NONPARAMETRIC AND SEMIPARAMETRIC APPROACHES TO FUNCTIONAL DATA MODELING

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Wentian Huang

December 2021

NONPARAMETRIC AND SEMIPARAMETRIC APPROACHES TO FUNCTIONAL DATA MODELING

Wentian Huang, Ph.D.

Cornell University 2021

We propose original nonparametric and semiparametric approaches to model the relationship between a pair of variables (X, Y), where $X \in \mathcal{L}^2(\mathcal{T})$ is a square-integrable random function over a compact interval $\mathcal{T}$, and $Y \in \mathbb{R}$. Modeling of functional data greatly extends the nonparameteric approaches that have been widely used in the multivariate setting for both classifications and regressions. However, such approaches can be problematic due to the infinite dimensional nature of functional data and the so-called curse-of-dimensionality. Fortunately, functional data often lie in a low-dimensional subspace. Therefore, one can project the data onto a subspace of dimension J, e.g., the first J principal components, where J is a tuning parameter, and the model performance is sensitive to the choice of J.

Our work develops methods controlling the cut-off basis J with respect to the finite sample size n for functional data, covering both classifications and scalar-on-function regressions. A semiparametric Bayes classifier is developed using the copula structure to model dependency between the J projected scores. Furthermore, for functional local linear models, a class of multidimensional data-adaptive ridge penalties is built, and an algorithm of empirically estimating the first-order derivatives is developed. The methods prove to have strong prediction performance and dimension reduction strength when compared to other popular approaches through comprehensive simulation scenarios and

real-data examples. We also point out the methods' effective bandwidth size control, indicating their strength in finite-sample variance reduction. The estimators' asymptotic performances are also discussed, when $J \to \infty$ as $n \to \infty$.

BIOGRAPHICAL SKETCH

Wentian Huang earned her Bachelor of Science degree in Mathematics from University of Illinois at Urbana-Champaign in 2014. Her interest in statistics started to grow when taking a probability course in the sophomore year. After college she started her Ph.D. study at Cornell University, with concentration in nonparametric statistics and functional data modeling. In her personal life, she is interested in street photography and Russian literature.

To my family

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my advisor, Professor David Ruppert, for all his time and effort in working with me. Professor Ruppert has given me great guidance as well as freedom in my research. To me, he is the role model of a meticulous researcher and patient mentor. I benefit tremendously from his detailed suggestions in both the content and the writing of the thesis.

My gratitude also extends to my commitee members, Professor Giles Hooker and Professor David Matteson for their careful review of my proposal and dissertation. They taught some of the best courses of my Ph.D. study, which provided me useful tools in my research.

Finally, I'd like to thank my parents, who never got the opportunity to go to college, but now have two kids whose names can be followed by the suffix Ph.D. Their unconditional love supported me through all the years. My thanks also go to my older sister, who has always been my best friend and appreciates my quirkiness since I was young. And to my incoming nephew, who I hope would find the world interesting and worth the journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In recent decades, the fast growing computing technology has enabled large-scale data processing and storage, which fosters the rapid development of functional data analysis (FDA). Functional data are samples from a random functional variable $\mathcal{X}$ which takes values in infinite dimensional space ([25]. In particular, we discuss $\mathcal{X} \in \mathcal{L}^2(\mathcal{T})$ for a compact interval $\mathcal{T}$, where $\int_{\mathcal{T}} \mathcal{X}^2(t)\, dt < \infty$ for $t \in \mathcal{T}$. Applications of functional data can be seen in industries like chemometrics, econometrics, environment science etc.

Due to the infinite dimensional nature of functional data, many parametric multivariate methods appear to be inefficient or non-applicable to model random functions. We are interested in using nonparametric or at least semi-parametric models for functional data, which cast fewer assumptions on the distribution of functions, and therefore have wider applications. They prove to have strong performance in FDA.

Our work of nonparametric/semi-parametric functional data models cover both discussions of functional data classifications and scalar-on-function regressions. It has three parts as follows.

1.1 Semiparametric Functional Bayes Classification

Similar to multivariate classification, classification of functional data assigns different labels to data groups based on their distributions. However, as the data have infinite dimensions, there is no density to describe the distribution pat-

terns, and classic methods such as Bayes classification would fail here. [34] extended the linear discriminant analysis (LDA) to functional data (FLDA), including the case where the curves are partially observed. [33] proposed a functional version of the generalized linear model (FGLM), including functional logistic regression. Thereafter, the FGLM was further researched extensively. Aside from the FGLM, other classifiers have also been studied. [55] applied support vector machines (SVM) to classify infinite-dimensional data. [14] explored the classification of functional data based on data depth. [39] suggested a functional segmented discriminant analysis combining an LDA and an SVM, and [11] proposed a nonlinear aggregation classifier.

However, these methods distinguish groups by the differences between their functional means. They achieve satisfactory results when the location difference is the dominant feature distinguishing classes, but functional data provide more information than just group means. We propose new semiparametric Bayes classifiers. We project the functions onto the eigenfunctions of the pooled covariance function, that is, the covariance function marginalized over groups. These eigenfunctions can be estimated by applying a functional principal components analysis (fPCA) to the combined groups. The projections will not be independent or even uncorrelated, unless these common eigenfunctions are also the eigenfunctions of the group-specific covariance functions, an assumption not likely to hold in many situations. Nonparametric kernel density estimation is used to estimate the marginal densities of scores projected onto a low-dimensional subspace, and parametric copulas are incorporated to model dependency between the scores.

This classification avoids the restricted range of applications imposed by the

assumption of equal group-specific eigenfunctions. It also avoids the curse of dimensionality that a multivariate nonparametric density estimation would entail.

In addition, we discuss the different projection basis for building the classifiers, and extend the binary classification to multiclass cases. Two real-data examples and a comprehensive simulation are included to demonstrate the strong performance of this semiparametric classifier. Asymptotic results are also investigated, including conditions for 'perfection classification', meaning that the classifier achieves zero error when $n \to \infty$ ([18]).

1.2 Adaptive Ridge-Penalized Functional Local Linear Regression

Then, we research about the scalar-on-function regression where an unknown function, m, describes the relationship between a predictor function X in some Hilbert space and a real scalar Y. The model is $Y = m(X) + \epsilon$ where ϵ is random error. We assume an independent, identically distributed sample (X_i, Y_i), $i = 1, \ldots, n$.

Past work such as Cai et al. (2006 [8]) and Reiss and Ogden (2007 [54]) discussed estimation when m is linear, so that $m(X) = \langle X, \beta \rangle$, the inner product of X and an unknown coefficient function β. However, the linearity assumption often fails to hold.

Fortunately, nonparameteric methods that have been widely used in multivariate regression have been extended to functional predictors and have shown

strong performance there, especially functional polynomial regressions. We choose the functional local linear regression (FLLR), which balances between prediction accuracy and model complexity.

To implement local polynomial functional regression, one can project the data onto a subspace of dimension J, e.g., the first J principal components, where J is a tuning parameter. However, the estimator can be sensitive to the choice of J and, even with the best choice of J, the estimator will likely be improved with a roughness penalty.

To improve the FLLR estimator, we propose a data-adaptive ridge roughness penalty. The most general ridge penalty matrix is a $J \times J$ positive semidefinite matrix. Data-based selection of this type of penalty matrix with $J(J+1)/2$ free parameters can be difficult and can result in an unstable and inefficient estimator. Therefore, we propose a data-adaptive ridge penalization that utilizes a specific class of positive semidefinite diagonalizable matrices. As will be shown later, this structure with only J free parameters enables a quadratic programming search for optimal tuning parameters that minimize the estimated mean squared error (MSE) of prediction. Our method of penalization also accommodates a different roughness penalty level on each basis function and avoids the computational cost of multivariate cross validation as J increases.

Our original estimator has strong prediction performance in both simulations and real data examples, especially when the model is nonlinear. In addition, the method shows effective bandwidth size control for finite data samples, proving its strength in variance reduction. Asymptotic properties of the new estimator are derived, and a detailed implementation is provided, including a two-step bandwidth selection for estimating m and its functional derivative m'.

1.3 Functional Derivative Estimation

During the work of multidimensional ridge penalty for functional local linear models, we discover that the functional derivative of the regression model m at the function X, which we denote as m'_X, is important for model roughness penalization. Also, similar to its multivariate counterpart, the functional derivative provides a quantitative perception of the relationship between the change in the predictor curve X and the response Y. Properties such as the bounded linearity of m'_X can be found in Chapter 4 of Zeidler (1995 [68]).

However, there are only limited past works discussing about the estimation of the functional derivatives, many of which use a parametric linear framework. We propose a more generalized approach. A new nonparametric estimator is constructed for the functional derivatives, where an empirical bias of the estimator is calculated, and tuning parameters like cut-off basis J and bandwidth h are selected by minimizing the empirical mean squared error directly. This method is extended from Ruppert (1997 [56]) where the empirical bias is calculated for multivariate nonparametric regressions as well as density estimation in order to select optimal local bandwidths. We adjust the empirical bias from Ruppert (1991 [56]) according to the non-asymptotic bounds of estimated derivatives on functional data, and explore the specific behaviors of the estimator under the infinite dimensional setting. Advantage of this method is demonstrated through simulation study. In addition, this functional derivative estimator shows its practical strength through a comprehensive real world data analysis about the COVID-19 pandemic.

CHAPTER 2

COPULA-BASED FUNCTIONAL BAYES CLASSIFICATION WITH PRINCIPAL COMPONENTS AND PARTIAL LEAST SQUARES

2.1 Introduction

Functional classification, where the features are continuous functions on a compact interval, is receiving increasing interest in fields such as chemometrics, medicine, economics, and environmental science. [34] extended the linear discriminant analysis (LDA) to functional data (FLDA), including the case where the curves are partially observed. [33] proposed a functional version of the generalized linear model (FGLM), including functional logistic regression. Thereafter, the FGLM was further researched by, among others, [48], [40], [69], [46], and [61]. Aside from the FGLM, other classifiers have also been studied. [55] applied support vector machines (SVM) to classify infinite-dimensional data. [14] explored the classification of functional data based on data depth. [39] suggested a functional segmented discriminant analysis combining an LDA and an SVM, and [11] proposed a nonlinear aggregation classifier.

However, certain issues remain. Current methods, such as the FLDA, SVM, and functional centroid classifier ([18]), distinguish groups by the differences between their functional means. They achieve satisfactory results when the location difference is the dominant feature distinguishing classes, but functional data provide more information than just group means. For example, Fig. 2.1 from the example in Section 2.4.1 compares the mean and standard deviation functions of raw and smoothed fractional anisotropy (FA) measured along the corpus callosum (cca) of 141 subjects, 99 with multiple sclerosis (MS) and 42

without. The disparity between the group standard deviations in panel (c) provides additional information that can identify MS patients. As shown in Section 2.4.1, the LDA and centroid classifiers fail to capture this information, and have higher misclassification rates than the classifiers we propose.

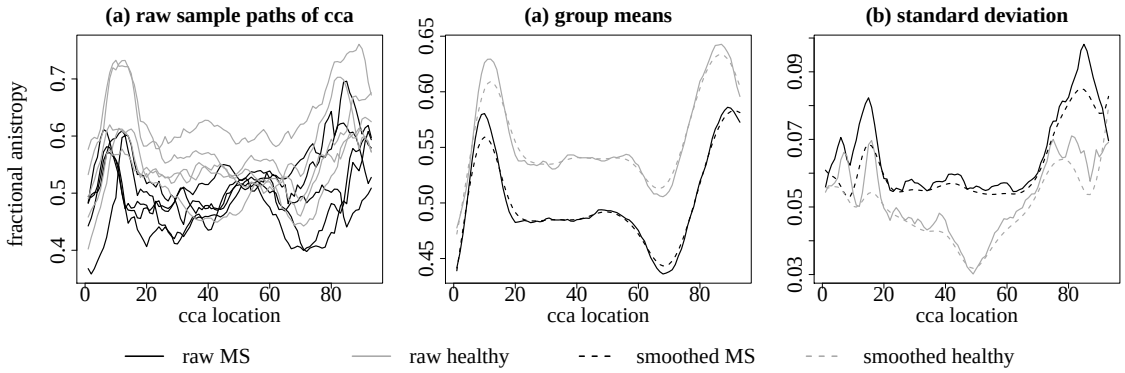

Figure 2.1: Panel (a) shows profiles of FA, five each of cases and controls, and panels (b) and (c) show the group means and standard deviations. Compared to the controls, the MS group has a lower mean and a higher standard deviation.

Both parametric and nonparametric methods have drawbacks in classifying functional data. Parametric models, such as linear and quadratic discriminant analysis, are popular in functional classification, especially because nonparametric methods are likely to encounter the curse of dimensionality. However, parametric methods can cast rigid assumptions on the class boundaries ([39]). Our interest is in methods that avoid stringent assumptions on the data. [15] proposed a nonparametric Bayes classifier, assuming that the subgroups share the same sets of eigenfunctions, and that the scores projected on them are independent. With these assumptions and the definition of the density of random functions proposed by [17], the joint densities of the truncated functional data can be estimated using a *univariate* kernel density estimation (KDE). The Bayes rules estimated this way avoid the curse of dimensionality, but require that the groups have equal sets of eigenfunctions and independent scores.

We propose new semiparametric Bayes classifiers. We project the functions onto the eigenfunctions of the pooled covariance function, that is, the covariance function marginalized over groups. These eigenfunctions can be estimated by applying a functional principal components analysis (fPCA) to the combined groups. The projections will not be independent or even uncorrelated, unless these common eigenfunctions are also the eigenfunctions of the group-specific covariance functions, an assumption not likely to hold in many situations. For instance, in Section 2.4 we discuss two real-data examples, and include a comparison of their group eigenfunctions in the supplementary materials. Both cases appear to violate the equal eigenfunction assumption. We estimate the marginal density of the projected scores using a univariate KDE, as in [15], and model the association between the scores using a parametric copula. Our semiparametric methodology avoids the restricted range of applications imposed by the assumption of equal group-specific eigenfunctions. It also avoids the curse of dimensionality that a multivariate nonparametric density estimation would entail.

In addition to the principal components (PC) basis, we also consider a partial least squares (PLS) projection basis. PLS has attracted recent attention owing to its effectiveness in prediction and classification problems with high-dimensional and functional data. [50] discuss a functional LDA combined with PLS. [18] mention the potential advantage of PLS scores in their functional centroid classifier, when the difference between the group means does not lie primarily in the space spanned by the first few eigenfunctions. We find that PLS scores can be more efficient than PC scores in capturing group mean differences.

This study contributes to the literature in two ways. In our numerical results,

the new method shows improved prediction accuracy and strength in dimension reduction, and extends the functional Bayes classification to multiclass classification. In the theoretical analysis, several new conditions are added for the functional data to achieve asymptotic optimality. These conditions are required because of the unequal group-specific eigenfunctions. Moreover, we propose asymptotic sparsity assumptions on the inverse of the copula correlations in our new method, following the design of [67] and [41] for high-dimensional data. We also build a new theorem that uses the special copula structure to achieve asymptotic perfect classification.

In Section 2.2, we introduce our model and the copula-based functional Bayes classifiers. Section 2.3 contains a comprehensive simulation study comparing our methods with existing classifiers on both binary and multiclass problems. Section 2.4 uses two real-data examples to show the strength of our classifiers in terms of accuracy and dimension reduction with respect to data size. In Section 2.5, we discuss the asymptotic properties of our classifiers. We also establish conditions for our classifiers to achieve perfect classification on data generated by Gaussian and non-Gaussian processes. Finally, in Section 2.6, we discuss future work, including extending the classification to the case where there are multiple functional predictors. Additional results and detailed proofs are provided in the supplementary materials.

2.2 Model Setup & Functional Bayes Classifiers with Copulas

2.2.1 Methodology

Suppose $(X_{i\cdot\cdot}, Y_i),\ i = 1, \ldots, n$ are independent and identically distributed (i.i.d.) from the joint distribution of (X, Y), where X is a square integrable function over some compact interval $\mathcal{T}$, that is, $X \in \mathcal{L}^2(\mathcal{T})$. Here $Y = 0, 1$ is an indicator of groups Π_0 and Π_1, respectively, and $\pi_k = P(Y = k)$. In addition, $X_{i\cdot k}$, for $i = 1, \ldots, n_k$ and $k = 0, 1$, denotes the ith sample curve of $X_{\cdot\cdot k} = (X|Y = k)$, and $n = \sum_{k=0,1} n_k$. Our goal is to classify a new observation, x.

Note that throughout the paper, we order the index of X by observation counts (i), joint basis (j), and group labels (k): for curves, $X_{i\cdot\cdot}$ denotes the ith observation of the random function X, and $X_{\cdot\cdot k}$ is the random function $X|Y = k$. Therefore, $X_{i\cdot k}$ is the ith sample curve of $X_{\cdot\cdot k}$. Furthermore, $X_{\cdot j\cdot}$ and $X_{\cdot jk}$ are random variables from projecting X and $X_{\cdot\cdot k}$, respectively, onto the jth joint basis function ψ_j, with X_{ijk} the ith observation of $X_{\cdot jk}$.

Dai et al.([15]) extended the Bayes classification from multivariate to functional data: a new curve x is classified into Π_1 if

$$Q(x) = \frac{P(Y = 1|X = x)}{P(Y = 0|X = x)} = \frac{\overline{f}_1(x)\pi_1}{\overline{f}_0(x)\pi_0} \approx \frac{f_1(x_1, \ldots, x_J)\pi_1}{f_0(x_1, \ldots, x_J)\pi_0} > 1, \tag{2.1}$$

where $\overline{f}_k$ is the density of $X_{\cdot\cdot k}$ and f_k is the joint density of the scores $X_{\cdot jk}$ on the basis ψ_j, for $1 \leq j \leq J$.

A key feature of the Bayes classification on functional data is that the classifiers vary with the choice of basis functions ψ_j and with the estimation of f_0, f_1. [15] built the original functional Bayes classifier (BC), upon two important as-

sumptions. First, the sets of the first J eigenfunctions, $\{\phi_1, \ldots, \phi_J\}$, of the covariance operators G_1 and G_0 of the two groups are equal. Here, $G_k(\phi_j)(t) = \int_{\mathcal{T}} G_k(s,t)\phi_j(s)ds = \lambda_{jk}\phi_j(t)$, $G_k(s,t) = \text{cov}\{X_{\cdot\cdot k}(s), X_{\cdot\cdot k}(t)\} = \sum_{j=1}^{\infty} \lambda_{jk}\phi_j(s)\phi_j(t)$, and λ_{jk} is the jth eigenvalue in group k. Second, letting $\psi_j = \phi_j$, for $1 \leq j \leq J$, the J projected scores $X_{\cdot jk} = \langle X_{\cdot\cdot k}, \phi_j \rangle$ are independent. Then, with f_{jk} as the marginal density of $X_{\cdot jk}$, the log ratio of $Q(x)$ in Eq.(2.1) becomes

$$\log Q(x) \approx \log Q_J(x) = \log\left(\frac{\pi_1}{\pi_0}\right) + \sum_{j=1}^{J} \log\left\{\frac{f_{j1}(x_j)}{f_{j0}(x_j)}\right\}. \tag{2.2}$$

A classifier that uses Eq.(2.2) avoids the curse of dimensionality and only needs to estimate the marginal densities, f_{jk}. However, as later simulations and examples show, its performance can degrade if the two aforementioned assumptions are not met. We propose new semiparametric Bayes classifiers based on copulas that do not require these two assumptions, and yet are free from the curse of dimensionality. The theoretical work in Section 2.5 proves that these classifiers maintain the advantages of BC over a wider range of data distributions, and are capable of perfect classification when $n \to \infty$ and $J \to \infty$.

2.2.2 Copula-Based Bayes Classifier with PC

Allowing for possibly unequal group eigenfunctions, the covariance function of group k is

$$G_k(s,t) = \text{cov}\left(X_{\cdot\cdot k}(s), X_{\cdot\cdot k}(t)\right) = \sum_{j=1}^{\infty} \lambda_{jk}\phi_{jk}(s)\phi_{jk}(t),\ k = 0, 1,$$

with $\phi_{1k}, \ldots, \phi_{Jk}$ as the eigenfunctions. For simplicity, we assume the group means are $E(X|Y=0) = 0$ and $E(X|Y=1) = \mu_d$. The joint covariance operator G then has the kernel $G(s,t) = \pi_1 G_1(s,t) + \pi_0 G_0(s,t) + \pi_1\pi_0\mu_d(s)\mu_d(t)$.

As later examples suggest, the unequal group eigenfunction case is common. To accommodate this case, we can project data from both groups onto the same basis functions. Therefore, we use the eigenfunctions $\phi_1, \ldots, \phi_J$ of G as the basis $\psi_1, \ldots, \psi_J$.

The joint density f_k,for $k = 0, 1$, in Eq.(2.1) allows for potential score correlation and tail dependency, which we use copulas to model. A copula is a multivariate cumulative distribution function (CDF) with univariate marginal distributions that are all uniform, and it characterizes only the dependency between the components; see, for example, [57]. Here, we extend its use to truncated scores of functional data.

Let $x_j = \langle x, \phi_j \rangle = \int_{\mathcal{T}} x(t)\phi_j(t)dt$ be the jth projected score of x. The copula function C_k describes the distribution of the first J scores in Π_k by

$$F_k\left(x_1, \ldots, x_J\right) = C_k\left\{F_{1k}(x_1), \ldots, F_{Jk}(x_J)\right\}, \tag{2.3}$$

$$f_k\left(x_1, \ldots, x_J\right) = c_k\left\{F_{1k}(x_1), \ldots, F_{Jk}(x_J)\right\} f_{1k}(x_1) \cdots f_{Jk}(x_J). \tag{2.4}$$

F_k in Eq.(2.3) is the joint CDF of $X_{\cdot 1k}, \ldots, X_{\cdot Jk}$, and C_k is the CDF of the uniformly distributed variables $F_{1k}(X_{\cdot 1k}), \ldots, F_{Jk}(X_{\cdot Jk})$, where F_{jk} is the univariate CDF of $X_{\cdot jk}$. In Eq.(2.4), the joint density f_k is decomposed into score marginal densities f_{jk} and the copula density c_k for the dependency between the projected scores. Our revised classifier is $\mathbb{1}\left\{\log Q_J^*(x) > 0\right\}$; that is, the new curve x belongs to Π_1 if

$$\log Q_J^*\left(x\right) = \log\left(\frac{\pi_1}{\pi_0}\right) + \sum_{j=1}^{J} \log\left\{\frac{f_{j1}(x_j)}{f_{j0}(x_j)}\right\} + \log\left\{\frac{c_1\{F_{11}(x_1), \ldots, F_{J1}(x_J)\}}{c_0\{F_{10}(x_1), \ldots, F_{J0}(x_J)\}}\right\} > 0. \tag{2.5}$$

We also consider situations in which Y has more than two classes. A more general procedure for multiclass classification is described in the supplementary

materials.

2.2.3 Choice of Copula and Correlation Estimator

There are a number of approaches to copula estimation. [26] studied the asymptotic properties of semiparametric estimation in copula models. [10] discussed semiparametric copula estimation to characterize the temporal dependence in time series data. [36] estimated the copula density nonparametrically using penalized splines, and [28] applied multivariate kernel density estimation to copulas.

To address the high dimensionality of functional data, we model the copula densities c_1 and c_0 parametrically, and use a kernel estimation for the univariate densities $f_{1k}, \ldots, f_{Jk}$, for $k = 0, 1$. We study the properties of Bayes classification using both Gaussian copulas and t-copulas, denoted by BCG and BCt, respectively. When c_k is modeled by a Gaussian copula in Eq.(2.4), $c_k(\cdot) = c_{G,k}(\cdot|\mathbf{\Omega}_{G,k})$, where $c_{G,k}$ is the Gaussian copula density with $J \times J$ correlation matrix $\mathbf{\Omega}_{G,k}$. When there is tail dependency between the scores, a t-copula is used: $c_k(\cdot) = c_{t,k}(\cdot|\mathbf{\Omega}_{t,k}, \nu_k)$, with $c_{t,k}$ the t-copula density, $\mathbf{\Omega}_{t,k}$ the correlation matrix, and ν_k the tail index.

There are several ways to estimate the correlation matrices $\mathbf{\Omega}_{G,k}$ or $\mathbf{\Omega}_{t,k}$. We use rank correlations, and specifically, Kendall's τ. Kendall's τ between the projected scores of $X_{\cdot\cdot k}$ on the jth and j'th basis is $\rho_\tau\left(X_{\cdot jk}, X_{\cdot j'k}\right) = E\left[\text{sign}\left\{\left(X^{(1)}_{\cdot jk} - X^{(2)}_{\cdot jk}\right)\left(X^{(1)}_{\cdot j'k} - X^{(2)}_{\cdot j'k}\right)\right\}\right]$, $\text{sign}(x) = \mathbb{1}\{x > 0\} - \mathbb{1}\{x < 0\}$, and $X^{(1)}_{\cdot\cdot k}$, $X^{(2)}_{\cdot\cdot k}$ are i.i.d. samples of $X_{\cdot\cdot k}$. The robustness of the rank correlation and its optimal asymptotic error rate are studied by [41].

A relationship exists between the (j, j')th entry of the copula correlation $\mathbf{\Omega}_k$ and Kendall's τ: $\mathbf{\Omega}_k^{jj'} = \sin\left(\frac{\pi}{2}\rho_\tau\left(X_{\cdot jk}, X_{\cdot j'k}\right)\right)$ for both Gaussian copulas and t-copulas ([37]; [38]; [57]). Then, $\mathbf{\Omega}_k^{jj'}$ is estimated by Kendall's τ as $\hat{\mathbf{\Omega}}_k^{jj'} = \sin\left(\frac{\pi}{2}\hat{\rho}_{\tau,k}^{jj'}\right)$, where

$$\hat{\rho}_{\tau,k}^{jj'} = \frac{2}{n_k\left(n_k-1\right)} \sum_{1\leq i\leq i'\leq n_k} \text{sign}\left\{\langle X_{i\cdot k} - X_{i'\cdot k}, \hat{\phi}_j\rangle\langle X_{i\cdot k} - X_{i'\cdot k}, \hat{\phi}_{j'}\rangle\right\}.$$

It is possible that $\hat{\mathbf{\Omega}}_k$ is not positive definite, but this problem is easily remedied ([57]). Another rank correlation, Spearman's ρ, is similar and is omitted here. In the supplementary material, we show that for Gaussian copulas, the difference between the log determinant of $\hat{\mathbf{\Omega}}_k$, as estimated, and that of $\mathbf{\Omega}_k$ is $Op\left(J\sqrt{(\log J)/n}\right)$.

Additionally for t-copulas with $\hat{\mathbf{\Omega}}_{t,k}$, we apply a pseudo-maximum likelihood to estimate the tail parameter $\nu_k > 0$ by maximizing the log copula density $\sum_{i=1}^{n_k} \log\left[c_{t,k}\left\{\hat{F}_{1k}\left(X_{i1k}\right), \ldots, \hat{F}_{Jk}\left(X_{iJk}\right)|\hat{\mathbf{\Omega}}_{t,k}, \nu_k\right\}\right]$, with $\hat{F}_{jk}\left(x\right) = \sum_{i=1}^{n_k} \mathbb{1}\left\{X_{ijk} \leq x\right\}/\left(n_k+1\right)$. [44] discuss the maximum pseudo-likelihood estimation of t-copulas, and apply it to model extreme co-movements of financial assets.

2.2.4 Marginal Density f_{jk} Estimation

We estimate the marginal density f_{jk} of the projected scores $X_{\cdot jk}$ using a kernel density estimation: $\hat{f}_{jk}\left(\hat{x}_j\right) = \frac{1}{n_k h_{jk}} \sum_{i=1}^{n_k} K\left(\frac{\langle x - X_{i\cdot k}, \hat{\phi}_j\rangle}{h_{jk}}\right)$, with K the standard Gaussian kernel, $\hat{\phi}_j$ the estimated jth joint eigenfunction, $h_{jk} = \hat{\sigma}_{jk}h$ the bandwidth for scores projected on $\hat{\phi}_j$ in group k, $\hat{\sigma}_{jk}$ as the estimated standard deviation of $\sigma_{jk} = \sqrt{\text{Var}\left(X_{\cdot jk}\right)}$, and $\hat{x}_j = \langle x, \hat{\phi}_j\rangle$. Then, $\log Q_J^*\left(x\right)$ in Eq.(2.5) is

estimated by

$$\log \hat{Q}_J^*(x) = \log\left(\frac{\hat{\pi}_1}{\hat{\pi}_0}\right) + \sum_{j=1}^{J} \log\left\{\frac{\hat{f}_{j1}(\hat{x}_j)}{\hat{f}_{j0}(\hat{x}_j)}\right\} + \log\left\{\frac{\hat{c}_1\{\hat{F}_{11}(\hat{x}_1), \ldots, \hat{F}_{J1}(\hat{x}_J)\}}{\hat{c}_0\{\hat{F}_{10}(\hat{x}_1), \ldots, \hat{F}_{J0}(\hat{x}_J)\}}\right\},$$

where $\hat{c}_k$ is the Gaussian copula or t-copula density with the estimated parameters, and $\hat{\pi}_k = n_k/n$. Proposition 1 in Section 2.5 shows that with an additional mild assumption, when the group eigenfunctions are unequal, $|\hat{f}_{jk}(\hat{x}_j) - f_{jk}(x_j)|$ is asymptotically bounded at the same rate as when the eigenfunctions are equal. Detailed proofs are included in supplementary materials.

2.2.5 Copula-Based Bayes Classifier with Partial Least Squares

An interesting alternative to using PCs is to use functional partial least squares (FPLS). FPLS finds directions that maximize the covariance between the projected X and Y scores, rather than focusing on the variation in X alone, as with PCA. As the algorithm in the supplementary materials describes, FPLS iteratively generates a weight function w_j at each step j, for $1 \leq j \leq J$, which solves $\max_{w_j \in \mathcal{L}^2(\mathcal{T})} \text{cov}^2\left\{Y^{j-1}, \langle X^{j-1}, w_j\rangle\right\}$, such that $\|w_j\| = 1$ and $\langle w_j, G(w_j')\rangle = 0$, for all $1 \leq j' \leq j-1$. Recall that G is the joint covariance operator of the random function X. Here, Y^{j-1} and X^{j-1} are the updated function X and the indicator Y at step $j-1$, respectively, and their corresponding sample values are denoted as Y_i^{j-1} and $X_{i\cdot}^{j-1}$, for $i = 1, \ldots, n$.

The algorithm gives the decomposition $X_{i\cdot}(t) = \sum_{j=1}^{J} s_{ij} P_j(t) + E_i(t)$, for $t \in \mathcal{T}$, where $\mathbf{s}_i = (s_{i1}, \ldots, s_{iJ})^T$ is the length J score vector, $P_j \in \mathcal{L}^2(\mathcal{T})$, for $1 \leq j \leq J$, are loading functions, and E_i is the residual. [50] investigated PLS in linear discriminant analysis (LDA), and defined score vectors $\mathbf{S}_j$ as eigenvectors of the product of the Escoufier's operators of X and Y ([19]). For our case,

the classifiers BCG and BCt now act on the PLS scores $\mathbf{s}_i = (s_{i1}, \ldots, s_{iJ})^T$ of each observation $X_{i\cdot}$. We refer to these classifiers as BCG-PLS and BCt-PLS, respectively.

The dominant PCA directions might only have large within-group variances and small between-group differences in means. Such directions will have little power to discriminate between groups. This problem can be fixed by FPLS. The advantages of FPLS have been discussed, for example, by [50] and [18]. The latter found that when the difference between the group means projected on the jth PC direction is large only for large j, their functional centroid classifier with PLS scores has lower misclassification rates than when using PCA scores. As later examples show, FPLS is especially effective in such situations.

2.3 Comparison of Classifiers using Simulated Data

2.3.1 Data Design

To set up the simulation, for simplicity, we use $\pi_1 = \pi_0 = 0.5$. By Karhunen–Loève expansions, the functions $X_{i\cdot k}$, for $i = 1, \ldots, n_k$, of group $k = 0, 1$ can be decomposed as $X_{i\cdot k} = \mu_k + \sum_{j=1}^{J} \sqrt{\lambda_{jk}} \xi_{ijk} \phi_{jk}$, where μ_k is the group mean, λ_{jk} is the jth eigenvalue in group k corresponding to eigenfunction ϕ_{jk}, and $\lambda_{1k} > \cdots > \lambda_{Jk}$. The variables ξ_{ijk} are distributed with $E(\xi_{ijk}) = 0$, $\text{var}(\xi_{ijk}) = 1$, and $\text{cov}(\xi_{ijk}, \xi_{ij'k}) = 0$, for $\forall j \neq j'$. The compact interval $\mathcal{T}$ is $[0, 1]$, and the functions $X_{i\cdot k}$ are observed at the equally spaced grid $t_1 = 0, t_2 = 1/50, \ldots, t_{51} = 1$, with i.i.d. Gaussian noise $\epsilon_{ik}(t)$ centered at zero and with standard deviation 0.5. The classifiers are implemented both with and

without pre-smoothing the data. Because they have similar performance, we report only the results using pre-smoothing. The total sample size is $n = 250$, with 100 training and 150 test cases. The number of eigenfunctions for curve generation is $J = 201$, double the size of the training data set, to imitate the infinite dimensions of the functional data. For each j, the bandwidth h_{jk} for KDE is selected by the direct plug-in method ([62]). Simulations are repeated $N = 1000$ times. The supplementary materials include additional results with increased training size.

The distribution of (X, Y) is determined by four factors: the eigenfunctions (whether common or group-specific), difference between group means, eigenvalues, and score distributions. The factors are varied according to a $2 \times 2 \times 2 \times 3$ full factorial design, described below. We adopt a four-letter system to label the 24 factor-level combinations, which we call "scenarios."

Factor 1: Eigenfunctions $\phi_{1k}, \ldots, \phi_{Jk}$ of group k: The first factor specifies the eigenfunctions of the covariance operators G_1 and G_0. When the two sets $\phi_{1k}, \ldots, \phi_{Jk}$, for $k = 0, 1$, are the same, let the common eigenfunctions be the Fourier basis on $\mathcal{T} = [0, 1]$, where $\phi_{1k}(t) = 1, \phi_{jk}(t) = \sqrt{2}\cos(j\pi t)$ or $\sqrt{2}\sin((j-1)\pi t)$, for $1 < j \leq 201$ even or odd.

When the two groups have unequal eigenfunctions, the group $k = 0$ uses the Fourier basis $\phi_{10}, \ldots, \phi_{J0}$ as above, but the group $k = 1$ has a Fourier basis rotated by iterative updating:

i) let the starting value of $\phi_{11}, \ldots, \phi_{J1}$ be the original Fourier basis functions, as above;

ii) at step (j, j'), where $1 \leq j \leq J-1$, $j' = j+1, \ldots, J$, the pair of

functions $(\phi^*_{j1}, \phi^*_{j'1})$ is generated by a Givens rotation of angle $\theta_{jj'}$ of the current pair $(\phi_{j1}, \phi_{j'1})$ such that $\phi^*_{j1}(t) = \cos(\theta_{jj'})\,\phi_{j1}(t) - \sin(\theta_{jj'})\,\phi_{j'1}(t)$, $\phi^*_{j'1}(t) = \sin(\theta_{jj'})\,\phi_{j1}(t) + \cos(\theta_{jj'})\,\phi_{j'1}(t)$.

iii) the rotation angle for each pair of (j, j') is $\theta_{jj'} = \frac{\pi}{3}(\lambda_{j0} + \lambda_{j'0})$, with $\lambda_{j0}, \lambda_{j'0}$ the jth and j'th eigenvalues, respectively, of group $k = 0$. Hence, the major eigenfunctions receive greater rotations, with the angles proportional to their eigenvalues;

iv) then, we update $\phi_{j1}, \phi_{j'1}$ with the new $\phi^*_{j1}, \phi^*_{j'1}$ and continue the rotations until each pair of (j, j'), with $1 \leq j \leq J-1$, $j' = j+1, \ldots, J$, is rotated.

The rotated Fourier basis of group $k = 1$ guarantees that both groups Π_1 and Π_0 span the same eigenspace and satisfy the null hypothesis of the test of equal eigenspaces developed by [4]. This test was used by [15] to check whether the two groups have the same eigenfunctions. However, having equal eigenspaces is a necessary, but not sufficient condition for having equal sets of eigenfunctions, as proved by the rotated basis. Because of the unequal eigenfunctions of the operators G_1 and G_0, the scores X_{ijk} are correlated, which can be modeled by the new copula-based classifiers.

We also tested other choices of the second set of eigenfunctions, including the Haar wavelet system on $\mathcal{L}^2([0,1])$. However, the results are similar, and so are omitted. We denote the scenario where Π_1 and Π_0 have equal eigenfunctions as S (same), and otherwise as R (rotated).

Factor 2: Difference, μ_d, Between the Group Means: The second factor, which is at two levels, S (same) and D (different), is the difference between the group means, $\mu_d = \mu_1 - \mu_0$. For simplicity, we let $\mu_0 = 0$, $\mu_1 = \mu_d$. Here, $\mu_d(t) = t$.

Factor 3: Eigenvalues $\lambda_{1k}, \ldots, \lambda_{Jk}$ of Group k: The third factor, at two levels labeled S and D, is whether the eigenvalues $\lambda_{1k}, \ldots, \lambda_{Jk}$ depend on k. We label the level where $\lambda_{j1} = \lambda_{j0} = 1/j^2$ as S, and that when $\lambda_{j1} = 1/j^3$ and $\lambda_{j0} = 1/j^2$ as D, for $1 \leq j \leq J$.

Factor 4: Distribution of the standardized scores ξ_{ijk}: The fourth factor, at three levels N (normal), T (tail dependence and skewness), and V (varied), is the distribution of ξ_{ijk}.

N: $\xi_{i1k}, \ldots, \xi_{iJk}$ have a Gaussian distribution $N(0,1)$ for both $k = 0$ and 1.

T: This level includes tail dependency by setting $\xi_{ijk} = (\delta_{ijk} - b)/\eta_{ik}$, where $\delta_{ijk} \sim \text{Exp}(\lambda^*), \lambda^* = 5\sqrt{3}/3, b = 1/\lambda^*$, and $\eta_{ik} \sim \chi^2(5)/5$, for all $j = 1, \ldots, J$. All δ_{ijk} and η_{ik} are mutually independent, whereas the scores ξ_{ijk} on each basis j are uncorrelated, but dependent, because they share the same denominator, η_{ik}. The scores are skewed in both groups.

V: In this level, the scores in the two groups have different types of distributions, with $\xi_{ij1} \sim N(0,1)$, and $\xi_{ij0} \sim \text{Exp}(1) - 1$. Simulation results of a different choice of the varied distributions of ξ_{ij1} and ξ_{ij0} are included in supplementary.

Table 2.1 lists all 24 scenarios used in the simulations:

	$\xi_{ijk} \sim$ N	$\xi_{ijk} \sim$ T	$\xi_{ijk} \sim$ V
$\mu_d = 0,\ \lambda_{j1} = \lambda_{j0}$	(R/S)SSN	(R/S)SST	(R/S)SSV
$\mu_d = 0,\ \lambda_{j1} \neq \lambda_{j0}$	(R/S)SDN	(R/S)SDT	(R/S)SDV
$\mu_d \neq 0,\ \lambda_{j1} = \lambda_{j0}$	(R/S)DSN	(R/S)DST	(R/S)DSV
$\mu_d \neq 0,\ \lambda_{j1} \neq \lambda_{j0}$	(R/S)DDN	(R/S)DDT	(R/S)DDV

Table 2.1: Simulation scenarios. The labels are ordered: eigenfunctions (R/S), group mean (S, D), eigenvalues (S, D), and ξ_{ijk} distributions (N, T, V). Note that in SSSN and SSST, functions from both groups have the same distribution. We simply include them to have a full factorial design.

2.3.2 Functional Classifiers

The classifiers used in this study are listed below. Five of them are Bayes classifiers, and the last three are non-Bayes. The methods proposed in this paper are described in (ii) - (iii).

(i) BC: the original Bayes classifier of [15], with the log density ratio given by Eq.(2.2). The scores are by projection onto PCs;

(ii) BCG, BCG-PLS: Bayes classifiers with a Gaussian copula to model correlation, using PC and PLS scores, respectively. The rank correlation used is Kendall's τ. Both the Gaussian copula and the t-copula densities can be implemented using the R package `copula` ([31]);

(iii) BCt, BCt-PLS: Bayes classifiers similar to (ii), but using a t-copula instead;

(iv) CEN: functional centroid classifier in [18], where observation x is classified to group $k = 1$ if $T(x) = (\langle x, \psi\rangle - \langle \mu_1, \psi\rangle)^2 - (\langle x, \psi\rangle - \langle \mu_0, \psi\rangle)^2 \leq 0$, with μ_1 and μ_0 the group means. Here, $\psi = \sum_{j=1}^{J^*} \lambda_j^{-1} \mu_j \phi_j$ is a function of the first J^* joint eigenfunctions ϕ_j, the corresponding eigenvalues λ_j, and $\mu_j = \langle \mu_1 - \mu_0, \phi_j\rangle$;

(v) PLSDA (PLS discriminant analysis): binary classifier using Fisher's linear discriminant rule, with FPLS as a dimension-reduction method. It is implemented in the R package `pls` ([47]);

(vi) Logistic regression: logistic regression on functional PCs, implemented by the R function `glm`. It is one of the functional generalized regressions discussed in [48].

In each simulation, J^* is selected using 10-fold cross validation on the training data. The candidate J values range from 1 to 30 (2 to 30 for classifiers using

copulas). The estimation of the joint eigenfunctions ϕ_j follows the discretization approach of the fPCA, as described in Chapter 8.4 of [51]. A similar discretization strategy is used for the PLS basis.

2.3.3 Classifier Performance

	BC	BCG	BCGPLS	BCt	BCtPLS	CEN	PLSDA	logistic	CV	Ratio (CV)
SSSN	0.502	0.502	0.500	0.500	0.501	0.502	0.501	0.500	0.501	0.23%
SSDN	**0.227**	0.244	0.345	0.258	0.443	0.464	0.495	0.466	0.232	2.43%
SDSN	0.347	0.351	0.361	0.351	0.363	**0.275**	0.304	0.279	0.291	5.88%
SDDN	**0.169**	0.173	0.303	0.175	0.327	0.231	0.262	0.234	0.173	2.64%
SSST	0.507	0.502	0.500	0.505	0.499	0.499	0.499	0.499	0.502	0.69%
SSDT	**0.438**	*0.441*	0.454	0.456	0.471	0.488	0.497	0.490	0.452	3.19%
SDST	0.188	0.183	0.270	0.184	0.311	**0.167**	0.234	*0.169*	0.170	1.96%
SDDT	0.166	0.161	0.237	0.160	0.296	**0.148**	0.233	*0.150*	0.152	2.59%
SSSV	**0.355**	0.361	0.484	0.363	0.493	0.476	0.481	0.489	0.363	2.20%
SSDV	**0.253**	0.270	0.373	0.276	0.430	0.455	0.477	0.462	0.257	1.78%
SDSV	**0.264**	0.275	0.401	0.276	0.408	0.279	0.315	0.283	0.273	3.27%
SDDV	**0.202**	0.209	0.309	0.207	0.313	0.236	0.280	0.238	0.210	3.95%
RSSN	0.327	**0.147**	0.183	**0.147**	0.180	0.494	0.497	0.485	0.151	2.67%
RSDN	0.252	**0.090**	0.140	0.093	0.164	0.489	0.500	0.482	0.093	2.93%
RDSN	0.287	**0.128**	0.154	**0.128**	0.152	0.327	0.333	0.329	0.131	2.71%
RDDN	0.208	**0.077**	0.112	*0.079*	0.128	0.287	0.300	0.288	0.080	3.44%
RSST	0.435	**0.354**	0.373	*0.357*	0.372	0.486	0.490	0.489	0.361	1.95%
RSDT	0.400	**0.326**	0.348	0.336	0.365	0.486	0.491	0.485	0.339	3.87%
RDST	0.178	**0.148**	0.248	0.154	0.261	0.174	0.252	0.175	0.156	5.80%
RDDT	0.166	**0.137**	0.217	0.142	0.255	0.159	0.249	0.158	0.147	7.68%
RSSV	0.266	**0.147**	0.202	*0.149*	0.204	0.472	0.481	0.475	0.150	1.71%
RSDV	0.233	**0.100**	0.143	0.105	0.157	0.465	0.475	0.469	0.104	3.85%
RDSV	0.241	**0.145**	0.183	*0.146*	0.191	0.332	0.349	0.337	0.148	2.28%
RDDV	0.238	**0.116**	0.157	0.120	0.167	0.299	0.325	0.300	0.121	3.97%

Table 2.2: Misclassification rates of eight classifiers on 24 scenarios, each an average from 1000 simulations. Lowest rates of each data case are in bold, and cases within margin of error (see text) of the lowest are in italics. The column labeled CV contains error rates of the classifier selected by cross-validation. Ratio(CV) is the percent difference from the best of the eight classifiers for that scenario. CV error rates are not included in the rankings that determine coloring. SSSN and SSST are in gray because there is actually no difference between groups in these scenarios, and, because $\pi_0 = \pi_1 = 1/2$, the true misclassification rate is 0.5.

Table 2.2 contains the average misclassification rates over 1000 simulations by each method on each scenario. In addition, for each simulation, we use 10-fold cross-validation to select the classifier with the best performance on the

training data among the eight classifiers in Section 2.3.2. The average misclassification rates of the CV-selected classifier are listed in the CV column. The column Ratio(CV) contains the percentage difference between the CV-selected (CV) and the best (opt) classifier: Ratio(CV) $= \{\text{err(CV)} - \text{err(opt)}\}/\text{err(opt)} \times 100\%$. For each scenario, the lowest error rates of the eight classifiers are in bold. We label those within the optimal case's margin of error (MOE) for each data scenario γ in italics: $\text{MOE}_\gamma = 1.96 \times \sigma^*_\gamma/\sqrt{1000}$, where σ^*_γ is the sample standard deviation of the best classifier's (at scenario γ) error rates from 1000 simulations. The simulations enable a comprehensive understanding of the classifiers' behaviors, which we now discuss.

– *Equal versus Unequal Eigenfunctions*. A comparison between the top and bottom half of Table 2.2 demonstrates the strength of our copula-based classifiers, especially on unequal eigenfunctions (bottom half). By its nature, BC has strong performance when the two groups have the same set of eigenfunctions and the scores ξ_{ijk} are mutually independent, for example, in SSDN and SSDV. However, when the data have a more complicated structure, such as score tail dependency and location difference, CEN and logistic obtain better results (SDST, SDDT). Note that in every case with equal eigenfunctions, BCG/BCt are always the ones with rates closest to those of BC.

  On the other hand, when the group eigenfunctions are different, BC and the three non-Bayes classifiers fail to outperform BCG/BCt in any scenario, even though the group eigenspaces remain equal. BCG maintains its robust performance of lowest error rates throughout all cases. BCt is not far behind, and falls into BCG's MOE 50% of the time as labeled.

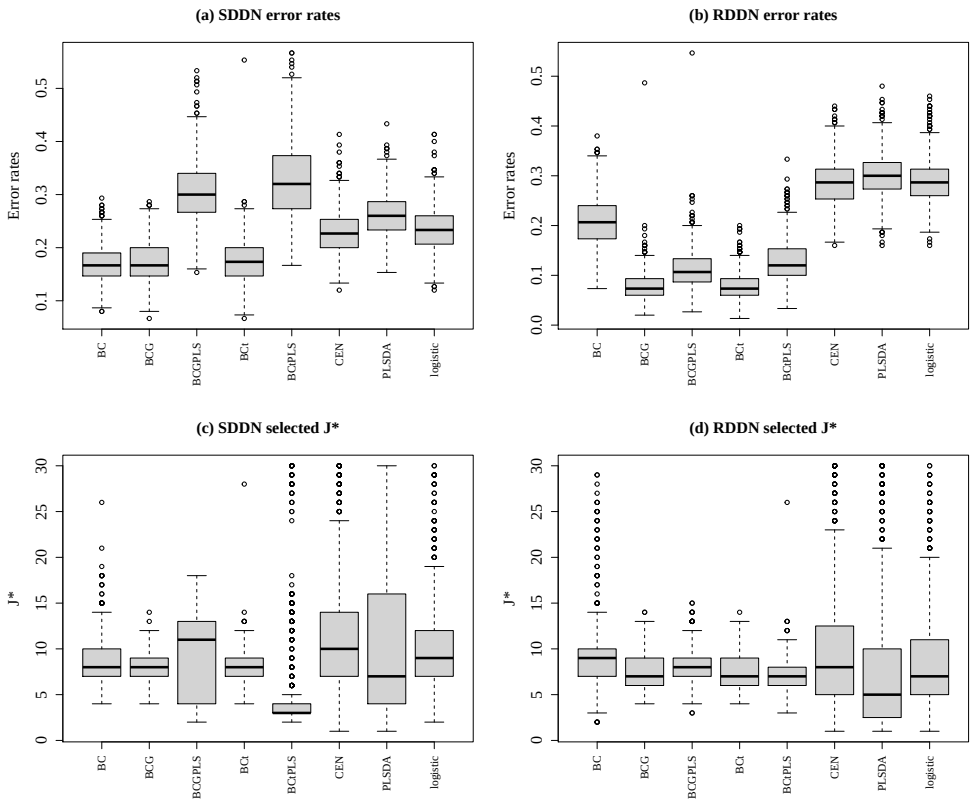

Figure 2.2: Part (a) and (b) are box plots of the error rates by the eight classifiers in scenarios SDDN and RDDN. The bottom two plots (c) and (d) are box plots of cross-validated J^* in each simulation.

Fig. 2.2 compares the misclassification rates and the corresponding J^* selected in each of the 1000 simulations at two scenarios, SDDN and RDDN. These two scenarios differ only in their eigenfunction setting. In Plot (a), where the groups have equal eigenfunctions, BC, BCG, and BCt show similar behaviors in classification. In Plot (b), where the group eigenfunctions differ, BCG and BCt have the lowest error rates and variation, followed by BCG-PLS and BCt-PLS. In Plots (c) and (d), we find that BCG and BCt are the only classifiers that have a stable choice of optimal J^*: both methods choose $J^* < 10$ more than 75% of the time with few outliers, regardless of whether the group eigenfunctions are equal or not.

– *Difference between the group means.* Under the equal eigenfunction setting,

non-Bayes classifiers such as CEN and the logistic regression are naturally sensitive to a location difference, especially when other factors are kept the same; see for example, SDSN, SDST. However, in the bottom half of Table 2.2, where the group eigenfunctions differ, BCG shows the strongest performance in all cases, with BCt a close second.

In this table, the PC-based methods BCG and BCt show an advantage over their PLS counterparts in scenarios with a location difference. That is because μ_d is effectively captured by PCs. In Section 2.3.4, when the new μ_d has nonzero projections only on the last several bases, PLS-based classifiers can do a better job than other methods in distinguishing such a difference, as mentioned in [18]. This phenomenon is also discussed in Section 2.4.

– *Difference in group eigenvalues and score distributions*. In general, we find that the marginal densities of the scores and their eigenvalues have similar effects on the classifiers' performance. They contribute to the difference of the functional distributions in each group, which the three non-Bayes methods (CEN, PLSDA, logistic) fail to detect. For all scenarios in Table 2.2 without a location difference, CEN, PLSDA, and the logistic regression all show very poor performance, with error rates close to 50%.

The two right-most columns in Table 2.2 show that the CV-selected method achieves comparable performance to the optimal result of each scenario. This demonstrates the stability and strength of our copula-based Bayes classifiers, especially under the unequal eigenfunction setting. The supplementary materials report the correlations between the first 10 scores in the scenarios RSDN and RSDT, respectively. These high correlations are consistent with the strong per-

formance of the copula-based classifiers in the scenarios where the two groups have different eigenfunctions.

2.3.4 Multiclass Classification Performance

We also investigate the performance of the aforementioned methods in terms of classifying data into more than two labels, because the group eigenfunctions from multiple different classes are more likely to be unequal, making it increasingly necessary to consider the dependency of the scores on the joint basis.

We now denote the group labels as $Y = k$, for $k = 0, 1, 2$, and set up the multiclass scenarios following the design in Section 2.3.1. The first column in Table 2.3 lists the 12 scenarios considered. The first letter M labels unequal group eigenfunctions: when $Y = 0$ and 1, the group eigenfunctions are the Fourier basis and its rotated counterpart, respectively, as described in type R of Factor 1 for binary data; when $Y = 2$, the group basis is again the rotated Fourier functions on $\mathcal{T} = [0, 1]$, but the rotation angle factor used in iii) of Factor 1 in Section 2.3.1 is now $\pi/4$ instead of $\pi/3$. We omit cases of equal group eigenfunctions, because similar results can be found in the binary setup, and the likelihood of an unequal basis increases as the levels of Y increase.

The second letter S or D again denotes equal group means or not, respectively. When the group means μ_k are unequal (labeled D), we set $\mu_0 = 0$, μ_1 is the identity function used previously, and $\mu_2 = \sum_{j=192}^{201} \phi_{j0}$. The function μ_2 follows a similar design to that of [18], where the group mean only has nonzero weights on the last three of 40 eigenfunctions. We assign the nonzero weights to the last 10 of the 201 bases.

Similarly, S or D in the third position represents the same or different group eigenvalues, respectively. When the group eigenvalues are equal, $\lambda_{jk} = 10/j^2$ for all k; otherwise, $\lambda_{jk} = 10/j^2, 10/j^3, 10/j$, respectively, for $k = 0, 1, 2$, for $j \geq 1$. Finally, the last letter inherits the design from Factor 4 of Section 2.3.1 to describe the standardized score distribution patterns: similarly to the binary case, N and T denote the Gaussian and skewed distributions, respectively, for all three levels, while for V, we define the scores ϵ_{ijk} to follow a standard Gaussian, centered exponential with rate one, or skewed distribution in T, for $k = 0, 1, 2$ respectively.

The other setup details of the noise, data pre-smoothing, and bandwidth selection are all similar to Section 2.3.1 for binary data. For each simulation, we have 100 training and 150 test cases. The optimal cut-off J^* is selected using cross-validation from $J \leq 10$. Table 2.3 presents the misclassification rates from 1000 Monte Carlo repetitions by seven of the eight classifiers in Section 2.3.2. Note that functional centroid classifier is not applicable to multiclass data, and thus is excluded here. As in the binary case, the supplementary material includes additional results with an increased training size and a different set of score distributions (V).

Table 2.3 indicates that for data of multiple labels, the behaviors of the seven classifiers follow a similar pattern to that of the binary case when the group eigenfunctions are unequal. In particular, BCt shows strength under increased data complexity, followed closely by BCG. BCG-PLS/BCt-PLS also prove their advantage in detecting location differences on minor basis functions in MDSN. Although they fail to outperform their PC-based counterparts under more complicated scenarios such as MDST and MDSV, we believe this is because the

	BC	BCG	BCGPLS	BCt	BCtPLS	PLSDA	logistic	CV	Ratio(CV)
MSSN	0.520	**0.325**	0.392	*0.327*	0.392	0.641	0.637	0.328	0.89%
MDSN	0.356	0.247	*0.237*	0.245	**0.235**	0.446	0.427	0.226	-3.88%
MSDN	0.213	*0.169*	0.281	**0.168**	0.310	0.636	0.618	0.173	3.00%
MDDN	0.194	*0.156*	0.272	**0.156**	0.295	0.540	0.509	0.157	1.11%
MSST	0.560	**0.450**	0.503	**0.450**	0.492	0.635	0.638	0.456	1.25%
MDST	0.343	**0.286**	0.303	**0.286**	0.333	0.424	0.364	0.284	-0.72%
MSDT	0.449	*0.399*	0.444	**0.397**	0.467	0.624	0.616	0.401	0.95%
MDDT	0.342	0.297	0.355	**0.287**	0.403	0.483	0.401	0.293	2.38%
MSSV	0.325	**0.259**	0.394	*0.261*	0.475	0.633	0.615	0.264	2.23%
MDSV	0.288	*0.237*	0.356	**0.234**	0.433	0.436	0.399	0.241	2.93%
MSDV	0.385	0.314	0.427	**0.302**	0.435	0.631	0.627	0.311	3.00%
MDDV	0.272	*0.223*	0.322	**0.219**	0.340	0.475	0.434	0.224	2.18%

Table 2.3: Misclassification rates averaged over 1000 simulations of the seven classifiers on 12 multiclass data scenarios. Best case in each scenario is in bold, and cases within margin of error of the lowest are in italic. $P(Y = k) = 1/3$, for $k = 0, 1, 2$, so the true misclassification rate of any method is approximately 0.667.

group means are not the only dominant difference in these two data cases.

Tables 2.2 and 2.3 give us clear guidelines that deciding whether or not to use copulas in a classification makes a more significant impact on the outcome than the type of copulas, because both BCG and BCt present competitive performance. The tables also reveal the strength of copula-based methods in dimension reduction. Classifiers using copulas are able to achieve high accuracy with small cut-off J^*, which indicates their advantage in small samples. In addition, in general, PCs are preferable to PLS, owing to their robustness and simplicity of implementation. BCG-PLS and BCt-PLS should be considered when the group mean difference is significant and located at minor eigenfunctions, which we discuss further in the real-data examples.

2.4 Real-Data Examples

In this section, we use two real-data examples to illustrate the strength of our new method in terms of classification and dimension reduction with respect to the data size n.

2.4.1 Classification of Multiple Sclerosis Patients

Our first example explores the classification of multiple sclerosis (MS) cases based on FA profiles of the cca tract. FA is the degree of anisotropy of water diffusion along a tract, and is measured by diffusion tensor imaging (DTI). Outside the brain, water diffusion is isotropic ([29]). MS is an autoimmune disease leading to lesions in white matter tracts such as the cca. These lesions decrease FA.

The DTI data set in the R package `refund` ([**?**]) contains FA profiles at 93 locations on the cca of 142 subjects. The data were collected at Johns Hopkins University and the Kennedy–Krieger Institute. The numbers of visits per subject range from one to eight, but we used the 142 FA curves from the first visits only. One subject with partially missing FA data was removed. Among the 141 subjects, 42 are healthy ($k = 0$) and 99 were diagnosed with MS ($k = 1$). We use local linear regression for data pre-smoothing. To determine the optimal number of dimensions J^* for each method, we use cross-validation with maximal $J = 30$. The misclassification rates from using 10-fold cross-validation were recorded for 1000 repetitions.

As discussed in Section 2.1, Panel (a) in Fig. 2.1 plots 5 FA profiles from each

group, and panels (b) and (c) display the group means and standard deviations of the cases and controls, using raw and pre-smoothed data. Compared with the controls, MS patients have lower mean FA values and greater variability. We see that smoothing removes some noise.

Method	BC	BCG	BCGPLS	BCt	BCtPLS	CEN	PLSDA	logistic
Error Rate	0.228	0.199	0.211	**0.192**	0.211	0.264	0.219	0.216

Table 2.4: Average misclassification rates of eight functional classifiers by 1000 repetitions of 10-fold CV. BCt has the best performance. The best case is in bold.

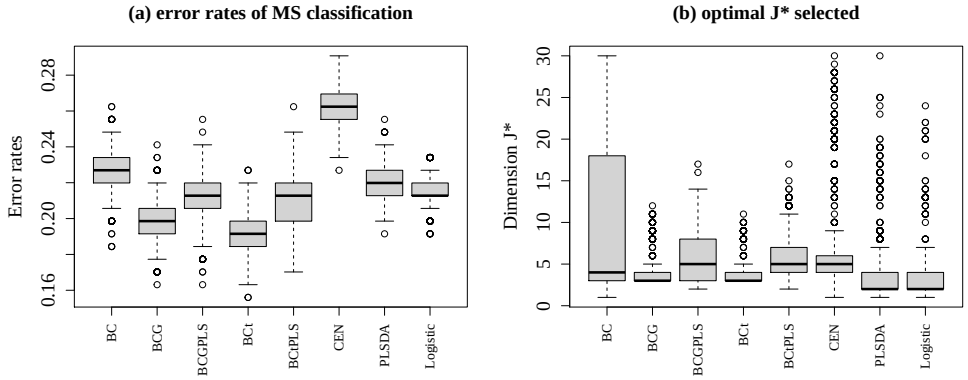

Figure 2.3: Box plots of misclassification rates and optimal number of components J^* in the MS study over 1000 repetitions of 10-fold cross-validation. BCt achieves the lowest average error rate, while requiring a very small number of components ($J^* < 5$) with lowest variation.

As shown in Table 2.4 and Part (a) of Fig. 2.3, BCt achieves the lowest error rate at 0.192, with a margin of error 0.0007. The rates of the other methods fail to fall into this range, and are all significantly higher than that of BCt. In fact, the third quartile for BCt is below the first quartile of all other methods, except BCG. Part (b) is a box plot of cross-validated J^* during each simulation for all classifiers. Here, BCt and BCG achieve the lowest error rates, with a minimal number of dimensions. In addition, compared with methods such as CEN, PLSDA, or logistic regression, their choice of optimal J^* is very stable, with the smallest variation and few outliers. In contrast, BC is prone to employing a

large number of components in classification. This tendency can be found in other examples too.

In the supplementary materials, we compare the loadings, score distributions, and group eigenfunctions between using PC and PLS. The difference explains why PC is a better choice for this example. Note that it is not our intent to develop DTI as a technique for diagnosing MS. DTI is too expensive and time-consuming for that purpose. Instead, we are looking for differences in FA between cases and controls, because these could inform researchers about the nature of the disease. We have found clear differences between cases and controls in the mean and variance of FA. The strong positive correlation between the second and the third PC scores in the healthy cases (Spearman's ρ at 0.525 and an adjusted p-value 2×10^{-2}) is diminished in the MS group. BCt and BCG are best able to use a compact model to capture subtle differences, such as correlations.

2.4.2 Particulate Matter (PM) Emission of Heavy-Duty Trucks

As a second example, we investigate the relationship between the movement patterns of heavy-duty trucks and particulate matter (PM) emissions. We use the data in [45], originally extracted from the Coordinating Research Council E55/59 emissions inventory program documentary ([12]). The data set contains 108 records of truck speed in miles/hour over 90-second intervals, and the logarithms of their PM emission in grams (log PM), captured by 70 mm filters.

We dichotomize log PM. The 41 of 108 cases with log PM above average are called high emission ($k = 1$), and the other cases are low emission ($k = 0$). We

classify log PM level using the 90-second velocity profiles. The misclassification rates are estimated using 10-fold cross-validation, repeated 1000 times.

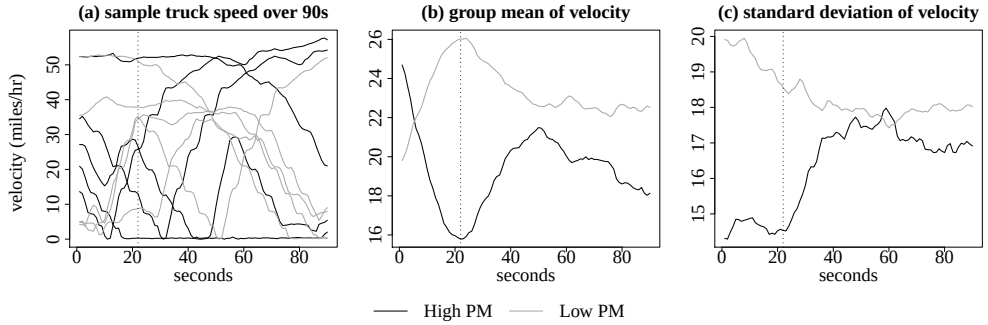

Figure 2.4: Plots of five sample paths in each PM group, as well as group mean and standard deviation of truck velocity data. On average, trucks in high PM group have lowest speed at 22 seconds, marked with a dashed line on each plot.

As Fig. 2.4 shows, during the first 20 seconds, vehicles in the high PM group, on average, decelerate to a minimum speed, whereas the low PM group tends to speed up. The high PM group also has much lower variation than the low PM group.

	BC	BCG	BCGPLS	BCt	BCtPLS	CEN	PLSDA	logistic
Error rate	0.285	0.280	**0.207**	0.280	**0.207**	0.278	0.256	0.228

Table 2.5: Average misclassification rates of eight functional classifiers by 1000 repetitions of 10-fold cross-validation. BCt-PLS and BCG-PLS have the best performance. The best cases are in bold.

From Fig. 2.5 and Table 2.5, BCG-PLS and BCt-PLS have the lowest misclassification rates. The third quartiles of their error rates are below the first quartiles of the other classifiers, except for the logistic regression. In addition, both methods keep the classification model compact by requiring small J^* with low variation. BC and the three methods on the right of plot (b) of Fig. 2.5 again demand more components with bigger variation in classification. In the supplementary materials, we include additional results for both data examples to validate their different choices of PC- and PLS-based classifiers.

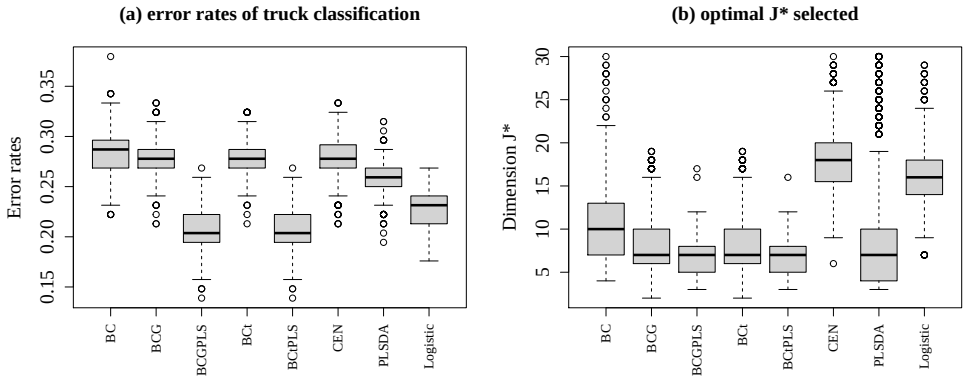

Figure 2.5: Box plots of misclassification rates and optimal number of components J^* in the truck emission case over 1000 repetitions of 10-fold cross-validation. BCt-PLS and BCG-PLS achieve the lowest average error rate with J^* concentrated around 7.

2.5 Theoretical Asymptotic Properties

An interesting feature of functional classifiers is asymptotic perfect classification. That is, under certain conditions, the error rate goes to zero as $J \to \infty$, owing to the infinite-dimensional nature of functional data ([18]). [15] discussed the perfect classification by BC under equal group eigenfunctions. In this section, we prove that when the group eigenfunctions differ, perfect classification is retained by our classifier $\mathbb{1}\{\log Q_J^*(X) > 0\}$ for both Gaussian and non-Gaussian processes. The scores $X_{\cdot jk}$, for $1 \leq j \leq J$, in this section are all projected onto joint eigenfunctions $\phi_1, \ldots, \phi_J$.

We first show that $\log Q_J^*(X)$ and the estimated $\log \hat{Q}_J^*(X)$ are asymptotically equivalent under mild conditions. Then, the behavior of the Bayes classifier $\mathbb{1}\{\log Q_J^*(X) > 0\}$ is studied in two settings: first, when the random function $X_{\cdot\cdot k}$ is a Gaussian process for both $k = 0, 1$; and second, the more general case, when X is non-Gaussian, but its projected scores are meta-Gaussian distributed in each group. For simplicity, we assume here that $\pi_1 = \pi_0$.

2.5.1 Asymptotic equivalence of $\log \hat{Q}_J^*(X)$ and $\log Q_J^*(X)$

We first list several assumptions, which help establish the asymptotic equivalence of both the marginal and the copula density components of $\log \hat{Q}_J^*(X)$ and $\log Q_J^*(X)$.

Assumption A1. *For all* $C > 0$ *and some* $\delta > 0$: $\sup_{t\in\mathcal{T}} E\{|X(t)|^C\} < \infty$, $\sup_{s,t\in\mathcal{T}:s\neq t} E[\{|s-t|^{-\delta}|X(s)-X(t)|\}^C] < \infty$.

Assumption A2. *For integers* $r \geq 1$, $\lambda_j^{-r} E[\int_{\mathcal{T}}\{X - E(X)\}\phi_j]^{2r}$ *is bounded uniformly in* j.

Assumption A3. *There are no ties among the eigenvalues* $\{\lambda_j\}_{j=1}^{\infty}$.

Assumption A4. *The density* g_j *of the* j*th standardized score* $\langle X - E(X), \phi_j\rangle/\sqrt{\lambda_j}$ *is bounded and has a bounded derivative; for some* $\delta > 0$, $h = h(n) = O(n^{-\delta})$ *and* $n^{1-\delta}h^3$ *is bounded away from zero as* $n \to \infty$. *The ratio* $f_{j1}(X_{\cdot j})/f_{j0}(X_{\cdot j})$ *is atomless for all* $j \geq 1$.

For all $c > 0$, let $\mathcal{S}(c) = \{x \in \mathcal{L}^2(\mathcal{T}) : \|x\| \leq c\}$. Assumptions A1–A4 are from [17], adapted here to bound the difference $D_{jk}(x_j) = \hat{g}_{jk}(\hat{x}_j) - \bar{g}_{jk}(x_j)$ s.t. $\sup_{x\in\mathcal{S}(c)} |D_{jk}(x_j)| = op\{(nh)^{-1/2}\}$. We let $\hat{g}_{jk}(\hat{x}_j) = 1/(n_k h)\sum_{i=1}^{n_k} K\left\{\langle X_{i\cdot k} - x, \hat{\phi}_j\rangle/(\hat{\sigma}_{jk}h)\right\}$ be the estimated density of the standardized scores of group k on basis $\hat{\phi}_j$, with $\bar{g}_{jk}(x_j)$ using ϕ_j and σ_{jk}. In addition, the following assumption is added for $D_{jk}(x_j)$, for both $k = 0, 1$:

Assumption A5. $\sup_{x\in\mathcal{S}(c)} |\hat{\pi}_k D_{jk}(x_j)/(\hat{\pi}_0 D_{j0}(x_j) + \hat{\pi}_1 D_{j1}(x_j))| = Op\left(1 + \sqrt{\frac{\log n}{nh^3}}\right)$.

We use A5 to give a mild bound simply to avoid the case where the magnitudes of both $D_{jk}(x_j)$, for $k = 0, 1$, are too large and close, but with opposite

signs. A5 guarantees that the difference between the estimated marginal density $\hat{f}_{jk}(\hat{x}_j)$ and $f_{jk}(x_j)$ is able to be bounded by the same rate as when the group eigenfunctions are equal. However, this is not a necessary condition for the asymptotic equivalence of $\log \hat{Q}_J^*(X)$ and $\log Q_J^*(X)$, and we can certainly relax its bound for Theorem 1 below.

Then, $\hat{f}_{jk}(\hat{x}_j) = (1/\hat{\sigma}_{jk})\,\hat{g}_{jk}(\hat{x}_j)$, and we have Proposition 1 (see the supplementary for the proof):

Proposition 1. *Under Assumptions A1–A5, when the group eigenfunctions are unequal, the estimated marginal density $\hat{f}_{jk}$ using scores $\langle X_{i\cdot k}, \hat{\phi}_j \rangle$ achieves an asymptotic error bound: $\sup_{x \in \mathcal{S}(c)} |\hat{f}_{jk}(\hat{x}_j) - f_{jk}(x_j)| = Op\left\{ h + \sqrt{\frac{\log n}{nh}} \right\}$, where the rate is the same as in [15], where the group eigenfunctions are equal.*

Assumption A6. *The CDFs F_{jk} of scores $X_{\cdot jk}$ are continuous and strictly increasing, with correspondent marginal densities f_{jk} continuous and bounded. In addition, f_{jk} are bounded away from zero on any compact interval within their supports.*

A6 ensures that the scores $X_{\cdot jk}$ and their monotonic transformations are atomless; this also follows Condition 5 in [15].

Then, in addition to the marginal densities, we establish the equivalence of $\mathbf{\Omega}_k^{-1}$ and $\hat{\mathbf{\Omega}}_k^{-1}$ in $\log Q_J^*(X)$ and $\log \hat{Q}_J^*(X)$, respectively, as $n \to \infty$. As mentioned in Section 2.2.3, we calculate $\hat{\mathbf{\Omega}}_k$ using rank correlations. In addition, when J is large, the inverse of $\hat{\mathbf{\Omega}}_k$ can be estimated using the graphical Dantzig selector ([67]), which solves the matrix inverse by connecting the entries of the inverse correlation matrix to a multivariate linear regression, and exploits the sparsity of the inverse matrices ([67]). [41] provided a q-norm Op bound of the difference between the inverse Gaussian copula matrix and its estimation by the

Dantzig estimator for high-dimensional problems, and is extended here for the difference between $\mathbf{\Omega}_k^{-1}$ and $\hat{\mathbf{\Omega}}_k^{-1}$.

Our sparsity assumptions on the inverse correlation matrices follow the design of [67] and [41]: let $\mathbf{\Omega}_k$ belong to the class of matrices $\mathcal{C}(\kappa, \tau, M, J) := \{\mathbf{\Omega}^{J\times J} : \mathbf{\Omega} \succ \mathbf{0}, \text{diag}(\mathbf{\Omega}) = \mathbf{1}, \|\mathbf{\Omega}^{-1}\|_1 \leq \kappa, \frac{1}{\tau} \leq \lambda_{\min}(\mathbf{\Omega}) \leq \lambda_{\max}(\mathbf{\Omega}) \leq \tau, \text{deg}(\mathbf{\Omega}^{-1}) \leq M\}$, where $\kappa, \tau \geq 1$ are constants determining the tuning parameter in the graphical Dantzig selector, and the parameter M bounding $\text{deg}(\mathbf{\Omega}^{-1}) = \max_{1\leq j\leq J} \sum_{j'=1}^{J} I(\mathbf{\Omega}_{jj'}^{-1} \neq 0)$ is dependent on J. Assuming these sparsity conditions, we have the following theorem.

Theorem 1. *Under A1–A6, $\forall \epsilon > 0$, as $n \to \infty$, there exists a sequence $J(n, \epsilon, M) \to \infty$, and a set S dependent on $J(n, \epsilon, M)$, $P(S) \geq 1 - \epsilon$, such that*

$$P\left(S \cap \left\{\mathbb{1}\left\{\log \hat{Q}_J^*(X) \geq 0\right\} \neq \mathbb{1}\left\{\log Q_J^*(X) \geq 0\right\}\right\}\right) \to 0,$$

provided that $MJ\sqrt{\log J} = o(\sqrt{n})$.

Theorem 1 proves that under unequal group eigenfunctions, $\log \hat{Q}_J^*(X)$ using copulas retains the property in Theorem A1 of [15] for the estimated Bayes classifiers with equal group eigenfunctions and independent scores: as $n \to \infty$, $\log \hat{Q}_J^*(X)$ gets arbitrarily close to the true Bayes classifier $\log Q_J^*(X)$, which enables us to discuss the performance of our method using the properties of the true Bayes classifier.

2.5.2 Perfect classification when X is a Gaussian process in both groups

Let $X_{\cdot k}$ be a centered Gaussian process such that $X_{\cdot k} = \sum_{q=1}^{\infty} \sqrt{\lambda_{qk}} \xi_{qk} \phi_{qk}$, with $\xi_{qk} \sim N(0,1)$, for $k = 0, 1$. We denote the $J \times J$ covariance matrix of scores $X_{\cdot jk}$, for $1 \leq j \leq J$, as $\mathbf{R}_k$, where its (j, j')th entry is equal to $\text{cov}\,(X_{\cdot jk}, X_{\cdot j'k}) = \sum_{q=1}^{\infty} \lambda_{qk} \langle \phi_{qk}, \phi_j \rangle \langle \phi_{qk}, \phi_{j'} \rangle$, and its eigenvalues are $d_{1k}, \ldots, d_{Jk}$. Let $\vec{\mu}_J$ be a length-J vector $(\mu_1, \ldots, \mu_J)^T$ by projecting μ_d on first J bases, $\mu_j = \langle \mu_d, \phi_j \rangle$. By the law of total covariance and the result that the trace of a matrix is equal to the sum of its eigenvalues, we derive the following relationship between the two sets of eigenvalues (i.e. λ_j, λ_{jk}, and d_{jk}): $\sum_{j=1}^{J} \lambda_j = \pi_1 \sum_{j=1}^{J} d_{j1} + \pi_0 \sum_{j=1}^{J} d_{j0} + \pi_1 \pi_0 \sum_{j=1}^{J} \mu_j^2$, and $\sum_{j=1}^{J} d_{jk} = \sum_{j=1}^{J} \sum_{q=1}^{\infty} \lambda_{qk} \langle \phi_{qk}, \phi_j \rangle^2$. The following assumption is standard in functional data for the distribution of X, and ensures that $d_{jk} > 0$, for $1 \leq j \leq J$, $k = 0, 1$:

Assumption A7. *Both the group covariance operators, G_1, G_0, and the covariance matrices $\mathbf{R}_0$, $\mathbf{R}_1$ are bounded and positive definite, and $\mu_d \in \mathcal{L}^2(\mathcal{T})$.*

When X is Gaussian in both groups, $\log Q_J^*(X)$ is a quadratic form in $\mathbf{X}_J$ ($\mathbf{X}_J$ is a length-J vector with jth entry $\langle X, \phi_j \rangle$):

$$\log Q_J^*(X) = -\frac{1}{2} (\mathbf{X}_J - \vec{\mu}_J)^T \mathbf{R}_1^{-1} (\mathbf{X}_J - \vec{\mu}_J) + \frac{1}{2} \mathbf{X}_J^T \mathbf{R}_0^{-1} \mathbf{X}_J + \log \sqrt{\frac{|\mathbf{R}_0|}{|\mathbf{R}_1|}}. \quad (2.1)$$

With potentially unequal group eigenfunctions, entries in $\mathbf{X}_J$ at $Y = k$ can be correlated, which complicates the distribution of $\log Q_J^*(X)$ in each group.

Therefore, we implement a linear transformation of $\mathbf{X}_J$ in Steps i)–iii), and for simplicity, we set $\mathbf{R}_0$ as the reference level:

i) The eigendecomposition of the matrix product gives $\mathbf{R}_0^{1/2}\mathbf{R}_1^{-1}\mathbf{R}_0^{1/2} = \mathbf{P}^T\mathbf{\Delta}\mathbf{P}$, where $\mathbf{\Delta} = \text{diag}\{\Delta_1, \ldots, \Delta_J\}$, Δ_j as eigenvalues of $\mathbf{R}_0^{1/2}\mathbf{R}_1^{-1}\mathbf{R}_0^{1/2}$. By the equivalence of the determinants, $\prod_{j=1}^{J} \frac{d_{j0}}{d_{j1}} = \prod_{j=1}^{J} \Delta_j$. In addition, $\Delta_j > 0$, for all j, under A7;

ii) Let $\mathbf{Z} = \mathbf{R}_0^{-1/2}\mathbf{X}_J$, $\mathbf{U} = \mathbf{P}\mathbf{Z}$;

iii) When $k = 0$, the jth entry U_j of the vector $\mathbf{U}$ has a standard Gaussian distribution; at $k = 1$, $U_j \sim N(-b_j, 1/\Delta_j)$, with b_j the jth entry of $\mathbf{b} = -\mathbf{P}\mathbf{R}_0^{-1/2}\vec{\mu}_J$.

Consequently, the entries of $\mathbf{U}$ are uncorrelated for both $k = 0$ and 1, Eq.(2.1) becomes

$$\log Q_J^*(X) = -\frac{1}{2}\sum_{j=1}^{J} \Delta_j \left(U_j + b_j\right)^2 + \frac{1}{2}\sum_{j=1}^{J} U_j^2 + \frac{1}{2}\sum_{j=1}^{J} \log \Delta_j,$$

and the asymptotic behaviors of the Bayes classifier for Gaussian processes are concluded.

Theorem 2. *With A7, when the random function X is a Gaussian process at both $Y = 0$ and 1 and the group eigenfunctions of G_0, G_1 are unequal, the functional Bayes classifier $\mathbb{1}\{\log Q_J^*(X) > 0\}$ achieves perfect classification when either $\|\mathbf{R}_0^{-1/2}\vec{\mu}_J\|^2 \to \infty$, or $\sum_{j=1}^{J}(\Delta_j - 1)^2 \to \infty$, as $J \to \infty$. Otherwise, its error rate $err(\mathbb{1}\{\log Q_J^*(X) > 0\}) \not\to 0$.*

Theorem 2 is a natural extension of Theorem 2 in [15]. It again reveals that the error rate of the Bayes classifier approaches zero asymptotically when Π_1 and Π_0 are sufficiently different in terms of either the group means or the scores' variances. In addition, recognizing the different correlation patterns between group scores helps improve the classification accuracy. Instead of

adopting $\mu_j/\sqrt{\lambda_{j0}}$ and $\lambda_{j0}/\lambda_{j1}$ to build conditions for perfect classification, as in [15], we use the transformed $\mathbf{R}_0^{-1/2}\vec{\mu}_J$ and Δ_j to accommodate the potentially unequal group eigenfunctions and the dependent scores. For the special case when the eigenfunctions are actually equal, the covariance matrices $\mathbf{R}_k = \text{diag}\{\lambda_{1k}, \ldots, \lambda_{Jk}\}$ with $\Delta_j = \lambda_{j0}/\lambda_{j1}$, and consequently the two conditions in Theorem 2 become the same as those proposed in [15]. The proof of Theorem 2 is in the supplementary materials.

2.5.3 When X is a non-Gaussian process

For non-Gaussian processes, when the projected scores $X_{\cdot jk}$, for $1 \leq j \leq J$, fit a Gaussian copula model, that is, they are meta-Gaussian distributed, we derive sufficient conditions in terms of the marginal densities f_{jk} and the score correlations in order to achieve an asymptotically zero misclassification rate.

First, we let $\mathbf{u}_k = (u_{1k}, \ldots, u_{Jk})^T$ be a length-J random vector with $u_{jk} = \Phi^{-1}(F_{jk}(X_{\cdot j\cdot}))$, where $\Phi(\cdot)$ is the CDF of $N(0,1)$. When $Y = k$, $(u_{jk}|Y = k) \sim N(0,1)$, and $\text{var}(\mathbf{u}_k|Y = k) = \mathbf{\Omega}_k$, as denoted before. Let the eigendecomposition be $\mathbf{\Omega}_k = \mathbf{V}_k\mathbf{D}_k\mathbf{V}_k^T$, with $\mathbf{D}_k$ the diagonal matrix with eigenvalues ω_{jk}, for $j = 1, \ldots, J$. On the other hand, $u_{jk}|Y = k'$ follows a more complicated distribution when $k' \neq k$. We denote $\text{var}(\mathbf{u_k}|Y = k') = \mathbf{M}_k$ with the eigendecomposition $\mathbf{M}_k = \mathbf{U}_k\tilde{\mathbf{D}}_k\mathbf{U}_k^T$, and the eigenvalues of $\mathbf{M}_k$ are v_{jk}, for $j = 1, \ldots, J$.

Therefore, the log density ratio $\log Q_J^*(X)$ in the Bayes classifier with a Gaus-

sian copula can be represented as

$$\log Q_J^*(X) = \sum_{j=1}^{J} \log \frac{f_{j1}\left(X_{\cdot j\cdot}\right)}{f_{j0}\left(X_{\cdot j\cdot}\right)} + \frac{1}{2}\log\frac{|\mathbf{\Omega}_0|}{|\mathbf{\Omega}_1|} - \frac{1}{2}\mathbf{u}_1^T\left(\mathbf{\Omega}_1^{-1} - \mathbf{I}\right)\mathbf{u}_1 + \frac{1}{2}\mathbf{u}_0^T\left(\mathbf{\Omega}_0^{-1} - \mathbf{I}\right)\mathbf{u}_0$$
$$= \sum_{j=1}^{J} \log \frac{f_{j1}\left(X_{\cdot j\cdot}\right)}{f_{j0}\left(X_{\cdot j\cdot}\right)} \Big/ \frac{\sqrt{\omega_{j1}}}{\sqrt{\omega_{j0}}} - \frac{1}{2}\mathbf{u}_1^T\left(\mathbf{\Omega}_1^{-1} - \mathbf{I}\right)\mathbf{u}_1 + \frac{1}{2}\mathbf{u}_0^T\left(\mathbf{\Omega}_0^{-1} - \mathbf{I}\right)\mathbf{u}_0. \tag{2.2}$$

Similarly to A7, we make an assumption on the covariances of $\mathbf{u}_k$, conditional on Y:

Assumption A8. *The matrices $\mathbf{\Omega}_k$ and $\mathbf{M}_k$, for $k = 0, 1$, are bounded and positive definite.*

Next, we define a sequence of ratios g_j, for $j = 1, 2, \ldots$, by $g_j = \frac{f_{j1}\left(X_{\cdot j\cdot}\right)}{f_{j0}\left(X_{\cdot j\cdot}\right)} \Big/ \frac{\sqrt{\omega_{j1}}}{\sqrt{\omega_{j0}}}$, where g_j compares the ratio of the marginal densities to the ratio of the eigenvalues of the correlation matrices. In addition, let

$$s_{jk} = \frac{\text{var}\left(\langle V_{jk}, \mathbf{u}_k\rangle | Y = k\right)}{\text{var}\left(\langle V_{jk}, \mathbf{u}_k\rangle | Y = k'\right)} = \frac{\mathbf{V}_{jk}^T \mathbf{\Omega}_k \mathbf{V}_{jk}}{\mathbf{V}_{jk}^T \mathbf{M}_k \mathbf{V}_{jk}} = \frac{\omega_{jk}}{\sum_{q=1}^{J} C_{(j,q)k}^2 \upsilon_{qk}},$$

where $C_{(j,q)k} = \langle \mathbf{U}_{qk}, \mathbf{V}_{jk}\rangle$, $\sum_{q=1}^{J} C_{(j,q)k} = 1$, and $\mathbf{U}_{qk}$ and $\mathbf{V}_{jk}$ are the qth and jth columns, respectively, of the eigenvector matrices $\mathbf{U}_k$ and $\mathbf{V}_k$. As a result, s_{jk} compares the jth eigenvalue of $\mathbf{\Omega}_k$ against a convex combination of the eigenvalues of $\mathbf{M}_k$, the individual weights of which are determined by projecting $\mathbf{V}_{jk}$ onto the eigenvalues of $\mathbf{M}_k$, $\mathbf{U}_{qk}$.

In terms of the sequences g_j and s_{jk}, for $j = 1, 2, \ldots$, we derive the following theorem for non-Gaussian processes; the proof is in Section **??** of the supplementary.

Theorem 3. *With Assumptions A6, A7, and A8, when the projected scores $X_{\cdot jk}$, for $j = 1, \ldots, J$, are meta-Gaussian distributed at each group Π_k, perfect classification*

by the Bayes classifier $\mathbb{1}\{\log Q_J^*(X) > 0\}$ *is achieved asymptotically if a subsequence* $g_r^* = g_{j_r}$ *of* g_j *exists, with corresponding* $s_{j_r k}$*, such that one of the following conditions is satisfied as* $r \to \infty$*:*

a) $g_{j_r} = op(1)$*, and* $s_{j_r 0} \to 0$*;*

b) $1/g_{j_r} = op(1)$*, and* $s_{j_r 1} \to 0$*;*

or when g_{j_r} *has distinct behaviors in subgroups:*

c) $g_{j_r} = op(1)$ *at* $Y = 1$*,* $1/g_{j_r} = op(1)$ *at* $Y = 0$*, with both* $s_{j_r 0}$ *and* $s_{j_r 1} \to 0$*;*

d) $1/g_{j_r} = op(1)$ *at* $Y = 1$*, and* $g_{j_r} = op(1)$ *at* $Y = 0$*.*

Based on the structure of the log density ratio described in Eq.(2.2), Theorem 3 discusses the occurrence of perfect classification in two aspects: g_j, which mainly depicts the relative magnitude of the score marginal densities at each $k = 0, 1$; and s_{jk}, which compares the correlation between the scores conditioned at each group. Either part showing enough disparity between groups results in perfect classification.

For example, in Theorem 3 a), when there exists a subsequence $g_{j_r} \to 0$ in probability, indicating the dominance of the marginal densities by the group $Y = 0$, the misclassification tends to occur at $Y = 1$. However, as $s_{j_r 0} \to 0$, the covariance of $\mathbf{u}_0$ conditioned at $Y = 1$ becomes much larger than at $Y = 0$. As a result, the nonnegative $\mathbf{u}_0^T \mathbf{\Omega}_0^{-1} \mathbf{u}_0^T$ in Eq.(2.2) with large variation when $Y = 1$ compensates to eventually avoid misclassifying X to group 0. When g_{j_r} behaves perfectly, as in case d), where the corresponding group marginal densities are dominant in each subgroup $Y = k$, we do not need to impose requirements on the copula correlation to achieve perfect classification.

Remark. Theorem 3 provides sufficient, but not necessary conditions for the Bayes classifier to achieve asymptotic perfect classification under unequal group eigenfunctions. Owing to the optimality of the Bayes classifier in minimizing the zero-one loss, various conditions from other functional classifiers to achieve an asymptotically zero error also work here. For example, [18] proposed conditions in terms of group eigenvalues and the mean difference for the functional centroid classifier to reach perfect classification. These also work as sufficient conditions for $\mathbb{1}\{\log Q_J^*(X) > 0\}$ in our case. With a copula model, which is not found in previous work, Theorem 3 uses the relation between the scores' marginal densities and correlations to reduce the error rate to zero asymptotically.

2.6 Discussion

2.6.1 Remarks

Our copula-based Bayes classifiers remove the assumptions of equal group eigenfunctions and independent scores. As our two examples show, it is not uncommon to have unequal group eigenfunctions. The new methods also prove to have stronger performance in terms of dimension reduction than that of the original BC. Our simulation results prove the strength of our method in distinguishing groups by the differences in their functional means and their covariance functions. We examined the two choices of projection directions, PC and PLS. PLS can detect location differences on eigenfunctions corresponding to smaller eigenvalues. We discussed new conditions for the estimated classi-

fier to be asymptotically equivalent to the true Bayes classifier, and for perfect classification to occur. These differ from those of previous works, owing to the unequal group eigenfunction setting. We also imposed sparsity conditions on the inverse of the copula correlations.

2.6.2 Future Work

In future work, we would like to extend the copula-based classification to the problem with multiple functional covariates. Some previous works discuss this situation in the framework of functional generalized models: [13] proposed a generalized multilevel regression model where there are repeated curve measurements for each subject; [69] discussed an FGLM approach for the classification of multilevel functions with Bayesian variable selection; and [40] present a generalized functional linear model where there are both functional and multivariate covariates, and use a semiparametric single-index function to model the interaction between them. We plan to approach the problem from a different angle, using functional Bayes classification again, owing to its strong performance in the single functional predictor case. Furthermore, because it is natural to assume that the response depends on the covariates and their interactions, it becomes more important for our method to model the dependency between the projected scores. Another aspect we would like to consider is how to choose a proper functional basis for multiple functional predictors.

Acknowledgements

The MRI/DTI data in the refund package were collected at Johns Hopkins

University and the Kennedy–Krieger Institute.

CHAPTER 3

ADAPTIVE RIDGE-PENALIZED FUNCTIONAL LOCAL LINEAR REGRESSION

3.1 Introduction

Functional data analysis has received increasing attention during the past few decades with applications in a variety of fields such as chemometrics, medicine, and environmental science. This article focuses on scalar-on-function regression where an unknown function, m, describes the relationship between a predictor function X in some Hilbert space and a real scalar Y. The model is $Y = m(X)+\epsilon$ where ϵ is random error. We assume an independent, identically distributed sample (X_i, Y_i), $i = 1, \ldots, n$.

Past work such as Cai et al. (2006 [8]) and Reiss and Ogden (2007 [54]) discussed estimation when m is linear, so that $m(X) = \langle X, \beta \rangle$, the inner product of X and an unknown coefficient function β. However, the linearity assumption often fails to hold. For instance, in Section 3.5 we plot the estimated derivatives of m at each observed function, X_i, in two real data sets (See Fig. 3.3 & Fig. 3.5). Linearity implies that the derivative of m at X is equal to β at all X. Variation of the derivative's shape as X_i varies shows the nonlinearity of m in these examples.

Nonparameteric methods that have been widely used in multivariate regression have been extended to functional predictors and have shown strong performance there. For example, Ferraty et al. (2007 [22]) applied the well known Nadaraya-Watson kernel estimator to regression with functional predictors. For

regression with a scalar or low-dimensional covariate, local polynomial regression has advantages over kernel regression, e.g., better behavior near the boundary of the covariate space [20, 59]. Therefore, it is natural to study local polynomial functional regression.

Baíllo and Grané (2009 [1]) first extended the multivariate local linear regression estimator of Ruppert and Wand (1994 [59]), where X is finite-dimensional, to functional data. Boj et al. (2010 [6]) and Barrientos-Marin et al. (2010 [3]), among many others, also studied local linear regression with functional predictors. Ferraty and Nagy (2019 [24]) discussed in detail the implementation of functional local linear regression (FLLR) and its asymptotic behavior, while Berlinet et al. (2011 [5]) explored the model from a purely theoretical perspective. As in multivariate regression, FLLR often has better prediction accuracy than kernel estimators.

Because of the so-called curse-of-dimensionality, nonparametric estimators such as local linear regression can be problematic in high dimensional spaces. Function spaces are infinite dimensional, so local polynomial regression might seem unsuitable for functional regression. Fortunately, functional data often lie in a low-dimensional subspace, e.g., in the space spanned by the first few principal component directions. Therefore, to implement local polynomial functional regression, one can project the data onto a subspace of dimension J, e.g., the first J principal components, where J is a tuning parameter. However, the estimator can be sensitive to the choice of J and, even with the best choice of J, the estimator will likely be improved with a roughness penalty.

To improve the FLLR estimator, we propose a data-adaptive ridge roughness penalty. The most general ridge penalty matrix is a $J \times J$ positive semidefinite

matrix. Data-based selection of this type of penalty matrix with $J(J+1)/2$ free parameters can be difficult and can result in an unstable and inefficient estimator. Therefore, we propose a data-adaptive ridge penalization that utilizes a specific class of positive semidefinite diagonalizable matrices. As will be shown later, this structure with only J free parameters enables a quadratic programming search for optimal tuning parameters that minimize the estimated mean squared error (MSE) of prediction. Our method of penalization also accommodates a different roughness penalty level on each basis function and avoids the computational cost of multivariate cross validation as J increases.

Reiss and Ogden (2007 [54]) suggested a univariate roughness penalty for functional linear models. Reiss et al. (2017 [53]) explored adding a fixed univariate ridge penalty onto nonparametric functional estimators with smoothing splines. Both papers select a single smoothing parameter by generalized cross validation or restricted maximum likelihood (REML) estimation of variances, and neither discussed the estimators' asymptotic behaviors. As far as we know, there is no previous work investigating multidimensional ridge penalties in functional nonparametric regression. Our estimator has strong prediction performance in both simulations and real data examples, especially when the model is nonlinear. In addition, the method shows effective bandwidth size control for finite data samples, proving its strength in variance reduction. Asymptotic properties of the new estimator are derived, and a detailed implementation is provided, including a two-step bandwidth selection for estimating m and its functional derivative m'.

In Section 3.2, we introduce our model and the design of the ridge penalty. In Section 3.3, we estimate the mean square error (MSE) of our estimator and

discuss its asymptotic estimator behavior. Section 3.4 provides a detailed description of the implementation of the estimator and includes a comprehensive simulation study to compare the performance of multiple nonparametric methods under different linearity levels of m. Section 3.5 uses two real datasets to examine the performance of our method. In the end, we discuss potential future work. Additional results and detailed proofs can be found in the supplementary materials.

3.2 Methodology

3.2.1 Functional Local Linear Regression

We consider a pair of variables $(X, Y) \in \mathcal{L}^2(\mathcal{T}) \times \mathbb{R}$, which means X is a square integrable random function over a compact interval $\mathcal{T}$, and Y is real valued. Suppose there exists a regression model $Y = m(X) + \epsilon$, where $m : \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ is first order differentiable, and $\epsilon \sim N(0, \sigma_e^2)$. In this article we are interested in the estimation of $E(Y|x) = m(x)$ at a point x, using n i.i.d. samples (X_i, Y_i) collected from the joint distribution (X, Y).

As discussed in the introduction, we use functional local linear regression (FLLR) to estimate $m(x)$. However, FLLR estimates not only $m(x)$ but also its first derivative, $m'_x : \mathcal{L}^2(\mathcal{T}) \to \Re$. Although FLLR has become a well-studied technique for nonparametric functional regression, there has been relatively little research on regularizing the high-dimensional estimate of m'_x. For this purpose, we suggest a new FLLR model with data-adaptive ridge penalization, which we denote as FLLR-r. As will be seen in Section 3.4, regularization of

$\hat{m}'_x$ also improves $\hat{m}(x)$.

Below are several assumptions needed for the FLLR-r estimator:

Assumption A9. *The continuously differentiable function $K : \mathbb{R} \to \mathbb{R}^+$ is a kernel of Type I, whose definition can be found in, for example, Ferraty and Vieu (2006 [25]): $\int K = 1$, and $c_K \mathbb{1}_{[0,1]} \leq K \leq C_K \mathbb{1}_{[0,1]}$ where $c_K, C_K > 0$;*

Assumption A10. *$\forall h > 0, \psi_x(h) = P(\|X_i - x\| < h) > 0$. Also, as $n \to \infty$, $h = h_n \to 0$, $n\psi_x(h) \to \infty$, $J = J(n) \to \infty$.*

Assumption A11. *For $x \in \mathcal{L}^2(\mathcal{T})$, the model $m : \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ is first and second order differentiable at its neighborhood $\mathcal{N}_x$, with the corresponding bounded derivative functional $m'_x : \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ and $m''_x : \mathcal{L}^2(\mathcal{T}) \times \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$. Also, for all $u \in \mathcal{L}^2(\mathcal{T})$ and $x + u \in \mathcal{N}_x$, there is $0 < \rho < 1$ and $r = x + \rho u$ s.t.*

$$m(x+u) = m(x) + m'_x(u) + \frac{1}{2} m''_r(u^2)$$

Assumption A11 is an application of Taylor's Theorem in function spaces (Zeidler, 1995 [68]), and is similar to Assumption H1 in Ferraty and Nagy (2019 [24]).

Let $\phi_1, \phi_2, \ldots \in \mathcal{L}^2(\mathcal{T})$ be a set of orthogonal basis functions, $X = \sum_{j=1}^{\infty} \langle X, \phi_j \rangle \phi_j$, $c_{ij} = \langle X_i - x, \phi_j \rangle$, and $X_i - x = \sum_{j=1}^{\infty} c_{ij} \phi_j$. With a first-order Taylor expansion at x, we have

$$E(Y_i|X_i) = m(X_i) = m(x) + m'_x(X_i - x) + o_p(\|X_i - x\|). \tag{3.1}$$

Eq. (3.1) can therefore be estimated using a basis truncated at J:

$$m(X_i) \approx m(x) + \sum_{j=1}^{\infty} c_{ij} m'_x(\phi_j) \approx m(x) + \mathbf{c}_i^T \mathbf{m}'_{x,J},$$

where $\mathbf{c}_i = (c_{i1}, \ldots, c_{iJ})^T$ and $\mathbf{m}'_{x,J} = \{m'_x(\phi_1), \ldots, m'_x(\phi_J)\}^T$. Define $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_J)^T = \mathbf{m}'_{x,J}$, so $\boldsymbol{\beta}^T\boldsymbol{\Phi}$ is the derivative functional m'_x projected on the subspace spanned by $\boldsymbol{\Phi} = \{\phi_1, \ldots, \phi_J\}^T$. In addition, $\mathbf{C}$ is an $n \times J$ matrix with rows $\mathbf{c}_i^T$, and $\mathbf{Y}$ is the vector of responses Y_i. FLLR then estimates $m(x)$ and $\boldsymbol{\beta}$ by minimizing the weighted sum of squared errors $\sum_{i=1}^n \left(Y_i - m(x) - \mathbf{c}_i^T\boldsymbol{\beta}\right)^2 \Delta_i$, with the kernel weights $\Delta_i = \dfrac{K\left(\|X_i - x\|/h\right)}{E\left\{K\left(\|X_i - x\|/h\right)\right\}}$.

3.2.2 Ridge Penalty in FLLR

By its nature, an FLLR model is characterized by the truncated basis count J and the bandwidth h, which in practice are usually determined by cross validation. As mentioned earlier, we'd like to introduce a multidimensional ridge penalty into the functional regression. The new method constructs the penalty using a data-adaptive basis learnt from sample functions, and enables a parameter selection that minimizes the finite sample estimation error of $m(x)$.

Let $\mathbf{H}^*$ be a $J \times J$ positive semidefinite penalty matrix, and $\mathbf{H} = \begin{pmatrix} 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{H}^* \end{pmatrix}$. Optimal estimates of $m(x)$ and $\boldsymbol{\beta}$ satisfy the following:

$$\hat{m}(x), \hat{\boldsymbol{\beta}} = \text{argmin}_{m(x),\boldsymbol{\beta}} \left\{ \sum_{i=1}^n \left(Y_i - m(x) - \mathbf{c}_i^T\boldsymbol{\beta}\right)^2 \Delta_i + \boldsymbol{\beta}^T\mathbf{H}^*\boldsymbol{\beta} \right\}. \tag{3.2}$$

Thus,

$$\hat{m}(x) = \mathbf{e}_1^T \left(\frac{1}{n}\mathbf{C}_x^T\boldsymbol{\Delta}\mathbf{C}_x + \mathbf{H}\right)^{-1} \frac{1}{n}\mathbf{C}_x^T\boldsymbol{\Delta}\mathbf{Y}, \tag{3.3}$$

where $\mathbf{C}_x = \begin{pmatrix} \mathbf{1} & \mathbf{C} \end{pmatrix}$, and $\boldsymbol{\Delta}$ is the diagonal weight matrix with $\{\Delta_i\}_{i=1}^n$ as entries.

We set up the matrix $\mathbf{H}^*$ to accommodate different levels of roughness

penalty for $\hat{\beta}_j$'s. As $\hat{\beta}_j$'s estimate the first order derivatives m'_x along each basis direction ϕ_j, variation in penalty is reasonable. There has been little work discussing multidimensional ridge penalization applied to nonparametric functional regressions. Reiss et al. (2017 [53]) established a real data example of signature verification using a scalar-on-function principal coordinate model, with a fixed-value ridge parameter. Seifert and Gasser (2000 [60]) pointed out that, in multivariate local linear regression, it was unlikely to find a stable minimum among the whole space of nonnegative ridge matrices. They mentioned a potential approach to iteratively search for optimal eigenvalues, given any set of eigenvectors, to minimize the mean squared error of the estimator, but they did not discuss this idea further.

Here we develop a data-adaptive $\mathbf{H}^*$ that is amenable to theoretical work and allows a stable implementation of a multidimensional ridge penalty. Let $\mathbf{H}^*$ be in the class of matrices diagonalizable by $\mathbf{V}$: $\mathbf{H}^* \in \mathcal{R}(J, \mathbf{V}) := \{\mathbf{R}^{J\times J} : \mathbf{V}^T\mathbf{R}\mathbf{V}$ is diagonal$\}$, where $\mathbf{V}$ is the eigenvector matrix of the weighted sample covariance of scores $\langle X_i - x, \phi_j\rangle$, $1 \leq j \leq J$ (detailed discussion of $\mathbf{V}$ is included in Section 3.3). Thus, $\mathbf{H}^* = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, $\mathbf{\Lambda}$ the diagonal matrix with entries $\lambda_j \geq 0$, $1 \leq j \leq J$.

The data-based matrix $\mathbf{V}$ carries out a change of basis along the eigenfunctions of the weighted covariance operators based on x: $\phi_j^* = \mathbf{V}_j^T\mathbf{\Phi} \in \mathcal{L}^2(\mathcal{T})$, $1 \leq j \leq J$, $\mathbf{\Phi}$ as defined at the end of Section 3.2.1. Consequently, $\boldsymbol{\beta}^* = \mathbf{V}^T\boldsymbol{\beta}$, the derivative functional m'_x projected onto the new directions, can be estimated with a multidimensional roughness penalty to avoid overfitting. When all λ_j's are equal, this becomes the special case of applying univariate ridge penalty λ on squared norm of $\boldsymbol{\beta}$. Throughout later calculations such as Eq. (3.9) and (3.10),

the structure of $\mathbf{H}^*$ facilities an asymptotic analysis and, in applications, avoids the potential instability of multivariate cross validation when using a more general penalty matrix.

3.3 Mean Squared Error (MSE) and Parameter Selection

In this section, we estimate the mean squared error (MSE) of the FLLR-r estimator $\hat{m}(x)$ and explore the FLLR-r estimator's asymptotic behavior. The optimal ridge parameters are selected by minimizing the finite-sample estimated MSE by quadratic programming.

3.3.1 Estimated Bias and Variance

We start with the conditional bias of $\hat{m}(x)$:

$$\begin{aligned}\text{bias}\left(\hat{m}(x)\right) &= E[\hat{m}(x)|X_1,\ldots,X_n] - m(x) \\ &= \mathbf{e}_1^T\left(\frac{1}{n}\mathbf{C}_x^T\mathbf{\Delta}\mathbf{C}_x + \mathbf{H}\right)^{-1}\frac{1}{n}\mathbf{C}_x^T\mathbf{\Delta}\left[B_1\left(x\right), B_2\left(x\right), \ldots, B_n\left(x\right)\right]^T,\end{aligned} \tag{3.4}$$

where $B_i\left(x\right) = m'_x(X_i - x) + \frac{1}{2}m''_{r_i}\left((X_i - x)^2\right)$ and $r_i = x + \rho_i\left(X_i - x\right)$ for some $\rho_i \in (0,1)$, $i = 1,\ldots,n$. We estimate the MSE using an estimate of the truncated bias which is

$$\text{bias}^{(J)}\left(\hat{m}(x)\right) = \mathbf{e}_1^T\left(\frac{1}{n}\mathbf{C}_x^T\mathbf{\Delta}\mathbf{C}_x + \mathbf{H}\right)^{-1}\frac{1}{n}\mathbf{C}_x^T\mathbf{\Delta}\mathbf{C}\mathbf{m}'_{x,J}. \tag{3.5}$$

In addition, the exact variance of $\hat{m}(x)$ is

$$\text{Var}\left(\hat{m}\left(x\right)\right) = n^{-2}\sigma_e^2 \mathbf{e}_1^T (n^{-1}\mathbf{C}_x^T \mathbf{\Delta} \mathbf{C}_x + \mathbf{H})^{-1} \mathbf{C}_x^T \mathbf{\Delta}^2 \mathbf{C}_x (n^{-1}\mathbf{C}_x^T \mathbf{\Delta} \mathbf{C}_x + \mathbf{H})^{-1} \mathbf{e}_1 \tag{3.6}$$

Based on Eq. (3.5) and (3.6), we define the truncated MSE as $\text{MSE}_x\left(J, \mathbf{H}, h\right) = \left\{\text{bias}^{(J)}\left(\hat{m}(x)\right)\right\}^2 + \text{Var}(\hat{m}(x))$. We then reconstruct MSE_x using the new orthogonal basis $\phi_1^*, \ldots \phi_J^*$.

3.3.2 Reconstructed MSE and Ridge Penalty Optimization

As stated in Section 3.2.2, the columns of $\mathbf{V}$ for the basis change are the eigenvectors of the weighted sample covariance of the projected scores $\langle X_i, \phi_j \rangle$, $1 \leq j \leq J$. We define two weighted sample statistics of the projected scores: let $\hat{\mu}_j$ be the weighted average where $\hat{\mu}_j = \sum_i \Delta_i \langle X_i - x, \phi_j \rangle / \sum_i \Delta_i$, and similarly the weighted sample covariance is $\hat{\sigma}_{jk} = \sum_i \Delta_i \langle X_i - x, \phi_j \rangle \langle X_i - x, \phi_k \rangle / \sum_i \Delta_i - \hat{\mu}_j \hat{\mu}_k$ for scores on basis ϕ_j and ϕ_k. Therefore, $\hat{\boldsymbol{\mu}} = \mathbf{C}^T \mathbf{\Delta} \mathbf{1} / \sum_i \Delta_i = (\hat{\mu}_1, \ldots, \hat{\mu}_k)^T$, and the $J \times J$ sample covariance matrix $\mathbf{W}_{x,J}$ is

$$\mathbf{W}_{x,J} = \left(\mathbf{C}^T - \hat{\boldsymbol{\mu}} \mathbf{1}^T\right) \mathbf{\Delta} \left(\mathbf{C} - \mathbf{1} \hat{\boldsymbol{\mu}}^T\right) \Big/ \sum_i \Delta_i = \mathbf{C}^T \mathbf{\Delta} \mathbf{C} \Big/ \sum_i \Delta_i - \hat{\boldsymbol{\mu}} \hat{\boldsymbol{\mu}}^T. \tag{3.7}$$

Columns of the matrix $\mathbf{V}$ are the eigenvectors of $\mathbf{W}_{x,J}$. Positive semi-definiteness of $\mathbf{W}_{x,J}$ is proved in the supplementary material.

With $a = (n^{-1} \sum_{i=1}^n \Delta_i)^{-1}$, we define

$$\begin{aligned} \tilde{\mathbf{M}} &= a^{-1} \mathbf{W}_{x,J} = \mathbf{V} \mathbf{\Lambda}^* \mathbf{V}^T \\ &= n^{-1} \mathbf{C}^T \mathbf{\Delta} \mathbf{C} - a \left(n^{-1} \mathbf{C}^T \mathbf{\Delta} \mathbf{1}\right) \left(n^{-1} \mathbf{1}^T \mathbf{\Delta} \mathbf{C}\right), \end{aligned} \tag{3.8}$$

where $\mathbf{\Lambda}^*$ is the diagonal matrix containing eigenvalues $\tilde{\gamma}_j > 0$, $\forall 1 \leq j \leq J$, of $\tilde{\mathbf{M}}$. By the asymptotic properties from Baíllo and Grané (2009 [1]), $a^{-1} =$

$1 + o_p\left((n\psi_x(h))^{-1/2}\right)$. Then, $\mathbf{M} = \tilde{\mathbf{M}} + \mathbf{H}^* = \mathbf{V}\mathbf{D}\mathbf{V}^T$, where $\mathbf{D} = \mathbf{\Lambda}^* + \mathbf{\Lambda}$. Some other key terms denoted are:

- $\mathbf{d}_1 = \mathbf{V}^T\left(\frac{1}{n}\mathbf{C}^T\mathbf{\Delta}\mathbf{1}\right) = a^{-1}\hat{\boldsymbol{\mu}}^*$, where $\hat{\boldsymbol{\mu}}^* = \mathbf{V}^T\hat{\boldsymbol{\mu}}$ is the weighted sample average of scores $\langle X_i - x, \phi_j\rangle$ on the new basis ϕ_j^*, $1 \le j \le J$;
- $d_2 = n^{-1}\mathbf{1}^T\mathbf{\Delta}\mathbf{C}\mathbf{m}'_{x,J} = a^{-1}\langle\hat{\boldsymbol{\mu}}, \mathbf{m}'_{x,J}\rangle$;
- $\mathbf{d}_3 = \mathbf{V}^T\left(\frac{1}{n}\mathbf{C}^T\mathbf{\Delta}\mathbf{C}\right)\mathbf{m}'_{x,J} = a^{-1}\mathbf{V}^T\left(\mathbf{W}_{x,J} + \hat{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}^T\right)\mathbf{m}'_{x,J}$.

After some calculation, the bias from Eq.(3.5) is re-expressed as

$$\begin{aligned}\text{bias}^{(J)}(\hat{m}(x)) &= \langle\hat{\boldsymbol{\mu}}, \mathbf{m}'_{x,J}\rangle + a^{-1}\langle\hat{\boldsymbol{\mu}}, \mathbf{m}'_{x,J}\rangle\langle\hat{\boldsymbol{\mu}}^*, \mathbf{D}^{-1}\hat{\boldsymbol{\mu}}^*\rangle \\ &\quad - \hat{\boldsymbol{\mu}}^{*T}\mathbf{D}^{-1}\mathbf{V}^T\left(\mathbf{W}_{x,J} + \hat{\boldsymbol{\mu}}\hat{\boldsymbol{\mu}}^T\right)\mathbf{m}'_{x,J} \\ &= ad_2 + a^2 d_2 \cdot \mathbf{d}_1^T\mathbf{D}^{-1}\mathbf{d}_1 - a \cdot \mathbf{d}_1^T\mathbf{D}^{-1}\mathbf{d}_3, \end{aligned} \tag{3.9}$$

and also the exact variance of $\hat{m}(x)$ in Eq.(3.6) is

$$\text{Var}(\hat{m}(x)) = \left\|\sigma_e\mathbf{\Delta}\frac{1}{n}\left[a\mathbf{1} + \left(a^2\mathbf{1}\mathbf{d}_1^T - a\mathbf{C}\mathbf{V}\right)\mathbf{D}^{-1}\mathbf{d}_1\right]\right\|^2 \tag{3.10}$$

Detailed calculations are included in the supplementary materials. Having $\mathbf{H}^*$ in the class of $\mathcal{R}(J, \mathbf{V})$ circumvents the complication of general matrix inversion, and transforms the problem of building $\text{MSE}_x\,(J, \mathbf{H}, h)$-optimal $\mathbf{H}^*$ into a quadratic programming problem with parameters $\lambda_1, \ldots, \lambda_J$, which are stored only in $\mathbf{D}$.

For brevity, we let

- $\mathbf{D}_1^* = \text{diag}\{\mathbf{d}_1\}$, $\mathbf{A}_1^* = (a^2 d_2\mathbf{d}_1^T - a\mathbf{d}_3^T)\mathbf{D}_1^*$, $\mathbf{A}_2^* = \frac{1}{n}\sigma_e\mathbf{\Delta}(a^2\mathbf{1}\mathbf{d}_1^T - a\mathbf{C}\mathbf{V})\mathbf{D}_1^*$;
- $S_1 = -ad_2$, $\mathbf{S}_2 = -\frac{1}{n}a\sigma_e\mathbf{\Delta}\mathbf{1}$;

- $1/\tilde{\boldsymbol{\gamma}} = (1/\tilde{\gamma}_1, \ldots, 1/\tilde{\gamma}_J)$, and $\mathbf{b} = \{(\tilde{\gamma}_1 + \lambda_1)^{-1}, \ldots, (\tilde{\gamma}_J + \lambda_J)^{-1}\}^T$.

With Eq. (3.9) and (3.10), we search for optimal $\mathbf{b}^*$, and therefore optimal $\lambda_1^*, \ldots, \lambda_J^*$, by minimizing $\text{MSE}_x\left(J, \mathbf{H}, h\right)$:

$$\begin{gathered}\min_{\mathbf{b}} \text{MSE}_x\left(J, \mathbf{H}, h\right) = \min\left\{\|\mathbf{A}_1^*\mathbf{b} - S_1\|^2 + \|\mathbf{A}_2^*\mathbf{b} - \mathbf{S}_2\|^2\right\}, \\ \text{s.t. } \mathbf{0} \leq \mathbf{b} \leq 1/\tilde{\gamma}.\end{gathered} \tag{3.11}$$

For $\mathbf{m}'_{x,J}$ in d_2 and $\mathbf{d}_3$, we use direct plug-in estimator $\hat{\boldsymbol{\beta}}^P$ from fitting Eq. (3.2) where $\mathbf{H}^*$ is zero matrix, i.e., from the original FLLR rule, and $\hat{\sigma}_e$ is the standard error from FLLR fitting.

Remark. The data-adaptive structure of $\mathbf{H}^*$ enables the estimated MSE of $\hat{m}(x)$ to be written in a quadratic form in terms of the ridge parameters $\lambda_1, \ldots, \lambda_J$ for optimization, while general multivariate diagonal matrices would fail to do so. Based on Seifert and Gasser's (2000 [60]) discussion of multivariate local polynomial regression, a more generic approach to find optimal eigenvalues of the general ridge matrix with other given sets of eigenvectors (unequal to $\mathbf{V}$) may be found iteratively, but at potentially high computational cost, while $\mathbf{H}^* = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T$ is not only empirically stable, but has desirable theoretical properties, which we discuss below.

3.3.3 Asymptotic Properties of FLLR-r

Let $\mathcal{P}_J m'_x$ be the projection of the bounded linear functional m'_x onto the subspace of $\mathcal{L}^2(\mathcal{T})$ spanned by $\phi_1, \ldots, \phi_J$ (also by $\phi_1^*, \ldots, \phi_J^*$), and $\mathcal{P}_{J^\perp} m'_x$ the projection onto the complementary subspace. We derive the asymptotic properties of $\hat{m}(x)$ as follows.

Theorem 4. *Let Assumptions A9 - A11 hold. As $n \to \infty$, the conditional bias and variance of FLLR-r estimator $\hat{m}(x)$ are*

i) $E\left(\hat{m}(x)|X_1,\ldots,X_n\right) - m(x) = O_P\left(\|\mathcal{P}_J m_x'\|h\right) + O_P\left(\|\mathcal{P}_{J^\perp} m_x'\|h\right) + O_P\left(h^2\right) + \kappa_J \cdot \{O_P\left(\|\mathcal{P}_{J^\perp} m_x'\|h^3\right) + O_P\left(h^4\right)\}$, *where* $\kappa_J = \max_{1\le j\le J} \frac{1}{\tilde{\gamma}_j + \lambda_j}$;

ii) $Var\left(\hat{m}(x)|X_1,\ldots,X_n\right) = O_P\left(\frac{1}{n\psi_x(h)}\right) + \kappa_J \cdot O_P\left(\frac{h^2}{n\psi_x(h)}\right) + \kappa_J^2 \cdot O_P\left(\frac{h^4}{n\psi_x(h)}\right)$.

For the projected derivatives $\mathcal{P}_J m_x'$ and $\mathcal{P}_{J^\perp} m_x'$ in i), $m_x' = \mathcal{P}_J m_x' + \mathcal{P}_{J^\perp} m_x'$, and $\|m_x'\|^2 = \|\mathcal{P}_J m_x'\|^2 + \|\mathcal{P}_{J^\perp} m_x'\|^2$. The sizes of both $\mathcal{P}_J m_x'$ and $\mathcal{P}_{J^\perp} m_x'$ are dependent on the magnitude of the derivative m_x'. As J increases, $\|\mathcal{P}_J m_x'\| \to \|m_x'\|$ and $\|\mathcal{P}_{J^\perp} m_x'\| \to 0$. The coefficient κ_J is the minimum sum of a weighted covariance eigenvalue plus a corresponding ridge penalty. We here add an additional assumption A12 on κ_J to discuss asymptotic behavior further.

Assumption A12. *As $n \to \infty$ and $J = J(n) \to \infty$, $h^2 / \min_{1\le j\le J}\{\tilde{\gamma}_j + \lambda_j\} = O_P(1)$. Or equivalently, $h^2\kappa_J = O_P(1)$.*

Assumptions about the minimum eigenvalues of the score covariance matrices are not uncommon in local regression. See e.g., Reiss et al.(2017 [53]), Ferraty and Nagy (2019 [24]). Here we are able to relax the restriction on the decay rates of the eigenvalues, as the ridge parameters can compensate for fast decreasing $\tilde{\gamma}_j$'s. With the additional Assumption A12, bias and variance of $\hat{m}(x)$ are

Corollary 4.1. *With Assumptions A9-A12,*

i) $E\left(\hat{m}(x)|X_1,\ldots,X_n\right) - m(x) = O_P\left(\|\mathcal{P}_J m_x'\|h\right) + O_P\left(\|\mathcal{P}_{J^\perp} m_x'\|h\right) + O_P\left(h^2\right)$;

ii) $Var\left(\hat{m}(x)|X_1,\ldots,X_n\right) = O_P\left(\frac{1}{n\psi_x(h)}\right)$.

Consequently, compared to FLLR estimator in Ferraty and Nagy (2019 [24]), the bias of $\hat{m}(x)$ has an additional term $O_P\left(\|\mathcal{P}_J m_x'\| h\right)$ from the ridge penalty, while the bound on the variance of $\hat{m}(x)$ is equivalent to FLLR's under A9 – A12.

3.4 Simulation

We use simulated data to compare the performance of FLLR-r with the unpenalized local linear model FLLR, as well as the functional Nadaraya-Watson estimator (NW). The functional Nadaraya-Watson estimator is a natural extension of its multivariate version, discussed in past work, e.g., Ferraty et al. (2007 [22]). NW estimates $m(x)$ as

$$\hat{m}^{NW}(x) = \frac{\sum_{i=1}^{n} Y_i K\left(\|X_i - x\| / h\right)}{\sum_{i=1}^{n} K\left(\|X_i - x\| / h\right)}$$

This section also discusses data-based selection of h and J.

3.4.1 Data Setup

We use 201 Fourier basis on $\mathcal{T} = [0, 1]$ for sample curve generation, where $\phi_1(t) = 1, \phi_2(t) = \sqrt{2}\cos(2\pi t), \phi_3(t) = \sqrt{2}\sin(2\pi t), \ldots, \phi_j(t) = \sqrt{2}\cos(j\pi t)$ or $\sqrt{2}\sin\left((j-1)\pi t\right)$ for $1 < j \leq 201$ according as j is even or odd. With eigenvalues $\theta_j = 1/j$, $X_i = \sum_{j=1}^{201} \sqrt{\theta_j} U_{ij} \phi_j$, where U_{ij} is uniformly distributed i.i.d. scores on $[-\sqrt{3}, \sqrt{3}]$. The curves X_i are observed on 51 equispaced points, $t = 0, 0.02, \ldots, 1$, on $\mathcal{T} = [0, 1]$, with observation error $\xi_t \sim N(0, \sigma_t = 0.2)$. Local linear pre-smoothing is applied with the direct plug-in bandwidth of Ruppert el al. ([58]).

We follow the spirit of Ferraty and Nagy (2019 [24]) to design the regression m: $\mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ as a combination of linear and nonlinear models. Let

$$\begin{aligned} m(X_i) &= (1-a)\,\langle X_i, \sum_{j=1}^{30} \phi_j \rangle + a \sum_{j=1}^{20} \exp\left(-\langle X_i, \phi_j \rangle^2\right) \\ &= (1-a) \sum_{j=1}^{30} \sqrt{\theta_j} U_{ij} + a \sum_{j=1}^{20} \exp\left(-\theta_j U_{ij}^2\right), \end{aligned} \tag{3.12}$$

where the sliding parameter $a \in [0,1]$ varies the shape of m between the linear regression (when $a = 0$) and strongly nonlinear regression (when $a = 1$). Random error $\epsilon_i \sim N(0, \sigma_e = 0.5)$ is added to each observation: $Y_i = m(X_i) + \epsilon_i$.

3.4.2 Selection of Tuning Parameters J^*, h_r, h_d

There are several global tuning parameters we must select for estimating $m(x)$: the optimal cut-off basis count J^*, regression bandwidth h_r and derivative bandwidth h_d.

As noted at the end of Section 3.3.2, an estimated derivative vector at x, $\hat{\boldsymbol{\beta}}_x^P$, is necessary for the constructing ridge penalty. As an estimator, we use

$$\hat{\boldsymbol{\beta}}_x^P = [\mathbf{0} \quad \mathbf{I}] \left(\frac{1}{n}\mathbf{C}_x^T \boldsymbol{\Delta} \mathbf{C}_x\right)^{-1} \frac{1}{n}\mathbf{C}_x^T \boldsymbol{\Delta} \mathbf{Y},$$

where $[\mathbf{0} \quad \mathbf{I}]$ is a $J \times (J+1)$ matrix with a first column of 0's followed by an identity matrix. We obtain the preliminary $\hat{\boldsymbol{\beta}}_x^P$ from FLLR, i.e., without a ridge penalty, using the bandwidth h_d discussed below. Then, using $\hat{\boldsymbol{\beta}}_x^P$ we estimate $m(x)$ with a different bandwidth, h_r, by FLLR-r fitting, i.e., with a ridge penalty. Ferraty and Nagy (2019 [24]) mentioned that the asymptotic behavior of the estimated regression operator and its derivative are different, which is the motivation for using two distinct bandwidths h_r, h_d.

Nested leave-one-out cross-validation (LOOCV) is used for J^* and h_r, but it is not suitable for h_d, as there is no direct way to measure the fitness of $\hat{\boldsymbol{\beta}}_x^P$. Instead, we adopt wild bootstrapping of residuals to select h_d. The wild bootstrap was proposed by Wu (1986 [65]), and Ferraty et al. (2007 [22]) introduced it for bandwidth selection in nonparameteric functional regression. Later, this method was applied to first-order functional derivative estimation by Ferraty and Nagy (2019 [24]). Also, Slaoui (2020 [63]) adopted the wild bootstrapping for bandwidth selection in recursive nonparametric functional regression.

The tuning of the parameters h_d, h_r, and J follows these steps:

i) For each candidate cut-off basis J: use LOOCV to select the optimal FLLR bandwidth h_{LL} which satisfies

$$\min_h \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{m}_{LL}^{(-i)}\left(X_i|h,J\right)\right)^2,$$

where $\hat{m}_{LL}^{(-i)}$ is the FLLR estimated regression operator at X_i, with X_i removed in training. In addition, denote the estimated derivative at X_i using $h_{LL}\left(J\right)$ as $\tilde{\boldsymbol{\beta}}_{X_i}(h_{LL})$, which is estimated simultaneously with $\hat{m}_{LL}^{(-i)}$. Note that h_{LL} is dependent on J.

ii) Define the residuals $\hat{\epsilon}_i = Y_i - \hat{m}_{LL}^{(-i)}\left(X_i|h_{LL},J\right)$. Let the wild bootstrapped residuals be $\epsilon_i^b = \hat{\epsilon}_i \cdot v_i^b$, where v_i^b, $i = 1,\ldots,n$ are i.i.d. random variables with $E(v_i^b) = 0$, and the next several moments equal to 1. We use the most common choice, Mammen's two-point distribution (Mammen, 1993 [43]):

$$v_i^b = \begin{cases} -\left(\sqrt{5}-1\right)/2, \text{ with probability } \left(\sqrt{5}+1\right)/\left(2\sqrt{5}\right), \\ \left(\sqrt{5}+1\right)/2, \text{ with probability } \left(\sqrt{5}-1\right)/\left(2\sqrt{5}\right). \end{cases}$$

In this case, $E(v_i^b) = 0$, $E\{(v_i^b)^2\} = 1$, and $E\{(v_i^b)^3\} = 1$, which ensure that the bootstrapped residuals ϵ_i^b have same first three moments as $\hat{\epsilon}_i$,

$i = 1, \ldots, n$ (see e.g. [42], [43]). Other choices of v_i include the Rademacher distribution (Davidson and Flachaire, 2008 [16]) and Mammen's continuous distribution (Mammen, 1993 [43]).

iii) Set $Y_i^b = \hat{m}_{LL}^{(-i)}(X_i|h_{LL}, J) + \epsilon_i^b$. For each of $b = 1, \ldots, B$ repetitions, estimate the derivative at X_i with bandwidth h as $\tilde{\boldsymbol{\beta}}_{X_i}^b(h)$ using the new set of data $\left(X_i, Y_i^b\right)$. Let $\hat{\boldsymbol{\beta}}_{X_i}(h) = \sum_{b=1}^B \tilde{\boldsymbol{\beta}}_{X_i}^b(h)/B$.

iv) Then, choose h_d as the global bandwidth for the preliminary derivative estimation:

$$h_d = \text{argmin}_h \frac{1}{n} \sum_{i=1}^n \left\| \tilde{\boldsymbol{\beta}}_{X_i}(h_{LL}) - \hat{\boldsymbol{\beta}}_{X_i}(h) \right\|^2. \tag{3.13}$$

Due to the difficulty of functional derivative estimation, Ferraty and Nagy (2019 [24]) designed the ad hoc bandwidth selector which minimizes the variation of the estimated derivative using h_d from the one using h_{LL} as in (3.13), but doing this ignores the bias introduced by the latter. Future research can focus on developing a more systematic estimator for the functional derivatives.

v) After the estimated derivative $\hat{\boldsymbol{\beta}}_x^P$ is calculated using bandwidth h_d, we plug $\hat{\boldsymbol{\beta}}_x^P$ into d_2 and $\mathbf{d_3}$ in Section 3.3.2 and search for the optimal λ_j's in Eq. 3.11. LOOCV is applied to select the global bandwidth h_r for FLLR-r regression. In addition, since h_d, h_r are all dependent on J, the optimal J^* for FLLR and FLLR-r is determined through the nested LOOCV steps i) to v).

3.4.3 Model Performance Comparison

We simulated 200 Monte Carlo repetitions of model Eq. (3.12), each with $n_T = 100$ training and $n_t = 50$ test cases. To compare estimator performance at different levels of linearity of the regression function m, we implemented multiple models with $a = 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$. Larger a implies stronger nonlinearity.

The candidate cut-off J values for LOOCV ranged from 1 to 15. In addition, for computational convenience, we translated each of the continuous bandwidths h_{LL}, h_d, h_r to a discrete parameter k_h, which is the number of nearest neighbors of x. This technique was adopted from Ferraty et al. (2007 [22]), and it was also applied in Ferraty and Nagy (2019 [24]). The maximum percentage of training cases that can selected as neighbors was set to 70%.

	$a = 0.3$	$a = 0.4$	$a = 0.5$	$a = 0.6$	$a = 0.7$	$a = 0.8$
FLLR	0.376	0.426	0.501	0.616	0.761	0.850
FLLR-r	**0.367**	**0.413**	**0.475**	**0.571**	**0.689**	**0.810**
NW	0.544	0.572	0.616	0.682	0.761	0.858

Table 3.1: Averaged error ratios of prediction by FLLR, FLLR-r, and NW. The optimal result for each a is in bold. FLLR-r has the smallest error ratio in all scenarios.

Table 3.1 records the averaged error ratios of prediction by each of the three methods at each level of nonlinearity, a. Error ratio (ER) is calculated by $\sum_{i=1}^{n_t} \left(Y_i - \hat{m}^* \left(X_i\right)\right)^2 / \sum_{i=1}^{n_t} \left(Y_i - \bar{Y}\right)^2$, where $\hat{m}^* \left(X_i\right)$ is estimated regression on i-th test case by each method, and $\bar{Y}$ is average of Y_i's. ER is essentially the same as the widely used metric for regression, $(1 - R^2)$. As a increases, the overall level of ER increases as well, but FLLR-r always achieves the best performance among the three methods.

Fig. 3.1 and 3.2 summarize, at different a levels, the performance of the three methods by error ratios and their selected bandwidth k_{h_r} for the ridge estimator.

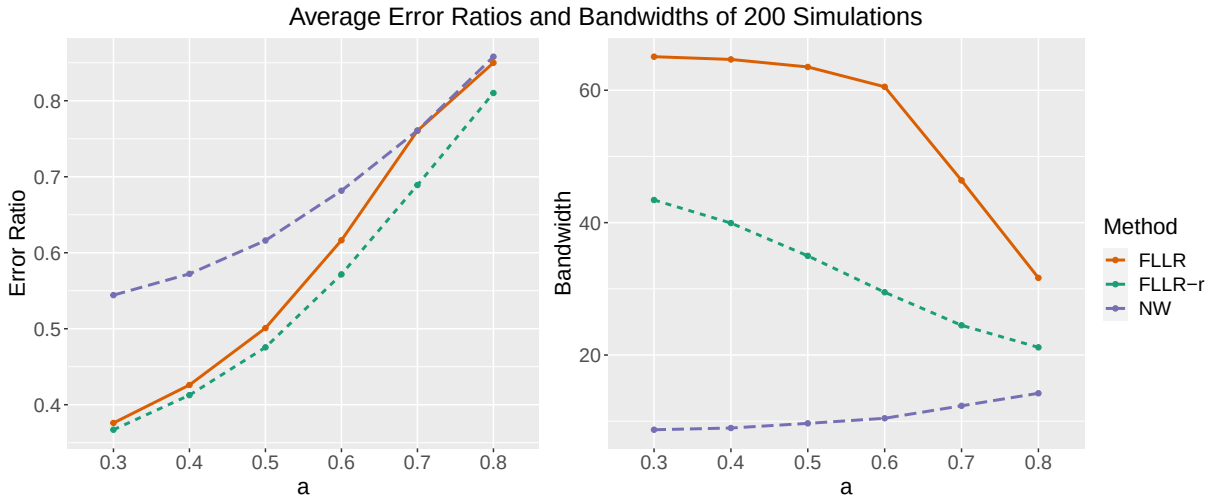

Figure 3.1: Plots of average error ratios (left) and bandwidths (right) by each method for a from 0.3 to 0.6. FLLR-r achieves the lowest ER among the three methods, and uses a smaller bandwidth than FLLR.

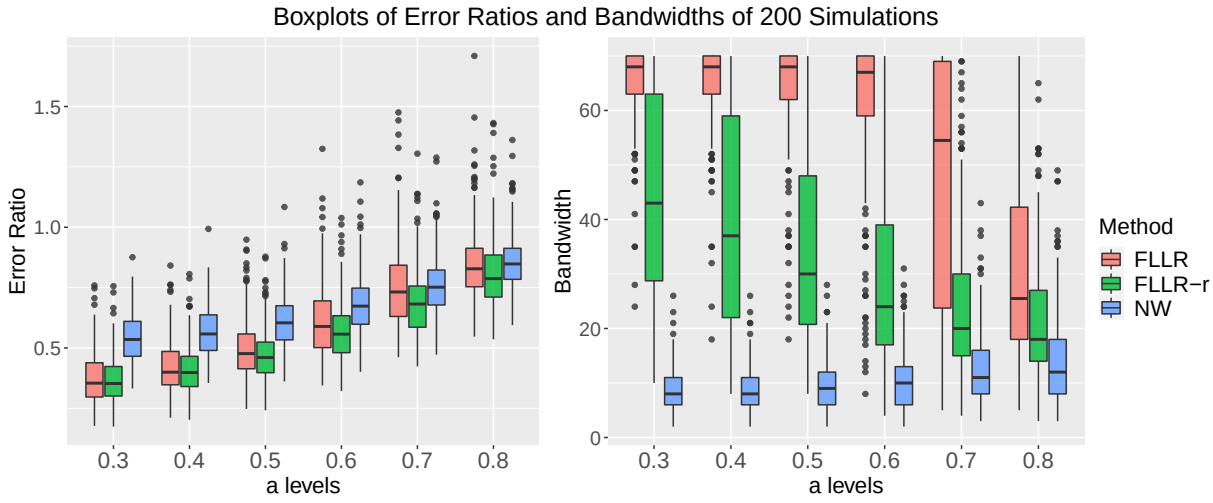

Figure 3.2: Boxplots of simulation error ratios (left) and cross-validated bandwidths (right) for FLLR, FLLR-r and NW at different levels. FLLR-r is advantageous in prediction especially at higher nonlinearity levels, and it needs smaller bandwidth for finite sample data.

According to the plots, FLLR and FLLR-r have very close prediction errors at lower levels of a. However, as the linearity of regression operator decreases with a increasing, the performance of FLLR and FLLR-r diverge. At higher a, FLLR shows larger prediction errors and more outliers than FLLR-r, as seen in the left panel of Fig. 3.2. Also, FLLR-r requires a smaller bandwidth at each level

of a in comparison with FLLR, as the right boxplot of Fig. 3.2 points out. The third quartile of the FLLR-r bandwidth among the 200 simulations is below the first quartile of FLLR's for $a \leq 0.6$. The simulations show that FLLR-r is able to achieve smaller variation that FLLR. Such behavior is consistent with the ridge penalty in multivariate regression, which is known to reduce the variance of estimation while increasing its bias.

In the supplementary material, we include results for derivative estimation by both FLLR and FLLR-r. Derivatives generated by FLLR-r tend to be flatter than FLLR.

3.5 Two Real Data Examples

3.5.1 Particulate Matter (PM) Emission of Heavy Duty Trucks

As our first example, we investigate the relationship between movement patterns of heavy duty trucks and particulate matter (PM) emissions. We use the dataset in McLean et al. (2015 [45]) originally extracted from the Coordinating Research Council E55/59 emissions inventory program documentary (Clark et al. 2007 [12]). The dataset contains 108 records of truck speeds in miles/hour over 90 second intervals, and the logarithms of their PM emission in grams (log PM), captured by 70 mm filters. We convert log PM back to the original PM weight by the exponential transformation. For each of the 200 simulations, the dataset is randomly split into training and test cases by a ratio of $2:1$. Percentage of training cases considered for bandwidth selection is 50%.

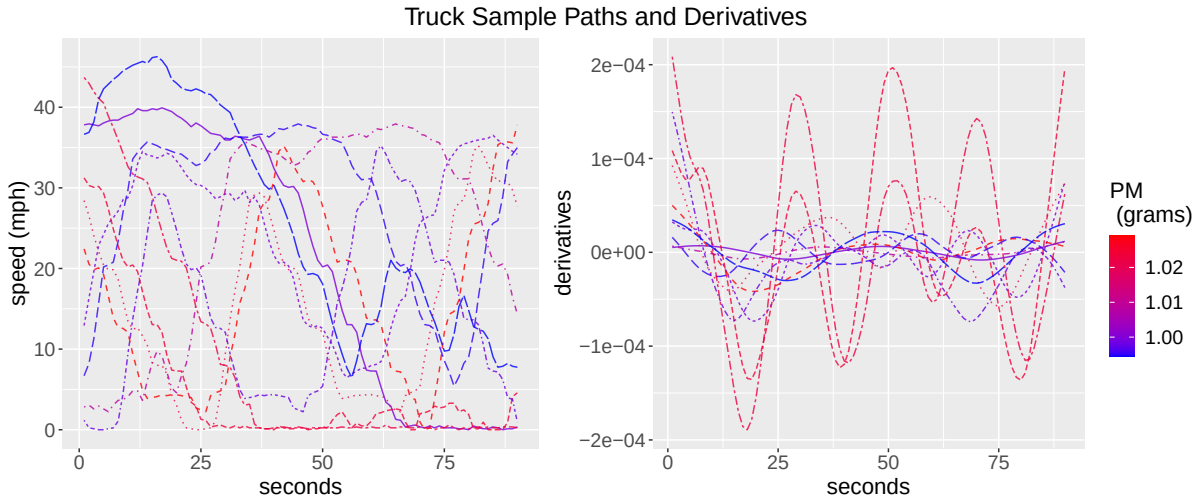

Figure 3.3: Plots of 10 randomly sampled paths (left) and their corresponding estimated derivatives (right). Gradient color scale is used to represent the PM emission related to each sample. Derivatives on the right plot are calculated from estimated derivative scores $\hat{\boldsymbol{\beta}}^P_{X_i}$ (as in Section 3.4.2), $i = 1, \ldots, 10$, applied to the functional basis.

The left panel of Fig. 3.3 shows 10 randomly sampled paths of truck speed, where the gradient color scale corresponds to PM emissions in grams. The right panel includes the estimated derivative functions by FLLR-r for each case, calculated from scores $\hat{\boldsymbol{\beta}}^P_{X_i}$, $i = 1, \ldots, 10$, applied to the functional basis. As the derivatives vary substantially across different records, we can safely infer that the regression function mapping truck movement patterns to PM emissions is nonlinear.

	FLLR	FLLR-r	NW	FLM
Error Ratio (ER)	0.715	**0.652**	0.771	0.862
Mean k_h	30.4	20.3	6.5	N/A

Table 3.2: Averaged error ratios of four models for 200 repetitions and average num of nearest neighbors k_h used by each method are also included.

Table 3.2 shows the averaged error ratios from 200 repetitions by each estimator, as well as the number of nearest neighbors selected by cross validation. An additional estimator, FLM, the functional linear model with scalar ridge penalty in the R package 'fda.usc'([2]), is included. Cross validation is used

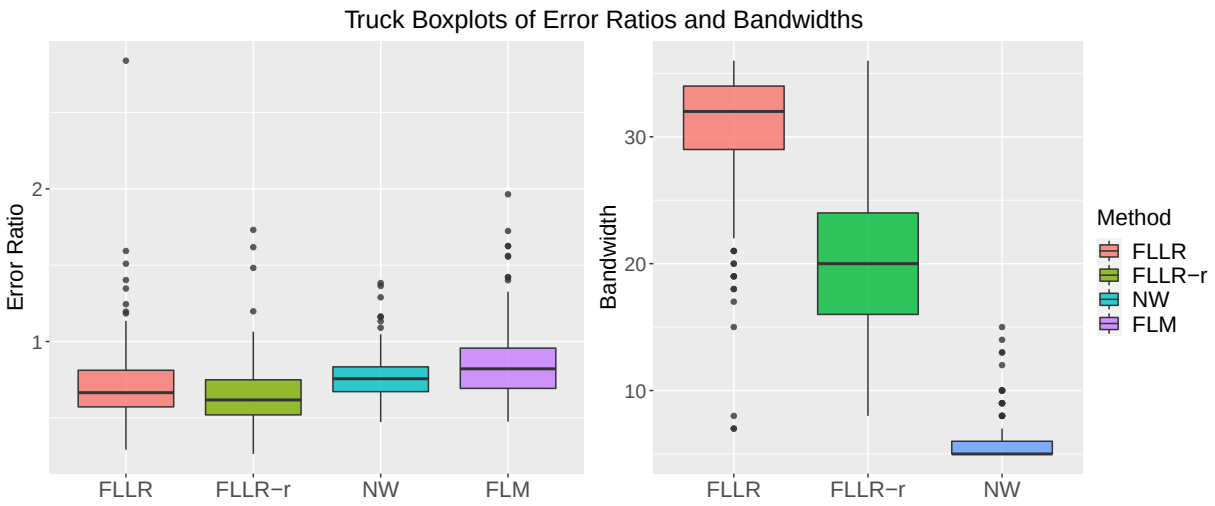

Figure 3.4: Boxplots of error ratios and bandwidths of the three methods for estimated regression of PM emission on truck speed.

for its ridge parameter tuning. We see that FLLR-r is able to achieve the lowest error ratio and uses fewer nearest neighbors than FLLR. Average cut-off J^* for FLLR and FLLR-r are 9.98 and 11 respectively. Boxplots in Fig. 3.4 show the advantages of FLLR-r in prediction accuracy and bandwidth choice.

3.5.2 Oil Content in Cargill Corn Samples

The second example uses a data set of 80 corn specimens measured with different NIR spectrometers at wavelengths 1100–2498nm at 2nm intervals. We choose instrument mp5 for analysis here. Oil content in percentage of total corn kernel weight is also recorded. We use FLLR, FLLR-r and NW to examine the regression mapping corn NIR data to oil content. The original data set can be accessed online at `https://eigenvector.com/resources/data-sets`. Again, the dataset is randomly split into training and test cases by a ratio of $2:1$ during each of the 200 simulations. Percentage of training cases considered for bandwidth selection is 50%. As in the previous truck example, Fig. 3.5 shows

10 randomly sampled NIR paths, with derivatives m'_{X_i} estimated by FLLR-r.

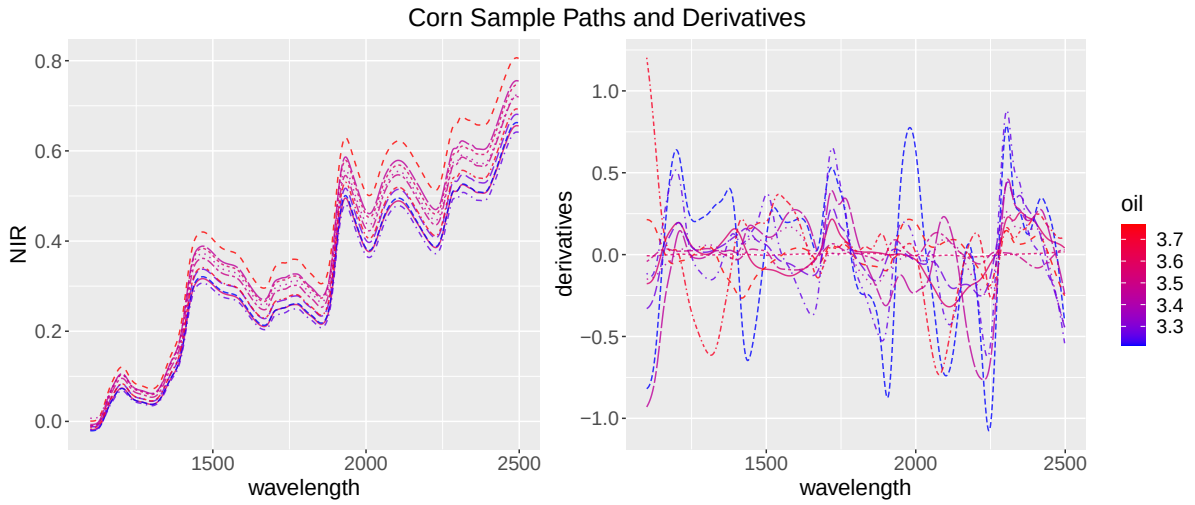

Figure 3.5: Plots of 10 randomly selected corn samples with NIR paths (left) and their corresponding estimated derivatives m'_{X_i} (right). A gradient color scale represents the oil content of each sample.

	FLLR	FLLR-r	NW	FLM
Error Ratio (ER)	0.480	0.421	1.017	**0.378**
Mean k_h	24.4	19.8	12.6	N/A

Table 3.3: Averaged error ratios of four models on corn NIR data, for 200 repetitions. Average numbers of nearest neighbors k_h used by each method are also included.

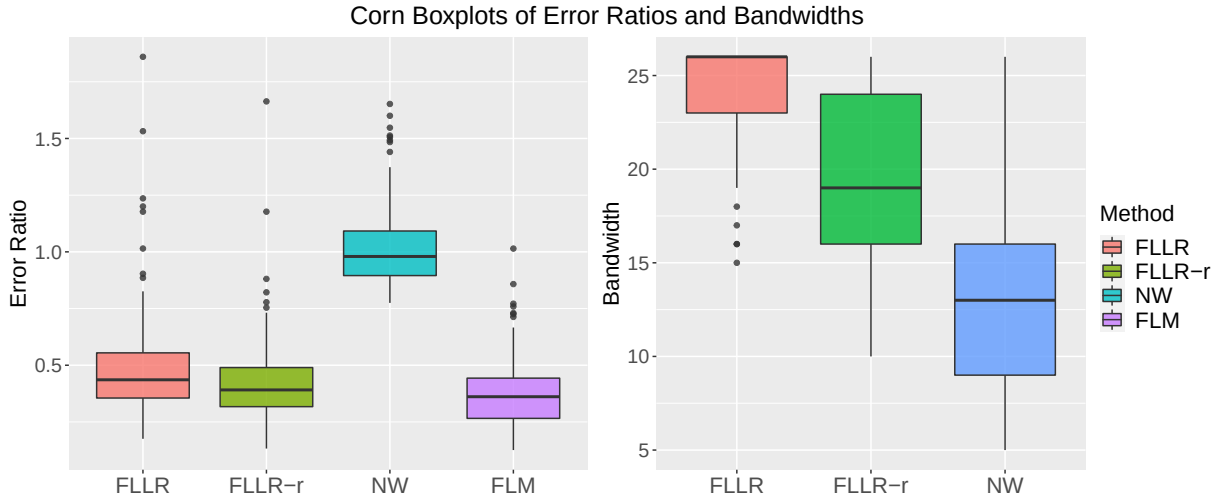

Figure 3.6: Boxplots of error ratios and bandwidths of the three methods for estimated regression of corn oil content on NIR.

Table 3.3 displays the averaged error ratios from 200 repetitions by each es-

timator, as well as the optimal number of nearest neighbors they selected by cross validation. Fig. 3.6 displays the repetition results in boxplots. Due to the relatively higher linearity level compared to the previous example, we see that FLM performs best in this example. However, FLLR-r still generates lowest error ratios for prediction among the three nonparametric methods, and FLLR-r uses a smaller bandwidth than FLLR.

3.6 Discussion

In this work, we extend multidimensional ridge penalization to functional regression and propose a specific type of ridge matrix that adapts to the weighted covariance of the sample scores. We discuss in detail our tuning parameter selector, which is designed to minimize the mean squared error of predictions. Both theoretical results and data analysis show the advantages of this model (FLLR-r), including higher prediction accuracy, especially in regression with higher degree of nonlinearity and a reduction of variance.

Estimation of functional derivatives is another important yet challenging topic in functional data analysis. Fan and Zhang (2000 [21]) discussed estimating derivatives in functional linear models, and Müller et al. (2010 [49]) covered derivative estimation in functional additive models. However, there is relatively little work on nonparametric estimation of functional derivatives. In ongoing work, we are developing further our methodology for functional derivative estimation. Potential directions include a efficient bandwidth selection for estimating derivatives and higher order functional local polynomial models for a more accurate approximation of the first-order differential operators. We antic-

ipate that a well-developed nonparametric derivative estimator can be applied to improve the FLLR-r model for better predictions.

CHAPTER 4

ESTIMATION OF FUNCTIONAL DERIVATIVES

4.1 Introduction

Functional data analysis (FDA), where the target data sets consist of random functions over some continuum, has received increasing attention and popularity in the past decades (Ramsay and Silverman 2005 [51]); Hsing and Eubank, 2015 [32]; Wang et al., 2016 [64]). In particular, there are many works about the scalar-on-function regression, where a model m describes the relationship between a function X from some Hilbert space and a real scalar Y: $Y = m(X) + \epsilon$, ϵ as some random error (see for example Müller and Stadtmüller, 2005 [48]; Cai and Hall, 2006 [7]; Gerthesis et al., 2013 [27]; Reiss et al., 2017 [52]; Ferraty and Nagy 2019 [24]).

We are interested in the functional derivative of the regression model m at the function X, which we denote as m'_X. Properties such as the bounded linearity of m'_X can be found in Chapter 4 of Zeidler (1995 [68]). Similar to its multivariate counterpart, the functional derivative provides a quantitative perception of the relationship between the change in the predictor curve X and the response Y. As a later example in Section 4.4 shows, the derivative functions can also assist in model interpretation in real world scenarios. However, there are only limited past works discussing about the estimation of the functional derivatives, many of which use a parametric framework: Cardot et al. (2003 [9]) studied a B-spline based coefficient function of the functional linear models; Hall and Horowitz discussed estimating derivatives of functional linear regressions through principal component analysis and developed asymp-

totic convergence rates; Chen and Müller (2012) extended conditional quantile analysis to the functional generalized linear models and their derivatives; other works include Fan and Zhang (2000 [21]), James and Silverman (2005 [35]), Yao et al. (2005 [66]).

However, the parametric methods have strict assumptions about the regression model shape, and therefore are limited to describe only certain relationships between function X and scalar Y. For example, the functional linear regressions as discussed above assume constant derivative functions. In this article, we aim to develop a nonparametric, local linear approach to functional derivative estimation that imposes fewer constraints on m and is applicable to a wider range of data scenarios. Nonparametric functional derivative estimation was first brought up in Hall et al. (2009 [30]), where the researchers focused on a kernel-based method. Berlinet et al. (2011 [5]) mentioned functional derivatives in the local linear regression, but is purely theoretical. Ferraty and Nagy (2019 [24]) designed an innovative algorithm of wild bootstrapping to select the optimal bandwidth for estimating m'_X. Their approach assigns a pilot estimation of derivatives $\hat{m}'_X$, which is a byproduct of the local linear regression fitting, and then builds a final estimator $\hat{\hat{m}}'_X$ by picking the optimal bandwidth that minimizes the squared distance between $\hat{\hat{m}}'_X$ and $\hat{m}'_X$. While it resolves the lack of optimization criterion in estimating m'_X, this ad-hoc design does not take the error between the pilot estimator $\hat{m}'_X$ and the true m'_X into consideration, which the performance of $\hat{\hat{m}}'_X$ highly depends on.

We propose a more generalized approach. A new nonparametric estimator is constructed for the functional derivatives, where an empirical bias of the estimator is calculated, and tuning parameters like cut-off basis J and band-

width h are selected by minimizing the empirical mean squared error directly. This method is extended from Ruppert (1997 [56]) where the empirical bias is calculated for multivariate nonparametric regressions as well as density estimation in order to select optimal local bandwidths. We adjust the empirical bias from Ruppert (1991 [56]) according to the non-asymptotic bounds of estimated derivatives on functional data, and explore the specific behaviors of the estimator under the infinite dimensional setting. Advantage of this method is demonstrated through simulation study. In addition, this functional derivative estimator shows its practical strength through a comprehensive real world data analysis about the COVID-19 pandemic, which is from one of the authors' work experience in the virtual health industry. Results prove the significance of the functional derivative estimator in regression prediction as well as model interpretation.

In Section 4.2, we introduce the framework of functional derivatives, and its local linear based estimation. Then the empirical bias is calculated as well as the mean squared error. Section 4.3 provides detailed implementation steps of the model, and performance of our method is compared with others in simulation results. A comprehensive study about the daily COVID-19 test case growth and its impact on winter fatality is included in Section 4.4. In the end, we discuss potential future work. Additional asymptotic proofs and detailed calculation steps can be found in the Supplementary Materials.

4.2 Functional Derivatives and An Empirical Estimation

We assume that a set of n i.i.d. samples (X_i, Y_i) are collected from the joint distribution of $(X, Y) \in \mathcal{L}^2(\mathcal{T}) \times \mathbb{R}$, where X is a square integrable random function over a compact interval $\mathcal{T}$, and Y is real valued. Relationship between X and Y is described by the regression model $Y = m(X) + \epsilon$, where $m : \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ is second-order differentiable, $\epsilon \sim N(0, \sigma_e^2)$. Thus for a new function $x \in \mathcal{L}^2(\mathcal{T})$ in a neigborhood of X_i, $i = 1, \ldots, n$, there exists $u_i = \rho x + (1-\rho) X_i$, $\rho \in [0, 1]$ such that

$$m(X_i) = m(x) + m'_x(X_i - x) + \frac{1}{2} m''_{u_i}(X_i - x)^2, \tag{4.1}$$

where by Riesz representation theorem, $m'_x \in \mathcal{L}^2(\mathcal{T})$, and $m'_x(X_i - x) = \langle m'_x, X_i - x \rangle$ is the corresponding inner product. The bounded bilinear functional m''_x maps from $\mathcal{L}^2(\mathcal{T}) \times \mathcal{L}^2(\mathcal{T})$ to $\mathbb{R}$, and $m''_{u_i}(X_i - x)^2$ is m's second-order derivative at u_i, evaluated at $(X_i - x, X_i - x)$. Our target is to estimate the bounded linear functional m'_x. By the Riesz theorem, m'_x can be represented as $\sum_{j=1}^{\infty} \langle m'_x, \phi_j \rangle \phi_j$ for a set of orthonormal basis functions ϕ_j, $j \geq 1$. In this paper we set the basis to be the eigenfunctions of X's covariance: $C(s,t) = \mathrm{cov}(X(s), X(t)) = \sum_{j \geq 1} \theta_j \phi_j(s) \phi_j(t)$, with $\{\theta_j\}_{j \geq 1}$ the nonincreasing eigenvalues.

Therefore, to estimate m'_x, it is sufficient to get the projections of the functional m'_x on ϕ_j's. The projected values $\langle m'_x, \phi_j \rangle$ can be interpreted as the perturbation of the regression m by a small increment at x alongside the direction of ϕ_j, i.e.,

$$\langle m'_x, \phi_j \rangle = \lim_{t \to 0} \{m(x + t\phi_j) - m(x)\} / t. \tag{4.2}$$

Without additional structural assumptions on the regression model, two pri-

mary nonparametric methods are discussed in past works to estimate the derivative functional m'_x: one is a Nadaraya-Watson based estimator proposed by Hall et al. (2009 [30]), which utilizes the interpretation of derivative projections as in Eq. (4.2); the other is studied in the articles of Baíllo and Grané (2009 [1]), Ferraty and Nagy (2019 [24]) etc., which uses the functional local linear regression (FLLR) to estimate $E(Y|x) = m(x)$, with the truncated derivative estimation as a byproduct.

4.2.1 FLLR-based Derivative Estimation

In this article we build our estimator based upon the latter approach. Several assumptions are made in the estimation process:

Assumption A13. *There is a continuously differentiable kernel function $K : \mathbb{R} \to \mathbb{R}^+$ such that $\int K = 1$, and $c_K \mathbb{1}_{[0,1]} \leq K \leq C_K \mathbb{1}_{[0,1]}$ where $C_K \geq c_K > 0$;*

Assumption A14. *Let $\gamma^{p_1,\ldots,p_M}_{j_1,\ldots,j_M}(t) = E(\langle \phi_{j_1}, X_1 - x\rangle^{p_1} \cdots \langle \phi_{j_M}, X_1 - x\rangle^{p_M} | \|X_1 - x\|^{p_1+\cdots+p_M} = t)$, and let $\gamma^{p_1,\ldots,p_M\prime}_{j_1,\ldots,j_M}(t)$ be its derivatives at t, for integers $j_1, \ldots, j_M, p_1, \ldots, p_M \geq 0$, $M \geq 1$. Functions $\gamma^1_{j_1}$, $\gamma^{1,1}_{j_1,j_2}$, ..., $\gamma^{1,\ldots,1}_{j_1,\ldots,j_4}$, $\gamma^2_{j_1}$, $\gamma^{2,1,1}_{j_1,j_2,j_3}$, $\gamma^{2,1,1,1,1}_{j_1,\ldots,j_5}$ and $\gamma^{2,2}_{j_1,j_2}$ are continuously differentiable around 0. Let $\mathbf{\Gamma}$ be the $J \times J$ matrix with (j,k)-th entry as $\gamma^{1,1'}_{j,k}$. The smallest eigenvalue of $\mathbf{\Gamma}$ is positive.*

Assumption A15. *$\forall h > 0, \pi_x(h) = P(\|X - x\| < h) > 0$ for $x \in \mathcal{L}^2(\mathcal{T})$. As $n \to \infty$, $h = h_n \to 0$, $n\pi_x(h) \to \infty$, $J = J(n) \to \infty$. Also, $\tau_{x,h}(s) := \dfrac{\pi_x(hs)}{\pi_x(h)} \to \tau_x(s)$ as $h \to 0$, $s \in (0,1]$, for some nonnegative $\tau_x(s)$.*

A13 and A15 are standard assumptions in nonparametric functional regression (e.g. [1], [23]). A14 is from Ferraty & Nagy (2019 [24]) to depict the distribution of the function X, which is heavily used in the asymptotic proofs.

Let $\phi_1, \phi_2, \ldots \phi_J \in \mathcal{L}^2(\mathcal{T})$ be the truncated set of eigenfunctions of X, $c_{ij} = \langle X_i - x, \phi_j \rangle$, and $\mathbf{C}$ be an $n \times J$ matrix with c_{ij} as (i,j)-th entry. We consider $m'_{x,J}$ which is the derivative m'_x projected onto the first J basis, and let a length J vector $\mathbf{m}'_{x,J} = \{\langle m'_x, \phi_1 \rangle, \ldots, \langle m'_x, \phi_J \rangle\}^T$ be the coefficients of this projection. FLLR estimates $\mathbf{m}'_{x,J}$ by minimizing the weighted sum of squared error

$$\min_{\beta_0, \beta} \sum_{i=1}^{n} \left(Y_i - \beta_0 - \mathbf{c}_i^T \beta\right)^2 \Delta_i,$$

where the kernel weights $\Delta_i = \dfrac{K\left(\|X_i - x\|/h\right)}{E\left\{K\left(\|X_i - x\|/h\right)\right\}}$. The resulting $\hat{\beta}_0$ and $\hat{\beta}$ are respectively estimators of $m(x)$ and $\mathbf{m}'_{x,J}$:

$$\hat{\mathbf{m}}'_{x,J} = \hat{\beta} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix} \left(\frac{1}{n} \mathbf{C}_x^T \mathbf{\Delta} \mathbf{C}_x\right)^{-1} \frac{1}{n} \mathbf{C}_x^T \mathbf{\Delta} \mathbf{Y}, \tag{4.3}$$

with $\mathbf{C}_x = \begin{pmatrix} \mathbf{1} & \mathbf{C} \end{pmatrix}$, $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$, $\mathbf{I}$ the $J \times J$ identity matrix, and $\mathbf{\Delta}$ the $n \times n$ diagonal matrix with Δ_i as entries.

The bandwidth h and cut-off basis J are key tuning parameters determining the performance of the estimated derivative $\hat{m}'_{x,J} = \mathbf{\Phi}^T \hat{\mathbf{m}}'_{x,J}$, with $\mathbf{\Phi} = [\phi_1, \ldots, \phi_J]^T$. As Ferraty and Nagy (2019 [24]) discovered, $\hat{m}'_{x,J}$ tends to select larger h compared to h_{reg}, due to the different convergence rates of $\hat{m}'_{x,J}$ and $\hat{m}(x)$. In FLLR, the two parameters (here denoted as h_{reg} and J_{reg}) for the regression predictor $\hat{m}(x)$ can be selected through cross validation by minimizing the sample MSE, but such sample MSE does not exist for derivative estimation.

Therefore, we build an estimated MSE to provide a clear loss function for estimated derivatives. A new $\tilde{m}'_{x,J^E} = \boldsymbol{\phi}^T \tilde{\mathbf{m}}'_{x,J^E}$ is proposed with corresponding bandwidth and cut-off parameters h^E and J^E which minimize an estimated MSE of $\hat{m}'_x$. The MSE estimation and consequently selection of h^E and J^E follow the EBBS method proposed by Ruppert (1997 [56]) for multivariate non-

parametric regressions, which concentrates on empirical calculation of the non-asymptotic bias term and is adapted here for functional setting, thus denoted as F-EBBS.

4.2.2 Estimated MSE of $\hat{m}'_x$ and F-EBBS

F-EBBS estimates MSE of $\hat{m}'_{x,J}$ through the non-asymptotic analysis of its bias and variance terms conditioned on the sample data $X_1, \ldots, X_n$, where

$$\begin{aligned} MSE_X\left(\hat{m}'_{x,J}\right) &:= E_X\left\|\hat{m}'_{x,J} - m'_{x,J}\right\|^2 \\ &= \left\|E_X\left(\hat{m}'_{x,J}\right) - m'_{x,J}\right\|^2 + E_X\left\|\hat{m}'_{x,J} - E_X\left(\hat{m}'_{x,J}\right)\right\|^2. \end{aligned} \tag{4.4}$$

As later part of the article will show, similar to EBBS, the exact variance $E_X\left\|\hat{m}'_{x,J} - E_X\left(\hat{m}'_{x,J}\right)\right\|^2$ can be estimated with finite sample data directly, and F-EBBS focuses on an empirical estimation of the squared bias $E_X\left\|\hat{m}'_{x,J} - m'_{x,J}\right\|^2$. We have the following theorem:

Theorem 5. *Under assumptions A13 - A15, the squared bias of the derivative estimator $\hat{m}'_{x,J}$ can be written in the form as:*

$$\left\|E_X\left(\hat{m}'_{x,J}\right) - m'_{x,J}\right\|^2 = \left\{c_0 + c_1 h + c_2 h^2\right\}\left(1 + op\left(1\right)\right),$$

where c_0, c_1, c_2 are coefficients dependent on the truncated basis J, the target function $m'_{x,J}$, and $\gamma_{j_1,\ldots,j_M}^{p_1,\ldots,p_M\prime}(0)$ from A14. The supplementary materials provide detailed calculation of these values.

With Theorem 5, F-EBBS estimates the empirical bias following the design of Ruppert (1997 [56]). For a truncated basis J, $\hat{m}'_{x,J,h_k}$ is the derivative estimated by h_k from a grid of bandwidths $\mathcal{H}_1 = \{h_1, \ldots, h_{K_1}\}$, $k = 1, \ldots, K_1$. Then for

each h_k, a set of neighboring bandwidths h_k^b of h_k, $b = 1, \ldots, N_B$ is selected. Their corresponding estimators $\hat{m}'_{x,J,h_k^b}$ are used to fit the model in Eq.(4.5) by ordinary least squares (OLS), in order to estimate the squared bias of $\hat{m}'_{x,J,h_k}$:

$$\left\|\hat{m}'_{x,J,h}\right\|^2 \approx c_0^* + 2\langle \beta, \hat{m}'_{x,J,h} \rangle + c_1 h + c_2 h^2. \tag{4.5}$$

Coefficients c_0^*, c_1, c_2 are scalars, and β is a function spanned by $\phi_1, \ldots, \phi_J$. Eq.(4.5) is derived from Theorem 5, which generates $\hat{c}_1$, $\hat{c}_2$ for c_1, c_2, and $\hat{c}_0 = \hat{c}_0^* + \|\hat{\beta}/2\|^2$ for c_0. Detailed calculation steps are included in the supplementary materials. Consequently, squared bias of $\hat{m}'_{x,J,h_k}$ is estimated by $\hat{c}_0 + \hat{c}_1 h_k + \hat{c}_2 {h_k}^2$.

In addition, exact variance of $\hat{m}'_{x,J,h_k}$ is

$$\begin{aligned} E_X \left\|\hat{m}'_{x,J,h_k} - E_X\left(\hat{m}'_{x,J,h_k}\right)\right\|^2 &= E \int_{\mathcal{T}} \left\{\hat{m}'_{x,J,h_k}(t) - E_X\left[\hat{m}'_{x,J,h_k}(t)\right]\right\}^2 dt \\ &= \int_{\mathcal{T}} \mathrm{var}_X\left(\hat{m}'_{x,J,h_k}(t)\right) dt. \end{aligned} \tag{4.6}$$

The term $\mathrm{var}_X\left(\hat{m}'_{x,J,h_k}(t)\right)$ can be estimated with sample data according to Eq.(4.3), and the trapezoidal rule is applied for the integral approximation in Eq.(4.6).

Hence, $\hat{MSE}\left(\hat{m}'_{x,J,h_k}\right)$ is a sum of estimated squared bias and variance from above. We then use cubic interpolation to fit $\hat{MSE}\left(\hat{m}'_{x,J,h_k}\right)$ over $\mathcal{H}_1$ and interpolate on a finer grid set $\mathcal{H}_2 = \left\{h_1^*, \ldots, h_{K_2}^*\right\}$. The double interpolation is used to generate a smooth MSE function on the bandwidth h for optimization. Note that we select $h_{k-N_1}, \ldots, h_{k+N_2}$ as the neighbors of h_k in empirical bias calculation, $N_B = N_1 + N_2$, and therefore MSE's are estimated for each bandwidth from h_{1+N_1} to $h_{K_1-N_2}$, which also sets the range for $\mathcal{H}_2$. As this set-up reuses derivatives estimated by adjacent bandwidths, we suggest to choose small N_1 and N_2 in order to reduce correlation between MSE estimation. The optimal bandwidth for J

is then the global minimum of empirical MSE: $h_J = \text{argmin}_{h\in\mathcal{H}_2} \hat{MSE}\left(\hat{m}'_{x,J,h}\right)$, and $J^E = \text{argmin}_J \hat{MSE}\left(\hat{m}'_{x,J,h_J}\right)$, $h^E = h_{J^E}$.

4.2.3 Remarks

In the simulation study, we find that the empirical MSE fitted over bandwidths $\mathcal{H}_2$ sometimes has a long tail at the right-hand side, therefore returns an overly large h_J at the global minimum of the curve. We would like to control the size of the bandwidth selected, to meet the assumptions listed above for asymptotic study and to avoid bias inflation. Ruppert (1997 [56]) chose the optimal bandwidth at the first local minimum, but our MSE curve is generally more wiggly. We provide detailed discussions about locally optimal bandwidth selection for bandwidth size control in the Supplementary Materials.

4.3 Simulation

In this section, a comprehensive simulation is designed to compare the strength of the two FLLR-based derivative estimators: F-EBBS and the wild bootstrap selector (WB) from Ferraty and Nagy (2019 [24]), under different linearity levels of the regression model m. Discussion about the estimation performance with respect to J and h is also included.

4.3.1 Data Setup

We build the sample data by the following scheme: the first 201 Fourier basis on $\mathcal{T} = [0, 1]$ are used for curve generation, where $\phi_1(t) = 1$, $\phi_2(t) = \sqrt{2}\cos(2\pi t)$, $\phi_3(t) = \sqrt{2}\sin(2\pi t)$, ..., $\phi_j(t) = \sqrt{2}\cos(j\pi t)$ or $\sqrt{2}\sin((j-1)\pi t)$ for $1 < j \leq 201$ even or odd. So $X_i = \sum_{j=1}^{201} U_{ij}\sqrt{\theta_j}\phi_j$, where eigenvalues $\theta_j = 10/j$, and U_{ij} is standard Gaussian i.i.d. scores. X_i's are observed on 101 equispaced points $t = 0, 0.01, \ldots, 1$ on $\mathcal{T} = [0, 1]$.

The regression model m follows the design from Ferraty and Nagy (2019 [24]) which combines a linear model with an exponential one, and a parameter $a \in [0, 1]$ is used to adjust the level of linearity:

$$m(x) = (1-a)\langle x, \sum_{j=1}^{100}\phi_j\rangle + a\sum_{j=1}^{150}\exp\left(-\langle x, \phi_j\rangle^2\right). \tag{4.7}$$

In addition, an i.i.d. random noise $\epsilon_i \sim N(0, 1)$ is added to the sample data so that $Y_i = m(X_i) + \epsilon_i$. Then the projected derivative score $\langle m'_x, \phi_{j^*}\rangle$ is:

$$\langle m'_x, \phi_{j^*}\rangle = \begin{cases} (1-a) - 2a\langle x, \phi_{j^*}\rangle\exp\left(-\langle x, \phi_{j^*}\rangle^2\right), & \text{when } j^* \leq 100, \\ -2a\langle x, \phi_{j^*}\rangle\exp\left(-\langle x, \phi_{j^*}\rangle^2\right), & \text{when } 100 < j^* \leq 150, \\ 0, & \text{when } j^* > 150. \end{cases} \tag{4.8}$$

We run a total of 500 simulations, each with 200 training and 100 test cases. FLLR with wild bootstrap method is implemented using the R package `fllr` by Ferraty and Nagy (2019 [24]). We project the functions to a subspace of $J = 20$ dimensions. $\mathcal{H}_1$ is a grid of 50 points, equally spaced on the log scale from $(J+4)$-th to 95th percentile of all the distances between training data sorted increasingly. $\mathcal{H}_2$ then takes a finer grid of 100 points, following the discussion in Section 4.2.2. To avoid singularity in equation solving and to control bias, N_1

and N_2 are both set to be the floor integer of $(J/2+3)$. The tuning parameter is $a = 0, 0.1, 0.25, 0.4, 0.5, 0.6, 0.7, 0.85$.

4.3.2 Model Performance

For all test cases, we compare the estimated derivatives, $\hat{m'}_{x,J}^{(WB)}$ (wild bootstrap) and $\hat{m'}_{x,J}^{(E)}$ (F-EBBS), with the true derivatives ${m'}_{x,J}$ by the squared ratio of error to norm:

$$SREN = \frac{\left\| \hat{m'}_{x,J} - {m'}_{x,J} \right\|^2}{\left\| {m'}_{x,J} \right\|^2}.$$

SREN is able to measure the relative error of each method to the truncated target norms. .

Table 4.1 records averaged SREN by each method from 500 simulations, at 8 different a values and $J = 20$. When the model is linear, both methods achieve good performance in estimating the constant derivative function. For $a = 0.1$ to $a = 0.4$, the wild bootstrap method outperforms F-EBBS, while both keep SREN at relatively low levels. With a increasing and the model linearity diminishes, SREN for both methods rise dramatically, due to the difficulty to estimate derivatives of highly nonlinear models. However, F-EBBS constantly shows better results for $a \geq 0.5$, proving its strength under nonlinear situations. It provides a guideline about when to choose each derivative estimator based on the model shape. Figure 4.1 shows 4 boxplots of SREN at $a = 0, 0.25, 0.5, 0.85$ across the 500 Monte Carlo repetitions.

The tuning parameters J and h selected by the two methods are also investigated. We find that F-EBBS constantly requires small bandwidth h, showing its strength in variance control under the finite sample size n.

	a = 0	a = 0.1	a = 0.25	a = 0.4	a = 0.5	a = 0.6	a = 0.7	a = 0.85
$\hat{m'}^{(WB)}_{x,J}$	**0.411**	**0.392**	**0.437**	**0.501**	0.733	0.968	1.194	2.205
$\hat{m'}^{(E)}_{x,J}$	**0.411**	0.432	0.483	0.536	**0.688**	**0.908**	**1.088**	**2.039**

Table 4.1: SREN averaged over 500 simulations for each method under different linearity levels. In general, SREN is monotonically increasing as a grows. Conditioned on the same J, F-EBBS shows a strong performance in derivative estimation of highly nonlinear models.

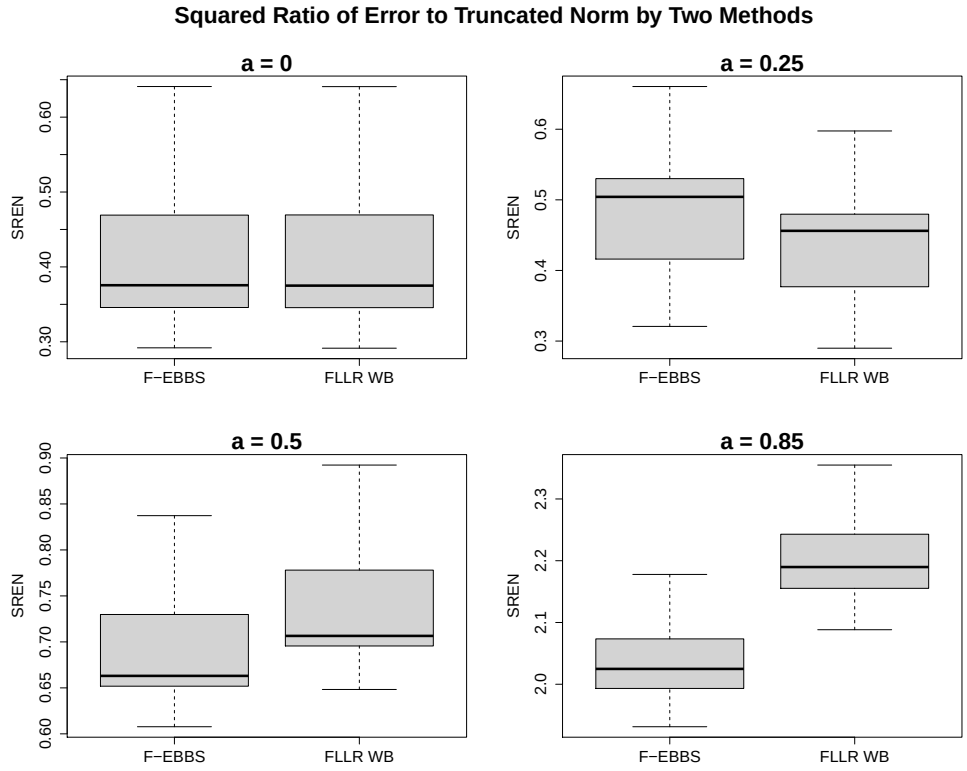

Figure 4.1: Boxplots of SREN by two methods at $a = 0, 0.25, 0.5, 0.85$. Both methods have similar variation when a is small. Distinction between accuracy widens as the model linearity level diminishes.

4.4 COVID-19 Testing Growth Tracking

The practical use of F-EBBS is explored during one of the authors' work experience in the virtual health industry studying the analysis of COVID-19 growth patterns. COVID-19 is an infectious disease caused by SARS-CoV-2 (WHO, 2020

[**?**]), and has led to an ongoing pandemic. An important preventive measure is to frequently test the population with suspected symptoms or high risk of exposure to the virus. Therefore, in this example we investigate the impact of COVID testing growth on the COVID-19 pandemic control using the method of F-EBBS. The dataset are collected and generously shared by The COVID Tracking Project (https://covidtracking.com/), which contain information like the daily new cases of COVID at each state, total population tested, hospitalized and in ICU, etc. The daily data cover the whole year of 2020, and are categorized into five levels based on data quality. We focus on the second and third quarters of the year, i.e., from April to September, 2020, to see how the growth of the testing rate was associated with the COVID death counts in November, 2020, which is a typical flu month. Upper respiratory infections and flu cases have been the most frequent reasons for doctor visits in the U.S., but it remains unclear whether the continuing COVID would change such pattern. Thus, it is of great interest for the virtual health industry to evaluate the severity of the pandemic, especially during the flu season, and optimize the medical resources.

Due to the data quality, a subset of 29 states are selected for model training purpose. For each of the 29 states, the curve X_i records the daily test cases of state i, $i = 1, \ldots, 29$ for 183 consecutive days, while the scalar Y_i denotes the death cases occurred in November, 2020. F-EBBS estimates a derivative curve $\hat{\beta}_i$ for each state by $\hat{\beta}_i = \hat{\mathbf{\Phi}}^T \hat{\mathbf{m}}'^{(E)}_{X_i,J}$, where $\hat{\mathbf{\Phi}} = \{\hat{\phi}_1, \ldots, \hat{\phi}_J\}$ is the list of sample estimated functional principal components. In the left part of Fig.4.2, we plot the daily COVID-19 test cases in ten thousands (10k) for five randomly selected states: CO, GA, MA, MI and NJ, with the gradient color scales corresponding to the numbers of death cases in November, 2020. The right plot contains their estimated derivative functions which are smoothed by local quadratic regressions,

with bandwidths selected by generalized cross validation (GCV). Smoothed derivative estimates are denoted as $\tilde{\beta}_i$, $i = 1, \ldots, 5$.

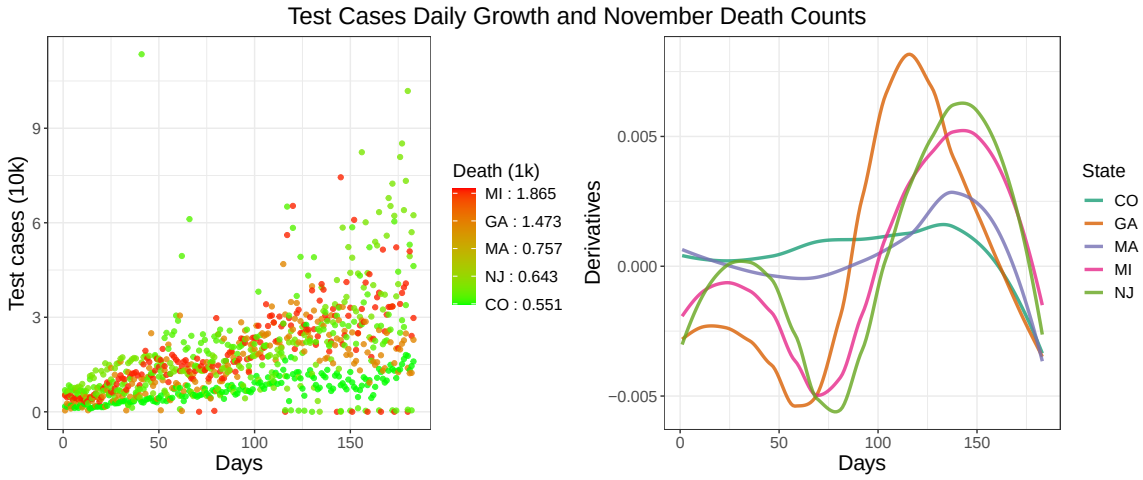

Figure 4.2: Daily COVID-19 test cases (left) in ten thousands (10k) for five randomly selected states: CO, GA, MA, MI and NJ. Gradient color scale is used to represent the different levels of death cases in November, 2020. Correspondent derivative estimates $\hat{\beta}_i$ are smoothed by local quadratic regressions and plotted in the right part.

The derivative plot in Fig.4.2 helps to reflect the relationship between daily test cases and November death counts: there are some upwards and downwards on the derivatives during the early stage, especially for the first 75 days. Due to the relatively small number of test cases, influence of this period is limited on November death counts. Between Days 100 to 150, there was a strong upward trend in the derivative functions, indicating the rising impact of daily test cases on the November COVID-19 fatality. A reasonable interpretation is that at this stage, daily testing was surging due to the widespread virus and therefore people's higher exposure risk to someone with confirmed symptoms, all of which are positively correlated to the fatal rates. On the other hand, after Day 150, the sample derivatives all show a downward pattern, and eventually fall below the horizontal zero line. It shows the number of deaths in November per test case after day 150 is decreasing. A potential explanation is that

the preventative measures were starting to control the outbreak, and growth in test cases helps to reduce fatal cases in the following months. However, further epidemiological analysis is needed to confirm whether such inference is reasonable. Such trend is also presented in the averaged derivative plot over all 29 states in Fig.4.3.

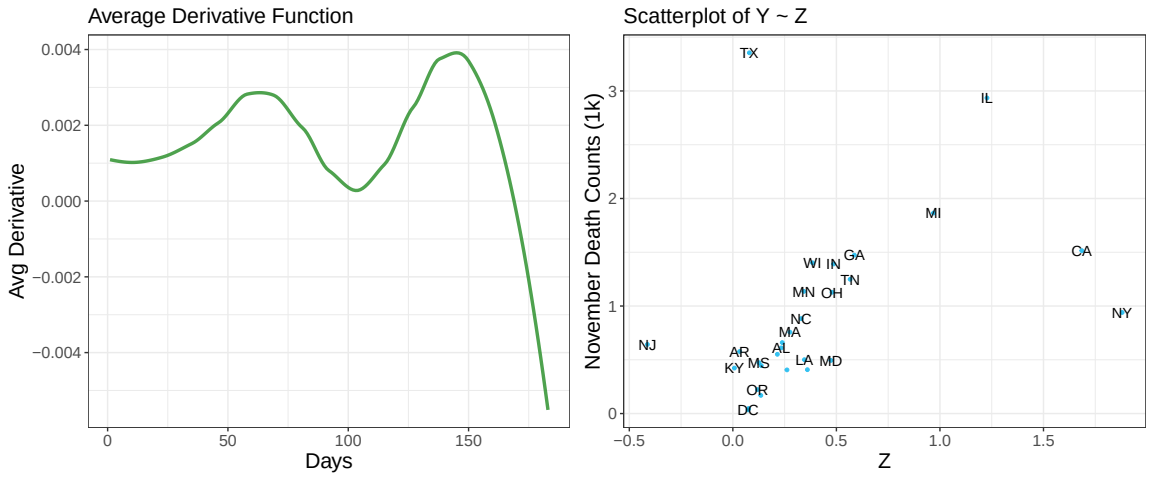

Figure 4.3: Left: the average of derivative function by all 29 states, again smoothed by the local quadratic regression with GCV bandwidth. Right: scatterplot of November death counts (Y_i) versus fitted Z_i scores, $i = 1, \ldots, 29$.

Both Hall et al. (2009 [30]) and Ferraty and Nagy (2019 [24]) investigated the performance of their estimated derivatives using a semiparametric functional single index model on the Berkeley growth data (Tuddenham, 1954 [**?**]). We follow their strategy to interpret further the effect of our estimated derivative functions on the response Y (i.e. November fatal cases). Let $g^* : \mathcal{L}^2(\mathcal{T}) \to \mathbb{R}$ be the functional single index model, and $g : \mathbb{R} \to \mathbb{R}$:

$$Y_i = g^*(X_i) + \epsilon_i = g(\langle X_i, \beta \rangle) + \epsilon_i, \tag{4.9}$$

where X_i and Y_i are as previously defined, and ϵ_i the unknown random error. As mentioned, for each state, a local derivative estimate $\hat{\beta}_i$ is calculated through F-EBBS, with all other states as the training set. Then, the slope β in Eq.4.9 is

estimated as $\hat{\hat{\beta}} = \sum_i \hat{\beta}_i/n$, n as the sample size (Härdle and Stoker, 1989 [**?**]). Fig.4.3 (left) plots $\hat{\hat{\beta}}$ smoothed by the local quadratic regression. In addition, we include a scatterplot of Y_i's versus the fitted scores $Z_i = \langle X_i, \hat{\hat{\beta}} \rangle$ in Fig.4.3 (right). It shows a roughly positive correlation between Y_i and Z_i, which again validates our interpretation in the previous paragraph about how the daily testing cases impact the November fatal counts through the estimated derivatives.

Moreover, we derive the fitted November fatal counts $\hat{Y}_i$'s using the kernel smoother with GCV bandwidth h:

$$\hat{Y}_i = \frac{\sum_{q \neq i} Y_i \cdot K\left(\left(Z_i - Z_q\right)/h\right)}{\sum_{q \neq i} K\left(\left(Z_i - Z_q\right)/h\right)},$$

with a mean squared error (MSE) of 0.45 and correlation between $\hat{Y}_i$ and Y_i as 0.46. The wild bootstrap selector (WB) as discussed in Section 4.3 is also tested on this data set for derivative estimation, which achieves a higher MSE of 0.59 while a slightly lower correlation 0.41. We should also note that for both methods, the small sample size causes unstable functional derivative estimation, especially in the bandwidth selection. Therefore, results discussed in this section would provide more insights if COVID data with good quality are available for the rest of the states.

4.5 Discussion

In this article, we propose a new, nonparametric functional derivative estimator F-EBBS, based on its multivariate version developed by Ruppert (1997 [56]). We are able to provide an empirically calculated bias as well as an exact sample variance of the derivative estimator, which is then used for bandwidth and

cut-off basis selection. The estimator shows good performance in simulation, especially when the regression is highly nonlinear. It also proves its strength in variance control under finite sample sizes. A comprehensive study about COVID-19 daily testing and its effect in winter fatality is conducted, using the functional derivative estimator. Accuracy of the method and its interpretation in real data are included.

As stated earlier, we find that the linearity level is closely related to the performance of F-EBBS. Therefore, it would be helpful to discuss the linearity or shape of the unknown model m when estimating functional derivatives. McLean et al. (2015 [45]) discussed a restricted likelihood ratio test for linearity in the scalar-on-function regressions, which could be helpful for future research. In addition, we would like to explore further about the optimal hyperparameters to use in F-EBBS, such as the sizes of $\mathcal{H}_1$ and $\mathcal{H}_2$. We currently set them to fixed values, but it be would of interest to examine their influence on the estimation performance.

BIBLIOGRAPHY

[1] Amparo Baíllo and Aurea Grané. Local linear regression for functional predictor and scalar response. *Journal of Multivariate Analysis*, 100(1):102–111, 2009.

[2] Manuel Febrero Bande, Manuel Oviedo de la Fuente, Pedro Galeano, Alicia Nieto, Eduardo Garcia-Portugues, and Maintainer Manuel Oviedo de la Fuente. Package 'fda. usc'. *CRAN Repository*, 2020.

[3] J Barrientos-Marin, Frédéric Ferraty, and P26822111327 Vieu. Locally modelled regression and functional data. *Journal of Nonparametric Statistics*, 22(5):617–632, 2010.

[4] Michal Benko, Wolfgang Härdle, and Alois Kneip. Common functional principal components. *The Annals of Statistics*, 37(1):1–34, 2009.

[5] Alain Berlinet, Abdallah Elamine, and André Mas. Local linear regression for functional data. *Annals of the Institute of Statistical Mathematics*, 63(5):1047–1075, 2011.

[6] Eva Boj, Pedro Delicado, and Josep Fortiana. Distance-based local linear regression for functional predictors. *Computational Statistics & Data Analysis*, 54(2):429–437, 2010.

[7] T Tony Cai and Peter Hall. Prediction in functional linear regression. *The Annals of Statistics*, 34(5):2159–2179, 2006.

[8] T Tony Cai, Peter Hall, et al. Prediction in functional linear regression. *The Annals of Statistics*, 34(5):2159–2179, 2006.

[9] Hervé Cardot, Frédéric Ferraty, and Pascal Sarda. Spline estimators for the functional linear model. *Statistica Sinica*, pages 571–591, 2003.

[10] Xiaohong Chen and Yanqin Fan. Estimation of copula-based semiparametric time series models. *Journal of Econometrics*, 130(2):307–335, 2006.

[11] Alejandro Cholaquidis, Ricardo Fraiman, Juan Kalemkerian, and Pamela Llop. A nonlinear aggregation type classifier. *Journal of Multivariate Analysis*, 146:269–281, 2016.

[12] Nigel N Clark, Mridul Gautam, W Scott Wayne, Donald W Lyons, Gregory Thompson, and Barbara Zielinska. Heavy-duty vehicle chassis dynamometer testing for emissions inventory, air quality modeling, source apportionment and air toxics emissions inventory. *Coordinating Research Council, incorporated*, 2007.

[13] Ciprian M Crainiceanu, Ana-Maria Staicu, and Chong-Zhi Di. Generalized multilevel functional regression. *Journal of the American Statistical Association*, 104(488):1550–1561, 2009.

[14] Antonio Cuevas, Manuel Febrero, and Ricardo Fraiman. Robust estimation and classification for functional data via projection-based depth notions. *Computational Statistics*, 22(3):481–496, 2007.

[15] Xiongtao Dai, Hans-Georg Müller, and Fang Yao. Optimal bayes classifiers for functional data and density ratios. *Biometrika*, 104(3):545–560, 2017.

[16] Russell Davidson and Emmanuel Flachaire. The wild bootstrap, tamed at last. *Journal of Econometrics*, 146(1):162–169, 2008.

[17] Aurore Delaigle and Peter Hall. Defining probability density for a distribution of random functions. *The Annals of Statistics*, 38(2):1171–1193, 2010.

[18] Aurore Delaigle and Peter Hall. Achieving near perfect classification for

functional data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):267–286, 2012.

[19] Yves Escoufier. *Echantillonnage dans une population de variables aléatoires réelles*. Department de math.; Univ. des sciences et techniques du Languedoc, 1970.

[20] Jianqing Fan. Design-adaptive nonparametric regression. *Journal of the American statistical Association*, 87(420):998–1004, 1992.

[21] Jianqing Fan and J-T Zhang. Two-step estimation of functional linear models with applications to longitudinal data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(2):303–322, 2000.

[22] Frédéric Ferraty, André Mas, and Philippe Vieu. Nonparametric regression on functional data: inference and practical aspects. *Australian & New Zealand Journal of Statistics*, 49(3):267–286, 2007.

[23] Frédéric Ferraty, André Mas, and Philippe Vieu. Nonparametric regression on functional data: inference and practical aspects. *Australian & New Zealand Journal of Statistics*, 49(3):267–286, 2007.

[24] Frédéric Ferraty and Stanislav Nagy. Scalar-on-function local linear regression and beyond. *arXiv preprint arXiv:1907.08074*, 2019.

[25] Frédéric Ferraty and Philippe Vieu. *Nonparametric functional data analysis: theory and practice*. Springer Science & Business Media, 2006.

[26] Christian Genest, Kilani Ghoudi, and L-P Rivest. A semiparametric estimation procedure of dependence parameters in multivariate families of distributions. *Biometrika*, 82(3):543–552, 1995.

[27] Jan Gertheiss, Jeff Goldsmith, Ciprian Crainiceanu, and Sonja Greven. Longitudinal scalar-on-functions regression with application to tractography data. *Biostatistics*, 14(3):447–461, 2013.

[28] Irene Gijbels, Marek Omelka, and Noël Veraverbeke. Multivariate and functional covariates and conditional copulas. *Electronic Journal of Statistics*, 6:1273–1306, 2012.

[29] Jeff Goldsmith, Ciprian M Crainiceanu, Brian Caffo, and Daniel Reich. Longitudinal penalized functional regression for cognitive outcomes on neuronal tract measurements. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3):453–469, 2012.

[30] Peter Hall, Hans-Georg Müller, and Fang Yao. Estimation of functional derivatives. *The Annals of Statistics*, pages 3307–3329, 2009.

[31] Marius Hofert, Ivan Kojadinovic, Martin Maechler, and Jun Yan. *copula: Multivariate Dependence with Copulas*, 2018. R package version 0.999-19.1.

[32] Tailen Hsing and Randall Eubank. *Theoretical foundations of functional data analysis, with an introduction to linear operators*, volume 997. John Wiley & Sons, 2015.

[33] Gareth M James. Generalized linear models with functional predictors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):411–432, 2002.

[34] Gareth M James and Trevor J Hastie. Functional linear discriminant analysis for irregularly sampled curves. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):533–550, 2001.

[35] Gareth M James and Bernard W Silverman. Functional adaptive model estimation. *Journal of the American Statistical Association*, 100(470):565–576, 2005.

[36] Göran Kauermann, Christian Schellhase, and David Ruppert. Flexible copula density estimation with penalized hierarchical b-splines. *Scandinavian Journal of Statistics*, 40(4):685–705, 2013.

[37] Maurice George Kendall. Rank correlation methods. 1948.

[38] William H Kruskal. Ordinal measures of association. *Journal of the American Statistical Association*, 53(284):814–861, 1958.

[39] Bin Li and Qingzhao Yu. Classification of functional data: A segmentation approach. *Computational Statistics & Data Analysis*, 52(10):4790–4800, 2008.

[40] Yehua Li, Naisyin Wang, and Raymond J Carroll. Generalized functional linear models with semiparametric single-index interactions. *Journal of the American Statistical Association*, 105(490):621–633, 2010.

[41] Han Liu, Fang Han, Ming Yuan, John Lafferty, and Larry Wasserman. High-dimensional semiparametric gaussian copula graphical models. *The Annals of Statistics*, 40(4):2293–2326, 2012.

[42] James G MacKinnon. Inference based on the wild bootstrap. In *Seminar presentation given to Carleton University in September*, 2012.

[43] Enno Mammen. Bootstrap and wild bootstrap for high dimensional linear models. *The annals of statistics*, pages 255–285, 1993.

[44] Roy Mashal and Assaf Zeevi. Beyond correlation: Extreme co-movements between financial assets. *Unpublished, Columbia University*, 2002.

[45] Mathew W McLean, Giles Hooker, and David Ruppert. Restricted likelihood ratio tests for linearity in scalar-on-function regression. *Statistics and Computing*, 25(5):997–1008, 2015.

[46] Mathew W McLean, Giles Hooker, Ana-Maria Staicu, Fabian Scheipl, and David Ruppert. Functional generalized additive models. *Journal of Computational and Graphical Statistics*, 23(1):249–269, 2014.

[47] Bjørn-Helge Mevik, Ron Wehrens, and Kristian Hovde Liland. pls: Partial least squares and principal component regression. *R package version*, 2(3), 2011.

[48] Hans-Georg Müller, Ulrich Stadtmüller, et al. Generalized functional linear models. *Annals of Statistics*, 33(2):774–805, 2005.

[49] Hans-Georg Müller and Fang Yao. Additive modelling of functional gradients. *Biometrika*, 97(4):791–805, 2010.

[50] Cristian Preda, Gilbert Saporta, and Caroline Lévéder. Pls classification of functional data. *Computational Statistics*, 22(2):223–235, 2007.

[51] J.Õ. Ramsay and B.W̃. Silverman. *Functional data analysis*. New York: Springer, 2005.

[52] Philip T Reiss, Jeff Goldsmith, Han Lin Shang, and R Todd Ogden. Methods for scalar-on-function regression. *International Statistical Review*, 85(2):228–249, 2017.

[53] Philip T Reiss, David L Miller, Pei-Shien Wu, and Wen-Yu Hua. Penalized nonparametric scalar-on-function regression via principal coordinates. *Journal of Computational and Graphical Statistics*, 26(3):569–578, 2017.

[54] Philip T Reiss and R Todd Ogden. Functional principal component regression and functional partial least squares. *Journal of the American Statistical Association*, 102(479):984–996, 2007.

[55] Fabrice Rossi and Nathalie Villa. Support vector machine for functional data classification. *Neurocomputing*, 69(7-9):730–742, 2006.

[56] David Ruppert. Empirical-bias bandwidths for local polynomial nonparametric regression and density estimation. *Journal of the American Statistical Association*, 92(439):1049–1062, 1997.

[57] David Ruppert and David S Matteson. *Statistics and Data Analysis for Financial Engineering with R examples*. Springer, 2015.

[58] David Ruppert, Simon J Sheather, and Matthew P Wand. An effective bandwidth selector for local least squares regression. *Journal of the American Statistical Association*, 90(432):1257–1270, 1995.

[59] David Ruppert and Matthew P Wand. Multivariate locally weighted least squares regression. *The annals of statistics*, pages 1346–1370, 1994.

[60] Burkhardt Seifert and Theo Gasser. Data adaptive ridging in local polynomial regression. *Journal of Computational and Graphical Statistics*, 9(2):338–360, 2000.

[61] Zuofeng Shang, Guang Cheng, et al. Nonparametric inference in generalized functional linear models. *The Annals of Statistics*, 43(4):1742–1773, 2015.

[62] Simon J Sheather and Michael C Jones. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society: Series B (Methodological)*, 53(3):683–690, 1991.

[63] Yousri Slaoui. Recursive nonparametric regression estimation for independent functional data. *Statistica Sinica*, 30(1):417–37, 2020.

[64] Jane-Ling Wang, Jeng-Min Chiou, and Hans-Georg Müller. Functional data analysis. *Annual Review of Statistics and Its Application*, 3:257–295, 2016.

[65] Chien-Fu Jeff Wu et al. Jackknife, bootstrap and other resampling methods in regression analysis. *the Annals of Statistics*, 14(4):1261–1295, 1986.

[66] Fang Yao, Hans-Georg Müller, and Jane-Ling Wang. Functional linear regression analysis for longitudinal data. *The Annals of Statistics*, pages 2873–2903, 2005.

[67] Ming Yuan. High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, 11(Aug):2261–2286, 2010.

[68] Eberhard Zeidler. *Applied functional analysis: main principles and their applications*, volume 109. Springer Science & Business Media, 1995.

[69] Hongxiao Zhu, Marina Vannucci, and Dennis D Cox. A bayesian hierarchical model for classification with selection of functional predictors. *Biometrics*, 66(2):463–473, 2010.