Longitudinal Data Clustering Via Kernel Mixture Models

LONGITUDINAL DATA CLUSTERING VIA KERNEL MIXTURE MODELS

BY

XI ZHANG, M.Ec.

A THESIS

SUBMITTED TO THE DEPARTMENT OF MATHEMATICS & STATISTICS

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

Master of Science (2020) McMaster University
(Statistics) Hamilton, Ontario, Canada

TITLE:	Longitudinal Data Clustering Via Kernel Mixture Models
AUTHOR:	Xi Zhang M.Ec. (Statistics), Shanxi University of Finance and Economics, TaiYuan, China
SUPERVISOR:	Dr. Paul D. McNicholas
NUMBER OF PAGES:	x, 54

To my family and friends. Thanks for your support.

Abstract

Kernel mixture models are proposed to cluster univariate, independent multivariate and dependent bivariate longitudinal data. The Gaussian distribution in finite mixture models is replaced by the Gaussian and gamma kernel functions, and the expectation-maximization algorithm is used to estimate bandwidths and compute log-likelihood scores. For dependent bivariate longitudinal data, the bivariate Gaussian copula is used to reveal the correlation between two attributes. After that, we use AIC, BIC and ICL to select the best model. In addition, we also introduce a kernel distance-based clustering method to compare with the kernel mixture models. A simulation is performed to illustrate the performance of this mixture model, and results show that the gamma kernel mixture model performs better than the kernel distance-based clustering method based on misclassification rates. Finally, these two models are applied to COVID-19 data, and sixty countries are classified into ten clusters based on growth rates and death rates.

Acknowledgements

Foremost, I would like to express my deepest gratitude to my supervisor, Dr. Paul D. McNicholas, for giving me an opportunity to do research at McMaster University. He also provided a lot of guidance and support throughout my research.

At the same time, I would like to thank Dr. Orla Murphy for her patience and encouragement. Without their help, my thesis would not have gone so smoothly.

Besides my supervisor, my sincere thanks go to the rest of my examination committee, Dr. Noah Forman and Dr. Sharon McNicholas, for their insightful comments and questions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Longitudinal data, also called panel data, are a type of data which are collected by observing a set of variables over time based on multiple subjects. This kind of data exists widely in various fields, such as economics, social science, finance and medicine. Compared with cross-sectional data, panel data record the changes in variables over time and add the random effect into the model, so panel data can reflect the correlation within time series. Compared with time series, panel data include more than one variable. In practice, it is difficult to cluster subjects based on one variable. Hence, panel data can provide more useful information for clustering and classification. Furthermore, homogeneity is required in current panel data models to find common effects among individuals, especially in the linear regression model. If we cluster longitudinal data first and then construct a panel data model, this model will be more accurate than models based on the original data set. Thus, a number of panel data clustering methods have been proposed, and these methods are mainly divided into two categories: distance-based clustering and model-based clustering (see Sections 2.1 and 2.2).

Overall, model-based clustering considers the autocorrelations within time series, but distance-based clustering regards observations in time as T independent samples and redefines distance matrix. Intuitively, model-based clustering appears to be more appropriate than distance-based clustering method. Another benefit of the model-based clustering method is that randomization is considered in the model, such as residuals and prior distribution. In addition, model-based cluster-

ing can account for dependence between attributes via dynamic models. Hence, in some cases, clustering results from model-based methods is more meaningful than results from distance-based methods.

Some basic concepts and methods, including distance-based and model-based clustering, are introduced in Chapter 2. In Chapter 3, we will construct a kernel mixture model and a distance-based clustering model based on Fourier transforms. In Chapter 4, we will perform a simulation for these two methods and make a comparison between them. Three cases are included in this simulation: Univariate longitudinal data, independent longitudinal data and dependent longitudinal data. Next, these two methods will be applied to COVID-19 data in Chapter 5. Finally, Chapter 6 contains concluding statements and a discussion of future work.

Chapter 2

Background

2.1 Distance-Based Clustering

Distance-based clustering can be used for balanced or unbalanced panel data. The most important step in this clustering method is to find a suitable distance function to measure dissimilarity between two subjects, and then a general clustering algorithm, such as relocation clustering, hierarchical clustering, K-means and fuzzy c-means, will be applied to these distance measurements. Common choices of distance functions include Euclidean distance (Golay et al., 1998; Košmelj and Batagelj, 1990; Policker and Geva, 2000), root mean square distance (Van Wijk and Van Selow, 1999), Minkowski distance (Genolini et al., 2015) and cross-correlation-based-distance (Golay et al., 1998). In addition, Nie et al. (2010) adjusted weights based on the time sequence of observations when computing Minkowski distance.

These distance functions do not consider the correlation within each time series and are easily affected by noise (Ferreira and Zhao, 2015), so some researchers proposed new distance functions to measure the dissimilarity between time series. Short time series (STS) distance introduced by Möller-Levet et al. (2003) reflects the differences of time series slopes between two subjects. Dynamic time warping (DTW) proposed by Berndt and Clifford (1994) finds a warping path with the minimum cumulative distance. Patterns from two time series can be detected even if there is a time lag between these observations, and this distance function is also suitable for unevenly spaced time series. Batista et al. (2014) added complexity

factors to correct distance measures (complexity-invariant distance), such as Euclidean distance, and results show that this method could improve the accuracy of classification and clustering. Ferreira and Zhao (2015) used community detection methods for time series clustering based on the above distance functions and verified that DTW works better than other functions.

After choosing a suitable distance function to measure the dissimilarity between subjects, we need an algorithm to cluster data. The most common algorithm is the K-means algorithm (Lloyd, 1982), and squared Euclidean distance is typically chosen to measure the dissimilarity between observations and arrive at the mean value within each cluster. This algorithm tries to partition subjects into K clusters by minimizing the average dissimilarity. The processes of this algorithm are: firstly, we need to determine the number of clusters and then select K data points as cluster centers. Next, we compute squared Euclidean distance between observations and every cluster center and assign observations to the closest cluster. After that, we compute the averages of each cluster and these mean values will be the new cluster centers. Finally, we repeat above steps until cluster centers do not change. Because we take the average of observations as the cluster centers, this algorithm is easily affected by outliers. For this reason, the K-medoids algorithm is a popular alternative and the most well-known algorithm for K-medoids, called the partitioning around medoids (PAM), has been proposed by Kaufmann and Rousseeuw (1987). In the PAM algorithm, cluster centers must be data points and the squared Euclidean distance is replaced by the pairwise distance. The iterative process is analogous to K-means clustering.

2.2 Model-Based Clustering

Model-based clustering assumes that the data are generated by a mixture of distributions. In essence, each data point ends up assigned to the component whose distribution is most likely to have generated it — McNicholas (2016b) provides a review of model-based clustering work and extensive details are given by McNicholas (2016a). To cluster longitudinal data, the components can be distri-

butions or dynamic models. Some researchers consider that observations should be in the same cluster when coefficients or errors from the dynamic model are similar, so they first measure the dissimilarity of coefficients or errors and then cluster observations. Other researchers assume the distributions of coefficients or residuals in the dynamic model, and then use a finite mixture model to cluster data. Thus, compared with distance-based clustering, model-based clustering can consider correlations within time series by constructing autoregressive or dynamic panel models.

Autoregressive (AR), autoregressive-moving-average (ARMA), autoregressive-integrated-moving average (ARIMA) and generalized autoregressive conditional heteroscedastic (GARCH) models are usually chosen to fit longitudinal data. Piccolo (1990) used the Euclidean distance matrix of coefficients from the AR model to measure the dissimilarity of two subjects. Baragona (2001) computed the cross-correlations of errors based on the ARMA model, and then made a comparison between three clustering algorithms. Maharaj (2000) constructed a chi-squared statistic to test whether coefficients from the AR model is different or not. Kalpakis et al. (2001) applied the partition algorithm to time series clustering based on the Euclidean distance of coefficients of the ARIMA model. More details on this kind of approach can be found in the review paper by Liao (2005).

When a finite mixture model is used for clustering, we assume distribution functions for coefficients or residuals in the dynamic model based on experience. Frühwirth-Schnatter and Kaufmann (2008) and Juárez and Steel (2010) assigned probability distributions for coefficients and residuals from a first-order autoregression model, and use a Bayesian estimation method to get class-specific parameters and a posterior probability of cluster membership. McNicholas and Murphy (2010) applied the modified Cholesky decomposition to the Gaussian mixture model and give eight different models based on restrictions to covariance structure, and this model is fitted for univariate longitudinal data. When computing the joint density of time series, Frühwirth-Schnatter and Kaufmann (2008) assumes the lagged rank is one, but McNicholas and Murphy (2010) considers all samples before T time points. In addition, Vermunt (2010) assumed that observations over time are inde-

pendent, and applied the locally independent clustering kernel function to mixture model. The application of these approaches were reviewed by Frühwirth-Schnatter (2011).

In recent years, some new machine learning methods are applied for panel data clustering. Falissard et al. (2018) applied artificial neural network to longitudinal clustering. The first step of fuzzy random effect estimation tree (Xu et al., 2020, FREETree) used weighted correlation network analysis to cluster features in panel data.

2.3 Fourier Transform

Baron Fourier (1878) showed that some wavelet form functions can be rewritten as the sum of sine and cosine functions, but these series can only be used to analyze the periodic phenomena. Hence, the Fourier transform instead of the Fourier series was introduced to analyze nonperiodic phenomena, and the frequency becomes a continuous variable. However, real-world samples in time domain are discrete not continuous, so a discrete Fourier transform (DFT) was proposed by Cooley and Tukey (1965).

Suppose that there are N observations and T time points. Let $\boldsymbol{x}$ denote observations x_{nt}, where $n = 1, \ldots, N$, $t = 1, \ldots, T$. The discrete Fourier transform can be written in the following form:

$$x_{nk} = \sum_{t=1}^{T} x_{nt} e^{-2\pi i t k/T} \tag{2.1}$$

for $k = 0, 1, \ldots, T-1$, where x_{nk} is a complex number at frequency k and x_{nt} is a real number observed at time point t.

Computing the DFT directly based on (2.1) is time-consuming, so the fast Fourier transform algorithm (FFT) was introduced where the number of operations is reduced from $\mathscr{O}(T^2)$ to $\mathscr{O}(T \log_2(T))$. We will use two important properties for our analysis. The first property is $x_{nk} = x_{n(k+T/2)}$, i.e.,

$$x_{n(k+T/2)} = \sum_{t=1}^{T} x_{nt} e^{-2\pi i t(k+T/2)/T} = \sum_{t=1}^{T} x_{nt} e^{-2\pi i t k/T} e^{-\pi i t} = x_{nk},$$

where $e^{-\pi it} = 1$. The spectral value at frequency k is the modulus of x_{nk}. Based on the first property, we know that the spectral plot is a symmetric plot in one period. To save computing time, spectral values between frequency $T/2$ and T will be discarded in my analysis.

The second property is that Fourier coefficients are independent of each other. Let

$$e_k(t) = e^{-2\pi i k t/T}$$

be the Fourier transform basis. The inner product of two Fourier coefficient functions is

$$\begin{aligned}
\langle x_{nk_1}, x_{nk_2} \rangle &= \left\langle \sum_{t=1}^{T} x_{nt} e_{k_1}(t), \sum_{s=1}^{T} x_{ns} e_{k_2}(s) \right\rangle \\
&= \sum_{t=1}^{T} \sum_{s=1}^{T} \langle x_{nt} e_{k_1}(t), x_{ns} e_{k_2}(s) \rangle \\
&= \sum_{t=1}^{T} \sum_{s=1}^{T} x_{nt} x_{ns} \int_0^T e^{-2\pi i k_1 t/T} e^{2\pi i k_2 t/T} dt \\
&= \sum_{t=1}^{T} \sum_{s=1}^{T} x_{nt} x_{ns} \frac{T}{2\pi i (k_2 - k_1)} (e^{2\pi i (k_2 - k_1)} - 1) = 0. \qquad (2.2)
\end{aligned}$$

Equation (2.2) shows that Fourier coefficients are orthogonal to each other, so they are independent in the frequency domain.

2.4 Kernel Density Estimation

As the spectrum value is a non-negative value, the Gaussian and gamma distribution are both used as kernel functions to cluster panel data. After the Fourier transform, the spectral data is denoted by $\boldsymbol{x} = \{x_{nk}\}$, where x_{nk} represents the spectral value of subject n at frequency k. The Gaussian and gamma kernel functions (Chen, 2000) of x_{nk} in cluster g can be written as below. Let b_g be bandwidth of the cluster g, $x_{ik,g}$ be the spectral value of subject i from cluster g at frequency

k, and n_g be the total number of subjects in cluster g.

$$\hat{f}_{\text{Gamma}}(x_{nk}|b_g) = \frac{1}{n_g}\sum_{i=1}^{n_g} \frac{x_{ik,g}^{\frac{x_{nk}}{b_g}} \exp\left\{-\frac{x_{ik,g}}{b_g}\right\}}{b_g^{\frac{x_{nk}}{b_g}+1}\Gamma(\frac{x_{nk}}{b_g}+1)},$$

$$\hat{f}_{\text{Gaussian}}(x_{nk}|b_g) = \frac{1}{n_g}\sum_{i=1}^{n_g} \frac{1}{\sqrt{(2\pi)}b_g} \exp\left\{\frac{-(x_{nk}-x_{ik,g})^2}{2b_g^2}\right\}.$$

The first property indicates that $T/2$ spectral values are enough to cluster data due to the symmetry of the spectral values. The second property of the Fourier transform shows that the spectral values under different frequencies are independent so we can get a joint distribution by multiplying marginal distributions together. Assume that the bandwidth under each frequency is different. The joint density functions are:

$$\hat{f}_{\text{Gamma}}(\boldsymbol{x}_n|\boldsymbol{B}_g) = \prod_{k=1}^{T/2}\left[\frac{1}{n_g}\sum_{i=1}^{n_g} \frac{x_{ik,g}^{\frac{x_{nk}}{b_{k,g}}} \exp\left\{-\frac{x_{ik,g}}{b_{k,g}}\right\}}{b_{k,g}^{\frac{x_{nk}}{b_{k,g}}+1}\Gamma(\frac{x_{nk}}{b_{k,g}}+1)}\right],$$

$$\hat{f}_{\text{Gaussian}}(\boldsymbol{x}_n|\boldsymbol{B}_g) = \prod_{k=1}^{T/2}\left[\frac{1}{n_g}\sum_{i=1}^{n_g} \frac{1}{\sqrt{(2\pi)}b_{k,g}} \exp\left\{\frac{-(x_{nk}-x_{ik,g})^2}{2b_{k,g}^2}\right\}\right].$$

2.5 Finite Mixture Model

2.5.1 Mixture Models and Model Fitting

Suppose that there are G clusters and N K-dimensional observations. A finite mixture model is a distribution function with G components, and each component has a density function with parameters $\boldsymbol{\theta}_g$. If component density functions are the Gaussian distribution, this finite mixture model is the Gaussian mixture model. Assume that each observation belongs to one of the components in the mixture model. However, the components are unknown, so we introduce latent variables Z_n into the finite mixture model to indicate the group that the observation belongs to. The random variable $\boldsymbol{X}_n$ arises from a finite mixture model if, for all $\boldsymbol{x}_n \subset \boldsymbol{X}_n$,

its density can be written

$$f(\boldsymbol{x}_n|\boldsymbol{\Theta}) = \sum_{g=1}^{G} p(Z_n = g|\boldsymbol{\phi}) f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n),$$

where $f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)$ is the component density function of the gth cluster, G component density functions always have the same distribution with different parameters, Z_n is a latent random variable used to assign clusters for $\boldsymbol{x}_n$, $p(Z_n = g|\boldsymbol{\phi})$ is the prior probability that $\boldsymbol{x}_n$ comes from the gth cluster, also known as the gth mixing proportion, and $\boldsymbol{\Theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_G, \boldsymbol{\phi})$.

In mixture models, the latent variable Z_n takes integer values from 1 to G. Generally, Z_n has a multinomial distribution with parameters $\boldsymbol{\phi}$. In my thesis, $Z_n \sim$ Multinomial($\boldsymbol{\pi}$), where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_G)$, and π_g is the probability that observation n belongs to the gth cluster. These parameters will be updated in EM algorithm. As N subjects are observed in the data set, the number of latent random variables are N and they are independent and identically distributed according to a multinomial distribution.

There exists two primary approaches to estimate parameters in a mixture model. They are the expectation-maximization (EM) algorithm (Dempster et al., 1977) and the Markov Chain Monte Carlo (MCMC) method (Diebolt and Robert, 1994; Frühwirth-Schnatter and Kaufmann, 2008; Juárez and Steel, 2010). The EM algorithm introduces latent variables to the log-likelihood function and maximizes the log-likelihood function to estimate parameters. In contrast, the MCMC method uses the Gibbs sampler to simulate the posterior distribution.

Comparisons have been made between the EM algorithm and the MCMC method (De la Cruz-Mesía et al., 2008; Lin, 2010; Dias and Wedel, 2004; Rydén et al., 2008). Rydén et al. (2008) proposed that the computational cost of Gibbs samplers is higher than the EM, so the EM algorithm is the simplest and quickest method for point estimation. However, one disadvantage of the EM algorithm is that it is hard to guarantee final result convergence to the global maximum log-likelihood value and so it is necessary to give multiple initial values and choose estimators with the maximum log-likelihood as the final result. In contrast, results

of the empirical study from Lin (2010) show that there is no significant difference between these methods. Thus, the EM algorithm will be used to estimate parameters in this thesis.

The EM algorithm contains two steps. The first step is called the expectation step. In this step, a latent variable is introduced to the mixture model and a lower bound on the log-likelihood function will be constructed through Jensen's inequality. In the finite mixture model, the lower bound on the log-likelihood function is shown below. The last step used Jensen's inequality, and $[p(Z_n = g|\boldsymbol{\phi})f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, z_n)/p(Z_n = g|\boldsymbol{x}_n)]$ is a constant, so Jensen's inequality holds with equality.

$$\begin{aligned}
\ell(\boldsymbol{\Theta}; \boldsymbol{x}) &= \sum_{n=1}^{N} \log\left\{f(\boldsymbol{x}_n|\boldsymbol{\Theta})\right\} = \sum_{n=1}^{N} \log\left\{\sum_{g=1}^{G} p(Z_n = g|\boldsymbol{\phi}) f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)\right\} \\
&= \sum_{n=1}^{N} \log\left\{\sum_{g=1}^{G} p(Z_n = g|\boldsymbol{x}_n) \frac{p(Z_n = g|\boldsymbol{\phi}) f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)}{p(Z_n = g|\boldsymbol{x}_n)}\right\} \\
&\geq \sum_{n=1}^{N} \sum_{g=1}^{G} p(Z_n = g|\boldsymbol{x}_n) \left[\log\left\{p(Z_n = g|\boldsymbol{\phi})\right\} + \log\left\{f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)\right\}\right. \\
&\qquad \left. - \log\left\{p(Z_n = g|\boldsymbol{x}_n)\right\}\right],
\end{aligned}$$

The second step is maximizing the log-likelihood function with respect to the parameters in the mixture models. These two steps are repeated until the convergence criterion is satisfied. Wu (1983) gave a proof of the convergence of the EM algorithm. The EM algorithm iterates between the E- and M-steps until satisfying a stopping criterion:

E-step: compute posterior probabilities as the predicted classifications:

$$p(Z_n = g|\boldsymbol{x}_n) = \frac{p(Z_n = g|\boldsymbol{\phi}) f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)}{\sum_{d=1}^{G} p(Z_n = d|\boldsymbol{\phi}) f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)}.$$

M-step: maximize the log-likelihood function based on the E-step

$$\max \sum_{n=1}^{N} \sum_{g=1}^{G} p(Z_n = g|\boldsymbol{x}_n) \left[\log \{p(Z_n = g|\boldsymbol{\phi})\} + \log \{f(\boldsymbol{x}_n|\boldsymbol{\theta}_g, Z_n)\} - \log \{p(Z_n = g|\boldsymbol{x}_n)\}\right].$$

2.5.2 Stopping Criterion

Due to the slow convergence of the EM algorithm, many stopping criteria have been proposed (reviewed in Karlis and Xekalaki (2003) and Abbi et al. (2008)). The most common approach is to give a threshold for the increase in the log-likelihood function. Böhning et al. (1994) developed a useful stopping criterion based on the Aitken acceleration. Let $l^{(i)}$ be the log-likelihood at iteration i. As this sequence converges to a constant ($\hat{l}$) linearly, the estimated log-likelihood value at iteration i is :

$$l_{\infty}^{(i)} = l^{(i-1)} + \frac{1}{1 - c^{(i)}}(l^{(i)} - l^{(i-1)}),$$

where

$$c^{(i)} = \frac{l^{(i+1)} - l^{(i)}}{l^{(i)} - l^{(i-1)}}.$$

If

$$|l_{\infty}^{(i)} - l_{\infty}^{(i-1)}| < \epsilon$$

(Böhning et al., 1994) or

$$l_{\infty}^{(i)} - l^{(i-1)} < \epsilon \tag{2.3}$$

(McNicholas et al., 2010), the iterations of the EM algorithm stop. The stopping rule in (2.3) will be used in this thesis.

2.5.3 Model Selection

In this thesis, model selection includes determining the optimal number of clusters, the type of kernel function, and whether to use a copula function. Information criteria are widely used in model-based clustering, and the three most common criteria are Akaike's information criterion (AIC; Akaike, 1998), the Bayesian in-

formation criterion (BIC; Schwarz, 1978) and the integrated completed likelihood (ICL) criterion (Biernacki et al., 2000).

$$\begin{aligned}
\text{AIC} &= -2\ell(\hat{\boldsymbol{\Theta}}; \boldsymbol{x}) + 2d, \\
\text{BIC} &= -2\ell(\hat{\boldsymbol{\Theta}}; \boldsymbol{x}) + d\log(N), \\
\text{ICL} &\approx \text{BIC} - 2\sum_{g=1}^{G}\sum_{i=1}^{N} p(z_i|\hat{\boldsymbol{\theta}}_{\boldsymbol{g}}, \boldsymbol{x}) \log p(z_i|\hat{\boldsymbol{\theta}}_{\boldsymbol{g}}, \boldsymbol{x}),
\end{aligned}$$

where d is the number of parameters in mixture model, N is the number of observations, $\hat{\boldsymbol{\Theta}}$ is the maximum likelihood estimate, and $\ell(\hat{\boldsymbol{\Theta}}; \boldsymbol{x})$ is the maximum log-likelihood value. Compared with the BIC, the ICL adds the entropy of posterior probabilities into the criterion. The model with the minimum AIC, BIC and ICL are selected as the best model.

2.5.4 Performance Assessment

The adjusted Rand index (ARI; Hubert and Arabie, 1985) and the misclassification rate are used to assess the performance of the models in the simulation study, and the ARI is used to compare the clustering results from the kernel mixture model and kernel distance-based clustering in the application. The ARI can measure the similarity between true classes and predicted classes or two clustering results. This measurement adjusts the Rand index (Rand, 1971) to consider randomness (i.e., random correct classification), and it can take negative values. When the ARI is close to zero, it means this clustering result is similar to the random clustering result. An ARI value of 1 indicates perfect class agreement.

2.6 Copula Functions

Copulas are multivariate cumulative distribution functions (CDF) which include multiple univariate uniform distributions and dependence parameters. Sklar's theorem (Sklar, 1959) states that the joint cumulative distribution function of a K-dimensional random vector can be written as the combination of its K marginal

distributions and a copula. This statement is summarized by Sklar's equation.

$$F(x_1, \ldots, x_K) = C\left(F_1(x_1; \boldsymbol{\theta_1}), \ldots, F_K(x_K, \boldsymbol{\theta_K})\right),$$

where $(x_1,\ldots,x_K)$ is a K-dimensional observation, $F_1,\ldots,F_K$ are CDFs with parameters $\boldsymbol{\theta}$, C is a K-copula and F is a K-dimensional joint density function. In addition, Sklar also proposes that if the K marginal distributions are continuous, then this copula function is unique.

Reviews of literature on copula families were conducted by Nelsen (2007) and Joe (1997). Di Lascio and Giannerini (2012) choose three copulas for clustering: the Gaussian copula from the elliptical family as well as Frank's and Clayton's copulas from the Archimedean family. Due to its computational simplicity, the Gaussian copula will be used in this thesis.

$$C(u_1, \ldots, u_K; \boldsymbol{\rho}) = \Phi_K\left\{\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_K); \boldsymbol{\rho}\right\},$$

where $u_1, \ldots, u_K$ are cumulative probabilities of K random variables, which are distributed in $[0, 1]$, Φ is the CDF of the standard univariate normal distribution, Φ_K is the CDF of the standard K-dimensional normal distribution, and $\boldsymbol{\rho}$ is a correlation matrix.

Genest and Favre (2007) proposed several rank-based methods to estimate the dependence parameter in the copula function, e.g., Kendall's tau, Spearman's rho, and the maximum pseudo-likelihood estimator (MPE). Compared with the MLE, the MPE uses an empirical distribution to replace the CDF. Let R_{ki} be the rank of x_{ki} within the variable. The pseudo-likelihood function is shown below:

$$\ell(R; \boldsymbol{x}) = \sum_{n=1}^{N} \log\left\{C\left(\frac{R_{1i}}{N+1}, ..., \frac{R_{Ki}}{N+1}; \boldsymbol{\rho}\right)\right\},$$

As the EM algorithm will be used to estimate parameters in the mixture model, maximizing the log-likelihood function is an essential step in this algorithm. However, maximizing the log-likelihood function with copulas is quite complicated, as mixing proportions exist in the log-likelihood function. Thus, a two-stage estima-

tion method is introduced to estimate parameters in the log-likelihood function.

The first step of the two-stage estimation method is to estimate the marginal parameters by using the marginal maximum likelihood estimators. The second step is to estimate the dependence parameters. Dependence parameter estimation includes parametric estimation and semiparametric estimation methods. The semiparametric method proposed by (Genest et al., 1995) maximizes the pseudolikelihood to estimate the dependence parameters, and it is called the pseudo maximum likelihood (PML). The parametric estimation proposed by Joe (2005) maximizes the multivariate log-likelihood function, and this method is called the inference function for margins (IFM).

Kim et al. (2007) made a comparison between these two methods. The results of simulation and application both show that the PML method is robust when marginal distributions are unknown, so we will choose the PML method to estimate the copula parameters.

2.7 Clustering functions in R

In this section, some clustering functions in R will be introduced. As the EM algorithm is easily stuck at the local optimum points, we need to choose different initial values. In this thesis, K-means and the Gaussian mixture model clustering results will be used as initial labels in the EM algorithm. The `kmeans` function is the base function in R, and the Gaussian mixture model is performed via the package `mclust`. In addition, the PAM algorithm will also be used to cluster longitudinal data based on the dissimilarity matrix in the kernel distance-based clustering method. The PAM algorithm is implemented by the function `PAM` in the `cluster` package.

Chapter 3

Methodology

3.1 Univariate Longitudinal Data Clustering

Suppose there are G clusters, and the length of the time series is T. Plugging the Gaussian kernel and gamma kernel into a finite mixture model, we get joint density functions of mixture models,

$$
\begin{aligned}
f_{\text{Gaussian}}(\boldsymbol{x}_n|\boldsymbol{B}) &= \sum_{g=1}^{G} p(Z_n = g)\hat{f}(\boldsymbol{x}_n|\boldsymbol{B}_g) \\
&= \sum_{g=1}^{G} \pi_g \left[\prod_{k=1}^{T/2} \left\{ \frac{1}{n_g} \sum_{i=1}^{n_g} \left(\frac{1}{\sqrt{2\pi} b_{k,g}} \exp\left(\frac{-(x_{nk} - x_{ik,g})^2}{2b_{k,g}^2} \right) \right) \right\} \right], \\
f_{\text{Gamma}}(\boldsymbol{x}_n|\boldsymbol{B}) &= \sum_{g=1}^{G} p(Z_n = g)\hat{f}(\boldsymbol{x}_n|\boldsymbol{B}_g) \\
&= \sum_{g=1}^{G} \pi_g \left[\prod_{k=1}^{T/2} \left\{ \frac{1}{n_g} \sum_{i=1}^{n_g} \frac{x_{ik,g}^{\frac{x_{nk}}{b_{k,g}}} \exp\left(-\frac{x_{ik,g}}{b_{k,g}} \right)}{b_{k,g}^{\frac{x_{nk}}{b_{k,g}}+1} \Gamma(\frac{x_{nk}}{b_{k,g}} + 1)} \right\} \right],
\end{aligned}
$$

where $Z_1, \ldots, Z_N$ are latent random variables used to assign clusters for every subject, and they are independent and identically distributed variables which have the multinomial distribution, π_g is the prior probability that subjects belong to the gth cluster, x_{nk} is the spectral value of subject n under the frequency k, and $b_{k,g}$ is the bandwidth of observations under the frequency k in cluster g.

Next, the EM algorithm is used to estimate parameters in the mixture models.

The E-step computes the posterior probability of the latent random variable based on the following formula:

$$p(Z_n = g|\boldsymbol{x}_n) = \frac{p(\boldsymbol{x}_n, Z_n = g)}{p(\boldsymbol{x}_n)} = \frac{\pi_g f(\boldsymbol{x}_n; \boldsymbol{B}_g)}{\sum_{g=1}^{G} \pi_g f(\boldsymbol{x}_n; \boldsymbol{B}_g)}. \tag{3.1}$$

The M-step maximizes the log-likelihood function with respect to all parameters in the mixture model. The lower bound of the gamma log-likelihood function is

$$\begin{aligned}\ell_{\text{Gamma}}(\boldsymbol{B}; \boldsymbol{x}) &= \sum_{n=1}^{N} \log \left[\sum_{g=1}^{G} \pi_g f(\boldsymbol{x}_n; \boldsymbol{B}_g) \right] \\ &\geq \sum_{n=1}^{N} \sum_{g=1}^{G} p(Z_n = g|\boldsymbol{x}_n) \left[\log \pi_g + \log f(\boldsymbol{x}_n; \mathbf{B}_g) - \log p(Z_n = g|\boldsymbol{x}_n) \right],\end{aligned}$$

where

$$\begin{aligned}\log f(\boldsymbol{x}_n; \boldsymbol{B}_g) &= \sum_{k=1}^{T/2} \log \left\{ \frac{1}{n_g} \sum_{i=1}^{n_g} f(\boldsymbol{x}_n; \boldsymbol{B}_g) \right\} \\ &\geq \frac{1}{n_g} \sum_{k=1}^{T/2} \sum_{i=1}^{n_g} \left[\frac{x_{nk}}{b_{k,g}} \log x_{ik,g} - \frac{x_{ik,g}}{b_{k,g}} - \left(\frac{x_{nk}}{b_{k,g}} + 1 \right) \log b_{k,g} - \log \left\{ \Gamma \left(\frac{x_{nj}}{b_{k,g}} + 1 \right) \right\} \right],\end{aligned}$$

then take the partial derivative with respect to $b_{k,g}$,

$$\begin{aligned}\frac{\partial \ell_{\text{Gamma}}(\boldsymbol{B}; \boldsymbol{x})}{\partial b_{k,g}} = \sum_{n=1}^{N} p(Z_n = g|\boldsymbol{x}_n) \Bigg[\sum_{i=1}^{n_g} \Bigg\{ &-\frac{x_{nk}}{b_{k,g}^2} \log x_{ik,g} + \frac{x_{ik,g}}{b_{k,g}^2} + \frac{x_{nk}}{b_{k,g}^2} \log b_{k,g} \\ &- \left(\frac{x_{nk}}{b_{k,g}^2} + \frac{1}{b_{k,g}} \right) + \frac{\Gamma'(\frac{x_{nk}}{b_{j,g}} + 1)}{\Gamma(\frac{x_{nk}}{b_{k,g}} + 1)} \left(\frac{x_{nk}}{b_{k,g}^2} \right) \Bigg\} \Bigg].\end{aligned}$$

Setting the above equation equal to zero yields (3.2). This function is a non-linear equation, so the Newton-Raphson method will be used to obtain an approximation of the root.

$$\begin{aligned}\sum_{n=1}^{N} p(Z_n = g|\boldsymbol{x}_n) \Bigg[\sum_{i=1}^{n_g} \Bigg\{ &-\frac{x_{nk}}{b_{k,g}^2} \log x_{ik,g} + \frac{x_{ik,g}}{b_{k,g}^2} + \frac{x_{nk}}{b_{k,g}^2} \log b_{k,g} \\ &- \left(\frac{x_{nk}}{b_{k,g}^2} + \frac{1}{b_{k,g}} \right) + \frac{\Gamma'(\frac{x_{nk}}{b_{j,g}} + 1)}{\Gamma(\frac{x_{nk}}{b_{k,g}} + 1)} \left(\frac{x_{nk}}{b_{k,g}^2} \right) \Bigg\} \Bigg] = 0.\end{aligned} \tag{3.2}$$

Similarly, we obtain the score function for the Gaussian:

$$\begin{aligned}\ell_{\text{Gaussian}}(\boldsymbol{B};\boldsymbol{x}) &= \sum_{n=1}^{N}\log\left[\sum_{g=1}^{G}\{\pi_g f(\boldsymbol{x}_n;\boldsymbol{B}_g)\}\right] \\ &\geq \sum_{n=1}^{N}\sum_{g=1}^{G} p(z_n=g|\boldsymbol{x}_n)\left[\log\pi_g + \log\{f(\boldsymbol{x}_n;\boldsymbol{B}_g)\} - \log\{p(z_n=g|\boldsymbol{x}_n)\}\right],\end{aligned}$$

where

$$\begin{aligned}\log\{f(\boldsymbol{x}_n;\boldsymbol{B}_g)\} &= \sum_{k=1}^{T/2}\log\left\{\frac{1}{n_g}\sum_{i=1}^{n_g} f(\boldsymbol{x}_n;\boldsymbol{B}_g)\right\} \\ &\geq \frac{1}{n_g}\sum_{k=1}^{T/2}\sum_{i=1}^{n_g}\left\{-\frac{(x_{nk}-x_{ik,g})^2}{2b_{k,g}^2} - \log(b_{k,g}) - \frac{1}{2}\log(2\pi)\right\}.\end{aligned}$$

Taking the partial derivative with respect to $b_{k,g}$ gives

$$\frac{\partial\ell_{\text{Gaussian}}(\boldsymbol{B};\boldsymbol{x})}{\partial b_{k,g}} = \sum_{n=1}^{N} p(z_n=g|\boldsymbol{x}_n)\left[\frac{1}{n_g}\sum_{i=1}^{n_g}\left\{\frac{(x_{nk}-x_{ik,g})^2}{b_{k,g}^3} - \frac{1}{b_{k,g}}\right\}\right], \tag{3.3}$$

and, setting (3.3) equal to zero yields,

$$b_{k,g} = \frac{\sum_{n=1}^{N} p(z_n=g|\boldsymbol{x}_n)\left\{\frac{1}{n_g}\sum_{i=1}^{n_g}(x_{nk}-x_{i,g})^2\right\}}{\sum_{n=1}^{N} p(z_g|\boldsymbol{x}_n)}.$$

Finally, we update π_g in the M-step. Due to the constraint that $\sum_{g=1}^{G}\pi_g = 1$, the Lagrange multiplier method is used to obtain π_g. Note that

$$\ell(\pi_g) = \sum_{n=1}^{N}\sum_{g=1}^{G}\left[p(z_n=g|\boldsymbol{x}_n)\log(\pi_g)\right] + \lambda\left(\sum_{g=1}^{G}\pi_g - 1\right),$$

where λ is the Lagrange multiplier. Taking the partial derivative,

$$\frac{\partial\ell}{\partial\pi_g} = \sum_{n=1}^{N}\frac{p(z_n=g|\boldsymbol{x}_n)}{\pi_g} + \lambda, \tag{3.4}$$

and setting (3.4) equal to zero yields

$$\pi_g = -\sum_{n=1}^{N} \frac{p(z_n = g|\boldsymbol{x}_n)}{\lambda}.$$

It follows that

$$\sum_{g=1}^{G} \pi_g = -\sum_{g=1}^{G}\sum_{n=1}^{N} \frac{p(z_n = g|\boldsymbol{x}_n)}{\lambda} = 1 \tag{3.5}$$

and, noting that

$$\sum_{g=1}^{G} p(z_n = g|\boldsymbol{x}_n) = 1,$$

it follows that $\lambda = -N$. Thus, we use the following equation to update π_g:

$$\pi_g = \frac{1}{N}\sum_{n=1}^{N} p(z_n = g|\boldsymbol{x}_n). \tag{3.6}$$

To avoid obtaining a local optimum, initial labels will be given by `kmeans` and `mclust` clustering results, and initial bandwidths and prior probabilities will be computed based on initial labels. To select a model, we will use the log-likelihood score when the number of parameters is the same or BIC when the number of parameters differs between models.

3.2 Multivariate Longitudinal Data Clustering

3.2.1 Independent Attributes

Suppose there are D independent random variables. The kernel density estimation functions are shown below:

$$\hat{f}_{\text{Gamma}}(\boldsymbol{x}_{nk}; \boldsymbol{B}_g) = \frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D} \frac{(x_{idk,g})^{\frac{x_{nd,k}}{b_{dk,g}}} \exp\left(-\frac{x_{idk,g}}{b_{d,g}}\right)}{(b_{d,g})^{\frac{x_{nd,k}}{b_{d,g}}+1}\Gamma(\frac{x_{nd,k}}{b_{d,g}}+1)},$$

$$\hat{f}_{\text{Gaussian}}(\boldsymbol{x}_{nk}; \boldsymbol{B}_g) = \frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\left[\frac{1}{\sqrt{2\pi}b_{d,g}}\exp\left\{-\frac{(x_{nd,k}-x_{idk,g})^2}{2(b_{d,g})^2}\right\}\right],$$

where $x_{nd,k}$ are observations taken under frequency k, from individual n attribute d, $x_{idk,g}$ are observations of the random variable d in the class g, and $b_{d,g}$ is the bandwidth of the random variable d in cluster g.

Based on the multi-dimensional kernel density function, joint density functions of frequencies are:

$$\hat{f}_{\text{Gamma}}(\boldsymbol{x}_n;\boldsymbol{B}_g)=\prod_{k=1}^{T/2}\left\{\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\frac{(x_{idk,g})^{\frac{x_{nd,k}}{b_{dk,g}}}\exp\left(-\frac{x_{idk,g}}{b_{d,g}}\right)}{(b_{d,g})^{\frac{x_{nd,k}}{b_{d,g}}+1}\Gamma(\frac{x_{nd,k}}{b_{d,g}}+1)}\right\},$$

$$\hat{f}_{\text{Gaussian}}(\boldsymbol{x}_n;\boldsymbol{B}_g)=\prod_{k=1}^{T/2}\left[\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\left[\frac{1}{\sqrt{2\pi}b_{d,g}}\exp\left\{-\frac{(x_{nd,k}-x_{idk,g})^2}{2(b_{d,g})^2}\right\}\right]\right].$$

Plugging $\hat{f}_{\text{Gamma}}(\boldsymbol{x}_n;\boldsymbol{B}_g)$ and $\hat{f}_{\text{Gaussian}}(\boldsymbol{x}_n;\boldsymbol{B}_g)$ into the finite mixture model, we get density functions of mixture models,

$$\begin{aligned}
f_{\text{Gaussian}}(\boldsymbol{x}_n;\boldsymbol{B}_g)&=\sum_{g=1}^{G}\pi_g\hat{f}(\boldsymbol{x}_n;\boldsymbol{B}_g)\\
&=\sum_{g=1}^{G}\pi_g\prod_{k=1}^{T/2}\left[\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\left[\frac{1}{\sqrt{2\pi}b_{d,g}}\exp\left\{-\frac{(x_{nd,k}-x_{idk,g})^2}{2(b_{d,g})^2}\right\}\right]\right],\\
f_{\text{Gamma}}(\boldsymbol{x}_n;\boldsymbol{B}_g)&=\sum_{g=1}^{G}\pi_g\hat{f}(\boldsymbol{x}_n;\boldsymbol{B}_g)\\
&=\sum_{g=1}^{G}\pi_g\left[\prod_{k=1}^{T/2}\left\{\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\frac{(x_{idk,g})^{\frac{x_{nd,k}}{b_{dk,g}}}\exp\left(-\frac{x_{idk,g}}{b_{d,g}}\right)}{(b_{d,g})^{\frac{x_{nd,k}}{b_{d,g}}+1}\Gamma(\frac{x_{nd,k}}{b_{d,g}}+1)}\right\}\right].
\end{aligned}$$

Next, using the EM algorithm to estimate parameters, we get posterior probabilities based on (3.1) in the E-step. In the M-step, lower bounds of the log-likelihood function are shown below:

$$\ell(\boldsymbol{B};\boldsymbol{X})=\sum_{n=1}^{N}\sum_{g=1}^{G}p(z_n=g|\boldsymbol{x}_n)\left[\log\pi_g+\log f(\boldsymbol{x}_n;\boldsymbol{B}_g)-\log p(z_n=g|\boldsymbol{x}_n)\right],$$

where

$$\log f_{\text{Gamma}}(\boldsymbol{x}_n;\boldsymbol{B}_g) = \sum_{k=1}^{T/2}\log\left[\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\frac{(x_{idk,g})^{\frac{x_{nd,k}}{b_{dk,g}}}\exp\left(-\frac{x_{idk,g}}{b_{d,g}}\right)}{(b_{d,g})^{\frac{x_{nd,k}}{b_{d,g}}+1}\Gamma\left(\frac{x_{nd,k}}{b_{d,g}}+1\right)}\right]$$

$$\geq \sum_{k=1}^{n_g}\frac{1}{n_g}\sum_{i=1}^{n_g}\sum_{d=1}^{D}\left[\frac{x_{nd,k}}{b_{dk,g}}\log x_{idk,g} - \frac{x_{idk,g}}{b_{dk,g}} - \left(\frac{x_{nd,k}}{b_{dk,g}}+1\right)\log b_{dk,g} - \log\left\{\Gamma\left(\frac{x_{nd,k}}{b_{dk,g}}+1\right)\right\}\right],$$

$$\log f_{\text{Gaussian}}(\boldsymbol{x}_n;\boldsymbol{B}_g) = \sum_{k=1}^{T/2}\log\left[\frac{1}{n_g}\sum_{i=1}^{n_g}\prod_{d=1}^{D}\frac{1}{\sqrt{2\pi}b_{dk,g}}\exp\left\{\frac{-(x_{nd,k}-x_{idk,g})^2}{2(b_{dk,g})^2}\right\}\right]$$

$$\geq \sum_{k=1}^{T/2}\frac{1}{n_g}\sum_{i=1}^{n_g}\sum_{d=1}^{D}\left\{\frac{1}{2}\log(2\pi) - \log b_{dk,g} - \frac{(x_{nd,k}-x_{idk,g})^2}{2(b_{dk,g})^2}\right\}.$$

Maximizing both gamma and Gaussian log-likelihood functions, we get estimators of bandwidths. The roots of (3.7) will be estimated via the Newton-Raphson method.

$$\text{Gamma:}\quad f(b_{dk,g}) = \sum_{n=1}^{N}p(z_g|\boldsymbol{x}_n)\sum_{i=1}^{n_g}\left\{-\frac{x_{nd,k}}{b_{dk,g}^2}\log x_{idk,g} + \frac{x_{idk,g}}{b_{dk,g}^2} + \frac{x_{nd,k}}{b_{dk,g}^2}\log b_{dk,g} - \frac{x_{nd,k}}{b_{dk,g}^2} - \frac{1}{b_{dk,g}} + \frac{\Gamma'(\frac{x_{nd,k}}{b_{dk,g}+1})}{\Gamma(\frac{x_{nd,k}}{b_{dk,g}}+1)}\frac{x_{nd,k}}{b_{dk,g}^2}\right\} = 0 \tag{3.7}$$

$$\text{Gaussian:}\quad b_{dk,g} = \frac{\sum_{n=1}^{N}p(z_g|\boldsymbol{x}_n)\frac{1}{n_g}\sum_{i=1}^{n_g}(x_{nd,k}-x_{idk,g})^2}{\sum_{n=1}^{N}p(z_g|\boldsymbol{x}_n)}. \tag{3.8}$$

The EM algorithm for kernel mixture models is summarized in Algorithm 1. The first step is to use `kmeans` and `Mclust` to cluster observations for every attribute in the data set, and then we use these clustering results as inital labels to iterate the E- and M-steps until convergence. Finally, we select the final model based on BIC scores.

Algorithm 1 Kernel mixture model clustering based on independence assumption
(1) Initialize the labels by Kmeans and Mclust methods to classify one spectrum data, and use these labels to find $x_{id,g}$.
(2) Initialize the bandwidth based on initial labels
(3) E-step: calculate posterior probability based on (3.1) and log-likelihood value.
(4) M-step: calculate π_g based on (3.6) and bandwidths based on (3.8) and (3.7)
(5) Iterate E-step and M-step until convergence
(6) Compare BIC scores and choose the minimum value as the final result.

3.2.2 Dependent Attributes

In this section, the independence assumption will be relaxed by using copulas. In practice, attributes in the same data sets are typically dependent. For example in disease modeling, the number of death cases is positively associated with the total number of infected cases. However, both these variables should be considered when analyzing an epidemic spreading process. Hence, it is necessary to consider dependent attributes in the clustering model. Herein, the bivariate Gaussian copula will be used to model dependence between two attributes:

$$C(u,v,\rho)=\int_{-\infty}^{\Phi^{-1}(u)}\int_{-\infty}^{\Phi^{-1}(v)}\frac{1}{2\pi\sqrt{1-\rho^2}}\exp\left\{-\frac{x^2-2\rho xy+y^2}{2(1-\rho^2)}\right\}dxdy,$$

where Φ represents standard normal distribution, u and v are the marginal CDFs from two attributes, $u=F_1(x_{n1,k})$ and $v=F_2(x_{n2,k})$, $c(u,v;\rho)$ is the joint CDF. As PML will be used to estimate parameters, we take partial derivatives with respect to x_1 and x_2 to get the joint density function. The joint PDF is shown below:

$$\begin{aligned}c(x_{n1,k},x_{n2,k};\rho)=&\frac{1}{\sqrt{1-\rho^2}}\exp\left\{-\frac{\rho^2\left[\Phi^{-1}(F_1(x_{n1,k}))\right]^2+\rho^2\left[\Phi^{-1}(F_2(x_{n2,k}))\right]^2}{2(1-\rho^2)}\right.\\&\left.-\frac{2\rho\Phi^{-1}(F_1(x_{n1,k}))\Phi^{-1}(F_2(x_{n2,k}))}{1-\rho^2}\right\}f(X_1)g(X_2).\end{aligned}\tag{3.9}$$

Plugging (3.9) into the finite mixture model, we get the log-likelihood function:

$$\begin{aligned}\ell(\rho,\boldsymbol{B}|\boldsymbol{X})=&\sum_{n=1}^{N}\sum_{g=1}^{G}p(z_n=g|\boldsymbol{x}_n)\left[\log\pi_g+\sum_{k=1}^{T/2}\left[\log\left\{c(F_1(x_{n1,k}),F_2(x_{n2,k});\rho_g,\boldsymbol{B}_g)\right\}\right.\right.\\&\left.\left.+\log\left\{f(x_{n1,k};\boldsymbol{B}_g)\right\}+\log\left\{f(x_{n2,k};\boldsymbol{B}_g)\right\}\right]-\log\left\{p(z_n=g|\boldsymbol{X}_n)\right\}\right],\end{aligned}\tag{3.10}$$

where

$$\begin{aligned}
&\log\left\{c(F_1(x_{n1,k}), F_2(x_{n2,k}); \rho_g, \boldsymbol{B}_g)\right\} \\
&\quad = -\frac{1}{2}\log(1-\rho_g^2) - \left[\left\{\Phi^{-1}(F_1(x_{n1,k}; \boldsymbol{B}_g))\right\}^2 + \left\{\Phi^{-1}(F_2(x_{n2,k}; \boldsymbol{B}_g))\right\}^2\right]\frac{\rho_g^2}{2(1-\rho_g^2)} \\
&\qquad + \Phi^{-1}(F_1(x_{n1,k}, \boldsymbol{B}_g))\Phi^{-1}(F_2(x_{n2,k}, \boldsymbol{B}_g))\frac{\rho_g}{1-\rho_g^2}, \\
&F_{\text{Gaussian}}(x_{nd,k}, b_{dk,g}) = \int_{-\infty}^{x_{nd,k}} \left\{\frac{1}{n_g}\sum_{i=1}^{n_g}\frac{1}{\sqrt{2\pi}b_{dk,g}}\exp\left(\frac{-(X_{d,k}-x_{idk,g})^2}{2b_{dk,g}^2}\right)\right\} dX_{d,k}, \\
&F_{\text{gamma}}(x_{nd,k}, b_{dk,g}) = \int_{0}^{x_{nd,k}} \left\{\frac{1}{n_g}\sum_{k=1}^{n_g}\frac{x_{idk,g}^{\frac{X_{d,k}}{b_{dk,g}}}\exp\left(-\frac{x_{idk,g}}{b_{dk,g}}\right)}{b_{dk,g}^{\frac{X_{d.k}}{b_{dk,g}}+1}\Gamma(\frac{X_{d,k}}{b_{dk,g}}+1)}\right\} dX_{d,k}, \\
&f_{\text{Gaussian}}(x_{nd,k}, b_{dk,g}) = \frac{1}{n_g}\sum_{i=1}^{n_g}\left[\frac{1}{\sqrt{(2\pi)}b_{dk,g}}\exp\left\{\frac{-(x_{nd,k}-x_{idk,g})^2}{2(b_{dk,g})^2}\right\}\right], \\
&f_{\text{gamma}}(x_{nd,k}, b_{dk,g}) = \frac{1}{n_g}\sum_{i=1}^{n_g}\frac{x_{idk,g}^{\frac{x_{nd,k}}{b_{dk,g}}}\exp\left(-\frac{x_{idk,g}}{b_{kd,g}}\right)}{(b_{kd,g})^{\frac{x_{nd,k}}{b_{kd,g}}+1}\Gamma(\frac{x_{nd,k}}{b_{dk,g}}+1)},
\end{aligned}$$

where d is the number of attributes. The log-likelihood function includes two parts: one part is from the marginal distribution and another part reflects the dependence parameter. It is difficult to maximize (3.10) as the derivative function is a non-linear function. Thus, we will use Newton-Raphson method to solve this equation. When the number of parameters increases, solving this non-linear function would be time-consuming. Hence, the PML method will be used to estimate the copula parameters.

To estimate parameters in the marginal distribution, we take the partial derivative with respect to the bandwidth and the result is the same as the estimator in (3.7) and (3.8). Next, we take the derivative of the log-likelihood function with respect to ρ:

$$\begin{aligned}
\frac{\partial \ell}{\partial \rho_g} = &\sum_{n=1}^{N} p(z_g|\boldsymbol{x}_n)\sum_{k=1}^{T/2}\left\{\rho_g(1-\rho_g^2) - \rho_g\left[\left\{\Phi^{-1}\left(\frac{R_{n1,k}}{N+1}\right)\right\}^2 + \left\{\Phi^{-1}\left(\frac{R_{n2,k}}{N+1}\right)\right\}^2\right]\right. \\
&\left. + (1-\rho_g+2\rho_g^2)\Phi^{-1}\left(\frac{R_{n1,k}}{N+1}\right)\Phi^{-1}\left(\frac{R_{n2,k}}{N+1}\right)\right\} = 0, \qquad (3.11)
\end{aligned}$$

where $R_{n1,k}$ is the rank of $x_{n1,k}$ among all observations of the first attribute under frequency k. This is a non-linear function, so the Newton-Raphson method will be applied to solve this equation. As we use the two-stage estimation method in the M-step, Algorithm 2 is quite different from Algorithm 1. M-step2 is added in this algorithm to estimate dependence parameters. The other steps in Algorithm 1 are the same as in Algorithm 2 .

Algorithm 2 Kernel mixture model clustering based on dependence assumption

(1) Initialize labels by Kmeans and Mclust methods based on one attribute, and apply this label for all attributes.
(2)Initialize bandwidth based on initial label.
(3) E-step: calculate posterior probability based on (3.1) and log-likelihood value.
(4) M-step1: calculate π_g based on (3.6) and bandwidths based on (3.7) and (3.8)
(5) M-step2: calculate dependent parameters based on (3.11)
(6) Iterate E-step and M-step until convergence
(7) Compare BIC scores and choose the minimum value as the final result.

3.3 Kernel Distance-Based Clustering

In this section, kernel distance-based clustering will be proposed to compare with the kernel mixture model. This method will fit spectral curves rather than the spectral density via the kernel smoothing method. As this is a classification problem not a prediction problem, over-fitting can be accepted to reveal more details about spectral curves. Thus, we will choose as small a bandwidth as possible to fit the curves.

The first step of the kernel distance-based clustering is to fit the spectral curve, and then the squared Euclidean distance will be calculated to form a dissimilarity matrix. After obtaining a dissimilarity matrix, a distance-based clustering method will be used for clustering. The PAM algorithm will be used for clustering, as the dissimilarity matrix is distances between observations and we do not choose the mean values as cluster centres. Assume that the kernel function is the normal distribution, and frequencies vary from -0.5 to 0.5. Then, the kernel smoothing model for $x_{i,k}$ is:

$$\hat{x}_{i,k} = \frac{\sum_{m=1}^{K} K_h(f_k, f_m) x_{i,m}}{\sum_{m=1}^{K} K_h(f_k, f_m)},$$

$$d_{ij} = \int_0^{0.5} (\hat{x}_{i,k} - \hat{x}_{j,k})^2 df_k,$$

where f's are frequencies, x's are spectrum values, and $K_h(f_k, f_m)$ is a kernel function with bandwidth h. The Gaussian kernel smoother can be expressed as:

$$K_h(f_k, f_m) = \exp\left\{-\frac{(f_k - f_m)^2}{2h^2}\right\}.$$

For multivariate longitudinal data, we need to find a way to combine dissimilarity matrices from different attributes together and get a total distance matrix. Two methods of combining dissimilarity matrices have been proposed in the `klmsd` package Genolini et al. (2015), and proved that these two methods get the same results. One of the methods is to compute T distances between different subjects for every attribute and then use the distance function to combine these T distances together. We define this combined distance as $d_r(x_{ir}, x_{jr})$, where r means attribute r. Finally, the author used the distance function again to combine these R distances. Hence, the Minkowski distance between $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ is:

$$d(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sqrt[p]{\sum_{R,T} |x_{ir,t} - x_{jr,t}|^p},$$

where R is the number of variables, and T is the number of time points. Based on this definition, we can get the squared Euclidean distance between variables.

$$d(\boldsymbol{x_i}, \boldsymbol{x_j}) = \sum_{r=1}^{R} \int_0^{0.5} (\hat{x}_{ir,k} - \hat{x}_{jr,k})^2 df_k.$$

3.4 Summary

In this chapter, we proposed the kernel mixture model to cluster longitudinal data. The Gaussian and gamma kernel functions replace the Gaussian distribution in finite mixture models, which can reduce the number of parameters and also make models more robust. Firstly, this model is used to fit univariate longitudinal data, and then we extended it to independent longitudinal data. Finally, we

introduced the bivariate Gaussian copula to the kernel mixture model to relax the independence assumption. Univariate, independent multivariate and dependent bivariate longitudinal data were discussed. In addition, we also provided a kernel distance-based clustering method based on the kernel smoother method, which will be used to compare with the kernel mixture model in Chapters 4 and 5.

Chapter 4

Simulation

4.1 Data Generation

In this chapter, AR and ARIMA models are applied to generate univariate data sets. Suppose that there are two classes, where coefficients of the AR and ARIMA models differ for each group. The first time series is from an ARIMA model, and the second and third time series are from an AR model. Coefficients have opposite signs for two classes in the second data set, but they are similar in the third data set. The models are shown below.

$$\begin{aligned}
&\text{ARIMA}_{\text{class1}}: \quad x_{i,t} - x_{i,(t-1)} = 0.8897(x_{i,(t-1)} - x_{i,(t-2)}) + 0.2279\epsilon_{i,(t-1)}, \\
&\text{ARIMA}_{\text{class2}}: \quad x_{i,t} - x_{i,(t-1)} = -0.4959(x_{i,(t-1)} - x_{i,(t-2)}) - 0.2488\epsilon_{i,(t-1)}, \qquad (4.1) \\
&\text{AR}_{\text{class1}}: \quad x_{i,t} = -0.23x_{i,(t-1)} + \epsilon_{i,t}, \\
&\text{AR}_{\text{class2}}: \quad x_{i,t} = 0.88x_{i,(t-1)} + \epsilon_{i,t}, \qquad (4.2) \\
&\text{AR}_{\text{class1}}: \quad x_{i,t} = 0.23x_{i,(t-1)} + \epsilon_{i,t}, \\
&\text{AR}_{\text{class1}}: \quad x_{i,t} = 0.5x_{i,(t-1)} + \epsilon_{i,t}, \qquad (4.3)
\end{aligned}$$

where $x_{i,t}$ means the observed value of subject i at the time point t. The errors are sampled from a standard normal distribution, and they are independent between subjects.

Next, we will use these models to generate bivariate panel data sets. Suppose there are two attributes in the panel data, and they are continuous variables.

Errors will be generated from a multivariate Gaussian distribution allowing for different correlation coefficients for each class. Six panel data sets will be formed based on Table 4.1.

Table 4.1: Panel data sets, where ρ_1 and ρ_2 represent correlation coefficients in two classes.

Panel	Attribute1	Attribute2	Correlation coefficients
1	(4.3)	(4.1)	ρ_1=ρ_2 =0
2	(4.2)	(4.3)	ρ_1=ρ_2 = 0.25
3	(4.2)	(4.3)	ρ_1=ρ_2 =0.75
4	(4.2)	(4.3)	ρ_1=ρ_2 = 0.98
5	(4.3)	(4.1)	ρ_1=0, ρ_2 = 0.75
6	(4.3)	(4.1)	ρ_1=0, ρ_2 = 0.98

4.2 Univariate Panel Data Simulation

In this section, three univariate panel data sets are generated based on equations (4.1), (4.2) and (4.3). Suppose that there are fifty observations over thirty time points, where half of them are from the first class and the other half are from the second class. The time series plots are shown in Figure 4.1.

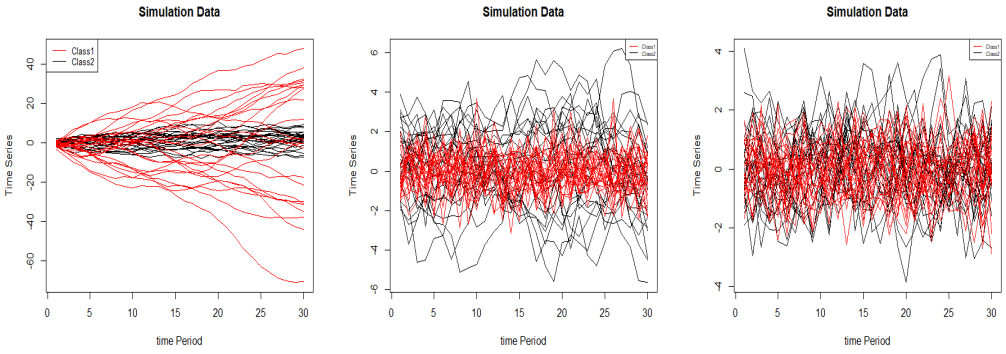

Figure 4.1: These time series are generated based on ARIMA and AR models. Red lines are class 1 and black lines are class 2. The time series in the left panel are generated through (4.1), medium panel is from (4.2), and right panel is from (4.3).

The left panel of Figure 4.1 shows that ARIMA time series is not stationary, but the divergence of class 1 is larger than that of class 2. Thus, the first-order differences between consecutive observations will be computed to get a stationary

time series. The other two time series are stationary. The middle panel indicates that the data in class 1 fluctuate on a smaller range than data from the second class. However, fluctuation ranges of two clusters in right panel are almost the same.

Suppose that every period consists of thirty time points. After a Fourier transformation, the spectral plot can be seen in Figure 4.2. The left panel shows spectral values of the first univariate panel data in class 1 is larger than values in class 2, but this is opposite in the middle panel. However, these two panels both indicate that it is easy to cluster attributes into two groups. For the third spectral plot, most of red lines and blacks lines are mixed together due to the similar coefficients in the AR model.

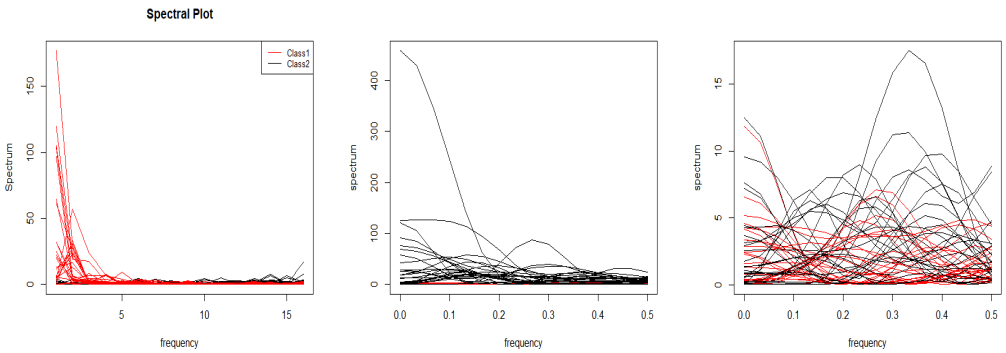

Figure 4.2: Spectral plots from the Fourier transformation. Red lines are values in class 1 and black lines are values in class 2. Due to the symmetry property of the Fourier transform, we just give spectral values with frequencies between 0 and 0.5. This can reduce the computational complexity.

4.2.1 Kernel Mixture Model Clustering

Next, spectral values will be used to cluster univariate panel data. Labels from `kmeans` and `mclust` both will be used as initial labels to get a global optimal result. As the number of parameters is the same among the Gaussian and gamma models, the log-likelihood values can be used to select the model. Hence, labels with the maximum log-likelihood will be chosen as the final clustering result in the simulation study. The following tables are confusion matrices of three panel data sets.

Table 4.2: Confusion matrix of the first univariate panel data set.

	time series				spectrum value											
	kmeans		Mclust		kmeans		Mclust		kernel mixture model							
									Gaussian				Gamma			
									kmeans		Mclust		kmeans		Mclust	
	predictted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	12	13	24	1	18	7	9	16	1	24	5	20	1	24	4	21
true class2	0	25	25	0	25	0	12	13	25	0	6	19	25	0	5	20
log-likelihood	-		-		-		-		-1410.645		-1593.590		-1360.698		-1550.224	

Table 4.3: Confusion matrix of the second univariate panel data set.

	time series				spectrum value											
	kmeans		Mclust		kmeans		Mclust		kernel mixture model							
									Gaussian				Gamma			
									kmeans		Mclust		kmeans		Mclust	
	predictted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	25	0	3	22	25	0	25	0	25	0	25	0
true class2	14	11	15	10	14	11	25	0	18	7	0	25	0	25	0	25
log-likelihood	-		-		-		-		-3123.9384		-2244.3741		-2201.5950		-2201.5950	

Table 4.4: Confusion matrix of the third univariate panel data set.

	time series				spectrum value											
	kmeans		Mclust		kmeans		Mclust		kernel mixture model							
									Gaussian				Gamma			
									kmeans		Mclust		kmeans		Mclust	
	predictted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
truen class1	14	11	25	0	25	0	14	11	25	0	25	0	25	0	24	1
true class2	13	12	24	1	16	9	6	19	10	15	17	8	10	15	15	10
log-likelihood	-		-		-		-		-1899.271		-1930.884		-1870.377		-1875.163	

Table 4.2 shows clustering results from the first univariate panel data set. The `kmeans` method (misclassification rate = 0.26, ARI = 0.2187) performs better than `Mclust` (misclassification rate = 0.48, ARI = 0) when we use original time series for clustering. After a Fourier transformation, `kmeans` (misclassification rate = 0.36, ARI = 0.063) is also better than `Mclust` (misclassification rate = 0.44, ARI = -0.0056). The result of the kernel mixture model shows the log-likelihood value of the gamma kernel mixture model with `kmeans` initialized label is a maximum, so this model is the best among the four models. The misclassification rate decreases to 0.02, and ARI increases to 0.9199.

Table 4.3 is based on the second panel data set. There is no big difference between the `kmeans` (misclassification rate = 0.28, ARI = 0.1821) and `Mclust` (misclassification rate = 0.3, ARI = 0.1488) methods when clustering time series directly. The performance of `Mclust` (misclassification rate = 0.06, ARI = 0.7697) has a significant improvement via a Fourier transformation. For the kernel mixture model, the best result is when the kernel function is a gamma distribution, and the log-likelihood score equals -2201.595. The misclassification rate decreases to

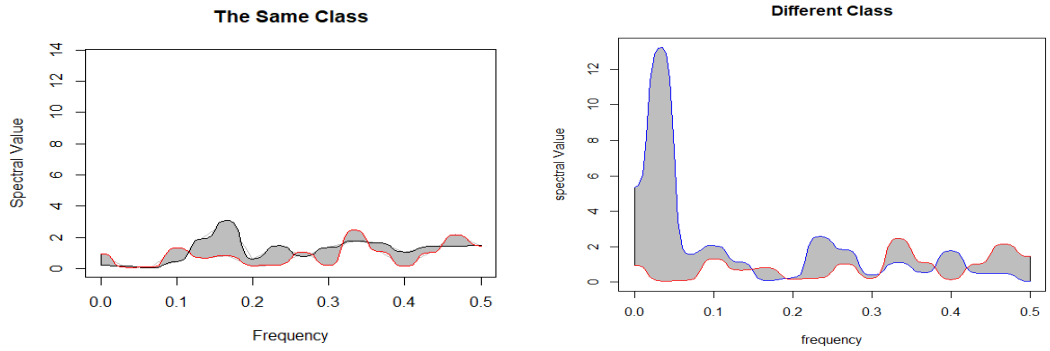

Figure 4.3: Estimated spectral curves through the kernel smoothing method. Bandwidth is 0.01. The left panel is two spectral curves from the same cluster, and the right panel is from different clusters.

zero and ARI equals 1. We also find that the kernel mixture model can improve accuracy no matter which kernel functions are used.

The misclassification rate in Table 4.4 is higher than in the first two tables. Twenty-four observations are misclassified based on the time series, and the ARI of `kmeans` (-0.0191) is smaller than that of `Mclust` (0). The Fourier transform makes results better, the ARI of `kmeans` and `Mclust` increases to 0.1189 and 0.0844, and misclassification rates decrease to 0.32 and 0.34. When we use the kernel mixture model for clustering, the best result is from the Gamma kernel function with the `kmeans` initial label (misclassification rate $= 0.2$, ARI $= 0.3488$), and the log-likelihood score is -1870.377.

Overall, based on the above three tables, we observe that using a Fourier transform for clustering may decrease the misclassification rate, kernel mixture model can improve the performance, and log-likelihood scores of the gamma kernel function are larger than those of the Gaussian kernel function.

4.2.2 Kernel Distance-based Clustering

To illustrate the kernel distance-based clustering method, three observations are chosen and form the following spectral plots. Observations in the left panel are from the same cluster and observation in the right panel are from different clusters. As we can see, the overlapping area in the left panel is smaller than that in the right panel. Thus, this is an efficient nonparametric method of clustering.

When we get the dissimilarity matrix, PAM will be used for clustering. The clustering results are included in Table 4.5, which are similar to that of the kernel mixture model clustering. The number of misclassified observations in the first and the second data sets increases by two and one, but this method reduces one misclassified observation in the third data set. For the univariate panel data set, there is no significant difference between the kernel mixture model and the distance-based model.

Table 4.5: Confusion matrix of the kernel distance-based clustering method.

	data set 1		data set 2		data set 3	
	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	22	3	25	0	23	2
true class2	0	25	1	24	7	28

4.3 Multivariate Panel Data Simulation

In this section, the kernel mixture model will be applied to the multivariate panel data. Suppose that correlation coefficients are the same in two clusters. Four multivariate panel data sets are generated in the first four ways shown in Table 4.1, but correlation coefficients in the time domain are quite different from those of errors due to the coefficients of the AR and ARIMA models. The following table and figures display this change.

Table 4.6: Correlation coefficients of panel 1, panel 2, panel 3 and panel 4.

	residual	time domain	frequency domain
panel 1	0	0.002	-0.01
panel 2	0.25	0.15	0.15
panel 3	0.75	0.53	0.43
panel 4	0.98	0.77	0.78

The first four plots in Figure 4.4 show weak relationships between two variables. Correlation coefficients of these two panel data sets are -0.01 and 0.15 in the frequency domain. The remaining four plots in Figure 4.4 show strong relationships

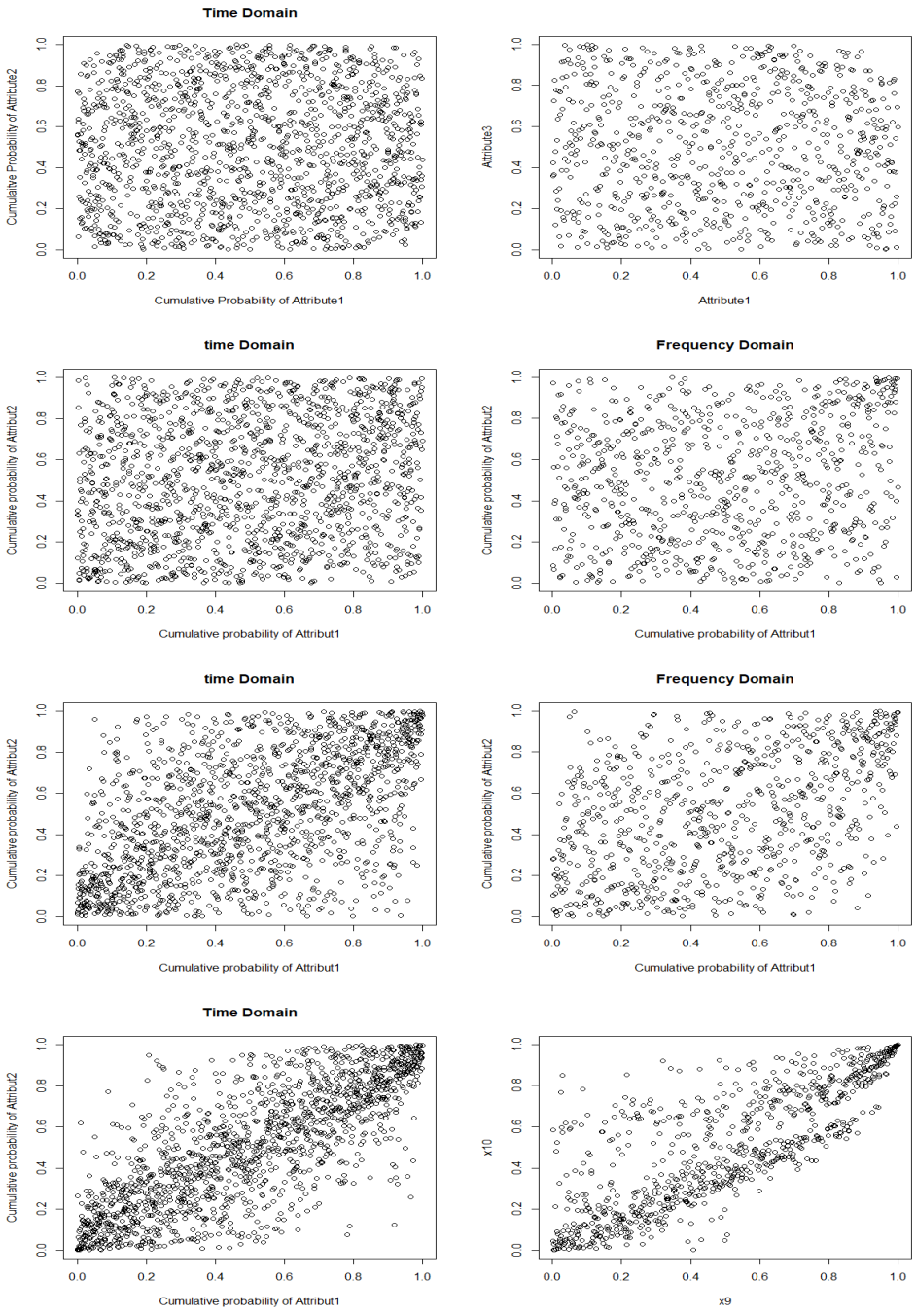

Figure 4.4: Correlation between two attributes in four multivariate panel data sets is shown in this figure. The left panel is correlation in the time domain and the right panel is correlation in frequency.

with corresponding correlation coefficients of 0.428 and 0.78.

Initially, we will make a classification for panel data based on the independence assumption. `kmeans` and `Mclust` clustering results will be used as initial labels. Tables 4.7–4.10 contain the four clustering results.

Table 4.7: Confusion matrix of panel 1 based on the independence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	24	1	24	1	8	17	24	1	24	1	24	1	24	1
true class2	11	14	9	16	9	16	10	15	6	19	7	18	6	19	7	18
BIC	7216.163		7140.817		7106.179		7637.669		6822.785		6851.403		6833.761		6834.845	

Table 4.8: Confusion matrix of panel 2 based on the independence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	22	3	25	0	22	3	25	0	22	3	25	0	18	7	25	0
true class2	23	2	16	9	23	2	0	25	12	13	11	14	12	13	0	25
BIC	11265.841		10654.855		11265.841		9219.119		10616.041		10075.897		10828.179		9011.449	

Table 4.9: Confusion matrix of panel 3 based on the independence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	25	0	25	0	25	0	25	0	25	0	25	0
true class2	15	10	14	11	15	10	2	23	11	14	14	11	8	17	2	23
BIC	10210.565		10018.175		10210.565		8437.261		9533.377		9705.975		9277.463		8358.717	

Table 4.10: Confusion matrix of Panel 4 based on the independence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	25	0	25	0	25	0	25	0	25	0	25	0
true class2	19	6	5	20	5	20	5	20	11	14	8	17	5	20	5	20
BIC	10449.787		8680.971		8680.971		8680.971		9013.473		8995.473		8437.801		8437.801	

Table 4.7 shows that the minimum BIC equals 6822.785 when the kernel function is the gamma distribution, and seven subjects are misclassified. In addition, the gamma kernel function performs better than the Gaussian kernel function no matter how initial labels are given. Two attributes in panel 1 are from the first and third univariate data sets. Compared with Table 4.2, the misclassification rate increases to 0.14. In contrast, the misclassification rate decreases by 0.06 compared with Table 4.4. That means that panel data could provide more information than time series especially when time series are difficult to classify.

The minimum BIC in Table 4.8 is 9011.449 when the initial label is given by the `Mclust` result of attribute 2 resulting in a misclassification rate of zero. Results of the gamma kernel function are better than those of the Gaussian kernel function. When the kernel function is the gamma distribution, misclassification rates all decrease no matter how the initial labels are given.

Table 4.9 indicates that the minimum BIC score is 8358.717 when the kernel function is a gamma distribution and the initial label is from `Mclust`, and the misclassification rate decreases to 0.04. Table 4.10 shows that the minimum BIC is 8437.801, and five samples are misclassified. Hence, the misclassification rate is 0.1.

Overall, these tables show that the gamma kernel function performs better than the Gaussian kernel function. Clustering results depend on the initial labels, and it is necessary to try different initial values. Panel data can provide more information for clustering compared with univariate time series. Based on Tables 4.8, 4.9 and 4.10, we find that the misclassification rates increase as the correlation becomes stronger between two attributes.

Next, we will relax the independence assumption and consider dependent attributes that are linked through a copula with a dependence parameter. Four clustering results are shown below and correlation coefficients increase in these panel data sets.

Table 4.11: Confusion matrix of panel 1 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	24	1	24	1	9	16	24	1	23	2	24	1	23	2
true class2	14	11	8	17	7	18	10	13	6	19	8	17	7	18	7	18
BIC	7233.243		7140.531		7057.239		7643.583		6843.167		6861.033		6861.092		6853.234	

Table 4.12: Confusion matrix of panel 2 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	23	2	25	0	23	2	25	0	25	0	25	0	19	6	25	0
true class2	22	3	16	9	22	3	0	25	11	14	11	14	9	16	0	25
BIC	11289.443		10683.591		11289.443		9234.539		10274.395		10099.085		10859.317		9032.759	

Table 4.13: Confusion matrix of Panel 3 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	24	1	24	1	25	0	25	0	24	1	23	2
true class2	21	4	12	13	0	25	0	25	16	9	14	11	0	25	0	25
BIC	11187.433		10212.299		8054.029		8054.017		10789.119		9435.164		7938.733		8058.521	

Table 4.14: Confusion matrix of panel 4 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predictted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	25	0	25	0	25	0	25	0	23	2	23	2
true class1	16	9	5	20	5	20	5	20	7	18	6	19	0	25	0	25
BIC	10140.843		8763.829		8763.829		8763.829		8465.861		7789.295		7511.971		7511.971	

As expected, the results of panel 1 (the independence case) in Table 4.11 are similar to when independence was assumed, and the number of misclassified observations remains seven. When the relationship of variables becomes stronger, the performance is improved by using a copula. For example, BIC scores of panel 3 and panel 4 become smaller when we add a copula in the kernel mixture model. In addition, models with different initial labels in Table 4.12 all show an improvement, and Table 4.13 shows that one observation is misclassified and the misclassification rate decreases from 0.04 to 0.02. The misclassification rate in panel 4 also has a decrease from 0.1 to 0.01.

The above simulation is based on the principle that the correlation coefficient of attributes is the same in panel data sets. Now, we assume that correlation coefficients are different in two clusters. Panels 5 and 6 will be used as an illustration. The correlation coefficients are shown in Table 4.15.

Table 4.15: Correlation coefficients of panel 5 and panel 6.

	ρ_1	ρ_2	ρ_{time1}	ρ_{time2}	$\rho_{\text{frequency1}}$	$\rho_{\text{frequency2}}$
panel5	0.0000	0.7500	-0.0009	0.5200	-0.0800	0.3800
panel6	0.0000	0.9800	-0.0050	0.7060	0.1126	0.9000

Table 4.16 shows results based on the independence assumption, and results of Table 4.17 are from the dependence algorithm. The best model is the gamma kernel mixture model and the initial lable is `Mclust` result of attribute 2, and the number of misclassified observations decreases to 3.

Tables 4.18 and 4.19 are based on the independence and dependence assumptions, respectively. We find that results of the Gaussian kernel function are improved via the dependence algorithm. The classification results are all the same in Table 4.19, though the BIC of the gamma kernel mixture models is a little bit smaller than that of the Gaussian kernel mixture models.

Table 4.16: Confusion matrix of panel 5 based on the independence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	14	11	25	0	25	0	25	0	25	0	25	0	23	2
true class2	9	16	8	17	13	12	5	20	9	16	11	14	11	14	7	18
BIC	8952.009		9631.157		9296.437		8501.553		8510.171		8636.851		8636.851		8334.529	

Table 4.17: Confusion matrix of panel 5 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	25	0	25	0	25	0	25	0	25	0	25	0	25	0	23	2
true class2	9	16	12	13	12	13	5	20	2	23	11	14	11	14	1	24
BIC	8959.833		9104.071		9122.071		8550.091		8261.879		8680.239		8680.239		8261.880	

Table 4.18: Confusion matrix of panel 6 based on the independent assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	24	1	25	0	25	0	24	1	24	1	24	1	24	1	24	1
true class2	0	25	7	18	6	19	0	25	0	25	0	25	0	25	0	25
BIC	8473.701		9679.945		9113.459		8473.701		8312.491		8312.491		8312.491		8312.491	

Table 4.19: Confusion matrix of panel 6 based on the dependence assumption.

	Gaussian								Gamma							
	kmeans				Mclust				kmeans				Mclust			
	Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2		Attribute 1		Attribute 2	
	predicttted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	24	1	24	1	24	1	24	1	24	1	24	1	24	1	24	1
true class2	0	25	0	25	0	25	0	25	0	25	0	25	0	25	0	25
BIC	7992.122		7992.122		7992.122		7992.122		7855.604		7855.604		7855.604		7855.604	

Finally, kernel distance-based clustering will be applied to these six panel data sets. The distance matrix of every attribute will be calculated separately, and the sum of these matrices will be regarded as a total distance matrix. The PAM algorithm will be used to make a classification based on this matrix. Table 4.20 shows the final result.

Table 4.20: Confusion matrix of the kernel distance-based clustering method.

	panel 1		panel 2		panel 3		panel 4		panel 5		panel 6	
	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2	predicted class1	predicted class2
true class1	24	1	17	8	25	0	25	0	25	0	25	0
true class2	6	19	12	13	10	15	11	14	9	16	8	17

For multivariate panel data, the distance-based clustering method performs worse than model-based clustering. Misclassification rates significantly increased in all four panel data sets. For example, the misclassification rate of the second

panel data sets increases from 0 to 0.4. Thus, it appears that the kernel mixture model is more suitable for multivariate panel data.

4.4 Summary

In this chapter, a simulation study was performed, including univariate, independent multivariate and dependent bivariate cases of panel data, and we also compare the proposed kernel mixture model with the proposed distance-based clustering method. For univariate panel data, kernel mixture models can improve the classification accuracy compared with `kmeans` and `Mclust` . In addition, the kernel distance-based clustering can perform as well as the kernel mixture model. For independent multivariate panel data, the log-likelihood values of the gamma kernel function are larger than those of the Gaussian mixture model, so the gamma kernel function consistently outperforms. Compared with univariate results, we find that multivariate panel data can provide more information for clustering especially when one attribute is difficult to classify, such as the third univarate data set. For dependent multivariate panel data, we simulated two cases. The first case was that the correlation between attributes is the same in different clusters. We find that as the correlation increases, the misclassification rate of the dependence algorithm becomes lower than that of the independence algorithm. If correlations are weak, the independent method performs as well as the dependent method. The second case assumed that the correlation between attributes varies as clusters. Based on BIC scores, we also observed that the performance of the dependent method becomes better as the relationship becomes stronger. In addition, when the number of attributes increases, the performance of the distance-based clustering method deteriorates. Overall, the performance of the algorithms depends on the degree of correlation between attributes, and the kernel mixture model is better than the kernel distance-based model when the data set has more than one attribute.

Chapter 5

Application

5.1 COVID-19 Data

The aim of this chapter is to analyze the development of COVID-19 in different countries based on all COVID-19 situation reports from the World Health Organization (WHO) before June 15th. The top ten countries with the highest number of confirmed cases will be selected from each region as observations. Reports include six regions, namely Africa, Americas, Eastern Mediterranean, Europe, South-East Asia and Western Pacific. Table 5.1 shows the selected countries in different regions.

Table 5.1: The top ten countries in six regions.

Africa	Americas	Eastern Mediterranean	Europe	South-East Asia	Western Pacific
Algeria	Argentina	Afghanistan	Belarus	Bangladesh	Australi
Cameroon	Brazil	Bahrain	Belgium	Bhutan	China
Cote d'Ivoire	Canada	Egypt	France	India	Japan
DRC	Chile	Iran	Germany	Indonesia	Malaysia
Gabon	Colombia	Kuwait	Italy	Maldives	Mongolia
Ghana	Dominican Republic	Oman	Netherlands	Myanmar	New Zealand
Guinea	Ecuador	Pakistan	Russia	Nepal	Philippines
Nigeria	Mexico	Qatar	Spain	Sri Lanka	South Korea
Senegal	Peru	Saudi Arabia	UK	Thailand	Singapore
South Africa	USA	United Arab Emirates	Turkey	Timor-Leste	Viet Nam

The total number of confirmed cases, total new cases, total deaths and total new deaths are included in situation reports. In order to reduce the impact of the population size and recording error in each countries, the growth rate of total new cases in five days and the death rate in five days will be used for clustering. Note

that

$$\text{Growth Rate}_{t+5} = \left(\frac{\text{Total New Cases}_{t+5}}{\text{Total New Cases}_t} - 1 \right) \times 100\%,$$

$$\text{Death Rate}_t = \frac{\text{Total Deaths}_t}{\text{Total Confirmed Cases}_t} \times 100\%,$$

where

$$\text{Total New Cases}_t = \sum_{i=0}^{4} \text{Daily New Cases}_{t-i}$$

for $t = 5, 10, \ldots, T$. The lower bound of the growth rate is -1, when the total new cases equals zero. The growth rate equals zero means the number of new cases in five days stays the same as that in the last five days. When the growth rate is larger than zero, exponential growth will occur in the total number of confirmed cases. Thus, this measurement can reflect transmission speed of COVID-19 in different countries. The range of the death rate is between zero and one. This measurement reflects medical levels and shows whether the health care system is weak in these countries.

5.2 Descriptive Statistics

The WHO reports the number of new daily cases from January 20th, and the total number of deaths from February 4th. Tables 5.2 and 5.3 give monthly statistics for the growth rate and the death rate. In January, COVID-19 mainly spread in the Western Pacific region. Most of the new cases in other areas were input cases, so the average growth rate was smaller than zero. However, COVID-19 became a global outbreak in March, and the average growth rate increased to 1.9188. In addition, the growth rate was stable after April especially in June.

Table 5.2: Statistical summary of the growth rate.

	January	February	March	April	May	June
Maximum	8.0000	28.5000	46.0000	30.0000	30.6667	4.0741
3rd Quantile	-1.0000	-1.0000	2.2682	0.5503	0.3353	0.2245
Mean	-0.4714	-0.5049	1.9188	0.4692	0.3676	0.0409
Median	-1.0000	-1.0000	0.5687	0.0926	0.0572	0.0261
1st Quantile	-1.0000	-1.0000	-0.2972	-0.1867	-0.1424	-0.1680
Minimum	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000	-1.0000
Std Deviation	1.5449	4.4421	4.64901	2.0607	2.3930	0.5362

In January and February, death rates in many countries were zero. The maximum in February was one, which happened in Iran. The average death rate was around 0.04 after April and the standard deviation was stable at 0.04. In addition, we find that the death rate in the European region is higher than that in other regions. The top four countries with the highest death rates are all from the European region after April. According to Figure 5.1, we find that the COVID-19

Table 5.3: Statistical summary of the death rate.

	February	March	April	May	June
Maximum	1.0000	0.3333	0.1893	0.1991	0.1954
3rd Quantile	0.0000	0.02139	0.0516	0.0549	0.0494
Mean	0.01047	0.01661	0.0376	0.0393	0.0372
Median	0.0000	0.0000	0.0220	0.0235	0.02124
1st Quantile	0.0000	0.0000	0.0081	0.0073	0.0068
Minimum	0.0000	0.0000	0.0000	0.0000	0.0000
Std Deviation	0.0687	0.03222	0.0411	0.0443	0.0442

outbreak began in China, and neighboring countries were also affected, such as South Korea and Thailand. Europe and Americas also had input cases. In February, COVID-19 occurred in Europe and Iran, and the global outbreak happened in March. The number of new cases also increased in April. During the last two months, there was a decreasing trend in most countries. New Zealand did not have new cases in June, but the number of cases rebounded in China due to the Beijing outbreak.

Figure 5.2 displays the change of growth and death rates within six months. The left panel shows that outbreak time in each countries are different, but most of them occurred in March. The right panel shows common patterns of the death rate. One pattern is to first increase, then decrease and finally become stable, e.g., the death rate of Iran. The second pattern is to increase gradually and then keep stable. After April, death rates in all countries are lower than 0.2.

5.3 Fourier Transform

The observation period of growth rates is from January 31 to June 15. In order to get a balanced panel data set, we will use zero to fill in missing data in

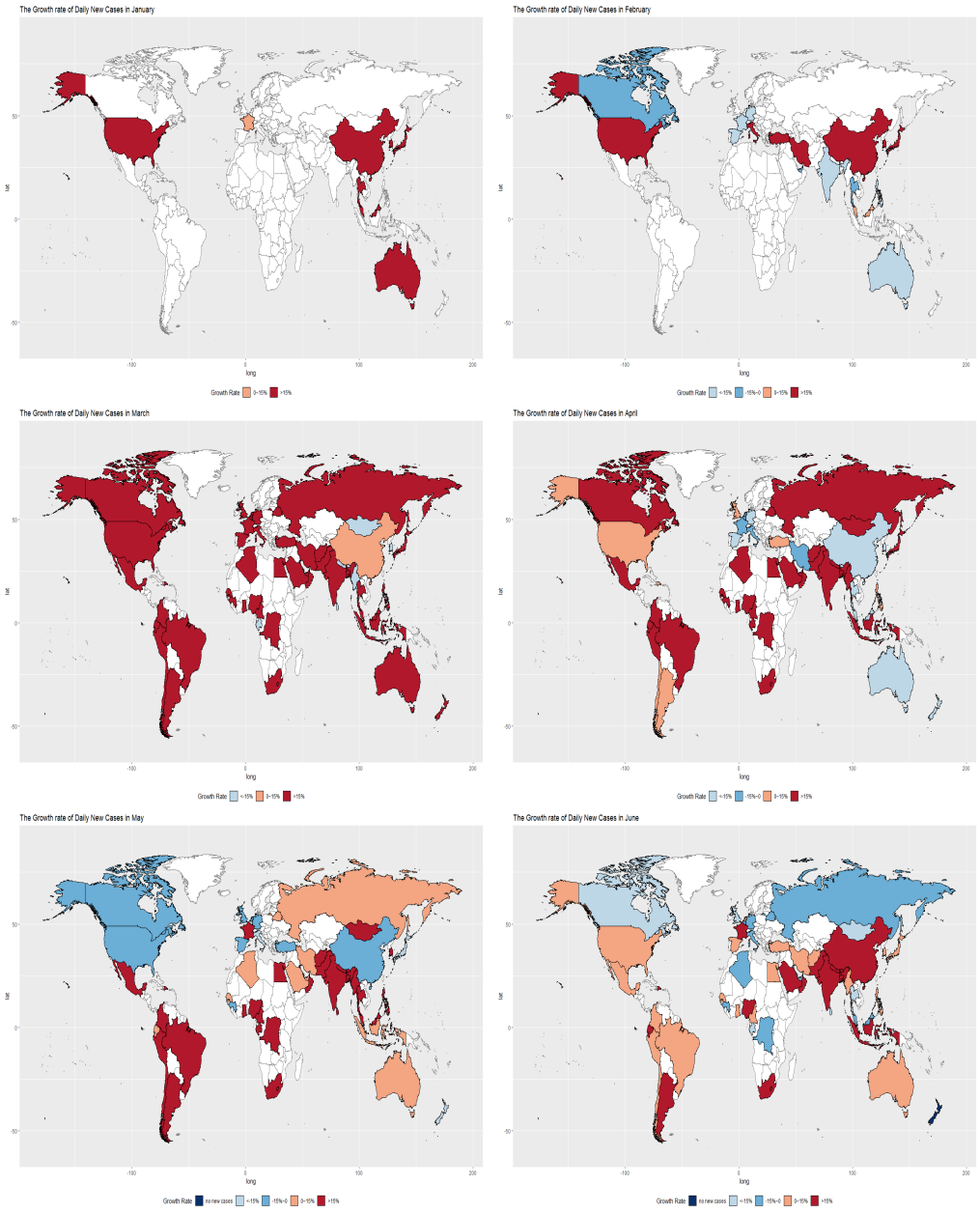

Figure 5.1: Development of COVID-19 during a 6 month period is shown in this figure. The red areas indicate a COVID-19 outbreak in this country, and an orange areas represent that the growth rate became stable. Dark blue areas represents no new cases in this country, and blue areas show that there is a decrease in the growth rate.

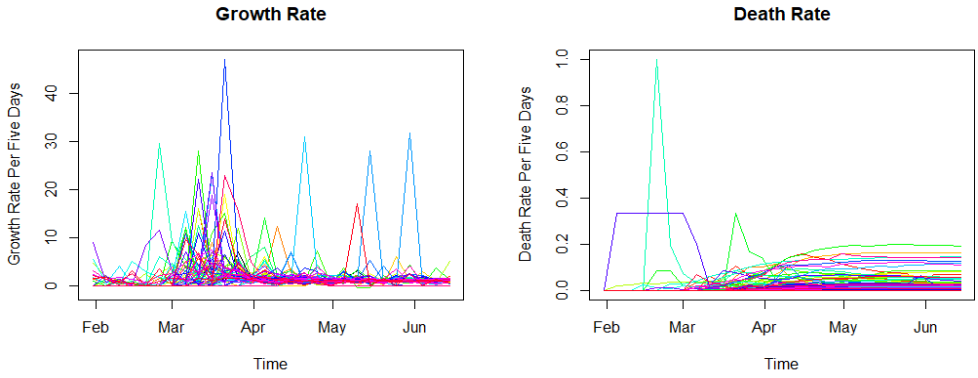

Figure 5.2: Time series plots for growth rates and death rates. The left panel shows growth rate in sixty countries and the right panel is death rate.

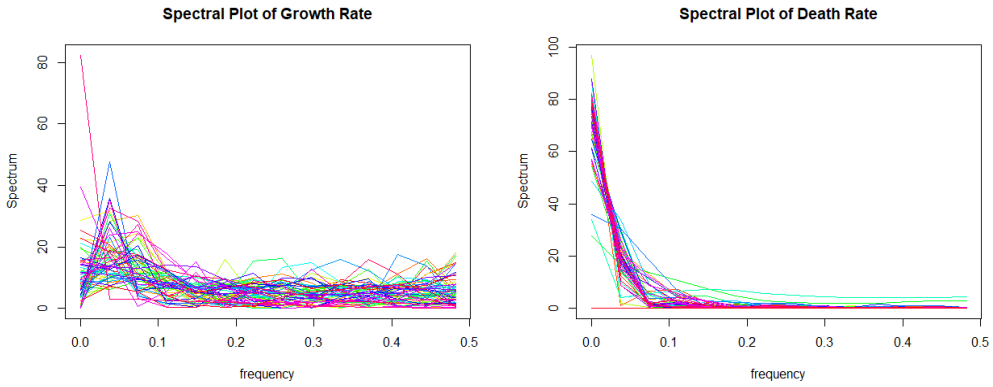

Figure 5.3: Spectral plot of the growth and death rate in sixty countries.

the death rate, as the death rate is a non-decreasing sequence. Thus, the length of time points is 28. Assume that the 28 time points form one period. Due to the symmetric property in the spectral plot, the number of spectral values is 15 and frequencies vary from zero to 0.5. Figure 5.3 shows the spectral plot after the Fourier transformation.

The left panel in Figure 5.3 is from the growth rate, and highest values of each country, such as Italy, are mainly concentrated between frequency 0 and 0.1, and then spectrum values decrease and form a long tail. However, spectrum values in some other countries, such as the USA, continuously decrease. The right panel shows that spectrum values in the death rate have a continuous decrease in all countries, as death rates in the time domain have an increasing trend. However, the highest spectral values are different in each country, and there are no deaths in four countries. They are Vietnam, Mongolia, Bhutan and the Maldives, respectively.

5.4 Kernel Mixture Model

Although sixty countries are selected from six regions, it is hard to say if countries from the same region belong to one cluster. Many factors can affect the development of this pandemic, such as medical level, living habits and measurements taken by governments. Thus, the number of clusters is unknown. We set $G = 1, \ldots, 15$ and initial labels will be given by `kmeans` and `mclust` separately. Finally, we will use the AIC, BIC and ICL to select the best model.

Table 5.4: BIC, AIC and ICL scores.

	BIC				AIC				ICL			
	Gaussian		Gamma		Gaussian		Gamma		Gaussian		Gamma	
	independent	dependent	independnt	dependent	independent	dependent	independent	dependent	independent	dependent	independent	dependent
2	7502.075	7533.184	6205.653	6237.038	7227.538	7249.18	5931.116	5953.034	7502.14	7533.2	6205.7	6237.08
3	7290.402	7340.736	5690.684	6077.028	6878.596	6914.73	5278.878	5651.022	7290.5	7340.84	5690.786	6077.132
4	7174.241	7236.116	5368.135	5391.338	6625.166	6668.108	4819.06	4823.33	7174.488	7236.396	5368.292	5391.862
5	7159.781	7244.976	5253.757	5350.136	6473.438	6534.966	4567.414	4640.126	7160.912	7247.29	5253.968	5350.714
6	7165.036	7151.734	5301.348	5369.13	6341.424	6299.722	4477.736	4517.118	7166.726	7152.58	5301.884	5379.99
7	6985.269	6950.168	4938.483	4998.36	6024.388	5956.154	3977.602	4004.346	6990.338	6955.142	4943.324	5009.776
8	6966.785	6929.162	4968.787	5060.49	5868.636	5793.146	3870.638	3924.474	6971.438	6933.764	4986.774	5075.766
9	6996.89	6995.861	5011.624	4741.343	5761.472	5717.842	3776.206	3463.324	6998.742	6998.18	5025.144	4741.674
10	7015.433	6964.475	4696.845	4797.305	5642.746	5544.454	3324.158	3377.284	7020.132	6970.452	4708.162	4894.846
11	7089.365	7069.979	4811.093	4921.675	5579.41	5507.956	3301.138	3359.652	7098.778	7077.472	4823.056	4930.524
12	7161.712	7137.277	4944.772	4980.911	5514.488	5433.252	3297.548	3276.886	7166.298	7142.564	4952.322	4989.582
13	7106.957	7105.771	4725.878	4703.267	5322.464	5172	2941.386	2857.24	7112.14	7020.164	4728.944	5189.428
14	7232.229	7173.037	4804.87	4850.42	5310.468	5185.008	2883.109	2862.391	7239.518	7182.91	4810.616	4858.904
15	7214.346	7157.281	4772.224	4948.782	5155.316	4983.536	2713.195	2818.751	7220.734	7119.638	4775.57	4953.428

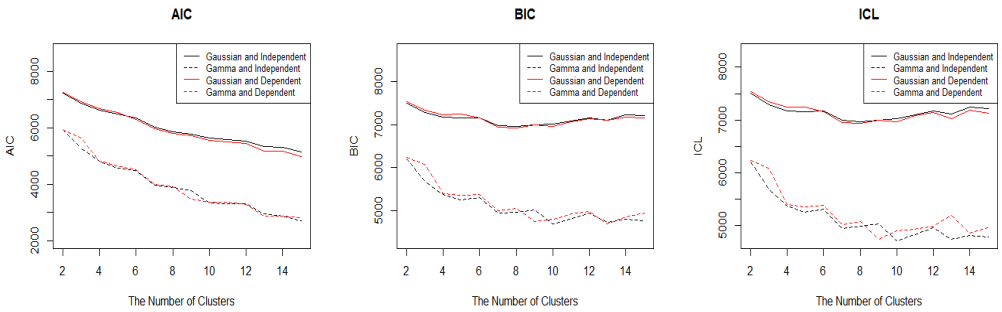

Figure 5.4: AIC, BIC and ICL plots. The higher value of the information scores indicates a better model.

Table 5.4 shows BIC, AIC and ICL values. The minimum BIC, AIC and ICL are 4696.8445, 2713.1946 and 4708.162. Figure 5.4 shows that BIC scores of the Gaussian kernel mixture models are larger than those of the gamma kernel mixture models, so the gamma kernel mixture model is consistently better than the Gaussian kernel mixture model. In addition, the AIC continuously decreases as the number of clusters increases, but the BIC and ICL are optimal when $G = 10$. However, there is no significant difference in these scores between the independence

and dependence assumptions, and most clustering results are similar for these two algorithms. Thus, the optimal number of clusters is ten, kernel function is gamma distribution, and the initial labels are given by the mclust clustering result of the death rate.

Table 5.5: Kernel mixture model clustering results.

Africa	Americas	Eastern Mediterranean	Europe	South-East Asia	Western Pacific
Algeria	Argentina	Afghanistan	Belarus	Bangladesh	Australia
Cameroon	Brazil	Bahrain	Belgium	Bhutan	China
Cote d'Ivoire	Canada	Egypt	France	India	Japan
DRC	Chile	Iran	Germany	Indonesia	Malaysia
Gabon	Colombia	Kuwait	Italy	Maldives	Mongolia
Ghana	Dominican Republic	Oman	Netherlands	Myanmar	New Zealand
Guinea	Ecuador	Pakistan	Russia	Nepal	Philippines
Nigeria	Mexico	Qatar	Spain	Sri Lanka	South Korea
Senegal	Peru	Saudi Arabia	UK	Thailand	Singapore
South Africa	USA	United Arab Emirates	Turkey	Timor-Leste	Viet Nam

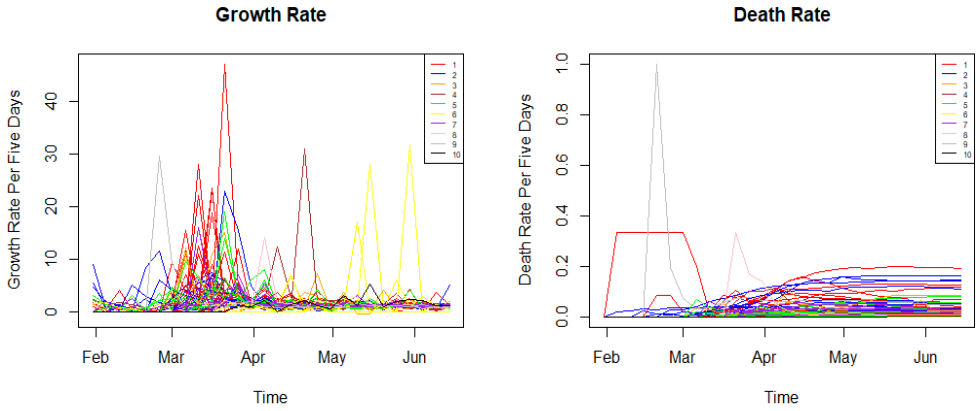

Figure 5.5: Clustering Results. Colors represents different clusters. The left panel is growth rate and the right panel is death rate.

Figure 5.4 and Table 5.5 show the clustering results. According to the left panel in Figure 5.5, the COVID-19 outbreak in the countries which are classified to cluster 1, cluster 3, cluster 5 and cluster 7 happened in March, but peak values are quite different in these clusters. The average of the peak values in cluster 1 is a maximum value of 15.2304, and the average in cluster 7 is a minimum value of 7.85. Similarly, the outbreaks in cluster 2 and cluster 9 happened in February, and peak values in cluster 9 are higher than values in cluster 2. The outbreaks in cluster 4 and cluster 9 happened in April, and the outbreaks in cluster 6 and cluster 10 happened in May. Peak values in cluster 6 are higher than cluster 10.

Based on the right panel in Figure 5.4, death rates in cluster 9 and cluster 8 have a similar pattern. The death rate reached the maximum value early in the outbreak, and then it decreased to a stable value. However, the time when the death rate reached the maximum value is different. Cluster 8 is later than cluster 9, since the date when the outbreak happened in cluster 8 is later than the date in cluster 9. In addition, there were no deaths until June 15 in cluster 6, so death rate equals zero in these four countries. Death rates in cluster 1 and cluster 2 are quite higher than those in other clusters, as the outbreaks happened in these countries earlier than in other countries. Overall, the kernel mixture model can cluster countries based on both outbreak time and peak values.

5.5 Comparison with Distance-based Clustering

In this section, kernel distance-based clustering will be applied to COVID-19 data and then we will make a comparison with the kernel mixture model.

Table 5.6: The top ten countries in six regions.

Africa	Americas	Eastern Mediterranean	Europe	South-East Asia	Western Pacific
Algeria	Argentina	Afghanistan	Belarus	Bangladesh	Australia
Cameroon	Brazil	Bahrain	Belgium	Bhutan	China
Cote d'Ivoire	Canada	Egypt	France	India	Japan
DRC	Chile	Iran	Germany	Indonesia	Malaysia
Gabon	Colombia	Kuwait	Italy	Maldives	Mongolia
Ghana	Dominican Republic	Oman	Netherlands	Myanmar	New Zealand
Guinea	Ecuador	Pakistan	Russia	Nepal	Philippines
Nigeria	Mexico	Qatar	Spain	Sri Lanka	South Korea
Senegal	Peru	Saudi Arabia	UK	Thailand	Singapore
South Africa	USA	United Arab Emirates	Turkey	Timor-Leste	Viet Nam

Table 5.6 and Figure 5.6 show the distance-based clustering results. Compared with Figure 5.5, countries where outbreaks happened in February and May are the same as countries given by the kernel mixture model, but this model cannot identify which countries had outbreaks in March and April. For death rates, this method clusters Iran and Gabon into the same group, as they have similar patterns.

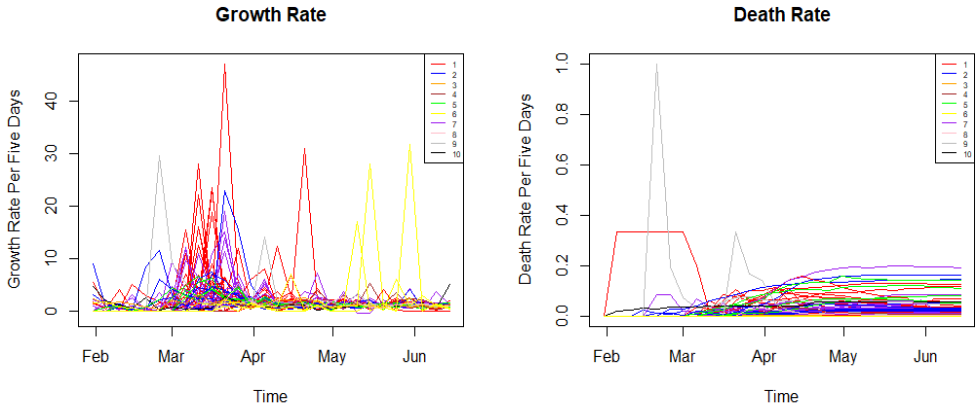

Figure 5.6: Kernel distance-based clustering results. Colors represents different clusters. The left panel is growth rate and the right panel is death rate.

Table 5.7: Cross-tabulation of mixture model clusters and distance-based clusters.

No. of clusters	1	2	3	4	5	6	7	8	9	10
A	4	0	2	0	3	0	4	0	0	0
B	16	1	3	2	2	0	1	0	0	0
C	0	6	0	0	1	0	0	0	0	0
D	0	0	0	0	0	3	0	0	0	0
E	0	3	0	0	1	0	0	0	0	0
F	0	1	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	1	1	0
H	0	0	0	2	0	0	0	0	0	1
I	0	0	0	1	0	0	0	0	0	0
J	0	0	0	0	0	1	0	0	0	0

We will use the ARI to measure the similarity between two clusters. As mentioned in Chapter 2, the closer the ARI is to 1, the more similar these two clustering results are. Based on Table 5.7, A∼J represent kernel mixture model clustering results and 1∼10 are kernel distance-based clustering results. An ARI equal to 0.3233 means the results from distance-based clustering have differences with the results from the mixture model. However, Figure 5.6 indicates that results from the kernel mixture model can reveal more details compared with the kernel distance-based model.

Chapter 6

Conclusions and Future Work

6.1 Conclusion

We proposed kernel mixture models to cluster univariate, independent multivariate and dependent bivariate longitudinal data by using a Fourier transform, a kernel estimation method and copulas. The Gaussian and gamma distributions were chosen as kernel functions, and the bivariate Gaussian copula was used as the dependence structure. In addition, we also proposed a kernel distance-based clustering method to make a comparison with the kernel mixture model.

The performances of the proposed methods were investigated through a simulation study. We found that the gamma kernel mixture models outperformed the Gaussian kernel mixture models. The BIC scores with copulas were smaller than scores without copulas, as the correlation became stronger. For univariate data sets, there were no significant differences in clustering results between the kernel mixture models and the kernel distance-based method, but the performance of the distance-based clustering deteriorated for bivariate longitudinal data. Compared with univariate clustering results, bivariate clustering results displayed that adding a useful attribute can improve accuracy rates, which means multivariate longitudinal data can provide more information for clustering.

Finally, we applied the kernel mixture model and distance-based method to COVID-19 data. Sixty countries were classified to ten clusters based on growth rates and death rates. The kernel mixture model considered the COVID-19 out-

break time and peak values for clustering, so we found that there were two clusters in each month, namely, high-peak and low-peak clusters. In addition, we also compared kernel mixture modesl and kernel distance-based clustering method. The ARI showed that the clustering results of the distance-based approach were different from the results of the mixture model, and the distance-based model did not distinguish countries which had outbreaks in March and April. Thus, the kernel mixture model appears more suitable for COVID-19 data.

6.2 Future Work

There are three natural extensions to this body of work. Firstly, different copulas, such as Clayton, Frank and Gumbel-Hougaard, can be used in the dependence algorithm. In this thesis, we used the bivariate Gaussian copula to fit the model for convenient application, but we could change the copulas to find the best copula function for a mixture model. Chen and Fan (2005), Grønneberg and Hjort (2014) and Huard et al. (2006) proposed to use the pseudolikelihood function for copula selection. Secondly, we can extend the bivariate longitudinal data to multivariate data by using multivariate copula functions. We have clarified that multiple attributes can provide more information for clustering, so an increase in the number of attributes is essential for kernel mixture models. Thirdly, kernel mixture models in this thesis are used for balanced longitudinal data, but unbalanced longitudinal data is more commonly found in practice. Unbalanced longitudinal data is a data set where not all individuals are observed at all time points. Thus, we could extend the kernel mixture model to fit unbalanced longitudinal data in the future.

Bibliography

Abbi, R., E. El-Darzi, C. Vasilakis, and P. Millard (2008). Analysis of stopping criteria for the EM algorithm in the context of patient grouping according to length of stay. In *2008 4th International IEEE Conference Intelligent Systems*, Volume 1, pp. 3–9–3–14.

Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pp. 199–213. Springer.

Baragona, R. (2001). A simulation study on clustering time series with metaheuristic methods. *Quaderni di Statistica 3*, 1–26.

Baron Fourier, J. B. J. (1878). *The analytical theory of heat*. The University Press.

Batista, G. E., E. J. Keogh, O. M. Tataw, and V. M. De Souza (2014). CID: An efficient complexity-invariant distance for time series. *Data Mining and Knowledge Discovery 28*(3), 634–669.

Berndt, D. J. and J. Clifford (1994). Using dynamic time warping to find patterns in time series. In *KDD workshop*, Volume 10, pp. 359–370. Seattle, WA, USA:.

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

Böhning, D., E. Dietz, R. Schaub, P. Schlattmann, and B. G. Lindsay (1994). The distribution of the likelihood ratio for mixtures of densities from the one-parameter exponential family. *Annals of the Institute of Statistical Mathematics 46*(2), 373–388.

Chen, S. X. (2000). Probability density function estimation using gamma kernels. *Annals of the Institute of Statistical Mathematics 52*(3), 471–480.

Chen, X. and Y. Fan (2005). Pseudo-likelihood ratio tests for semiparametric multivariate copula model selection. *Canadian Journal of Statistics 33*(3), 389–414.

Cooley, J. W. and J. W. Tukey (1965). An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation 19*(90), 297–301.

De la Cruz-Mesía, R., F. A. Quintana, and G. Marshall (2008). Model-based clustering for longitudinal data. *Computational Statistics and Data Analysis 52*(3), 1441–1457.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological) 39*(1), 1–22.

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

Dias, J. G. and M. Wedel (2004). An empirical comparison of EM, SEM and MCMC performance for problematic Gaussian mixture likelihoods. *Statistics and Computing 14*(4), 323–332.

Diebolt, J. and C. P. Robert (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society: Series B (Methodological) 56*(2), 363–375.

Falissard, L., G. Fagherazzi, N. Howard, and B. Falissard (2018). Deep clustering of longitudinal data. *arXiv preprint arXiv*: *1802.03212*.

Ferreira, L. N. and L. Zhao (2015). A time series clustering technique based on community detection in networks. *Procedia Computer Science 53*, 183–190.

Frühwirth-Schnatter, S. (2011). Panel data analysis: a survey on model-based clustering of time series. *Advances in Data Analysis and Classification 5*(4), 251–280.

Frühwirth-Schnatter, S. and S. Kaufmann (2008). Model-based clustering of multiple time series. *Journal of Business and Economic Statistics 26*(1), 78–89.

Genest, C. and A.-C. Favre (2007). Everything you always wanted to know about copula modeling but were afraid to ask. *Journal of Hydrologic Engineering 12*(4), 347–368.

Genest, C., K. Ghoudi, and L.-P. Rivest (1995). A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika 82*(3), 543–552.

Genolini, C., X. Alacoque, M. Sentenac, C. Arnaud, et al. (2015). kml and kml3d: R packages to cluster longitudinal data. *Journal of Statistical Software 65*, 1–34.

Golay, X., S. Kollias, G. Stoll, D. Meier, A. Valavanis, and P. Boesiger (1998). A new correlation-based fuzzy logic clustering algorithm for FMRI. *Magnetic Resonance in Medicine 40*(2), 249–260.

Grønneberg, S. and N. L. Hjort (2014). The copula information criterion. *Scandinavian Journal of Statistics 41*(2), 436–459.

Huard, D., G. Évin, and A.-C. Favre (2006). Bayesian copula selection. *Computational Statistics and Data Analysis 51*(2), 809–822.

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

Joe, H. (1997). *Multivariate Models and Multivariate Dependence Concepts*. New York: Chapman & Hall/CRC Press.

Joe, H. (2005). Asymptotic efficiency of the two-stage estimation method for copula-based models. *Journal of Multivariate Analysis 94*(2), 401–419.

Juárez, M. A. and M. F. Steel (2010). Model-based clustering of non-Gaussian panel data based on skew-t distributions. *Journal of Business and Economic Statistics 28*(1), 52–66.

Kalpakis, K., D. Gada, and V. Puttagunta (2001). Distance measures for effective clustering of ARIMA time-series. In *Proceedings 2001 IEEE International Conference on Data Mining*, pp. 273–280. IEEE.

Karlis, D. and E. Xekalaki (2003). Choosing initial values for the EM algorithm for finite mixtures. *Computational Statistics and Data Analysis 41*(3-4), 577–590.

Kaufmann, L. and P. Rousseeuw (1987, 01). Clustering by means of medoids. *Data Analysis based on the L1-Norm and Related Methods*, 405–416.

Kim, G., M. J. Silvapulle, and P. Silvapulle (2007). Comparison of semiparametric and parametric methods for estimating copulas. *Computational Statistics and Data Analysis 51*(6), 2836–2850.

Košmelj, K. and V. Batagelj (1990). Cross-sectional approach for clustering time varying data. *Journal of Classification 7*(1), 99–109.

Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern Recognition 38*(11), 1857–1874.

Lin, T. H. (2010). A comparison of multiple imputation with EM algorithm and MCMC method for quality of life missing data. *Quality and Quantity 44*(2), 277–287.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory 28*(2), 129–137.

Maharaj, E. A. (2000). Cluster of time series. *Journal of Classification 17*(2), 297–314.

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

McNicholas, P. D. (2016b). Model-based clustering. *Journal of Classification 33*(3), 331–373.

McNicholas, P. D. and T. B. Murphy (2010). Model-based clustering of longitudinal data. *Canadian Journal of Statistics 38*(1), 153–168.

McNicholas, P. D., T. B. Murphy, A. F. McDaid, and D. Frost (2010). Serial and parallel implementations of model-based clustering via parsimonious Gaussian mixture models. *Computational Statistics & Data Analysis 54*(3), 711–723.

Möller-Levet, C. S., F. Klawonn, K.-H. Cho, and O. Wolkenhauer (2003). Fuzzy clustering of short time-series and unevenly distributed sampling points. In *International Symposium on Intelligent Data Analysis*, pp. 330–340. Springer.

Nelsen, R. B. (2007). *An Introduction to Copulas*. Springer Science & Business Media.

Nie, G., Y. Chen, L. Zhang, and Y. Guo (2010). Credit card customer analysis based on panel data clustering. *Procedia Computer Science 1*(1), 2489–2497.

Piccolo, D. (1990). A distance measure for classifying ARIMA models. *Journal of Time Series Analysis 11*(2), 153–164.

Policker, S. and A. B. Geva (2000). Nonstationary time series analysis by temporal clustering. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 30*(2), 339–343.

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

Rydén, T. et al. (2008). EM versus Markov chain Monte Carlo for estimation of hidden Markov models: A computational perspective. *Bayesian Analysis 3*(4), 659–688.

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

Sklar, M. (1959). Fonctions de repartition an dimensions et leurs marges. *Publications de l'Institut de Statistique de l'Université de Paris 8*, 229–231.

Van Wijk, J. J. and E. R. Van Selow (1999). Cluster and calendar based visualization of time series data. In *Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis' 99)*, pp. 4–9. IEEE.

Vermunt, J. K. (2010). Longitudinal research using mixture models. In *Longitudinal Research with Latent Variables*, pp. 119–152. Springer.

Wu, C. J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics 11*(1), 95–103.

Xu, Y., A. Zafirov, R. M. Alvarez, D. Kojis, M. Tan, and C. M. Ramirez (2020). FREEtree: A tree-based approach for high dimensional longitudinal data with correlated features.