Dimensionality Reduction with Non-Gaussian Mixtures

McMaster University

DIMENSIONALITY REDUCTION WITH NON-GAUSSIAN MIXTURES

BY

YANG TANG

A Thesis Submitted to the School of Graduate Studies in Partial Fulfilment of the Requirements for the Degree Doctor of Philosophy

Science Doctor of Philosophy (2017) McMaster University
(Department of Mathematics and Statistics) Hamilton, Ontario, Canada

TITLE:	Dimensionality Reduction with Non-Gaussian Mixtures
AUTHOR:	Yang Tang Ph.D., (Mathematics and Statistics) McMaster University, Hamilton, Canada
SUPERVISOR:	Dr. Paul D. McNicholas
NUMBER OF PAGES:	xv, 103

For my parents , with love

Abstract

Broadly speaking, cluster analysis is the organization of a data set into meaningful groups and mixture model-based clustering is recently receiving a wide interest in statistics. Historically, the Gaussian mixture model has dominated the model-based clustering literature. When model-based clustering is performed on a large number of observed variables, it is well known that Gaussian mixture models can represent an over-parameterized solution. To this end, this thesis focuses on the development of novel non-Gaussian mixture models for high-dimensional continuous and categorical data. We developed a mixture of joint generalized hyperbolic models (JGHM), which exhibits different marginal amounts of tail-weight. Moreover, it takes into account the cluster specific subspace and, therefore, limits the number of parameters to estimate. This is a novel approach, which is applicable to high, and potentially very-high, dimensional spaces and with arbitrary correlation between dimensions. Three different mixture models are developed using forms of the mixture of latent trait models to realize model-based clustering of high-dimensional binary data. A family of mixture of latent trait models with common slope parameters are developed to reduce the number of parameters to be estimated. This approach facilitates a low-dimensional visual representation of the clusters. We further developed the penalized latent trait models to facilitate ultra high dimensional binary data which performs automatic variable selection as well. For all models and families of models developed in this thesis, the algorithms used for model-fitting and parameter estimation are presented. Real and simulated data sets are used to assess the clustering ability of the models.

Acknowledgements

First and foremost I would like to express my gratitude toward my advisors Dr. Paul D. McNicholas and Dr. Ryan P. Browne, for their support, guidance and continuous encouragement throughout the course of my degree. The joy and enthusiasm they have for research were contagious and motivational for me, even during tough times in the Ph.D. pursuit. I am also truly grateful for the valuable opportunities Dr. McNicholas has provided for me and the excellent example he has set as a professor and scientist.

Thank you to Dr. Aaron Childs for serving as a members of my supervisory committee, Dr. Joel Dubin for examining this thesis, Dr. Roman Viveros-Aguilera for serving on my examining committee, and to Dr. Donald Sprung for chairing my thesis defence.

Thank you to all members of the McNicholas research group for their support, advices as well as friendship, throughout this journey.

A special thank you to Dr. Salvatore Ingrassia and Dr. Antonio Punzo for the collaboration and for providing me with the wonderful opportunity to research at the University of Catania, Italy.

I would like to acknowledge the financial support I received from an Ontario Graduate Scholarship.

Finally, my deepest thanks to my family, particularly my parents, for all their unconditional love and encouragement. Thank you for giving me the opportunities and being

patient and supportive when I needed. Thank you to Derek for you endless patience and unconditional love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

1.1.1 What is High-Dimensional Data?

How to analyze high dimensional data is a topic that has been receiving increasing attention over the past few years. Data collection has become easier and faster due to technology advances, and every aspect of our lives is influenced by data. Companies are keen to enlarge their traditional data sets with social media data, browser logs, as well as text analytics and sensor data to get a more complete picture of their customers; most sports teams use GPS equipment and video analytics that track the performance of players and use smart technology to track athletes' nutrition and sleep as well as their emotional wellbeing; and a number of cities are trying to optimize traffic flows based on real time traffic information as well as social media and weather data. So not only do we have a large amount of data, but it also tends to be complex and messy in comparison to traditional data.

1.1.2 Cluster Analysis

Cluster analysis is the organization of large data sets into meaningful clusters (groups). Clustering algorithms can be categorized based on how the underlying models operate. The similarity between objects can be determined using distance measures over the various dimensions in the data set or by gathering individuals that have arisen from the same distribution using a probabilistic framework.

Clustering algorithms based on distance measures (e.g., k-means clustering (MacQueen, 1967), hierarchical clustering (Ward J., 1963)) have received much attention in literature. The k-means paradigms are simply divisions of the set of data objects into non-overlapping subsets such that each data object is in exactly one subset. On the other hand, hierarchical clustering joins data objects together in a hierarchical fashion from the closest, that is most similar, to the furthest apart, that is the most different. Therefore we obtain a graphical representation of the matrix of distances (i.e., a dendrogram or tree). Traditional clustering algorithms based on distance measures cannot cope with high-dimensional and/or complex data sets because of their high complexity and computational cost. The standard hierarchical clustering methods can handle data with both numerical and categorical values (e.g., Anderberg, 1973; Jain and Dubes, 1988). However, the quadratic computational cost makes them less suitable for clustering large data sets. On the other hand, the k-means clustering method is efficient for processing large data sets. It minimizes a cost function by changing the means of a cluster. This prohibits it from being used in applications where categorical data are involved.

Model-based clustering provides an alternative to distance-based methods for clustering. McNicholas (2016) defines a cluster as a unimodal component within an appropriate finite mixture model. Herein, we use the definition by McNicholas (2016). However, when model-based clustering is performed on a large number of observed variables, it is well known that it can be over-parameterized and very computationally intensive, as, besides the mixing weights, it is required to estimate the mean vector and the covariance matrix for each component (McLachlan and Peel, 2000a). McNicholas (2016) further explains that an "appropriate" mixture model is one that is appropriate in light of the data under consideration, that the model has the necessary flexibility, or parameterization to fit the data. Increasing attention has been drawn to this problem, aimed at parameterizing the generic component-covariance matrix (Celeux and Govaert, 1995; Fraley and Raftery, 1998; Browne and McNicholas, 2014) or at performing dimensional reduction in each component through latent variables (McLachlan *et al.*, 2003; Bouveyron *et al.*, 2007; McNicholas and Murphy, 2008, 2010; Baek *et al.*, 2010; Vermunt, 2007; Browne and McNicholas, 2012; Gollini and Murphy, 2014; Murray *et al.*, 2014). The models developed in this thesis add to this growing body of work and offer advantages not necessarily provided by these other models.

1.2 Thesis Structure

1.2.1 Chapter 2

Background details on finite mixture models, some non-Gaussian distributions often used in model-based clustering and latent trait models for the analysis of binary data. Dimension reduction techniques for clustering high-dimensional data are discussed, followed by parameter estimation techniques. Methods for model-selection and performance assessment are also discussed.

1.2.2 Chapter 3

Mixtures of joint generalized hyperbolic models are developed via a novel distribution, a joint generalized hyperbolic model (JGHM). Algorithm for model parameter estimation is presented. The clustering ability of the model is illustrated on real and simulated data and the models are compared.

1.2.3 Chapter 4

A mixture of latent trait models via contaminated Gaussian distributions (MLTCG) is proposed. We assume that the low dimensional continuous latent variable comes from a contaminated Gaussian distribution and therefore picks up extreme patterns in the observed binary data while clustering. The clustering performance is demonstrated on real and simulated data and the models are compared.

1.2.4 Chapter 5

A penalized mixture of latent trait models (PMLTM) for clustered binary data is developed: we assume that the data have been generated by a mixture of latent trait models and we shrink the slope parameters, with a gamma-Laplace penalty function. The newly developed variational EM algorithm provides closed-form estimates for model parameters and avoids intensive searches of the tuning parameters through model selection criterion such as Bayesian information criterion. The clustering results are reported for several data sets. A comparison between selected programming languages is shown on simulated and real data.

1.2.5 Chapter 6

We propose a mixture of multinomial latent trait models with common slope parameters (MMCLT). We implement a multinomial logistic response function for the use of clustering categorical data when there exists more than two categories. The sharing of the slope parameters reduces the number of parameters to a manageable size; however, each latent trait still has a different effect in each group. A new variational EM algorithm based on two quadratic lower bounds to the multinomial likelihood is developed.

1.2.6 Chapter 7

The ideas and methods demonstrated in this thesis are summarized in this last chapter. Suggestions for future work are discussed.

1.3 Impact

The impact of this work is summarized here. The principal novel features of this work are:

(i) The mixture of joint generalized hyperbolic models is developed based on generalized hyperbolic distributions which represent perhaps the most flexible in a recent series of alternatives to the Gaussian mixture model for clustering and classification. The component specific subspaces reduce the number of parameters significantly to realize high-dimensional data clustering.

(ii) We extend the mixture of latent trait analyzers to accommodate extreme patterns while clustering. The MLTCG model can automatically detect extreme observations and therefore be more accurate about cluster identification. The MLTCG model represents a useful tool for finding extreme patterns in clustered high-dimensional binary data because they cannot be easily visualized.

(iii) The PMLTM model enables us to encourage sparsity in estimating the slope parameters, thus reducing the number of free parameters considerably, and achieves automatic variable selection for clustered high dimensional binary data. The component-specific independent tuning parameters avoid the over-penalization that can occur when inferring a shared tuning parameter on clustered data. The new EM algorithm provides efficient parameter estimation.

iv The development of the MMCLT model makes an important contribution to literature on mixture models capable of handling high-dimensional categorical data. More specifically, this work extends a vein of research on mixture of latent trait models. Furthermore, the new variational EM algorithm is developed based on two quadratic lower bounds to the multinomial likelihood.

Chapter 2

Background

2.1 Model-Based Clustering

2.1.1 Overview

Model-based clustering is a fundamental statistical approach for clustering, where data are clustered using some assumed mixture modelling structure and the group memberships are "learned" in an unsupervised fashion. A finite mixture model is a convex combination of a finite number of simple component distributions. Therefore, the density of a general finite mixture model is given by

$$f(\boldsymbol{x}_i|\boldsymbol{\Theta}) = \sum_{g=1}^{G} \pi_g p_g \left(\boldsymbol{x}_i; \boldsymbol{\theta}_g\right), \tag{2.1}$$

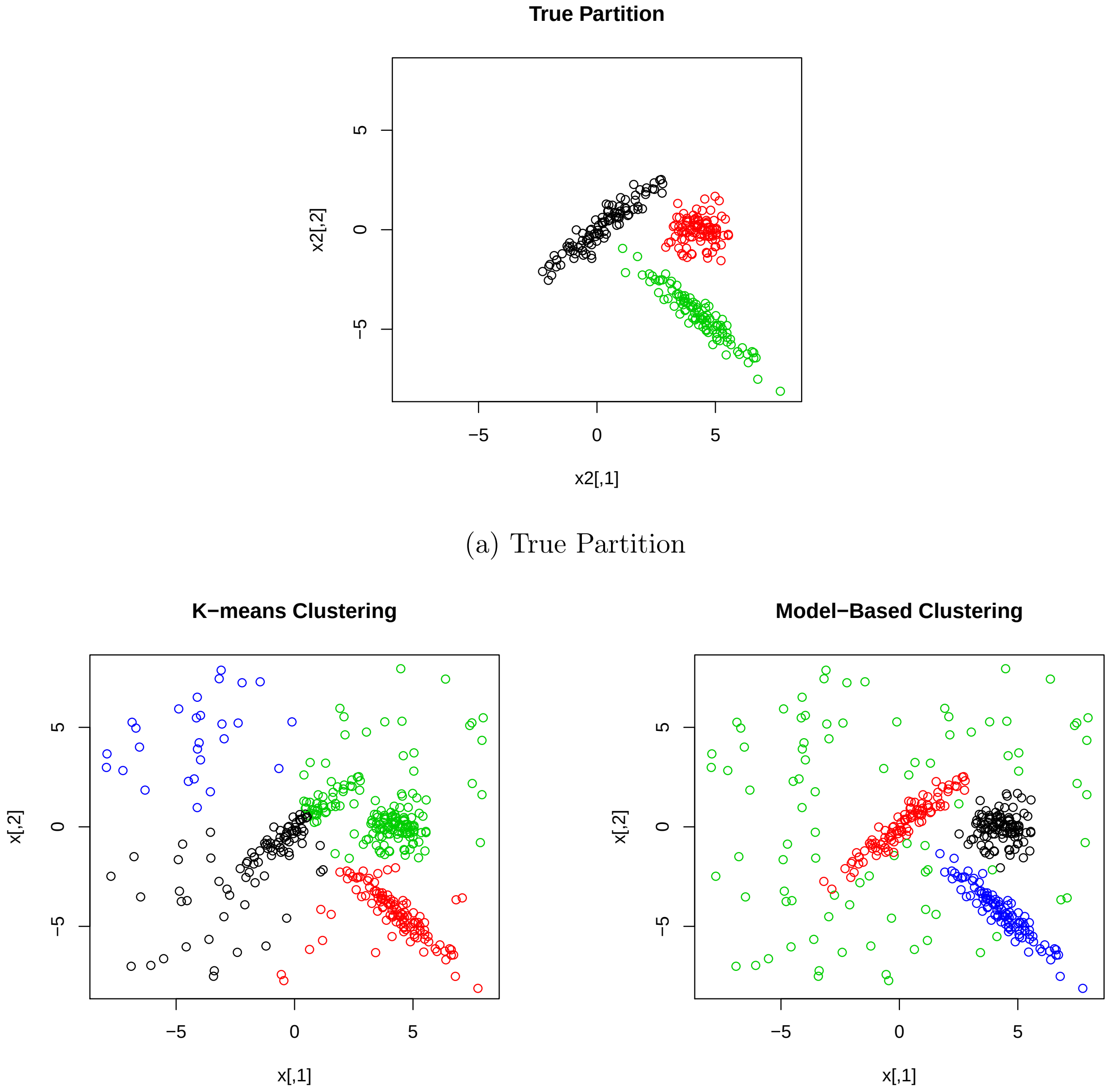
where G is the number of components, π_g ($\pi_g \in (0, 1]$, and $\sum_{g=1}^{G} \pi_g = 1$) is the probability that an observation $\boldsymbol{x}_i$ belongs to group g and $\boldsymbol{\theta}_g$ contains unknown parameters in the mixture model. The EM algorithm (Bock and Aitkin, 1981) is used to find the MLE (maximum likelihood estimates) of the parameters.

2.1.2 Finite Mixture Models for Real-Valued Data

The Gaussian mixture model is probably the most well known in literature (e.g., Wolfe, 1963; Banfield and Raftery, 1993; Celeux and Govaert, 1995; Fraley and Raftery, 2002). Mixture model approaches have shown promising results with large data sets and data sets with noise. We can use a simple example to illustrate the need for model-based clustering. Consider simulated data with two variables, three components and added noise ($X_1 \sim \text{Uniform}[-10, 10]$, $X_2 \sim \text{Uniform}[-10, 10]$). When k-means is used to cluster this data set, it does a poor job of finding the true clusters because of the background noise. Instead, we try using a model-based algorithm. Figure 2.1 shows the plots of classification when $G = 4$. Moreover, model-based methods offer better interpretability because the model parameters directly characterizes the clusters. A limitation of model-based clustering with high-dimensional data is that if the dimension of the data is high relative to the number of observations, then the covariance estimates in the ellipsoidal models will often be singular. Hence, more parsimonious models are proposed (e.g., McNicholas and Murphy, 2008; Browne and McNicholas, 2012).

2.1.2.1 Finite Mixture Models for Clustering Non-Continuous Data and Mixed Data

For non-continuous data, one needs to specify $p_g(\boldsymbol{x}_i; \boldsymbol{\theta}_g)$ in Equation 2.1 through probability mass functions. While there are plenty of choices for univariate non-continuous distributions, the use of multivariate non-continuous distributions for the definition of the mixture models is limited due to the difficulty in constructing models that allow flexibility on the dependence structure. Recent work include finite mixtures of multinomial distributions (Jorgensen, 2004) for categorical data and finite mixtures of multivariate Poisson distributions (Karlis and



(a) True Partition

(b) K-means Partition

(c) Model-Based Clustering

Figure 2.1: Comparison between k-means and model-based clustering.

Meligkotsidou, 2007) for count data. However, these models are limited in application due to the high parameterization.

A new framework via mixtures of copulas has been proposed to accommodate the modelling of data with either continuous or non-continuous domains. Copulas offer the means for constructing multivariate models due to the flexibility in describing dependence among mixture components and allows the easy construction of multivariate models with prescribed marginals. A few attempts on copula-based mixture models have already been made (see, for examples, Jajuga and Papla, 2006; Di Lascio and Giannerini, 2012; Vrac *et al.*, 2012; Marbac *et al.*, 2014b; Kosmidis and Karlis, 2016). More investigations are needed for the application of the framework on scenarios with high-dimensional data. A wide-range of parsimonious parameterization between exchangeable and unstructured correlation matrices can be obtained by drawing ideas from parsimonious parameterizations in Gaussian mixture models like the eigenvalue decomposition proposed in Celeux and Govaert (1995). It can be directly applied to any copula family that is parameterized in terms of a full covariance matrix, allowing the comparison of a wide range of parsimonious models.

2.2 Non-Gaussian Distributions

2.2.1 The Generalized Hyperbolic Distribution

The generalized hyperbolic distribution (GHD) represents perhaps the most flexible among the recent series of alternatives to the Gaussian mixture model. The GHD is capable of handling skewness and heavy tails, and has many well-known distributions as special or limiting cases (Table 2.1).

The generalized hyperbolic distribution (McNeil *et al.*, 2015) takes the form

$$\begin{aligned} f_{\text{GHD}}(\boldsymbol{x}|\boldsymbol{\theta}) = & \left[\frac{\chi + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{\psi + \boldsymbol{\alpha}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}}\right]^{(\lambda-p/2)/2} \\ & \times \frac{[\psi/\chi]^{\lambda/2} K_{\lambda-p/2}\left(\sqrt{[\psi + \boldsymbol{\alpha}\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}][\chi + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})]}\right)}{(2\pi)^{p/2} \mid \boldsymbol{\Sigma} \mid^{1/2} K_{\lambda}(\sqrt{\chi\psi})\exp(\boldsymbol{\mu}-\boldsymbol{x})'\boldsymbol{\Sigma}^{-1}\boldsymbol{\alpha}}, \end{aligned} \tag{2.2}$$

where $\boldsymbol{x}$ is a p-dimensional data vector, $K_{\lambda}(\cdot)$ is the modified Bessel function of the third

Table 2.1: Generalized hyperbolic distribution and its special and limiting cases (Prause, 1999).

GHD	GIG	λ	χ	ψ
General case	General case	$\in \mathbb{R}$	> 0	> 0
Hyperbolic	Positive Hyperbolic	$(p+1)/2$	> 0	> 0
GIG		$\in \mathbb{R}$	$\to 0$	const.
Normal-inverse Gaussian	Inverse Gaussian	$-\frac{1}{2}$	> 0	> 0
Variance gamma	Inverse gamma	> 0	> 0	> 0
Student's t	Gamma	$-\frac{v}{2} < 0$	$v > 0$	0
Skewed Laplace		$(p+1)/2$	0	> 0
Normal		$\in \mathbb{R}$	$\to \infty$	$\to \infty$

kind with index λ, and $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}, \lambda, \chi, \psi)$ is a vector of parameters. A mixture of generalized hyperbolic distributions has been developed for clustering by Browne and McNicholas (2015a).

One attractive feature of the GHD is that the p-dimensional random variable $\boldsymbol{X}$ can be generated via the relationship

$$\boldsymbol{X} = \boldsymbol{\mu} + W\boldsymbol{\beta} + \sqrt{W}\boldsymbol{V},$$

where $\boldsymbol{V} \sim \text{MVN}(\boldsymbol{0}, \boldsymbol{\Sigma})$ and $W \sim \text{GIG}(\Omega, 1, \lambda)$, cf. Browne and McNicholas (2015a). Under this parameterization, the density of the generalized hyperbolic distribution is

$$\begin{aligned} f_{\text{GHD}}(\boldsymbol{x}|\boldsymbol{\theta}) = & \left[\frac{\omega + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{\omega + \boldsymbol{\beta}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}}\right]^{(\lambda-p/2)/2} \\ & \times K_{\lambda-p/2}\left(\sqrt{[\omega + \boldsymbol{\beta}\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}][\omega + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})]}\right) \\ & \times (2\pi)^{p/2} \mid \boldsymbol{\Sigma} \mid^{1/2} K_{\lambda}(\sqrt{\omega})\exp(\boldsymbol{\mu}-\boldsymbol{x})'\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}, \end{aligned} \tag{2.3}$$

and $\boldsymbol{W}|\boldsymbol{x} \sim \text{GIG}(\omega + \boldsymbol{\beta}\boldsymbol{\Sigma}^{-1}\boldsymbol{\beta}, \omega + (\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}, \lambda - p/2)$. This parameterization yields tractable expected values that lead to the development of a computationally efficient E-step for fitting the MJGHM model in Chapter 3.

2.2.2 Multiple Scaled Distributions

Forbes and Wraith (2014) show that the density of the random variable $\boldsymbol{X}$ from a multiple scaled normal variance mixture can be written

$$f(\boldsymbol{x}|\mu, \boldsymbol{D}, \boldsymbol{A}, \theta) = \int_0^\infty \cdots \int_0^\infty \phi_p(\boldsymbol{x}|\mu, \boldsymbol{D}\Delta_{\boldsymbol{w}}\boldsymbol{A}\boldsymbol{D}') \times f_{\boldsymbol{W}}(w_1, \ldots, w_p|\theta) dw_1 \ldots dw_p, \tag{2.4}$$

where $\phi_p(\boldsymbol{X}|\mu, \boldsymbol{D}\Delta_w\boldsymbol{A}\boldsymbol{D}')$ is the multivariate Gaussian density and $\Delta_{\boldsymbol{w}} = \text{diag}(w_1, \ldots, w_p)$ is a diagonal weight matrix where $w_1, \ldots, w_p$ are independent, and $f_{\boldsymbol{W}}(w_1, \ldots, w_p|\theta)$ is a p-variate density function with parameter θ. Forbes and Wraith (2014) focused on a multiple scaled t-distribution. Later, Franczak *et al.* (2015) proposed a mixture of multiple scaled shifted asymmetric Laplace (MSSAL) distributions for clustering applications. The upper level sets of the MSSAL density are guaranteed to be convex, and therefore ideal for clustering.

2.3 Dimensionality Reduction

2.3.1 Latent Variable Models for Continuous Data

Mixtures of factor analyzers is a statistical method which concurrently performs clustering and within each cluster, local dimensionality reduction. Assume that the distribution of the observation $\boldsymbol{X}_i$ is modelled using a q-dimensional vector of continuous latent variable, $\boldsymbol{U}_i \sim \text{MVN}(0, \boldsymbol{I}_q)$, where q is generally much smaller than p. The generative model is given by:

$$\boldsymbol{X}_i = \boldsymbol{\mu}_g + \boldsymbol{\Lambda}_g\boldsymbol{U}_{ig} + \boldsymbol{\epsilon}_{ig} \qquad \text{with prob. } \pi_g,$$

where $\boldsymbol{\Lambda}_g$ is known as the factor loading matrix and $\boldsymbol{\epsilon}_{ig} \sim \text{MVN}(0, \boldsymbol{\Psi}_g)$ with $\boldsymbol{\Psi}_g = \text{diag}(\psi_{1g}, \ldots, \psi_{pg})$. The $\boldsymbol{U}_i$ and $\boldsymbol{\epsilon}_i$ are both independently distributed and are independent of each other. Thus the marginal distribution of $\boldsymbol{X}$ in the gth component is $\text{MVN}(\boldsymbol{\mu}_g, \boldsymbol{\Lambda}_g\boldsymbol{\Lambda}'_g + \boldsymbol{\Psi}_g)$. Many versions of the mixture of factor analyzers have been developed over time (e.g., Ghahramani and Hinton, 1996; Tipping and Bishop, 1999; McLachlan and Peel, 2000b; Yoshida *et al.*, 2004; McNicholas and Murphy, 2008; Baek *et al.*, 2010).

2.3.2 Latent Variable Models for Non-Continuous Data and Mixed Data

Mixture models with latent structure have been considered for the analysis of non-continuous data and mixed type data. Mixture of latent class analysis and mixture of latent trait analysis are two commonly used latent variable models for categorical data and mixed data. In latent class models, the dependence in the data is explained by a categorical latent variable that identifies groups and the response variables are independent (known as the local independence assumption). Latent class models are widely used for model-based clustering of categorical data and mixed data (e.g., Goodman, 1974; Celeux and Govaert, 1991; Biernacki *et al.*, 2010). However, if the condition of independence within the group is violated, latent class models tend to overestimate the number of components and can be potentially misleading. Different models relax the conditional independence assumption. The multilevel latent class models (Vermunt, 2003, 2007) assume that the conditional dependency between the response variables can be explained by other unobserved variables. Marbac *et al.* (2014a) propose a conditional modes model which groups the response variables into conditionally independent blocks. The corresponding block is a parsimonious multinomial distribution which brings out the intra-class dependency between variables. However, the model is difficult to estimate if the data set has a large number of variables.

The latent trait models use a continuous univariate or multivariate latent variable to model the dependence among the categorical or mixed response variables. Recently proposed, mixtures of latent trait models for the analysis of categorical and mixed response variables include work by Muthen *et al.* (2006), Vermunt (2007), Khan *et al.* (????), Browne and McNicholas (2012), Cagnone and Viroli (2012), Gollini and Murphy (2014).

2.3.3 Subspace Clustering via Gaussian Mixture Models

A unified approach for model-based subspace clustering is introduced by Bouveyron *et al.* (2007). Within the Gaussian mixture model framework, this approach assumes that class conditional densities are Gaussian MVN$(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ for $g = 1, \ldots, G$. Let $\boldsymbol{Q}_g$ consist of the eigenvectors of $\boldsymbol{\Sigma}_g$ as columns and $\boldsymbol{\Phi}_g$ are the eigenvalues. Then the component covariance matrices $\boldsymbol{\Sigma}_g$, for $g = 1, \ldots, G$, can be written $\boldsymbol{\Sigma}_g = \boldsymbol{Q}_g \boldsymbol{\Phi}_g \boldsymbol{Q}_g'$, where $\boldsymbol{\Phi}_g$ is divided into two blocks

$$\boldsymbol{\Phi}_g = \left(\begin{array}{ccc|ccc} \phi_{g1} & & 0 & & & \\ & \ddots & & & 0 & \\ 0 & & \phi_{gq_g} & & & \\ \hline & & & b_g & & 0 \\ & 0 & & & \ddots & \\ & & & 0 & & b_g \end{array}\right)$$

with $\phi_{gj} > b_g$, $j = 1, \ldots, q_g$ and $q_g < p$. The component-specific subspace $\mathcal{E}_g$ is defined as the affine space rotated by the q_g eigenvectors associated with the eigenvalues ϕ_{gj}. Drawing ideas from model-based subspace clustering, Chapter 3 describes the joint generalized hyperbolic model (JGHM) which projects p-dimensional $\boldsymbol{X}$ onto two subspaces.

2.4 Mixture of Latent Trait Models and Penalized Latent Variable Models

2.4.1 Mixture of Latent Trait Models

2.4.1.1 Overview

Gollini and Murphy (2014) assume that the conditional distribution of $\boldsymbol{x}_i$ given that the observation is from group g (i.e., $z_{ig} = 1$) is a latent trait model with parameters b_{mg}, $\boldsymbol{w}_{mg}$; and the latent variable $\boldsymbol{Y}_i \sim N(0, I)$. Thus, the MLTA model is of the form,

$$p(\boldsymbol{x}_i) = \sum_{g=1}^{G} \eta_g \int_{\boldsymbol{\mathcal{Y}}_i} p(\boldsymbol{x}_i|\boldsymbol{y}_i, z_{ig} = 1)p(\boldsymbol{y}_i)d\boldsymbol{y}_i, \tag{2.5}$$

where

$$p(\boldsymbol{x}_i|\boldsymbol{y}_i,\, z_{ig} = 1) = \prod_{m=1}^{M} [\pi_{mg}(\boldsymbol{y}_i)]^{x_{im}}[1 - \pi_{mg}(\boldsymbol{y}_i)]^{1-x_{im}},$$

and the response function for each group is given by

$$\pi_{mg}(\boldsymbol{y}_i) = p(x_{im} = 1|\boldsymbol{y}_i,\, z_{ig} = 1) = \frac{1}{1 + \exp(-(b_{mg} + \mathbf{w}'_{mg}\boldsymbol{y}_i))},$$

where b_{mg} and $\boldsymbol{w}_{mg}$ are the model parameters.

In particular, b_{mg} has a direct effect on the probability of a positive response to the variable m given by an individual in group g,

$$\pi_{mg}(0) = p(x_{nm} = 1|\boldsymbol{y}_n = 0, z_{ng} = 1) = \frac{1}{1 + \exp(-b_{mg})}.$$

The value $\pi_{mg}(0)$ is the probability that the median individual in group g has a positive response for the variable m. The value of the slope parameters account for the correlation between the observed response variables.

The log-likelihood can be written as,

$$l = \sum_{i=1}^{n} \log \left(\sum_{g=1}^{G} \eta_g \int_{\boldsymbol{\mathcal{Y}}_i} \prod_{m=1}^{M} p(x_{im}|\boldsymbol{y}_i, z_{ig}=1) p(\boldsymbol{y}_i) d\boldsymbol{y}_i \right). \tag{2.6}$$

2.4.2 Mixture of Latent Trait Models with Common Slope Parameters

Tang *et al.* (2015) introduce the mixture of latent trait models with common slope parameters (MLCT) that restricts the MLTA model by using common slope parameters that reduce the number of parameters to a manageable size; still, each latent trait has a different effect in each component. It also facilitates low-dimensional visual representation of components with posterior means of the continuous latent variables corresponding to the observed data. The MCLT model assumes there is a d-dimensional continuous latent variable $\boldsymbol{Y}$ underlying the behaviour of the M binary response variables, where $\boldsymbol{Y}$ comes from G different components, i.e., $Y_{ig} \sim \text{MVN}(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$. In addition, all latent traits share a set of common slope parameters $\boldsymbol{W} = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_M)$ in the logistic function.

Therefore , the MCLT model takes the form,

$$p(\boldsymbol{x}_i) = \sum_{g=1}^{G} \eta_g p(\boldsymbol{x}_i | z_{ig}=1) = \sum_{g=1}^{G} \eta_g \int_{\boldsymbol{\mathcal{Y}}_{ig}} p(\boldsymbol{x}_i|\boldsymbol{y}_{ig}, z_{ig}=1) p(\boldsymbol{y}_{ig}) d\boldsymbol{y}_{ig},$$

where

$$p(\boldsymbol{x}_i|\boldsymbol{y}_{ig}, z_{ig}=1) = \prod_{m=1}^{M} [\pi_{mg}(\boldsymbol{y}_{ig})]^{x_{im}} [1 - \pi_{mg}(\boldsymbol{y}_{ig})]^{1-x_{im}},$$

and the response function for each categorical variable in each component is

$$\pi_{mg}(\boldsymbol{y}_{ig}) = p(x_{im}=1|\boldsymbol{y}_{ig}, z_{ig}=1) = \frac{1}{1+\exp\{-\boldsymbol{w}_m' \boldsymbol{y}_{ig}\}},$$

where $\boldsymbol{w}_m$ is the common model parameter and the latent variable $\boldsymbol{Y}_{ig} \sim \text{MVN}(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$.

Therefore, the model is a finite mixture model in which the gth component latent variable $\boldsymbol{Y}_{ig}$ is MVN$(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ and the mixing proportions are $\eta_1, \eta_2, \ldots, \eta_G$.

2.4.2.1 Approximations to the Log-Likelihood of Latent Trait Models

In the mixture of latent trait models, the integral in Equation 2.5 is intractable. A variational approach (Jaakkola and Jordan, 2000) can be used for a fast algorithm since it has a closed-form solution for parameter updates and provides a lower bound approximation to the log-likelihood (Chapters 4, 5, and 6)

Jaakkola and Jordan (2000) first proposed a variational approximation for the predictive likelihood in a Bayesian logistic regression model and also briefly considered the "dual" problem, which is closely related to the latent trait model. The variational approximation of the logistic function

$$p(\boldsymbol{x}_i|\boldsymbol{y}_i,\, z_{ig}=1) = \frac{\exp(\boldsymbol{y}_i)}{1+\exp(\boldsymbol{y}_i)} = (1+\exp(-\boldsymbol{y}_i))^{-1},$$

can be approximated by the exponential of a quadratic form involving variational parameters $\boldsymbol{\xi}_{ng} = (\xi_{i1g}, ..., \xi_{img})$, where $\xi_{img} \neq 0$ for all $m = 1, ..., M$.

Now, the lower bound of each term in the log-likelihood takes the form,

$$L(\boldsymbol{\xi}_{ig}) = \log(\underline{p}(\boldsymbol{x}_i|\boldsymbol{\xi}_{ig}) = \log\left(\int \prod_{m=1}^{M} \underline{p}(x_{im}|\boldsymbol{y}_i, z_{ig}=1, \xi_{img})p(\boldsymbol{y}_i)\, d\boldsymbol{y}_i\right),$$

where

$$\underline{p}(x_{im}|\boldsymbol{y}_i, z_{ig}=1, \xi_{img}) = \sigma(\xi_{img})\exp\left(\frac{A_{img}-\xi_{img}}{2} + \lambda(\xi_{img})(A_{img}^2 - \xi_{img}^2)\right),$$

$$A_{img} = (2x_{im}-1)(b_{mg} + \boldsymbol{w}'_{mg}\boldsymbol{y}_i),$$

$$\lambda(\xi_{img}) = (\frac{1}{2} - \sigma(\xi_{img}))/2\xi_{img},$$

$$\sigma(\xi_{img}) = (1+\exp(-\xi_{img}))^{-1}.$$

This approximation is used to obtain a lower bound for the log-likelihood estimation that leads to the development of a computationally efficient E-step for fitting the MLTCG, PMLTM and MMCLT models in Chapters 4, 5 and 6.

2.4.3 Penalized Latent Variable Models

A problem that arises with fitting high-dimensional binary data via a latent variable model is the large number of parameters required. Penalized latent variable models are developed to carry out dimension reduction and variable selection simultaneously. There has recently been an increasing interest in penalized latent variable models (see Houseman *et al.*, 2007; DeSantis *et al.*, 2008). Houseman *et al.* (2007) propose a penalized item response theory model with univariate traits which penalizes the item-response slopes with ridge penalties. However, it doesn't take into account the potential group structure of the data, and Gauss-Hermite quadrature is required to approximate the likelihood. DeSantis *et al.* (2008) develop a penalized latent class model to facilitate analysis of high-dimensional ordinal data. A ridge penalty is introduced to the feature-based parameterization of class-specific response probabilities to stabilize maximum likelihood estimation. Both methods adopt a shared tuning parameter among variables and require a model selection criterion to choose it.

The Lasso estimator (Tibshirani, 1996) often yields solutions with some parameter estimates being exactly 0. Galimberti *et al.* (2009) develop a penalized factor mixture analysis which contextually performs dimension reduction and variable selection by shrinking the factor loadings through a penalized likelihood method with a Lasso penalty. Results proved the capability of the model to select the relevant variables in the presence of a large number of irrelevant ones. However, they use shared tuning parameters for all variables and an exhaustive search using BIC is required to chose the tuning parameters.

2.5 The EM Algorithm and Extensions

2.5.1 The EM Algorithm

The EM algorithm (Dempster *et al.*, 1977) is an iterative computational procedure for calculating maximum likelihood estimates when data are incomplete. The EM algorithm is commonly used for the fitting of mixture models and parameter estimation in model-based clustering where the incomplete data arises from the unobserved cluster label, and in some cases, other latent variables. On each iteration of the EM algorithm, there are two steps – an expectation step (E-step), where the complete-data likelihood is calculated based on current model parameters, and a maximization step (M-step), where the expected value of the complete-data log-likelihood is maximized with respect to the model parameters. The algorithm alternates between the E and M-steps until some convergence criterion is reached. In this thesis, we use several alternatives of the classical expectation-maximization for model fitting. McLachlan and Krishnan (2007) gave a detailed review of the EM algorithm and its extensions.

2.5.2 The Generalized EM Algorithm

The generalized EM algorithm (GEM) is applied when the solution to the M step does not exist in closed-form, and it may not be feasible to attempt to find the value of $\boldsymbol{\Theta}$ that globally maximizes the function $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$. Thus, the M-step in GEM requires $\boldsymbol{\Theta}^{(t+1)}$ to be chosen such that

$$Q(\boldsymbol{\Theta}^{(t+1)}; \boldsymbol{\Theta}^{(t)}) \geq Q(\boldsymbol{\Theta}^{(t)}; \boldsymbol{\Theta}^{(t)})$$

holds. Instead of maximizing the Q-function $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$, one chooses $\boldsymbol{\Theta}^{(t+1)}$ to increase the Q function over its value at $\boldsymbol{\Theta} = \boldsymbol{\Theta}^{(k)}$. Dempster *et al.* (1977) showed that the likelihood is not decreased after a GEM iteration so a GEM sequence of likelihood values must converge if bounded above.

2.5.3 The ECM Algorithm

Meng and Rubin (1993) introduce a variant to the EM algorithm, the expectation-conditional maximization (ECM) algorithm, for use in the instance that the complete-data likelihood estimation is relatively complicated. The ECM algorithm replaces a complicated M-step by a number of computationally simpler conditional maximization steps (CM-steps). The ECM algorithm shares the convergence properties of the classic EM algorithm. Moreover, the algorithm usually converges more quickly in terms of total computation time, though it may require more iterations to reach convergence.

2.5.4 The EM Algorithm for Maximum a Posteriori and Maximum Penalized Estimation

The EM algorithm is easily modified to find the mode of a posterior distribution in a Bayesian framework, producing the maximum a posteriori (MAP) estimate corresponding to some prior density $p(\boldsymbol{\Theta})$ for $\boldsymbol{\Theta}$. On the $(t+1)$th iteration, instead of estimating the complete-data likelihood, we calculate the conditional expectation of the log complete-data posterior density given the observed data using the current MAP estimate $\boldsymbol{\Theta}^{(t)}$. That is,

$$\mathbb{E}_{\boldsymbol{\Theta}^{(t)}}\{\log p(\boldsymbol{\Theta}|\text{complete-data})|\text{observed data}\} = Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)}) + \log p(\boldsymbol{\Theta}).$$

We choose $\boldsymbol{\Theta}^{(t+1)}$ to maximize the log complete-data posterior density in the M-steps. The objective function in the M-step is equal to $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ augmented by the log prior density

$\log p(\boldsymbol{\Theta})$. The imposition of a Bayesian prior for $\boldsymbol{\Theta}$ almost always makes the objective function more concave.

2.6 The MM Algorithm

In minimization problems, MM stands for majorize-minimize, and in maximization problems, MM stands for minorize-maximize. The majorization-minizmation algorithm (Hunter and Lange, 2004) is an optimization method that is particularly useful in high-dimensional problems. The MM algorithm substitutes a simple optimization problem for a difficult one by separating the variables, avoiding large matrix inversions or turning a non-differentiable problem into a smooth problem. Let $g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ denote a real-valued function of $\boldsymbol{\Theta}$ whose form depends on $\boldsymbol{\Theta}^{(t)}$. The function $g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ is said to majorize $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ at point $\boldsymbol{\Theta}^{(t)}$ provided

$$
\begin{aligned}
g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}) &\geq Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)}), \\
g(\boldsymbol{\Theta}^{(t)}|\boldsymbol{\Theta}^{(t)}) &= Q(\boldsymbol{\Theta}^{(t)}).
\end{aligned}
$$

Therefore, in the MM algorithm, we minimize the majorizing function $g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ rather than the actual function $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ and the MM procedure is numerically stable because it forces $Q(\boldsymbol{\Theta})$ to not increase after a MM iteration. With straightforward changes, the MM algorithm also applies to maximization rather than minimization: To maximize the function $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$, we minorize it by a surrogate function $g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ and maximize $g(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ to produce the next iteration $\boldsymbol{\Theta}^{(t+1)}$.

Indeed, every EM algorithm is a special case of the more general class of MM algorithms, which typically exploit convexity rather than missing data in majorizing or minorizing an objective function.

2.6.1 Convergence

There are a variety of ways to measure convergence. One common approach is in terms of either the size of the relative change in the parameter estimates or the log likelihood. One stops the algorithm when the increase in the log-likelihood between continuous iterations is less than a given threshold ϵ, i.e.,

$$\log \mathcal{L}(\boldsymbol{\Theta}^{(t+1)} \mid \boldsymbol{X}) - \log \mathcal{L}(\boldsymbol{\Theta}^{(t)} \mid \boldsymbol{X}) < \epsilon.$$

However, this is a measure of "lack of progress" and not of actual convergence.

Böhning *et al.* (1994) exploited Aitken's acceleration procedure (Aitken, 1926) which can be used as a convergence criterion. This stopping criterion determines convergence by estimating the limiting value of the log-likelihood at each iteration of the EM algorithm. The Aitken acceleration at iteration t is

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

where $l^{(t)}$ is the log-likelihood at iteration t. An asymptotic estimate of the log-likelihood at iteration t is

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

Böhning *et al.* (1994) suggest considering an algorithm to be converged when

$$|l_\infty^{(t+1)} - l_\infty^{(t)}| < \epsilon.$$

For all algorithms developed in this thesis, convergence is determined via Aitken's acceleration. The stopping criterion suggested by Böhning *et al.* (1994) is used with $\epsilon = 0.01$.

2.7 Model Selection and Performance Assessment

2.7.1 Model Selection

The Bayesian information criterion (BIC; Schwarz, 1978) is commonly used for model selection in model-based clustering. The BIC takes the form,

$$\text{BIC} = -2l + k \log n, \tag{2.7}$$

where l is the maximized log-likelihood, k is the number of free parameters to be estimated in the model, and n is the number of observations. Schwarz (1978) proved that, if one of the models $M_1, \ldots, M_m$ is correct, so that there is a true $\mathbf{\Theta}$ in that model, as n becomes large, the probability approaching 1, BIC will select the best model. Poskitt (1987) and Haughton (1988) extended and improved Schwarz's work, showing that consistency held also under less restrictive conditions. The BIC is used as a model selection criterion throughout the thesis to select the number of clusters G and the dimension of the latent variable Y, the covariance structure and the number of dimensions of the component specific subspaces where appropriate. When defined as in Equation 2.7, models with lower values of BIC are preferable.

2.7.2 The Adjusted Rand Index

For high-dimensional binary data, particularly when the number of observations n is not very large relative to their dimension m, it is common to have a large number of patterns with small observed frequency. Accordingly, we cannot use a χ^2 test to check the goodness of the model fit. To assess the model performance, a measure of agreement is needed. When the true underlying groups are known, the clustering performance of the models can be assessed by comparing the known class labels to the estimated group memberships. We assign each observation $\boldsymbol{x}_i \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ to one and only one group g, $g = 1, \ldots, G$, with the largest corresponding z_{ig} value after convergence. These are referred to as the MAP classifications. The Rand index (Rand, 1971) is simply calculated as pairwise agreements between the true class labels and the MAP classification,

$$\frac{\text{number of pairwise agreements}}{\text{total number of pairs}}. \tag{2.8}$$

The Rand index lies between 0 and 1. When two partitions agree perfectly, the Rand index is 1. A problem with the Rand index is that the expected value of the Rand index of two random partitions does not take a constant value (say zero) and smaller values are difficult to interpret. Therefore, Hubert and Arabie (1985) proposed the adjusted Rand index (ARI). The ARI is the corrected-for-chance version of the Rand index. The general form of the ARI is

$$\frac{\text{index} - \text{expected index}}{\text{maximum index} - \text{expected index}},$$

which is bounded above by 1, and has expected value 0 under random classification.

Chapter 3

Mixtures of Joint Generalized Hyperbolic Models

3.1 Introduction

In this chapter, mixtures of joint generalized hyperbolic models are developed via a novel distribution, a joint generalized hyperbolic model (JGHM). The JGHM exhibits different marginal amounts of tail-weight. Drawing ideas from model-based subspace clustering (Bouveyron *et al.*, 2007), we take into account the cluster specific subspace to limit the number of parameters to estimate. This is a novel approach, which is applicable to high – and potentially very-high – dimensional spaces and with arbitrary correlation between dimensions. A multi-cycle expectation-conditional maximization (MCECM) algorithm (Meng and Rubin, 1993) is used for parameter estimation and BIC is used to determine the number of components and the dimensions of the subspaces. This method is a robust asymmetric clustering method for high-dimensional data — “asymmetric" in the sense that the clusters can be asymmetric. Our proposed method is illustrated and compared to standard clustering methods, on simulated and real data.

In Section 3.2, we introduce our mixture of joint generalized hyperbolic models for asymmetric clustering of high-dimensional data (MJGHM-HDClust). Then, the performance of our model is assessed (Section 3.3) and some potential real world applications for subspace clustering are discussed (Section 3.4). The paper concludes with a discussion and suggestions for future work in Section 3.5.

3.2 A Mixture of Joint Generalized Hyperbolic Models

3.2.1 Overview

Drawing ideas from model-based subspace clustering (Bouveyron *et al.*, 2007), the joint generalized hyperbolic model (JGHM) chooses to project p-dimensional $\boldsymbol{X}$ onto two subspaces. We assume there is a q-dimensional subspace which best preserves the variance of the data and is much smaller than the original space. There exists a q-dimensional latent variable $\boldsymbol{W}$ that controls the concentration in the first q dimensions of $[\boldsymbol{\Gamma}'\boldsymbol{x}]$, where $\boldsymbol{\Gamma}$ is a matrix of eigenvectors associated with the eigenvalues $\boldsymbol{\Phi} = (\phi_1, \phi_2, \ldots, \phi_p)$ with $\phi_1 > \phi_2 \cdots > \phi_p$; and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_q)'$ is a q-dimensional index parameter. In addition, outside the q-dimensional subspace, the noise variance is modelled by a single parameter b and a univariate latent variable A where $A \sim \text{GIG}(\omega_0, 1, \lambda_0)$. Therefore, the JGHM takes the form

$$\begin{aligned} f(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\phi}, b, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \omega_0, \lambda_0) &= \prod_{j=1}^{q} \int_0^\infty \rho_1([\boldsymbol{\Gamma}'\boldsymbol{x} - \boldsymbol{\mu} - \boldsymbol{\Delta}_w \boldsymbol{\beta}]_j | \mathbf{0}, \phi_j w_j) h_W(w_j|\Omega_j, 1, \lambda_j) dw_j \\ &\times \int_0^\infty \prod_{k=q+1}^{p} \rho_1([\boldsymbol{\Gamma}'\boldsymbol{x} - \boldsymbol{\mu} - a\boldsymbol{\beta}]_k | \mathbf{0}, b \times a) h_A(a|\omega_0, 1, \lambda_0) da \\ = \prod_{j=1}^{q} \left[\frac{\Omega_j + \phi_j^{-1}([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j)^2}{\Omega_j + \beta_j^2 \phi_j^{-1}} \right]^{\frac{\lambda_j - \frac{1}{2}}{2}} & \frac{K_{\lambda_j - \frac{1}{2}} \left(\sqrt{\left[\Omega_j + \beta_j^2 \phi_j^{-1}\right] \left[\Omega_j + \phi_j^{-1}\left([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j\right)^2\right]} \right)}{(2\pi)^{\frac{1}{2}} \phi_j^{\frac{1}{2}} K_{\lambda_j}(\Omega_j) \exp\left\{ -\frac{([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j)\beta_j}{\phi_j} \right\}} \\ \times \left[\frac{\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)^2}{\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} \beta_k^2} \right]^{\frac{(\lambda_0 - \frac{p-q}{2})}{2}} & \frac{K_{\lambda_0 - \frac{p-q}{2}} \left(\sqrt{\left[\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} \beta_k^2\right] \left[\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)^2\right]} \right)}{(2\pi)^{\frac{p-q}{2}} b^{\frac{p-q}{2}} K_{\lambda_0}(\omega_0) \exp\left\{ -\frac{\sum\limits_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)\beta_k}{b} \right\}}. \end{aligned} \tag{3.1}$$

As such,

$$W_j|\boldsymbol{x} \sim \text{GIG}(\Omega_j + \beta_j^2 \phi_j^{-1}, \Omega_j + \frac{[\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j}{\phi_j}, \lambda_j - \frac{1}{2}),$$

and

$$A|\boldsymbol{x} \sim \text{GIG}(\omega_0 + b^{-1} \sum_{k=q+1}^{p} \beta_k^2, \omega_0 + b^{-1} \sum_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)^2, \lambda_0 - \frac{p-q}{2}).$$

We use a mixture of JGHMs for model-based clustering and classification. The MJGHM is then given by

$$f(\boldsymbol{x}|\boldsymbol{\Psi}) = \sum_{g=1}^{G} \pi_g f_{\text{JGHM}}(\boldsymbol{x}|\boldsymbol{\Gamma}_g, \boldsymbol{\mu}_g, \boldsymbol{\beta}_g, \boldsymbol{\phi}_g, b, \boldsymbol{\Omega}_g, \boldsymbol{\lambda}_g, \omega_{0g}, \lambda_{0g}),$$

in which we assume component specific subspaces and the dimension q_g of the subspace can be considered as the number of dimensions required to describe the main features of the gth component. The mixing proportions are $\pi_1, \pi_2, \ldots, \pi_G$. It would generally be advantageous to use the MJGHM because the GHD is a flexible distribution, capable of handling skewness and heavy tails, and has many well known distributions as special or limiting cases.

3.2.2 Parameter Estimation

To fit the models, we adopt the MCECM, which is a variant of the well-known EM algorithm. In our case, the missing data comprise the group memberships z_{ig}, where $z_{ig} = 1$ if observation i belongs to component g and $z_{ig} = 0$ otherwise. The multidimensional latent variables $\mathbf{\Delta}_{W_g} = \text{diag}(W_{1g}, \cdots, W_{q_g g}, A_g I_{p-q_g})(g = 1, \cdots, G)$ are assumed to follow GIG distributions. Therefore, the complete-data CD consist of the observed $\boldsymbol{x}_i$ together with the z_{ig} and the $\mathbf{\Delta}_{W_{ig}}$ and complete-data log-likelihood is given by:

$$l_c(\mathbf{\Psi}|\text{CD}) = l_{1c}(\boldsymbol{\pi}|\text{CD}) + l_{2c}(\boldsymbol{\theta}|\text{CD}) + l_{3c}(\boldsymbol{\upsilon}|\text{CD}) + l_{4c}(\boldsymbol{\tau}|\text{CD}),$$

where

$$\begin{aligned}
l_{1c}(\boldsymbol{\pi}|\text{CD}) &= \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\log\pi_g,\\
l_{2c}(\boldsymbol{\theta}|\text{CD}) &= \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\Big\{\log f_p\left([\mathbf{\Gamma}'_g\boldsymbol{x}_i]|\boldsymbol{\mu}_g + \mathbf{\Delta}_{w_{ig}}\boldsymbol{\beta}_g, \mathbf{\Delta}_{w_{ig}}\mathbf{\Phi}_g\right)\Big\},\\
l_{3c}(\boldsymbol{\upsilon}|\text{CD}) &= \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\Big\{\sum_{j=1}^{q_g}(\log h_W(w_{ijg}|\Omega_{jg}, 1, \lambda_{jg}))\Big\} \text{ and}\\
l_{4c}(\boldsymbol{\tau}|\text{CD}) &= \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\Big\{\log h_A(a_{ig}|\omega_{0g}, 1, \lambda_{0g})\Big\},
\end{aligned}$$

where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_G)$, $f_p\left([\mathbf{\Gamma}'_g\boldsymbol{x}_i]|\boldsymbol{\mu}_g + \mathbf{\Delta}_{w_{ig}}\boldsymbol{\beta}_g, \mathbf{\Delta}_{w_{ig}}\mathbf{\Phi}_g\right)$ is the density of a multivariate Gaussian distribution with mean $\boldsymbol{\mu}_g + \mathbf{\Delta}_{w_{ig}}\boldsymbol{\beta}_g$ and $\mathbf{\Phi}_g = \text{diag}(\phi_1, \phi_2, \ldots, \phi_{q_g}, b_g\boldsymbol{I}_{p-q_g})$; accordingly, $\boldsymbol{\theta} = \{\mathbf{\Gamma}_g, \boldsymbol{\mu}_g, \boldsymbol{\beta}_g, \boldsymbol{\phi}_g, b_g\}_{g=1}^{G}$.

We also have $\boldsymbol{\upsilon} = \{\mathbf{\Omega}_g, \boldsymbol{\lambda}_g\}_{g=1}^{G}$ and $\boldsymbol{\tau} = \{\omega_{0g}, \lambda_{0g}\}_{g=1}^{G}$.

The MCECM algorithm iterates between two CM-steps and an E-step is performed before each CM-step. They arise from the partition $\mathbf{\Psi} = (\mathbf{\Psi}_1, \mathbf{\Psi}_2)$,

where $\mathbf{\Psi}_1 = (\pi_g, \boldsymbol{\mu}_g, \boldsymbol{\beta}_g, \boldsymbol{\phi}_g, b_g, \mathbf{\Omega}_g, \boldsymbol{\lambda}_g, \omega_{0g}, \lambda_{0g})$ and $\mathbf{\Psi}_2 = \mathbf{\Gamma}_g$.

1. The E-step: We compute the expected value of the complete-data log-likelihood in the E-step using the expected values of the missing data in $l_c(\boldsymbol{\Psi}|\text{CD})$. We require the following expectations:

$$\mathbb{E}\left[Z_{ig}|\boldsymbol{x}_i\right] = \frac{\pi_g f(\boldsymbol{x}_i|\boldsymbol{\Psi}_g)}{\sum_{h=1}^{G}\pi_h f(\boldsymbol{x}_i|\boldsymbol{\Psi}_h)} =: \hat{z}_{ig}.$$

The expected values of the multidimensional latent variables can be written as:

$$\begin{aligned}
\mathbb{E}\left[W_{ijg}|\boldsymbol{x}_i, z_{ig}=1\right] &= \sqrt{\frac{e_{ijg}}{d_{jg}}}\frac{K_{\lambda_{jg}+1/2}(\sqrt{e_{ijg}d_{jg}})}{K_{\lambda_{jg}-1/2}(\sqrt{e_{ijg}d_{jg}})} =: E_{1ijg}\\
\mathbb{E}\left[W_{ijg}^2|\boldsymbol{x}_i, z_{ig}=1\right] &= \frac{e_{ijg}}{d_{jg}}\frac{K_{\lambda_{jg}+3/2}(\sqrt{e_{ijg}d_{jg}})}{K_{\lambda_{jg}-1/2}(\sqrt{e_{ijg}d_{jg}})} =: E_{2ijg}\\
\mathbb{E}\left[\frac{1}{W_{ijg}}|\boldsymbol{x}_i, z_{ig}=1\right] &= \sqrt{\frac{d_{jg}}{e_{ijg}}}\frac{K_{\lambda_{jg}+1/2}(\sqrt{e_{ijg}d_{jg}})}{K_{\lambda_{jg}-1/2}(\sqrt{e_{ijg}d_{jg}})} - \frac{2\lambda_{jg}-1}{e_{ijg}} =: E_{3ijg}\\
\mathbb{E}\left[\log W_{ijg}|\boldsymbol{x}_i, z_{ig}=1\right] &= \log\sqrt{\frac{e_{ijg}}{d_{jg}}} + \frac{\partial}{\partial \upsilon}\log\left(K_\upsilon(\sqrt{e_{ijg}d_{jg}})\right)|_{\upsilon=\lambda_{jg}-1/2} =: E_{4ijg}
\end{aligned}$$

where $d_{jg} = \Omega_{jg} + \beta_{jg}^2\phi_{jg}^{-1}$ and $e_{ijg} = \Omega_{jg} + \frac{[\boldsymbol{\Gamma}_g'\boldsymbol{x}_i]_j - \mu_{jg}}{\phi_{jg}}$.

$$\begin{aligned}
\mathbb{E}\left[A_{ig}|\boldsymbol{x}_i, z_{ig}=1\right] &= \sqrt{\frac{e_{0ig}}{d_{0g}}}\frac{K_{\lambda_{0g}-(p-q_g)/2+1}(\sqrt{e_{0ig}d_{0g}})}{K_{\lambda_{0g}-(p-q_g)/2}(\sqrt{e_{0ig}d_{0g}})} =: EA_{1ig}\\
\mathbb{E}\left[A_{ig}^2|\boldsymbol{x}_i, z_{ig}=1\right] &= \frac{e_{0ig}}{d_{0g}}\frac{K_{\lambda_{0g}-(p-q_g)/2+2}(\sqrt{e_{0ig}d_{0g}})}{K_{\lambda_{0g}-(p-q_g)/2}(\sqrt{e_{0ig}d_{0g}})} =: EA_{2ig}\\
\mathbb{E}\left[\frac{1}{A_{ig}}|\boldsymbol{x}_i, z_{ig}=1\right] &= \sqrt{\frac{d_{0g}}{e_{0ig}}}\frac{K_{\lambda_{0g}-(p-q_g)/2+1}(\sqrt{e_{0ig}d_{0g}})}{K_{\lambda_{0g}-(p-q_g)/2}(\sqrt{e_{0ig}d_{0g}})} - \frac{2\lambda_{0g}-(p-q_g)}{e_{0ig}} =: EA_{3ig}\\
\mathbb{E}\left[\log A_{ig}|\boldsymbol{x}_i, z_{ig}=1\right] &= \log\sqrt{\frac{e_{0ig}}{d_{0g}}} + \frac{\partial}{\partial \upsilon}\log\left(K_\upsilon(\sqrt{e_{0ig}d_{0g}})\right)|_{\upsilon=\lambda_{0g}-(p-q_g)/2} =: EA_{4ig},
\end{aligned}$$

where $d_{0g} = \omega_{0g} + b_g^{-1}\sum_{k=q_g+1}^{p}\beta_{kg}^2$, and $e_{0ig} = \Omega_{0g} + b_g^{-1}\sum_{k=q_g+1}^{p}([\boldsymbol{\Gamma}_g'\boldsymbol{x}_i]_k - \mu_{kg})^2$.

Thus we have

$$\begin{aligned}
\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}}] &= \mathrm{diag}(E_{1i1g}, E_{1i2g}, \dots, E_{1iq_gg}, EA_{1ig}\boldsymbol{I}_{p-q_g}),\\
\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}] &= \mathrm{diag}(E_{3i1g}, E_{3i2g}, \dots, E_{3iq_gg}, EA_{3ig}\boldsymbol{I}_{p-q_g}),\\
\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}^2}] &= \mathrm{diag}(E_{2i1g}, E_{2i2g}, \dots, E_{2iq_gg}, \hat{E}A_{2ig}\boldsymbol{I}_{p-q_g}).
\end{aligned}$$

2. CM-step 1: The first CM-step on the $(t{+}1)$th iteration requires the calculation of $\boldsymbol{\Psi}_1^{(t+1)}$ as the value of $\boldsymbol{\Psi}_1$ that maximizes $Q(\boldsymbol{\Psi}|\boldsymbol{\Psi}^{(t)})$ with $\boldsymbol{\Psi}_2$ fixed at $\boldsymbol{\Psi}_2^{(t)}$. In particular, we obtain the update for the mixing proportions from $\hat{\pi}_g^{(t+1)} = n_g^{(t)}/n$, where $n_g = \sum_{i=1}^n \hat{z}_{ig}^{(t)}$. The elements of the location parameter $\boldsymbol{\mu}_g$ and skewness parameter $\boldsymbol{\beta}_g$ are replaced with

$$\mu_{jg}^{(t+1)} = \frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j \left(\frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}}]_j^{(t)}}{n_g^{(t)}}\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]_j^{(t)} - 1\right)}{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\left(\frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}}]_j^{(t)}}{n_g^{(t)}}\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]_j^{(t)} - 1\right)}$$

and

$$\beta_{jg}^{(t+1)} = \frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j \left(\frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]_j^{(t)}}{n_g^{(t)}} - \mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]_j^{(t)}\right)}{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\left(\frac{\sum_{i=1}^n \hat{z}_{ig}^{(t)}\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}}]_j^{(t)}}{n_g^{(t)}}\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]_j^{(t)} - 1\right)},$$

respectively, where $j = 1, 2, \dots, p$ and $[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j$ is the j^{th} element of the matrix $[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]$. We update the diagonal elements h_{jg} of the empirical covariance matrix of $[\boldsymbol{\Gamma}_g'\boldsymbol{X}]_j|\boldsymbol{\Delta}_{W_g}$,

$$h_{jg}^{(t+1)} = \frac{1}{n_g^{(t)}}\sum_{n=1}^n\left\{\hat{z}_{ig}^{(t)}([\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j - \mu_{jg}^{(t+1)})^2 - 2\hat{z}_{ig}^{(t)}([\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j - \mu_{jg}^{(t+1)})\beta_{jg}^{(t+1)}\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}}]_j^{(t)} + \hat{z}_{ig}^{(t)}(\mathbb{E}[\boldsymbol{\Delta}_{W_{ig}^2}]_j^{(t)}(\beta_{jg}^2)^{(t+1)}\right\}.$$

We then order $h_{jg}^{(t+1)}$ from the largest to the smallest in order to determine the subspaces. Now we obtain

$$\phi_{jg}^{(t+1)} = \frac{1}{n_g^{(t)}} \sum_{i=1}^{n} \hat{z}_{ig}^{(t)} \left[E_{3ijg}^{(t)}([\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j - \mu_{jg}^{(t+1)})^2 - 2([\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_j - \mu_{jg}^{(t+1)})\beta_{jg}^{(t+1)} + E_{1ijg}^{(t)}(\beta_{jg}^2)^{(t+1)} \right]$$

and

$$b_g^{(t+1)} = \frac{1}{n_g^{(t)}(p-q_g)} \sum_{i=1}^{n} \hat{z}_{ig}^{(t)} \sum_{k=q_g+1}^{p} \left[EA_{3ig}^{(t)}[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_k^2 + EA_{3ig}^{(t)}(\mu_{kg}^2)^{(t+1)} + EA_{1ig}^{(t)}(\beta_{kg}^2)^{(t+1)} \right.$$
$$\left. -2EA_{3ig}^{(t)}[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_k\mu_{kg}^{(t+1)} - 2[\boldsymbol{\Gamma}_g'^{(t)}\boldsymbol{x}_i]_k\beta_{kg}^{(t+1)} - 2\mu_{kg}^{(t+1)}\beta_{kg}^{(t+1)} \right],$$

respectively.

The q_j-dimensional concentration parameter $\boldsymbol{\Omega}_g$ and index parameter $\boldsymbol{\lambda}_g$ are estimated by maximizing the function

$$q_{jg}(\Omega_{jg}, \lambda_{jg}) = -\log K_{\lambda_{jg}}(\Omega_{jg}) + (\lambda_{jg}-1)\frac{\sum_{i=1}^{n}\hat{z}_{ig}E_{4ijg}}{n_g} - \frac{\Omega_{jg}}{2}\left(\sum_{i=1}^{n}\hat{z}_{ig}E_{1ijg} + \sum_{i=1}^{n}z_{ig}E_{3ijg}\right).$$

This leads to

$$\lambda_{jg}^{(t+1)} = \frac{\sum_{i=1}^{n}\hat{z}_{ig}E_{4ijg}}{n_g}\lambda_{jg}^{(t)}\left[\frac{\partial}{\partial v}\log K_v(\Omega_{jg}^{(t)})|_{v=\lambda_{jg}^{(t)}}\right]^{-1}$$

and

$$\Omega_{jg}^{(t+1)} = \Omega_{jg}^{(t)} - \left[\frac{\partial}{\partial v}q_{jg}(v, \lambda_{jg}^{(t+1)})|_{v=\Omega_{jg}^{(t)}}\right]\left[\frac{\partial^2}{\partial v^2}q_{jg}(v, \lambda_{jg}^{(t+1)})|_{v=\Omega_{jg}^{(t)}}\right]^{-1}.$$

The univariate parameters ω_{0g} and λ_{0g} are estimated following Browne and McNicholas (2015a).

3. CM-step 2: To update the component eigenvector matrices $\boldsymbol{\Gamma}_g$, our goal is to minimize the matrix trace function

$$f(\boldsymbol{\Gamma}_g) = \frac{1}{2}\,\mathrm{Tr}\left(\sum_{i=1}^{n}\hat{z}_{ig}\boldsymbol{x}_i\boldsymbol{x}_i'\boldsymbol{\Gamma}_g\hat{\boldsymbol{\Phi}}_g\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]\boldsymbol{\Gamma}_g'\right) - \mathrm{Tr}\left(\sum_{i=1}^{n}\hat{z}_{ig}\boldsymbol{\Phi}_g^{-1}(\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]\boldsymbol{\mu}_g + \boldsymbol{\beta}_g)\boldsymbol{x}_i'\boldsymbol{\Gamma}_g\right) + \text{constant}.$$

We follow Kiers (2002) and Browne and McNicholas (2014) by using a majorization function for the minimization of $f(\boldsymbol{\Gamma}_g)$ and it takes the form

$$f(\boldsymbol{\Gamma}_g) \leq \text{constant} + \text{Tr}\left(\boldsymbol{F}_t \boldsymbol{\Gamma}_g\right),$$

where

$$\boldsymbol{F}_t = \sum_{i=1}^{n} \left(-\hat{z}_{ig}^{(t)} (\boldsymbol{\Phi}_g^{-1})^{(t+1)} (\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]^{(t)} \boldsymbol{\mu}_g^{(t+1)} + \boldsymbol{\beta}_g^{(t+1)}) \boldsymbol{x}_i' \right) \\ + \sum_{i=1}^{n} \left(\hat{z}_{ig}^{(t)} \boldsymbol{x}_i \boldsymbol{x}_i' \boldsymbol{\Gamma}_g \mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]^{(t)} (\boldsymbol{\Phi}_g^{-1})^{(t+1)} - \hat{z}_{ig}^{(t)} \alpha_{ig}^{(t+1)} \boldsymbol{x}_i \boldsymbol{x}_i' \boldsymbol{\Gamma}_g \right),$$

$\boldsymbol{\Phi}_g^{(t+1)} = \text{diag}(\phi_{1g}^{(t+1)}, \phi_{2g}^{(t+1)}, \ldots, \phi_{qgg}^{(t+1)}, b_g^{(t+1)} I_{p-d})$, and α_{ig} is the largest value of the diagonal matrix $(\mathbb{E}[\boldsymbol{\Delta}_{\frac{1}{W_{ig}}}]^{(t)} \boldsymbol{(\Phi}_g^{-1})^{(t+1)})$. Suppose we obtain the singular value decomposition

$$-F_t = \boldsymbol{P}_t \boldsymbol{B}_t \boldsymbol{R}_t' \quad \text{and} \quad \boldsymbol{\Gamma}_g^{(t+1)} = \boldsymbol{R}_t \boldsymbol{P}_t'.$$

4. Convergence criterion: The convergence of our MCECM algorithm is determined using a criterion based on the Aitken acceleration (Chapter 2.5.3).

3.2.3 Model Identifiability

Before investigating the identifiability of our mixture of joint generalized hyperbolic models (MJGHM), it is convenient to rewrite the model density as

$$f(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\Gamma},\boldsymbol{\phi},b,\boldsymbol{\Omega},\boldsymbol{\lambda},\omega_0,\lambda_0) = \prod_{j=1}^{q} \int_0^\infty \rho_1([\boldsymbol{\Gamma}'\boldsymbol{x} - \boldsymbol{\mu} - \boldsymbol{\Delta}_w \boldsymbol{\beta}]_j | \boldsymbol{0}, \phi_j w_j) h_W(w_j|\Omega_j,1,\lambda_j) dw_j$$

$$\times \int_0^\infty \prod_{k=q+1}^{p} \rho_1([\boldsymbol{\Gamma}'\boldsymbol{x} - \boldsymbol{\mu} - a\boldsymbol{\beta}]_k | \boldsymbol{0}, b \times a) h_A(a|\omega_0,1,\lambda_0) da$$

$$= \prod_{j=1}^{q} \left[\frac{\Omega_j + \phi_j^{-1}([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j)^2}{\Omega_j + \beta_j^2 \phi_j^{-1}} \right]^{\frac{\lambda_j - \frac{1}{2}}{2}} \frac{K_{\lambda_j - \frac{1}{2}}\left(\sqrt{\left[\Omega_j + \beta_j^2 \phi_j^{-1}\right]\left[\Omega_j + \phi_j^{-1}\left([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j\right)^2\right]} \right)}{(2\pi)^{\frac{1}{2}} \phi_j^{\frac{1}{2}} K_{\lambda_j}(\Omega_j) \exp\left\{ -\frac{([\boldsymbol{\Gamma}'\boldsymbol{x}]_j - \mu_j)\beta_j}{\phi_j} \right\}}$$

$$\times \left[\frac{\omega_0 + b^{-1} \sum_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)^2}{\omega_0 + b^{-1} \sum_{k=q+1}^{p} \beta_k^2} \right]^{\frac{(\lambda_0 - \frac{p-q}{2})}{2}} \frac{K_{\lambda_0 - \frac{p-q}{2}}\left(\sqrt{\left[\omega_0 + b^{-1} \sum_{k=q+1}^{p} \beta_k^2\right]\left[\omega_0 + b^{-1} \sum_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)^2\right]} \right)}{(2\pi)^{\frac{p-q}{2}} b^{\frac{p-q}{2}} K_{\lambda_0}(\omega_0) \exp\left\{ -\frac{\sum_{k=q+1}^{p} ([\boldsymbol{\Gamma}'\boldsymbol{x}]_k - \mu_k)\beta_k}{b} \right\}}.$$

The identifiability of the MJGHM depends on the identifiability of the mixture of univariate generalized hyperbolic distributions which has been proved in Browne and McNicholas (2015a). In Proposition 1, we extend the results in Browne and McNicholas (2015a), and show that the mixture of joint generalized hyperbolic models (MJGHM) is identifiable assuming correct choice of q_g $(g = 1, \dots, G)$.

Theorem 3.2.1. *Let $\boldsymbol{\Sigma}$ be a square, symmetric real-valued $p \times p$ matrix with p linearly independent eigenvectors. Then there exists a symmetric diagonal decomposition*

$$\boldsymbol{\Sigma} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}',$$

where the columns of $\boldsymbol{Q}$ are the orthogonal and normalized eigenvectors of $\boldsymbol{\Sigma}$, and $\boldsymbol{\Lambda}$ is the diagonal matrix whose entries are the eigenvalues of $\boldsymbol{\Sigma}$. Further, all entries of $\boldsymbol{Q}$ are real and we have $\boldsymbol{Q}^{-1} = \boldsymbol{Q}'$.

Proposition 3.2.2. *The joint generalized hyperbolic models generate identifiable finite mixtures assuming the correct choice of q_g $(g = 1, \dots, G)$.*

Proof. Similar to Tortora *et al.* (2014), we consider moving the amount t in a direction $\boldsymbol{z}$, setting $\boldsymbol{x} = t\boldsymbol{z}$. If $\boldsymbol{z}$ is equal to the k^{th} eigenvector $(k = 1, \dots q)$ then the density reduces to

$$c_k \left[\frac{\Omega_k + \phi_k^{-1}([t-\mu_k)^2}{\Omega_k + \beta_k^2\phi_k^{-1}}\right]^{\frac{\lambda_k - \frac{1}{2}}{2}} \frac{K_{\lambda_k - \frac{1}{2}}\left(\sqrt{\left[\Omega_k + \beta_k^2\phi_k^{-1}\right]\left[\Omega_k + \phi_k^{-1}(t-\mu_k)^2\right]}\right)}{(2\pi)^{\frac{1}{2}}\phi_k^{\frac{1}{2}} K_{\lambda_k}(\Omega_k)\exp\{-\frac{(t-\mu_k)\beta_k}{\phi_k}\}},$$

where

$$c_k = \prod_{j=1, j\neq k}^{q} \left[\frac{\Omega_j + \phi_j^{-1}(\mu_j)^2}{\Omega_j + \beta_j^2 \phi_j^{-1}} \right]^{\frac{\lambda_j - \frac{1}{2}}{2}} \frac{K_{\lambda_j - \frac{1}{2}} \left(\sqrt{\left[\Omega_j + \beta_j^2 \phi_j^{-1}\right] \left[\Omega_j + \phi_j^{-1} (\mu_j)^2\right]} \right)}{(2\pi)^{\frac{1}{2}} \phi_j^{\frac{1}{2}} K_{\lambda_j}(\Omega_j) \exp\{\frac{(\mu_j)\beta_j}{\phi_j}\}}$$

$$\times \left[\frac{\bar{\Omega} + b^{-1} \sum\limits_{d=q+1}^{p} (\mu_d)^2}{\bar{\Omega} + b^{-1} \sum\limits_{d=q+1}^{p} \beta_d^2} \right]^{\frac{(\bar{\lambda} - \frac{p-q}{2})}{2}} \frac{K_{\bar{\lambda} - \frac{p-q}{2}} \left(\sqrt{\left[\bar{\Omega} + b^{-1} \sum\limits_{d=q+1}^{p} \beta_d^2\right] \left[\bar{\Omega} + b^{-1} \sum\limits_{d=q+1}^{p} (\mu_d)^2\right]} \right)}{(2\pi)^{\frac{p-q}{2}} b^{\frac{p-q}{2}} K_{\bar{\lambda}}(\bar{\Omega}) \exp\left\{ \frac{\sum_{d=q+1}^{p} (\mu_d)\beta_d}{b} \right\}}.$$

Therefore, the density is proportional to

$$f(t|\boldsymbol{\theta}) \propto \prod_{j=1}^{q} \left[\frac{\Omega_j + \phi_j^{-1}([t - \mu_j)^2}{\Omega_j + \beta_j^2 \phi_j^{-1}} \right]^{\frac{\lambda_j - \frac{1}{2}}{2}} \frac{K_{\lambda_j - \frac{1}{2}} \left(\sqrt{\left[\Omega_j + \beta_j^2 \phi_j^{-1}\right] \left[\Omega_j + \phi_j^{-1} (t - \mu_j)^2\right]} \right)}{(2\pi)^{\frac{1}{2}} \phi_j^{\frac{1}{2}} K_{\lambda_j}(\Omega_j) \exp\left\{-\frac{(t-\mu_j)\beta_j}{\phi_j}\right\}}$$

$$\times \left[\frac{\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} (t - \mu_k)^2}{\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} \beta_k^2} \right]^{\frac{(\lambda_0 - \frac{p-q}{2})}{2}} \frac{K_{\lambda_0 - \frac{p-q}{2}} \left(\sqrt{\left[\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} \beta_k^2\right] \left[\omega_0 + b^{-1} \sum\limits_{k=q+1}^{p} (t - \mu_k)^2\right]} \right)}{(2\pi)^{\frac{p-q}{2}} b^{\frac{p-q}{2}} K_{\lambda_0}(\omega_0) \exp\left\{ \frac{\sum_{k=q+1}^{p} (y-\mu_k)\beta_k}{b} \right\}}.$$

Browne and McNicholas (2015a) state that "if the parameterizations are one-to-one, then if one parameterization is shown to be identifiable, the others are identifiable as well." Accordingly, similar to Browne and McNicholas (2015a), we let $\delta_j = \beta_j/\phi_j$, $\alpha_j = \sqrt{\Omega_j/\phi_j + \beta_j^2/\phi_j^2}$ and $\kappa_j = \sqrt{\phi_j \Omega_j}$. For large z, the Bessel function can approximated by

$$K_{\lambda(z)} = \sqrt{\frac{\pi}{2z}} e^{-z} \left[1 + O\left(\frac{1}{z}\right) \right],$$

and the characteristic function for a normal variance-mean density can be written as

$$\varphi_X(t) = \exp\{it\mu\} M_W \left(\beta t i - \frac{1}{2}\sigma^2 t^2 | \lambda, \Omega \right).$$

Therefore, the characteristic function for the joint generalized hyperbolic models can be written as

$$
\begin{aligned}
&\varphi_{\boldsymbol{X}}(\boldsymbol{v}) = \\
&\prod_{j=1}^{q} \exp\{i|\boldsymbol{\Gamma}'\boldsymbol{v}|_j\mu_j\} \left[1 + \frac{\phi_j|\boldsymbol{\Gamma}'\boldsymbol{v}|_j^2 - 2\beta_j|\boldsymbol{\Gamma}'\boldsymbol{v}|_j i}{\Omega_j}\right]^{-\lambda_j/2} \frac{K_{\lambda_j}\left(\sqrt{\Omega_j\left[\Omega_j + (\phi_j|\boldsymbol{\Gamma}'\boldsymbol{v}|_j^2 - 2\beta_j|\boldsymbol{\Gamma}'\boldsymbol{v}|_j i)\right]}\right)}{K_{\lambda_j}(\Omega_j)} \\
&\times \exp\{i|\boldsymbol{\Gamma}'\boldsymbol{v}|_2'\boldsymbol{\mu}_2\} \left[1 + \frac{b|\boldsymbol{\Gamma}'\boldsymbol{v}|_2'|\boldsymbol{\Gamma}'\boldsymbol{v}|_2 - 2\boldsymbol{\beta}_2'|\boldsymbol{\Gamma}'\boldsymbol{v}|i}{\omega_0}\right]^{-\lambda_0/2} \\
&\times \frac{K_{\lambda_0}\left(\sqrt{\omega_0\left[\omega_0 + (b|\boldsymbol{\Gamma}'\boldsymbol{v}|_2'|\boldsymbol{\Gamma}'\boldsymbol{v}|_2 - 2\boldsymbol{\beta}_2'|\boldsymbol{\Gamma}'\boldsymbol{v}|_2 i)\right]}\right)}{K_{\lambda_0}(\omega_0)},
\end{aligned}
$$

where $|\boldsymbol{\Gamma}'\boldsymbol{v}|_2$ is the $(q+1)$th to pth columns of $|\boldsymbol{\Gamma}'\boldsymbol{v}|$, $\boldsymbol{\mu}_2 = (\mu_{q+1}, \ldots, \mu_p)'$, and $\boldsymbol{\beta}_2 = (\beta_{q+1}, \ldots, \beta_p)'$. Now if we consider moving t in the direction $\boldsymbol{z}$, $\boldsymbol{v} = t\boldsymbol{z}$ and for large t, the characteristic function is

$$
\begin{aligned}
&\varphi_{\boldsymbol{X}}(\boldsymbol{v} = t\boldsymbol{z}) \\
&\propto \exp\left\{it\sum_{j=1}^{p}|\boldsymbol{\Gamma}'\boldsymbol{z}|_j\mu_j - t\sum_{j=1}^{q}\kappa_j||\boldsymbol{\Gamma}'\boldsymbol{z}|_j| - t\sqrt{b\omega_0}\sum_{k=q+1}^{p}|\boldsymbol{\Gamma}'\boldsymbol{z}|_k - \log(t)\left(\sum_{j=1}^{q}\lambda_j I(|\boldsymbol{\Gamma}'\boldsymbol{z}|_j \neq 0) + \lambda_0\right) + O(1)\right\} \\
&\propto \exp\left\{it\boldsymbol{z}'\boldsymbol{\Gamma}\boldsymbol{\mu} - t\left(\sum_{j=1}^{q}\kappa_j||\boldsymbol{\Gamma}'\boldsymbol{z}|_j| + \sqrt{b\omega_0}\sum_{k=q+1}^{p}|\boldsymbol{\Gamma}'\boldsymbol{z}|_k\right) - \log(t)\left(\sum_{j=1}^{q}\lambda_j I(|\boldsymbol{\Gamma}'\boldsymbol{z}|_j \neq 0) + \lambda_0\right) + O(1)\right\}.
\end{aligned}
$$

From Yakowitz and Spragins (1968), there exists $\boldsymbol{z}$ such that the tuple

$$
\left(\boldsymbol{z}'\boldsymbol{\Gamma}\boldsymbol{\mu}, \sum_{j=1}^{q}\kappa_j||\boldsymbol{\Gamma}'\boldsymbol{z}|_j| + \sqrt{b\omega_0}\sum_{k=q+1}^{p}|\boldsymbol{\Gamma}'\boldsymbol{z}|_k, \sum_{j=1}^{q}\lambda_j I(|\boldsymbol{\Gamma}'\boldsymbol{z}|_j \neq 0) + \lambda_0\right)
$$

is pairwise distinct for all $g = 1, \ldots, G$ and reduces to a mixture of univariate hyperbolic distributions, which has been proved identifiable in Browne and McNicholas (2015a). □

3.2.4 Computational Aspects

We start with ten random initializations of the algorithm by randomly assigning each observation to one of the G components. After fitting models for all values of G and q_g, we select the model with the lowest BIC. We compare our approach with the classic Gaussian parsimonious clustering models (GPCM) from R package **mixture** (Browne *et al.*, 2015) and high-dimensional data clustering (HDDC) approach from R package **HDclassif** (Bergé *et al.*, 2012). It is worth noting that the MJGHM-HDClust model proposed here does not need to numerically invert covariance matrices, which often fails for singularity reasons. During our experiments, we found the classical Gaussian parsimonious clustering models (GPCM) from R package **mixture** (Browne *et al.*, 2015) do not always converge, especially for high-dimensional data. We also compare with the parsimonious Gaussian mixture models from R package **pgmm** (McNicholas *et al.*, 2011) in our real data application.

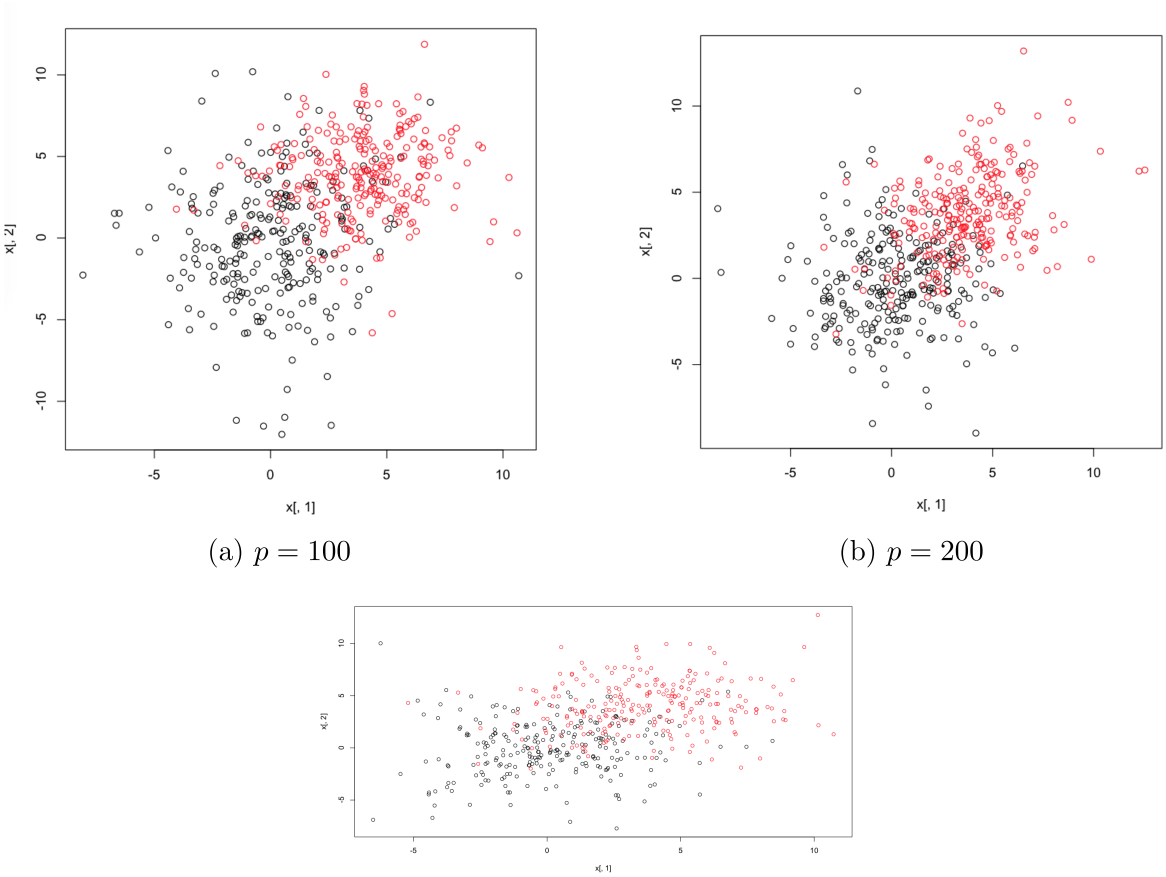
3.3 Simulation Studies

To illustrate the accuracy of the proposed MJGHM-HDClust model, we perform simulation experiments on three datasets. Each one of the three datasets consists of 500 data points (i.e., $n = 500$) with different dimensionality ($p_1 = 100$, $p_2 = 200$, $p_3 = 500$). For each data set, two Gaussian densities are simulated in $\mathbb{R}^p$ with the mixing proportions $\pi_1 = \pi_2 = 0.5$. The BIC is used for selecting the best model and ARI can be calculated because the true partitions are known. Thus, a comparison of approaches MJGHM-HDClust, HDDC (via R package **HDclassif**), and GPCM (via R package **mixture**) is carried out. We present the model structures selected by the lowest BIC values and the ARI values associated with the clustering results (Table 3.1). With the MJGHM-HDClust approach, the BIC selected $G = 2$ components in all cases, which is correct. The ARI values for selected models are very high for all p, even when p is as large as the sample size n. On the other hand, for the HDDC and GMM approaches, the BIC failed to choose the correct number of groups in all cases, and the ARI value decreases as p increases. In addition, the GPCMs do not always converge for $p = 200$ and $p = 500$. It is notable that the MJGHM-HDClust approach outperformed

HDDC and GPCM in all cases, even though the samples were generated from Gaussian mixture models. We have also given plots of the first two dimensions of the transformed spaces of the data (Figure 3.2). The clusters are very well separated in these projections when compared to the original space (Figure 3.1).

Table 3.1: Model selection, and ARI values for the MJGHM-HDClust, HDDC, and GPCM approaches for different values of p.

	MJGHM-HDClust			HDDC			GPCM		
	$p = 100$	$p = 200$	$p = 500$	$p = 100$	$p = 200$	$p = 500$	$p = 100$	$p = 200$	$p = 500$
G	2	2	2	3	5	5	3	3	5
q_g/Σ_g	$(2, 3)$	$(15, 12)$	$(23, 15)$	$(14, 32, 22)$	$(14, 23, 18, 3, 18)$	$(4, 68, 41, 9, 24)$	VII	VII	VII
BIC	282252	482159	727036	240254	472508	794696	234448	491205	694239
ARI	0.97	0.97	0.95	0.43	0.25	0.21	0.67	0.47	0.33

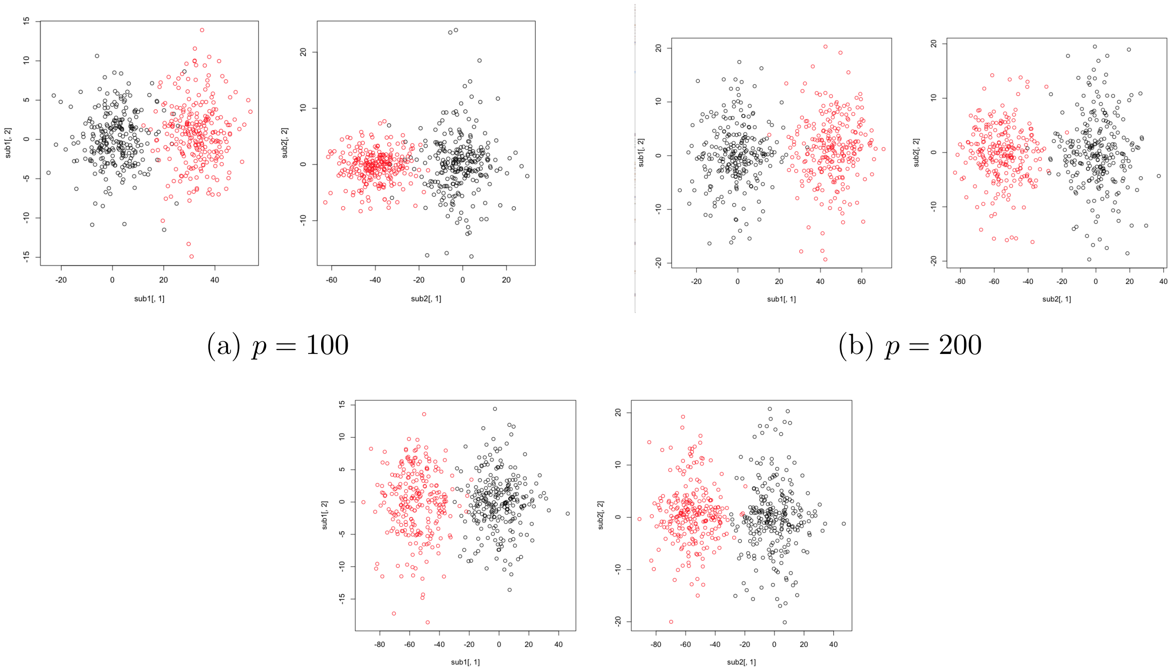


(a) $p = 100$

(b) $p = 200$

(c) $p = 500$

Figure 3.1: Plots of the first two dimensions from the original data sets for different values of p.



(a) $p = 100$

(b) $p = 200$

(c) $p = 500$

Figure 3.2: Plots of the first two dimensions from the transformed spaces (MJGHM-HDClust).

3.4 Real Data

3.4.1 Italian Wines

The Italian wines data (Forina *et al.*, 1986) has been widely used in literature. The data set includes 178 wines and each wine belongs to one of the three types: Barolo, Grignolino or Barbera. The chemical and physical properties of each wine are listed in Table 3.2. This data set is available from the R package **pgmm** (McNicholas *et al.*, 2011). The MJGHM-HDClust models were fitted to these data for $G = 1, 2, \ldots, 4$ and $q_g = 2, 3, 5, 8, 10$. The lowest BIC occurs at the 3-cluster, $\boldsymbol{q} = (8, 5, 3)$ model. The BIC value is 16984.

Table 3.2: Twenty-seven chemical and physical properties for the Italian wines.

Item	Property	Item	Property
1	Alcohol	15	Total phenols
2	Suger-free extract	16	Flavanoids
3	Fixed acidity	17	Nonflavanoid phenols
4	Tartaric acid	18	Proanthocyanins
5	Malic acid	19	Color intensity
6	Uronic acids	20	Hue
7	pH	21	$\text{OD}_{280}/\text{OD}_{315}$ of diluted wines
8	Ash	22	$\text{OD}_{280}/\text{OD}_{315}$ of flavanoids
9	Alcalinity of ash	23	Glycerol
10	Potassium	24	2,3-butanediol
11	Calcium	25	Total nitrogen
12	Magnesium	26	Proline
13	Phosphate	27	Methanol
14	Chloride		

A summary of the best models from the MJGHM-HDClust, HDDC, GPCM, and the PGMM approaches is shown in Table 3.3. The MJGHM-HDClust approach and PGMM approach yield excellent clustering results (ARI= 0.95) and outperform the chosen two-component Gaussian mixture models and the HDDC approach. There are only three misclassified wines in comparison with the true labels (Table 3.4). It is worth noting that MJGHM-HDClust is one of the few methods in the literature using all 27 variables of the wine data set and yielding excellent clustering results.

Table 3.3: A comparison of four different approaches.

Approach	G	Model	BIC	ARI
MJGHM-HDClust	3	$(q_1, q_2, q_3) = (8, 5, 3)$	16984	0.95
HDDC	2	$(q_1, q_2) = (2, 1)$	12657	0.41
PGMM	3	$q = 4$ & CUU	11428	0.96
GPCM	2	EVE	12068	0.49

Table 3.4: Cross-tabulation of the type of wines and predicted classifications.

Class/True	Barolo	Grignolino	Barbera	ARI
1	59	2	0	
2	0	68	0	0.95
3	0	1	48	

3.4.2 Breast Cancer Diagnostic Data Set

The breast cancer diagnostic data was originally reported on by Street *et al.* (1993). They give data on 569 cases of breast tumours: 357 benign and 212 malignant. Ten real-valued features are computed for each cell nucleus (Table 3.5). The mean, standard error, and the "worst" or the largest of these features were computed for each image, resulting in 30 attributes. For instance, attribute 3 is mean radius, attribute 13 is the standard error of radius and attribute 23 is the worst radius. The MJGHM-HDClust models were fitted to these data for $G = 1, 2, \ldots, 4$ and $q_g = 2, 3, 5, 8, 10$. The lowest BIC occurs at the 2-cluster, $\boldsymbol{q} = (8, 5)$ model. The BIC value is 20432. Figure 3.3 shows the minimum BIC for two-component models for each pair $(q_1,\, q_2)$.

A summary of the best models from the MJGHM-HDClust, HDDC, PGMM and GPCM approaches is shown in Table 3.6. The respective classification results reveal that the chosen two-component MJGHM-HDClust yields relatively good clustering result (ARI$=$ 0.70) and

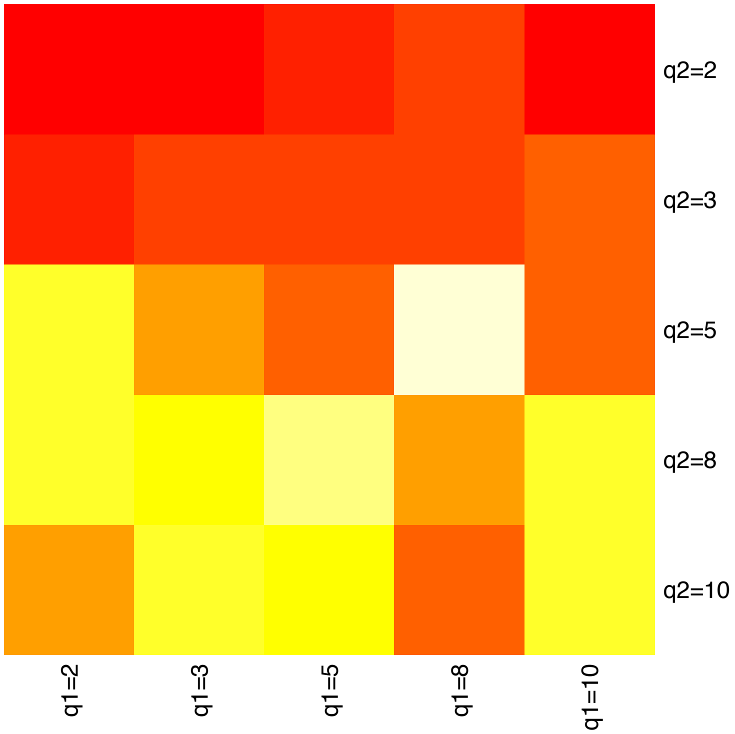


Figure 3.3: A heat map representation of the minimum BIC value for each value of (q_1, q_2) where the minimum is taken over the two-component models.

outperforms the other approaches we compared with. Moreover, the MJGHM-HDClust gives us the correct number of components compared to the other approaches.

Plots of the first three dimensions of the transformed spaces for each component with group labels (Figure 3.4) indicate that the groups are well separated in the latent space.

Table 3.5: Ten features are considered for breast cancer diagnosis.

Item	Feature	Item	Feature
1	Radius	6	Compactness
2	Texture Intensity	7	Concavity
3	Perimeter	8	Concave Points
4	Area	9	Symmetry
5	Smoothness acid	10	Fractal Dimension

Table 3.6: A comparison of four different approaches.

Approach	G	Model	BIC	ARI
MJGHM-HDClust	2	$(q_1, q_2) = (8, 5)$	20432	0.70
HDDC	4	$(q_1, q_2, q_3, q_4) = (4, 3, 3, 3)$	26673	0.09
PGMM	4	$q = 4$ & UUU	12083	0.35
GPCM	4	VEE	24367	0.22

Table 3.7: Cross-tabulation of type of the tumours and predicted classifications.

Class/True	Benign	Malignant	ARI
1	30	343	0.70
2	164	15	

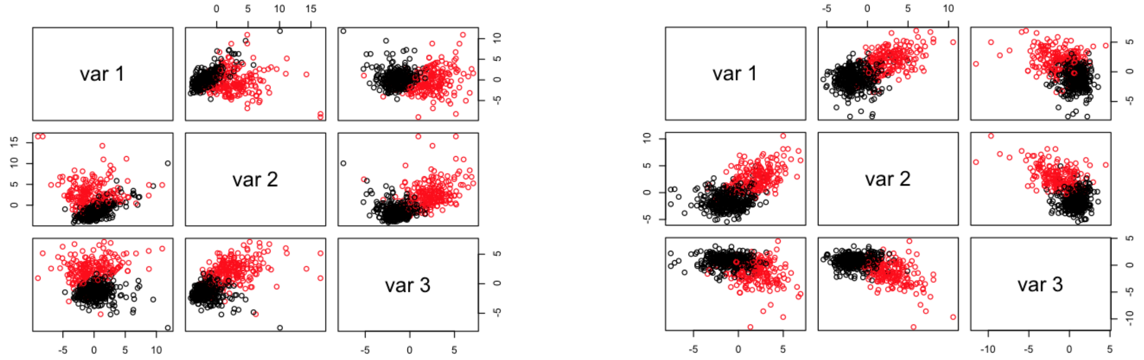


Figure 3.4: Left: Plot of the first three dimensions of the transformed spaces for Group 1. Right: Plot of the first three dimensions of the transformed spaces for Group 2.

3.5 Discussion

We introduce the asymmetric clustering for high-dimensional data via a mixture of joint generalized hyperbolic models, referred to as the MJGHM-HDClust, for model-based clustering. The MJGHM-HDClust approach proposed here does not need to numerically invert covariance matrices, which makes it ideal for high-dimensional data clustering. We develop the MJGHM-HDClust approach based on the generalized hyperbolic distributions which represent perhaps the most flexible in a recent series of alternatives to the Gaussian mixture model for clustering and classification. Parameter estimation is carried out using a multi-cycle ECM algorithm and Bayesian information criterion is used for model selection.

Comparing the MJGHM-HDClust, HDDC, PGMM, and GPCM approaches yielded some interesting results. Two real data sets were considered for illustration: the Italian wine data and the breast cancer diagnostic data. The MJGHM-HDClust approach was the only approach giving great classification performance in both cases. The PGMM approach gave excellent classification results for the Italian wine data but performed poorly when fitted to the breast cancer diagnostic data.

Although illustrated for clustering, the MJGHM-HDClust approach can also be applied for semi-supervised classification and discriminant analysis.

Chapter 4

Mixture of Latent Trait Models via the Contaminated Gaussian Distributions

4.1 Introduction

In this chapter, we explore the possibility of discovering "extreme patterns" of binary data by drawing ideas from the mixture of contaminated Gaussian distributions (Punzo and McNicholas, 2016). A contaminated Gaussian distribution (Figure 4.1) is a two-component Gaussian mixture in which one of the components – with a large prior probability – represents normal observations, and the other – with a small prior probability, the same mean and an inflated covariance matrix – represents the extreme points (Aitkin and Wilson, 1980). For continuous multivariate random variables, the mixture of contaminated Gaussian distributions accommodates outlying observations, spurious observations, or noise. Our goal is to automatically detect patterns with lower probability of appearing while clustering in multi-dimensional binary data, which we collectively refer to as extreme patterns. We propose a mixture of latent trait models which assumes the low dimensional continuous latent variable comes from a contaminated Gaussian distribution and therefore picks up extreme patterns in the observed binary data while clustering.

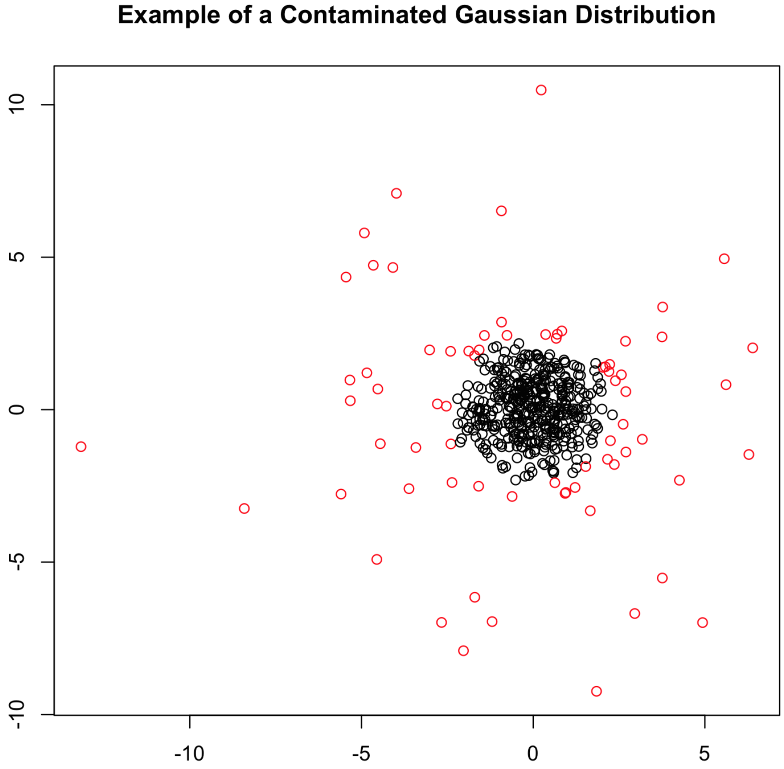


Figure 4.1: Plot of data from a contaminated Gaussian distribution.

The model-based clustering framework is outlined in Section 4.2, and an expectation-conditional maximization (ECM) algorithm for parameter estimation is outlined. Application on artificial and real data are presented in Section 4.3 and the chapter concludes with some discussion in Section 4.4.

4.2 The Mixture of Latent Trait Model via Contaminated Gaussian Distributions

4.2.1 The Latent Trait Models via Contaminated Gaussian Distributions

Latent trait models via a contaminated Gaussian distribution can be used to model a set of n multivariate binary (categorical) observations. We assume that there is a D dimensional continuous latent variable $Y_i = \{y_{i1}, \ldots, y_{iD}\}$ underlying the behaviour of the M categorical response variables within each observation. The latent trait model via a contaminated Gaussian distribution assumes that,

$$p(\boldsymbol{x}_i|\boldsymbol{\Theta}) = \int_{\boldsymbol{y}_i} p(\boldsymbol{x}_i|\boldsymbol{y}_i; \boldsymbol{\alpha}, \boldsymbol{w})p(\boldsymbol{y}_i|\tau, \eta)d\boldsymbol{y}_i, \tag{4.1}$$

where the conditional distribution of $\boldsymbol{x}_i$ given $\boldsymbol{y}_i$ is

$$p(\boldsymbol{x}_i|\boldsymbol{y}_i) = \prod_{m=1}^{M} (p_m(\boldsymbol{y}_i))^{x_{im}}(1 - p_m(\boldsymbol{y}_i))^{(1-x_{im})},$$

and the response function is a logistic function

$$p_m(\boldsymbol{y}_i) = p(x_{im} = 1|\boldsymbol{y}_i) = \frac{1}{1 + \exp(-(\alpha_m + \boldsymbol{w}'_m\boldsymbol{y}_i))},$$

where α_m is the intercept and $\boldsymbol{w}_m$ are the slope parameters in the logistic function. The continuous latent variable $\boldsymbol{Y}_i$ comes from a contaminated Gaussian distribution

$$\begin{aligned}
&p(\boldsymbol{y}_i; \tau, \eta) = \tau p(\boldsymbol{y}_i \mid c_i = 1) + (1 - \tau)p(\boldsymbol{y}_i \mid c_i = 0),\\
&\boldsymbol{Y}_i \mid c_i = 1 \sim \text{MVN}(\boldsymbol{0}, \boldsymbol{I}),\\
&\boldsymbol{Y}_i \mid c_i = 0 \sim \text{MVN}(\boldsymbol{0}, \eta\boldsymbol{I}),
\end{aligned}$$

where $\tau \in (0.5, 1)$ is the prior probability of a randomly chosen $\boldsymbol{Y}_i$ coming from MVN$(\boldsymbol{0}, \boldsymbol{I})$ and η denotes the degree of contamination. Because of the assumption $\eta > 1$, it can be interpreted as the increase in variability due to the extreme values.

The mixture of latent trait model via contaminated Gaussian distributions (MLTCG) is a mixture latent trait model and the latent variables are random variables from a contaminated Gaussian distribution

$$p(\boldsymbol{x}_i) = \sum_{g=1}^{G} \pi_g \int_{\boldsymbol{y}_{ig}} p(\boldsymbol{x}_i | \boldsymbol{y}_{ig}; \boldsymbol{\alpha}_g, \boldsymbol{w}_g) p(y_{ig} | \tau_g, \eta_g) d\boldsymbol{y}_{ig}, \tag{4.2}$$

where $\pi_g, \boldsymbol{\alpha}_g, \boldsymbol{w}_g, \tau_g, \eta_g$ are component specific parameters.

4.2.2 Model Identifiability

The identifiability of our model depends on the identifiability of the latent trait part as well as the identifiability of the mixture of contaminated Gaussian distributions. The identifiability of the mixture of contaminated Gaussian distributions has been proved by Punzo and McNicholas (2016) and Knott and Bartholomew (1999) gives a detailed explanation of model identifiability in the latent trait models context.

4.2.3 Model Fitting

To fit the MLTCG model, we adopt the ECM algorithm (Section 2.5.3). In this case, there are two sources of incomplete data: one arises from the fact that we do not know the component labels $\boldsymbol{z}_i$ and the other arises from the fact that we do not know whether an observation in group g is normal or extreme. To denote the second source of missing data, we use $\boldsymbol{c}_i = (c_{i1}, c_{i2}, \dots c_{iG})$ where $c_{ig} = 1$ if observation i in group g is normal and $c_{ig} = 0$ if observation i in group g is extreme. Therefore, the complete-data log-likelihood can be written

$$l_c = \sum_{i=1}^{n}\sum_{g=1}^{G} z_{ig}\Bigg(\log \pi_g + c_{ig}\log p(\boldsymbol{x}_i|\boldsymbol{y}_{ig1};\boldsymbol{\alpha}_g,\boldsymbol{w}_g) + c_{ig}\log\tau_g + c_{ig}\log p(\boldsymbol{y}_{ig1};\boldsymbol{0},\boldsymbol{I})$$

$$+ (1-c_{ig})\log(1-\tau_g) + (1-c_{ig})\log p(\boldsymbol{x}_i|\boldsymbol{y}_{ig2};\boldsymbol{\alpha}_g,\boldsymbol{w}_g) + (1-c_{ig})\log p(\boldsymbol{y}_{ig2};\boldsymbol{0},\eta_g\boldsymbol{I})\Bigg). \tag{4.3}$$

The ECM algorithm iterates between three steps, an E-step and two CM-steps, until convergence. The parameter vector $\boldsymbol{\theta}_g$ is partitioned in $\boldsymbol{\theta}_g = \{\boldsymbol{\theta}_{g1},\boldsymbol{\theta}_{g2}\}$, where $\boldsymbol{\theta}_{g1} = \{\boldsymbol{\xi}_g\}$ and $\boldsymbol{\theta}_{g2} = \{\boldsymbol{\alpha}_g,\boldsymbol{w}_g,\eta_g,\pi_g\}$.

1. We estimate z_{ig} and c_{ig}

$$z_{ig}^{(t+1)} = \frac{\pi_g^{(t)}\exp(L_{ig}^{(t)})}{\sum_{g=1}^{G}\pi_g\exp(L_{ig}^{(t)})},$$
$$c_{ig}^{(t+1)} = \frac{\tau_g\exp(L(\boldsymbol{\xi}_{ig1}^{(t)}))}{\tau_g\exp(L(\boldsymbol{\xi}_{ig1}^{(t)})) + (1-\tau_g)\exp(L(\boldsymbol{\xi}_{ig0}^{(t)}))}.$$

2. We then update π_g and τ_g

$$\tau_g^{(t+1)} = \frac{\sum_{i=1}^{n} z_{ig}^{(t+1)}c_{ig}^{(t+1)}}{\sum_{i=1}^{n} z_{ig}^{(t+1)}}.$$

When the MLTCG models are used for detecting extreme patents, $(1-\tau_g)$ represents the percentage of extreme observations and the proportion of normal observations is at least equal to a pre-determined value τ_g^* (i.e., $\tau_g^* = 0.5$). In this case, we perform a numerical search of the maximum $\tau_g^{(t+1)}$ using the `optimize()` function, over the interval $(\tau_g^*, 1)$, of the function

$$\sum_{i=1}^{n} z_{ig}^{(t+1)}\left(c_{ig}^{(t+1)}\log\tau_g + (1-c_{ig}^{(t+1)})\log(1-\tau_g)\right).$$

Herein, we use this approach to update τ_g and we take $\tau_g^* = 0.5$ for $g = 1, \ldots, G$.

$$\pi_g^{(t+1)} = \frac{\sum_{i=1}^n z_{ig}^{(t+1)}}{n}.$$

3. Estimate the likelihood: We approximate the posterior density for $p(\boldsymbol{y}_{ig}|\boldsymbol{x}_i, z_{ig}^{(t+1)} = 1)$ by its variational lower bound $\underline{p}(\boldsymbol{y}_{ig}|\boldsymbol{x}_i, z_{ig}^{(t+1)} = 1, \boldsymbol{\xi}_{ig}^{(t)})$ (Section 2.4.2.1), which is a $\text{MVN}(\boldsymbol{\mu}_{ig}^{(t+1)}, \boldsymbol{\Sigma}_{ig}^{(t+1)})$ density, where

$$\begin{aligned}
\mathbb{E}(\text{Cov}(\boldsymbol{Y}_{ig})|c_{ig}^{(t+1)} = 1) &= \left[\boldsymbol{I}_D - 2\sum_{m=1}^{M} B(\xi_{img1}^{t})\boldsymbol{w}_{mg}^{(t)}\boldsymbol{w}_{mg}^{\prime(t)}\right]^{-1} =: \boldsymbol{\Sigma}_{ig1}^{(t+1)},\\
\mathbb{E}(\boldsymbol{Y}_{ig}|c_{ig}^{(t+1)} = 1) &= \boldsymbol{\Sigma}_{ig1}^{(t+1)}\left[\sum_{m=1}^{M}\left(x_{im} - \frac{1}{2} + 2B(\xi_{img1}^{(t)})\alpha_{mg}^{(t)}\right)\boldsymbol{w}_{mg}^{(t)},\right] := \boldsymbol{\mu}_{ig1}^{(t+1)},\\
\mathbb{E}(\text{Cov}(\boldsymbol{Y}_{ig})|c_{ig}^{(t+1)} = 0) &= \left[\frac{1}{\eta_g}\boldsymbol{I}_D - 2\sum_{m=1}^{M} B(\xi_{img0}^{t})\boldsymbol{w}_{mg}^{(t)}\boldsymbol{w}_{mg}^{\prime(t)}\right]^{-1} =: \boldsymbol{\Sigma}_{ig0}^{(t+1)},\\
\mathbb{E}(\boldsymbol{Y}_{ig}|c_{ig}^{(t+1)} = 0) &= \boldsymbol{\Sigma}_{ig0}^{(t+1)}\left[\sum_{m=1}^{M}\left(x_{im} - \frac{1}{2} + 2B(\xi_{img0}^{(t)})\alpha_{mg}^{(t)}\right)\boldsymbol{w}_{mg}^{(t)},\right] := \boldsymbol{\mu}_{ig0}^{(t+1)},
\end{aligned}$$

where $B(\xi_{img}^{(t)}) = (\frac{1}{2} - \sigma(\xi_{img}^{(t)}))/2\xi_{img}^{(t)}$ and $\sigma(\xi_{img}^{(t)}) = \left(1 + \exp(-\xi_{img}^{(t)})\right)^{-1}$.

4. CM steps 1: Optimize the variational parameter $\xi_{img}^{(t+1)}$. Owing to the EM formulation, each update for ξ_{img} corresponds to a monotone improvement to the posterior approximation. The updates are

$$\xi_{img1}^{2(t+1)} = \boldsymbol{w}_{mg}^{\prime(t)}\left(\boldsymbol{\Sigma}_{ig1}^{(t+1)} + \boldsymbol{\mu}_{ig1}^{(t+1)}\boldsymbol{\mu}_{ig1}^{\prime(t+1)}\right)\boldsymbol{w}_{mg}^{(t)} + 2\alpha_{mg}^{(t)}\boldsymbol{w}_{mg}^{\prime(t)}\boldsymbol{\mu}_{ig1}^{(t+1)} + \alpha_{mg}^{2(t)}$$

and

$$\xi_{img0}^{2(t+1)} = \boldsymbol{w}_{mg}^{\prime(t)}\left(\boldsymbol{\Sigma}_{ig0}^{(t+1)} + \boldsymbol{\mu}_{ig0}^{(t+1)}\boldsymbol{\mu}_{ig0}^{\prime(t+1)}\right)\boldsymbol{w}_{mg}^{(t)} + 2\alpha_{mg}^{(t)}\boldsymbol{w}_{mg}^{\prime(t)}\boldsymbol{\mu}_{ig0}^{(t+1)} + \alpha_{mg}^{2(t)}.$$

5. CM step 2:

Update parameter $\alpha_{mg}, \boldsymbol{w}_{mg}$ based on the posterior distributions corresponding to the observations in the data set:

$$\hat{\boldsymbol{w}}_{mg}^{(t+1)} = \\ -\left[2\sum_{i=1}^{n} z_{ig}^{(t+1)}\left(c_{ig}^{(t+1)}B(\xi_{img1}^{(t+1)})\mathbb{E}[\boldsymbol{y}_{ig1}\boldsymbol{y}'_{ig1}]^{(t+1)}\right)\left((1-c_{ig}^{(t+1)})B(\xi_{img0}^{(t+1)})\mathbb{E}[\boldsymbol{y}_{ig0}\boldsymbol{y}'_{ig0}]^{(t+1)}\right)\right]^{-1} \\ \times\left[\sum_{i=1}^{n} z_{ig}^{(t+1)}(x_{im}-1/2)\left(c_{ig}^{(t+1)}\hat{\boldsymbol{\mu}}_{ig1}^{(t+1)}+(1-c_{ig}^{(t+1)})\hat{\boldsymbol{\mu}}_{ig0}^{(t+1)}\right)\right],$$

where $\hat{\boldsymbol{w}}_{mg}^{(t+1)} = (\boldsymbol{w}_{mg}^{'(t+1)}, \alpha_{mg}^{(t+1)})'$, $\hat{\boldsymbol{\mu}}_{ig}^{(t+1)} = (\boldsymbol{\mu}_{ig}^{'(t+1)}, 1)'$, and

$$\mathbb{E}[\boldsymbol{y}_{igk}\boldsymbol{y}'_{igk}] = \begin{bmatrix} \boldsymbol{\Sigma}_{igk}^{(t+1)} + \boldsymbol{\mu}_{igk}^{(t+1)}\boldsymbol{\mu}_{igk}^{(t+1)} & \boldsymbol{\mu}_{igk}^{(t+1)} \\ \boldsymbol{\mu}_{igk}^{'(t+1)} & 1 \end{bmatrix}.$$

Update η_g by optimizing the following log likelihood with respect to η_g and subject to $\eta_g > 1$,

$$-\frac{d}{2}\sum_{i=1}^{n}\left(z_{ig}^{(t+1)}(1-c_{ig}^{(t+1)})\log\eta_g\right) - \frac{1}{2}\sum_{i=1}^{n} z_{ig}^{(t+1)}\left(\frac{1-c_{ig}^{(t+1)}}{\eta_g}\right)\mathbb{E}[\boldsymbol{y}_{ig0}^{'(t+1)}\boldsymbol{y}_{ig0}^{(t+1)}],$$

where $\mathbb{E}[\boldsymbol{y}_{ig0}^{'(t+1)}\boldsymbol{y}_{ig0}^{(t+1)}] = \mathrm{Tr}(\mathbb{E}[\boldsymbol{y}_{ig0}^{(t+1)}\boldsymbol{y}_{ig0}^{'(t+1)}]) = \mathrm{Tr}(\mathbb{E}[\boldsymbol{y}_{ig0}^{(t+1)}\boldsymbol{y}_{ig0}^{'(t+1)}])$.

6. Obtain the lower bound of the log likelihood at the expansion point $\boldsymbol{\xi}_{ng}$

$$\begin{aligned} L(\boldsymbol{\xi}_{ig1}^{(t+1)}) = &\sum_{m=1}^{M}\left(\log(\delta(\xi_{img1}^{(t+1)})) - \frac{\xi_{img1}^{(t+1)}}{2} - B(\xi_{img1}^{(t+1)})\xi_{img1}^{2(t+1)}\right) \\ &+ \frac{1}{2}\log|\Sigma_{ig1}^{(t+1)}| + \frac{1}{2}\boldsymbol{\mu}_{ig1}^{'(t+1)}\boldsymbol{\Sigma}_{ig1}^{-1(t+1)}\boldsymbol{\mu}_{ig1}^{(t+1)}, \\ L(\boldsymbol{\xi}_{ig0}^{(t+1)}) = &\sum_{m=1}^{M}\left(\log(\delta(\xi_{img0}^{(t+1)})) - \frac{\xi_{img0}^{(t+1)}}{2} - B(\xi_{img0}^{(t+1)})\xi_{img0}^{2(t+1)}\right) \\ &+ \frac{1}{2}\log|\Sigma_{ig0}^{(t+1)}| + \frac{1}{2}\boldsymbol{\mu}_{ig0}^{'(t+1)}\boldsymbol{\Sigma}_{ig0}^{-1(t+1)}\boldsymbol{\mu}_{ig0}^{(t+1)}, \end{aligned}$$

the $L_{ig}^{(t+1)} = \log\left(\tau_g^{(t+1)}\exp(L(\boldsymbol{\xi}_{ig1}^{(t+1)})) + (1-\tau_g^{(t+1)})\exp(L(\boldsymbol{\xi}_{ig0}^{(t+1)}))\right)$, and

$$l^{(t+1)} \approx \sum_{i=1}^{n}\log\left(\sum_{g=1}^{G}\pi_g^{(t+1)}\exp(L_{ig}^{(t+1)})\right).$$

4.3 Data Analysis

In this section, we evaluate the performance of the MLTCG model on artificial and real data sets. Particular attention will be devoted to the problem of contamination parameter recovery and model selection in simulation study, clustering results and detecting extreme patterns in the application of real data.

4.3.1 Simulation Studies

To illustrate the ability of parameter recovery for the proposed MLTCG model, we perform a simulation experiment on a 25-dimensional binary data set (i.e., $M = 25$). The observations are generated from a MLTCG model with a two-component mixture ($G = 2, \pi_1 = \pi_2 = 0.5$). The latent variables are two-dimensional multivariate Gaussian distribution. Each component consists of 80% normal patterns and 20% extreme patterns (i.e., $\tau_1 = \tau_2 = 0.8$) and degree of contamination $\eta_1 = \eta_2 = 2.5$. We choose sample sizes $n \in \{100, 200, 500\}$ and run 100 simulations for each sample. Data were fitted using $G = 2$ and $D = 2$, and starting randomly. Table 4.1 presents the value of the estimated contamination parameters and standard errors of these estimates for $n = 100, 250, 500$. The standard errors are relatively low when $n = 100$ and decrease with increasing sample size n.

Table 4.1: Estimated values for $\boldsymbol{\eta}$ and $\boldsymbol{\tau}$.

	$n = 100$		$n = 250$		$n = 500$	
	η_g (SE)	τ_g (SE)	η_g (SE)	τ_g (SE)	η_g (SE)	τ_g (SE)
Group 1	2.37 (0.25)	0.79 (0.03)	2.51 (0.10)	0.80 (0.01)	2.51 (0.08)	0.80 (0.01)
Group 2	2.45 (0.22)	0.80 (0.03)	2.50 (0.10)	0.80 (0.01)	2.50 (0.07)	0.80 (0.01)

The samples with 500 observations were fitted using $G \in \{1, 2, 3, 4, 5\}$ and $D = 2$. The left panel of Figure 4.2 displays the BIC values averaged on the 100 samples for each value of G. As shown in the right panel of Figure 4.2, on average the ARI has a maximum for $G = 2$. This result is an evidence of the BIC selecting the “best” model.

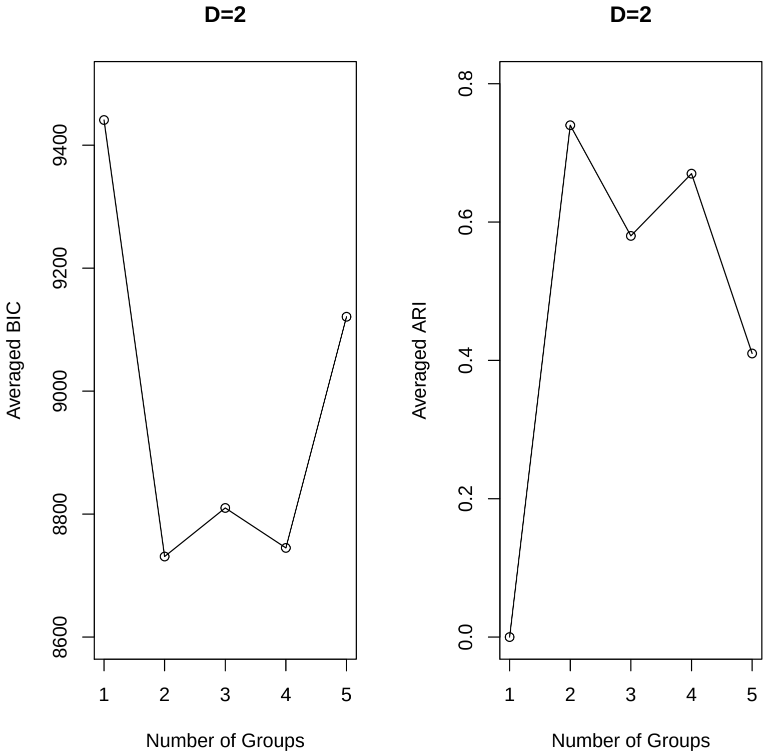


Figure 4.2: Left: BIC values averaged on the 100 samples for each value of G. Right: ARI averaged on the 100 samples for each value of G.

4.3.2 U.S. Congressional Voting

We assess the performance of the MLTCG model on the U.S. Congressional Voting data. A U.S. congressional voting data set (Lichman, 2013) has been widely used in literature (e.g., Ratanamahatana and Gunopulos, 2002; Gollini and Murphy, 2014; Tang *et al.*, 2015). This data set includes votes of 435 members of the U.S. House of Representatives on sixteen key issues in 1984 with three different type of votes: yes, no, or undecided. The representative's party is labeled as a Democrat or a Republican. The issues voted on are listed in Table 4.2. There are 11% undecided votes for issue 2 and 23% for issue 16. All other issues have less than 5% undecided votes.

Table 4.2: The issues voted on in the U.S. congressional voting data.

Item	Issue	Item	Issue
1	Handicapped Infants	9	MX Missile
2	Water Project Cost-Sharing	10	Immigration
3	Adoption of the Budget Resolution	11	Synfuels Corporation Cutback
4	Physician Fee Freeze	12	Education Spending
5	El Salvador Aid	13	Superfund Right to Sue
6	Religious Groups in Schools	14	Crime
7	Anti-Satellite Test Ban	15	Duty- Free Exports
8	Aid to Nicaraguan ‘Contras’	16	Export Administration Act/South Africa

We compare our results to those obtained by fitting a MLTA model and a MCLT model. The MLTCG were fitted to these data for $D = 1, 2, \ldots, 4$ and $G = 1, 2, \ldots, 4$. The minimum BIC (Table 4.3) occurs at the 2-components, 2 dimensional model. The BIC value is 9918.

Table 4.3: The estimated BIC for the models with $D = 1, 2, 3, 4$ and $G = 1, 2, \ldots, 4$.

	G=1	G=2	G=3	G=4
D=1	10509	11480	10282	10558
D=2	10557	**9918**	10305	10713
D=3	12176	10282	10712	11364
D=4	12258	10612	11354	12051

A summary of the best models for the MLTA, PMLTA, MCLT and MLTCG approaches is shown in Table 4.4. It can be seen that the highest ARI value (0.77) is obtained using the MLTCG model.

Table 4.4: A comparison of 4 different approaches.

	Model	G	D	BIC	$\boldsymbol{\Sigma}_g$	ARI
1	MLTA	3	1	9812	n/a	0.42
2	PMLTA	4	2	9681	n/a	0.47
3	MCLT	2	5	9597	EVI	0.64
4	MLTCG	2	2	9918	n/a	**0.77**

The classification table for group membership versus party membership for the selected model ($G = 2$, $D = 2$) is presented in Table 4.5. In comparison with the true party membership, there are only 26 misclassified representatives (i.e., 94.02% accuracy) associated with the chosen model. Group 1 consists mainly of Republican representatives, and Group 2 consists mainly of Democratic representatives. Due to the number of variables, it is difficult to know the possible presence of extreme patterns. The selected MLTCG model recognizes the presence of the two groups when we consider the normal points together with the extreme points. The advantage of our approach is that not only can we cluster in the presence of extreme patterns, but we can also identify them. When we view the results of our analysis, we see that there are 161 extreme observations, and it is not surprising that 20 out of 26 misclassified observations are considered extreme observations.

Table 4.5: Cross-tabulation of the parties and predicted classification for our chosen model ($G = 2$, $D = 2$) for the U.S. Congressional Voting data.

	Group 1	Normal/Extreme	Group 2	Normal/Extreme
Republican	7	1 6	161	118 43
Democrat	248	150 98	19	5 14

Table 4.6 and 4.7 shows the median probability $\pi_{mg}(0)$ for each of the clusters. The probabilities of positive responses for the "A" variables (yes/no vs. undecided) for the median individuals in all clusters are always high with only one exception in the normal observations in Group 1, for variable number 16, where $\pi_{16\,1n}(0) = 0.37$. Thus, the majority of representatives voted on most issues, but with a slightly higher voting rate in extreme observations on all issues. Due to the high voting rates, most probabilities given for "B" variables (yes vs. no/undecided) can be interpreted in terms of voting yes versus no.

It can be observed that the responses for the median individual in Group 1 are opposite to the ones given by the median individual in Group 2 for most issues. The extreme observations in Group 1 showed different voting behaviour on Issue 5 (El Salvador Aid), 9 (MX Missile) and 16 (Export Administration Act/South Africa) (Table 4.6). The extreme observations in Group 2 showed different voting behaviour on Issue 7 (Anti-Satellite Test Ban), 12 (Education Spending), 13 (Superfund Right to Sue), and 16 (Export Administration Act/South Africa) (Table 4.7).

Table 4.6: A comparison of the probability of a positive response for individuals classified as "Normal" vs. "Extreme" in Group 1.

Y/N vs. Undecided	Normal	Extreme	Y vs. N/Undecided	Normal	Extreme
1A	0.97	0.98	1B	0.63	0.58
2A	0.89	0.92	2B	0.54	0.35
3A	0.98	0.98	3B	0.88	0.91
4A	0.96	0.99	4B	0.03	0.05
5A	0.96	0.99	5B	**0.33**	**0.00**
6A	0.94	0.99	6B	0.56	0.30
7A	0.96	1.00	7B	0.64	0.93
8A	0.98	0.99	8B	0.76	0.98
9A	0.86	1.00	9B	**0.51**	**0.96**
10A	0.98	1.00	10B	0.39	0.59
11A	0.94	0.99	11B	0.54	0.39
12A	0.93	0.95	12B	0.18	0.05
13A	0.96	0.95	13B	0.37	0.14
14A	0.96	0.98	14B	0.36	0.27
15A	0.93	0.97	15B	0.57	0.69
16A	**0.37**	**1.00**	16B	**0.33**	**1.00**

Table 4.7: A comparison of the probability of a positive response for individuals classified as “Normal” vs. “Extreme” in Group 2.

Y/N vs. Undecided	Normal	Extreme	Y vs. N/Undecided	Normal	Extreme
1A	0.95	1.00	1B	0.14	0.29
2A	0.85	0.92	2B	0.54	0.25
3A	0.95	1.00	3B	0.06	0.33
4A	0.96	1.00	4B	0.93	0.92
5A	0.96	1.00	5B	0.94	0.98
6A	0.98	1.00	6B	0.95	0.81
7A	0.93	1.00	7B	**0.04**	**0.67**
8A	0.92	0.98	8B	0.03	0.33
9A	0.96	1.00	9B	0.03	0.27
10A	0.97	1.00	10B	0.44	0.67
11A	0.90	0.98	11B	0.17	0.17
12A	0.89	0.96	12B	**0.17**	**0.77**
13A	0.92	0.94	13B	**0.89**	**0.58**
14A	0.93	1.00	14B	0.91	1.00
15A	0.91	0.94	15B	0.02	0.21
16A	0.80	1.00	16B	**0.36**	**1.00**

4.4 Discussion

The MLTCG model has been introduced for robust clustering of binary data. The MLTCG model can be viewed as a generalization of the MLTA that accommodates extreme patterns in binary data via contaminated Gaussian distributions. The MLTCG model can automatically detect extreme observations while clustering. It is demonstrated that the MLTCG model is effective in clustering. Real data are often “contaminated” and it is difficult to detect extreme observations in high-dimensional binary data because the data cannot be easily visualized. When applied to the U.S. Congressional Voting data, our approach performed better in terms of classification when compared to the MLTA and MCLT models. The model parameters are interpretable and provide a characterization of the extreme observations.

Chapter 5

Penalized Mixture of Latent Trait Models

5.1 Introduction

In this chapter, we propose a penalized mixture of latent trait models (PMLTM) for clustered binary data: we assume that the data have been generated by the MLTA model and we shrink the slope parameters, with a gamma-Laplace penalty function. The PMLTM model enables us to encourage sparsity in estimating the slope parameters, thus reducing the number of free parameters considerably and achieving automatic variable selection. Moreover, the component-specific independent tuning parameters avoids the over-penalization that can occur when inferring a shared tuning parameter on clustered data. The newly developed variational EM algorithm provides closed-form estimates for model parameters and avoids intensive searches of the tuning parameters through a model selection criterion such as BIC.

This chapter is outlined as follows: In Section 5.2, we first introduce a penalized mixture of latent trait models via non-convex penalties to realize automatic variable selection; then a new variational EM algorithm is developed for obtaining the penalized model parameters as well as estimates for the component specific tuning parameter $\boldsymbol{\lambda}_g$. The data simulations are presented in Section 5.3. Our approach is then applied to two real data sets (Section 5.4), and we conclude in Section 5.5.

5.2 Model-Based Clustering via a Penalized Mixture of Latent Trait Models

5.2.1 Penalized Mixture of Latent Trait Models via Non-Convex Penalties

We assume that the conditional distribution of $\boldsymbol{x}_i$ in component g is a latent trait model (see details in Section 2.4.1.1). A potential drawback of the MLTA for high-dimensional data is the large number of parameters to be estimated. In particular, the model in Equation 2.5 involves $(G-1)+(G\times M)+G\times(M\times D-D\times(D-1)/2)$ free parameters, of which $G\times(M\times D-D\times(D-1)/2)$ are the parameters $\boldsymbol{w}_1,\ldots,\boldsymbol{w}_G$.

Therefore, we propose the use of a penalized log-likelihood using the form,

$$Q(\boldsymbol{\Theta}) = l(\boldsymbol{\Theta}) - C(\boldsymbol{\Theta}), \tag{5.1}$$

where $l(\boldsymbol{\Theta})$ is the log-likelihood of the model (5.1) and $C(\boldsymbol{\Theta})$ is a penalty term. Similar to the Lasso penalty for regression (Tibshirani, 1996), we propose the use of fat-tailed and sparsity-inducing independent Laplace prior for each coefficient $\boldsymbol{w}_{mg}$. To account for uncertainty about the appropriate level of variable-specific regularization, each Laplace rate parameter

λ_{jg} is left unknown with a gamma hyperprior. Thus,

$$\pi(\boldsymbol{w}_{mg}, \lambda_{mg}) = \frac{r^s}{\Gamma(s)} \lambda_{mg}^{s-1} \exp\{-r\lambda_{mg}\} \prod_{d=1}^{D} \frac{\lambda_{mg}}{2} \exp\{-\lambda_{mg}|w_{dmg}|\}, \quad s, r, \lambda_{mg} > 0 \tag{5.2}$$

This is a departure from the usual shared λ model. However, available cross-validation (e.g., via solution paths) and fully Bayesian (i.e., through Monte-Carlo marginalization) methods for estimating $\boldsymbol{w}_{mg}$ under unknown λ_{mg} are prohibitively expensive. A novel algorithm is proposed for finding posterior mode estimates of the slope parameters (MAP estimates) while treating λ_{mg} as missing data via an EM algorithm. The MAP inference with fixed λ_{mg} is equivalent to likelihood maximization under an L_1-penalty in the Lasso estimation and $\lambda_{mg} \sim \text{Gamma}(s, r)$ leads us to a non-convex penalty (Figure 5.1),

$$\begin{aligned} C(\boldsymbol{w}_{mg}) &= -\log \int_{\lambda_{mg}} \pi(\boldsymbol{w}_{mg}, \lambda_{mg}; s, r) d\lambda_{mg}, \quad s, r, \lambda_{mg} > 0 \\ &= (s + D) \log(1 + \sum_{d=1}^{D} |w_{dmg}|/r) + \text{constant}. \end{aligned} \tag{5.3}$$

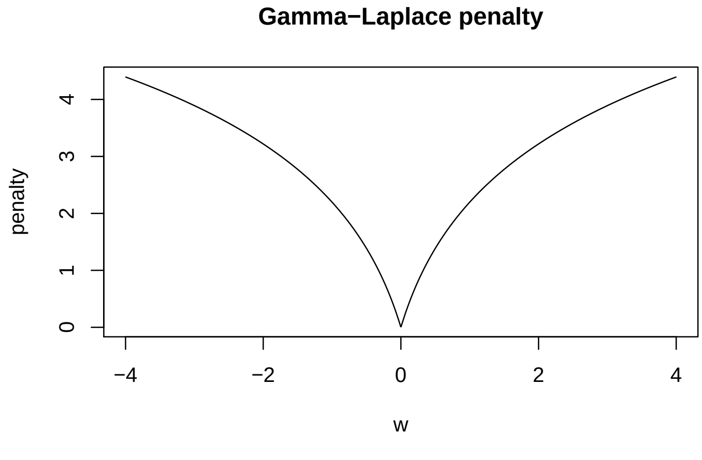


Figure 5.1: The gamma-Laplace penalty $(s+D)\log(1+\sum_{d=1}^{D}|w_d|/r)$ for $s = 1$ and $r = 1/2$.

5.2.2 Motivation for Gamma-Laplace Penalties

One unique aspect of our approach is the use of independent gamma-Laplace priors for each slope parameter $\boldsymbol{w}_{mg}$. The Laplace prior for $\boldsymbol{w}_{mg}$ encourages sparsity in $\boldsymbol{w}_{mg}$ through a sharp density spike at $\boldsymbol{w}_{mg} = 0$ and MAP inference with fixed λ_{mg} is equivalent to likelihood maximization under an L_1 penalty in the Lasso estimation and selection procedure of Tibshirani (1996). In the Bayesian inference for Lasso regression, conjugate gamma hyperpriors are a common choice for the rate parameter λ, e.g., Park and Casella (2008); Yuan and Wei (2014). However, we feel that independent rate parameter λ_{mg} provides a better representation of prior utility, and it avoids the over-penalization that can occur when inferring a shared rate parameter on clustered data.

As detailed in Section 5.2.1, our approach yields an estimation procedure that corresponds to likelihood maximization under a specific non-convex penalty that can be seen as a reparametrization of the 'log-penalty' described in Mazumder *et al.* (2012). Like the standard Lasso, singularity at zero in $C(\boldsymbol{w}_{mg})$ causes some coefficients to be set to zero. However, unlike the Lasso, the gamma-Laplace has gradient $C'(\boldsymbol{w}_{mg}) = \pm(s+D)/\log(1+\sum_{d=1}^{D}|w_{dmg}|/r)$ which disappears as $\sum_{d=1}^{D}|w_{dmg}| \to \infty$, leading to the property of unbiasedness for large coefficients (Fan and Li, 2001).

Commonly, the rate parameter λ is selected using cross-validation (CV) or an information criterion such as BIC. However, our independent λ_{mg} would require searches of impossibly massive dimension. Moreover, CV is sensitive to the data sample where it is applied. That said, one may wish to use CV to choose s or r in the hyperprior, because results are less sensitive to these parameters than they are to a fixed penalty; a small grid of search locations should be sufficient.

5.2.3 Interpretation of the Model Parameters

The interpretation of the model parameters can be exactly as in MLTA and IRT models. In the finite mixture model, η_g is the proportion of observations in the gth component. The characteristics of component g are determined by the parameters α_{mg} and $\boldsymbol{w}_{mg}$. In particular, the intercept α_{mg} has a direct effect on the probability of a positive response to the variable m given by an individual in group g, through the relationship

$$\pi_{mg}(0) = p(x_{im} = 1 | \boldsymbol{y}_n = 0, z_{ng=1}) = \frac{1}{1 + \exp(-\alpha_{mg})}.$$

The value $\pi_{mg}(0)$ is the probability that the median individual in group g has a positive response for the variable m. However, when the data set has very low percentage of positive responses (e.g., text data), the value of $\pi_{mg}(0)$ can be very low for all items across all components. Thus we use the slope parameters to characterize each component in Section 5.4.

The slope parameters $\boldsymbol{w}_g$ are known as discrimination parameters in the item response theory. The larger the value of w_{dmg}, the greater the effect of factor $\boldsymbol{y}_d$ on the probability of a positive response to item m in group g. The quantity w_{dmg} can be used to calculate the correlation coefficient between the observed item $\boldsymbol{x}_i$ and the multivariate latent variable $\boldsymbol{Y}_i$. In the latent trait case, the slope parameters cannot be interpreted as correlation coefficients, because they are not bounded by 0 and 1 as a correlation would be. However, it is possible to transform the loadings so that they can be interpreted as correlation coefficients in exactly the same way as in factor analysis. The standardized w_{dmg} is given by

$$w^*_{dmg} = \frac{w_{dmg}}{\sqrt{1 + \sum_{d=1}^{D} w^2_{dmg}}}. \tag{5.4}$$

The purpose of the Laplace prior for $\boldsymbol{w}_{mg}$ is to encourage sparsity in $\boldsymbol{w}_{mg}$, therefore identifying non-informative variables for each component. When the mth row of the slope parameter matrix in gth component is zero everywhere $(w_{1mg}, w_{2mg} \dots w_{Dmg} = 0)$, then the corresponding variable is not informative. In addition, $w^*_{dmg} = 0$ indicates that item m is independent from latent trait $\boldsymbol{y}_d$ in component g.

5.2.4 Model Identifiability

The identifiability of our model depends on the identifiability of the latent trait part as well as the identifiability of the mixture model. The identifiability of the mixture models has been discussed by several authors (e.g., McLachlan and Peel, 2000a). Knott and Bartholomew (1999) introduces the model identifiability issue in the latent trait analysis. A necessary condition for model identifiability is that the number of the free parameters to be estimated not exceed the number of possible data patterns. However, this condition is not sufficient. The slope parameters $\boldsymbol{w}_g$ are only identifiable with a $d \times d$ constraints. This is important when determining the number of free parameters in the model.

5.2.5 A New Variational EM Algorithm for Parameter Estimation

5.2.5.1 Prior Specification

A classical assumption is to suppose the independence between the prior distribution, thus

$$p(\boldsymbol{\Theta}) = \prod_{g=1}^{G} p(\eta_g) \left(\prod_{m=1}^{M} \prod_{d=1}^{D} p(w_{dmg}) p(\alpha_{mg}) \prod_{i=1}^{n} p(\xi_{img}) \right)$$

where $\eta_g \sim \text{Dirichlet}(\frac{1}{2}, \ldots, \frac{1}{2})$, $\alpha_{mg} \sim N(0,1)$, $W_{dmg} \sim \text{Laplace}(0, \lambda_{mg})$, and $\xi_{img} \sim \text{Uniform}[0, 20]$.

5.2.5.2 Parameter Estimation

We use the EM algorithm to fit our model which is a natural approach for MAP estimation when data are incomplete. In our case, there are three sources of missing data: $\{\boldsymbol{z}_i\}_{i=1}^n$

arises from the fact that we do not know the cluster labels, $\{\boldsymbol{y}_i\}_{i=1}^{n}$ is the D dimensional continuous latent variable, and $\{\boldsymbol{\lambda}_m\}_{m=1}^{M}$ is the unknown Laplace rate parameter.

The purpose of the M-steps of the EM algorithm is to find the MAP estimates of $\boldsymbol{\Theta}$ by maximizing the conditional expectation of the log (complete-data) posterior $\log p(\boldsymbol{\Theta}|\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{\lambda})$ which could be easily obtained:

$$\log p(\boldsymbol{\Theta}|\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{\lambda}) \propto \sum_{i=1}^{n} \sum_{g=1}^{G} \log p(\boldsymbol{x}_i|\boldsymbol{\theta}_g, \boldsymbol{y}_i, z_{ig}) + \log p(\boldsymbol{\theta}_g|z_{ig}, \boldsymbol{\lambda}_g). \tag{5.5}$$

1. E-step: Estimate z_{ig} with
$$z_{ig}^{t+1} = \frac{\eta_g p(\boldsymbol{\theta}_g|\boldsymbol{x}_i)}{\sum_{h=1}^{G} \eta_h p(\boldsymbol{\theta}_h|\boldsymbol{x}_i)}.$$

2. M-step: Estimate η_n
$$\eta_g^{t+1} = \frac{n_g^{(t+1)} - 1/2}{n - G/2}.$$

3. E-step: Estimate the log posterior

   (a) E-step: Compute the latent posterior statistics for $p(\boldsymbol{y}_i|\boldsymbol{x}_i, z_{ig}^{(t+1)}, \boldsymbol{\xi}_{ig}^{(t)}, \boldsymbol{\alpha}_g^{(t)}, \boldsymbol{w}_g^{(t)})$ which is a $N(\boldsymbol{\mu}_{ig}^{(t+1)}, \boldsymbol{\Sigma}_{ig}^{(t+1)})$ density:
$$\begin{aligned} \boldsymbol{\Sigma}_{ig}^{(t+1)} &= \left[\boldsymbol{I}_D - 2\sum_{m=1}^{M} B(\xi_{img}^{t}) \boldsymbol{w}_{mg}^{(t)} \boldsymbol{w}_{mg}^{'(t)}\right]^{-1}, \\ \boldsymbol{\mu}_{ig}^{(t+1)} &= \boldsymbol{\Sigma}_{ig}^{(t+1)} \left[\sum_{m=1}^{M} \left(x_{im} - \frac{1}{2} + 2B(\xi_{img}^{(t)})\alpha_{mg}^{(t)}\right) \boldsymbol{w}_{mg}^{(t)}, \right], \end{aligned}$$
   where $B(\xi_{img}^{(t)}) = (\frac{1}{2} - \sigma(\xi_{img}^{(t)}))/2\xi_{img}^{(t)}$ and $\sigma(\xi_{img}^{(t)}) = \left(1 + \exp(-\xi_{img}^{(t)})\right)^{-1}$.

   (b) E-step: Expected value of the latent posterior statistics for $p(\lambda_{mg}|\boldsymbol{w}_{mg}^{(t)}, \boldsymbol{z}_g^{(t+1)})$ which is a Gamma$(s + D, \sum_{d=1}^{D} |w_{dmg}| + r)$ density:
$$\lambda_{mg}^{(t+1)} = \frac{s + D}{\sum_{d=1}^{D} |w_{dmg}^{(t)}| + r},$$

where $s, r > 0$ are shape and rate parameters of the gamma hyperprior which are pre-determined.

(c) M-step: Optimize the variational parameter ξ_{img} in order to make the approximation $\tilde{p}(\boldsymbol{x}_i | z_{ig}^{(t+1)} = 1, \boldsymbol{\xi}_{ig}^{(t+1)})$ as close as possible to $p(\boldsymbol{x}_i | z_{ig} = 1)$

$$\xi_{img}^{2(t+1)} = \boldsymbol{w}_{mg}^{\prime(t)} \left(\boldsymbol{\Sigma}_{ig}^{(t+1)} + \boldsymbol{\mu}_{ig}^{(t+1)} \boldsymbol{\mu}_{ig}^{\prime(t+1)} \right) \boldsymbol{w}_{mg}^{(t)} + 2\alpha_{mg}^{(t)} \boldsymbol{w}_{mg}^{\prime(t)} \boldsymbol{\mu}_{ig}^{(t+1)} + \alpha_{mg}^{2(t)}.$$

(d) M-step: Optimize the parameters $\boldsymbol{w}_g$ and $\boldsymbol{\alpha}_g$ in order to increase the log (complete-date) posterior $\log p(\boldsymbol{w}_g, \boldsymbol{\alpha}_g | \boldsymbol{x}, \boldsymbol{y}^{(t+1)}, \boldsymbol{\lambda}_g^{(t+1)}, \boldsymbol{\xi}_g^{(t+1)}, \boldsymbol{z}_g^{(t+1)})$:

$$\begin{aligned}\boldsymbol{w}_{mg}^{(t+1)} = & \Bigg[2\textstyle\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}\boldsymbol{y}_{ig}') + \frac{2[\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig})][\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}')]}{\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) - n_g^{(t+1)}} \\ & - n_g \lambda_{mg} \boldsymbol{\Gamma}_{mg}^{(t)} \Bigg]^{-1} \times \Bigg[- \textstyle\sum_{i=1}^n z_{ig}^{(t+1)} (x_{im} - \frac{1}{2}) \boldsymbol{\mu}_{ig}^{(t+1)} - \frac{\sum_{i=1}^n z_{ig}^{(t+1)} (x_{im} - \frac{1}{2}) \sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \boldsymbol{\mu}_{ig}^{(t+1)}}{\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) - n_g^{(t+1)}} \Bigg],\end{aligned}$$

$$\alpha_{mg}^{(t+1)} = - \left[2\textstyle\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) - n_g^{(t+1)} \right]^{-1} \left[\textstyle\sum_{i=1}^n z_{ig}^{(t+1)} \left(x_{im} - \frac{1}{2} + 2B(\xi_{img}^{(t+1)}) \boldsymbol{w}_{mg}^{\prime(t+1)} \boldsymbol{\mu}_{ig}^{(t+1)} \right) \right],$$

where $n_g^{(t+1)} = \sum_{i=1}^n z_{ig}^{(t+1)}$, $\mathbb{E}(\boldsymbol{y}_{ig}\boldsymbol{y}_{ig}') = \boldsymbol{\Sigma}_{ig}^{(t+1)} + \boldsymbol{\mu}_{ig}^{(t+1)} \boldsymbol{\mu}_{ig}^{\prime(t+1)}$,
$\boldsymbol{\Gamma}_{mg}^{(t)} = \text{diag} \left(\frac{1}{|w_{1mg}^{(t)}|}, \ldots, \frac{1}{|w_{Dmg}^{(t)}|} \right)$, and

$$\begin{aligned}& \left[\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}) \right] \left[\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}') \right] \\ & = \sum_{i=1}^n z_{ig}^{2(t+1)} B(\xi_{img}^{2(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}\boldsymbol{y}_{ig}') + 2 \sum_{i<j} z_{ig}^{2(t+1)} z_{jg}^{2(t+1)} B(\xi_{img}^{2(t+1)}) B(\xi_{jmg}^{2(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}\boldsymbol{y}_{jg}').\end{aligned}$$

We adopt a numerically more convenient form of $\boldsymbol{w}_{mg}$

$$\begin{aligned}\boldsymbol{w}_{mg}^{(t+1)} = \boldsymbol{\Upsilon}_{mg}^{(t)} & \left[\boldsymbol{\Upsilon}_{mg}^{(t)} \left(2 \sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}\boldsymbol{y}_{ig}') \right.\right. \\ & \left.\left. + \frac{2[\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig})][\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \mathbb{E}(\boldsymbol{y}_{ig}')]}{\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) - n_g^{(t+1)}} \right) \boldsymbol{\Upsilon}_{mg}^{(t)} - n_g \lambda_{mg} \boldsymbol{I}_D \right]^{-1} \\ & \times \boldsymbol{\Upsilon}_{mg}^{(t)} \left[- \sum_{i=1}^n (x_{im} - \frac{1}{2}) \boldsymbol{\mu}_{ig}^{(t+1)} - \frac{\sum_{i=1}^n z_{ig}^{(t+1)} (x_{im} - \frac{1}{2}) \sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) \boldsymbol{\mu}_{ig}^{(t+1)}}{\sum_{i=1}^n z_{ig}^{(t+1)} B(\xi_{img}^{(t+1)}) - n_g^{(t+1)}} \right],\end{aligned}$$

where $\boldsymbol{\Upsilon}_{mg}^{(t)} = \text{diag} \left(|w_{1mg}^{(t)}|^{1/2}, \ldots, |w_{Dmg}^{(t)}|^{1/2} \right)$. This avoids estimating $|w_{dmg}^{-1(t)}|$, some of which are expected to go to zero.

4. Obtain the lower bound of the log-likelihood at the expansion point ξ_{ig}:

$$L(\boldsymbol{\xi}_{ig}^{(t+1)}) = \sum_{m=1}^{M}\left[\log\sigma(\xi_{img}^{(t+1)}) - \frac{\xi_{img}^{(t+1)}}{2} - B(\xi_{img}^{(t+1)})(\xi_{img})^{2(t+1)} + (x_{im} - \frac{1}{2})\alpha_{mg}^{(t+1)} + B(\xi_{img}^{(t+1)})\alpha_{mg}^{2(t+1)}\right] + \log\frac{|\boldsymbol{\Sigma}_{ig}^{(t+1)}|}{2} + \frac{\boldsymbol{\mu}_{ig}^{'(t+1)}[\Sigma_{ig}^{(t+1)}]^{-1}\boldsymbol{\mu}_{ig}^{(t+1)}}{2}.$$

5. Convergence criterion: The convergence of our variational EM algorithm is determined by the criterion described in Section 2.6.1.

5.2.6 Selection of Programming Languages

When fitting the PMLTM model using R, as the number of items becomes large, the task becomes increasing burdensome. Therefore, we implement our algorithm in two scripting languages, R and Python, and compare the performance between them. Python is an elegant open-source language that has become popular in the scientific community. We use the **Numpy** library for matrix operations and **Scipy.stats** library for the use of probability distributions and statistical functions. Moreover, Python does not generate copies of the objects in an array or a list when slicing arrays and lists which may save memory and shorten the runtime.

5.3 Simulation Study

To illustrate the proposed PMLTM model, we perform simulation experiments. A set of 100 samples of $n = 500$ observations has been generated from a PMLTG model with a two-component mixture ($G = 2, \pi_1 = \pi_2 = 0.5$). The latent variable is generated from a

Gaussian distribution (i.e. $Y \sim N(0, 1)$). Table 5.1 reports the slope parameters $\boldsymbol{w}$ used to generate a set of $M = 10$ observed variables. For each sample, the value of the gamma hyperprior $[s, r]$ was selected from $[0.1, 0.5]$, $[0.5, 0.5]$, $[1, 0.5]$, and $[2, 0.5]$. Table 5.2 shows the BIC and ARI values averaged on the 100 samples for each combination of s and r. As shown in Table 5.2, on average, the BIC has a minimum and ARI has a maximum when $[s, r] = [1, 0.5]$. Therefore, we use $[s, r] = [1, 0.5]$ in future data application.

Table 5.1: Component specific slope parameters.

	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}
$\boldsymbol{w}_1$	0	0	0	0	0	0.5	-0.4	0.3	0.7	1.5
$\boldsymbol{w}_2$	-1.0	-3.8	0.6	-0.7	4.5	0	0	0	0	0

Table 5.2: BIC and ARI values averaged on the 100 samples for each combination of $[s, r]$.

	$s = 0.1, r = 0.5$	$s = 0.5, r = 0.5$	$s = 1, r = 0.5$	$s = 2, r = 0.5$
BIC	17512	13620	13525	13584
ARI	0.70	0.72	0.74	0.74

5.3.1 A Comparison of the Selected Programing Language: R vs. Python

Table 5.3 shows a comparison of the average run time over 100 loops of the E-step function and M-step function using R and Python ($G = 2$, $D = 2$, $n = 100$). Python runs approximately 103 times faster than R for the E-step function and 190 times faster for the M-step function.

Table 5.3: A comparison between R vs. Python

Function	Number of Loops	Python	R
E-Step	100 loops	15.4ms/loop	1.6s/loop
M-Step	100 loops	4.19ms/loop	0.8s/loop

5.4 Application

5.4.1 U.S. Congressional Voting

We assess the performance of the PMLTM model using the U.S. Congressional Voting data. We compare our approach to MLTA, MCLT, and MLTCG models. A summary of the best models for the MLTA, PMLTA, MCLT, MLTCG and PMLTM approaches is shown in Table 5.4. The ARI value obtained using PMLTM model is higher than using the MLTA and PMLTA models. Due to the presence of possible extreme pattens (Table 4.5), the ARI value obtained by the MLTCG model is the highest.

Table 5.4: A comparison of 5 different approaches.

	Model	G	D	BIC	ARI
1	MLTA	3	1	9812	0.42
2	PMLTA	4	2	9681	0.47
3	MCLT	2	5	9597	0.64
4	MLTCG	2	2	9918	0.77
5	PMLTM	2	1	9288	0.58

The classification table for group membership versus party membership for the selected model ($G = 2$, $d = 1$) is presented in Table 5.5. In comparison with the true party membership, there are 52 misclassified representatives associated with the chosen model. Group 1 consists mainly of Republican representatives, and Group 2 consists mainly of Democratic representatives.

Table 5.5: Cross-tabulation of party and predicted classification for our chosen model ($G = 2$, $d = 1$) for the U.S. Congressional Voting data.

	1	2
Republican	151	17
Democrat	35	232

Table 5.6 shows the correlation coefficients for each of the 16 items for each group g.

All the correlation coefficients are positive and large for the “A” variables except item 2 and item 16, which indicates that the latent variable $\boldsymbol{Y}$ has a large effect on the probability of a positive response to vote in both groups. Thus, the majority of representatives voted on most issues and most probabilities given for “B” variables can be interpreted in terms of voting yes versus no.

It can be observed that the correlation coefficients for Group 1 are opposite to the ones for Group 2 for all non-zero items. Items 2 (Water Project Cost-Sharing), 10 (Immigration), and 16 (Export Administration Act) have zero correlation coefficients for both groups, which implies that the corresponding variables are not informative to clustering. In addition, the correlation coefficients for items 1 (Handicapped Infants), 6 (Religious Groups in Schools), 14 (Crime), and 15 (Duty- Free Exports) are zero for Group 2, which implies that the multivariate latent variable $\boldsymbol{Y}$ has no effect on the probability of a positive response to these items. Therefore, the latent variable can be interpreted as a “general” factor relating to the areas of military and foreign affairs. Crespin and Rohde (2010) drew a similar conclusion and they found distinct voting differences in the area of military, foreign affairs and agriculture in appropriations voting.

5.4.1.1 Computational Time: R vs. Python

Table 5.7 shows a comparison of the average run time for 10 runs using R and Python when $G = 2$ and $D = 1$. Python runs approximately 107 times faster than R.

5.4.2 Boston Airbnb Reviews

This data set includes detailed English comments on the Airbnb website in the Boston area from 65275 guests ($n = 65275$) since 2008. We perform some pre-processing of the text data (i.e., converting the text to lower case, removing numbers and punctuation, removing

Table 5.6: Correlation Coefficients w^*_{mg} for Group g for each of 16 votes in the U.S. Congressional Voting data.

Y/N vs. Undecided	G1	G2	Y vs. N/Undecided	G1	G2
1A	0.54	0.66	1B	-0.25	0
2A	0.36	0.38	2B	0	0
3A	0.53	0.93	3B	-0.29	0.51
4A	0.55	0.57	4B	0.35	-0.43
5A	0.55	0.57	5B	0.53	-0.44
6A	0.57	0.65	6B	0.47	0
7A	0.54	0.79	7B	-0.30	0.42
8A	0.51	0.70	8B	-0.46	0.55
9A	0.52	0.57	9B	-0.46	0.29
10A	0.56	0.56	10B	0	0
11A	0.49	0.87	11B	-0.17	0.15
12A	0.46	0.48	12B	0.27	-0.39
13A	0.50	0.53	13B	0.33	-0.26
14A	0.54	0.65	14B	0.51	0
15A	0.48	0.69	15B	-0.40	0
16A	0.34	0.10	16B	0	0.09

Table 5.7: A comparison between R vs. Python

Number of Runs	Python	R
10 runs	288ms/run	31s/run

stop words, and stemming). These basic transforms are available within the R package **tm** (Feinerer and Hornik, 2015). We then create a matrix with each comment as a row and each word as a column. If a word is mentioned in a comment, the response for the corresponding cell is coded as 1, and otherwise is 0. The term matrix contains 43584 words, most of them being infrequently used, which we refer to as "sparse terms". We remove sparse terms that appear less than 2% in all reviews because we are often not interested in such terms. At the end of this pre-processing step, the term matrix consists of 278 words (i.e., $M = 278$) and we generate a word cloud (Figure 5.2) to provide a quick visual overview of the frequency of the words in the final term matrix.

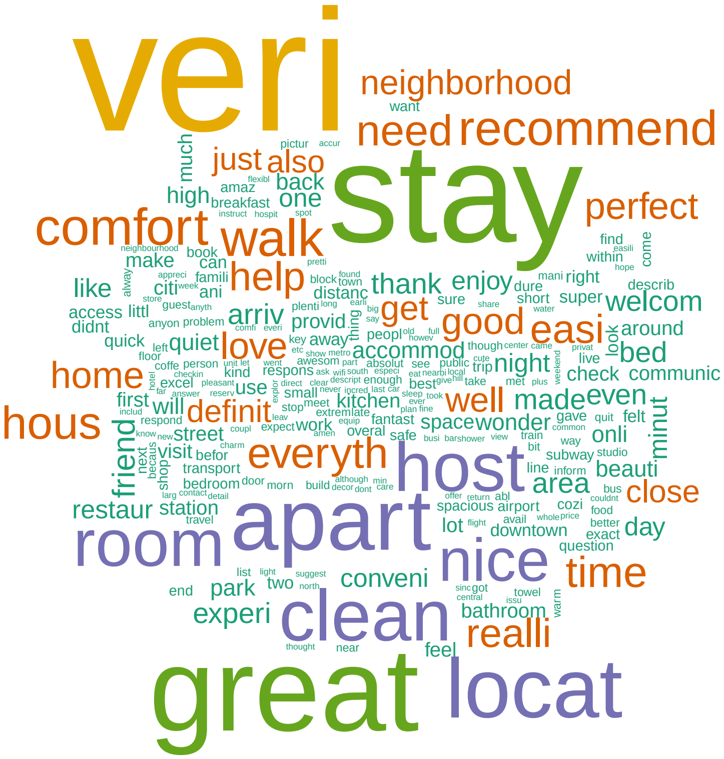


Figure 5.2: A word cloud for the final term matrix of the Airbnb comments.

The PMLTM model is fitted to these data for $D = 1, 2, 3, 4, 5$ and $G = 1, 2, 3, 4, 5$. We run all models in both R and Python. It takes us more than 24 hours to get the results using R while Python only takes 106 minutes. The minimum BIC occurs at the 3-cluster, 2-dimensional PMLTM model. The BIC value is 869496. The classification table for group membership of the selected model ($G = 3, d = 2$) is presented in Table 5.8. We also include average sentiment scores for each group which are calculated using the built-in Python library **nlkt** (natural language toolkit). The sentiment of each comment: positive, negative, or neutral is presented using a score ranging from 0 to 1. Because each comment could contain positives and negatives at the same time, a compound score is presented as well. Each compound score ranges from -1 to 1; -1 for an overall unpleasant tone and 1 for an overall pleasant tone. Group 1 consists of mainly very positive comments. The compound score is 0.95 which indicates that the overall tone of the comments is very positive. The average positivity score in Group 1 is 0.30. Group 2 is a small group that consists of comments that have a slightly negative tone overall. It is worth noting that the average negativity score is higher than the average positivity score in Group 2. Group 3 consists of mainly positive comments as well, and the average compound score is lower in Group 3 when compared to Group 1.

Table 5.8: The predicted classification and the sentiment scores for our chosen model ($G = 3$, $d = 2$) for the Airbnb data.

Group	Number of Observations	Compound	Negativity	Neutrality	Positivity
Group 1	39492	0.95	0	0.70	0.30
Group 2	5605	-0.35	0.07	0.86	0.05
Group 3	20178	0.58	0	0.68	0.32

Table 5.9 shows the high-loading words for each latent trait in each group. We note that the first latent trait $\boldsymbol{y}_1$ is concerned with the listings (e.g., location of the property, condition of the property, etc.) whereas the second latent trait $\boldsymbol{y}_2$ is concerned with the host. We can further characterize our groups from this observation. The comments in Group 1 and Group

3 both have an overall positive tone. However, comments in Group 1 are more specific about the listings and the hosts. Moreover, the positive comments are more intense in Group 1 by using words such as “absolute”, “amazing” and “awesome”. Group 3 mainly consists of generic positive comments (see Table 5.10); they are less specific about the properties or the hosts. Most high-loading words in Group 2 are considered neutral, but words such as “disappoint” and “never” are negative terms. It is worth noting that there are comments in Group 2, which, even though we would say the sentiment with regards to the host is positive, the sentiment of the overall paragraph is negative (see Table 5.10).

Table 5.9: High-loading words for each latent trait of our chosen model ($G = 3$, $d = 2$) for the Airbnb data.

Group1	$\boldsymbol{y}_1$	absolute, accur, amaz, awesom, bar, bedroom, big, bus, easili, equip, floor, lot, metro, store
	$\boldsymbol{y}_2$	answer, anything, apprici, ask, next, plus, return, reserv
Group2	$\boldsymbol{y}_1$	busi, discript, cute, detail, ever, par, never, old, explor, north, plan, south, studio, view
	$\boldsymbol{y}_2$	checkin, common, contact, couldn’t, disappoint, suggust
Group3	$\boldsymbol{y}_1$	found, stay, spot, care
	$\boldsymbol{y}_2$	welcome, help, pleasent, next, good, book, host, friend, suggest

Table 5.10: Sample reviews of our chosen model ($G = 3$, $d = 2$).

Group	Reviews
Group 1	1. "The place is really well furnished, pleasant and clean. Islam was very helpful, you can feel free to ask him virtually anything and he'll help you. He was fun too, very cool talking to him. Oh, and the place is pretty conveniently located too. Highly recommended. The neighbourhood might not be the cleanest in Boston (my gf liked Brooklyne much more in that matter), but this is a great location and price for value overall." 2. "Perry's house is much cleaner and bigger than it is in the pictures. We are very happy to stay at his apartment. Perry is also very friendly and thoughtful. He explained all the instructions very clearly and he kept contacting us to know if we had any question. The house is located in a nice neighborhood, about 5 minute walking to a train/subway station." 3. "We stayed here for almost 2 months when we relocated to Boston quite quickly. The apartment was very clean and very new. Perry went out of his way on multiple occasions to make sure that me, my husband and our 18 month old son had everything we needed. The kitchen and bathroom are very newly renovated and the kitchen had everything we needed (appliances, pots/pans, etc). We had a great experience here and would definitely recommend it."
Group 2	1. "Izzy's communication is very good. All communication was done via text or AirBnB messaging. Directions and house details were well spelled out and clear. I was in the basement room of the 3 rooms he rents out. Everything is clean but spares. I would not consider it cozy but it was a very good value." 2. "We were rather disappointed with this accommodatiion. The host did not even meet us, but left rather complicated instructions to access the keys to the apartment. We did not meet the host at all during our stay, or even hear from him as to how we were getting on. The apartment was somewhat shabby, and not really like the image indicated, as this only showed a small corner of one room. The kitchen was tiny, and although quite well equipped, it badly needed redecoration and a good clean. In addition, the apartment backed onto a yard with three dumpsters, and on 4 occasions we were awakened early in the morning by the noise of the dumpsters being emptied." 3. "I fell in love with the view of this apartment. Fenway out the window as promised. My expectations were pretty low going in because I realized it was very basic budget accommodations. Sean was helpful with the different questions I had about the city. The instructions for obtaining lockbox key were very clear. The location is great and the building old and had a lot of character. I came to town with a friend of mine for the night to catch the Red Sox game. We understood it to have a large enough bed to accommodate us since it says 1 to 4 people. When we arrived the bed seemed quite small. When I asked Sean about it he told me that there was 2 mattresses on top of each other and to take them apart and he thought that there were sheets in the closet for both (there were not) we had explored Boston all day and didn't return til 1 am..pulling a mattress apart was not what I wanted to do. We were so tired and since there was only 1 sheet we decided to just be very cozy. The bed was comfortable and we slept well until around 5 am when people were down in the alley going through glass bottles in the trash dumpsters which was very loud. (Not sure if that happens all the time) The kitchen is small but would be helpful if you needed one. I would not recommend having 4 people stay as it would be quite cramped (but if you are looking for a budget place with a great view..this would work.)"
Group 3	1. "GREAT SPACE, PERFECT LOCATION, AWESOME PEOPLE!! Definately will be back!!!!" 2. "We liked the apartment but not the three flights of steps to get to it." 3. "Everything was great - as described and expected."

5.5 Discussion

In this chapter, we extend the MLTA model by introducing gamma-Laplace penalties on the slope parameters. The PMLTM model enables us to encourage sparsity in estimating the slope parameters, thus reducing the number of free parameters considerably. The component-specific tuning parameters avoid the over-penalization that can occur when inferring a shared tuning parameter on clustered data. The PMLTM model retains the ability to investigate the dependence between variables while clustering with the added advantage of being able to model very high-dimensional binary data (e.g., text data).

The excellent clustering behaviour of this method has been shown by two applications: on the U.S. Congressional Voting and the Boston Airbnb reviews data sets. In both cases, the model found groups that were intuitive in their interpretation. Applying the PMLTM model to the Boston Airbnb reviews data showed that the method scales to far larger datasets than any other model-based clustering methods for binary data.

Chapter 6

Mixture of Multinomial Latent Trait Models with Common Slope Parameters

6.1 Introduction

In this chapter, we introduce the MMCLT models that generalize the MCLT model by implementing multinomial logistic response function for clustering categorical data. The sharing of the slope parameters reduces the number of parameters to a manageable size; however, each latent trait still has a different effect in each group. A variational EM algorithm based on two quadratic lower bounds to the multinomial likelihood, a loose static bound (Böhning, 1992, Böhning bound) and a multivariate sharp bound (Browne and McNicholas, 2015b, Browne-McNicholas bound), is developed.

This chapter is organized as follows: The framework of MMCLT is introduced in Section 6.2. The variational EM algorithm and the model fitting are laid out in Section 6.3. Our approach is then applied to a real data set (Section 6.4), and we conclude in Section 6.5.

6.2 Mixture of Multinomial Latent Trait Models with Common Slope Parameters

6.2.1 Overview

Let $k = 0, 1, \ldots, (K-1)$ denote the number of categories of variable m. The indicator variable $\boldsymbol{x}_{nm} = k$ is represented by a K-dimensional vector in which k^{th} element is set to 1 and all remaining elements to 0 (Equation 6.1).

$$\boldsymbol{x}_{nm}(k) \begin{cases} 1 & \text{if the response falls in category } k \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

We denote a full response pattern for one individual by $\boldsymbol{x}_i = (\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \ldots, \boldsymbol{x}_{iM})$. Similarly to the MCLT model, we assume that each observation $\boldsymbol{x}_n$ comes from one of the G components and we have $\boldsymbol{z}_i = (z_{i1}, \ldots, z_{iG})$ to identify the group membership. Further, given that the observation is from group g (i.e. $z_{ig} = 1$), we assume that the conditional distribution is a latent trait model with a multinomial logistic response function. Thus, the MMCLT model takes the form

$$p(\boldsymbol{x}_i) = \sum_{g=1}^{G} \eta_g p(\boldsymbol{x}_i | z_{ig} = 1) = \sum_{g=1}^{G} \eta_g \int_{\boldsymbol{y}_{ig}} p\left(\boldsymbol{x}_i | \boldsymbol{y}_{ig}, z_{ig} = 1\right) p\left(\boldsymbol{y}_{ig}\right) d\boldsymbol{y}_{ig},$$

where

$$p\left(\boldsymbol{x}_i | \boldsymbol{y}_{ig}, z_{ig} = 1\right) = \prod_{m=1}^{M} \prod_{k=0}^{K-1} \left(\pi_{mgk}(\boldsymbol{y}_{ig})\right)^{\boldsymbol{x}_{im(k)}},$$

and the multinomial response function for $x_{im}(k)$ of each group is

$$\pi_{mgk}(y_{ig}) \begin{cases} p\left(\boldsymbol{x}_{im}(0) = 1|\boldsymbol{y}_{ig}, z_{ig} = 1\right) & \\ = \exp\left(1 - \log\left(1 + \sum_{k=1}^{K-1}\exp(\boldsymbol{\gamma}_{imgk})\right)\right) & k = 0 \\ p\left(\boldsymbol{x}_{im}(k) = 1|\boldsymbol{y}_{ig}, z_{ig} = 1\right) & \\ = \exp\left(\boldsymbol{\gamma}_{imgk} - \log\left(1 + \sum_{k=1}^{K-1}\exp(\boldsymbol{\gamma}_{imgk})\right)\right) & k = 1, \ldots K-1, \end{cases} \tag{6.2}$$

where $\boldsymbol{\gamma}_{imgk} = \boldsymbol{w}_{mk}\boldsymbol{y}_{ig}$ and the d-dimensional latent variable $\boldsymbol{Y}_{ig} \sim \text{MVN}(\boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$. We denote the common model parameter w_m as

$$\boldsymbol{w}_m = \begin{bmatrix} 0 & \cdots & 0 \\ w_{m11} & \cdots & w_{m1d} \\ \vdots & \vdots & \vdots \\ w_{mK1} & \cdots & w_{mKd} \end{bmatrix}_{K\times d}.$$

The first row of w_m must be set to zero to ensure identifiability. The complete-data log-likelihood is then given by

$$l = \sum_{i=1}^{n}\log\left(\sum_{g=1}^{G}\eta_g\int_{\boldsymbol{y}_{ig}}\prod_{m=1}^{M}\prod_{k=0}^{K-1}p(\boldsymbol{x}_{im}(k)|\boldsymbol{y}_{ig}, z_{ig} = 1)p(\boldsymbol{y}_{ig})d\boldsymbol{y}_{ig}\right). \tag{6.3}$$

Similar to Tang *et al.* (2015), we consider a further parametrization of the covariance matrices $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_g$ of the mixture components following the work of Celeux and Govaert (1995). The 14 models in Table 6.1 are considered.

6.2.2 Interpretation of Model Parameters

The interpretation of η_g is the same as in the finite mixture model; η_g is the mixing proportion for component g and corresponds to the prior probability that a randomly chosen individual is in the gth component.

The property of the observations within the group g is represented by the common slope

Table 6.1: Some important characteristics of parameterizations of $\boldsymbol{\Sigma_g}$ and the number of free parameters in the model.

Model	$\boldsymbol{\Sigma}_g$	Vol/Shape/Orientation	Number of free parameters
1	$S\boldsymbol{QAQ}'$	EEE	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+d(d+1)/2-d^2$
2	$S_g\boldsymbol{QAQ}'$	VEE	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+d(d+1)/2+G-1-d^2$
3	$S\boldsymbol{QA}_g\boldsymbol{Q}'$	EVE	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+d(d+1)/2+(G-1)(d-1)-d^2$
4	$S_g\boldsymbol{QA}_g\boldsymbol{Q}'$	VVE	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+d(d+1)/2+(G-1)d-d^2$
5	$S\boldsymbol{Q}_g\boldsymbol{AQ}'_g$	EEV	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G(d(d+1)/2)-(G-1)d-d^2$
6	$S_g\boldsymbol{Q}_g\boldsymbol{AQ}'_g$	VEV	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G(d(d+1)/2)-(G-1)(d-1)-d^2$
7	$S\boldsymbol{Q}_g\boldsymbol{A}_g\boldsymbol{Q}'_g$	EVV	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G(d(d+1)/2)-(G-1)-d^2$
8	$S_g\boldsymbol{Q}_g\boldsymbol{A}_g\boldsymbol{Q}'_g$	VVV	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G(d(d+1)/2)-d^2$
9	$S\boldsymbol{V}$	EEI	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+d-d^2$
10	$S_g\boldsymbol{V}$	VEI	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G+d-1-d^2$
11	$S\boldsymbol{V}_g$	EVI	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+Gd-G+1-d^2$
12	$S_g\boldsymbol{V}_g$	VVI	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+Gd-d^2$
13	$S\boldsymbol{I}$	EII	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+1-d^2$
14	$S_g\boldsymbol{I}$	VII	$G-1+d\left(\sum_{m=1}^{M} m(K_m-1)+G\right)+G-d^2$

parameter $\boldsymbol{w}_m$ and the hyperparameters of the latent variable $\boldsymbol{Y}_{ng}$. In particular, we can write the response function π_{mgk} with "median" response probabilities

$$\pi_{mgk}(\boldsymbol{\mu}_g) = \exp\left(\boldsymbol{w}_{mk}\boldsymbol{\mu}_g - \log\left(1+\sum_{k=1}^{K-1}\exp(\boldsymbol{w}_{mk}\boldsymbol{\mu}_g)\right)\right). \tag{6.4}$$

The interpretation of $\pi_{mgk}(\boldsymbol{\mu}_g)$ is the probability that the median individual in group g has a positive response for the variable m and category k.

The discriminating power of the latent variable $\boldsymbol{y}_{ng}$ is indicated by the spread of the $\boldsymbol{w}_m(k)^* = \boldsymbol{w}_m \times \text{diag}(\boldsymbol{\Sigma}_g^{\frac{1}{2}})$ considered as a function of k. A large spread produces large differences between the corresponding response probabilities in the variable m for observations from group g.

Moreover, the mean $\boldsymbol{\mu}_g$ and covariance matrix $\boldsymbol{\Sigma}_g$ of component g can be used to provide low dimensional plots of the cluster.

6.2.3 Related Models

The MMCLT model has a lot of common characteristics with a number of models in literature, especially, with regards to item response theory.

The MMCLT model can be seen as a categorical version of the mixture of factor analyzers with common factor loadings (MCFA) (Baek *et al.*, 2010). The loading matrix in the MCFA model is analogous to the common slope parameters; the component means and the mixing proportions take identical roles in both models.

Bolt *et al.* (2001) considers a mixture of nominal response model (MNRM) for multiple-choice data. A natural parameter y_{gmk} is written as:

$$y_{gmk} = \lambda_{mk}\theta + \xi_{gmk},$$

and the resulting mixture nominal response model as

$$P_{gmk} = \frac{\exp(y_{gmk})}{\sum_{k=1}^{K} \exp(y_{gmk})}.$$

This model has a close connection to the proposed MMCLT model. Key differences between our model and the MNRM model are that we focus on common slope parameters and multivariate trait parameters; we further introduce a computationally efficient alternative algorithm for fitting the model without the need to adopt to a Markov Chain Monte Carlo estimation or quadrature methods.

6.2.4 Model Identifiability

The identifiability of our model depends on the identifiability of the latent trait part as well as the identifiability of the mixture model. The identifiability of the mixture models has been discussed by several authors (e.g., McLachlan and Peel, 2000a). Tang *et al.* (2015) give a explanation of the model identifiability in the latent trait models with common slope parameters. They also introduce a cut-off value for the common slope parameters, i.e., $|w_{mkd}| \leq 10$. By restricting the slope parameters, they prevent the estimated covariance matrix from converging to singular matrices. Knott and Bartholomew (1999) mention that this estimation of slope parameters can increase without limit due to small sample sizes. Therefore, we adopt the same cut-ff for the common slope parameters in this chapter.

6.3 Variational Bounds for Model Fitting

6.3.1 Variational Methods

Because there is no conjugate family for the multinomial logistic model, variational approximations have been proposed to fit latent trait models. We take a second order Taylor series expansion of the log-sum-exp (LSE) function around a point ξ

$$\text{lse}(\boldsymbol{\gamma}_{img}) = \text{lse}(\boldsymbol{\xi}_{im}) + \nabla\text{lse}(\boldsymbol{\xi}_{im})(\boldsymbol{\gamma}_{img} - \boldsymbol{\xi}_{im}) + \frac{1}{2}(\boldsymbol{\gamma}_{img} - \boldsymbol{\xi}_{im})' H(\boldsymbol{\xi}_{im})(\boldsymbol{\gamma}_{img} - \boldsymbol{\xi}_{im}),$$

where $\boldsymbol{\gamma}_{img} = \boldsymbol{W}_m \boldsymbol{y}_{ig}$. An upper bound to the LSE function can be found by replacing the Hessian matrix $H(\xi)$ which appears in the second order term. Böhning (1992) and Krishnapuram *et al.* (2005) propose a fixed matrix $\boldsymbol{B}$ such that $\boldsymbol{B} - H(\xi)$ is positive definite for all $\boldsymbol{\xi}$. Browne and McNicholas (2015b) derive a sharp quadratic bound for a multivariate LSE function by replacing the Hessian matrix $H(\xi)$ with B^*, which can be seen as a multivariate analogue of the variational bound for binary responses proposed by Jaakkola and Jordan (2000). For a given point ξ, $\text{lse}(\boldsymbol{\gamma}_{img}|B^*) \leq \text{lse}(\boldsymbol{\gamma}_{img}|B)$ for all x or equivalently, $B \succeq B^* \succeq 0$. Both variational bounds allows for the computation of an approximate log-likelihood in closed-form. In this case the lower bound of each term in the log-likelihood,

$$L(\boldsymbol{\xi}_i) = \log(\underline{p}(\boldsymbol{x}_i|\boldsymbol{\xi}_i) = \log\left(\int \prod_{m=1}^{M} \underline{p}(\boldsymbol{x}_{im}|\boldsymbol{y}_i, \boldsymbol{\xi}_{im}) p(\boldsymbol{y}_i)\, d\boldsymbol{y}_i\right),$$

where

$$\begin{aligned}\underline{p}(\boldsymbol{x}_{im}|\boldsymbol{y}_i,\boldsymbol{\xi}_{im}) = (&\boldsymbol{x}_{im}\boldsymbol{\gamma}_{im} - \text{lse}(\boldsymbol{\xi}_{im}) + \pi'_m(\boldsymbol{\xi}_{im})\boldsymbol{\xi}_{im} \\ &+(\boldsymbol{B}(\xi_{im})\boldsymbol{\xi}_{im} - \pi_m(\boldsymbol{\xi}_{im}))'\boldsymbol{\gamma}_{im} - \frac{1}{2}\boldsymbol{\gamma}'_{im}\boldsymbol{B}(\xi_{im})\boldsymbol{\gamma}_{im} - \frac{1}{2}\boldsymbol{\xi}'_{im}\boldsymbol{B}(\xi_{im})\boldsymbol{\xi}_{im}\Big),\end{aligned}$$

$$\boldsymbol{B}(\xi_{im}) = \begin{cases} \frac{1}{2}\left(\boldsymbol{I}_{K-1} - \mathbf{1}_{K-1}\mathbf{1}'_{K-1}/K\right) & \text{for Böhning bound,} \\ \left(\text{diag}[b(\xi_{im0}),\ldots,b(\xi_{im(K-1)})] + b(\xi_{imK})\mathbf{1}_{K-1}\mathbf{1}'_{K-1}\right)^{-1} & \text{for Browne-McNicholas bound,} \end{cases} \tag{6.5}$$

where $b(\xi_{imk}) = 2 \times \max\left\{\frac{\xi_{imk}-1-\log(\xi_{imk})}{(1-\xi_{imk})^2}, 1\right\}$;

$$\begin{aligned}\boldsymbol{\gamma}_{im} &= \boldsymbol{w}_m\boldsymbol{y}_i, \\ \text{lse}\,(\boldsymbol{\xi}_{im}) &= \log\left(1 + \sum_{k=1}^{K-1}\boldsymbol{\xi}_{im}\right),\end{aligned}$$

where K is the number of categories, $\boldsymbol{\xi}_{im}$ is the vector of variational parameters, $\boldsymbol{I}_K$ is the identity matrix of size $K \times K$ and $\mathbf{1}_K$ is a vector of ones of length K.

Although the Browne-McNicholas bound is the tightest bound for a given point, it has higher computational complexity. The reason is that $\boldsymbol{B}(\xi_{im})$ now depends on $\boldsymbol{\xi}$ and hence on i, which means we need to compute a different posterior covariance matrix for each i (Equation 6.5). By using the Böhning bound we need only invert the posterior covariance matrix once. Its computational efficiency becomes the most attractive feature, especially in the multinomial case. In this chapter, we study both bounds to explore the speed vs accuracy trade-off.

6.3.2 Model Fitting

Here we derive a variational EM algorithm to obtain the approximation of the likelihood:

1. E-Step: Estimate $z_{ig}^{(t+1)}$ with:
$$z_{ig}^{(t+1)} = \frac{\eta_g^{(t)} \exp(L(\boldsymbol{\xi}_{ig}^{(t)}))}{\sum_{g=1}^{G} {\eta'}_g^{(t)} \exp(L({\boldsymbol{\xi}'}_{ig}^{(t)}))}.$$

2. M-Step: Estimate $\eta_g^{(t+1)}$ using
$$\eta_g^{(t+1)} = \frac{\sum_{i=1}^{n} z_{ig}^{(t+1)}}{N}.$$

3. Estimate the lower bound of log-likelihood via a $K \times 1$ variational parameter vector $\boldsymbol{\xi}_{img}$:

   (a) E-Step: We approximate the latent posterior statistics for $p(\boldsymbol{y}_{ig}|\boldsymbol{x}_i, z_{ig}^{(t+1)} = 1)$ by its variational lower bound $\underline{p}(\boldsymbol{y}_{ig}|\boldsymbol{x}_i, z_{ig}^{(t+1)} = 1, \boldsymbol{\xi}_{ig}^{(t)})$, which is a $N(\boldsymbol{v}_{ig}^{(t+1)}, \boldsymbol{\varphi}_{ig}^{(t+1)}))$ density:
$$(\boldsymbol{\varphi}_{ig})^{(t+1)} = \left((\boldsymbol{\Sigma}_g^{-1})^{(t)} + \sum_{m=1}^{M} \boldsymbol{w}'^{(t)}_m \, \boldsymbol{B}(\boldsymbol{\xi}_{img}) \, \boldsymbol{w}_m^{(t)} \right)^{-1},$$
$$\boldsymbol{v}_{ig}^{(t+1)} = \boldsymbol{\varphi}_{ig}^{(t+1)} \left((\boldsymbol{\Sigma}_g^{-1})^{(t)} \boldsymbol{\mu}_g^{(t)} + \sum_{m=1}^{M} (\boldsymbol{w}'^{(t)}_m \left(x_{im} + \boldsymbol{B}(\boldsymbol{\xi}_{img}) \boldsymbol{\xi}_{img}^{(t)} - \pi_{mg}(\boldsymbol{\xi}_{img}^{(t)}) \right) \right),$$
   where
$$\boldsymbol{B}(\boldsymbol{\xi}_{img}) = \begin{cases} \frac{1}{2} \left(\boldsymbol{I}_K - \boldsymbol{1}_K \boldsymbol{1}'_K / (K+1) \right) & \text{Böhning bound,} \\ \left(\mathrm{diag}[b(\xi_{img1}), \ldots, b(\xi_{imgK})] + b(\xi_{img(K+1)}) \boldsymbol{1}_K \boldsymbol{1}'_K \right)^{-1} & \text{Browne-McNicholas bound,} \end{cases} \tag{6.6}$$
   where $b(\xi_{imgk}) = 2 \times \max \left\{ \frac{\xi_{imgk} - 1 - \log(\xi_{imgk})}{(1 - \xi_{imgk})^2}, 1 \right\}$;
   and $\pi_{mg}(\boldsymbol{\xi}_{img}^{(t)}) = \exp \left\{ \boldsymbol{\xi}_{img}^{(t)} - \log \left(1 + \sum_{k=1}^{k-1} \exp(\boldsymbol{\xi}_{imgk}^{(t)}) \right) \right\}$.

   (b) M-Step: Optimize the variational parameter vector $\boldsymbol{\xi}_{img}^{(t+1)}$.

      i. By using the Böhning bound, one can show that the optimal value is
$$\boldsymbol{\xi}_{img}^{(t+1)} = \boldsymbol{w}_m^{(t)} \boldsymbol{v}_{ig}^{(t+1)}.$$

This follows from the fact that the Böhning bound is tight for the LSE function when $\xi_{imgk} = \gamma_{imgk}$.

ii. For Browne-McNicholas Bound, the update for $\boldsymbol{\xi}_{img}$ can be written as

$$\boldsymbol{\xi}_{img}^{(t+1)} = \boldsymbol{\xi}_{img}^{(t)} + \frac{1}{2}\frac{\partial Q_{img}}{\partial \boldsymbol{\xi}_{img}}|_{\boldsymbol{\xi}_{img}=\boldsymbol{\xi}_{img}^{(t)}}.$$

(c) Update accumulated sufficient statistics for the parameters $\boldsymbol{w}_m$, $\boldsymbol{\mu}_g$, and $\boldsymbol{\Sigma}_g$ based on the closed-form posterior distributions corresponding to the observations in the data set:

$$\boldsymbol{\Sigma}_g^{(t+1)} = \frac{1}{n_g}\sum_{i=1}^{n} z_{ig}^{(t+1)}\boldsymbol{\varphi}_{ig}^{(t+1)},$$
$$\boldsymbol{\mu}_g^{(t+1)} = \frac{1}{n_g}\sum_{i=1}^{n} z_{ig}^{(t+1)}\boldsymbol{\upsilon}_{ig}^{(t+1)},$$

where $n_g = z_{1g} + \cdots + z_{ng}$ and

$$\begin{aligned}\text{vec}(\boldsymbol{w}_m^{(t+1)}) = &\left(\sum_{g=1}^{G}\sum_{i=1}^{n} z_{ig}\boldsymbol{B}(\xi_{img}^{(t+1)}) \otimes \left(\boldsymbol{\psi}_{ig}^{(t+1)} + \boldsymbol{\upsilon}_{ig}^{(t+1)}\boldsymbol{\upsilon}_{ig}^{\prime(t+1)}\right)\right)^{-1}\\ &\text{vec}\left(\sum_{g=1}^{G}\sum_{i=1}^{n} z_{ig}^{(t+1)}\left(\boldsymbol{x}_{im} + \boldsymbol{B}(\xi_{img}^{(t+1)})\boldsymbol{\xi}_{img}^{(t+1)} - \pi_{mg}(\boldsymbol{\xi}_{img}^{(t+1)})\right)\boldsymbol{\upsilon}_{ig}^{\prime(t+1)}\right).\end{aligned}$$

(d) Obtain the lower bound:

$$\begin{aligned}L(\boldsymbol{\xi}_{ig}^{(t+1)}) = &\sum_{m=1}^{M}\left(-\text{lse}(\boldsymbol{\xi}_{img}^{(t+1)}) + \pi'_{mg}(\boldsymbol{\xi}_{img}^{(t+1)})\boldsymbol{\xi}_{img}^{(t+1)} - \frac{1}{2}\boldsymbol{\xi}_{img}^{\prime(t+1)}\boldsymbol{B}(\boldsymbol{\xi}_{img})\boldsymbol{\xi}_{img}^{(t+1)}\right)\\ &- \frac{\boldsymbol{\mu}_g^{\prime(t+1)}(\boldsymbol{\Sigma}_g^{-1})^{(t+1)}\boldsymbol{\mu}_g^{(t+1)}}{2} + \frac{1}{2}\log\frac{|\boldsymbol{\varphi}_{ig}^{(t+1)}|}{|\boldsymbol{\Sigma}_g^{(t+1)}|} + \frac{\boldsymbol{\upsilon}_{ig}^{\prime(t+1)}(\boldsymbol{\varphi}_{ig}^{-1})^{(t+1)}\boldsymbol{\upsilon}_{ig}^{(t+1)}}{2},\end{aligned}$$

where

$$\text{lse}\left(\boldsymbol{\xi}_{img}^{(t+1)}\right) = \log\left(1 + \sum_{k=1}^{K-1}\boldsymbol{\xi}_{imgk}^{(t+1)}\right),$$

and the log-likelihood:

$$l^{(t)} \approx \sum_{i=1}^{n}\log\left(\sum_{g=1}^{G}\eta_g^{(t+1)}\exp(L(\boldsymbol{\xi}_{ig}^{(t+1)}))\right).$$

4. Return to Step 1.(a). Stops when convergence is reached.

6.4 Application

6.4.1 Mushroom Data

The mushroom data set includes descriptions of 8124 hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family. Each species is classified as edible or poisonous. We adopt 21 attributes: three describe the cap, one describes the bruises, one describes the odor, four describe the gill, five describe the stalk, two describe the veil, two describe the ring, one describes the spore, one describes the population and one describes the habitat. All attributes have different numbers of nominal categories (Table 6.2).

Table 6.2: Attributes for the mushroom data.

	Attribute Name	Number of Categories
1	Cap-shape	6
2	Cap-surface	4
3	Cap-color	10
4	Bruises	2
5	Odor	9
6	Gill-attachment	2
7	Gill-spacing	2
8	Gill-size	2
9	Gill-color	12
10	Stalk-shape	2
11	Stalk-surface-above-ring	5
12	Stalk-surface-below-ring	4
13	Stalk-color-above-ring	4
14	Stalk-color-below-ring	9
15	Veil-type	9
16	Veil-color	4
17	Ring-number	3
18	Ring-type	5
19	Spore-print-color	9
20	Population	6
21	Habitat	7

The fourteen MMCLT models were fitted to these data for $d = 1, \ldots, 3$ and $G = 1, \ldots, 3$.

The minimum BIC occurs at the 2-component, 2-dimensional latent trait model via Browne-McNicholas bound and $\mathbf{\Sigma}_g = S\boldsymbol{V}_g$, which is considered as the "best" model. The BIC value is 584926. Table 6.3 presents the classification of the group membership with the true label.

Table 6.3: Cross-tabulation of true and predicted classification for our chosen model (EVV, $G = 2$, $d = 2$, Browne-McNicholas bound) for the mushroom data.

	1	2
Edible	527	3681
Poisonous	3469	477

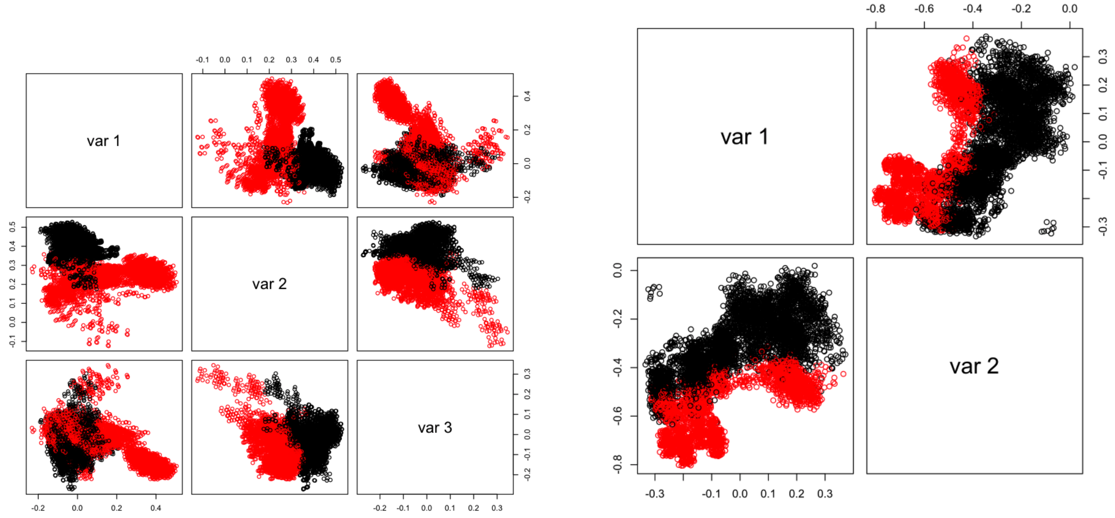
6.4.1.1 A Comparison of variational bound: Böhning and Browne-McNicholas

The key statistics on the best models via Böhning bound and Browne-McNicholas bound are shown in Table 6.4. The highest ARI value (0.57) is obtained using the Browne-McNicholas bound, which can be expected, because the Browne-McNicholas bound is the sharp quadratic bound of the LSE function. However, the Browne-McNicholas bound has much higher computational complexity. The speed issue becomes serious when dealing with categorical variables. For computational simplicity, we use Böhning bound for the next data example.

Table 6.4: A comparison of two different variational bounds.

	Variational Bound	G	D	BIC	$\mathbf{\Sigma}_g$	ARI	Time per Iteration
1	Böhning	2	3	591379	VEV	0.53	20 sec
2	Browne-McNicholas	2	2	584926	EVV	0.57	121 sec

The estimated posterior mean with true group labels using different bounds are presented in Figure 6.1.



(a) Böhning bound

(b) Browne-McNicholas bound

Figure 6.1: Plots of the estimated posterior mean for different bounds.

6.4.2 U.S. Congressional Voting

The U.S. congressional voting data set is also used to illustrate the MMCLT model. The responses are coded into three categories ($K = 3$): 1=yes, 2=no and 3=undecided and we treat response "no" ($k = 2$) as a reference category. The fourteen MMCLT models were fitted to these data for $d = 1, 2, \ldots, 5$ and $G = 1, 2, \ldots, 5$. The minimum BIC (BIC= 20197) occurs at the 3-group, 2-dimensional latent trait model and $\boldsymbol{\Sigma}_g = S_g\boldsymbol{Q}_g\boldsymbol{A}_g\boldsymbol{Q}_g$, which is considered as the "best" model. The divergence of the estimation occurs sometimes due to the small sample size. One or more of the elements in $\boldsymbol{\mu}$ appear to be increasing without limit. However, the divergence of the estimation happens much less when using the fixed Böhning bound than the Browne-McNicholas bound. The classification table of the group membership with party membership is presented in Table 6.5. A 3-components and 2-dimensional latent trait model is selected according to our model selection criteria. Group 1 consists mainly of Democratic congressman, and Group 2 consists mainly of Republican congressman. Group 3 is a small group consists of voters from both parties.

Table 6.5: Cross-tabulation of party and predicted classification for our chosen model (VVV, $G = 3$, $d = 2$) for the U.S. Congressional Voting Data.

	1	2	3
Republican	35	225	7
Democrat	141	12	15

A visual representation of the estimated posterior mean of the best model is presented (Figure 6.2). Group 1 and Group 2 are well separated. Group 3 is a very small group consists of voters from both group.

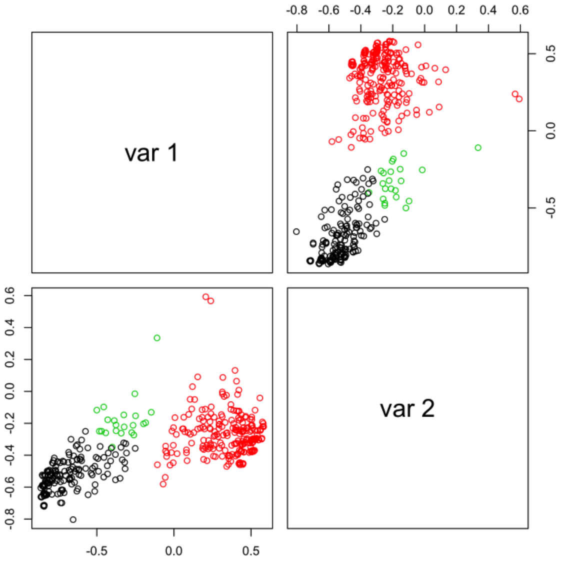


Figure 6.2: Projection of the estimated posterior mean for the best model with true group labels

6.5 Discussion

The mixture of multinomial latent trait models with common slope parameters (MMCLT) provides an additional methodology for model-based clustering of high-dimensional multinomial categorical data. A comparison of two different variational lower bounds (Böhning bound vs. Browne-McNicholas bound) has been carried out. The Browne-McNicholas bound is the first direct extension of the Jaakkola bound to the general categorical case. However, we demonstrate that the Böhning bound is useful and efficient when the number of categories K is large.

Chapter 7

Conclusions

7.1 Summary

The work developed in this thesis represents a significant contribution to the growing body of work on mixture models capable of clustering high-dimensional continuous and non-continuous data. In Chapter 3, we introduce asymmetric clustering for high-dimensional data via a mixture of joint generalized hyperbolic models. This is a novel approach which is applicable to high, and potentially very-high, dimensional continuous data. The use of generalized hyperbolic distribution is particularly useful for clustering applications to avoid making assumptions about the distribution of the underlying groups.

In the next three chapters, we develop models for model-based clustering of high-dimensional binary and categorical data, a topic that has received relatively little attention. In Chapter 4, we explore the possibility of discovering "extreme patterns" of binary data while clustering by drawing ideas from the mixture of contaminated Gaussian distributions. This is the first instance of a mixture model handling binary data with possible extreme patterns.The PMLTM is developed in Chapter 5. The motivation is to develop a model-based clustering framework for very-high dimensional binary data (e.g., text data). The proposed gamma-Laplace penalties on the slope parameters successfully reduce the number of free parameters and identify non-informative variables to clustering. The component-specific tuning parameters avoid the over-penalization that can occur when inferring a shared tuning parameter on clustered data.

In Chapter 6, a mixture of latent trait models is extended by using a multinomial logistic response function to accommodate categorical data, a topic that has remained relatively unexplored in clustering literature. We use two variational lower bounds to approximate the likelihood. We also apply both bounds to real data in order to explore the speed vs accuracy trade-off.

7.2 Future Work

7.2.1 Improvements to Computational Efficiency

There are computational challenges to be addressed in fitting the MJGHM-HDClust model in Chapter 3. We will investigate the possibility to avoid calculating the full $p \times p$ eigenvector matrix $\mathbf{\Gamma}$ which will greatly reduce the runtime. Alternatives to BIC for selecting the dimensionality of the component-specific subspace q_g can also be studied.

In Chapter 5, we demonstrate the usefulness of using Python when compared to R. Developing analogous Python code for MJGHM-HDClust (Chapter 3), MLTCG (Chapter 4), and MMCLT (Chapter 6) will possibly improve the runtime dramatically. In addition, parallel computing implementation are possible solutions to help address the computational challenge.

7.2.2 Parsimonious Extensions to the PMLTM Model

A parsimonious family of the mixture of latent trait models is developed by using common slope parameters and applying restrictions to the components of the decomposed covariance matrices in Tang *et al.* (2015). Analogous families of parsimonious models could be developed for the PMLTM model to further reduce the number of parameters to be estimated, which would make this model even more powerful for the analysis of high-dimensional binary data. Probabilities on restricting the turning parameter $\boldsymbol{\lambda}$ can be explored as well.

7.2.3 Properties of the Variational Approximation in the MMCLT Model

We wish to study the asymptotic properties of the variational approximation in the MMCLT model in Chapter 6. A simulation study can be carried out to investigate the speed vs. accuracy trade-off in detail between Böhning bound and Browne-McNicholas bound. Additionally, we are interested in other approximation methods to the multinomial likelihood.

Bibliography

Aitken, A. C. (1926). A series formula for the roots of algebraic and transcendental equations. *Proceedings of the Royal Society of Edinburgh*, **45**(01), 14–22.

Aitkin, M. and Wilson, G. T. (1980). Mixture models, outliers, and the EM algorithm. *Technometrics*, **22**(3), 325–331.

Anderberg, M. R. (1973). Cluster analysis for applications, monographs and textbooks on probability and mathematical statistics.

Baek, J., McLachlan, G. J., and Flack, L. K. (2010). Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(7), 1298–1309.

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

Bergé, L., Bouveyron, C., and Girard, S. (2012). HDclassif: An R package for model-based clustering and discriminant analysis of high-dimensional data. *Journal of Statistical Software*, **46**(6), 1–29.

Biernacki, C., Celeux, G., and Govaert, G. (2010). Exact and Monte Carlo calculations of integrated likelihoods for the latent class model. *Journal of Statistical Planning and Inference*, **140**(11), 2991–3002.

Bock, R. D. and Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. *Psychometrika*, **46**(4), 443–459.

Böhning, D. (1992). Multinomial logistic regression algorithm. *Annals of the Institute of Statistical Mathematics*, **44**(1), 197–200.

Böhning, D., Dietz, E., Schaub, R., Schlattmann, P., and Lindsay, B. G. (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**(2), 373–388.

Bolt, D. M., Cohen, A. S., and Wollack, J. A. (2001). A mixture item response model for multiple-choice data. *Journal of Educational and Behavioral Statistics*, **26**(4), 381–409.

Bouveyron, C., Girard, S., and Schmid, C. (2007). High-dimensional data clustering. *Computational Statistics & Data Analysis*, **52**(1), 502–519.

Browne, R. P. and McNicholas, P. D. (2012). Model-based clustering, classification, and discriminant analysis of data with mixed type. *Journal of Statistical Planning and Inference*, **142**(11), 2976–2984.

Browne, R. P. and McNicholas, P. D. (2014). Estimating common principal components in high dimensions. *Advances in Data Analysis and Classification*, **8**(2), 217–226.

Browne, R. P. and McNicholas, P. D. (2015a). A mixture of generalized hyperbolic distributions. *Canadian Journal of Statistics*, **43**(2), 176–198.

Browne, R. P. and McNicholas, P. D. (2015b). Multivariate sharp quadratic bounds via sigma-strong convexity and the Fenchel connection. *Electronic Journal of Statistics*, **9**(2), 1913–1938.

Browne, R. P., Elsherbiny, A., and McNicholas, P. D. (2015). *mixture: Mixture Models for Clustering and Classification*. R package version 1.4.

Cagnone, S. and Viroli, C. (2012). A factor mixture analysis model for multivariate binary data. *Statistical Modelling*, **12**(3), 257–277.

Celeux, G. and Govaert, G. (1991). Clustering criteria for discrete data and latent class models. *Journal of Classification*, **8**(2), 157–176.

Celeux, G. and Govaert, G. (1995). Gaussian parsimonious clustering models. *Pattern Recognition*, **28**(5), 781–793.

Crespin, M. H. and Rohde, D. W. (2010). Dimensions, issues, and bills: Appropriations voting on the house floor. *The Journal of Politics*, **72**(4), 976–989.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **53**(1), 1–38.

DeSantis, S. M., Houseman, E. A., Coull, B. A., Stemmer-Rachamimov, A., and Betensky, R. A. (2008). A penalized latent class model for ordinal data. *Biostatistics*, **9**(2), 249–262.

Di Lascio, F. M. L. and Giannerini, S. (2012). A copula-based algorithm for discovering patterns of dependent observations. *Journal of Classification*, **29**(1), 50–75.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, **96**(456), 1348–1360.

Feinerer, I. and Hornik, K. (2015). *tm: A Framework for Text Mining Applications within R*. R package version 0.7-1.

Forbes, F. and Wraith, D. (2014). A new family of multivariate heavy-tailed distributions with variable marginal amounts of tailweight: application to robust clustering. *Statistics and Computing*, **24**(6), 971–984.

Forina, M., Armanino, C., Castino, M., and Ubigli, M. (1986). Multivariate data analysis as a discriminating method of the origin of wines. *Vitis*, **25**(3), 189–201.

Fraley, C. and Raftery, A. E. (1998). How many clusters? which clustering method? answers via model-based cluster analysis. *The Computer Journal*, **41**(8), 578–588.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, **97**(458), 611–631.

Franczak, B. C., Tortora, C., Browne, R. P., and McNicholas, P. D. (2015). Unsupervised learning via mixtures of skewed distributions with hypercube contours. *Pattern Recognition Letters*, **58**, 69–76.

Galimberti, G., Montanari, A., and Viroli, C. (2009). Penalized factor mixture analysis for variable selection in clustered data. *Computational Statistics & Data Analysis*, **53**(12), 4301–4310.

Ghahramani, Z. and Hinton, G. E. (1996). The EM algorithm for mixtures of factor analyzers. Technical report, Technical Report CRG-TR-96-1, University of Toronto.

Gollini, I. and Murphy, T. B. (2014). Mixture of latent trait analyzers for model-based clustering of categorical data. *Statistics and Computing*, **24**(4), 569–588.

Goodman, L. A. (1974). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, **61**(2), 215–231.

Haughton, D. M. A. (1988). On the choice of a model to fit data from an exponential family. *The Annals of Statistics*, **16**(1), 342–355.

Houseman, E. A., Marsit, C., Karagas, M., and Ryan, L. M. (2007). Penalized item response theory models: application to epigenetic alterations in bladder cancer. *Biometrics*, **63**(4), 1269–1277.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, **2**(1), 193–218.

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

Jaakkola, T. S. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, **10**(1), 25–37.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data.* Upper Saddle River: Prentice-Hall, Inc.

Jajuga, K. and Papla, D. (2006). Copula functions in model-based clustering. In *From Data and Information Analysis to Knowledge Engineering*, pages 606–613. New York: Springer.

Jorgensen, M. (2004). Using multinomial mixture models to cluster internet traffic. *Australian & New Zealand Journal of Statistics*, **46**(2), 205–218.

Karlis, D. and Meligkotsidou, L. (2007). Finite mixtures of multivariate Poisson distributions with application. *Journal of Statistical Planning and Inference*, **137**(6), 1942–1960.

Khan, M. E., Bouchard, G., Murphy, K. P., and Marlin, B. M. (????). Variational bounds for mixed-data factor analysis. In *Advances in Neural Information Processing Systems*, volume 23, pages 1108–1116, Vancouver, Canada.

Kiers, H. A. L. (2002). Setting up alternating least squares and iterative majorization algorithms for solving various matrix optimization problems. *Computational Statistics & Data Analysis*, **41**(1), 157–170.

Knott, M. and Bartholomew, D. J. (1999). *Latent variable models and factor analysis*. Number 7. London: Edward Arnold.

Kosmidis, I. and Karlis, D. (2016). Model-based clustering using copulas with applications. *Statistics and Computing*, **26**(5), 1079–1099.

Krishnapuram, B., Carin, L., Figueiredo, M. A., and Hartemink, A. J. (2005). Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(6), 957–968.

Lichman, M. (2013). UCI machine learning repository. http://archive.ics.uci.edu/ml.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297.

Marbac, M., Biernacki, C., and Vandewalle, V. (2014a). Finite mixture model of conditional dependencies modes to cluster categorical data. *arXiv preprint arXiv:1402.5103*.

Marbac, M., Biernacki, C., and Vandewalle, V. (2014b). Model-based clustering of Gaussian copulas for mixed data. *arXiv preprint arXiv:1405.1299*.

Mazumder, R., Friedman, J. H., and Hastie, T. (2012). SparseNet: Coordinate descent with nonconvex penalties. *Journal of the American Statistical Association*, **106**(495), 1125–1138.

McLachlan, G. J. and Krishnan, T. (2007). *The EM Algorithm and Extensions.* New York: John Wiley & Sons.

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

McLachlan, G. J. and Peel, D. (2000b). Mixtures of factor analyzers. In *Finite Mixture Models*, pages 238–256. New York: John Wiley & Sons.

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

McNeil, A. J., Frey, R., and Embrechts, P. (2015). *Quantitative Risk Management: Concepts, Techniques and Tools.* Princeton: Princeton University Press.

McNicholas, P. D. (2016). *Mixture Model-Based Classification.* Boca Raton: Chapman & Hall/CRC Press.

McNicholas, P. D. and Murphy, T. B. (2008). Parsimonious Gaussian mixture models. *Statistics and Computing*, **18**(3), 285–296.

McNicholas, P. D. and Murphy, T. B. (2010). Model-based clustering of microarray expression data via latent gaussian mixture models. *Bioinformatics*, **26**(21), 2705–2712.

McNicholas, P. D., Jampani, K. R., McDaid, A. F., Murphy, T. B., and Banks, L. (2011). *pgmm: Parsimonious Gaussian Mixture Models.* R package version 1.2.

Meng, X. L. and Rubin, D. B. (1993). Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, **80**(2), 267–278.

Murray, P. M., Browne, R. P., and McNicholas, P. D. (2014). Mixtures of skew-t factor analyzers. *Computational Statistics and Data Analysis*, **77**, 326–335.

Muthen, B., Asparouhov, T., *et al.* (2006). Item response mixture modeling: Application to tobacco dependence criteria. *Addictive Behaviors*, **31**(6), 1050–1066.

Park, T. and Casella, G. (2008). The Bayesian Lasso. *Journal of the American Statistical Association*, **103**(482), 681–686.

Poskitt, D. S. (1987). Precision, complexity and Bayesian model determination. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **49**(2), 199–208.

Prause, K. (1999). *The Generalized Hyperbolic Model: Estimation, Financial Derivatives, and Risk Measures*. Ph.D. thesis, Freiburg im Breisgau: Albert Ludwigs University of Freiburg.

Punzo, A. and McNicholas, P. D. (2016). Parsimonious mixtures of multivariate contaminated normal distributions. *Biometrical Journal*, **58**(6), 1506–1537.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, **66**(336), 846–850.

Ratanamahatana, C. A. and Gunopulos, D. (2002). Scaling up the naive Bayesian classifier: Using decision trees for feature selection. In *Proceedings of the IEEE Workshop on Data Cleaning and Preprocessing (DCAP). at IEEE International Conference on Data Mining*, pages 475–487.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics.*, **6**(2), 461–464.

Street, N. W., Wolberg, W. H., and Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, volume 1905, pages 861–870, San Jose, California.

Tang, Y., Browne, R. P., and McNicholas, P. D. (2015). Model-based clustering of high-dimensional binary data. *Computational Statistics & Data Analysis*, **87**, 84–101.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **58**(1), 267–288.

Tipping, M. E. and Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **61**(3), 611–622.

Tortora, C., Franczak, B. C., Browne, R. P., and McNicholas, P. D. (2014). A mixture of coalesced generalized hyperbolic distributions. *arXiv preprint arXiv:1403.2332*.

Vermunt, J. K. (2003). Multilevel latent class models. *Sociological Methodology*, **33**(1), 213–239.

Vermunt, J. K. (2007). Multilevel mixture item response theory models: An application in education testing. In *Proceedings of the 56th session of the International Statistical Institute.*, pages 22–28, Lisbon, Portugal.

Vrac, M., Billard, L., Diday, E., and Chédin, A. (2012). Copula analysis of mixture models. *Computational Statistics*, **27**(3), 427–457.

Ward J., J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, **58**(301), 236–244.

Wolfe, J. H. (1963). *Object cluster analysis of social areas*. Ph.D. thesis, Berkeley: University of California.

Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *The Annals of Mathematical Statistics*, **39**(1), 209–214.

Yoshida, R., Higuchi, T., and Imoto, S. (2004). A mixed factors model for dimension reduction and extraction of a group structure in gene expression data. In *Proceedings. 2004 IEEE Computational Systems Bioinformatics Conference, 2004. CSB 2004.*, pages 161–172, Stanford, California.

Yuan, J. and Wei, G. (2014). An efficient Monte Carlo EM algorithm for Bayesian Lasso. *Journal of Statistical Computation and Simulation*, **84**(10), 2166–2186.