6-2022

Mixture Of Functional Graphical Models

Qihai Liu
Western Michigan University

Follow this and additional works at: https://scholarworks.wmich.edu/dissertations

Part of the Statistics and Probability Commons

Recommended Citation
https://scholarworks.wmich.edu/dissertations/3857

This Dissertation-Open Access is brought to you for free and open access by the Graduate College at ScholarWorks at WMU. It has been accepted for inclusion in Dissertations by an authorized administrator of ScholarWorks at WMU. For more information, please contact wmu-scholarworks@wmich.edu.
With the development of data collection technologies that use powerful monitoring devices and computational tools, many scientific fields are now obtaining more detailed and more complicedly structured data, e.g., functional data. This leads to increasing challenges of extracting information from the large complex data. Making use of these data to gain insight into complex phenomena requires characterizing the relationships among a large number of functional variables. Functional data analysis (FDA) is a rapidly developing area of statistics for data which can be naturally viewed as a smooth curve or function. It is a method that changes the frame of data and thus the fundamental statistical unit is now a function or curve, other than the vector of measurements. Graphical models have been widely used to explicitly capture the statistical relationships between the variables of interest in the form of a graph. The central question in these models is to infer significant conditional dependencies or independencies from high-dimensional data. In the current literature, it is common to assume that the high-dimensional data come from a homogeneous source and follow a parametric or semi-parametric graphical model. However, in real-world context
the observed data often come from different sources and may have heterogeneous dependencies across the whole population. Therefore, a single functional graphical model is no longer adequate for the data. As finite mixture models offer powerful statistical techniques to identify subpopulations with certain commonality within an overall population from heterogeneous sources, one solution to this issue may be the application of mixture analysis techniques in functional graphical models.

As a part of such effort, a functional graphical model is developed to extract the conditional dependence structure among random functions. In this dissertation, we propose the mixture of functional graphical models (MFGM), which detects the heterogeneous subgroups of the population and estimates the conditional dependencies in each subgroup. We also introduce an estimation method for MFGM using an iterative Expectation-Maximization (EM) algorithm.

The performance of our algorithm on the overall clustering accuracy and accuracy of the estimation for the conditional dependence structures in the heterogeneous subgroups is shown through the simulation studies. Our MFGM algorithm outperforms the two potential competing algorithms: the alternating direction method of multipliers (ADMM) algorithm provided by the R fgm package that assumes partial separability in the proposed functional Gaussian graphical models, and the mixggm algorithm that implements mixture of Gaussian graphical models in multivariate vector setting. The application to high-dimensional electroencephalography (EEG) dataset taken from an alcoholism study is also discussed. The results from the real data analysis also
corroborate the performance of our MFGM algorithm.

Our work, motivated by inferring heterogeneous conditional dependencies of high-dimensional data, may greatly extend the methodology and applicability of high-dimensional graphical models, and provide a novel strategy for complex functional data analysis.
MIXTURE OF FUNCTIONAL GRAPHICAL MODELS

by

Qihai Liu

A dissertation submitted to the Graduate College
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
Statistics
Western Michigan University
June 2022

Doctoral Committee:

Hyun Bin Kang, Ph.D., Chair
Kevin H Lee, Ph.D.
Joshua Naranjo, Ph.D.
Sangwoo Lee, Ph.D.
Copyright by
Qihai Liu
2022
ACKNOWLEDGMENTS

First and foremost I am extremely grateful to the chair of my doctoral committee, Dr. Hyun Bin Kang, and my co-supervisor, Dr. Kevin Lee. In the past three years, they showed really keen insight and provided invaluable advice, continuous support and patience for my PhD study. I am really lucky and honored to have so charming supervisors, and I attribute all my improvements at WMU to them.

Secondly, I would like to thank my committee members, Prof. Joshua Naranjo and Prof. Sangwoo Lee, for their valuable comments and suggestions that greatly improved the quality of my dissertation. Meanwhile, my gratitude extends to Prof. Joseph McKean and Dr. Duy Ngo, the courses provided by whom broadened my horizons and enhanced my understanding of statistics.

I would also like to express gratitude to Prof. Magdalena Niewiadomska-Bugaj, our former department chair. I will never forget how excited I was when Prof. Bugaj kindly offered me an opportunity to work as TA for our department; I will never forget how much inspired I was when she encouraged me to finish a PhD study in our department. I appreciate all of her trust. She changed my life. She is my lifelong mentor.

Finally and most importantly, I would like to thank Dr. Joshua Li and Dr. Li
Jin, my former supervisors at University of Virginia. I will always keep in mind their kind greetings and tireless encouragement during the hardest times of my PhD study at WMU. I could not achieve anything here without their help. They deserve all my respect and gratitude.

Qihai Liu
Table of Contents

ACKNOWLEDGMENTS .................................................. ii

LIST OF TABLES ................................................... vii

LIST OF FIGURES .................................................. viii

1 Introduction ....................................................... 1

1.1 Motivation ...................................................... 1

1.2 Contribution ................................................... 2

1.3 Organization ................................................... 4

2 Background and Literature Review ............................. 6

2.1 Functional Data Analysis ..................................... 6

2.1.1 Data Registration or Feature Alignment ................. 7

2.1.1.1 Shift Registration .................................. 8

2.1.1.2 Landmark Registration with Warping Function .... 9

2.1.2 Basis Expansion .......................................... 10

2.2 Functional Principal Component Analysis ................. 13

2.2.1 Mercer’s Theorem ........................................ 13

2.2.2 Karhunen-Loève Expansion ............................ 14
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3 Functional Regression</td>
<td>15</td>
</tr>
<tr>
<td>2.4 Functional Graphical Models</td>
<td>16</td>
</tr>
<tr>
<td>2.5 Mixture of Functional Graphical Models</td>
<td>19</td>
</tr>
<tr>
<td>3 Mixture of Functional Graphical Models</td>
<td>21</td>
</tr>
<tr>
<td>3.1 Functional Data Smoothing</td>
<td>21</td>
</tr>
<tr>
<td>3.2 Functional Graphical Models</td>
<td>22</td>
</tr>
<tr>
<td>3.3 Mixture of Functional Graphical Models</td>
<td>26</td>
</tr>
<tr>
<td>3.4 Computation</td>
<td>27</td>
</tr>
<tr>
<td>3.5 Algorithmic Convergence</td>
<td>30</td>
</tr>
<tr>
<td>4 Simulation Studies</td>
<td>35</td>
</tr>
<tr>
<td>4.1 Functional Graphical Models</td>
<td>36</td>
</tr>
<tr>
<td>4.2 Mixture of Functional Graphical Models</td>
<td>38</td>
</tr>
<tr>
<td>4.2.1 Two-Cluster Mixture Models</td>
<td>38</td>
</tr>
<tr>
<td>4.2.2 Three-Cluster Mixture Models</td>
<td>38</td>
</tr>
<tr>
<td>4.3 Data Analysis of Simulated Mixture of Functional Graphical Models</td>
<td>39</td>
</tr>
<tr>
<td>4.3.1 Two-Cluster Mixture Models Analysis</td>
<td>41</td>
</tr>
<tr>
<td>4.3.2 Three-Cluster Mixture Models Analysis</td>
<td>43</td>
</tr>
<tr>
<td>5 Real Data Analysis</td>
<td>45</td>
</tr>
<tr>
<td>5.1 EEG Data and Alcoholism</td>
<td>45</td>
</tr>
<tr>
<td>5.2 Results</td>
<td>48</td>
</tr>
<tr>
<td>6 Conclusion and Future Work</td>
<td>52</td>
</tr>
<tr>
<td>6.1 Conclusion</td>
<td>52</td>
</tr>
</tbody>
</table>
6.2 Future Work ......................................................... 54
List of Tables

4.1 Definitions of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). ................................................... 41
4.2 Comparisons of MFGM with ADMM and mixggm in terms of overall clustering in the two-cluster mixture simulation studies. ............... 42
4.3 Comparisons of MFGM with ADMM and mixggm in terms of estimates of edge structures in subgroups in the two-cluster mixture simulation studies. .......................................................... 43
4.4 Comparisons of MFGM with ADMM and mixggm in terms of overall clustering in the three-cluster mixture simulation studies. ............... 43
4.5 Comparisons of MFGM with ADMM and mixggm in terms of estimates of edge structures in subgroups in the three-cluster mixture simulation studies. .......................................................... 44
5.1 Comparisons of MFGM with ADMM and mixggm in terms of overall clustering in EEG data analysis. ................................................. 48
5.2 Overall clustering of EEG by MFGM (Left), ADMM (Middle), and mixggm (Right). ................................................................. 49
List of Figures

2.1 Randomly selected ten observations in the raw data for medfly laying eggs in their first twenty-five days (A); and cubic B-spline smoothing with basis number 5 performed on the raw data (B). . . . . . . . . . . 12

4.1 Covariance structures of $\mathbb{R}^{Lp}$-valued random coefficients from different $L$-truncated Karhunen-Loève type expansions. (a) and (b): covariance and precision matrices, respectively, of functional principal component coefficients $(\xi_1^\top, \ldots, \xi_p^\top)\top$ as in Qiao et al. (2019). (c) and (d): block diagonal covariance and precision matrices, respectively, of functional principal component coefficients $(\theta_1^\top, \ldots, \theta_L^\top)\top$ under partial separability as in Zapata et al. (2019). . . . . . . . . . . . . . . . . . . . . . . . . . 35

4.2 Conditional dependencies in each simulated functional graphical model. Thickness of the edge denotes strength of connection. . . . . . . . . 37

5.1 The 19-channel montage specified according to the 10–20 International system (indicated by blue nodes). . . . . . . . . . . . . . . . . . . . 47
5.2 Comparisons of conditional dependence structures in Alcoholic and Control subgroups estimated by MFGM, ADMM, and mixggm algorithms. Solid edges in black color: common in both of Alcoholic and Control; dash-dotted edges in red color: specific in Alcoholic; dash-dotted edges in blue color: specific in Control.
Chapter 1

Introduction

1.1 Motivation

Functional data analysis (FDA) is a rapidly developing area of statistics for data that can be naturally viewed as a smooth curve or function. It is a method that changes the frame of data and thus the fundamental statistical unit is now a function or curve, other than the vector of measurements (Ramsay and Silverman, 2007, 2008). With the development of data collection technologies that use powerful monitoring devices and computational tools, many scientific fields are now obtaining more detailed and more complicatedly structured data (Ferraty, 2014). Making use of these data to gain insight into complex phenomena requires characterizing the relationships among a large number of variables (Uhler, 2017). Graphical models have been widely used to explicitly capture the statistical relationships between the variables of interest in the form of a graph. The central question in these models is to infer significant conditional dependencies or independencies from high-dimensional data (Qiao et al., 2019). In the current literature, it is common to assume that the high-dimensional data come from a homogeneous source and follow a parametric or semi-parametric graphical model (Yuan
and Lin, 2007; Friedman et al., 2008; Meinshausen and Bühlmann, 2006). However, in real-world scenarios the observed data often come from different sources and may have heterogeneous dependencies across the whole population. Therefore, a single functional graphical model is no longer adequate for the data. As finite mixture models offer powerful statistical techniques to identify subpopulations with certain commonality within an overall population from heterogeneous sources, one solution to this issue may be the application of mixture analysis techniques in functional graphical models. Our motivated example is a high-dimensional electroencephalography (EEG) dataset taken from a large study that examined EEG correlates of genetic predisposition to alcoholism (https://kdd.ics.uci.edu/databases/eeg/eeg.html) (Zhang et al., 1995). The subjects from alcoholic group and control group were exposed to visual stimuli and their EEG activities were recorded from 64 electrodes placed on the scalps. We aim to detect heterogeneous subgroups and infer the human brain network structures in each subgroup by analyzing the multivariate functional EEG data.

### 1.2 Contribution

Mixture of graphical models have been well shown powerful and efficient to identify subpopulations that share hidden commonality and further depict heterogeneous conditional dependencies across the whole population in vector-valued scenarios, but very few studies have been carried out to explore the applicability of mixture of graphical models in functional context. In this dissertation, we propose the mixture of functional graphical models (MFGM) and provide an estimation method for the model.

Our estimation algorithm involves performing functional principal component analysis (FPCA) on the multivariate functional data. That is, we decompose each functional variable in a multivariate setting by truncated Karhunen-Loève expansion for dimension
reduction. Then the functional graphical lasso (fglasso) algorithm (Qiao et al., 2019) is employed to estimate the conditional dependence structures in each subpopulation.

To generalize the functional graphical models to mixture scenarios, one challenge is solving the maximization problem of log-likelihood with penalty, which is non-convex and non-smooth, to estimate graphical model parameters for each subgroup. As the Expectation–Maximization (EM) algorithm provides a powerful tool to solve latent variable problems in mixture models, we introduce the iterative EM process to estimate the parameters for mixture of graphical models. In the EM framework, the mixture problem is formulated as an incomplete-data problem. It is critical to design an effective and efficient EM algorithm with theoretically guaranteed ascent property and convergence to a stationary point under high dimensions. The optimization of the EM algorithm is explored in this work.

We also provide a tuning parameter selection criterion. As discussed in Qiao et al. (2019), to choose the tuning parameter for the penalty term in the maximization problem, there exist a number of possible approaches such as Akaike information criterion (AIC), Bayesian information criterion (BIC), and others. However, given the complicated functional structure of functional graphical models, it is unclear how to compute the effective degrees of freedom for AIC/BIC, let alone the more complicated MFGM studied in this work. We propose the cross-validation score criterion that is reliable for the selection of optimal tuning parameters in the MFGM context.

As revealed by the comparisons with the two potential competing algorithms ADMM and mixggm in both of simulation studies and real data analysis, our work proposes a method with high efficiency to detect heterogeneous subgroups from the whole population and estimate heterogeneous conditional dependence structures across the whole population at the same time. The algorithm with high efficiency to analyze
multivariate functional data of heterogeneity explored by our study would greatly extend the methodology and applicability of high-dimensional mixture of graphical models, and provide a novel strategy for complex functional data analysis.

1.3 Organization

The dissertation is organized as follows. In Chapter 2, we review literature related to the research background of this study, including FDA, graphical models, functional graphical models, and mixture of graphical models. We also discuss the significance of our proposed model which extends mixture of graphical models from finite vector-valued context to infinite functional context.

In Chapter 3, we present the methodologies for MFGM. We first introduce the B-spline basis decomposition for functional data smoothing. Then we introduce functional principal component analysis (FPCA), the truncated Karhunen-Loève expansion that is employed to reduce the dimension of multivariate functional data from infinite to finite scale, and the cross-validation score criterion to find the proper number of functional principal components. Next we present the computation to implement the proposed MFGM. First, we introduce the fglasso algorithm that estimates the conditional dependence structures of Gaussian functional graphical models. Then we propose the iterative EM process to solve the maximization problem that estimates the parameters for mixture of Gaussian functional graphical models. In addition, we introduce the cross-validation score criterion for the selection of optimal tuning parameters for the penalty term in the maximization problem in mixture of functional model settings. In the end, the algorithmic convergence is proved.

In Chapter 4, we apply our proposed MFGM algorithm in simulation studies with two- and three-cluster mixture multivariate functional designs, and the performance of
our algorithm is evaluated by checking the overall clustering accuracy and accuracy of the estimation for the conditional dependence structures in the heterogeneous subgroups. Comparisons with the two competing algorithms ADMM and mixgmg are performed.

In Chapter 5, we apply our proposed MFGM algorithm in the analysis of real-world data, the high-dimensional EEG dataset taken from a large study that examined EEG correlates of genetic predisposition to alcoholism. The subjects from alcoholic group and control group were exposed to visual stimuli and their EEG activities were recorded from electrodes placed across the scalps. We aim to detect heterogeneous subgroups and infer the human brain network structures in each subgroup by analyzing the multivariate functional EEG data with our algorithm. Comparisons with the two competing algorithms ADMM and mixgmg are performed.

In Chapter 6, we draw conclusion for our current work, and discuss future work that may improve our study.
Chapter 2

Background and Literature Review

2.1 Functional Data Analysis

Functional data analysis (FDA) is a collection of statistical techniques which analyzes data that provide information in the form of functions, images, shapes, and more (Wang et al., 2016). The basic philosophy of FDA is to consider the practical discrete measurements as continuous functions of infinite dimensions instead of treating them as a set of vectors. Ramsay, Dalzell, and Silverman made fundamental contributions to FDA (Ramsay and Dalzell, 1991; Ramsay and Silverman, 2007, 2008). They described the reasons for performing FDA as follows: 1) smoothing and interpolation procedures can yield functional representations of finite sets of observations; 2) it is more natural to think through in functional terms even though only finite numbers of observations are available; 3) the objectives of an analysis can be functional in nature, as would be the case if finite data are used to estimate an entire function, its derivatives, or the values of other functionals; 4) taking considerations such as smoothness into account for multivariate data arising from functional processes can have important implications for their analyses. They also summarized the goals of FDA as follows: 1) to present
data in ways that help further analysis; 2) to display data in ways that highlight various characteristics; 3) to study important sources of pattern and variation among the data; 4) to explain variation in an outcome or dependent variable by using input or independent variable information.

Since its foundation, FDA has shown a wide range of applications in different fields of sciences including epidemiology to help forecast measles (Kowal, 2019) and to identify patterns of malaria incidence (Dieng et al., 2020), meteorology to study the spatiotemporal variability of particulate matter components (King et al., 2018), biomechanics to study back pain (Page et al., 2006), knee osteoarthritis (Deluzio and Astephen, 2007), as well as age, gender, and speed effects on gait (Røislien et al., 2009), neuroimaging for positron emission tomography (PET), functional magnetic resonance imaging (fMRI) (Tian, 2010), EEG data analysis to study the relationship between brain and mind (Hasenstab et al., 2017), and many others. The abundant applications presented the features of FDA: based on smoothing, aligning, dimension reduction or testing effects on outcomes as functions, certain goals of FDA can be exploratory, confirmatory or predictive. The methodologies and theories to support these applications were detailed in the books by Ramsay and Silverman (2007) and Horváth and Kokoszka (2012).

### 2.1.1 Data Registration or Feature Alignment

Random functions typically contain both phase (horizontal) and amplitude (vertical) variability. Phase variation is a variation in the location of curve features along the horizontal axis, a phenomenon which is opposed to the height or amplitude variation. The presence of phase variation artificially often inflates data variance, blurs underlying data structures, and distorts principal components (Marron et al., 2015). Ignorance
of phase variation entails a loss of structure in the data and inefficiency in data models since representative curve features are dampened in a group average (Ramsay and Silverman, 2007; Robertson et al., 2013). Therefore, the separation/removal of phase variation from amplitude variation, which accomplishes a similar goal as time normalization, has always been desirable in FDA. Registration of the functional curves transforms their arguments rather than the values to help align the corresponding peak locations of the curves. In other words, it is used to reduce phase variability between curves while preserving the shape and amplitude of the individual curves.

2.1.1.1 Shift Registration

Some functional observations must be aligned by moving each curve horizontally such that any meaningful cross-curve analysis is possible. This often happens when the time at which the recording process begins is arbitrary, and is unrelated to the beginning of the interesting segment of the data. Let the interval $\mathcal{T}$ over which the functions $x_i$ are to be registered be $[T_1, T_2]$. The values $x_i^*(t) = x_i(t + \delta_i)$ are of interest, where the shift parameter $\delta_i$ is chosen in order to appropriately align the curves. Estimation of the shift parameter can be characterized as either fixed effects or random effects. The estimation requires a criterion that defines when several curves are properly registered. One possibility is to identify a specific feature or landmark for a curve, and shift each curve so that this feature occurs at a fixed point in time. Least squares is a common criterion for shift registration. For example, a mean function $\hat{\mu}(t)$ is first estimated by the sample average $\bar{x}$. The least squares criterion is minimized with respect to $\delta_i$. This process is iterated by re-computing the mean function $\hat{\mu}(t)$ from the registered curves $x_i^*(t)$, and re-computing a new set of shifts $\delta_i$, until the convergence is reached.
2.1.1.2 Landmark Registration with Warping Function

A landmark is a feature with a location that is identifiable and shared across all curves. Landmarks are typically extrema, inflection points, and so on. The landmark registration process requires for each curve $x_i$ the identification of the argument values $t_{if}, f = 1, \ldots, F$, associated with each of the $F$ features. The goal is to construct a transformation function $h_i$ for each curve such that the registered curves with values $x^*(t) = x_i[h_i(t)]$ have more or less identical argument values for any given landmark. To calculate the registered function values $x^*(t) = x[h(t)]$, first, estimate the inverse warping function $h^{-1}(t)$ with the property $h^{-1}[h(t)] = t$. The inverse function $h^{-1}(t)$ is computed by smoothing or interpolating the relationship between $h(t)$ plotted on the horizontal axis and $t$ plotted on the vertical axis. Then the simple interpolation is used to get the values of this inverse function at an equally spaced set of values of $t$ if required. As it is essential that this smoothing or interpolation function should be strictly monotonic, lots of values of $t$ have to be used and/or the monotone smoothing has to be employed. The second step is to smooth or interpolate the relationship between $h^{-1}(t)$ plotted on the horizontal axis and $x(t)$ plotted on the vertical axis. Simple interpolation can be used to get the values of this registered function at an equally spaced set of values of $t$ if required. As introduced by Kneip and Gasser (1992) and Gasser and Kneip (1995), special features, such as the peak locations in functions or derivatives, are aligned to their average location, and then smooth transformations from the average location to the location of the feature for a specific subject are implemented. For example, in human growth, the biological age of different children varies. This variation has a direct bearing on the growth rate, which follows a general shape with subject-specific timing of the two major growth spurts, the prepubertal and pubertal growth spurts (Gasser et al., 1984). In a study to analyze the growth
data, the functional curves were registered by using the zero-crossing in the middle of the pubertal growth spurt as a single landmark. Then the curves resulting from this preliminary registration were used as inputs to a continuous registration (Ramsay and Silverman, 2008). In another study, Bigot proposed a fast and automatic nonparametric landmark registration method based on the alignment of the structural intensity of the zero-crossings of a wavelet transform, to align the significant landmarks of a set of noisy signals (Bigot, 2006). To point out, landmark registration requires clearly identifiable landmarks and manual care in defining and finding landmarks.

2.1.2 Basis Expansion

In effect, basis expansion represents the infinite-dimensional functions within the finite-dimensional space. Due to the dimension of the expansion, a functional dataset reduces to a vector space instead of a functional space with ideally appropriate basis functions. Once basis functions are well estimated from the observed signals, a linear approximation is derived and certain characteristics of the signals are then included in the coefficients, which become appropriate descriptive variables of the signals. The number of the basis functions is regarded as a tuning parameter defined according to the characteristics of the data. The first step in FDA is to perform smoothing that converts the raw discrete observation points into a smoothly varying function. This smoothing process emphasizes patterns in the data by minimizing short-term deviations due to observational errors, such as measurement errors or inherent system noise (Ullah and Finch, 2013). Figure 2.1 (A) shows a typical example of functional data that is a subset of the Mediterranean fruit flies laying eggs data (Carey et al., 1998). The association of the individual mortality and longevity with the time-dynamics of the egg-laying trajectory is explored in the study. The eggs laid for each medfly
are counted on a daily basis. Therefore, smoothing process must be performed to represent the functional trajectories before further analysis. Basis expansion, expanding the functions using basis functions, is one of the common nonparametric smoothing methods that represent the potentially infinite-dimensional world of functions within the finite-dimensional framework of vectors. The advantage of the basis expansion approach is that nonlinearity and local features in the data can be easily absorbed by the basis functions while the model is still linear in the transformations. As reviewed by Ullah and Finch (2013), B-spline expansion has been shown to be the most popular smoothing technique ever used, presumably due to its simplicity and flexibility for tackling a wide range of nonparametric and semiparametric modeling situations. B-splines are constructed from polynomial pieces, joined at certain values of the knots. Once the knots are given, it is easy to compute the B-splines recursively, for any desired degree of the polynomial (Eilers et al., 1996). Large number of knots are chosen to reduce the effective degrees of freedom and increase smoothness in the overall function estimate. Figure 2.1 (B) shows cubic B-spline smoothing with basis number 5 performed on the raw medfly laying eggs data to denote the inherent functional features. Fourier-basis expansion has been the second most popular smoothing method. The Fourier basis functions are sine and cosine functions of increasing frequency, and therefore are especially useful for periodic data smoothing, where the temporal pattern is stable. Wavelet basis functions are localized in both frequency and time domains simultaneously, which allows for the extraction of features that are less smooth from temporal data (Vidakovic, 2009). The wavelet bases are chosen to represent data displaying discontinuities and/or rapid changes in behavior (Ruppert et al., 2003; Simonoff, 2012). Wavelet basis function decomposition has also been successfully applied to the analysis of fetal movement monitoring data (Reislien and Winje, 2013).
Other basis functions include step functions, polygons, exponential functions, \textit{et al.}

Figure 2.1: Randomly selected ten observations in the raw data for medfly laying eggs in their first twenty-five days (A); and cubic B-spline smoothing with basis number 5 performed on the raw data (B).

The discrete observations $y_i, \ i = 1, \ldots, N,$ are represented as $y_i = x_i(t_i) + \epsilon_i, \ t_i \in \mathcal{T},$ where $\epsilon_i$ are the error terms, and the random functions $x_i(t)$ are decomposed by a basis function expansion:

$$x_i(t) \approx \sum_{k=1}^{K} c_{ik} \phi_k(t), \ i = 1, \ldots, N, \ t \in \mathcal{T}.$$  

A common way to estimate the coefficients $c_{ik}$ is via penalized least squares method by minimizing

$$\sum_{i=1}^{n} \left( y_i - \sum_{k=1}^{K} c_{ik} \phi_k(t_i) \right)^2 + \lambda \int \left( L \left[ \sum_{k=1}^{K} c_{ik} \phi_k(t) \right] \right)^2 dt,$$

where $L$ is an operator that measures the roughness of functions, e.g., typically, taking the second derivative of the functions, and $\lambda$ is the smoothing parameter which governs the trade-off between the fit to the measurements and the smoothness of resulting functional objects. The optimal $\lambda$ can be chosen from a grid search that refers to cross-validation (CV), generalized cross-validation (GCV), AIC, BIC, \textit{et al.} Other ways
to estimate the basis coefficients, such as weighted least squares, localized least squares, are detailed in the book by Ramsay and Silverman (2008).

### 2.2 Functional Principal Component Analysis

Principal component analysis (PCA) is a statistical methodology used to reveal the internal structure of the data in order to explain variability (Jolliffe and Cadima, 2016). Functional principal component analysis (FPCA) is a generalization of PCA from multivariate data analysis to functional context. It has been the most prevalent tool in FDA for dimension reduction. Functional data consist of multiple series of observations with an underlying correlation structure. FPCA explores the covariance structure of the functional objects, and identifies functional principal components that explain the most variability of a sample of curves. In addition, FPCA expands the functions with their orthonormal eigenfunctions, which is analogous to the decomposition with orthogonal eigenvectors in multivariate case. The truncated expansion with eigenfunctions reduces the dimension of functions, as the multivariate PCA is used for linear dimension reduction.

#### 2.2.1 Mercer’s Theorem

In FDA, Mercer’s theorem is a representation of a symmetric positive-definite function on a square as a sum of a convergent sequence of product functions. It is a continuous analog of the singular-value or eigenvalue decomposition of a symmetric positive-definite matrix. Suppose $\Sigma(s, t)$ is a symmetric, continuous, and nonnegative
definite kernel function. Mercer’s theorem provides the spectral decomposition of $\Sigma$ as

$$\Sigma(s, t) = \sum_{k=1}^{\infty} \tau_k v_k(t) v_k(s),$$

where $\tau_k$’s are eigenvalues in descending order and $v_k$’s are the corresponding orthonormal eigenfunctions.

### 2.2.2 Karhunen-Loève Expansion

The Karhunen-Loève theorem (Karhunen, 1946; Loève, 1946) is a representation of a stochastic process as an infinite linear combination of orthogonal functions. Karhunen-Loève expansion decomposes the stochastic process into a series of orthogonal functions with the random coefficients. Suppose that $X(t)$ is a stochastic process for $t$ in some interval $[a, b]$. The process is often characterized by its mean $\mu(t)$, and its covariance $\Sigma(s, t)$. Under mild assumptions, with Mercer’s theorem applied on $\Sigma$, the process can be expressed by the Karhunen-Loève expansion

$$X(t) = \mu(t) + \sum_{k=1}^{\infty} \lambda_k v_k(t),$$

where $v_k$’s are the orthonormal eigenfunctions and $\lambda_k$’s are the corresponding eigenscores. The eigenscores are pairwise uncorrelated random variables. If the process is Gaussian, then the eigenscores are Gaussian and stochastically independent. The polygonal, B-spline, Fourier, and wavelet bases, et al., can be employed to decompose each eigenfunction and further to estimate the eigencomponents. In practical applications, truncated Karhunen-Loève expansions are implemented to convert the inherently infinite-dimensional functional data to a finite-dimensional vector of random scores. The choice of the optimal number of components $K$ for the truncation of Karhunen-
Loève expansion, which gives the best trade-off between bias and variance, has been an open question. Besides scree plot or the fraction of variance explained by the first few principal components, the criteria such as AIC and BIC, as well as the CV method, have been widely used for the selection of $K$. In terms of applications, the Karhunen-Loève expansion has been explored for both of densely observed FDA and much more difficult situation of sparse FDA (Wang et al., 2016).

2.3 Functional Regression

Functional regression is an active area of research, and the functional regression models are developed to explore the relationship between functional data and other variables. The model structure changes depending on whether the functional observations are the predictors or the responses.

Scalar-on-function regression is when the functional data are included as the predictors or the covariates and the response variable is scalar.

$$Y_i = \int_T \beta(t)X_i(t) \, dt + \epsilon_i.$$  

Function-on-scalar regression is when the functional data are included as the response while the predictors are scalar.

$$Y_i(t) = \sum_{p=1}^P x_{ip}\beta_p(t) + \epsilon_i(t).$$

Function-on-function regression also follows the same rule. This is when both the
response and the predictor are functional data.

\[ Y_i(t) = \int_T \beta(t,s)X_i(s) \, ds + \epsilon_i(t). \]

**Concurrent linear model** is a special case in which both the response and the predictor are functional data. The model looks like

\[ Y_i(t) = \beta_0(t) + \beta_1(t)X_i(t) + \epsilon_i(t). \]

In this case, \( Y_i \) at time \( t_j \) is affected only by \( X_i \) at the same time \( t_j \) whereas in function-on-function regression model, \( X_i \) at time \( t_j \) can affect \( Y_i \) at all time points.

### 2.4 Functional Graphical Models

Graphical models provide a powerful tool to describe statistical relationships between variables of interest in the form of a graph. In a graphical model, the conditional dependence structure among the components of a multivariate random vector is depicted. Let \( \mathbf{X} = (X_1, \ldots, X_p) \) be a \( p \)-dimensional random vector. Let \( G = (V, E) \) be an undirected graph, where \( V = \{1, \ldots, p\} \) represents the set of vertices corresponding to \( p \) random variables, and \( E = \{(i,j) \in V \times V \, : \, i \neq j\} \) represents the set of undirected edges. The edges describe the conditional dependence structure of the \( p \) variables, i.e., nodes \( i \) and \( j \) are connected by an edge if and only if \( X_i \) and \( X_j \) are dependent, conditional on the other \( p-2 \) variables. The absence of an edge between two nodes means the corresponding random variables are conditionally independent, given other variables. Even though the Gaussian assumption is very restrictive, the Gaussian distribution is widely used for such graphical models due to its convenient
analytical properties. It has been shown that, if $X$ follows a multivariate Gaussian distribution, estimating the edge set is equivalent to identifying the locations of the nonzero elements in the precision matrix, i.e., the inverse covariance matrix of $X$ (Lauritzen, 1996; Uhler, 2017). Hence, the Gaussian graph model describes the sparsity pattern of the precision matrix. Penalized regression methods for inducing sparsity in the precision matrix, which have been well studied in both low- and high-dimensional settings, are central to the construction of Gaussian graphical models. Yuan and Lin (2007) and Friedman et al. (2008) considered methods that optimize the graphical lasso (glasso) criterion, a maximum likelihood approach with the addition of lasso type penalty imposed on the off-diagonal entries of the precision matrix. This glasso algorithm and its subsequent improvements (Witten et al., 2011; Hsieh et al., 2013) are widely used to solve the problem efficiently. Under Gaussianity, Lam and Fan (2009) studied the sparsistency and rates of convergence for estimating sparse covariance and precision matrices with smoothly clipped absolute deviation (SCAD) and adaptive lasso penalty. Moreover, Li and Gui (2006) introduced a threshold gradient descent (TGD) regularization procedure for estimating the sparse precision matrix in the setting of Gaussian graphical models. Besides focusing on the estimation of precision matrix, the developments on Gaussian graphical models include neighborhood selection approach that estimates the neighborhood of each node separately by solving lasso linear regression problems for each node, and then stitches the neighborhoods together to form the global graph estimate, i.e., estimates which components are zero, rather than fully estimates precision matrix (Meinshausen and Bühlmann, 2006); non-zero partial correlations selection to estimate a sparse Gaussian graphical model by imposing the lasso penalty on the partial correlations, based on a relation between partial correlation and regression coefficient (Peng et al., 2009); and many others.
Frequently in real-world applications, unlike the static vector-valued graphical models, the nodes in the models are realized by random functional processes that vary over a continuum such as a time interval or a spatial domain. The conditional dependence structure of functional data is of interest in a wide range of applications. For example, in neuroimaging, the dependence networks across brain regions are modeled, where data for each region are of functional forms, e.g., EEG signals (Qiao et al., 2019), fMRI signals (Li and Solea, 2018); in bioinformatics, the gene networks are fitted based on time-course gene expression data (Wang et al., 2005), with each time-course being treated as a continuous process. Although methods for vector-valued data are well established, the generalization to functional data remains relatively underdeveloped, since in functional graphical model problems, not only can the number of functions measured per sample be large, but also each function is itself a high-dimensional object, thus making the estimation of model parameters challenging. In functional data, unlike the finite-dimensional case, the covariance operator is compact and thus not invertible, with the consequence that the connection between conditional independence and an inverse covariance operator is lost, as the latter does not exist (Zapata et al., 2019). As discussed in Qiao et al. (2019), one possible approach to handling this sort of functional data would be to sample the functions over a grid of time points, estimate separate networks for each time point, and then either report all networks or construct a single graphical model by somehow merging all the networks. There are some drawbacks for this strategy: 1) the time points for each functional observation may differ; 2) simultaneously interpreting all different networks would significantly increase the complexity of the dependence structure; 3) each of the networks would only correspond to dependencies among the functions at a common time point, however, it seems likely that some dependencies may only exist at different time points. To
overcome these drawbacks, under the assumption that the random processes observed at the vertices are Hilbert-space-valued Gaussian random functions, Qiao et al. (2019) performed FPCA as a dimension reduction approach to approximating each function by a finite representation, and extended the glasso criterion to fglasso, which estimates the functional graphical model by imposing a block sparsity constraint on the precision matrix via a group lasso penalty. The fglasso algorithm to fit functional graphical models has been the most popular one under Gaussian assumption. Following a fundamentally different approach, based on extending Markov distributions and hyper Markov laws from random variables to random processes, Zhu et al. (2016) constructed a framework for Bayesian inference of undirected, decomposable graphs in the Gaussian multivariate functional data context. Still under Gaussian assumption, Qiao et al. (2020) proposed a class of doubly functional graphical models to capture the evolving conditional dependence relationship among a large number of sparsely or densely sampled functions. Furthermore, Li and Solea (2018) introduced a nonparametric functional graphical model based on the additive conditional independence, with no distributional assumption imposed on the random functions. Moreover, this group went further to explore the functional copula Gaussian graphical model (FCGGM), which removes the marginal Gaussian assumption but retains the simplicity of the Gaussian dependence structure (Solea and Li, 2020).

2.5 Mixture of Functional Graphical Models

The derivation of mixture models arises when one samples from a population that consists of several homogeneous subpopulations, the components of the population. This is very common in real-world applications. For instance, the Attention Deficit Hyperactivity Disorder (ADHD)-200 Global Competition provides a large dataset
covering participants from three subgroups of the population: healthy control, ADHD combined (ADHD-C) type, and ADHD inattentive (ADHD-I) type, and aims to learn a machine learning classifier that uses a participant’s resting state fMRI scan to diagnose (classify) that individual into one of three subgroups (Brown et al., 2012). Mixture models are powerful to effectively identify subpopulations with hidden commonality within the whole population (Lindsay, 1995; McLachlan et al., 2019), and mixture of graphical models can further depict heterogeneous conditional dependencies across the whole population. A lot of studies have been carried out to explore mixture of vector-valued graphical models. Rodriguez et al. (2011) developed the nonparametric Bayesian estimation of mixtures of Gaussian graphical models. Under the fixed dimensional setting, Mallapragada et al. (2010) and Huang et al. (2013) studied the nonparametric finite mixture model for clustering and regression analyses, respectively. Mixture models were extensively studied in the classical low-dimensional scenarios, but received less attention in high-dimensional statistical learning. Recently, Städler et al. (2010) developed an efficient EM algorithm to study $\ell_1$ penalization for mixture of high-dimensional regression models, and Ruan et al. (2011) went further to study mixture of Gaussian graphical models in the high-dimensional scenarios. It would be even more challenging to analyze the data that consist of observations coming from different sources and sharing hidden commonality of infinite dimension, i.e., the nodes in each subgroup are of functional forms, which is a somewhat more complicated setting, and very little literature on mixture of functional graphical models (MFGM) is available so far.
Chapter 3

Mixture of Functional Graphical Models

In this Section we present our MFGM to detect heterogeneous subgroups and recover conditional dependence structures in each subgroup of multivariate functional data.

3.1 Functional Data Smoothing

The first step in FDA is to perform smoothing that converts the raw discrete observation points into a smoothly varying function. This smoothing process emphasizes patterns in the data by minimizing short-term deviations due to observational errors, such as measurement errors or inherent system noise (Ullah and Finch, 2013). Basis expansion, expanding the functions using basis functions, is one of the common nonparametric smoothing methods that represent the potentially infinite-dimensional world of functions within the finite-dimensional framework of vectors. It has been shown that B-spline expansion is the most popular smoothing technique ever used, presumably...
due to its simplicity and flexibility for tackling a wide range of nonparametric and semiparametric modeling situations (Ullah and Finch, 2013).

The discrete observations \((y_{ij}, t_j), i = 1, \ldots, N, j = 1, \ldots, J\), are represented as 
\[ y_{ij} = x_i(t_j) + \epsilon_{ij}, \quad t_j \in T, \]
where \(\epsilon_{ij}\) are the error terms, and the random functions \(x_i(t)\) are decomposed by B-spline expansion:

\[ x_i(t) \approx \sum_{k=1}^{K} c_{ik} B_k(t), \]

for \(i = 1, \ldots, N, \ t \in T\). We find the coefficients for the basis functions \(\{c_{ik} : i = 1, \ldots, N, k = 1, \ldots, K\}\) by

\[
\{\hat{c}_{ik}\} = \arg\min_{\{c_{ik}\}} \sum_{i=1}^{N} \left( y_i - \sum_{k=1}^{K} c_{ik} B_k(t_i) \right)^2 + \lambda \int \left( L \left[ \sum_{k=1}^{K} c_{ik} B_k(t) \right] \right)^2 dt,
\]

where \(\lambda\) is roughness penalty term parameter and \(L\) is an operator that measures the roughness of functions, e.g., typically, taking the second derivative of the functions.

The optimal basis dimension \(K\) can be chosen by the GCV method, the grid search to find the minimum GCV score: 
\[
\frac{n \times SSE}{(n - df)^2},
\]
where the error sums of squares 
\[ SSE = \sum_{i=1}^{N} \left( y_i - \sum_{k=1}^{K} c_{ik} B_k(t_i) \right)^2, \]
and \(df\) is degrees of freedom measure of the smooth that is related to the roughness term 
\[
\int \left( L \left[ \sum_{k=1}^{K} c_{ik} B_k(t) \right] \right)^2 dt.
\]

### 3.2 Functional Graphical Models

The graphical models are used to understand the conditional dependence structure of a set of random variables. Given a \(p\)-dimensional random variable \(X = (X_1, \ldots, X_p)^\top\),

we can find the conditional covariance as

\[ C_{ij} = \text{cov} (X_i, X_j | \{X_k, k \neq i, j\}) . \]

Using this, we can construct the graph \( G = (V, E) \), where \( V = \{1, \ldots, p\} \) represents the vertices and \( E = \{(i, j) : C_{ij} \neq 0, (i, j) \in V^2, i \neq j\} \) represents the edges.

It is typically assumed that the random variable \( X \) follows multivariate normal distribution for a Gaussian graphical model. The benefit of Gaussian distribution is that when we have a covariance matrix \( \Sigma \) for \( X \), we can find the precision matrix, \( \Theta = \Sigma^{-1} \), and this \( \Theta \) will convey the conditional dependence structure. If \( \Theta_{ij} \neq 0 \), then \( X_i \) and \( X_j \) are conditionally dependent; and if \( \Theta_{ij} = 0 \), then \( X_i \) and \( X_j \) are conditionally independent. Therefore the edge set \( E \) can be represented as

\[ E = \{(i, j) : \Theta_{ij} \neq 0, (i, j) \in V^2, i \neq j\}. \]

As discussed in Qiao et al. (2019), functional graphical models are the extension of graphical models to functional variables. Instead of \( X \), the set of variables is a \( p \)-dimensional functional variable \( X(t) = (X_1(t), \ldots, X_p(t))^\top \) for \( t \in \mathcal{T} \). We assume they jointly follow a \( p \)-dimensional multivariate Gaussian process. The conditional covariance is defined by

\[ C_{ij}(s, t) = \text{cov} (X_i(s), X_j(t) | \{X_k(\cdot), s, t \in \mathcal{T}, k \neq i, j\}) . \]

Then \( X_i \) and \( X_j \) are conditionally independent if and only if \( C_{ij}(s, t) = 0 \) for all \( s, t \in \mathcal{T} \).
Therefore, the edge set for $G = (V, E)$ would be

$$E = \{(i, j) : C_{ij}(s, t) \neq 0 \text{ for some } s, t \in \mathcal{T}, \ (i, j) \in V^2, \ i \neq j\}.$$ 

We observe $N$-replications of $\mathbf{X}(t)$. With Karhunen-Loève expansion, each functional variable is represented with

$$x_{ij}(t) = \sum_{l=1}^{\infty} a_{ijkl}(t),$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, p$.

Since we cannot fully observe the infinite-dimensional functional objects, to estimate the edge set, we use the truncated version of functional object. Following the logic of Qiao et al. (2019), we assume the $M$-truncated version of Karhunen-Loève expansion follows multivariate Gaussian distribution:

$$x_{ij}(t) \approx \sum_{l=1}^{M} a_{ijkl}(t),$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, p$. The truncated multivariate random vector

$$\mathbf{a}_i^M = \left((\mathbf{a}_{i1}^M)^T, \ldots, (\mathbf{a}_{ip}^M)^T\right)^T \in \mathbb{R}^{Mp} \sim \mathcal{N}(\mathbf{\mu}, \mathbf{\Theta}^M)$$

represents the first $M$ principal component scores for the $i$th set of functions for $i = 1, \ldots, N$, where $\mathbf{a}_{ij}^M = (a_{ij1}, \ldots, a_{ijM})^T$. We can define the $M$-truncated conditional cross-covariance function by

$$C_{ij}^M(s, t) = \text{cov} \left(\mathbf{X}_i^M(s), \mathbf{X}_j^M(t)|\{\mathbf{X}_k^M(\cdot), \ s, t \in \mathcal{T}, \ k \neq i, j\}\right).$$

According to Qiao et al. (2019), for $(i, j) \in V^2$, let $\Theta_{ij}^M$ be the $M \times M$ matrix corresponding to the $(i, j)$th submatrix of $\mathbf{\Theta}^M$, it can be shown that,
\[ E = \{(i, j) : \|\Theta_{ij}^M\|_F \neq 0, \ (i, j) \in V^2, \ i \neq j\}, \]

where \(\|\cdot\|_F\) denotes the Frobenius norm.

Therefore, the log-likelihood would look like

\[ l = \sum_{i=1}^{N} \log \mathcal{N}(a_i^M | \mu, \Theta^M), \]

where

\[ \mathcal{N}(a_i^M | \mu, \Theta^M) = (2\pi)^{-\frac{M^2}{2}} |\Theta^M|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (a_i^M - \mu) ^\top \Theta^M (a_i^M - \mu) \right\}. \]

Let \(S^M\) be the sample covariance matrix of \(a_i^M\). Referring to Qiao et al. (2019), fglasso is proposed to estimate the network structure. The fglasso modifies the glasso by incorporating a group lasso penalty (Yuan and Lin, 2006) to produce a block sparsity structure. Specifically, the fglasso is defined as the solution to

\[ \hat{\Theta}^M = \arg \max_{\Theta^M} \left\{ \log |\Theta^M| - \text{tr}(S^M \Theta^M) - \lambda \sum_{i \neq j} \|\Theta_{ij}^M\|_F \right\}, \]

where \(\Theta^M \in \mathbb{R}^{Mp \times Mp}\) is symmetric positive-definite and \(\lambda\) is a nonnegative tuning parameter. The group lasso penalty forces the elements of \(\Theta_{ij}^M\) to either all be zero (a sparse solution) or all nonzero (a connected edge between node \(i\) and node \(j\)). Hence, as \(\lambda\) increases, \(\Theta^M\) grows sparser in a blockwise fashion. The final estimated edge set is then defined as

\[ \hat{E}^M = \{(i, j) : \|\hat{\Theta}_{ij}^M\|_F \neq 0, \ (i, j) \in V^2, \ i \neq j\}. \]
3.3 Mixture of Functional Graphical Models

In multivariate vector context, let $G(X) = (V, E)$ represent the mixture graphical model of vector $X \in \mathbb{R}^p$ with vertex set $V = \{1, \ldots, p\}$ and edge set $E$. Suppose the $X$ jointly follows a $p$-dimensional multivariate Gaussian distribution, and each distribution can be depicted by an undirected graph $G_k = (V, E_k), \ k = 1, \ldots, K$. Denote by $K$ the number of communities. In each community, the edge set $E_k$, namely, conditional dependence structure, is reflected by the precision matrix $\Theta_k$. Define $Z = (Z_1, \ldots, Z_N) \in \mathcal{Z} = \{1, \ldots, K\}^N$ as the community membership indicators. Assume that $Z_i$ are independently drawn from a multinomial distribution with parameter $\pi = (\pi_1, \ldots, \pi_K)^\top$, with $\sum_{k=1}^K \pi_k = 1$. Namely, $Z_1, \ldots, Z_N \overset{i.i.d}{\sim} \text{Multinomial}(1; \pi)$. Therefore, $G(X) = \pi_1 G_1(X) + \pi_2 G_2(X) + \cdots + \pi_K G_K(X)$, with $\sum_{k=1}^K \pi_k = 1$. Now, the goal of mixture of graphical models is to estimate $\pi$ and $\Theta$, and then infer membership label of each individual via maximizing the log-likelihood of the observed data with sparsity penalty.

Our proposed MFGM is the generalization of mixture of graphical models from finite vector-valued context to infinite functional context. Suppose the functional variables $g_1(t), \ldots, g_p(t)$ jointly follow a $p$-dimensional multivariate Gaussian process with vertex set $V = \{1, \ldots, p\}$ and edge set $E$, and each process belongs to an undirected graph $G_k = (V, E_k), \ k = 1, \ldots, K$, where $K$ denotes the number of communities.

With Karhunen-Loève expansion, we represent each functional variable with

$$g_{ij}(t) = \sum_{l=1}^{\infty} a_{ijl} \phi_{jl}(t),$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, p$.

To go further, the $M$-truncated version of Karhunen-Loève expansion would be
\[ g_{ij}(t) \approx \sum_{i=1}^{M} a_{ijl} \phi_{jl}(t), \]

for \( i = 1, \ldots, N \) and \( j = 1, \ldots, p \). We assume the mixture of multivariate Gaussianity of the truncated multivariate random vector

\[
\mathbf{a}_i^M = \left( (\mathbf{a}_{i1}^M)\top, \ldots, (\mathbf{a}_{ip}^M)\top \right)\top \in \mathbb{R}^{Mp} \sim \sum_{k=1}^{K} \pi_k \mathcal{N} (\mu_k; \Theta_k),
\]

which represents the first \( M \) principal component scores for the \( i \)th set of functions for \( i = 1, \ldots, N \), where \( \mathbf{a}_{ij}^M = (a_{ij1}, \ldots, a_{ijM})\top \).

Now, the log-likelihood would look like

\[
l = \sum_{i=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{a}_i^M | \mu_k; \Theta_k),
\]

where

\[
\mathcal{N}(\mathbf{a}_i^M | \mu_k; \Theta_k) = (2\pi)^{-\frac{M}{2}} |\Theta_k|^{\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\mathbf{a}_i^M - \mu_k)\top \Theta_k (\mathbf{a}_i^M - \mu_k) \right\}.
\]

Therefore, for MFGM the fglasso is defined as the solution to

\[
\left\{ (\hat{\pi}_k, \hat{\Theta}_k) \right\}_{k=1, \ldots, K} = \arg \max_{(\pi_k, \Theta_k)} \left\{ \sum_{i=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{a}_i^M | \mu_k; \Theta_k) - \sum_{k=1}^{K} \lambda_k \sum_{j \neq l} \| \Theta_{kjl}^M \|_{\mathcal{F}} \right\}.
\]

### 3.4 Computation

The EM algorithm provides a powerful tool to deal with latent variables in mixture models. Following the spirit of the EM algorithm, we view the functional data to be incomplete, and treat the latent variables as “missing data”. Moreover, unlike
traditional approaches, the sparse estimation imposes the non-smooth penalty function to regularize the likelihood function, which leads to solving a challenging non-convex and non-smooth optimization problem.

We introduce the latent random variables $Z_i = (Z_{i1}, \ldots, Z_{iK})^T$, $i = 1, \ldots, N$, satisfying that

$$Z_{ik} = \begin{cases} 
1 & \text{if } g_i(t) \text{ belongs to the } k\text{th community,} \\
0 & \text{otherwise.}
\end{cases}$$

Now given the complete data, the complete log-likelihood would be

$$\ell_{\text{comp}} = \sum_{i=1}^{N} \sum_{k=1}^{K} Z_{ik} \log \pi_k + Z_{ik} \log \mathcal{N}(a_i^M|\mu_k, \Theta_k),$$

and the complete penalized log-likelihood function becomes

$$L_{\text{comp}} = \ell_{\text{comp}} - \sum_{k=1}^{K} \lambda_k \sum_{j \neq l} \|\Theta_{kj}\|_F.$$ 

**E step:** Let $\pi_k^{(l)}$, $\mu_k^{(l)}$, and $\Theta_k^{(l)}$ be the estimates of $\pi_k$, $\mu_k$, and $\Theta_k$ for $k$th community at the $l$th iteration. In the E step of the $(l+1)$th iteration, we compute the conditional expectation of membership probabilities $\gamma_{ik}$ given current estimates $\pi_k^{(l)}$, $\mu_k^{(l)}$, and $\Theta_k^{(l)}$ for $k = 1, \ldots, K$. From Bayes’ rule, the conditional expectation of $\gamma_{ik}$ is of the following form:

$$\gamma_{ik}^{(l+1)} = \frac{\pi_k^{(l)} \mathcal{N}(a_i^M|\mu_k^{(l)}, \Theta_k^{(l)})}{\sum_{k=1}^{K} \pi_k^{(l)} \mathcal{N}(a_i^M|\mu_k^{(l)}, \Theta_k^{(l)})}.$$ 

These estimates are also called the membership probabilities as the output of the E step.
**M step:** In the M step of the \((l + 1)\)th iteration, we estimate the parameters \(\pi_k, \mu_k, \) and \(\Theta_k\) that maximize the conditional expectation of \(\mathcal{L}^{\text{comp}} = \ell_{\text{comp}} - \sum_{k=1}^{K} \lambda_k \sum_{j \neq l} \| \Theta_{kj}^{l} \|_F\) given the updated membership probabilities \(\gamma_{ik}^{(l+1)}\), namely:

\[
\sum_{k=1}^{K} \left( \sum_{i=1}^{N} \gamma_{ik}^{(l+1)} \left( \log \pi_k^{(l)} + \log \mathcal{N}(a_i^M | \mu_k^{(l)}, \Theta_k^{(l)}) \right) - \lambda_k \sum_{j \neq l} \| \Theta_{kj}^{(l)} \|_F \right).
\]

Now the closed-form solution of \(\pi_k^{(l+1)}\) and \(\mu_k^{(l+1)}\) can be solved from the maximization of the above function. We update \(\pi_k^{(l+1)}\) by

\[
\pi_k^{(l+1)} = \frac{1}{N} \sum_{i=1}^{N} \gamma_{ik}^{(l+1)},
\]

and update \(\mu_k^{(l+1)}\) by

\[
\mu_k^{(l+1)} = \frac{\sum_{i=1}^{N} \gamma_{ik}^{(l+1)} a_i^M}{\sum_{i=1}^{N} \gamma_{ik}^{(l+1)}}.
\]

Next, we update \(\Theta_k^{(l+1)}\) with the competing ADMM algorithm or by solving the below optimization problem with the state-of-art optimization algorithm fglasso.

\[
\Theta_k^{(l+1)} = \arg \max_{\Theta_k^{(l)}} \left\{ \log |\Theta_k^{(l)}| - \text{tr}(S_k^{(l+1)} \Theta_k^{(l)}) - \lambda_k \sum_{j \neq l} \| \Theta_{kj} \|_F \right\},
\]

where

\[
S_k^{(l+1)} = \frac{\sum_{i=1}^{N} \gamma_{ik}^{(l+1)} (a_i^M - \mu_k^{(l+1)})(a_i^M - \mu_k^{(l+1)})^\top}{\sum_{i=1}^{N} \gamma_{ik}^{(l+1)}}.
\]

We alternate between the E step and the M step until the estimates of parameters converge. The EM algorithm is sensitive to the initial values of the parameters, so care must be taken in the first step. In this work, the Mclust function, acquired from the R mclust package, and the split_comp function, acquired from the R gmgm package, are applied to the multivariate principal component score vectors to provide good initials...
for the EM iterations.

Now we discuss the tuning parameter selection of our algorithm via CV approach. The $J$-fold cross-validation score ($CV$) for mixture case ($K$ clusters) is represented with:

$$CV(\lambda_1, \cdots, \lambda_K) = \sum_{j=1}^{J} \sum_{k=1}^{K} n_j \left( \log \hat{\pi}_{-j}^k - \log |\hat{C}_{\lambda_k, -j}^k| + \text{tr}(\hat{C}_{\lambda_k, -j}^k \Sigma_j^k) \right),$$

where $n_j$ is the sample size of test data in $j$th CV, $\hat{\pi}_{-j}^k$ is the estimated $k$th community proportion by using training data in $j$th CV, $\hat{C}_{\lambda_k, -j}^k$ is the estimated precision matrix of community $k$ by using training data with the tuning parameter $\lambda_k$ in $j$th CV, and $\Sigma_j^k$ is the test data sample covariance matrix in $j$th CV. As the regular grid search process takes too much time for finding the optimal tuning parameters, the more efficient random search process is performed to find the optimal tuning parameter vector $(\lambda_1, \cdots, \lambda_K)^T$ that results in smallest value of $CV$. Then, the optimal tuning parameter vector is used for MFGM for total data analysis to perform overall clustering $(\hat{\gamma}_{ik}, i = 1, \ldots, N, k = 1, \ldots, K)$ and to estimate conditional dependence structures in each cluster $(\hat{\Theta}_k, k = 1, \ldots, K)$.

### 3.5 Algorithmic Convergence

In this section, we prove that our EM algorithm actually ascertains an algorithmic convergence, meaning that in each iteration, it approximates to the desired direction of maximizing our penalized likelihood.

**Theorem 1** Our EM algorithm as described in Section 3.4 satisfies ascent property.

Here, the ascent property means that the unknown parameters of the mixture
graphical models estimated by the EM iterations incrementally increase the observed incomplete-data log-likelihood with penalty. Proof of Theorem 1 is as follows.

In multivariate vector context, let $G(X) = (V, E)$ represent the mixture graphical model of vector $X \in \mathbb{R}^p$ with vertex set $V = \{1, \ldots, p\}$ and edge set $E$. Suppose the $X$ jointly follows a $p$-dimensional multivariate Gaussian distribution, and each distribution can be depicted by an undirected graph $G_k = (V, E_k)$, $k = 1, \ldots, K$. Denote by $K$ the number of communities. In each community, the edge set $E_k$, namely, conditional dependence structure, is reflected by the precision matrix $\Theta_k$. Define latent variables $Z = (Z_1, \ldots, Z_N) \in \mathcal{Z} = \{1, \ldots, K\}^N$ as the community membership indicators. Assume that $Z_i$ are independently drawn from a multinomial distribution with parameter $\pi = (\pi_1, \ldots, \pi_K)^T$, where $\sum_{k=1}^{K} \pi_k = 1$. Namely, $Z_1, \ldots, Z_N \overset{i.i.d}{\sim}$ Multinomial$(1; \pi)$, satisfying that

$$Z_{ik} = \begin{cases} 1 & \text{if } X_i \text{ belongs to the } k\text th \text{ community}, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, our unknown parameters are: $\omega = \{\pi_1, \ldots, \pi_K, \mu_1, \ldots, \mu_K, \Theta_1, \ldots, \Theta_k\}$. Then the observed incomplete-data log-likelihood is:

$$\ell_{(X)}(\omega) = \sum_{i=1}^{N} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(X_i|\mu_k, \Theta_k),$$

which is what we want to optimize.

Now given the complete data, the complete log-likelihood would be

$$\ell_{(X,Z)}(\omega) = \log \left(P(X, Z|\omega)\right) = \sum_{i=1}^{N} \sum_{k=1}^{K} Z_{ik} \left( \log \pi_k + \log \mathcal{N}(X_i|\mu_k, \Theta_k) \right).$$
In the **E-step**, we use the current value of the parameters \( \omega^{(l)} \) to find the posterior distribution of the latent variables given by \( P(Z|X, \omega^{(l)}) \), and then use this to find the expectation of the complete-data log-likelihood, with respect to this posterior, evaluated at an arbitrary \( \omega \). This expectation is denoted \( Q(\omega|\omega^{(l)}) \):

\[
Q(\omega|\omega^{(l)}) = E_{Z|X,\omega^{(l)}} \left[ \log \left( P(X, Z|\omega) \right) \right] = \sum_Z \log \left( P(X, Z|\omega) \right) P(Z|X, \omega^{(l)}).
\]

In the **M-step**, we determine the new parameter \( \hat{\omega} \) by maximizing \( Q \) function:

\[
\hat{\omega} = \arg \max_{\omega} Q(\omega|\omega^{(l)}).
\]

Next, we show that iteratively maximizing \( Q \) function is equivalent to incrementally maximizing the observed incomplete-data log-likelihood \( \ell(X)(\omega) \).

According to the density identity, we can write

\[
\ell(X)(\omega) = \ell(X,Z)(\omega) - \ell(Z|X)(\omega).
\]

On both sides, taking expectation with respect to \( Z \), given \( X \) and \( \omega^{(l)} \), we have

\[
\ell(X)(\omega) = Q(\omega|\omega^{(l)}) - H(\omega|\omega^{(l)}),
\]

where \( H(\omega|\omega^{(l)}) = E_{Z|X,\omega^{(l)}} \left[ \ell(Z|X)(\omega) \right] \).

According to the non-negativity of relative entropy or Kullback-Leibler divergence:

\[
\int \log \frac{p(x)}{q(x)} p(x) \, dx \geq 0, \text{ equality iff } p = q, \text{ where } P \text{ and } Q \text{ are distributions},
\]
we have
\[ \sum Z \log \frac{P(Z|X, \omega^{(l)})}{P(Z|X, \omega)} P(Z|X, \omega^{(l)}) \geq 0, \text{ equality iff } \omega^{(l)} = \omega, \]

which is equivalent to
\[ H(\omega^{(l)}|\omega^{(l)}) - H(\omega|\omega^{(l)}) \geq 0, \forall \omega. \]

Let \( \omega \) take the value of \( \omega^{(l+1)} \), we have
\[ -H(\omega^{(l+1)}|\omega^{(l)}) \geq -H(\omega^{(l)}|\omega^{(l)}). \]

At the same time, the EM process warrants
\[ Q(\omega^{(l+1)}|\omega^{(l)}) \geq Q(\omega^{(l)}|\omega^{(l)}), \forall l. \]

Taken together, we have
\[ Q(\omega^{(l+1)}|\omega^{(l)}) - H(\omega^{(l+1)}|\omega^{(l)}) \geq Q(\omega^{(l)}|\omega^{(l)}) - H(\omega^{(l)}|\omega^{(l)}), \forall l, \]

which is equivalent to
\[ \ell(\omega^{(l+1)}) \geq \ell(\omega^{(l)}), \forall l. \]

This does not imply that the EM updates will necessarily converge to the MLE, just that they are surely moving in the right direction, and local maxima convergence is warranted. The approaches to provide optimal initials are to be explored.

In our study, what we actually want to optimize is adjusted to the penalized
incomplete-data log-likelihood:

$$\ell_{(X)}(\omega) + \sum_{k=1}^{K} \lambda_k \sum_{j \neq l} \| \Theta_{kjl}^M \|_F.$$ 

Accordingly, in EM process, the Q function is adjusted to

$$E_{Z|X,\omega^{(t)}} \left[ \log \left( P(X, Z|\omega) \right) \right] + \sum_{k=1}^{K} \lambda_k \sum_{j \neq l} \| \Theta_{kjl}^M \|_F.$$ 

Therefore, the proof for our study considering glasso penalty is the same to that for the scenario of traditional log-likelihood only.
Chapter 4

Simulation Studies

We perform a number of simulations to compare our MFGM algorithm to potential competing methods, the ADMM algorithm provided by the R fgm package that assumes partial separability in the proposed functional Gaussian graphical models (depicted in Figure 4.1), and the mixgmm algorithm that takes average of observations across the time interval for each node and implements mixture of Gaussian graphical models in a multivariate vector context.

Figure 4.1: Covariance structures of $\mathbb{R}^{Lp}$-valued random coefficients from different $L$-truncated Karhunen-Loève type expansions. (a) and (b): covariance and precision matrices, respectively, of functional principal component coefficients $(\xi_1^T, \ldots, \xi_p^T)^T$ as in Qiao et al. (2019). (c) and (d): block diagonal covariance and precision matrices, respectively, of functional principal component coefficients $(\theta_1^T, \ldots, \theta_L^T)^T$ under partial separability as in Zapata et al. (2019).
4.1 Functional Graphical Models

In each setting, the multivariate Gaussian functional variables are generated via
\[ g_{ij} = s(t)^T \delta_{ij} \] for \( i = 1, \ldots, N \) and \( j = 1, \ldots, p \), where \( s(t) \) is a five-dimensional Fourier basis function, and \( \delta_{ij} \in \mathbb{R}^5 \) is a mean zero Gaussian random vector. Hence, \( \delta_i = (\delta_{i1}^T, \ldots, \delta_{ip}^T)^T \in \mathbb{R}^{5p} \) follows a multivariate Gaussian distribution with covariance \( \Sigma = \Theta^{-1} \) (Qiao et al., 2019). Different block sparsity patterns in the precision matrix \( \Theta \) correspond to different conditional dependence structures. We consider five general structures as follows:

1. Model 1: An identity precision matrix of dimension \( 5p \times 5p \) is generated. Hence, all of the \( p \) nodes are disconnected. This is called Independent model.

2. Model 2: A block banded matrix \( \Theta \) is generated with \( \Theta_{jj} = I_5 \) for \( j = 1, \ldots, p \), \( \Theta_{j,j+1} = \Theta_{j+1,j} = 0.5I_5 \) for \( j = 1, \ldots, p - 1 \), and 0 at all other locations. Hence, only the adjacent two nodes are connected. This is called Autoregressive One (AR1) model.

3. Model 3: A block banded matrix \( \Theta \) with \( \Theta_{jj} = I_5 \) for \( j = 1, \ldots, p \), \( \Theta_{j,j+1} = \Theta_{j+1,j} = 0.4I_5 \) for \( j = 1, \ldots, p - 1 \), \( \Theta_{j,j+2} = \Theta_{j+2,j} = 0.2I_5 \) for \( j = 1, \ldots, p - 2 \), and 0 at all other locations. Hence, the consecutively adjacent three nodes are pair-wise connected. This is called Autoregressive Two (AR2) model with weak connection.

4. Model 4: Similar to Model 3, a block banded matrix \( \Theta \) is generated with \( \Theta_{jj} = I_5 \) for \( j = 1, \ldots, p \), \( \Theta_{j,j+1} = \Theta_{j+1,j} = 0.6I_5 \) for \( j = 1, \ldots, p - 1 \), \( \Theta_{j,j+2} = \Theta_{j+2,j} = 0.35I_5 \) for \( j = 1, \ldots, p - 2 \), and 0 at all other locations. Hence, the consecutively
adjacent three nodes are pair-wise connected. This is called Autoregressive Two (AR2) model.

5. Model 5: A block banded matrix $\Theta$ is generated with random sparse connection structure: $\Theta_{jj} = \mathbf{I}_5$ and $\Theta_{j,k} = \Theta_{k,j} = 0.25B_{j,k}\mathbf{I}_5$ for $j = 1, \ldots, p$, and $j \neq k$, where $B_{j,k}$ is a Bernoulli random variable which takes the value 1 with probability 0.05. The precision matrix $\Theta$ is generated iteratively until the positive-definite requirement is reached. This is called Random model.

![Figure 4.2: Conditional dependencies in each simulated functional graphical model. Thickness of the edge denotes strength of connection.](image)

The five simulation models are depicted in Figure 4.2. In all settings, we consider dimension parameter $p = 20$, and generate observations of $\delta_i$ from the associated multivariate Gaussian distribution, and the observed values $h_{ik}$ are sampled using

$$h_{ijl} = g_{ij}(t_l) + e_{ijl}, \text{ for } i = 1, \ldots, n, \ j = 1, \ldots, p, \text{ and } l = 1, \ldots, T, \ e_{ijl} \sim \mathcal{N}(0, 0.5^2),$$

where each function is observed at 100 equally spaced time points, $0 = t_1, \ldots, t_{100} = 1$. 
4.2 Mixture of Functional Graphical Models

4.2.1 Two-Cluster Mixture Models

Three two-cluster mixture of multivariate functional models with mixing proportion $\frac{1}{2}$ as the below. The models are based on the settings described in Section 4.1.

1. Model (1,4): $h_i = B_i \times h_i_{(\text{Model 1})} + (1 - B_i) \times h_i_{(\text{Model 4})}$
2. Model (2,3): $h_i = B_i \times h_i_{(\text{Model 2})} + (1 - B_i) \times h_i_{(\text{Model 3})}$
3. Model (4,5): $h_i = B_i \times h_i_{(\text{Model 4})} + (1 - B_i) \times h_i_{(\text{Model 5})}$

where $B_i$ is a Bernoulli random variable which takes the value 1 with probability 0.5. We generate $N = 100$ mixture functional observations of $h_i$ for each mixture model. We expect that it is not challenging to do clustering and to estimate connection structures in Model (1,4) as there is obvious distinction between the Identity precision matrix and AR2 precision matrix in this mixture model. Model (2,3) should be hard in this simulation study as the AR1 precision matrix and AR2 precision matrix with weak connections are way close to each other. We go further to set the design of Model (4,5) to explore whether our method can perform good analysis in the mixture model in which a subgroup with random connection structure is involved.

4.2.2 Three-Cluster Mixture Models

To explore even more complex scenarios, two three-cluster mixture of multivariate functional models with mixing proportions $\frac{1}{3}$ of the following combinations are simulated. The models are based on the settings described in Section 4.1.
1. Model (1,2,4): \( h_i = B_{i1} \times h_i^{(\text{Model 1})} + B_{i2} \times h_i^{(\text{Model 2})} + B_{i3} \times h_i^{(\text{Model 4})} \)

2. Model (2,4,5): \( h_i = B_{i1} \times h_i^{(\text{Model 2})} + B_{i2} \times h_i^{(\text{Model 4})} + B_{i3} \times h_i^{(\text{Model 5})} \)

where \( B_i = (B_{i1}, B_{i2}, B_{i3})^\top \) is a random variable that follows multinomial distribution with probability parameter \( (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^\top \). We generate \( N = 50 \) mixture functional observations of \( h_i \) for each mixture model. In model (1,2,4), the three basic graphical structures, Independent, AR1, and AR2, are involved; and in Model (2,4,5), the subgroup with random graphical structure is considered for the mixture with two other heterogeneous subgroups. We expect that the three-cluster mixture models are more challenging than the two-cluster mixture models to analyze.

### 4.3 Data Analysis of Simulated Mixture of Functional Graphical Models

To apply our proposed MFGM algorithm to the analysis of simulated mixture data, first, the total functional observations are fitted by using an \( L \)-dimensional cubic B-spline basis. The GCV method is used to choose the optimal dimension parameter \( L \). Then the smoothed functions are each decomposed by \( M \)-truncated Karhunen-Loève expansion, and the optimal harmonic number \( M \) is determined by eight-fold CV. It turns out that \( M = 5 \), which is consistent with our design, and our further checking shows that five principal components already explain more than 99% of the total variation in the signal trajectories for each node. The multivariate Karhunen-Loève expansion basis coefficient (principal component score) vectors \( a_i^M \) with \( M = 5 \) are thus acquired for further mixture analysis assuming Gaussianity.

In the iterative EM process to analyze the mixture of blocked Gaussian multivariate
vector graphical models, the fglasso algorithm employed in our method is compared with the ADMM algorithm, to solve the maximization problem of log-likelihood with penalty for estimating the conditional dependence structures in each cluster. To provide good initials for the EM iterations, the Mclust function, acquired from the R mclust package, and the split_comp function, acquired from the R gmgm package, are applied to the multivariate principal component score vectors, for two-cluster and three-cluster mixture models analysis, respectively. Our MFGM algorithm is also compared with the mixgmm algorithm to confirm the advantage of considering inherent functional nature of the data.

The overall clustering and the estimates of the edge structures in each subgroup are checked. The metrics considered in the comparisons are as follows:

\[
\text{Accuracy (Accu): } \frac{TP+TN}{TP+TN+FP+FN},
\]

\[
\text{Matthews correlation coefficient (MCC): } \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}},
\]

\[
\text{True Positive Rate (TPR): } \frac{TP}{TP+FN},
\]

\[
\text{False Positive Rate (FPR): } \frac{FP}{FP+TN},
\]

where TP, TN, FP, FN represent true positives, true negatives, false positives, false negatives, respectively, which are depicted in Table 4.1.

In three-cluster mixture models analysis, the metric

\[
\text{MCC: } \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}},
\]

is still employed for overall clustering assessment. The Micro-Averaging approach is
Table 4.1: Definitions of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

<table>
<thead>
<tr>
<th>Estimate vs. Real</th>
<th>Real Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subgroup 1</td>
</tr>
<tr>
<td>Estimated Labels</td>
<td>TN</td>
</tr>
<tr>
<td></td>
<td>FP</td>
</tr>
</tbody>
</table>

used to calculate TP, TN, FP, FN as the below:

\[
\begin{align*}
TP &= TP_{\text{Subgroup1}} + TP_{\text{Subgroup2}} + TP_{\text{Subgroup3}}, \\
TN &= TN_{\text{Subgroup1}} + TN_{\text{Subgroup2}} + TN_{\text{Subgroup3}}, \\
FP &= FP_{\text{Subgroup1}} + FP_{\text{Subgroup2}} + FP_{\text{Subgroup3}}, \\
FN &= FN_{\text{Subgroup1}} + FN_{\text{Subgroup2}} + FN_{\text{Subgroup3}}.
\end{align*}
\]

In the assessment of overall clustering for two-cluster and three-cluster mixture models, the confusion tables are created by referring to the maximum MCC values, which resolves the label-switching issue. We run each simulation 100 times for two-cluster mixture models analysis and 50 times for three-cluster mixture models analysis, and the means of all metrics calculated by the three algorithms are acquired for comparison.

4.3.1 Two-Cluster Mixture Models Analysis

As shown in Table 4.2, both of our method and the ADMM algorithm do a very good job in overall clustering of the three two-cluster mixture models, but the mixgmm algorithm does a relatively poor job reflected by the poor values of all the four metrics for each mixture model. The results of overall clustering of the most challenging
mixture model, Model (2,3), reveal a solid advantage of our method and the ADMM algorithm compared with the mixggm algorithm that ignores the inherent functional nature in the data.

Table 4.2: Comparisons of MFGM with ADMM and mixggm in terms of overall clustering in the two-cluster mixture simulation studies.

<table>
<thead>
<tr>
<th>Model (1,4)</th>
<th>Accu</th>
<th>MCC</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFGM</td>
<td>0.9992</td>
<td>0.9985</td>
<td>0.9985</td>
<td>0.0000</td>
</tr>
<tr>
<td>ADMM</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.9283</td>
<td>0.8568</td>
<td>0.8787</td>
<td>0.0299</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model (2,3)</th>
<th>Accu</th>
<th>MCC</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFGM</td>
<td>0.9923</td>
<td>0.9848</td>
<td>0.9848</td>
<td>0.0000</td>
</tr>
<tr>
<td>ADMM</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.7470</td>
<td>0.5424</td>
<td>0.5493</td>
<td>0.0506</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model (4,5)</th>
<th>Accu</th>
<th>MCC</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFGM</td>
<td>0.9981</td>
<td>0.9962</td>
<td>1.0000</td>
<td>0.0040</td>
</tr>
<tr>
<td>ADMM</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.8904</td>
<td>0.7921</td>
<td>0.9492</td>
<td>0.1725</td>
</tr>
</tbody>
</table>

Table 4.3 shows further estimates of the conditional dependence structures in each subgroup in the designed two-cluster mixture models. In the analysis of Model (1,4), all of the three methods do a good job to estimate the edge structure in subgroup 1. Our method and mixggm outperform ADMM in estimating the edge structure in subgroup 2. In analyzing the challenging mixture model, Model (2,3), the three methods show similar decent performances, which are a little worse than that in analyzing Model (1,4), however. In analyzing Model (4,5), our method and mixggm algorithm do a decent job in estimating the conditional dependencies in both of the two subgroups, but the ADMM algorithm does a little worse for the estimate in subgroup one, which is denoted by small values of MCC and TPR.
Table 4.3: Comparisons of MFGM with ADMM and mixggm in terms of estimates of edge structures in subgroups in the two-cluster mixture simulation studies.

<table>
<thead>
<tr>
<th>Subgroup 1</th>
<th>Subgroup 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accu MCC TPR FPR</td>
<td>Accu MCC TPR FPR</td>
</tr>
<tr>
<td>Model (1,4)</td>
<td></td>
</tr>
<tr>
<td>MFGM</td>
<td>0.9911</td>
</tr>
<tr>
<td>ADMM</td>
<td>1.0000</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.9484</td>
</tr>
<tr>
<td>Model (2,3)</td>
<td></td>
</tr>
<tr>
<td>MFGM</td>
<td>0.8462</td>
</tr>
<tr>
<td>ADMM</td>
<td>0.8150</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.8694</td>
</tr>
<tr>
<td>Model (4,5)</td>
<td></td>
</tr>
<tr>
<td>MFGM</td>
<td>0.8378</td>
</tr>
<tr>
<td>ADMM</td>
<td>0.7858</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.8614</td>
</tr>
</tbody>
</table>

4.3.2 Three-Cluster Mixture Models Analysis

As shown in Table 4.4, in overall clustering, our method MFGM does best in both of the two three-cluster mixture models, which is indicated by relatively higher MCC values; and mixggm does worst, which is indicated by the lowest MCC values.

Table 4.4: Comparisons of MFGM with ADMM and mixggm in terms of overall clustering in the three-cluster mixture simulation studies.

<table>
<thead>
<tr>
<th>Model (1,2,4)</th>
<th>Model (2,4,5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCC</td>
<td>0.6525</td>
</tr>
<tr>
<td>MFGM</td>
<td>0.6465</td>
</tr>
<tr>
<td>ADMM</td>
<td>0.5801</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.6196</td>
</tr>
</tbody>
</table>

Table 4.5 further compares the three algorithms in terms of the estimates of the conditional dependence structures in each subgroup in the designed three-cluster mixture models. It shows that the three algorithms do better for Model (1,2,4) than for Model (2,4,5) in estimating the graphical structures in the first two subgroups. However, they do worse for Model (1,2,4) than for Model (2,4,5) in estimating the graphical structure in the third subgroup. Moreover, it is revealed that MFGM does the best to estimate the heterogeneous networks almost in all of the three subgroups.
in both of the two mixture models, which is confirmed by the best turnouts of the four metrics. In comparison, ADMM and mixggm do worse than MFGM. There is no significant difference between these two algorithms.

Table 4.5: Comparisons of MFGM with ADMM and mixggm in terms of estimates of edge structures in subgroups in the three-cluster mixture simulation studies.

<table>
<thead>
<tr>
<th></th>
<th>Model (1,2,4)</th>
<th></th>
<th>Model (2,4,5)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subgroup 1</td>
<td>Subgroup 2</td>
<td>Subgroup 3</td>
<td>Subgroup 1</td>
</tr>
<tr>
<td>MFGM</td>
<td>Accu</td>
<td>1.000</td>
<td>0.9606</td>
<td>0.8481</td>
</tr>
<tr>
<td></td>
<td>MCC</td>
<td>1.000</td>
<td>0.8291</td>
<td>0.5351</td>
</tr>
<tr>
<td></td>
<td>TPR</td>
<td>1.000</td>
<td>0.7552</td>
<td>0.3545</td>
</tr>
<tr>
<td></td>
<td>FPR</td>
<td>0.000</td>
<td>0.0046</td>
<td>0.0003</td>
</tr>
<tr>
<td>ADMM</td>
<td>Accu</td>
<td>0.9937</td>
<td>0.9164</td>
<td>0.8181</td>
</tr>
<tr>
<td></td>
<td>MCC</td>
<td>0.9291</td>
<td>0.6189</td>
<td>0.4160</td>
</tr>
<tr>
<td></td>
<td>TPR</td>
<td>0.8760</td>
<td>0.4621</td>
<td>0.2332</td>
</tr>
<tr>
<td></td>
<td>FPR</td>
<td>0.0001</td>
<td>0.0065</td>
<td>0.0022</td>
</tr>
<tr>
<td>mixggm</td>
<td>Accu</td>
<td>0.9692</td>
<td>0.9490</td>
<td>0.8554</td>
</tr>
<tr>
<td></td>
<td>MCC</td>
<td>0.8323</td>
<td>0.7895</td>
<td>0.6588</td>
</tr>
<tr>
<td></td>
<td>TPR</td>
<td>1.0000</td>
<td>0.7593</td>
<td>0.8557</td>
</tr>
<tr>
<td></td>
<td>FPR</td>
<td>0.0324</td>
<td>0.0188</td>
<td>0.1447</td>
</tr>
</tbody>
</table>

To sum up, mixggm does a poor job in overall clustering for all of the simulated mixture models. This confirms a solid advantage of our method and the ADMM compared with mixggm that ignores the inherent functional nature in the data. The simulation studies also confirm that our method outperforms or is as good as the two potential competing algorithms ADMM and mixggm, in analyzing both of the two-cluster and three-cluster multivariate functional mixture models, which lays a solid foundation for the next step to investigate its ability to analyze the real-world functional mixture data.
Chapter 5

Real Data Analysis

5.1 EEG Data and Alcoholism

Alcoholism is a common neurological disorder caused by the mutual effect of genetic and environmental factors. It not only damages the brain system but also leads to cognitive and mobility impairments (Oscar-Berman and Marinković, 2007). It is of great importance to not only find a way that is reliable to distinguish alcoholics from normal subjects, but also recover the distinction of the brain patterns between alcoholics and normal subjects, which helps to explore the underlying mechanisms for alcoholism. EEG is a very effective tool for studying the complex dynamics of brain activities. It can visualize complex brain activities as dynamic outputs (Zhu et al., 2014). Therefore, it can be used to distinguish alcoholics from normal subjects based on the differences in the signals. A functional brain network accounts for the neuro-dynamical interactions between neural regions. Functional connectivity defines statistical interdependence between the dynamics of all pairs of the network nodes without taking into account causal effects (Ahmadi et al., 2017). Therefore, the analysis of the functional EEG data by mixture of graphical models is expected to depict the distinct brain networks
in the two subgroups.

We apply the proposed MFGM algorithm on the EEG dataset acquired from the online UCI Knowledge Discovery in Databases Archive (https://kdd.ics.uci.edu/databases/eeg/eeg.html). Zhang et al. (1995) describes in detail the data collection process. This data arose from a large study to examine EEG correlates of genetic predisposition to alcoholism. The study consisted of 122 subjects, of which 77 belonged to the alcoholism group and 45 to the control group. The data were initially obtained from 64 electrodes placed on the subjects’ scalps that captured EEG signals at 256 Hz during a one-second period. Each subject completed 120 trials under either a single stimulus (a single picture) or two stimuli (a pair of pictures) shown on a computer monitor. As the 64 electrodes were located at standard positions, to reduce the dimension of the data, we select the electrodes that detect signals in the 19-channel montage as specified according to the 10–20 International system (Fp1, Fp2, F7, F3, Fz, F4, F8, T7, C3, Cz, C4, T8, P7, P3, Pz, P4, P8, O1, O2) (Hayden et al., 2006), which are depicted in Figure 5.1 by the blue nodes. Furthermore, referring to the case considered in Hayden et al. (2006); Qiao et al. (2019), we focus on the EEG signals filtered at α frequency bands between 8 and 12.5 Hz that are acquired by applying the eegfilter function (R eegkit package) on the raw data. To remove the potential dependence between the measurements and the influence of different stimulus types, we only select observations under single stimulus for the use in this study (Li et al., 2010; Zhu et al., 2016; Qiao et al., 2019). Moreover, it shows that many studies used multiple samples per subject in order to obtain a sufficiently large sample, which violated the independence assumption inherent in most methods. Following the analysis in Li et al. (2010); Qiao et al. (2019), we average the valid band-filtered EEG signals across all trials for each subject.
First, the filtered EEG functional observations are fitted by using an $L$-dimensional cubic B-spline basis. The GCV method is used to choose the optimal dimension parameter $L$. Then the smoothed functions are each decomposed by $M$-truncated Karhunen-Loève expansion. Different from that in the simulation studies, the CV method always selects the highest value from the search grid as the harmonic number $M$, which leads to a very high dimension of the multivariate Karhunen-Loève expansion basis coefficient vector, making it too difficult for the following mixture model analysis.

As the FPCA turns out that six principal components already explain more than 90% of the total variation in the signal trajectories for each node, we fix $M = 6$ as the truncation number for the Karhunen-Loève decomposition. The multivariate Karhunen-Loève expansion basis coefficient (principal component score) vectors $\mathbf{a}_i^M$ with $M = 6$ are thus acquired for further mixture analysis assuming Gaussianity.

In the iterative EM process to analyze the mixture of blocked Gaussian multivariate vector graphical models, the fglasso algorithm employed in our method is compared with the ADMM algorithm, to solve the maximization problem of log-likelihood with
penalty for estimating the conditional dependence structures in each cluster. To provide
good initials for the EM iterations, the Mclust function, acquired from the R mclust
package, is applied to the multivariate principal component score vectors. Our MFGM
algorithm is also compared with the mixggm algorithm to confirm the advantage of
considering inherent functional nature of the data. First, the overall clustering is
checked. The metrics considered are Accu, MCC, TPR, and FPR. The corresponding
raw clustering results to calculate the four metrics by each method are detailed in Table
5.2. Then, more importantly, the estimates of the edge structures in each subgroup by
the three methods are compared.

5.2 Results

As shown in Table 5.1, in overall clustering, our method and ADMM do well as
reflected by the high values of turnouts for the metrics Accuracy and TPR, but both
provide relatively poor results as reflected by low MCC values and high FPR values.
In contrast, the mixggm does a very poor job as revealed by poor turnouts for all of
the four metrics.

Table 5.1: Comparisons of MFGM with ADMM and mixggm in terms of overall
clustering in EEG data analysis.

<table>
<thead>
<tr>
<th></th>
<th>Accu</th>
<th>MCC</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFGM</td>
<td>0.7295</td>
<td>0.3904</td>
<td>0.9091</td>
<td>0.5778</td>
</tr>
<tr>
<td>ADMM</td>
<td>0.7131</td>
<td>0.3708</td>
<td>0.8052</td>
<td>0.4444</td>
</tr>
<tr>
<td>mixggm</td>
<td>0.5984</td>
<td>0.1334</td>
<td>0.6883</td>
<td>0.5556</td>
</tr>
</tbody>
</table>

Figure 5.2 depicts the estimated brain nodes connection structures in each clustered
subgroup by the three methods. Our MFGM method reveals that, in both subgroups,
the electrode locations from the frontal region are densely connected, and the electrode
locations from other regions of the scalp tend to be only sparsely connected. This is consistent with the findings reported by a popular functional graphical models study that analyzed the same EEG dataset (Qiao et al., 2019). We also notice that, the nodes connection structure in the frontal region in the alcoholic subgroup has an asymmetric pattern compared to a symmetric pattern in the control; and the Fz electrode-located region has a little more connection with the adjacent regions in the alcoholic subgroup than that in the control, but the Cz electrode-located region has less connection with the adjacent regions in the alcoholic subgroup than that in the control. Moreover, very sparse connections in the lower left Temporal region and Occipital region are revealed in the alcoholic subgroup compared to none in the control. The ADMM algorithm reveals significant distinction between the two subgroups. Very dense regional connections are found all over the whole brain in the control. In contrast, very sparse regional connections are shown in the alcoholic subgroup except the Occipital region and the lower Temporal regions. These findings do not match that in Qiao et al. (2019) at all. Finally, the mixggm algorithm estimates super dense regional connections in both of the two subgroups, which doesn’t make any meaningful sense.

To point out, our MFGM algorithm doesn’t provide very good overall clustering for the EEG data, which is reflected by the relatively weak values of the two metrics MCC and FPR, probably due to the high variation and complexity of the nature of the functional EEG signals, or due to the heterogeneity in each subgroup caused by some covariates, e.g., gender, age, et al., the information of which is not available from

<table>
<thead>
<tr>
<th>MFGM</th>
<th>Real Labels</th>
<th>Estimated Labels</th>
<th>ADMM</th>
<th>Real Labels</th>
<th>Estimated Labels</th>
<th>mixggm</th>
<th>Real Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Control</td>
<td>Alcoholic</td>
<td></td>
<td>Control</td>
<td>Alcoholic</td>
<td></td>
<td>Control</td>
</tr>
<tr>
<td>Estimated Labels</td>
<td>19</td>
<td>7</td>
<td>25</td>
<td>15</td>
<td>20</td>
<td>24</td>
<td>20</td>
</tr>
<tr>
<td>Estimated Labels</td>
<td>29</td>
<td>79</td>
<td>20</td>
<td>62</td>
<td>25</td>
<td>53</td>
<td>25</td>
</tr>
</tbody>
</table>
Figure 5.2: Comparisons of conditional dependence structures in Alcoholic and Control subgroups estimated by MFGM, ADMM, and mixggm algorithms. Solid edges in black color: common in both of Alcoholic and Control; dash-dotted edges in red color: specific in Alcoholic; dash-dotted edges in blue color: specific in Control.

the dataset, unfortunately. However, our estimated conditional dependence structures of the brain networks in each subgroup are very consistent with a previously reported study (Qiao et al., 2019). Moreover, by analyzing the frequency power of the EEG signals, Hayden et al. (2006) showed that alcoholism in people may lead to frontal lobe dysfunction and alcoholism subjects exhibit a pattern of frontal asymmetry similar to that found in other psychiatric group. Our study echoes their findings from another aspect by estimating asymmetric brain network in the alcoholic subgroup and symmetric brain network in the control.

The two electrodes O1 and O2 are located above the primary visual cortex, the simplest, earliest cortical visual area, which is highly specialized for processing information about static and moving objects and is excellent in pattern recognition. The
two electrode P3 and P7 are located in the Brodmann area 37 in the left hemisphere. The Brodmann area 37 is part of the temporal cortex in the human brain. It contains the fusiform gyrus which in turn contains the fusiform face area, a part of the human visual system, an area important for the recognition of faces. Therefore, the exclusive connection structures between O1 and O2 as well as P3 and P7 in the alcoholic subgroup may suggest the distinction of Cerebral response to visual stimuli and visual recognition between alcoholic subjects and controls. The brain networks estimated by the ADMM algorithm do not match that in our study and that in the previously reported study (Qiao et al., 2019), which may suggest that the assumption of partial separability in the proposed functional Gaussian graphical models could be invalid for the EEG data analysis. The mixggm algorithm provides very poor estimates for the brain networks in each subgroup, which may be due to the reasons as follows: 1) The performance that takes average of observations across the time interval for each node, ignoring the inherent functional nature in the data, could be really invalid in functional data analysis; 2) The mixggm algorithm provides very poor overall clustering for the EEG data, therefore, it is hard to expect good estimates of the networks in each subgroup based on the poor overall clustering results.

To sum up, our MFGM algorithm outperforms the other two competing algorithms in the real-world EEG data analysis, especially in estimating the brain networks in the alcoholic subgroup and the control.
Chapter 6

Conclusion and Future Work

6.1 Conclusion

Graphical models use a graph-based representation to express the conditional dependence structure between random variables. The real-world data often come from different sources and may have heterogeneous dependencies across the whole population, which violates the common assumption of the graphical models that the high-dimensional data come from a homogeneous source and follow a parametric or semi-parametric graphical model. Mixture of graphical models show high efficiency to identify subpopulations that share hidden commonality and further depict heterogeneous conditional dependencies across the whole population in vector-valued context. In addition, FDA changes the frame of functional data by making the fundamental statistical unit a function or curve, other than the vector of measurements. We propose the MFGM method that takes advantage of both mixture of graphical models and FDA for the generalization of mixture of graphical models from vector context to functional context. Our MFGM method employs an effective and efficient EM algorithm with theoretically guaranteed ascent property and convergence to a stationary point under
high dimensions, solving the maximization problem of log-likelihood with penalty, which is non-convex and non-smooth, to estimate the graphical model parameters for each subgroup.

In the simulation studies, we first prepare two-cluster mixture multivariate functional designs: Independent model mixed with AR2, AR1 mixed with AR2 that contains weak connection edges, and AR2 mixed with model that contains random weak connection edges. To explore even more complex scenarios, we prepare three-cluster mixture multivariate functional designs to analyze: Independent model mixed with AR1 and AR2, Random model mixed with AR1 and AR2. The simulation studies show that, our MFGM method either outperforms or is as good as the two potential competing algorithms ADMM and mixggm, evaluated by both of overall clustering and estimating the conditional dependencies in each subgroup. This lays a solid foundation for the next step to investigate its ability to analyze the real-world functional mixture data.

The real-world EEG dataset consisted of multivariate functional observations from two subgroups, alcoholc and control, with certain commonality as well as heterogeneous conditional dependencies within an overall population. Our MFGM method doesn’t provide very good overall clustering for the EEG data, which is reflected by the relatively weak values of the two metrics MCC and FPR, probably due to the high variation and complexity of the nature of the functional EEG signals, or due to the impacts of some confounding variables, e.g., gender, age, et al., the information of which is not available from the dataset, unfortunately. However, our estimated regional networks of the brain in each subgroup show that, in both subgroups, the electrode locations from the frontal region are densely connected, and the electrode locations from other regions of the scalp tend to be only sparsely connected, which is very consistent with a previously reported study (Qiao et al., 2019). Moreover, another reported study (Hayden et al.,
2006) analyzed the frequency power of the EEG signals and showed that alcoholism in people may lead to frontal lobe dysfunction and alcoholism subjects exhibit a pattern of frontal asymmetry similar to that found in other psychiatric group. Our study echoes their findings from another aspect by estimating asymmetric brain network in the alcoholic subgroup and symmetric brain network in the control. Our MFGM method outperforms the two potential competing algorithms ADMM and mixggm in this real-world data analysis, especially in estimating the brain regional networks in each subgroup.

To sum up, our proposed MFGM method, motivated by not only clustering mixture functional observations into subgroups, but more importantly, inferring heterogeneous conditional dependencies in each subgroup of the high-dimensional data, may greatly extend the methodology and applicability of high-dimensional graphical models, and provide a novel strategy for complex functional data analysis.

6.2 Future Work

Even though our work reveals high efficiency of the proposed MFGM method as evaluated by both of overall clustering and estimation of the conditional dependence structures in each subpopulation, there are some challenges yet to overcome in this project and further work is needed to improve our proposed strategy: 1) The truncation number $M$ for Karhunen-Loève expansion is selected based on CV in the simulation studies, which doesn’t work well in EEG real-world data analysis. Other criteria, such as AIC (Yao et al., 2005) or BIC, can be performed and compared. 2) In this work, we arbitrarily select the number of subclusters $K$ for the whole population. Some reliable criteria, such as AIC, BIC, or Integrated Classification Likelihood (ICL) (Morvan et al., 2021), can be explored for selection of optimal cluster number, although
it is challenging to find the right degrees of freedom in our complex models (Qiao et al., 2019). 3) To reduce the dimension of the multivariate data, and further reduce the computation time, we only choose nineteen (10–20 International system) out of sixty-four electrode regions, which may omit some regional connections that are very informative. In the following study, we will take into account all sixty-four electrode regions for brain network analysis and see if our method can handle the even higher dimensional scenarios or not.


