FUNCTIONAL DATA ANALYSIS: CLASSIFICATION AND REGRESSION

A Dissertation

by

HO-JIN LEE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2004

Major Subject: Statistics

FUNCTIONAL DATA ANALYSIS: CLASSIFICATION AND REGRESSION

A Dissertation

by

HO-JIN LEE

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Tailen Hsing (Chair of Committee)	
Bani K. Mallick (Member)	Faming Liang (Member)
Jianxin Zhou (Member)	Michael T. Longnecker (Head of Department)

August 2004

Major Subject: Statistics

ABSTRACT

Functional Data Analysis: Classification and Regression. (August 2004)

Ho-Jin Lee, B.E. Sung Kyun Kwan University, Korea;

M.S., Texas A&M University

Chair of Advisory Committee: Dr. Tailen Hsing

Functional data refer to data which consist of observed functions or curves evaluated at a finite subset of some interval. In this dissertation, we discuss statistical analysis, especially classification and regression when data are available in function forms. Due to the nature of functional data, one considers function spaces in presenting such type of data, and each functional observation is viewed as a realization generated by a random mechanism in the spaces. The classification procedure in this dissertation is based on dimension reduction techniques of the spaces. One commonly used method is Functional Principal Component Analysis (Functional PCA) in which eigen decomposition of the covariance function is employed to find the highest variability along which the data have in the function space. The reduced space of functions spanned by a few eigenfunctions are thought of as a space where most of the features of the functional data are contained. We also propose a functional regression model for scalar responses. Infinite dimensionality of the spaces for a predictor causes many problems, and one such problem is that there are infinitely many solutions. The space of the parameter function is restricted to Sobolev-Hilbert spaces and the loss function, so called, ϵ-insensitive loss function is utilized. As a robust technique of function estimation, we present a way to find a function that has at most ϵ deviation from the observed values and at the same time is as smooth as possible.

To my parents,

Arline

and You Young

ACKNOWLEDGEMENTS

I would like to express a measure of my sincere gratitude to my advisor Tailen Hsing, without whose guidance and support through the work, this accomplishment could not have been possible. His enthusiasm and encouragement made the entire course of this study very much enjoyable. I am also grateful for the committee members Bani Mallick, Faming Liang and Jianxin Zhou. Their critical readings and sharp comments have made the dissertation richer. They have been an invaluable source in tackling my research problems.

My special thanks first go to my mother. Her steady love and encouragement have sustained me in my growing up years. She has always been with me in good times or bad, showing unselfish support and unconditional love throughout the years of my life. Without her love, I doubt I could have accomplished my work successfully. I am also thankful to be blessed with my parents-in-law, who have supported me spiritually and who awoke before dawn each day to pray for me.

I thank my brother, Ho Kyoung, and his family, who have supported me during my study. I owe them a debt of gratitude. I also thank my friends in College Station, Joon Jin Song, Jeesun Jung, and Deukwoo Kwon, who helped and encouraged me in many ways.

Finally, I am indebted to two special people. My wife, You-Young, whose companionship, prayers, sacrifice, and never-ending support throughout the years of studying abroad have been a source of strength and encouragement to complete my study. My lovely daughter, Arline, has brought me great joy and made my life unique, vivid, and colorful. I am grateful for the support from both of them.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

Data arising in a wide range of fields are often obtained in a form of functions. That is, one or more observations are taken on each of a number of individuals in a sample. Advancement of scientific technology requires development of statistical analysis, with the aim of making inferences about population from which functional data are drawn. While the analysis of functional data (FD) and that of multivariate data share many common principles, the infinite-dimensional nature of functional data presents many new challenges that are absent in the traditional multivariate analysis. The book by Ramsay and Silverman (1997) gives a clear account of the basic considerations of functional data analysis (FDA). A software developed for both the Matlab and S-PLUS by Ramsay and Silverman is available from http://www.psych.mcgill.ca/faculty/ramsay/fda.html.

Functional data refer to data which consist of observed functions or curves evaluated at a finite subset of some interval. In a conceptual sense, however it is thought of as being defined continuously. Due to the nature of functional data, modeling such type of data requires to consider function spaces such as Hilbert spaces, and each functional observation is viewed as a realization generated by a random mechanism in the spaces. What distinguishes FDA from other conventional statistics is the atom of data. The numbers are regarded as the atoms in real random variables, and vectors of numbers as the atoms in random vectors. In FDA, however, data come in a form

---

The format and style follow that of *Journal of the American Statistical Association*.

of functions or curves as their atoms. It should be emphasized that the individual datum in FDA is a whole function defined on some interval, rather than focusing on the observed value at a particular point in the interval.

Functional Data Analysis has a wide range of flexibility in the sense that the time points are not required to be equally spaced in subjects and furthermore they can vary from one subject to another. Functional data do not necessarily assume that an observation evaluated at one time point in the interval is independent that of another point within the same functional datum. It can be assumed to be independent from one functional datum to another, but not necessarily to be independent of observed values at distinct time points within the same functional datum. In some cases, functional data are functions of time, but it may not always be true. For example, functional data on a higher dimensional space might be functions of quantities other than time. Ramsey and Silverman (2002) may be consulted for more accounts of case studies in FDA. For thorough mathematical aspects of functional analysis, see Conway (1985), Lebedev et al.(2002) and Rynne et al. (2001).

1.1 Basis Function Approach

Consider the situation where we observe unsupervised sample curves, which is partially observed on the subset of an interval. Let $\{X(t),\ t \in T\}$ be a second order stochastic process defined on T, e.g., $X \in L_2[0,1]$. The stochastic process is a collection $\{X(t),\ t \in T\}$ defined on a common probability space $(\Omega, \mathcal{F}, P)$. Let P_X be the corresponding probability distribution of X. In order to clarify the use of the index set in stochastic processes, one needs to write $X(t)$ as a function $X(\omega, t)$ of two variables ω and t. For fixed $t \in T$, the function $X(\cdot, t)$ is a measurable map from Ω into $\mathbf{R}$. For fixed $\omega \in \Omega$, the function $X(\omega, \cdot)$ becomes a sample path of the

stochastic process. Denoted by $\mu(t)$,

$$\mu(t) := \mathrm{E}X(\omega, t) = \int x(\omega, t) dP_X(x), \tag{1.1}$$

for fixed t. It may be reasonable to assume that the probability distribution P_X be a mixture distribution in which each of component distributions in P_X represents the underlying distributional substructure. The difficulty in implementing the idea is that information of the structure of P_X is rarely available.

An alternative in FDA setting to avoid the difficulty is to consider function spaces where sample paths reside in. With fixed ω, a sample path $X(\omega, t)$ is an equivalent class of functions in the function space L_2. Since functions in the space L_2 can be expressed in terms of basis functions generating the space and furthermore the space is a separable Hilbert space, each function in the space can be written as a countable linear combination of the basis functions. Let $\{\phi_k\}$ be a set of basis functions of L_2, then we see that for each $X(\omega, t)$ with fixed ω, there is a unique $\mathbf{c}' = (c_1, c_2, \ldots) \in l_2$ such that

$$X(t) = \sum_{k=1}^{\infty} c_k \phi_k(t). \tag{1.2}$$

It should be emphasized that the stochastic process is decomposed into two parts c_k and $\phi_k(t)$ and the random mechanism only involves in the coefficients $c_k = c_k(\omega)$.

Once the representation by basis functions is adopted, three types of inquires need to be answered for computational issues.

- How many basis functions are selected to describe the sample paths.
- Which basis functions are appropriate.
- How the coefficients $\mathbf{c}$ are determined based on partially observed functions.

The choice of the number of basis functions clearly involves the decision of smoothness as well as dimension reduction of the process. Ramsey and Silverman (1997) suggest that 20-30 basis functions are in general enough to extract the prominent features. Choosing a basis is a more controversial issue since no basis is universally good. However there are advisable guidelines on specific occasions. For example, if the paths are uniformly smooth with limited features and especially if the curves appear to be periodic, then the Fourier basis seems to be a good choice. On the other hand, splines or wavelets may be a better choice if there are a number of local features which may be relevant for the statistical analysis.

Admitting a bit of abuse of notations, we may write

$$X(t) = \sum_{k=1}^{K} c_k \phi_k(t). \tag{1.3}$$

In reality, $X(t)$ is only observed on a finite set of time interval, and suppose that we have a set of data $x_i(t_{ij}),\ i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, J_i$, where the time points t_{ij}'s can be irregularly spaced. For simplicity, we assume that the time points are the same for all the sample curves, which is denoted by $t_1, \ldots, t_J$.

The least squares approach is the standard method to determine the approximating basis expansion by minimizing the sum of squares, for $i = 1, \ldots, n$,

$$\begin{aligned} & \sum_{j=1}^{J} \Big[x_i(t_j) - \sum_{k=1}^{K} c_{i,k} \phi_k(t_j) \Big]^2 \\ = \ & (\mathbf{x}_i - \Phi \mathbf{c}_i)'(\mathbf{x}_i - \Phi \mathbf{c}_i) \\ = \ & \|\mathbf{x}_i - \Phi \mathbf{c}_i\|^2_{\mathbf{R}^J}, \end{aligned} \tag{1.4}$$

where $\mathbf{x}_i' = (x_i(t_1), \ldots, x_i(t_J))$, $\mathbf{c}_i' = (c_{i,1}, \ldots, c_{i,K})$ and $\Phi = \{\phi_k(t_j)\}_{j,k=1}^{J,K}$. The solution vector to the minimization problem (1.4) is, for $i = 1, \ldots, n$,

$$\mathbf{c}_i = (\Phi'\Phi)^{-1}\Phi'\mathbf{x}_i, \tag{1.5}$$

if Φ has full rank.

The computation in $\mathbf{c}_i$ requires to obtain the inverse matrix, which can be challenging with higher dimension. However expensive computation can be lessened if $\Phi'\Phi$ is a "band matrix" with nonzero elements only close to the diagonal. A special case of band matrices is a diagonal matrix. One such example is when the t_j are equally spaced and the Fourier basis is used then $\Phi'\Phi$ is a diagonal matrix.

1.2 Using the Data to Represent Curves

Suppose that $X(t)$ is a second order stochastic process on $[0,1]$ with zero mean function and a covariance function $v(s,t) = \mathrm{E}X(s)X(t)$. In the previous Section 1.1, we are interested in the estimation of all of $X(t)$ by forcing the data to be adapted into the space spanned by basis functions. As discussed before, the choice of basis functions is a debatable problem, and one alternative approach is to take the data itself as the basis.

To implement this idea, we use a linear combination

$$X(t) = \sum_{j=1}^{J} c_j X(t_j) \tag{1.6}$$

and determine the c_j's by minimizing

$$\begin{aligned} L(\boldsymbol{c}) &= \mathrm{E}\Big[X(t) - \sum_{j=1}^{J} c_j X(t_j)\Big]^2 \\ &= v(t,t) - 2\sum_{j=1}^{J} c_j v(t,t_j) + \sum_{j=1}^{J}\sum_{k=1}^{J} c_j c_k v(t_j,t_k). \end{aligned} \tag{1.7}$$

Differentiating with respect to c_l and equating it to zero, we obtain

$$v(t,t_l) = \sum_{j=1}^{J} c_j v(t_l,t_j) = \Big(v(t_l,t_1),\cdots,v(t_l,t_J)\Big) \begin{bmatrix} c_i \\ \vdots \\ c_J \end{bmatrix}. \tag{1.8}$$

It is easily seen that

$$\begin{bmatrix} \hat{c}_i \\ \vdots \\ \hat{c}_J \end{bmatrix} = \Sigma^{-} \begin{bmatrix} v(t,t_1) \\ \vdots \\ v(t,t_J) \end{bmatrix}, \tag{1.9}$$

where $\Sigma = \{v(t_i,t_j)\}_{i,j=1}^{J}$ and Σ^{-} is a Monroe-Penrose generalized inverse of Σ.

To give clear exposition of $X(t_j)$, we again employ two variables t_j and w. Observe that $X(t_j)$ is a real value evaluated at $t = t_j$ and $X(t_j)$ is viewed as $X(t_j, w)$, and hence $X(t_j)$ is a random variable. It is now clear that each c_j is a function of t, $(1 \leq j \leq J)$.

When the data is used to represent curves, a disadvantage to the approach can be found if the number of time points is relatively larger than the number of curves in a sample ($n << J$). From (1.6), one needs to estimate the functions $c_j,\ (1 \leq j \leq J)$ to get a single approximation of $X(t)$. One of the desirable qualities of estimation is parsimony. Specifically, we need estimation procedures to be as efficient as possible without heavy computation. Comparing with the basis function approach in which we need to estimate K real values for c_k's in (1.3), the method using the data to construct curves can not be an efficient way. In addition to that, it may not be easy to interpret the functions c_j in (1.8).

1.3 Treating Functional Data as Multivariate Vectors

One of the simplest ways to handle functional data is to treat them as multivariate vectors. That is, the space where each datum resides is not a space of functions, but a finite dimensional $\mathbf{R}^J$. One should note that this method does not consider any dependencies of different values over subsequent time-points within the same functional datum, so called *horizontal dependencies*. Employing the method implies that permuting time points arbitrarily, which is equivalent to exchanging the order of the

indexes in a multivariate vector, should not change the result of statistical analysis. Accounting for the inherent nature of the data and using the dependencies along the time-axis should lead to higher quality results.

CHAPTER II

FUNCTIONAL PRINCIPAL COMPONENT ANALYSIS

Principal Component Analysis (PCA) is an effective technique for understanding the structure of data. Analogous to the classical multivariate PCA, the essential goal of functional PCA is to obtain a few orthogonal functions that most efficiently describe the variations in the data.

Let $\{X(t), t \in T\}$ be a zero-mean stochastic process where T is some index set which is taken to be a bounded or unbounded interval here. Assume that the sample paths belong to the usual L_2 space of measurable functions on T with inner product

$$\langle f_1, f_2 \rangle = \int_T f_1(x) f_2(x) dx.$$

Let v be the covariance function of the $\{X(t)\}$, i.e. $v(s,t) = \mathrm{E}X(s)X(t)$. The covariance operator V is defined to be

$$Vf \to \langle v(x,\cdot), f(x) \rangle = \int_T v(x,\cdot) f(x) dx, \ f \in L_2.$$

Since the operator V is a Hilbert-Schmidt operator (Rynn and Youngson, 2001), V admits an eigenvalue decomposition, namely V has a sequence of eigenvalues and eigenfunctions $\rho_i, \ \xi_i, \ i = 1, 2, \ldots$, satisfying

$$V\xi_i = \rho_i \xi_i \text{ and } \langle \xi_i, \xi_j \rangle = \delta_{i,j} \text{ for all } i, j.$$

In practice, we do not know the true function v but rather have a sample $x_i(t), \ 1 \leq i \leq n$, where for each i, $x_i(t)$ is observed on a discrete set of points $T_i = \{t_{i,1}, \ldots, t_{i,J_i}\}$ for some finite J_i. In principle, v can be estimated from the data and the $\rho_i, \ \xi_i$ can then be computed from the estimated covariance operator.

There are a number of ways to do this. Here we adopt the basis function approach in Ramsay and Silverman (1997). First, let $\{\phi_1, \ldots, \phi_K\}$ be the first K basis functions in a basis, where K is picked to be large enough, say, between 20 and 30, so that these functions will be able to described most of features of the data. The basis are selected based on the nature of the data; for example if the data are smooth and periodic then a Fourier basis might be ideal and for data that have a lot of local features then B-splines might work better. Approximate each x_i by

$$\tilde{x}_i(t) = \sum_{k=1}^{K} c_{i,k}\phi_k(t)$$

where the coefficients $c_{i,k}$ are obtained by minimizing the least squares criterion function:

$$\sum_{j=1}^{J_i}\left[x_i(t_{i,j}) - \sum_{k=1}^{K} c_{i,k}\phi_k(t)\right]^2.$$

The centered version of $\tilde{x}_i$ is then

$$\hat{x}_i(t) = \sum_{k=1}^{K} \hat{c}_{i,k}\phi_k(t),$$

where

$$\hat{c}_{i,k} = c_{i,k} - \frac{1}{n}\sum_{i=1}^{n} c_{i,k}.$$

Then the sample covariance functions is

$$\begin{aligned} \hat{v}(s,t) &= \frac{1}{n}\sum_{i=1}^{n} \hat{x}_i(s)\hat{x}_i(t) && (2.1)\\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{l=1}^{K} \hat{c}_{i,k}\hat{c}_{i,l}\phi_k(s)\phi_l(t). && (2.2) \end{aligned}$$

Hence the estimated covariance operator is

$$\hat{V}f = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{l=1}^{K} \hat{c}_{i,k}\hat{c}_{i,l}\langle\phi_k, f\rangle\phi_l, \qquad (2.3)$$

and if $f = \sum_{m=1}^{K} a_m \phi_m$, then

$$\hat{V} f = \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{K} \sum_{l=1}^{K} \sum_{m=1}^{K} \hat{c}_{i,k} \hat{c}_{i,l} a_m \langle \phi_k, \phi_m \rangle \phi_l, \tag{2.4}$$

which can be conveniently expressed as

$$\hat{V} f = \boldsymbol{\phi}' C \boldsymbol{\Phi} \boldsymbol{a}, \tag{2.5}$$

where

$$C = \left[\frac{1}{n} \sum_{i=1}^{n} \hat{c}_{i,k} \hat{c}_{i,l} \right]_{k,l=1}^{K}, \ \boldsymbol{\Phi} = \left[\langle \phi_k, \phi_m \rangle \right]_{k,m=1}^{K}, \ \boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)', \ \boldsymbol{a} = (a_1, \ldots, a_K)'.$$

Hence the eigenvalue problem in the functions space

$$\hat{V} f = \lambda f$$

can be expressed as

$$\boldsymbol{\phi}' C \boldsymbol{\Phi} \boldsymbol{a} = \lambda \boldsymbol{\phi}' \boldsymbol{a}. \tag{2.6}$$

and can be solved as an eigenvalue problem in the finite dimensional space:

$$C \Phi \boldsymbol{a} = \lambda \boldsymbol{a}. \tag{2.7}$$

Thus, the jth principle component eigenvector $\boldsymbol{a}_j$ of $C\boldsymbol{\Phi}$ leads to an estimate $\hat{\xi}_j = \boldsymbol{\phi}' \boldsymbol{a}_j$ of the jth principal component eigenfunction of V.

Following the above procedure, the jth principle component score of $\hat{x}_i$ is defines to be

$$\alpha_{i,j} = \langle \hat{x}_i, \hat{\xi}_j \rangle, \tag{2.8}$$

and we can write

$$\hat{x}_i = \hat{x}_{i,p} + r_{i,p}, \tag{2.9}$$

where

$$\hat{x}_{i,p} = \sum_{j=1}^{p} \alpha_{i,j} \hat{\xi}_j \text{ and } r_{i,p} = \hat{x}_i - \hat{x}_{i,p}. \tag{2.10}$$

Denoting the principal component score vectors by $\boldsymbol{\alpha}_i^p := (\alpha_{i,1}, \ldots, \alpha_{i,p})'$, $1 \leq i \leq n$, a larger p will allow $\hat{x}_{i,p}(t)$ to approximate $\hat{x}_i$ and hence x_i better, but it could also result in over-fitting in the classification.

CHAPTER III

SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM) is a recently developed classification technique by Vapnik (1995). The main idea behind the technique is embodied in the Structural Risk Minimization (SRM) principle in which it aims at minimizing an upper bound on the generalization error of a model via a nested sequence of function classes. The SRM principle is proposed to overcome the problem that while the empirical risk converges to the expected risk by the law of large numbers, this does not necessarily imply that the minimizer of the empirical risk converges to that of the expected risk in the limit of sample sizes. The structure of the function class in the SRM principle finds a decision function having a small training error and the function comes from an element of the structure that has low capacity or VC dimension.

Kernels used in the SVM generalize the concept of linear decision boundaries for classification, producing nonlinear boundaries by building a linear boundary in an enlarged feature space $\mathcal{H}$. Transforming of the data into the larger feature space achieves linear separation easier, and the linearly separating boundary in $\mathcal{H}$ is translated into a nonlinear boundary in the original space. SVM is an extremely powerful general methodology which has a wide range of applications, including pattern recognition (Burges, 1998), gene classification (Brown et al., 1999) and spam filtering (Drucker et al., 1999). For complete details of SVM, see Vapnik (1995, 1998), Burges (1998), Cristianini and Shawe-Taylor (2000) and Gunn (1998).

3.1 Support Vector Machine For Classification

In this chapter we describe SVM for classification to determine a rule from the observed data $(\mathbf{x_i}, y_i),\ 1 \leq i \leq n$, to classify any new observation for which the class label is not observed. The classification problem here is confined to the case where $y_i \in \{-1, +1\}$ for simplicity. For the most part, $\mathbf{x_i}$ is only required to be in a dot product space. However, to conform with the literature at large, we assume that $\mathbf{x_i} \in \mathbf{R}^p$ for some positive integer p.

For an overview we briefly investigate the main ideas here. Suppose that we have a linearly separable data set $\{(\mathbf{x_i}, y_i) \in \mathbf{R}^p \times \{-1, +1\}\}_{i=1}^n$ for which the positive examples $(y_i = 1)$ can be perfectly separated from the negative examples $(y_i = -1)$ by a hyperplane in $\mathbf{R}^p$. Then there exists $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, $\mathbf{w} \in \mathbf{R}^p$ and $b \in \mathbf{R}$, satisfying that for $i = 1, \ldots, n$,

$$f(\mathbf{x_i}) \geq 1 \text{ if } y_i = 1 \text{ and } f(\mathbf{x_i}) \leq -1 \text{ if } y_i = -1.$$

Combining these two conditions,

$$y_i f(\mathbf{x_i}) \geq 1 \text{ for } i = 1, \ldots, n. \tag{3.1}$$

There are many possible linear hyperplanes that satisfy (3.1) but SVM selects the one that maximizes margin (maximizes the distance from the hyperplane to the nearest positive and negative data points). Consider the points for which the equality holds in (3.1), and these points lie on either $\mathbf{w} \cdot \mathbf{x} + b = 1$ or $\mathbf{w} \cdot \mathbf{x} + b = -1$ with normal vector $\mathbf{w}$ and perpendicular distances from the hyperplanes $|1 - b|/\|\mathbf{w}\|$ and $|-1 - b|/\|\mathbf{w}\|$, respectively. Thus, the margin is simply given by $2/\|\mathbf{w}\|$. Note that the hyperplane maximizing $2/\|\mathbf{w}\|$ is obtained by minimizing

$$\frac{1}{2}\|\mathbf{w}\|^2, \tag{3.2}$$

subject to (3.1). Note that changing b will move it in the normal direction provided (3.1) holds, thus the margin remains unchanged but the hyperplane is no longer optimal.

The solution of the primal optimization problem (3.2) subject to the constraint (3.1) can be written in terms of the Lagrange functional

$$L(\mathbf{w}, b, \lambda) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \lambda_i((\mathbf{w} \cdot \mathbf{x}_i + b)y_i - 1), \tag{3.3}$$

where $\lambda_i \geq 0$ are the Lagrange multipliers. The Lagrangian has to be minimized with respect to $\mathbf{w}$ and b, and maximized with respect to λ_i. We now introduce the duality of the primal problem, which is easier to solve. The dual problem is found by differentiating with respect to $\mathbf{w}$ and b and imposing stationarity, then the duality is given by

$$\max_{\lambda} W(\lambda) = \max_{\lambda} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j), \tag{3.4}$$

subject to

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \text{ and } \lambda_i \geq 0,\ i = 1, \ldots, n. \tag{3.5}$$

Denoted λ_i^* by the solution to the dual problem (3.4), we find the solution to the primal problem (3.3) is

$$\mathbf{w}^* = \sum_{i=1}^{n} \lambda_i^* \mathbf{x}_i y_i \tag{3.6}$$

$$b^* = -\frac{1}{2}\mathbf{w}^* \cdot (\mathbf{x}_r + \mathbf{x}_s) \tag{3.7}$$

where $\mathbf{x}_r$ and $\mathbf{x}_s$ are any support vector from each class satisfying,

$$\lambda_r^*,\ \lambda_s^* > 0 \text{ and } y_r = 1,\ y_s = -1. \tag{3.8}$$

Finally, our decision rule is defined to be

$$\psi(\mathbf{x}) = sgn(f(\mathbf{x})). \tag{3.9}$$

The arguments above require that the training data are linearly separable but it may not be realistic. We can relax this condition by introducing an additional cost function or soft margin associated with misclassification error. To enable the optimal separating hyperplane method to be generalized, non-negative variables ν_i need to be incorporated into the problem for linearly separable case. For a given value of C,

$$\min_{\mathbf{w},b,\nu} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \nu_i, \tag{3.10}$$

subject to

$$y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \nu_i,\ i = 1, \ldots, n. \tag{3.11}$$

The Lagrangian for the optimization problem of (3.10) under the constraints of (3.11) is given by

$$L(\mathbf{w}, b, \nu, \lambda, \mu) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\nu_i - \sum_{i=1}^{n}\lambda_i((\mathbf{x}_i\cdot\mathbf{w} + b)y_i - 1) - \sum_{i=1}^{n}\mu_i\nu_i, \tag{3.12}$$

where $\lambda_i \geq 0$ and $\mu_i \geq 0$ are the Lagrange multipliers. The corresponding duality is obtained from the similar argument,

$$\max_{\lambda} W(\lambda) = \max_{\lambda} \sum_{i=1}^{n}\lambda_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j), \tag{3.13}$$

subject to

$$\sum_{i=1}^{n}\lambda_i y_i = 0 \text{ and } 0 \leq \lambda_i \leq C,\ i = 1, \ldots, n. \tag{3.14}$$

The solution to the optimization is identical to the separable case except for a modification of the bound C of the Lagrange multipliers. The objective functional $W(\lambda)$ can be written in a more compact form as follows:

$$\lambda\cdot\mathbf{1}_n - \frac{1}{2}\lambda^t\mathbf{K}\lambda, \tag{3.15}$$

where $\mathbf{1}_n = (1, \ldots, 1)^t$ and $\mathbf{K}$ is an $n \times n$ positive-definite matrix whose entries are $\mathbf{K}_{ij} = y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j$. Note that the dual optimization problem (3.13) is in the form of dot product $\mathbf{x}_i \cdot \mathbf{x}_j$. From the fact the problem is completely described by the inner products of the training data we can extend the concept of inner product to the so-called *kernel method* by mapping the data into high dimensional feature spaces.

3.2 Nonlinear Support Vector Machines

In cases where a linear decision function is not appropriate we map the data $\mathbf{x}$ into a high dimensional feature space $\mathcal{H}$. It can be thought of as a generalization of inner products by choosing a nonlinear mapping $\phi : \mathbf{R}^p \to \mathcal{H}$. Then the SVM constructs an optimal separating hyperplane in this higher dimensional space based on the data through a dot product defined on $\mathcal{H}$, i.e. on functions of the form $\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$. If there were a kernel function k such that $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$, the use of kernels makes it possible to map the data implicitly into a feature space, and we would never need to know what ϕ is used.

Using the kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$, the optimization problem (3.3) becomes

$$\begin{aligned} \max_{\lambda} W(\lambda) &= \max_{\lambda} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \\ &= \max_{\lambda} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j), \end{aligned} \tag{3.16}$$

subject to

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \text{ and } \lambda_i \geq 0, \ i = 1, \ldots, n. \tag{3.17}$$

Solving the equation (3.16) under the constraints (3.17) determines the Lagrange multipliers, and the solution is given by

$$f(\mathbf{x}) = \sum_{n_s} \lambda_i y_i k(\mathbf{s}_i, \mathbf{x}) + b, \tag{3.18}$$

where the $\mathbf{s}_i$ are support vectors and the summation is defined over the set of support vectors. A list of various kernels is given in Table 1.

Table 1: Kernel functions.

Kernel Function	Regularization Network
$(\mathbf{x} \cdot \mathbf{y})$	Linear
$(\mathbf{x} \cdot \mathbf{y})^d$	Polynomial of degree d
$\exp(-\gamma\|\mathbf{x} - \mathbf{y}\|^2)$	Gaussian Radial Basis Function
$\tanh(\mathbf{x} \cdot \mathbf{y} - \theta)$	Multi Layer Perceptron
$B_{2n+1}(x - y)$	B-spline
$(\|\mathbf{x} - \mathbf{y}\|^2 + c^2)^{1/2}$	Multiquadric
$(\|\mathbf{x} - \mathbf{y}\|^2 + c^2)^{-1/2}$	Inverse Multiquadric
$\frac{\sin(d + 1/2)(x - y)}{\sin((x - y)/2)}$	Fourier

CHAPTER IV

CLASSIFICATION

4.1 Introduction

Classification of multivariate data has long been an important problem in statistics. For example, linear discriminant analysis goes as far back as Fisher (1936). In this chapter, we consider a classification problem in FDA. Consider the situation where we observe a sample from each of two different populations of curves, where the identity of each observed curve is known. The goal is to use the sample information to construct a classifier to classify any future curve for which the identity is unknown. For example we might be interested in classifying normal brain waves from those that belong to jet pilots under large g-force and about to pass out; the classifier can be used in any future mission in judging the state of alertness of the pilot, or we might be interested in classifying curves that describe harmless seismic activities from those that describe activities which will lead to major earthquakes. In Section 4.5 we will consider a data set in which X records the amount of eggs per day laid by fruit flies over a time period and we wish to classify if the flies are long or short-lived. Mathematically, we consider data in which each observation is pair of values (X, Y) where X is a curve which is wholly or partially observed and Y is its class label which assumes a finite number of values. The objective is to obtain a rule from a sample to classify a new observation X by estimating the corresponding value of Y.

FDA is of growing interest in the scientific literature. The books by Ramsay and Silverman (1997, 2002) are an excellent source for methodological aspects of FDA, and in particular chapter 6 addresses the principal components approach. Focusing on classification of functional data, Hall et al. (2001) proposes a nonparametric pro-

cedure for signal discrimination in which dimension reduction is obtained using the Karhunen-Loève expansion of covariance function, and then a new observation is assigned to the signal type with the highest posterior probability calculated from use of kernel methods. Müller and Stadtmüller (unpublished manuscript) study FD classification problem based on a parametric approach. They also use the Karhunen-Loève expansion with the aim to reduce the dimension, and then apply the machinery of the generalized linear model with logit link function. Alter et al. (2000) adopt the Generalized Singular Value Decomposition to classify dynamic genes for genome-scale expression arrays data with repeated measurements. James and Hastie (2001) employ the functional clustering model, producing low-dimensional representations of sample curves via parameterization of cluster means. The technique they use is particularly useful when curve data are observed at a sparse set of time points.

We only consider the situation where the curves are dense. When the curves are only irregularly sampled, the challenges will be different (James and Hastie, 2001). The classification approach to be addressed in this chapter is composed of two procedures: Functional Principal Component Analysis (functional PCA) and Support Vector Machine (SVM). Dimension reduction of a function space is achieved by using functional PCA, which extracts the modes of variation of curves, and then the SVM is applied to the principal component scores from functional PCA to classify functional data. The SVM based on the idea of Vapnik's theory (Vapnik, 1995) is related to regularization theory, which induces a general decision function for classification of multidimensional space. For details of regularization theory, see chapter 4 in Schölkopf and Smola (2002). The SVM has seen increasing attention from the statistics community. For an overview, see Vapnik (1995), Burges (1998) and Cristianini and Shawe-Taylor (2000). Refer to Smola et al. (1998) for the connection between regularization networks and SVM.

4.2 Application of Functional Data to Support Vector Machine

We have described the functional PCA for the purpose of feature extraction of data and SVM for classification of vector-valued input data. Assume that $X(t)$ is a second order stochastic process such that square integral is finite and Y is a real valued random variable. For a binary case, we would have $Y \in \{-1, 1\}$. Without loss of generality, we can always assume that $\mathrm{E}X(t) = 0$.

Remember that a stochastic process is a collection $\{X(t),\ t \in T\}$ defined on a common probability space $(\Omega, \mathcal{F}, P)$ and P_X is the corresponding distribution function. In order to clarify the use of the index sets in stochastic processes, we again write $X(t)$ as a function of two variables $X(\omega, t)$. Let us assume that $X(t)$ is centered, i.e., for fixed t,

$$\mathrm{E}X(t) = \mathrm{E}X(\omega, t) = \int x(\omega, t) dP_X(x) = 0. \tag{4.1}$$

Suppose that the process is written as

$$X(t) = X(\omega, t) = \sum_{j=1}^{\infty} \alpha_j \xi_j(t) = \sum_{j=1}^{\infty} \alpha_j(\omega) \xi_j(t), \tag{4.2}$$

where $\xi_j(t)$ form orthonormal basis of the function space L_2 and random variables $\alpha_j = \alpha_j(\omega)$ are the coefficients of the projection to eigenfunctions $\xi_j(t)$. Note that for fixed j,

$$\begin{aligned} \langle \xi_j(\cdot), X(\cdot) \rangle &= \int_T \xi_j(t) X(\omega, t) dt \\ &= \int_T \xi_j(t) \sum_{k=1}^{\infty} \alpha_k(\omega) \xi_k(t) dt \\ &= \alpha_j(\omega) \end{aligned}$$

Thus,

$$\mathrm{E}[\alpha_j(\omega)] = \int \int_T \xi_j(t) x(t) dt dP_X(x) = \int_T \xi_j(t) \int x(\omega, t) dP_X(x) dt = 0,$$

by the assumption of the centered $X(t)$. Now consider

$$\begin{aligned}
\mathrm{Var}(\alpha_i(\omega)) &= \int \Big(\int_T \xi_i(t)x(t)dt \Big) \Big(\int_T \xi_i(s)x(s)ds \Big) dP_X(x) \\
&= \int_T \int_T \xi_i(s) \Big(\int x(s)x(t)dP_X(x) \Big) \xi_i(t)dsdt \\
&= \int_T \int_T \xi_i(s)v(s,t)\xi_i(t)dsdt \\
&= \rho_i^2.
\end{aligned}$$

It can be seen that

$$\begin{aligned}
\int_T \mathrm{E}X^2(t)dt &= \int_T \int x^2(t)dP_X(x)dt \\
&= \int_T \int \Big(\sum_{i=1}^{\infty} \alpha_i(\omega)\xi_i(t) \Big) \Big(\sum_{j=1}^{\infty} \alpha_j(\omega)\xi_j(t) \Big) dP_X(x)dt \\
&= \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \int \alpha_i(\omega)\alpha_j(\omega) \int_T \xi_i(t)\xi_j(t)dtdP_X(x) \\
&= \sum_{i=1}^{\infty} \int \alpha_i^2(\omega)dP_X(x) \\
&= \sum_{i=1}^{\infty} \mathrm{Var}(\alpha_i(\omega)) \\
&= \sum_{i=1}^{\infty} \rho_i^2 < \infty.
\end{aligned}$$

Since the predictor variable forms a curve, it is necessary to reduce its dimensionality by using functional PCA on $X(t)$. With a choice of $p = p_n$ increasing as $n \to \infty$ the predictor $X(t)$ can be split into two components for which the first component comprises the first p terms in the expansion of $X(t)$ in (4.2) and the second one keeps the remaining of the expansion. A subjective decision for the choice of p can be made from a scree plot which shows percentages of variation of the predictor $X(t)$. For

fixed t, let

$$A_p(t) = A_p(\omega, t) = \sum_{j=1}^{p} \alpha_j(\omega)\xi_j(t), \tag{4.3}$$

$$B_p(t) = B_p(\omega, t) = \sum_{j=p+1}^{\infty} \alpha_j(\omega)\xi_j(t). \tag{4.4}$$

Then we find that

$$\begin{aligned} \mathrm{E}(X(t) - \mathrm{E}(X(t)|A_p(t)))^2 &= \mathrm{E}(B_p(t) - \mathrm{E}(B_p(t)|A_p(t)))^2 \\ &= \mathrm{E}B_p^2(t) - 2\mathrm{E}(\mathrm{E}^2(B_p(t)|A_p(t))) + \mathrm{E}(\mathrm{E}^2(B_p(t)|A_p(t))) \\ &= \mathrm{E}B_p^2(t) - \mathrm{E}(\mathrm{E}^2(B_p(t)|A_p(t))). \end{aligned}$$

Thus the approximation error of $X(t)$ truncated at the pth term is bounded above. If $p = p_n$ goes to infinity as $n \to \infty$, the approximation error tends to be zero. However, the larger value p might add a lot of noise to the approximation, which leads to over-fitting in classification procedures.

We note that for $X(t)$, the α_j's are uniquely determined with respect to the set of $\xi_i(t)$'s. For a given value of p, the p-truncated process, denoted by $X_p(t)$ can be expressed as

$$X_p(t) = \sum_{j=1}^{p} \alpha_j \xi_j(t). \tag{4.5}$$

Our aim is classification of functional data, each of which is expressed in terms of a p dimensional vector $\boldsymbol{\alpha}^p = \boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_p)'$, called principal component scores. Considering the $\boldsymbol{\alpha}$'s to be input examples in SVM instead of using the functional data, we are able to assign the functional data to their class labels through principal component scores $\boldsymbol{\alpha}$ obtained from functional PCA of $X(t)$. For a linearly separable case with binary label ($Y \in \{-1, 1\}$), SVM allows us to reconstruction of a linear function which generalizes relation of between $\boldsymbol{\alpha}$ and its class label. The various

methods in SVM literature introduced in the previous section such as non-linear cases and kernel methods are also applied to classify functional data. All the simulations and real data analysis in the dissertation are done via SVM^{light} by Joachims (1998)

4.3 Performance Measures

Natural concerns arising in any statistical classification procedures are how well a classification rule performs given the training examples available and how one estimates the generalization error of the rule. In this section we discuss the training error as an estimator of the generalization error, and other performance measures such as the recall and the precision will be introduced in the line of cross-validation.

Suppose the training examples are generated from a unknown distribution $P(\boldsymbol{\alpha}, y)$, where $\boldsymbol{\alpha}$ has been defined in section 4.2. The generalization error of a classification rule ψ based on data $D_n = \{(\boldsymbol{\alpha}_1, y_1), \ldots, (\boldsymbol{\alpha}_n, y_n)\}$ is defined to be

$$Err(\psi) = P[\psi(\boldsymbol{\alpha}) \neq y | D_n] = \int L(\psi(\boldsymbol{\alpha}), y) dP(\boldsymbol{\alpha}, y), \tag{4.6}$$

where L is the 0-1 loss function. A natural estimate of the error rate is the training error, which is defined to be

$$\hat{Err}(\psi) = \frac{1}{n} \sum_{i=1}^{n} L(\psi(\boldsymbol{\alpha}_i), y_i). \tag{4.7}$$

When we seek for a ψ such that it minimizes the training error $\hat{Err}(\psi)$, noticing that it is measured on the same data set implies that it is expected to have an underestimate of the generalization error $Err(\psi)$. A common alternative for estimating the generalization error rate is cross-validation or leave-one-out (loo) estimate. From the training example $D_n = \{(\boldsymbol{\alpha}_1, y_1), \ldots, (\boldsymbol{\alpha}_n, y_n)\}$ the first example $(\boldsymbol{\alpha}_1, y_1)$ is removed. The resulting sample $D_n^{(-1)}$ is used for training, leading to a classification rule $\psi^{(-1)}$. This classification rule is tested on the held-out example $(\boldsymbol{\alpha}_1, y_1)$. This process is

repeated until all training examples are completed. The number of misclassification divided by n is the loo estimate of the generalization error.

$$\hat{Err}_{\text{loo}}(\psi) = err(\psi) = \frac{1}{n}\sum_{i=1}^{n} L(\psi^{(-i)}(\boldsymbol{\alpha}_i), y_i). \tag{4.8}$$

Various measures of performance other than the training error based on loo estimator have been developed by Joachims (2000). In the chapter, the recall and the precision of a decision rule ψ are defined.

The recall $Rec(\psi)$ of a decision rule ψ is defined to be the probability that an example $\boldsymbol{\alpha}$ with label $y = 1$ is classified correctly i.e. $\psi(\boldsymbol{\alpha}) = 1$:

$$\begin{aligned} Rec((\psi)) &= P(\psi(\boldsymbol{\alpha}) = 1 | y = 1) \\ &= \frac{P(\psi(\boldsymbol{\alpha}) = 1, y = 1)}{P(\psi(\boldsymbol{\alpha}) = 1, y = 1) + P(\psi(\boldsymbol{\alpha}) = -1, y = 1)}. \end{aligned} \tag{4.9}$$

Similarly, the precision $Pre(\psi)$ of a decision rule ψ is defined to be the probability that an example $\boldsymbol{\alpha}$ classified as $\psi(\boldsymbol{\alpha}) = 1$ is indeed with the same label, i.e., $y = 1$:

$$\begin{aligned} Pre((\psi)) &= P(y = 1 | \psi(\boldsymbol{\alpha}) = 1) \\ &= \frac{P(\psi(\boldsymbol{\alpha}) = 1, y = 1)}{P(\psi(\boldsymbol{\alpha}) = 1, y = 1) + P(\psi(\boldsymbol{\alpha}) = 1, y = -1)}. \end{aligned} \tag{4.10}$$

For technical details of the estimators of the three measures, see Joachims (1998, 2000) .

4.4 Simulation Studies

In this section we compare the performances of SVM to other traditional classification procedures through numerical examples. We generate two sets of functional data: non-Gaussian and Gaussian stochastic processes. For the non-Gaussian stochastic process we set two true curves with each of 500 simulated sample curves observed on 25 time points in $[0, 1]$. The two true functions are taken such as

$$\begin{aligned} \Pi_1 &= \sqrt{2}\Big(0.9\sin(\pi t) + 1.1\sin(2\pi t) + 0.8\cos(3\pi t) + 0.80\cos(4\pi t)\Big) \\ \Pi_2 &= \sqrt{2}\Big(1.0\sin(\pi t) + 1.5\sin(2\pi t) + 0.7\cos(3\pi t) + 0.75\cos(4\pi t)\Big). \end{aligned}$$

Each sinusoidal function is multiplied by a random number from uniform distributions to generate a total of 1000 sample curves. See Figure 1. Note that the shapes and patterns of all the curves are similar, indicating that it does not seem possible to detect one class from the other. In order to apply the functioanl PCA and then SVM for classification, we need to estimate the eigenfunctions from the estimated sample covariance function. See Figure 2. Solving the eigenequation from Chapter II leads to the principal component functions. Figure 3 shows the first four PC functions from the non-Gaussian processes. All PC functions are clearly periodic and sinusoidal. The first PC function finds that at time point 0.3 the processes have the greatest variability. The second PC function represents the contrast between the first and the second half. The third PC function measures uniformity over the time interval and the fourth PC function concerns sinusoidal decreasing variability. They are orthonormal to each other from the definition of PC functions.

The functional data of size 1000 are projected onto the space spanned by the set of PC functions in order to get the coefficients $\boldsymbol{\alpha}$. Various kernel functions are chosen to compare their performances via leave-one-out cross validation error rates. For comparison with traditional statistical classification procedures using the PC scores $\boldsymbol{\alpha}$, two alternatives are included in the simulation study: Fisher linear discriminant analysis (Fisher LDA) as parameteric approach and kernel discriminant analysis as nonparameteric approach. To illustrate the parametric method, suppose that each class has a multivariate normal distribution. The Fisher LDA develops a discriminant function or classification criterion using Bayes decision rule. The classification criterion is based on the pooled covariance matrix of $\boldsymbol{\alpha}$'s. With the identical prior probability of all classes, each observation is classified into a class in which it has the largest posterior probability of the observation. While the parametric approaches are to assume particular distrubutional forms such as multivariate normal, nonparametric

discriminant methods are based on nonparametric estimates of class-specific probability densities. The kernel method can be used to estimate a nonparametric density in each class and to produce a classification criterion. The kernel method uses uniform, normal, Epanechnikov, biweight, or triweight kernels in the density estimation with the bandwidth r. Large values of r lead to very smooth density estimates and amall values of r lead to rough estimates. Once the densities are estimated, the posterior probabilities of class membership at each observation are evaluated. An observation is classified into a class in which its posterior probability has the largest value.

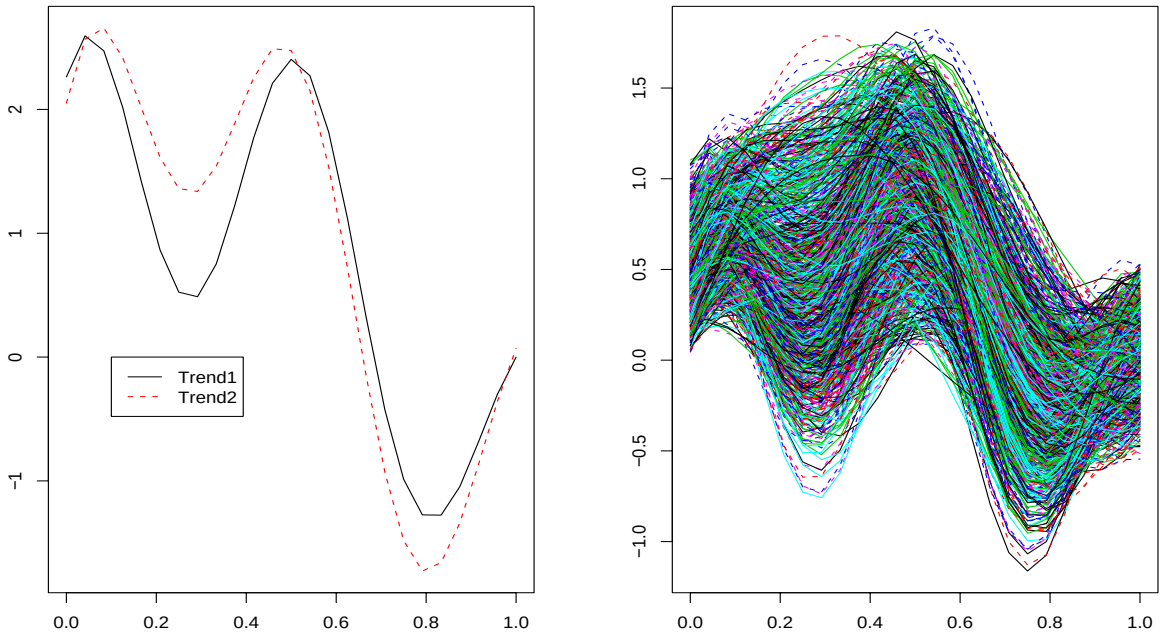

Figure 1: Non-Gaussian stochastic processes.

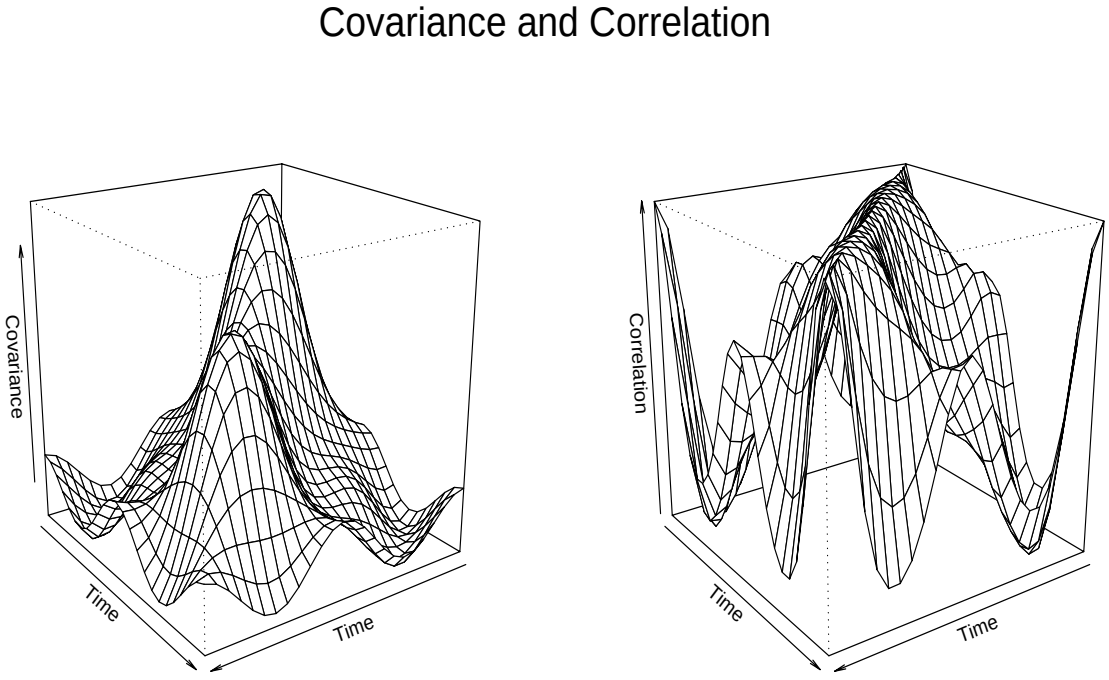

Figure 2: Sample covariance and correlation functions for non-Gaussian stochastic processes.

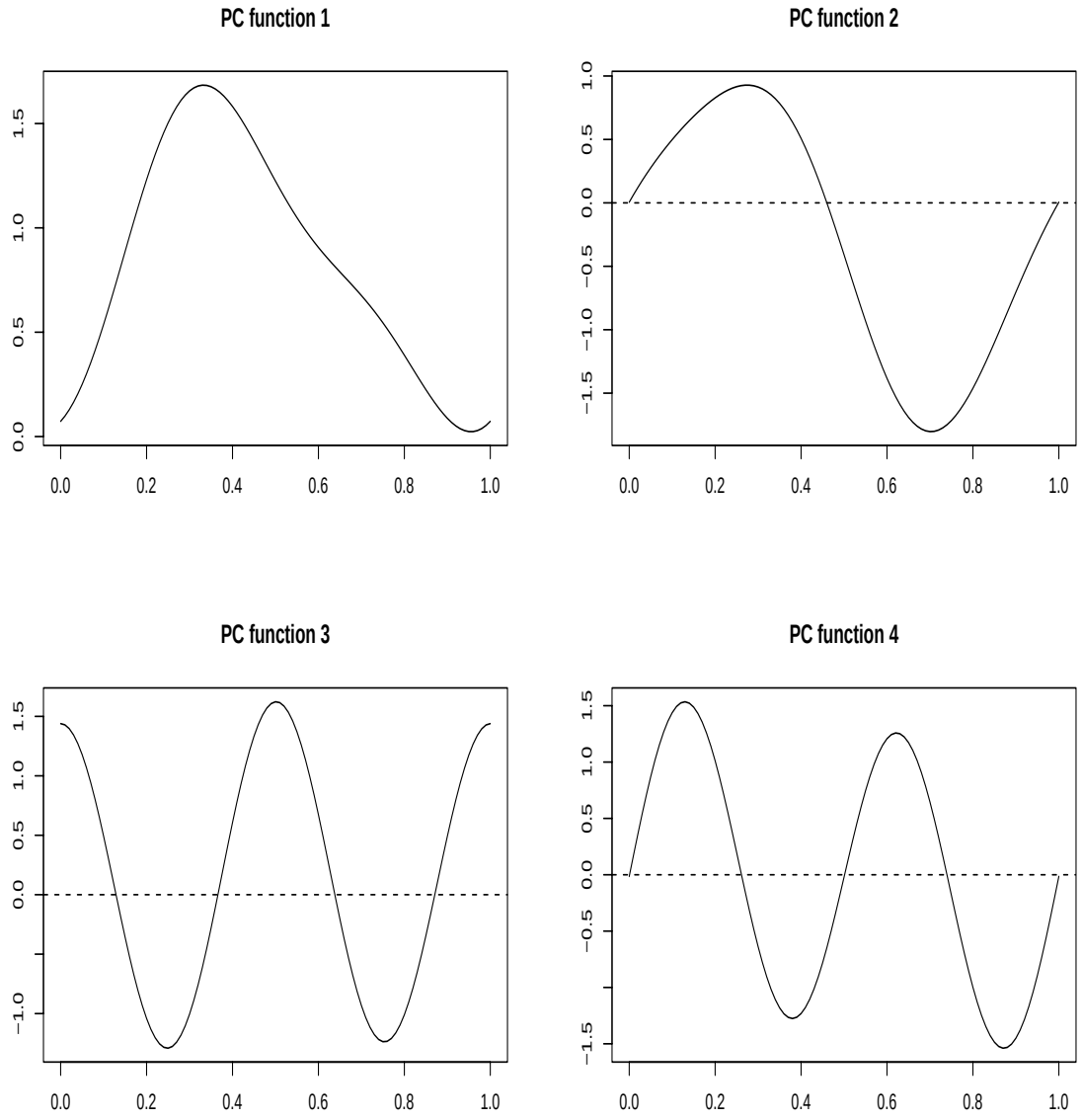

Figure 3: The first four principal component functions of non-Gaussian stochastic processes.

Table 2 summarizes the performances of the SVM, the Fisher LDA and the nonparametric approach using Epanechnikov kernel with the bandwidth r being 0.6 with repect to leave-one-out cross validation error rates with 40 runs. The SVM outperforms the other methods. The SVM using the quadratic polynomial kernel with $p = 4$ yields better result than other SVM kernel functions. The table shows that only 8 out of 1000 curves have been missclassified through the SVM classification procedures.

For the sumulation study of stationary Gaussian processes, 40 sets of stationary Gaussian stochastic processes of size 100 are generated for each class over 30 time points. In the experiment of the Gaussian processes, covariance function is of the form

$$v(s,t) = \sigma^2 r(s,t), \tag{4.11}$$

where σ^2 is the constant variance, producing the processes to homoschedastic and the correlation function is of the form

$$r(s,t) = \exp(-\theta u^{\delta}) \tag{4.12}$$

for $\delta = 2,\ \theta = 1$, and $\sigma^2 = 1$. See Trosset (1999).

Table 2: Averages of leave-one-out cross validation error rates for non-Gaussian functional data.

	$p = 2$	$p = 3$	$p = 4$
LDA	0.0186	0.0147	0.0490
NonPara ($h = 0.6$)	0.0148	0.0209	0.1989
SVM Linear	0.0159	0.0097	0.0095
SVM Polynomial 2	0.0140	0.0099	0.0084
SVM Polynomial 3	0.0140	0.0095	0.0091
SVM RBF ($\gamma = 0.3$)	0.0141	0.0093	0.0091

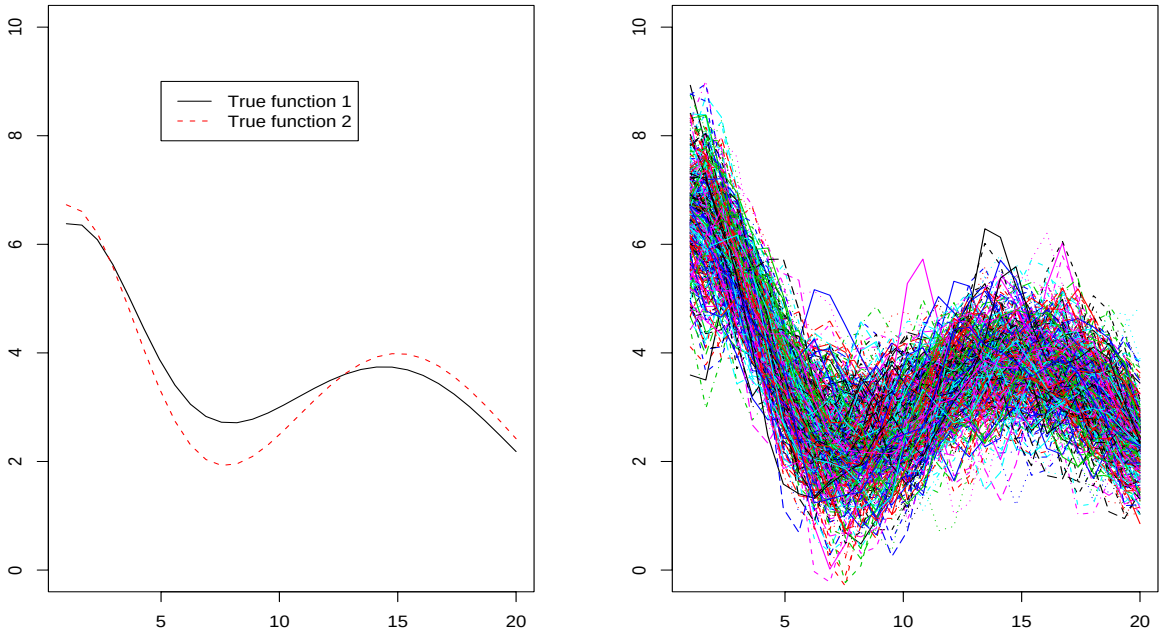

Figure 4: Gaussian stochastic processes.

A run out of 40 is shown in Figure 4. The corresponding covariance and correlation functions are in Figure 5. The first four PC functions in Figure 6 show how the covariance function is decomposed into the orthonormal functions. The first PC function captures the overall mean function of the sample processes, noting that PC functions are constant up to their sign. The second and third PC functions explain variability of the processes in the first and third quarter on the time interval. The forth PC function measures uniformity over the interval. Table 3 provides summary of the performances and the results show that the cross validation error rates depend on the choices of p. The Fisher LDA yields better performance when $p = 3$; nonparametric classification with normal kernel and SVM work better when $p = 4$ and $p = 2$, respectively. It should also be noted that larger value of p does not gurantee better performances. All nine classification procedures including the Fisher LDA and nonparametric method result in the smallest error rates when p is small. In general, using superfluous variables in classification analysis adds noise to the analysis, and indeed, it is not always true that more variables are better in classification analysis.

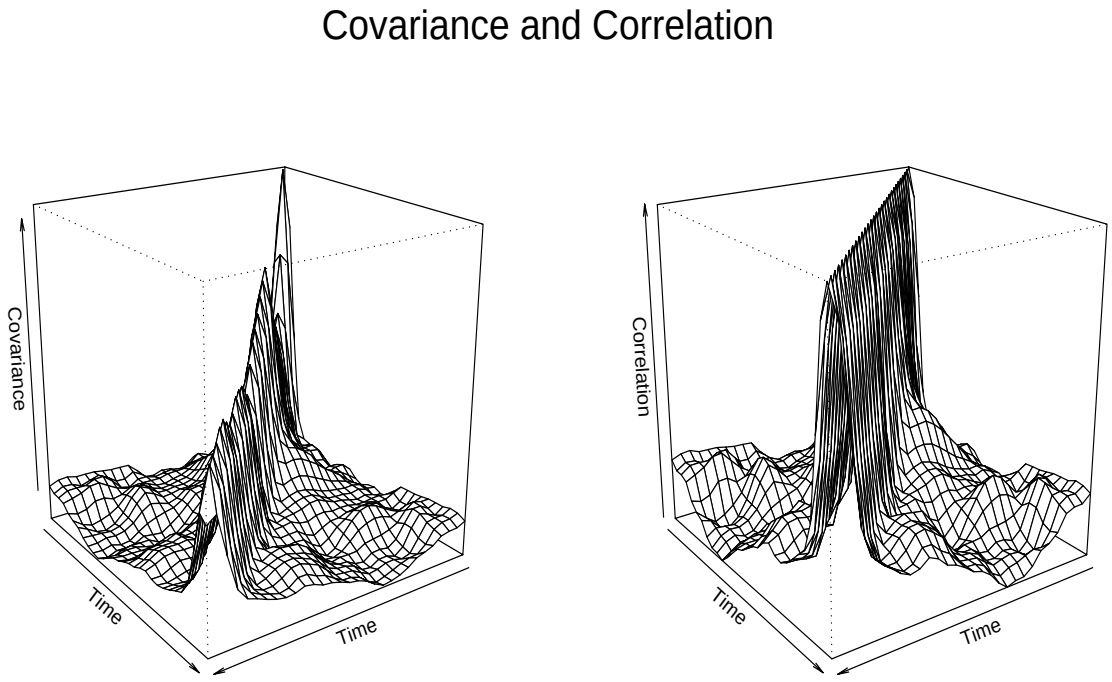

Figure 5: Sample covariance and correlation functions for Gaussian stochastic processes.

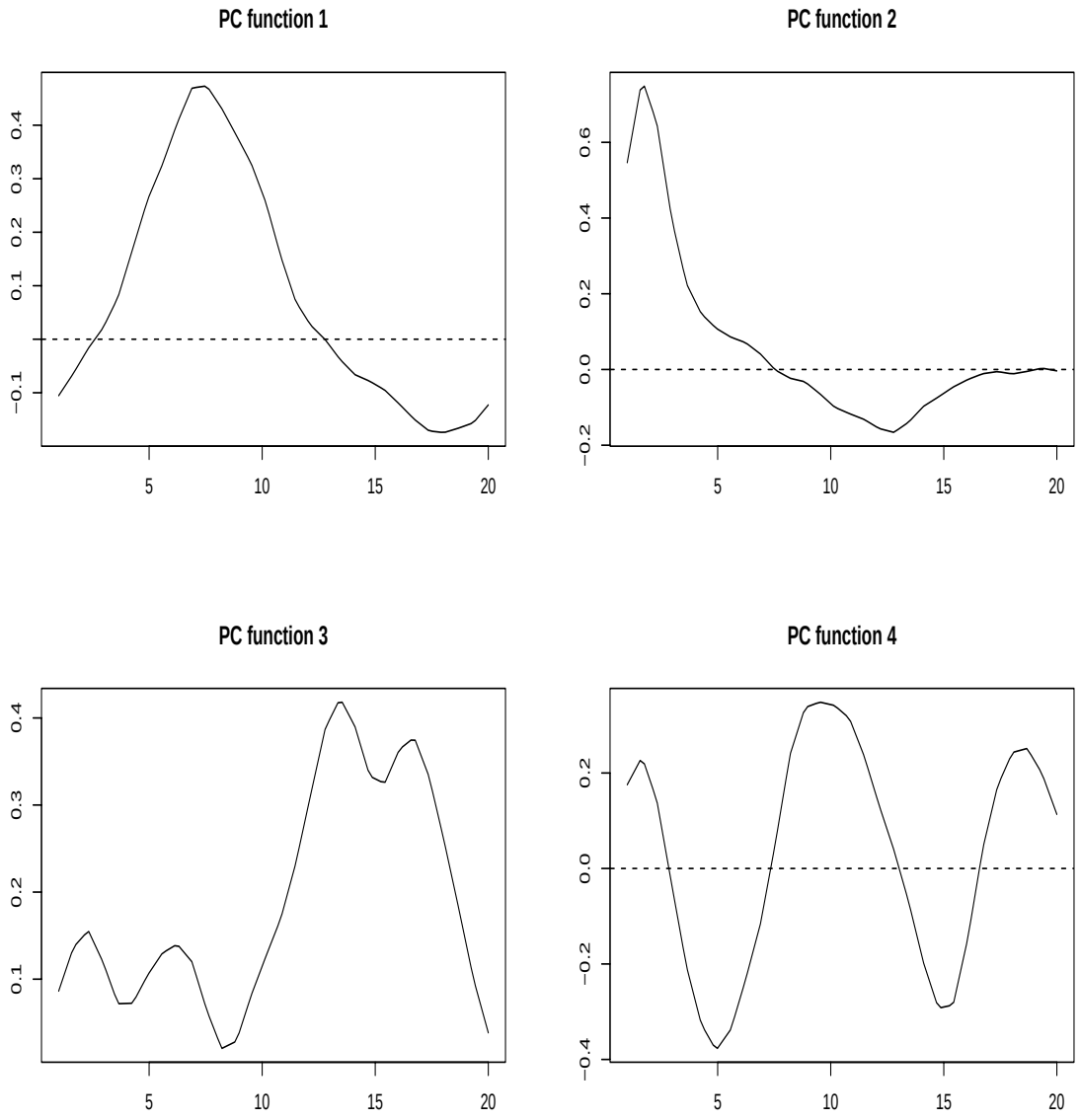

Figure 6: The first four principal component functions of Gaussian stochastic processes.

Table 3: Averages of leave-one-out cross validation error rates for Gaussian functional data.

	$p = 2$	$p = 3$	$p = 4$	$p = 5$	$p = 6$	$p = 7$	$p = 8$	$p = 9$	$p = 10$
Fisher LDA	0.1588	0.1570	0.1585	0.1583	0.1610	0.1643	0.1688	0.1648	0.1695
NonPara (h=2)	0.1570	0.1573	0.1553	0.1548	0.1578	0.1598	0.1618	0.1600	0.1605
NonPara (h=5)	0.1563	0.1583	0.1535	0.1565	0.1558	0.1578	0.1608	0.1593	0.1578
NonPara (h=10)	0.1565	0.1588	0.1540	0.1565	0.1545	0.1563	0.1605	0.1588	0.1593
NonPara (h=15)	0.1568	0.1588	0.1540	0.1565	0.1543	0.1555	0.1578	0.1583	0.1583
Linear	0.1603	0.1618	0.1663	0.1650	0.1720	0.1695	0.1730	0.1708	0.1723
Polynomial 2	0.1678	0.1705	0.1720	0.1815	0.1825	0.1913	0.1883	0.1925	0.1968
Polynomial 3	0.1770	0.1720	0.1760	0.1763	0.1783	0.1788	0.1873	0.1840	0.1805
RBF γ=0.3	0.1698	0.1673	0.1713	0.1695	0.1750	0.1735	0.1810	0.1818	0.1805

4.5 Example: Medfly Fecundity Data

Clear understanding the relationship between reproduction and longevity is a long research interest in ecology and evolution. One suspects that an increment in reproduction might cause a decrement in longevity on organisms. The concept of a "cost of reproduction" explains that a high degree of reproduction prevents an organism from being prolonged lifespan, and reduces its ability to survive due to the fact that resources are dissipated.

Mediterranean fruit flies (*Ceratits capitata*) or medflies in short have been studied by many researchers (Carey et al., 1998, and Müller et al., 2001). The experiment conducted by Carey at al. consists of one thousand of medflies as experimental units and resulting data is collected by counting the daily eggs laid by each individual fly over a certain period of time (30 days) as well as its lifespan. Out of one thousand 534 flies are selected, which lived past 34 days. A fly was assigned to the value of $Y = 1$ indicating as long-lived if the remaining lifetime past 30 days was 14 days or longer. Otherwise, $Y = -1$ was assigned to a short-lived fly. Of the 534 medflies, 256 were classified as long-lived and 278 were classified as short-lived. Applying the basis expansion techniques to the raw data of daily egg counts, the data can then be represented by $(X_i(t), Y_i)$ for $i = 1, \ldots, 534$, where $X_i(t)$ are the stochastic processes of reproductory trajectories on $t \in [1, 30]$ and $Y_i \in \{-1, 1\}$ are class labels.

Various basis functions such as the polygonal, the B-spline and the Fourier basis can be applied to the raw data to get the predictor representations. We plot the reproductory trajectories with polygonal basis for all medflies without distinction between short-lived and long-lived (Figure 7) and the corresponding covariance and correlation functions (Figure 8). We apply the functional PCA to the medflies data expanded by the polygonal, the B-spline and the Fourier basis, respectively to get

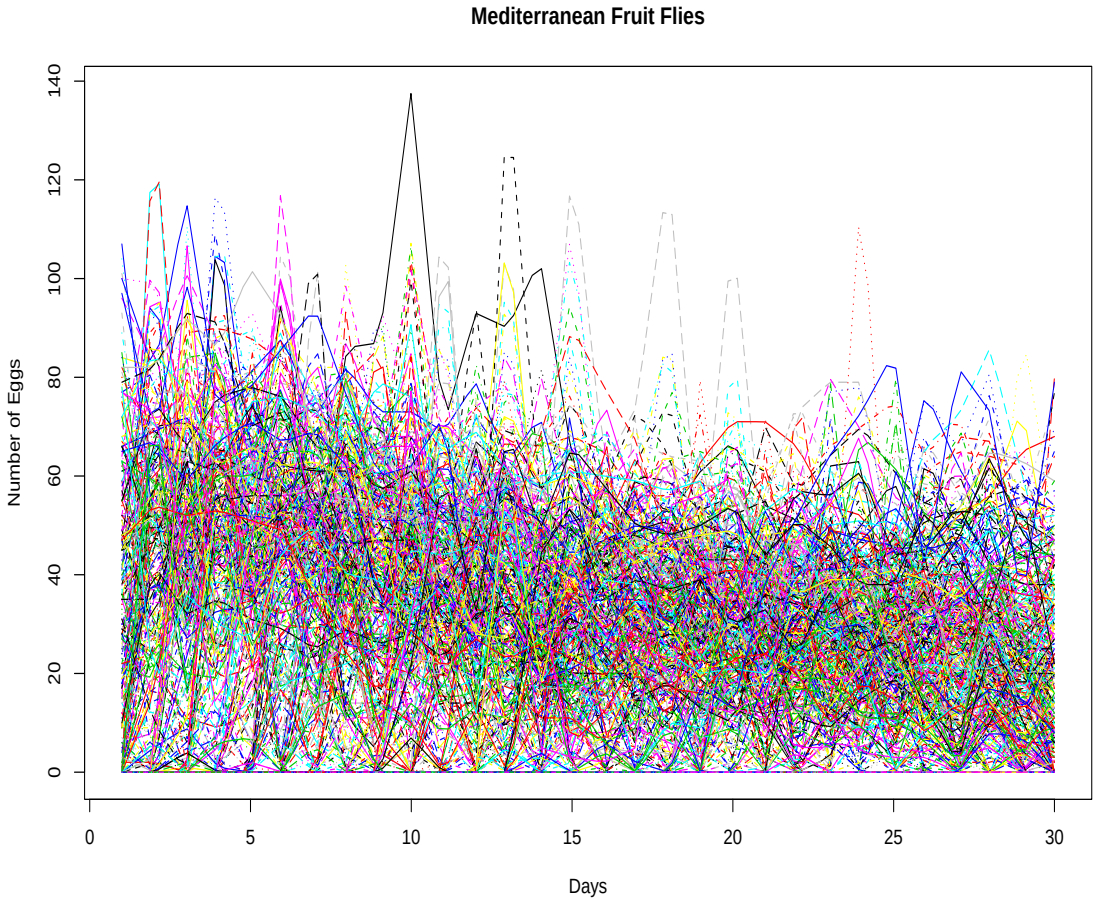

Figure 7: Plot of medflies data expanded by polygonal basis.

the principal component functions $\xi(t)$'s and the principal component scores $\boldsymbol{\alpha}$'s for each $X_j(t)$, $j = 1, \ldots, 534$. The first four eigenfunctions are selected and shown in Figure 9. From these plots we find that the first principal functions are overall positive and indicate that the number of eggs laid by individual medflies has the greatest variability at the fifth day. The second PC functions represent the contrast the number of eggs between the first 10 days and the rest days and it is a measure of a day-shift effect of the number of eggs over the two exclusive sets of days. Note that the second PC functions for the polygonal and the B-spline basis have positive weights between about 1-10 days. However the function for the Fourier basis shows

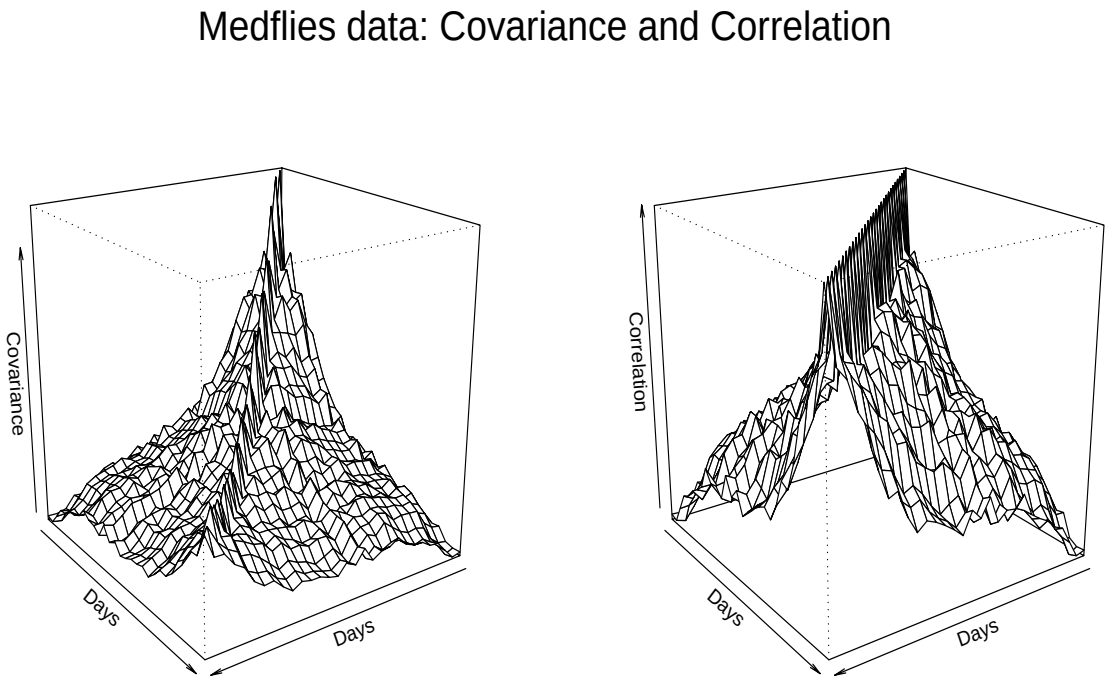

Figure 8: Sample covariance and correlation functions of medflies data expanded by polygonal basis.

negative weights between the same days. The third PC functions measures uniformity of the number of eggs over the 30 days. The forth PC functions compare between the number of eggs in the first and third quarters with the second and forth quarters over the 30 days. For choices of p to determine the dimension in SVM, we find that there are knees at $p = 6$ for the polygonal basis, $p = 7$ for the B-spline basis, and $p = 6$ for the Fourier basis in the scree plots (Figure 10). Consequently, we note that six to seven principal components are enough to explain the variation in the medflies data.

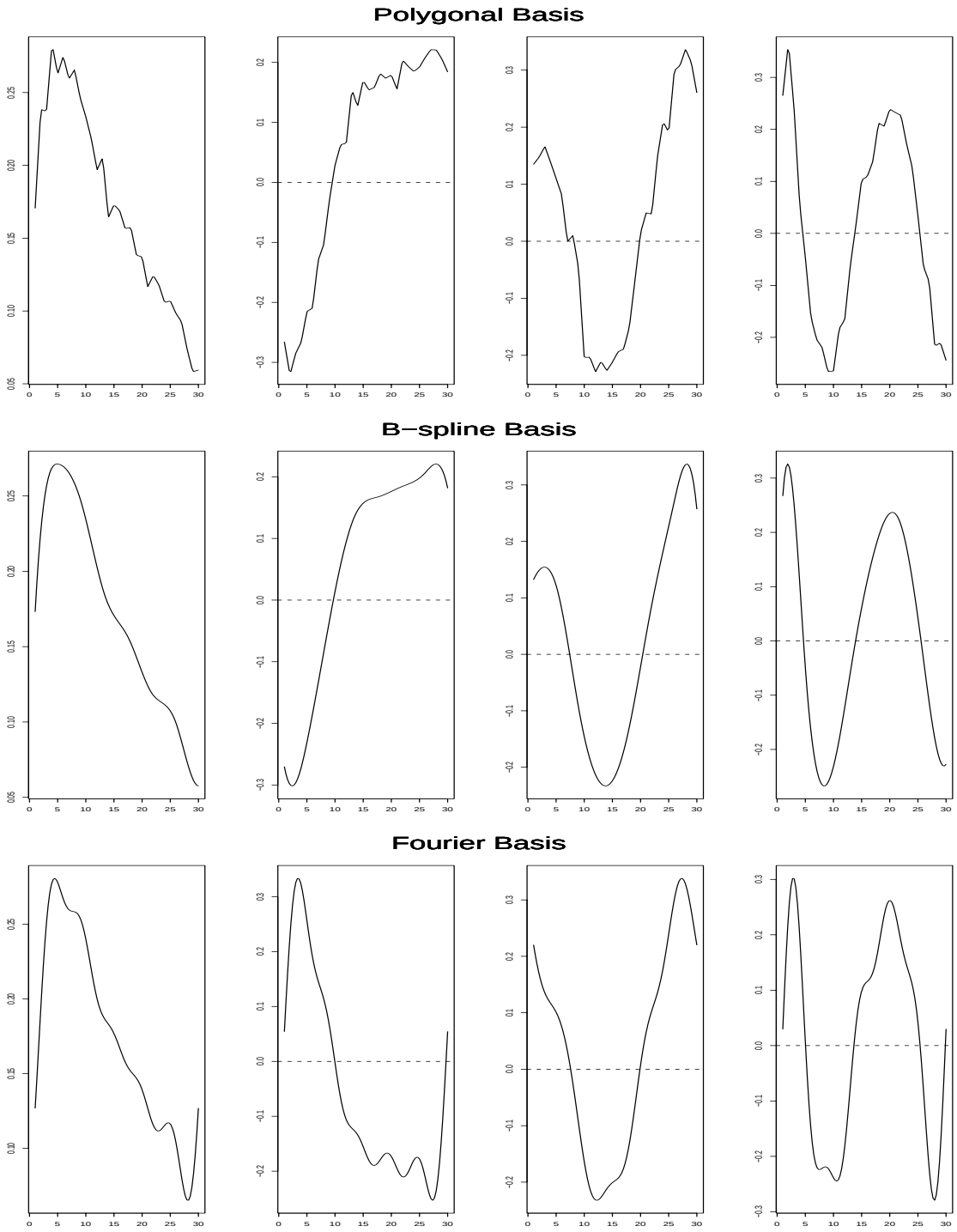

Figure 9: The first four estimated principal component functions of the medflies data expanded by polygonal basis (the first row), B-spline basis (the second row), and Fourier basis (the third row).

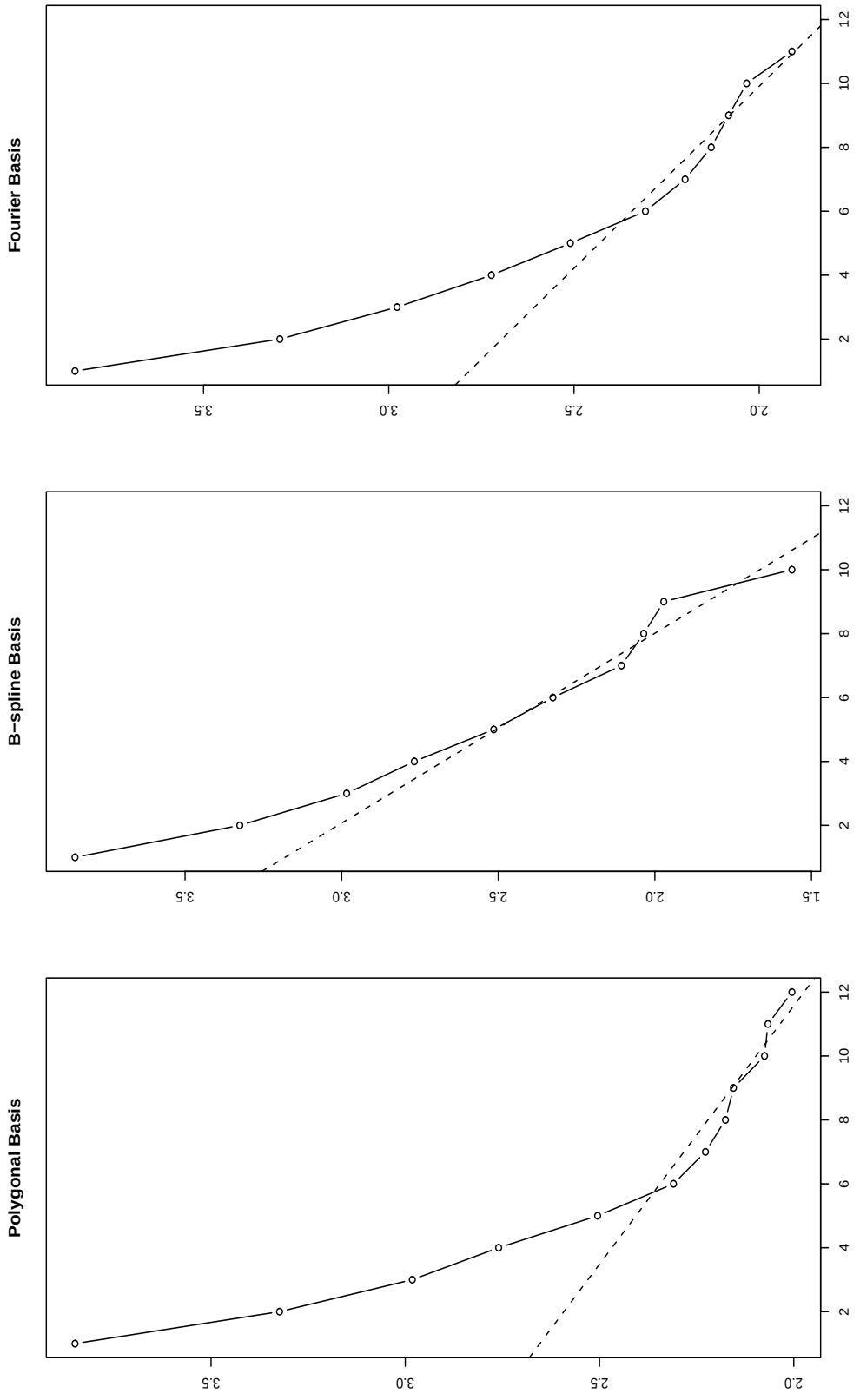

Figure 10: Scree plots to detect the number p of eigenfunctions.

A properly chosen value of p determines the dimension of the input space in the SVM. With the set of training examples composed of the principal component scores and their class labels, the SVM selects a classifier that separates the examples, maximizing the margin. The classifier generalizes our decision rule for classification and makes it possible to predict the class membership when a new input example is given. For the estimates of the generalization error and measures of efficiency such as the training error, the recall and the precision are obtained under three different types of kernel functions used in the SVM (Tables 4-6). We find that the SVM performs better under the three models when $p = 6$ for the polygonal basis, $p = 10$ for the B-spline basis, and $p = 6$ for the Fourier basis. It is consistent with the results from the scree plots that determine the number of eigenfunctions for dimension reduction except B-spline basis. Notably, the Model 3 (Gaussian radial basis) shows 100 percentages in the recall whereas relatively lower percentages in the precision for all p compared to other Models. The Model 1 (linear kernel) works better than the other two models in consideration of the leave-one-out estimates of the generalization error.

There are many approaches developed in multivariate statistical analysis for classification. We compare the performance of classification in SVM with those of two other mothods: Fisher linear discriminant analysis (Fisher LDA) as a parametric approach and kernel method as a nonparametric aproach. In Figure 11, we graph the leave-one-out error estimates of the generalization errors for the three classification methods over the various values of p. The results of the nonparametric method are obtained from a choice of kernel being Epanechnikov and a bandwidth being 4 based on cross validation criteria. Note that each of the classification methods does not show monotonicity in the error estimates as p increases. The error estimates are influenced by the decision of the truncation p in the functional PCA of $X(t)$.

Table 4: Leave-one-out estimates of the training error, the Recall and the Precision under three SVM models. Types of kernel functions used in the models are linear, the third order polynomial and gaussian radial basis functions, respectively. The predictors are smoothed by polygonal basis.

		$p = 3$	$p = 4$	$p = 5$	$p = 6$	$p = 7$	$p = 8$	$p = 9$	$p = 10$
Model 1	*err*	46.44	39.89	38.20	35.02	35.21	35.96	40.07	37.64
	Rec	58.27	73.74	73.38	81.65	81.65	79.50	74.46	74.10
	Pre	55.10	59.42	61.08	62.53	62.36	62.08	59.14	61.49
Model 2	*err*	49.81	49.06	43.07	44.94	44.76	40.07	45.13	43.63
	Rec	83.45	82.37	91.73	86.69	83.45	85.97	82.01	82.37
	Pre	51.33	51.81	55.19	54.28	54.59	55.58	54.42	55.45
Model 3	*err*	48.13	47.94	47.94	47.94	47.94	47.94	47.94	47.94
	Rec	99.64	100	100	100	100	100	100	100
	Pre	51.97	52.06	52.06	52.06	52.06	52.06	52.06	52.06

Table 5: The same as Table 4 except the predictors are expanded by the B-spline basis.

		$p = 3$	$p = 4$	$p = 5$	$p = 6$	$p = 7$	$p = 8$	$p = 9$	$p = 10$
Model 1	err	47.19	42.32	37.27	35.02	34.64	35.96	38.01	33.15
	Rec	57.19	71.22	74.82	79.86	80.94	78.78	77.34	76.26
	Pre	54.45	57.56	61.72	62.89	63.03	62.22	60.56	65.63
Model 2	err	50.19	45.51	45.13	41.20	42.13	42.51	41.57	39.51
	Rec	83.81	82.01	85.61	92.81	93.53	92.45	88.13	92.45
	Pre	51.10	54.16	54.21	56.33	55.67	55.51	56.45	57.49
Model 3	err	48.13	47.94	47.94	47.94	47.94	47.94	47.94	47.94
	Rec	99.64	100	100	100	100	100	100	100
	Pre	51.97	52.06	52.06	52.06	52.06	52.06	52.06	52.06

Table 6: The same as Table 4 except the predictors are expanded by the Fourier basis.

		$p = 3$	$p = 4$	$p = 5$	$p = 6$	$p = 7$	$p = 8$	$p = 9$	$p = 10$
Model 1	err	47.57	41.76	40.07	36.14	36.89	38.39	36.33	36.70
	Rec	58.99	73.38	70.50	81.65	79.14	73.38	74.82	74.82
	Pre	53.95	57.79	59.76	61.52	61.28	60.90	62.65	62.28
Model 2	err	50.00	49.06	45.13	41.76	42.13	42.88	44.76	45.51
	Rec	82.37	82.37	85.61	92.81	92.45	90.65	86.33	85.61
	Pre	51.23	51.81	54.21	55.97	55.75	55.38	54.42	53.97
Model 3	err	48.13	47.94	47.94	47.94	47.94	47.94	47.94	47.94
	Rec	99.64	100	100	100	100	100	100	100
	Pre	51.97	52.06	52.06	52.06	52.06	52.06	52.06	52.06

However, the SVM classification shows the superiority in a missclassification error sense to other two methods. We find that SVM performs better than the others at $p = 6$ for the polygonal basis, $p = 10$ for the B-spline basis and $p = 6$ for the Fourier basis. To investigate the classification performances over the various combination of K and p, the configuration of combination are set to be $K = 45$ and $p = 15,\ 25,\ 35,\ 45$, $K = 75$ and $p = 15,\ 35,\ 55,\ 75$ and $K = 105$ and $p = 15,\ 35,\ 55,\ 75,\ 105$. Figures 12 and 13 provides the leave-one-out error rates over the combinations under the Fisher LDA, nonparametric discriminant analysis with the Epanechnikov kernel and the SVMs with polynomial kernels. The SVM with the linear kernel has been shown to perform well both the B-spline and the Fourier basis. It is seen that larger values of p and K do not warrant a small error rate, and specifically, the SVM with polynomial $d = 3$ in the B-spline basis provides the error rates such as 41.57% at $(K, p) = (45, 15)$, 44.01% at $(75, 15)$, 44.94% at $(45, 35)$ and 45.69% at $(75, 35)$.

The way of treating functional data as multivariate vector has been discussed in Section 1.3, in which, as its weakness, we found that the horizontal dependencies within the same functional data are neglected. Instead of using the basis expansion approach, if the medfly data are considered to be vectors of size 30 by viewing the data as multivariate vectors we discover that lack of incoporation of the dependencies results in a higher leave-one-out error rate: the SVM shows 47.94% error rate.

We also provide the results of two methods; methods using PCA and without using PCA. See Table 7. We find that the SVM using PCA show slightly better performances than the SVM without PCA. It can be thought to support the idea why we need to use PCA. Projection onto orthonormal eigenfunctions, capturing the prominent directions, makes the process of classification easier.

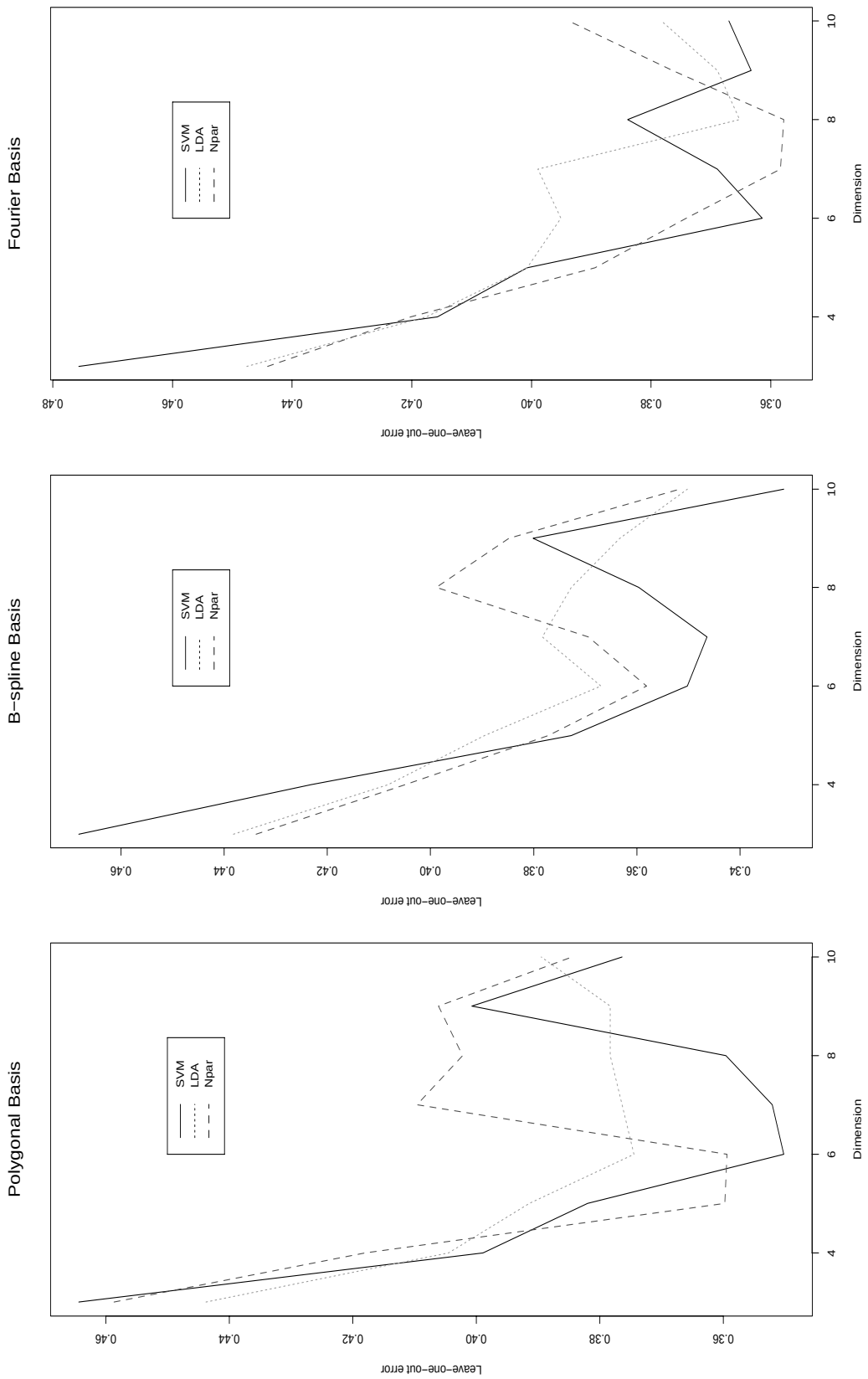

Figure 11: Comparisons of leave-one-out error estimates of SVM (linear kernel) with Fisher LDA and nonparametric discriminant analysis.

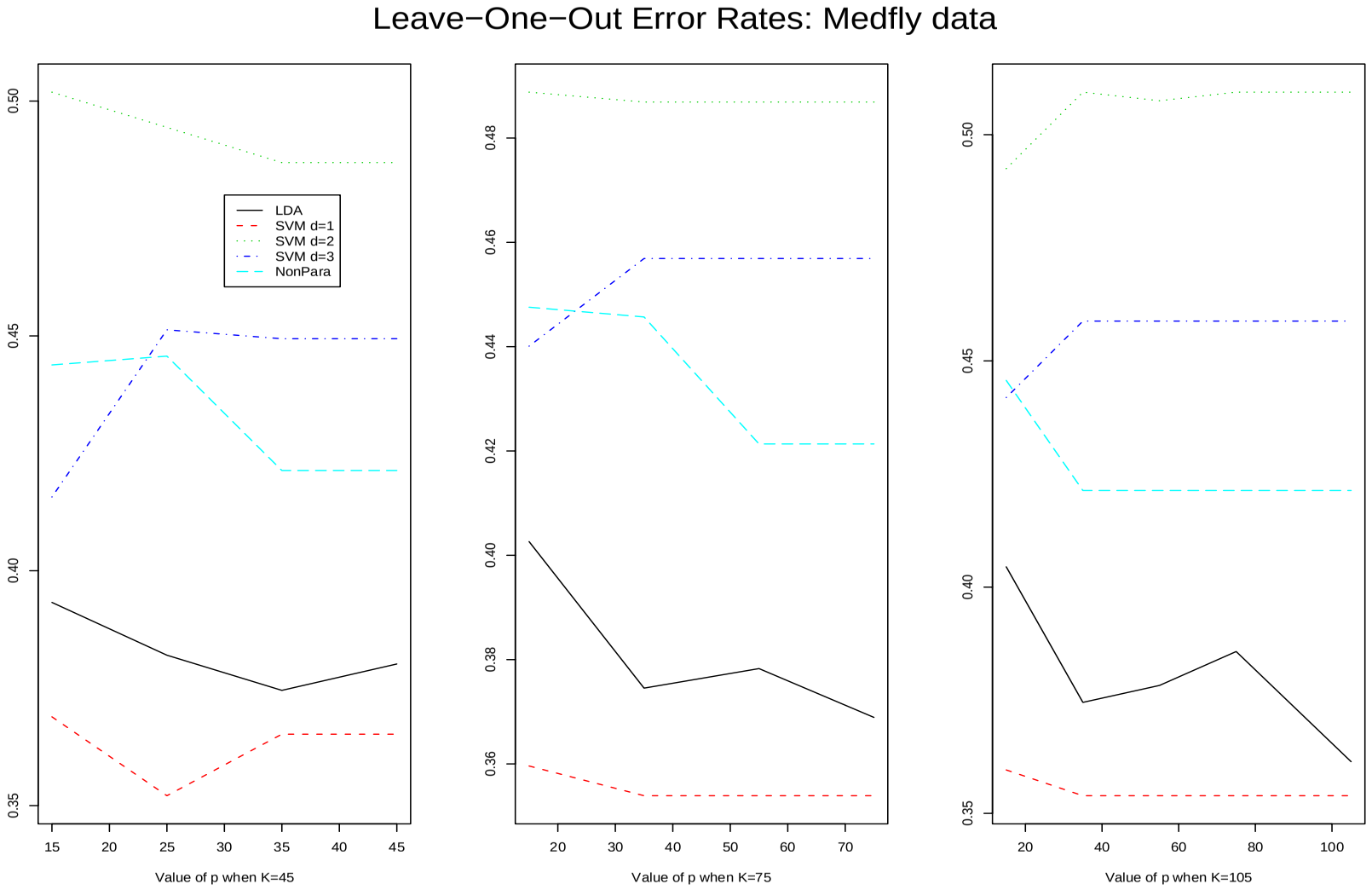

Figure 12: The B-spline basis. Comparisons of leave-one-out error estimates of SVM (linear, polynomial $d = 2$ and polynimial $d = 3$ kernels) with Fisher LDA and nonparametric discriminant analysis.

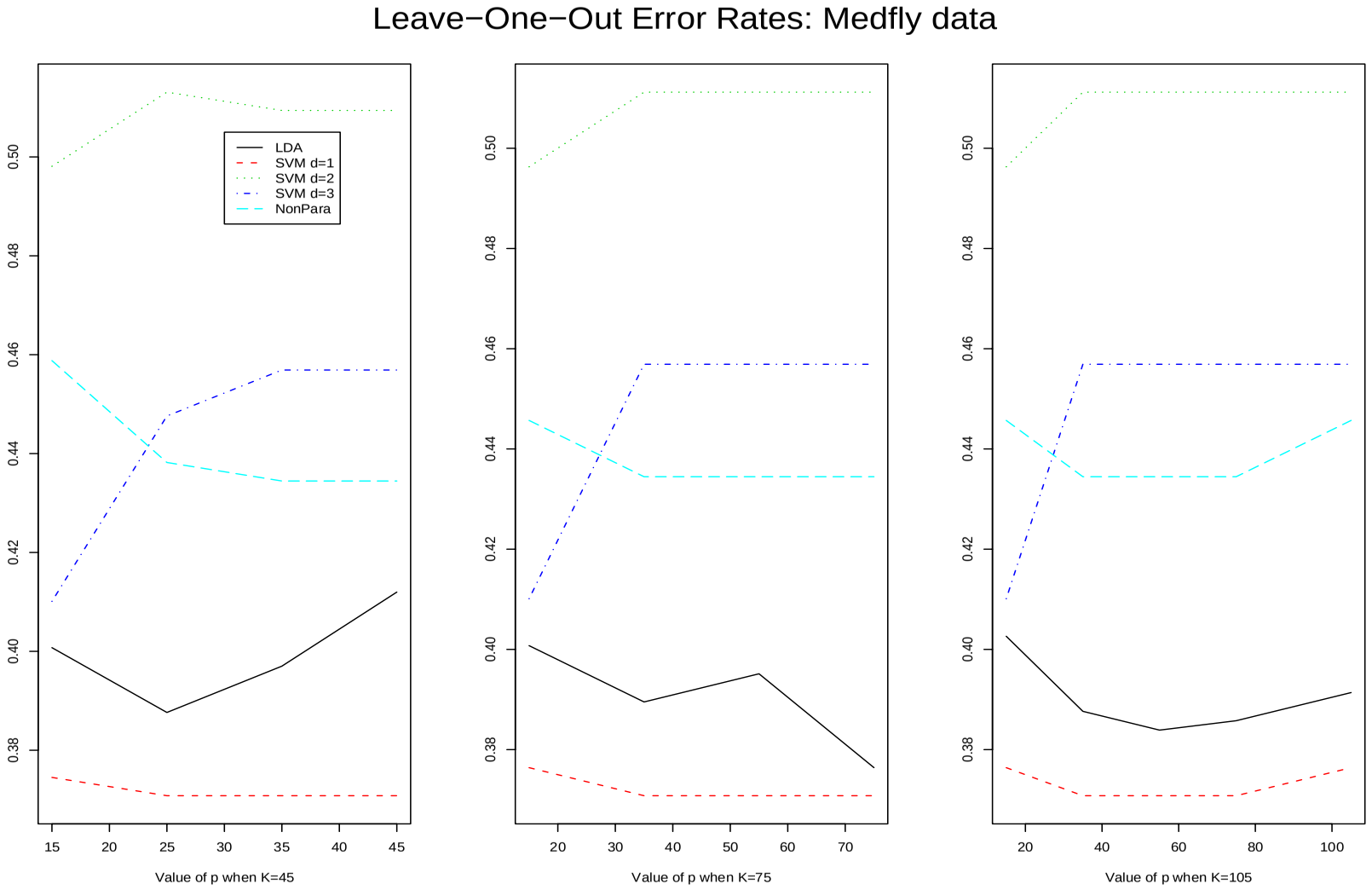

Figure 13: The Fourier basis. Comparisons of leave-one-out error estimates of SVM (linear, polynomial $d = 2$ and polynimial $d = 3$ kernels) with Fisher LDA and nonparametric discriminant analysis.

Table 7: Performances of SVM classifications with PCA and without PCA. Type of kernel used is linear.

		PCA	Not PCA
Polygonal	err	35.02	34.27
$(p = 6)$	Rec	81.65	75.54
	Pre	62.53	64.62
B spline	err	33.15	35.02
$(p = 10)$	Rec	76.26	76.26
	Pre	65.63	63.66
Fourier	err	36.14	36.70
$(p = 6)$	Rec	81.65	75.18
	Pre	61.52	62.20

4.6 Summary

The main goal of this chapter was to provide a method for classification of functional data. Since functional data is considered to be realizations of an infinite dimensional space, it is a difficult task to classify them. However, the functional PCA presented in Chapter II allows us to reduce the dimensionality by projecting the functional data onto the set of orthonormal basis functions. The set of principal component functions extracts the main sources of variability that the data contains. Orthogonality of the set of the basis functions induces the uniqueness of principal component scores for each $X_j(t)$, and then each principal component score in the lower dimensional space is used for classification, instead of the data in a functional form.

Support Vector Machine seeks for an optimal separating hyperplane to identify the decision boundary for classification in a feature space. Applying SVM, one builds up a generalized classifier capable of detecting the classes of new members. Application to medflies data reveals that among the models that we examined, SVM that uses a linear kernel results in the best performance. Determination of the values of p from scree plots is shown to be appropriate. Finally measures of efficiency such as the training error, the *recall* and the *precision* under different SVM models have been presented for comparisons with application to the medflies data.

CHAPTER V

FUNCTIONAL ROBUST REGRESSION

5.1 Theoretical Foundation

Consider the functional linear model

$$Y = \langle \beta, X \rangle_{L_2[0,1]} + e. \tag{5.1}$$

In reality, $x_i,\ (i = 1, \ldots, n)$, a realization of the stochastic precess $X(t), t \in [0, 1]$, can only be observed at a set of discrete points. However, in order to clarify reasoning here, it is assumed that the discrete raw data is functionalized over a set of fine grid time points $\{0 \leq t_1, t_2, \ldots, t_N \leq 1\}$. With appropriate choices of basis functions ψ's and the number of them being used p, we have

$$x_i = \sum_{l=1}^{p} c_{i,l}\psi_l,\ \ i = 1, \ldots, n. \tag{5.2}$$

The Fourier basis is a good choice if the observed curves are uniformly smooth with limited features, and especially if the curves appear to be periodic waves. On the other hand, splines or wavelets may be a better choice if there are lots of local features which may be relevant for the statistical analysis.

Under the model, the goal is to estimate the parameter function $\beta(t),\ t \in [0, 1]$ with the following criterion:

$$\min_{\beta \in \mathcal{B}_m} \frac{1}{n} \sum_{i=1}^{n} |y_i - \langle \beta, x_i \rangle|_\epsilon + \lambda \|\beta\|^2_{\mathcal{B}_m}, \tag{5.3}$$

where

$$|x|_\epsilon = \begin{cases} 0 & \text{if } |x| < \epsilon \\ |x| - \epsilon & \text{otherwise} \end{cases} \tag{5.4}$$

and $\mathcal{B}_m$ is the Sobolev-Hilbert space defined by

$$\mathcal{B}_m = \{\beta : \beta, \beta', \beta^{(2)}, \ldots, \beta^{(m-1)} \text{ are absolutely continuous and } \beta^{(m)} \in L_2[0,1]\}.$$

One refers to Vapnik (1995) and Smola and Schölkopf (1998) for the complete details of the ϵ-insensitive loss function and the Support Vector regression. For each $\beta \in \mathcal{B}_m$ we have by Taylor's expansion with remainder

$$\beta(t) = \sum_{i=0}^{m-1} \frac{\beta^{(i)}(0)}{i!} t^i + \int_0^1 \frac{(t-u)_+^{m-1}}{(m-1)!} \beta^{(m)}(u) du, \tag{5.5}$$

where $u_+ = u\mathbf{1}_{[0,1]}(u)$.

Define

$$\phi_i(t) = \frac{t^{i-1}}{(i-1)!},\ i = 1, 2, \ldots, \tag{5.6}$$

and

$$\mathcal{H}_0 = \text{span}\{\phi_1, \ldots, \phi_m\}. \tag{5.7}$$

For any $f, g \in \mathcal{H}_0$, define the inner product

$$\langle f, g \rangle = \sum_{i=0}^{m-1} f^{(i)}(0) g^{(i)}(0). \tag{5.8}$$

It is easy to see that the ϕ_i form an orthonormal basis for $\mathcal{H}_0$.

Consider the space

$$\mathcal{H}_1 = \{\beta \ : \ \beta^{(i)}(0) = 0,\ 0 \le i \le m-1,\ \beta, \beta' \ldots \beta^{(m-1)} \text{ are absolutely continuous and } \beta^{(m)} \in L_2[0,1]\}.$$

Note that any function in $\mathcal{H}_1$ satisfies

$$\beta(t) = \int_0^1 \frac{(t-u)_+^{m-1}}{(m-1)!} \beta^{(m)}(u) du. \tag{5.9}$$

The inner product on $\mathcal{H}_1$ is defined to be

$$\langle f, g\rangle = \int_0^1 f^{(m)} g^{(m)}(u)du. \tag{5.10}$$

Furthermore, it can be seen that the spaces $\mathcal{H}_0$ and $\mathcal{H}_1$ are the reproducing kernel Hilbert spaces (r.k.h.s.) with reproducing kernels

$$\begin{aligned} R_0(s,t) &= \sum_{i=1}^{m} \phi_i(s)\phi_i(t) && (5.11)\\ R_1(s,t) &= \int_0^1 \frac{(s-u)_+^{m-1}(t-u)_+^{m-1}}{[(m-1)!]^2}du. && (5.12) \end{aligned}$$

Thus, $\mathcal{B}_m$ can be written as the direct sum of two reproducing kernel Hilbert spaces, and hence each element in the space has a unique representation of the form: for all $\beta \in \mathcal{B}_m$,

$$\beta = \beta_0 + \beta_1,$$

where $\beta_0 \in \mathcal{H}_0$ and $\beta_1 \in \mathcal{H}_1$. Furthermore we have $R = R_0 + R_1$. See Wahba (1990) for details of r.k.h.s.

It is easily seen that

$$\beta = \beta_\lambda = \sum_{j=1}^{m} a_j\phi_j(\cdot) + \sum_{k=1}^{N} b_k R_1(t_k, \cdot). \tag{5.13}$$

Consequently, the norm is defined to be, with $\boldsymbol{a} = (a_1, \ldots, a_m)'$, $\boldsymbol{b} = (b_1, \ldots, b_N)'$,

$$\begin{aligned} \|\beta\|^2_{\mathcal{B}_m} &= \langle\beta, \beta\rangle_{\mathcal{B}_m}\\ &= \langle\sum_{j=1}^{m} a_j\phi_j(\cdot) + \sum_{k=1}^{N} b_k R_1(t_k, \cdot), \sum_{j'=1}^{m} a_{j'}\phi_{j'}(\cdot) + \sum_{k'=1}^{N} b_{k'} R_1(t_{k'}, \cdot)\rangle_{\mathcal{B}_m}\\ &= \sum_{j=1}^{m}\sum_{j'=1}^{m} a_j a_{j'}\langle\phi_j, \phi_{j'}\rangle + \sum_{k=1}^{N}\sum_{k'=1}^{N} b_k b_{k'} R_1(t_k, t_{k'})\\ &= \underbrace{\boldsymbol{a}'\tilde{R}_0\boldsymbol{a}}_{:=\|\beta\|_0^2} + \underbrace{\boldsymbol{b}'\tilde{R}_1\boldsymbol{b}}_{:=\|\beta\|_1^2}, && (5.14) \end{aligned}$$

where $\tilde{R}_0 = \Big[\langle \phi_j, \phi_{j'} \rangle\Big]_{j,j'=1}^{m}$ and $\tilde{R}_1 = \Big[R_1(t_k, t_{k'})\Big]_{k,k'=1}^{N}$. We can give more flexibility to the second term in (5.3): since the squared norm of β is decomposed into two components, we can have, instead of $\lambda \|\beta\|^2_{\mathcal{W}_m}$,

$$\lambda_0 \|\beta\|_0^2 + \lambda_1 \|\beta\|_1^2, \tag{5.15}$$

where λ_0 and $\lambda_1 \geq 0$. For special cases, either λ_0 or λ_1 or both could be zero, then further simplification of the models is made. However the simplification may cause the problem of estimation to be of ill-posed. Depending upon research interest, we may impose restrictions on λ's such as $0 \leq \lambda_0 < \lambda_1$: the bigger value of λ_1 is, the more contribution β_1 makes to the estimation of β. The simplest situation is the case of $\lambda = \lambda_0 = \lambda_1$.

Now

$$\begin{aligned} \langle \beta, x_i \rangle &= \langle \sum_{j=1}^{m} a_j \phi_j + \sum_{k=1}^{N} b_k R_1(t_k, \cdot), \sum_{l=1}^{p} c_{i,l} \psi_l \rangle \\ &= \sum_{j=1}^{m} \sum_{l=1}^{p} a_j c_{i,l} \langle \phi_j, \psi_l \rangle + \sum_{k=1}^{N} \sum_{l=1}^{p} b_k c_{i,l} \langle R_1(t_k, \cdot), \psi_l \rangle \\ &= \boldsymbol{a}' \Sigma_0 \boldsymbol{c}_i + \boldsymbol{b}' \Sigma_1 \boldsymbol{c}_i, \end{aligned} \tag{5.16}$$

where

$$\Sigma_0 = \Big[\langle \phi_j, \psi_l \rangle\Big]_{m \times p},\ \Sigma_1 = \Big[\langle R_1(t_j, \cdot), \psi_l \rangle\Big]_{N \times p} \text{ and } \boldsymbol{c}_i = (c_{i,1}, \ldots, c_{i,p})'.$$

As a result, it follows from (5.14), (5.15) and (5.16) that the minimization problem in (5.3) can be written as

$$\min \frac{1}{n} \sum_{i=1}^{n} |y_i - \boldsymbol{a}' \Sigma_0 \boldsymbol{c}_i - \boldsymbol{b}' \Sigma_1 \boldsymbol{c}_i|_\epsilon + (\lambda_0 \boldsymbol{a}' \tilde{R}_0 \boldsymbol{a} + \lambda_1 \boldsymbol{b}' \tilde{R}_1 \boldsymbol{b}). \tag{5.17}$$

Introducing slack variables $\gamma_i,\ \gamma_i^*$, we have the following equivalent problem:

$$\min \frac{1}{n} \sum_{i=1}^{n} (\gamma_i + \gamma_i^*) + (\lambda_0 \boldsymbol{a}' \tilde{R}_0 \boldsymbol{a} + \lambda_1 \boldsymbol{b}' \tilde{R}_1 \boldsymbol{b}), \tag{5.18}$$

subject to

$$\begin{aligned} y_i - \boldsymbol{a}'\Sigma_0\boldsymbol{c}_i - \boldsymbol{b}'\Sigma_1\boldsymbol{c}_i &\leq \epsilon + \gamma_i,\ i = 1, \ldots, n, \\ \boldsymbol{a}'\Sigma_0\boldsymbol{c}_i + \boldsymbol{b}'\Sigma_1\boldsymbol{c}_i - y_i &\leq \epsilon + \gamma_i^*,\ i = 1, \ldots, n, \\ \gamma_i,\ \gamma_i^* &\geq 0,\ i = 1, \ldots, n. \end{aligned} \tag{5.19}$$

Thus we have the Lagrangian functional:

$$\begin{aligned} \mathcal{L}(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\gamma}, \boldsymbol{\gamma}^*; \boldsymbol{u}, \boldsymbol{u}^*, \boldsymbol{w}, \boldsymbol{w}^*) &= \frac{1}{n}\sum_{i=1}^{n}(\gamma_i + \gamma_i^*) + \lambda_0\boldsymbol{a}'\tilde{R}_0\boldsymbol{a} + \lambda_1\boldsymbol{b}'\tilde{R}_1\boldsymbol{b} \\ &\quad - \sum_{i=1}^{n}\alpha_i\Big(\boldsymbol{a}'\Sigma_0\boldsymbol{c}_i + \boldsymbol{b}'\Sigma_1\boldsymbol{c}_i - y_i + \epsilon + \gamma_i\Big) \\ &\quad - \sum_{i=1}^{n}\alpha_i^*\Big(y_i - \boldsymbol{a}'\Sigma_0\boldsymbol{c}_i - \boldsymbol{b}'\Sigma_1\boldsymbol{c}_i + \epsilon + \gamma_i^*\Big) \\ &\quad - \sum_{i=1}^{n}\beta_i\gamma_i - \sum_{i=1}^{n}\beta_i^*\gamma_i^*. \end{aligned}$$

Let $\boldsymbol{C}$ be a $p \times n$ matrix by $\Big[\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_n\Big]$. Then we may express $\mathcal{L}$ as

$$\begin{aligned} \mathcal{L} &= \frac{1}{n}(\boldsymbol{\gamma} + \boldsymbol{\gamma}^*)'\mathbf{1} + \lambda_0\boldsymbol{a}'\tilde{R}_0\boldsymbol{a} + \lambda_1\boldsymbol{b}'\tilde{R}_1\boldsymbol{b} + (\boldsymbol{u} - \boldsymbol{u}^*)'\boldsymbol{y} - \epsilon(\boldsymbol{u} + \boldsymbol{u}^*)'\mathbf{1} \\ &\quad -(\boldsymbol{a}'\Sigma_0 + \boldsymbol{b}'\Sigma_1)\boldsymbol{C}(\boldsymbol{u} - \boldsymbol{u}^*) - (\boldsymbol{u} + \boldsymbol{w})'\boldsymbol{\gamma} - (\boldsymbol{u}^* + \boldsymbol{w}^*)'\boldsymbol{\gamma}^*. \end{aligned}$$

Differentiation of $\mathcal{L}$ with respect to $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{\gamma}$ and $\boldsymbol{\gamma}^*$ produces, assuming all inverse matrices are well defined,

$$\boldsymbol{a} = \frac{1}{2\lambda_0}\tilde{R}_0^{-1}\boldsymbol{\Sigma}_0\boldsymbol{C}(\boldsymbol{u} - \boldsymbol{u}^*). \tag{5.20}$$

$$\boldsymbol{b} = \frac{1}{2\lambda_1}\tilde{R}_1^{-1}\boldsymbol{\Sigma}_1\boldsymbol{C}(\boldsymbol{u} - \boldsymbol{u}^*). \tag{5.21}$$

$$n(\boldsymbol{u} + \boldsymbol{w}) = \mathbf{1}. \tag{5.22}$$

$$n(\boldsymbol{u}^* + \boldsymbol{w}^*) = \mathbf{1}. \tag{5.23}$$

From (5.22) and (5.23), it follows that in $\mathcal{L}$

$$\frac{1}{n}(\boldsymbol{\gamma}+\boldsymbol{\gamma}^*)'\mathbf{1}-(\boldsymbol{u}+\boldsymbol{w})'\boldsymbol{\gamma}-(\boldsymbol{u}^*+\boldsymbol{w}^*)'\boldsymbol{\gamma}^*=0.$$

It can then be seen from (5.20) and (5.21) that

$$\begin{aligned}&\lambda_0\boldsymbol{a}'\tilde{R}_0\boldsymbol{a}+\lambda_1\boldsymbol{b}'\tilde{R}_1\boldsymbol{b}-(\boldsymbol{a}'\Sigma_0+\boldsymbol{b}'\Sigma_1)\boldsymbol{C}(\boldsymbol{u}-\boldsymbol{u}^*)\\ =\;&-\frac{1}{2}(\boldsymbol{u}-\boldsymbol{u}^*)'\mathbf{K}(\boldsymbol{u}-\boldsymbol{u}^*),\end{aligned}\tag{5.24}$$

where $\mathbf{K}=\frac{1}{2}\boldsymbol{C}'(\boldsymbol{\Sigma}_0'\tilde{R}_0^{-1}\boldsymbol{\Sigma}_0/\lambda_0+\boldsymbol{\Sigma}_1'\tilde{R}_1^{-1}\boldsymbol{\Sigma}_1/\lambda_1)\boldsymbol{C}$. Substituting the results above into $\mathcal{L}$ reduces the dual problem to minimization of $\mathcal{W}$ with respect to $\boldsymbol{u}$ and $\boldsymbol{u}^*$ where

$$\mathcal{W}(\boldsymbol{u},\boldsymbol{u}^*)=\frac{1}{2}(\boldsymbol{u}-\boldsymbol{u}^*)'\mathbf{K}(\boldsymbol{u}-\boldsymbol{u}^*)-(\boldsymbol{u}-\boldsymbol{u}^*)'\boldsymbol{y}+\epsilon(\boldsymbol{u}+\boldsymbol{u}^*)'\mathbf{1}.\tag{5.25}$$

The minimization is carried out subject to having

$$\mathbf{0}\le\boldsymbol{u},\;\boldsymbol{u}^*\le\frac{1}{n}\mathbf{1}.\tag{5.26}$$

From the optimizers $\hat{\boldsymbol{u}}$ and $\hat{\boldsymbol{u}}^*$, we can estimate $\boldsymbol{a}$ in (5.20) and $\boldsymbol{b}$ in (5.21), hence β in (5.13). That is, we have

$$\hat{\boldsymbol{a}}=\frac{1}{2\lambda_0}\tilde{R}_0^{-1}\boldsymbol{\Sigma}_0\boldsymbol{C}(\hat{\boldsymbol{u}}-\hat{\boldsymbol{u}}^*)\text{ and }\hat{\boldsymbol{b}}=\frac{1}{2\lambda_1}\tilde{R}_1^{-1}\boldsymbol{\Sigma}_1\boldsymbol{C}(\hat{\boldsymbol{u}}-\hat{\boldsymbol{u}}^*),\tag{5.27}$$

which produces

$$\hat{\beta}\;=\;\sum_{j=1}^{m}\hat{a}_j\phi_j+\sum_{k=1}^{N}\hat{b}_kR_1(t_k,\cdot).\tag{5.28}$$

$$\hat{y}\;=\;\hat{\boldsymbol{a}}'\Sigma_0\boldsymbol{C}+\hat{\boldsymbol{b}}'\Sigma_1\boldsymbol{C}=(\hat{\boldsymbol{u}}-\hat{\boldsymbol{u}}^*)'\mathbf{K}.\tag{5.29}$$

It must be pointed out that the parameters λ_0,λ_1 and ϵ are assumed to be known so far. However, in practical issues, it may not be true. Since the function β in (5.3) depends on the parameters, investigation of the analytical search for the parameters from the training data is so.

5.2 SMO Algorithm for the Minimization Problem

The Sequential Minimal Optimization (SMO) algorithm, introduced by Platt (1999), searches the optimizers through the feasible region and minimizes the dual problem (5.25). In this Section, we describe a modified SMO algorithm in which the bias term is not needed. See Vogt (2002). Let $\mathbf{K} = [k_{ij}]_{i,j=1}^n$. The objective dual functional is given by

$$\begin{aligned} \mathcal{W} &= \frac{1}{2}(\boldsymbol{u}-\boldsymbol{u}^*)'\mathbf{K}(\boldsymbol{u}-\boldsymbol{u}^*) - (\boldsymbol{u}-\boldsymbol{u}^*)'\boldsymbol{y} + \epsilon(\boldsymbol{u}+\boldsymbol{u}^*)'\mathbf{1} \\ &= \frac{1}{2}\sum_{i=1}^n\sum_{i=1}^n(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)k_{ij} - \sum_{i=1}^n y_i(\alpha_i-\alpha_i^*) + \epsilon\sum_{i=1}^n(\alpha_i+\alpha_i^*) \end{aligned} \tag{5.30}$$

subject to

$$0 \le \alpha_i,\ \alpha_i^* \le n^{-1} \text{ for } i = 1,\ldots,n. \tag{5.31}$$

Without loss of generality, we can assume that α_1 and α_1^* are the variables to be optimized, and then go to the next α's. The choice of an index i is based on the reasoning of Platt. The objective functional $\mathcal{W}$ can be expressed in terms of α_1 and α_1^*.

$$\mathcal{W} = \frac{1}{2}k_{11} + \Big(\sum_{j=2}^n(\alpha_j-\alpha_j^*)k_{1j} - y_1\Big)(\alpha_1-\alpha_1^*) + \epsilon(\alpha_1-\alpha_1^*) + const. \tag{5.32}$$

Let $\alpha_i = \alpha_i^{old}$ and $\alpha_i^* = \alpha_i^{*old}$ for $i = 2,\ldots,n$ and $E_i = \hat{y}_i - y_i$, where $\hat{y}_i = \sum_{j=1}^n(\alpha_j^{old} - \alpha_j^{*old})k_{ij}$. Then it can be easily seen that

$$\begin{aligned} \mathcal{W} &= \frac{1}{2}k_{11}\alpha_1^2 + \Big(E_1 - (\alpha_1^{old}-\alpha_1^{*old})k_{11} + \epsilon\Big)\alpha_1 \\ &+ \frac{1}{2}k_{11}\alpha_1^{*2} - \Big(E_1 - (\alpha_1^{old}-\alpha_1^{*old})k_{11} - \epsilon\Big)\alpha_1^* + const. \end{aligned} \tag{5.33}$$

Taking derivatives with respect to α_1 and α_1^* and equating them to zero imply that

$$\alpha_1 = \alpha_1^{old} - \alpha_1^{*old} - \frac{E_1+\epsilon}{k_{11}} \tag{5.34}$$

$$\alpha_1^* = -\alpha_1 - \frac{2\epsilon}{k_{11}}. \tag{5.35}$$

Note that $\mathcal{W}$ contains no cross product term, and the solutions of the constrained problem are found by clipping α_1 and α_1^* to the interval $[0, n^{-1}]$:

$$\alpha_1^{new} = \min\{\max\{\alpha_1, 0\}, n^{-1}\} \tag{5.36}$$

$$\alpha_1^{*new} = \min\{\max\{\alpha_1^*, 0\}, n^{-1}\}. \tag{5.37}$$

In order for a point to be the optimal, the Karush-Kuhn-Tucker (KKT) conditions are fulfilled. The KKT conditions for the optimization are particularly simple: For $i = 1 \ldots, n$, the optimal solution α_i should satisfy one of the three conditions:

$$\begin{aligned} \alpha_i = 0 \quad &\wedge \quad \epsilon + E_i \geq 0 \\ 0 < \alpha_i < n^{-1} \quad &\wedge \quad \epsilon + E_i = 0 \\ \alpha_i = n^{-1} \quad &\wedge \quad \epsilon + E_i \leq 0. \end{aligned}$$

Similarly, for α_i^*,

$$\begin{aligned} \alpha_i^* = 0 \quad &\wedge \quad \epsilon - E_i \geq 0 \\ 0 < \alpha_i^* < n^{-1} \quad &\wedge \quad \epsilon - E_i = 0 \\ \alpha_i^* = n^{-1} \quad &\wedge \quad \epsilon - E_i \leq 0. \end{aligned}$$

5.3 Simulation Study

In this section, we compare the performance of our functional regression method on two datasets with SMO algorithm. We generate stationary Gaussian processes as a predictor $X(t),\ t \in [0, 1]$ using (4.11) and (4.12) and choose three true functions β, linear, quadratic and sinusoidal functions, respectively. A standard Gaussian noise adds to the dependent variable Y. With the choices of $m = 2$ or 3, the Sobolev-Hilbert space is decomposed to two orthogonal spaces, leading to express the parameter function as a direct sum in (5.13). They amount to present the function as a composite

of a linear and a nonlinear but smooth enough function if m is chosen to be 2 and as a quadratic and a higher order funtion if $m = 3$.

When the true function is linear, our estimate is close to the true function and the estimates of the dependent variable are so. In the case where the true function is quadratic and m is chosen as 3, we find that the values of the dependent variable is well estimated. Figure 14 compares the values of the estimated dependent variable with the observed values. The parameter estimate in the case of the sinusoidal function with $m = 3$ is shown in Figure 15, which provides the performance as reliable as those of the linear and quadratic functions. The value m and the regularization parameters λ_0 and λ_1 play crucial roles in the estimation of the function. The parameters λ's control the trade-off between goodness-of-fit and smoothness of the estimated β function. It is clear that if the parameter function β is a cubic function, it would be desirable to choose m to be 4, accommodating all the cubic functions. Consequently, the space $\mathcal{H}_0$ consisting of all the cubic functions presents a primary structure of the function β, and the corresponding λ_0 will be larger to give more weights. However consider the case where the function β is a higher order polynomial function or even not an analytic function, then determination of the value m would be challenging or impossible. Once m is chosen, which is equivalent to decomposition of the space $\mathcal{H}$ into two orthogonal spaces, each of the spaces are weighted by the same regularizers. The process of function estimation is sensitive to the regularizers, and even worse when the value m is not appropriately chosen.

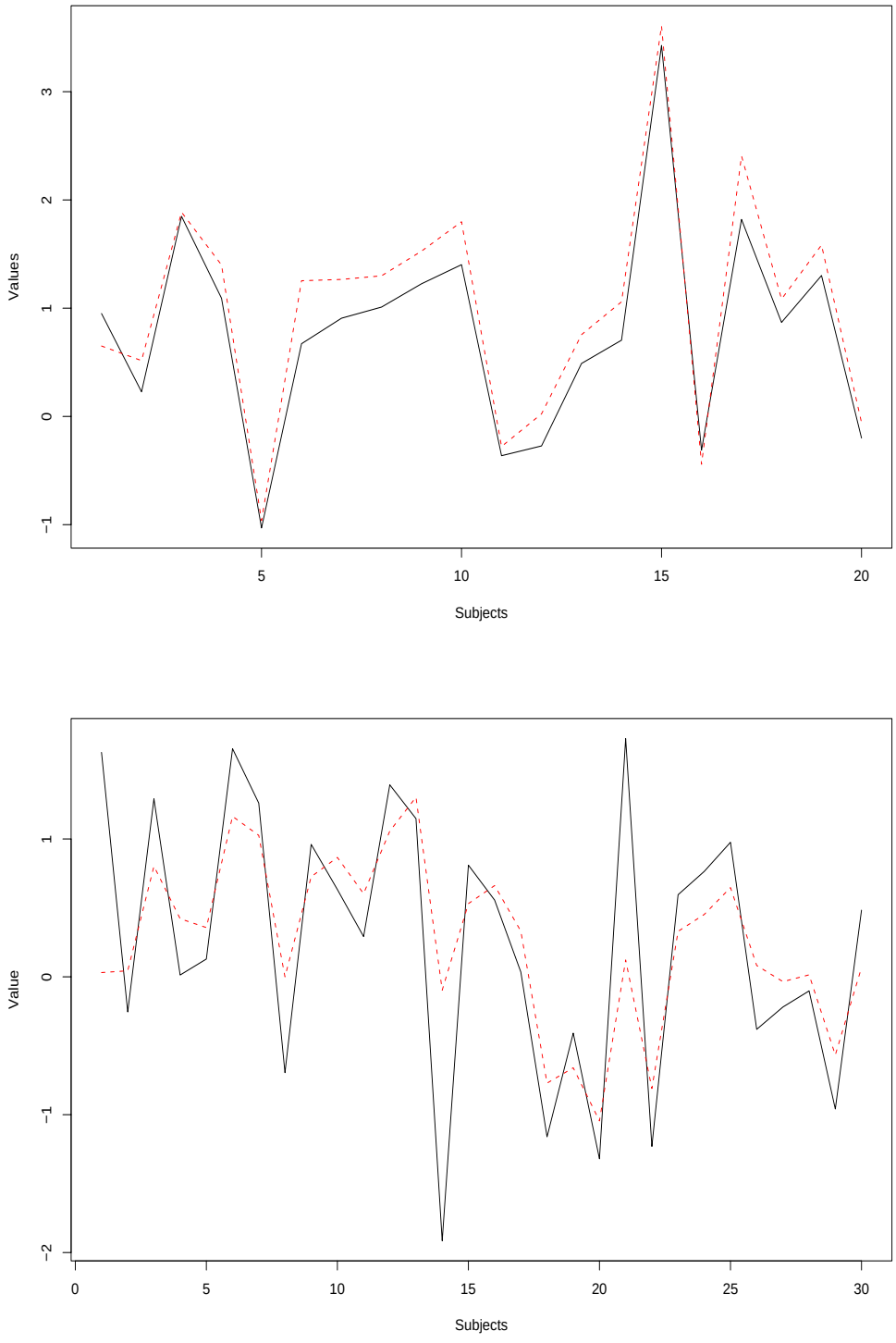

Figure 14: Simulation data. The estimated values (red dotted lines) and data values (black solid lines) are shown. Upper: the β function is linear with $m = 2$. Bottom: the β function is quadratic with $m = 3$.

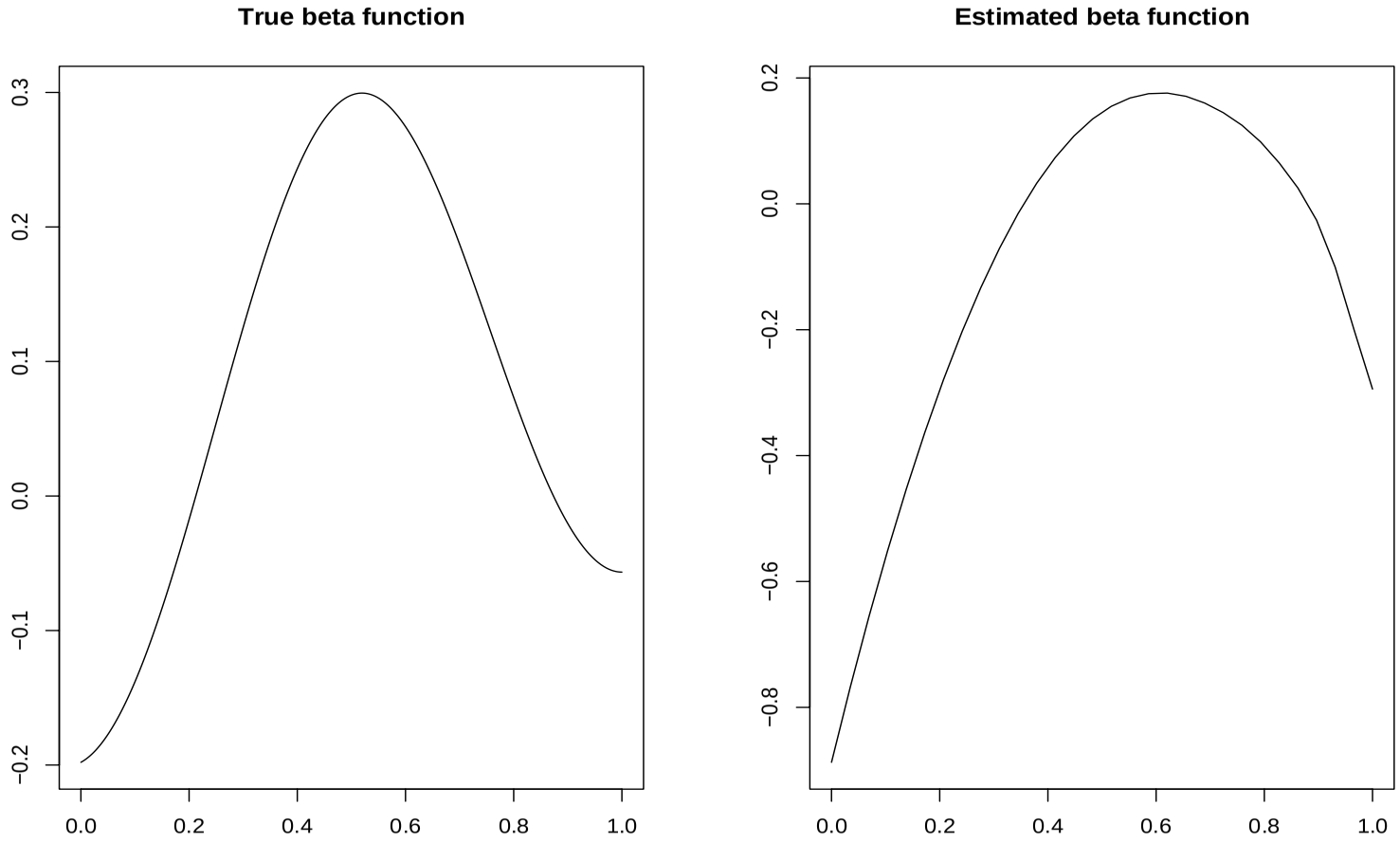

Figure 15: Comparision of the true β function (left) and an estimated function (right). Functional data X is Gaussian Process, β is a sinusoidal function, $Y = \langle \beta, X \rangle + N(0,1)$, $\epsilon = 3$, $\lambda_0 = 0.7$, $\lambda_1 = 1/100000$ and $m = 3$.

5.4 Example: Lipoprotein Density Profiles Data

It is well known that abundance cholesterol in bloodstream causes a buildup of plague in artery walls, resulting in higher risks for the development of atherosclerosis and coronary heart disease. Establishing serum cholesterol as a useful indicator for an individual's predisposition for coronary heart disease motivates its routine measurement. The precision and accuracy has been improved in cholesterol measurement through technical advances of instrumentation, however, serum cholesterol concentration is still subject to some variation, and hence is unreliable.

The degree of heterogeneity of serum lipoprotein particles is an important factor in investigating the assessment of cardiovascular risk. No conclusive agreement has yet been made on the lipoprotein profile for accessing risk of cardiovascular disease accurately. However, much attention has been focused recently as possibility of heart disease on the relation of lipoprotein subclasses: Very Low Density Lipoprotein (VLDL), Low Density Lipoprotein (LDL) and High Density Lipoprotein (HDL). It is known that all the major lipoproteins show differing sub-distributions in serum. Excess cholesterol in the bloodstream in the form of LDL subclass can increase the plague buildup on artery walls. The higher the level of LDL, the greater a patient's risk of cardiovascular disease. Studies of structure in the lipoprotein density distribution provide a guide for assessment of cardiovascular disease risk.

The data in Figure 16 is a functional data in which the lipoprotein profile functions for each of 24 individuals patients are obtained. Each profile is defined over an interval which corresponds to the density (weight/volumn) scale of lipoprotein particles, and the value of profile function at a point in the density scale is the value of abundance of lipoprotein particles evaluated at the point. The three picks in each profile in the plot, from left to right, correspond to VLDL, LDL and HDL, respectively.

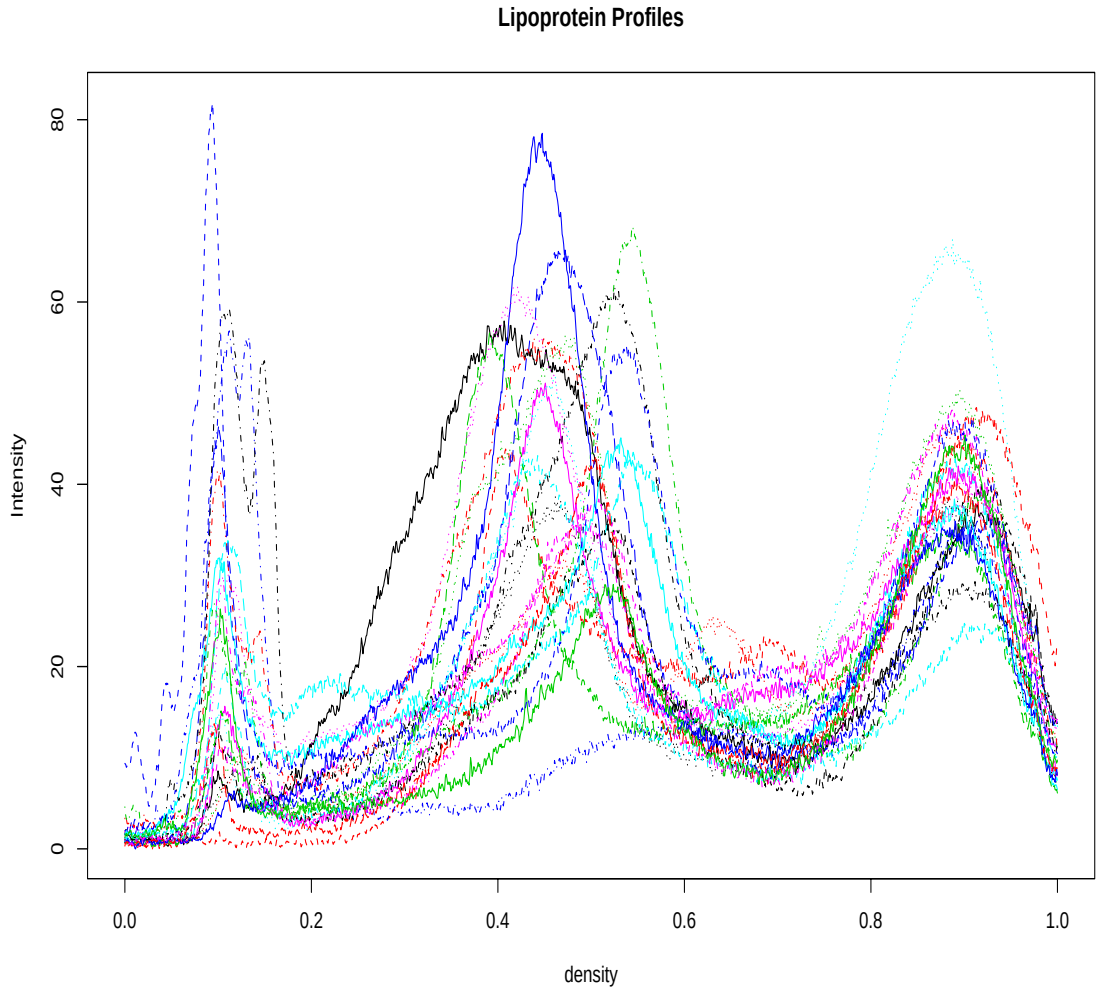

Figure 16: Lipoprotein density profiles data.

The response variable Y is taken as the total cholestrol level of 24 patients. Figure 17 shows the estimator of the regression weight fuction β described in Section 5.1 and the estimated values of the total cholestrol level via the model for each patients. The regularization paramters λ_0 and λ_1 are obtained through the grid searches, minimizing the ϵ insensitive loss function. From the estimated function, we find that VLDL and LDL have larger weights in predicting the total cholestrol level, supporting that the LDL subclass in the bloodstream can increase the risk of cardiovascular disease.

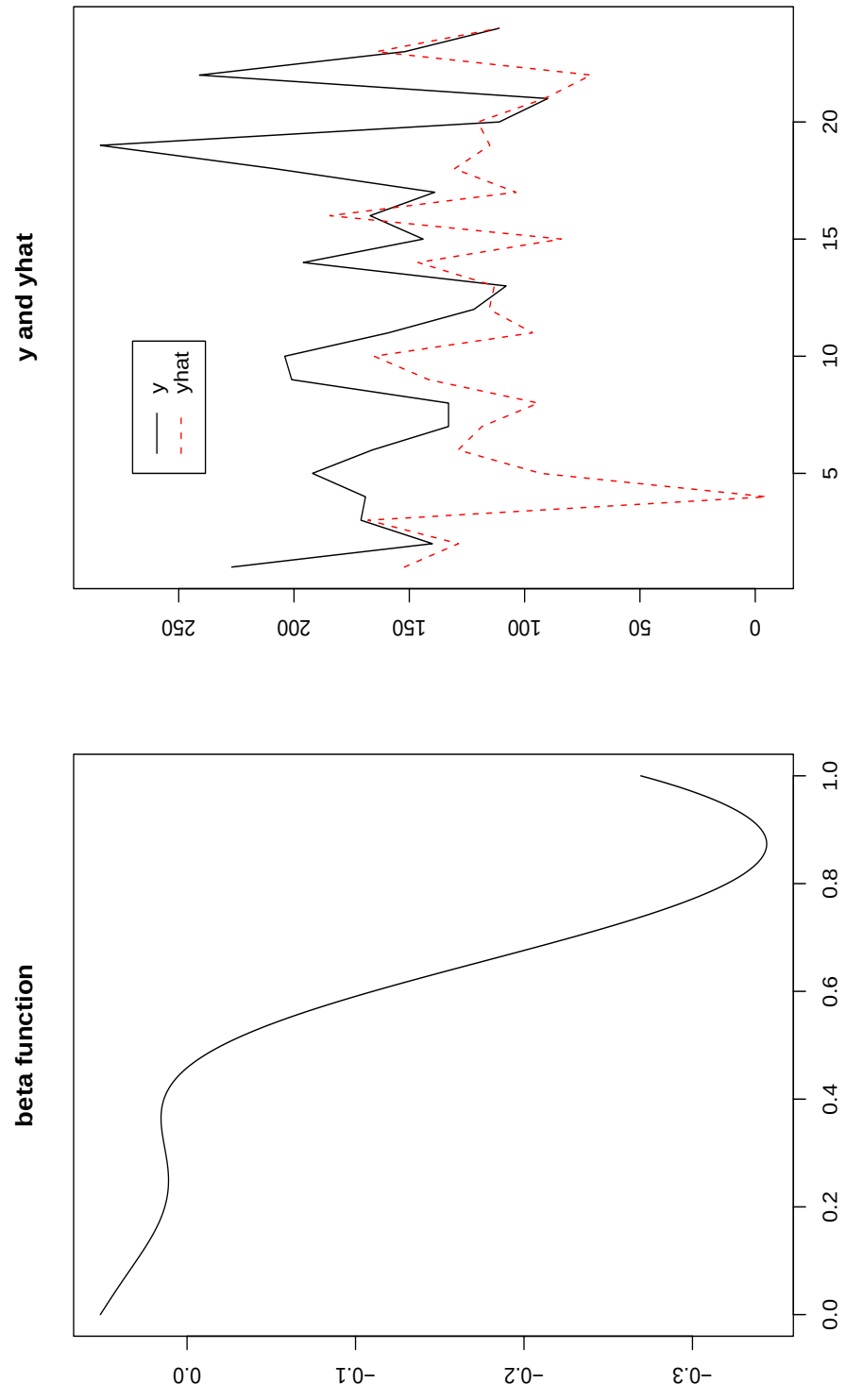

Figure 17: Lipoprotein profiles data. $\epsilon = 0.3$, $\lambda_0 = 75$, $\lambda_1 = 0.001$ and $m = 3$.

5.5 Discussion

As in the classical least squares estimator, functional linear regression is a standard method because of rather easy computation. Unfortunately, the standard functional regression analysis is quite sensitive when higher dimensional data are analyzed. An example of robust regression analysis is SVM regression in which the regression problem is solved by introduction of the ϵ-insensitive loss function. The ϵ-insensitive loss function ignores errors that are within the ϵ distance, and the points outside the ϵ tube only contribute to the cost function, leading to the sparse representation of the dual variables. Using a small number of data points warrants a significant advantage in practical computation. The parameter function β is restricted to be in the Sobolev-Hilbert space, which insures the smoothness of the function. The norm, in an L_2 sense, on the space plays a role of balancing the bias and variance trade-off.

Lipoprotein profiles data has been used to show how this idea is applied. The parameter function β is restricted to be an element in the Sobolev-Hilbert space, which imposes on the smoothness of the function. In that regards, selection of the value m is closely related to the level of smoothness, and hence it is crucial. Further study on the determination of the regularization parameters λ_0, λ_1 and ϵ are still needed, suggesting that cross validation criterion is useful.

CHAPTER VI

CONCLUSION

Most challenging aspects in studying functional data analysis emerge from that data are not observed completely and making inference about population from which data are drawn requires to consider the infinite dimensional function spaces. These may lead to that the process of estimation is not reliable, even impossible. One way to avoid the problems is to represent the discrete data in terms of basis expansions. We discussed about other possible alternatives in Chapter I.

Our classification analysis of functional data has employed the Support Vector Machine in which the input vectors consist of the projection coefficients of the functional data onto the space by a few orthonormal functions. Application to medflies data suggested that among the models that we examined, SVM with the linear kernel results in the best performance in the sense of the cross validation error rate. A variety of statistical methods, including SVM as well as a parametric method and nonparametric methods were used in the simulation studies.

We proposed the functional robust regression model for scalar responses. Under the regularization framework, the spaces of the parameter function was restricted to the Sobolev-Hilbert spaces, which insure that our estimate is a smooth function, and ϵ-insensitive loss functions were utilized. It has been well known that the regularization parameters make significant effects for estimation procedures. As discussed in the simulation study, the determination of the value m is another important parameter, which judges the primary structures of the Sobolev-Hilbert spaces.

REFERENCES

Alter, O., Brown, P., and Bostein, D. (2000). "Singular Value Decomposition for Genome-Wide Expression Data Processing and Modeling." *Proc. Nat. Acad. Sci., USA*, 97, 10101–10106.

Brown, M., Grundy, W., Lin, D., Christianini, N., Sugnet, C., Ares, J., and Haussler, D. (1999). "Support Vector Machine Classification of Microarray Gene Expression Data." Technical report. Department of Computer Science, University of California, Santa Cruz.

Burges, C. (1998). "A Tutorial on Support Vector Machines for Pattern Recognition." *Data Mining and Knowledge Discovery*, 2, 121–167.

Carey, J. R., Liedo, P., Müller, H. G., Wang, J. L., and Vaupel, J. W. (1998). "Dual Modes of Aging in Mediterranean Fruit Fly Females." *Science*, 281, 996–998.

Conway, J. (1985). *A Course in Functional Analysis*. New York: Springer-Verlag.

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines*. Cambridge, U.K.: Cambridge University Press.

Drucker, H., Wu, D., and Vapnik, V. (1999). "Support Vector Machines for Spam Categorization." *IEEE Transactions on Neural Networks*, 10(5), 1048–1054.

Fisher, R. (1936). "The Use of Multiple Measurements in Taxonomic Problems." *Annals of Eugenics*, 7, 179–188.

Gunn, S. R. (1998). "Support Vector Machines for Classification and Regression."

Technical report. Deptartment of Electronics and Computer Science, University of Southampton, UK.

Hall, P., Poskitt, D. S., and Presnell, B. (2001). "A Functional Data-Analytic Approach to Signal Discrimination." *Technometrics*, 43(1), 1–9.

James, G. and Hastie, J. (2001). "Functional Linear Discriminant Analysis for Irregularly Sampled Curves." *Journal of the Royal Statistical Society, Series B*, 63, 533–550.

Joachims, T. (1998). "Making Large-Scale Support Vector Machine Learning Practical." In *Advances in Kernel Methods: Support Vector Machines*, eds. B, Schölkopf, C, Burges, and A, Smola. Cambridge, MA: MIT Press.

——— (2000). "Estimating the Generalization Performance of a SVM Efficiently." In *Proceedings of ICML-00, 17th International Conference on Machine Learning*, ed. P, Langley, 431–438. San Francisco, CA: Morgan Kaufmann Publishers.

Lebedev, L., Vorovich, I., and Gladwell, G. (2002). *Functional Analysis*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Müller, H. G., Carey, J. R., Wu, D., Liedo, P., and Vaupel, J. W. (2001). "Reproductive Potential Predicts Longevity of Female Mediterranean Fruit Flies." In *Proceedings of the Royal Society Series B*, Vol. 268, 445–450.

Müller, H. G. and Stadtmüller, U. (2001). "Generalized Functional Linear Models." Unpublished manascript, http://anson.ucdavis.edu/∼mueller/preprints.html.

Platt, J. (1999). "Fast Training of Support Vector Machines using Sequential Minimal Optimization." In *Advances in Kernel Methods: Support Vector Machines*, eds. B, Schölkopf, C, Burges, and A, Smola, 185–208. Cambridge, MA: MIT Press.

Ramsay, J. and Silverman, B. (1997). *Functional Data Analysis*. New York: Springer.

——— (2002). *Functional Data Analysis - Methods and Case Studies*. New York: Springer.

Rynne, B. and Youngson, M. (2001). *Linear Functional Analysis*. London: Springer-Verlag.

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels - Support Vector Machines, Regulations, Optimization, and Beyond*. Cambridge, MA: MIT Press.

Smola, A. and Schölkopf, B. (1998). "A Tutorial on Support Vector Regression." Technical report. NeuroCOLT2, NC2-TR-1998-030.

Smola, A. J., Schölkopf, B., and Müller, K.-R. (1998). "The Connection between Regularization Operators and Support Vector Kernels." *Neural Networks*, 11(4), 637–649.

Trosset, M. (1999). "The Krigifer: A Procedure for Generating Pseudorandom Nonlinear Objective Functions for Computational Experimentation." Interim report 35. Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. New York: Springer Verlag.

——— (1998). *Statistical Learning Theory*. New York: Wiley.

Vapnik, V., Golowich, S., and Smola, A. (1997). "Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing." In *Advances in Neural Information Processing Systems,* volumn 9, eds. M, Mozer, M, Jordan, and T, Petsche, 281–287. Cambridge, MA: MIT Press.

Vogt, M. (2002). "SMO Algorithms for Support Vector Machines without Bias Term." Technical report. Institute of Automatic Control, Darmstadt University of Technology, Darmstadt, Germany.

Wahba, G. (1990). *Spline Models for Observational Data*. CBMS-NSF Regional Conference Series in Applied Mathematics 59, SIAM, Philadelphia, PA.

VITA

Ho-Jin Lee, son of Youngdeuk Lee and Hyunae Yoo, was born on December 10, 1971, in Kimjae, Korea. He received a Bachelor of Economics degree in statistics from Sung Kyun Kwan University in Seoul, Korea in 1996. He received a Master of Science degree in statistics from Texas A&M University in College Station, Texas, under the direction of P. Fred Dahm in 2002. He continued his studies in statistics under the direction of Tailen Hsing, and received a Doctor of Philosophy degree in statistics from Texas A&M University in August 2004. He is employed at Schering-Plough Research Institute as a research statistician. His permanent address is Woosung APT 108-301 Seongseong, Chunan Choongnam, Korea.