2014

Methods for Clustering Mixed Data

JeanMarie L. Hendrickson
University of South Carolina - Columbia

Methods for Clustering Mixed Data

by

JeanMarie Hendrickson

Bachelor of Science
University of Minnesota, Duluth 2006
Master of Science
Minnesota State University, Mankato 2009

Submitted in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy in

Statistics

College of Arts and Sciences

University of South Carolina

2014

Accepted by:

David Hitchcock, Major Professor

John Grego, Committee Member

Joshua Tebbs, Committee Member

Diansheng Guo, Committee Member

Lacy Ford, Vice Provost and Dean of Graduate Studies

Dedication

I dedicate this to John and Eleanor Warntjes, Fred and Norma Thompson, and Norman and Laura Thompson.

Acknowledgments

I'd like to thank my advisor Dr. David Hitchcock for guiding my research, editing my dissertation, writing recommendation letters and for giving me advice whenever I needed it. I'd also like to thank the members of my dissertation committee.

Next, I'd like to thank my family. Thank you mom and dad for pushing me each step of the way and supporting me for all these years. I truly couldn't have asked for better parents. Thank you Josh, Jenna, Jessica and Jerica for being there for me when I needed encouragement and for when I needed a laugh. Nolan, your favorite aunt finished her dissertation so that she could finally hold you! Hunter, thank you for being the best stepson; you are truly an amazing little boy! I am truly blessed to have had many friends and family that have given me encouragement over the years and I am thankful for each one of you.

Lastly, I'd like to thank my husband. Jason, without your love, support, encouragement and humor, I'd have never made it through these last few years of my education, sanely. Thank you for making me smile.

Abstract

We give a brief introduction to cluster analysis and then propose and discuss a few methods for clustering mixed data. In particular, a model-based clustering method for mixed data based on Everitt's (1988) work is described, and we use a simulated annealing method to estimate the parameters for Everitt's model. A penalized log likelihood with the simulated annealing method is proposed as a remedy for the parameter estimates being drawn to extremes. Everitt's approach and the proposed method are compared based on their performance in clustering simulated data. We then use the penalized log likelihood method on a heart disease data set, in which our clustering result was compared to an expert's diagnosis using the Rand Index. We also describe an extension to Gower's (1971) coefficient, based on Kaufman and Rousseeuw's (1990) definition, which allows for other types of variables to be used in the coefficient. A clustering algorithm based on the extended Gower coefficient is used to to find a clustering solution for a buoy data set to see how well this method classified a variety of sites into their "true" regions. To display how the method based on the extended Gower coefficient performed in clustering data having a variety of structures, we show the results of a simulation study.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

As most real data contain different types of variables, an important area in cluster analysis deals with clustering mixed data. Mixed data sets arise when the variables observed consist of several types, e.g., continuous, categorical, functional, directional, etc. As will be mentioned in the literature review, there are several options for clustering mixed data, and each comes with its own problems. We will examine model-based clustering and an extension to the Gower coefficient for mixed data. When implementing model-based clustering for mixed data, the main problem is estimating the parameters for the model, especially with a large number of variables. Many methods of clustering objects rely explicitly or implicitly on the dissimilarities between pairs of objects, which we discuss in Chapter 2. To motivate our discussion of the extended Gower coefficient, there is a need for dissimilarities to be defined for other types of variables, besides those that have already been discussed in the literature.

We estimated the parameters for a model-based clustering approach for mixed data (following the model proposed by Everitt (1988)) using a simulated annealing approach. Our parameter estimates using this approach alone were drawn to extreme values for each parameter, which is when we decided that using a penalized log-likelihood within the simulated annealing would deter the parameters from being drawn to those extremes. This approach worked well our simulated data. We also applied our approach to a heart disease data set to investigate how accurate the clustering solution using our approach was to an expert's diagnosis.

Another approach we investigate for clustering mixed data was based on an extension of the Gower (1971) coefficient. For this, we follow the definition of the Gower dissimilarity of Kaufman and Rousseeuw (1990) and extend the dissimilarity measure they described to create a generalization of the Gower coefficient. While they only describe dissimilarities for binary (or nominal) and continuous variables, we extend the Gower coefficient, defining dissimilarities for directional and functional variables as well. For the functional data, we describe several different options to calculate the dissimilarity between two curves.

In this dissertation, the second chapter contains a literature review, which introduces cluster analysis and then summarizes the literature for clustering mixed data. Chapter 3 describes our stochastic search method based on the simulated annealing, and compares our results with those of Everitt's (1988). In this chapter, we also describe a penalized log-likelihood approach. In Chapter 4, we describe an extension of the Gower coefficient that can be used for mixed data. We include a real data example, as well as a simulation study we undertook using the extended Gower coefficient.

Chapter 2

Literature Review

2.1 Background on Cluster Analysis

Cluster analysis involves examining multivariate data and grouping objects into clusters. These clusters are constructed in such a way that the items within a cluster are homogeneous, or highly similar, and items in separate clusters are highly dissimilar. Some commonly used clustering procedures include agglomerative hierarchical methods, k-means type methods and classification maximum likelihood methods, which include model-based clustering.

Agglomerative hierarchical methods involve several steps, producing solutions ranging from partitions of n clusters each containing one individual to one cluster containing all individuals (Everitt et al. 2011). The dissimilarities between objects are often measured by distances, of which the Euclidean distance is the most widely used measure; the formula for the Euclidean distance for two p-dimensional observations $\mathbf{x}$ and $\mathbf{y}$ is as follows:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_p - y_p)^2}.$$

To see how this is done we will look at an example data set. The data are from Kaufman and Rousseeuw (1990, p. 200). The dissimilarity matrix $\mathbf{D}$ for the data in Table 2.1 was computed using Euclidean distance.

Table 2.1: Data from Kaufman and Rousseeuw (1990)

Object	Variable 1	Variable 2
1	2	2
2	5.5	4
3	5	5
4	1.5	2.5
5	1	1
6	7	5
7	5.75	6.5

$$\mathbf{D} = \begin{bmatrix} 0 & 4.03 & 4.24 & 0.71 & 1.41 & 5.83 & 5.86 \\ 4.03 & 0 & 1.12 & 4.27 & 5.41 & 1.80 & 2.51 \\ 4.24 & 1.12 & 0 & 4.30 & 5.66 & 2.00 & 1.68 \\ 0.71 & 4.27 & 4.30 & 0 & 1.58 & 6.04 & 5.84 \\ 1.41 & 5.41 & 5.66 & 1.58 & 0 & 7.21 & 7.27 \\ 5.83 & 1.80 & 2.00 & 6.04 & 7.21 & 0 & 1.95 \\ 5.86 & 2.51 & 1.68 & 5.84 & 7.27 & 1.95 & 0 \end{bmatrix}$$

As an example of how the dissimilarities are computed, consider the calculation of the dissimilarity between the first two objects,

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{(2-5.5)^2 + (2-4)^2} = 4.03.$$

Some other popular measures of distance between items are the Minkowski metric,

$$d(\mathbf{x}, \mathbf{y}) = \left[\sum |x_i - y_i|^m\right]^{\frac{1}{m}},$$

the Canberra metric,

$$d(\mathbf{x}, \mathbf{y}) = \sum \frac{|x_i - y_i|}{(x_i + y_i)},$$

and the Czekanowski coefficient,

$$d(\mathbf{x}, \mathbf{y}) = 1 - \frac{2\sum \min(x_i, y_i)}{\sum (x_i + y_i)}.$$

These distance measures can be used in single linkage, complete linkage and group average clustering, which are examples of agglomerative hierarchical clustering procedures. Single linkage clustering defines the distance between two clusters A and B to be

$$d(A,B) = \min_{i \in A, j \in B} d_{ij}.$$

Complete linkage clustering defines the distance between two clusters, A and B, to be

$$d(A,B) = \max_{i \in A, j \in B} d_{ij};$$

here, all of the objects in a cluster are guaranteed to be within some maximum distance of each other. Group average clustering defines the distance between two clusters, A and B, to be the average of the distances between all item pairs (i, j), where i belongs to cluster A and j belongs to cluster B, i.e.,

$$d(A,B) = \frac{1}{n_A n_B} \sum_{i \in A} \sum_{j \in B} d_{ij},$$

where n_A is the number of objects in cluster A and n_B is the number of objects in cluster B and the d_{ij} values represent distances between objects. In all these methods, clusters are formed by merging the groups that are the most similar, or that have the smallest distance from each other (where the distance is defined as above for single linkage, complete linkage and group average clustering). One of the drawbacks to agglomerative hierarchical techniques is that once an item is merged into a cluster, this merge cannot be undone; hence, if a merge decision is not chosen well, it may lead to low quality clusters.

K-means clustering methods seek to group data into k groups by minimizing some criterion; the within-group sum of squares over all variables is often used as the criterion to minimize (Everitt et al. 2011). The k-means method is a partitioning technique which uses variation about the mean of the cluster to measure objects' similarity. This method begins with an initial partition of the items into the k groups.

Then, the items are moved individually from one cluster to another and the criterion is calculated at each move; the move to be kept is the one that minimizes the criterion. The k-means method proceeds by continuing to move items individually and to calculate the criterion until no move causes the criterion to be decreased further. Some of the problems that can arise with k-means clustering methods are as follows: It can only be applied when the mean of a cluster is defined; one must specify the number of groups in advance; finally, it is sensitive to outliers.

Model-based clustering offers solutions to some of the problems that arise from the previously discussed methods, namely the choice of a clustering method, the estimation of the number of clusters and the lack of formal inference available without a model. Model-based clustering methods include cluster analyses on finite mixture densities (Everitt et al. 2011); these methods try to construct a model for the set of clusters, and find the best fit of the data to the model. Finite mixture densities are of the form:

$$f(\mathbf{x};\mathbf{p},\boldsymbol{\theta}) = \sum_{j=1}^{k} p_j g_j(\mathbf{x};\boldsymbol{\theta}),$$

where $\mathbf{x}$ is a p-dimensional random variable, $p_j \geq 0$ are the mixing proportions, where $\sum p_j = 1$ and $g_j(\cdot|\cdot)$, for $j = 1, ..., k$, are the component densities. As stated by Everitt et al. (2011), "Finite mixtures provide suitable models for cluster analysis if we assume that each group of observations in a data set suspected to contain clusters comes from a population with a different probability distribution." To define clusters using finite mixture densities, one would group observations based on the maximum value of the following cluster membership probability:

$$P(\text{cluster } j|\mathbf{x}_i) = \frac{\hat{p}_j g_j(\mathbf{x}_i, \hat{\boldsymbol{\theta}})}{f(\mathbf{x}_i; \hat{\mathbf{p}}, \hat{\boldsymbol{\theta}})},$$

with this probability being calculated after estimating the parameters of the distribution. These parameters are often estimated using either the expectation-maximization algorithm or Bayesian estimation methods.

Another well-known model-based formulation is the classification maximum likelihood procedure, which Banfield and Raftery (1993) and Fraley and Raftery (2002) extended, and which was originally proposed by Scott and Symons (1971). The procedure first assumes that there are d subpopulations, each corresponding to a cluster, with the subpopulation densities denoted by $g_j(\mathbf{x}; \boldsymbol{\theta}_j)$, where $\boldsymbol{\theta}_j$ is an unknown vector of parameters. Also, let $\boldsymbol{\gamma}' = [\gamma_1, \gamma_2, ..., \gamma_n]$ be a vector of cluster labels, where $\gamma_i = j$ if the i^{th} observation is from the j^{th} population. Then to fit the model, one finds the $\boldsymbol{\gamma}$ and the $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_k)$ that maximize the likelihood function associated, namely $L(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \prod_{i=1}^{n} g_{\gamma_i}(\mathbf{x}_i; \boldsymbol{\theta}_{\gamma_i})$. Clusters are then merged if they yield the greatest increase in the classification likelihood. The biggest difference between the finite mixture densities approach and this one is that for the classification maximum likelihood, each $\mathbf{x}_i$ is assumed to be from a single distribution whereas in the finite mixture approach the $\mathbf{x}_i$ are assumed to be from mixture distributions.

It can be noted that agglomerative hierarchical and k-means clustering methods are used more as "exploratory" tools, whereas model-based clustering allows for formal inference, once a model is chosen. Also, some of the concerns and problems that arise with hierarchical and k-means clustering can be handled with model-based clustering. This has led to more work being done in the model-based clustering realm.

2.2 Review of Previous Methodology for Clustering Mixed Data

There are several possible approaches when clustering data consisting of variables of different types, commonly called mixed data. One option is to perform a separate cluster analysis for each type of variable; one problem with this method is that the conclusions from these separate analyses may not agree (Kaufman and Rousseeuw, 1990). Another option is to convert all of the variables to a single type of variable and then perform a cluster analysis on the data. One could transform the data to have the variables be all interval-scaled or all binary. However, treating binary variables as if they were continuous introduces an arbitrary numerical coding to the categories. Categorizing continuous variables via thresholds raises the question of what threshold value is appropriate and furthermore sacrifices a great deal of information (Kaufman and Rousseeuw, 1990).

DISSIMILARITY MEASURES FOR MIXED DATA

Another approach to clustering mixed data is to combine the variables into a single proximity matrix (Kaufman and Rousseeuw, 1990). In cluster analysis, the focus is on finding the dissimilarity between two objects, but one could also find the similarity between objects. Gower (1971) came up with a coefficient to measure the similarity between two objects based on mixed data. We describe the Gower coefficient further in Chapter 4.

Gower's formulation assigned equal weight to either one of the variable types (continuous or binary) (Chae, Kim, and Yang, 2006). Since this can lead to clusters that are dominated by one kind of variable type, assigning weights to the variable types may alleviate this problem (Chae, Kim, and Yang, 2006). The dissimilarity measure described by Chae, Kim and Yang (2006), is as follows:

$$d^*_{ij} = \tau_{ij} \sum_{l=1}^{c} \frac{1}{c}\Big(\frac{|x_{il} - x_{jl}|}{R_l}\Big) + (1 - \tau_{ij})\sqrt{1 - \frac{\Sigma_{l=c+1}^{r} s_{ijl}}{\Sigma_{l=c+1}^{r} w_{ijl}}},$$

where τ_{ij}, $0 \leq \tau_{ij} \leq 1$, is a balancing weight such that

$$\tau_{ij} = \begin{cases} 1.0 - \frac{|\rho^c_{ij}|}{|\rho^c_{ij}|+|\rho^d_{ij}|} & \text{if } 1.0 < \frac{|\rho^c_{ij}|}{|\rho^d_{ij}|}, \\ 1.0 - \frac{|\rho^d_{ij}|}{|\rho^c_{ij}|+|\rho^d_{ij}|} & \text{if } 1.0 > \frac{|\rho^c_{ij}|}{|\rho^d_{ij}|} \\ 0.5 & \text{if } |\rho^c_{ij}| = |\rho^d_{ij}| \end{cases},$$

with $-1.0 \leq \rho^c_{ij}$ being a similarity measure for the quantitative variables, and with $\rho^d_{ij} \leq 1.0$ being a similarity measure for the binary variables, $i = 2, 3, \ldots, n$, $j = 1, 2, \ldots, n-1$, $i > j$. R_l is the range of the lth variable, $w_{ijl} = 1.0$ for continuous variables, $s_{ijl} = 1.0$ if $x_i = x_j$ and 0 otherwise, for binary variables, and w_{ijl} is either 0 or 1, for binary variables, depending on whether the comparison between the ith and jth objects is valid for the lth variables. Chae, Kim and Yang (2006) use the Pearson correlation coefficient for ρ_{ij}^c and the product moment correlation coefficient for ρ_{ij}^d, which is the Pearson correlation coefficient applied to binary data.

Ichino and Yaguchi (1994) proposed an interesting dissimilarity measure for mixed-type variables. The approach, called a generalized Minkowski metric, can handle continuous, discrete, ordinal and nominal variables. It can also deal with "tree-structured" variables whose values are a finite set of nominal values. The dissimilarity itself is based on a Cartesian space model, involving the Cartesian meet and Cartesian join operations, which have various outcomes depending on the nature of the variable under consideration (see Ichino and Yaguchi (1994) for details). It can be shown that the proposed dissimilarity measure is a metric distance. Ichino and Yaguchi (1994) also suggest alternate versions of the measure may be (1) normalized to account for the variables' differing measurement scales or (2) weighted differently for the different variables. The family of dissimilarity metrics is quite unusual compared to competing measures, but is useful, being general enough to handle a wide range of variable types, including "trees."

Friedman and Meulman (2004) describe two algorithms for clustering objects on subsets of attributes (COSA), based on previously defined distance measures. They note that the COSA algorithms as they are primarily described focus on clustering objects based on unspecified similar joint values for the attributes. However, with some slight modifications, the COSA algorithms can be used to cluster based on certain values for the attributes, which they refer to as "Single-target clustering", or based on two extreme values, which they refer to as "Dual-target clustering" (Friedman and Meulman, 2004).

An important early work on model-based clustering of mixed data was that of Everitt (1988). Everitt (1988) proposed a clustering model for data sets with both continuous variables and binary or ordinal variables. For this model, there are several assumptions. One is that an observed vector $\boldsymbol{x}$, which contains $p+q$ random variables, has density function

$$f(\boldsymbol{x}) = \sum_{i=1}^{c} p_i MVN_{(p+q)}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}),$$

where c is the number of clusters, $p_1, p_2, \ldots, p_c$ are the mixing proportions such that $\sum_{i=1}^{c} p_i = 1$ and $MVN(\cdot,\cdot)$ represents a multivariate normal density. Everitt then proposed that the binary and ordinal variables come from an underlying continuous distribution, where the q ordinal or categorical variables are “constructed” by setting certain threshold values to be cut-off points. The $x_{p+1}, x_{p+2}, \ldots, x_{p+q}$ are observed only through the categorized variables $z_1, z_2, \ldots, z_q$, where the z_j's are constructed in the following manner:

$$z_j = \begin{cases} 1 & \text{if} \quad -\infty = \alpha_{ij1} < x_{p+j} < \alpha_{ij2}, \\ 2 & \text{if} \quad \alpha_{ij2} < x_{p+j} < \alpha_{ij3}, \\ k_j & \text{if} \quad \alpha_{ijk_j} < x_{p+j} < \alpha_{ijk_j+1} = \infty. \end{cases}$$

The α_{ijl}, $i = 1, \ldots, c$, $j = 1, \ldots, q$, $l = 2, \ldots, k_j$, values are the threshold values used to construct the ordinal variables, $z_1, z_2, \ldots, z_q$, from the continuous variables, $x_{p+1}, x_{p+2}, \ldots, x_{p+q}$. The density he proposed is

$$h(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{c} p_i MVN_{(p)}(\boldsymbol{\mu}_i^{(p)}, \boldsymbol{\Sigma}_p) \int_{a_1}^{b_1} \cdots \int_{a_q}^{b_q} MVN_q(\boldsymbol{\mu}_i^{(q|p)}, \boldsymbol{\Sigma}_{q|p}) dy_1 \ldots dy_q,$$

where $\boldsymbol{\mu}_i^{(q|p)} = \boldsymbol{\Sigma}_{pq}' \boldsymbol{\Sigma}_p^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i^{(p)})$ and $\boldsymbol{\Sigma}_{q|p} = \boldsymbol{\Sigma}_q - \boldsymbol{\Sigma}_{pq}' \boldsymbol{\Sigma}_p^{-1} \boldsymbol{\Sigma}_{pq}$. These are, respectively, the mean and covariance matrix for the conditional density of $x_{p+1}, \ldots, x_{p+q}$ given $x_1, \ldots, x_p$. $\boldsymbol{\Sigma}_{pq}$ is the matrix of covariances between $x_1, \ldots, x_p$ and $x_{p+1}, \ldots, x_{p+q}$; $\boldsymbol{\Sigma}_p$ is the covariance matrix of $x_1, \ldots, x_p$; $\boldsymbol{\Sigma}_q$ is the covariance matrix of $x_{p+1}, \ldots, x_{p+q}$, where the diagonal elements are assumed (without loss of generality since $x_{p+1}, ..., x_{p+q}$ are unobserved) to be 1; and $\boldsymbol{\mu}_i^{(p)}$ is the mean vector of $x_1, \ldots, x_p$ for the objects in cluster i. Now, with a given set of observations, we need to estimate the parameters for this density in order to calculate the probabilities which are used to identify the clusters. To estimate the parameters, we maximize the log-likelihood,

$$\log L = \sum_{i=1}^{c} \log h(\boldsymbol{x}_i, \mathbf{z}_i).$$

Lawrence and Krzanowski (1996) adopt an approach previously known in discriminant analysis of mixed data (Krzanowski, 1993) and in graphical modeling (Whittaker, 1990). The approach replaces the q categorical variables with a single m-cell multinomial variable, where m is the number of possible sets (patterns) of values that the q categorical variables could take. Then the *conditional Gaussian model* assumes a multivariate normal distribution for the p continuous variables that is conditional on which of the m cells the individual falls in. So the probability of falling in the sth cell is p_s, and the vector of continuous variables for the jth individual in the sth cell is $\mathbf{x}_{sj} \sim N(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$. Lawrence and Krzanowski (1996) set all covariance matrices to $\boldsymbol{\Sigma}$, and adapted the resulting likelihood to the clustering problem. They use the EM algorithm analogously to McLachlan and Basford (1988) to estimate parameters, including the mixture subpopulation indicators that serve as cluster-membership parameters. Notably, in a simulation study, the estimated mean vectors for the continuous variables revealed a shrinkage property.

This shrinkage was attributed by Willse and Boik (1999) to a non-identifiability in the conditional Gaussian model. They found that the unrestricted model of Lawrence and Krzanowski (1996) led to multiple equally good solutions to the likelihood equations, and thus they suggested a restricted-model version of this approach.

Moustaki and Papageorgiou (2005) extended the work of Moustaki (1996) and developed a latent class mixture model that handles binary, nominal, ordinal and continuous variables using Bernoulli, multinomial, cumulative-probability multinomial, and normal distributions, respectively. The joint distribution (for the h^{th} object) of the observed variables $\mathbf{x}_h = (x_{1h}, \ldots, x_{ph})'$ is $f(\mathbf{x}_h) = \sum_{j=1}^{K} \eta_j g(\mathbf{x}_h|j)$, where $g(\mathbf{x}_h|j)$ is a density from the exponential family and $h(j|\mathbf{x}_h) = \eta_j g(\mathbf{x}_h|j)/f(\mathbf{x}_h)$ is a posterior probability that object h belongs to class j. The AIC and BIC are suggested for model selection.

Hunt and Jorgensen (1999) and Jorgensen and Hunt (1996) describe an approach to mixture model clustering which, as they say, can be thought of as an extension of some of the earlier approaches. To use the following approach, the observations should be in the form of an $n \times p$ matrix, which come from variables which are thought of as a random sample from $f(x) = \sum \pi_k f_k(x)$ (Hunt and Jorgensen, 1999). The vector of variables $\mathbf{x} = (x_1, \ldots, x_p)'$ has been partitioned into $(\bar{\mathbf{x}}_1', \ldots, \bar{\mathbf{x}}_L')$ and, within each of the K subpopulations, the variables in $\bar{\mathbf{x}}_l$ are independent of the variables in $\bar{\mathbf{x}}_{l'}$ for $l \neq l'$ (Hunt and Jorgensen, 1999). Their MULTIMIX model, for the ith observation, can be written as

$$f(\mathbf{x}_i; \boldsymbol{\phi}) = \sum_{k=1}^{K} \pi_k \prod_{l=1}^{L} f_{kl}(\bar{\mathbf{x}}_{il}; \boldsymbol{\theta}_{kl}),$$

where $\boldsymbol{\theta}_{kl}$ contains the parameters of the distribution f_{kl}, and π_k are the mixing proportions (Hunt and Jorgenson, 1999). Hunt and Jorgenson (1999) mention that if the f_{kl} belong to the exponential family, the model's parameters can then be estimated using the EM algorithm of Dempster et al. (1977).

He, Xu, and Deng (2005) describe a cluster ensemble approach to deal with data that contain both numeric and categorical data. It aims to "combine several runs of different clustering algorithms to get a common partition of the original dataset, aiming for consolidation of results from a portfolio of individual clustering results" (He, Xu, and Deng, 2005). Their approach is to first divide the dataset into two datasets, the categorical dataset and the numeric dataset, then use appropriate, existing clustering algorithms for each. The final step is to combine the clustering results from the categorical and numeric datasets as a categorical dataset, which is then input into a clustering algorithm to achieve the final clustering result (He, Xu, and Deng, 2005). Letting N_c and N_n represent the number of variables that are categorical and numeric, $N_c + N_n = N$ where N is the total number of variables, in many cases, $N_c \neq N_n$; thus, He, Xu, and Deng (2005) assign N_c and N_n as weights. This notion led to the clustering algorithm He, Xu, and Deng (2005) describe for the categorical dataset and the combined clustering categorical dataset, which is the weighted Squeezer algorithm.

Chapter 3

Stochastic Search for Model-Based Clustering of Mixed Data

In this chapter we consider Everitt's (1988) approach to model-based clustering of mixed data, for which the main issue is estimating the parameters for the density $h(\cdot)$ in order to be able to calculate the probabilities which are used to identify the clusters. The values that maximize the log likelihood will be our parameter estimates. One approach is to solve this maximization problem using a deterministic, rather than a stochastic, optimization method. In this case, we do not know what the likelihood looks like, and there may be several local modes, which for a deterministic gradient method would be problematic (Robert and Casella, 2004). One drawback of a stochastic optimization method is that it is slow, in comparison to a deterministic optimization method (specifically if the likelihood is unimodal). Also, we are trying to estimate numerous parameters, up to 33 in one example given in Everitt (1988). As Robert and Casella (2004, pg. 22) explained, "when the statistician needs to study the details of a likelihood surface or posterior distribution, or needs to simultaneously estimate several features of these functions, or when the distributions are highly multimodal, it is preferable to use a simulation-based approach." In Everitt (1988), the Simplex method of Nelder and Mead (1965) was used to optimize the log-likelihood.

Sections 3.1 - 3.5 include our initial findings. In Section 3.1, we discuss our approach, the simulated annealing method, for optimizing $\log h(\cdot)$. Section 3.2 includes some initial results using this method with bivariate data, simulated according to the settings described by Everitt (1988). Using the simulated annealing approach alone led our parameter estimates to be drawn to extreme values for each parameter. We attempted to use a penalized log-likelihood within the simulated annealing to deter the parameters from being drawn to those extremes; the discussion of this approach and results for the bivariate data are included in Section 3.3. Section 3.4 details our results for a multivariate data set described by Everitt (1988). We investigated the effect that specifying different initial values had on our approach; the results are in Section 3.5. Section 3.6 describes a way to find an optimal tuning parameter for the penalized log likelihood. We include a real data analysis in Section 3.7 that uses all of our most recent findings.

3.1 Simulated Annealing Method

The parameter estimates we want are those that maximize the log likelihood, denoted here generically by $l(\theta)$, where θ is a (possibly vector-valued) parameter. As mentioned previously, a stochastic optimization method may be preferable to a deterministic optimization method in this case. The simulated annealing method, a Monte Carlo optimization method, "defines a sequence $\pi_t(\theta)$ of densities whose maximum arguments are confounded with the arguments of max l and with higher and higher concentrations around this argument" (Robert and Casella, 1999). The Boltzman-Gibbs transform of l,

$$\pi_t(\theta) \propto \exp\{l(\theta)/T_t\},$$

where T_t is a sequence of decreasing temperatures, is the most common choice for constructing π_t. "The general recommendation for the temperature decrease is that it should be logarithmic, rather than geometric" (Robert and Casella, 1999). As the temperature decreases toward 0, the simulated values become concentrated in a smaller neighborhood of the maximum of the function l. Suppose at iteration t the current parameter vector in the algorithm is θ_t and we generate a candidate parameter vector ξ. The update from θ_t to θ_{t+1} comes from the Metropolis-Hastings algorithm and the value θ_{t+1} is generated as

$$\theta_{t+1} = \begin{cases} \xi & \text{with probability} \;\; \rho = \min\{\exp(\frac{l(\xi)-l(\theta_t)}{T_t}), 1\}, \\ \theta_t & \text{with probability} \;\; 1-\rho. \end{cases}$$

If $l(\xi) > l(\theta_t)$, the new value is accepted and we set $\theta_{t+1} = \xi$. If, however, $l(\theta_t) > l(\xi)$, the move to the new value ξ may still be accepted, with probability ρ. Note that ρ will be near 1 if $l(\xi)$ is nearly as large as $l(\theta_t)$. Furthermore, a large temperature T_t will yield a larger acceptance probability ρ. By allowing these moves to be random, the objective function h may decrease temporarily during the algorithm which allows the method to escape local modes.

In applying this method to our problem, we first chose initial values for each of the parameters such that the log-likelihood was a finite value; these initially served as the current parameter vector and current log-likelihood. Then at each iteration, a new parameter vector was made by adding a random number from a symmetric uniform distribution $U(-\delta, \delta)$ to the current parameter values. The values of δ varied for the different parameters, and were as follows: for p_1, $\delta = 0.25$; for μ_1, μ_2, α_{112}, and α_{212}, $\delta = 5$; for Σ_{pq}, $\delta = 1$. A new log-likelihood was calculated using the new parameter vector. Based upon the criterion explained above, we either accepted or rejected the new parameter and log-likelihood values and updated the values as needed. At each iteration, we also had to make sure that the values generated were valid values, such as $0 \leq p_1 \leq 1$. In the following example from Everitt (1988), since $p = 1$ and $q = 1$, Σ_p, Σ_q and Σ_{pq} were actually scalar-valued and we had the following,

$$\Sigma_{q|p} = \Sigma_q - \Sigma_{pq}^T \Sigma_p^{-1} \Sigma_{pq} = 1 - \Sigma_{pq}^T \Sigma_p^{-1} \Sigma_{pq} > 0.$$

For this to hold, we did the following:

1. Choose initial values for Σ_{pq} and Σ_p such that $\rho^2 = \frac{\Sigma_{pq}^2}{\Sigma_p}$ starts at 0.5.

2. Generate values for Σ_{pq} as described above.

3. Generate values for ρ^2 by adding a random number from $U(-0.25, 0.25)$.

4. At each iteration, check that $0 \leq \rho^2 \leq 1$.

5. If $\rho^2 < 0$, let $\rho^2 = |\rho^2|$. If $\rho^2 > 1$, let $\rho^2 = 2 - \rho^2$.

6. Solve for Σ_p, where $\Sigma_p = \frac{\Sigma_{pq}^2}{\rho^2}$.

3.2 Bivariate Mixed Data Example from Everitt (1988)

Everitt (1988) simulated data with different parameter settings in order to determine how well the parameters were estimated. For the estimates in Table 3.20, data were simulated having the following parameter values: $n = 50$, $p = 1$, $q = 1$, $c = 2$, $k_1 = 2$, $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$, $\Sigma_{pq} = 0.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, and $\alpha_{212} = 1.0$. With these data, following Everitt we set different values of $\mu_2^{(p)}$, letting it equal 1.0, 2.0 and 3.0. These values are used to compare the results with Everitt's (1988) results, given in Table 3.2.

Table 3.1: Simulated Annealing method with 10,000 draws: data set 1

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.98	0.33	−10.66	−0.52	0.98	−27.23	−8.37
2.0	0.99	0.88	36.86	1.15	1.80	−10.76	2.19
3.0	0.995	1.59	33.51	1.60	2.82	−11.12	8.13

Table 3.2: Everitt's Nelder-Mead Simplex method results (1988): data set 1

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.83	0.25	2.16	−0.23	0.90	0.29	88.66
2.0	0.36	0.46	1.45	−1.20	1.82	−1.17	22.87
3.0	0.49	0.27	2.87	−0.22	1.49	−0.22	1.50

As can be seen from the Tables 3.1 and 3.2, the simulated annealing did not estimate the parameters quite as well as the Simplex method did. Everitt used the values that generated the data as the initial values in the Simplex method. In the simulated annealing method, we used initial values that were further from the values used to generate the data.

For the estimates in Table 3.3, data were simulated based on the parameter values: $n = 50$, $p = 1$, $q = 1$, $c = 2$, $k_1 = 2$, $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$, $\mu_2^{(p)} = 3.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, and $\alpha_{212} = 1.0$. With these data, we tried different values of Σ_{pq}, letting it equal 0.2, 0.5 and 0.8. These values are used to compare the results with those in Table 3.4 of Everitt (1988).

Table 3.3: Simulated Annealing method with 10,000 draws: data set 2

Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.2	0.99	1.75	−16.60	1.86	3.99	−6.33	33.34
0.5	0.40	3.41	0.36	−0.60	1.10	−4.60	−5.93
0.8	0.56	2.62	0.01	−1.07	1.41	−4.93	−5.32

Table 3.4: Everitt's Nelder-Mead Simplex method results (1988): data set 2

Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.2	0.44	0.14	2.78	0.47	1.47	−0.17	1.00
0.5	0.42	0.02	2.75	0.60	1.37	−0.05	0.95
0.8	0.52	0.24	2.89	1.25	1.56	0.13	1.18

In this case, the Simplex method achieved better estimates as well. As the results from the simulated annealing method do not provide parameter estimates that are comparable to those achieved using the Simplex method, the next step was to investigate how the likelihood was behaving with respect to each parameter. Figures 3.1 through 3.7 show the plots of the log-likelihood versus each parameter as the parameter values are updated, holding other parameters fixed.

Upon inspection of Figures 3.1 through 3.7, it appeared that some of the parameters, namely p_1, Σ_{pq}, α_{112}, α_{212}, would benefit from either using a Bayesian frame-

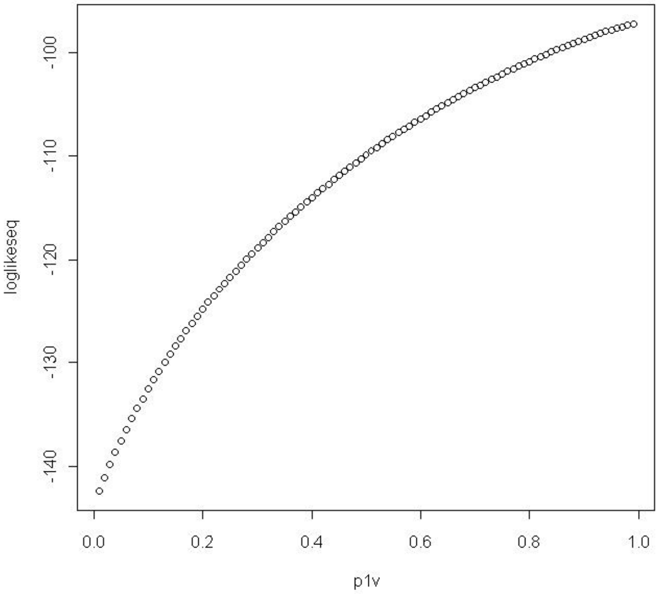

Figure 3.1: Log-likelihood Investigation: p_1

work with priors on the parameters listed or using a penalized likelihood approach. A problem with the Bayesian approach is that it is not obvious how to specify prior parameter values in a real data setting when the nature of the clusters is unknown a priori.

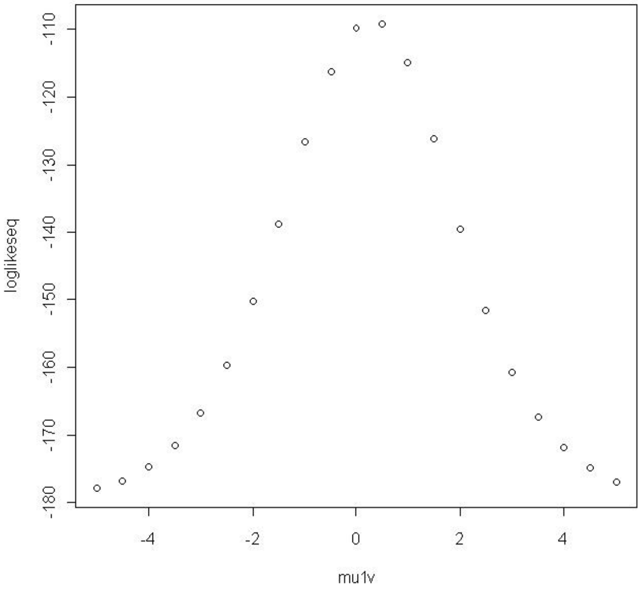

Figure 3.2: Log-likelihood Investigation: μ_1

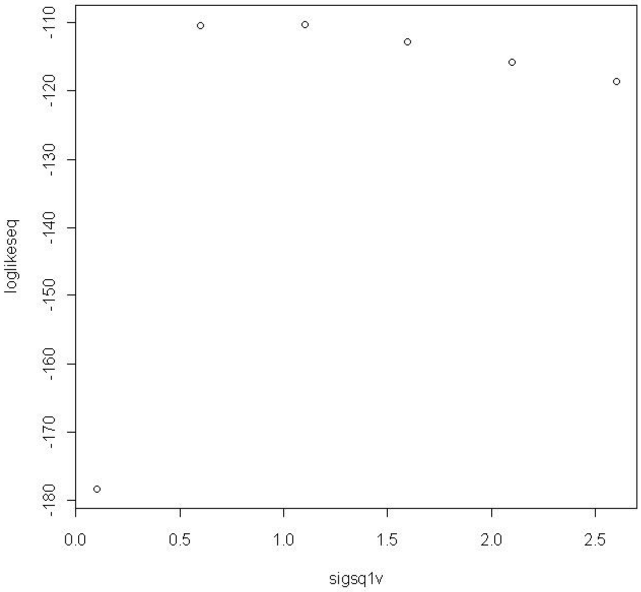

Figure 3.3: Log-likelihood Investigation: Σ_1

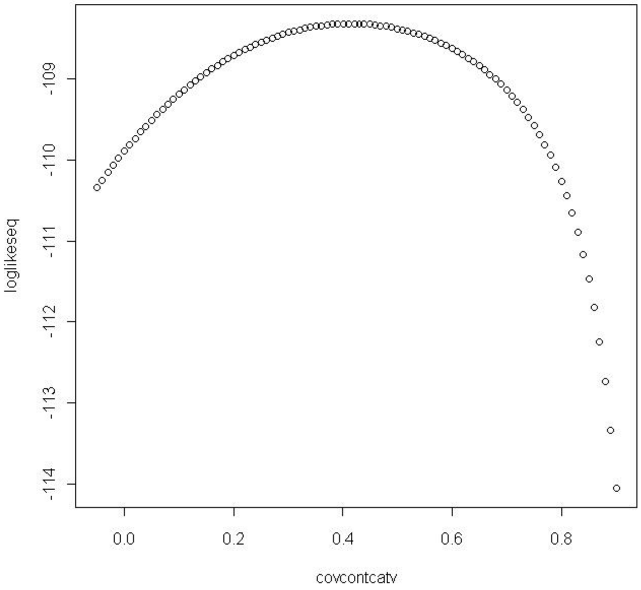

Figure 3.4: Log-likelihood Investigation: Σ_{pq}

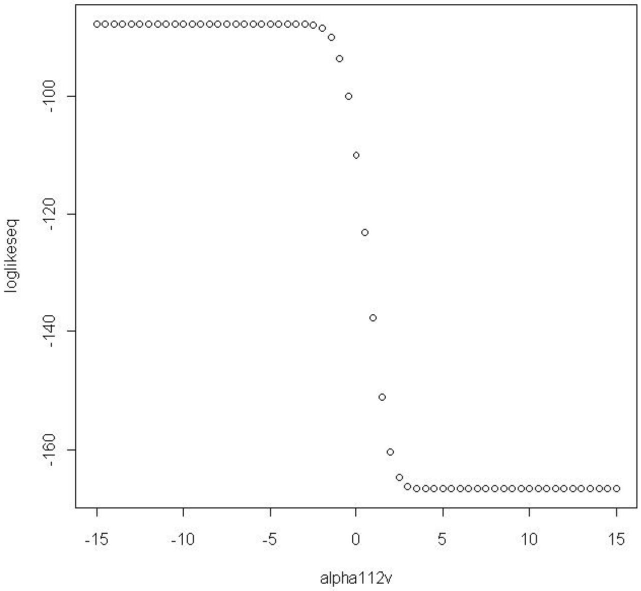

Figure 3.5: Log-likelihood Investigation: α_{112}

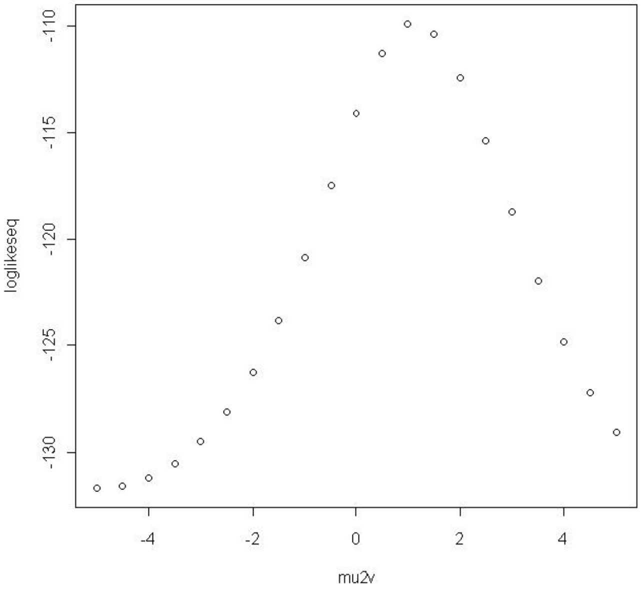

Figure 3.6: Log-likelihood Investigation: μ_2

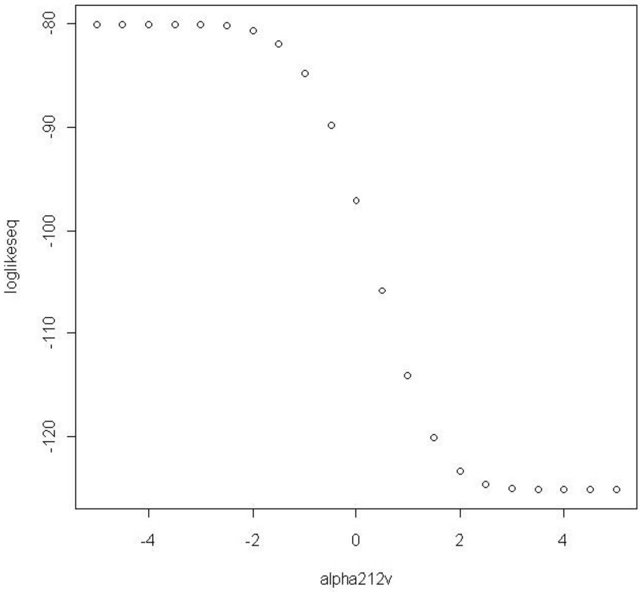

Figure 3.7: Log-likelihood Investigation: α_{212}

3.3 Penalized Likelihood Approach

Now, consider a penalized likelihood approach. The penalized likelihood approach is described as follows: "A penalty term is added to the log likelihood of the data, to circumvent difficulties related to dimensionality" (Green, 1996). In trying to maximize the log-likelihood, we encountered problems where the parameter estimates diverge toward extreme values; by using a penalized likelihood approach, we can penalize updates that yield extreme parameter estimates.

Our penalized likelihood approach was carried out in two stages. The first consisted of running the simulated annealing method with 5,000 draws, after a burn-in of 500, and calculating the standard deviations of the parameter estimates using all 5,000 values for each parameter. In the second stage, we calculate a scaled parameter vector, using the standard deviations that were found in the first stage, as follows: $\boldsymbol{\theta}^{(scaled)} = (\frac{|p_1 - 0.5|}{s_1}, \frac{\mu_1}{s_2}, \frac{\mu_2}{s_3}, \frac{\Sigma_p}{s_4}, \frac{\Sigma_{pq}}{s_5}, \frac{\alpha_{112}}{s_6}, \frac{\alpha_{212}}{s_7})$, where $s_1, \ldots, s_7$ represent the estimated standard deviations for each parameter estimate based on the initial stage. Then we continue with our stochastic search, using $\boldsymbol{\theta}^{curr}$ to obtain $\boldsymbol{\theta}^{new}$. We evaluate the penalized log likelihood for both $\boldsymbol{\theta}^{curr}$ and $\boldsymbol{\theta}^{new}$, where the penalized log likelihood is

$$\log L_{pen}(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta}) - \lambda \sum_{i=1}^{7} |\theta_i^{(scaled)}|.$$

At this step, we accept or reject $\boldsymbol{\theta}^{new}$ based on the probability in the simulated annealing method. The parameter λ is a tuning parameter that determines the degree of penalization.

To compare our results using the penalized log likelihood approach, the log likelihoods computed with the true parameter values and with Everitt's estimates were penalized (using the same standard deviations computed for the different parameters from stage 1 in our approach). For the data in Table 3.5 and Table 3.7, the burn-in was 500, the first stage had a total of 5000 draws, while the second stage had 50000 draws and λ was 20. The true parameter values were $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$,

$\Sigma_{pq} = 0.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and $\mu_2^{(p)}$ was varied to be 1.0, 2.0, and 3.0, respectively.

Table 3.5: Penalized log likelihood approach: data set 1 estimates

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.787	0.383	2.677	0.637	1.048	−30.683	−1.523
2.0	0.342	−71.516	0.978	1.146	1.838	63.161	−19.749
3.0	0.84	2.056	15.44	−1.829	3.400	−4.898	22.606

Table 3.6: Everitt's Nelder-Mead Simplex method results (1988): data set 1

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.662	0.908	0.632	−1.359	2.456	−9.502	−8.303
2.0	0.36	0.46	1.45	−1.20	1.82	−1.17	22.87
3.0	0.49	0.27	2.87	−0.22	1.49	−0.22	1.50

Table 3.7: Penalized log likelihood approach: data set 1 penalized log likelihoods

$\mu_2^{(p)}$	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
1.0	−130.733	−104.968	−210.363
2.0	−147.5821	−132.1312	−169.9999
3.0	−173.788	−141.357	−159.152

As can be seen from Table 3.7, our parameter estimates produced a better penalized log likelihood than Everitt's parameter estimates for two out of the three data sets; the case where $\mu_2^{(p)} = 3.0$ is where Everitt's penalized log likelihood is better than ours.

Figure 3.8 through Figure 3.14 are plots of the parameter values generated by the simulated annealing method, for the case when $\mu_2 = 1.0$. These show how much of the parameter space is being explored and whether or not we should adjust some of the values in the simulated annealing algorithm.

As can be seen from Figures 3.8 through 3.14, the parameter spaces for most of the parameters, aside from p_1, look as if they should be explored more completely,

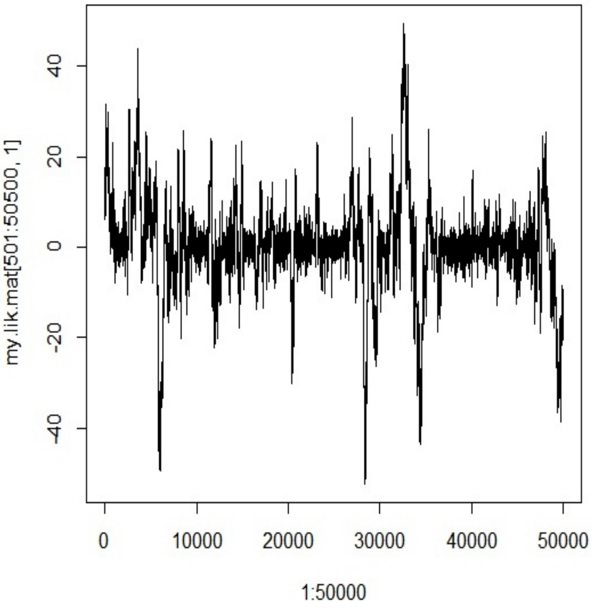

Figure 3.8: Parameter Space of μ_1

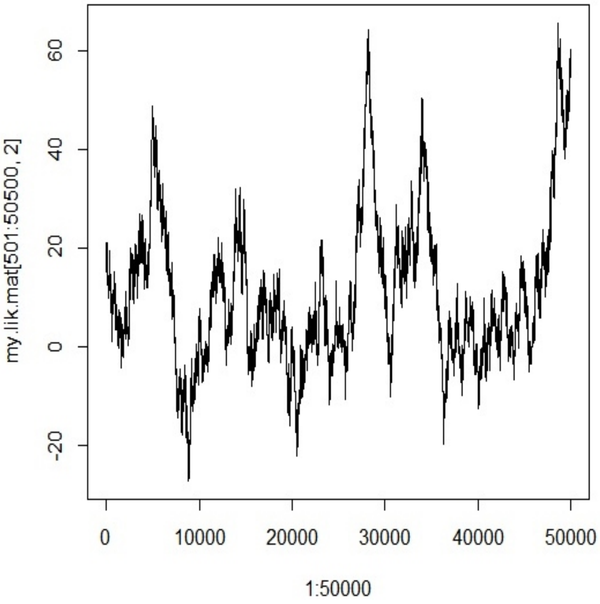

Figure 3.9: Parameter Space of Σ_{pq}

especially because at each iteration, the values of the parameter estimates change with each other. To achieve the best parameter estimates, we increased the number of iterations to let the parameter space be explored more fully, and adjusted both the temperature T and the tuning parameter λ.

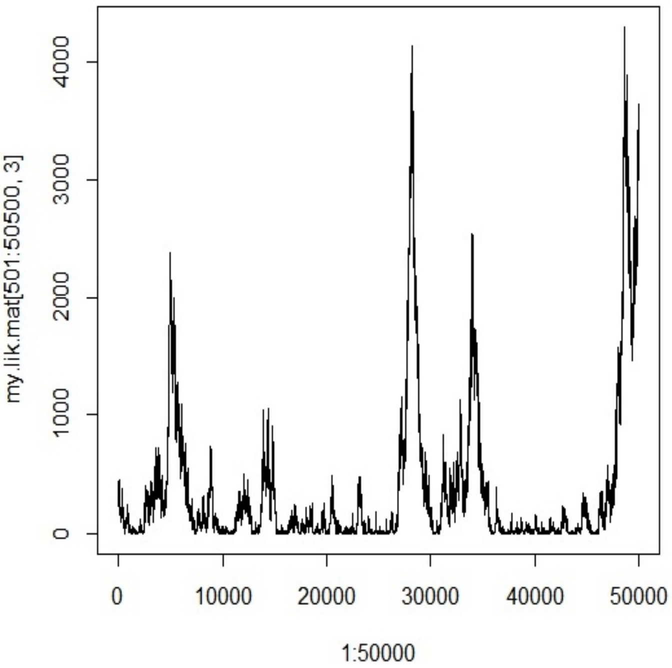

Figure 3.10: Parameter Space of Σ_p

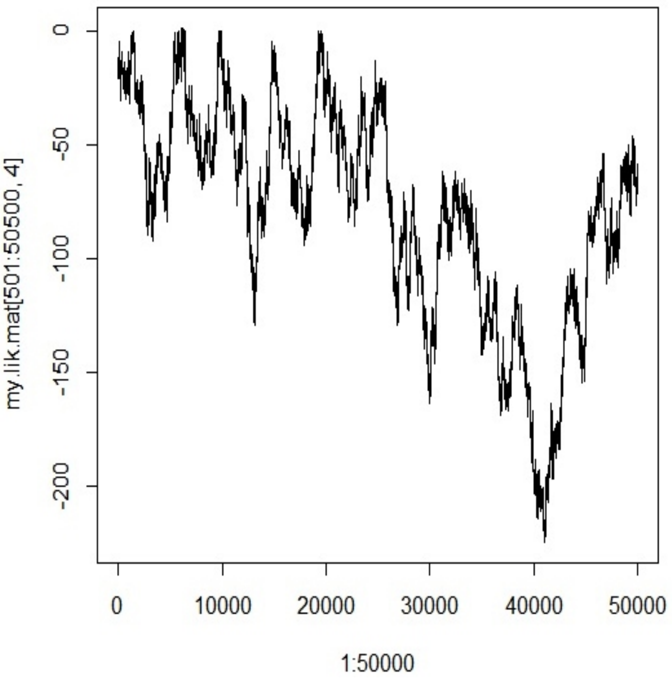

Figure 3.11: Parameter Space of α_{112}

Table 3.8 and Table 3.10 were attained by generating data with a burn-in of 500, $250,000$ draws for the first stage, $500,000$ draws for the second stage and λ was 65.

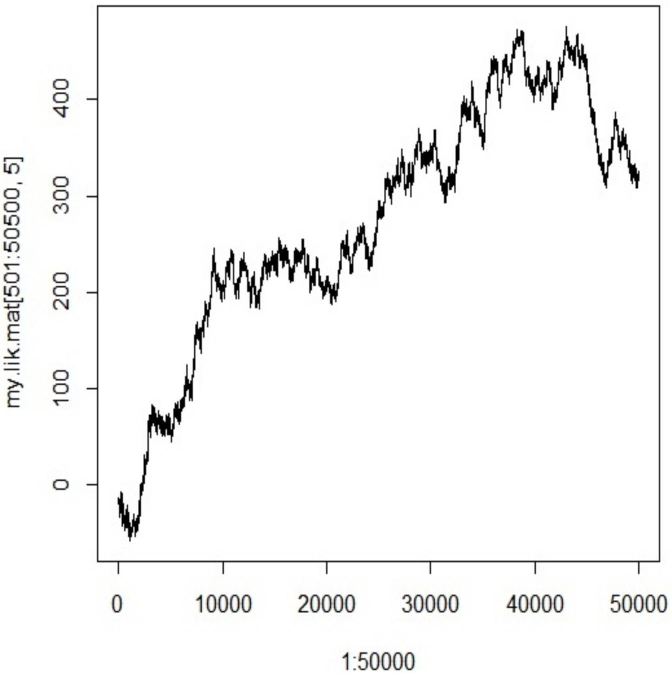

Figure 3.12: Parameter Space of μ_2

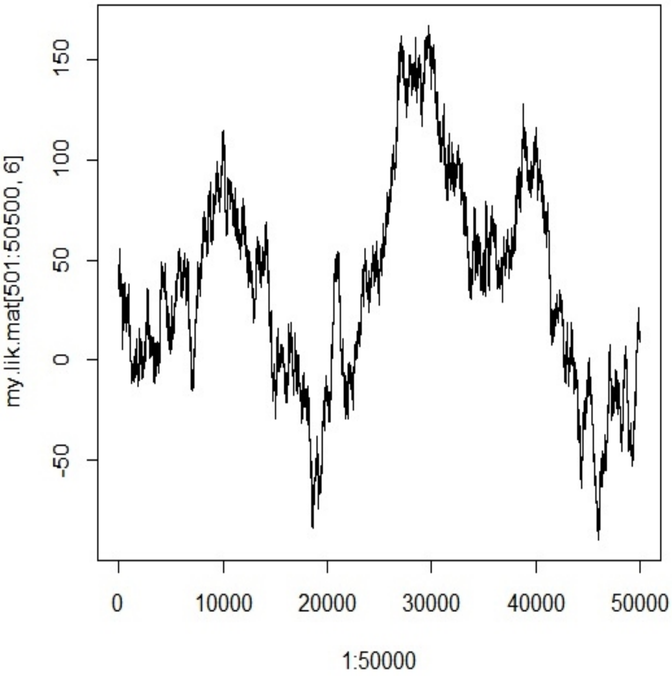

Figure 3.13: Parameter Space of α_{212}

The true parameter values were $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$, $\mu_2^{(p)} = 3.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and Σ_{pq} was varied to be 0.2, 0.5, and 0.8, respectively.

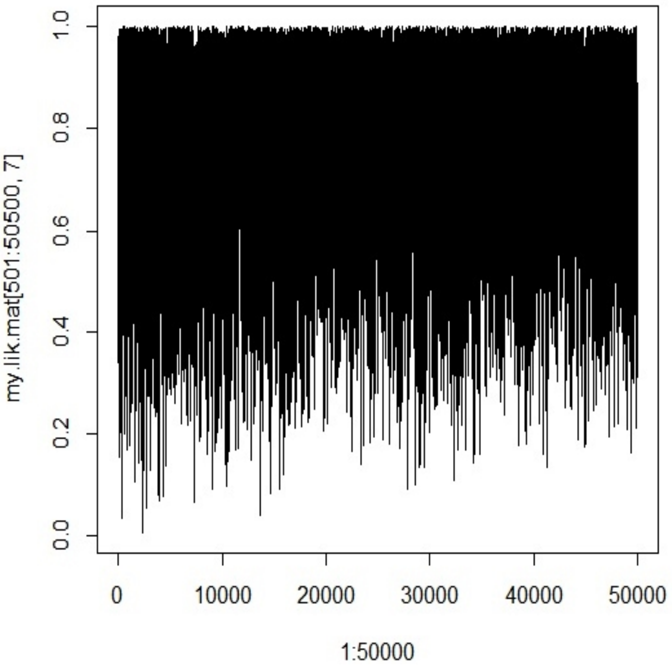

Figure 3.14: Parameter Space of p_1

Table 3.8: Penalized log likelihood approach: data set 2 estimates

Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.2	0.493	-0.340	-5.307	5.010	25.186	-38.353	-3.911
0.5	0.210	153.010	7.226	-32.357	1909.07	0.763	-75.806
0.8	0.87	-32.31	7.034	-19.31	2264.243	-43.06	-106.016

Table 3.9: Everitt's Nelder-Mead Simplex method results (1988): data set 2

Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.2	0.44	0.14	2.78	0.47	1.47	-0.17	1.00
0.5	0.42	0.02	2.75	0.60	1.37	-0.05	0.95
0.8	0.52	0.24	2.89	1.25	1.56	0.13	1.18

As can be seen from Table 3.10, Everitt's penalized log likelihood is better than ours in all but one case. When $\Sigma_{pq} = 0.8$, Everitt's penalized log likelihood is calculated to be not a real number. This may have to do with the randomness of the generated data or this may be due to rounding, as Everitt gave his values to two decimal places.

Table 3.10: Penalized log likelihood approach: data set 2 log likelihoods

Σ_{pq}	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.2	-193.613	-150.967	-163.343
0.5	-471.5350	-141.5402	-160.8764
0.8	-303.5461	-148.9736	NaN

Table 3.11 and Table 3.13 were attained by generating data with a burn-in of 500, $250,000$ draws for the first stage, $500,000$ draws for the second stage and λ was 65. The true parameter values were $\mu_1^{(p)} = 0.0$, $\mu_2^{(p)} = 3.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and p_1 was varied to be 0.6, 0.7, and 0.8, respectively.

Table 3.11: Penalized log likelihood approach: data set 3 estimates

p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.6	0.512	1.885	-28.766	2.600	8.165	-86.944	3.604
0.7	0.554	0.285	-2.765	1.701	4.150	-6.351	25.817
0.8	0.540	0.231	-14.182	11.386	132.457	-29.207	21.486

Table 3.12: Everitt's Nelder-Mead Simplex method results (1988): data set 3

p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.6	0.6	0.07	2.86	0.58	1.29	0.06	0.89
0.7	0.79	0.17	3.33	0.44	1.24	0.22	0.35
0.8	0.83	0.11	3.67	0.36	1.17	0.21	0.12

Table 3.13: Penalized log likelihood approach: data set 3 penalized log likelihoods

p_1	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.6	-268.190	-160.364	-158.631
0.7	-205.649	-201.756	-213.611
0.8	-304.428	-213.000	-213.649

As can be seen from Table 3.13, Everitt's penalized log likelihood is better than ours in all but one case. When $p_1 = 0.7$, our penalized log likelihood is better than

Everitt's.

For all of the previous comparisons, we compared our penalized likelihood, calculated with our simulated data, to a penalized likelihood for Everitt that was calculated by using the likelihood he found from his simulated data. To better compare our method with Everitt's, our next step was to use the method he used on our simulated data to find the parameter estimates. Everitt used the Nelder-Mead method to estimate the parameters. One can use the `optim` function in R to implement this method; however, using the `optim` function with the Nelder-Mead option does not allow one to place bounds on the parameter values. Instead, we used the option "L-BFGS-B" (Byrd et al., 1995) which corresponds to the method that allows appropriate bounds for the different parameters. Table 3.14 includes the results for data set 1 using `optim` with the "L-BFGS-B" option, when starting at the true values. The table also lists the results using our method and the results when calculating the penalized log likelihood with the true parameter values. To obtain these results, we ran our simulated annealing loop for 500,000 iterations, with step sizes of 1 for μ_1, 0.25 for Σ_{pq}, 0.15 for α_{112}, α_{212} and μ_2, and 0.1 for p_1.

Table 3.14: Comparisons using Everitt's method: data set 1 penalized log likelihoods

$\mu_2^{(p)}$	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
1.0	-102.488	-151.481	-95.092
2.0	-170.934	-159.291	-109.306
3.0	-182.115	-182.115	-133.360

Since we had results in which our penalized log likelihood was lower than penalized log likelihood that was calculated using the true parameter values, we wanted to employ a measure that would give us a sense of how close our parameter estimates were to the true parameter values. We first examined the following goodness of fit

Table 3.15: Comparisons using Everitt's method: data set 2 penalized log likelihoods

Σ_{pq}	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.2	-155.694	-171.008	-137.127
0.5	-206.713	-206.713	-146.998
0.8	-221.452	-221.452	-132.984

Table 3.16: Comparisons using Everitt's method: data set 3 penalized log likelihoods

p_1	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.6	-149.944	-226.074	-145.602
0.7	-194.597	-252.348	-115.633
0.8	-201.435	-229.926	-110.612

measure for our estimated parameter vector,

$$G_1 = \frac{1}{d}\sum_{j=1}^{d} \frac{|\hat{\theta}_j - \theta_j|}{|\hat{\theta}_j^{Ev} - \theta_j|},$$

where d is the number of parameters, θ_j is the true value for parameter j, $\hat{\theta}_j$ is our estimate for parameter j, $\hat{\theta}_j^{Ev}$ is the value for parameter j estimated by Everitt's approach. With this measure, if $G_1 < 1$ our parameter estimates were closer to the true parameter values, and if $G_1 > 1$, Everitt's method produced better estimates. This goodness of fit measure had the problem in which some $\hat{\theta}_j^{Ev}$ values were in certain cases equal to their respective θ_j values, and hence G_1 was undefined.

We then decided on the following goodness of fit measures

$$G_2^{SA} = \frac{1}{d}\sum_{j=1}^{d} \frac{|\hat{\theta}_j - \theta_j|}{s(\hat{\theta}_j)},$$

and

$$G_2^{Ev} = \frac{1}{d}\sum_{j=1}^{d} \frac{|\hat{\theta}_j^{Ev} - \theta_j|}{s(\hat{\theta}_j)},$$

where again d is the number of parameters, θ_j is the true value for parameter j, $\hat{\theta}_j$ is our estimate for parameter j, $\hat{\theta}_j^{Ev}$ is the value for parameter j estimated by Everitt's approach, and $s(\hat{\theta}_j)$ is the estimated standard deviation for the jth parameter, as estimated in stage 1. With these goodness of fit measures, the closer G_2^{SA} or G_2^{Ev} is to 0, the closer their respective parameter estimates are to the true parameter values, hence, we have the following: if $G_2^{SA} < G_2^{Ev}$, then our parameter estimates were closer to the true parameter values; if $G_2^{SA} > G_2^{Ev}$, then Everitt's method produced better parameter estimates.

Table 3.17: Goodness of fit measure: data set 1

$\mu_2^{(p)}$	G_2^{SA}	G_2^{Ev}
1.0	0.1112	0.09338
2.0	0.01916	0.08610
3.0	0	0.07693

Table 3.18: Goodness of fit measure: data set 2

Σ_{pq}	G_2^{SA}	G_2^{Ev}
0.2	0.4908	0.2702
0.5	0	0.08076
0.8	0	0.07563

Table 3.19: Goodness of fit measure: data set 3

p_1	G_2^{SA}	G_2^{Ev}
0.6	0.1560	0.1278
0.7	0.2413	0.2423
0.8	0.1793	0.2365

One can see from Table 3.17 that our parameter estimates were closer to the true parameter values when $\mu_2 = 2.0$ and when $\mu_2 = 3.0$. Since we chose the initial values to be the true parameter values, in the case when $\mu_2 = 3.0$, our method found that the best estimates were the true values. When $\mu_2 = 1.0$, Everitt's method produced estimates that were only slightly closer to the true parameter values than ours were.

From Table 3.18 we see that our estimates were the true parameter values, as $G_2^{SA} = 0$, when $\Sigma_{pq} = 0.5$ and $\Sigma_{pq} = 0.8$. When $\Sigma_{pq} = 0.2$, Everitt's method provided results that were considerably closer to the true parameter values than our estimates were. As can be seen from Table 3.19, our estimates were slightly closer to the true parameter values than Everitt's estimates when $p_1 = 0.8$, were minutely closer when $p_1 = 0.7$, and were not as close as Everitt's estimates when $p_1 = 0.6$.

3.4 Multivariate Mixed Data Example from Everitt (1988)

The first data set that was investigated by Everitt (1988) required estimating 7 parameters, a "simple" example. He then presented a more complex example, a data set that required estimating 33 parameters, his Data Set 4. In this data set, there are 3 continuous variables and 2 categorical variables, with 100 observations each. It is assumed that there are 3 clusters present in the data. In trying to achieve comparable results to Everitt's Data Set 4, data were simulated to have the following parameter values: $n = 100$, $p = 3$, $q = 2$, $c = 3$, $k_1 = 2$, $k_2 = 3$, $p_1 = 0.4$, $p_2 = 0.3$, $p_3 = 0.3$,

$$\mu_1^{(p)} = (0, 0, 0)^T, \mu_2^{(p)} = (3.0, 3.0, 3.0)^T, \mu_3^{(p)} = (1.5, -1.5, 4.5)^T,$$

$$\Sigma_{pq} = \begin{bmatrix} 0.2 & 0.0 \\ 0.5 & 0.5 \\ 0.0 & 0.4 \end{bmatrix},$$

$$\Sigma_p = I_3,$$

$$\Sigma_q = \begin{bmatrix} 1.0 & 0.5 \\ 0.5 & 1.0 \end{bmatrix},$$

$\alpha_{112} = 0.0$, $\alpha_{122} = -1.0$, $\alpha_{123} = 1.0$, $\alpha_{212} = 1.0$, $\alpha_{222} = -0.5$, $\alpha_{223} = 1.5$, $\alpha_{312} = -1.0$, $\alpha_{322} = 0.0$, $\alpha_{323} = 0.5$.

As defined previously, c is the number of clusters assumed for the data, p is the number of observed continuous variables, q is the number of unobserved latent continuous variables (used to generate the categorical variables) and the α_{ijl} values are the threshold values used to define the categorical, or ordinal, variables based on the latent variables $x_{p+1}, \ldots, x_{p+q}$. Hence, for the data set described above, we would have the following: for cluster 1 observations,

$$z_1 = \begin{cases} 1 & \text{if } \; x_4 < 0.0, \\ 2 & \text{if } \; x_4 \geq 0.0 \end{cases},$$

$$z_2 = \begin{cases} 1 & \text{if } \; x_5 < -1.0, \\ 2 & \text{if } \; -1.0 \leq x_5 < 1.0, \\ 3 & \text{if } \; x_5 \geq 1.0 \end{cases}$$

for cluster 2 observations,

$$z_1 = \begin{cases} 1 & \text{if } \; x_4 < 1.0, \\ 2 & \text{if } \; x_4 \geq 1.0 \end{cases},$$

$$z_2 = \begin{cases} 1 & \text{if } \; x_5 < -0.5, \\ 2 & \text{if } \; -0.5 \leq x_5 < 1.5, \\ 3 & \text{if } \; x_5 \geq 1.5 \end{cases}$$

and for cluster 3 observations,

$$z_1 = \begin{cases} 1 & \text{if } \; x_4 < -1.0, \\ 2 & \text{if } \; x_4 \geq -1.0 \end{cases}$$

$$z_2 = \begin{cases} 1 & \text{if } \; x_5 < 0.0, \\ 2 & \text{if } \; 0.0 \leq x_5 < 0.5. \\ 3 & \text{if } \; x_5 \geq 0.5 \end{cases}$$

In this case, instead of having to estimate seven parameters, as in the bivariate case, we need to estimate 33 parameters: p_1 and p_2; the 3 components of $\mu_1^{(p)}$; the 3 components of $\mu_2^{(p)}$; the 3 components of $\mu_3^{(p)}$; all 6 entries in Σ_{pq}; the $(1,1)$, $(1,2)$, $(1,3)$, $(2,2)$, $(2,3)$, and $(3,3)$ entries in Σ_p; the $(1,2)$ entry of Σ_q; and 9 α_{ijl} values.

Generating certain parameters in the simulated annealing algorithm was more complicated in the multivariate case. We defined a matrix

$$\Sigma = \begin{bmatrix} \Sigma_p & \Sigma_{pq} \\ \Sigma_{pq}^T & \Sigma_q \end{bmatrix}.$$

In order to generate initial values for the stochastic search, we needed to generate a correlation matrix P. Also, we generated a diagonal matrix D containing population variances for the components with the first three diagonal entries being any value, and the last two entries being 1. Then $\Sigma = D^{1/2}\mathrm{P}D^{1/2}$. At this step, we had to ensure Σ was positive definite, by checking all of the eigenvalues of Σ, making sure they were all greater than 0.000001, using the R function `eigen`. Then in the simulated annealing search, we changed the values of Σ componentwise. At each loop through the simulated annealing process, we had to check that Σ was positive definite in order for it to be valid. We also had to make sure that $\Sigma_{q|p}$ was a positive definite matrix. In case either of these matrices were not positive definite, we generated new values for the matrices until they were both positive definite.

To generate values for Σ, we started with the function `genPositiveDefMat`. We had to alter the function so that the range for the variances could be different for each variable, specifically that the 4th and 5th variances were restricted to be 1. We used the method `unifcorrmat` to generate the positive definite covariance matrix, which first generates a correlation matrix and then generates variance components within the ranges specified by the user. These are then combined to create the covariance matrix.

The generation of (p_1, p_2, p_3) was different. A simple uniform perturbation of the current p_1 and p_2 values in the chain would not ensure the condition that $p_1+p_2+p_3 = 1$. Instead, at each iteration, we randomly generated values for p_1 and p_2 using the `rdirichlet` function in R. This function ensures that $p_1 + p_2 + p_3 = 1$. This results in a more complete exploration of the parameter space (p_1, p_2, p_3), but also a loss in efficiency, requiring the algorithm to be run for more iterations. Also, with the parameter space being 33-dimensional, we looked at some initial plots to check if the parameter space was being thoroughly explored. Based on the plots, we decided to adjust the temperature so that the rate at which it decreased was slower; we changed the temperature function from $T(i) = \frac{500}{\log(i)}$ to $T(i) = \frac{500}{\log(\log(i))}$.

The following results are Everitt's parameter estimates for the data that was simulated as described above: $\hat{p}_1 = 0.36$, $\hat{p}_2 = 0.25$,

$$\hat{\mu}_1^{(p)} = (0.14, 0.20, -0.06)^T, \hat{\mu}_2^{(p)} = (2.81, 2.80, -3.12)^T, \hat{\mu}_3^{(p)} = (1.56, -1.49, 4.66)^T,$$

$$\hat{\Sigma}_{pq} = \begin{bmatrix} 0.38 & -0.11 \\ 0.39 & 0.65 \\ 0.00 & 0.19 \end{bmatrix},$$

$$\hat{\Sigma}_p = \begin{bmatrix} 1.13 & 0.13 & 0.04 \\ 0.13 & 0.95 & 0.06 \\ 0.04 & 0.06 & 0.78 \end{bmatrix},$$

$$\hat{\Sigma}_q = \begin{bmatrix} 1.0 & 0.04 \\ 0.04 & 1.0 \end{bmatrix},$$

$\hat{\alpha}_{112} = -0.29$, $\hat{\alpha}_{122} = -1.71$, $\hat{\alpha}_{123} = 1.0$, $\hat{\alpha}_{212} = 0.69$, $\hat{\alpha}_{222} = -0.36$, $\hat{\alpha}_{223} = 1.16$, $\hat{\alpha}_{312} = -0.78$, $\hat{\alpha}_{322} = 0.01$, and $\hat{\alpha}_{323} = 0.51$.

Table 3.20 contains our initial results, with $\lambda = 17$, and with $5,000$ iterations and $T(i) = \frac{500}{\log i}$ in the first stage, and $100,000$ iterations and $T(i) = \frac{500}{\log(\log i)}$ in the second stage. Table 3.20 also contains the true values and Everitt's parameter estimates. As can be seen from Table 3.20, our estimates for p_1, p_2, were the only parameters that our method estimated better than Everitt's; however, our penalized log likelihood was lower than Everitt's, but it was also lower than the penalized log likelihood calculated using the true parameter values, as can be seen in Table 3.21.

Table 3.20: Penalized likelihood approach: data set 4 estimates

	Truth	Everitt's Estimate	Our Estimate
p_1	0.4	0.36	0.418
p_2	0.3	0.25	0.252
$\mu_1^{(p)T}$	$(0, 0, 0)$	$(0.14, 0.20, -0.06)$	$(-19.25, -36.03, 18.0)$
$\mu_2^{(p)T}$	$(3.0, 3.0, 3.0)$	$(2.81, 2.80, -3.12)$	$(54.51, 56.20, 18.49)$
$\mu_3^{(p)T}$	$(1.5, -1.5, 4.5)$	$(1.56, -1.49, 4.66)$	$(1.23, 0.14, 0.69)$
Σ_{pq}	$\begin{bmatrix} 0.2 & 0.0 \\ 0.5 & 0.5 \\ 0.0 & 0.4 \end{bmatrix}$	$\begin{bmatrix} 0.38 & -0.11 \\ 0.39 & 0.65 \\ 0.00 & 0.19 \end{bmatrix}$	$\begin{bmatrix} 0.674 & 0.022 \\ 0.585 & -0.083 \\ 0.398 & -0.094 \end{bmatrix}$
Σ_p	I_3	$\begin{bmatrix} 1.13 & 0.13 & 0.04 \\ 0.13 & 0.95 & 0.06 \\ 0.04 & 0.06 & 0.78 \end{bmatrix}$	$\begin{bmatrix} 2.675 & 2.693 & 0.197 \\ 2.693 & 5.142 & 0.625 \\ 0.197 & 0.625 & 5.308 \end{bmatrix}$
Σ_q	$\begin{bmatrix} 1.0 & 0.5 \\ 0.5 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.04 \\ 0.04 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1 & -0.153 \\ -0.153 & 1 \end{bmatrix}$
α_{112}	0.0	-0.29	19.130
α_{122}	-1.0	-1.71	-27.024
α_{123}	1.0	1.0	-24.342
α_{212}	1.0	0.69	-14.800
α_{222}	-0.5	-0.36	-4.257
α_{223}	1.5	1.16	19.256
α_{312}	-1.0	-0.78	0.629
α_{322}	0.0	0.01	0.748
α_{323}	0.5	0.51	1.278

Table 3.21: Penalized log likelihood approach: data set 4 penalized log likelihoods

True Penalized $\log L$	Everitt's Penalized $\log L$	Our Penalized $\log L$
-1012.158	-1490.504	-1696.759

3.5 Investigating Different Initial Values for Algorithm

As Everitt (1988) started his simplex method at the true values, in this section we attempt to start our algorithm at both (1) the true values of the parameters and (2) at Everitt's estimates. For Everitt's data set 1, the true parameter values were: $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$, $\Sigma_{pq} = 0.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and $\mu_2^{(p)}$ was 1.0, 2.0 and 3.0, respectively (Everitt, 1988). Everitt's parameter estimates are given in Table 3.22.

Table 3.22: Everitt's Nelder-Mead Simplex method results (1988): data set 1

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.83	0.25	2.16	−0.23	0.90	0.29	88.66
2.0	0.36	0.46	1.45	−1.20	1.82	−1.17	22.87
3.0	0.49	0.27	2.87	−0.22	1.49	−0.22	1.50

Table 3.23 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 1, when starting our algorithm at the true parameter values.

Table 3.23: Penalized log likelihood approach (starting algorithm at the true values): data set 1 estimates

$\mu_2^{(p)}$	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
1.0	−101.383	−101.383	−335.169
2.0	−140.689	−140.689	−218.164
3.0	−139.344	−142.576	−161.235

As can be seen in Table 3.23, when starting our algorithm at the true values, our best penalized $\log L$ is achieved with those true values for the cases when $\mu_2^{(p)} = 1.0$ and $\mu_2^{(p)} = 2.0$. When $\mu_2^{(p)} = 3.0$, however, our best penalized $\log L$ is lower than the true penalized $\log L$. Table 3.24 contains our parameter estimates and the true values for the case when $\mu_2^{(p)} = 3.0$. As our estimates lie fairly close to the true values, our result may be due to the randomness of the simulated data.

Table 3.24: Penalized log likelihood approach: comparing our estimates and the true values for data set 1

	Our Estimates	True Values
$\hat{p}_1$	0.501	0.5
$\hat{\mu}_1^{(p)}$	-0.134	0.0
$\hat{\mu}_2^{(p)}$	2.328	3.0
$\hat{\Sigma}_{pq}$	0.214	0.0
$\hat{\Sigma}_p$	1.031	1.0
$\hat{\alpha}_{112}$	-0.609	0.0
$\hat{\alpha}_{212}$	0.945	1.0

Table 3.25 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 1, when starting our algorithm at Everitt's parameter estimates.

Table 3.25: Penalized log likelihood approach (starting at Everitt's estimates (1988)): data set 1 penalized log likelihoods

$\mu_2^{(p)}$	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
1.0	-289.193	-120.921	-377.074
2.0	-161.844	-126.167	-230.494
3.0	-117.383	-131.772	-146.181

For this set of simulated data, when starting our algorithm at Everitt's estimates, our penalized $\log L$ was better than Everitt's penalized $\log L$ in all three cases, and for the case when $\mu_2^{(p)} = 3.0$, our penalized $\log L$ was better than the true penalized $\log L$. Our best parameter estimates for this set of simulated data, when starting our algorithm at Everitt's estimates, are given in Table 3.26.

Table 3.26: Penalized log likelihood approach (starting at Everitt's estimates (1988)): data set 1 estimates

$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
1.0	0.516	0.768	1.901	-0.623	0.707	-1.740	89.307
2.0	0.499	13.787	0.559	-1.233	1.833	12.831	-2.928
3.0	0.465	1.169	3.097	1.413	2.092	-2.171	-1.994

For Everitt's data set 2, the true parameter values were: $p_1 = p_2 = 0.5$, $\mu_1^{(p)} = 0.0$, $\mu_2^{(p)} = 3.0$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and Σ_{pq} was 0.2, 0.5 and 0.8, respectively. Everitt's parameter estimates are given in Table 3.27.

Table 3.27: Everitt's Nelder-Mead Simplex method results (1988): data set 2

Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.2	0.44	0.14	2.78	0.47	1.47	−0.17	1.00
0.5	0.42	0.02	2.75	0.60	1.37	−0.05	0.95
0.8	0.52	0.24	2.89	1.25	1.56	0.13	1.18

Table 3.28 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 2, when starting our algorithm at the true parameter values.

Table 3.28: Penalized log likelihood approach (starting at the true values): data set 2 penalized log likelihoods

Σ_{pq}	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.2	−160.177	−160.177	−168.627
0.5	−128.342	−147.654	−163.218
0.8	−190.824	−239.413	NaN

Table 3.29 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 2, when starting our algorithm at Everitt's parameter estimates.

Table 3.29: Penalized log likelihood approach (starting at Everitt's estimates (1988)): data set 2 penalized log likelihoods

Σ_{pq}	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.2	−135.750	−149.922	−161.177
0.5	−159.471	−143.440	−159.471
0.8	−160.052	−180.659	NaN

As can be seen from Tables 3.28 and 3.29, the penalized log likelihood when using Everitt's parameter estimates, for the case when $\Sigma_{pq} = 0.8$, is not a real number.

This may have to do with the randomness of the generated data or this may be due to rounding, as Everitt gave his values to two decimal places.

For Everitt's data set 3, the true parameter values were: $\mu_1^{(p)} = 0.0$, $\mu_2^{(p)} = 3.0$, $\Sigma_{pq} = 0.5$, $\Sigma_p = 1.0$, $\alpha_{112} = 0.0$, $\alpha_{212} = 1.0$, and p_1 was 0.6, 0.7 and 0.8. Everitt's parameter estimates are given in Table 3.30.

Table 3.30: Everitt's Nelder-Mead Simplex method results (1988): data set 3

p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.6	0.6	0.07	2.86	0.58	1.29	0.06	0.89
0.7	0.79	0.17	3.33	0.44	1.24	0.22	0.35
0.8	0.83	0.11	3.67	0.36	1.17	0.21	0.12

Table 3.31 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 3, when starting our algorithm at the true parameter values.

Table 3.31: Penalized log likelihood approach (starting at the true values): data set 3 penalized log likelihoods

p_1	Our	True	Everitt's
	Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
0.6	−136.845	−171.938	−166.368
0.7	−118.035	−174.916	−195.018
0.8	−156.622	−176.898	−184.654

Table 3.32: Penalized log likelihood approach (starting at the true values): data set 3 estimates

p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.6	0.556	0.074	3.843	0.291	0.877	−0.580	−0.500
0.7	0.563	−0.873	2.680	−2.036	4.319	−1.253	−0.601
0.8	0.685	0.599	4.088	0.631	1.292	−0.499	1.510

Table 3.33 compares the penalized log likelihoods for our estimates, the true parameter values, and for Everitt's estimates for Everitt's data set 3, when starting our algorithm at Everitt's parameter estimates. As one can see, our penalized $\log L$ is

not only better than Everitt's penalized $\log L$, in each case, but it is also better than the true penalized $\log L$, in each case. Again, this is probably due to the fact that the simulated data are randomly generated and reflect certain random deviation from the true underlying parameter values.

Table 3.33: Penalized log likelihood approach (starting at Everitt's estimates (1988)): data set 3 penalized log likelihoods

p_1	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
0.6	−151.356	−167.217	−164.331
0.7	−135.076	−175.894	−193.319
0.8	−142.255	−199.451	−203.341

Table 3.34 contains our parameter estimates when starting at Everitt's estimates.

Table 3.34: Penalized log likelihood approach (starting at Everitt's estimates (1988)): data set 3 estimates

p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
0.6	0.482	−3.904	1.826	4.039	16.654	−12.891	−2.179
0.7	0.558	−1.950	2.968	1.273	4.428	−0.770	−1.669
0.8	0.554	2.996	2.088	−4.242	19.666	−0.563	−1.133

For Everitt's data set 4, the true parameter values are given in Table 3.35, along with Everitt's parameter estimates and our parameter estimates when starting at the true parameter values.

Table 3.35: Multivariate penalized log likelihood approach: penalized log likelihoods (starting at the true parameter values)

Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
−1238.481	−1238.481	−1890.201

Table 3.36 contains our parameter estimates, along with the true parameter values and Everitt's parameter estimates, when starting at Everitt's estimates and Table 3.37 contains the penalized log likelihoods calculated using the true parameter values, Everitt's parameter estimates and our parameter estimates.

Table 3.36: Multivariate penalized log likelihood approach: comparing our estimates and Everitt's estimates (1988) with the true values

	Truth	Everitt's Estimate	Our Estimate
p_1	0.4	0.36	0.511
p_2	0.3	0.25	0.149
$\mu_1^{(p)T}$	$(0, 0, 0)$	$(0.14, 0.20, -0.06)$	$(1.31, 0.42, 0.06)$
$\mu_2^{(p)T}$	$(3.0, 3.0, 3.0)$	$(2.81, 2.80, -3.12)$	$(2.48, 3.18, -4.5)$
$\mu_3^{(p)T}$	$(1.5, -1.5, 4.5)$	$(1.56, -1.49, 4.66)$	$(1.60, -2.0, 4.49)$
Σ_{pq}	$\begin{bmatrix} 0.2 & 0.0 \\ 0.5 & 0.5 \\ 0.0 & 0.4 \end{bmatrix}$	$\begin{bmatrix} 0.38 & -0.11 \\ 0.39 & 0.65 \\ 0.00 & 0.19 \end{bmatrix}$	$\begin{bmatrix} -0.21 & 0.6 \\ 0.17 & 0.29 \\ -0.01 & -0.15 \end{bmatrix}$
Σ_p	I_3	$\begin{bmatrix} 1.13 & 0.13 & 0.04 \\ 0.13 & 0.95 & 0.06 \\ 0.04 & 0.06 & 0.78 \end{bmatrix}$	$\begin{bmatrix} 1.60 & -0.07 & 0.25 \\ -0.07 & 1.28 & -0.11 \\ 0.25 & -0.11 & 1.13 \end{bmatrix}$
Σ_q	$\begin{bmatrix} 1.0 & 0.5 \\ 0.5 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.04 \\ 0.04 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.5 \\ 0.5 & 1.0 \end{bmatrix}$
α_{112}	0.0	−0.29	0.18
α_{122}	−1.0	−1.71	−2.18
α_{123}	1.0	1.0	0.26
α_{212}	1.0	0.69	2.01
α_{222}	−0.5	−0.36	−0.13
α_{223}	1.5	1.16	2.04
α_{312}	−1.0	−0.78	−1.48
α_{322}	0.0	0.01	0.75
α_{323}	0.5	0.51	1.25

Table 3.37: Multivariate penalized log likelihood approach: penalized log likelihoods (starting at Everitt's estimates 1988)

Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
−1305.512	−996.358	−1478.077

As can be seen from Table 3.37, our parameter estimates provide us with a lower penalized log likelihood than Everitt's parameter estimates do.

3.6 Optimal Lambda Values

The tuning parameter λ determines the degree of penalization for our penalized likelihood. Initially, we selected the value for λ in an ad hoc manner. We found that 65 seemed to work well with the data we were using, but we sought an approach to find an optimal λ value for any data set. To find the optimal value of λ using the data, we first formed a grid of λ values, e.g., $(1, 11, 21, 31, 41, 51, 61, 71, 81, 91, 101, 111, 121, 131, 141)$. We then ran the first stage of our simulated annealing method to find the standard deviations used in the penalty term of the penalized log likelihood. We then used a method similar to Everitt's method to find parameter estimates using each λ value in the grid; we used Everitt's method as it is computationally faster. After the parameter estimates were found for each λ value, we calculated G_2^{Ev} for each. The λ value that minimized G_2^{Ev} was chosen as the optimal value to use.

Tables 3.38, 3.39, 3.40 and 3.41 display the results for data set 1 using the optimal λ value in each case. Table 3.38 displays the penalized log likelihoods, where Everitt's penalized log likelihood was found using the `optim` function in R. As can be seen, our penalized log likelihood is greater than Everitt's penalized log likelihood when μ_2 is 1.0 and 2.0 but when looking at the goodness of fit measures in Table 3.39, we see that our estimates are closer, overall, to the truth only when $\mu_2 = 2.0$. Tables 3.40 and 3.41 display our parameter estimates and Everitt's parameter estimates, respectively.

Table 3.38: Penalized log likelihood approach with optimal λ value: data set 1 penalized log likelihoods

λ	$\mu_2^{(p)}$	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
51	1.0	−115.895	−123.308	−139.372
141	2.0	−190.487	−165.798	−204.445
131	3.0	−296.581	−190.041	−195.194

Table 3.39: Goodness of fit measure with optimal λ value: data set 1

λ	$\mu_2^{(p)}$	G_2^{SA}	G_2^{Ev}
51	1.0	0.139	0.083
141	2.0	0.093	0.142
131	3.0	0.297	0.158

Table 3.40: Penalized log likelihood approach with optimal λ value: data set 1 estimates

λ	$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
51	1.0	0.492	0.011	0.114	0.316	2.131	−1.205	−0.602
141	2.0	0.496	0.555	1.489	−0.555	1.051	−1.258	−1.028
131	3.0	0.520	3.411	−1.053	0.472	1.136	−1.612	−4.397

Table 3.41: Penalized log likelihood approach with optimal λ value: data set 1 estimates using an approach similar to Everitt's approach

λ	$\mu_2^{(p)}$	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
51	1.0	0.5	0.0	0.0	0.0	17.103	−0.018	−0.802
141	2.0	0.5	−0.395	−0.001	0.0	27.524	−4.794	0.0
131	3.0	0.5	0.028	−0.003	−0.001	26.916	−4.454	−0.022

Tables 3.42, 3.43, 3.44 and 3.45 display the results for data set 2 using the optimal λ value in each case. Table 3.42 displays the penalized log likelihoods, where Everitt's penalized log likelihood was found using the `optim` function in R. As can be seen, our penalized log likelihood is greater than Everitt's penalized log likelihood only when $\Sigma_{pq} = 0.2$ but when looking at the goodness of fit measures in Table 3.43, we see that Everitt's estimates are closer, overall, to the truth in each case. Tables 3.44 and 3.45 display our parameter estimates and Everitt's parameter estimates, respectively.

Tables 3.46, 3.47, 3.48 and 3.49 display the results for data set 2 using the optimal λ value in each case. Table 3.46 displays the penalized log likelihoods, where Everitt's penalized log likelihood was found using the `optim` function in R. As can be seen, Everitt's penalized log likelihood is greater than our penalized log likelihood in each case; for this data set, the goodness of fit measures in Table 3.47 reflect a similar

Table 3.42: Penalized log likelihood approach with optimal λ value: data set 2 log likelihoods

λ	Σ_{pq}	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
141	0.2	-211.308	-199.242	-212.494
121	0.5	-272.699	-202.876	-152.730
131	0.8	-226.896	-235.312	-171.19

Table 3.43: Goodness of fit measure with optimal λ value: data set 2

λ	Σ_{pq}	G_2^{SA}	G_2^{Ev}
141	0.2	0.192	0.180
121	0.5	0.188	0.116
131	0.8	0.203	0.157

Table 3.44: Penalized log likelihood approach with optimal λ value: data set 2 estimates

λ	Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
141	0.2	0.489	-0.023	-0.066	1.388	13.873	-1.991	-1.348
121	0.5	0.485	3.955	-0.007	-1.725	4.391	0.505	1.518
131	0.8	0.503	-0.562	0.713	-0.668	4.965	-5.017	-0.624

result, in that Everitt's estimates are closer to the true parameter values, overall. Tables 3.48 and 3.49 display our parameter estimates and Everitt's parameter estimates, respectively.

For the multivariate data set, we ran the second simulated annealing loop for $200,000$ iterations; the results are included in Tables 3.50 to 3.52. As one can see from Table 3.50, our penalized log likelihood is quite a bit greater than the penalized log likelihood found using an approach similar to that of Everitt's (1988), and the true log likelihood. We can then turn our attention to the goodness of fit measures in Table 3.51 to see how our parameter estimates compare to the true parameter values, and how the parameter estimates found using the approach similar to Everitt's compare to the true parameter values. In this case, our parameter estimates were

Table 3.45: Penalized log likelihood approach with optimal λ value: data set 2 estimates using an approach similar to Everitt's approach

λ	Σ_{pq}	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
141	0.2	0.5	−0.058	0.0	−0.01	27.356	−5.429	−0.001
121	0.5	0.5	0.0	0.0	0.0	25.362	−0.602	−0.260
131	0.8	0.5	−0.231	0.0	0.0	27.663	−0.853	−1.455

Table 3.46: Penalized log likelihood approach with optimal λ value: data set 3 log likelihoods

λ	p_1	Our Penalized $\log L$	True Penalized $\log L$	Everitt's Penalized $\log L$
111	0.6	−231.401	−227.741	−176.260
141	0.7	−299.085	−246.284	−153.729
141	0.8	−212.326	−256.063	−159.757

Table 3.47: Goodness of fit measure with optimal λ value: data set 3

λ	p_1	G_2^{SA}	G_2^{Ev}
111	0.6	0.286	0.215
141	0.7	0.384	0.219
141	0.8	0.306	0.290

closer to the true parameter values than Everitt's parameter estimates were, as $G_2^{SA} < G_2^{Ev}$. The goodness of fit measures provide a quick way to determine which parameter estimates were closer to the true parameter values but we include Table 3.52 for further inspection of the differences between the true value and the estimated value for each parameter.

Table 3.48: Penalized log likelihood approach with optimal λ value: data set 3 estimates

λ	p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
111	0.6	0.499	-0.117	0.242	-2.309	6.954	-5.455	0.023
141	0.7	0.488	-1.064	0.472	-1.949	8.056	-1.794	-4.625
141	0.8	0.508	-0.156	0.983	1.051	2.367	-1.635	-2.268

Table 3.49: Penalized log likelihood approach with optimal λ value: data set 3 estimates using an approach similar to Everitt's approach

λ	p_1	$\hat{p}_1$	$\hat{\mu}_1^{(p)}$	$\hat{\mu}_2^{(p)}$	$\hat{\Sigma}_{pq}$	$\hat{\Sigma}_p$	$\hat{\alpha}_{112}$	$\hat{\alpha}_{212}$
111	0.6	0.5	2.779	0.0	0.0	25.341	-1.540	-0.084
141	0.7	0.5	-0.059	0.0	0.0	28.242	-0.023	-0.001
141	0.8	0.5	0.008	0.001	0.002	29.200	-1.745	0.001

Table 3.50: Multivariate penalized log likelihood approach with optimal $\lambda = 1$: penalized log likelihoods (starting at Everitt's estimates 1988)

Our	True	Everitt's
Penalized $\log L$	Penalized $\log L$	Penalized $\log L$
-1274.205	-4274.408	-3664.513

Table 3.51: Goodness of fit measure with optimal λ value: multivariate data set

λ	G_2^{SA}	G_2^{Ev}
1	1.956	2.842

Table 3.52: Multivariate penalized log likelihood approach with optimal $\lambda = 1$: comparing our estimates and the estimates found using Everitt's method with the true values

	Truth	Everitt's Estimate	Our Estimate
p_1	0.4	0.50	0.16
p_2	0.3	0.22	0.09
$\mu_1^{(p)}$	$(0, 0, 0)^T$	$(-3.0, 0.1, -7.1)^T$	$(-2.6, -1.3, -7.2)^T$
$\mu_2^{(p)}$	$(3.0, 3.0, 3.0)^T$	$(3.4, -2.2, -6.1)^T$	$(2.2, -3.7, -4.4)^T$
$\mu_3^{(p)}$	$(1.5, -1.5, 4.5)^T$	$(2.9, -1.9, 2.5)^T$	$(3.0, -0.1, 3.8)^T$
Σ_{pq}	$\begin{bmatrix} 0.2 & 0.0 \\ 0.5 & 0.5 \\ 0.0 & 0.4 \end{bmatrix}$	$\begin{bmatrix} 0.46 & 0.14 \\ -0.4 & -0.11 \\ -0.01 & 0.69 \end{bmatrix}$	$\begin{bmatrix} 0.75 & -0.64 \\ 0.26 & 0.08 \\ 0.54 & -0.84 \end{bmatrix}$
Σ_p	I_3	$\begin{bmatrix} 1.3 & 0.2 & -0.3 \\ 0.2 & 0.7 & 0.02 \\ -0.3 & 0.02 & 1.3 \end{bmatrix}$	$\begin{bmatrix} 2.0 & 0.7 & 0.9 \\ 0.7 & 1.2 & 0.2 \\ 0.9 & 0.2 & 1.8 \end{bmatrix}$
Σ_q	$\begin{bmatrix} 1.0 & 0.5 \\ 0.5 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & -0.05 \\ -0.05 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & 0.14 \\ 0.14 & 1.0 \end{bmatrix}$
α_{112}	0.0	2.48	3.13
α_{122}	-1.0	-7.43	-5.17
α_{123}	1.0	4.04	0.63
α_{212}	1.0	-5.05	-5.80
α_{222}	-0.5	-5.13	-6.14
α_{223}	1.5	2.00	0.80
α_{312}	-1.0	-0.52	-0.56
α_{322}	0.0	0.53	0.79
α_{323}	0.5	3.44	3.10

3.7 Analysis of Heart Disease Data

The data we used for the real data analysis is the processed Cleveland heart disease data set, archived in the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/index.html). This data set consists of 14 variables, measured on 297 patients. Each patient was diagnosed by experts to have one of five types of heart disease; we use this variable to identify the true clustering structure for the data. There are three binary variables that were measured: whether or not the patient's fasting blood sugar exceeded 120 mg/dl; whether or not the patient was male; whether or not the patient suffered from exercise-induced angina. There are five ordinal variables: chest pain type (1: typical angina, 2: atypical angina, 3: non-anginal pain, 4: asymptomatic); resting electrocardiographic results (0: normal, 1: having ST-T wave abnormality, 2: showing probable or definite left ventricular hypertrophy by Estes' criteria); slope of the peak exercise ST segment (1: upsloping, 2: flat, 3: downsloping); number of major vessels (0-3) colored by fluoroscopy; and a defect category, with levels (normal, fixed defect, reversable defect). Lastly, there are five continuous variables: resting blood pressure in mmHg; maximum heart rate achieved; ST depression induced by exercise relative to rest; serum cholesterol in mg/dl; age in years. The description of these variables were taken from the UCI Repository data page, (http://archive.ics.uci.edu/ml/datasets/Heart+Disease).

To illustrate our method, we clustered a subset of the data that consisted of one continuous and one ordinal variable from the heart disease data (maximum heart rate achieved and chest pain type, respectively). We then clustered a larger subset consisting of three continuous variables and two ordinal variables:

- maximum heart rate achieved,
- resting blood pressure in mmHg,
- serum cholesterol in mg/dl,

- whether or not the patient's fasting blood sugar exceeded 120 mg/dl,
- resting electrocardiographic results (0: normal, 1: having ST-T wave abnormality, 2: showing probable or definite left ventricular hypertrophy by Estes' criteria).

Table 3.53 displays the results from clustering the bivariate mixed data using our simulated annealing method, with the second loop having a total of $50,000$ iterations. Table 3.53 displays the probabilities for the observations belonging to each cluster; the cluster with the highest probability for that observation is the one to which that observation is assigned. To illustrate how the probabilities are found, we'll look at how the probabilities were found for observation 2 from Table 3.53. As given in Chapter 2, the cluster membership probability is found using the following forumula:

$$P(\text{cluster } j|\mathbf{x}_i) = \frac{\hat{p}_j g_j(\mathbf{x}_i, \hat{\boldsymbol{\theta}})}{f(\mathbf{x}_i; \hat{\mathbf{p}}, \hat{\boldsymbol{\theta}})}.$$

For this data, our estimates for $\hat{p}_j$ are $\hat{p}_1 = 0.06$, $\hat{p}_2 = 0.25$, $\hat{p}_3 = 0.51$, $\hat{p}_4 = 0.16$, and $\hat{p}_5 = 0.02$. Thus, for observation 2, we have the following:

$$P(\texttt{Cl}_1) = \frac{0.06 * 0.02}{0.06 * 0.02 + 0.25 * 0.004 + 0.51 * 0.004 + 0.16 * 0.009 + 0.02 * 0.03} = 0.22,$$

$$P(\texttt{Cl}_2) = \frac{0.25 * 0.004}{0.06 * 0.02 + 0.25 * 0.004 + 0.51 * 0.004 + 0.16 * 0.009 + 0.02 * 0.03} = 0.15,$$

$$P(\texttt{Cl}_3) = \frac{0.51 * 0.004}{0.06 * 0.02 + 0.25 * 0.004 + 0.51 * 0.004 + 0.16 * 0.009 + 0.02 * 0.03} = 0.31,$$

$$P(\texttt{Cl}_4) = \frac{0.16 * 0.009}{0.06 * 0.02 + 0.25 * 0.004 + 0.51 * 0.004 + 0.16 * 0.009 + 0.02 * 0.03} = 0.24,$$

and

$$P(\texttt{Cl}_5) = \frac{0.02 * 0.03}{0.06 * 0.02 + 0.25 * 0.004 + 0.51 * 0.004 + 0.16 * 0.009 + 0.02 * 0.03} = 0.08.$$

Thus, observation 2 would be included in cluster 3, as this is the largest clustering membership probability.

Table 3.53: Heart Disease bivariate data cluster probabilities and cluster assignment

Obs.	P(Cl. 1)	P(Cl. 2)	P(Cl. 3)	P(Cl. 4)	P(Cl. 5)	Cluster Assign.
1	1.0	≈ 0	≈ 0	≈ 0	≈ 0	1
2	0.22	0.15	0.31	0.24	0.08	3
3	0.08	0.20	0.42	0.25	0.05	3
4	≈ 0	0.27	0.55	0.17	0.01	3
5	0.01	0.26	0.53	0.19	0.01	3
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
295	0.04	0.22	0.46	0.24	0.03	3
296	0.16	0.17	0.35	0.25	0.07	3
297	0.01	0.26	0.53	0.19	0.01	3

Our clustering solution grouped the data into 2 clusters. Two measures of the goodness of a clustering solution, relative to a gold standard structure, are the Rand and Adjusted Rand indices. The higher the value of each index, the better the solution reflects the gold-standard structure. These indices are formally defined in Section 4.1. The Adjusted Rand Index for the clustering result when compared to the expert diagnosis for each patient was -0.001 and the Rand Index was 0.41. As the expert's diagnosis had a total of 5 categories, our clustering solution could shed new perspective on the nature of the clustering structure. We now look at using our algorithm with a multivariate subset of the data.

The results in Table 3.54 are those from the clustering of the multivariate mixed data. This table displays the probabilities for the observations belonging to each cluster; the cluster with the highest probability for that observation is the one in which that observation is assigned to. For these results, we ran the second loop in the simulated annealing method a total of $50,000$ iterations. The Adjusted Rand Index for the clustering result when compared to the expert diagnosis for each patient was -0.03 and the Rand Index was 0.49.

For the multivariate mixed data clustering solution, the patients were separated

Table 3.54: Heart Disease multivariate data cluster probabilities and cluster assignment

Obs.	P(Cl. 1)	P(Cl. 2)	P(Cl. 3)	P(Cl. 4)	P(Cl. 5)	Cluster Assign.
1	≈ 0	0.70	≈ 0	0.30	≈ 0	2
2	≈ 0	0.87	≈ 0	0.13	≈ 0	2
3	≈ 0	0.80	≈ 0	0.20	≈ 0	2
4	0.02	0.16	0.64	0.06	0.12	3
5	≈ 0	0.74	≈ 0	0.26	≈ 0	2
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
295	0.19	0.01	0.45	0.01	0.34	3
296	0.01	0.50	0.28	0.14	0.07	3
297	≈ 0	0.74	≈ 0	0.26	≈ 0	2

into 4 clusters. Figure 3.15 is a scatterplot matrix of the continuous variables from the multivariate mixed data subset from the Heart Disease data, with the clusters distinguished. As this plot only displays the continuous variables, it does not depict the entire framework behind the clustering structure of the data but it allows us to make some conclusions. As one can see from Figure 3.15, patients that had higher maximum heart rate achieved values and lower resting blood pressure were clustered together with our algorithm, as represented in the plot by squares; these patients also had lower values of serum cholesterol. Patients with mid-level values of serum cholesterol and mid- to high-level values of maximum heart rate achieved were clustered together with our algorithm, as represented in the plot by triangles; these patients also had mid- to high-level values for resting blood pressure. The largest cluster included a wide range of values for the maximum heart rate achieved, a wide range of values for resting blood pressure, and a wide range of values for serum cholesterol, as represented in the plot by circles. There was one patient that was placed into a cluster by his/herself with our algorithm, as represented by an upside down triangle; this patient had a relatively high value for resting blood pressure, a low- to mid- level value for maximum heart rate achieved and a mid-level value for serum cholesterol.

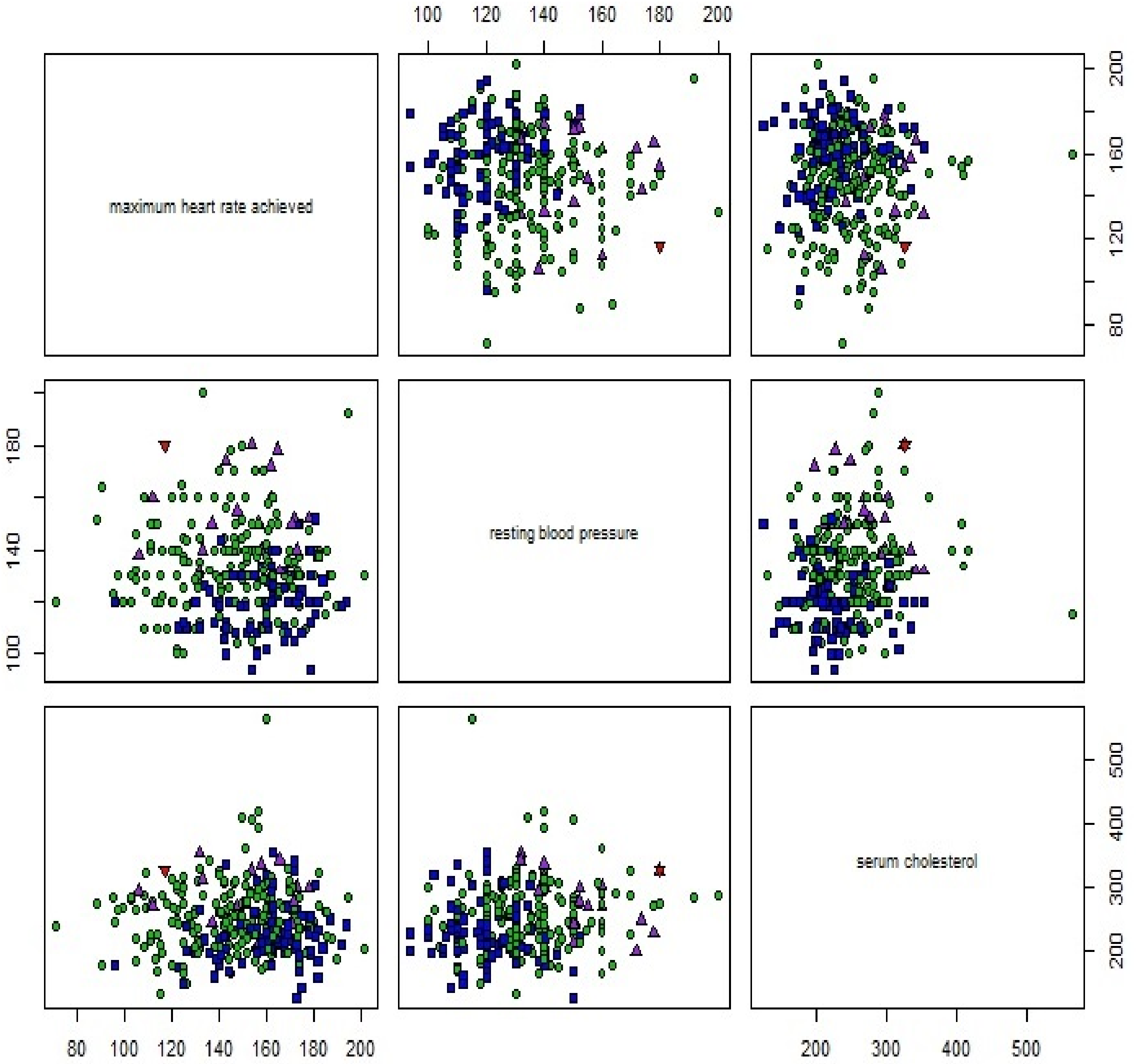

Figure 3.15: Plot of the continuous variables from the multivariate mixed data subset from the Heart Disease data (clusters distinguished)

Chapter 4

Extended Gower Coefficient

Gower (1971) described a measure of similarity that can be used for mixed data, namely, data that contain continuous, categorical and nominal variables. Kaufman and Rousseeuw (1990) describe a dissimilarity measure that is a generalization of the Gower coefficient; to find the similarity between two objects using the dissimilarity measure described below, one simply takes $1 - d(i, j)$. The dissimilarity between two objects, i and j, is defined as follows

$$d(i, j) = \frac{\sum_f \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_f \delta_{ij}^{(f)}},$$

where $\delta_{ij}^{(f)}$ is 1 if both measurements x_{if} and x_{jf} for the fth variable are nonmissing, and is 0 otherwise; $\delta_{ij}^{(f)}$ is also 0 if f is an asymmetric binary attribute and objects i and j give a $0 - 0$ match (Kaufman & Rousseeuw, 1990). If variable f is binary or nominal, then

$$d_{ij}^{(f)} = \begin{cases} 1 & \text{if } \; x_{if} \neq x_{jf}, \\ 0 & \text{if } \; x_{if} = x_{jf} \end{cases}$$

If all variables are nominal or symmetric binary variables, then $d(i, j)$ becomes the simple matching coefficient (the number of matches over the total number of number of available pairs); if the variables are asymmetric binary variables then $d(i, j)$ becomes the Jaccard coefficient (Kaufman & Rousseeuw, 1990). If variable f is interval-scaled, then

$$d_{ij}^{(f)} = \frac{|x_{if} - x_{jf}|}{\max_h(x_{hf}) - \min_h(x_{hf})}.$$

To create an extension of the Gower coefficient, we would like to be able to include other types of variables, such as directional and functional. In order to include those types of variables, we need to have a reasonable measure for the dissimilarity between two objects. Ackermann (1997) described a dissimilarity measure if the variables are directional; in this case,

$$d_{ij}^{(f)} = \pi - |\pi - |\theta_i - \theta_j||,$$

where θ_i is the angle measured on object i.

To measure the dissimilarity between two curves, or functions, there are several options, the most common being being the L_2 distance. The L_2 distance between functions $y_i(t)$ and $y_j(t)$ is defined as

$$d_{ij}^{(f)} = \sqrt{\int_T \Big[y_i(t) - y_j(t)\Big]^2 dt}.$$

Another option is to use a weighted L_2 distance. The weighted L_2 distance between functions $y_i(t)$ and $y_j(t)$ is defined as

$$d_{w_{ij}}^{(f)} = \sqrt{\int_T w(t)\Big[y_i(t) - y_j(t)\Big]^2 dt}.$$

Chen et al. (2014) considered the following weighting function

$$w(t) = \frac{\frac{1}{\hat{\sigma}^2(t)}}{\int_T (\frac{1}{\hat{\sigma}^2(u)}) du},$$

where $\hat{\sigma}^2(t)$ is the sample variance of all $y_i(t) - y_j(t)$ values within some small interval around t such that $t \,\epsilon\, [t - \delta, t + \delta]$. This function $w(t)$ puts more weight on those regions where the curves are further apart and less weight on the regions where the curves are closer together (Chen et al. 2014). To prevent the functional variables from having more or less influence than the other variables, we scale either of the dissimilarity measures described above as follows:

$$\frac{d_{ij}^{(f)}}{\max_{i,j} d_{ij}^{(f)} - \min_{i,j} d_{ij}^{(f)}}.$$

To estimate the curves for each functional variable, one option is to use B-splines, a form of nonparametric regression; in that case, the dissimilarities could be computed using the B-spline coefficients. Suppose the coefficients for curve i were $\hat{\beta}_{0i}, \hat{\beta}_{1i}, \ldots, \hat{\beta}_{ki}$ and for curve j were $\hat{\beta}_{0j}, \hat{\beta}_{1j}, \ldots, \hat{\beta}_{kj}$. The dissimilarity between the ith and jth curves would be calculated as follows:

$$d_{ij} = \sqrt{\left(\hat{\beta}_{0i} - \hat{\beta}_{0j}\right)^2 + \left(\hat{\beta}_{1i} - \hat{\beta}_{1j}\right)^2 + \cdots + \left(\hat{\beta}_{ki} - \hat{\beta}_{kj}\right)^2}.$$

4.1 Data Analysis

The data we will be using are from the National Data Buoy Center, given in the historical section of the National Oceanic and Atmospheric Administration (NOAA) website, `www.ndbc.noaa.gov`. The objects to be clustered are a set of stations at which measurements are taken via weather buoys. The variables of interest to us are wind direction, wind speed, air temperature, and water temperature, along with the latitude, longitude and water depth that corresponds to each buoy. Certain buoys were excluded from the analysis because of excessive missing values. To impute missing values in variables that were time series, we took the average of the observations before and after the missing observation. The number of observations and the measurement times for the functional data varied from buoy to buoy. To align the data, we defined our time index to be the number of minutes since Jan. 1, 2011 at 12:50 AM. The time in the original data sets was given as year, month, day, hour, and minute. In total, our data set consisted of 26 buoys, with one buoy being included in two different regions. In addition, we considered the time zone (a nominal variable) for each buoy. The time zones in our buoy data were Central, Eastern, and Pacific. In order to use this variable in the Gower coefficient, each time zone was given a numerical label, ‘Eastern’=1, ‘Central’=2, and ‘Pacific’=3.

As a starting point, we performed k-means clustering on fitted values from applying a B-spline smoother for each curve, representing each station, separately on each functional variable: wind direction, wind speed, air temperature, and water temperature. Since each buoy had differing numbers of measurement points, we specified a common set of points at which the B-spline smoother would estimate the mean response. We then did k-means clustering on these fitted values for each curve, based on that common grid of time points. We also did k-means clustering on water depth and latitude and longitude, treating these two as bivariate data. Since we are using an extension of the Gower coefficient, we could instead transform latitude and longitude to a directional variable. To transform from Cartesian coordinates to polar coordinates, we use the following formulas: $x = r\cos\theta$ and $y = r\sin\theta$, where $r = \sqrt{x^2 + y^2}$. Treating x as the longitude and y as the latitude, we can find our direction by solving for θ. Since latitude and longitude are measured using the intersection of the prime meridian and the equator as the reference point, we allow for the possibility of changing our reference point. Since all of the buoys in our data surround North America, all of them lie in the same general direction if we use 0° latitude, 0° longitude as our reference point. Thus, a more general formula for the direction, where we can change our reference point, is $\theta = \arcsin \frac{y-y_0}{\sqrt{(x-x_0)^2+(y-y_0)^2}}$, where x is the longitude and y is the latitude for each buoy and x_0 is the longitude and y_0 is the latitude at the reference point. So, for example, buoy 41046 is at 23.838° N 68.333° W; the value for θ for this buoy would be 0.335 radians, when using the intersection of the prime meridian and the equator as the reference point ($x_0 = 0$ and $y_0 = 0$).

To perform k-means clustering in R, we used the function `kmeans` in the `stats` package. In this function, we specified the number of clusters by the `centers` argument, and the number of random starts of the algorithm by the `nstart` argument. In order to have a stable clustering of the data, we set `nstart` to be 25. Table 4.1 through Table 4.6 give the results from the k-means clustering under the assumption of $k = 7$ clusters, since there are 7 regions in which these buoys are located according to the National Data Buoy Center. We also present the clustering solution when the "true" number of clusters is underspecified ($k = 6$) or overspecified ($k = 8$).

Table 4.1: k-means on Wind Direction

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	6	5	2
41047	West Atlantic	2	1	1
41048	West Atlantic	2	6	8
44007	NE USA	5	3	4
44009	NE USA	2	6	8
41004	SE USA	2	6	8
41012	SE USA	2	6	8
44020	SE USA	2	6	8
46012	NW USA	3	4	3
46027	NW USA	6	5	5
46041	NW USA	2	1	1
46042	NW USA	3	4	3
46059	NW USA	4	7	6
46011	SW USA	3	4	3
46025	SW USA	6	5	5
46028	SW USA	3	4	3
46053	SW USA	6	5	5
46054	SW USA	3	4	5
46086	SW USA	6	5	5
42020	FL/Gulf of Mexico	6	5	2
41012	FL/Gulf of Mexico	2	6	8
42036	FL/Gulf of Mexico	2	1	1
42039	FL/Gulf of Mexico	5	3	4
42040	FL/Gulf of Mexico	6	5	2
42056	West Caribbean	2	1	1
42055	West Caribbean	1	2	7

Table 4.2: *k*-means on Wind Speed

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	5	3	3
41047	West Atlantic	3	1	6
41048	West Atlantic	2	6	1
44007	NE USA	2	6	1
44009	NE USA	5	3	3
41004	SE USA	2	6	1
41012	SE USA	2	6	1
44020	SE USA	2	6	1
46012	NW USA	5	3	3
46027	NW USA	1	7	7
46041	NW USA	2	6	1
46042	NW USA	5	3	3
46059	NW USA	4	5	5
46011	SW USA	4	4	8
46025	SW USA	4	4	8
46028	SW USA	5	3	3
46053	SW USA	4	4	8
46054	SW USA	6	5	5
46086	SW USA	4	4	8
42020	FL/Gulf of Mexico	5	3	3
41012	FL/Gulf of Mexico	2	6	1
42036	FL/Gulf of Mexico	2	6	1
42039	FL/Gulf of Mexico	6	2	2
42040	FL/Gulf of Mexico	6	2	2
42056	West Caribbean	1	7	4
42055	West Caribbean	6	2	7

Table 4.3: *k*-means on Air Temperature

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	4	3	5
41047	West Atlantic	4	3	5
41048	West Atlantic	4	6	1
44007	NE USA	1	1	2
44009	NE USA	1	1	8
41004	SE USA	3	4	7
41012	SE USA	3	4	7
44020	SE USA	1	1	2
46012	NW USA	6	5	3
46027	NW USA	6	5	3
46041	NW USA	2	7	4
46042	NW USA	6	5	3
46059	NW USA	2	7	4
46011	SW USA	6	7	3
46025	SW USA	6	7	3
46028	SW USA	6	5	3
46053	SW USA	6	7	3
46054	SW USA	6	7	3
46086	SW USA	6	7	3
42020	FL/Gulf of Mexico	3	4	7
41012	FL/Gulf of Mexico	3	4	7
42036	FL/Gulf of Mexico	3	4	7
42039	FL/Gulf of Mexico	3	4	7
42040	FL/Gulf of Mexico	5	2	6
42056	West Caribbean	4	3	5
42055	West Caribbean	4	6	1

Table 4.4: *k*-means on Water Temperature

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	5	2	6
41047	West Atlantic	4	3	5
41048	West Atlantic	4	3	5
44007	NE USA	1	6	8
44009	NE USA	1	5	1
41004	SE USA	2	7	4
41012	SE USA	2	7	4
44020	SE USA	1	5	1
46012	NW USA	6	4	3
46027	NW USA	6	4	3
46041	NW USA	6	4	2
46042	NW USA	6	4	3
46059	NW USA	3	1	7
46011	SW USA	3	1	7
46025	SW USA	3	1	7
46028	SW USA	6	4	3
46053	SW USA	3	1	7
46054	SW USA	3	1	7
46086	SW USA	3	1	7
42020	FL/Gulf of Mexico	4	3	5
41012	FL/Gulf of Mexico	2	7	4
42036	FL/Gulf of Mexico	2	7	4
42039	FL/Gulf of Mexico	4	3	5
42040	FL/Gulf of Mexico	4	3	5
42056	West Caribbean	5	2	6
42055	West Caribbean	5	2	6

Table 4.5: *k*-means on Water Depth

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	1	3	7
41047	West Atlantic	1	3	7
41048	West Atlantic	1	3	7
44007	NE USA	4	7	5
44009	NE USA	4	7	5
41004	SE USA	4	7	5
41012	SE USA	4	7	5
44020	SE USA	4	7	5
46012	NW USA	4	6	3
46027	NW USA	4	7	5
46041	NW USA	4	7	5
46042	NW USA	5	2	1
46059	NW USA	6	5	6
46011	SW USA	4	6	3
46025	SW USA	2	1	4
46028	SW USA	2	1	4
46053	SW USA	2	6	2
46054	SW USA	2	6	2
46086	SW USA	5	2	1
42020	FL/Gulf of Mexico	4	7	5
41012	FL/Gulf of Mexico	4	7	5
42036	FL/Gulf of Mexico	4	7	5
42039	FL/Gulf of Mexico	4	6	3
42040	FL/Gulf of Mexico	4	7	3
42056	West Caribbean	6	5	6
42055	West Caribbean	3	4	8

Table 4.6: *k*-means on Latitude and Longitude

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	6	7	5
41047	West Atlantic	6	7	5
41048	West Atlantic	6	7	5
44007	NE USA	5	1	2
44009	NE USA	5	1	2
41004	SE USA	2	4	8
41012	SE USA	2	4	8
44020	SE USA	5	1	2
46012	NW USA	3	6	1
46027	NW USA	1	5	6
46041	NW USA	1	5	6
46042	NW USA	3	6	1
46059	NW USA	1	5	3
46011	SW USA	3	6	1
46025	SW USA	3	6	1
46028	SW USA	3	6	1
46053	SW USA	3	6	1
46054	SW USA	3	6	1
46086	SW USA	3	6	1
42020	FL/Gulf of Mexico	4	3	7
41012	FL/Gulf of Mexico	2	4	8
42036	FL/Gulf of Mexico	2	2	4
42039	FL/Gulf of Mexico	2	2	4
42040	FL/Gulf of Mexico	2	2	4
42056	West Caribbean	4	2	4
42055	West Caribbean	4	3	7

Rand (1971) introduced a concordance index to compare two clusterings, C_1 and C_2, based on pairs of objects, where C_1 is one clustering result from the data and C_2 is another (different) clustering result for the same data. This index has a range from 0 to 1, with 0 implying the two clusterings have nothing in common and 1 occurring when the two clusterings are the same (Rand, 1971). The Rand Index is defined as follows:

$$R = \frac{a+b}{a+b+c+d},$$

where a is the number of pairs of objects placed in the same cluster by both clusterings, b is the number of pairs of objects placed in different clusters by both clusterings, c is the number of pairs of objects placed in the same cluster by the clustering outcome C_1 but in different clusters by clustering outcome C_2, and d is the number of pairs of objects placed in the same cluster by clustering outcome C_2 but in different clusters by clustering outcome C_1. Another measure used to compare two clusterings is the Adjusted Rand Index (Hubert and Arabie, 1985), which is defined as

$$ARI = \frac{\sum\limits_{ij} \binom{n_{ij}}{2} - [\sum\limits_{i} \binom{a_i}{2} \sum\limits_{j} \binom{b_j}{2}]/\binom{N}{2}}{\frac{1}{2}[\sum\limits_{i} \binom{a_i}{2} + \sum\limits_{j} \binom{b_j}{2}] - [\sum\limits_{i} \binom{a_i}{2} \sum\limits_{j} \binom{b_j}{2}]/\binom{N}{2}},$$

where n_{ij} denotes the number of objects that are common to clusters C_{1i} and C_{2j}, where C_{1i} is the i^{th} cluster in the first clustering and C_{2j} is the j^{th} cluster in the second clustering, the a_i and b_j's are the marginal sums, $a_i = \sum\limits_{j} n_{ij}$ and $b_j = \sum\limits_{i} n_{ij}$, and $\sum\limits_{ij} n_{ij} = N$. (Vinh, Epps and Bailey, 2009). If one of the clustering outcomes (say, C_1) represents the "true" partition of the objects, then these indices measure the closeness of clustering outcome C_2 to the "truth" (i.e., the goodness of C_2).

We let the partitioning of the stations into their seven regions represent the "true" clustering, C_1, and the clustering result from our k-means clustering specifying 7 clusters be the other clustering, C_2. We also clustered the stations based on each functional variable individually using the R package `fda.usc` (Febrero-Bande, Oviedo de la Fuente, 2012). Using the R function `kmeans.fd`, which performs k-means clustering for functional data, we get the clustering results given in Tables 4.8−4.11. Then we compared the "truth" to the achieved clustering using the two criteria defined above. Table 4.7 shows the Adjusted Rand Index and the Rand Index for the clustering based on each variable separately.

Table 4.7: Adjusted Rand Index and Rand Index values

Variable	ARI	RI
Water Depth	0.1808	0.76
Latitude & Longitude	0.5359	0.8892
Wind Direction	0.2050	0.8031
Wind Speed	0.0757	0.7292
Air Temperature	0.3981	0.84
Water Temperature	0.4577	0.8092

An extension of the Gower coefficient will allow for distances based on other types of variables, specifically functional and directional. In order to cluster the data using all of the variables jointly, we used the Gower extension. To use the Gower extension with these data, we calculated the dissimilarities as described previously for each variable type. In order to calculate a dissimilarity component for interval-scaled, or continuous, variables, we needed to calculate the L_1 distance,

$$d_{L_1}^{(f)}(i,j) = \left| x_{if} - x_{jf} \right|,$$

between observations x_i and x_j for the f^{th} variable. Once these dissimilarities were calculated, we then divided by the range of the dissimilarities for the f^{th} variable.

To calculate a dissimilarity component for functional variables, we needed to calculate the functional L_2 distance,

$$d_{L_2}^{(f)}(i,j) = \Big[\int_T [x_{if} - x_{jf}]^2 dt \Big]^{\frac{1}{2}},$$

(in order to calculate the dissimilarity measure previously described for functional variables) using the R function `metric.lp` in the R package `fda.usc`. This function calculates an approximate L_p distance for functional data using Simpson's rule

Table 4.8: *k*-means functional data clustering: Wind Direction

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	2	6	4
41047	West Atlantic	2	3	2
41048	West Atlantic	4	2	7
44007	NE USA	3	3	7
44009	NE USA	4	3	7
41004	SE USA	4	2	7
41012	SE USA	4	3	7
44020	SE USA	4	3	7
46012	NW USA	1	7	1
46027	NW USA	6	7	8
46041	NW USA	6	1	8
46042	NW USA	1	5	6
46059	NW USA	6	2	8
46011	SW USA	1	5	6
46025	SW USA	6	1	8
46028	SW USA	1	5	1
46053	SW USA	6	1	8
46054	SW USA	1	5	1
46086	SW USA	6	1	6
42020	FL/Gulf of Mexico	2	6	2
41012	FL/Gulf of Mexico	4	3	7
42036	FL/Gulf of Mexico	3	2	3
42039	FL/Gulf of Mexico	3	2	3
42040	FL/Gulf of Mexico	4	2	3
42056	West Caribbean	2	6	4
42055	West Caribbean	5	4	5

(Febrero-Bande and Oviedo de la Fuente, 2012). Since the arguments in the R function `metric.lp` require the data to be a functional data object, we converted the fitted values from fitting the B-splines for each functional variable to a functional data object using the R function `fdata` in the R package `fda.usc`.

Table 4.9: *k*-means functional data clustering: Wind Speed

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	1	6	6
41047	West Atlantic	5	6	6
41048	West Atlantic	6	1	7
44007	NE USA	5	2	6
44009	NE USA	5	2	6
41004	SE USA	1	1	6
41012	SE USA	5	2	6
44020	SE USA	1	6	6
46012	NW USA	1	1	1
46027	NW USA	5	1	6
46041	NW USA	5	2	6
46042	NW USA	1	1	1
46059	NW USA	6	1	4
46011	SW USA	3	1	1
46025	SW USA	4	7	8
46028	SW USA	2	4	5
46053	SW USA	3	7	2
46054	SW USA	2	3	5
46086	SW USA	3	7	2
42020	FL/Gulf of Mexico	1	1	6
41012	FL/Gulf of Mexico	5	2	6
42036	FL/Gulf of Mexico	5	5	3
42039	FL/Gulf of Mexico	5	5	3
42040	FL/Gulf of Mexico	5	2	6
42056	West Caribbean	1	1	6
42055	West Caribbean	1	1	1

To calculate a dissimilarity component for directional variables, we wrote a function that calculated the Ackerman (1997) distance

$$d_{ij}^{(f)} = \pi - |\pi - |\theta_i - \theta_j||,$$

between each pair of objects. We also wrote a simple function to calculate the dis-

Table 4.10: *k*-means functional data clustering: Air Temperature

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	5	6	7
41047	West Atlantic	5	6	7
41048	West Atlantic	1	7	6
44007	NE USA	2	3	3
44009	NE USA	6	5	3
41004	SE USA	1	4	6
41012	SE USA	1	4	4
44020	SE USA	6	5	3
46012	NW USA	3	1	1
46027	NW USA	3	1	1
46041	NW USA	3	1	5
46042	NW USA	3	1	1
46059	NW USA	3	1	2
46011	SW USA	3	1	2
46025	SW USA	3	1	2
46028	SW USA	3	1	2
46053	SW USA	3	1	2
46054	SW USA	3	1	2
46086	SW USA	3	1	2
42020	FL/Gulf of Mexico	1	4	4
41012	FL/Gulf of Mexico	1	4	4
42036	FL/Gulf of Mexico	1	4	4
42039	FL/Gulf of Mexico	1	4	4
42040	FL/Gulf of Mexico	1	4	4
42056	West Caribbean	4	2	8
42055	West Caribbean	5	6	7

similarity measure

$$d_{ij}^{(f)} = \begin{cases} 1 & \text{if} \quad x_{if} \neq x_{jf}, \\ \\ 0 & \text{if} \quad x_{if} = x_{jf} \end{cases}$$

for nominal variables.

Table 4.11: *k*-means functional data clustering: Water Temperature

Station	Region	Cluster Label (6 clusters)	Cluster Label (7 clusters)	Cluster Label (8 clusters)
41046	West Atlantic	5	4	8
41047	West Atlantic	2	6	4
41048	West Atlantic	2	1	3
44007	NE USA	3	7	2
44009	NE USA	6	7	7
41004	SE USA	2	1	3
41012	SE USA	2	1	3
44020	SE USA	3	7	5
46012	NW USA	1	3	1
46027	NW USA	1	2	1
46041	NW USA	1	5	1
46042	NW USA	1	3	1
46059	NW USA	1	3	1
46011	SW USA	1	3	1
46025	SW USA	1	3	1
46028	SW USA	1	3	1
46053	SW USA	1	3	1
46054	SW USA	1	3	1
46086	SW USA	1	3	1
42020	FL/Gulf of Mexico	2	6	4
41012	FL/Gulf of Mexico	2	1	3
42036	FL/Gulf of Mexico	2	1	3
42039	FL/Gulf of Mexico	2	6	4
42040	FL/Gulf of Mexico	2	6	4
42056	West Caribbean	4	4	8
42055	West Caribbean	5	4	6

Then, the dissimilarity between two objects, i and j, is

$$d(i,j) = \frac{\sum_f \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_f \delta_{ij}^{(f)}},$$

where $\delta_{ij}^{(f)}$ is 1 if both measurements x_{if} and x_{jf} for the fth variable are nonmissing, and is 0 otherwise (Kaufman & Rousseeuw, 1990). Thus, if both measurements

x_{if} and x_{jf} for the fth variable are nonmissing, the dissimilarity between the ith and jth objects will simply be the sum (over the number of variables) of all of the dissimilarities calculated for the ith and jth objects, divided by the sum of the number of variables.

For our data, we had water depth, a continuous variable, and recall that we treated latitude and longitude as one bivariate continuous variable in our first clustering. We also had the functional variables wind direction, wind speed, air temperature, and water temperature; the nominal variable time zone; and for the second clustering, we calculated a direction from the latitude and longitude for a directional variable. We formed a 26×26 dissimilarity matrix filled with the extended Gower dissimilarities calculated as described above. We then used this dissimilarity matrix as an input to the R package `hclust` to produce a clustering result. Based upon the dendrogram, Figure 4.2, and the choice of 7 clusters (to represent the 7 regions), Tables 4.12 and 4.13 show the clusterings of the data, based on all of the variables. The Adjusted Rand Index is 0.5675 and the Rand Index is 0.8831. Note that the index values for the clustering based on all variables compare favorably to those for the clusterings based on each value separately, given in Table 4.7.

As can be seen from Tables 4.12 and 4.13, the 7-cluster solution gets the correct structure for the Western Atlantic Region, Northeast USA, Southwest USA, and the Western Caribbean. Some stations the solution misclassified were: Station 44020 (nominally in the Southeast USA region) was included in the Northeast USA cluster; and Station 46059 (nominally in the Northwest USA region) was the only one in its own cluster; the other stations in Northwest USA were included in the Southwest USA cluster. Station 41012 (nominally in the Florida/ Gulf of Mexico region and in the Southeast USA region) was included in the Southeast USA cluster. A map reveals where each of those ‘trouble’ stations were located. Station 44020 was the Nantucket Sound buoy; and Station 46059 was located west of California. Figure 4.1 is a US map with the buoy locations illustrated with our clustering solution, as indicated with different symbols for each cluster. With the locations of these buoys in mind, the clustering solution classified them into regions relatively close geographically to their "true" regions and so it seems that the 7-cluster solution does a reasonable job at putting the stations in their correct regions.

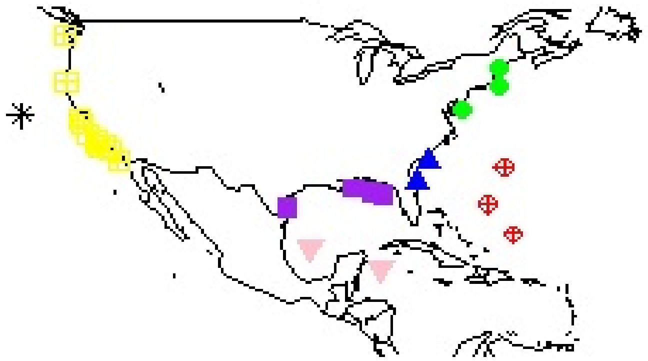

Figure 4.1: US map with buoy locations

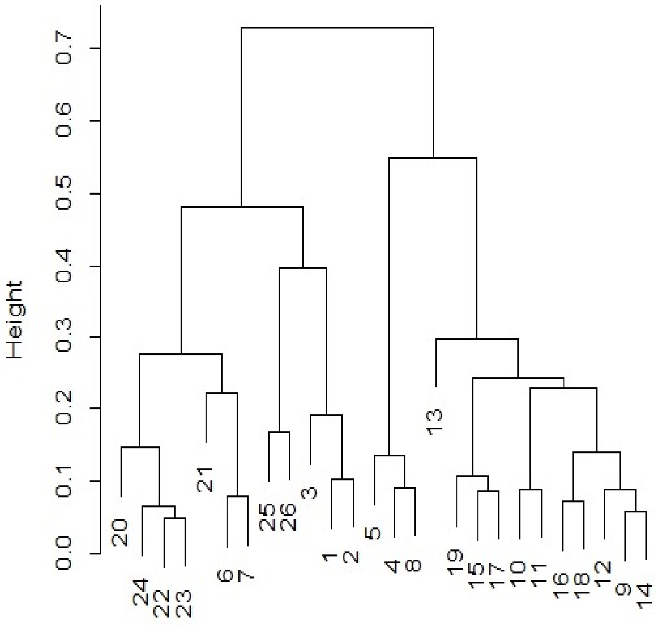

Figure 4.2: Dendrogram using the Gower Extension

Table 4.12: Clustering solution using the Gower Extension

Station	Region	Cluster Label, (7 clusters)
41046	Western Atlantic	1
41047	Western Atlantic	1
41048	Western Atlantic	1
44007	Northeast USA	2
44009	Northeast USA	2
41004	Southeast USA	3
41012	Southeast USA	3
44020	Southeast USA	2
46012	Northwest USA	4
46027	Northwest USA	4
46041	Northwest USA	4
46042	Northwest USA	4
46059	Northwest USA	5
46011	Southwest USA	4
46025	Southwest USA	4
46028	Southwest USA	4
46053	Southwest USA	4
46054	Southwest USA	4
46086	Southwest USA	4
42020	Florida/Gulf of Mexico	6
41012	Florida/Gulf of Mexico	3
42036	Florida/Gulf of Mexico	6
42039	Florida/Gulf of Mexico	6
42040	Florida/Gulf of Mexico	6
42056	Western Caribbean	7
42055	Western Caribbean	7

Table 4.13: Clustering solution using the Gower Extension with the directional variable

Station	Region	Cluster Label, (7 Clusters)
41046	Western Atlantic	1
41047	Western Atlantic	1
41048	Western Atlantic	1
44007	Northeast USA	2
44009	Northeast USA	2
41004	Southeast USA	3
41012	Southeast USA	3
44020	Southeast USA	2
46012	Northwest USA	4
46027	Northwest USA	4
46041	Northwest USA	4
46042	Northwest USA	4
46059	Northwest USA	5
46011	Southwest USA	4
46025	Southwest USA	4
46028	Southwest USA	4
46053	Southwest USA	4
46054	Southwest USA	4
46086	Southwest USA	4
42020	Florida/Gulf of Mexico	6
41012	Florida/Gulf of Mexico	3
42036	Florida/Gulf of Mexico	6
42039	Florida/Gulf of Mexico	6
42040	Florida/Gulf of Mexico	6
42056	Western Caribbean	7
42055	Western Caribbean	7

4.2 Simulation Study

We conducted a simulation study to determine how the method based on the extended Gower coefficient performed in clustering data having a variety of structures. The extended Gower coefficient is a method that allows for continuous, categorical, directional and functional variables to be used simultaneously to cluster a data set. To compare how the results from using the extended Gower coefficient differ from methods that allow only one or two of the variable types to be used in the cluster analysis, we compared the average Rand Index, for 1000 simulations, for the clustering results using (1) the extended Gower coefficient, (2) the original Gower coefficient, (3) the Ackermann distance, and (4) the L_2 distance. The purpose of the simulations was to see whether clustering results would be different and how such a difference would depend on the underlying cluster structure. With that in mind, we simulated data similar to the buoy data from the real data analysis.

To simulate a categorical variable (like time zone), we used the R function `sample.int`, which allows us to sample from a multinomial distribution. With this function, we were able to sample with replacement from five categories. Thus, the multinomial probability function we are using has the following form

$$\frac{N!}{x_1!x_2!x_3!x_3!x_4!x_5!}p_1^{x_1}p_2^{x_2}p_3^{x_3}p_4^{x_4}p_5^{x_5},$$

where $N = \sum_{i=1}^{5} x_i$, p_i is the corresponding probability for each category for $i = 1, 2, 3, 4, 5$ and x_i is the number of outcomes in each category for $i = 1, 2, 3, 4, 5$ (Lehmann and Casella, 1998). We were also able to let the cluster sizes differ and for each cluster we tried different probability vectors, in order to simulate clusters of data that have different probabilities of coming from each category. The probability vectors were $(0.8, 0.05, 0.05, 0.05, 0.05)$, which corresponds to one dominant category, and $(0.2, 0.2, 0.2, 0.2, 0.2)$, which corresponds to equiprobable categories.

To simulate a continuous variable (like water depth), we simulated from a normal distribution with mean μ and standard deviation σ, and then took the absolute value of those values. We considered the following values for μ and σ: for cluster 1, $\mu = 5000$ and $\sigma = 100$, for cluster 2, $\mu = 5000 + k\sigma$ and $\sigma = 100$, for cluster 3, $\mu = 5000 + 2k\sigma$ and $\sigma = 100$, and for cluster 4, $\mu = 5000 + 3k\sigma$ and $\sigma = 100$. We allowed k to vary from small to moderate to large; for example, $k = 5$, $k = 20$, and $k = 50$. Increasing k implies allowing a greater separation between clusters.

To simulate a directional variable, θ, we used the von Mises distribution. The von Mises distribution is a continuous probability distribution with parameters μ and κ; μ is the mean direction of the distribution, and κ is the concentration parameter of the distribution (Agostinelli and Lund, 2011). The density function of the von Mises distribution is as follows

$$\frac{\exp \kappa \cos(\theta - \mu)}{2\pi I_0(\kappa)}, 0 \leq \theta < 2\pi,$$

where $0 \leq \mu < 2\pi$, $\kappa \geq 0$ and $I_0(\kappa)$ is the modified Bessel function given by

$$I_0(\kappa) = \frac{1}{2\pi} \int\limits_0^{2\pi} \exp(\kappa \cos\theta))d\theta = \sum_{r=0}^{\infty} \left(\frac{\kappa}{2}\right)^{2r} \left(\frac{1}{r!}\right)^2$$

(Jammalamadaka and Sengupta, 2001). To simulate from this distribution in R, we used the function `rvonmises` in the `circular` package (Agostinelli and Lund, 2011). To simulate data for 4 different clusters, we considered the following values for μ and κ: for cluster 1, μ was 0 and κ was 50, so that the data were highly concentrated around 0, for cluster 2, μ was $0+k$ and κ was 50, for cluster 3, μ was $0+2k$ and κ was 50, for cluster 4, μ was $0+3k$ and κ was 50, where k varied from small to moderate to large; for example, we used $k = 0.5$, $k = 1.0$, and $k = 2.5$. As an example, Figure 4.3 is a plot for simulated data from all 4 clusters, where the mean for cluster 1 was 0, the mean for cluster 2 was 2.5 radians (143.2°), the mean for cluster 3 was 5 (286.5°) and the mean for cluster 4 was 7.5 (429.7°, or 69.7°).

Ferreira and Hitchcock (2009) provided a template for simulating functional data. The signal functions we used to simulate our functional data were the first group of

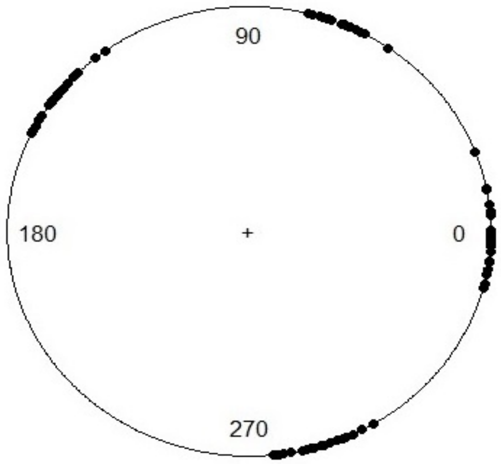

Figure 4.3: Plot of directional data

signal functions described by Ferreira and Hitchcock (2009), and are the following:

$$\mu_1(t) = \frac{1}{28}t + \exp(-t) + \frac{1}{5}\sin(t/3) + 0.5, t \in [0, 100]$$

$$\mu_2(t) = \frac{1}{20}t + \exp(-t) + \frac{1}{5}\sin(t/2), t \in [0, 100]$$

$$\mu_3(t) = \frac{1}{15}t + \exp(-t) + \frac{1}{5}\sin(t/2) - 1, t \in [0, 100]$$

$$\mu_4(t) = \frac{1}{18}t + \exp(-t) + \frac{1}{5}\sin(t/2), t \in [0, 100];$$

Figure 4.4 displays a graph of these signal functions.

As Ferreira and Hitchcock (2009) explain, these functions were chosen so that a good clustering is possible but not trivial. For our simulated data, we generated 30 discretized curves based on the functions given above. Following the framework laid out by Ferreira and Hitchcock (2009), the data was simulated over 200 points from $t = 0$ to $t = 100$ in increments of 0.5. By adding random error to the signal functions, we allowed variation among the simulated data within each cluster; in this case, we used the same process Ferreira and Hitchcock (2009) did, “a discretized

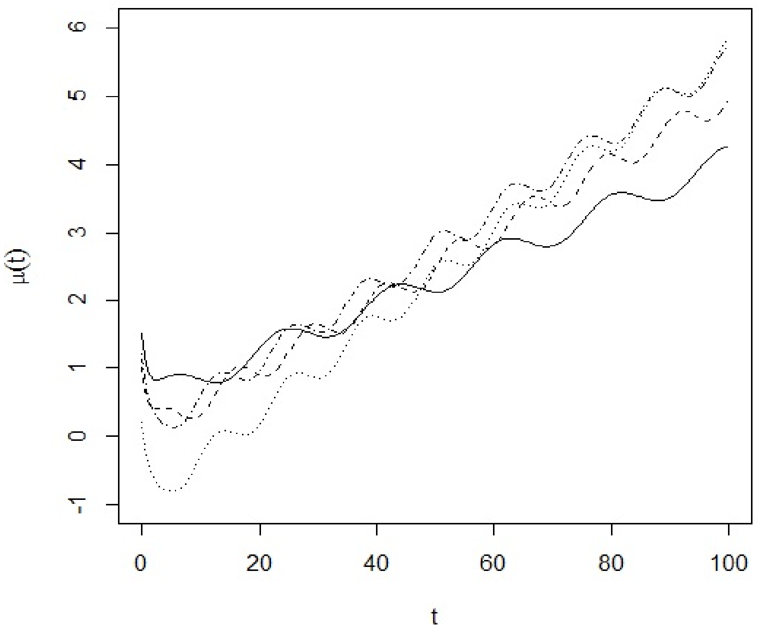

Figure 4.4: Plot of the signal functions used to simulate the functional data

approximation of the stationary Ornstein−Uhlenbeck process." This Gaussian process has a mean of zero and the covariance between errors measured at t_i and t_j is given by $\sigma^2(2\beta)^{-1}\exp(-\beta|t_i - t_j|)$ (Ferreira and Hitchcock, 2009). We kept the drift variable, β, at 0.5 and let $\sigma = 1.75$ for small distance between the clusters and $\sigma = 1$ for large distance between the clusters.

We did numerous simulations; for some settings we set one (or a few) variable(s) to have different means for each cluster, and for other simulation settings, all of the variables had different means for each cluster. In this way, we determined whether the clustering performance is more affected by differences relative to one type of variable or another. In certain settings we wanted simulated data in which there was a large distance between the clusters. In other settings, we wanted the clusters to be closer. This would give an assessment of how the extended Gower coefficient performed under a variety of data structures. The quality of the resulting clustering was judged using the Rand and Adjusted Rand indices.

Tables 4.14 and 4.15 display the settings we used for each simulation. For all of our simulations, we chose to have four clusters in the data. For each setting, we let the cluster sizes change; the cluster sizes were as follows:

- 25 objects in each cluster,
- 33 objects in clusters 1, 2, and 3 and 1 object in cluster 4,
- 10 objects in cluster 1, 20 objects in cluster 2, 30 objects in cluster 3, and 40 objects in cluster 4.

In each simulation, we let our categorical variable have five categories, with the probability for each category listed in the table. As previously described, for our two continuous variables we let $\sigma = 100$, and the mean for each cluster is given in Tables 4.14 and 4.15.

For each of the 15 simulation settings, we simulated 1000 data sets for each combination of parameter setting and cluster size allocation, and computed the average Rand Index and its standard error. Figure 4.5 compares the average Rand Index for each method we used to cluster the data (note that in the key, the capital letters indicate large separation between the clusters for that type of variable, whereas the lowercase letters indicate small separation between the clusters for that type of variable). As can be seen from Figure 4.5, where there is large separation between the clusters for all variable types, the average Rand Index is the largest when using the extended Gower coefficient, indicating that it gives the best clustering compared to the other methods. The cases where the extended Gower coefficient does not produce the largest average Rand Index are the following:

Table 4.14: Simulation Study Settings: Settings 1:8

Setting	Categorical	Continuous	Continuous	Directional	Functional
	Variable	Variable 1	Variable 2	Variable	Variable
	Probs.	Mean	Mean	Mean	sigma
1	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1$
2	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1$
3	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1$
4	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1$
5	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1.75$
6	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1$
7	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1$
8	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1.75$

- when there is large separation between the clusters for the categorical, continuous and functional variables, with small separation between the clusters for the directional variable

Table 4.15: Simulation Study Settings: Settings 9:15

Setting	Categorical Variable Probs.	Continuous Variable 1 Mean	Continuous Variable 2 Mean	Directional Variable Mean	Functional Variable sigma
9	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\10000\\15000\\20000\end{bmatrix}$	$\begin{bmatrix}500\\5500\\10500\\15500\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1.75$
10	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1$
11	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1.75$
12	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1$
13	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1$
14	$\begin{bmatrix}(0.8, 0.05, 0.05, 0.05, 0.05)\\(0.05, 0.8, 0.05, 0.05, 0.05)\\(0.05, 0.05, 0.8, 0.05, 0.05)\\(0.05, 0.05, 0.05, 0.05, 0.8)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.1\\0.2\\0.3\end{bmatrix}$	$\sigma = 1.75$
15	$\begin{bmatrix}(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\\(0.2, 0.2, 0.2, 0.2, 0.2)\end{bmatrix}$	$\begin{bmatrix}5000\\5500\\6000\\6500\end{bmatrix}$	$\begin{bmatrix}500\\1000\\1500\\2000\end{bmatrix}$	$\begin{bmatrix}0\\0.5\\1\\1.5\end{bmatrix}$	$\sigma = 1.75$

- when there is large separation between the clusters for the continuous and functional variables, and small separation between the clusters for the categorical and directional variables

- when there is large separation between the clusters for the categorical, functional and directional variables, and small separation between the clusters for the continuous variable

- when there is large separation between the clusters for the categorical and functional variables, and small separation between the clusters for the continuous and directional variables

- when there is large separation between the clusters for the functional and directional variables, and small separation between the clusters for the continuous and categorical variables

- when there is large separation between the clusters for the directional variable and small separation between the clusters for the continuous, categorical and functional variables

- when there is large separation between the clusters for the functional variable and small separation between the clusters for the continuous, categorical, and directional variables.

The average Rand Index using the extended Gower coefficient method was only slightly lower than the average Rand Index using the L_2 distance or Ackermann distance when (1) there is small separation between the clusters for the continuous variable only, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance being as large as 0.0401 and as small as 0.0216, (2) there is large separation between the clusters for the categorical, continuous and functional variables, with small separation between the clusters for the directional variable, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance being as large as 0.0948 and as small as 0.0743, (3) there is small separation between the clusters for the continuous, categorical, and functional variables, the difference between the average Rand indices for the extended Gower coefficient method and the Ackermann distance being as large as 0.1016 and as small as 0.0718, and (4) there is large separation between the clusters for the categorical and functional variables, and small separation between the clusters for the continuous and directional variables, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance being as large as 0.125 and as small as 0.0858. The average Rand Index using the extended Gower coefficient method was moderately lower when there is large separation between the clusters for the functional and directional variables, and small separation between the clusters for the continuous and categorical variables, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance was at most 0.1533 and at least 0.1251. The average Rand Index using the extended Gower coefficient method was beaten by a fairly large margin when (1) there is large separation between the clusters for the continuous and functional variables, and small separation between the clusters for the categorical and directional variables, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance being as large as 0.2393 and as small as 0.211, and (2) there is large separation between the clusters for the functional variable and small separation between the clusters for the continuous, categorical, and directional variables, with the difference between the average Rand indices for the extended Gower coefficient method and the L_2 distance being as large as 0.2669 and as small as 0.2328.

As one can see, when there is large separation between the clusters from the functional variable, it tends to dominate the clustering, making the L_2 distance clustering produce a larger average Rand Index value. Also, the average Rand Index for the extended Gower coefficient is smallest when the cluster sizes are 33, 33, 33 and 1, with the average Rand Index being higher when the cluster sizes for each cluster are the same and when the cluster sizes are 10, 20, 30, and 40. In some settings, the average Rand Index for the extended Gower coefficient with equal cluster sizes is minutely higher than the average Rand Index for the extended Gower coefficient with cluster sizes being 10, 20, 30, and 40. In other settings, the average Rand Index for the extended Gower coefficient with cluster sizes being 10, 20, 30, and 40 is minutely higher than the average Rand Index for the extended Gower coefficient with equal cluster sizes.

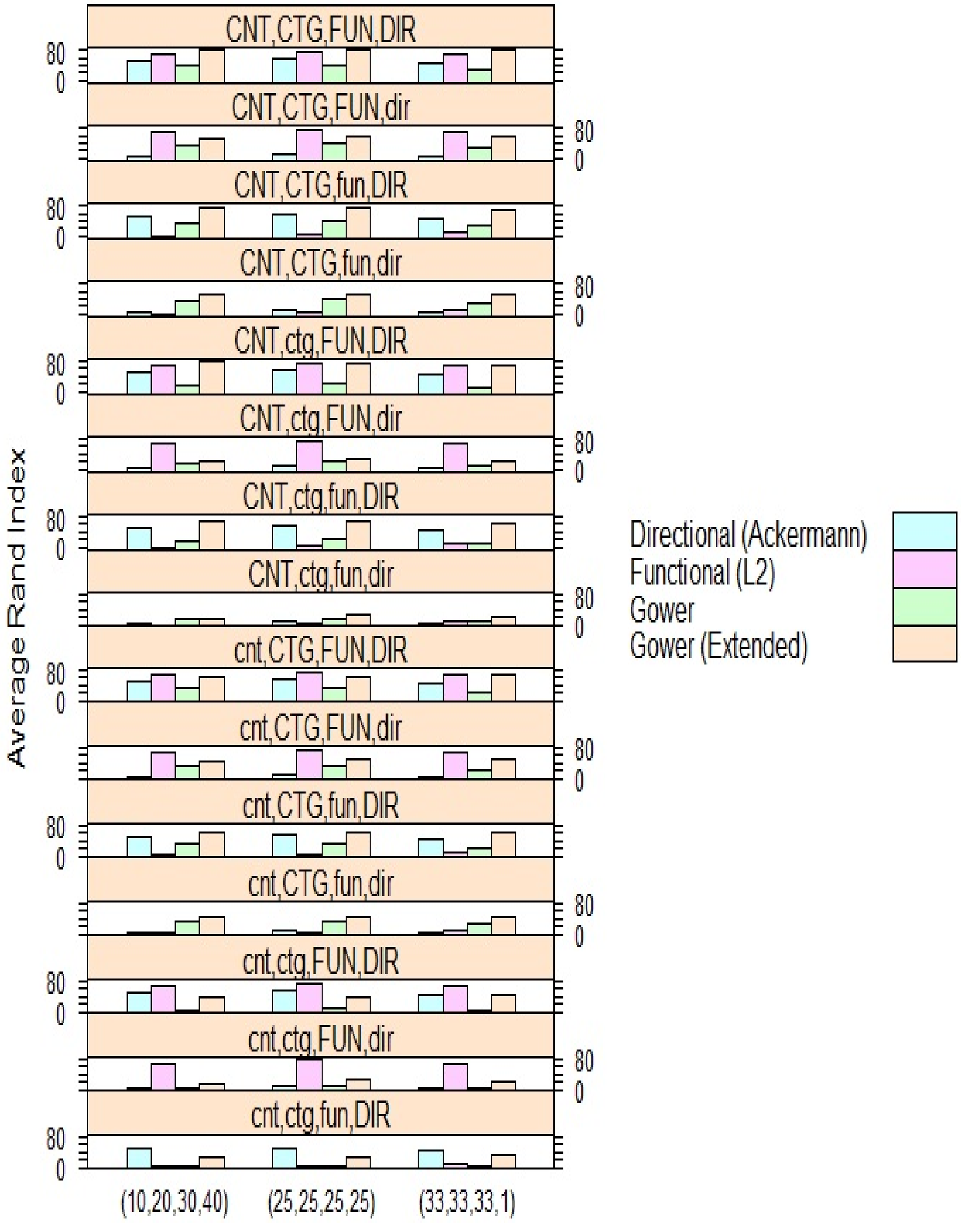

Figure 4.5: Comparisons with the Extended Gower coefficient

Chapter 5

Discussion/ Future Research

We considered Everitt's (1988) approach to model-based clustering of mixed data, for which the main issue is estimating the parameters for the density $h(\cdot)$, in order to be able to calculate the probabilities which are used to identify the clusters. We first approached this problem with the simulated annealing method; using this approach alone led our parameter estimates to be drawn to extreme values for each parameter. We found that using a penalized log-likelihood within the simulated annealing deters the parameters from being drawn to those extremes. To see how well our penalized log likelihood approach estimated the parameters, we initially compared our penalized $\log L$ to a penalized $\log L$ using Everitt's estimates (1988), and then, for a more fair comparison, compared our penalized $\log L$ to a penalized $\log L$ obtained using an estimation approach that mirrored Everitt's approach (1988). As all of these analyses were done using simulated data, we next applied our method to a real data set. For our real data set, we had an idea (albeit one that was far from certain) of the true clustering, which was informed by the expert diagnosis. This standard was used to determine how accurately our method performed.

We then presented a method of clustering based on an extended Gower coefficient. We first used this method on a real data set, the buoy data, and the method seemed to perform well, as the clustering solution classified the buoys into regions relatively close geographically to their "true" regions. We performed a simulation study to see how effective the method using the extended Gower coefficient is, in terms of being able to correctly assign the objects into the correct clusters, under different settings.

We wanted to see high values of the extended Gower coefficient between observations that are in truly separate clusters, and low values between observations that are in the same cluster. To compare the extended Gower coefficient to methods that allow only one or two types of variables to determine the clustering, we computed an average Rand Index for each method. We found that the extended Gower coefficient performed reasonably well in all of the settings, although the adjusted Rand Index for other methods was higher in several cases. In particular, when there was a large separation between the clusters for the functional data, with the L_2 distance yielding the largest adjusted Rand Index, and when there was a large separation between the clusters for the directional data, with Ackermann's distance yielding the largest adjusted Rand Index.

In our current work, we have been using small step sizes (the δ values) in the simulated annealing loop, with a large number of iterations. In the future, there could be work done to fine tune the appropriate step size(s) in the simulated annealing loop. One of the drawbacks in using the simulated annealing method is that it requires a significant amount of time to run, especially with multivariate data. A future direction in research would be to develop methods, whether via statistical or programming techniques, to enable the algorithm to run more efficiently.

Another option for future research is to simulate data having different types of signal functions. As mentioned by Ferreira and Hitchcock (2009), the signal functions we used for our simulation study (described in Section 4.2) represented "some form of periodic data." Other types of functional data we could look at include strictly decreasing functions, functions that peak and then decrease, and a mix of signal functions that had one or more different types of functions being used.

Bibliography

[1] H. Ackermann. A note on circular nonparametrical classification. *Biometrical Journal*, 5:577–587, 1997.

[2] Claudio Agostinelli and Ulric Lund. *circular: Circular Statistics*, 2011. R package version 0.4-3.

[3] Jeffrey D. Banfield and Adrian E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3):803–821, 1993.

[4] Kim Jong-Min Chae, Seong San and Wan Youn Yang. Cluster analysis with balancing weight on mixed-type data. *The Korean Communications in Statistics*, 13(3):719–732, 2006.

[5] Huaihou Chen, Philip T. Reiss, and Thaddeus Tarpey. Optimally weighted L2 distance for functional data. *Biometrics*, 2014. (in press).

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B*, 39(1):1–38, 1977. With discussion.

[7] B. S. Everitt. A finite mixture model for the clustering of mixed-mode data. *Statist. Probab. Lett.*, 6(5):305–309, 1988.

[8] Brian S. Everitt, Sabine Landau, Morven Leese, and Daniel Stahl. *Cluster Analysis*, volume 848 of *Wiley series in probability and statistics*. John Wiley & Sons, 2011.

[9] Manuel Febrero-Bande and Manuel Oviedo de la Fuente. Statistical computing in functional data analysis: The r package `fda.usc`. *Journal of Statistical Software*, 51(4), 2012.

[10] Laura Ferreira and David B. Hitchcock. A comparison of hierarchical methods for clustering functional data. *Communications in Statistics- Simulation and Computation*, 38(9):1925–1949, 2009.

[11] Chris Fraley and Adrian E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *J. Amer. Statist. Assoc.*, 97(458):611–631, 2002.

[12] Jerome H. Friedman and Jacqueline J. Meulman. Clustering objects on subsets of attributes. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 66(4):815–849, 2004. With discussion and a reply by the authors.

[13] J.C. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, 27:857–871, 1971.

[14] Peter J. Green. Penalized likelihood. In *In Encyclopedia of Statistical Sciences, Update Volume 2*, pages 578–586. John Wiley & Sons, 1996.

[15] Zengyou He, Xiaofei Xu, and Shengchun Deng. Clustering mixed numeric and categorical data: A cluster ensemble approach. *CoRR*, abs/cs/0509011, 2005.

[16] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[17] Lynette Hunt and Murray Jorgenson. Mixture model clustering using the multimix program. *Australia & New Zealand Journal of Statistics*, 41(2):153–171, 1999.

[18] Manabu Ichino and Hiroyuki Yaguchi. Generalized Minkowski metrics for mixed feature-type data analysis. *IEEE Trans. Systems Man Cybernet.*, 24(4):698–708, 1994.

[19] S.R. Jammalamadaka and A. Sengupta. *Topics in Circular Statistics*. Series on multivariate analysis. World Scientific, 2001.

[20] Murray Jorgenson and Lynette Hunt. Mixture model clustering of data sets with categorical and continuous variables. In D.L. Dowe, K.B. Korb, and J.J. Oliver, editors, *Proceedings of the conference, Information, Statistics and Induction in Science*, pages 375–384. World Scientific, 1996.

[21] Leonard Kaufman and Peter J. Rousseeuw. *Finding groups in data*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. John Wiley & Sons Inc., New York, 1990. An introduction to cluster analysis, A Wiley-Interscience Publication.

[22] W.J. Krzanowski. The location model for mixtures of categorical and continuous variables. *Journal of Classification*, 10:25–49, 1993.

[23] C.J. Lawrence and W.J. Krzanowski. Mixture separation for mixed-mode data. *Statistics and Computing*, 6:85–92, 1996.

[24] E.L. Lehmann and G. Casella. *Theory of Point Estimation*. Springer Texts in Statistics. Springer, 1998.

[25] Geoffrey J. McLachlan and Kaye E. Basford. *Mixture models*, volume 84 of *Statistics: Textbooks and Monographs*. Marcel Dekker Inc., New York, 1988. Inference and applications to clustering.

[26] Irini Moustaki. A latent trait and a latent class model for mixed observed variables. *The British Journal of Mathematical and Statistical Psychology*, 49:313–334, 1996.

[27] Irini Moustaki and Ioulia Papageorgiou. Latent class models for mixed variables with applications in archaeometry. *Comput. Statist. Data Anal.*, 48(3):659–675, 2005.

[28] John A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–317, 1965.

[29] W.M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846–850, 1971.

[30] Christian P. Robert and George Casella. *Monte Carlo statistical methods*. Springer Texts in Statistics. Springer-Verlag, New York, 1999.

[31] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer, 2004. 2nd Edition.

[32] A.J. Scott and M.J. Symons. Clustering methods based on likelihood ratio criteria. *Biometrics*, 27:387–397, 1971.

[33] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Is a correction for chance necessary? In *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, June 2009.

[34] J. Whitaker. *Graphical models in Applied Multivariate Statistics.* Wiley, New York, 1990.

[35] A.R. Willse and Robert J. Boik. Identifiable finite mixtures of location models for clustering mixed-mode data. *Statistics and Computing*, 9:111–121, 1999.

Appendix A

Average Rand Index along with Standard Errors

Table A.1: Setting 1: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.986(0.0005)	0.815(0.001)	0.894(0.001)	0.979(0.001)
$(33, 33, 33, 1)$	0.983(0.0006)	0.782(0.002)	0.855(0.002)	0.963(0.0007)
$(10, 20, 30, 40)$	0.987(0.0006)	0.802(0.002)	0.879(0.002)	0.970(0.002)

Table A.2: Setting 2: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.883(0.002)	0.810(0.001)	0.618(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.889(0.002)	0.782(0.002)	0.586(0.0009)	0.963(0.0007)
$(10, 20, 30, 40)$	0.876(0.002)	0.802(0.002)	0.601(0.001)	0.970(0.002)

Table A.3: Setting 3: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.979(0.0009)	0.732(0.001)	0.894(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.971(0.001)	0.618(0.002)	0.855(0.002)	0.963(0.0007)
$(10, 20, 30, 40)$	0.982(0.001)	0.690(0.001)	0.879(0.002)	0.970(0.002)

Table A.4: Setting 4: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.938(0.001)	0.799(0.001)	0.894(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.942(0.001)	0.748(0.002)	0.855(0.002)	0.963(0.0007)
$(10, 20, 30, 40)$	0.937(0.002)	0.791(0.002)	0.879(0.002)	0.970(0.002)

Table A.5: Setting 5: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.974(0.0009)	0.810(0.001)	0.894(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.971(0.0008)	0.782(0.002)	0.855(0.002)	0.620(0.001)
$(10, 20, 30, 40)$	0.977(0.0009)	0.802(0.002)	0.879(0.002)	0.579(0.001)

Table A.6: Setting 6: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.767(0.0008)	0.732(0.001)	0.619(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.736(0.0007)	0.618(0.002)	0.586(0.0009)	0.963(0.0007)
$(10, 20, 30, 40)$	0.731(0.0001)	0.690(0.001)	0.601(0.001)	0.970(0.002)

Table A.7: Setting 7: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.863(0.002)	0.799(0.001)	0.619(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.877(0.002)	0.748(0.002)	0.586(0.0009)	0.963(0.0007)
$(10, 20, 30, 40)$	0.845(0.002)	0.791(0.002)	0.601(0.001)	0.970(0.002)

Table A.8: Setting 8: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.944(0.002)	0.732(0.001)	0.894(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.940(0.002)	0.618(0.002)	0.855(0.002)	0.620(0.001)
$(10, 20, 30, 40)$	0.944(0.002)	0.689(0.001)	0.879(0.002)	0.579(0.001)

Table A.9: Setting 9: Average Rand Index (Standard Error)

Cluster	Extended	Gower	Directional	Functional
Allocation	Gower			
$(25, 25, 25, 25)$	0.875(0.002)	0.810(0.001)	0.619(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.879(0.002)	0.782(0.002)	0.586(0.0009)	0.620(0.001)
$(10, 20, 30, 40)$	0.865(0.002)	0.802(0.002)	0.601(0.001)	0.579(0.001)

Table A.10: Setting 10: Average Rand Index (Standard Error)

Cluster	Extended	Gower	Directional	Functional
Allocation	Gower			
$(25, 25, 25, 25)$	0.824(0001)	0.620(0.0005)	0.894(0.001)	0.978(0.001)
$(33, 33, 33, 1)$	0.838(0.002)	0.582(0.0003)	0.855(0.002)	0.963(0.0007)
$(10, 20, 30, 40)$	0.824(0.002)	0.598(0.0006)	0.879(0.002)	0.970(0.002)

Table A.11: Setting 11: Average Rand Index (Standard Error)

Cluster	Extended	Gower	Directional	Functional
Allocation	Gower			
$(25, 25, 25, 25)$	0.920(0.001)	0.799(0.001)	0.894(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.922(0.001)	0.748(0.002)	0.855(0.002)	0.620(0.001)
$(10, 20, 30, 40)$	0.918(0.002)	0.791(0.002)	0.879(0.002)	0.579(0.001)

Table A.12: Setting 12: Average Rand Index (Standard Error)

Cluster	Extended	Gower	Directional	Functional
Allocation	Gower			
$(25, 25, 25, 25)$	0.752(0.0006)	0.620(0.0005)	0.619(0.001)	0.985(0.0009)
$(33, 33, 33, 1)$	0.730(0.0008)	0.582(0.0003)	0.586(0.0009)	0.963(0.0007)
$(10, 20, 30, 40)$	0.703(0.0009)	0.598(0.0006)	0.601(0.001)	0.970(0.002)

Table A.13: Setting 13: Average Rand Index (Standard Error)

Cluster	Extended	Gower	Directional	Functional
Allocation	Gower			
$(25, 25, 25, 25)$	0.760(0.0007)	0.623(0.0006)	0.619(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.731(0.0008)	0.618(0.002)	0.586(0.0009)	0.620(0.001)
$(10, 20, 30, 40)$	0.729(0.001)	0.690(0.001)	0.601(0.001)	0.579(0.001)

Table A.14: Setting 14: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.857(0.002)	0.800(0.001)	0.619(0.001)	0.616(0.001)
$(33, 33, 33, 1)$	0.850(0.002)	0.749(0.002)	0.586(0.0009)	0.620(0.001)
$(10, 20, 30, 40)$	0.839(0.002)	0.791(0.002)	0.601(0.001)	0.579(0.001)

Table A.15: Setting 15: Average Rand Index (Standard Error)

Cluster Allocation	Extended Gower	Gower	Directional	Functional
$(25, 25, 25, 25)$	0.777(0.002)	0.598(0.0006)	0.879(0.002)	0.579(0.001)
$(33, 33, 33, 1)$	0.783(0.002)	0.582(0.0003)	0.855(0.002)	0.620(0.001)
$(10, 20, 30, 40)$	0.777(0.002)	0.598(0.0006)	0.879(0.002)	0.579(0.001)

Appendix B

Further Details about Simulation Study

To simulate a categorical variable (like time zone), we used the R function `sample.int`. The arguments for this function are `n`, which specifies the number of categories to choose from, `size`, which specifies the number of items to choose, `replace`, which specifies whether sampling is done with or without replacement and `prob`, which is a vector of probabilities for obtaining the observations. For our simulations, we set `n` at 5 (implying five categories), and we let `size` differ. For our simulations, `replace` was set to `TRUE`, and for each cluster we tried different probability vectors for `prob`, in order to simulate clusters of data that have different probabilities of coming from each category.

To simulate a directional variable, θ, we used the von Mises distribution. To simulate from this distribution in R, we used the function `rvonmises` in the `circular` package (Agostinelli and Lund, 2011). The arguments for this function are `n`, the total number of observations, `mu`, mean direction of the distribution, `kappa`, the concentration parameter of the distribution, and `control.circular`, which allows one to specify whether the units should be degrees or radians (Agostinelli and Lund, 2011). To simulate data for 4 different clusters, we considered the following values for `mu` and `kappa`: for cluster 1, `mu` was 0 and `kappa` was 50, so that the data were highly concentrated around 0, for cluster 2, `mu` was $0 + k$ and `kappa` was 50, for cluster 3, `mu` was $0 + 2k$ and `kappa` was 50, for cluster 3, `mu` was $0 + 3k$ and `kappa` was 50, and for cluster 4, `mu` was $0 + 4k$ and `kappa` was 50, where k varied from small to moderate to large; for example, $k = 0.5$, $k = 1.0$, and $k = 2.5$. We also changed `kappa` in other settings so that the data were not so highly concentrated around their respective means.