

$\mathbf{y}^{A\prime} = \{y_{ij} : i,j = 1,2\} = (392, 55, 76, 38)$ and $\mathbf{p}^A = \dfrac{1}{561}\,\mathbf{y}^A$

$\mathbf{y}^{B\prime} = \left\{y^B_{i+} : i = 1,2\right\} = (33, 9)$ and $\mathbf{p}^B = \dfrac{1}{42}\,\mathbf{y}^B$

$\mathbf{y}^{C\prime} = \left\{y^C_{+j} : i = 1,2\right\} = (31, 7)$ and $\mathbf{p}^C = \dfrac{1}{38}\,\mathbf{y}^C$.

Let $\mathbf{p}'_{obs} = (\mathbf{p}^{A\prime}, \mathbf{p}^{B\prime}, \mathbf{p}^{C\prime})$ with $E(\mathbf{p}_{obs}) = \boldsymbol{\pi}_{obs}$.

For the saturated model the constraint $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$ must hold, where the elements of $\mathbf{A}$ are

$$\mathbf{A}: \begin{array}{cccccccc} \pi^A_{11} & \pi^A_{12} & \pi^A_{21} & \pi^A_{22} & \pi^B_{1+} & \pi^B_{2+} & \pi^C_{+1} & \pi^C_{+2} \\ 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 \end{array}$$

The ML estimate of $\boldsymbol{\pi}_{obs}$ under the constraint $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$ is obtained with

$$\widehat{\boldsymbol{\pi}}_{obs,c} = \left(\widehat{\boldsymbol{\pi}}^{A\prime}_c, \widehat{\boldsymbol{\pi}}^{B\prime}_c, \widehat{\boldsymbol{\pi}}^{C\prime}_c\right)' = \mathbf{p}_{obs} - (\mathbf{A}\mathbf{V}_{\pi_{obs}})'(\mathbf{A}\mathbf{V}_{\pi_{obs}}\mathbf{A}')^{-1}\mathbf{A}\mathbf{p}_{obs} \tag{67}$$

where

$$\mathbf{V}_{\pi_{obs}} = \begin{pmatrix} \frac{1}{561}\left(D_{\pi^A} - \boldsymbol{\pi}^A\boldsymbol{\pi}^{A\prime}\right) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{42}\left(D_{\pi^B} - \boldsymbol{\pi}^B\boldsymbol{\pi}^{B\prime}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{1}{38}\left(D_{\pi^C} - \boldsymbol{\pi}^C\boldsymbol{\pi}^{C\prime}\right) \end{pmatrix}.$$

The ML estimates of the cell probabilities in the 2×2 table are given by the elements of $\widehat{\boldsymbol{\pi}}^A_c$ in $\widehat{\boldsymbol{\pi}}_{obs,c}$. This procedure gives the same values for the ML estimates as obtained with the EM algorithm in Example 5.1. Results obtained under constraints and from the Genmod procedure in SAS are shown in Table 5.4. The programs are given in the Appendix.

TABLE 5.4: ML estimates and standard errors.

	Estimate	Std Err
π_{11}	0.6971	0.0187
π_{12}	0.0986	0.0124
π_{21}	0.1358	0.0141
π_{22}	0.0685	0.0104

EXAMPLE 5.3

Determining the ML estimates of cell probabilities in an incomplete contingency table under constraints.

Consider the data in Table 5.5 (from Lipsitz, Parzen, Molenberghs (1998)) which contains the data from the Six Cities Study, a study conducted to assess the health effects of air polution. The columns corresponds to the wheezing status (no wheeze, wheeze with cold, wheeze apart from cold) of a child at age 10. The rows represent the smoking status of the child's mother (none, medium, heavy) during that time. For some individuals the maternal smoking variable is missing, while for others the child's wheezing status is missing. The objective is to estimate the probabilities of the joint distribution of maternal smoking and respiratory illness.

TABLE 5.5: Six Cities Data: Maternal Smoking Cross-Classified by Child's Wheeze Status.

	Child's wheeze status			
Maternal Smoking	No Wheeze	Wheeze with Cold	Wheeze apart from cold	Missing
None	287	39	38	279
Moderate	18	6	4	27
Heavy	91	22	23	201
Missing	59	18	26	

Similar as in Example 5.2:
$\mathbf{y}'_{obs} = (\mathbf{y}^{A\prime}, \mathbf{y}^{B\prime}, \mathbf{y}^{C\prime}) = (287, 39, 38, 18, 6, 4, 91, 22, 23, 279, 27, 201, 59, 18, 26)$. For the constraint $\mathbf{A}\boldsymbol{\pi}_{obs} = \mathbf{0}$ the elements of $\mathbf{A}$ are given by

$$\mathbf{A}: \begin{array}{cccccccccccccc} \pi^A_{11} & \pi^A_{12} & \pi^A_{13} & \pi^A_{21} & \pi^A_{22} & \pi^A_{23} & \pi^A_{31} & \pi^A_{32} & \pi^A_{33} & \pi^B_{1+} & \pi^B_{2+} & \pi^B_{3+} & \pi^C_{+1} & \pi^C_{+2} & \pi^C_{+3} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \end{array}.$$

The ML estimate of $\boldsymbol{\pi}'_{obs} = (\boldsymbol{\pi}^{A\prime}, \boldsymbol{\pi}^{B\prime}, \boldsymbol{\pi}^{C\prime})$ is obtained iteratively with

$$\widehat{\boldsymbol{\pi}}_{obs,c} = \mathbf{p}_{obs} - (\mathbf{A}\mathbf{V}_{\boldsymbol{\pi}_{obs}})' (\mathbf{A}\mathbf{V}_{\boldsymbol{\pi}_{obs}}\mathbf{A}')^{-1} \mathbf{A}\mathbf{p}_{obs}.$$

The ML estimates of the cell probabilities, given in Table 5.6, are the same as those obtained by Lipsitz, Parzen and Molenberghs (1998). Procedures give asymptotically equivalent results. Slight differences in the standard errors are indicated.
Table 5.6 also gives the ML estimates of cell probabilities when using only the 528 fully classified cases.

TABLE 5.6. ML estimates and standard errors.

	Fully Classified Cases $n = 528$		Fully and Partially Classified Cases $n = 528 + 610$		
	Estimate	Std Err	Estimate	Std Err(Genmod)	
π_{11}	0.5436	0.0217	0.4747	0.0179	(0.0174)
π_{12}	0.0739	0.0114	0.0701	0.0105	(0.0102)
π_{13}	0.0720	0.0112	0.0742	0.0108	(0.0107)
π_{21}	0.0341	0.0079	0.0327	0.0065	(0.0064)
π_{22}	0.0114	0.0046	0.0120	0.0044	(0.0045)
π_{23}	0.0076	0.0038	0.0087	0.0039	(0.0041)
π_{31}	0.1723	0.0164	0.2060	0.0149	(0.0158)
π_{32}	0.0417	0.0087	0.0558	0.0094	(0.0106)
π_{33}	0.0436	0.0089	0.0658	0.0100	(0.0116)

5.2 LOGLINEAR MODELS FOR INCOMPLETE CONTINGENCY TABLES

In this section the EM algorithm and ML estimation under constraints are discussed as methods to determine the ML estimates of the cell probabilities in the complete table for any loglinear model where both the fully and partially classified cases are used.
It is assumed that the data are MAR and the missing data mechanism is ignorable.

5.2.1 The EM algorithm

The starting values used in the EM algorithm are the ML estimates of cell probabilities obtained by using only the data in the fully classified table. The process then iterates between the E-step and the M-step. In the E-step the counts in the partially classified table are distributed into the full table by using the ML estimates of the cell probabilities obtained in the M-step. In the M-step ML estimates of the cell probabilities for the filled in table are obtained and used in the E-step as the next approximation of the ML estimates of the cell probabilities in the complete table. The ML estimation procedure under constraints for loglinear models (Section 3.1) can be used in the M-step of the algorithm.

5.2.2 ML Estimation under constraints

Consider an $I \times J \times K$ contingency table with C_1, C_2 and C_3 the three categorical variables where $C_1 = \{1,2,\ldots,I\}$, $C_2 = \{1,2,\ldots,J\}$ and $C_3 = \{1,2,\ldots,K\}$. Suppose that for n^A cases, information for C_1, C_2 and C_3 is known and for n^B cases C_1 is missing. The n^A cases are classified in an $I \times J \times K$ table and the n^B cases in a $J \times K$ table. The objective is to determine the ML estimates of the cell probabilities in the $I \times J \times K$ contingency table, for a specific loglinear model, by using both the n^A fully classified cases and the n^B partially classified cases. A specific loglinear model is assumed.
Suppose $I = J = K = 2$. Let $\mathbf{Y}^{A\prime} = \left(Y^A_{111}, Y^A_{112}, Y^A_{121}, Y^A_{122}, Y^A_{211}, Y^A_{212}, Y^A_{221}, Y^A_{222}\right)$ be the $IJK \times 1$ vector of cell counts for the fully classified table with $E\left(\mathbf{Y}^A\right) = \boldsymbol{\mu}^A$ and let $\mathbf{Y}^{B\prime} = \left(Y^B_{+11}, Y^B_{+12}, Y^B_{+21}, Y^B_{+22}\right)$ the $JK \times 1$ vector of cell counts for the partially classified table with $E\left(\mathbf{Y}^B\right) = \boldsymbol{\mu}^B$. Furthermore let $\mathbf{y}'_{obs} = \left(\mathbf{y}^{A\prime}, \mathbf{y}^{B\prime}\right)$, $\boldsymbol{\mu}'_{obs} = \left(\boldsymbol{\mu}^{A\prime}, \boldsymbol{\mu}^{B\prime}\right)$ and $\boldsymbol{\pi}'_{obs} = \left(\frac{1}{n_A}\boldsymbol{\mu}^{A\prime}, \frac{1}{n_B}\boldsymbol{\mu}^{B\prime}\right) = \left(\boldsymbol{\pi}^{A\prime}, \boldsymbol{\pi}^{B\prime}\right)$.
Two sets of constraints are imposed; the first pertains to the specific loglinear model that is fitted and the second is used to constrain the marginal probabilities in the fully and partially classified tables.

Constraint 1
The saturated loglinear model for the fully classified data is

$$\log \boldsymbol{\mu}^A = \mathbf{X}\boldsymbol{\beta} \tag{68}$$

where $\boldsymbol{\mu}^A$ is the vector with expected cell frequencies, $\mathbf{X} : IJK \times IJK$ is the design matrix and $\boldsymbol{\beta} : IJK \times 1$ is the parameter vector for the saturated model.
The unsaturated model can be written as

$$\log \boldsymbol{\mu}^A = \mathbf{X}_u\boldsymbol{\beta}_u \tag{69}$$

where $\mathbf{X}_u$ is a submatrix of $\mathbf{X}$ given in (68) and $\boldsymbol{\beta}_u$ is the parameter vector of the model.
Let $\mathbf{P} = \mathbf{I} - \mathbf{X}_u\left(\mathbf{X}'_u\mathbf{X}_u\right)^{-1}\mathbf{X}'_u$. The constraint for the model in (69) is

$$\mathbf{g}_1\left(\boldsymbol{\mu}^A\right) = \mathbf{P}\log\boldsymbol{\mu}^A = \mathbf{P}\mathbf{X}_u\boldsymbol{\beta}_u = \mathbf{0}. \tag{70}$$

Constraint 2
The sum of the expected cell probabilities in the $I \times J \times K$ fully classified table over category C_1, gives the expected marginal cell probabilities,

$$\pi^A_{+jk} = \sum_{i=1}^{I} \pi^A_{ijk}, \text{ for } j = 1,2,\ldots,J \text{ and } k = 1,2,\ldots,K.$$

The constraint which must hold betw[illegible] ssified tables is

$$\pi^A_{+jk} = \pi^B_{+jk}, \text{ for all } j, k.$$

Hence, the second constraint can be written as

$$\mathbf{g}_2(\boldsymbol{\mu}_{obs}) = \begin{pmatrix} \frac{1}{n^A}\mathbf{1}'_I \otimes \mathbf{I}_{JK} & -\frac{1}{n^B}\mathbf{I}_{JK} \end{pmatrix} \boldsymbol{\mu}_{obs} = \mathbf{0}. \tag{71}$$

Combining (70) and (71) gives

$$\mathbf{g}(\boldsymbol{\mu}_{obs}) = \begin{pmatrix} \mathbf{g}_1(\boldsymbol{\mu}^A) \\ \mathbf{g}_2(\boldsymbol{\mu}_{obs}) \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}$$

The ML estimate of $\boldsymbol{\mu}_{obs}$ subject to $\mathbf{g}(\boldsymbol{\mu}_{obs}) = \mathbf{0}$ is determined iteratively with

$$\widehat{\boldsymbol{\mu}}_{obs,c} = \mathbf{y}_{obs} - \left(\mathbf{G}_{\boldsymbol{\mu}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\right)' \left(\mathbf{G}_{\mathbf{y}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\mathbf{G}'_{\boldsymbol{\mu}_{obs}}\right)^{-1} \mathbf{g}(\mathbf{y}_{obs}) + o(\|\mathbf{y}_{obs} - \boldsymbol{\mu}_{obs}\|) \tag{72}$$

where

$$\mathbf{G}_{\boldsymbol{\mu}_{obs}} = \frac{\partial \mathbf{g}(\boldsymbol{\mu}_{obs})}{\partial \boldsymbol{\mu}_{obs}} = \begin{pmatrix} \frac{\partial \mathbf{g}_1(\boldsymbol{\mu}^A)}{\partial \boldsymbol{\mu}_{obs}} \\ \frac{\partial \mathbf{g}_2(\boldsymbol{\mu}_{obs})}{\partial \boldsymbol{\mu}_{obs}} \end{pmatrix} = \begin{pmatrix} \mathbf{P}\mathbf{D}^{-1}_{\boldsymbol{\mu}^A} & \mathbf{0}_{IJK \times JK} \\ \frac{1}{n^A}\mathbf{1}'_I \otimes \mathbf{I}_{JK} & -\frac{1}{n^B}\mathbf{I}_{JK} \end{pmatrix},$$

$$\mathbf{G}_{\mathbf{y}_{obs}} = \frac{\partial \mathbf{g}(\boldsymbol{\mu}_{obs})}{\partial \boldsymbol{\mu}_{obs}}\Big|_{\boldsymbol{\mu}_{obs}=\mathbf{y}_{obs}} \text{ and}$$

$$\mathbf{V}_{\boldsymbol{\mu}_{obs}} = \mathbf{D}_{\boldsymbol{\mu}_{obs}} - \frac{1}{n^A + n^B}\boldsymbol{\mu}_{obs}\boldsymbol{\mu}'_{obs}.$$

The ML estimates of the cell probabilities in the $I \times J \times K$ table is

$$\widehat{\boldsymbol{\pi}}^A = \frac{\widehat{\boldsymbol{\mu}}^A_c}{n^A}.$$

EXAMPLE 5.4

Determining the maximum likelihood estimates of cell probabilities in an incomplete contingency table for any loglinear model.

In Table 5.7, from Little and Rubin (1987), the survival of infants are related according to the amount of prenatal care received by the mothers and the clinic they attended. For data in Table 5.7(a) information on survival, prenatal care and clinic attended were recorded but in Table 5.7(b) information of the clinic attended is missing.

TABLE 5.7. A 2^3 contingency table with partially classified observations.

		Survival (S)		
Clinic (C)	Prenatal Care (P)	Died	Survived	
	(a) *Fully Classified Cases*			
A	Less	3	176	
	More	4	293	
B	Less	17	197	
	More	2	23	715 cases
	(b) *Partially Classified Cases (Clinic missing)*			
	Less	10	150	
	More	5	90	255 cases

The ML estimates of cell probabilities for different loglinear models are given in Table 5.8. The cell probabilities are given in the form $100\widehat{\pi}_{CPS}$ where

$C = 1$ if Clinic = "A" and $C = 2$ if Clinic = "B";
$P = 1$ if Care = "Less" and $P = 2$ if Care = "More";
$S = 1$ if Survival = "Died" and $S = 2$ if Survival = "Survived".

The saturated model $\{CPS\}$ was fitted to the incomplete data as explained in section 5.1.2 and the models $\{PS, CS, CP\}$, $\{CS, CP\}$ and $\{PS, CS\}$ were fitted by using the EM algorithm and the ML procedure under constraints.

TABLE 5.8: ML estimates of cell probabilities for different loglinear models.

	$\{CPS\}$	$\{PS, CS, CP\}$	$\{PS, CS\}$	$\{CS, CP\}$
$100\widehat{\pi}_{111}$	0.4639	0.4350	0.8327	0.4963
$100\widehat{\pi}_{112}$	25.4410	25.4680	36.7015	25.4203
$100\widehat{\pi}_{121}$	0.7560	0.7913	0.3053	0.7579
$100\widehat{\pi}_{122}$	38.8092	38.7845	28.4910	38.8208
$100\widehat{\pi}_{211}$	2.6289	2.6578	2.2601	2.6787
$100\widehat{\pi}_{212}$	28.4765	28.4495	17.2160	28.4150
$100\widehat{\pi}_{221}$	0.3780	0.3427	0.8287	0.2939
$100\widehat{\pi}_{222}$	3.0465	3.0712	13.3647	3.1172

Only the $\{CS, CP\}$ loglinear model is discussed in more detail. The programs are given in the Appendix.

The EM algorithm

The observed frequency vector for the 715 fully classified cases is $\mathbf{y}^{A\prime}=(3,176,4,293,17,197,2,23)$ and for the 255 partially classified cases $\mathbf{y}^{B\prime}=(10,150,5,90)$.

The EM algorithm is used to determine $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\pi}}$, the ML estimates of the cell frequencies and cell probabilities in the 2^3 table.

The steps for the EM algorithm are as follows:

Step 1: Starting value for the EM algorithm

The starting value of the EM algorithm are the ML estimates obtained by using only the data in the fully classified table.

From section 3.1.3, $\boldsymbol{\mu}^{(0)}$, the first approximation of $\widehat{\boldsymbol{\mu}}$, is determined iteratively with

$$\boldsymbol{\mu}^{(0)}=\mathbf{y}^{A}-\mathbf{A}_C\left(\mathbf{A}_C'\mathbf{D}_{\mathbf{y}^A}^{-1}\mathbf{A}_C\right)^{-1}\mathbf{g}\left(\mathbf{y}^{A}\right)+o\left(\left\|\mathbf{y}^{A}-\boldsymbol{\mu}\right\|\right) \tag{73}$$

and from this, $\boldsymbol{\pi}^{(0)}=\dfrac{\boldsymbol{\mu}^{(0)}}{715}$.

Step 2: E-Step

In the E-step $\boldsymbol{\pi}^{(r)}$, $r=0,1,\ldots$ is used to distribute the 255 partially classified counts into the full table. The filled in frequency vector at the rth step of the EM algorithm is

$$\mathbf{y}^{(r)}=\mathbf{y}^{A}+\frac{\boldsymbol{\pi}^{(r)}}{\left[\begin{pmatrix}1&1\\1&1\end{pmatrix}\otimes\mathbf{I}_{JK}\right]\boldsymbol{\pi}^{(r)}}\#\left[\begin{pmatrix}1\\1\end{pmatrix}\otimes\mathbf{y}^{B}\right]$$

where the division and multiplication indicated with "#" in the last term is elementwise.

Step 3: M-Step

In the M-step $\mathbf{y}^{(r)}$ is used to obtain the next approximation of the ML estimate of $\boldsymbol{\mu}$,

$$\boldsymbol{\mu}^{(r+1)}=\mathbf{y}^{(r)}-\mathbf{A}_C\left(\mathbf{A}_C'\mathbf{D}_{\mathbf{y}^{(r)}}^{-1}\mathbf{A}_C\right)^{-1}\mathbf{g}\left(\mathbf{y}^{(r)}\right)+o\left(\left\|\mathbf{y}^{(r)}-\boldsymbol{\mu}\right\|\right).$$

The next approximation of $\widehat{\boldsymbol{\pi}}$ is $\boldsymbol{\pi}^{(r+1)}=\dfrac{\boldsymbol{\mu}^{(r+1)}}{970}$, $r=0,1,2,\ldots$.

The EM algorithm iterates between Step 2 and Step 3 until covergence is attained.

Table 5.9 gives values at different steps of the algorithm.

TABLE 5.9: Values at different steps of the EM algorithm for the $\{CS,CP\}$ model.

		$r=0$		$r=1$		$r=2$		$r=10$	
		M-Step	E-Step	M-Step	E-Step	M-Step	E-Step	M-Step	E-Step
Cell	$\mathbf{y}^A$	$100\boldsymbol{\pi}^{(0)}$	$\mathbf{y}^{(0)}$	$100\boldsymbol{\pi}^{(1)}$	$\mathbf{y}^{(1)}$	$100\boldsymbol{\pi}^{(2)}$	$\mathbf{y}^{(2)}$	$100\boldsymbol{\pi}^{(10)}$	$\mathbf{y}^{(10)}$
111	3	0.3682	4.3400	0.4802	4.5030	0.4919	4.5468	0.4963	4.5632
112	176	24.6668	246.8579	25.4165	246.8436	25.4201	246.8343	25.4203	246.8280
121	4	0.6109	7.4363	0.7338	7.5561	0.7513	7.5894	0.7579	7.6031
122	293	40.9276	376.4384	38.8409	376.3140	38.8230	376.3048	38.8208	376.3106
211	17	2.3794	25.6600	2.7148	25.4970	2.6883	25.4532	2.6787	25.4368
212	197	27.5507	276.1421	28.3988	276.1564	28.4099	276.1657	28.4150	276.1720
221	2	0.2780	3.5637	0.2980	3.4439	0.2953	3.4106	0.2939	3.3969
222	23	3.2185	29.5616	3.1170	29.6860	3.1202	29.6952	3.1171	29.6894

Maximum likelihood estimation under constraints

Let $\mathbf{y}^{A\prime}=(3,176,4,293,17,197,2,23)$ and $\mathbf{y}^{B\prime}=(10,150,5,90)$ be the observed frequency vectors for the 715 fully and 255 partially classified cases respectively with $E\left(\mathbf{Y}^{A}\right)=\boldsymbol{\mu}^{A}$ and $E\left(\mathbf{Y}^{B}\right)=\boldsymbol{\mu}^{B}$. Furthermore $\mathbf{y}'_{obs}=\left(\mathbf{y}^{A\prime},\mathbf{y}^{B\prime}\right)$ and $\boldsymbol{\mu}'_{obs}=\left(\boldsymbol{\mu}^{A\prime},\boldsymbol{\mu}^{B\prime}\right)$. Assume a multinomial sampling scheme.

From Section 5.2.2 the ML estimate of $\boldsymbol{\mu}'_{obs}=\left(\boldsymbol{\mu}^{A\prime},\boldsymbol{\mu}^{B\prime}\right)$ subject to $\mathbf{g}\left(\boldsymbol{\mu}_{obs}\right)=\mathbf{0}$ is determined iteratively with

$$\widehat{\boldsymbol{\mu}}_{obs,c}=\mathbf{y}_{obs}-\left(\mathbf{G}_{\boldsymbol{\mu}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\right)'\left(\mathbf{G}_{\mathbf{y}_{obs}}\mathbf{V}_{\boldsymbol{\mu}_{obs}}\mathbf{G}'_{\boldsymbol{\mu}_{obs}}\right)^{-1}\mathbf{g}\left(\mathbf{y}_{obs}\right)+o\left(\left\|\mathbf{y}_{obs}-\boldsymbol{\mu}_{obs}\right\|\right) \tag{74}$$

where

$$\mathbf{g}\left(\boldsymbol{\mu}_{obs}\right)=\begin{pmatrix}\mathbf{g}_1\left(\boldsymbol{\mu}^{A}\right)\\ \mathbf{g}_2\left(\boldsymbol{\mu}_{obs}\right)\end{pmatrix}=\begin{pmatrix}\mathbf{P}\log\boldsymbol{\mu}^{A}\\ \frac{1}{715}(1,1)\otimes\mathbf{I}_4\boldsymbol{\mu}_{obs} \quad -\frac{1}{255}\mathbf{I}_4\boldsymbol{\mu}_{obs}\end{pmatrix}$$

$$\mathbf{G}_{\boldsymbol{\mu}_{obs}}=\frac{\partial\mathbf{g}\left(\boldsymbol{\mu}_{obs}\right)}{\partial\boldsymbol{\mu}_{obs}}=\begin{pmatrix}\frac{\partial\mathbf{g}_1\left(\boldsymbol{\mu}^{A}\right)}{\partial\boldsymbol{\mu}_{obs}}\\ \frac{\partial\mathbf{g}_2\left(\boldsymbol{\mu}_{obs}\right)}{\partial\boldsymbol{\mu}_{obs}}\end{pmatrix}=\begin{pmatrix}\mathbf{P}\mathbf{D}^{-1}_{\boldsymbol{\mu}^{A}} & \mathbf{0}_{8\times4}\\ \frac{1}{715}\mathbf{1}'_2\otimes\mathbf{I}_4 & -\frac{1}{255}\mathbf{I}_4\end{pmatrix},$$

$$\mathbf{G}_{\mathbf{y}_{obs}}=\frac{\partial\mathbf{g}\left(\boldsymbol{\mu}_{obs}\right)}{\partial\boldsymbol{\mu}_{obs}}\Big|_{\boldsymbol{\mu}_{obs}=\mathbf{y}_{obs}}\text{ and}$$

$$\mathbf{V}_{\boldsymbol{\mu}_{obs}}=\mathbf{D}_{\boldsymbol{\mu}_{obs}}-\frac{1}{970}\boldsymbol{\mu}_{obs}\boldsymbol{\mu}'_{obs}.$$

The ML estimates of the cell probabilities in the incomplete contingency table are the elements of $\frac{\widehat{\boldsymbol{\mu}}^{A}_{c}}{n^{A}}$ and are the same as those obtained with the EM algorithm.

Table 5.10 gives the ML estimates of cell probabilities obtained under constraints when using only the 715 fully classified cases and when using all 970 counts. The standard errors are also given.

TABLE 5.10: ML estimates obtained under constraints for the $\{CS,CP\}$ model.

	$n=715$		$n=970$	
	Estimate	Std Error	Estimate	Std Error
π_{111}	0.0037	0.0014	0.0050	0.0014
π_{112}	0.2467	0.0160	0.2542	0.0153
π_{121}	0.0061	0.0023	0.0076	0.0022
π_{122}	0.4093	0.0183	0.3882	0.0159
π_{211}	0.0238	0.0054	0.0268	0.0050
π_{212}	0.2755	0.0166	0.2842	0.0158
π_{221}	0.0028	0.0008	0.0029	0.0008
π_{222}	0.0322	0.0064	0.0312	0.0063

5.3 CONCLUSION

This dissertation has illustrated maximum likelihood estimation procedures for a number of generalized linear models for categorical data. The results obtained with the method under constraints are the same as those obtained with the more generally used Newton-Raphson, Fisher scoring and EM algorithms. The advantage of the method under constraints is that it is computationally less intensive and also more flexible to incorporate different models.

In this chapter the method was further developed to determine maximum likelihood estimates for loglinear models when the contingency table is incomplete and the missing data mechanism is ignorable. This illustrates the elegance with which the method under constraints can be applied.

This opens up new opportunities for the study of maximum likelihood estimation. This includes models for incomplete data when the missing data mechanism is ignorable, such as logistic regression and analysis of variance. Furthermore the same models for incomlete data can be studied when the missing data mechanism is not ignorable.

6 REFERENCES

Agresti, A. (1984). *Analysis of Ordinal Categorical Data.* New York: Wiley.

Agresti, A. (1990). *Categorical Data Analysis.* New York: Wiley.

Barndorff-Nielsen, O. (1978). *Information and Exponential Families in Statistical Theory.* Chichester: Wiley.

Cornfield, J. (1962). Joint dependence of risk of coronary heart disease on serum cholesterol and systolic blood pressure: a discriminant function analysis. *Fed. Proc.*, **21**, Supplement No. 11: 58-61.

Crowther, N.A.S. and Matthews, G.B. (1998). A maximum likelihood estimation procedure for the generalized linear model with restrictions. *South African Statist J.*, **32**, 119-144.

Dempster, A.P., Laird, N.M., and Rubin, D.B. (1977). Maximum likelihood for incomplete data via the EM algorithm (with discussion), *J. Roy. Statist. Soc.* **B39**, 1-38.

Goodman, L.A. (1979). Multiplicative models for square contingency tables with ordered categories. *Biometrika*, **66**, 413-418.

Lipsitz, S.R., Parzen, M., and Molenberghs, G. (1998). Obtaining the maximum likelihood estimates in incomplete $R \times C$ contingency tables using a Poisson generalized linear model. *Journal of Computational and Graphical Statistics*, **7**, 356-376.

Little, R.J.A., Rubin, D.B. (1987). *Statistical Analysis with Missing Data.* New York: Wiley.

Matthews, G.B. (1995). *Maximum likelihood estimation when modelling in tems of constraints.* Ph.D.: University of Pretoria.

Matthews, G.B. and Crowther, N.A.S. (1995). A maximum likelihood estimation procedure when modelling in terms of constraints. *South African Statist J.*, **29**, 29-50.

Nelder, J. and Wedderburn, R.W.M. (1972). Generalized linear models. *J. Roy. Statist. Soc.* **A135**, 370-384.

Pugh, M.D. (1983). Contributory fault and rape convictions: Loglinear models for blaming the victim. *Social Psychology Quarterly*, **46**, 233-242.

Rao, C.R. (1972). *Linear Statistical Inference and its Applications.* New York: Wiley.

Schafer, J.L. (1997). *Analysis of incomplete multivariate data. London*: Chapman &Hall.

Tomizawa, S. (1987). Decompositions for 2-ratios-parameter symmetry model in square contingency tables with ordered categories. *Biometrical J.*, **29**, 45-55.

Tomizawa, S. (1990). Another linear diagonals-parameter symmetry mode; for square contingency tables with ordered categories. *South African Statist. J.*, **24**, 117-125.

7 APPENDIX

The IML programs for examples are given in the Appendix and appear under the appropriate chapter heading and example number.

CHAPTER 2

EXAMPLE 2.1

```
proc iml; reset nolog;
y={80, 15, 5};
b={80,0.1875};
diff=1;
j=0;
do while (diff>0.000001); j=j+1;
  q=j(2,1,1);
  q[1]=-(1+b[2]+b[2]*b[2])+y[+]/b[1];
  q[2]=-b[1]*(1+2*b[2])+(y[2]+2*y[3])/b[2];
  H=j(2,2,1);
  H[1,1]=-y[+]/(b[1]*b[1]); H[1,2]=-(1+2*b[2]);
  H[2,1]=H[1,2]; H[2,2]=-2*b[1]-(y[2]+2*y[3])/(b[2]*b[2]);
  b1=b-inv(H)*q;
  diff=(b-b1)`*(b-b1);
  b=b1;
end;
m=j(3,1,0);
m[1]=b[1]; m[2]=b[1]*b[2]; m[3]=b[1]*b[2]*b[2];
print j b m;
```

EXAMPLE 2.2

```
proc iml; reset nolog;
y={80, 15, 5};
x={1 0, 1 1, 1 2};
b=ginv(x`*x)*x`*y;
b={80,0.1875};
diff=1;
j=0;
do while (diff>0.000001); j=j+1;
  m=exp(x*b);
  b1=b+ginv(x`*diag(m)*x)*x`*(y-m);
  diff=sqrt((b-b1)`*(b-b1));
  b=b1;
end;
m=exp(x*b); print j b m;
```

EXAMPLE 2.3

```
proc iml; reset nolog;
y={80, 15, 5}; ybegin=y;
b={80,0.1875};
diff=1;
j=0;
do while (diff>0.000001); j=j+1;
  q=j(2,1,1);
  q[1]=-(1+b[2]+b[2]*b[2])+ybegin[+]/b[1];
  q[2]=-b[1]*(1+2*b[2])+(ybegin[2]+2*ybegin[3])/b[2];
  Inf=j(2,2,1);
  Inf[1,1]=y[+]/(b[1]*b[1]); Inf[1,2]=(1+2*b[2]);
  Inf[2,1]=Inf[1,2]; Inf[2,2]=2*b[1]+(y[2]+2*y[3])/(b[2]*b[2]);
  b1=b+inv(Inf)*q;
  diff=sqrt((b-b1)`*(b-b1));
  b=b1;
  y[1]=b[1]; y[2]=b[1]*b[2]; y[3]=b[1]*b[2]*b[2];
end;
m=j(3,1,0); m[1]=b[1]; m[2]=b[1]*b[2]; m[3]=b[1]*b[2]*b[2];
print j b m;
```

EXAMPLE 2.9

```
proc iml; reset nolog;
Gm=j(1,3,0);  Gy=j(1,3,0);
y={80,15,5}; ybegin=y; m=y; muhat=y;
i=0; j=0;
diff1=1; diff2=1;
do while (diff1>0.000001);
i=i+1; j=0;
diff2=1;
Dm=diag(m);
Gm[1]=m[3]; Gm[2]=-2*m[2]; Gm[3]=m[1];
y=ybegin;
  do while (diff2>0.000001);
  j=j+1;
  g=y[1]*y[3]-y[2]*y[2];
  Gy[1]=y[3]; Gy[2]=-2*y[2]; Gy[3]=y[1];
  muhat=y-(Gm*Dm)`*ginv(Gy*Dm*Gm`)*g;
  diff2=sqrt((muhat-y)`*(muhat-y));
  y=muhat;
  end;
diff1=sqrt((muhat-m)`*(muhat-m));
m=muhat;
end;
print i j m;
```

EXAMPLE 2.10

```
proc iml; reset nolog;
Gy=j(1,3,0);
y={80,15,5};
j=0;
diff1=1;
do while (diff1>0.000001);
  j=j+1;
  Gy[1]=1/y[1]; Gy[2]=-2/y[2]; Gy[3]=1/y[3];
  GmDm={1 -2 1};
  g=log(y[1]*y[3]/(y[2]*y[2]));
  muhat=y-GmDm`*ginv(Gy*GmDm`)*g;
  diff1=sqrt((muhat-y)`*(muhat-y));
  y=muhat;
end;
print j y;
```

or

```
proc iml; reset nolog;
y={80, 15, 5};      m=y;
x={1 0, 1 1, 1 2};
p=i(3)-x*ginv(x`*x)*x`;
diff=1;
j=0;
do while (diff>0.000001); j=j+1;
  idy=inv(diag(y));
  muhat=y-p*ginv(p*idy*p)*p*log(y);
  diff=sqrt((muhat-y)`*(muhat-y));
  y=muhat;
end;
print j muhat;
```

EXAMPLE 2.11

```
proc iml;
reset nolog;
yobs={125,18,20,34}; mu=yobs;
x={1 -1 -1 -3,
   0  1 -1  0};
diff=1; r=0;
do while (diff>1e-10);
r=r+1;
v=diag(mu)-(1/197)#mu*mu`;
mu1=yobs-(x*v)`*ginv(x*v*x`)*x*yobs;
diff=(mu-mu1)`*(mu-mu1);
mu=mu1;
end;
print r mu;
pi=mu[4]/197*4; print pi;
```

CHAPTER 3

EXAMPLE 3.1 : Proc Catmod for reduced Loglinear model

```
data verdict;
input m v f n @@;
cards;
1 1 1 42   1 1 2 23
1 2 1  4   1 2 2 11
2 1 1 79   2 1 2 65
2 2 1 12   2 2 2 41
3 1 1 32   3 1 2 17
3 2 1  8   3 2 2 24
;
proc catmod;
weight n;
model m*v*f=_response_/ml nogls noprofile pred=freq;
loglin m v f m*v v*f;
run;
```

CATMOD PROCEDURE

Response: M*V*F	Response Levels (R)= 12
Weight Variable: N	Populations (S)= 1
Data Set: VERDICT	Total Frequency (N)= 358
Frequency Missing: 0	Observations (Obs)= 12

MAXIMUM-LIKELIHOOD ANALYSIS

Iteration	Sub Iteration	-2 Log Likelihood	Convergence Criterion
0	0	1779.1932	1.0000
1	0	1621.7743	0.0885
2	0	1590.2147	0.0195
3	0	1590.0846	0.0000819
4	0	1590.0846	1.2263E-8
5	0	1590.0846	4.29E-16

Parameter Estimates

Iteration	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0
1	-0.3296	0.6508	0.4413	-0.0112	-0.0223	0.3212	0.2793
2	-0.4090	0.6050	0.5376	-0.1947	0.2463	0.007680	0.3846
3	-0.4219	0.6068	0.5518	-0.1941	0.2509	0.0178	0.3823
4	-0.4221	0.6067	0.5520	-0.1941	0.2512	0.0178	0.3823
5	-0.4221	0.6067	0.5520	-0.1941	0.2512	0.0178	0.3823

MAXIMUM-LIKELIHOOD ANALYSIS-OF-VARIANCE TABLE

Source	DF	Chi-Square	Prob
M	2	55.92	0.0000
V	1	56.51	0.0000
F	1	8.50	0.0036
M*V	2	8.60	0.0135
V*F	1	32.99	0.0000
LIKELIHOOD RATIO	4	2.81	0.5898

ANALYSIS OF MAXIMUM-LIKELIHOOD ESTIMATES

Effect	Parameter	Estimate	Standard Error	Chi-Square	Prob
M	1	-0.4221	0.1062	15.81	0.0001
	2	0.6067	0.0811	55.92	0.0000
V	3	0.5520	0.0734	56.51	0.0000
F	4	-0.1941	0.0666	8.50	0.0036
M*V	5	0.2512	0.1062	5.60	0.0180
	6	0.0178	0.0811	0.05	0.8266
V*F	7	0.3823	0.0666	32.99	0.0000

MAXIMUM-LIKELIHOOD PREDICTED VALUES FOR RESPONSE FUNCTIONS AND FREQUENCIES

					-------Observed-------		--------Predicted------		
Sample	M	V	F	Function Number	Function	Standard Error	Function	Standard Error	Residual
1				1	0.55961579	0.25588316	0.46056655	0.22902508	0.09904924
				2	-0.0425596	0.29179604	0.08408898	0.23545775	-0.1266486
				3	-1.7917595	0.54006172	-1.9103652	0.3908212	0.11860574
				4	-0.7801586	0.36410954	-0.7576857	0.31291637	-0.0224729
				5	1.19139402	0.23307701	1.25599258	0.20979113	-0.0645986
				6	0.99633344	0.2388541	0.87951501	0.21679525	0.11681843
				7	-0.6931472	0.35355339	-0.6481235	0.32394827	-0.0450237
				8	0.53551824	0.25701539	0.50455601	0.22387033	0.03096223
				9	0.28768207	0.27003086	0.17799958	0.2397416	0.10968249
				10	-0.3448405	0.31700189	-0.198478	0.24589408	-0.1463625
				11	-1.0986123	0.40824829	-1.1526795	0.23414645	0.05406722
	1	1	1	F1	42	6.0887294	38.5465116	4.76034564	3.45348837
	1	1	2	F2	23	4.63921829	26.4534884	3.57260801	-3.4534884
	1	2	1	F3	4	1.98879543	3.6	1.11274377	0.4
	1	2	2	F4	11	3.26527352	11.4	2.95150493	-0.4
	2	1	1	F5	79	7.84646666	85.3953488	7.04763999	-6.3953488
	2	1	2	F6	65	7.29371812	58.6046512	5.80126523	6.39534884
	2	2	1	F7	12	3.4055492	12.72	2.77929216	-0.72
	2	2	2	F8	41	6.02531902	40.28	5.58608559	0.72
	3	1	1	F9	32	5.39811678	29.0581395	4.13758176	2.94186047
	3	1	2	F10	17	4.02402006	19.9418605	3.04155594	-2.9418605
	3	2	1	F11	8	2.79664604	7.68	1.88314117	0.32
	3	2	2	F12	24	4.73191943	24.32	4.32421628	-0.32

EXAMPLE 3.1 : ML Estimation with the Newton-Raphson algorithm

```
proc iml;
reset nolog;
y={42, 23, 4, 11, 79, 65, 12, 41, 32, 17, 8, 24};
x={1     1     0     1     1     1     0     1,
   1     1     0     1    -1     1     0    -1,
   1     1     0    -1     1    -1     0    -1,
   1     1     0    -1    -1    -1     0     1,
   1     0     1     1     1     0     1     1,
   1     0     1     1    -1     0     1    -1,
   1     0     1    -1     1     0    -1    -1,
   1     0     1    -1    -1     0    -1     1,
   1    -1    -1     1     1    -1    -1     1,
   1    -1    -1     1    -1    -1    -1    -1,
   1    -1    -1    -1     1     1     1    -1,
   1    -1    -1    -1    -1     1     1     1};
m=y;
b=ginv(x`*x)*x`*log(m);
m=exp(x*b);
diff=1; i=0;

do while (diff>1e-15); i=i+1;
b1=b+ginv(x`*diag(m)*x)*x`*(y-m);
diff=(b-b1)`*(b-b1);
b=b1;
m=exp(x*b);
end;

sebhat=sqrt(vecdiag(ginv(x`*diag(m)*x)));
print i b sebhat;
```

EXAMPLE 3.1 : ML Estimation under constraints

```
proc iml;
reset nolog;
y={42, 23, 4, 11, 79, 65, 12, 41, 32, 17, 8, 24}; ybegin=y;
m=y;
x={1    1    0    1    1       1       0       1       0    1    1    0,
   1    1    0    1   -1       1       0      -1       0   -1   -1    0,
   1    1    0   -1    1      -1       0       1       0   -1   -1    0,
   1    1    0   -1   -1      -1       0      -1       0    1    1    0,
   1    0    1    1    1       0       1       0       1    1    0    1,
   1    0    1    1   -1       0       1       0      -1   -1    0   -1,
   1    0    1   -1    1       0      -1       0       1   -1    0   -1,
   1    0    1   -1   -1       0      -1       0      -1    1    0    1,
   1   -1   -1    1    1      -1      -1      -1      -1    1   -1   -1,
   1   -1   -1    1   -1      -1      -1       1       1   -1    1    1,
   1   -1   -1   -1    1       1       1      -1      -1   -1    1    1,
   1   -1   -1   -1   -1       1       1       1       1    1   -1   -1};

c={0 0 0 0 0 0 0 1 0 0 0 0,
   0 0 0 0 0 0 0 0 1 0 0 0,
   0 0 0 0 0 0 0 0 0 0 1 0,
   0 0 0 0 0 0 0 0 0 0 0 1};
acp=c*ginv(x`*x)*x`; ac=acp`;
gy=ac`*log(y);
wald=gy`*ginv(ac`*diag(1/y)*ac)*gy;
diff=1; i=0;

do while (diff>1e-10);
y1=y-ac*ginv(ac`*diag(1/y)*ac)*ac`*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;
bhat=ginv(x`*x)*x`*log(y);
print bhat;

v=diag(y)-y*y`/y[+];
v=diag(y);
covy=v-ac*ginv(ac`*diag(1/y)*ac)*ac`;
se_y=sqrt(vecdiag(covy));
print y se_y;

est1=diag(1/y)*covy*diag(1/y);
cov_bhat=ginv(x`*x)*x`*est1*x*ginv(x`*x);
se_bhat=sqrt(vecdiag(cov_bhat));
print bhat se_bhat;

chi2=sum((ybegin-y)#(ybegin-y)/y);
dev=2#ybegin`*log(ybegin/y); print dev;
print chi2 dev wald;
```

EXAMPLE 3.2 : Proc Logistic and Proc Genmod

```
data blood;
input pressure ypres yabs;
events=ypres;
trials=ypres+yabs;
cards;
111.5  3 153
121.5 17 235
131.5 12 272
141.5 16 255
151.5 12 127
161.5  8  77
176.5 16  83
191.5  8  35
;
proc logistic;
model events/trials=pressure;
run;

proc genmod;
model events/trials=pressure/link=logit dist=bin;
run;
```

```
                    The LOGISTIC Procedure
  Data Set: WORK.BLOOD
  Response Variable (Events): EVENTS
  Response Variable (Trials): TRIALS
  Number of Observations: 8
  Link Function: Logit

                    Response Profile
                Ordered  Binary
                  Value  Outcome      Count
                      1  EVENT           92
                      2  NO EVENT      1237

   Model Fitting Information and Testing Global Null Hypothesis BETA=0
                          Intercept
              Intercept       and
  Criterion     Only      Covariates    Chi-Square for Covariates
  AIC          670.831      648.718          .
  SC           676.024      659.102          .
  -2 LOG L     668.831      644.718      24.113 with 1 DF (p=0.0001)
  Score           .            .         26.556 with 1 DF (p=0.0001)

              Analysis of Maximum Likelihood Estimates
            Parameter Standard    Wald       Pr >    Standardized     Odds
  Variable DF Estimate   Error  Chi-Square Chi-Square   Estimate      Ratio
  INTERCPT 1   -6.0820  0.7243    70.5098     0.0001          .          .
  PRESSURE 1    0.0243 0.00484    25.2523     0.0001     0.269349    1.025
```

Association of Predicted Probabilities and Observed Responses

Concordant	= 56.8%	Somers' D	= 0.273
Discordant	= 29.5%	Gamma	= 0.316
Tied	= 13.7%	Tau-a	= 0.035
(113804 pairs)		c	= 0.636

The GENMOD Procedure

Model Information

Description	Value
Data Set	WORK.BLOOD
Distribution	BINOMIAL
Link Function	LOGIT
Dependent Variable	EVENTS
Dependent Variable	TRIALS
Observations Used	8
Number Of Events	92
Number Of Trials	1329

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	6	5.9092	0.9849
Scaled Deviance	6	5.9092	0.9849
Pearson Chi-Square	6	6.2899	1.0483
Scaled Pearson X2	6	6.2899	1.0483
Log Likelihood	.	-322.3590	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	1	-6.0820	0.7243	70.5076	0.0001
PRESSURE	1	0.0243	0.0048	25.2513	0.0001

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
SCALE	0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

EXAMPLE 3.2 : ML Estimation using the Newton-Raphson algorithm

```
proc iml;
reset nolog;
x={1 111.5, 1  121.5, 1  131.5, 1  141.5, 1  151.5, 1 161.5, 1  176.5, 1 191.5};
y={3 153,   17 235,   12 272,   16 255,   12 127,   8 77,   16 83,   8 35};
xr=nrow(x);

yi=y[,1]; yi0=yi;
ni=y[,1]+y[,2];
pi=yi/ni;
e=j(xr,1,1);

logit=log(pi/(e-pi));
bhat=ginv(x`*x)*x`*logit;

diff=1;  i=0;
do while (diff>1e-10); i=i+1;
  pi=exp(x*bhat)/(e+exp(x*bhat));
  var=ni#pi#(e-pi); v=diag(var); ivar=1/var;
  yi1=ni#pi;
  bhat1=bhat+ginv(x`*v*x)*x`*(yi-yi1);
  diff=(bhat-bhat1)`*(bhat-bhat1);
  bhat=bhat1;
end;

sebhat=sqrt(vecdiag(ginv(x`*v*x)));
print i bhat sebhat;
```

EXAMPLE 3.2 : ML Estimation under constraints

```
proc iml;
reset nolog;
x={1 111.5, 1  121.5, 1  131.5, 1  141.5, 1  151.5, 1 161.5, 1  176.5, 1 191.5};
y={3 153,   17 235,  12 272,  16 255,  12 127,  8 77,    16 83,   8 35};
xr=nrow(x);
p=i(xr)-x*ginv(x`*x)*x`;

yi=y[,1]; yi0=yi;
ni=y[,1]+y[,2];
e=j(xr,1,1);

diff=1; i=0;
do while (diff>1e-10); i=i+1;
  pi=yi/ni;
  logit=log(pi/(e-pi));
  var=ni#pi#(e-pi); v=diag(var); ivar=1/var;
  g=p*diag(ivar);
  yi1=yi-p*ginv(p*diag(ivar)*p)*p*logit;
  diff=(yi1-yi)`*(yi1-yi);
  yi=yi1;
end;

bhat=ginv(x`*x)*x`*logit;
sebhat=sqrt(vecdiag(ginv(x`*v*x)));
print i yi0 yi1;
print bhat sebhat;

pi=yi/ni;
var=ni#pi#(e-pi); v=diag(var); iv=diag(1/var);
covy=v-p*ginv(p*iv*p)*p;
se_y=sqrt(vecdiag(covy));

est1=iv*covy*iv;
cov_bhat=ginv(x`*x)*x`*est1*x*ginv(x`*x);
se_bhat=sqrt(vecdiag(cov_bhat));
print bhat se_bhat;

chi2=sum((yi0-yi)#(yi0-yi)/yi)+sum((yi-yi0)#(yi-yi0)/(ni-yi));
dev=2#yi0`*log(yi0/yi)+2#(ni-yi0)`*log((ni-yi0)/(ni-yi));

print chi2 dev;
```

EXAMPLE 3.3: Proc Catmod, Proc Logistic and Proc Genmod

```
data verdict;
input m v f n @@;
cards;
1 1 1 42   1 1 2 23
1 2 1  4   1 2 2 11
2 1 1 79   2 1 2 65
2 2 1 12   2 2 2 41
3 1 1 32   3 1 2 17
3 2 1  8   3 2 2 24
;
proc catmod;
weight n;
model v=m f/ml nogls noprofile;
run;

data verdict;
input m1 m2 f1 guilty n_guilty @@;
events=guilty;
trials=guilty+n_guilty;
cards;
 1  0  1 42 4
 0  1  1 79 12
-1 -1  1 32 8
 1  0 -1 23 11
 0  1 -1 65 41
-1 -1 -1 17 24
;
proc logistic;
model events/trials=m1 m2 f1;
run;

proc genmod;
model events/trials=m1 m2 f1/link=logit dist=bin;
run;
```

The CATMOD Procedure

Data Summary

Response	v	Response Levels	2
Weight Variable	n	Populations	6
Data Set	VERDICT	Total Frequency	358
Frequency Missing	0	Observations	12

Maximum Likelihood Analysis

Iteration	Sub Iteration	-2 Log Likelihood	Convergence Criterion	Parameter Estimates 1	2	3	4
0	0	496.29338	1.0000	0	0	0	0
1	0	382.97715	0.2283	0.8530	0.3128	0.1136	0.5613
2	0	378.42388	0.0119	1.0465	0.4339	0.1224	0.7443
3	0	378.34289	0.000214	1.0776	0.4548	0.1211	0.7732
4	0	378.34285	1.0572E-7	1.0783	0.4553	0.1210	0.7739
5	0	378.34285	2.749E-14	1.0783	0.4553	0.1210	0.7739

Maximum likelihood computations converged.

Maximum Likelihood Analysis of Variance

Source	DF	Chi-Square	Pr > ChiSq
Intercept	1	53.91	<.0001
m	2	8.38	0.0152
f	1	32.61	<.0001
Likelihood Ratio	2	0.26	0.8801

Analysis of Maximum Likelihood Estimates

Parameter		Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept		1.0783	0.1469	53.91	<.0001
m	1	0.4553	0.2226	4.18	0.0408
	2	0.1210	0.1717	0.50	0.4809
f	1	0.7739	0.1355	32.61	<.0001

The LOGISTIC Procedure

Data Set: WORK.VERDICT
Response Variable (Events): EVENTS
Response Variable (Trials): TRIALS
Number of Observations: 6
Link Function: Logit

Response Profile

Ordered Value	Binary Outcome	Count
1	EVENT	258
2	NO EVENT	100

Model Fitting Information and Testing Global Null Hypothesis BETA=0

Criterion	Intercept Only	Intercept and Covariates	Chi-Square for Covariates
AIC	426.100	386.343	.
SC	429.981	401.865	.
-2 LOG L	424.100	378.343	45.758 with 3 DF (p=0.0001)
Score	.	.	43.571 with 3 DF (p=0.0001)

Analysis of Maximum Likelihood Estimates

Variable	DF	Parameter Estimate	Standard Error	Wald Chi-Square	Pr > Chi-Square	Standardized Estimate	Odds Ratio
INTERCPT	1	1.0783	0.1469	53.9107	0.0001	.	.
M1	1	0.4553	0.2226	4.1840	0.0408	0.168558	1.577
M2	1	0.1210	0.1717	0.4968	0.4809	0.054754	1.129
F1	1	0.7739	0.1355	32.6053	0.0001	0.427234	2.168

Association of Predicted Probabilities and Observed Responses

Concordant	= 62.6%	Somers' D	= 0.434
Discordant	= 19.2%	Gamma	= 0.531
Tied	= 18.2%	Tau-a	= 0.175
(25800 pairs)		c	= 0.717

The GENMOD Procedure

Model Information

Description	Value
Data Set	WORK.VERDICT
Distribution	BINOMIAL
Link Function	LOGIT
Dependent Variable	EVENTS
Dependent Variable	TRIALS
Observations Used	6
Number Of Events	258
Number Of Trials	358

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	2	0.2554	0.1277
Scaled Deviance	2	0.2554	0.1277
Pearson Chi-Square	2	0.2552	0.1276
Scaled Pearson X2	2	0.2552	0.1276
Log Likelihood	.	-189.1714	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	1	1.0783	0.1469	53.9106	0.0001
M1	1	0.4553	0.2226	4.1840	0.0408
M2	1	0.1210	0.1717	0.4968	0.4809
F1	1	0.7739	0.1355	32.6053	0.0001
SCALE	0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

EXAMPLE 3.3 : ML Estimation under constraints and using the Newton-Raphson algorithm

```
proc iml;
reset nolog;
x={1  1  0  1,
   1  0  1  1,
   1 -1 -1  1,
   1  1  0 -1,
   1  0  1 -1,
   1 -1 -1 -1};
y={42 4, 79 12, 32 8, 23 11, 65 41, 17 24};
xr=nrow(x);

yi=y[,1]; yi0=yi;
ni=y[,1]+y[,2];
pi=yi/ni; pi0=pi;
e=j(xr,1,1);

print 'ML ESTIMATION SUBJECT TO CONSTRAINTS';
p=i(xr)-x*ginv(x`*x)*x`;
diff=1; i=0;
do while (diff>1e-10); i=i+1;
  pi=yi/ni;
  logit=log(pi/(e-pi));
  var=ni#pi#(e-pi); v=diag(var); ivar=1/var;
  g=p*diag(ivar);
  yi1=yi-p*ginv(p*diag(ivar)*p)*p*logit;
  diff=(yi1-yi)`*(yi1-yi);
  yi=yi1;
end;

bhat=ginv(x`*x)*x`*logit;
sebhat=sqrt(vecdiag(ginv(x`*v*x)));
print i yi0 yi1; print bhat sebhat;

pi=yi/ni; var=ni#pi#(e-pi); v=diag(var); iv=diag(1/var);
covy=v-p*ginv(p*iv*p)*p; se_y=sqrt(vecdiag(covy));

est1=iv*covy*iv;
cov_bhat=ginv(x`*x)*x`*est1*x*ginv(x`*x); se_bhat=sqrt(vecdiag(cov_bhat));
print bhat se_bhat;

chi2=sum((yi0-yi)#(yi0-yi)/yi)+sum((yi-yi0)#(yi-yi0)/(ni-yi));
dev=2#yi0`*log(yi0/yi)+2#(ni-yi0)`*log((ni-yi0)/(ni-yi));
print chi2 dev;

print 'NEWTON-RAPHSON ALGORITHM';
logit=log(pi0/(e-pi0)); bhat=ginv(x`*x)*x`*logit;

diff=1;  i=0;
do while (diff>1e-10); i=i+1;
  pi=exp(x*bhat)/(e+exp(x*bhat));
  var=ni#pi#(e-pi); v=diag(var); ivar=1/var;
  yi1=ni#pi;
  bhat1=bhat+ginv(x`*v*x)*x`*(yi-yi1);
  diff=(bhat-bhat1)`*(bhat-bhat1);
  bhat=bhat1;
end;

sebhat=sqrt(vecdiag(ginv(x`*v*x)));
print i bhat sebhat;
```

CHAPTER 4

EXAMPLE 4.1

```
proc iml; reset nolog;
/*********************************************************/
/* Give the observed values of y from the square table    */
/*********************************************************/
y={50  45   8  18   8
   28 174  84 154  55
   11  78 110 223  96
   14 150 185 714 447
    3  42  72 320 411};
/* y={11607 100 366 124 87 13677 515 302 172 225 17819 270 63 176 286 10192}; */
/* y={1520 266 124 66 234 1512 432 78 117 362 1772 205 36 82 179 492}; */

y=y`; ybeg=y;

/*********************************************************/
/* Create C matrix for the test under constraints        */
/*********************************************************/
n=sqrt(nrow(y)); nn=n#(n-1)/2;
C=j(nn,n*n,0);
r=0;
do j=1 to (n-1);
k1begin=(j-1)*(n+1)+2; k1end=n*j;
lc=0;
  do k1=k1begin to k1end;
  lc=lc+1;
  r=r+1; k2=k1+(n-1)*lc;
  C[r,k1]=1; C[r,k2]=-1;
  end;
end;

/*************************************************/
/* 1 Test for CS model under constraints         */
/*************************************************/
print 'Model CS';
x=j(nn,1,1);
P=I(nn)-x*ginv(x`*x)*x`;
K=P*C;
diff=1;
i=0;

do while (diff>1e-10); i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(x`*x)*x`*C*log(y));
print delta chi2 g2;
print ybeg y;
```

```
/************************************************/
/* 2 Test for S model under constraints         */
/************************************************/
print 'Model S';
y=ybeg;
diff=1;
i=0;

do while (diff>1e-10);
i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-C`*ginv(C*Di*C`)*C*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
print chi2 g2;
print ybeg y;

/************************************************/
/* 3 Test for DPS model under constraints       */
/************************************************/
print 'Model DPS';
y=ybeg;

X=I(n-1);
do h=2 to n-1;
YY=I(n-h)||j(n-h,h-1,0);
X=X//YY;
free YY;
end; print X;

P=I(nn)-X*ginv(X`*X)*X`;
K=P*C;

diff=1;
i=0;

do while (diff>1e-10);
i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(X`*X)*X`*C*log(y));
print delta chi2 g2;
print ybeg y;
```

```
/*************************************************/
/* 4 Test for LDPS model under constraints       */
/*************************************************/
print 'Model LDPS';
y=ybeg;

X1=I(n-1);
do h=2 to n-1;
YY=I(n-h)||j(n-h,h-1,0);
X1=X1//YY;
free YY;
end;
L=1;
do h=2 to n-1;
L=L//h;
end;
X=X1*L; print X;

P=I(nn)-X*ginv(X`*X)*X`;
K=P*C;

diff=1;
i=0;

do while (diff>1e-10);
i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(X`*X)*X`*C*log(y));
print delta chi2 g2;
print ybeg y;


/*************************************************/
/* 5 Test for ALDPS model under constraints      */
/*************************************************/
print 'Model ALDPS';
y=ybeg;

X1=I(n-1);
do h=2 to n-1;
YY=I(n-h)||j(n-h,h-1,0);
X1=X1//YY;
free YY;
end;
L=1;
do h=2 to n-1;
L=L//h;
end;
```

```
X=j(2*n,1,n)-X1*L; print X;

P=I(nn)-X*ginv(X`*X)*X`;
K=P*C;

diff=1;
i=0;

do while (diff>1e-10);
i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(X`*X)*X`*C*log(y));
print delta chi2 g2;
print ybeg y;

/*************************************************/
/* 6 Test for 2RPS model under constraints       */
/*************************************************/
print 'Model 2RPS';
y=ybeg;

X1=I(n-1);
do h=2 to n-1;
YY=I(n-h)||j(n-h,h-1,0);
X1=X1//YY;
free YY;
end;
L=0;
do h=1 to n-2;
L=L//h;
end;
X2=X1*L; X3=j(n*2,1,1); X=X3||X2; print X;

P=I(nn)-X*ginv(X`*X)*X`;
K=P*C;

diff=1; i=0;

do while (diff>1e-10); i=i+1;
Dy=diag(y); Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;
chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(X`*X)*X`*C*log(y));
print delta chi2 g2;
print ybeg y;
```

```
/*************************************************/
/* 7 Test for QS model under constraints         */
/*************************************************/
print 'Model QS';
y=ybeg;

free X;
plusi=I(n-1); mini=-plusi;
een=j(n-1,1,1); mineen=-een;
X=een||mini||mineen||plusi;

do k=1 to n-2;
nul=j(n-k-1,k,0);
plusi=I(n-k-1); mini=-plusi;
een=j(n-k-1,1,1); mineen=-een;
YY=nul||een||mini||nul||mineen||plusi;
X=X//YY;
free YY;
end;


P=I(nn)-X*ginv(X`*X)*X`;
K=P*C;

diff=1;
i=0;

do while (diff>1e-10);
i=i+1;
Dy=diag(y);
Di=inv(Dy);
y1=y-K`*ginv(K*Di*K`)*K*log(y);
diff=(y1-y)`*(y1-y);
y=y1;
end;

chi2=(ybeg-y)`*((1/y)#(ybeg-y));
g2=2*ybeg`*log(ybeg/y);
delta=exp(ginv(X`*X)*X`*C*log(y));
print delta chi2 g2;
print ybeg y;
```

CHAPTER 5

EXAMPLE 5.2 and EXAMPLE 5.3

```
/*********************************************************/
/* ML estimation of cell probabilities for incomplete    */
/* IxJ contingency tables if data is missing on either   */
/* categories and the missing data mecahnism is          */
/* ignorable                                             */
/*********************************************************/

proc iml; reset nolog;

/*********************************************************/
/* ENTER FREQUENCY VECTORS A, B and C:                   */
/* A: both row and column categories observed            */
/*    enter rowwise                                      */
/* B: row category observed and column category missing   */
/* C: column category observed and row category missing   */
/*********************************************************/

/***************/
/* Example 5.2 */
/***************/
A={392,55,76,38};
B={33,9};
C={31,7};

/***************/
/* Example 5.3 */
/***************/
A={287,39,38,18,6,4,91,22,23};
B={279,27,201};
C={59,18,26};

i=nrow(B);
j=nrow(C);
na=nrow(A); nb=i; nc=j;

y=A//B//C;
sy=y[+];
ya=y[1:na,]; yb=y[na+1:na+nb,]; yc=y[na+nb+1:na+nb+nc,];
som_ya=ya[+]; som_yb=yb[+]; som_yc=yc[+];
pa=ya/ya[+]; pb=yb/yb[+]; pc=yc/yc[+];
p=pa//pb//pc; tot=p[+];
p0=p; pbegin=p;

ij=i+j;
ej=J(1,j,1); ei=J(1,i,1);
i_i=I(i);    i_j=I(j);
i_ij=-I(ij);
c_row=i_i@ej;
c_col=ei@i_j;
g1=c_row//c_col;
G=g1||i_ij;
```

```
diff=1; t=0;

do while (diff>1e-20); t=t+1;

   pa=p[1:na,]; pb=p[na+1:na+nb,]; pc=p[na+nb+1:na+nb+nc,];
   cova=diag(pa)/som_ya-pa*pa`/som_ya;
   covb=diag(pb)/som_yb-pb*pb`/som_yb;
   covc=diag(pc)/som_yc-pc*pc`/som_yc;
   V=block(cova,covb,covc);

   p=pbegin;
     print g; print p;
   gp=G*p;
   pt=p-(G*V)`*ginv(G*V*G`)*gp;
   diff=(pt-p0)`*(pt-p0);
   p0=pt;
   p=pt;

end;

stderr=sqrt(vecdiag(v-(g*v)`*ginv(g*v*g`)*g*v));
print pt stderr;
```

EXAMPLE 5.2: GENMOD

```
data one;
input count p11 p12 p21 off;
cards;
392  561    0    0   0
 55    0  561    0   0
 76    0    0  561   0
 38 -561 -561 -561 561
 33   42   42    0   0
  9  -42  -42    0  42
 31   38    0   38   0
  7  -38    0  -38  38
;
proc genmod data=one;
model count=p11 p12 p21/dist=poi link=id offset=off noint;
run;
```

The GENMOD Procedure

Model Information

Description	Value
Data Set	WORK.ONE
Distribution	POISSON
Link Function	IDENTITY
Dependent Variable	COUNT
Offset Variable	OFF
Observations Used	8

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	5	0.1125	0.0225
Scaled Deviance	5	0.1125	0.0225
Pearson Chi-Square	5	0.1149	0.0230
Scaled Pearson X2	5	0.1149	0.0230
Log Likelihood	.	2642.6805	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	0	0.0000	0.0000	.	.
P11	1	0.6971	0.0187	1389.0323	0.0001
P12	1	0.0986	0.0124	63.5830	0.0001
P21	1	0.1358	0.0141	92.2715	0.0001
SCALE	0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

Lagrange Multiplier Statistics

Parameter	ChiSquare	Pr>Chi
Intercept	0.0309	0.8605

EXAMPLE 5.3: GENMOD

```
data one;
input count p11 p12 p13 p21 p22 p23 p31 p32 off;
cards;
287    528      0      0      0      0      0      0      0      0
39       0    528      0      0      0      0      0      0      0
38       0      0    528      0      0      0      0      0      0
18       0      0      0    528      0      0      0      0      0
6        0      0      0      0    528      0      0      0      0
4        0      0      0      0      0    528      0      0      0
91       0      0      0      0      0      0    528      0      0
22       0      0      0      0      0      0      0    528      0
23    -528   -528   -528   -528   -528   -528   -528   -528    528
279    507    507    507      0      0      0      0      0      0
27       0      0      0    507    507    507      0      0      0
201   -507   -507   -507   -507   -507   -507      0      0    507
59     103      0      0    103      0      0    103      0      0
18       0    103      0      0    103      0      0    103      0
26    -103   -103      0   -103   -103      0   -103   -103    103
;
proc genmod data=one;
model count=p11 p12 p13 p21 p22 p23 p31 p32/dist=poi link=id offset=off noint;
run;
```

The GENMOD Procedure

Model Information

Description	Value
Data Set	WORK.ONE
Distribution	POISSON
Link Function	IDENTITY
Dependent Variable	COUNT
Offset Variable	OFF
Observations Used	15

Criteria For Assessing Goodness Of Fit

Criterion	DF	Value	Value/DF
Deviance	7	36.0006	5.1429
Scaled Deviance	7	36.0006	5.1429
Pearson Chi-Square	7	36.8259	5.2608
Scaled Pearson X2	7	36.8259	5.2608
Log Likelihood	.	4471.6801	.

Analysis Of Parameter Estimates

Parameter	DF	Estimate	Std Err	ChiSquare	Pr>Chi
INTERCEPT	0	0.0000	0.0000	.	.
P11	1	0.4747	0.0174	748.0998	0.0001
P12	1	0.0701	0.0102	47.1384	0.0001
P13	1	0.0742	0.0107	47.7219	0.0001
P21	1	0.0327	0.0064	25.9330	0.0001
P22	1	0.0120	0.0045	6.9284	0.0085
P23	1	0.0087	0.0041	4.4891	0.0341
P31	1	0.2060	0.0158	169.1698	0.0001
P32	1	0.0558	0.0106	28.0104	0.0001
SCALE	0	1.0000	0.0000	.	.

NOTE: The scale parameter was held fixed.

Lagrange Multiplier Statistics

Parameter	ChiSquare	Pr>Chi
Intercept	0.0571	0.8111

EXAMPLE 5.3 (Fully Classified cases)

```
data wheeze;
input smoke status f @@;
cards;
1 1 287 1 2 39 1 3 38
2 1  18 2 2  6 2 3  4
3 1  91 3 2 22 3 3 23
;
proc catmod data=wheeze;
weight f;
model smoke*status= _response_/ml noprofile pred=prob;
loglin smoke status smoke*status;
run;
```

CATMOD PROCEDURE

MAXIMUM-LIKELIHOOD PREDICTED VALUES FOR RESPONSE FUNCTIONS AND PROBABILITIES

				Observed		Predicted		
Sample	SMOKE	STATUS	Function Number	Function	Standard Error	Function	Standard Error	Residual
1			1	2.523988	0.21670852	2.523988	0.2167086	0
			2	0.52806743	0.26290547	0.52806743	0.26290557	0
			3	0.50209194	0.26418564	0.50209194	0.26418573	0
			4	-0.2451225	0.31469639	-0.2451225	0.31469648	0
			5	-1.3437347	0.45841567	-1.3437347	0.45840475	-1.8143E-9
			6	-1.7491999	0.54173634	-1.7491999	0.5417358	0
			7	1.37536529	0.23338224	1.37536529	0.23338233	0
			8	-0.0444518	0.29821604	-0.0444518	0.29821615	0
	1	1	P1	0.54356061	0.02167697	0.54356061	0.02167697	0
	1	2	P2	0.07386364	0.01138245	0.07386364	0.01138246	0
	1	3	P3	0.0719697	0.01124706	0.0719697	0.01124706	0
	2	1	P4	0.03409091	0.00789715	0.03409091	0.00789715	0
	2	2	P5	0.01136364	0.00461275	0.01136364	0.00461261	0
	2	3	P6	0.00757576	0.0037735	0.00757576	0.0037735	0
	3	1	P7	0.17234848	0.01643654	0.17234848	0.01643655	0
	3	2	P8	0.04166667	0.00869632	0.04166667	0.00869633	0
	3	3	P9	0.04356061	0.00888298	0.04356061	0.00888298	0

EXAMPLE 5.4: Model {SPC}

```
proc iml; reset nolog;
yc={3,176,4,293,17,197,2,23}; som_yc=yc[+]; pc=yc/som_yc;
ym={10,150,5,90};             som_ym=ym[+]; pm=ym/som_ym;
G={1 0 0 0 1 0 0 0 -1  0  0  0,
   0 1 0 0 0 1 0 0  0 -1  0  0,
   0 0 1 0 0 0 1 0  0  0 -1  0,
   0 0 0 1 0 0 0 1  0  0  0 -1};

y=yc//ym;
p=pc//pm; p0=p; pbegin=p;

diff=1; t=0;

do while (diff>1e-20); t=t+1;
 pc=p[1:8,]; pm=p[9:12,];
 covc=diag(pc)/som_yc-pc*pc`/som_yc;
 covm=diag(pm)/som_ym-pm*pm`/som_ym;
 V=block(covc,covm);
 p=pbegin;
 gp=G*p;
 pt=p-(G*V)`*ginv(G*V*G`)*gp;
 diff=(pt-p0)`*(pt-p0);
 p0=pt; p=pt;
end;
pc=100*pt[1:8,]; print pc;
```

EXAMPLE 5.4: (ML estimation with EM algorithm: Model SC PC)

```
proc iml;
reset nolog;
/***********************************/
/* design matrix: S P C SP SC PC SPC */
/***********************************/
X={1  1  1  1  1  1  1  1,
   1 -1  1  1 -1 -1  1 -1,
   1  1 -1  1 -1  1 -1 -1,
   1 -1 -1  1  1 -1 -1  1,
   1  1  1 -1  1 -1 -1 -1,
   1 -1  1 -1 -1  1 -1  1,
   1  1 -1 -1 -1 -1  1  1,
   1 -1 -1 -1  1  1  1 -1};

/** model:SP,SC,PC **/  ah=X[,8];
/** model:SP,SC    **/  ah=X[,7:8];
/** model:SC,PC    **/  ah=X[,5]||X[,8];
y={3,176,4,293,17,197,2,23,10,150,5,90};

ya=y[1:8,];  na=ya[+];  pa=ya/na;     yabeg=ya; ya1=ya;

diff2=1;  r=0;
do while (diff2>1e-10);
diff1=1;

/*******************************************************/
/* First iteration:   Starting values of EM algorithm */
/* Higher iterations: M-Step of EM algorithm          */
/*******************************************************/
do while (diff1>1e-20);
  yt=ya-ah*ginv(ah`*diag(1/ya)*ah)*ah`*log(ya);
  diff1=(yt-ya)`*(yt-ya);
  ya=yt;
end;

/***********************************/
/* E-Step of EM algorithm          */
/***********************************/
r=r+1;
pa=ya/ya[+];
pfill=j(2,2,1)@i(4)*pa;
yb=j(2,1,1)@y[9:12,];
ya=yabeg+yb#pa/pfill;
ya2=ya;

diff2=(ya2-ya1)`*(ya2-ya1);
ya1=ya2;
end;
print r pa;

sig=diag(ya)-ah*ginv(ah`*diag(1/ya)*ah)*ah`;
cov=ginv(X`*X)*X`*(diag(1/ya)*sig*diag(1/ya))*X*ginv(X`*X);
var=diag(cov);

pa=100*pa;
/*print pa var;*/
```

EXAMPLE 5.4: (ML estimation under constraints: Model CS CP)

```
proc iml; reset nolog;
y={3,176,4,293,17,197,2,23,10,150,5,90}; ybegin=y; y0=y; n=y[+]; mu0=y;
ya=y[1:8,];  na=ya[+];
yb=y[9:12,]; nb=yb[+];
i=2; j=2; k=2; jk=j*k; ijk=i*j*k;
x={1  1  1  1  1  1  1  1,
   1 -1  1  1 -1 -1  1 -1,
   1  1 -1  1 -1  1 -1 -1,
   1 -1 -1  1  1 -1 -1  1,
   1  1  1 -1  1 -1 -1 -1,
   1 -1  1 -1 -1  1 -1  1,
   1  1 -1 -1 -1 -1  1  1,
   1 -1 -1 -1  1  1  1 -1};
xu=x[,1:4]||x[,6:7];
p1=i(8)-xu*ginv(xu`*xu)*xu`;
cr=(1/na)#j(1,i,1)@i(jk);

diff1=1; j1=0;
diff2=1; j2=0;

do while (diff1>1e-10); j1=j1+1; j2=0; diff2=1;
 ya=y[1:8,]; yb=y[9:12,];
 cov=diag(y)-1/n#y*y`;
 gmu1=(p1*diag(1/ya))||j(8,4,0);
 gmu2=cr||((-1/nb)#i(jk));
 gmu=gmu1//gmu2;
 y=ybegin;

do while (diff2>1e-10); j2=j2+1;
 ya=y[1:8,]; yb=y[9:12,];
 g1=p1*log(ya);
 g2=(cr||((-1/nb)#i(jk)))*y;
 g=g1//g2;
 gy1=(p1*diag(1/ya))||j(ijk,jk,0);
 gy2=cr||((-1/nb)#i(jk));
 gy=gy1//gy2;

 yt=y-(gmu*cov)`*ginv(gy*cov*gmu`)*g;
 diff2=(yt-y0)`*(yt-y0);
 y0=yt;
 y=yt;
end;

mut=yt;
diff1=(mut-mu0)`*(mut-mu0);
mu0=mut;
end;

ya=y[1:8,];
pa=ya/na; pb=yb/nb;
print i j pa pb;
```

```
cov=diag(y)-1/n#y*y`;
gmu1=(p1*diag(1/ya))||j(ijk,jk,0);
gmu2=cr||((-1/nb)#i(jk));
gmu=gmu1//gmu2;

sig=sqrt(1/na#1/na#vecdiag(cov-(gmu*cov)`*ginv(gmu*cov*gmu`)*gmu*cov));
sig=sig[1:8,];
p=p[1:8,];
print ybegin yt pa sig;
```

I	J	PA	PB
2	2	0.0049631	0.0317501
		0.254203	0.538353
		0.0075794	0.010518
		0.3882079	0.4193789
		0.026787	
		0.28415	
		0.0029385	
		0.0311711	

YBEGIN	YT	PA	SIG
3	3.5486225	0.0049631	0.0015509
176	181.75517	0.254203	0.0155327
4	5.4193025	0.0075794	0.002343
293	277.56862	0.3882079	0.0158979
17	19.152686	0.026787	0.0051063
197	203.16723	0.28415	0.0160086
2	2.1010385	0.0029385	0.0007932
23	22.287327	0.0311711	0.0061783
10	8.0962709		
150	137.28002		
5	2.6820797		
90	106.94163		