Mixture of Gaussian Processes and its Applications

Mian Huang, Runze Li, Hansheng Wang, and Weixin Yao

*The Pennsylvania State University*

*Technical Report Series*  
#10-102

*College of Health and Human Development*  
*The Pennsylvania State University*
Mixture of Gaussian Processes and its Applications

MIAN HUANG, RUNZE LI, HANSHENG WANG, AND WEIXIN YAO

July 19, 2010

Abstract

The rapid development of information technologies enables researchers to collect and store functional data at a low cost. As a result, the quantitative analysis of functional data becomes practically feasible, which naturally calls for new statistical methods to serve such a purpose. To this end, we propose a new model, namely, “Mixture of Gaussian Processes” in this paper. Our method can be viewed as a natural extension of high-dimensional normal mixtures. However, the key difference is that smoothed structures are imposed for both the mean and covariance functions. As a consequence, our model can be estimated efficiently by a novel combination of the ideas from EM algorithm, kernel regression, and functional principal component analysis. It is remarkable that no high-dimensional covariance matrix needs to be estimated in our computational process. This has been an inevitable step for a general normal mixture model and suffers from its estimation instability and inaccuracy. The proposed methodology is empirically justified by Monte Carlo simulations and illustrated by an analysis of a supermarket dataset is illustrated.

Keywords: EM algorithm, kernel regression, Gaussian process, Functional principal component analysis

1Mian Huang is Assistant Professor, School of Statistics and Management, Shanghai University of Finance and Economics, Shanghai, P. R. China. Runze Li is Professor, Department of Statistics, The Pennsylvania State University University Park, PA 16801, USA. Hansheng Wang is Professor, Department of Business Statistics and Econometrics, Peking University, Beijing, P.R. China. Weixin Yao is Assistant Professor, Department of Statistics, Kansas State University, Manhattan, Kansas 66506, USA. Huang’s research is partially supported by a National Science Foundation grant DMS 0348869, National Institute on Drug Abuse grants P50 DA10075, and funding through Projet 211 Phase 3 of SUFE, and Shanghai Leading Academic Discipline Project, B803. Li’s research is supported by National Institute on Drug Abuse grants R21 DA024260. Wang’s research is supported in part by a NSFC grant 10771006. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIDA or the NIH.
1 Introduction

The rapid development of information technologies enables researchers to collect and store functional data at a low cost. As a result, the quantitative analysis of functional data becomes practically feasible; see Ramsay and Silverman (2005) for a comprehensive and excellent treatment. The basis of functional data analysis consists of two parts: the estimations of the mean function and of the covariance structure. Among many approaches, functional principal component (FPC) analysis serves as a key technique in functional data analysis. Rice and Silverman (1991) and James et al. (2000) studied the spline smoothing methods in FPC analysis; Staniswalis and Lee (1998) and Yao et al. (2003) applied kernel-based smoothing methods for FPC analysis in irregular and sparse longitudinal data. The asymptotic properties of principal component functions are investigated in Yao et al. (2005) and Hall et al. (2006).

As an illustration of functional data, Figure 1 depicts the plot of a set of collected curves. This data set contains the number of customers who visited a particular supermarket in China on each of 139 days. For each day, the number of customers shopping in the supermarket is observed every half hour from 7:00am to 5:30pm. Thus, there are 22 observations for each day. The collected time was coded as 1 for 7:00am, 2 for 7:30am, and so on. In the analysis of this dataset, we regard each day as one subject. Thus, we have a total of 139 subjects. Figure 1 shows that the variability may be large in certain time periods. Intuitively, the customer flow (i.e., the number of customers) may show different patterns in weekdays, weekends and holiday season, and hence the data is likely inhomogeneous. To statistically model such a multivariate response, we may naively consider a mixture of 22-dimensional multivariate normal distributions. Nevertheless, we find this method less effective because the $22 \times 22$ covariance matrix for each component has to be estimated. With such a limited sample size (i.e, 139), the estimated covariance matrices are likely to be ill-conditioned. As a consequence, the estimation accuracy of its inverse is very poor. This motivates us to develop a class of new models for analysis of inhomogeneous functional data.

As an interesting alternative to mixture of high-dimensional normals, we propose a mixture of Gaussian processes for functional data. As compared with a general normal mixture,
the major advantage of our method is that smoothed structures are imposed for both the mean and covariance functions. As a consequence, the unknown functions can be estimated efficiently by a novel combination of the ideas from EM algorithm, kernel regression, and functional principal component analysis. Therefore, the challenging task of high-dimensional covariance matrix estimation may be completely avoided. In addition, the proposed mixture models can deal with data collected at irregular, possibly subject-dependent time points. This leads to variation in the number of observations from subject to subject. It is clear that a mixture of multivariate normals is not applicable for such data. In this paper, we shall systematically study the mixture of Gaussian processes. We first propose an estimation procedure for the mean functions using kernel regression. We further develop an estimation procedure for covariance functions using functional principal component analysis. We empirically test the proposed algorithms by Monte Carlo simulations, and further apply the proposed procedure to analysis of a supermarket dataset.

The rest of this paper is structured as follows. We present the definition of mixture of
Gaussian processes in section 2. In section 3, we develop an estimation procedure for the newly proposed models. Simulation results and an empirical analysis of supermarket dataset are presented in section 4. Concluding remarks and discussion are given in section 5.

2 Model Definition and Observed Data

Let $C$ be a latent class variable with a discrete distribution $P(C = c) = \rho_c$ for $c = 1, 2, \cdots, C$. It is assumed in this paper that $C$ is fixed and known. We will briefly discuss how to determine $C$ in section 3. Given $C = c$, $\{X(t), t \in T\}$ follows a Gaussian process with mean $\mu_c(t)$ and covariance function $\text{Cov}\{X(s), X(t)\} = G_c(s, t)$. We refer to $\{X(t) : t \in T\}$ as a mixture of Gaussian processes. Typically, $T$ is a closed and bounded time interval. It is assumed throughout this paper that $\mu_c(t)$ is a smooth function of $t$, and $G_c(s, t)$ is a positive definite and bivariate smooth function of $s$ and $t$. Thus, the path of $X(t)$ indeed is a smooth function.

As a covariance function, $G_c(s, t)$ can be represented as

$$G_c(s, t) = \sum_{q=1}^{\infty} \lambda_q v_q(s)v_q(t),$$

where $\lambda_q$’s are eigenvalues, and $v_q(\cdot)$’s are eigenfunctions. Furthermore, we have $\lambda_{1c} \geq \lambda_{2c} \geq \cdots$, and $\sum_q \lambda_q < \infty$, for $c = 1, \cdots, C$. By the Karhunen-Loève theorem, if the $i$-th subject $X_i(t)$ is from the $c$-th component, then it can be represented as follows

$$X_i(t) = \mu_c(t) + \sum_{q=1}^{\infty} \xi_{iqc} v_q(t),$$

where the functional principal component score $\xi_{iqc}$ is considered as independent random variables with $E(\xi_{iqc}) = 0$, and $\text{Var}(\xi_{iqc}) = \lambda_q$.

Since the sample path of $X_i(t)$ is a smooth function of $t$, $X_i(t)$ is termed a smooth random function (Yao, et al., 2005). As depicted in Figure 1, the collected sample of random curves are typically not smooth in practice. Following Yao et al. (2003), it is assumed that the observed curve $\{y_i(t), t = t_{ij}, j = 1, \cdots, N_i\}$ is

$$y_i(t) = X_i(t) + \epsilon_i(t),$$
where $\epsilon_i(t)$ is additive measurement error, and it is assumed that $\epsilon_i(t_{ij})$, for all $i$ and $j$, are independent and identically distributed as $N(0, \sigma^2)$. Denote $y_{ij} = y_i(t_{ij})$ and $\epsilon_{ij} = \epsilon_i(t_{ij})$. Throughout this paper, it is assumed that conditioning on $C = c$, the observations $y_{ij}$, $j = 1, \cdots, N_i$ and $i = 1, \cdots, n$, follows

$$y_{ij} = \mu_c(t_{ij}) + \sum_{q=1}^{\infty} \xi_{iqc}v_{qc}(t_{ij}) + \epsilon_{ij},$$

(2.1)

where $\epsilon_{ij}$s are independent and identically distributed of $N(0, \sigma^2)$. We shall propose an estimation procedure for $\rho_c$, $\mu_c(\cdot)$, $v_{qc}(\cdot)$ and $\sigma^2$ in section 3.

3 Estimation Procedure

3.1 Initial Estimation of the Mean Functions

For the mixture of Gaussian processes, it is inevitable to estimate the mean functions first, and then estimate the covariance function based on the residuals. As demonstrated in Lin and Carroll (2000), the kernel generalized estimating equation (GEE) method for repeated measurement data yields an optimal estimate in a certain sense by pretending the data within subjects are independent. Furthermore, kernel GEE method with working independent covariance structure is easy to implement. Therefore it is natural to develop an estimation procedure for the mean function with working independent covariance structure. In what follows, we shall develop an estimation procedure for $\mu_c(\cdot)$ by using kernel regression techniques and pretending that the data within subjects are independent. We will further develop a refined estimation procedure by incorporating within-subjects correlation in section 3.3.

By working independent correlation, we mean that conditioning on $C = c$, $G_c(s, t) = 0$ if $s \neq t$. Let $\sigma^*c^2(t) = G_c(t, t) + \sigma^2$, it follows that

$$y_{ij} = \mu_c(t_{ij}) + \epsilon^*_{ij},$$

(3.1)

where $\epsilon^*_{ij}$ are independent with $E(\epsilon^*_{ij}) = 0$ and $\text{Var}(\epsilon^*_{ij}) = \sigma^*c^2(t_{ij})$. This is equivalent to treating $y_{ij}$s sampled from the following distribution:

$$y(t) \sim \sum_{c=1}^{C} \rho_cN\{\mu_c(t), \sigma^*c^2(t)\}.$$

(3.2)
For model (3.2), the log-likelihood function of the collected data is

\[
\sum_{i=1}^{n} \log \left[ \sum_{c=1}^{C} \rho_c \prod_{j=1}^{N_i} \phi\{y_{ij} | \mu_c(t_{ij}), \sigma_c^2(t_{ij})\} \right].
\]

(3.3)

We now propose an EM-type algorithm to maximize (3.3). Define the membership identity random variables

\[
z_{ic} = \begin{cases} 
1, & \text{if } \{X_i(t), t \in T \} \text{ is in the } c^{th} \text{ group,} \\
0, & \text{otherwise.}
\end{cases}
\]

Thus, the complete likelihood of \{\{(y_{ij}, Z_{ic}), j = 1, \cdots, N_i, i = 1, \cdots, n, c = 1, \cdots C\}\} is

\[
\prod_{i=1}^{n} \prod_{c=1}^{C} \prod_{j=1}^{N_i} \rho_c \phi\{y_{ij} | \mu_c(t_{ij}), \sigma_c^2(t_{ij})\}^{z_{ic}}.
\]

After the \(l\)-th iteration of the EM algorithm, suppose that we have \(\rho_c^{(l)}, \sigma_{c}^2(t_{ij})^{(l)}\), and \(\mu_c^{(l)}(\cdot)\). Thus, in the E-step of the \((l+1)\)-th iteration, the expectation of the latent variable \(z_{ic}\) is given by

\[
r_{ic}^{(l+1)} = \frac{\rho_c^{(l)} \left[ \prod_{j=1}^{N_i} \phi\{y_{ij} | \mu_c^{(l)}(t_{ij}), \sigma_{c}^2(t_{ij})^{(l)}\} \right]}{\sum_{c=1}^{C} \rho_c^{(l)} \left[ \prod_{j=1}^{N_i} \phi\{y_{ij} | \mu_c^{(l)}(t_{ij}), \sigma_{c}^2(t_{ij})^{(l)}\} \right]}.
\]

(3.4)

In the M-step of the \((l+1)\)-th iteration, we would maximize the logarithm of complete likelihood function with \(z_{ic}\) replaced by \(r_{ic}^{(l+1)}\), which is

\[
\sum_{i=1}^{n} \sum_{c=1}^{C} \left[ r_{ic}^{(l+1)} \log(\rho_c) + r_{ic}^{(l+1)} \sum_{j=1}^{N_i} \log \phi\{y_{ij} | \mu_c(t_{ij}), \sigma_{c}^2(t_{ij})\} \right].
\]

This leads to

\[
\rho_c^{(l+1)} = \frac{1}{n} \sum_{i=1}^{n} r_{ic}^{(l+1)}.
\]

(3.5)

Note that both \(\mu_c(\cdot)\) and \(\sigma_{c}^2(\cdot)\) are nonparametric smoothing function. Here we use kernel regression to estimate \(\mu_c(\cdot)\)'s and \(\sigma_{c}^2(\cdot)\)'s. For any \(t_0 \in T\), we approximate \(\mu_c(t_{ij})\) by \(\mu_c(t_0)\) and \(\sigma_{c}^2(t_{ij})\) by \(\sigma_{c}^2(t_0)\) for \(t_{ij}\) in the neighborhood of \(t_0\). Thus, the corresponding local log-likelihood function is

\[
\sum_{i=1}^{n} \sum_{c=1}^{C} \sum_{j=1}^{N_i} \left[ \log \phi\{y_{ij} | \mu_c(t_0), \sigma_{c}^2(t_0)\} \right] K_h(t_{ij} - t_0),
\]

(3.6)
where \( K_h(t) \) is a rescaled kernel function \( h^{-1}K(t/h) \) with a kernel function \( K(t) \). Maximizing (3.6) with respect to \( \mu_c(t_0) \) and \( \sigma_c^2(t_0), c = 1, \cdots, C \), yields

\[
\begin{align*}
\mu_c^{(l+1)}(t_0) &= \frac{\sum_{i=1}^{n} \sum_{j=1}^{N_i} w_{cij}^{(l+1)} y_{ij}}{\sum_{i=1}^{n} \sum_{j=1}^{N_i} w_{cij}^{(l+1)}}, \\
\sigma_c^{2(l+1)}(t_0) &= \frac{\sum_{i=1}^{n} \sum_{j=1}^{N_i} w_{cij}^{(l+1)} \left\{ y_{ij} - \mu_c^{(l+1)}(t_0) \right\}^2}{\sum_{i=1}^{n} \sum_{j=1}^{N_i} w_{cij}^{(l+1)}},
\end{align*}
\]

(3.7) (3.8)

where \( w_{cij}^{(l+1)} = r_{ic}^{(l+1)} K_h(t_{ij} - t_0) \). In practice, we evaluate the estimates at a set of grid points for the given label in the E-step. Let \( \{u_1, \cdots, u_{n_{grid}}\} \) be a set of grid points at which the estimated functions are evaluated, where \( n_{grid} \) is the number of grid points. If the total number of observations \( J = \sum_{i=1}^{n} N_i \), is not very large, we can directly use all the time points as the grid points. Otherwise, we update \( \mu_c(t_{ij}) \) and \( \sigma_c^{2}(t_{ij}), i = 1, \cdots, n, j = 1, \cdots, N_i \) by linearly interpolating \( \mu_c^{(l+1)}(u_k) \) and \( \sigma_c^{2(l)}(u_k), k = 1, \cdots, n_{grid} \). Denoted by \( \hat{\rho}_c, \hat{\mu}_c(\cdot), \) and \( \hat{\sigma}_c^{2}(\cdot) \) the resulting estimate of \( \rho_c, \mu_c(\cdot), \) and \( \sigma_c^{2}(\cdot) \), respectively.

### 3.2 Estimation of Covariances

We now deal with estimation of covariance functions using functional principal analysis. Let \( \bar{G}_{ic}(t_{ij}, t_{il}) = \{y_{ij} - \hat{\mu}_c(t_{ij})\} \{y_{il} - \hat{\mu}_c(t_{il})\} \). Note that given \( C = c \), \( \text{Cov}\{Y(t), Y(t)\} = G_c(t, t) + \sigma^2 \), and \( \text{Cov}\{Y(s), Y(t)\} = G_c(s, t) \) for \( s \neq t \). If \( z_{ic} \) were observable, then the covariance function \( G_c(s, t) \) could be estimated by a two-dimensional kernel smoother, which is to minimize

\[
\sum_{i=1}^{n} z_{ic} \sum_{1 \leq j \neq l \leq N} \left[ \bar{G}_{ic}(t_{ij}, t_{il}) - \beta_0 \right]^2 K_h^*(t_{ij} - s)K_h^*(t_{il} - t),
\]

(3.9)

with respect to \( \beta_0 \). In practice, \( z_{ic} \) is a latent variable. Following the idea of the EM algorithm, we replace \( z_{ic} \) by its expectation \( r_{ic} \), which was obtained in the estimation procedure for \( \mu_c(\cdot) \) described in section 3.1. Thus, we minimize

\[
\sum_{i=1}^{n} r_{ic} \sum_{1 \leq j \neq l \leq N} \left[ \bar{G}_{ic}(t_{ij}, t_{il}) - \beta_0 \right]^2 K_h^*(t_{ij} - s)K_h^*(t_{il} - t),
\]

(3.10)
with respect to $\beta_0$. The minimizer $\hat{G}_c(s, t) \equiv \hat{\beta}_0$ of (3.10) has a closed form solution, given by

$$\hat{G}_c(s, t) = \frac{\sum_{i=1}^{n} r_{ic} \sum_{1 \leq j \neq l \leq N_i} \hat{G}_c(t_{ij}, t_{il}) K_{h^*}(t_{ij} - s) K_{h^*}(t_{iq} - t)}{\sum_{i=1}^{n} r_{ic} \sum_{1 \leq j \leq N_i} K_{h^*}(t_{ij} - s) K_{h^*}(t_{iq} - t)}. \quad (3.11)$$

Following Rice and Silverman (1991), the estimation of eigenvalues and eigenfunctions are based on discretizing the covariance estimate $\hat{G}_c(s, t)$. The estimates of eigenvalues $\hat{\lambda}_{qc}$ and eigenfunctions $\hat{v}_{qc}(\cdot)$ are determined by eigenfunctions

$$\int_T \hat{G}_c(s, t) \hat{v}_{qc}(s) ds = \hat{\lambda}_{qc} \hat{v}_{qc}(t), \quad (3.12)$$

where $\hat{v}_{qc}(t)$ satisfies $\int_T \hat{v}_{qc}^2(t) dt = 1$, and $\int_T \hat{v}_{pc}(t) \hat{v}_{qc}(t) dt = 0$ if $p \neq q$. Then, in order for the resulting estimate of $G_c(s, t)$ to be positive definite, we set

$$\hat{G}_c(s, t) = \sum_q \hat{\lambda}_{qc} I(\hat{\lambda}_{qc} > 0) \hat{v}_{qc}(s) \hat{v}_{qc}(t).$$

### 3.3 A Refined Estimation Procedure for the Mean Functions

Given $\hat{\mu}_c(t)$ and $\hat{v}_{qc}(t)$, the functional principal component score $\xi_{iqc}$ can be estimated by

$$\hat{\xi}_{iqc} = \int_T \{y_i(t) - \hat{\mu}_c(t)\} \hat{v}_{qc}(t) dt. \quad (3.13)$$

Furthermore, for $j = 1, \cdots, N_i$ and $i = 1, \cdots, n$, define

$$\hat{\eta}_c(t_{ij}) = \sum_q \hat{\xi}_{iqc} I(\hat{\lambda}_{qc} > 0) \hat{v}_{qc}(t_{ij}), \quad (3.14)$$

which is an estimate of

$$\eta_c(t_{ij}) = \sum_q \xi_{iqc} I(\lambda_{qc} > 0) v_{qc}(t_{ij}).$$

Let

$$y^*_c(t_{ij}) = y_{ij} - \hat{\eta}_c(t_{ij}). \quad (3.15)$$

Then, conditioning on $C = c$, the model (2.1) can be approximated by

$$y^*_c(t_{ij}) \approx \mu_c(t_{ij}) + \epsilon_{ij}, \quad (3.16)$$
where \( \epsilon_{ij} \)'s are independent and identically distributed as \( N(0, \sigma^2) \). Hence, with the aid of functional PCA, we can transform the correlated data to uncorrelated data with a few eigenvalues and eigenfunctions from the estimate of \( G_c(s, t) \). Based on \( \{ y_c^*(t_{ij}), i = 1, \ldots, n, j = 1, \ldots, N_i, c = 1, \ldots, C \} \), the EM-type algorithm in section 3.1 can be adapted here to further improve the estimate of \( \mu_c(t), \sigma^2, \) and \( \rho_c \)'s. Slight revision is made according to the constant variance of (3.16), which is different from (3.1). Specifically, in the E-step we find the probability

\[
r_{ic}^{(l+1)} = \frac{\rho_c^{(l)}}{\sum_{c=1}^C \rho_c^{(l)}} \left[ \prod_{j=1}^{N_i} \phi\{y_c^*(t_{ij}) | \mu_c^{(l)}(t_{ij}), \sigma_c^{2(l)} \} \right] \]

(3.17)

In the M-step, we update the estimates of \( \mu_c(t), \rho_c, \) and \( \sigma^2 \). For \( t_0 \in \{ u_1, \ldots, u_{n_{grid}} \} \),

\[
\mu_c^{(l+1)}(t_0) = \frac{\sum_{i=1}^n \sum_{j=1}^{N_i} w_{cij}^{(l+1)} y_c^*(t_{ij})}{\sum_{i=1}^n \sum_{j=1}^{N_i} w_{cij}^{(l+1)}},
\]

(3.18)

where \( w_{cij}^{(l+1)} = r_{ic}^{(l+1)} K_h(t_{ij} - t_0) \), and

\[
\rho_c^{(l+1)} = \frac{1}{n} \sum_{i=1}^n r_{ic}^{(l+1)},
\]

(3.19)

\[
\sigma_c^{2(l+1)} = \frac{1}{n^2 \sum_{i=1}^n N_i \sum_{c=1}^C \sum_{j=1}^{N_i} r_{ic}^{(l+1)}} \sum_{i=1}^n \sum_{j=1}^{N_i} \sum_{c=1}^C \sum_{j=1}^{N_i} \left( y_{ij} - \mu_c^{(l+1)}(t_{ij}) \right)^2.
\]

(3.20)

Furthermore, we update \( \{ \mu_c^{(l+1)}(t_{ij}), i = 1, \ldots, n, j = 1, \ldots, N_i \} \) by linearly interpolating \( \mu_c^{(l+1)}(u_k), k = 1, \ldots, n_{grid} \).

To improve the estimation, we can also further iterate the above procedure between estimating the covariance structure \( G_c(\cdot, \cdot) \) and the component mean functions \( \mu_c(\cdot) \). Hence we have the following iterative estimation procedure.

### 3.4 An Iterative Estimation Procedure

The proposed estimation procedures in sections 3.1, 3.2 and 3.3 can be summarized in the following algorithm:

**Step 1:** Calculate \( \hat{\mu}_c(\cdot) \) using the EM-type algorithm of (3.4)--(3.8).


**Step 2:** Given $\hat{\mu}_c(\cdot)$, and $r_{ic}s$, obtain $\hat{G}_c(s, t)$ using (3.11) and calculate $\hat{\eta}_c(t_{ij})$ using (3.12), (3.13), and (3.14).

**Step 3:** Calculate $y^*_c(t_{ij})$ in (3.15), update $\mu_c(t), \sigma^2$, and $\rho_{cs}$ using (3.17)–(3.20).

Iteratively calculate Step 2 and Step 3 until convergence.

It is worth noting that this newly proposed procedure is easy to implement, since it avoids the disadvantages of high-dimensional mixture of normals, i.e., the calculation of inverse of the covariance matrix.

### 3.5 Practical Implementation Issues

**Choice of the number of component**

Choosing the number of component $C$ is a critical issue for mixture models. Here we outline a simple approach to determine $C$ by using the AIC for finite mixture of low dimensional multivariate normals. Direct implementation of the AIC for mixture of Gaussian processes is difficult since the degrees of freedom for mixture of Gaussian processes is not well defined. As a practical alternative, we recommend applying the AIC with a finite mixture of multivariate normals for partial of the observed data. Specifically, like the supermarket data introduced in section 1, if the data are observed at $(t_1, \cdots, t_N)$ for all subjects, then we may take the partial data observed at $(t_{k_1}, \cdots, t_{k_N'})$, a subset of $(t_1, \cdots, t_N)$. In practice, the subset $(t_{k_1}, \cdots, t_{k_N'})$ can be every $d$ points of $(t_1, \cdots, t_N)$ for some $d \geq 2$. For irregular and unbalanced data, one may either bin the data over the observed times or interpolate the data over a regular grid points, and then further use the AIC to the partial data of the binned data or interpolated data. By using partial data, we are able to determine $C$ before analysis using the proposed procedure, and avoid the disadvantages of high-dimensional mixtures of normals. This has been implemented in our analysis in section 4.2.

**Bandwidth selection**

Bandwidth selection is another important issue to be addressed. In the proposed estimation procedure, we need to select bandwidths for both mean functions and covariance functions. For the mean functions, we use the multi-fold cross-validation method to determine
the bandwidth. For the covariance functions, we use one-curve-leave-out cross-validation to choose this smoothing parameter, which has been used in the literature of covariance function smoothing (Rice and Silverman, 1991; Yao et al. 2005). The simulation results in section 4 demonstrate that the proposed estimation procedure works quite well in a wide range of bandwidths.

**Choice of the number of eigenfunctions**

A proper number of eigenfunctions is vital to provide a reasonable approximation to the Gaussian process in each component. Rice and Silverman (1991) suggested using the cross-validation method based on the one-curve-leave-out prediction error. Yao et al. (2005) investigated AIC-type criteria in functional principal component analysis, and found that while the AIC and cross-validation give similar results, the AIC is computationally more efficient than cross-validation method. In practice, empirical criterions are also useful to select the number of eigenfunctions. We may choose the number of eigenfunctions so that the percentage of total variation explained by the eigenfunctions is above a certain threshold, e.g., 85 percent or 90 percent.

**4 Simulation and Application**

In this section, we conduct numerical simulations to demonstrate the performance of the proposed estimation procedures. To assess the performance of the estimates of the unknown regression functions \( \mu_c(t) \), we consider the square root of the average squared errors (RASE) for mean functions,

\[
RASE^2_\mu = n^{-1}_{\text{grid}} \sum_{c=1}^{C} \sum_{j=1}^{n_{\text{grid}}} \{ \hat{\mu}_c(u_j) - \mu_c(u_j) \}^2,
\]

where \( \{u_j, j = 1, \ldots, n_{\text{grid}}\} \) are the grid points at which the unknown functions \( \mu_c(\cdot) \) are evaluated. For simplification, the grid points are taken evenly on the range of the \( t_{ij} \)'s. In the simulation, we set \( n_{\text{grid}} = 50 \). Similarly, we can define the RASE of the eigenfunctions for the \( c \)-th component, which is

\[
RASE^2_{v_c} = n^{-1}_{\text{grid}} \sum_{q=1}^{Q_c} \sum_{j=1}^{n_{\text{grid}}} \{ \hat{v}_{qc}(u_j) - v_{qc}(u_j) \}^2.
\]
We are also interested in the average of mean square of predicted error, given by

\[
\text{MSE} = \left( \sum_{i=1}^{N_i} \frac{1}{n} \sum_{i=1}^{N_i} \sum_{j=1}^{N_i} \left\{ y_{ij} - \sum_{c=1}^{C} \hat{r}_{ic} \hat{X}_{ic}(t_{ij}) \right\}^2 \right),
\]

where \( \hat{X}_{ic}(t_{ij}) = \hat{\mu}_c(t_{ij}) + \hat{\eta}_c(t_{ij}) \). MSE can be considered as a natural estimate of \( \sigma^2 \).

### 4.1 Simulation Study

In the following example, we generate data from a two-component mixture of Gaussian processes with

\[
\begin{align*}
\rho_1 &= 0.45, \quad \rho_2 = 1 - \rho_1 = 0.55, \quad \text{and} \quad \sigma^2 = 0.01, \\
\mu_1(t) &= \delta + 1.5 \sin(\pi t), \quad \text{and} \quad \mu_2(t) = \sin(\pi t), \\
v_{11}(t) &= \sqrt{2} \sin(4\pi t), \quad \text{and} \quad v_{12}(t) = \sqrt{2} \cos(4\pi t), \\
v_{21}(t) &= \sqrt{2} \sin(\pi t), \quad \text{and} \quad v_{22}(t) = \sqrt{2} \cos(\pi t).
\end{align*}
\]

The simulated data with sample size \( n = 50 \) are observed at grid points \( \{k/N, k = 1, \ldots, N\} \) for both components, where \( N \) is set to be 20 and 40. Let the eigenvalues for both components be \( \lambda_{11} = 0.04, \lambda_{12} = 0.01, \lambda_{21} = 0.04, \lambda_{22} = 0.01, \) and \( \lambda_{qc} = 0, \) for \( q > 2, c = 1, 2, \) and let the principal component scores \( \xi_{iqc} \) be generated from \( N(0, \lambda_{qc}), \) \( q = 1, 2, \) and \( c = 1, 2. \)

We consider two scenarios of simulation data sets from the above data generation scheme. In the first scenario, we set \( \delta = 0.5. \) As demonstrated in the typical sample depicted in Figure 2 (a), the subjects from the two components are well separated for this scenario. In the second scenario, we set \( \delta = 0, \) and the mean functions of the two components are close to each other. Thus, the subjects from the two components are heavily overlapping. A typical sample generated from this scenario is depicted in Figure 2 (b). In the following simulation, we compare the performance of two estimation procedures: the estimation of (3.2) using the EM-type algorithm, referred to as procedure of ‘working independent’; and the estimation of (2.1) using the iterative estimation procedure, referred to as procedure of ‘incorporating correlation.’ The comparisons are conducted in both the well-separated setting, and the heavy-overlap setting. For the heavy-overlap setting, we further investigate the performance of eigenfunction estimation.
Figure 2: (a) Typical sample data for the well-separated setting, $\delta = 0.5$; (b) Typical sample data for the heavy-overlap setting $\delta = 0$.

Table 1: Simulation Results for $\delta = 0.5$

<table>
<thead>
<tr>
<th>$N$</th>
<th>$h$</th>
<th>Working independent $\text{RASE}_{\mu}$</th>
<th>$\rho_1 = 0.45$</th>
<th>Incorporating correlation $\text{RASE}_{\mu}$</th>
<th>$\rho_1 = 0.45$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean(Std)</td>
<td>Mean(Std)</td>
<td>Mean(Std)</td>
<td>Mean(Std)</td>
</tr>
<tr>
<td>20</td>
<td>0.15</td>
<td>0.2157(0.0104)</td>
<td>0.4382(0.0688)</td>
<td>0.2200(0.0075)</td>
<td>0.4458(0.0679)</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>0.2061(0.0108)</td>
<td>0.4384(0.0687)</td>
<td>0.2105(0.0082)</td>
<td>0.4458(0.0679)</td>
</tr>
<tr>
<td></td>
<td>0.067</td>
<td>0.2063(0.0113)</td>
<td>0.4488(0.0698)</td>
<td>0.2105(0.0087)</td>
<td>0.4562(0.0694)</td>
</tr>
<tr>
<td>40</td>
<td>0.09</td>
<td>0.2119(0.0115)</td>
<td>0.4538(0.0707)</td>
<td>0.2164(0.0085)</td>
<td>0.4614(0.0702)</td>
</tr>
<tr>
<td></td>
<td>0.06</td>
<td>0.2089(0.0118)</td>
<td>0.4540(0.0705)</td>
<td>0.2135(0.0090)</td>
<td>0.4614(0.0702)</td>
</tr>
<tr>
<td></td>
<td>0.04</td>
<td>0.2086(0.0116)</td>
<td>0.4520(0.0713)</td>
<td>0.2110(0.0090)</td>
<td>0.4558(0.0711)</td>
</tr>
</tbody>
</table>
Table 2: Simulation Results for $\delta = 0$

<table>
<thead>
<tr>
<th></th>
<th>Working independent</th>
<th>Incorporating correlation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RASE$_{\mu}$</td>
<td>$\rho_1 = 0.45$</td>
</tr>
<tr>
<td>$N$</td>
<td>$h$</td>
<td>Mean(Std)</td>
</tr>
<tr>
<td>20</td>
<td>0.15</td>
<td>0.3533(0.0263)</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>0.3436(0.0232)</td>
</tr>
<tr>
<td></td>
<td>0.067</td>
<td>0.3508(0.0303)</td>
</tr>
<tr>
<td>40</td>
<td>0.09</td>
<td>0.3392(0.0222)</td>
</tr>
<tr>
<td></td>
<td>0.06</td>
<td>0.3358(0.0211)</td>
</tr>
<tr>
<td></td>
<td>0.04</td>
<td>0.3355(0.0246)</td>
</tr>
</tbody>
</table>

Table 3: Estimation of Eigenfunctions and Measurement Error ($\delta = 0$)

<table>
<thead>
<tr>
<th></th>
<th>RASE$_{v_1}$</th>
<th>RASE$_{v_2}$</th>
<th>MSE</th>
<th>$\hat{\sigma}^2 = 0.01$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>$h$</td>
<td>Mean(Std)</td>
<td>Mean(Std)</td>
<td>Mean(Std)</td>
</tr>
<tr>
<td>20</td>
<td>0.15</td>
<td>0.3974(0.3934)</td>
<td>0.3749(0.2800)</td>
<td>0.0112(0.0011)</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td>0.3862(0.3568)</td>
<td>0.3357(0.2796)</td>
<td>0.0093(0.0008)</td>
</tr>
<tr>
<td></td>
<td>0.067</td>
<td>0.3912(0.3582)</td>
<td>0.3341(0.2798)</td>
<td>0.0087(0.0007)</td>
</tr>
<tr>
<td>40</td>
<td>0.09</td>
<td>0.2136(0.0986)</td>
<td>0.2473(0.1469)</td>
<td>0.0100(0.0003)</td>
</tr>
<tr>
<td></td>
<td>0.06</td>
<td>0.2342(0.1165)</td>
<td>0.2302(0.1410)</td>
<td>0.0094(0.0004)</td>
</tr>
<tr>
<td></td>
<td>0.04</td>
<td>0.2182(0.1038)</td>
<td>0.2420(0.1449)</td>
<td>0.0093(0.0003)</td>
</tr>
</tbody>
</table>
In our simulation, we first generate several simulation datasets for a given sample size, and then use the CV bandwidth selectors to choose a bandwidth for each dataset. This provides an idea about the optimal bandwidth for a given sample size. To evaluate the performance of proposed procedures in a wide range of bandwidths, we consider three different bandwidths: two-thirds of the selected bandwidth, the selected bandwidth, and 1.5 times the selected bandwidth, which correspond to the under-smoothing, appropriate smoothing and over-smoothing, respectively. Table 1 displays the simulation results for the case of $\delta = 0.5$. The mean and standard deviation of $\text{RASE}_\delta$, and the estimate of $\rho_1$ over 100 simulations are recorded for both procedures. The results summarized in Table 1 show that the proposed procedures perform quite well for all three different bandwidths in the two procedures. This suggests that when the components are well separated, the estimation procedure incorporating correlations does not provide significant improvements compared to the working independent procedure. Table 2 depicts the simulation results for the case of $\delta = 0$. The mean and standard deviation of $\text{RASE}_\delta$, and the estimate of $\rho_1$ over 100 simulations are recorded for both procedures. Table 2 shows that the estimation procedure for working independent correlation performs poorly since the estimate of proportion parameter $\rho_1$ has large bias. However, as shown in Table 2, the estimation procedure incorporating correlations does give better results: smaller $\text{RASE}_\delta$s for the mean functions, and more accurate estimates of $\rho_1$ in a wide range of bandwidths. For $\delta = 0$, we further summarize the estimation of $\sigma^2$, MSE, and the RASE of the eigenfunctions for each component in Table 3. The results show that both the $\hat{\sigma}^2$ yielded by the iterative procedure and the MSE are good estimates of $\sigma^2$. In the heavy overlap setting, the proposed iterative procedure is able to provide a good estimate of the eigenfunctions.

### 4.2 Analysis of Supermarket Data

We use the proposed mixture of Gaussian processes and estimation procedure to analyze the supermarket dataset, which is depicted in Figure 1. We determine the number of component $C$ using some partial sparse data. To this end, the AIC for multivariate mixture of normals with one, two, three and four components are calculated and compared. We choose 4 sparse
datasets, which are taken from the original data for every 2,3,4,5 time locations. The AIC scores achieve the minimum at $C = 3$ for all the sparse datasets; thus, it is reasonable to select a 3-component model for analysis.

We first analyze the data using the working independent correlation model (3.1) with three components. Without loss of information, we transform the time interval of the data to $[0, 1]$. The smoothing parameter chosen by CV selector is 0.065. The estimated proportions of the three components (from up to down) are 0.1632, 0.4311, and 0.4057. The estimated mean functions and a hard-clustering result are shown in Figure 3(a). The hard-clustering is obtained by assigning component identities according to the largest $r_{ic}, c = 1, \cdots, C$. From this result and the original data with actual calender dates, we found that the days in the upper class are mainly from the month of Chinese spring festivals. Most Saturdays and Sundays fall in the middle class, and the weekdays generally fall in the lower class. The estimated mean functions can be viewed as estimated average customer flows of the three classes. We observed that there are two peaks of customer flows for 3 components. The first peak occurs around 9:00 am in all components. The second peak occurs around 2:00 pm for the first component, and 3:00 pm for the second and third component. This pattern may indicate that people tend to buy earlier in the afternoon during the days of spring festival. We further plot the estimated variance functions of the three components in Figure 3(b).
Combining Figure 3(a) and Figure 3(b), we observed that the variance functions followed a similar pattern with the mean functions in three components, in that a higher mean was associated with a higher variance.

The next step is to analyze the data by incorporating the estimated correlations. Based on the estimated posterior, we estimate the covariance functions and obtain estimates of the eigenfunctions of all the components. We plot the first two eigenfunctions of all components in Figure 4. For the upper class, the first eigenfunction explains 51.70% of the total variation, and has a negative value along its time interval from 9:00 am to 5:30 pm. It means that a subject of this class (i.e., a day) with a positive (negative) functional principal component score on this direction tends to have smaller (larger) customer flows than the population average in a whole observed time interval. We also observe that there are two negative peaks (corresponding to two lowest local minimums) in the first eigenfunction, which occurs around 9:00 am and 2:00 pm. It means that the variations of the customer flows are large in the two peaks, especially for the peak at 9:00 am. Note that these peaks are also observed in the first estimated variance function; therefore the results agree with each other as we expected.

The second eigenfunction, which explains 22.80% of the total variation, has relatively small negative values in the morning and large positive values in the afternoon. This means that a subject with a positive functional principal component score on this direction tends to have smaller customer flow in the morning and a higher customer flow in the afternoon. The variation characterized by the second eigenfunction has a minor magnitude compared to the

Figure 4: (a) First two eigenfunctions of the upper class; (b) First two eigenfunctions of the middle class; (c) First two eigenfunctions of the lower class.
variation in the first eigenfunction, where the magnitude is determined by the eigenvalues. The third and fourth eigenfunction explains 7.58% and 4.28% of the total variation, and is of little interest. The four principal components explain more than 85% percent of the total variation. Therefore, we think that using 4 eigenfunctions is enough for the analysis of the upper class. Similarly, we can compute and interpret the eigenfunctions of the second component and third component. Further analysis shows that incorporating the estimated covariances does not lead to significant improvements, and thus those results are not reported.

5 Discussion

In this paper, we propose mixture of Gaussian processes for analysis of inhomogeneous functional data. We impose smoothed structures for both mean and covariance functions in each component, and develop estimation procedures using kernel regression, EM algorithm, and functional principal component analysis. The proposed procedure overcomes several disadvantages of mixture of multivariate normals, such as “curse of dimensionality”, computational instability, and inability to handle irregular sparse data. We develop a kernel-type nonparametric regression technique for estimation of mixture of Gaussian processes.

The proposed mixture of Gaussian processes model provides an alternative to model-based functional clustering models. Cluster analysis for functional data has been studied by several authors. James and Sugar (2003) proposes a mixture-model-based clustering approach for functional data. In their approach, individual curves are represented by splines, and cast the problem into a parametric finite mixture of normals. In genetic research, model-based functional clustering methods have been used in Luan and Li (2003, 2004), in which spline method are used to model the mean function. Bayesian approaches for functional clustering models are studied in Heard et al. (2006), and Ma and Zhong (2008). Compared with the aforementioned works, we introduced functional principal component analysis for the estimation procedure, which provides the advantage of effective computation, i.e., avoids the inverse of high-dimensional covariance matrix. Functional principal component analysis also provides a powerful tool to interpret the results via the eigenvalue and eigenfunctions. We empirically justified the proposed estimation procedure by Monte Carlo simulations, and
gave detailed illustration in real data analysis.

References


