Terms and Conditions of Use of Digitised Theses from Trinity College Library Dublin

Copyright statement

All material supplied by Trinity College Library is protected by copyright (under the Copyright and Related Rights Act, 2000 as amended) and other relevant Intellectual Property Rights. By accessing and using a Digitised Thesis from Trinity College Library you acknowledge that all Intellectual Property Rights in any Works supplied are the sole and exclusive property of the copyright and/or other IPR holder. Specific copyright holders may not be explicitly identified. Use of materials from other sources within a thesis should not be construed as a claim over them.

A non-exclusive, non-transferable licence is hereby granted to those using or reproducing, in whole or in part, the material for valid purposes, providing the copyright owners are acknowledged using the normal conventions. Where specific permission to use material is required, this is identified and such permission must be sought from the copyright holder or agency cited.

Liability statement

By using a Digitised Thesis, I accept that Trinity College Dublin bears no legal responsibility for the accuracy, legality or comprehensiveness of materials contained within the thesis, and that Trinity College Dublin accepts no liability for indirect, consequential, or incidental, damages or losses arising from use of the thesis for whatever reason. Information located in a thesis may be subject to specific use constraints, details of which may not be explicitly described. It is the responsibility of potential and actual users to be aware of such constraints and to abide by them. By making use of material from a digitised thesis, you accept these copyright and disclaimer provisions. Where it is brought to the attention of Trinity College Library that there may be a breach of copyright or other restraint, it is the policy to withdraw or take down access to a thesis while the issue is being resolved.

Access Agreement

By using a Digitised Thesis from Trinity College Library you are bound by the following Terms & Conditions. Please read them carefully.

I have read and I understand the following statement: All material supplied via a Digitised Thesis from Trinity College Library is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of a thesis is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form providing the copyright owners are acknowledged using the normal conventions. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Topics in Unsupervised Learning

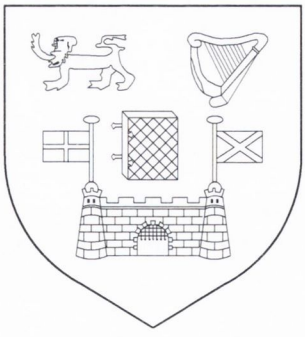
by

Paul David McNicholas

A thesis submitted to the University of Dublin in satisfaction of the requirement for the degree of

Doctor of Philosophy

Department of Statistics,
University of Dublin, Trinity College.



June 2007

Declaration

This thesis has not been submitted as an exercise for a degree at this or any other University and it is entirely my own work. I agree that the Library may lend or copy this thesis upon request; this permission covers only single copies made for study purposes, subject to normal conditions of acknowledgement. The copyright belongs jointly to the University of Dublin and Paul David McNicholas.

Paul David McNicholas

Summary

Two topics in unsupervised learning are reviewed and developed; namely, model-based clustering and association rule mining.

A new family of Gaussian mixture models, with a parsimonious covariance structure, is introduced. The mixtures of factor analysers and mixtures of principal component analysers models are special cases of this new family of models. This family exhibit the feature that their number of covariance parameters grows linearly with the dimensionality of the data, which leads to relatively fast computation time. These models perform excellently, compared to popular model-based clustering techniques, when applied to real data.

A new family of Gaussian mixture models with a Cholesky-decomposed covariance structure is also introduced. Four members of this family are developed and applied to real data. This family of models has great potential for further development in future work.

A novel approach, *via* association rules, is taken to the analysis of college applications data. This analysis contributes to the discussion about the existence of a 'points race'. A new method of quantifying and visualising the interestingness of an association rule is also introduced and an argument for the inclusion of negations in the association rule mining process is given.

Acknowledgements

I gratefully acknowledge the guidance, support and patience of my supervisors, Dr. Brendan Murphy and Dr. Myra O'Regan. Specifically, to Dr. Murphy for demonstrating and explaining to me many invaluable research techniques, encouraging me to apply mathematical reasoning to statistical problems and for giving me tremendous support throughout my time as a postgraduate student. Specifically, to Dr. O'Regan for frequently reminding me to refer both to first principles and the practical applications of my work.

In addition, I would like to thank Prof. John Haslett, Mr. Eamonn Mullins and Ms. Deirdre Toher for their advice on particular aspects of my work. I would like to add a special note of thanks to Mr. Aaron McDaid, Mr. Dermot Frost and Mr. Michael Salter–Townshend for their generous advice on programming in C.

Thanks to Prof. Patrick Prendergast, Dean of Graduate Studies, and Dr. Mike Peardon, Course Director for the MSc in High Performance Computing, for allowing me to take an MSc in High Performance Computing while completing this work. The knowledge gained from this course has been valuable while completing some of the research outlined herein, and will be of great benefit in future research.

I would like to extend a very special note of personal gratitude to Prof. Petros Florides, Mr. Eamonn Mullins, Dr. Donal O'Donovan and Dr. Myra O'Regan who have each given me fantastic support and encouragement throughout my years at university. Furthermore, Ms. Sharon King, Mr. John McCarthy, Mr. Aaron McDaid, Ms. Éimhín Ní Mhuircheartaigh, Mr. Stuart O'Neill and Mr. Michael Salter-Townshend provided me with great personal support throughout my time as a postgraduate student, for which I am very grateful. I am also grateful to my family for their continued support.

I would like to gratefully acknowledge the contribution of my examiners, Prof. John Hinde and Dr. Krzysztof Mosurski, for their helpful comments and suggestions. Thanks are also due to Ms. Sharon King for translating the variable names from the coffee data from German into English.

This research was funded by a Science Foundation Ireland Basic Research Grant and LaTeX was used to produce this thesis.

Publications

The following articles, arising from this work, are to be published or are under review.

McNicholas, P. D., Murphy, T. B. and O'Regan, M., 'Standardising the Lift of an Association Rule', *Computational Statistics and Data Analysis* (under review).

McNicholas, P. D. and Murphy, T. B., 'Parsimonious Gaussian Mixture Models', *Computational Statistics* (conditionally accepted).

McNicholas, P. D. (2007), 'Association rule analysis of CAO data', *Journal of the Statistical and Social Inquiry Society of Ireland* **26**. Includes discussion and rejoinder.

To Sharon, without whom nothing would be truly worthwhile.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Unsupervised Learning

1.1.1 Definition

Hastie *et al.* (2001) give the analogy of "learning without a teacher" when introducing unsupervised learning. Formally, they consider a set of n realisations $\{x_1, x_2, \ldots, x_n\}$ of a random p-dimensional vector $\mathbf{x}$ with joint probability distribution $\mathbb{P}(\mathbf{x})$. The objective is then to deduce properties of $\mathbb{P}(\mathbf{x})$ without knowing the correct answers, that is "without the help of a supervisor".

This approach to statistical learning can also be distinguished by whether or not the outcome, or target, variable is present in the learning process. A model learns in a supervised fashion when the outcome variable is present during the learning process, while in the unsupervised learning context, the outcome variable is either missing, non-existent or there is no obvious outcome variable.

This work focuses on unsupervised learning for two types of data — interval data and binary data. The analysis of these different data types necessitates two very different approaches; model-based clustering techniques are applied to interval data and a data mining technique is used for the analysis of binary data.

1.1.2 Model-Based Clustering

Clustering based on mixture models has appeared in the literature with increased frequency in recent years. The approach followed herein involves a family of Gaussian mixture models with parsimonious covariance structure. The clustering of the data, according to this family of models, is given by the values assigned to the group membership labels through the learning process. Well-established mixture modelling techniques are used for motivation and the mathematical foundation of this family of models is established. These models are then applied to real data and perform favorably when compared to well-established techniques. A family of Gaussian mixture models that can be applied to longitudinal data is also introduced.

1.1.3 Association Rule Mining

Association rule mining is developed as an effective method of analysing binary data. A novel application of association rules to a 'clustering' problem is demonstrated and a method of standardising the lift of an association rule is introduced that leads to a new measure of interestingness. Additionally, the number of potentially interesting association rules that can be mined from a binary dataset is quantified and an argument is put forward for the inclusion of negations in association rule analysis.

1.2 Thesis Structure

1.2.1 Chapter 2

Ideas around model-based clustering and Gaussian mixture models are introduced and the popular literature is reviewed. The expectation-maximisation algorithm is discussed, as are three methods of model evaluation; the Bayesian information criterion, the Rand index and the adjusted Rand index.

1.2.2 Chapter 3

Parsimonious Gaussian mixture models are developed using a latent Gaussian model which is closely related to the factor analysis model. These models provide a unified modelling framework which includes the mixtures of factor of analysers and mixtures of probabilistic principal component analysers models as special cases. In particular, a class of eight parsimonious Gaussian mixture models, which are based on the mixtures of factor analysers model, are introduced.

A method for obtaining the maximum likelihood estimates for the parameters in these models, *via* an AECM algorithm, is demonstrated. This family of models includes five parsimonious models that have not previously been developed. All members of this family have the attractive property that their number of covariance parameters is linear in the dimensionality of the data. These models are applied to the analysis of chemical and physical properties of Italian wines and the chemical properties of coffee; the models are shown to give superior clustering results when compared to well-established techniques.

1.2.3 Chapter 4

Chapter 4 builds on the ideas in Chapter 3 with the covariance structure of the mixture models being further generalised. This results in four extra models. These models also have a number of covariance parameters that is linear in the dimensionality of the data. None of these four models have previously appeared in the literature and combined with the eight models from Chapter 3, they give a family of twelve parsimonious Gaussian mixture models. These models are applied to crabs data where they give excellent results when compared to well-established techniques.

1.2.4 Chapter 5

Chapter 5 presents a different slant on the ideas of Chapter 3 and Chapter 4 with the covariance structures of the models being constructed *via* a modified Cholesky decomposition. This leads to a family of four mixture models, two of which have not previously appeared in the literature, that can be used to cluster longitudinal data. This family of mixture

models is shown to perform well when applied to real data. Further developments of this covariance structure, that could potentially increase the number of models in this family, are also suggested.

1.2.5 Chapter 6

This chapter marks the change in emphasis from interval data to binary data and thus the switch from model-based clustering to association rule mining. In Chapter 6 association rules are introduced. The term 'association rule' is defined, along with related functions, and the popular literature is summarised. Various methods of generating, pruning and ranking rules are introduced and reviewed.

1.2.6 Chapter 7

Chapter 7 introduces the novel idea of using association rules to reveal grouping in data. Central Applications Office college application data is analysed using association rule mining to investigate relationships between course choices across applicants. The role of gender as a factor in course selection is examined as well as a larger question around the functionality of the application system — what attracts students to a course; is it a topic of interest or is it the perceived status of the course associated with high entry points?
The expected gender imbalances in areas like primary teaching and engineering appear, along with some others. Association rules generated suggest that students select courses based primarily on topic but sometimes with geographical location in mind. No evidence is found to suggest that students are selecting courses based on perceived points status.

1.2.7 Chapter 8

The range of values that the lift of an association rule may take is used to standardise lift so that it is more effective as a measure of interestingness. This standardisation is extended to account for minimum support and confidence thresholds. A method of visualising standardised lift, through the relationship between lift and its upper and lower bounds, is proposed. The application of standardised lift as a measure of interestingness is demonstrated on the

college application data that was analysed in Chapter 7 and on German social data. In the latter case, negations are introduced into the mining paradigm and an argument for their inclusion is put forward. This argument includes a quantification of the number of extra rules that arise when negations are considered.

1.2.8 Chapter 9

The ideas and methods demonstrated in this work are summarised in Chapter 9. Suggestions for further work are given, both based upon and arising from this work.

1.3 The Impact of this Work

The impact of this work on the body of literature is summarised here based on the most significant original ideas contained herein. The principal novel features of this work are:

- A new family of Gaussian mixture models, with a parsimonious covariance structure, is introduced. The covariance structure is similar to that of the mixtures of factor analysers model, which is in fact a member of this family. This new family of models exhibit the feature that their number of covariance parameters grows linearly with the dimensionality of the data, which leads to relatively fast computation time. These models perform excellently, compared to popular model-based clustering techniques, when applied to real data.

- A new family of Gaussian mixture models with a Cholesky-decomposed covariance structure is introduced. Four members of this family are developed and applied to real data. This family of models has great potential for further development in future work.

- A new approach is taken to the analysis of college applications data that contributes to the discussion about the existence of a 'points race'.

- A new method of quantifying and visualising the interestingness of an association rule is introduced and an argument for the inclusion of negations in the association rule paradigm is put forward.

Chapter 2

Model-Based Clustering

2.1 Background

Chapters 3, 4 and 5 of this work focus on families of mixture models. Each of these families is applied to interval data as a model-based clustering technique.

2.2 Mixture Models

2.2.1 Finite Mixture Models

Model-based clustering techniques can be traced at least as far back as Wolfe (1963). In more recent years model-based clustering has appeared in the statistics literature with increased frequency (Fraley & Raftery, 1998; McLachlan *et al.*, 2003; Raftery & Dean, 2006). Typically the data are clustered using some assumed mixture modelling structure and the parameters associated with these models are usually estimated using an expectation-maximisation (EM) algorithm.

Finite mixture models assume that data are collected from a finite collection of sub-populations and that the data within each sub-population can be modelled using some standard statistical model. Therefore, such mixture models can be especially useful when modelling data that may have been collected from multiple sources. Mixture models are the basis of most modern model-based clustering algorithms; partly because methods based on

mixtures offer the advantage of allowing for uncertainties to be quantified using probabilities. Extensive reviews of finite mixture models are contained in Titterington *et al.* (1985), McLachlan & Basford (1988) and McLachlan & Peel (2000*a*).

2.2.2 Gaussian Mixture Models

The Gaussian mixture model has received particular attention, both in the three aforementioned texts and in the wider body of statistical literature. A Gaussian model is assumed for each sub-population, or group. The model density is of the form

$$f(\mathbf{x}) = \sum_{g=1}^{G} \pi_g \phi(\mathbf{x}|\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g), \tag{2.1}$$

where

$$\phi(\mathbf{x} \mid \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) = \frac{1}{\sqrt{(2\pi)^p \mid \boldsymbol{\Sigma}_g \mid}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_g)'\boldsymbol{\Sigma}_g^{-1}(\mathbf{x} - \boldsymbol{\mu}_g)\right\}$$

is the density of a multivariate Gaussian with mean $\boldsymbol{\mu}_g$ and covariance $\boldsymbol{\Sigma}_g$, and π_g is the probability of membership of group g. A review of Gaussian mixture models with particular emphasis on applications to cluster analysis, discriminant analysis and density estimation is given by Fraley & Raftery (2002*b*).

2.3 The Expectation-Maximisation Algorithm

The EM algorithm is an iterative method for finding the maximum of the expected value of the complete-data log-likelihood. Before introducing the EM algorithm, a larger class of algorithms to which the EM algorithm belongs is introduced.

2.3.1 MM Algorithms

MM algorithms are a blueprint for a broad, ever-expanding, class of algorithms. A review of MM algorithms is given by Hunter & Lange (2004) and the remainder of this section essentially presents a summary of their work.

The 'MM' stands for 'minorise-maximise' or 'majorise-minimise', depending on whether the purpose of the algorithm is to minimise or maximise the function of interest. Application of

the MM principle can be seen in the literature at least as far back as Ortega & Rheinboldt (1970) but the first use of the term MM came in Hunter & Lange (2000).

Let $\varphi^{(m)}$ be a fixed value of a parameter φ and let $g(\cdot)$ and $f(\cdot)$ be real-valued functions. Then $g\left(\varphi \mid \varphi^{(m)}\right)$ is said to majorize $f(\varphi)$ at the point $\varphi^{(m)}$ if

$$g\left(\varphi \mid \varphi^{(m)}\right) \geq f(\varphi) \quad \text{for all } \varphi, \text{ and}$$
$$g\left(\varphi^{(m)} \mid \varphi^{(m)}\right) = f\left(\varphi^{(m)}\right).$$

Furthermore, $g\left(\varphi \mid \varphi^{(m)}\right)$ is said to minorize $f(\varphi)$ at $\varphi^{(m)}$ if $-g\left(\varphi \mid \varphi^{(m)}\right)$ majorizes $-f(\varphi)$ at $\varphi^{(m)}$.

Further details, including the casting of the EM algorithm as a special case, are given by Hunter & Lange (2004). In the case of the EM algorithm, the minorising function is the expected value of the complete-data log-likelihood.

2.3.2 The EM Algorithm

The EM algorithm (Dempster *et al.*, 1977) provides an iterative method of finding maximum likelihood estimates (MLEs) where the data is incomplete or some of the data is missing. Importantly, the data does not actually need to be incomplete but framing problems as incomplete data problems often leads to efficient solutions *via* the EM algorithm.

In the E-step, the expected value of the log-likelihood is computed based on the current estimates of the model parameters and the 'complete-data' vector — that is, the vector of observed data plus missing data. This function, the expected value of the complete-data log-likelihood, is a minorising function — a fact which follows from Jensen's inequality (Jensen, 1906). In the M-step, this expected value is maximised with respect to the model parameters.

These two steps are repeated iteratively until convergence is reached. There are a variety of ways to measure convergence; one common approach is to take convergence as the point at which the difference in successive estimates of the log-likelihood is sufficiently small. A comprehensive overview of EM algorithms is given by McLachlan & Krishnan (1997) and alternative convergence criteria are given by Lindsay (1995) and McLachlan & Krishnan (1997).

2.4 MCLUST: Software for Model-Based Cluster Analysis

The general Gaussian mixture model, given in Equation 2.1, has a total of

$$(G-1)+Gp+Gp(p+1)/2$$

parameters, of which $Gp(p+1)/2$ are from the group covariance matrices $\mathbf{\Sigma}_g$. A simpler form of the mixture assumes that the covariances are constrained to be equal across groups, which reduces to a total of $(G-1)+Gp+p(p+1)/2$ parameters, of which $p(p+1)/2$ are from the common group covariance matrix $\mathbf{\Sigma}_g = \mathbf{\Sigma}$.

Banfield & Raftery (1993), Celeux & Govaert (1995) and Fraley & Raftery (1998, 2002*b*) exploit an eigenvalue decomposition of the group covariance matrices to give a wide range of covariance structures that use between one and $Gp(p+1)/2$ parameters. The eigenvalue decomposition of the covariance matrix is of the form

$$\mathbf{\Sigma}_g = \lambda_g \mathbf{D}_g \mathbf{A}_g \mathbf{D}_g', \tag{2.2}$$

where λ_g is a constant, $\mathbf{D}_g$ is a matrix consisting of the eigenvectors of $\mathbf{\Sigma}_g$ and $\mathbf{A}_g$ is a diagonal matrix with entries proportional to the eigenvalues of $\mathbf{\Sigma}_g$.

The model-based clustering techniques that have been developed using the covariance structure given in Equation 2.2 allow for a variety of constraints. There is the option to constrain the components of the eigenvalue decomposition of $\mathbf{\Sigma}_g$ across groups of the mixture model. Furthermore, the matrices $\mathbf{D}_g$ or $\mathbf{D}_g$ and $\mathbf{A}_g$ may be set equal to $\mathbf{I}$, or not.

Fraley & Raftery (2002*a*, 2003) describe the `mclust` software, which is available as a library in the software package R (R Development Core Team, 2006). The `mclust` software allows efficient model-based clustering and incorporates the work of Fraley & Raftery (1998, 1999, 2002*b*). Table 2.1, which is taken from Dean *et al.* (2006), gives the covariance decompositions that are available using the `mclust` software. Figure 2.1, which is similar to a figure in Dean *et al.* (2006), provides an illustration of these covariance decompositions.

Fraley & Raftery (2002*b*) demonstrate that the parsimonious mixture models derived in this model-based clustering framework give excellent results in a variety of applications. Dean *et al.* (2006) illustrate the application of these models to give classification rules in food authenticity problems.

Table 2.1: A variety of covariance structures, $\boldsymbol{\Sigma}_g$, available using `mclust`, along with the number of covariance parameters in each case.

ID	Volume	Shape	Orientation	Covariance Decomp.	Number of Covariance Parameters
EII	Equal	Spherical	—	$\lambda\mathbf{I}$	1
VII	Variable	Spherical	—	$\lambda_k\mathbf{I}$	G
EEI	Equal	Equal	Axis-aligned	$\lambda\mathbf{A}$	p
VEI	Variable	Equal	Axis-aligned	$\lambda_g\mathbf{A}$	$p+G-1$
EVI	Equal	Variable	Axis-aligned	$\lambda\mathbf{A}_g$	$pG-G+1$
VVI	Variable	Variable	Axis-aligned	$\lambda_g\mathbf{A}_g$	pG
EEE	Equal	Equal	Equal	$\lambda\mathbf{DAD}'$	$p(p+1)/2$
EEV	Equal	Equal	Variable	$\lambda\mathbf{D}_k\mathbf{AD}_k'$	$Gp(p+1)/2-(G-1)p$
VEV	Variable	Equal	Variable	$\lambda_k\mathbf{D}_k\mathbf{AD}_k'$	$Gp(p+1)/2-(G-1)(p-1)$
VVV	Variable	Variable	Variable	$\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k'$	$Gp(p+1)/2$

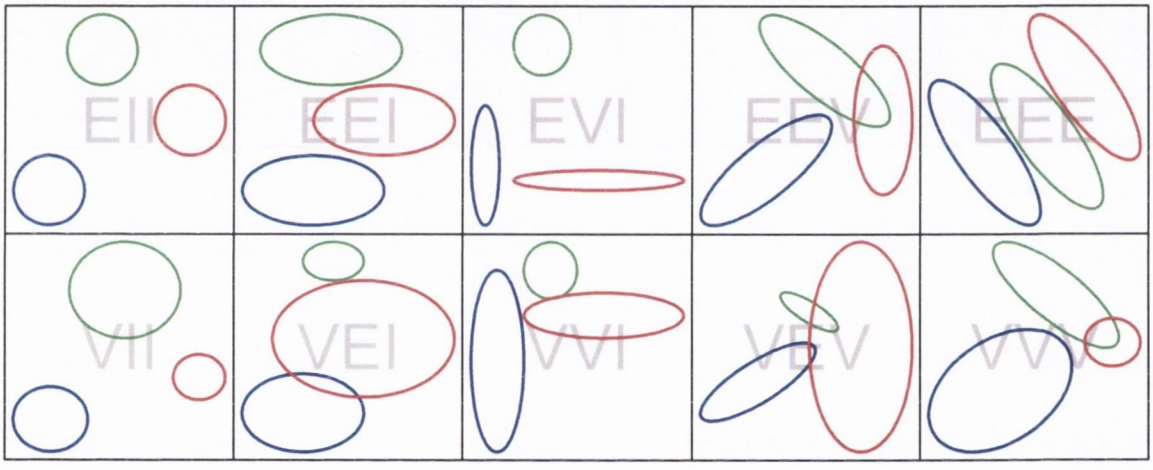


Figure 2.1: Cluster shapes that correspond to the covariance structures given in Table 2.1.

Notably, the number of covariance parameters given in Table 2.1 for the `mclust` models are either constant, linear or quadratic in p. Therefore, not all of these models are well suited to modelling high dimensional data. In particular, fitting a mixture model with any of the more general covariance structures available in `mclust`, that is the last four covariance structures given in Table 2.1, to high-dimensional data will be extremely computationally intensive.

2.5 Variable Selection

Raftery & Dean (2006) propose a variable selection method based on the use of Bayes factors. Their approach is that which would be taken to a model selection problem. Two models, M_1 and M_2 say, for data X are compared using the using Bayes factors. The Bayes factor, B_{12}, for model M_1 versus model M_2, is defined as

$$B_{12} = \frac{p(X \mid M_1)}{p(X \mid M_2)},$$

where

$$p(X \mid M_k) = \int p(X \mid \boldsymbol{\theta}_k, M_k) p(\boldsymbol{\theta}_k \mid M_k) d\boldsymbol{\theta}_k,$$

$\boldsymbol{\theta}_k$ is the vector of parameters for model M_k and $p(\boldsymbol{\theta}_k \mid M_k)$ is the prior distribution of M_k (Kass & Raftery, 1995). Variables are then selected based on which model is the 'best'. Variable selection can be carried out using the `clustvarsel` package (Dean & Raftery, 2006) in R. The user only needs to preset the maximum number of groups and `clustvarsel` automatically selects the variables. However, due to its nature, variable selection involves many runs of `mclust` and once the variables are selected the user needs to run `mclust` on the chosen variables to get the classifications. Therefore, this method has even greater limitations, in terms of computation time, than `mclust`, and thus will be particularly apparent in applications to high-dimensional data.

2.6 Mixture Model Selection & Performance

2.6.1 The Bayesian Information Criterion

The Bayesian information criterion (BIC) (Schwartz, 1978) is often used to select an appropriate mixture model; in the case of `mclust`, for example. For a model with parameters Φ, the BIC is given by

$$\text{BIC} = 2l(x, \hat{\Phi}) - m \log n,$$

where $l(x, \hat{\Phi})$ is the maximised log-likelihood, $\hat{\Phi}$ is the MLE of Φ, m is the number of free parameters in the model and n is the number of observations. The use of the BIC can

be motivated through an asymptotic approximation of the log posterior probability of the models (Kass & Raftery, 1995).

The usual regularity conditions for the asymptotic approximation used in the development of the BIC are not generally satisfied by mixture models. However, Leroux (1992) showed that the BIC, asymptotically, does not underestimate the true number of mixture components and Keribin (1998, 2000) showed that the BIC gives consistent estimates of the number of components in a mixture model. Furthermore, Fraley & Raftery (1998, 2002*b*) provide practical evidence that BIC performs well as a model selection criterion for mixture models. Note that there are alternatives to the BIC for mixture model selection; for example, the integrated completed likelihood (Biernacki *et al.*, 2000) penalises the BIC by subtracting the estimated mean entropy. However, the viability of such alternatives is not discussed herein and the BIC alone is used for model selection.

2.6.2 The Rand & Adjusted Rand Indices

The performance of mixture models in revealing group structure in data can measured using the Rand index (Rand, 1971) and the adjusted Rand index (Hubert & Arabie, 1985). These indices are computed on a cross-tabulation of the maximum *a posteriori* (MAP) classification of the observations with the true group membership. The rand index can be expressed as

$$\frac{\text{number of agreements}}{\text{number of agreements} + \text{number of disagreements}}, \tag{2.3}$$

where the 'number of agreements' is computed based on pair agreement and the 'number of disagreements' is based on pair disagreement. Consider an example similar to that given by Rand (1971), where a variable with two groups $y = \{(a, b), (c, d, e)\}$ is classified by $\hat{y} = \{(a, b, c), (d, e)\}$. Now, the number of agreements is the number of pairs that are together in both y and $\hat{y}$ plus the number of pairs that are separate in both y and $\hat{y}$. Therefore, there are six agreements; *ab*, *de*, *ad*, *ae*, *bd* and *be*. The denominator of Equation 2.3 can be easily computed as

$${}^5C_2 = \frac{5!}{2!3!} = 10,$$

and so the Rand index for this example is 0.6.

The adjusted Rand index corrects the Rand index for chance by accounting for the fact that if classification is performed randomly some cases will be correctly classified by chance. The indices each take values in $[0, 1]$ with '0' indicating that the MAP classification and true groups never agree and '1' indicating that the they are exactly the same. Large values of these indices indicate strong agreement between the true groupings and the classifications proposed by the mixture model.

Note that the possibility of using alternative or additional measures of class agreement, such as Cohen's kappa (Cohen, 1960), was considered but the Rand and adjusted Rand indices were thought to be sufficient for use in this work.

2.6.3 Note on Performance Assessment for Model-Based Clustering

Although the model-based clustering techniques described herein are unsupervised, their performance is assessed *via* application to real data and subsequent comparison of the resulting classifications to the true groups. However, in many applications the true group structure would not be known, which is reflected in the use of the BIC for model selection in the analyses that are carried out in this work.

Chapter 3

Parsimonious Gaussian Mixture Models

3.1 Introduction

A new family of finite mixture models is introduced, with a Gaussian model used to model each mixture component. These mixture components have a parsimonious factor analysis-like covariance structure. The factor analysis and probabilistic principal component analysis models are used for motivation. Crucially, the members of this new family of mixture models each have a number of covariance parameters that grows linearly in the dimensionality of the data.

3.2 Factor Analysis

3.2.1 The Model

Factor analysis is a data reduction technique that replaces the observed variables with unobservable factors that explain a satisfactory amount of the variability in the data. This technique is only useful if it returns 'many' less factors than there are variables. The model, which originated in the psychology literature (Spearman, 1904), assumes that a p-dimensional real-valued data vector $\mathbf{x}$ is modelled using a q-dimensional vector of real-valued

factors $\mathbf{u}$, where $q \ll p$. The model may be expressed in the form

$$\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \\ \vdots \\ \boldsymbol{\mu}_p \end{pmatrix} + \begin{pmatrix} \lambda_{11} & \lambda_{12} & \dots & \lambda_{1q} \\ \lambda_{21} & \lambda_{22} & \dots & \lambda_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{p1} & \lambda_{p2} & \dots & \lambda_{pq} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_q \end{pmatrix} + \begin{pmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \\ \vdots \\ \boldsymbol{\epsilon}_p \end{pmatrix},$$

or

$$\mathbf{x} = \boldsymbol{\mu} + \boldsymbol{\Lambda}\mathbf{u} + \boldsymbol{\epsilon}, \tag{3.1}$$

where $\boldsymbol{\Lambda}$ is a $p \times q$ matrix of factor loadings, the factors $\mathbf{u} \sim N(0, \mathbf{I}_q)$ and $\boldsymbol{\epsilon} \sim N(0, \boldsymbol{\Psi})$, where $\boldsymbol{\Psi} = \text{diag}(\psi_1, \psi_2, \dots, \psi_p)$. It is a feature of the factor analysis model that $\boldsymbol{\Lambda}$ is not uniquely defined; if $\boldsymbol{\Lambda}$ is replaced by $\boldsymbol{\Lambda}^* = \boldsymbol{\Lambda}\mathbf{D}$, where $\mathbf{D}$ is orthonormal, then

$$\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi} = (\boldsymbol{\Lambda}^*)(\boldsymbol{\Lambda}^*)' + \boldsymbol{\Psi},$$

which allows the results of factor analyses to be rotated, altering their interpretation. This is a large part of the reason why factor analysis has spent much time as "the black sheep of statistical theory" (Lawley & Maxwell, 1962).

3.2.2 The Likelihood Function

From Equation 3.1, the the density of an $\mathbf{x}_i$ is

$$f(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}) = \frac{1}{\sqrt{(2\pi)^p \mid \boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi} \mid}} \exp\left\{ -\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu})'(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) \right\}, \tag{3.2}$$

and it follows that the log-likelihood of $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)'$ is given by

$$\begin{aligned} l(\boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}) &= \sum_{i=1}^{n} \log f(\mathbf{x}_i \mid \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}) \\ &= -\frac{np}{2}\log 2\pi - \frac{n}{2}\log|\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi}| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})'(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) \\ &= -\frac{np}{2}\log 2\pi - \frac{n}{2}\log|\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi}| - \frac{n}{2}\text{tr}\left\{\mathbf{S}(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1}\right\}, \end{aligned} \tag{3.3}$$

where $\mathbf{S} = (1/n)\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})'$. Note that the data only appears in the model through $\mathbf{S}$ and that since the matrix $\boldsymbol{\Psi}$ is diagonal, the $p \times p$ matrix $(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})$ can be inverted using the formula

$$(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1} = \boldsymbol{\Psi}^{-1} - \boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}(\mathbf{I}_q + \boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda})^{-1}\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1} \tag{3.4}$$

(McLachlan & Peel, 2000*a*), which leaves only $q \times q$ matrices to be inverted. McLachlan & Peel (2000*a*) also give a convenient formula for finding the determinant of this matrix;

$$| \mathbf{\Lambda\Lambda}' + \mathbf{\Psi} | = | \mathbf{\Psi} | / | \mathbf{I}_p - \mathbf{\Lambda}' \left(\mathbf{\Lambda\Lambda}' + \mathbf{\Psi} \right)^{-1} \mathbf{\Lambda} | . \tag{3.5}$$

The formulae given in equations 3.4 and 3.5 have the potential to greatly reduce computation time.

Now, the MLE of $\boldsymbol{\mu}$ is easily obtained by differentiating Equation 3.3 with respect to $\boldsymbol{\mu}$ and setting the result equal to zero to get $\hat{\boldsymbol{\mu}} = \overline{\mathbf{x}}$. In order to obtain the MLEs of $\mathbf{\Lambda}$ and $\mathbf{\Psi}$, an EM algorithm is used.

3.2.3 The EM Algorithm for the Factor Analysis Model

E-step

The vector $\mathbf{x}$ is taken as the observed data and $\mathbf{u}$ as the missing data. The complete-data consist of $(\mathbf{x}, \mathbf{u})$; the observed data and the missing data. Before computing the complete-data log-likelihood $l_c(\boldsymbol{\mu}, \mathbf{\Lambda}, \mathbf{\Psi}) = \log f(\mathbf{x}, \mathbf{u})$, it is necessary to compute $\log f(\mathbf{x}_i \mid \mathbf{u}_i)$;

$$\begin{aligned}
\log f(\mathbf{x}_i \mid \mathbf{u}_i) &= -\frac{p}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{\Psi}| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu} - \mathbf{\Lambda}\mathbf{u}_i)' \mathbf{\Psi}^{-1} (\mathbf{x}_i - \boldsymbol{\mu} - \mathbf{\Lambda}\mathbf{u}_i) \\
&= -\frac{p}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{\Psi}| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) + (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i - \frac{1}{2} \mathbf{u}_i' \mathbf{\Lambda}' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i \\
&= -\frac{p}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{\Psi}| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}) + (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i - \frac{1}{2} \operatorname{tr} \left\{ \mathbf{\Lambda}' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i \mathbf{u}_i' \right\} \\
&= -\frac{p}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{\Psi}| - \frac{1}{2} \operatorname{tr} \left\{ \mathbf{\Psi}^{-1} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})' \right\} + (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i \\
&\quad - \frac{1}{2} \operatorname{tr} \left\{ \mathbf{\Lambda}' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i \mathbf{u}_i' \right\} .
\end{aligned}$$

Now,

$$\begin{aligned}
l_c(\boldsymbol{\mu}, \mathbf{\Lambda}, \mathbf{\Psi}) = \log f(\mathbf{x}, \mathbf{u}) &= \sum_{i=1}^{n} \log f(\mathbf{x}_i \mid \mathbf{u}_i) f(\mathbf{u}_i) \\
&= C - \frac{n}{2} \log |\mathbf{\Psi}| - \frac{1}{2} \operatorname{tr} \left\{ \mathbf{\Psi}^{-1} \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})' \right\} + \sum_{i=1}^{n} (\mathbf{x}_i - \boldsymbol{\mu})' \mathbf{\Psi}^{-1} \mathbf{\Lambda}\mathbf{u}_i \\
&\quad - \frac{1}{2} \operatorname{tr} \left\{ \mathbf{\Lambda}' \mathbf{\Psi}^{-1} \mathbf{\Lambda} \sum_{i=1}^{n} \mathbf{u}_i \mathbf{u}_i' \right\} ,
\end{aligned}$$

where C is constant with respect to $\boldsymbol{\mu}$, $\boldsymbol{\Lambda}$, and $\boldsymbol{\Psi}$. Now, the expected value of $\mathbf{u}_i$, conditional on $\mathbf{x}_i$ and the current model parameters, is

$$\mathbb{E}[\mathbf{u}_i|\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}] = \boldsymbol{\Lambda}'(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1}(\mathbf{x}_i - \boldsymbol{\mu}) = \boldsymbol{\beta}(\mathbf{x}_i - \boldsymbol{\mu}),$$

where $\boldsymbol{\beta} = \boldsymbol{\Lambda}'(\boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi})^{-1} = \boldsymbol{\Lambda}'\boldsymbol{\Sigma}^{-1}$, and the expected value of $\mathbf{u}_i\mathbf{u}_i'$ conditional on $\mathbf{x}_i$ and the current model parameters is

$$\mathbb{E}[\mathbf{u}_i\mathbf{u}_i'|\mathbf{x}_i, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}] = \mathbf{I}_q - \boldsymbol{\beta}\boldsymbol{\Lambda} + \boldsymbol{\beta}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})'\boldsymbol{\beta}'.$$

To complete the E-step, the expected value of the complete-data log-likelihood, which shall be denoted $Q(\boldsymbol{\Lambda}, \boldsymbol{\Psi})$, is computed at $\hat{\boldsymbol{\mu}} = \overline{\mathbf{x}}$.

$$\begin{aligned}
Q(\boldsymbol{\Lambda}, \boldsymbol{\Psi}) &= C - \frac{n}{2}\log|\boldsymbol{\Psi}| - \frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Psi}^{-1}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})'\right\} \\
&\quad + \sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\mathbb{E}\big[\mathbf{u}_i|\mathbf{x}_i, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\Psi}}\big] - \frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\sum_{i=1}^{n}\mathbb{E}\big[\mathbf{u}_i\mathbf{u}_i'|\mathbf{x}_i, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\Psi}}\big]\right\} \\
&= C - \frac{n}{2}\log|\boldsymbol{\Psi}| - \frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Psi}^{-1}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})'\right\} \\
&\quad + \operatorname{tr}\left\{\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})'\right\} - \frac{n}{2}\operatorname{tr}\big\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}(\mathbf{I}_q - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\Lambda}})\big\} \\
&\quad - \frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\left[\hat{\boldsymbol{\beta}}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})'\hat{\boldsymbol{\beta}}'\right]\right\} \\
&= C + \frac{n}{2}\log|\boldsymbol{\Psi}^{-1}| - \frac{n}{2}\operatorname{tr}\big\{\boldsymbol{\Psi}^{-1}\mathbf{S}\big\} + n\operatorname{tr}\big\{\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\mathbf{S}\big\} - \frac{n}{2}\operatorname{tr}\big\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}\big\},
\end{aligned}$$

where $\boldsymbol{\Theta} = \mathbf{I}_q - \hat{\boldsymbol{\beta}}\hat{\boldsymbol{\Lambda}} + \hat{\boldsymbol{\beta}}\mathbf{S}\hat{\boldsymbol{\beta}}'$ is a symmetric $q \times q$ matrix. Although $\boldsymbol{\Theta}$ was introduced as a notational convenience it could be used to devise a model diagnostic, since if $\boldsymbol{\Sigma} = \mathbf{S}$ then $\boldsymbol{\Theta} = \mathbf{I}_q$.

M-step

It is necessary to maximise Q with respect to $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}$ in the M-step of the EM algorithm. Differentiating Q with respect to $\boldsymbol{\Lambda}$ and utilising results from Graybill (1983), Lütkepohl

(1996) and Magnus & Neudecker (1999) gives

$$\begin{aligned} S_1(\boldsymbol{\Lambda}, \boldsymbol{\Psi}) = \frac{\partial Q}{\partial \boldsymbol{\Lambda}} &= n\frac{\partial}{\partial \boldsymbol{\Lambda}} \operatorname{tr}\{\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\mathbf{S}\} - \frac{n}{2}\frac{\partial}{\partial \boldsymbol{\Lambda}} \operatorname{tr}\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}\} \\ &= n(\boldsymbol{\Psi}^{-1})'(\hat{\boldsymbol{\beta}}\mathbf{S})' - \frac{n}{2}\frac{\partial}{\partial \boldsymbol{\Lambda}} \operatorname{tr}\{\boldsymbol{\Lambda}\boldsymbol{\Theta}\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\} \\ &= n\boldsymbol{\Psi}^{-1}\mathbf{S}'\hat{\boldsymbol{\beta}}' - \frac{n}{2}[(\boldsymbol{\Psi}^{-1})'\boldsymbol{\Lambda}\boldsymbol{\Theta}' + \boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}] \\ &= n\boldsymbol{\Psi}^{-1}\mathbf{S}\hat{\boldsymbol{\beta}}' - n\,\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}. \end{aligned}$$

Now, solving the equation $S_1(\hat{\boldsymbol{\Lambda}}, \boldsymbol{\Psi}) = 0$ gives $\hat{\boldsymbol{\Lambda}} = \mathbf{S}\hat{\boldsymbol{\beta}}'\boldsymbol{\Theta}^{-1}$. Differentiating Q with respect to $\boldsymbol{\Psi}^{-1}$ gives,

$$\begin{aligned} S_2(\boldsymbol{\Lambda}, \boldsymbol{\Psi}) = \frac{\partial Q}{\partial \boldsymbol{\Psi}^{-1}} &= \frac{n}{2}\boldsymbol{\Psi} - \frac{n}{2}\mathbf{S}' + n(\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\mathbf{S})' - \frac{n}{2}(\boldsymbol{\Lambda}')'(\boldsymbol{\Lambda}\boldsymbol{\Theta})' \\ &= \frac{n}{2}\boldsymbol{\Psi} - \frac{n}{2}\mathbf{S}' + n\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\mathbf{S} - \frac{n}{2}\boldsymbol{\Lambda}\boldsymbol{\Theta}'\boldsymbol{\Lambda}'. \end{aligned}$$

Now, solving $\operatorname{diag}\{S_2(\hat{\boldsymbol{\Lambda}}, \hat{\boldsymbol{\Psi}})\} = 0$ gives

$$\begin{aligned} &\frac{n}{2}\hat{\boldsymbol{\Psi}} = \operatorname{diag}\left\{\frac{n}{2}\mathbf{S}' - n\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\beta}}\mathbf{S} + \frac{n}{2}\hat{\boldsymbol{\Lambda}}\boldsymbol{\Theta}'[\mathbf{S}\hat{\boldsymbol{\beta}}'\boldsymbol{\Theta}^{-1}]'\right\} \\ \Rightarrow \quad &\hat{\boldsymbol{\Psi}} = \operatorname{diag}\{\mathbf{S}' - 2\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\beta}}\mathbf{S} + \hat{\boldsymbol{\Lambda}}\boldsymbol{\Theta}'(\boldsymbol{\Theta}^{-1})'\hat{\boldsymbol{\beta}}\mathbf{S}\} = \operatorname{diag}\{\mathbf{S} - \hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\beta}}\mathbf{S}\}. \end{aligned}$$

The matrix results used in this chapter are listed in Appendix A.1. Note that all matrices and vectors herein are real, all objects that are differentiated are continuously differentiable and that all differentials are well defined.

3.2.4 The Probabilistic Principal Component Analysis Model

The probabilistic principal component analysis (PPCA) model (Tipping & Bishop, 1999*b*) is a special case of the factor analysis model with $\boldsymbol{\Psi} = \psi\mathbf{I}_p$.

3.2.5 Mixtures of Factor Analysers & Mixtures of Probabilistic Principal Component Analysers

Developing the factor analysis model, Ghahramani & Hinton (1997) develop a mixture of factor analysers model, which is further developed by McLachlan & Peel (2000*a*). The density of an observation in group g is as in Equation 3.2 with $\boldsymbol{\mu} = \boldsymbol{\mu}_g$, $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi} = \boldsymbol{\Psi}_g$. The probability of membership of group g is denoted by π_g. Mixtures of factor

analysers traditionally differ in whether the $\mathbf{\Psi}_g$ term is constrained to be equal in different groups or not. Tipping & Bishop (1999*a*) develop the PPCA model to get a mixture of probabilistic principal component analysers model, which is a special case of the mixtures of factor analysers model.

3.3 Parsimonious Gaussian Mixture Models

3.3.1 Covariance Structures

The models mentioned in Section 3.2.5 can be extended to a wider family of models by constraining, or not, $\mathbf{\Lambda}_g = \mathbf{\Lambda}$, $\mathbf{\Psi}_g = \mathbf{\Psi}$ and whether $\mathbf{\Psi}_g$ is isotropic. The matrix $\mathbf{\Psi}_g$ is said to be isotropic if $\mathbf{\Psi}_g = \psi_g \mathbf{I}_p$. The option to apply, or not, these constraints, leads to eight parsimonious Gaussian mixture models (PGMMs). These models, along with their respective covariance structures, are given in Table 3.1.

Table 3.1: The covariance structure and number of covariance parameters for each PGMM.

Model	Loading Matrix	Error Variance	Isotropic ($\mathbf{\Psi}_g = \psi_g \mathbf{I}_p$)	Number of Covariance Parameters
CCC	Constrained	Constrained	Constrained	$\{pq - q(q-1)/2\} + 1$
CCU	Constrained	Constrained	Unconstrained	$\{pq - q(q-1)/2\} + p$
CUC	Constrained	Unconstrained	Constrained	$\{pq - q(q-1)/2\} + G$
CUU	Constrained	Unconstrained	Unconstrained	$\{pq - q(q-1)/2\} + Gp$
UCC	Unconstrained	Constrained	Constrained	$G\{pq - q(q-1)/2\} + 1$
UCU	Unconstrained	Constrained	Unconstrained	$G\{pq - q(q-1)/2\} + p$
UUC	Unconstrained	Unconstrained	Constrained	$G\{pq - q(q-1)/2\} + G$
UUU	Unconstrained	Unconstrained	Unconstrained	$G\{pq - q(q-1)/2\} + Gp$

The last three models given in Table 3.1 have been developed previously. Ghahramani & Hinton (1997) assume the equal noise model (UCU) and in the context of the mixtures of PPCAs model, Tipping & Bishop (1999*a*) assume unequal, isotropic, noise (UUC). McLachlan & Peel (2000*a*) and McLachlan *et al.* (2003) assume unequal noise (UUU), however they comment that assuming equal noise (UCU) can give more stable results.

Notably, if q and G are fixed then the number of covariance parameters of each model grows linearly in p. That is, if the number of factors and the number of groups are fixed then the number of covariance parameters grows linearly in the number of variables. This is a very

important and convenient feature of each member of this family of models, since it leads to relatively fast computation time in cases with very many variables.

To illustrate how the numbers of covariance parameters given in Table 3.1 were arrived at, consider the CCU case. The group covariance stricture is of the form

$$\boldsymbol{\Sigma} = \boldsymbol{\Lambda}\boldsymbol{\Lambda}' + \boldsymbol{\Psi},$$

and $q(q-1)/2$ constraints are required so that $\boldsymbol{\Lambda}$ is uniquely defined (Lawley & Maxwell, 1971; McLachlan & Peel, 2000*a*). Therefore, recalling that $\boldsymbol{\Psi}$ is a $p \times p$ diagonal matrix, it follows that the CCU covariance structure has a total of

$$pq - q(q-1)/2 + p$$

free parameters. An analogous argument is used in the other seven cases.

Note that there is a relationship between the PGMM family of models and the MCLUST family of models. Setting $q = 0$ makes the PGMM covariance structure $\boldsymbol{\Sigma}_g = \boldsymbol{\Psi}_g$, which, depending on constraints, can also be $\boldsymbol{\Sigma}_g = \boldsymbol{\Psi}$, $\boldsymbol{\Sigma}_g = \psi_g \mathbf{I}_p$ or $\boldsymbol{\Sigma}_g = \psi \mathbf{I}_p$. Herein, we only consider the models in Table 3.1 for $q > 0$.

3.3.2 Model Fitting

The PGMMs are fitted using the alternating expectation-conditional maximisation (AECM) algorithm (Meng & van Dyk, 1997). The ECM algorithm (Meng & Rubin, 1993) replaces the M-step by a series of conditional maximization steps. The AECM algorithm allows a different specification of complete-data for each conditional maximization step. McLachlan & Krishnan (1997) give an extensive review of the EM algorithm and variants. McLachlan & Peel (2000*b*) give extensive details of fitting the AECM algorithm in the case where no constraints are imposed. An example of an AECM algorithm is given with the coding details in Appendix D.

3.3.3 Likelihoods

Denote by $\mathbf{z}$, the unobserved group labels where $z_{ig} = 1$ if observation i belongs to group g and $z_{ig} = 0$ otherwise. At the first stage of the AECM algorithm, when estimating π_g

and $\boldsymbol{\mu}_g$, the group membership labels $\mathbf{z}$ are taken as the missing data. The complete-data likelihood is given by

$$\mathcal{L}_1(\pi_g, \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = \prod_{i=1}^{n}\prod_{g=1}^{G}\left[\pi_g f(\mathbf{x}_i \mid \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)\right]^{z_{ig}},$$

and the expected value of the complete-data log-likelihood is

$$\begin{aligned} Q_1(\pi_g, \boldsymbol{\mu}) = &\sum_{g=1}^{G} n_g \log \pi_g - \frac{np}{2}\log 2\pi - \sum_{g=1}^{G}\frac{n_g}{2}\log|\boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}_g| \\ &- \sum_{g=1}^{G}\frac{n_g}{2}\operatorname{tr}\left\{\mathbf{S}_g(\boldsymbol{\Lambda}_g\boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}_g)^{-1}\right\}, \end{aligned}$$

where

$$\hat{z}_{ig} = \frac{\hat{\pi}_g f(\mathbf{x}_i \mid \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Lambda}}_g, \hat{\boldsymbol{\Psi}}_g)}{\sum_{g'=1}^{G}\hat{\pi}_{g'} f(\mathbf{x}_i \mid \hat{\boldsymbol{\mu}}_{g'}, \hat{\boldsymbol{\Lambda}}_{g'}, \hat{\boldsymbol{\Psi}}_{g'})}, \tag{3.6}$$

$n_g = \sum_{i=1}^{n}\hat{z}_{ig}$ and $\mathbf{S}_g = (1/n_g)\sum_{i=1}^{n}\hat{z}_{ig}(\mathbf{x}_i - \boldsymbol{\mu}_g)(\mathbf{x}_i - \boldsymbol{\mu}_g)'$. Now, maximising Q_1 with respect to $\boldsymbol{\mu}_g$ and π_g yields

$$\hat{\boldsymbol{\mu}}_g = \frac{\sum_{i=1}^{n}\hat{z}_{ig}\mathbf{x}_i}{\sum_{i=1}^{n}\hat{z}_{ig}} \quad \text{and} \quad \hat{\pi}_g = \frac{n_g}{n}.$$

At the second stage of the AECM algorithm, when estimating $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}_g$, the group membership labels $\mathbf{z}$ and the latent factors $\mathbf{u}$ are taken as the missing data. It follows that the complete-data log-likelihood is given by

$$\begin{aligned} l(\pi_g, \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) &= \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\left[\log \pi_g + \log f(\mathbf{x}_i|\mathbf{u}_i, \boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) + \log f(\mathbf{u}_i)\right] \\ &= C + \sum_{g=1}^{G}\left[-\frac{n_g}{2}\log|\boldsymbol{\Psi}_g| - \frac{n_g}{2}\operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1}\mathbf{S}_g\right\} + \sum_{i=1}^{n} z_{ig}(\mathbf{x}_i - \boldsymbol{\mu}_g)'\boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}_g\mathbf{u}_i\right. \\ &\quad \left. -\frac{1}{2}\operatorname{tr}\left\{\boldsymbol{\Lambda}_g'\boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}_g\sum_{i=1}^{n} z_{ig}\mathbf{u}_i\mathbf{u}_i'\right\}\right], \end{aligned}$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}_g$. Now, the expected value of the

complete-data log-likelihood, evaluated with $\boldsymbol{\mu}_g = \hat{\boldsymbol{\mu}}_g$ and $\pi_g = \hat{\pi}_g$, is of the form

$$
\begin{aligned}
Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = C + \sum_{g=1}^{G} \Bigg[& -\frac{n_g}{2} \log |\boldsymbol{\Psi}_g| - \frac{n_g}{2} \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \mathbf{S}_g\right\} \\
& + \sum_{i=1}^{n} \hat{z}_{ig} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_g)' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \mathbb{E}\left[\mathbf{u}_i | \mathbf{x}_i, \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Lambda}}_g, \hat{\boldsymbol{\Psi}}_g\right] \\
& - \frac{1}{2} \operatorname{tr}\left\{ \boldsymbol{\Lambda}_g' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \sum_{i=1}^{n} \hat{z}_{ig} \mathbb{E}\left[\mathbf{u}_i \mathbf{u}_i' | \mathbf{x}_i, \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Lambda}}_g, \hat{\boldsymbol{\Psi}}_g\right] \right\} \Bigg],
\end{aligned}
$$

which becomes;

$$
\begin{aligned}
Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = C + \sum_{g=1}^{G} \Bigg[& -\frac{n_g}{2} \log |\boldsymbol{\Psi}_g| - \frac{n_g}{2} \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \mathbf{S}_g\right\} + \sum_{i=1}^{n} \hat{z}_{ig} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_g)' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \hat{\boldsymbol{\beta}}_g (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_g) \\
& - \frac{1}{2} \operatorname{tr}\left\{ \boldsymbol{\Lambda}_g' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \sum_{i=1}^{n} \hat{z}_{ig} \mathbb{E}\left[\mathbf{u}_i \mathbf{u}_i' | \mathbf{x}_i, \hat{\boldsymbol{\mu}}_g, \hat{\boldsymbol{\Lambda}}_g, \hat{\boldsymbol{\Psi}}_g\right] \right\} \Bigg] \\
= C + \sum_{g=1}^{G} n_g & \left[\frac{1}{2} \log |\boldsymbol{\Psi}_g^{-1}| - \frac{1}{2} \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \mathbf{S}_g\right\} + \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\right\} - \frac{1}{2} \operatorname{tr}\left\{\boldsymbol{\Lambda}_g' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \boldsymbol{\Theta}_g\right\} \right] \\
= C + \frac{1}{2} \sum_{g=1}^{G} n_g & \left[\log |\boldsymbol{\Psi}_g^{-1}| - \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \mathbf{S}_g\right\} + 2 \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\right\} - \operatorname{tr}\left\{\boldsymbol{\Lambda}_g' \boldsymbol{\Psi}_g^{-1} \boldsymbol{\Lambda}_g \boldsymbol{\Theta}_g\right\} \right],
\end{aligned}
$$

where $\boldsymbol{\Theta}_g = \mathbf{I}_q - \hat{\boldsymbol{\beta}}_g \hat{\boldsymbol{\Lambda}}_g + \hat{\boldsymbol{\beta}}_g \mathbf{S}_g \hat{\boldsymbol{\beta}}_g'$ is a symmetric $q \times q$ matrix and the $\hat{z}_{ig}$ are computed as in Equation 3.6 using the estimates of $\hat{\boldsymbol{\mu}}_g$ and $\hat{\pi}_g$ from the first stage of the algorithm. Now, it only remains to maximise $Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)$ with respect to $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}_g$ respectively to get MLEs of these parameters.

3.3.4 The Models

Results of this maximisation follow for all eight PGMMs. Details are given in Appendix A. Let,

$$
\tilde{\mathbf{S}} = \sum_{g=1}^{G} \hat{\pi}_g \mathbf{S}_g \quad \text{and} \quad \tilde{\boldsymbol{\Theta}} = \mathbf{I}_q - \hat{\boldsymbol{\beta}} \hat{\boldsymbol{\Lambda}} + \hat{\boldsymbol{\beta}} \tilde{\mathbf{S}} \hat{\boldsymbol{\beta}}'.
$$

Model CCC

$$
\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\Lambda}}' (\hat{\boldsymbol{\Lambda}} \hat{\boldsymbol{\Lambda}}' + \hat{\psi} \mathbf{I}_p)^{-1}, \; \hat{\boldsymbol{\Lambda}}^{\text{new}} = \tilde{\mathbf{S}} \hat{\boldsymbol{\beta}}' \tilde{\boldsymbol{\Theta}}^{-1}, \; (\hat{\psi})^{\text{new}} = \frac{1}{p} \operatorname{tr}\left\{\tilde{\mathbf{S}}' - \hat{\boldsymbol{\Lambda}}^{\text{new}} \hat{\boldsymbol{\beta}} \tilde{\mathbf{S}}\right\}.
$$

Model CCU

$$\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\boldsymbol{\Psi}})^{-1},\ \hat{\boldsymbol{\Lambda}}^{\text{new}} = \tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}'\tilde{\boldsymbol{\Theta}}^{-1},\ \hat{\boldsymbol{\Psi}}^{\text{new}} = \operatorname{diag}\{\tilde{\mathbf{S}} - \hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\}.$$

Model CUC

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\psi}_g\mathbf{I}_p)^{-1},\ \hat{\boldsymbol{\Lambda}}^{\text{new}} = \left[\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\right]\left[\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\boldsymbol{\Theta}_g\right]^{-1},$$

$$(\hat{\psi}_g)^{\text{new}} = \frac{1}{p}\operatorname{tr}\{\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\}.$$

Model CUU

In this case the loading matrix must be solved in a row-by-row manner. This slows the fitting of this model to a great degree.

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\boldsymbol{\Psi}}_g)^{-1},\ \hat{\boldsymbol{\lambda}}_i^{\text{new}} = \mathbf{r}_i\left(\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_{g(i)}}\boldsymbol{\Theta}_g\right)^{-1},$$

$$\hat{\boldsymbol{\Psi}}_g^{\text{new}} = \operatorname{diag}\{\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\},$$

where $\hat{\boldsymbol{\lambda}}_i^{\text{new}}$ is the ith row of the matrix $\hat{\boldsymbol{\Lambda}}^{\text{new}}$, $\hat{\psi}_{g(i)}$ denotes the ith element along the diagonal of $\hat{\boldsymbol{\Psi}}_g$, $\mathbf{r}_i$ represents the ith row of the matrix $\sum_{g=1}^{G} n_g\hat{\boldsymbol{\Psi}}_g^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g'$ and $i = 1, 2, \ldots, p$.

Model UCC

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\psi}\mathbf{I}_p)^{-1},\ \hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},\ (\hat{\psi})^{\text{new}} = \frac{1}{p}\sum_{g=1}^{G}\hat{\pi}_g\operatorname{tr}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}.$$

Model UCU

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\boldsymbol{\Psi}})^{-1},\ \hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},\ \hat{\boldsymbol{\Psi}}^{\text{new}} = \sum_{g=1}^{G}\hat{\pi}_g\operatorname{diag}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}.$$

Model UUC

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\psi}_g\mathbf{I}_p)^{-1},\ \hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},\ (\hat{\psi}_g)^{\text{new}} = \frac{1}{p}\operatorname{tr}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}.$$

Model UUU

$$\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\boldsymbol{\Psi}}_g)^{-1},\ \hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},\ \hat{\boldsymbol{\Psi}}_g^{\text{new}} = \operatorname{diag}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}.$$

3.3.5 Convergence Criteria

Aitken's Acceleration

The Aitken's acceleration procedure, described in McLachlan & Krishnan (1997), was used to estimate the asymptotic maximum of the log-likelihood. This allowed a decision to be made on whether the EM algorithm had converged or not. The Aitken acceleration at iteration k is given by

$$a^{(k)} = \frac{l^{(k+1)} - l^{(k)}}{l^{(k)} - l^{(k-1)}},$$

where $l^{(k+1)}$, $l^{(k)}$ and $l^{(k-1)}$ are the log-likelihood values from iteration $k+1$, k and $k-1$ respectively. Then the asymptotic estimate of the log-likelihood at iteration $k+1$ is given by

$$l_{\infty}^{(k+1)} = l^{(k)} + \frac{1}{1 - a^{(k)}}(l^{(k+1)} - l^{(k)}).$$

Böhning *et al.* (1994) contend that the algorithm can be considered to have converged when

$$|l_{\infty}^{(k+1)} - l_{\infty}^{(k)}| < \epsilon,$$

where ϵ is a 'small' number. Lindsay (1995) give an alternative stopping criterion; that the algorithm can be stopped when

$$|l_{\infty}^{(k)} - l^{(k)}| < \epsilon. \tag{3.7}$$

The latter criterion is used herein, with $\epsilon = 10^{-2}$.

Lack of Progress

Some model-based clustering algorithms, such as that described by Fraley & Raftery (1998), use the difference in successive log-likelihoods as a convergence criterion. That is, the algorithm is considered to have converged when

$$|l^{(k+1)} - l^{(k)}| < \epsilon, \tag{3.8}$$

where ϵ is a 'small' number. The condition in Equation 3.8, however, is not a convergence criterion but rather an indication of lack of progress. Figure 3.1 shows a classically shaped plot of iteration number versus log-likelihood and it can be argued that in this case the criteria in equations 3.7 and 3.8 will give similar results.

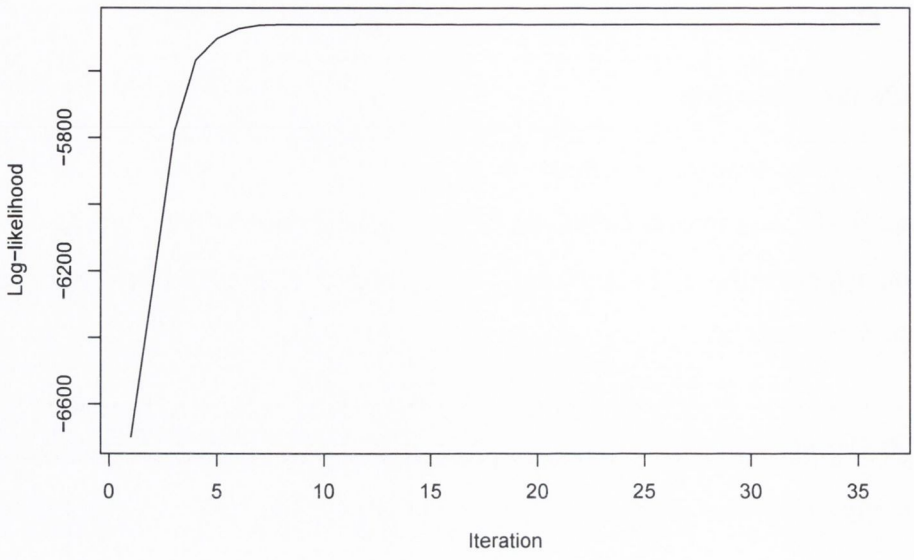


Figure 3.1: A 'classic' plot of iteration number versus log-likelihood for an AECM algorithm.

However, figures 3.2 and 3.3 illustrate situations where the difference between a lack of progress criterion and a genuine convergence criterion can become very apparent. In both of these cases, a lack of progress criterion might greatly underestimate the correct value of the log-likelihood.

Note that figures 3.1, 3.2 and 3.3 all arise from real analyses. In each case the data are the log-likelihoods at each iteration of an AECM algorithm for the CUU model, applied to the wine data that is introduced in Section 3.5.

3.3.6 Software

Software, called `pgmm`, was written in the C programming language to execute the AECM algorithm; that is, to find MLEs for the model parameters and the estimates of the group membership labels. The initial values for the elements of $\mathbf{\Lambda}_g$ were randomly generated from a uniform distribution on $[0, 1)$, while those for $\mathbf{\Psi}_g$ were some function of diag$\{\mathbf{S}_g\}$. It is a feature of `pgmm` that data are automatically standardised prior to analysis. Further details of the coding are given in Appendix D.

Finding the best method of initialising the z_{ng} was very difficult. Through trial-and-error it was discovered that the best method was to choose random values for $z_{ng} \in \{0, 1\}$ and

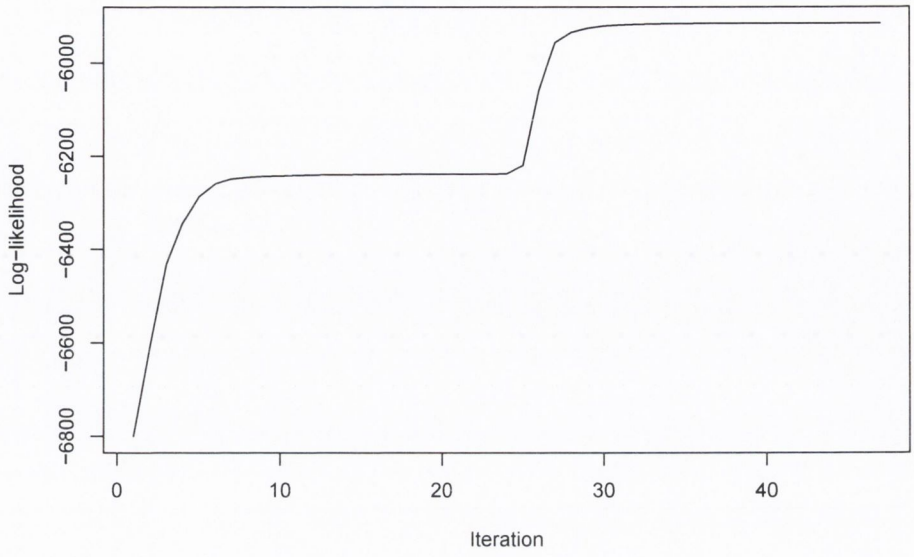


Figure 3.2: A plot of iteration number versus log-likelihood for an AECM algorithm, illustrating a single 'step'.

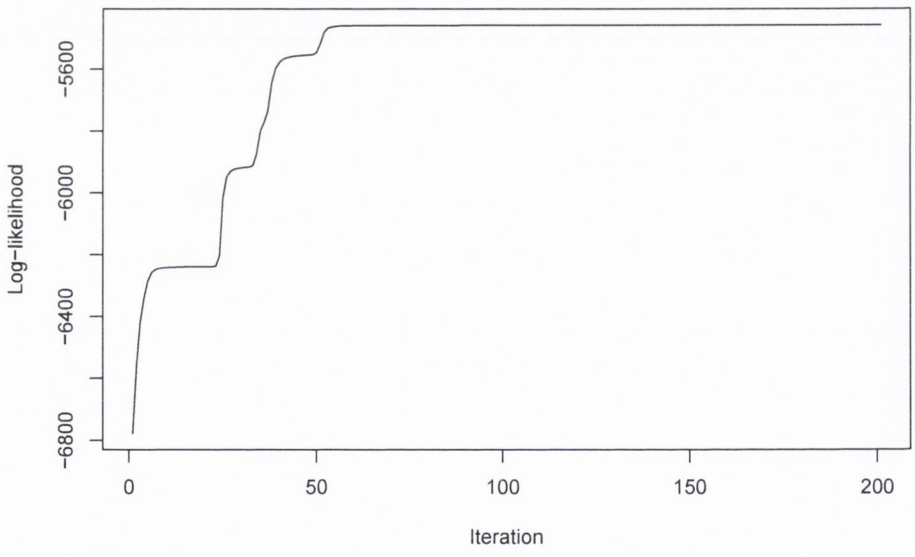


Figure 3.3: A plot of iteration number versus log-likelihood for an AECM algorithm, illustrating multiple 'steps'.

then to run the CCC model and take its output as the starting point for every other model. That said, this process was repeated multiple times with different random starting values of the z_{ng} to prevent a local, rather than global, maximum log-likelihood being attained.

The AECM algorithm has no 'hard classification' step; that is, $\hat{z}_{ng} \in [0, 1]$ at each iteration, rather than forcing $\hat{z}_{ng} \in \{0, 1\}$. A consequence of this feature is that at the end of the AECM algorithm, after convergence has been reached, it is not necessarily the case that each $\hat{z}_{ng}$ will take a value in $\{0, 1\}$. Therefore, `pgmm` classifies each case based on the maximum value of $\hat{z}_{ng}$ over $g = 1, 2, \ldots, G$.

The option to hard classify at each iteration may seem attractive and can lead to faster convergence in some cases. However, if a hard classification step is used at each iteration, the algorithm can become more prone to producing sub-optimal results. An example of an EM algorithm that involves a hard classification step is the classification EM algorithm (Celeux & Govaert, 1992).

3.3.7 Model Selection & Performance

The BIC was used to select an appropriate PGMM. Specifically, the BIC was used to choose the 'best' model from the members of the PGMM family (CCC, CCU, ..., UUU), the number of latent factors q and the number of mixture components, or groups, G. In addition, the BIC can be used to choose between the best PGMM and models arising from alternative model-based clustering techniques, such as MCLUST.

The performance of the PGMMs in revealing group structures in data is measured herein using the Rand index and the adjusted Rand index. These indices are also used herein to assess the performance of competing techniques, such as MCLUST.

3.4 Coffee Data

3.4.1 The Data

Streuli (1973) reported thirteen chemical properties of coffee from across 29 countries. These coffees were of two types — Robusta and Arabica. Twelve of the thirteen chemical properties that were recorded are given in Table 3.2.

The data were sourced from from `www.parvus.unige.it`. There was a thirteenth variable, Total Chlorogenic Acid, which was not used, both because it was not precisely the sum of

Table 3.2: Twelve of the thirteen chemical properties of coffee given by Streuli (1973).

Water	Bean Weight	Extract Yield
pH Value	Free Acid	Mineral Content
Fat	Caffeine	Trigonellin
Chlorogenic Acid	Neochlorogenic Acid	Isochlorogenic Acid

Chlorogenic Acid, Neochlorogenic Acid and Isochlorogenic Acid and because the information it contained was already given by these three variables.

3.4.2 The PGMM Family of Models

The PGMM family of models was fitted to these data for $G \in \{1, 2, \ldots, 5\}$ and $q \in \{1, 2, \ldots, 5\}$. Further, each model was run three times from different starting values of the group membership labels. The BIC was computed for all 600 of these models and the best member for each (G, q) is illustrated in Figure 3.4.

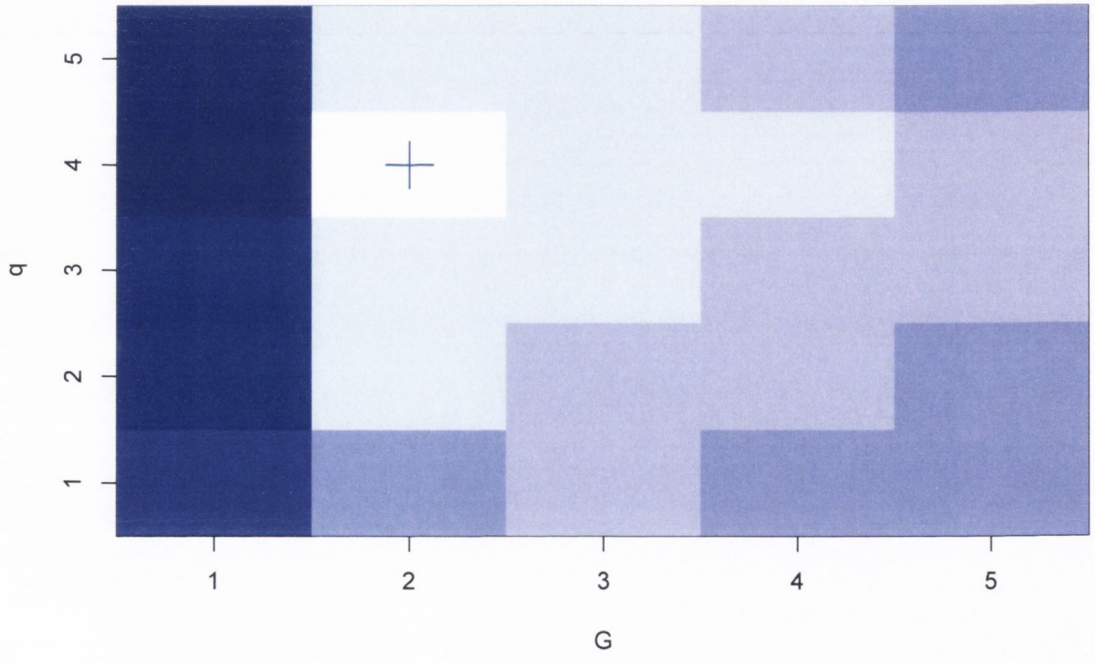


Figure 3.4: A 'heat map' giving the maximum BIC value for each PGMM at (G, q) for the coffee data.

The best model, in terms of BIC, was a CUU model with $G = 2$ and $q = 4$. The BIC for

this model was -1148.23. Table 3.3 gives the classification for this model, which is an exact match — two groups directly corresponding to the two varieties of coffee.

Table 3.3: Classification table for the best PGMM for the coffee data.

	1	2
Arabica	36	
Robusta		7

3.4.3 The MCLUST Family of Models

The `mclust` software for model-based clustering was also used to classify these data. Table 3.4 gives the classification for the best MCLUST model, which was a VEI model with three groups.

Table 3.4: Classification table for the best MCLUST model for the coffee data.

	1	2	3
Arabica	22	14	
Robusta			7

This classification could also be thought of as 'correct', however Arabica is split across two groups. The BIC for this model was -1301.70, which is less than the BIC for the best PGMM.

3.4.4 Model Comparison

Table 3.5 gives the Rand and adjusted Rand indices for both families of models that were applied to the coffee data. The chosen PGMM was the best model, in terms of BIC, Rand index and adjusted Rand index.

Table 3.5: Rand and adjusted Rand indices for both families of models that were applied to the coffee data.

Model	**Rand Index**	**Adjusted Rand Index**
PGMM	1.00	1.00
MCLUST	0.66	0.38

3.5 Italian Wine Data

3.5.1 The Data

Forina *et al.* (1986) used twenty-eight chemical properties of Italian wines from the Piedmont region to classify them into their specific type — Barolo, Grignolino or Barbera. Table 3.6 gives the subset of these thirteen variables that are available from the `gclus` library (Hurley, 2004) in R. Note that the variable 'malic acid' represents total malic acid; see Ooche *et al.* (1981) for further details.

Table 3.6: The thirteen chemical properties of wine, used by Forina *et al.* (1986), that are available in gclus.

Alcohol	Malic acid	Ash
Alcalinity of ash	Magnesium	Total ahenols
Flavanoids	Nonflavanoid phenols	Proanthocyanins
Color intensity	Hue	OD_{280}/OD_{315} of diluted wines
Proline		

3.5.2 The PGMM Family of Models

The family of PGMMs were fitted to the data for $G \in \{1, 2, \ldots, 5\}$ and $q \in \{1, 2, \ldots, 5\}$ by running the `pgmm` software from three random starting values, so that a total of 600 models were fitted. Figure 3.5 shows the BIC for the best PGMM for each couplet (G, q) over the three runs — the best model was a CUU model with $G = 4$ and $q = 2$. The BIC for this model was -5294.68.

A classification table for this best PGMM is given in Table 3.7. This model classifies the Barolo and Barbera varieties into clusters 1 and 4 respectively but Grignolino is spread mostly across clusters 2 and 3, with two observations in cluster 4.

Table 3.7: Classification table for the best PGMM for the wine data from `gclus`.

	1	2	3	4
Barolo	59			
Grignolino		38	31	2
Barbera				48

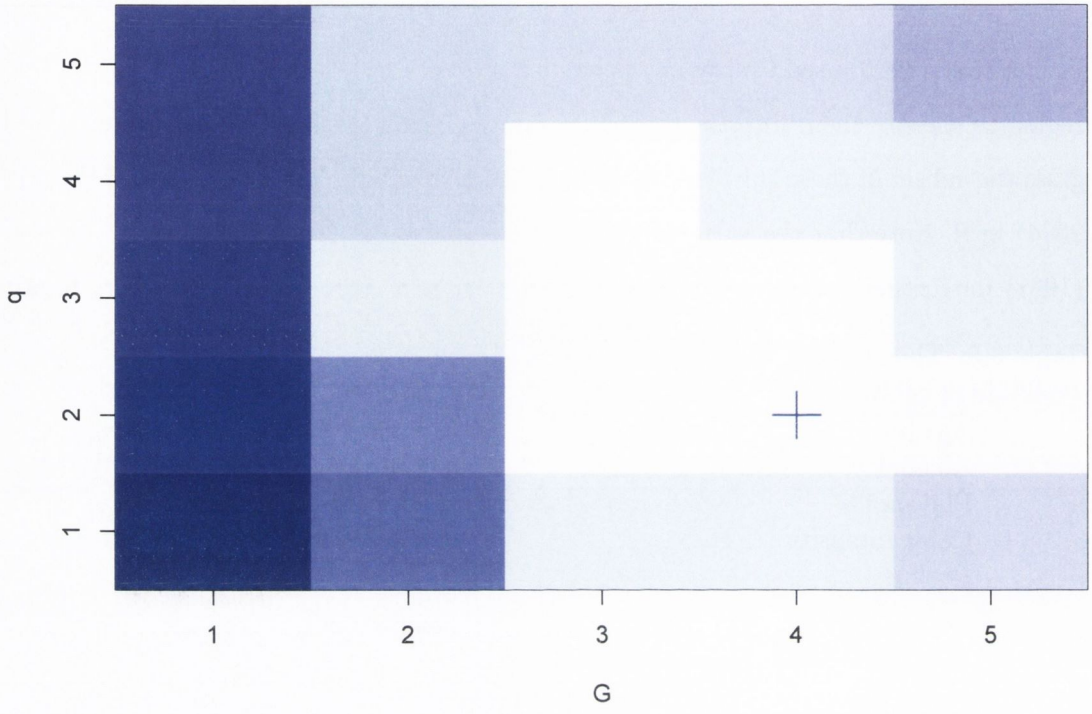


Figure 3.5: A 'heat map' giving the greatest BIC values for each PGMM at (G, q) for the wine data from `gclus`.

3.5.3 The MCLUST Family of Models

The best model using `mclust` was a VEI model with eight groups. The BIC for this model was -5469.95 and classifications for this model are given in Table 3.8.

Table 3.8: Classification table for the best MCLUST model for the wine data from `gclus`.

	1	2	3	4	5	6	7	8
Barolo	40	18	1					
Grignolino			21	22		27	1	
Barbera					4		17	27

3.5.4 Variable Selection

Variable selection was carried out using the `clustvarsel` package in R; the maximum number of groups was preset at eight. The variables selected were Malic acid, Proline, Flavanoids, colour intensity and OD_{280}/OD_{315} of diluted wines. The selected model was a VEV model with three groups and the classification results for variable selection are given in Table 3.9.

Table 3.9: Classification table for variable selection for the wine data from `glus`.

	1	2	3
Barolo	51	8	
Grignolino	3	67	1
Barbera		1	47

3.5.5 Model Comparison

Table 3.10 gives the Rand and adjusted Rand indices for all three models that were applied to the wine data. The best PGMM was the best model, in terms of both Rand index and adjusted Rand index.

Table 3.10: Rand and adjusted Rand indices for all of models that were applied to the wine data from `gclus`.

Model	**Rand Index**	**Adjusted Rand Index**
PGMM	0.91	0.79
MCLUST	0.80	0.48
Variable Selection	0.90	0.78

The best PGMM also had greater BIC than the best MCLUST model. No comparison can be made between the PGMM family of models and variable selection, which involves many `mclust` computations, in terms of BIC.

Furthermore, it is remarkable that the PGMM family of models has outperformed variable selection. The latter approach has the advantage that all of the 'unhelpful' variables are absent during the final clustering, yet it is still outperformed by the former. Rather than disregard variables, the PGMM approach weights variables appropriately using the elements of the load matrices $\mathbf{\Lambda}_g$. This approach is not only more natural than that of variable

selection, but based on the results of this analysis it may be superior as a model-based clustering technique.

3.5.6 Deeper into the Italian Wine Dataset

The wines in this study were from the years 1970–1979, as shown in Table 3.11.

Table 3.11: The wine data from `gclus` ordered by year of production.

	70	71	72	73	74	75	76	77	78	79
Barolo		19		20	20					
Grignolino	9	9	7	9	16	9	12			
Barbera					9		5		29	5

Now, returning to the PGMM results and assuming that the data are in year order, a more detailed classification of the PGMM results emerges and is given in Table 3.12. It is apparent, from Table 3.12, that clusters 2 and 3 are almost grouped by year; Cluster 2 comprising mainly Grignolino wines from 1970–1974 and Cluster 3 consisting primarily of Grignolino wines from 1974–1976.

Table 3.12: Classification table for the best PGMM model for the wine data from `gclus`, ordered by year.

Cluster	Barolo			Grignolino							Barbera			
	71	73	74	70	71	72	73	74	75	76	74	76	78	79
1	19	20	20											
2				7	8	4	8	8	2	1				
3				2	1	2	1	8	7	10				
4						1				1	9	5	29	5

Furthermore, applying the information in Table 3.11 to the results of the MCLUST and variable selection analysis gave results that were not interpretable in this fashion.

3.6 The Larger Wine Dataset

3.6.1 The Remaining Variables

The remaining 15 variables used by Forina *et al.* (1986) are given in Table 3.13. All but one of these 28 variables were sourced from `www.parvus.unige.it`; the variable Sulphate was not available.

Table 3.13: The fifteen chemical properties of wine, used by Forina *et al.* (1986), that were not available in gclus.

Sugar-free extract	Fixed acidity	Tartaric acid	Uronic acids
pH	Potassium	Calcium	Phosphate
Chloride	Sulphate	Glycerol	OD_{280}/OD_{315} of flavanoids
2,3-butanediol	Total nitrogen	Methanol	

3.6.2 The PGMM Family of Models

The family of PGMMs was fitted to this larger wine dataset for $G \in \{1, 2, \ldots, 5\}$ and $q \in \{1, 2, \ldots, 5\}$ and `pgmm` was set to loop three times so that a total of 600 models were fitted. Figure 3.6 shows the BIC for the best PGMM for each couplet (G, q) — the best model overall was a CUU model with $G = 3$, $q = 4$ and a BIC of -11454.32.

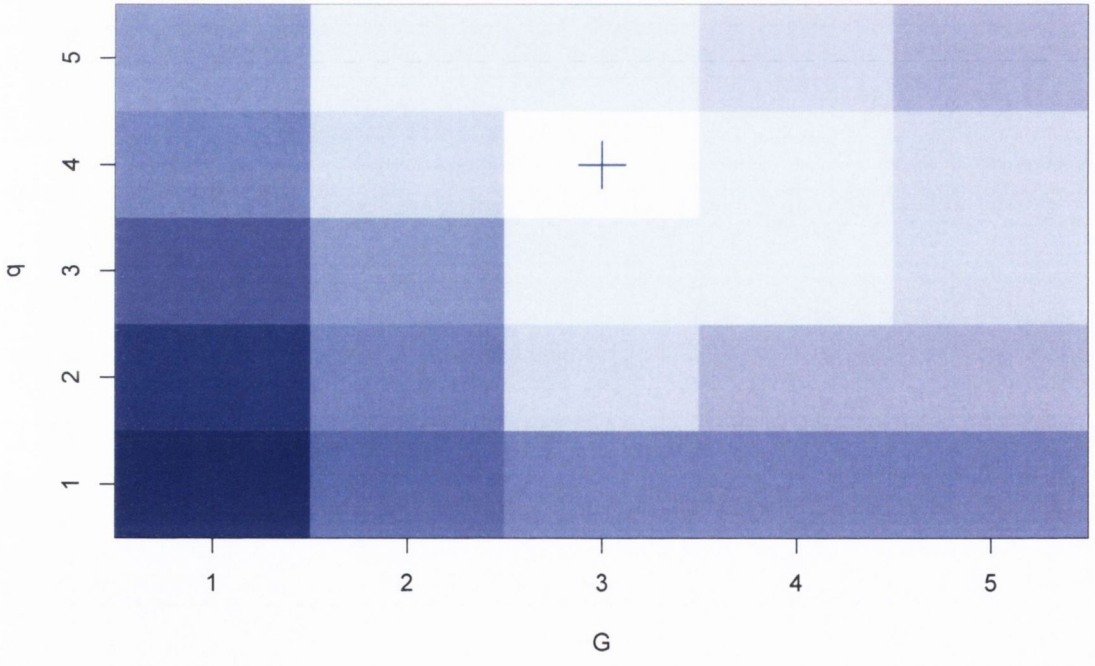


Figure 3.6: A 'heat map' giving the greatest BIC values for each PGMM at (G, q) for the larger wine dataset.

A classification table for this model is given in Table 3.14. This model classifies the Barolo

and Barbera into clusters 1 and 3 respectively and gets the Grignolino almost right, misclassifying only one sample.

Table 3.14: Classification table for the best PGMM for the larger wine dataset.

	1	2	3
Barolo	59		
Grignolino		70	1
Barbera			48

3.6.3 The MCLUST Family of Models

Classifications for the best model, in terms of BIC, using the `mclust` software in R are given in Table 3.15.

Table 3.15: Classification table for the best MCLUST model for the larger wine dataset.

	1	2	3
Barolo	58	1	
Grignolino	4	66	1
Barbera			48

The best MCLUST model was a VVI model with three groups and a BIC value of -12119.31.

3.6.4 Variable Selection

Variable selection was carried out using `clustvarsel` and, once again, the maximum number of groups was preset at eight. Nineteen variables were selected and they are given in Table 3.16.

Table 3.16: Variables selected for the larger wine dataset.

Chloride	Malic acid	Flavanoids	Proline
Colour intensity	Uronic acids	Magnesium	2-3-butanediol
Tartaric acid	Total nitrogen	Calcium	Alcalinity of ash
Hue	OD_{280}/OD_{315} of diluted wines	Methanol	Nonflavanoid phenols
Alcohol	Sugar-free extract	Phosphate	

A VVI model with four groups was chosen and the classifications for this model are given in Table 3.17.

Table 3.17: Classification table for variable selection on the larger wine dataset.

	1	2	3	4
Barolo	52	7		
Grignolino		17	54	
Barbera		1		47

3.6.5 Model Comparison

Table 3.18 gives the Rand and adjusted Rand indices for all three models that were applied to the wine data. The chosen PGMM was the best model, in terms of both Rand index and adjusted Rand index. Further, the best PGMM also outperformed the best MCLUST model in terms of BIC.

Table 3.18: Rand and adjusted Rand indices for all models applied to the larger wine dataset.

Model	**Rand Index**	**Adjusted Rand Index**
PGMM	0.99	0.98
MCLUST	0.95	0.90
Variable Selection	0.91	0.78

Interestingly, MCLUST performed better than variable selection on these data.

3.7 Discussion

A new family of PGMMs has been introduced. This family of models can be seen as a generalisation of the mixtures of factor analysers and mixtures of principal components analysers models and contains them both as special cases. The number of covariance parameters in this family of models grows linearly with data dimensionality, which is especially convenient in high-dimensional situations. This is not true of the MCLUST models, where the number of parameters in any of the diagonal covariance structures is linear or constant in the data dimensionality but is quadratic in data dimension for the non-diagonal covariance structures. The PGMM family therefore offers a more implementable modelling structure for high-dimensional data than MCLUST.

Chang (1983) shows that the principal components corresponding to the highest eigenvalues do not necessarily contain the most group information. Directly modelling data using

the PGMM family avoids the problems of implementing data reduction using principal components analysis before clustering. Modelling *via* the PGMM family also avoids the need for variable selection prior to, or concurrently with, model-based clustering.

The application of the PGMM family to the coffee and the wine data indicates that the models give excellent clustering performance. The clusters found using the PGMM family exhibit superior group structure-capturing of these data compared to other methods. In the case of the coffee data, choosing between the PGMM family and the MCLUST family, using the BIC, led to the selection a member of the PGMM family; this is in agreement with the results (Fraley & Raftery, 2002*b*) that show that choosing models within the MCLUST framework using the BIC gives models with good classification and clustering performance. The best PGMM also gave superior clustering performance on the wine data when compared to the best model using MCLUST or variable selection.

Although not reported, in the course of the analyses featured in this chapter, it was noted that the 'best' model using BIC was not always the best classifier. There is, therefore, future scope for finding an alternative to BIC for choosing the best member of the PGMM family of models and, perhaps, the MCLUST family of models. The ICL may present a meaningful alternative in this situation. Moreover, since the methods introduced herein are presented as unsupervised, it would be inappropriate to use known classifications for model selection.

Chapter 4

Generalised Parsimonious Gaussian Mixture Models

4.1 Structure

4.1.1 Redefining $\boldsymbol{\Psi}_g$

The family of PGMMs introduced in Chapter 3 are extended by further parameterising the group covariance structure. This further parameterisation comes about by writing

$$\boldsymbol{\Psi}_g = \omega_g \boldsymbol{\Delta}_g,$$

where $\omega_g \in \mathbb{R}$ and $\boldsymbol{\Delta}_g = \text{diag}\{\delta_1, \delta_2, \ldots, \delta_p\}$ such that $|\boldsymbol{\Delta}_g| = 1$, for $g \in \{1, 2, \ldots, G\}$. This new family of models will be referred to as generalised parsimonious Gaussian mixture models (GPGMMs).

4.1.2 The Q Equation

Now, the equation for Q given in Section 3.3.3 can be written as

$$Q(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}_g) = C + \sum_{g=1}^{G} n_g \left[\frac{1}{2} \log \left| (\omega_g \boldsymbol{\Delta}_g)^{-1} \right| - \frac{1}{2} \text{tr} \left\{ (\omega_g \boldsymbol{\Delta}_g)^{-1} \mathbf{S}_g \right\} + \text{tr} \left\{ (\omega_g \boldsymbol{\Delta}_g)^{-1} \boldsymbol{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g \right\} \right.$$
$$\left. - \frac{1}{2} \text{tr} \left\{ \boldsymbol{\Lambda}_g' (\omega_g \boldsymbol{\Delta}_g)^{-1} \boldsymbol{\Lambda}_g \boldsymbol{\Theta}_g \right\} \right],$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\boldsymbol{\Lambda}_g$, ω_g and $\boldsymbol{\Delta}_g$, $\boldsymbol{\Theta}_g = (\mathbf{I}_q - \hat{\boldsymbol{\beta}}_g\hat{\boldsymbol{\Lambda}}_g + \hat{\boldsymbol{\beta}}_g\mathbf{S}_g\hat{\boldsymbol{\beta}}_g')$ is a symmetric $q \times q$ matrix and $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\omega}_g\boldsymbol{\Delta}_g)^{-1}$. Now following from Corollary A.1 of Appendix A;

$$\log|(\omega_g\boldsymbol{\Delta}_g)^{-1}| = \log|\omega_g^{-1}\boldsymbol{\Delta}_g^{-1}| = \log\left[\omega_g^{-p}|\boldsymbol{\Delta}_g^{-1}|\right] = p\log\omega_g^{-1} + \log|\boldsymbol{\Delta}_g^{-1}|,$$

and so Q can be written as

$$\begin{aligned} Q(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}_g) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \Bigg[& p\log\omega_g^{-1} + \log|\boldsymbol{\Delta}_g^{-1}| - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}_g^{-1}\mathbf{S}_g\} \\ & + 2\omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}_g^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Delta}_g^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\Bigg]. \end{aligned} \tag{4.1}$$

4.2 Parameter Estimates for GPGMMs

4.2.1 Estimating the Parameters

Estimation of the model parameters, *via* the AECM algorithm, is analogous to that of the PGMM parameters of Section 3.3. The number of covariance parameters associated with each GPGMM is given in Table 4.1.

Table 4.1: Covariance structures and number of covariance parameters for each member of the GPGMM family.

Model	$\boldsymbol{\Lambda}_g = \boldsymbol{\Lambda}$	$\boldsymbol{\Delta}_g = \boldsymbol{\Delta}$	$\omega_g = \omega$	$\boldsymbol{\Delta} = I$	Covariance Parameters
CCCC	Yes	Yes	Yes	Yes	$[pq - q(q-1)/2] + 1$
CCUC	Yes	Yes	No	Yes	$[pq - q(q-1)/2] + G$
UCCC	No	Yes	Yes	Yes	$G[pq - q(q-1)/2] + 1$
UCUC	No	Yes	No	Yes	$G[pq - q(q-1)/2] + G$
CCCU	Yes	Yes	Yes	No	$[pq - q(q-1)/2] + p$
CCUU	Yes	Yes	No	No	$[pq - q(q-1)/2] + [G + (p-1)]$
UCCU	No	Yes	Yes	No	$G[pq - q(q-1)/2] + p$
UCUU	No	Yes	No	No	$G[pq - q(q-1)/2] + [G + (p-1)]$
CUCU	Yes	No	Yes	No	$[pq - q(q-1)/2] + [1 + G(p-1)]$
CUUU	Yes	No	No	No	$[pq - q(q-1)/2] + Gp$
UUCU	No	No	Yes	No	$G[pq - q(q-1)/2] + [1 + G(p-1)]$
UUUU	No	No	No	No	$G[pq - q(q-1)/2] + Gp$

Table 4.1 contains a total of four new models when compared to Table 3.1. The equivalences between the models in these tables, that is the equivalences between the PGMM family and the GPGMM family, are shown in Table 4.2.

Table 4.2: Equivalences between PGMMs and GPGMMs.

PGMM	GPGMM	PGMM	GPGMM
CCC	C*CC*C	UCU	U*CC*U
CUC	C*CU*C	–	U*CU*U
UCC	U*CC*C	–	C*UC*U
UUC	U*CU*C	CUU	C*UU*U
CCU	C*CC*U	–	U*UC*U
–	C*CU*U	UUU	U*UU*U

From Table 4.2, the four new models are CCUU, CUCU, UCUU and UUCU. The derivation of the model parameter estimates for the CCUU model are demonstrated in Section 4.2.3 and details for the remaining three new models are given in Appendix B.

The estimates for the eight pre-existing models are obtained from the PGMM estimates given in Section 3.3.4 by writing

$$\boldsymbol{\Psi}_g = |\boldsymbol{\Psi}_g|^{1/g} \frac{\boldsymbol{\Psi}_g}{|\boldsymbol{\Psi}_g|^{1/g}},$$

and setting

$$\omega_g = |\boldsymbol{\Psi}_g|^{1/g} \text{ and } \boldsymbol{\Delta}_g = \frac{\boldsymbol{\Psi}_g}{|\boldsymbol{\Psi}_g|^{1/g}}.$$

Before outlining the derivation of the MLEs of the model parameters for the CCUU model, it is necessary to introduce the method of Lagrange multipliers.

4.2.2 Method of Lagrange Multipliers

Lagrange (1788) introduced a method of finding local extrema of a function subject to some constraint which is also expressed as a function. This method, which is summarised here, is described in detail by Fraleigh (1990).

In order to find local extrema of a function $f(x, y, z)$ subject to the constraint $g(x, y, z) = 0$, use the fact that there exists some constant λ such that

$$\nabla f = \lambda(\nabla g),$$

where λ is called the 'Lagrange multiplier'. This, along with the constraint $g(x, y, z) = 0$, leads to what Fraleigh (1990) calls the "conditions of the method of Lagrange multipliers";

$$\frac{\partial f}{\partial x} = \lambda \frac{\partial g}{\partial x}, \quad \frac{\partial f}{\partial y} = \lambda \frac{\partial g}{\partial y}, \quad \frac{\partial f}{\partial z} = \lambda \frac{\partial g}{\partial z} \quad \text{and} \quad g(x, y, z) = 0.$$

Furthermore, writing

$$L(x, y, z, \lambda) = f(x, y, z) - \lambda g(x, y, z),$$

makes these conditions equivalent to the conditions

$$\frac{\partial L}{\partial x} = 0, \quad \frac{\partial L}{\partial y} = 0, \quad \frac{\partial L}{\partial z} = 0 \quad \text{and} \quad \frac{\partial L}{\partial \lambda} = 0.$$

4.2.3 Parameter Estimates for the CCUU Model

For the CCUU model, $\boldsymbol{\Lambda}_g = \boldsymbol{\Lambda}$ and $\boldsymbol{\Delta}_g = \boldsymbol{\Delta}$. Therefore, $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\omega}_g\hat{\boldsymbol{\Delta}})^{-1}$ and from Equation 4.1, Q can be written

$$Q(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \Big[p \log \omega_g^{-1} + \log|\boldsymbol{\Delta}^{-1}| - \omega_g^{-1} \operatorname{tr}\{\boldsymbol{\Delta}^{-1}\mathbf{S}_g\} + 2\omega_g^{-1} \operatorname{tr}\{\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \omega_g^{-1} \operatorname{tr}\{\boldsymbol{\Lambda}'\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}_g\} \Big].$$

Now, form

$$L(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = Q(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}) - \lambda(|\boldsymbol{\Delta}| - 1),$$

and differentiating L with respect to $\boldsymbol{\Lambda}$, ω_g^{-1}, $\boldsymbol{\Delta}^{-1}$ and λ respectively gives the following score functions.

$$S_1(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \boldsymbol{\Lambda}} = \sum_{g=1}^{G} \frac{n_g}{\omega_g} \Big[\boldsymbol{\Delta}^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}\tilde{\boldsymbol{\Theta}}_g \Big],$$

$$S_2(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \omega_g^{-1}} = \frac{n_g}{2} \Big[p\omega_g - \operatorname{tr}\{\boldsymbol{\Delta}^{-1}\mathbf{S}_g\} + 2\operatorname{tr}\{\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \operatorname{tr}\{\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}_g\boldsymbol{\Lambda}'\} \Big],$$

$$S_3(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \boldsymbol{\Delta}^{-1}} = \frac{1}{2}\sum_{g=1}^{G} n_g \Big[\boldsymbol{\Delta} - \omega_g^{-1}\mathbf{S}_g' + 2\omega_g^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \omega_g^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}_g'\boldsymbol{\Lambda}' \Big] + \lambda|\boldsymbol{\Delta}|\boldsymbol{\Delta},$$

$$S_4(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \lambda} = |\boldsymbol{\Delta}| - 1.$$

Note that S_4 is included for completeness only and solving $S_4(\boldsymbol{\Lambda}, \omega_g, \boldsymbol{\Delta}, \lambda) = 0$ just returns the constraint $|\boldsymbol{\Delta}| = 1$. Now, solving $S_1(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\omega}_g, \hat{\boldsymbol{\Delta}}, \lambda) = 0$, in an analogous fashion to the CUC solution in Appendix A.4.3, gives

$$\hat{\boldsymbol{\Lambda}}^{\text{new}} = \left[\sum_{g=1}^{G} \frac{n_g}{\hat{\omega}_g}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' \right] \left[\sum_{g=1}^{G} \frac{n_g}{\hat{\omega}_g}\boldsymbol{\Theta}_g \right]^{-1}.$$

Again, imitating the PGMM solution of Appendix A.4.3, solving $S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\omega}_g^{\text{new}}, \hat{\boldsymbol{\Delta}}, \lambda) = 0$ gives

$$(\hat{\omega}_g)^{\text{new}} = \frac{1}{p}\operatorname{tr}\left\{\hat{\boldsymbol{\Delta}}^{-1}\left[\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\right]\right\},$$

and solving $\operatorname{diag}\{S_3(\hat{\boldsymbol{\Lambda}}^{\text{new}}, (\hat{\omega}_g)^{\text{new}}, \hat{\boldsymbol{\Delta}}^{\text{new}}, \lambda)\} = 0$ gives

$$\frac{n}{2}\hat{\boldsymbol{\Delta}}^{\text{new}} - \frac{1}{2}\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\operatorname{diag}\left\{\mathbf{S}_g' - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\right\} + \lambda|\hat{\boldsymbol{\Delta}}^{\text{new}}|\hat{\boldsymbol{\Delta}}^{\text{new}} = 0$$

$$\Rightarrow \quad \hat{\boldsymbol{\Delta}}^{\text{new}} = \frac{1}{n + 2\lambda|\hat{\boldsymbol{\Delta}}^{\text{new}}|}\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\operatorname{diag}\left\{\mathbf{S}_g' - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\right\}$$

$$= \frac{1}{n + 2\lambda}\operatorname{diag}\left\{\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\left[\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\right]\right\}.$$

But $\hat{\boldsymbol{\Delta}}^{\text{new}}$ is a diagonal matrix with $|\hat{\boldsymbol{\Delta}}^{\text{new}}| = 1$, therefore

$$n + 2\lambda = \left(\prod_{j=1}^{p}\xi_j\right)^{\frac{1}{p}},$$

where ξ_j is the jth element along the diagonal of the matrix

$$\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\left[\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\right],$$

and it follows that

$$\lambda = \frac{1}{2}\left[\left(\prod_{j=1}^{p}\xi_j\right)^{\frac{1}{p}} - n\right].$$

4.3 Leptograpsus Crabs Data

4.3.1 The Data

The data consist of biological measurements on 200 crabs — 50 male and 50 female, for each of two species; 100 orange and 100 blue. The data were sourced from the `MASS` library in R; which contains datasets from Venables & Ripley (1999). There are five variables in these data, corresponding to the five measurements that were taken on each crab; these measurements are given in Table 4.3.

Table 4.3: Details of the five measurements that were taken on the leptograpsus crabs.

Variable	Measurement
FL	Frontal lobe size in millimeters.
RW	Rear width in millimeters.
CL	Carapace length in millimeters.
CW	Carapace width in millimeters.
BD	Body depth in millimeters.

These data first appeared in Campbell & Mahon (1974) and were subsequently analysed by Ripley (1996), McLachlan & Peel (1998, 2000*a*) and Raftery & Dean (2006). These data were selected for analysis herein because the results will be comparable with the MCLUST and variable selection analyses of Raftery & Dean (2006).

4.3.2 The GPGMM Family of Models

The `pgmm` software was expanded to include the four new models. The family of GPGMMs were fitted to the data for $G \in \{1, 2, \ldots, 5\}$ and $q \in \{1, 2, \ldots, 5\}$ by running the `pgmm` software from three random starting values, so that a total of 900 models were fitted. Figure 4.1 shows the BIC for the best GPGMM for each couplet (G, q) over the three runs.

The best model was a UCUU model with $G = 4$ and $q = 1$. The BIC for this model was 201.79 and a classification table for this best GPGMM is given in Table 4.4.

Table 4.4: Classification table for the best GPGMM for the crabs data.

		1	2	3	4
Blue	Male	40	10		
	Female		50		
Orange	Male			50	
	Female			4	46

Looking at a pairs plot of the variables in the crabs data, given in Figure 4.2, it becomes apparent that a linear relationship exists between each of the five variables. Therefore, it is natural that the best GPGMM had one latent variable; that is, one linear combination of the variables.

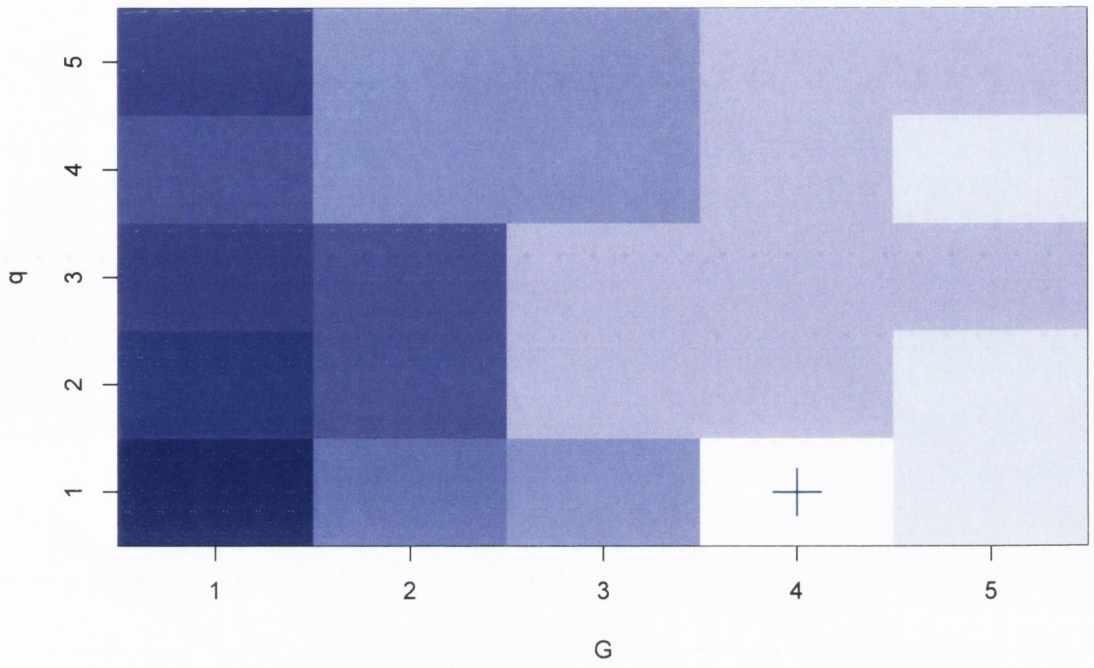


Figure 4.1: A 'heat map' giving the greatest BIC values for each GPGMM at (G, q) for the crabs data.

4.3.3 The MCLUST Family of Models & Variable Selection

Raftery & Dean (2006) report the results of applying MCLUST to the crabs data. Although they do not state a BIC value, they do report a classification table, which is given in Table 4.5.

Table 4.5: Classification table from the MCLUST analysis of the crabs data.

		1	2	3	4	5	6	7
Blue	Male	32					18	
	Female		31				19	
Orange	Male			28				22
	Female				24	21		5

Raftery & Dean (2006) also used variable selection to analyse the crabs data; the classification table for this analysis is given in Table 4.6.

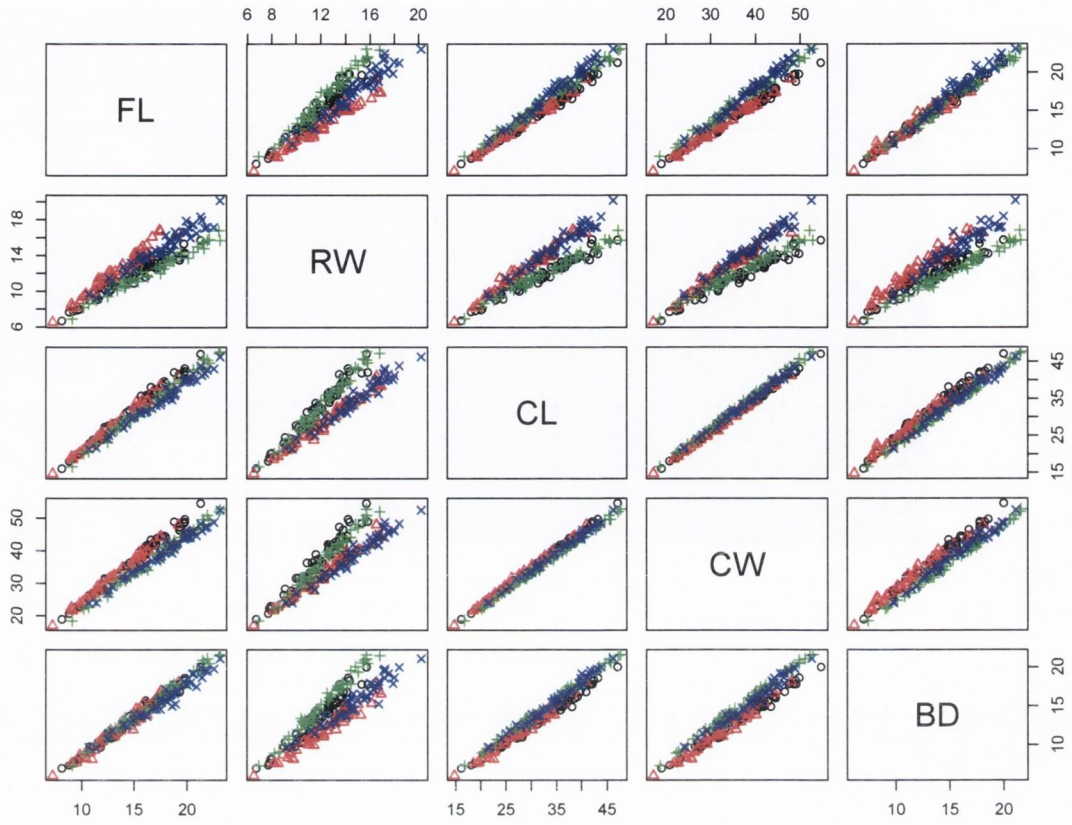


Figure 4.2: A pairs plot of the crabs data, constructed using R

Table 4.6: Classification table from the variable selection analysis of the crabs data.

		1	2	3	4
Blue	Male	40	10		
	Female		50		
Orange	Male			50	
	Female			5	45

4.3.4 Model Comparison

Table 4.7 gives the Rand and adjusted Rand indices and the error rate for the three models that were applied to the crabs data. The error rate is included here because it was used by Raftery & Dean (2006) for model comparison.

The best GPGMM was the best model; it had the largest Rand and adjusted Rand indices and it had the smallest error rate. This is another remarkable result; the GPGMM family

Table 4.7: Rand and adjusted Rand indices and error rate for all of the models that were applied to the crabs data.

Model	Rand Index	Adjusted Rand Index	Error Rate
GPGMM	0.935	0.828	0.07
MCLUST	0.851	0.533	0.425
Variable Selection	0.931	0.815	0.075

of models has outperformed the far less parsimonious variable selection method, albeit by a small margin, and the MCLUST family of models.

Furthermore, the PGMM family of models also performed excellently when applied to these data. The best PGMM had greater Rand and adjusted Rand indices than the variable selection method but had the same error rate.

4.4 Summary

The PGMM family of models that was introduced in Chapter 3 has been extended from eight to twelve parsimonious Gaussian mixture models. This new family of models, the GPGMMs, retain the attractive feature that their number of covariance parameters is linear in the dimensionality of the data. The GPGMM family of models was applied to leptograpsus crabs data and gave excellent classification results when compared to the MCLUST and variable selection methods used by Raftery & Dean (2006) to analyse the same data.

The superior performance of the GPGMMs when compared to MCLUST and variable selection on the data analysed in this chapter and in Chapter 3 may be related, in part, to the different convergence criteria used. Running MCLUST and variable selection on these data, using Aitken's acceleration might give better results for these models. That said, the performance of the GPGMMs compared to variable selection is remarkable when one considers the relative simplicity of the GPGMM method.

Note that some of the models depicted on heat maps herein may not in fact be parsimonious; that is, some such models may have a greater number of covariance parameters than the 'raw' covariance matrices. No such models were removed because the penalty that the BIC imposes on the log-likelihood for lack of parsimoniousness ensured that no such model was selected as the 'best'.

Chapter 5

Parsimonious Gaussian Mixture Models with Cholesky-Decomposed Covariance Structure

5.1 Introduction

The ideas around model-based clustering, described in chapters 2 and 3, are developed *via* modified Cholesky decomposition of the covariance matrices $\mathbf{\Sigma}_g$. The resulting family of mixture models can be applied to classification problems involving longitudinal, or naturally ordered, data.

5.2 Background

5.2.1 Cholesky Decomposition

Cholesky decomposition (Benoît, 1924) is a method, often used in numerical analysis, for decomposing a matrix into the product of a lower triangular matrix and its transpose. More precisely, let $\mathbf{A}$ be a real, positive definite matrix. Then $\mathbf{A}$ can be decomposed uniquely, using the Cholesky decomposition, as

$$\mathbf{A} = \mathbf{L}\mathbf{L}',$$

where $\mathbf{L}$ is a lower triangular matrix.

In the numerical analysis application, Cholesky decomposition is used to decompose the matrix $\mathbf{A}$ in order to simplify the solution to a linear system of equations, like

$$\mathbf{Ax} = \mathbf{b}.$$

In this work, a Cholesky decomposition will be applied to the group covariance matrix in a mixture modelling framework. More precisely, a modified Cholesky decomposition will be used.

5.2.2 Modified Cholesky Decomposition

Pourahmadi (1999, 2000) exploited the fact that covariance matrix $\boldsymbol{\Sigma}$ of a random variable can be decomposed using the relation

$$\mathbf{T}\boldsymbol{\Sigma}\mathbf{T}' = \mathbf{D},$$

where $\mathbf{T}$ is a unique unit lower triangular matrix with diagonal elements $t_{ii} = 1$ and $\mathbf{D}$ is a unique diagonal matrix with strictly positive entries. An alternative version of this relationship is written

$$\boldsymbol{\Sigma}^{-1} = \mathbf{T}'\mathbf{D}^{-1}\mathbf{T}. \tag{5.1}$$

In these decompositions, the matrices $\mathbf{T}$ and $\mathbf{D}$ can be interpreted statistically in terms of an autoregressive model. This decomposition was also used by Pan & MacKenzie (2003, 2006), while Pourahmadi *et al.* (2007) extended this decomposition to account for multiple covariance matrices. This covariance decomposition, however, has not previously been applied in a model-based clustering context.

5.3 Parsimonious Gaussian Mixture Models with Cholesky-Decomposed Covariance Structure

5.3.1 The Model

We assume a mixture model with a Gaussian mixture component used to model each subpopulation, with a modified Cholesky-decomposed covariance structure. Therefore, the

density of an observation $\mathbf{x}_i$ in group g is

$$f(\mathbf{x}_i \mid \boldsymbol{\mu}_g, \mathbf{T}_g, \mathbf{D}_g) = \frac{1}{\sqrt{(2\pi)^p \mid (\mathbf{T}_g'\mathbf{D}_g^{-1}\mathbf{T}_g)^{-1} \mid}} \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_g)'\mathbf{T}_g'\mathbf{D}_g^{-1}\mathbf{T}_g(\mathbf{x}_i - \boldsymbol{\mu}_g)\right\},$$

where $\mathbf{T}_g$ is the lower triangular matrix and $\mathbf{D}_g$ is the diagonal matrix, that follow from the modified Cholesky decomposition of $\boldsymbol{\Sigma}_g$.

Now, there is the option to constrain the $\mathbf{T}_g$ or the $\mathbf{D}_g$ to be equal across groups, which leads to four parsimonious Gaussian mixture models, which shall be called Cholesky-Decomposed Gaussian Mixture Models (CDGMMs). Each CDGMM, along with their respective nomenclature and number of covariance parameters is given in Table 5.1.

Table 5.1: The covariance structure and number of covariance parameters for each CDGMM.

Model	$\mathbf{T}_g$	$\mathbf{D}_g$	Covariance Parameters
VV	Variable	Variable	$G[p(p-1)/2] + Gp$
VE	Variable	Equal	$G[p(p-1)/2] + p$
EV	Equal	Variable	$p(p-1)/2 + Gp$
EE	Equal	Equal	$p(p-1)/2 + p$

Note that two of the models in Table 5.1, EE and VV, are equivalent to models that already exist in the MCLUST framework. The EE model is equivalent to the EEE model and the VV model is equivalent to the VVV model.

5.3.2 Model Fitting

The CDGMMs are fitted using the ECM algorithm. The E-step is very similar to the first E-step of the AECM algorithm from Chapter 3. The complete-data likelihood for the mixture model is

$$\mathcal{L}(\pi_g, \boldsymbol{\mu}_g, \mathbf{T}_g, \mathbf{D}_g) = \prod_{i=1}^{n}\prod_{g=1}^{G}\left[\pi_g f(\mathbf{x}_i \mid \boldsymbol{\mu}_g, \mathbf{T}_g, \mathbf{D}_g)\right]^{z_{ig}},$$

where $z_{ig} = 1$ if $i = g$ and $z_{ig} = 0$ otherwise. Similarly to Q_1 from Section 3.3.3, the expected value of the complete-data log-likelihood for the mixture model is

$$\begin{aligned} Q(\pi_g, \boldsymbol{\mu}_g\mathbf{T}_g, \mathbf{D}_g) = &\sum_{g=1}^{G} n_g \log \pi_g - \frac{np}{2}\log 2\pi - \sum_{g=1}^{G}\frac{n_g}{2}\log\left|(\mathbf{T}_g'\mathbf{D}_g^{-1}\mathbf{T}_g)^{-1}\right| \\ &- \sum_{g=1}^{G}\frac{n_g}{2}\operatorname{tr}\left\{\mathbf{S}_g\mathbf{T}_g'\mathbf{D}_g^{-1}\mathbf{T}_g\right\}, \end{aligned} \tag{5.2}$$

where

$$\hat{z}_{ig} = \frac{\hat{\pi}_g f(\mathbf{x}_i \mid \hat{\boldsymbol{\mu}}_g, \hat{\mathbf{T}}_g, \hat{\mathbf{D}}_g)}{\sum_{g'=1}^{G} \hat{\pi}_{g'} f(\mathbf{x}_i \mid \hat{\boldsymbol{\mu}}_{g'}, \hat{\mathbf{T}}_{g'}, \hat{\mathbf{D}}_{g'})},$$

$n_g = \sum_{i=1}^{n} \hat{z}_{ig}$ and $\mathbf{S}_g = (1/n_g) \sum_{i=1}^{n} \hat{z}_{ig}(\mathbf{x}_i - \boldsymbol{\mu}_g)(\mathbf{x}_i - \boldsymbol{\mu}_g)'$. Now, maximising Q with respect to π_g and $\boldsymbol{\mu}_g$ gives

$$\hat{\boldsymbol{\mu}}_g = \frac{\sum_{i=1}^{n} \hat{z}_{ig}\mathbf{x}_i}{\sum_{i=1}^{n} \hat{z}_{ig}} \text{ and } \hat{\pi}_g = \frac{n_g}{n},$$

as in Section 3.3.3.

The following results (Lütkepohl, 1996, Section 4.2) were used to simplify the term

$$\sum_{g=1}^{G} \frac{n_g}{2} \log \left| (\mathbf{T}_g' \mathbf{D}_g^{-1} \mathbf{T}_g)^{-1} \right|,$$

in Equation 5.2.

$$\mathbf{A}_{m\times m}, \mathbf{B}_{m\times m} : \mid \mathbf{AB} \mid = \mid \mathbf{A} \mid\mid \mathbf{B} \mid$$

$$\mathbf{A}_{m\times m} : \mid \mathbf{A}' \mid = \mid \mathbf{A} \mid$$

$$\mathbf{A}_{m\times m}, \text{ nonsingular} : \mid \mathbf{A}^{-1} \mid = \mid \mathbf{A} \mid^{-1}$$

$$\mathbf{A}_{m\times m} = [a_{ij}], \text{ triangular} : \mid \mathbf{A} \mid = \prod_{i=1}^{m} a_{ii}$$

The simplification is achieved as follows.

$$\log \left| (\mathbf{T}_g' \mathbf{D}_g^{-1} \mathbf{T}_g)^{-1} \right| = \log \left| \mathbf{T}_g' \mathbf{D}_g^{-1} \mathbf{T}_g \right|^{-1} = \log \left\{ \left| \mathbf{T}_g' \right|^2 \left| \mathbf{D}_g^{-1} \right| \right\}^{-1}$$

$$= -2 \log \left| \mathbf{T}_g' \right| + \log \left| \mathbf{D}_g \right| = \log \left| \mathbf{D}_g \right|,$$

and it follows that

$$\sum_{g=1}^{G} \frac{n_g}{2} \log \left| (\mathbf{T}_g' \mathbf{D}_g^{-1} \mathbf{T}_g)^{-1} \right| = \sum_{g=1}^{G} \frac{n_g}{2} \log \left| \mathbf{D}_g \right|.$$

Now, it is also true that

$$\operatorname{tr}\{\mathbf{S}_g \mathbf{T}_g' \mathbf{D}_g^{-1} \mathbf{T}_g\} = \operatorname{tr}\{\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g' \mathbf{D}_g^{-1}\},$$

and so Equation 5.2 can be written

$$Q(\mathbf{T}_g, \mathbf{D}_g) = C - \sum_{g=1}^{G} \frac{n_g}{2} \log |\mathbf{D}_g| - \sum_{g=1}^{G} \frac{n_g}{2} \operatorname{tr}\{\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g' \mathbf{D}_g^{-1}\}, \tag{5.3}$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\mathbf{T}_g$ and $\mathbf{D}_g$. The parameter estimates for the VV model are derived in Section 5.3.3 and estimates for the parameters of the other models are given in Appendix C.

5.3.3 Parameter Estimates for the VV Model

Differentiating Equation 5.3 with respect to $\mathbf{T}_g$ and $\mathbf{D}_g^{-1}$ respectively gives the following score functions.

$$S_1(\mathbf{T}_g, \mathbf{D}_g) = \frac{\partial Q(\mathbf{T}_g, \mathbf{D}_g)}{\partial \mathbf{T}_g} = -\frac{n_g}{2}\left[\left(\mathbf{D}_g^{-1}\right)' \mathbf{T}_g \mathbf{S}_g' + \mathbf{D}_g^{-1}\mathbf{T}_g\mathbf{S}_g\right] = -n_g\mathbf{D}_g^{-1}\mathbf{T}_g\mathbf{S}_g.$$

$$S_2(\mathbf{T}_g, \mathbf{D}_g) = \frac{\partial Q(\mathbf{T}_g, \mathbf{D}_g)}{\partial \mathbf{D}_g^{-1}} = \frac{n_g}{2}\mathbf{D}_g - \frac{n_g}{2}\left(\mathbf{T}_g\mathbf{S}_g\mathbf{T}_g'\right)' = \frac{n_g}{2}\left(\mathbf{D}_g - \mathbf{T}_g\mathbf{S}_g\mathbf{T}_g'\right).$$

Solving $S_1(\hat{\mathbf{T}}_g, \mathbf{D}_g)$ is not entirely straightforward because it is only the lower triangular part of $\mathbf{T}_g$ that needs to be estimated. Following Pourahmadi *et al.* (2007), ϕ_{ij} is used to denote those elements of $\mathbf{T}_g$ that are to be estimated, so that

$$\mathbf{T}_g = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ \phi_{21}^{(g)} & 1 & 0 & 0 & \cdots & 0 \\ \phi_{31}^{(g)} & \phi_{32}^{(g)} & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & & \ddots & & \vdots \\ \phi_{p-1,1}^{(g)} & \phi_{p-1,2}^{(g)} & \cdots & \phi_{p-1,p-2}^{(g)} & 1 & 0 \\ \phi_{p1}^{(g)} & \phi_{p3}^{(g)} & \cdots & \phi_{p,p-2}^{(g)} & \phi_{p,p-1}^{(g)} & 1 \end{pmatrix}, \tag{5.4}$$

and write $S_1(\mathbf{T}_g, \mathbf{D}_g) \equiv S_1(\Phi_g, \mathbf{D}_g)$, where $\Phi_g = \{\phi_{ij}^{(g)}\}$ for $i > j$; $i, j \in \{1, 2, \ldots, p\}$. Also, let $\mathrm{LT}\{\cdot\}$ denote the lower triangular part of a matrix. Now, solving $\mathrm{LT}\{S_1(\hat{\Phi}_g), \mathbf{D}_g\} = 0$ for $\hat{\Phi}_g$ leads to a total of $p-1$ linear equations. The 1×1 solution is straightforward;

$$\frac{s_{11}^{(g)}\hat{\phi}_{21}^{(g)}}{\hat{d}_{22}^{(g)}} + \frac{s_{21}^{(g)}}{\hat{d}_{22}^{(g)}} = 0,$$

and so

$$\hat{\phi}_{21}^{(g)} = -\frac{s_{21}^{(g)}}{s_{11}^{(g)}}.$$

The 2×2 case involves the equations

$$\frac{s_{11}^{(g)}\hat{\phi}_{31}^{(g)}}{\hat{d}_{33}^{(g)}} + \frac{s_{21}^{(g)}\hat{\phi}_{32}^{(g)}}{\hat{d}_{33}^{(g)}} + \frac{s_{31}^{(g)}}{\hat{d}_{33}^{(g)}} = 0$$

$$\frac{s_{12}^{(g)}\hat{\phi}_{31}^{(g)}}{\hat{d}_{33}^{(g)}} + \frac{s_{22}^{(g)}\hat{\phi}_{32}^{(g)}}{\hat{d}_{33}^{(g)}} + \frac{s_{32}^{(g)}}{\hat{d}_{33}^{(g)}} = 0$$

which can be expressed in the form

$$\begin{pmatrix} s_{11}^{(g)} & s_{21}^{(g)} \\ s_{12}^{(g)} & s_{22}^{(g)} \end{pmatrix} \begin{pmatrix} \hat{\phi}_{31}^{(g)} \\ \hat{\phi}_{32}^{(g)} \end{pmatrix} = - \begin{pmatrix} s_{31}^{(g)} \\ s_{32}^{(g)} \end{pmatrix}, \tag{5.5}$$

and so

$$\begin{pmatrix} \hat{\phi}_{31}^{(g)} \\ \hat{\phi}_{32}^{(g)} \end{pmatrix} = - \begin{pmatrix} s_{11}^{(g)} & s_{21}^{(g)} \\ s_{12}^{(g)} & s_{22}^{(g)} \end{pmatrix}^{-1} \begin{pmatrix} s_{31}^{(g)} \\ s_{32}^{(g)} \end{pmatrix}. \tag{5.6}$$

It follows that the solution to the general $(r-1) \times (r-1)$ system of equations will be given by

$$\begin{pmatrix} \hat{\phi}_{r1}^{(g)} \\ \hat{\phi}_{r2}^{(g)} \\ \vdots \\ \hat{\phi}_{r,r-1}^{(g)} \end{pmatrix} = - \begin{pmatrix} s_{11}^{(g)} & s_{21}^{(g)} & \cdots & s_{r-1,1}^{(g)} \\ s_{12}^{(g)} & s_{22}^{(g)} & \cdots & s_{r-1,2}^{(g)} \\ \vdots & \vdots & \ddots & \vdots \\ s_{1,r-1}^{(g)} & s_{2,r-2}^{(g)} & \cdots & s_{r-1,r-1}^{(g)} \end{pmatrix}^{-1} \begin{pmatrix} s_{r1}^{(g)} \\ s_{r2}^{(g)} \\ \vdots \\ s_{r,r-1}^{(g)} \end{pmatrix}, \tag{5.7}$$

for $r = 2, 3, \ldots, p$.

Solving $\text{diag}\{S_2(\hat{\mathbf{T}}_g, \hat{\mathbf{D}}_g)\} = 0$ is much more straightforward, giving

$$\hat{\mathbf{D}}_g = \text{diag}\{\hat{\mathbf{T}}_g \mathbf{S}_g \hat{\mathbf{T}}_g'\},$$

which is a very natural estimator.

5.4 Rats Data

5.4.1 The Data

Data on the body weights of rats on one of three different dietary supplements were published in Crowder & Hand (1990). A total of 16 rats were put on one of three different diets; eight rats were on Diet 1, four were put on Diet 2 and four on Diet 3. Weights were first recorded after a settling-in period and then weekly for a period of nine weeks. An extra measurement was taken at 44 days to help gauge the effect of a treatment that occurred during the sixth week. These measurements can be seen on the time series plot in Figure 5.1

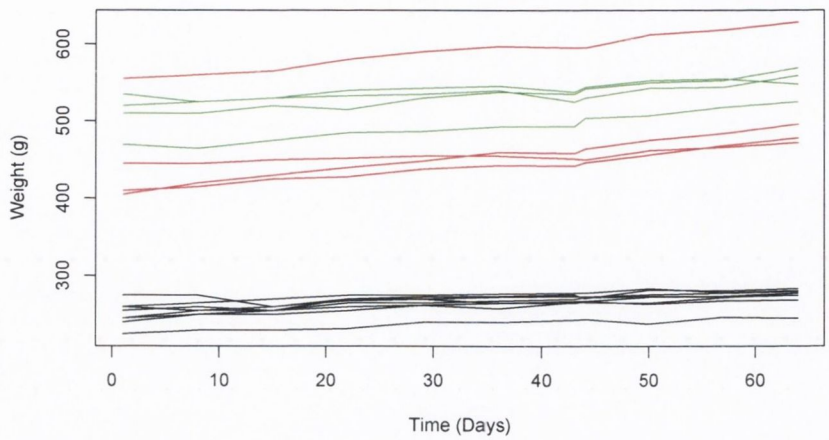


Figure 5.1: Time series for each rat, coloured by group — black for rats from Diet 1, red for Diet 2 and green for Diet 3.

5.4.2 Analysis

An ECM algorithm for each member of the CDGMM family was implemented in C, similarly to the `pgmm` software. As with the PGMM and GPGMM analyses, the Aitken acceleration convergence criterion was used for this ECM algorithm. Each member of the CDGMM family of models was fitted to the data for $G = 1, 2, \ldots, 6$.

5.4.3 Results

Using the BIC as a model selection criterion, the best CDGMM was an EE model with $G = 5$. The classifications according to this EE model, which is equivalent to an EEE model in the MCLUST framework, can be seen in Table 5.2 and Figure 5.2.

Table 5.2: Classification table for the best CDGMM on the rats dataset.

	1	2	3	4	5
Diet 1	8				
Diet 2		3	1		
Diet 3				1	3

From Table 5.2 and Figure 5.2, it is apparent that the best CDGMM classifies all rats on Diet 1 into Group 1 and that one rat from each of the remaining two diets is misclassified into a group on its own. However, these two misclassified rats do have unusual weights

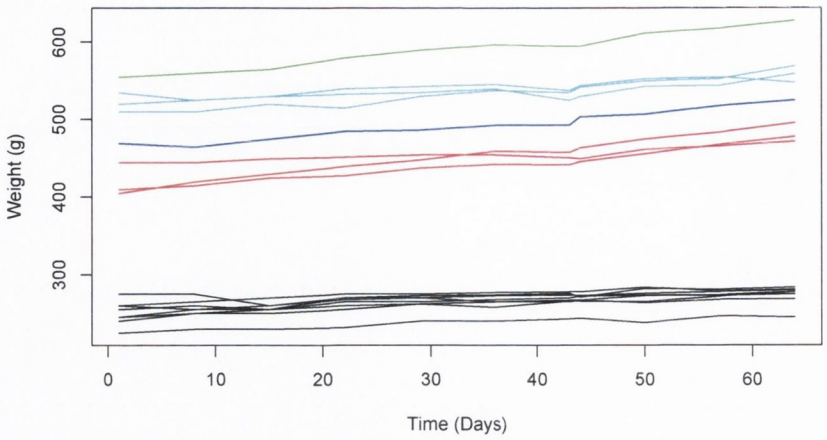


Figure 5.2: Time series for each rat, coloured by classifications — black for Group 1, red for Group 2, green for Group 3, purple for Group 4 and blue for Group 5.

relative to the other rats that were on the same diet. The Rand index associated with this model is 0.95 and the adjusted Rand index is 0.88.

5.5 Summary

A new model-based clustering paradigm has been introduced for the analysis of longitudinal data. This family of mixture models, called CDGMMs, utilise a modified Cholesky decomposition to model the covariance structure and so they are suitable for the analysis of longitudinal data. Four members of the CDGMMs are given herein and the associated maximum likelihood estimates for the parameters have been derived. Two of these models have already appeared in the literature, through the MCLUST framework, while the other two models are new. This family of models performed well when applied to real data on the weight of rats.

Note that this chapter is only intended as the first step in the creation of the CDGMM family of models. Only basic constraints are imposed herein and there is scope for the addition of many more members. For example, constraints could be imposed on elements on one or more of the sub-diagonals of $\mathbf{T}_g$, and some of the other constraints applied by Pourahmadi *et al.* (2007) could also be used.

The scope for expansion of the CDGMM family of models is great and it could grow to have

at least as many members as the GPGMM family of models. However, further expansion will be left for future work, with the emphasis of this thesis now shifting from interval to binary data.

Chapter 6

Association Rules

6.1 Introduction

This chapter marks the shift in this work from techniques applicable to interval data to techniques that can be applied to binary data. This shift, from model-based clustering to association rule mining, also marks the transition from statistical modelling into an area of data mining that is largely algorithm-driven.

Association rules are distinct from other methods of analysing binary data, such as loglinear models, in that they present an efficient method of analysing large binary data sets. Association rules are used to discover relationships between binary variables in transaction databases. A transaction database may be considered as one that consists solely of binary variables; an item can either be present in a transaction or not.

Although formally introduced by Agrawal *et al.* (1993), many of the ideas behind association rules can be seen in the literature at least as far back as Yule (1903). Applicable to data from a wide range of sources, such as convenience stores, credit card companies, electorial commissions, college application offices and healthcare providers; association rule analysis is one of the most versatile data mining techniques available to the analyst.

The apriori algorithm (Agrawal & Srikant, 1994; Borgelt & Kruse, 2002; Borgelt, 2003) presents an easy-to-implement method of generating, or mining, association rules.

6.2 Definition of an Association Rule

6.2.1 Association Rules

Given a non-empty set, I, an association rule is a statement of the form $A \Rightarrow B$, where $A, B \subset I$ such that $A \neq \emptyset, B \neq \emptyset$, and $A \cap B = \emptyset$. The set A is called the antecedent of the rule, the set B is called the consequent of the rule, and I is called the itemset. Association rules are generated, or mined, over a large set of transactions $\{\tau_1, \tau_2, \ldots, \tau_n\}$.

An association rule is deemed interesting if the items involved occur together often and there is evidence to suggest that one of the sets might in some sense lead to the presence of the other set. Consider a convenience store example; the association rule $\{\text{bread}\} \Rightarrow \{\text{butter}\}$ suggests that those who purchase bread, tend to also purchase butter.

The 'interestingness' of an association rule is commonly described by mathematical notions called support, confidence and lift. These functions are defined in Section 6.3.

6.2.2 Negations

In many applications, it is not only the presence of items in the antecedent and the consequent parts of an association rule that may be of interest. The absence of items from the antecedent part can be related to the presence or absence of items from the consequent part and *vice versa*. An example of a rule of this form is $\{\text{sunflower spead}\} \Rightarrow \{\text{not butter}\}$, and such a rule is sometimes referred to as a 'replacement rule'. Widening the mining paradigm to include this type of association rule gives rise to the possibility of mining many more association rules from each transaction dataset.

For convenience, we introduce the term 'negation'. Let $X \in I$ and write $\overline{X}$ to denote the absence or negation of the item, or items, in X from a transaction. Considering X as a binary $\{0, 1\}$ variable, the presence of the items in X is equivalent to $X = 1$, while $\overline{X}$ is equivalent to $X = 0$. Mining an association rule $A \Rightarrow B$ can actually lead to the discovery of related rules, which involve negations, such as $A \Rightarrow \overline{B}$; one way this can be achieved is through doubledecker plots, which are discussed in Section 6.6.2. An analysis involving negations along with a quantification of the number of rules that can be generated when negations are considered is given in Section 8.4.

6.3 Support, Confidence, Expected Confidence & Lift

The notation $P(A)$ is introduced to represent the proportion of times that the set A appears in transaction set Υ. Similarly, $P(A, B)$ represents the proportion of times that the sets A and B coincide in transactions in Υ. It is also necessary to define

$$P(B \mid A) = \frac{P(A, B)}{P(A)}.$$

That is, the proportion of times that the set B appears in all of the transactions involving the presence of the set A. Having established these definitions, Table 6.1 gives the functions by which association rules are typically characterised.

Table 6.1: Common functions of association rules.

Function	**Definition**
Support	$s(A \Rightarrow B) = P(A, B)$, $s(A) = P(A)$
Confidence	$c(A \Rightarrow B) = P(B \mid A)$
Expected Confidence	$EC(A \Rightarrow B) = P(B)$
Lift	$L(A \Rightarrow B) = c(A \Rightarrow B)/EC(A \Rightarrow B)$

Note that the expected confidence is the value that the confidence of a rule would take if the antecedent and consequent were statistically independent.

6.4 Rule Generation

In the original formulation of the association rule mining problem (Agrawal *et al.*, 1993), mining was performed by generating rules with support and confidence above predefined thresholds. The apriori algorithm (Agrawal & Srikant, 1994) was soon introduced; it also used a predefined support threshold but did so in a much more efficient manner. Recent improvements on the implementation of the apriori algorithm by Borgelt & Kruse (2002) and Borgelt (2003) have made it yet more efficient.

There have, however, been some viable alternatives to support based pruning. These alternatives mean that rules with relatively low support can be produced, which can lead to the discovery of rules involving negations. The idea of discovering association rules based on

conditional independencies was introduced by Castelo *et al.* (2001) through the MAMBO algorithm.

Pasquier *et al.* (1999) showed that it is not necessary to use support pruning in order to find all interesting association rules and that it is sufficient to mine the closed frequent itemsets. To define precisely a 'closed frequent itemset', it is necessary to consider Galois connection and Galois closure operators. These notions, together with other supporting definitions, are given by Pasquier *et al.* (1999). Further, Pei *et al.* (2000) and Zaki & Hsiao (2002) have proposed algorithms for this purpose, called CLOSET and CHARM respectively.

6.5 Lift & Odds Ratios

6.5.1 The Lift

The lift of an association rule is a symmetric function, since

$$L(A \Rightarrow B) = \frac{c(A \Rightarrow B)}{P(B)} = \frac{P(B \mid A)}{P(B)} = \frac{P(A,B)}{P(A)P(B)} = \frac{P(A \mid B)}{P(A)} = \frac{c(B \Rightarrow A)}{P(A)} = L(B \Rightarrow A).$$

This is extremely useful when it is not clear which of the rules $A \Rightarrow B$ and $B \Rightarrow A$ is of more interest, or when both may be of interest. Equivalently, lift can be useful when it is not clear whether $P(A \mid B)$ or $P(B \mid A)$ is of more interest, or when both may be of interest. Further, lift can also be used in any situation, even outside of the area of association rule mining, when it is not clear whether $P(A \mid B)$ or $P(B \mid A)$ is of interest, or when both may be of interest.

The distance of lift from one can be taken as a measure of the interestingness of an association rule, since if the lift of the association rule $A \Rightarrow B$ is one then

$$P(A,B) = P(A)P(B),$$

and so A and B are statistically independent. Further, the lift of an association rule gives insight into the interestingness of rules containing negations. Consider the rule $A \Rightarrow B$ where A and B are singleton sets; if the lift is less than one then only rules containing one negation will potentially be of interest, whilst if the lift is greater than one then only

rules involving two or no negations will be potentially of interest. Further details on this argument are given in relation to odds ratios in Section 6.5.2.

One problem with the interpretation of lift is that it is not symmetric about one. A possible solution to this is to consider log(lift), and another is to standardise lift as outlined in Chapter 8.

6.5.2 2×2 Tables & Odds Ratios

Table 6.2, which is a cross-tabulation of A versus B, can be regarded as a representation of the supports of all of the rules that can be formed with antecedent A (or $A = 1$) or $\overline{A}$ (or $A = 0$) and consequent B (or $B = 1$) or $\overline{B}$ (or $B = 0$). For example, the rule $A \Rightarrow B$ has support a and the rule $\overline{B} \Rightarrow A$ has support c.

Table 6.2: A cross-tabulation of A versus B.

		A	
		1	0
B	1	a	b
	0	c	d

The odds ratio associated with Table 6.2 is given by

$$\text{Odds Ratio} = \frac{a}{b} \div \frac{c}{d} = \frac{ad}{bc}.$$

This odds ratio can be related to association rules as follows. Note that $A = 1$ is considered to be the event here, without loss of generality. Consider the following three possibilities: odds ratio $= 1$, odds ratio < 1 and odds ratio > 1.

If odds ratio $= 1$ then $B = 0$ and $B = 1$ have exactly the same affect on the event. Therefore, no association rules involving A and B will be of interest.

If odds ratio < 1 then $B = 1$ reduces the chances of the event occurring and it is only the association rules containing exactly one negation that may be of interest, namely; $\overline{A} \Rightarrow B$, $A \Rightarrow \overline{B}$, $\overline{B} \Rightarrow A$ and $B \Rightarrow \overline{A}$.

If odds ratio > 1 then $B = 1$ increases the chances of the event occurring and it is only the association rules containing either no or two negations that may be of interest, namely $A \Rightarrow B$, $B \Rightarrow A$, $\overline{A} \Rightarrow \overline{B}$ or $\overline{B} \Rightarrow \overline{A}$.

Therefore, the odds ratio can be viewed as providing a partition of the eight possibly interesting association that can be formed from A, B and their respective negations. Moreover, lift can be viewed in the same way.

Furthermore, the relationship between association rules, contingency tables and odds ratios can be extended to larger tables where collapsibility could be used to determine which rules may be of interest. For example, if a $2 \times 2 \times 2$ table representing A, B and C can be collapsed (see Agresti, 2002, Section 9.1.2) to a 2×2 table representing A and B, then of the rules that can be formed from the sets A, B, C, $\overline{A}$, $\overline{B}$ and $\overline{C}$ we know that those involving C and $\overline{C}$ will not be of interest.

6.6 Association Rule Visualisation

Hofmann & Wilhelm (2001) discuss methods of visualising association rules, amongst which are two-key plots and doubledecker plots.

6.6.1 Two-Key Plots

Plotting confidence against support results in a scatter plot where each point lies on a straight line through the origin. Each such line represents all rules of the form $A \Rightarrow B_i$, so that each antecedent is represented by a line. Such a plot is often called a 'two-key plot' and an example is given in Figure 6.1. A two-key plot can be used to visualise pruning, for example all rules with support of at least 40% and confidence of at least 60% lie in the blue rectangle on Figure 6.2.

6.6.2 Doubledecker Plots

A doubledecker plot is a mosaic plot (Hartigan & Kleiner, 1981) showing various combinations of absence and presence of the items involved in an association rule. Figure 6.3 shows the doubledecker plot arising from the rule $A \Rightarrow B$.

Each of the rules $A \Rightarrow B$, $A \Rightarrow \overline{B}$, $\overline{A} \Rightarrow B$ and $\overline{A} \Rightarrow \overline{B}$ are represented on the plot in Figure 6.3. Specifically, the antecedents A and $\overline{A}$ are represented by a portion of the thin horizontal bar at the bottom of the plot, while the consequents B and $\overline{B}$ are each illustrated,

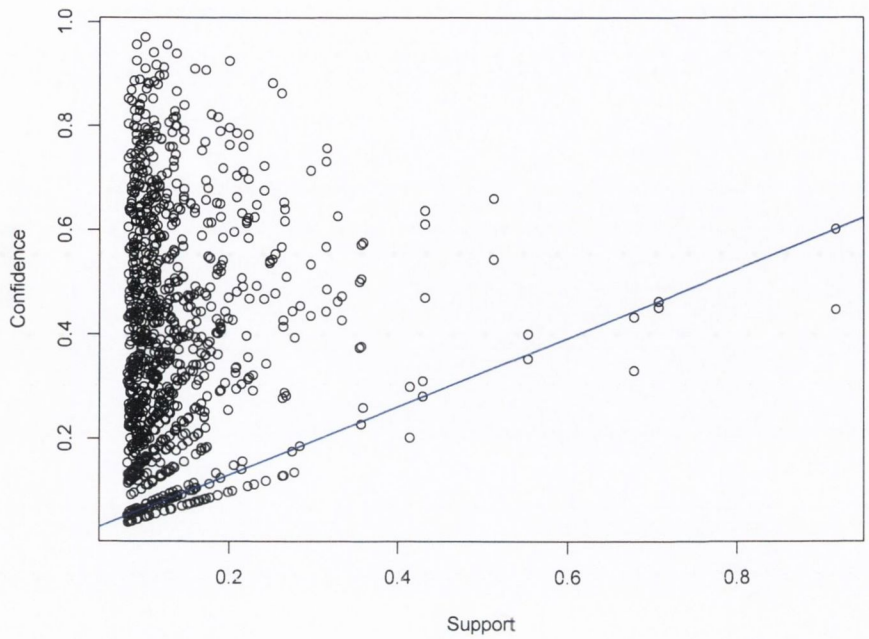


Figure 6.1: An example of a two-key plot.

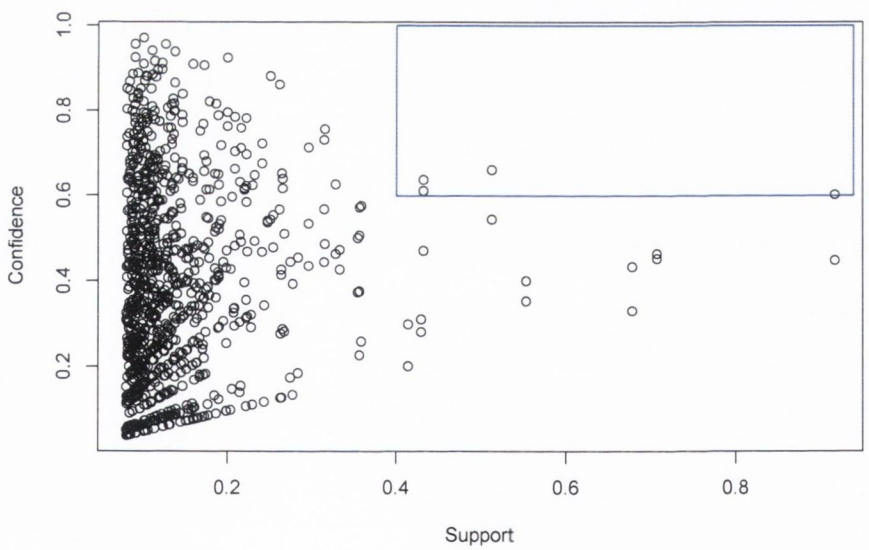


Figure 6.2: A two-key plot with a rectangle enclosing all rules with confidence of at least 0.6 and support of at least 0.4.

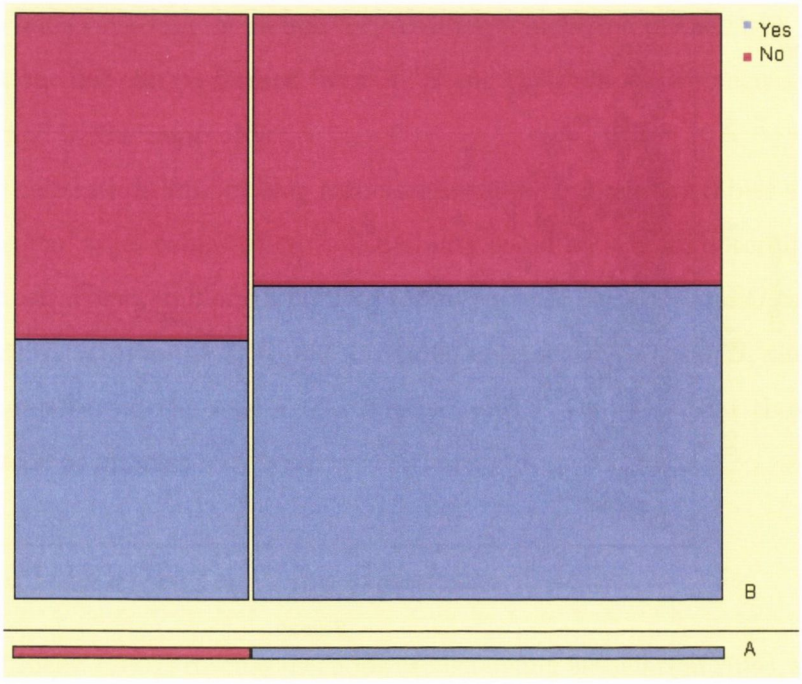


Figure 6.3: Doubledecker plot arising from the rule $A \Rightarrow B$.

for each antecedent, by a portion of a bin. These bins have an interpretation; the support of each rule is reflected in the area of the corresponding bin, while the height of the bin gives the confidence.

Doubledecker plots represent a method of extracting association rules involving negations implicitly. However, there is a limitation to their usefulness for this purpose. Consider Figure 6.4, which arises from the rule $\{A, B\} \Rightarrow \{C, D\}$. Figure 6.4 only illustrates rules with consequent $\{C, D\}$ or $\overline{\{C, D\}}$, the former is that of the original rule while the latter is the union of the three sets $\{C, \overline{D}\}$, $\{\overline{C}, D\}$ and $\{\overline{C}, \overline{D}\}$.

Therefore, while doubledecker plots can be used to visualise all possible antecedents when negations are considered, they are limited in that they can only display two possible consequents. That is, they are not sufficient to display all combinations of presence or absence of items from a given rule unless the consequent of that rule is singleton. The doubledecker plots shown in figures 6.3 and 6.4 were produced using the software package PISSARRO (Keller & Schlögl, 2002, `stats.math.uni-augsburg.de/PISSARRO`).

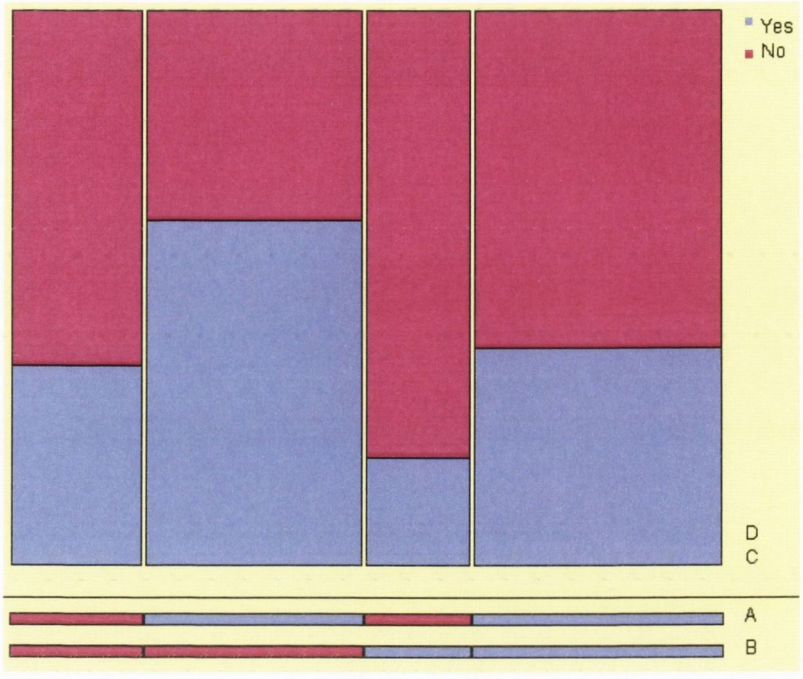


Figure 6.4: Doubledecker plot arising from the rule $\{A, B\} \Rightarrow \{C, D\}$.

6.7 Pruning

There have been few statistically sound, yet useful pruning methods mentioned in the association rules literature. One attempt at such a method was given by Bruzzese & Davino (2001), who presented three tests designed to avoid support and confidence pruning. Initially, association rules are mined with certain support and confidence thresholds, then three statistical tests are employed; an association rule is only retained if it 'passes' all three tests. These tests can be carried out using the software package PISSARRO.

6.7.1 Test 1

The first test is used to prune rules that have high confidence due to the presence of frequent items in the consequent part of the rule. Statistically, this test compares the confidence of the rule, $c(A \Rightarrow B)$, to the expected confidence of the rule, $EC(A \Rightarrow B)$, and is based upon the following hypotheses.

$$H_0 : c(A \Rightarrow B) = EC(A \Rightarrow B)$$
$$H_1 : c(A \Rightarrow B) > EC(A \Rightarrow B)$$

Under H_0 the antecedent and the consequent of the rule $A \Rightarrow B$ are statistically independent since $P(B \mid A) = P(B)$.

6.7.2 Test 2

The second test involves performing a chi-squared test on each set of rules with the same antecedent part. This test suffers the usual problems associated with chi-squared tests, as outlined by Wilhelm (2002).

The effect of Test 2 can be visualised on a two-key plot because if the test statistic is not significant then Test 2 will have the effect of deleting a line from the associated two-key plot. This relationship arises because, as mentioned is Section 6.6.1, each line on a two-key plot represents all rules with a common antecedent. Moreover, lines on the left hand side of a two-key plot face greater risk of deletion by Test 2 because these are the lines with higher slope and so represent rules with similar support.

6.7.3 Test 3

The third test is performed in order to evaluate the strength of the association between the antecedent and the consequent of a rule. The support of a rule is compared to the support of the antecedent part of the rule. Applying this test will cause the deletion of all rules for which the confidence is not close to one. This test, if applied, may cause the deletion of rules with confidence that might in fact be considered high, such as 90%.

6.7.4 Remark

While this, and other pruning methods can be effective in the analysis of association rules, they can slow down the mining process and often require that rules be initially generated using relatively low support or confidence thresholds. There is therefore no tangible advantage to pruning rules after generation over an effective mining method or a good system for ranking mined rules.

6.8 Interestingness

An alternative to pruning association rules is to rank them in order of interestingness, although this is not the sole intended use of measures of interestingness. In fact, the question of what makes an association rule interesting is at the heart of association rule mining. An important consideration regarding the interestingness of an association rule is the context in which it is presented; a rule may be mathematically interesting but yet may not be interesting in practice. Since what is interesting in practice will vary depending on the data and the purpose of the analysis, it is interestingness in the mathematical sense that is focused upon in this section.

Measures of interestingness have already been introduced in Section 6.3; for example, an association rule $A \Rightarrow B$ could be considered interesting if $c(A \Rightarrow B) > c_1$, where c_1 is a predefined threshold for confidence. However, basing interestingness on confidence alone is not satisfactory since there is no guarantee that the antecedent and consequent are not statistically independent.

Lift could also be used to rank mined association rules but the distance of lift from one cannot necessarily be taken as a reliable measure of interestingness. Lift is discussed further in Chapter 8.

Within the literature, alternative definitions of interestingness have been proposed. Two types of interestingness, that are described by Vazirgiannis *et al.* (2003), are given in the following sections.

6.8.1 Gray & Orlowska's Interestingness

Gray & Orlowska (1998) define the interestingness of the association rule $A \Rightarrow B$ as

$$\text{Int}(A \Rightarrow B; K, M) = \left[\left(\frac{P(A,B)}{P(A)P(B)} \right)^K - 1 \right] (P(A)P(B))^M,$$

where K and M are fixed constants used to weight the relative importance of the lift and the support components, $P(A)P(B)$, respectively. $\text{Int}(A \Rightarrow B; K, M)$ could potentially be viewed as an objective function to be maximised in the mining process.

As a consequence of the symmetry of lift, Gray and Orlowska's interestingness is also symmetric, for every value of K and M, since

$$(P(A).P(B))^M = (P(B).P(A))^M .$$

Therefore, Gray and Orlowska's interestingness could also be used in any situation, even outside of the area of association rule mining, where there is doubt over whether $P(A \mid B)$ or $P(B \mid A)$ is of interest, or when both may be of interest.

Another advantage of Gray and Orlowska's interestingness is ease of implementation; in fact, this type of interestingness is used is Chapter 7. Although, some justification for selection of the values of K and M used in Chapter 7 is given therein, the degree of arbitrariness in the choice of K and M is a weakness of this kind of interestingness.

6.8.2 Dong & Li's Interestingness

Dong & Li (1998) define interestingness in a more mathematical fashion than either Silberschatz & Tuzhilin (1995) or Gray & Orlowska (1998). Dong and Li's interestingness involves computing a distance metric between a rule and the other association rules in its neighborhood. An association rule is deemed interesting if the corresponding point is unexpected in the context of the neighboring rules.

Dong & Li (1998) actually define two types of interestingness; 'unexpected confidence' and 'isolated interestingness'. In both cases, the value taken by interestingness is either '0' or '1'. A value of '1' is assigned if the distance, as defined by some metric, is bigger than some predefined threshold; otherwise, a value of '0' is assigned. The binary nature of this type of interestingness means that it cannot be used to obtain any useful ranking of association rules.

Chapter 7

Association Rule Analysis of CAO Data

7.1 Introduction

Selected methods described in Chapter 6 are used and developed to facilitate the analysis of college application data. The purpose of this analysis is to discover whether or not there is a meaningful relationship between the points that a school-leaving student expects to attain and the courses, at an Irish university, that the student applies for. In particular, the question of primary interest is do students select university courses based on the points 'value' of the course, or based on subjects of interest? The relationship between course selection and gender is also explored.

7.2 Background

On Monday 21st August 2006 the Irish Times carried an article entitled *'Points race' may be over as CAO requirements tumble.* The author, Seán Flynn explained that the points race may be coming to an end; a claim supported in the article by John McGinnity, deputy registrar at NUI Maynooth (NUIM), who was quoted as stating that "this year has seen a rebalancing between the supply and demand for places."

The term 'points race' is used to describe the struggle to attain points; the metric by which

school leavers have come to judge their Leaving Certificates'. In past generations the vital statistic in assessing the strength of one's Leaving Certificate was the number of passes and honours attained. Today however, the assessment is usually given in cold hard points, so much so that the attainment of points has become, for many, an end as well as a means.

Points are awarded by the Central Applications Office (CAO) on the basis of grades attained at the Leaving Certificate examination. The intended purpose of these points is to allow colleges to decide which students should get into which courses. In this sense, it is understandable that students should want to get as many points as possible; the more points attained, the less likely a student is to be pipped at the finishing line and lose out on their preferred college course.

However, it would be a disturbing prospect if students were choosing their courses based on points value rather than on topic; if the status of taking a college course with high points outweighed the wisdom of taking one that the student might enjoy or even excel at. This would be a points race in a different sense: points for points-sake.

Considering that the points requirement for a course is dictated by the demand for entry to the course versus the amount of places available, it is not necessarily true that courses requiring higher points are better than those requiring lower points — high or low points reflect only supply versus demand for a course. Therefore, since the number of places for each course is limited, universities can control the points 'value' of a course simply by lowering the quota for the course; which must be set before the applications are made.

Socially, in a situation where this type of points race existed, our prospects of succeeding as a knowledge-based society in the future would have to viewed as questionable.

7.3 Literature

Tuohy (1998) completed the first research paper for the Commission on the Points System, which was established in 1997 by then Minister for Education and Science, Micheál Martin. Amongst his conclusions, the author found that gender may have been an issue in course choice. Moreover, he maintained that the answer to the question over whether or not the points system directly effected students choices was unclear.

The final report of the commission, including recommendations (Hyland, 1999) found that the Leaving Certificate was a valid method of predicting higher education performance, pointing to Lynch *et al.* (1999) for substantiation. In fact the report largely vindicated the Leaving Certificate together with the CAO system as a college entrance system. The report highlighted application for medicine courses as an exceptional case and recommended further investigation by the relevant state and healthcare bodies.

Clancy (2001) and O'Connell *et al.* (2006) completed an extensive study of access to higher education. They found that socioeconomic background was having a notable effect on the profile of entrants to various third-level courses and that more females than males were entering third-level education, with the stereotypical gender biases still present in areas like engineering and primary teaching.

Gormley & Murphy (2006) analysed the CAO data from 2000 by assuming the existence of an underlying mixture model with a Plackett-Luce model (Plackett, 1975) used for each mixture component. This analysis yielded a model with 22 mixture components, which is essentially claiming that there are 22 different sub-populations of students filling in CAO forms; 21 subject-based groups plus a noise group. These results supported the CAO system since no group emerged that chose courses that were linked only by high points value and not by topic.

They went further and looked at the subset of students that their model assigned to the health sciences component, listing the 30 courses with the greatest probability of appearing on their CAO forms. The medicine courses across the five institutions — University College Dublin (UCD), Trinity College Dublin (TCD), National University of Ireland Galway (NUIG), University College Cork (UCC) and the Royal College of Surgeons in Ireland (RCSI) — topped this list, which also contained two law courses and an engineering course. The authors ascertained that this could be regarded as adding "some weight" to the argument that a points race does exist.

Lynch *et al.* (2006) looked at the CAO system as an admission system for dentistry courses. They concluded that while there was a statistically significant relationship between Leaving Certificate performance and first year dental examination results, there was "no association between the number of points achieved by students in their Leaving Certificate examination

and their performance in the final dental examination".

Interestingly, Moran & Crowley (1979), even though they analysed data based on a slightly different application system over 25 years ago, also found that there was a relationship between Leaving Certificate performance and first year college exam performance. This relationship continuing in a less definite fashion through the remainder of the university years.

Notably, Moran & Crowley (1979) concluded at the time that science-based subjects, in particular Mathematics, had better predictive value than non-science-based subjects. They also vindicated, to an extent, the Leaving Certificate as a college entrance exam, concluding that "the prospects for other selection systems which can equal or improve on the Leaving Certificate seem poor".

7.4 Methodology & Data

7.4.1 Methodology

CAO applications data from the year 2000 were analysed using association rule mining. Whereas similar data have been analysed before using fairly complicated statistical models — Moran & Crowley (1979), Tuohy (1998) and Gormley & Murphy (2006) — the analysis herein is conducted at a very intuitive level. In particular, no underlying statistical model is assumed and no hypotheses are proposed. Therefore, the analysis essentially represents a convenient way of looking at the data.

7.4.2 Data

The data represent CAO degree applications from the year 2000; the same data that was analysed by Gormley & Murphy (2006). In the year 2000, 53,757 applicants — 24,419 male, 29,337 female and one with unspecified, misspecified or incorrectly recorded gender — chose up to ten of 533 degree courses; note that the two-subject moderatorship offered at TCD was counted as separate courses since, unlike the arts degrees offered at the NUI colleges, each applicant explicitly selected their course options at the application stage.

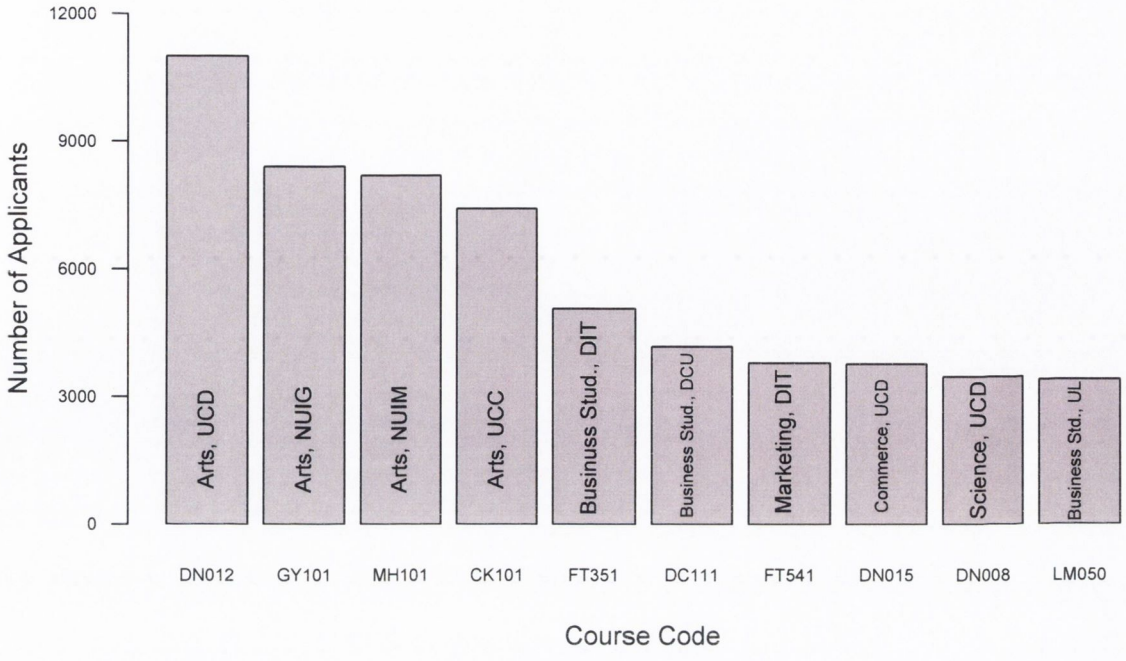


Figure 7.1: The ten most popular course choices in the year 2000.

The most popular course was Arts at UCD, which was chosen by 11,007 (20.5%) applicants. The ten most popular courses are shown in Figure 7.1; these are all either arts or business courses, with the exception of Science at UCD. The top four courses in Figure 7.1 are all common entry arts courses offered at NUI colleges in Dublin, Galway, Maynooth and Cork.

Note that the abbreviation DIT is used herein to denote 'Dublin Institute of Technology' and IT is used in general to denote 'Institute of Technology'.

Interestingly, the profile of the ten most popular first choices (Figure 7.2) is different; arts at UCD and NUIG switch places as the first and second most popular courses respectively. Furthermore, in addition to arts and business courses, primary teaching courses also feature this time along with the computer applications course in DCU.

Figure 7.3 represents the distribution of the number of courses chosen by applicants on the degree section of their respective CAO forms in the year 2000. The form was completed in full by 16,138 (30%) students and so 70% of applicants did not select the maximum number of degree courses. One was the next most popular number of courses selected, with

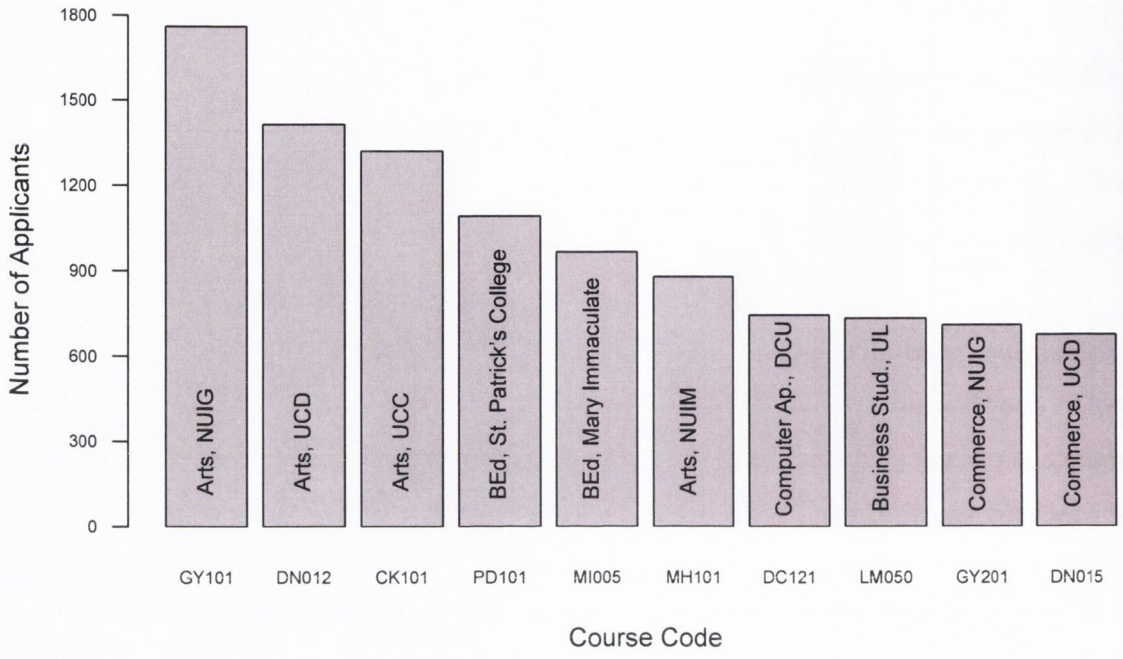


Figure 7.2: The ten most popular first preference course choices in the year 2000.

4,952 (9%) students opting to gamble on getting their first and sole choice. The number of applicants choosing 2–9 courses was roughly the same, taking values from 3,618 to 4,518.

7.5 Analysis of Rules Interrelating Courses

7.5.1 Rule Generation

Association rules were generated using the apriori algorithm of the arules package (Hahsler *et al.*, 2005) in R. The minimum support for a rule was set at 0.5%, the minimum confidence at 80% and the maximum length of a rule was set at ten. Practically, support of 0.5% means that at least 269 students must have selected a particular combination of courses for a rule comprised of that combination to feature within the mined rules. In analysing these rules, two subsets of rules were of particular interest; those interrelating courses and those relating courses and gender.

Figure 7.3: The number of courses selected by applicants in the year 2000.

7.5.2 Pruning Method

Discovery of interesting rules that interrelate courses was facilitated by using only the course choices in the input data, thereby omitting the gender variable; 145 association rules were generated using R. In order to view courses at the highest level of grouping, the following pruning technique was devised:

1. Consider the items in an association rule.

2. If a larger rule contains those items then delete the smaller association rule.

This method was applied and more than halved the number of association rules, leaving 72 rules, some of which comprised the same courses in different order.
For example, a rule comprised of four medicine courses, such as
{Medicine at NUIG, Medicine at TCD, Medicine at RCSI} $\Rightarrow$ {Medicine at UCC},
was pruned in favor of a rule comprised of these four, plus one additional medicine course, such as

{Medicine at NUIG, Medicine at UCC, Medicine at TCD, Medicine at RCSI}
$\Rightarrow$ {Medicine at UCD}.

Note that when this method is faced with two or more rules of the same length, containing the same items (courses), all rules are retained. The option of a third step where ties of this sort can be broken, using confidence say, was not availed of in this case. It can be seen in Section 7.5.3 that the order of such rules, when ranked by confidence, can provide information about their constituent courses.

The pruning method described in this section is somewhat unusual and perhaps even counterintuitive. Typically, the rule $\{A, B\} \Rightarrow C$ would be pruned in favor of the rule $A \Rightarrow C$, provided that the presence of B had no perceived 'benefit'. However, in the specific application of this section, it is desirable to retain the larger rule because if no rule relating courses like medicine and law existed amongst the pruned rules then none could have existed amongst the original rules.

7.5.3 Top Twenty Rules

Table 7.1 gives the top twenty rules, by confidence, relating distinct items. Following inspection of these rules, they can each be grouped into one of a few categories so that a non-model-specific 'clustering' of the courses emerges. An explanation of all course codes appearing in this work is given in Table E.1 (Appendix E).

Medicine

Rules 1, 10 and 15 in Table 7.1 relate the medicine courses across the five institutions offering medicine: NUIG, UCC, TCD, RCSI and UCD. In this case, the fact that the UCD course is in the consequent in Rule 1 suggests that it was the most popular of the five medicine courses; in rules 10 and 15 and elsewhere amongst the 72 remaining rules, rules relating the medicine courses appeared with different courses in the consequent (NUIG was the second most popular medicine course and TCD was the third).

Rule 14 associates three courses within Dublin; Science in UCD, Medicine in TCD and Medicine in UCD. This is a possible example of a geographical effect on course selection.

Table 7.1: The top twenty rules, ranked by confidence, interrelating courses.

	Rule	Support	Confidence	Lift
1	{GY501, CK701, TR051, RC001} ⇒ {DN002}	0.53%	97.92%	33.76
2	{MI005, DN012, CM001, FR001} ⇒ {PD101}	0.55%	97.67%	21.22
3	{PD101, FR001, LM047} ⇒ {MI005}	0.55%	96.71%	21.17
4	{CM001, FR001, PD103} ⇒ {PD101}	0.60%	96.41%	20.95
5	{MI005, PD103} ⇒ {PD101}	0.51%	95.82%	20.82
6	{CM001, LM047} ⇒ {MI005}	0.56%	95.54%	20.91
7	{GY101, MI005, CM001, FR001} ⇒ {PD101}	0.58%	95.44%	20.74
8	{DN012, CM001, MH101, FR001} ⇒ {PD101}	0.64%	95.04%	20.65
9	{MI005, CM001, MH101, FR001} ⇒ {PD101}	0.58%	94.24%	20.48
10	{CK701, TR051, DN002, RC001} ⇒ {GY501}	0.53%	92.76%	38.84
11	{CK101, CM001} ⇒ {MI005}	0.54%	92.33%	20.21
12	{TR084, FT472} ⇒ {FT471}	0.59%	92.11%	17.38
13	{GY101, DN012, FR001} ⇒ {PD101}	0.52%	92.11%	20.01
14	{DN008, TR051} ⇒ {DN002}	0.54%	91.25%	31.47
15	{GY501, CK701, DN002, RC001} ⇒ {TR051}	0.53%	89.52%	41.31
16	{DC111, FT542, DN015} ⇒ {FT351}	0.66%	89.44%	9.47
17	{GY101, MH101, FR001} ⇒ {PD101}	0.57%	89.24%	19.39
18	{CK101, FR001} ⇒ {MI005}	0.56%	89.05%	19.49
19	{MI005, DN012, PD101, CM001} ⇒ {FR001}	0.55%	88.82%	29.33
20	{TR004, GY251} ⇒ {DN009}	0.50%	88.67%	27.93

Teaching & Arts

Rules 2–9, 11, 13 and 17–19 relate arts (BA) and education (BEd) courses across nine institutions throughout Ireland. The course PD101 (BEd), at St. Patrick's College, is the consequent part of the majority of these courses, suggesting that it was a significant draw to such-minded people and the most popular of the BEd courses.

Social Care

Rule 12 suggests that people who applied for Social Studies (Social Work) at TCD and Early Childhood Care and Education at DIT also selected Social Care at DIT. In fact less than 8% of applicants choosing the former pair failed to choose the latter course. This may be indicative of a geographical effect — applicants wishing to study social care in Dublin.

Business

Rule 16 relates four business courses in Dublin: Business Studies at DIT and DCU, Management and Marketing at DIT and Commerce at UCD. There are further rules within the 72 that related business courses in Dublin.

Law

Rule 20 associates the Law degrees at TCD, NUIG and UCD; the UCD degree being the most popular choice. This rule can be used to illustrate the lift of an association rule; given that it is known that an applicant selected Law at both TCD and NUIG, they are 27.93 times more likely to have selected Law at UCD than if no information was available regarding their other course choices. Moreover, since lift is symmetric it is also true that if it is known that an applicant selected Law at UCD, they are 27.93 times more likely to have selected Law at TCD and NUIG than if no information regarding their other course choices was available.

7.5.4 Remaining Rules

All 72 rules are given in Table E.2 (Appendix E). When the antecedent and consequent of these rules are considered, almost all of them can be considered as belonging to one of the following categories, or 'clusters'.

- Arts
- Business[1]
- Business & Communications
- Engineering
- Law
- Medicine
- Primary Teaching & Arts
- Psychology & Arts
- Science
- Social Care[2]
- Theology & Arts

All of the remaining rules fall under some combination of these categories — typically, but not exclusively, a combination involving Arts. Most importantly, there were no combinations, like medicine and law, that associated two very different disciplines that are regarded as prestigious in terms of CAO points. Therefore, it can be said that all of the 145 mined

[1] Includes Business Studies, Commerce, Finance and Marketing.
[2] Includes Social Care, Social Science, Early Childhood Care and Early Childhood Studies.

rules were grouped by topic of interest and not by perceived prestige associated with high points value.

Looking within the categories into which these rules fall, there is evidence of further grouping by geographical area. For example, Rule 12 in Table 7.1 relates social care courses within Dublin whereas Rule 38 in Table E.2 relates the same type of courses without this geographical constraint.

Moreover, Rules 29 and 49 (Table E.2) relate science courses in the Dublin area, while Rule 51 relates business courses within Dublin. The geographical effect is not, however, restricted to the Dublin area. Rule 57 relates three courses in Cork: Commerce, Government and Public Policy, and Arts, all at UCC. This may in fact be an example of a rule where topic was second to geographical location as the main motivator for course selection.

7.5.5 Remarks

By inspection of the top 20 rules, five 'clusters' emerge. These 'clusters' are consistent with the findings of Tuohy (1998) and Gormley & Murphy (2006), yet come about from the output of a very simple analysis that assumed no underlying model. Furthermore, inspection of the whole 72 rules led to the discovery of six additional 'clusters'.

The presence of these 'clusters', especially arrived at in such a natural fashion, is evidence of the effectiveness of the points system; this analysis yielded no evidence to suggest that students choose courses with the highest points, instead the data suggest that they choose courses by topic, with geography sometimes a factor. Moreover, due to the pruning method that was employed, it can be inferred that amongst all 145 mined rules, there was no evidence to suggest that course choices are made based on prestige associated with high CAO points.

7.6 Further Exploratory Analysis

7.6.1 The Idea

Further analysis was carried out to search for any weak evidence that might suggest that a points race exists amongst a smaller cohort of students. The support threshold was

reduced and pruning was effected. Then subsets of prestige (high-point) courses within the remaining rules were examined.

7.6.2 Methodology

The rules were mined once again, this time the support threshold was set at just 0.1%, meaning that a rule would be mined so long as it was supported by at least 54 applicants. The confidence threshold remained at 80% and the pruning method employed in Section 7.5.2 was applied to the 2,537 rules that were generated; this reduced the number of rules to 1,020.

7.6.3 Results

Whilst three rules were found that contained law and medicine and one that contained psychology and law, these rules were each supported by less than 70 applicants.

7.7 Analysis of Gender Related Choices — Female

7.7.1 Initial Analysis

The gender variable was included in the mining process, the support threshold was reset to 0.5% and 536 association rules were generated using the arules package in R. The resulting rules were pruned, as in Section 7.5.2. The remaining rules were syntactically constrained to have 'female' (denoted F herein) in the consequent and the top ten such rules are given in Table 7.2.

From Table 7.2 it is apparent that the Early Childhood Care and Education course at DIT is in the antecedent of eight of the ten rules. The aim of this section was to find relationships between course choice and gender (female), and the rules in Table 7.2 largely failed in this aim. Therefore an alternative approach was necessary.

Note that rules were chosen with $\{F\}$ in the consequent rather than the antecedent because the proportion of applicants to a particular course that were female was viewed as more interesting than the proportion of females that applied to a particular course. In any event, since lift is symmetric, it applies identically to the converse of each rule.

Table 7.2: The top ten rules, ranked by confidence, with consequent {F}.

Antecedent	Supp.	Conf.	Lift
{BEd at Froebel, Early Childhood Care & Ed. at DIT}	0.54%	99.32%	1.82
{Arts at NUIG, Early Childhood Care & Ed. at DIT}	0.86%	99.14%	1.82
{BEd (Home Economics) at St. Catherine's}	1.26%	99.12%	1.82
{E.C.C. & Ed. at DIT, Tourism Marketing at DIT}	0.59%	99.07%	1.82
{Arts at UCC, E.C.C. & Education at DIT}	0.54%	98.97%	1.81
{E.C.C. & Ed. at DIT, Soc. Care at DIT, E.C.Std. at UCC}	0.84%	98.90%	1.81
{BEd at St. Patrick's, E.C.C. & Ed. at DIT}	0.66%	98.89%	1.81
{BEd at Mary Immaculate, Early Childhood Std. at UCC}	0.60%	98.80%	1.81
{E.C.C. & Ed. at DIT, Hosp. (H. & C.) Man. at DIT}	0.58%	98.42%	1.80
{Social Science at UCC, E.C.C. & Ed. at DIT}	0.51%	98.21%	1.80

7.7.2 An Alternative Approach — Gray & Orlowska's Interestingness

Due to the failure of the approach described in Section 7.7.1, a different tact was taken. Gray and Orlowska's interestingness was computed for all 536 mined rules, with $K = M = 2$. The top ten rules with consequent $\{F\}$, ranked in order of interestingness, are given in Table 7.3. The magnitude of the interestingness column (Interest.) of Table 7.3 is not of particular importance here; it is used for ranking purposes only.

Table 7.3: The top ten rules, ranked by interestingness, with consequent {F}.

	Antecedent	Supp.	Conf.	Lift	Interest.
1	{Social Care at DIT}	4.99%	94.10%	1.72	0.00165
2	{Early Childhood Care & Ed. at DIT}	4.49%	98.13%	1.80	0.00140
3	{BEd at St. Patrick's}	4.01%	87.23%	1.60	0.00098
4	{BEd at Mary Immaculate}	3.89%	85.18%	1.56	0.00089
5	{Social Science at UCD}	3.47%	85.79%	1.57	0.00072
6	{Early Childhood Studies at UCC}	3.19%	96.30%	1.78	0.00069
7	{Social Science at UCC}	3.02%	87.88%	1.61	0.00056
8	{BEd at Froebel}	2.77%	91.34%	1.68	0.00049
9	{E.C. Care & Ed. at DIT, Soc. Care at DIT}	2.49%	98.10%	1.80	0.00042
10	{BEd at Coláiste Mhuire}	2.51%	88.27%	1.62	0.00039

Choosing $K = M = 2$ can be viewed as giving a balanced compromise between the distance of lift from one and the respective magnitudes of $P(A)$ and $P(B)$. However, a more concrete argument can be made for setting $K = M$. Gray and Orlowska's interestingness can be

written as

$$\text{Int}(A \Rightarrow B; K, M) = \left[c(A \Rightarrow B)^K - P(B)^K\right] P(A)^M P(B)^{M-K}. \tag{7.1}$$

To see why this is so, notice that

$$\begin{aligned}
\text{Int}(A \Rightarrow B; K, M) &= \left[\left(\frac{P(A,B)}{P(A)P(B)}\right)^K - 1\right] (P(A).P(B))^M \\
&= \left[\frac{\left(\frac{P(A,B)}{P(A)}\right)^K - P(B)^K}{P(B)^K}\right] (P(A).P(B))^M \\
&= \left[P(B \mid A)^K - P(B)^K\right] P(A)^M P(B)^{M-K} \\
&= \left[c(A \Rightarrow B)^K - P(B)^K\right] P(A)^M P(B)^{M-K}.
\end{aligned}$$

From Equation 7.1, an argument for choosing $K = M$ becomes apparent; since all of the rules are to be syntactically constrained to have a particular consequent (F or later, M), $P(B) = P(F)$ will be equal across all rules and choosing $K = M$ gives $P(B)^{M-K} = 1$. Setting $K = M = 2$, which is somewhat arbitrary, achieves a suitable compromise between measuring the distance of $c(A \Rightarrow B)$ from $P(B)$ and the magnitude of $P(A)$ giving,

$$\text{Int}(A \Rightarrow B; 2, 2) = \left[c(A \Rightarrow B)^2 - P(B)^2\right] P(A)^2.$$

7.7.3 Top Ten Rules

By inspection, the rules in Table 7.3 fall neatly into two distinct categories. This gives an insight into the areas of undergraduate study that are most strongly linked to female applicants.

Primary Teaching

Rules 3, 4, 8 and 10 associated various types of primary teaching degrees (BEd) to female applicants. These rules all have confidence in the range 85.18% – 91.34%, which is consistent with the proportion of primary school teachers in Ireland that are female. Drew (2006) reports UNESCO educational statistics that reveal that 86% of primary school teachers in Ireland in 2002 were female. Table 7.4 gives the corresponding figures for the years 1999, 2001 and 2004 as well as the year 2002.

Table 7.4: Gender breakdown of primary teaching staff in Ireland, 1999–2004.

	1999	2000	2001	2002	2003	2004
Total	21,148	20,865	21,865	22,979	23,972	24,792
Female	17,924	–	18,680	19,772	–	20,671
Male	3,224	–	3,185	3,207	–	4,121
% Female	84.8	–	85.4	86.0	–	83.4
% Male	15.2	–	14.6	14.0	–	16.6

The raw data for Table 7.4 was sourced from the UNESCO Institute for Statistics website `http://www.uis.unesco.org`, UIS Database, updated following the *Final release of data from the 2005 education survey.* The gender-specific data for the years 2000 and 2003 were listed as unavailable.

Over the four years for which data on the proportion of female primary teachers is available, this number is between 83.4% and 86.0%. Drew (2006) suggests a variety of reasons that may explain why men are not attracted to primary teaching courses. Moreover, this issue is not confined to applicants within the Republic of Ireland. The situation is similar north of the border. The Stranmillis Annual Report 2003/04 stated that "male entrants to the Primary B.Ed. represent 17% of the Primary intake [in 2003], compared with 9% in 2002." However, their 2004/05 annual report revealed that this number had fallen to 10% in 2004. Earlier this year it appeared that the Government recognised this imbalance as an issue and was planning to address it. In an article entitled *Competition Intense for Places on Teaching Degree Courses* in the Irish Times on 25th January 2006, Brian Mooney wrote that "Some 90 per cent of primary school teachers under 40 are women and today, the Department of Education is launching a promotion campaign to attract men into primary teaching."

Social Care & Childhood Care

Rules 1, 5 and 7 link courses in Social Care and Social Science to female applicants. Rules 2 and 6 link courses in Early Childhood Care / Studies to female applicants; for example, over 96% of applicants to Early Childhood Studies at UCC were female. Rule 9 shows that over 98% of applicants who applied for both Social Care and Early Childhood Care and

Education at DIT were female.

This area shows a gender imbalance on par with, or worse than, that shown in primary teaching. The reasons for such an imbalance are likely to be similar; such as the perception of such jobs as 'women's jobs'. However, there is far less literature around this topic; suggesting that it may be perceived as being of less importance to society.

7.7.4 Further Rules

Extending to the twenty most interesting rules, as listed in Table E.3 (Appendix E), all courses that emerge are in the areas of primary teaching, social studies and arts. Therefore, the stereotype that primary teachers and social workers are predominantly female is, the data suggests, apparent from the application stage.

7.8 Miscellaneous

7.8.1 Analysis of Gender Related Choices — Male

The 536 rules generated in Section 7.7.1 were ranked according to their interestingness, with the syntactic constraint that the consequent is 'male', denoted {M}. The top twenty such rules are given in Table E.4; these rules all associated courses in engineering, construction and manufacturing. The data suggests, as in the female case, that the usual stereotypes hold in the male case and are entrenched in the application process.

7.8.2 Results of Analyses for Alternative Support & Confidence Thresholds

One of the criticisms sometimes levelled at association rule analysis is the sensitivity of the results to perturbation of the support or confidence thresholds. Therefore, in the interest of completeness, analyses were carried out with support as low as 0.1% and confidence as low as 50%. Note that while a support threshold of 0.1% may be considered 'too' low, it is certainly the case that a rule with confidence 50% is not useful. If a rule $A \Rightarrow B$ has confidence 50% then B is as likely to occur as not, given that A occurs.

The results of this analysis supported the earlier findings in this chapter, with rules linking courses by high points alone confined to a very small number of students.

7.9 Conclusions

Through the application of a simple analysis to CAO application data from the year 2000 — which consisted only of a useful way to look at the data without imposing any statistical model, testing any hypotheses or computing any confidence intervals — some important questions regarding the CAO application system have been answered, while others may have been raised.

Definite patterns of course choice amongst applicants emerge, based primarily on course topic, with geography also a factor for some applicants. In no instance, save for very small cohorts of students, did points alone dictate course choice. Moreover no evidence to link courses purely based on points was found. This is a very important result, supporting the CAO application system as an effective third level application channel.

There is strong gender linkage amongst the data. The vast majority of applicants to courses in social care, early childhood care and primary teaching are female; whereas the majority of applicants to courses in engineering, construction and manufacturing management are male.

Whereas the gender imbalances in primary teaching and engineering are active issues, comparatively little is being done to address the imbalances that exist in the social care and construction sectors. Whether or not these imbalances are detrimental to our society is debatable, however, it is clear from this analysis that they can all be traced back to the application stage.

7.9.1 A Word of Warning

The conclusions of this chapter should not be taken as a vindication of the CAO college application system. Whereas it is true that there is no evidence herein to suggest that a 'points race', in the negative sense, is prevalent, it is not necessarily true that this application system puts the best applicants into each college course.

Tuohy (1998) gives the analogy of a bank using a round of golf to decide which applicant should get the management post; it is a fair, consistent system but the best golfer will not necessarily be the best branch manager.

Chapter 8

Standardising the Lift of an Association Rule

8.1 Introduction

Various measures of the 'interestingness' of an association rule were introduced in Chapter 6. One of these, Gray and Orlowska's interestingness, was used in the analysis of the CAO data in Chapter 7. While Gray and Orlowska's interestingness was useful in this application, the somewhat arbitrary nature of the choice of K and M is a drawback of this interestingness.

This chapter aims to use one of the most commonly used stand-alone measure of interestingness, the lift, to devise a new measure of interestingness. This new kind of interestingness, standardised lift, relies only on the most commonly used functions of an association rule; support, confidence and lift. This new interestingness can be visualised elegantly and it has no parameters, so there are no ambiguities like those around Gray & Orlowska's interestingness.

8.2 Range of Values of Lift

8.2.1 Range in Terms of $P(A)$ & $P(B)$

The range of values that the lift of an association rule $A \Rightarrow B$ can take is restricted by the respective values of $P(A)$ and $P(B)$;

$$\frac{\max\{P(A)+P(B)-1, 1/n\}}{P(A)P(B)} \leq L(A \Rightarrow B) \leq \frac{1}{\max\{P(A), P(B)\}}, \tag{8.1}$$

where n is the the number of transactions τ_i. Equation 8.1 is derived in Appendix F.1 and gives bounds that are almost identical to those derived by Fréchet (1951).

8.2.2 Range in Terms of the Minimum Support Threshold or the Number of Transactions

Equation 8.2, which is derived in Appendix F.1, is an expression for the range of values of lift in terms of minimum support threshold s.

$$\frac{4s}{(1+s)^2} \leq L(A \Rightarrow B) \leq \frac{1}{s}. \tag{8.2}$$

Now, substituting $s = 1/n$ in Equation 8.2 gives the bounds in terms of n,

$$\frac{4n}{(n+1)^2} \leq L(A \Rightarrow B) \leq n. \tag{8.3}$$

Equation 8.3 will give realistic bounds for lift in situations where minimum support and confidence thresholds are not used in the mining process.

Standardisation will not be useful in general if achieved through equations 8.2 or 8.3 because the limits given in these equations do not vary from rule to rule. In other words, a ranking of rules by lift would be invariant under any standardisation involving these limits alone.

Equation 8.3 does, however, give an insight into why it is necessary to standardise lift: the maximum value that lift can take is n and this occurs when a rule is supported by only one transaction. Rules supported by only one transaction will not usually be of interest and may even be pruned in the rule generation stage of the mining process.

8.2.3 Range in Terms of $P(A)$, $P(B)$, Support & Confidence Thresholds

Equation 8.1 can be expanded to account for both the minimum values of support (s) and confidence (c) that may be set at the mining stage;

$$\max\left\{\frac{P(A)+P(B)-1}{P(A)P(B)}, \frac{4s}{(1+s)^2}, \frac{s}{P(A)P(B)}, \frac{c}{P(B)}\right\} \leq L(A \Rightarrow B) \leq \frac{1}{\max\{P(A), P(B)\}}. \tag{8.4}$$

Equation 8.4 is derived in Appendix F.1 and follows from equations 8.1 and 8.2 with little extra work. This equation provides the most general, and accurate, upper and lower bounds for lift given herein.

8.2.4 Standardisation

Equation 8.1 can be used to standardise the lift of an association rule according to the formula

$$\mathcal{L}(A \Rightarrow B) = \frac{L(A \Rightarrow B) - \lambda}{\upsilon - \lambda}, \tag{8.5}$$

where

$$\lambda = \frac{\max\{P(A)+P(B)-1, 1/n\}}{(P(A)P(B))},$$

and

$$\upsilon = \frac{1}{\max\{P(A), P(B)\}},$$

are the limits given in Equation 8.1.

The standardised lift $\mathcal{L}$ must take a value inside the closed interval $[0, 1]$. A value close to 1 is indicative of an interesting rule. By refining λ to account for the lower bound given in Equation 8.4, the expression for $\mathcal{L}$ in Equation 8.5 can be altered to account for minimum support and confidence thresholds. We shall denote this type of standardised lift by $\mathcal{L}^*$, which is given by

$$\mathcal{L}^*(A \Rightarrow B) = \frac{L(A \Rightarrow B) - \lambda^*}{\upsilon - \lambda^*}, \tag{8.6}$$

where

$$\lambda^* = \max\left\{\frac{P(A)+P(B)-1}{P(A)P(B)}, \frac{4s}{(1+s)^2}, \frac{s}{P(A)P(B)}, \frac{c}{P(B)}\right\}$$

and υ is defined as before.

8.3 Example I: College Application Data

8.3.1 Background & Data

Chapter 7 presented an analysis of CAO data from the year 2000. These data are based on 53,757 applicants who each chose up to ten of 533 degree courses. Association rules were initially generated as outlined in sections 7.5.1 and 7.5.2.

8.3.2 Initial Analysis

Rules

Following the application of the pruning procedure mentioned in Section 7.5.2, 72 of the 145 rules that were initially mined remained. Four of these rules contained only the medicine courses offered across five institutions and they are given, ranked by confidence, in Table 8.1.

Table 8.1: The four mined rules that contained only medicine courses.

Id.	Rule	Support	Confidence	Lift
1	{GY501, CK701, TR051, RC001} ⇒ {DN002}	0.53%	97.92%	33.76
2	{CK701, TR051, DN002, RC001} ⇒ {GY501}	0.53%	92.76%	38.84
3	{GY501, CK701, DN002, RC001} ⇒ {TR051}	0.53%	89.52%	41.31
4	{TR051, GY501, DN002, RC001} ⇒ {CK701}	0.53%	85.20%	46.54

Note that all of the rules in Table 8.1 have equal support since each of the four rules comprise the same items.

Standardised Lift

The bounds given in equations 8.1 and 8.4 were calculated for each of the rules in Table 8.1 and are given in Table 8.2.

Table 8.2: The four rules from Table 8.1, with their bounds for lift.

Id.	Rule	Lift	λ	λ^*	υ
1	{GY501, CK701, TR051, RC001} ⇒ {DN002}	33.76	0.12	31.85	34.48
2	{CK701, TR051, DN002, RC001} ⇒ {GY501}	38.84	0.14	36.64	41.87
3	{GY501, CK701, DN002, RC001} ⇒ {TR051}	41.31	0.14	38.97	46.15
4	{TR051, GY501, DN002, RC001} ⇒ {CK701}	46.54	0.16	43.91	54.62

The lift of each rule in Table 8.2 and its relationship with the upper and lower limits can be visualised as shown in Figure 8.1 and Figure 8.2.

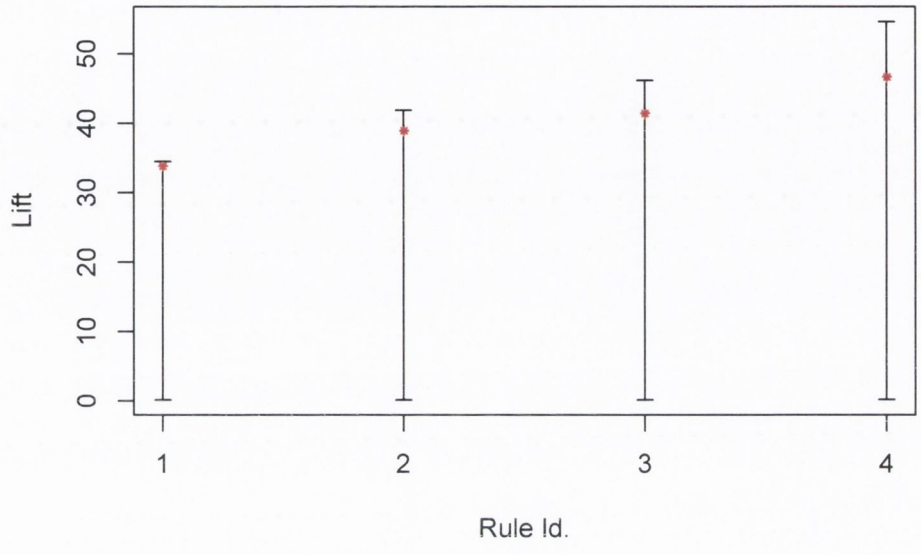


Figure 8.1: The lift of the rules in Table 8.2 with upper (υ) and lower (λ) bounds.

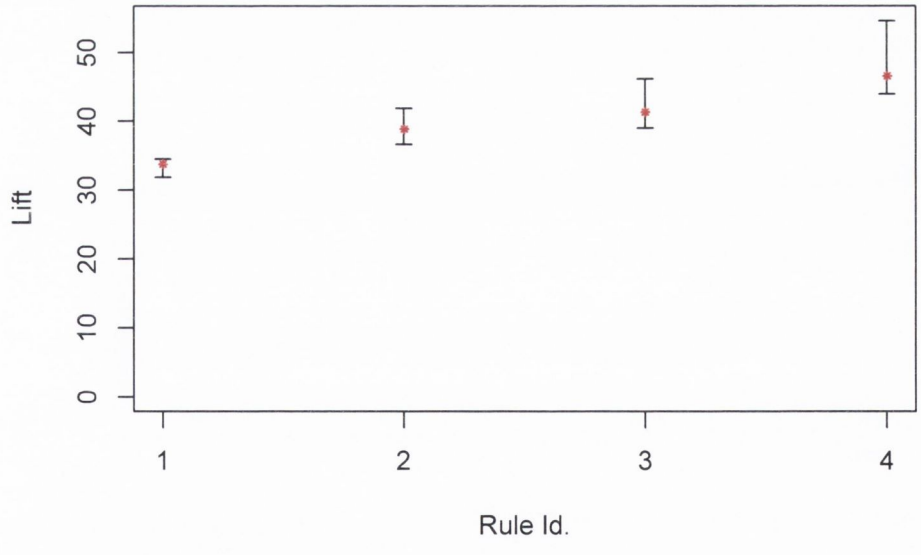


Figure 8.2: The lift of the rules in Table 8.2 with upper (υ) and lower (λ^*) bounds.

Figures 8.1 and 8.2 confirm that ordering these rules by confidence gives a ranking that

is consistent with the position of the lifts in relation to their respective upper and lower bounds. Figure 8.2 is, however, much better for this purpose, providing much tighter and more revealing lower bounds. These figures, and Figure 8.2 in particular, illustrate why rules with greater lift are not necessarily better rules.

The lift of each rule in Table 8.2 was then standardised according to Equation 8.5 and Equation 8.6 and their standardised lifts, $\mathcal{L}$ and $\mathcal{L}^*$ respectively, are given in Table 8.3.

Table 8.3: Standardised lifts for the four rules listed in Table 8.1.

Id.	Rule	Conf.	Lift	$\mathcal{L}$	$\mathcal{L}^*$
1	{GY501, CK701, TR051, RC001} ⇒ {DN002}	97.92%	33.76	0.979	0.727
2	{CK701, TR051, DN002, RC001} ⇒ {GY501}	92.76%	38.84	0.927	0.420
3	{GY501, CK701, DN002, RC001} ⇒ {TR051}	89.52%	41.31	0.895	0.326
4	{TR051, GY501, DN002, RC001} ⇒ {CK701}	85.20%	46.54	0.852	0.246

The values of $\mathcal{L}^*$ in Table 8.3 give a better, more discriminatory, range of values for standardised lift than those given by $\mathcal{L}$. This example illustrates how standardised lift may be used to choose the 'best' rule from amongst multiple rules that are comprised of the same items.

Ranking association rules by standardised lift reflects how close the lift of each rule is to the maximum value that it can take given the support of its antecedent and consequent, and, if applicable, minimum support and confidence thresholds. This presents a natural and unambiguous method of ranking association rules.

8.3.3 Analysis of Rules with Consequent 'Female'

Rules

In Chapter 7 rules with a gender variable in the (singleton) consequent were also considered. In these cases Gray & Orlowska's Interestingness, $\text{Int}(A \Rightarrow B, 2, 2)$, was used to rank these rules. The three highest ranked rules with consequent 'female' are given in Table 8.4.

The ranking of the rules in Table 8.4 by $\text{Int}(A \Rightarrow B, 2, 2)$ is different to that by lift or by confidence.

Table 8.4: The highest ranked, by $\text{Int}(A \Rightarrow B; 2, 2)$, rules with consequent $\{F\}$.

Id.	Antecedent	Support	Confidence	Lift	Int.
1	{Social Care at DIT}	4.99%	94.10%	1.72	0.00165
2	{Early Child. Care & Ed. at DIT}	4.49%	98.13%	1.80	0.00140
3	{BEd at St. Patrick's}	4.01%	87.23%	1.60	0.00098

Standardised Lift

The upper and lower limits of lift were computed along with the standardised lift $\mathcal{L}^*$ for each rule in Table 8.4. These results can be seen in Figure 8.3 and Table 8.5.

Figure 8.3: The lift of the rules in Table 8.4 with upper (υ) and lower (λ^*) bounds.

This ranking is different to that by Gray and Orlowska's interestingness and it does, in this instance, agree with the ranking by confidence. The agreement between the ranking by standardised lift and that by confidence is discussed further in Section 8.3.4.

Table 8.5: Standardised lift for the three highest ranked rules with consequent $\{F\}$.

Id.	Antecedent	Supp.	Conf.	Lift	Int.	$\mathcal{L}^*$
1	{Social Care at DIT}	4.99%	94.10%	1.72	0.00165	0.71
2	{Early Child. Care & Ed. at DIT}	4.49%	98.13%	1.80	0.00140	0.91
3	{BEd at St. Patrick's}	4.01%	87.23%	1.60	0.00098	0.36

8.3.4 Remark

Despite the results of the analyses in this section, ranking by standardised lift and ranking by confidence is not generally the same. It has been the same in the rules that we have considered with consequent 'female' because the lift was bounded by $[c/P(B), 1/P(B)]$ in all of these cases and so $\mathcal{L}^*$ was given by

$$\frac{L-\lambda^*}{v-\lambda^*} = \frac{\frac{P(A,B)}{P(A)P(B)} - \frac{c}{P(B)}}{\frac{c}{P(B)} - \frac{1}{P(B)}} = \frac{P(B \mid A) - c}{1-c} = \frac{c(A \Rightarrow B) - c}{1-c},$$

which is a linear function of confidence for fixed c. The ranking of the rules in Section 8.3.2 by standardised lift also agreed with the ranking by confidence.

8.4 Example II: German Social Life Feelings

8.4.1 The Data

The data analysed in this section are taken from a study of 'social life feelings' that appeared in Scheussler (1982) and Krebs & Schuessler (1987), and has been analysed many times. These analyses include Bartholomew & Schuessler (1991), Bartholomew (1991), Bartholomew *et al.* (1997), de Menezes & Bartholomew (1996) and Bartholomew & Knott (1999). The purpose of the analysis in this section is to demonstrate that the standardised lift can give a different ranking to either confidence or lift and to demonstrate the potential importance of negations in association rule analysis. Due to the extensive body of literature on these data, no in-depth analysis in conducted herein.

Table 8.6: The questions that were asked in the German 'social life feeling' study.

1.	Anyone can raise his standard of living if he is willing to work at it.
2.	Our country has too many poor people who can do little to raise their standard of living.
3.	Individuals are poor because of the lack of effort on their part.
4.	Poor people could improve their lot if they tried.
5.	Most people have a good deal of freedom in deciding how to live.

The data used herein are the answers given by a sample of 1,490 Germans to the questions in Table 8.6. These are the data that were analysed by Bartholomew & Knott (1999) and

they were sourced from `www.kendallslibrary.com`. Each of the five questions that were asked could be answered either 'yes' or 'no'. Overall, there were 3,224 'yes' answers and 3,227 'no' answers. The distribution of the number of 'yes' answers per question is given in Figure 8.4, which reveals that only questions 2 and 3 had more than 50% 'yes' answers.

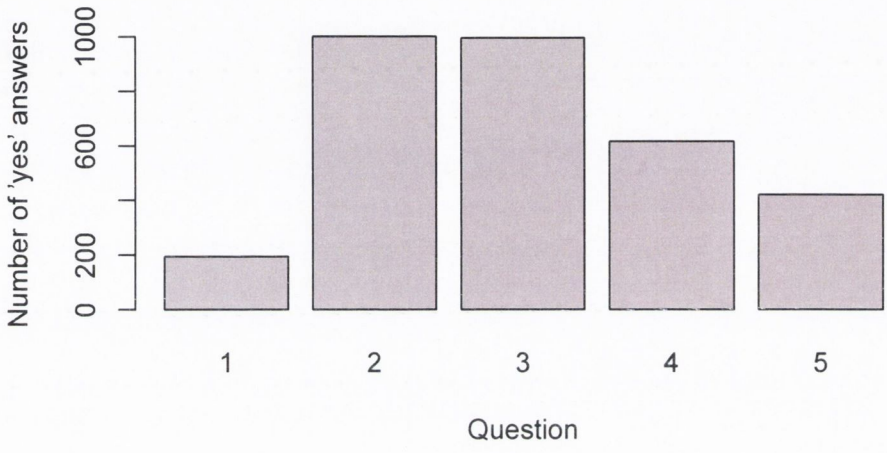


Figure 8.4: The number of 'yes' answers given to each question by the sample of surveyed Germans.

8.4.2 Negations & Coding

These data raise an interesting point; the fact that 'yes' is coded '1' and 'no' is coded '0' can be viewed as arbitrary. Further, had the questions been worded differently, the '1's and '0's could have been be flipped in some or all of the questions. Negations were therefore used in this analysis. In this case, the '1's were coded y1, y2, y3, y4 and y5 respectively while the '0's, or negations, were coded n1, n2, n3, n4 and n5 respectively. Then the apriori algorithm was used to mine the rules.

This straightforward method of mining rules containing negations is facilitated by a relatively small data set and would not be practical in most cases. Further discussion on negations is given in Section 8.5.

8.4.3 Resulting Rules

Association rules were generated using the arules package in R with minimum support set at 20% and minimum confidence at 80%. This approach led to the generation of 38 association rules, the top ten of which, ranked by $\mathcal{L}^*$, are given in Table 8.7.

Table 8.7: Top ten rules from the German 'social life feeling' data, ranked by $\mathcal{L}^*$.

Id.	Rule	Supp.	Conf.	Lift	$\mathcal{L}^*$	LB	UB
1	{n3,n4} ⇒ {n1}	0.262	0.951	1.095	0.757	0.920	1.151
2	{n3,n5} ⇒ {n1}	0.255	0.945	1.088	0.726	0.920	1.151
3	{n4,n5} ⇒ {n1}	0.446	0.945	1.087	0.723	0.920	1.151
4	{n3} ⇒ {n1}	0.311	0.935	1.076	0.677	0.920	1.151
5	{n4} ⇒ {n1}	0.548	0.935	1.076	0.674	0.920	1.151
6	{n2,n4} ⇒ {n1}	0.225	0.949	1.092	0.673	0.971	1.151
7	{n4,n5,y2} ⇒ {n1}	0.256	0.934	1.075	0.670	0.920	1.151
8	{n3,n4,n5} ⇒ {n1}	0.221	0.954	1.097	0.667	0.991	1.151
9	{n4,y2} ⇒ {n1}	0.323	0.925	1.064	0.626	0.920	1.151
10	{n4,n5,y3} ⇒ {n1}	0.225	0.936	1.077	0.617	0.958	1.151

All of the rules in Table 8.7 had the same upper bound for lift because they all had the same consequent, which had greater support in each case, than the antecedent. The lift of each rule can be seen in context of its upper and lower bounds in Figure 8.5.

8.4.4 Results

Although the rules had not been pruned to any great extent, looking at Table 8.7, some interesting rules became apparent. The rule with the highest value of standardised lift was {n3, n4} ⇒ {n1}. That is, 95.1% of those who did not agree that people were poor because of lack of effort or that poor people could improve their lot if they tried also did not agree that people could raise their standard of living if they were willing to work at it.

Note that the ranking of the rules in Table 8.7 is by $\mathcal{L}^*$ and is different than the ranking by either confidence or lift. This example illustrates $\mathcal{L}^*$ as a new and useful method of ranking association rules. Furthermore, if a pruning technique like that described in Section 7.5.2 was applied to these data, $\mathcal{L}^*$ could be used to effectively break ties.

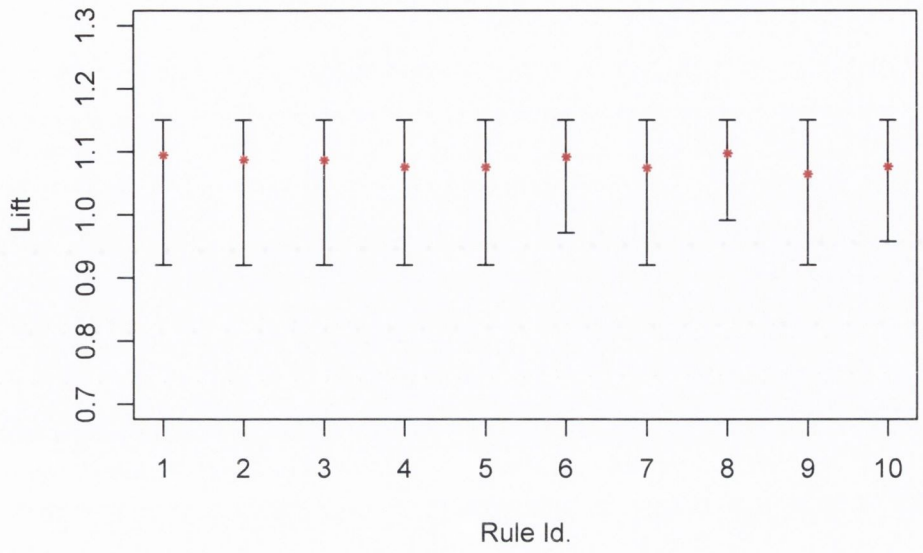


Figure 8.5: The lift of the ten rules, given in Table 8.7, along with their upper (υ) and lower (λ^*) bounds.

8.5 Negations

8.5.1 Background — Negative Association Rules

Silverstien *et al.* (1998) proposed the concept of association rules involving negations. Savasere *et al.* (1998) demonstrated a method of mining rules of the form $A \not\Rightarrow B$, which they referred to as 'negative association rules'. Hofmann & Wilhelm (2001) proposed mosaic plots as a means of visualising all combinations of presence or absence of the antecedent and consequent of an association rule; this is a method of deriving rules containing negations after mining, however it is limited in its effectiveness when the consequent is not singleton. Wu *et al.* (2002, 2004) considered rules of the form $A \Rightarrow \overline{B}$, $\overline{A} \Rightarrow B$ and $\overline{A} \Rightarrow \overline{B}$ and provided algorithms to mine such rules.

8.5.2 How Many More Rules

In Appendix F.2 it is shown that the number of rules that can possibly be generated from an itemset consisting of n items and their negations is given by

$$5^n - 2\,(3^n) + 1. \tag{8.7}$$

If negations are not considered, this number is given by

$$\sum_{i=2}^{n} {}^{n}C_i \left(2^i - 2\right) = 3^n - 2\left(2^n\right) + 1$$

(Hipp *et al.*, 2002), where

$${}^{n}C_r = \frac{n!}{r!(n-r)!}.$$

These bounds and, in particular, results that came about in the process of deriving these bounds are similar to the results of Fréchet (1951).

Now, it is trivial to work out the number of 'extra' rules that can be mined when negations are included;

$$5^n - 2\left(3^n\right) + 1 - \left[3^n - 2\left(2^n\right) + 1\right] = 5^n - 3^{n+1} + 2^{n+1}.$$

When viewed as a proportion of the total number, the amount of rules that contain at least one negation is given by

$$\frac{5^n - 3^{n+1} + 2^{n+1}}{5^n - 2\left(3^n\right) + 1}. \tag{8.8}$$

Therefore, when an itemset of just 20 items is considered, it follows from Equation 8.8 that 99.996% of potential association rules involve negations.

Of course, it may be argued that the number of rules given by Equation 8.7 is unrealistic because a minimum of 2^n transactions would be required in order that all potential rules may exist. Furthermore, it is highly unlikely in most practical applications that a transaction involving all, or even most, of the items will occur. For example, in a convenience store application it is highly unlikely that any one customer will purchase more than 0.5% of the items. However, regardless of these considerations, Equation 8.8 provides a useful figure for the proportion of potential rules that will contain negations.

8.5.3 Mining Association Rules Involving Negations

The huge amount of extra rules that can be mined when negations are included in the mining process presents many computational problems. Furthermore, the method of mining association rules involving negations demonstrated in Section 8.4 is not viable in general. Thiruvady & Webb (2004) and Gan *et al.* (2005) presented efficient algorithms to mine rules containing negations.

Similarly to the interestingness of a rule, the importance of negations in association rule mining depends on the data and on the purpose of the analysis. In the context of the German social life feeing data analysed herein, an argument was presented for the inclusion of negations in the mining paradigm. However, no such argument was made in relation to the CAO data because applicants were restricted to selecting at most ten courses and so each applicant would have over 500 negations.

8.6 Summary

A new function for ranking association rules, standardised lift, has been introduced. A method of visualising standardised lift or, more precisely, lift relative to its upper and lower bounds, was also introduced. This function, and all theory around it, calls only upon support, confidence and lift — all of which are common and widely used functions of association rules — and, if applicable, minimum support and confidence thresholds.

This new method of ranking rules was illustrated on two data sets and compared to existing functions of interestingness. In the analysis of second of these datasets, the German social life feeling data, negations were introduced to facilitate the mining process. An argument was given for the inclusion of negations in the association rule mining process, including a quantification of the amount of rules that can be mined when negations are included.

Chapter 9

Conclusions

9.1 Summary

This work has focused on the development and implementation of two topics in unsupervised learning; model-based clustering *via* parsimonious Gaussian mixture models and analysis of binary data *via* association rule mining. Significant additions have been made to the body of literature on these topics.

9.1.1 Model-Based Clustering

A new family of mixture models for model-based clustering has been introduced. Well-established mixture models were used for motivation and the model parameters for each member of this new family of models were estimated using an AECM algorithm. The 'best' member of the family was chosen using a well-established criterion — the Bayesian information criterion. The clustering of the data, according to each model, was given by the values assigned to the group membership labels through the learning process.

These models were then applied to coffee and wine data and performed favorably when compared to well-established techniques. Although initially comprised of eight members, this family was extended to twelve members and then applied to crabs data giving excellent results when compared to other model-based clustering techniques.

A modified Cholesky decomposition was used to build the framework for another family of

Gaussian mixture mixture models that are suitable for the analysis of longitudinal data. Four members of this family were introduced and applied to real data, where they performed well, giving good clustering results.

9.1.2 Association Rule Mining

The application of association rule mining to college application data was presented as a novel application that included a simple, but very effective, pruning strategy. This analysis contributed to the discussion over the existence of a 'points race'. A new and very natural measure of interestingness was introduced and a method of visualising this new type of interestingness was presented. The number of association rules that can arise from a given itemset when negations are included in the mining paradigm was quantified for the first time. This new measure of interestingness was demonstrated on the college application data and on German social data. The analysis of the German data was also used to illustrate the potential importance of negations.

9.2 Further Work

9.2.1 Modelling the Mean in Model-Based Clustering

In the earlier part of this work there was some focus on modelling the covariance structure in Gaussian mixture models. There is also scope to model the mean structure. The saving, in terms of potential number of parameters, that can be realised by modelling the mean is not as great as that which may be realised by modelling the covariance structure. For a Gaussian mixture model involving G groups and p variables, modelling the group means $\boldsymbol{\mu}_g$ will require Gp parameters. However, in situations where there may be a lot of groups, this saving may be worthwhile.

For example, some of the group means could be considered equal, but not others, or, in the case of longitudinal data, the mean of each group could be modelled using a general polynomial spline, with some group-specific correction applied where necessary.

9.2.2 Allowing the Number of Factors to Vary Across Groups

The PGMM family of models could be further extended by allowing the number of factors q to vary across groups. This approach would allow greater flexibility for model fitting. However, the resulting increase in the number of models would be so great that this approach would be highly computationally intensive.

9.2.3 Application of Generalised Procrustes Analysis to Model-Based Clustering

Generalised Procrustes analysis (Gower, 1975), or GPA, is a method of matching or aligning one matrix with another. The elements of the matrices are matched using rotation, translation and scaling. Parameters of mixture models, or even a collection of parameters, could be subjected to GPA. This would have the effect of aligning parameters across groups. Consider the PGMM family of models as an example; the loading matrices $\boldsymbol{\Lambda}_g$ could be subjected to GPA to facilitate estimation of the posterior expected value of the latent variables given the data, the model parameters and membership of group g; that is

$$\mathbb{E}\left[\mathbf{u}_i \mid \boldsymbol{\mu}_g, \mathbf{x}_i, \boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g\right] = \boldsymbol{\Lambda}_g' \boldsymbol{\Sigma}_g^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_g),$$

where $\boldsymbol{\Lambda}_g'$ is the only term that is not rotation invariant.

9.2.4 Parallelisation

There is scope for parallelisation of the C code that was written to perform the AECM algorithm on the PGMM family of models. The code already written, and described in Appendix D, is 'embarrassingly parallelisable' because it can easily be compartmentalised into independent sections defined by the triple (model, G, q). Parallelising the code in such a fashion would greatly reduce the computation time and allow full exploitation of the linear relationship between the number of covariance parameters and the dimensionality of the data.

9.2.5 Further Expansion of the CDGMM Family

There is great scope to expand the CDGMM family of models beyond the four initial members that were presented herein. As mentioned in Section 5.5, this could be achieved by imposing constraints on the sub-diagonals of the matrix $\mathbf{T}_g$.

9.2.6 Model Selection & Convergence Criteria

In the analyses of data carried out using the PGMM and GPGMM families of models it became apparent that the model with the greatest BIC was not always the best model in terms of classification. There is work to be done on the viability of alternatives to the BIC, such as the ICL, for model selection. There is also work to be done on the use of different convergence criteria, such as Aitken's acceleration, with MCLUST.

9.2.7 Association Rules

There is some work to be done on an association rule approach to categorical data. There has already been some work carried out in this area, such as that given in Tan *et al.* (2006, Chapter 7), but much is left to do. The application of some of the ideas around standardised lift that were raised herein, may lead to the emergence of interesting concepts in this area. Furthermore, the resulting theory may be so different from association rule mining as to constitute a new data mining technique.

9.2.8 The CAO Data

An analysis of the CAO data that uses the entry points from the previous year to try to predict the courses selected, or the order of the courses selected, by applicants may reveal interesting results. Repeating the analysis of Chapter 7 with data from another year may also be useful. The repetition of the analysis herein with the inclusion of a variable indicating the geographical location of the applicant's school may confirm the tendency towards nearby universities, for a cohort of students, that is suggested herein.

Bibliography

Agrawal, R. & Srikant, R. (1994), Fast algorithms for mining association rules in large databases, *in* '20th International Conference on Very Large Data Bases', Santiago, Chile.

Agrawal, R., Imielinski, T. & Swami, A. (1993), Mining association rules between sets of items in large databases, *in* 'ACM SIGMOD International Conference on Management of Data', Washington DC, USA.

Agresti, A. (2002), *Categorical Data Analysis*, 2nd edn, John Wiley & Sons, New York, USA.

Anton, H. & Rorres, C. (1994), *Elementary Linear Algebra*, 7th edn, John Wiley & Sons, New York, USA.

Banfield, J. D. & Raftery, A. E. (1993), 'Model-based Gaussian and non-Gaussian clustering', *Biometrics* **49**(3), 803–821.

Bartholomew, D. & Knott, M. (1999), *Latent Variable Models and Factor Analysis*, Kendall's Library of Statistics, 2nd edn, Arnold, London.

Bartholomew, D., de Menezes, L. & Tzamourani, P. (1997), Latent trait and latent class models applied to survey data, *in* J. Rost & R. Langeheine, eds, 'Applications of Latent Traits and Latent Class Models in the Social Sciences', Waxmann, Münster.

Bartholomew, D. J. (1991), 'Estimating relationships between latent variables', *Sankhyā. The Indian Journal of Statistics. Series A* **55**, 409–419.

Bartholomew, D. J. & Schuessler, K. F. (1991), 'Reliability of attitude scores based on a latent trait model', *Sociological Methodology* **21**, 97–123.

Benoît (1924), 'Note sur une méthode de résolution des équations normales provenant de l'application de la méthode des moindres carrés à un système d'équations linéaires en nombre inférieur celui des inconnues (Procédé du Commandant Cholesky)', *Bulletin Géodésique* **2**, 67–77.

Biernacki, C., Celeux, G. & Govaert, G. (2000), 'Assessing a mixture model for clustering with the integrated completed likelihood', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(7), 719–725.

Böhning, D., Dietz, E., Schaub, R., Schlattmann, P. & Lindsay, B. (1994), 'The distribution of the likelihood ratio for mixtures of densities from the one-parameter exponential family', *Annals of the Institute of Statistical Mathematics* **46**, 373–388.

Borgelt, C. (2003), Efficient implementations of apriori and eclat, *in* 'Workshop of Frequent Item Set Mining Implementations', Florida, USA.

Borgelt, C. & Kruse, R. (2002), Induction of association rules: Apriori implementation, *in* '15th COMPSTAT Conference', Berlin, Germany.

Bruzzese, D. & Davino, C. (2001), 'Statistical pruning of discovered association rules', *Computational Statistics* **16**(3), 387–398.

Campbell, N. A. & Mahon, R. J. (1974), 'A multivariate study of variation in two species of rock crab of genus leptograpsus', *Australian Journal of Zoology* **22**, 417–425.

Castelo, R., Feelders, A. & Siebes, A. (2001), MAMBO: Discovering association rules based on conditional independencies, *in* 'Proceedings of 4th International Symposium on Intelligent Data Analysis', Cascais, Portugal.

Celeux, G. & Govaert, G. (1992), 'A classification EM algorithm for clustering and two stochastic versions', *Computational Statistics and Data Analysis* **14**(3), 315–332.

Celeux, G. & Govaert, G. (1995), 'Gaussian parsimonious clustering models', *Pattern Recognition* **28**, 781–793.

Chang, W. C. (1983), 'On using principal components before separating a mixture of two multivariate normal distributions', *Applied Statistics* **32**(3), 267–275.

Clancy, P. (2001), *College Entry in Focus: a fourth national survey of access to higher education*, The Higher Education Authority.

Cohen, J. (1960), 'A coefficient of agreement for nominal scales', *Educational and Psychological Measurement* **20**, 37–46.

Crowder, M. J. & Hand, D. J. (1990), *Analysis of repeated measures*, Chapman and Hall, London.

de Menezes, L. M. & Bartholomew, D. J. (1996), 'New developments in latent structure analysis applied to social attitudes', *Journal of the Royal Statistical Society. Series A* **159**, 213–224.

Dean, N. & Raftery, A. E. (2006), *The clustvarsel Package.* R package version 0.2-4.

Dean, N., Murphy, T. B. & Downey, G. (2006), 'Using unlabelled data to update classification rules with applications in food authenticity studies', *Journal of the Royal Statistical Society. Series C* **55**(1), 1–14.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum likelihood from incomplete data via the EM algorithm', *Journal of the Royal Statistical Society. Series B* **39**(1), 1–38.

Dong, G. & Li, J. (1998), Interestingness of discovered association rules in terms of neighbourhood-based unexpectedness, *in* 'Pacific-Asia Conference on Knowledge Discovery and Data Mining'.

Drew, E. (2006), *Facing Extinction? Why Men are not Attracted to Primary Teaching*, The Liffey Press, Dublin.

Forina, M., Armanino, C., Castino, M. & Ubigli, M. (1986), 'Multivariate data analysis as a discriminating method of the origin of wines', *Vitis* **25**, 189–201.

Fraleigh, J. B. (1990), *Calculus with Analytic Geometry*, 3rd edn, Addison-Wesley.

Fraley, C. & Raftery, A. E. (1998), 'How many clusters? Which clustering methods? Answers via model-based cluster analysis', *The Computer Journal* **41**(8), 578–588.

Fraley, C. & Raftery, A. E. (1999), 'MCLUST: Software for model-based cluster analysis', *Journal of Classification* **16**, 297–306.

Fraley, C. & Raftery, A. E. (2002*a*), MCLUST: Software for model-based clustering, density estimation, and discriminant analysis, Technical Report 415, University of Washington, Department of Statistics.

Fraley, C. & Raftery, A. E. (2002*b*), 'Model-based clustering, discriminant analysis, and density estimation', *Journal of the American Statistical Association* **97**, 611–631.

Fraley, C. & Raftery, A. E. (2003), 'Enhanced software for model-based clustering, density estimation, and discriminant analysis: MCLUST', *Journal of Classification* **20**, 263–286.

Fréchet, M. (1951), 'Sur les tableaux de corrélation dont les marges sont données', *Annales de l'Université de Lyon, Sciences Mathématiques et Astronomie* **14**, 53–77.

Gan, M., Zhang, M.-Y. & Wang, S.-W. (2005), One extended form for negative association rules and the corresponding mining algorithm, *in* 'Proceedings of 2005 International Conference on Machine Learning and Cybernetics', Vol. 3, pp. 1716–1721.

Ghahramani, Z. & Hinton, G. E. (1997), The EM algorithm for factor analyzers, Technical Report CRG-TR-96-1, University Of Toronto, Toronto.

Gormley, I. C. & Murphy, T. B. (2006), 'Analysis of Irish third-level college applications data', *Journal of the Royal Statistical Society. Series A* **169**(2), 361–379.

Gower, J. C. (1975), 'Generalized Procrustes analysis', *Psychometrika* **40**(1), 33–51.

Gray, B. & Orlowska, M. (1998), CCAIIA: Clustering categorical attributes into interesting association rules, *in* 'Pacific-Asia Conference on Knowledge Discovery and Data Mining'.

Graybill, F. A. (1983), *Matrices with Applications in Statistics*, 2nd edn, Wadsworth, Belmont, California.

Hahsler, M., Gruen, B. & Hornik, K. (2005), *arules: Mining Association Rules and Frequent Itemsets*. R package version 0.2-4.

Hartigan, J. A. & Kleiner, B. (1981), Mosaics for contingency tables, *in* W. F. Eddy, ed., '13th Symposium on the Interface Between Computer Science and Statistics', Springer-Verlag, New York.

Hastie, T., Tibshirani, R. & Friedman, J. (2001), *The Elements of Statistical Learning*, Springer, New York.

Hipp, J., Güntzer, U. & Nakhaeizadeh, G. (2002), Data mining of association rules and the process of knowledge discovery in databases, *in* 'Industrial Conference on Data Mining', Springer-Verlag, London, UK, pp. 15–36.

Hofmann, H. & Wilhelm, A. (2001), 'Visual comparison of association rules', *Computational Statistics* **16**(3), 399–415.

Hubert, L. & Arabie, P. (1985), 'Comparing partitions', *Journal of Classification* **2**, 193–218.

Hunter, D. L. & Lange, K. (2000), 'Rejoinder to discussion of "Optimization transfer using surrogate objective functions"', *Journal of Computational and Graphical Statistics* **9**, 52–59.

Hunter, D. L. & Lange, K. (2004), 'A tutorial on MM algorithms', *The American Statistician* **58**(1), 30–37.

Hurley, C. (2004), 'Clustering visualizations of multivariate data', *Journal of Computational and Graphical Statistics* **13**(4), 788–806.

Hyland, A. (1999), *Commission on the Points System. Final Report and Recommendations*, The Stationary Office, Dublin. [Áine Hyland was the chairperson. A full list of authors is given on page 175 of the report.].

Jensen, J. L. W. V. (1906), 'Sur les fonctions convexes et les inégalités entre les valeurs moyennes', *Acta Mathematica* **30**, 175–193.

Kass, R. E. & Raftery, A. E. (1995), 'Bayes factors', *Journal of the American Statistical Association* **90**, 773–795.

Keller, R. & Schlögl, A. (2002), 'PISSARRO: Picturing interactively statistically sensible association rules reporting overviews'. Department of Computer Oriented Statistics and Data Analysis, Augsburg University, Germany.

Keribin, C. (1998), 'Estimation consistante de l'ordre de modèles de mélange', *Comptes Rendus de l'Académie des Sciences. Série I. Mathématique* **326**(2), 243–248.

Keribin, C. (2000), 'Consistent estimation of the order of mixture models', *Sankhyā. The Indian Journal of Statistics. Series A* **62**(1), 49–66.

Krebs, D. & Schuessler, K. F. (1987), *Soziale Empfindungen: ein interkultureller Skalenvergleich bei Deutschen und Amerikanern*, Monographien: Sozialwisenschaftliche Methoden, Campus Verlag, Frankfurt, Milan and New York.

Lagrange, J. L. (1788), *Méchanique Analitique*, Chez le Veuve Desaint, Paris.

Lawley, D. N. & Maxwell, A. E. (1962), 'Factor analysis as a statistical method', *The Statistician* **12**(3), 209–229.

Lawley, D. N. & Maxwell, A. E. (1971), *Factor Analysis as a Statistical Method*, Butterworths, London.

Leroux, B. G. (1992), 'Consistent estimation of a mixing distribution', *The Annals of Statistics* **20**, 1350–1360.

Lindsay, B. G. (1995), Mixture models: Theory, geometry and applications, *in* 'NSF-CBMS Regional Conference Series in Probability and Statistics', Vol. 5, Hayward, California: Institute of Mathematical Statistics.

Lütkepohl, H. (1996), *Handbook of Matrices*, John Wiley & Sons, Chicester.

Lynch, C. D., McConnell, R. J. & Hannigan, A. (2006), 'Dental school admissions in Ireland: can current selection criteria predict success?', *European Journal of Dental Education* **10**, 73–79.

Lynch, K., Brannick, T., Clancy, P. & Drudy, S. (1999), *Commission on the points system. Research paper no. 4. Points and performance in higher education: A study of the predictive validity of the points system*, The Stationary Office, Dublin.

Magnus, J. R. & Neudecker, H. (1999), *Matrix Differential Calculus with Applications in Statistics and Econometrics*, Revised edn, John Wiley & Sons, Chicester.

McLachlan, G. J. & Basford, K. E. (1988), *Mixture Models: Inference and applications to clustering*, Marcel Dekker Inc., New York.

McLachlan, G. J. & Krishnan, T. (1997), *The EM Algorithm and Extensions*, Wiley, New York.

McLachlan, G. J. & Peel, D. (1998), Robust cluster analysis via mixtures of multivariate t-distributions, *in* 'Lecture Notes in Computer Science', Vol. 1451, Springer-Verlag, Berlin, pp. 658–666.

McLachlan, G. J. & Peel, D. (2000*a*), *Finite Mixture Models*, John Wiley & Sons, New York.

McLachlan, G. J. & Peel, D. (2000*b*), Mixtures of factor analyzers, *in* 'Seventh International Conference on Machine Learning', San Francisco.

McLachlan, G. J., Peel, D. & Bean, R. W. (2003), 'Modelling high-dimensional data by mixtures of factor analyzers', *Computational Statistics & Data Analysis* **41**(3–4), 379–388.

Meng, X. L. & Rubin, D. B. (1993), 'Maximum likelihood estimation via the ECM algorithm: a general framework', *Biometrika* **80**, 267–278.

Meng, X. L. & van Dyk (1997), 'The EM algorithm — an old folk song sung to the fast tune (with discussion)', *Journal of the Royal Statistical Society Series B* **59**, 511–567.

Moran, M. A. & Crowley, M. J. (1979), 'The leaving certificate and first year university performance', *Journal of the Statistical and Social Inquiry Society of Ireland* **14**(1), 231–266.

O'Connell, P. J., Clancy, P. & McCoy, S. (2006), *Who went to college in 2004? A national survey of new entrants to higher education*, The Higher Education Authority.

Ooche, W., Kastelijn, H. & DeWaele, A. (1981), 'Détermination de l'origine d'un vin rouge á l'aide du spectre des acides aminés', *Annales des Falsifications et de l'Expertise Chimique* **74**, 381–408.

Ortega, J. M. & Rheinboldt, W. C. (1970), *Iterative Solutions of Nonlinear Equations in Several Variables*, Academic Press, New York.

Pan, J. & MacKenzie, G. (2003), 'On modelling mean-covariance structures in longitudinal studies', *Biometrika* **90**(1), 239–244.

Pan, J. & MacKenzie, G. (2006), 'Regression models for covariance structures in longitudinal studies', *Statistical Modelling* **6**, 43–57.

Pasquier, N., Bastide, Y., Taouil, R. & Lakhal, L. (1999), Discovering frequent closed itemsets for association rules, *in* 'Seventh International Conference on Database Theory'.

Pei, J., Han, J. & Mao, R. (2000), CLOSET: An efficient algorithm for mining frequent closed itemsets, *in* 'Proceedings of ACM-SIGMOD International Workshop on Data Mining and Knowledge Discovery', Dallas, Texas.

Plackett, R. L. (1975), 'The analysis of permutations', *Applied Statistics* **24**, 193–202.

Pourahmadi, M. (1999), 'Joint mean-covariance models with applications to longitudinal data: unconstrained parameterisation', *Biometrika* **86**(3), 677–690.

Pourahmadi, M. (2000), 'Maximum likelihood estimation of generalised linear models for multivariate normal covariance matrix', *Biometrika* **87**(2), 425–435.

Pourahmadi, M., Daniels, M. & Park, T. (2007), 'Simultaneous modelling of the Cholesky decomposition of several covariance matrices', *Journal of Multivariate Analysis* **98**, 568–587.

R Development Core Team (2006), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.

Raftery, A. E. & Dean, N. (2006), 'Variabel selection for model-based clustering', *Journal of the American Statistical Association* **101**(473), 168–178.

Rand, W. M. (1971), 'Objective criteria for the evaluation of clustering methods', *Journal of the American Statistical Association* **66**, 846–850.

Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, UK.

Savasere, A., Omiecinski, E. & Navathe, S. (1998), Mining for strong negative associations in a large database of customer transactions, *in* 'Proceedings 14th International Conference on Data Engineering', Florida, USA, pp. 494–502.

Scheussler, K. F. (1982), *Measuring Social Life Feelings*, Jossey-Bass Social and Behavioural Science Series, Jossey-Bass, San Francisco.

Schwartz, G. (1978), 'Estimating the dimension of a model', *Annals of Statistics* **6**, 31–38.

Silberschatz, A. & Tuzhilin, A. (1995), On subjective measures of interestingness in knowledge discovery, *in* 'Knowledge Discovery and Data Mining', pp. 275–281.

Silverstien, C., Brin, S. & Motwani, R. (1998), 'Beyond market baskets: Generalizing association rules to dependence rules', *Data Mining and Knowledge Discovery* **2**, 39–68.

Spearman, C. (1904), 'The proof and measurement of association between two things', *American Journal of Psychology* **15**, 72–101.

Streuli, H. (1973), Der heutige stand der kaffeechemie, *in* 'Association Scientifique International du Cafe, 6th International Colloquium on Coffee Chemisrty', Bogatá, Columbia, pp. 61–72.

Tan, P.-N., Steinbach, M. & Kumar, V. (2006), *Introduction to Data Mining*, Addison-Wesley, Boston, MA, USA.

Thiruvady, D. R. & Webb, G. I. (2004), Mining negative rules in large databases using GRD, *in* 'Pacific-Asia Conference on Knowledge Discovery and Data Mining', Sydney, Austrilia, pp. 161–165.

Tipping, T. E. & Bishop, C. M. (1999*a*), 'Mixtures of probabilistic principal component analysers', *Neural Computation* **11**(2), 443–482.

Tipping, T. E. & Bishop, C. M. (1999*b*), 'Probabilistic principal component analysers', *Journal of the Royal Statistical Society. Series B* **61**, 611–622.

Titterington, D. M., Smith, A. F. M. & Makov, U. E. (1985), *Statistical Analysis of Finite Mixture Distributions*, John Wiley & Sons, Chichester.

Tuohy, D. (1998), *Commission on the points system. Research paper no.1. Demand for third-level places. Interests, fields of study and the effect of the points system in the application process for 1997*, The Stationary Office, Dublin.

Vazirgiannis, M., Halkidi, M. & Gunopulos, D. (2003), *Uncertainty Handling and Quality Assessment in Data Mining*, Springer-Verlag, London.

Venables, W. N. & Ripley, B. D. (1999), *Modern Applied Statistics with S-PLUS*, Springer.

Wilhelm, A. (2002), Statistical tests for pruning association rules, *in* 'Computational Statistics: 15th Symposium', Berlin, Germany.

Wolfe, J. H. (1963), Object cluster analysis of social areas, Master's thesis, University of California, Berkley.

Wu, X., Zhang, C. & Zhang, S. (2002), Mining both positive and negative association rules, *in* '19th International Conference on Machine Learning', Sydney , Australia, pp. 658–665.

Wu, X., Zhang, C. & Zhang, S. (2004), 'Efficient mining of both positive and negative association rules', *ACM Transactions on Information Systems* **22**(3), 381–405.

Yule, G. U. (1903), 'Notes on the theory of association of attributes in statistics', *Biometrika* **2**(2), 121–134.

Zaki, M. & Hsiao, C. (2002), CHARM: An efficient algorithm for closed itemset mining, *in* '2nd SIAM International Conference on Data Mining', Arlington.

Appendix A

Calculations for PGMMs

A.1 Important Results

The following results from Lütkepohl (1996) and Magnus & Neudecker (1999) are utilised herein;

$$\mathbf{X}_{m\times n}: \frac{\partial \log|\mathbf{X}|}{\partial \mathbf{X}} = \mathbf{X}^{-1}.$$

$$\mathbf{A}_{m\times n}, \mathbf{B}_{n\times m}: \operatorname{tr}(\mathbf{AB}) = \operatorname{tr}(\mathbf{BA}).$$

$$\mathbf{X}_{m\times n}, \mathbf{A}_{n\times m}: \frac{\partial \operatorname{tr}(\mathbf{XA})}{\partial \mathbf{X}} = \frac{\partial \operatorname{tr}(\mathbf{AX})}{\partial \mathbf{X}} = \mathbf{A}'.$$

$$\mathbf{X}_{m\times n}, \mathbf{A}_{p\times m}, \mathbf{B}_{n\times p}: \frac{\partial \operatorname{tr}(\mathbf{AXB})}{\partial \mathbf{X}} = \mathbf{A}'\mathbf{B}'.$$

$$\mathbf{X}_{m\times n}, \mathbf{A}_{n\times n}, \mathbf{B}_{m\times m}: \frac{\partial \operatorname{tr}(\mathbf{XAXB})}{\partial \mathbf{X}} = \mathbf{B}'\mathbf{X}'\mathbf{A}' + \mathbf{A}'\mathbf{X}'\mathbf{B}'.$$

$$\mathbf{X}_{m\times n}, \mathbf{A}_{n\times m}, \mathbf{B}_{n\times m}: \frac{\partial \operatorname{tr}(\mathbf{XAX}'\mathbf{B})}{\partial \mathbf{X}} = \mathbf{B}'\mathbf{XA}' + \mathbf{BXA}.$$

The following Theorems, taken from Graybill (1983), are also important.

Theorem A.1. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be any matrices such that* $\mathbf{AB}$ *is defined, then*

$$(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'.$$

Theorem A.2. *If* $\mathbf{A}$ *is any matrix, then* $\mathbf{A}'\mathbf{A}$ *and* $\mathbf{AA}'$ *are symmetric.*

Theorem A.3. *If* $\mathbf{A}$ *is a nonsingular matrix, then* $\mathbf{A}'$ *and* $\mathbf{A}^{-1}$ *are nonsingular and*

$$(\mathbf{A}')^{-1} = (\mathbf{A}^{-1})'.$$

Theorem A.4. *If each element of the i^{th} row of an $n \times n$ matrix* $\mathbf{A}$ *contains a given factor k, then we may write $|\mathbf{A}| = k|\mathbf{B}|$, where the rows of* $\mathbf{B}$ *are the same as the rows of* $\mathbf{A}$ *except that the number k has been factored from each element of the i^{th} row of* $\mathbf{A}$.

A corollary of this theorem follows without further work; it is also stated as a result in Anton & Rorres (1994).

Corollary A.1. *Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be $n \times n$ matrices such that*

$$\mathbf{A} = k\mathbf{B},$$

where k is a scalar. Then

$$|\mathbf{A}| = k^n|\mathbf{B}|.$$

An important consequence of this corollary is that

$$\log|\psi^{-1}\mathbf{I}_p| = \log\psi^{-p} + \log|\mathbf{I}_p| = p\log\psi^{-1}.$$

A.2 Q Equation

Recall the formula for $Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)$ that was derived in Section 3.3.3;

$$Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \left[\log|\boldsymbol{\Psi}_g^{-1}| - \operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1}\mathbf{S}_g\right\} + 2\operatorname{tr}\left\{\boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\right\} - \operatorname{tr}\left\{\boldsymbol{\Lambda}_g'\boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\right\}\right],$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}_g$, and $\boldsymbol{\Theta}_g = \mathbf{I}_q - \hat{\boldsymbol{\beta}}_g\hat{\boldsymbol{\Lambda}}_g + \hat{\boldsymbol{\beta}}_g\mathbf{S}_g\hat{\boldsymbol{\beta}}_g'$ is a symmetric $q \times q$ matrix.

A.3 Differentiation

It is necessary to work out

$$S_1(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Lambda}_g} \text{ and } S_2(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Psi}_g^{-1}},$$

for each of the eight cases in Table 3.1 and to then solve $S_1(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}}) = 0$ for $\hat{\boldsymbol{\Lambda}}^{\text{new}}$ and $\operatorname{diag}\{S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}}^{\text{new}})\} = 0$ for $\hat{\boldsymbol{\Psi}}^{\text{new}}$ or $S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, (\hat{\psi})^{\text{new}}) = 0$ for $(\hat{\psi})^{\text{new}}$ in each case to get maximum likelihood estimates of these parameters.

A.4 Maximum Likelihood Estimates

A.4.1 Model CCC

For Model CCC; $\mathbf{\Lambda}_g = \mathbf{\Lambda}$ and $\mathbf{\Psi}_g = \mathbf{\Psi} = \psi \mathbf{I}_p$. Therefore, $\hat{\boldsymbol{\beta}} = \hat{\mathbf{\Lambda}}'(\hat{\mathbf{\Lambda}}\hat{\mathbf{\Lambda}}' + \hat{\psi}\mathbf{I}_p)^{-1}$ and Q can be written as

$$Q(\mathbf{\Lambda}, \psi) = C + \frac{n}{2}\Big[p \log \psi^{-1} - \psi^{-1} \operatorname{tr}\{\tilde{\mathbf{S}}\} + 2\psi^{-1} \operatorname{tr}\{\mathbf{\Lambda}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} - \psi^{-1} \operatorname{tr}\{\mathbf{\Lambda}'\mathbf{\Lambda}\tilde{\mathbf{\Theta}}\}\Big].$$

Differentiating Q with respect to $\mathbf{\Lambda}$ and ψ^{-1} respectively gives the score functions below.

$$S_1(\mathbf{\Lambda}, \psi) = \frac{\partial Q(\mathbf{\Lambda}, \psi)}{\partial \mathbf{\Lambda}} = \frac{n}{\psi}\Big[\tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}' - \mathbf{\Lambda}\mathbf{\Theta}\Big].$$

$$S_2(\mathbf{\Lambda}, \psi) = \frac{\partial Q(\mathbf{\Lambda}, \psi)}{\partial \psi^{-1}} = \frac{n}{2}\Big[p\psi - \operatorname{tr}\{\tilde{\mathbf{S}}\} + 2\operatorname{tr}\{\mathbf{\Lambda}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} - \operatorname{tr}\{\mathbf{\Lambda}'\mathbf{\Lambda}\tilde{\mathbf{\Theta}}\}\Big].$$

Now, solving $S_1\big(\hat{\mathbf{\Lambda}}^{\text{new}}, \psi\big) = 0$ gives

$$\tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}'\tilde{\mathbf{\Theta}}^{-1},$$

and solving $S_2\big(\hat{\mathbf{\Lambda}}^{\text{new}}, (\hat{\psi})^{\text{new}}\big) = 0$ gives

$$\begin{aligned}
p(\hat{\psi})^{\text{new}} &= \operatorname{tr}\{\tilde{\mathbf{S}}\} - 2\operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} + \operatorname{tr}\{(\hat{\mathbf{\Lambda}}^{\text{new}})'\hat{\mathbf{\Lambda}}^{\text{new}}\tilde{\mathbf{\Theta}}\} \\
&= \operatorname{tr}\{\tilde{\mathbf{S}}\} - 2\operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} + \operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\tilde{\mathbf{\Theta}}'(\hat{\mathbf{\Lambda}}^{\text{new}})'\} \\
&= \operatorname{tr}\{\tilde{\mathbf{S}}\} - 2\operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} + \operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\tilde{\mathbf{\Theta}}'(\tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}'\tilde{\mathbf{\Theta}}^{-1})'\} \\
&= \operatorname{tr}\{\tilde{\mathbf{S}}\} - 2\operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} + \operatorname{tr}\{\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\},
\end{aligned}$$

therefore,

$$(\hat{\psi})^{\text{new}} = \frac{1}{p}\operatorname{tr}\{\tilde{\mathbf{S}} - \hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\}.$$

A.4.2 Model CCU

For Model CCU; $\mathbf{\Lambda}_g = \mathbf{\Lambda}$ and $\mathbf{\Psi}_g = \mathbf{\Psi}$. Therefore, $\hat{\boldsymbol{\beta}} = \hat{\mathbf{\Lambda}}'(\hat{\mathbf{\Lambda}}\hat{\mathbf{\Lambda}}' + \hat{\mathbf{\Psi}})^{-1}$ and Q can be written as

$$Q(\mathbf{\Lambda}, \psi) = C + \frac{n}{2}\Big[\log|\mathbf{\Psi}^{-1}| - \operatorname{tr}\{\mathbf{\Psi}^{-1}\tilde{\mathbf{S}}\} + 2\operatorname{tr}\{\mathbf{\Psi}^{-1}\mathbf{\Lambda}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} - \operatorname{tr}\{\mathbf{\Lambda}'\mathbf{\Psi}^{-1}\mathbf{\Lambda}\tilde{\mathbf{\Theta}}\}\Big].$$

Differentiating Q with respect to $\mathbf{\Lambda}$ and $\mathbf{\Psi}^{-1}$ respectively gives the score functions below.

$$S_1(\mathbf{\Lambda}, \mathbf{\Psi}) = \frac{\partial Q(\mathbf{\Lambda}, \mathbf{\Psi})}{\partial \mathbf{\Lambda}} = n\Big[\mathbf{\Psi}^{-1}\tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}' - \mathbf{\Psi}^{-1}\mathbf{\Lambda}\tilde{\mathbf{\Theta}}\Big].$$

$$S_2(\mathbf{\Lambda}, \mathbf{\Psi}) = \frac{\partial Q(\mathbf{\Lambda}, \mathbf{\Psi})}{\partial \mathbf{\Psi}^{-1}} = \frac{n}{2}\Big[\mathbf{\Psi} - \tilde{\mathbf{S}}' + 2\mathbf{\Lambda}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}} - \mathbf{\Lambda}\tilde{\mathbf{\Theta}}'\mathbf{\Lambda}'\Big].$$

Now, solving $S_1(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}})$ gives

$$\hat{\boldsymbol{\Lambda}}^{\text{new}} = \tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}'\tilde{\boldsymbol{\Theta}}^{-1},$$

and solving $\text{diag}\{S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}}^{\text{new}})\} = 0$ gives

$$\begin{aligned}
\hat{\boldsymbol{\Psi}}^{\text{new}} &= \text{diag}\{\tilde{\mathbf{S}} - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}} + \hat{\boldsymbol{\Lambda}}^{\text{new}}\tilde{\boldsymbol{\Theta}}'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\} \\
&= \text{diag}\{\tilde{\mathbf{S}} - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}} + \hat{\boldsymbol{\Lambda}}^{\text{new}}\tilde{\boldsymbol{\Theta}}'(\tilde{\mathbf{S}}\hat{\boldsymbol{\beta}}'\tilde{\boldsymbol{\Theta}}^{-1})'\} \\
&= \text{diag}\{\tilde{\mathbf{S}} - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}} + \hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} \\
&= \text{diag}\{\tilde{\mathbf{S}} - \hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\}.
\end{aligned}$$

A.4.3 Model CUC

For Model CUC; $\boldsymbol{\Lambda}_g = \boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}_g = \psi_g\mathbf{I}_p$, so that $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\psi}_g\mathbf{I}_p)^{-1}$. Therefore, Q can be written as

$$Q(\boldsymbol{\Lambda}, \psi_g) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \left[p\log\psi_g^{-1} - \psi_g^{-1}\,\text{tr}\{\mathbf{S}_g\} + 2\psi_g^{-1}\,\text{tr}\{\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \psi_g^{-1}\,\text{tr}\{\boldsymbol{\Lambda}'\boldsymbol{\Lambda}\boldsymbol{\Theta}_g\}\right].$$

Differentiating Q with respect to $\boldsymbol{\Lambda}$ and ψ_g^{-1} respectively gives the score functions below.

$$\begin{aligned}
S_1(\boldsymbol{\Lambda}, \psi_g) &= \frac{\partial Q(\boldsymbol{\Lambda}, \psi)}{\partial \boldsymbol{\Lambda}} = \sum_{g=1}^{G} n_g \left[\frac{1}{\psi_g}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \frac{1}{\psi_g}\hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g\right]. \\
S_2(\boldsymbol{\Lambda}, \psi_g) &= \frac{\partial Q(\boldsymbol{\Lambda}, \psi)}{\partial \psi_g^{-1}} = \frac{n_g}{2}\left[p\psi_g - \,\text{tr}\{\mathbf{S}_g\} + 2\,\text{tr}\{\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \,\text{tr}\{\boldsymbol{\Lambda}'\boldsymbol{\Lambda}\boldsymbol{\Theta}_g\}\right].
\end{aligned}$$

Now, solving $S_1(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\psi}_g) = 0$ gives

$$\hat{\boldsymbol{\Lambda}}^{\text{new}}\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\boldsymbol{\Theta}_g = \sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g',$$

and so

$$\hat{\boldsymbol{\Lambda}}^{\text{new}} = \left[\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\right]\left[\sum_{g=1}^{G}\frac{n_g}{\hat{\psi}_g}\boldsymbol{\Theta}_g\right]^{-1}.$$

Solving $S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, (\hat{\psi}_g)^{\text{new}}) = 0$ gives

$$(\hat{\psi}_g)^{\text{new}} = \frac{1}{p}\,\text{tr}\{\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\}.$$

A.4.4 Model CUU

For Model CUU; $\boldsymbol{\Lambda}_g = \boldsymbol{\Lambda}$ and so $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}'(\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{\Lambda}}' + \hat{\boldsymbol{\Psi}}_g)^{-1}$. Therefore, Q can be written as

$$Q(\boldsymbol{\Lambda}, \boldsymbol{\Psi}_g) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \left[\log|\boldsymbol{\Psi}_g^{-1}| - \operatorname{tr}\{\boldsymbol{\Psi}_g^{-1}\tilde{\mathbf{S}}\} + 2\operatorname{tr}\{\boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}\tilde{\mathbf{S}}\} - \operatorname{tr}\{\boldsymbol{\Lambda}'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}\tilde{\boldsymbol{\Theta}}\}\right].$$

Differentiating Q with respect to $\boldsymbol{\Lambda}$ and $\boldsymbol{\Psi}_g^{-1}$ respectively gives the score functions below.

$$S_1(\boldsymbol{\Lambda}, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Lambda}} = \sum_{g=1}^{G} n_g \left[\boldsymbol{\Psi}_g^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \boldsymbol{\Psi}_g^{-1}\boldsymbol{\Lambda}\boldsymbol{\Theta}_g\right].$$

$$S_2(\boldsymbol{\Lambda}, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Psi}_g^{-1}} = \frac{n_g}{2}\left[\boldsymbol{\Psi}_g - \mathbf{S}_g + 2\boldsymbol{\Lambda}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \boldsymbol{\Lambda}\boldsymbol{\Theta}_g'\boldsymbol{\Lambda}'\right].$$

Now, solving $S_1(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}}_g) = 0$ gives

$$\sum_{g=1}^{G} n_g\hat{\boldsymbol{\Psi}}_g^{-1}\hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g = \sum_{g=1}^{G} n_g\hat{\boldsymbol{\Psi}}_g^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g', \tag{A.1}$$

which must be solved for $\hat{\boldsymbol{\Lambda}}^{\text{new}}$ in a row-by-row manner. Write

$$\boldsymbol{\lambda}_i = \begin{pmatrix} \lambda_{i1} & \lambda_{i2} & \cdots & \lambda_{iq} \end{pmatrix},$$

to represent the ith row of the matrix $\boldsymbol{\Lambda}$ and let $\mathbf{r}_i$ represent the ith row of the matrix on the right-hand-side of Equation A.1. Now, row i of Equation A.1 can be written

$$\hat{\boldsymbol{\lambda}}_i^{\text{new}} \sum_{g=1}^{G} \frac{n_g}{\hat{\psi}_{g(i)}}\boldsymbol{\Theta}_g = \mathbf{r}_i,$$

where $\hat{\psi}_{g(i)}$ is the ith entry along the diangonal of $\boldsymbol{\Psi}_g$. Therefore,

$$\hat{\boldsymbol{\lambda}}_i^{\text{new}} = \mathbf{r}_i \left(\sum_{g=1}^{G} \frac{n_g}{\hat{\psi}_{g(i)}}\boldsymbol{\Theta}_g\right)^{-1},$$

for $i = 1, 2, \ldots, p$. Unfortunately, the row-by-row nature of this solution slows the fitting of the CUU model to a great degree.

Solving $\operatorname{diag}\{S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, \hat{\boldsymbol{\Psi}}_g^{\text{new}})\} = 0$ gives

$$\hat{\boldsymbol{\Psi}}_g^{\text{new}} = \operatorname{diag}\{\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}}\boldsymbol{\Theta}_g(\hat{\boldsymbol{\Lambda}}^{\text{new}})'\}.$$

A.4.5 Model UCC

For Model UCC; $\boldsymbol{\Psi}_g = \psi\mathbf{I}_p$ and so $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\psi}\mathbf{I}_p)^{-1}$. Therefore, Q can be written as

$$Q(\boldsymbol{\Lambda}_g, \psi) = C + \frac{1}{2}\sum_{g=1}^{G} n_g \left[p\log\psi^{-1} - \psi^{-1}\operatorname{tr}\{\mathbf{S}_g\} + 2\psi^{-1}\operatorname{tr}\{\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \psi^{-1}\operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\right].$$

Differentiating Q with respect to $\boldsymbol{\Lambda}_g$ and ψ^{-1} respectively gives the score functions below.

$$S_1(\boldsymbol{\Lambda}_g, \psi) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \psi)}{\partial \boldsymbol{\Lambda}_g} = \frac{n_g}{\psi}\left[\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \boldsymbol{\Lambda}\boldsymbol{\Theta}_g\right].$$

$$S_2(\boldsymbol{\Lambda}_g, \psi) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \psi)}{\partial \psi^{-1}} = \frac{1}{2}\sum_{g=1}^{G} n_g \left[p\psi - \operatorname{tr}\{\mathbf{S}_g\} + 2\operatorname{tr}\{\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \operatorname{tr}\{\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g'\boldsymbol{\Lambda}_g'\}\right].$$

Now, solving $S_1(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\psi}) = 0$ we obtain

$$\hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},$$

in the familiar way. Solving $S_2(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, (\hat{\psi})^{\text{new}}) = 0$ gives

$$p(\hat{\psi})^{\text{new}}\sum_{g=1}^{G} n_g = \sum_{g=1}^{G} n_g\left[\operatorname{tr}\{\mathbf{S}_g\} - 2\operatorname{tr}\{\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} + \operatorname{tr}\{\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}_g^{\text{new}})'\}\right],$$

and so

$$p(\hat{\psi})^{\text{new}} = \sum_{g=1}^{G}\frac{n_g}{n}\left[\operatorname{tr}\{\mathbf{S}_g\} - \operatorname{tr}\{\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}\right].$$

Therefore,

$$(\hat{\psi})^{\text{new}} = \frac{1}{p}\sum_{g=1}^{G}\hat{\pi}_g\operatorname{tr}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}.$$

A.4.6 Model UCU

For Model UCU, $\boldsymbol{\Psi}_g = \boldsymbol{\Psi}$, so $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\boldsymbol{\Psi}})^{-1}$ and so Q can be written as

$$Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}) = C + \frac{1}{2}\sum_{g=1}^{G} n_g\left[\log|\boldsymbol{\Psi}^{-1}| - \operatorname{tr}\{\boldsymbol{\Psi}^{-1}\mathbf{S}_g\} + 2\operatorname{tr}\{\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\right].$$

Differentiating Q with respect to $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}^{-1}$ respectively gives the score functions below.

$$S_1(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi})}{\partial \boldsymbol{\Lambda}_g} = \frac{n_g}{2}\left[\boldsymbol{\Psi}^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \boldsymbol{\Psi}^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\right].$$

$$S_2(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi})}{\partial \boldsymbol{\Psi}^{-1}} = \frac{1}{2}\sum_{g=1}^{G} n_g\left[\boldsymbol{\Psi} - \mathbf{S}_g + 2\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g'\boldsymbol{\Lambda}_g'\right].$$

Now, solving $S_1\big(\hat{\mathbf{\Lambda}}_g^{\text{new}}, \hat{\mathbf{\Psi}}\big) = 0$ gives the familiar result

$$\hat{\mathbf{\Lambda}}_g^{\text{new}} = \mathbf{S}_g \hat{\boldsymbol{\beta}}_g' \mathbf{\Theta}_g^{-1},$$

and solving $\operatorname{diag}\big\{S_2\big(\hat{\mathbf{\Lambda}}_g^{\text{new}}, \hat{\mathbf{\Psi}}^{\text{new}}\big)\big\} = 0$ gives

$$\hat{\mathbf{\Psi}}^{\text{new}} \sum_{g=1}^{G} n_g = \sum_{g=1}^{G} n_g \operatorname{diag}\{\mathbf{S}_g - 2\hat{\mathbf{\Lambda}}_g^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g + \hat{\mathbf{\Lambda}}_g^{\text{new}} \mathbf{\Theta}_g' (\hat{\mathbf{\Lambda}}_g^{\text{new}})'\},$$

and therefore,

$$\hat{\mathbf{\Psi}}^{\text{new}} = \sum_{g=1}^{G} \hat{\pi}_g \operatorname{diag}\{\mathbf{S}_g - \hat{\mathbf{\Lambda}}_g^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\}.$$

A.4.7 Model UUC

For Model UUC, $\mathbf{\Psi}_g = \psi_g \mathbf{I}_p$, so that $\hat{\boldsymbol{\beta}}_g = \hat{\mathbf{\Lambda}}_g'(\hat{\mathbf{\Lambda}}_g \hat{\mathbf{\Lambda}}_g' + \hat{\psi}_g \mathbf{I}_p)^{-1}$. Therefore, Q can be written as

$$Q(\mathbf{\Lambda}_g, \psi_g) = C + \frac{1}{2} \sum_{g=1}^{G} n_g \left[p \log \psi_g^{-1} - \psi_g^{-1} \operatorname{tr}\{\mathbf{S}_g\} + 2\psi_g^{-1} \operatorname{tr}\{\mathbf{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\} - \psi_g^{-1} \operatorname{tr}\{\mathbf{\Lambda}_g' \mathbf{\Lambda}_g \mathbf{\Theta}_g\} \right].$$

Differentiating Q with respect to $\mathbf{\Lambda}_g$ and ψ_g^{-1} respectively gives the score functions below.

$$S_1(\mathbf{\Lambda}_g, \psi_g) = \frac{\partial Q(\mathbf{\Lambda}_g, \psi_g)}{\partial \mathbf{\Lambda}_g} = \frac{n_g}{2} \left[\psi_g^{-1} \mathbf{S}_g \hat{\boldsymbol{\beta}}_g' - \psi_g^{-1} \mathbf{\Lambda}_g \mathbf{\Theta}_g \right].$$

$$S_2(\mathbf{\Lambda}_g, \psi_g) = \frac{\partial Q(\mathbf{\Lambda}_g, \psi)}{\partial \psi_g^{-1}} = \frac{n_g}{2} \left[p\psi_g - \operatorname{tr}\{\mathbf{S}_g\} + 2 \operatorname{tr}\{\mathbf{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\} - \operatorname{tr}\{\mathbf{\Lambda}_g \mathbf{\Theta}_g' \mathbf{\Lambda}_g'\} \right].$$

Now, solving $S_1\big(\hat{\mathbf{\Lambda}}_g^{\text{new}}, \hat{\psi}_g\big) = 0$ gives

$$\hat{\mathbf{\Lambda}}_g^{\text{new}} = \mathbf{S}_g \hat{\boldsymbol{\beta}}_g' \mathbf{\Theta}_g^{-1},$$

and solving $S_2\big(\hat{\mathbf{\Lambda}}_g^{\text{new}}, (\hat{\psi}_g)^{\text{new}}\big) = 0$, in the familiar fashion, leads to the result

$$(\hat{\psi}_g)^{\text{new}} = \frac{1}{p} \operatorname{tr}\{\mathbf{S}_g - \hat{\mathbf{\Lambda}}_g^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\}.$$

A.4.8 Model UUU

No constrainits are imposed upon $\mathbf{\Lambda}_g$ or $\mathbf{\Psi}_g$ for the UUU model, therefore,

$$Q(\mathbf{\Lambda}_g, \mathbf{\Psi}_g) = C + \frac{1}{2} \sum_{g=1}^{G} n_g \left[\log |\mathbf{\Psi}_g^{-1}| - \operatorname{tr}\{\mathbf{\Psi}_g^{-1} \mathbf{S}_g\} + 2 \operatorname{tr}\{\mathbf{\Psi}_g^{-1} \mathbf{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\} - \operatorname{tr}\{\mathbf{\Lambda}_g' \mathbf{\Psi}_g^{-1} \mathbf{\Lambda}_g \mathbf{\Theta}_g\} \right].$$

Differentiating Q with respect to $\boldsymbol{\Lambda}_g$ and $\boldsymbol{\Psi}_g^{-1}$ respectively gives the score functions below.

$$S_1(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Lambda}_g} = \frac{n_g}{2}\left[\hat{\boldsymbol{\Psi}}_g^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \hat{\boldsymbol{\Psi}}_g^{-1}\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\boldsymbol{\Theta}_g\right].$$

$$S_2(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g) = \frac{\partial Q(\boldsymbol{\Lambda}_g, \boldsymbol{\Psi}_g)}{\partial \boldsymbol{\Psi}_g^{-1}} = \frac{n_g}{2}\left[\hat{\boldsymbol{\Psi}}_g^{\text{new}}\mathbf{I}_p - \mathbf{S}_g + 2\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}_g^{\text{new}})'\right].$$

Solving $S_1\big(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\boldsymbol{\Psi}}_g\big) = 0$ gives

$$\hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g'\boldsymbol{\Theta}_g^{-1},$$

and solving $\operatorname{diag}\big\{S_2\big(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\boldsymbol{\Psi}}_g^{\text{new}}\big)\big\} = 0$ gives

$$\begin{aligned}\hat{\boldsymbol{\Psi}}_g^{\text{new}} &= \operatorname{diag}\big\{\mathbf{S}_g - 2\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\boldsymbol{\Theta}_g'(\hat{\boldsymbol{\Lambda}}_g^{\text{new}})'\big\} \\ &= \operatorname{diag}\big\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\big\}.\end{aligned}$$

Appendix B

Calculations for GPGMMs

B.1 Important Results

In addition to the results mentioned in Section A.1, the following result from Lütkepohl (1996) and Magnus & Neudecker (1999) is used in this section;

$$\mathbf{X}_{n\times n} \text{ (nonsingular)} : \frac{\partial|\mathbf{X}^{-1}|}{\partial\mathbf{X}} = -|\mathbf{X}|^{-1}(\mathbf{X}')^{-1}.$$

From which it follows that

$$\frac{\partial|\mathbf{X}|}{\partial\mathbf{X}^{-1}} = -|\mathbf{X}|\mathbf{X}'.$$

B.2 Q Equation

Recall Equation 4.1;

$$Q(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}_g) = C + \frac{1}{2}\sum_{g=1}^{G} n_g\Big[p\log\omega_g^{-1} + \log|\boldsymbol{\Delta}_g^{-1}| - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}_g^{-1}\mathbf{S}_g\} + 2\omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}_g^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Delta}_g^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\Big],$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\boldsymbol{\Lambda}_g$, ω_g and $\boldsymbol{\Delta}_g$, $\boldsymbol{\Theta}_g = \mathbf{I}_q - \hat{\boldsymbol{\beta}}_g\hat{\boldsymbol{\Lambda}}_g + \hat{\boldsymbol{\beta}}_g\mathbf{S}_g\hat{\boldsymbol{\beta}}_g'$ is a symmetric $q \times q$ matrix and $|\boldsymbol{\Delta}_g| = 1$. Now, maximising Q with respect to $\boldsymbol{\Lambda}_g$, $\boldsymbol{\Delta}_g$ and ω_g gives maximum likelihood estimates of these parameters for each of the the models listed in Table 4.1.

B.3 Constrained Maximum Likelihood Estimates

B.3.1 Model UCUU

For Model UCUU, $\boldsymbol{\Delta}_g = \boldsymbol{\Delta}$ and so $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g'(\hat{\boldsymbol{\Lambda}}_g\hat{\boldsymbol{\Lambda}}_g' + \hat{\omega}_g\boldsymbol{\Delta})^{-1}$ and Q can be written

$$Q(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}) = C + \frac{1}{2}\sum_{g=1}^{G} n_g\Big[p\log\omega_g^{-1} + \log|\boldsymbol{\Delta}^{-1}| - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}^{-1}\mathbf{S}_g\} + 2\omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \omega_g^{-1}\operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\Big].$$

Now, form $L(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}, \lambda) = Q(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}) - \lambda(|\boldsymbol{\Delta}| - 1)$ and differentiating Q with respect to $\boldsymbol{\Lambda}_g$, ω_g^{-1}, $\boldsymbol{\Delta}^{-1}$ and λ respectively gives the score functions below.

$$S_1(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \boldsymbol{\Lambda}} = \sum_{g=1}^{G}\frac{n_g}{\omega_g}\left[\boldsymbol{\Delta}^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}_g\tilde{\boldsymbol{\Theta}}_g\right],$$

$$S_2(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \omega_g^{-1}} = \frac{n_g}{2}\left[p\omega_g - \operatorname{tr}\{\boldsymbol{\Delta}^{-1}\mathbf{S}_g\} + 2\operatorname{tr}\{\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \operatorname{tr}\{\boldsymbol{\Lambda}_g'\boldsymbol{\Delta}^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g\}\right],$$

$$S_3(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \boldsymbol{\Delta}^{-1}} = \frac{1}{2}\sum_{g=1}^{G} n_g\left[\boldsymbol{\Delta} - \omega_g^{-1}\mathbf{S}_g' + 2\omega_g^{-1}\boldsymbol{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \omega_g^{-1}\boldsymbol{\Lambda}_g\boldsymbol{\Theta}_g'\boldsymbol{\Lambda}_g'\right] + \lambda|\boldsymbol{\Delta}|\boldsymbol{\Delta},$$

$$S_4(\boldsymbol{\Lambda}_g, \omega_g, \boldsymbol{\Delta}, \lambda) = \frac{\partial L}{\partial \lambda} = |\boldsymbol{\Delta}| - 1.$$

Solving $S_1(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\omega}_g, \hat{\boldsymbol{\Delta}}, \lambda) = 0$, we obtain

$$\hat{\boldsymbol{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g\boldsymbol{\Theta}_g^{-1}.$$

Now, setting $S_2(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\omega}_g^{\text{new}}, \hat{\boldsymbol{\Delta}}, \lambda) = 0$ we have

$$(\hat{\omega}_g)^{\text{new}} = \frac{1}{p}\operatorname{tr}\{\hat{\boldsymbol{\Delta}}^{-1}\mathbf{S}_g - \hat{\boldsymbol{\Delta}}^{-1}\hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}$$

and solving $\operatorname{diag}\{S_3(\hat{\boldsymbol{\Lambda}}_g^{\text{new}}, \hat{\omega}_g^{\text{new}}, \hat{\boldsymbol{\Delta}}^{\text{new}}, \lambda)\} = 0$ gives

$$\begin{aligned}\hat{\boldsymbol{\Delta}}^{\text{new}} &= \frac{1}{n + 2\lambda|\hat{\boldsymbol{\Delta}}^{\text{new}}|}\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\operatorname{diag}\{\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} \\ &= \frac{1}{n + 2\lambda}\operatorname{diag}\left\{\sum_{g=1}^{G}\frac{n_g}{(\hat{\omega}_g)^{\text{new}}}\left[\mathbf{S}_g - \hat{\boldsymbol{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\right]\right\}.\end{aligned}$$

But $\hat{\boldsymbol{\Delta}}^{\text{new}}$ is a diagonal matrix with $|\hat{\boldsymbol{\Delta}}^{\text{new}}| = 1$, therefore

$$n + 2\lambda = \left(\prod_{i=1}^{p}\xi_j\right)^{\frac{1}{p}},$$

where ξ_j is the jth element along the diagonal of the matrix

$$\sum_{g=1}^{G} \frac{n_g}{(\hat{\omega}_g)^{\text{new}}} \left[\mathbf{S}_g - \hat{\mathbf{\Lambda}}_g^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g \right].$$

It follows that

$$\lambda = \frac{1}{2} \left[\left(\prod_{j=1}^{p} \xi_j \right)^{\frac{1}{p}} - n \right].$$

B.3.2 Model CUCU

For Model CUCU, $\mathbf{\Lambda}_g = \mathbf{\Lambda}$ and $\omega_g = \omega$. Therefore, $\hat{\boldsymbol{\beta}}_g = \hat{\mathbf{\Lambda}}'(\hat{\mathbf{\Lambda}}\hat{\mathbf{\Lambda}}' + \hat{\omega}\hat{\mathbf{\Delta}}_g)^{-1}$ and Q can be written

$$Q(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g) = C + \frac{1}{2} \sum_{g=1}^{G} n_g \Big[p \log \omega^{-1} + \log |\mathbf{\Delta}_g^{-1}| - \omega^{-1} \operatorname{tr}\{\mathbf{\Delta}_g^{-1} \mathbf{S}_g\} + 2\omega^{-1} \operatorname{tr}\{\mathbf{\Delta}_g^{-1} \mathbf{\Lambda} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\}$$
$$- \omega^{-1} \operatorname{tr}\{\mathbf{\Lambda}' \mathbf{\Delta}_g^{-1} \mathbf{\Lambda} \mathbf{\Theta}_g\} \Big].$$

Now, we form

$$L(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g, \lambda) = Q(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g) - \sum_{g=1}^{G} \lambda_g (|\mathbf{\Delta}_g| - 1).$$

Differentiating Q with respect to $\mathbf{\Lambda}$, ω^{-1}, $\mathbf{\Delta}_g^{-1}$ and λ respectively gives the score functions below.

$$S_1(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \mathbf{\Lambda}} = \sum_{g=1}^{G} \frac{n_g}{\omega} \left[\mathbf{\Delta}_g^{-1} \mathbf{S}_g \hat{\boldsymbol{\beta}}_g' - \mathbf{\Delta}_g^{-1} \mathbf{\Lambda} \tilde{\mathbf{\Theta}}_g \right],$$

$$S_2(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \omega^{-1}}$$
$$= \frac{1}{2} \sum_{g=1}^{G} n_g \left[p\omega - \operatorname{tr}\{\mathbf{\Delta}_g^{-1} \mathbf{S}_g\} + 2 \operatorname{tr}\{\mathbf{\Delta}_g^{-1} \mathbf{\Lambda} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\} - \operatorname{tr}\{\mathbf{\Delta}_g^{-1} \mathbf{\Lambda} \mathbf{\Theta}_g \mathbf{\Lambda}'\} \right],$$

$$S_3(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \mathbf{\Delta}_g^{-1}} = \frac{n_g}{2} \left[\mathbf{\Delta}_g - \omega^{-1} \mathbf{S}_g' + 2\omega^{-1} \mathbf{\Lambda} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g - \omega^{-1} \mathbf{\Lambda} \mathbf{\Theta}_g' \mathbf{\Lambda}' \right] + \lambda_g |\mathbf{\Delta}_g| \mathbf{\Delta}_g,$$

$$S_4(\mathbf{\Lambda}, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \lambda_g} = |\mathbf{\Delta}_g| - 1.$$

Unfortunately, we must solve $S_1(\hat{\mathbf{\Lambda}}^{\text{new}}, \hat{\omega}, \hat{\mathbf{\Delta}}_g, \lambda_g) = 0$ in a row by row manner, like that of the CUU solution in Appendix A.4.4, so that

$$\hat{\boldsymbol{\lambda}}_i^{\text{new}} = \mathbf{r}_i \left(\sum_{g=1}^{G} \frac{n_g}{\hat{\delta}_{g(i)}} \mathbf{\Theta}_g \right)^{-1},$$

where $\mathbf{r}_i$ is the ith row of the matrix

$$\sum_{g=1}^{G} \frac{n_g}{\hat{\delta}_{g(j)}} \mathbf{S}_g \hat{\boldsymbol{\beta}}_g'$$

and $\hat{\delta}_{g(i)}$ is the ith element along the diagonal of the matrix $\hat{\boldsymbol{\Delta}}_g$. Solving $S_2(\hat{\boldsymbol{\Lambda}}^{\text{new}}, (\hat{\omega})^{\text{new}}, \hat{\boldsymbol{\Delta}}_g, \lambda_g) = 0$ gives

$$(\hat{\omega})^{\text{new}} = \frac{1}{p} \sum_{g=1}^{G} \hat{\pi}_g \operatorname{tr}\{\hat{\boldsymbol{\Delta}}_g^{-1} \mathbf{S}_g - 2\hat{\boldsymbol{\Delta}}_g^{-1} \hat{\boldsymbol{\Lambda}}^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g - \hat{\boldsymbol{\Delta}}_g^{-1} \hat{\boldsymbol{\Lambda}}^{\text{new}} \boldsymbol{\Theta}_g (\hat{\boldsymbol{\Lambda}}^{\text{new}})'\}.$$

Solving $\operatorname{diag}\{S_3(\hat{\boldsymbol{\Lambda}}^{\text{new}}, (\hat{\omega})^{\text{new}}, \hat{\boldsymbol{\Delta}}_g^{\text{new}}, \lambda_g)\} = 0$ gives

$$(\hat{\omega}_g)^{\text{new}} = \frac{1}{\hat{\omega}^{\text{new}} \left(1 + \frac{2\lambda_g}{n_g}\right)} \operatorname{diag}\{\mathbf{S}_g' - 2\hat{\boldsymbol{\Lambda}}^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}} \boldsymbol{\Theta}_g' (\hat{\boldsymbol{\Lambda}}^{\text{new}})'\}. \tag{B.1}$$

But $\hat{\boldsymbol{\Delta}}_g^{\text{new}}$ is a diagonal matrix with $|\hat{\boldsymbol{\Delta}}_g^{\text{new}}| = 1$, therefore

$$(\hat{\omega}_g)^{\text{new}} \left(1 + \frac{2\lambda_g}{n_g}\right) = \left(\prod_{j=1}^{p} \xi_{gj}\right)^{\frac{1}{p}},$$

where ξ_j is the jth element along the diagonal of the matrix

$$\mathbf{S}_g' - 2\hat{\boldsymbol{\Lambda}}^{\text{new}} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{\text{new}} \boldsymbol{\Theta}_g' (\hat{\boldsymbol{\Lambda}}^{\text{new}})'.$$

It follows that

$$\lambda_g = \frac{n_g}{2} \left[\frac{1}{(\hat{\omega}_g)^{\text{new}}} \left(\prod_{j=1}^{p} \xi_{gj}\right)^{\frac{1}{p}} - 1 \right].$$

B.3.3 Model UUCU

For Model UUCU, $\omega_g = \omega$. Therefore, $\hat{\boldsymbol{\beta}}_g = \hat{\boldsymbol{\Lambda}}_g' (\hat{\boldsymbol{\Lambda}}_g \hat{\boldsymbol{\Lambda}}_g' + \hat{\omega} \hat{\boldsymbol{\Delta}}_g)^{-1}$ and Q can be written

$$Q(\boldsymbol{\Lambda}_g, \omega, \boldsymbol{\Delta}_g) = C + \frac{1}{2} \sum_{g=1}^{G} n_g \Big[p \log \omega^{-1} + \log |\boldsymbol{\Delta}_g^{-1}| - \omega^{-1} \operatorname{tr}\{\boldsymbol{\Delta}_g^{-1} \mathbf{S}_g\} + 2\omega^{-1} \operatorname{tr}\{\boldsymbol{\Delta}_g^{-1} \boldsymbol{\Lambda}_g \hat{\boldsymbol{\beta}}_g \mathbf{S}_g\}$$
$$- \omega^{-1} \operatorname{tr}\{\boldsymbol{\Lambda}_g' \boldsymbol{\Delta}_g^{-1} \boldsymbol{\Lambda}_g \boldsymbol{\Theta}_g\} \Big]$$

and we form

$$L(\boldsymbol{\Lambda}_g, \omega, \boldsymbol{\Delta}_g, \lambda) = Q(\boldsymbol{\Lambda}_g, \omega, \boldsymbol{\Delta}_g) - \sum_{g=1}^{G} \lambda_g (|\boldsymbol{\Delta}_g| - 1).$$

Differentiating Q with respect to $\mathbf{\Lambda}_g$, ω^{-1}, $\mathbf{\Delta}_g^{-1}$ and λ_g respectively gives the score functions below.

$$S_1(\mathbf{\Lambda}_g, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \mathbf{\Lambda}_g} = \frac{n_g}{\omega}\left[\mathbf{\Delta}_g^{-1}\mathbf{S}_g\hat{\boldsymbol{\beta}}_g' - \mathbf{\Delta}_g^{-1}\mathbf{\Lambda}_g\tilde{\mathbf{\Theta}}_g\right],$$

$$\begin{aligned} S_2(\mathbf{\Lambda}_g, \omega, \mathbf{\Delta}_g, \lambda_g) &= \frac{\partial L}{\partial \omega^{-1}} \\ &= \frac{1}{2}\sum_{g=1}^{G} n_g\left[p\omega - \operatorname{tr}\{\mathbf{\Delta}_g^{-1}\mathbf{S}_g\} + 2\operatorname{tr}\{\mathbf{\Delta}_g^{-1}\mathbf{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\} - \operatorname{tr}\{\mathbf{\Lambda}_g'\mathbf{\Delta}_g^{-1}\mathbf{\Lambda}_g\mathbf{\Theta}_g\}\right], \end{aligned}$$

$$S_3(\mathbf{\Lambda}_g, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \mathbf{\Delta}_g^{-1}} = \frac{n_g}{2}\left[\mathbf{\Delta}_g - \omega^{-1}\mathbf{S}_g' + 2\omega^{-1}\mathbf{\Lambda}_g\hat{\boldsymbol{\beta}}_g\mathbf{S}_g - \omega^{-1}\mathbf{\Lambda}_g\mathbf{\Theta}_g'\mathbf{\Lambda}_g'\right] + \lambda_g|\mathbf{\Delta}_g|\mathbf{\Delta}_g,$$

$$S_4(\mathbf{\Lambda}_g, \omega, \mathbf{\Delta}_g, \lambda_g) = \frac{\partial L}{\partial \lambda_g} = |\mathbf{\Delta}_g| - 1.$$

Solving $S_1(\hat{\mathbf{\Lambda}}_g^{\text{new}}, \hat{\omega}, \hat{\mathbf{\Delta}}_g, \lambda_g) = 0$ gives

$$\hat{\mathbf{\Lambda}}_g^{\text{new}} = \mathbf{S}_g\hat{\boldsymbol{\beta}}_g\mathbf{\Theta}_g^{-1},$$

and solving $S_2(\hat{\mathbf{\Lambda}}_g^{\text{new}}, (\hat{\omega})^{\text{new}}, \hat{\mathbf{\Delta}}_g, \lambda_g) = 0$ gives

$$(\hat{\omega})^{\text{new}} = \frac{1}{p}\sum_{g=1}^{G}\hat{\pi}_g \operatorname{tr}\{\hat{\mathbf{\Delta}}_g^{-1}(\mathbf{S}_g - \hat{\mathbf{\Lambda}}_g^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g)\}.$$

Solving $\operatorname{diag}\{S_3(\hat{\mathbf{\Lambda}}_g^{\text{new}}, (\hat{\omega})^{\text{new}}, \hat{\mathbf{\Delta}}_g^{\text{new}}, \lambda_g)\} = 0$ gives

$$\begin{aligned} \hat{\mathbf{\Delta}}_g^{\text{new}} &= \frac{1}{(\hat{\omega}_g)^{\text{new}}\left(1 + \frac{2\lambda_g}{n_g}\right)}\operatorname{diag}\{\mathbf{S}_g' - 2\hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g + \hat{\mathbf{\Lambda}}^{\text{new}}\mathbf{\Theta}_g'(\hat{\mathbf{\Lambda}}^{\text{new}})'\} \\ &= \frac{1}{(\hat{\omega}_g)^{\text{new}}\left(1 + \frac{2\lambda_g}{n_g}\right)}\operatorname{diag}\{\mathbf{S}_g' - \hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g\}. \end{aligned}$$

But $\hat{\mathbf{\Delta}}_g^{\text{new}}$ is a diagonal matrix with $|\hat{\mathbf{\Delta}}_g^{\text{new}}| = 1$, therefore

$$(\hat{\omega}_g)^{\text{new}}\left(1 + \frac{2\lambda_g}{n_g}\right) = \left(\prod_{j=1}^{p}\xi_{gj}\right)^{\frac{1}{p}},$$

where ξ_{gj} is the jth element along the diagonal of the matrix

$$\mathbf{S}_g' - \hat{\mathbf{\Lambda}}^{\text{new}}\hat{\boldsymbol{\beta}}_g\mathbf{S}_g.$$

It follows that

$$\lambda_g = \frac{n_g}{2}\left[\frac{1}{(\hat{\omega}_g)^{\text{new}}}\left(\prod_{j=1}^{p}\xi_{gj}\right)^{\frac{1}{p}} - 1\right].$$

Appendix C

Calculations for CDGMMs

C.1 Q Equation

From Equation 5.3,

$$Q(\mathbf{T}_g, \mathbf{D}_g) = C - \sum_{g=1}^{G} \frac{n_g}{2} \log |\mathbf{D}_g| - \sum_{g=1}^{G} \frac{n_g}{2} \operatorname{tr}\{\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g' \mathbf{D}_g^{-1}\},$$

where C is constant with respect to $\boldsymbol{\mu}_g$, $\mathbf{T}_g$ and $\mathbf{D}_g$. Now, it is necessary to maximise Q with respect to $\mathbf{T}_g$ and $\mathbf{D}_g$ to get maximum likelihood estimates of these parameters for each of the four cases VV, VE, EV and EE. Solutions for these latter three cases are similar to the VV case detailed in Section 5.3.3.

C.2 Maximum Likelihood Estimates

C.2.1 Model VE

For Model VE, $\mathbf{D}_g = \mathbf{D}$ and so Q can be written

$$Q(\mathbf{T}_g, \mathbf{D}) = C - \frac{n}{2} \log |\mathbf{D}| - \sum_{g=1}^{G} \frac{n_g}{2} \operatorname{tr}\left\{\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g' \mathbf{D}^{-1}\right\}. \tag{C.1}$$

Differentiating Equation C.1 with respect to $\mathbf{T}_g$ and $\mathbf{D}_g^{-1}$ respectively gives the following score functions.

$$S_1(\mathbf{T}_g, \mathbf{D}) = \frac{\partial Q(\mathbf{T}_g, \mathbf{D})}{\partial \mathbf{T}_g} = -\frac{n_g}{2}\left[\left(\mathbf{D}^{-1}\right)' \mathbf{T}_g \mathbf{S}_g' + \mathbf{D}^{-1}\mathbf{T}_g \mathbf{S}_g\right] = -n_g \mathbf{D}^{-1}\mathbf{T}_g \mathbf{S}_g.$$

$$S_2(\mathbf{T}_g, \mathbf{D}) = \frac{\partial Q(\mathbf{T}_g, \mathbf{D})}{\partial \mathbf{D}^{-1}} = \frac{n}{2}\mathbf{D} - \sum_{g=1}^{G} \frac{n_g}{2}\left(\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g'\right)' = \frac{n}{2}\mathbf{D}_g - \sum_{g=1}^{G} \frac{n_g}{2}\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g'.$$

Solving $\mathrm{LT}\{S_1(\hat{\mathbf{T}}_g, \mathbf{D})\} = 0$ gives an identical solution to that of Section 5.3.3, with a general $(r-1) \times (r-1)$ system of linear equations given by

$$\begin{pmatrix} \hat{\phi}_{r1}^{(g)} \\ \hat{\phi}_{r2}^{(g)} \\ \vdots \\ \hat{\phi}_{r,r-1}^{(g)} \end{pmatrix} = -\begin{pmatrix} s_{11}^{(g)} & s_{21}^{(g)} & \cdots & s_{r-1,1}^{(g)} \\ s_{12}^{(g)} & s_{22}^{(g)} & \cdots & s_{r-1,2}^{(g)} \\ \vdots & \vdots & \ddots & \vdots \\ s_{1,r-1}^{(g)} & s_{2,r-2}^{(g)} & \cdots & s_{r-1,r-1}^{(g)} \end{pmatrix}^{-1} \begin{pmatrix} s_{r1}^{(g)} \\ s_{r2}^{(g)} \\ \vdots \\ s_{r,r-1}^{(g)} \end{pmatrix},$$

for $r = 2, 3, \ldots, p$, where the $\{\phi_{ij}^{(g)}\}$ are those elements of $\mathbf{T}_g$ which must be estimated. Solving $\mathrm{diag}\{S_2(\hat{\mathbf{T}}_g, \hat{\mathbf{D}})\} = 0$ gives

$$\hat{\mathbf{D}} = \sum_{g=1}^{G} \pi_g \,\mathrm{diag}\{\hat{\mathbf{T}}_g \mathbf{S}_g \hat{\mathbf{T}}_g'\}.$$

C.2.2 Model EV

For Model EV, $\mathbf{T}_g = \mathbf{T}$ and so Q can be written

$$Q(\mathbf{T}, \mathbf{D}_g) = C - \sum_{g=1}^{G} \frac{n_g}{2} \log|\mathbf{D}_g| - \sum_{g=1}^{G} \frac{n_g}{2} \mathrm{tr}\left\{\mathbf{T}\mathbf{S}_g\mathbf{T}'\mathbf{D}_g^{-1}\right\}. \tag{C.2}$$

Differentiating Equation C.2 with respect to $\mathbf{T}_g$ and $\mathbf{D}_g^{-1}$ respectively gives the following score functions.

$$S_1(\mathbf{T}, \mathbf{D}_g) = \frac{\partial Q(\mathbf{T}, \mathbf{D}_g)}{\partial \mathbf{T}} = -\sum_{g=1}^{G} \frac{n_g}{2}\left[\left(\mathbf{D}_g^{-1}\right)' \mathbf{T}\mathbf{S}_g' + \mathbf{D}_g^{-1}\mathbf{T}\mathbf{S}_g\right] = -\sum_{g=1}^{G} n_g \mathbf{D}_g^{-1}\mathbf{T}\mathbf{S}_g.$$

$$S_2(\mathbf{T}, \mathbf{D}_g) = \frac{\partial Q(\mathbf{T}, \mathbf{D}_g)}{\partial \mathbf{D}_g^{-1}} = \frac{n_g}{2}\mathbf{D}_g - \frac{n_g}{2}\left(\mathbf{T}_g \mathbf{S}_g \mathbf{T}_g'\right)' = \frac{n_g}{2}\left(\mathbf{D}_g - \mathbf{T}\mathbf{S}_g\mathbf{T}'\right).$$

To solve $S_1(\hat{\mathbf{T}}, \mathbf{D}_g)$, once again we use ϕ_{ij} to denote those elements of $\mathbf{T}_g$ that are to be estimated. Now, solving $\mathrm{LT}\{S_1(\hat{\mathbf{T}}, \mathbf{D}_g)\} \equiv S_1(\hat{\Phi}, \mathbf{D}_g) = 0$ for $\hat{\Phi}$ leads again to a total of

$p-1$ linear equations. The 1×1 solution is;

$$\sum_{g=1}^{G} \left[\frac{s_{11}^{(g)} \hat{\phi}_{21}}{\hat{d}_{22}^{(g)}} + \frac{s_{21}^{(g)}}{\hat{d}_{22}^{(g)}} \right] = 0,$$

and so

$$\hat{\phi}_{21} = -\frac{\sum_{g=1}^{G} \left[\frac{s_{11}^{(g)}}{\hat{d}_{22}^{(g)}} \right]}{\sum_{g=1}^{G} \left[\frac{s_{21}^{(g)}}{\hat{d}_{22}^{(g)}} \right]}. \tag{C.3}$$

For convenience, we introduce the notation

$$\kappa_m^{ij} = \sum_{g=1}^{G} \left[\frac{s_{ij}^{(g)}}{\hat{d}_{mm}^{(g)}} \right],$$

so that Equation C.3 can be written

$$\hat{\phi}_{21} = \kappa_2^{11} / \kappa_2^{21}.$$

The 2×2 case involves the equations

$$\kappa_3^{11} \hat{\phi}_{31} + \kappa_3^{21} \hat{\phi}_{32} + \kappa_3^{31} = 0$$

$$\kappa_3^{12} \hat{\phi}_{31} + \kappa_3^{22} \hat{\phi}_{32} + \kappa_3^{32} = 0$$

which can be expressed in the form

$$\begin{pmatrix} \kappa_3^{11} & \kappa_3^{21} \\ \kappa_3^{12} & \kappa_3^{22} \end{pmatrix} \begin{pmatrix} \hat{\phi}_{31} \\ \hat{\phi}_{32} \end{pmatrix} = - \begin{pmatrix} \kappa_3^{31} \\ \kappa_3^{32} \end{pmatrix},$$

and so

$$\begin{pmatrix} \hat{\phi}_{31} \\ \hat{\phi}_{32} \end{pmatrix} = - \begin{pmatrix} \kappa_3^{11} & \kappa_3^{21} \\ \kappa_3^{12} & \kappa_3^{22} \end{pmatrix}^{-1} \begin{pmatrix} \kappa_3^{31} \\ \kappa_3^{32} \end{pmatrix}.$$

It follows that the solution to the general $(r-1) \times (r-1)$ system of equations will be given by

$$\begin{pmatrix} \hat{\phi}_{r1} \\ \hat{\phi}_{r2} \\ \vdots \\ \hat{\phi}_{r,r-1}^{(g)} \end{pmatrix} = - \begin{pmatrix} \kappa_r^{11} & \kappa_r^{21} & \cdots & \kappa_r^{r-1,1} \\ \kappa_r^{12} & \kappa_r^{22} & \cdots & \kappa_r^{r-1,2} \\ \vdots & \vdots & \ddots & \vdots \\ \kappa_r^{1,r-1} & \kappa_r^{2,r-2} & \cdots & \kappa_r^{r-1,r-1} \end{pmatrix}^{-1} \begin{pmatrix} \kappa_r^{r1} \\ \kappa_r^{r2} \\ \vdots \\ \kappa_r^{r,r-1} \end{pmatrix}, \tag{C.4}$$

for $r = 2, 3, \ldots, p$. Note that $\kappa_m^{ij} = \kappa_m^{ji}$ and so the $(r-1) \times (r-1)$ matrix in Equation C.4 is symmetric. Now, solving $\operatorname{diag}\{S_2(\hat{\mathbf{T}}_g, \hat{\mathbf{D}}_g)\} = 0$ gives

$$\hat{\mathbf{D}}_g = \pi_g \operatorname{diag}\{\hat{\mathbf{T}} \mathbf{S}_g \hat{\mathbf{T}}'\}.$$

C.2.3 Model EE

For Model VE, $\mathbf{T}_g = \mathbf{T}$ and $\mathbf{D}_g = \mathbf{D}$ and so Q can be written

$$Q(\mathbf{T}, \mathbf{D}) = C - \frac{n}{2} \log |\mathbf{D}| - \frac{n}{2} \operatorname{tr}\{\mathbf{T}\tilde{\mathbf{S}}\mathbf{T}'\mathbf{D}^{-1}\}, \tag{C.5}$$

where $\tilde{\mathbf{S}} = \sum_{g=1}^{G} \pi_g \mathbf{S}_g$. Differentiating Equation C.5 with respect to $\mathbf{T}$ and $\mathbf{D}^{-1}$ respectively gives the following score functions.

$$S_1(\mathbf{T}, \mathbf{D}) = \frac{\partial Q(\mathbf{T}, \mathbf{D})}{\partial \mathbf{T}} = -\frac{n}{2}\left[(\mathbf{D}^{-1})'\mathbf{T}\tilde{\mathbf{S}}' + \mathbf{D}^{-1}\mathbf{T}\tilde{\mathbf{S}}\right] = -n\mathbf{D}^{-1}\mathbf{T}\tilde{\mathbf{S}}.$$
$$S_2(\mathbf{T}, \mathbf{D}) = \frac{\partial Q(\mathbf{T}, \mathbf{D})}{\partial \mathbf{D}^{-1}} = \frac{n}{2}\mathbf{D} - \frac{n}{2}(\mathbf{T}\tilde{\mathbf{S}}\mathbf{T}')' = \frac{n}{2}(\mathbf{D} - \mathbf{T}\tilde{\mathbf{S}}\mathbf{T}').$$

Solving $\mathrm{LT}\{S_1(\hat{\mathbf{T}}, \mathbf{D})\} = 0$ gives a similar solution to that of Section 5.3.3, with a general $(r-1) \times (r-1)$ system of linear equations given by

$$\begin{pmatrix} \hat{\phi}_{r1} \\ \hat{\phi}_{r2} \\ \vdots \\ \hat{\phi}_{r,r-1} \end{pmatrix} = - \begin{pmatrix} \tilde{s}_{11} & \tilde{s}_{21} & \cdots & \tilde{s}_{r-1,1} \\ \tilde{s}_{12} & \tilde{s}_{22} & \cdots & \tilde{s}_{r-1,2} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{s}_{1,r-1} & \tilde{s}_{2,r-2} & \cdots & \tilde{s}_{r-1,r-1} \end{pmatrix}^{-1} \begin{pmatrix} \tilde{s}_{r1} \\ \tilde{s}_{r2} \\ \vdots \\ \tilde{s}_{r,r-1} \end{pmatrix},$$

for $r = 2, 3, \ldots, p$, where the $\{\phi_{ij}\}$ are those elements of $\mathbf{T}$ which must be estimated. Solving $\operatorname{diag}\{S_2(\hat{\mathbf{T}}, \hat{\mathbf{D}})\} = 0$ gives

$$\hat{\mathbf{D}} = \operatorname{diag}\{\hat{\mathbf{T}}\tilde{\mathbf{S}}\hat{\mathbf{T}}'\}.$$

Appendix D

Coding

D.1 Background

This aim of this section is to outline the C code that was written in order to execute the AECM algorithm for the PGMM family of models and later extended to account for the GPGMM family with the addition of four extra models. At the early stages of its development, simulation studies were used to test the code. An overview of the code is given in the following sections.

D.2 Functionality

D.2.1 Organisation & Input

A makefile is used for compilation so that the code compiles on entering `make` and the executable is called `pgmm`. To remove old program and data files, the command `make allclean` is used. This code dynamically allocates memory so that the size of the dimensionality of the data is not hard-written into the code. Data is read in from a file containing a single column; given the number of variables, this column is easily separable into variables.

D.2.2 Command Line

Arguments are taken on the command line; `getopt()` was used to facilitate this. The method 'flag' must be one of those given in Table D.1.

Table D.1: The command line options available in `pgmm`.

Flag	Method
a	Run all models; the CCC model is randomly seeded and every other model takes the classification output from this CCC model as their starting classification values. Must be followed by filename for input.
f	Filename for input; only to be used when just one member of the family is to be run.
h	Invoke a help file.
l	Set the number of times to loop over the AECM algorithm, default is two.
m	Run one (specified) model with random initial classification. Must be followed by model number; 1 (CCC), 2 (CCU), 3 (CUC),..., 12 (UUCU).
g	Set the maximum number of groups, default is 8.
n	Set the number of observations.
p	Set the number of variables.
q	Set the number of factors, default is 8.
s	Set the seed for the random starting classifications. By default, a function of time is used.

For example, to run all of the models on data with 19 variables and 200 observations contained in `file.txt`, with 1–6 groups and 1–4 factors, the command line might read

```
./pgmm -a file.txt -g 6 -q 4 -p 19 -n 200.
```

To run just the UUC model on these data, the command line may read

```
./pgmm -m 7 -f file.txt -g 6 -q 4 -p 19 -n 200.
```

Convieniently, the order of the command line arguments is not at all important due to the nature of `getopt()`.

D.2.3 Output to Files

The files produced by running `pgmm` are described in Table D.2.

Table D.2: The files produced by running `pgmm`.

Filename	Contents
`bicstore`	The maximum BIC values for each (G, q) over all models.
`o.model_name`	The maximum BIC values for each (G, q) for a particular model.
`o.seed`	The random seed that was used to obtain the initial classifications.
`o.class`	The classifications for the model with the greatest BIC.

D.2.4 Output to Screen

The name, number of groups and number of factors are given for the best model. The classification for the best model is also printed. Another feature of this code is that a warning message is printed to the screen if the log-likelihood has decreased from one iteration to another. Decreasing log-likelihood during an AECM algorithm should only happen due to round-off error and a frequent or systematic decrease in log-likelihood would indicate an error in the algorithm, the code, or both.

The name of each member of the PGMM family is printed to screen just before the AECM algorithm is run on it and each pair (G, q) is printed to the screen in turn. An example of such output is given below.

```
CUC
(1, 1)  (1, 2)  (1, 3)  (1, 4)  (1, 5)
(2, 1)  (2, 2)  (2, 3)  (2, 4)  (2, 5)
(3, 1)  (3, 2)  (3, 3)  (3, 4)  (3, 5)
(4, 1)  (4, 2)  (4, 3)  (4, 4)  (4, 5)
(5, 1)  (5, 2)  (5, 3)  (5, 4)  (5, 5)
```

D.2.5 Notes on the Structure of the Code

The function `main()` is in `pgmm.c`; `main()` is quite terse and mostly farms out tasks to other functions. One of these functions is `op1()`, which is responsible for randomly seeding the array of classifications and sending them to the appropriate algorithm. The function `op1()`, together with the necessary headers, is given below.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "functions.h"
```

```
void op1(double **x, double ****z, double **bayes, char *fname, int N, int p,
    int q, int G,
    void (*func)(double **z1, double **x1, double **b1, int, int, int, int)){

    int i,j,g,m;
    double rd,count=0;
    FILE *file_out;

    file_out = fopen(fname, "w");
    if(!file_out){
        printf("Error opening %s!\a",fname);
        exit(1);
    }

    /* The random seeding happens here */
    {FILE *seedfile;
    int seed;
    seed = time(NULL)+G*q+N;
    srand48(seed);
    seedfile = fopen("o.seed","a");
    fprintf(seedfile,"seed %d\n",seed);
    fclose(seedfile);}

    /* The 'stick breaking' happens here */
    for(g=0;g<G;g++){
       count=0;
       for(j=1;j<=q;j++){
          for(i=0; i<N; i++){
             rd = drand48();
             for(m=0;m<=g;m++){
                z[g][j-1][i][m]=0;
                if(rd<((double)(m+1)/(double)(g+1)) &&
                   rd>((double)m/(double)(g+1))){
                   z[g][j-1][i][m]=1;
                }
             }
          }
          printf("(%d, %d)\t",g+1,j); fflush(stdout);
          /* The function call is made and timed */
          start_timer();
          func(z[g][j-1], x, bayes, j, p, g+1, N);
          if(!timedout())
          fprintf(file_out,"BIC[G=%d][q=%d]:\t%f\n",g+1,j,bayes[g+1][j]);
          else   bayes[g+1][j] = -100000.00;
```

```
        }
        printf("\n");
    }
    fclose(file_out);
}
```

There is another function in the file `operator.c`, called `op()`, that is very similar to `op1()` except that it uses a given array of classifications. The file `functions.h` contains function declarations for all of the functions used either directly or indirectly by main. The function `void (*func)()` can be thought of as a place-holder for one of the functions `aecm()`, `aecm2()`,..., `aecm12()`.

D.2.6 Example of an AECM Algorithm

Code for the application of the AECM to the CCC model is given below. This code is contained in the file `aecm.c`.

```
/******* Function to execute AECM algorithm for G groups, q factors
and p variables in CCC case ********/

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include "functions.h"

void aecm(double **z, double **x, double **bic, int q, int p, int G,
    int N){

    double **sigma, **lambda, **mu, **w, **beta, **theta,
       **sampcovtilde, *l, *at, *pi, *n;
    int it,stop,paras;
    double psi, detpsi, c=1.0, geometry_pi, detsig;
    geometry_pi = 3.14159265;

   /* Allocate memory - matrices and vectors */
    my_alloc_vec(&pi,G);
    my_alloc_vec(&n,G);
    my_alloc_vec(&at,150000);
    my_alloc(&sigma,p,p);
    my_alloc_vec(&l,150000);
    my_alloc(&sampcovtilde,p,p);
    my_alloc(&lambda,p,q);
```

```
my_alloc(&beta,q,p);
my_alloc(&theta,q,q);
my_alloc(&mu,G,p);
my_alloc(&w,G,N);

update_n(n, z, G, N);
update_pi(pi, n, G, N);
update_mu(mu, n, x, z, G, N, p);

psi = initialise_psi(z, x, mu, p, G, N);
initialise_lambda(lambda, p, q);

it=0; stop=0;
while(stop==0){

  update_n(n, z, G, N);
  update_pi(pi, n, G, N);
  update_mu(mu, n, x, z, G, N, p);

  if(it>0) BOMBOUT(update_z(w, x, z, sigma, mu, pi, c, N, G,p));

  update_stilde(sampcovtilde,x, z, mu, G, N, p);

  update_sigmainv(sigma, lambda, psi, p, q);

  update_beta(beta, sigma, lambda, p, q);

  update_theta(theta, beta, lambda, sampcovtilde, p, q);

  update_lambda(lambda, beta, sampcovtilde, theta, p, q);

  psi = update_psi(lambda, beta, sampcovtilde, p, q);

  update_sigmainv(sigma, lambda, psi, p, q);

  detpsi = pow(psi,p);

  detsig = update_det_sigma(lambda, sigma, detpsi, p, q);

  c = pow(2*geometry_pi,p/2.0)*sqrt(detsig);

  /* Abort if calculations go outside machine precision */
  BOMBOUT(update_z(w, x, z, sigma, mu, pi, c, N, G,p));
```

```
      stop = convergtest(l, at, w, N, it, G);
      it++;
    }

    /* no of parameters */
    paras = G-1 + G*p + p*q - q*(q-1)/2 + 1;

    /* BIC = 2log_likelihood - mlog(n) */
    bic[G][q] = 2*l[it-1] - paras*log(N);


    /* Deallocate memory */
    my_free(sigma,p); my_free(lambda,p); my_free(mu,G); my_free(w,G);
    free(n); my_free(beta,q); my_free(theta,q); my_free(sampcovtilde,p);
    free(l); free(at); free(pi);
}
```

D.2.7 Compatibility of Output with R

The contents of the files `bicstore` and `o.class` can easily be read into the software package R. This facilitates the production of the contingency tables and heat maps given in chapters 3 and 4.

Appendix E

Tables from the Analysis of CAO Data

E.1 Explanation of Course Codes

Table E.1: Course codes appearing in this work.

CK	**Cork Institute of Technology**
CR107	Electronic Engineering
CR108	Mechanical Engineering
CK	**University College Cork (NUI)**
CK101	Arts
CK102	Social Science
CK111	Early Childhood Studies
CK201	Commerce
CK204	Finance
CK210	Government and Public Policy
CK301	Law
CK402	Biological and Chemical Sciences
CK701	Medicine
CM	**Coláiste Mhuire**
CM001	BEd
DC	**Dublin City University**
DC111	Business Studies
DC121	Computer Applications
DC126	Financial and Actuarial Mathematics
DC131	Communication Studies

DC132	Journalism
DC181	Biotechnology
DC191	Electronic Engineering
DC192	Telecommunications Engineering
DC201	Common Entry into Science (Undenominated Entry)
FT	**Dublin Institute of Technology**
FT101	Arcgitecture
FT111	Construction Economics and Management (Quantity Surveying)
FT125	Engineering
FT221	Electrical and Electronic Engineering
FT222	Applied Sciences
FT224	Optometry
FT281	Computer Engineering
FT351	Business Studies
FT352	Media Arts
FT353	Communications — Journalism
FT354	Information Systems Development
FT401	Hospitality (Hotel and Catering) Management
FT402	Tourism Marketing
FT471	Social Care
FT472	Early Childhood Care and Education
FT541	Marketing
FT542	Management and Marketing
FT543	Retail and Services Management
FR	**Froebel College of Education**
FR001	BEd
NC	**National Coll of Ireland**
NC001	Accounting and Human Resource Management
RC	**Royal College of Surgeons in Ireland**
RC001	Medicine
CS	**St Catherine's College of Education**
CS001	BEd (Home Economics)
PD	**St. Patrick's College of Education**
PD101	BEd
PD103	BA in Humanities
TR	**Trinity College Dublin**
TR004	Law
TR032	Engineering
TR051	Medicine
TR071	Science
TR084	Social Studies (Social Work)
DN	**University College Dublin (NUI)**
DN001	Architecture

DN002	Medicine
DN003	Engineering
DN005	Veterinary Medicine
DN007	Social Science
DN008	Science
DN009	Law (BCL)
DN012	Arts
DN015	Commerce
DN020	Actuarial and Financial Studies
DN021	Business and Legal Studies
DN054	Psychology plus 2 permissible subjects under DN012 [Arts] for first year only
LC	**Limerick Institute of Technology**
LC017	Construction Economics (Quantity Surveying)
LC019	Construction Management
MI	**Mary Immaculate College**
MI005	BEd
LM	**University of Limerick**
LM047	Arts
LM050	Business Studies
LM069	Computer Engineering
LM081	Manufacturing Technology
GA	**Galway-Mayo Institute of Technology**
GA042	Construction Management
GY	**National University of Ireland, Galway**
GY101	Arts
GY103	Arts (Public and Social Policy)
GY201	Commerce
GY251	Bachelor of Civil Law (BCL)
GY301	Science
GY401	Engineering (Undenominated)
GY402	Civil Engineering
GY501	Medicine
MH	**National University of Ireland, Maynooth**
MH101	Arts
MH102	Finance
MH106	Psychology
MH201	Science
MU	**Pontifical University, Maynooth**
MU001	Theology and Arts
WD	**Waterford Institute of Technology**
WD025	Construction Management
WD026	Electronics

E.2 Rules Interrelating Courses

Table E.2: The 72 rules mentioned in Section 7.5.4, ranked by confidence.

	Rule	Support	Confidence	Lift
1	{GY501, CK701, TR051, RC001} ⇒ {DN002}	0.525%	97.92%	33.76
2	{MI005, DN012, CM001, FR001} ⇒ {PD101}	0.547%	97.67%	21.22
3	{PD101, FR001, LM047} ⇒ {MI005}	0.547%	96.71%	21.17
4	{CM001, FR001, PD103} ⇒ {PD101}	0.599%	96.41%	20.95
5	{MI005, PD103} ⇒ {PD101}	0.512%	95.82%	20.82
6	{CM001, LM047} ⇒ {MI005}	0.558%	95.54%	20.91
7	{GY101, MI005, CM001, FR001} ⇒ {PD101}	0.584%	95.44%	20.74
8	{DN012, CM001, MH101, FR001} ⇒ {PD101}	0.642%	95.04%	20.65
9	{MI005, CM001, MH101, FR001} ⇒ {PD101}	0.579%	94.24%	20.48
10	{CK701, TR051, DN002, RC001} ⇒ {GY501}	0.525%	92.76%	38.84
11	{CK101, CM001} ⇒ {MI005}	0.538%	92.33%	20.21
12	{TR084, FT472} ⇒ {FT471}	0.586%	92.11%	17.38
13	{GY101, DN012, FR001} ⇒ {PD101}	0.521%	92.11%	20.01
14	{DN008, TR051} ⇒ {DN002}	0.543%	91.25%	31.47
15	{GY501, CK701, DN002, RC001} ⇒ {TR051}	0.525%	89.52%	41.31
16	{DC111, FT542, DN015} ⇒ {FT351}	0.662%	89.44%	9.47
17	{GY101, MH101, FR001} ⇒ {PD101}	0.571%	89.24%	19.39
18	{CK101, FR001} ⇒ {MI005}	0.560%	89.05%	19.49
19	{MI005, DN012, PD101, CM001} ⇒ {FR001}	0.547%	88.82%	29.33
20	{TR004, GY251} ⇒ {DN009}	0.501%	88.67%	27.93
21	{MI005, PD101, CM001, MH101} ⇒ {FR001}	0.579%	88.60%	29.26
22	{TR071, MH201} ⇒ {DN008}	1.049%	88.13%	13.62
23	{CK101, PD101} ⇒ {MI005}	0.766%	88.03%	19.27
24	{GY101, CM001, MH101} ⇒ {PD101}	0.526%	87.89%	19.10
25	{DN021, TR004} ⇒ {DN009}	0.698%	87.41%	27.53
26	{TR071, GY301} ⇒ {DN008}	1.019%	87.12%	13.47
27	{DC111, FT541, DN015} ⇒ {FT351}	0.841%	86.92%	9.20
28	{CK402, TR071} ⇒ {DN008}	0.655%	86.91%	13.43
29	{TR071, FT222} ⇒ {DN008}	0.536%	86.23%	13.33
30	{MI005, FR001, LM047} ⇒ {PD101}	0.547%	85.96%	18.68
31	{DN012, MU001} ⇒ {MH101}	0.737%	85.90%	5.64
32	{DC131, DC132, FT352} ⇒ {FT353}	0.549%	85.51%	26.42
33	{DN054, MH106, MH101} ⇒ {DN012}	0.809%	85.29%	4.17
34	{DC111, FT541, FT542} ⇒ {FT351}	0.969%	85.27%	9.03
35	{GY501, TR051, DN002, RC001} ⇒ {CK701}	0.525%	85.20%	46.54
36	{PD101, FR001, PD103} ⇒ {CM001}	0.599%	85.19%	30.01
37	{GY101, MI005, PD101, CM001} ⇒ {FR001}	0.584%	85.09%	28.10
38	{FT471, CK111} ⇒ {FT472}	0.845%	84.86%	18.53

39	{TR032, GY401} ⇒ {DN003}	0.541%	84.84%	22.35
40	{MH101, GY103} ⇒ {GY101}	0.590%	84.76%	5.42
41	{DC111, NC001} ⇒ {FT351}	0.783%	84.54%	8.95
42	{CK101, DN012, MH101, LM047} ⇒ {GY101}	0.939%	84.17%	5.38
43	{DN012, PD101, CM001, MH101} ⇒ {FR001}	0.642%	84.15%	27.79
44	{GY101, MU001} ⇒ {MH101}	0.582%	84.14%	5.52
45	{GY201, MH101} ⇒ {GY101}	0.757%	83.92%	5.37
46	{GY101, DN012, PD103} ⇒ {MH101}	0.543%	83.91%	5.51
47	{MI005, DN012, MH101} ⇒ {PD101}	0.588%	83.60%	18.17
48	{MH106, GY101, DN012} ⇒ {MH101}	0.580%	83.42%	5.47
49	{TR071, DC201} ⇒ {DN008}	0.504%	83.38%	12.89
50	{CK102, LM047} ⇒ {CK101}	0.512%	83.08%	6.02
51	{FT541, FT542, DN015} ⇒ {FT351}	0.727%	83.01%	8.79
52	{CK101, GY101, DN012, LM047} ⇒ {MH101}	0.939%	82.92%	5.44
53	{DN012, TR004} ⇒ {DN009}	0.765%	82.86%	26.10
54	{CK101, FR001} ⇒ {PD101}	0.521%	82.84%	18.00
55	{CK301, TR004} ⇒ {DN009}	0.558%	82.64%	26.03
56	{LM050, FT351, DN015} ⇒ {GY201}	0.532%	82.18%	13.34
57	{CK201, CK210} ⇒ {CK101}	0.512%	82.09%	5.95
58	{GY101, PD101, CM001, FR001} ⇒ {MI005}	0.584%	81.98%	17.95
59	{MI005, DN012, PD101, FR001} ⇒ {CM001}	0.547%	81.89%	28.85
60	{GY101, MI005, DN012} ⇒ {PD101}	0.526%	81.56%	17.72
61	{DC111, FT354} ⇒ {FT351}	0.621%	81.46%	8.63
62	{FT351, MH102, DN015} ⇒ {DC111}	0.584%	81.35%	10.46
63	{FT351, FT542, FT543} ⇒ {FT541}	0.521%	80.92%	11.45
64	{TR071, DC181} ⇒ {DN008}	0.519%	80.87%	12.50
65	{MI005, PD101, MH101, FR001} ⇒ {CM001}	0.579%	80.78%	28.46
66	{PD101, CM001, PD103} ⇒ {FR001}	0.599%	80.70%	26.65
67	{DN007, MH101} ⇒ {DN012}	1.287%	80.65%	3.94
68	{CK204, CK101} ⇒ {CK201}	0.662%	80.54%	18.38
69	{DN054, MH106, DN012} ⇒ {MH101}	0.809%	80.41%	5.28
70	{DC111, LM050, DN015} ⇒ {GY201}	0.593%	80.35%	13.05
71	{DC111, FT541, DN012} ⇒ {FT351}	0.515%	80.29%	8.50
72	{DN012, PD101, MH101, FR001} ⇒ {CM001}	0.642%	80.23%	28.26

E.3 Rules with Consequent $\{F\}$

Table E.3: Top twenty rules, ranked by interestingness, with consequent {F}.

	Antecedent	Support	Confidence	Lift	Interest.
1	{FT471}	4.987%	94.10%	1.72	0.00165
2	{FT472}	4.494%	98.13%	1.80	0.00140
3	{PD101}	4.014%	87.23%	1.60	0.00098
4	{MI005}	3.892%	85.18%	1.56	0.00089
5	{DN007}	3.469%	85.79%	1.57	0.00072
6	{CK111}	3.192%	96.30%	1.78	0.00069
7	{CK102}	3.021%	87.88%	1.61	0.00056
8	{FR001}	2.766%	91.34%	1.68	0.00049
9	{FT472, FT471}	2.491%	98.10%	1.80	0.00042
10	{CM001}	2.506%	88.27%	1.62	0.00039
11	{PD101, FR001}	2.390%	91.46%	1.68	0.00037
12	{PD101, CM001}	2.245%	89.61%	1.64	0.00032
13	{DN007, DN012}	2.247%	85.07%	1.59	0.00030
14	{MI005, PD101}	2.175%	87.96%	1.61	0.00029
15	{TR084}	2.111%	92.13%	1.69	0.00029
16	{CM001, FR001}	1.961%	90.86%	1.66	0.00025
17	{PD103}	1.987%	84.03%	1.54	0.00023
18	{PD101, CM001, FR001}	1.823%	91.33%	1.67	0.00021
19	{DN012, PD101}	1.780%	89.77%	1.65	0.00020
20	{MI005, FR001}	1.717%	90.67%	1.66	0.00019

E.4 Rules with Consequent $\{M\}$

Table E.4: Top twenty rules, ranked by interestingness, with consequent {M}.

	Antecedent	Support	Confidence	Lift	Interest.
1	{FT125}	2.907%	87.12%	1.92	0.00062
2	{TR032}	2.556%	80.30%	1.77	0.00044
3	{WD025}	2.318%	90.09%	1.98	0.00040
4	{GA042}	2.269%	91.59%	2.02	0.00039
5	{LC017}	2.273%	87.85%	1.93	0.00038
6	{LC019}	2.17%	91.75%	2.02	0.00036
7	{FT111}	2.18%	88.74%	1.95	0.00035
8	{FT221}	2.015%	89.65%	1.97	0.00030
9	{FT281}	1.767%	84.07%	1.85	0.00022
10	{DC191}	1.468%	88.55%	1.95	0.00016
11	{CR107}	1.432%	88.71%	1.95	0.00015
12	{CR108}	1.397%	91.25%	2.01	0.00015
13	{DN003, FT125}	1.269%	83.99%	1.85	0.00011
14	{LC019, LC017}	1.205%	93.51%	2.06	0.00011
15	{LM069}	1.259%	81.86%	1.80	0.00011
16	{TR032, FT125}	1.161%	85.95%	1.89	0.000097
17	{WD026}	1.127%	90.99%	2.00	0.000095
18	{GY402}	1.168%	82.52%	1.82	0.000095
19	{DC192}	1.068%	84.16%	1.85	0.000081
20	{LM081}	1.027%	89.76%	1.98	0.000078

Appendix F

Lift & Negations

F.1 Range of Values for Lift

Theorem F.1. *Suppose I is an item set, and $A \Rightarrow B$ is an association rule on a set of transactions $\{\tau_1, \ldots, \tau_n\}$ over I. Then*

$$\frac{\max\{P(A) + P(B) - 1, 1/n\}}{P(A)P(B)} \leq L(A \Rightarrow B) \leq \frac{1}{\max\{P(A), P(B)\}}.$$

Proof. First consider the bounds of $P(A, B)$. If $P(A) + P(B) \leq 1$, then the appropriate lower bound is given by

$$P(A, B) \geq \frac{1}{n},$$

otherwise the appropriate lower bound is $P(A, B) \geq P(A) + P(B) - 1$.

The upper bound is given by $P(A, B) \leq \min\{P(A), P(B)\}$. Therefore,

$$\max\{P(A) + P(B) - 1, 1/n\} \leq P(A, B) \leq \min\{P(A), P(B)\},$$

and so,

$$\frac{\max\{P(A) + P(B) - 1, 1/n\}}{P(A)P(B)} \leq L(A \Rightarrow B) \leq \frac{1}{\max\{P(A), P(B)\}}.$$

□

Theorem F.2. *Suppose I is an item set, and $A \Rightarrow B$ is an association rule mined over a set of transactions $\{\tau_1, \ldots, \tau_n\}$ over I, using support threshold s. Then*

$$\frac{4s}{(1+s)^2} \leq L(A \Rightarrow B) \leq \frac{1}{s}.$$

Proof. Let $s = \vartheta/n$. The maximum value of lift will occur when

$$P(A, B) = P(A) = P(B) = \frac{\vartheta}{n}$$

and so

$$L(A \Rightarrow B) \leq \frac{\vartheta/n}{(\vartheta/n)^2} = \frac{n}{\vartheta} = \frac{1}{s}.$$

To find the minimum value of lift, first consider the case where $n+\vartheta$ is even, then $L(A \Rightarrow B)$ is minimised when

$$P(A, B) = \frac{\vartheta}{n} \text{ and } P(A) = P(B) = \frac{(n+\vartheta)/2}{n} = \frac{(n+\vartheta)}{2n},$$

however, if $n + \vartheta$ is odd then $L(A \Rightarrow B)$ is minimised when

$$P(A, B) = \frac{\vartheta}{n}, \quad P(A) = \frac{(n+\vartheta+1)/2}{n} = \frac{(n+\vartheta+1)}{2n}$$
$$\text{and } P(B) = \frac{(n+\vartheta-1)/2}{n} = \frac{(n+\vartheta-1)}{2n}.$$

Now,

$$\frac{(n+\vartheta-1)}{2n} \cdot \frac{(n+\vartheta+1)}{2n} = \frac{(n+\vartheta+1)(n+\vartheta-1)}{4n^2} < \frac{(n+\vartheta)^2}{4n^2},$$

therefore,

$$L(A \Rightarrow B) \geq \frac{\vartheta/n}{(n+\vartheta)^2/4n^2} = \frac{4n\vartheta}{(n+\vartheta)^2} = \frac{4s}{(1+s)^2}.$$

□

Lemma F.1. *Suppose I is an item set, and $A \Rightarrow B$ is an association rule on a set of transactions $\{\tau_1, \ldots, \tau_n\}$ over I. Suppose that minimum thresholds for support and confidence are used in the mining process, denoted s and c respectively. Then*

$$\max\left\{\frac{P(A)+P(B)-1}{P(A)P(B)}, \frac{4s}{(1+s)^2}, \frac{s}{P(A)P(B)}, \frac{c}{P(B)}\right\} \leq L(A \Rightarrow B)$$
$$\leq \frac{1}{\max\{P(A), P(B)\}}.$$

Proof. In addition to the results of theorems F.1 and F.2, note that if $P(A, B) \geq s$ then

$$L(A \Rightarrow B) = \frac{P(A, B)}{P(A)P(B)} \geq \frac{s}{P(A)P(B)},$$

and if $P(B \mid A) \geq c$ then

$$L(A \Rightarrow B) = \frac{P(B \mid A)}{P(B)} \geq \frac{c}{P(B)}.$$

□

F.2 Negations Theorem

Theorem F.3. *The number of rules that can possibly be generated from an itemset consisting of n items and their negations is given by*

$$5^n - 2\left(3^n\right) + 1.$$

Proof. If A and B contain a total of m items then the number of rules involving these m items and their negations is given by

$$\left[\sum_{r=1}^{m-1} {}^{m}C_r\right] 2^m = \left[\sum_{r=0}^{m} {}^{m}C_r - 2\right] 2^m = \left(2^m - 2\right) 2^m = 2^{2m} - 2^{m+1}.$$

Therefore, from an itemset of size n there are $2^{2n} - 2^{n+1}$ rules of length n, there are ${}^{n}C_{n-1}\left[2^{2(n-1)} - 2^{(n-1)+1}\right]$ rules of length $n-1$, ${}^{n}C_{n-2}\left[2^{2(n-2)} - 2^{(n-2)+1}\right]$ rules of length $n-2$ and so on. It follows that the total number of rules that can be generated from these n items and their negations is given by

$$\left(2^{2n} - 2^{n+1}\right) + {}^{n}C_{n-1}\left[2^{2(n-1)} - 2^{(n-1)+1}\right] + \cdots + {}^{n}C_2\left[2^{2(2)} - 2^{2+1}\right].$$

Which can be expressed as

$$\sum_{i=2}^{n} {}^{n}C_i\left(2^{2i} - 2^{i+1}\right). \tag{F.1}$$

Now, the binomial theorem states that

$$(1+x)^n = \sum_{i=0}^{n} {}^{n}C_i x^i,$$

and so developing Equation F.1 gives,

$$\begin{aligned}
\sum_{i=2}^{n} {}^{n}C_i\left(2^{2i} - 2^{i+1}\right) &= \sum_{i=2}^{n} {}^{n}C_i 2^{2i} - \sum_{i=2}^{n} {}^{n}C_i 2^{i+1} = \sum_{i=2}^{n} {}^{n}C_i 4^i - 2\sum_{i=2}^{n} {}^{n}C_i 2^i \\
&= \left[\sum_{i=0}^{n} {}^{n}C_i 4^i - \left({}^{n}C_0 + {}^{n}C_1(4)\right)\right] - 2\left[\sum_{i=0}^{n} {}^{n}C_i 2^i - \left({}^{n}C_0 + {}^{n}C_1(2)\right)\right] \\
&= \left[5^n - (1+4n)\right] - 2\left[3^n - (1+2n)\right] = 5^n - 2\left(3^n\right) + 1.
\end{aligned}$$

□