Statistica Sinica Preprint No: SS-2020-0294	
Title	Modeling Spiky Functional Data With Derivatives of Smooth Functions in Function-on-Function Regression
Manuscript ID	SS-2020-0294
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202020.0294
Complete List of Authors	Ruiyan Luo and Xin Qi
Corresponding Author	Ruiyan Luo
E-mail	rluo@gsu.edu
Notice: Accepted version subject to English editing.	

Modeling spiky functional data with derivatives of smooth functions in function-on-function regression

Ruiyan Luo and Xin Qi

Georgia State University

Abstract: Smoothness penalty is an efficient regularization and dimension reduction tool for functional regression. However, for spiky functional data observed on a dense grid, the coefficient function in functional regression can be spiky and hence the smoothness regularization is inefficient and leads to over-smoothing. In this paper, we propose a novel approach to fit the functional-on-function regression model by viewing the spiky coefficient functions as the derivatives of smooth auxiliary functions. Compared to smoothness regularization or sparsity regularization which are imposed directly on the spiky coefficient function in existing methods, imposing smoothness regularization on the smooth auxiliary functions can more efficiently reduce the dimension and improve the performance of fitted model. With the estimated smooth auxiliary functions and taking derivatives, we can fit the model and make prediction. Simulation studies and real data applications show that compared to the existing methods, the new method can greatly improve model performance when the coefficient function is spiky, and performs similarly well when the coefficient function is smooth.

Key words and phrases: auxiliary function, derivative, function-on-function regression, smoothness regularization, spiky functional data

1. Introduction

The function-on-function (FOF) linear regression model is a useful tool to study the association between functional variables. The past two decades have witnessed the development of methods to fit the FOF model for relatively smooth functional data observed on a moderately sized grid.

With the development of technology, densely observed curves have been collected in different areas, and they usually display complex local features. For example, the spectrum curves contain a number of narrow and high peaks, whereas the electroencephalography time series curves exhibit high local variations over the entire time interval. When we apply the FOF model to these spiky curves, assuming the coefficient functions to be smooth is inadequate to capture the association between the complex local features of these curves. In this paper, we allow the coefficient functions to be spiky with various local features. Let $Y(t)$ denote the functional response and $X_j(s)$, $1 \le j \le p$, denote multiple functional predictors. The FOF linear regression model has the form

$$Y(t) = \mathfrak{U}(t) + \int_0^1 X_1(s)\mathfrak{B}_1(s,t)ds + \cdots + \int_0^1 X_p(s)\mathfrak{B}_p(s,t)ds + \varepsilon(t), \quad 0 \le t \le 1, \tag{1.1}$$

where without loss of generality, we assume that the domains of $X_j(s)$ and $Y(t)$ are all $[0,1]$, and the $X_j(s)$ have mean zero. To illustrate our idea, we first focus on the model with a single functional predictor ($p = 1$),

$$Y(t) = \mathfrak{U}(t) + \int_0^1 X(s)\mathfrak{B}(s,t)ds + \varepsilon(t). \tag{1.2}$$

In most literature for the FOF model, such as Ramsay and Dalzell (1991), Besse and Cardot (1996), Yao *et al.* (2005), Scheipl *et al.* (2015), Luo and Qi (2017), and the references therein, both $X(s)$ and $Y(t)$ are relatively smooth, and a key assumption is that the coefficient functions are smooth. These coefficient functions are estimated by smooth basis expansion with smoothness regularization imposed. However, for spiky functional data, this smoothness assumption on coefficients may not be true. Another category of popular methods in functional data analysis (FDA) are based on wavelet transformation. With its ability to cope well with discontinuities or rapid changes in functions, wavelet expansion has been used to functional data with sharp local features, such as Zhao *et al.* (2012) and Reiss *et al.* (2015) for the scalar-on-function linear

regression, and Luo *et al.* (2016) for the FOF linear regression. Typically, these methods first conduct wavelet transformation on the observed predictor and/or response curves, then impose sparsity regularization in the wavelet domain, and finally transform back to the original time domain to get estimates of coefficient functions. The major assumption of wavelet-based methods is that the wavelet coefficient vector of $\mathfrak{B}(s,t)$ is sparse, which implies that $\mathfrak{B}(s,t)$ is smooth except a few possible discontinuities or rapid changes (Nason, 2010). However, this sparsity assumption may not be true for spiky functional data with a large number of peaks or rapid fluctuations spread over the whole time range. Therefore, for FOF model with spiky functional data, both the smoothness assumption on $\mathfrak{B}(s,t)$ and the sparsity assumption on the wavelet coefficient vector of $\mathfrak{B}(s,t)$ can be false, and hence these methods can be inefficient.

Without the smoothness or the sparsity assumption on $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ and different from the existing methods that estimate the coefficient functions directly, we will introduce a new method to fit the FOF model, where the coefficients can be spiky for densely observed functional data. We introduce a novel viewpoint to explore the spiky data—viewing the spiky functions as certain transformation of unknown smooth functions. As it is easier and more efficient to control the smoothness of a smoother function, we propose to first estimate the smooth functions via smooth regularization and then get estimates of the spiky functions by taking the inverse transformation of the smooth functions. The unknown smooth functions play the role of auxiliary variables in this method. In this paper, we utilize the relationship between integration and differentiation, and view the spiky coefficient functions $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ as derivatives of smooth auxiliary functions denoted by $\mu(t)$ and $\beta(s,t)$, respectively. We first estimate $\mu(t)$ and $\beta(s,t)$, and then by taking derivatives, we obtain the estimates of $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ and fit the model. Since $\mu(t)$ and $\beta(s,t)$ are smooth functions, to estimate them, we can impose relatively strong smoothness regularization to achieve efficient dimension reduction.

Specifically, we write the model (1.2) as $Y(t) = D^{d_2}\mu(t) + \int_0^1 X(s)D_s^{d_1}D_t^{d_2}\beta(s,t)ds + \varepsilon(t)$, where D is the differential operator, D_s and D_t are the partial differential operators with respect to s and t, respectively, and the non-negative integers d_1 and d_2 are the orders of differential operators. Functions $\mu(t)$ and $\beta(s,t)$ satisfy the equations $D^{d_2}\mu(t) = \mathfrak{U}(t)$ and $D_s^{d_1}D_t^{d_2}\beta(s,t) = \mathfrak{B}(s,t)$. When $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are not smooth, there are d_1 and d_2 large enough such that their antiderivatives, $\mu(t)$ and $\beta(s,t)$, are smooth functions. Hence, we can estimate $\mu(t)$ and $\beta(s,t)$ using smoothness regularization. Specifically, let $F(t) = \mathfrak{U}(t) + \int_0^1 X(s)\mathfrak{B}(s,t)ds$ denote the true linear regression function of $X(s)$ in model (1.2). We will find appropriate orders d_1 and d_2 and smooth functions $\mu(t)$ and $\beta(s,t)$, such that the linear function $\widetilde{F}(t) = D^{d_2}\mu(t) + \int_0^1 X(s)D_s^{d_1}D_t^{d_2}\beta(s,t)ds$ is a good approximation to $F(t)$. Different orders d_1 and d_2 of derivatives result in different functions $\mu(t)$ and $\beta(s,t)$. Larger values of d_1 and d_2 lead to smoother $\mu(t)$ and $\beta(s,t)$ which allow stronger smoothness regularization and more efficient dimension reduction. But higher order derivatives can increase variations of estimation and reduce the performance of the fitted model. So in practice, we will view d_1 and d_2 as tuning parameters and adaptively choose them to reach a balance. Using two different orders, d_1 and d_2, for the partial derivative of s and t, respectively, we can tackle the situation when $\mathfrak{B}(s,t)$ has different roughness levels along the directions of s and t, respectively. The top plots in Figure 2 illustrate the estimated $\mathfrak{B}(s,t)$ and the estimated $\beta(s,t)$, respectively, for a real spiky functional dataset. The top-left plot shows that the estimated $\mathfrak{B}(s,t)$ has rather coarse surface with lots of spiky peaks, whereas the top-right exhibits a quite smooth estimate of $\beta(s,t)$.

The rest of this paper is organized as follows. In Section 2, we introduce our method for the model (1.2) with one functional predictor, study its theoretical property and propose algorithms for computation. In Sections 3 we extend the method to the general model (1.1) with multiple functional predictors. Simulation studies and real data analysis are provided in Sections 4

and 5, respectively. Proofs of theorems and additional information about computational issues, simulations, and real data analysis are provided in supplementary material.

2. Function-on-function regression with one predictor

A common approach to fit the FOF model (1.2) is to represent $\mathfrak{B}(s,t)$ with basis expansions. Some methods use predetermined basis functions and represent $\mathfrak{B}(s,t)$ as $\sum_{k=1}^{K}\sum_{l=1}^{K} b_{kl}\eta_k(s)\theta_l(t)$, where the $\eta_k(s)$ and $\theta_l(t)$ are prespecified bases such as B-spline or Fourier bases, and the b_{kl} are the corresponding expanding coefficients (Ramsay and Silverman, 2005; Scheipl *et al.*, 2015). Some methods use data-driven basis functions. For example, Yao *et al.* (2005) and Chiou *et al.* (2016), based on the functional principal component analysis (FPCA), represent $\mathfrak{B}(s,t)$ as $\sum_{k=1}^{K}\sum_{l=1}^{K} b_{kl}\eta_k^X(s)\theta_l^Y(t)$, where the $\eta_k^X(s)$ and $\theta_l^Y(t)$ are the eigenfunctions of the covariance functions of $X(s)$ and $Y(t)$, respectively. Luo and Qi (2017) considers all representations of $\mathfrak{B}(s,t)$ of the form $\sum_{k=1}^{K}\varphi_k(s)\zeta_k(t)$, where the $\varphi_k(s)$ and $\zeta_k(t)$ can be any square integrable functions. This is a large family of representations and includes the aforementioned representations as special cases. For example, for the representation based on FPCA, let $\varphi_k^{\mathrm{FPCA}}(s)=\eta_k^X(s)$ and $\zeta_k^{\mathrm{FPCA}}(t)=\sum_{l=1}^{K} b_{kl}\theta_l^Y(t)$, then $\sum_{k=1}^{K}\sum_{l=1}^{K} b_{kl}\eta_k^X(s)\theta_l^Y(t)=\sum_{k=1}^{K}\varphi_k^{\mathrm{FPCA}}(s)\zeta_k^{\mathrm{FPCA}}(t)$ is in this family. Similarly, the tensor product basis representation $\sum_{k=1}^{K}\sum_{l=1}^{K} b_{kl}\eta_k(s)\theta_l(t)$ is also in this family. Among all representations of $\mathfrak{B}(s,t)$ of the form $\sum_{k=1}^{K}\varphi_k(s)\zeta_k(t)$, Luo and Qi (2017) identifies the optimal one for estimating the linear regression function $F(t)=\mathfrak{U}(t)+\int_0^1 X(s)\mathfrak{B}(s,t)ds$.

In this paper, we do not estimate $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ directly. Instead we will find smooth functions $\mu(t)$ and $\beta(s,t)$ such that the linear function, $\widetilde{F}(t)=D^{d_2}\mu(t)+\int_0^1 X(s)D_s^{d_1}D_t^{d_2}\beta(s,t)ds$, generated by their derivatives, is a good estimation of $F(t)$. Since $\beta(s,t)$ is a function defined in a two-dimensional region, we consider the large family of representations introduced above. Specifically, given d_1, d_2, and the number K of components, among all possible rep-

resentations of the form $\beta(s,t) = \sum_{k=1}^{K} \phi_k(s)\xi_k(t)$, we will identify the optimal one, denoted by $\beta_K^{(opt)}(s,t) = \sum_{k=1}^{K} \phi_k^{(opt)}(s)\xi_k^{(opt)}(t)$, so that $\mathfrak{U}(t) + \int_0^1 X(s)D_s^{d_1}D_t^{d_2}\beta_K^{(opt)}(s,t)ds = \mathfrak{U}(t) + \int_0^1 X(s)\{\sum_{k=1}^{K} D^{d_1}\phi_k^{(opt)}(s)D^{d_2}\xi_k^{(opt)}(t)\}ds$ is the best approximation to $F(t)$ among all linear functions of the form $\mathfrak{U}(t) + \int_0^1 X(s)\{\sum_{k=1}^{K} D^{d_1}\phi_k(s)D^{d_2}\xi_k(t)\}ds$, where the $\phi_k(s)$ are arbitrary functions with square integrable derivatives of order d_1, and the $\xi_k(t)$ are arbitrary functions with square integrable derivatives of order d_2.

2.1 Optimal representation for $\beta(s,t)$ for given orders d_1 and d_2

Given the number K of components and the orders of derivatives d_1 and d_2, we call $\beta_K^{(opt)}(s,t) = \sum_{k=1}^{K} \phi_k^{(opt)}(s)\xi_k^{(opt)}(t)$ *an optimal representation* if $\{\phi_k^{(opt)}(s), \xi_k^{(opt)}(t) : 1 \le k \le K\}$ solves

$$\min_{\substack{\phi_k(s),\xi_k(t),\\ 1\le k\le K}} \mathbf{E}\left[\int_0^1 \left(F(t) - \left\{\mathfrak{U}(t) + \int_0^1 X(s)\sum_{k=1}^{K} D^{d_1}\phi_k(s)D^{d_2}\xi_k(t)ds\right\}\right)^2 dt\right], \tag{2.3}$$

where the objective function is the expected integrated squared approximation error to $F(t)$, the minimization is over all possible functions $\phi_k(s)$ with square integrable derivatives of order d_1 and all possible functions $\xi_k(t)$ with square integrable derivatives of order d_2. Two facts about the solutions to (2.3) need to be pointed out. First, even if $\mathfrak{B}(s,t)$ is spiky, there always exist smooth solutions to (2.3) when d_1 and d_2 are large enough. Second, with derivatives involved in (2.3), the solutions to (2.3) are not unique. But for any solution $\{\phi_k^{(opt)}(s), \xi_k^{(opt)}(t) : 1 \le k \le K\}$ to (2.3), $\mathfrak{U}(t) + \int_0^1 X(s)\sum_{k=1}^{K} D^{d_1}\phi_k^{(opt)}(s)D^{d_2}\xi_k^{(opt)}(t)ds$ provides the best approximation to $F(t)$ as defined in (2.3). With these two facts, we will estimate a smooth solution to (2.3) by imposing smoothness penalty. The following theorem provides characterization of the solution to (2.3) which leads to our estimation approach.

Theorem 1. *Let* $\phi_k^{(opt)}(s)$ *and* $\xi_k^{(opt)}(t)$, $1 \le k \le K$, *be any solution to* (2.3).

(a). The $\phi_k^{(opt)}(s)$ *are solutions to the following sequential optimization problems,*

$$\max_{\phi} \int_0^1 \int_0^1 D^{d_1}\phi(s)\mathbf{B}(s,s')D^{d_1}\phi(s')dsds', \quad \text{subject to} \quad \int_0^1 \int_0^1 D^{d_1}\phi(s)\mathbf{\Sigma}(s,s')D^{d_1}\phi(s')dsds' = 1,$$
$$\text{and} \int_0^1 \int_0^1 D^{d_1}\phi(s)\mathbf{\Sigma}(s,s')D^{d_1}\phi_l^{(opt)}(s')dsds' = 0 \quad \text{for all} \quad 1 \le l \le k-1, \tag{2.4}$$

where $\mathbf{B}(s,s') = \int_0^1 \mathbf{E}\left[X(s)F(t)\right]\mathbf{E}\left[F(t)X(s')\right]dt$ *and* $\mathbf{\Sigma}(s,s') = \mathbf{E}\left[X(s)X(s')\right]$ *is the covariance function of* $X(s)$.

(b). As $K \to \infty$, $\mathfrak{U}(t) + \int_0^1 X(s)\sum_{k=1}^K D^{d_1}\phi_k^{(opt)}(s)D^{d_2}\xi_k^{(opt)}(t)ds$ *converges to* $F(t)$ *in terms of mean integrated squared error.*

Based on Theorem 1 (a), we will propose a sample version of the optimization problem (2.4) with smoothness penalty to obtain smooth estimates $\widehat{\phi}_k(s)$. To estimate the $\xi_k(t)$, by Theorem 1 (b), for a large enough K, we have

$$\begin{aligned} Y(t) = F(t) + \varepsilon(t) &\approx \mathfrak{U}(t) + \int_0^1 X(s)\sum_{k=1}^K D^{d_1}\phi_k^{(opt)}(s)D^{d_2}\xi_k^{(opt)}(t)ds + \varepsilon(t), \\ &= D^{d_2}\mu(t) + \sum_{k=1}^K \mathbf{Z}_k D^{d_2}\xi_k^{(opt)}(t) + \varepsilon(t), \end{aligned} \tag{2.5}$$

where $\mathbf{Z}_k = \int_0^1 X(s)D^{d_1}\phi_k^{(opt)}(s)ds$ is a scalar random variable, and we take $\mathfrak{U}(t) = D^{d_2}\mu(t)$ so that $\mu(t)$ is smooth for a large enough d_2. With the estimates $\widehat{\phi}_k(s)$, we can estimate the values of $\mathbf{Z}_k$ for different samples. Then motivated by (2.5), we propose a penalized least squares approach with smoothness penalty to obtain smooth estimates $\widehat{\mu}(t)$ and the $\widehat{\xi}_k(t)$. We provide the details of our estimation procedure in the following section.

2.2 Estimation procedure

Let $\{X_i(s), Y_i(t) : 1 \le i \le n\}$ be a set of independent observations from model (1.2). Let $\overline{Y}(t) = \sum_{i=1}^n Y_i(t)/n$ and $\overline{X}(s) = \sum_{i=1}^n X_i(s)/n$ denote their mean curves, respectively. We propose a two-step procedure to estimate the smooth functions $\widehat{\phi}_k(s)$, $\widehat{\mu}(t)$, and $\widehat{\xi}_k(t)$. First, we

estimate the $\widehat{\phi}_k(t)$ sequentially by solving a sample version of the optimization problem (2.4) with smoothness regularization. Second, we propose a penalized least squares problem with smoothness penalty to estimate $\widehat{\mu}(t)$ and the $\widehat{\xi}_k(t)$.

To get the $\widehat{\phi}_k(t)$, we note that $\mathbf{B}(s,s')$ and $\mathbf{\Sigma}(s,s')$ in (2.4) respectively can be estimated by

$$\widehat{\mathbf{B}}(s,s') = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\{X_i(s)-\overline{X}(s)\}\left[\int_0^1\{Y_i(t)-\overline{Y}(t)\}\{Y_j(t)-\overline{Y}(t)\}dt\right]\{X_j(s')-\overline{X}(s')\},$$
$$\widehat{\mathbf{\Sigma}}(s,s') = \frac{1}{n}\sum_{i=1}^{n}\{X_i(s)-\overline{X}(s)\}\{X_i(s')-\overline{X}(s')\}. \tag{2.6}$$

Then we propose to get $\widehat{\phi}_k(s)$, $1\le k\le K$, by sequentially solving the optimization problem,

$$\begin{aligned}
\max_{\phi}\quad & \frac{\int_0^1\int_0^1 D^{d_1}\phi(s)\widehat{\mathbf{B}}(s,s')D^{d_1}\phi(s')dsds'}{\int_0^1\int_0^1 D^{d_1}\phi(s)\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\phi(s')dsds' + \lambda\left[\int_0^1\phi(s)^2ds+\tau\int_0^1\{D^2\phi(s)\}^2ds\right]},\\
\text{subject to}\quad & \int_0^1\int_0^1 D^{d_1}\phi(s)\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\phi(s')dsds'=1,\\
\text{and}\quad & \int_0^1\int_0^1 D^{d_1}\widehat{\phi}_l(s)\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\phi(s')dsds'=0,\quad \text{for all } 1\le l<k.
\end{aligned} \tag{2.7}$$

Problem (2.7) is a penalized sample version of (2.4) by noticing that the objective function of (2.4) can be written as $\int_0^1\int_0^1 D^{d_1}\phi(s)\mathbf{B}(s,s')D^{d_1}\phi(s')dsds' \Big/ \int_0^1\int_0^1 D^{d_1}\phi(s)\mathbf{\Sigma}(s,s')D^{d_1}\phi(s')dsds'$ due to the first constraint $\int_0^1\int_0^1 D^{d_1}\phi(s)\mathbf{\Sigma}(s,s')D^{d_1}\phi(s')dsds'=1$ in (2.4). In the denominator of the objective function in (2.7), we add the penalty $\lambda\left[\int_0^1\phi(s)^2ds+\tau\int_0^1\{D^2\phi(s)\}^2ds\right]$ which consists of two parts. The first part, $\int_0^1\phi(s)^2ds$, is used to control the magnitude of the estimated function $\widehat{\phi}_k(s)$ in L_2-norm, and guarantees the uniqueness of the solution to (2.7). Indeed, without this term, the solution to (2.7) is not unique when $d_1>0$, because adding a scalar constant to a solution does not change its derivative and the obtained function is still a solution to (2.7). Hence, the first part in the penalty can reduce the estimate variation and improve the performance of the fitted model. The second part, $\int_0^1\{D^2\phi(s)\}^2ds$, controls the smoothness of $\widehat{\phi}_k(s)$. More detailed discussion on the effect of this smoothness penalty is provided after Theorem 2 in Section 2.

With estimates $\widehat{\phi}_1(s), \cdots, \widehat{\phi}_K(s)$, we next calculate the estimates $\widehat{\mu}(t)$ and $\{\widehat{\xi}_k(t)\}$ using a penalized least squares approach motivated by (2.5). Let $z_{ik} = \int_0^1 \{X_i(s) - \overline{X}(s)\} D^{d_1}\phi_k^{(opt)}(s)ds$ denote the i-th centered sample value of the random variable $\mathbf{Z}_k = \int_0^1 X(s)D^{d_1}\phi_k^{(opt)}(s)ds$ defined in (2.5), and $\widehat{z}_{ik} = \int_0^1 \{X_i(s) - \overline{X}(s)\} D^{d_1}\widehat{\phi}_k(s)ds$ denote its estimate for $1 \le i \le n$ and $1 \le k \le K$. By (2.5), we regress $Y_i(t)$ on $\{\widehat{z}_{ik} : 1 \le k \le K\}$ to calculate $\widehat{\mu}(t)$ and the $\widehat{\xi}_k(t)$ by solving the penalized least squares problem

$$\min_{\substack{\mu(t),\\ \xi_1(t),\cdots,\xi_K(t)}} \left[\frac{1}{n}\sum_{i=1}^{n} \int_0^1 \left\{ Y_i(t) - D^{d_2}\mu(t) - \sum_{k=1}^{K} \widehat{z}_{ik} D^{d_2}\xi_k(t) \right\}^2 dt \right.$$
$$\left. + \kappa \int_0^1 \{D^2\mu(t)\}^2 dt + \kappa \sum_{k=1}^{K} \int_0^1 \{D^2\xi_k(t)\}^2 dt \right]. \tag{2.8}$$

The first term in the objective function of (2.8) is the mean integrated squared residuals and the other terms are the smoothness penalties.

With all the estimates, $\widehat{\mu}(t)$, the $\widehat{\phi}_k(s)$ and $\widehat{\xi}_k(t)$, we can calculate the following estimates

$$\widehat{\beta}(s,t) = \sum_{k=1}^{K} \widehat{\phi}_k(s)\widehat{\xi}_k(t), \quad \widehat{\mathfrak{B}}(s,t) = \sum_{k=1}^{K} D_s^{d_1}\widehat{\phi}_k(s) D_t^{d_2}\widehat{\xi}_k(t),$$
$$\widehat{F}(t) = D^{d_2}\widehat{\mu}(t) + \int_0^1 X(s) D_s^{d_1} D_t^{d_2}\widehat{\beta}(s,t)ds.$$

Given a new observed predictor curve $X_{\text{new}}(s)$, we predict the response curve as

$$Y_{\text{pred}}(t) = D^{d_2}\widehat{\mu}(t) + \int_0^1 X_{\text{new}}(s) D_s^{d_1} D_t^{d_2}\widehat{\beta}(s,t)ds. \tag{2.9}$$

Designed to fit the FOF model, this method shares some similarities with Luo and Qi (2017) in that both methods express $\mathcal{B}(s,t)$ as the sum of products of separate functions of s and t. Both of them try to find the optimal expansions by minimizing the mean squared error ((2.3) in this paper and (2.2) in Luo and Qi (2017)) which can be similarly characterized via generalized eigenvalue problems. But these two methods have the following key differences. First, in this paper, we

consider the FOF model for spiky functional data, hence we do not assume that $\mathfrak{B}(s,t)$ is smooth. However, Luo and Qi (2017) is tailored for smooth functional data and makes smoothness assumption on $\mathfrak{B}(s,t)$, which is essential for its two major estimation steps. The method in Luo and Qi (2017) can be inefficient for spiky functional data as illustrated in simulations and real data analysis in Sections 4 and 5. Second, let $\mathfrak{B}^{(opt)}(s,t) = \sum_{k=1}^{K} \varphi_k^{(opt)}(s)\zeta_k^{(opt)}(t)$ denote the optimal decomposition with K components (note that the notations in Luo and Qi (2017) are different). With smoothness assumption on $\mathfrak{B}(s,t)$ which implies that the component functions $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ in the optimal decomposition $\mathfrak{B}^{(opt)}(s,t)$ are also smooth, Luo and Qi (2017) identifies the optimal decomposition in the set $\mathcal{S}_1 = \{\sum_{k=1}^{K} \varphi_k(s)\zeta_k(t) : \varphi_k(s) \text{ and } \zeta_k(t) \text{ are smooth}\}$ and imposes smooth penalties on the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ directly. However, as $\mathfrak{B}(s,t)$ can be spiky in this paper, the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ may not be smooth, and the approach in Luo and Qi (2017) may not be applicable. Instead, we identify $\mathfrak{B}^{(opt)}(s,t) = \sum_{k=1}^{K} D^{d_1}\phi_k^{(opt)}(s)D^{d_2}\xi_k^{(opt)}(t)$ from the set $\mathcal{S}_2 = \{\sum_{k=1}^{K} D^{d_1}\phi_k(s)D^{d_2}\xi_k(t) : \phi_k(s) \text{ and } \xi_k(t) \text{ are smooth}\}$ which is much larger than $\mathcal{S}_1$ and includes both smooth functions and nonsmooth functions. This avoids the smoothness assumption on $\mathfrak{B}(s,t)$, but we can still use smooth regularity on the $\phi_k(s)$ and $\xi_k(t)$ to efficiently reduce dimension. Third, the optimization problems in characterizing the component functions of the optimal decomposition are different. The generalized eigenvalue problem (2.4) characterizes the antiderivatives $\phi_k^{(opt)}(s)$ with derivative operator D^{d_1} involved, whereas the generalized eigenvalue problem in Luo and Qi (2017) characterizes $\varphi_k^{(opt)}(s)$ without any derivative operator involved. Fourth, the asymptotic results in Luo and Qi (2017) essentially rely on the smoothness assumption of $\mathfrak{B}(s,t)$. Our asymptotic results do not need this assumption, and can be applied to much more general situations. Even in some cases where the asymptotic results in Luo and Qi (2017) are applicable, the theorem in this paper can provide smaller upper bounds.

Details are given in Section 2.3.

2.3 Asymptotic results

Luo and Qi (2017) provides the asymptotic results for the estimation of $F(t)$ and the prediction error under the assumption that the optimal decomposition $\mathfrak{B}^{(opt)}(s,t) = \sum_{k=1}^{K} \varphi_k^{(opt)}(s)\zeta_k^{(opt)}(t)$ of $\mathfrak{B}(s,t)$ is smooth, and the results depend on the second derivatives of the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$. For spiky functional data, the smooth assumption and the asymptotic results in Luo and Qi (2017) may not hold. Hence, we need to study the asymptotic properties of the proposed method for spiky data without smoothness assumption on $\mathfrak{B}(s,t)$, and provide the results not depending on the derivatives of the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$.

For the i-th sample predictive curve $X_i(s)$, let $F_i(t) = \mathfrak{U}(t) + \int_0^1 X_i(s)\mathfrak{B}(s,t)ds$ be the corresponding sample curve of $F(t)$, $1 \le i \le n$, and define the vector $\mathbf{F}(t) = (F_1(t), \cdots, F_n(t))^{\mathrm{T}}$. Let $\widehat{\mathbf{F}}(t) = \left(\widehat{F}_1(t), \cdots, \widehat{F}_n(t)\right)^{\mathrm{T}}$ denote the estimate of $\mathbf{F}(t)$, where we have $\widehat{F}_i(t) = D^{d_2}\widehat{\mu}(t) + \sum_{k=1}^{K} \int_0^1 X_i(s) D_s^{d_1}\widehat{\phi}_k(s) D_t^{d_2}\widehat{\xi}_k(t)ds$, and $\widehat{\mu}(t)$, the $\widehat{\phi}_k(s)$ and $\widehat{\xi}_k(t)$ are the estimates described in Section 2.2. We will provide the convergence rate of the estimation error $\widehat{\mathbf{F}}(t) - \mathbf{F}(t)$. In addition, let $X_{\mathrm{new}}(s)$ be a new observed predictor curve and $Y_{\mathrm{new}}(t)$ be the corresponding response curve. This new observation is independent of the data $\{X_i(s), Y_i(t) : 1 \le i \le n\}$ used for estimation. Let $\widehat{Y}_{\mathrm{pred}}(t) = D^{d_2}\widehat{\mu}(t) + \sum_{k=1}^{K} \int_0^1 X_{\mathrm{new}}(s) D_s^{d_1}\widehat{\phi}_k(s) D_t^{d_2}\widehat{\xi}_k(t)ds$ be the predicted response in our approach. We will provide an upper bound for the prediction error $\widehat{Y}_{\mathrm{pred}}(t) - Y_{\mathrm{new}}(t)$.

Let $\|\cdot\|$ denote the L_2 norm of a function and $\|\cdot\|_2$ denote the l_2 norm of a vector. Let σ_k^2 denote the maximum value of the optimization problem (2.4) in Theorem 1, and $\widehat{\sigma}_k^2$ denote the maximum value of the optimization problem (2.7) in our estimation procedure, for $1 \le k \le K$. We assume the following regularity condition that is commonly used in FDA, such as Yao *et al.*

(2005), Delaigle and Hall (2012), etc.

Condition 1. $\mathrm{E}[\|X\|^4] < \infty$, $\mathrm{E}[\|\varepsilon\|^2] < \infty$, and $\sigma_1^2 > \cdots > \sigma_K^2 > 0$.

With derivatives involved in the definition (2.3) of the optimal representation, the $\phi_k^{(opt)}(s)$ and $\xi_k^{(opt)}(t)$ are not uniquely defined. In the following Condition 2, we assume that there exists at least one set of $\phi_k^{(opt)}(s)$ and $\xi_k^{(opt)}(t)$ which are smooth.

Condition 2. There exist a set of $\phi_k^{(opt)}(s)$ and $\xi_k^{(opt)}(t)$, $1 \le k \le K$, such that $\|D^2\phi_k^{(opt)}\| < \infty$ and $\|D^2\xi_k^{(opt)}\| < \infty$ for $1 \le k \le K$.

In the following theorem, we arbitrarily choose and fix a set of $\phi_k^{(opt)}(s)$ and $\xi_k^{(opt)}(t)$ satisfying Condition 2. The different choice of the $\phi_k^{(opt)}(s)$ and $\xi_k^{(opt)}(t)$ does not affect the convergence rates provided in the theorem, but affects the multiplication constants in these convergence rates.

Theorem 2. *Under Conditions 1, 2 and for $0 \le d_1 \le 2$, if we choose $\lambda = C/\sqrt{n}$, $c_\tau \le \tau \le C_\tau$ and $\kappa = C_\kappa/\sqrt{n}$, where C and C_κ are constants large enough, $0 < c_\tau < C_\tau$, and all these constants do not depend on n, then for any $\epsilon > 0$ and any n, there exists an event $\Omega_{n,\epsilon}$ with $P(\Omega_{n,\epsilon}) > 1 - \epsilon$, such that in $\Omega_{n,\epsilon}$, we have*

$$\|\widehat{\phi}_k\|^2 \le H_{k,1}, \quad \|D^2\widehat{\phi}_k\|^2 \le H_{k,2}, \quad \|D^{d_1}\widehat{\phi}_k\|^2 \le H_{k,d_1}, \tag{2.10}$$

$$|\widehat{\sigma}_k^2 - \sigma_k^2| \le \frac{H_{k,3}}{\sqrt{n}}, \quad \|D^{d_2}\widehat{\mu} - \mathfrak{U}\|^2 \le \frac{H_0}{\sqrt{n}}, \quad \|D^{d_2}\widehat{\xi}_k - D^{d_2}\xi_k^{(opt)}\|^2 \le \frac{H_{k,5}}{\sqrt{n}}, \tag{2.11}$$

$$\frac{1}{n}\int_0^1 \|\widehat{\mathbf{F}}(t) - \mathbf{F}(t)\|_2^2 dt \le \frac{M_K}{\sqrt{n}} + 2\sum_{k=K+1}^{\infty}\sigma_k^2, \tag{2.12}$$

$$\mathrm{E}\left[\|\widehat{Y}_{\mathrm{pred}} - Y_{\mathrm{new}}\|^2 \middle| X_i(s), Y_i(t), 1 \le i \le n\right] \le \frac{M_K}{\sqrt{n}} + 2\sum_{k=K+1}^{\infty}\sigma_k^2 + \mathrm{E}\left[\|\varepsilon\|^2\right], \tag{2.13}$$

for all $n \ge n_0(\epsilon)$, where $n_0(\epsilon)$, H_0, $H_{k,i}$ and M_K are all constants only depending on ϵ, C, C_κ, c_τ, C_τ, σ_k^2, $\|\phi_k^{(opt)}\|$, $\|D^2\phi_k^{(opt)}\|$, $\|\xi_k^{(opt)}\|$ and $\|D^2\xi_k^{(opt)}\|$, $1 \le k \le K$, and not depending on n.

The condition $0 \leq d_1 \leq 2$ stems from the penalties in (2.7) and offers practical guidance in choosing d_1. It guarantees that the norm of $D^{d_1}\widehat{\phi}_k$ is bounded as $n \to \infty$. Indeed, with penalties imposed on $\|\phi\|^2$ and $\|D^2\phi\|^2$ in the optimization problem (2.7), we can bound $\|\widehat{\phi}_k\|^2$ and $\|D^2\widehat{\phi}_k\|^2$ when $n \to \infty$, as shown in the first two inequalities in (2.10). Based on the Gagliardo-Nirenberg interpolation inequality, if $0 \leq d_1 \leq 2$, the norm of $D^{d_1}\widehat{\phi}_k$ can also be bounded as $n \to \infty$, which is shown in the third inequality in (2.10). This, together with the third inequality in (2.11), leads to the boundedness of the norm of $\widehat{\mathfrak{B}}(s,t)$ as $n \to \infty$, which is necessary for the good performance of our method in both theory and practice. So in our algorithm, we choose d_1 from $\{0,1,2\}$. If one wants to consider a candidate value of d_1 larger than 2, one needs to replace the second derivative in the penalty term $\tau \int_0^1 \{D^2\phi(s)\}^2 ds$ with a derivative of order at least as high as the upper bound of d_1. For example, if we want to consider $0 \leq d_1 \leq 4$, we need to replace the penalty term by $\tau \int_0^1 \{D^4\phi(s)\}^2 ds$, and then obtain similar results as in Theorem 2.

In the proof of Theorem 2, we obtain the following upper bound ((S.21) in Section S.1.2 of supplementary material) when n is large,

$$\|D^2\widehat{\phi}_1\|^2 \leq \frac{1}{\tau}\left(\frac{c_0c_1}{C}+1\right)\|\phi_1^{(opt)}\|^2 + \left(\frac{c_0c_2}{\tau C}+1\right)\|D^2\phi_1^{(opt)}\|^2, \tag{2.14}$$

where c_0, c_1, c_2 and C are constants not depending on n and the choice of $\phi_1^{(opt)}(s)$. The inequality (2.14) holds for any choice of $\phi_1^{(opt)}(s)$. With a relatively large τ, the first term and $c_0c_2/(\tau C)$ can be small, and $\|D^2\widehat{\phi}_1\|^2$ has almost the same or smaller magnitude than $\|D^2\phi_1^{(opt)}\|^2$. So the estimate $\widehat{\phi}_1(s)$ can have at least the same smoothness level as the smoothest choice of $\phi_1^{(opt)}(s)$. Similar results hold for $\widehat{\phi}_k(s)$ with $k > 1$. Meanwhile, (2.12) shows that the estimated regression function is close to the true regression function $F(t)$. Therefore, for models with spiky coefficient functions, unlike the existing methods which are prone to over-smoothing, our method can impose relatively strong smoothness regularization on the smooth auxiliary functions, and at the same

time, have good model estimation and prediction.

The upper bound of the mean integrated error of $\widehat{\mathbf{F}}(t)$ in (2.12) consists of two terms. The first is due to the estimation error and the second is the truncation error when we only estimate the first K terms in the optimal representation. With the increase of K, the truncation error decreases, but the estimation error will increase since more terms are estimated. A proper choice of K will balance these two types of errors. The upper bound of the prediction error (2.13) has an additional term due to the noise in the new response function. The first inequality in (2.11) shows that $\widehat{\sigma}_k^2$ is a good estimate of σ_k^2, which will be used to choose the number of components K in Section 2.4.3. The second inequality in (2.11) shows that $D^{d_2}\widehat{\mu}$ is a good estimate of the intercept function $\mathfrak{U}$ in the FOF model (1.2).

To compare the asymptotic results in Theorem 2 to those in Luo and Qi (2017), recall that $\mathfrak{B}_K^{(opt)}(s,t) = \sum_{k=1}^K \varphi_k^{(opt)}(s)\zeta_k^{(opt)}(t)$ denotes the optimal decomposition of $\mathfrak{B}(s,t)$ with K components, and $\beta_K^{(opt)}(s,t) = \sum_{k=1}^K \phi_k^{(opt)}(s)\xi_k^{(opt)}(t)$ is the solution to the optimization problem (2.3). From (2.3) and the (2.2) in Luo and Qi (2017), we have the following relationships,

$$D^{d_1}\phi_k^{(opt)}(s) = \varphi_k^{(opt)}(s), \quad D^{d_2}\xi_k^{(opt)}(t) = \zeta_k^{(opt)}(t), \quad 1 \le k \le K. \tag{2.15}$$

Although Theorem 2 provides similar asymptotic convergence rates for $\widehat{\mathbf{F}}(t)$ and prediction error as in Theorem 3(a) of Luo and Qi (2017), they have the following important differences. First, Condition 2 and (2.15) imply that in this paper, the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ are only required to belong to the Sobolev space W^1 when $d_1 = d_2 = 1$, and belong to the L_2 space when $d_1 = d_2 = 2$. In Luo and Qi (2017), the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ are required to belong to the Sobolev space W^2. It is well known that $W^2 \subset W^1 \subset L_2$ by the Sobolev embedding theorem (Theorem 4.12 in Adams and Fournier (2003)), and the L_2 space is much larger than the W^2 space. Hence, the asymptotic results in this paper cover much wider situations than those in Luo and Qi (2017).

Especially, when $d_1 = d_2 = 2$ (the $\varphi_k^{(opt)}(s)$ and $\zeta_k^{(opt)}(t)$ belong to L_2), we do not impose any smooth assumptions on $\mathfrak{B}_K^{(opt)}(s,t)$, and hence, the asymptotic results can be applied to the models with any spiky coefficient surfaces. Second, the upper bounds for the estimation error of $F(t)$ and the prediction error can be lower in Theorem 2 than those in Luo and Qi (2017) even though they have the same convergence rates, because the multiplicative constants for these convergence rates are different in these two papers. In Theorem 2 of this paper, the constant M_K depends on and increases with the $\|D^2\phi_k^{(opt)}\|$ and $\|D^2\xi_k^{(opt)}\|$, and in Theorem 3 of Luo and Qi (2017), the corresponding constant depends on and increases with the $\|D^2\varphi_k^{(opt)}\|$ and $\|D^2\zeta_k^{(opt)}\|$. To show the difference, we take the case $d_1 = d_2 = 2$ as an example. By (2.15), the constant M_K in Theorem 2 increases with the $\|D^2\phi_k^{(opt)}\| = \|\varphi_k^{(opt)}\|$ and $\|D^2\xi_k^{(opt)}\| = \|\zeta_k^{(opt)}\|$. For spiky data, the $\varphi_k^{(opt)}$ and $\zeta_k^{(opt)}$ can be spiky, $\|D^2\varphi_k^{(opt)}\|$ and $\|D^2\zeta_k^{(opt)}\|$ may not exist, then in this case, the convergence rates in Luo and Qi (2017) do not hold anymore. Even if $\|D^2\varphi_k^{(opt)}\|$ and $\|D^2\zeta_k^{(opt)}\|$ exist, their values will be large for spiky $\varphi_k^{(opt)}$ and $\zeta_k^{(opt)}$, hence in this situation, the upper bounds in Theorem 3 of Luo and Qi (2017) can be much larger than those in Theorem 2.

2.4 Computation

2.4.1 Solving (2.7)

To solve the optimization problem (2.7), we represent the function $\phi(s)$ using basis expansion. Let $\mathbf{\Gamma}(s) = (b_1(s), \ldots, b_M(s))^{\mathrm{T}}$ be the vector of M basis functions of s. We use B-spline basis functions with equally spaced knots. We represent $\phi(s) = \mathbf{a}^{\mathrm{T}}\mathbf{\Gamma}(s)$, where $\mathbf{a}$ is the M-dimensional vector of expansion coefficients, and convert (2.7) to an optimization problem of $\mathbf{a}$ as follows. We fist consider the objective function of (2.7). The numerator can be expressed as

$$\int_0^1 \int_0^1 D^{d_1}\phi(s)\widehat{\mathbf{B}}(s,s')D^{d_1}\phi(s')dsds' = \mathbf{a}^{\mathrm{T}}\mathbf{\Xi}\mathbf{a}, \tag{2.16}$$

where $\mathbf{\Xi} = \int_0^1 \int_0^1 D^{d_1}\mathbf{\Gamma}(s)\widehat{\mathbf{B}}(s,s')D^{d_1}\mathbf{\Gamma}(s')^{\mathrm{T}}dsds'$ is an $M \times M$ nonnegative definite symmetric matrix, and $D^{d_1}\mathbf{\Gamma}(s)$ is an M-dimensional vector of the d_1-th derivatives of M basis functions in $\mathbf{\Gamma}(s)$. The first term in the denominator of the objective function in (2.7) can be expressed as

$$\int_0^1 \int_0^1 D^{d_1}\phi(s)\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\phi(s')dsds' = \mathbf{a}^{\mathrm{T}}\mathbf{H}\mathbf{a}, \tag{2.17}$$

where $\mathbf{H} = \int_0^1 \int_0^1 D^{d_1}\mathbf{\Gamma}(s)\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\mathbf{\Gamma}(s')^{\mathrm{T}}dsds'$ is an $M \times M$ nonnegative definite symmetric matrix. The penalty term in the denominator of the objective function can be expressed as

$$\int_0^1 \phi(s)^2 ds + \tau \int_0^1 \{D^2\phi(s)\}^2 ds = \mathbf{a}^{\mathrm{T}}(\mathbf{J}_0 + \tau\mathbf{J}_2)\mathbf{a}, \tag{2.18}$$

where $\mathbf{J}_0 = \int_0^1 \mathbf{\Gamma}(s)\mathbf{\Gamma}(s)^{\mathrm{T}}ds$ and $\mathbf{J}_2 = \int_0^1 D^2\mathbf{\Gamma}(s)D^2\mathbf{\Gamma}(s)^{\mathrm{T}}ds$ are $M \times M$ nonnegative definite symmetric matrices. By (2.16)-(2.18), the optimization problem (2.7) can be converted to the following sequential optimization problem of $\mathbf{a}$. Suppose that we have obtained the solutions of the first $k-1$ optimization problems, denoted by $\widehat{\mathbf{a}}_l$, $1 \le l \le k-1$, then the k-th problem is

$$\max_{\mathbf{0}\neq\mathbf{a}\in\mathbb{R}^M} \frac{\mathbf{a}^{\mathrm{T}}\mathbf{\Xi}\mathbf{a}}{\mathbf{a}^{\mathrm{T}}\mathbf{Q}\mathbf{a}}, \quad \text{subject to} \quad \mathbf{a}^{\mathrm{T}}\mathbf{H}\mathbf{a} = 1, \quad \widehat{\mathbf{a}}_l^{\mathrm{T}}\mathbf{H}\mathbf{a} = 0 \text{ for } 1 \le l \le k-1, \tag{2.19}$$

where $\mathbf{Q} = \mathbf{H} + \lambda(\mathbf{J}_0 + \tau\mathbf{J}_2)$ is an $M \times M$ positive definite symmetric matrix. When $k = 1$, we only have the constraint $\mathbf{a}^{\mathrm{T}}\mathbf{H}\mathbf{a} = 1$. The details for solving (2.19) are given in Section S.2.1 of supplementary material. Let $\widehat{\mathbf{a}}_k$ denote the solution to (2.19). Then we have the estimated function $\widehat{\phi}_k(s) = \widehat{\mathbf{a}}_k^{\mathrm{T}}\mathbf{\Gamma}(s)$ and $D^{d_1}\widehat{\phi}_k(s) = \widehat{\mathbf{a}}_k^{\mathrm{T}}D^{d_1}\mathbf{\Gamma}(s)$.

2.4.2 Solving (2.8)

To solve (2.8), we represent $\mu(t)$ and the $\xi_k(t)$ by basis expansions. Let $\mathbf{\Pi}(t) = (d_1(t), \ldots, d_L(t))^{\mathrm{T}}$ be a vector of L basis functions of t. Let $\mu(t) = \mathbf{b}_0^{\mathrm{T}}\mathbf{\Pi}(t)$ and $\xi_k(t) = \mathbf{b}_k^{\mathrm{T}}\mathbf{\Pi}(t)$, $1 \le k \le K$, where the coefficient vectors $\mathbf{b}_k$ are L-dimensional. Then (2.8) can be converted to

$$\min_{\mathbf{b}_0,\cdots,\mathbf{b}_K} \left[\frac{1}{n}\sum_{i=1}^n \int_0^1 \left\{ Y_i(t) - \mathbf{b}_0^{\mathrm{T}}D^{d_2}\mathbf{\Pi}(t) - \sum_{k=1}^K \widehat{z}_{ik}\mathbf{b}_k^{\mathrm{T}}D^{d_2}\mathbf{\Pi}(t) \right\}^2 dt \right.$$

$$+\kappa \int_0^1 \{\mathbf{b}_0^{\mathrm{T}} D^{d_2}\mathbf{\Pi}(t)\}^2 dt + \kappa \sum_{k=1}^{K} \int_0^1 \{\mathbf{b}_k^{\mathrm{T}} D^{d_2}\mathbf{\Pi}(t)\}^2 dt \Bigg], \tag{2.20}$$

which is a convex quadratic optimization problem of the $\mathbf{b}_k$. The explicit solution is given in Section S.2.2 of supplementary material.

2.4.3 Choice of tuning parameters and the number of components

In addition to the tuning parameters λ, τ and κ in the optimization problems (2.7) and (2.8), we also need to determine the two orders d_1 and d_2 of derivatives, the number K of components and the number of basis functions. We first consider the choice of the number of basis functions. Then by viewing d_1, d_2 and K also as tuning parameters, we propose a cross-validation procedure to choose λ, τ, κ, d_1, d_2 and K, simultaneously.

To capture the complicated local features in densely observed spiky functional data, we usually need a large number of basis functions. We choose the (default) number of B-spline basis functions in $\mathbf{\Gamma}(s)$ and $\mathbf{\Pi}(t)$ equal to the number of observation time points in $X_i(s)$ and $Y_i(t)$, respectively. For highly spiky coefficient functions in our simulations, we found that when we reduce the number of basis functions from the default value, the prediction errors increase quickly. On the other hand, when the number of basis functions is increased from our default value, the prediction errors keep the same or are slightly improved. For relative smooth coefficient functions, the number of basis functions can be greatly reduced without impairing the predictive performance of our approach. But as the smoothness level of the coefficient function is unknown in practice, we propose the above default number of basis functions to achieve good predictive performance and computational efficiency.

As discussed after Theorem 2, d_1 cannot exceed the order of derivative used in the smoothness penalty. Since we use $\int_0^1 \{D^2\phi(s)\}^2 ds$ as the smoothness penalty in (2.7), we choose d_1 from

$\{0, 1, 2\}$. Similarly we choose d_2 from $\{0, 1, 2\}$. To take such order derivatives, we need to choose B-spline functions with continuous derivatives up to order two, that is, cubic or higher order splines. Our empirical studies did not find significant improvement of performance using higher order B-splines than cubic splines, so we use cubic splines in our implementation.

Algorithm 1 : CV algorithm

- Partition n samples into five CV sets with roughly the same sizes. The v-th validation set includes $\{X_j^{(\text{valid})}(s), Y_j^{(\text{valid})}(t) : 1 \le j \le N_v\}$ with size N_v, $1 \le v \le 5$.
- **for** $1 \le \ell \le L$ **do**
  - Based on (2.21), use all the data to calculate the upper bound $\widehat{K}^{\ell}_{\text{upp}}$, which depends on the tuning parameters.
  - **for** $1 \le v \le 5$ **do**
    - Use the training set to calculate $\widehat{\phi}_{v,k,\ell}(s)$, $\widehat{\mu}_{v,\ell}(t)$ and $\widehat{\xi}_{v,k,\ell}(t)$ for $1 \le k \le \widehat{K}^{\ell}_{\text{upp}}$.
    - **for** $1 \le k \le \widehat{K}^{\ell}_{\text{upp}}$ **do**
      - Use the first k components and the formula (2.9) to get the predicted response, denoted by $Y^{(\text{pred})}_{j,k,\ell}(t)$, for $X_j^{(\text{valid})}(s)$ in the v-th validation set, $1 \le j \le N_v$.
      - Calculate the CV error $e_{v,k,\ell} = \sum_{j=1}^{N_v} \|Y^{(\text{pred})}_{j,k,\ell} - Y_j^{(\text{valid})}\|^2$.
  - Calculate the total CV error $e_{\text{total},k,\ell} = \sum_{v=1}^{5} e_{v,k,\ell}$ for $1 \le k \le \widehat{K}_{\text{upp}}$.
  - Let $e^{\ell}_{\min} = \min_{1 \le k \le \widehat{K}^{\ell}_{\text{upp}}} e_{\text{total},k,\ell}$ and $K^{\ell}_{\text{opt}} = \text{argmin}_{1 \le k \le \widehat{K}^{\ell}_{\text{upp}}} e_{\text{total},k,\ell}$ respectively be the minimum CV error and the corresponding optimal number of components for the ℓ-th combination of the candidate values of λ, τ, κ, d_1, d_2.
- Let $\ell_{\text{opt}} = \text{argmin}_{1 \le \ell \le L} e^{\ell}_{\min}$, which indexes the optimal combination of tuning parameters λ, τ, κ, d_1, d_2. Then $K^{\ell_{\text{opt}}}_{\text{opt}}$ gives the optimal number of components.

We next propose a cross-validation (CV) procedure to determine all the tuning parameters.

We choose λ and κ from $\{10^{-10}, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 1, 10^2\}$, choose τ, the ratio of the penalty on $\|D^2\phi\|^2$ and $\|\phi\|^2$, from $\{10^{-3}, 10^{-1}, 10, 10^3\}$, and choose d_1 and d_2 from $\{0, 1, 2\}$. Twice denser grids for λ and κ and three times denser grid for τ in a larger range do not lead to obvious improvement of prediction in our simulation. The number K of components can be any positive integer, but in practice, we need to determine an upper bound $\widehat{K}_{\text{upp}}$ and choose K from $\{1, 2, \ldots, \widehat{K}_{\text{upp}}\}$. The calculation of $\widehat{K}_{\text{upp}}$ is motivated by the upper bound in (2.12) of the estimation error of regression function, where the second part, $2\sum_{k=K+1}^{\infty}\sigma_k^2$, is due to truncation after K terms in the optimal expansion. When K is large enough, this part will be small and the upper bound will be dominated by the first term which is due to estimation error and increases with K. Then it is unnecessary to explore larger K. So we choose $\widehat{K}_{\text{upp}}$ such that $\sum_{k=\widehat{K}_{\text{upp}}+1}^{\infty}\sigma_k^2$ is small enough. Since $\widehat{\sigma}_k^2$ is an estimate of σ_k^2 as shown in Theorem 2, we determine $\widehat{K}_{\text{upp}}$ as

$$\widehat{K}_{\text{upp}} = \min\left\{K : \frac{\widehat{\sigma}_K^2}{\widehat{\sigma}_1^2 + \cdots + \widehat{\sigma}_K^2} < 0.001\right\}, \tag{2.21}$$

which is the first K such that $\widehat{\sigma}_K^2$ only accounts for 0.1% of the accumulated sum $\sum_{k=1}^{K}\widehat{\sigma}_k^2$.

We use a five-fold CV procedure (Algorithm 1) to choose tuning parameters, where L denote the number of all possible combinations of the candidate values of λ, τ, κ, d_1, d_2.

3. Function-on-function regression with multiple predictors

To generalize our approach to the FOF model with multiple functional predictors, we will consider smooth auxiliary functions $\mu(t)$ and $\beta_j(s,t)$, $1 \le j \le p$, such that $\mathfrak{U}(t) = D^{d_2}\mu(t)$ and $\mathfrak{B}_j(s,t) = D_s^{d_1}D_t^{d_2}\beta_j(s,t)$, where d_1 and d_2 are nonnegative integers. Moreover, we will consider all representations for $\beta_j(s,t)$ of the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_k(t)$, $1 \le j \le p$, and identify the optimal one in approximating the linear regression function $F(t) = \mathfrak{U}(t) + \sum_{j=1}^{p}\int_0^1 X_j(s)\mathfrak{B}_j(s,t)ds$.

We consider all representations for $\beta_j(s,t)$ of the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_k(t)$, $1 \le j \le p$, rather

than the more general form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{kj}(t)$, where given k, the $\xi_{kj}(t)$ can be different for different j, for the following reasons. First, the expansion $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{kj}(t)$ involves much more functions than $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{k}(t)$, $1\le j\le K$, and hence may require a lot of constraints (e.g. orthogonality of the $\xi_{kj}(t)$) to ensure the stability of estimation. This will increase the difficulty and error of estimation. Second, with the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{k}(t)$, we can characterize the optimal expansion using optimization problems similar to those in Theorem 1, which lead to an efficient estimate procedure. But there is no convenient characterization for the optimal expansion of the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{kj}(t)$. Third, when K is large enough, the approximation error of the optimal expansion of the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{k}(t)$, $1\le j\le p$, is small, and the benefit of considering the more general expansion form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{kj}(t)$, $1\le j\le p$, is limited.

We take the same order partial derivatives, $D_s^{d_1}D_t^{d_2}$, for all $\beta_j(s,t)$, $1\le j\le p$, for the following reasons. First, in the expansions of the form $\sum_{k=1}^{K}\phi_{kj}(s)\xi_{k}(t)$ for $\beta_j(s,t)$, $1\le j\le p$, $\xi_k(t)$ does not depend on j. Hence, we can choose the same order d_2 of partial derivatives with respect to t for all $1\le j\le p$. Second, using different d_1 for different j will greatly increase the number of tuning parameters, which could result in large variations in estimation and heavy computational burden. Third, although the $\mathfrak{B}_j(s,t)$ may have different smoothness levels along s or t, we can always choose d_1 and d_2 large enough so that the $\beta_j(s,t)$ are all smooth functions.

Now we define the optimal representation of the coefficient functions in approximating the linear regression function $F(t)=\mathfrak{U}(t)+\sum_{j=1}^{p}\int_0^1 X_j(s)\mathfrak{B}_j(s,t)ds$. Let $\mathbf{\Phi}_k=(\phi_{k1}(s),\ldots,\phi_{kp}(s))^{\mathrm{T}}$ and $D^{d_1}\mathbf{\Phi}_k=(D^{d_1}\phi_{k1}(s),\ldots,D^{d_1}\phi_{kp}(s))^{\mathrm{T}}$ be the coordinate-wise derivative of $\mathbf{\Phi}_k$ of order d_1. Given d_1, d_2, and K, $\sum_{k=1}^{K}\phi_{kj}^{(opt)}(s)\xi_{k}^{(opt)}(t)$, $1\le j\le p$, is called an optimal representation for $\beta_j(s,t)$ if $\mathbf{\Phi}_k^{(opt)}=(\phi_{k1}^{(opt)}(s),\ldots,\phi_{kp}^{(opt)}(s))^{\mathrm{T}}$ and $\xi_k^{(opt)}(t)$, $1\le k\le K$, is a solution to the following

optimization problem which extends (2.3)

$$\min_{\substack{\mathbf{\Phi}_k(s),\xi_k(t),\\ 1\le k\le K}} \mathbf{E}\left[\int_0^1\left(F(t)-\left\{\mathfrak{U}(t)+\sum_{j=1}^{p}\int_0^1 X_j(s)\sum_{k=1}^{K}D^{d_1}\phi_{kj}(s)D^{d_2}\xi_k(t)ds\right\}\right)^2 dt\right]. \tag{3.22}$$

Analogous to Theorem 1, $\mathbf{\Phi}_k^{(opt)}=(\phi_{k1}^{(opt)}(s),\ldots,\phi_{kp}^{(opt)}(s))^{\mathrm{T}}$ is characterized as the solution to

$$\begin{aligned}
&\max_{\mathbf{\Phi}}\int_0^1\int_0^1 D^{d_1}\mathbf{\Phi}(s)^{\mathrm{T}}\mathbf{B}(s,s')D^{d_1}\mathbf{\Phi}(s')dsds', \qquad (3.23)\\
\text{subject to}\quad &\int_0^1\int_0^1 D^{d_1}\mathbf{\Phi}(s)^{\mathrm{T}}\mathbf{\Sigma}(s,s')D^{d_1}\mathbf{\Phi}(s')dsds'=1,\\
\text{and}\quad &\int_0^1\int_0^1 D^{d_1}\mathbf{\Phi}(s)^{\mathrm{T}}\mathbf{\Sigma}(s,s')D^{d_1}\mathbf{\Phi}_l(s')dsds'=0,\quad \text{for}\quad 1\le l\le k-1,
\end{aligned}$$

where $\mathbf{B}(s,s')$ and $\mathbf{\Sigma}(s,s')$ are both symmetric $p\times p$ matrices with the (j,j') element as $\mathbf{B}_{jj'}(s,s')=\int_0^1 \mathrm{E}\left[X_j(s)F(t)\right]\mathrm{E}\left[F(t)X_{j'}(s')\right]dt$ and $\mathbf{\Sigma}_{jj'}(s,s')=\mathrm{E}\left[X_j(s)X_{j'}(s')\right]$, respectively.

Suppose that we have n independent observations $\{(Y_i(t),X_{i1}(t),\ldots,X_{ip}(t)),1\le i\le n\}$ from model (1.1). The sample versions of $\mathbf{B}(s,s')$ and $\mathbf{\Sigma}(s,s')$ are respectively denoted as $\widehat{\mathbf{B}}(s,s')$ and $\widehat{\mathbf{\Sigma}}(s,s')$ with the (j,j') elements

$$\widehat{\mathbf{B}}_{jj'}(s,s')=\frac{1}{n^2}\sum_{i=1}^{n}\sum_{i'=1}^{n}\{X_{ij}(s)-\overline{X}_j(s)\}\left[\int_0^1\{Y_i(t)-\overline{Y}(t)\}\{Y_{i'}(t)-\overline{Y}(t)\}dt\right]\{X_{i'j'}(s')-\overline{X}_{j'}(s')\},$$

$$\widehat{\mathbf{\Sigma}}_{jj'}(s,s')=\frac{1}{n}\sum_{i=1}^{n}\{X_{ij}(s)-\overline{X}_j(s)\}\{X_{ij'}(s')-\overline{X}_{j'}(s')\},$$

where $1\le j,j'\le p$. Then motivated by (3.23), we propose the following sequential penalized optimization problem to calculate the estimate $\widehat{\mathbf{\Phi}}_k=(\widehat{\phi}_{k1}(s),\cdots,\widehat{\phi}_{kp}(s))^{\mathrm{T}}$ for $1\le k\le K$,

$$\begin{aligned}
&\max_{\mathbf{\Phi}}\ \frac{\int_0^1\int_0^1 D^{d_1}\mathbf{\Phi}(s)^{\mathrm{T}}\widehat{\mathbf{B}}(s,s')D^{d_1}\mathbf{\Phi}(s')dsds}{\int_0^1\int_0^1 D^{d_1}\mathbf{\Phi}(s)^{\mathrm{T}}\widehat{\mathbf{\Sigma}}(s,s')D^{d_1}\mathbf{\Phi}(s')dsds'+\lambda\{\int_0^1\|\mathbf{\Phi}(s)\|_2^2ds+\tau\int_0^1\|D^2\mathbf{\Phi}(s)\|_2^2ds\}}\\
\text{subject to}\quad &\int_0^1\int_0^1\mathbf{\Phi}(s)^{\mathrm{T}}\widehat{\mathbf{\Sigma}}(s,s')\mathbf{\Phi}(s')dsds'=1,\\
\text{and}\quad &\int_0^1\int_0^1\mathbf{\Phi}(s)^{\mathrm{T}}\widehat{\mathbf{\Sigma}}(s,s')\widehat{\mathbf{\Phi}}_l(s')dsds'=0,\quad \text{for}\quad 1\le l\le k-1, \qquad (3.24)
\end{aligned}$$

where $\|\mathbf{\Phi}(s)\|_2^2=\sum_{j=1}^{p}\phi_{kj}^2(s)$ and $\|D^2\mathbf{\Phi}(s)\|_2^2=\sum_{j=1}^{p}\{D^2\phi_{kj}(s)\}^2$ are the squared l_2 norms of the vectors. Let $\widehat{z}_{ik}=\sum_{j=1}^{p}\int_0^1(X_{ij}(s)-\overline{X}_j(s))\widehat{\phi}_{kj}(s)ds$ for $1\le k\le K$ and $1\le i\le n$. The

estimates $\widehat{\mu}(t), \widehat{\xi}_1(t), \cdots \widehat{\xi}_K(t)$ are obtained by solving the same problem as (2.8) in Section 2. Then we can calculate the following estimates

$$\widehat{\beta}_j(s,t) = \sum_{k=1}^{K} \widehat{\phi}_{kj}(s)\widehat{\xi}_k(t), \quad \widehat{\mathfrak{B}_j}(s,t) = \sum_{k=1}^{K} D_s^{d_1}\widehat{\phi}_{kj}(s) D_t^{d_2}\widehat{\xi}_k(t),$$

$$\widehat{F}(t) = D^{d_2}\widehat{\mu}(t) + \sum_{j=1}^{p} \int_0^1 X_j(s) \sum_{k=1}^{K} D_s^{d_1}\widehat{\phi}_{kj}(s) D_t^{d_2}\widehat{\xi}_k(t) ds.$$

Given new observed predictor curves $X_{\text{new},j}(s)$, $1 \le j \le p$, we predict the response curve as

$$Y_{\text{pred}}(t) = D^{d_2}\widehat{\mu}(t) + \sum_{j=1}^{p} \int_0^1 X_{\text{new},j}(s) \sum_{k=1}^{K} D_s^{d_1}\widehat{\phi}_{kj}(s) D_t^{d_2}\widehat{\xi}_k(t) ds.$$

For practical computation, we use the cubic B-spines and choose the number of basis functions and tuning parameters using the same procedure as in Section 2.4.

4. Simulation

We conduct three sets of simulations to assess the performance of the proposed method. The first two focus on FOF models with one functional predictor. We consider coefficient functions with different smoothness levels, from highly spiky to relatively smooth, and also study the effect of the smoothness level of predictors on the performance of our method. In Simulation 3 (supplementary material), we evaluate the performance of proposed method on models with multiple functional predictors.

We compare our new method based on derivatives (*fof.deriv*) with the following methods. The *sSigComp* (Luo and Qi, 2017) estimates $\mathfrak{B}(s,t)$ directly by considering its optimal representation as mentioned in Section 2.1 and imposes smooth penalties. The *wSigComp* (Luo *et al.*, 2016) first conducts wavelet transformation on functional predictors, and then regresses the functional response on the wavelet coefficients with both sparse and smooth penalties imposed. Both *sSigComp* and *wSigComp* are implemented in the `R` package `FRegSigCom`, and to

make the results comparable, we choose the same number of basis functions as in *fof.deriv*. We have also considered the following three methods. The *fdapace* (Yao *et al.*, 2005) implemented in the `R` package `fdapace`, performs the FPCA on both predictor and response curves and uses these eigenfunctions to expand the coefficient kernel function. The *pffr* (Ivanescu *et al.*, 2014) implemented in the package `refund`, uses the tensor product bases to expand $\mathfrak{B}(s,t)$ and fits a penalized additive regression model by the restricted maximum likelihood approach. The *FDboost* (Brockhaus *et al.*, 2017) implemented in `FDBoost`, uses the tensor product bases to expand $\mathfrak{B}(s,t)$ and fits the model by a component-wise gradient boosting algorithm.

For each setting of all the three simulations, we conduct 100 simulation runs and each run has $N_{\text{train}} = 100$ observations as the training data and another $N_{\text{test}} = 500$ independent observations as the test data. All sample curves are defined in $[0,1]$. For each method, we use the training set to select tuning parameters and fit the model, and then apply the fitted model to the test data to estimate the regression function $F(t)$ and calculate the mean integrated squared estimation error $\text{MISEE} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \int_0^1 \left\{ \widehat{F}_i(t) - F_i^{\text{test}}(t) \right\}^2 dt$, where $(X_{i1}^{\text{test}}, \ldots, X_{ip}^{\text{test}})$ is the vector of predictor curves in the i-th sample in the test data, p is the number of predictor curves, $F_i^{\text{test}}(t) = \mathfrak{U}(t) + \sum_{j=1}^{p} \int_0^1 X_{ij}^{\text{test}}(s) \mathfrak{B}_j(s,t) ds$ is the corresponding true regression function, and $\widehat{F}_i(t)$ is its estimate.

4.1 Simulation 1

We generate data from model (1.2) with one functional predictor ($p = 1$) as follows, with specific forms of $\mathfrak{B}_1(s,t)$, $\mathfrak{B}_2(s,t)$, and $\mathfrak{U}_1(t)$ given in Section S.3.1 of supplementary material.

(1). We consider two types of $X(s)$ (Figure S.1 of supplementary material). The first type has wiggly sample curves generated from a Gaussian process with covariance function $\exp\{-2500(s-s')^2\}$, and the second type has smooth sample curves $X(s) = \sum_{k=1}^{10} \{V_{k1}\sin(2k\pi s) + V_{k2}\cos(2k\pi s)\}$,

where $V_{kj} \sim N(0, 1/k^2)$ independently for $1 \le k \le 10$ and $j = 1, 2$.

(2). We consider three types of $\mathfrak{B}(s,t)$, denoted by $\mathfrak{B}_1(s,t) \sim \mathfrak{B}_3(s,t)$ and shown in Figure S.2 of supplementary material, where $\mathfrak{B}_1(s,t)$ and $\mathfrak{B}_2(s,t)$ are highly spiky and generated from triangle or square waves with different frequencies, and $\mathfrak{B}_3(s,t) = e^{-20(s-0.5)^2-20(t-0.5)^2}$ is smooth.

(3). We consider two types of $\mathfrak{U}(t)$, denoted by $\mathfrak{U}_1(t)$ and $\mathfrak{U}_2(t)$ and shown in Figure S.4 of supplementary material, where $\mathfrak{U}_1(t)$ is highly spiky and is a linear combination of square waves with different frequencies, and $\mathfrak{U}_2(t) = \sin(2\pi t)$ is a smooth function.

(4). The random error $\varepsilon(t) \sim N(0, \sigma^2)$ independently for all $0 \le t \le 1$. We consider three noise levels, $\sigma = 0.01$, 0.1, and 1. In each simulation, we scale the coefficient function by a scalar factor such that when $\sigma = 1$, the signal to noise ratio is equal to 1.

The method *wSigComp* needs the fast wavelet transformation which requires that the number of observation points is equal to a power of two. Hence, we choose $T = 2^7$ or $T = 2^9$ observation points equally spaced between 0 and 1 for all sample curves. The smaller number, 2^7, is chosen to run all methods, and their MISEEs and running times from 100 runs are summarized in Tables S.1 and S.2, respectively, of Section S.3.1.1 in supplementary materials. These tables show that the methods *pffr* and *FDboost*, which are designed for small or moderate number of observation points from smooth curves, have much higher MISEEs in all settings (except when both $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are smooth) and need much longer running times than other methods. So for the cases with denser observations, $T = 2^9$, we exclude these two smooth methods and summarize the averages and standard deviations of the MISEEs for the other four methods in Table 1. For the method *fof.deriv*, we summarize the most frequently selected orders of derivatives, d_1 and d_2, in Table S.3, the frequencies of the selected number K_{opt} of components in Figure S.5, and the frequencies of selected tuning parameters, κ, λ, and τ, in Figures S.6∼S.8, Section S.3.1.2

of supplementary material, for $T = 2^9$. The following discussion focuses the results of *fof.deriv*, *sSigComp* and *wSigComp* with $T = 2^9$, as *fdapace* has obviously much higher MISEE than them in all settings. From these tables and figures, we observe the following patterns.

(1). When $\mathfrak{B}(s,t)$ is spiky (Types 1 and 2), the *fof.deriv* has the lowest prediction error in all settings. Especially, when the noise is relatively small ($\sigma = 0.01, 0.1$), *fof.deriv* has a great advantage over the smooth method *sSigComp* and the wavelet-based method *wSigComp* no matter whether $X(s)$ and $\mathfrak{U}(t)$ are smooth or spiky. For example, for Type 1 $\mathfrak{B}(s,t)$ and $\sigma = 0.01$, the average MISEEs of *sSigComp* and *wSigComp* are respectively 36.2 and 7.8 times as high as that of *fof.deriv* when both $X(s)$ and $\mathfrak{U}(t)$ are spiky (Type 1), and 1.9 and 15.4 times as high when both $X(s)$ and $\mathfrak{U}(t)$ are smooth (Type 2). The advantage of *fof.deriv* over the other methods declines when the noise level increases. The *sSigComp* is much more sensitive to the smoothness of $X(s)$ than the other two methods.

(2). When $\mathfrak{U}(t)$, $\mathfrak{B}(s,t)$ and $X(s)$ are all smooth (Type 2 $\mathfrak{U}(t)$, Type 3 $\mathfrak{B}(s,t)$ and Type 2 $X(s)$), the three methods have close average MISEEs with the new method *fof.deriv* and the smooth method *sSigComp* slightly better than the wavelet-based method *wSigComp*. When $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are smooth and $X(s)$ is spiky, *fof.deriv* and *sSigComp* still have close performance and are better than *wSigComp*. When $\mathfrak{B}(s,t)$ is smooth and $\mathfrak{U}(t)$ is spiky, if $\sigma = 0.01, 0.1$, *fof.deriv* is much better than the other two, and if $\sigma = 1$, all the three methods have close performance.

(3). As shown in Table S.3 of supplementary material, when $\mathfrak{B}(s,t)$ is spiky (Types 1 and 2) and the noise is relatively small ($\sigma = 0.01, 0.1$), both selected d_1 and d_2 are nonzero except a few iterations. This indicates that for spiky $\mathfrak{B}(s,t)$, estimating auxiliary smooth functions is more efficient than directly estimating spiky coefficient surface. When the noise is large ($\sigma = 1$), zero d_1 or d_2 is chosen in more iterations. This is because the large noise masks the signal in response

Table 1: Average (and standard deviation) of MISEEs from 100 replicates for Simulation 1 with 2^9 observation time points on each curve.

X	σ	$\mathfrak{U}$	$\mathfrak{B}$	*fof.deriv*	*sSigComp*	*wSigComp*	*fdapace*
1	.01	1	1	$7.12(1.40)\cdot10^{-4}$	$2.58(0.55)\cdot10^{-2}$	$5.52(5.20)\cdot10^{-3}$	$6.61(0.12)\cdot10^{-1}$
			2	$1.69(0.12)\cdot10^{-4}$	$1.29(0.17)\cdot10^{-2}$	$6.52(4.10)\cdot10^{-3}$	$6.53(0.04)\cdot10^{-1}$
			3	$7.61(0.20)\cdot10^{-6}$	$3.11(0.00)\cdot10^{-3}$	$9.60(0.01)\cdot10^{-4}$	$6.45(0.01)\cdot10^{-1}$
		2	1	$7.12(1.27)\cdot10^{-4}$	$2.32(0.56)\cdot10^{-2}$	$4.32(3.00)\cdot10^{-3}$	$2.21(1.36)\cdot10^{-2}$
			2	$1.64(0.13)\cdot10^{-4}$	$1.02(0.20)\cdot10^{-2}$	$5.06(3.53)\cdot10^{-3}$	$1.22(0.33)\cdot10^{-2}$
			3	$1.85(0.33)\cdot10^{-7}$	$1.81(0.35)\cdot10^{-7}$	$5.47(5.18)\cdot10^{-7}$	$3.22(1.19)\cdot10^{-3}$
	0.1	1	1	$1.70(0.12)\cdot10^{-3}$	$2.66(0.57)\cdot10^{-2}$	$7.16(4.16)\cdot10^{-3}$	$6.69(0.10)\cdot10^{-1}$
			2	$1.92(0.11)\cdot10^{-3}$	$1.46(0.18)\cdot10^{-2}$	$1.02(0.81)\cdot10^{-2}$	$6.62(0.04)\cdot10^{-1}$
			3	$2.13(0.19)\cdot10^{-4}$	$3.30(0.03)\cdot10^{-3}$	$1.19(0.03)\cdot10^{-3}$	$6.54(0.01)\cdot10^{-1}$
		2	1	$1.69(0.12)\cdot10^{-3}$	$2.34(0.54)\cdot10^{-2}$	$7.32(7.20)\cdot10^{-3}$	$3.22(1.72)\cdot10^{-2}$
			2	$1.93(0.12)\cdot10^{-3}$	$1.18(0.18)\cdot10^{-2}$	$8.82(7.83)\cdot10^{-3}$	$2.04(0.32)\cdot10^{-2}$
			3	$1.82(0.36)\cdot10^{-5}$	$1.65(0.32)\cdot10^{-5}$	$3.24(1.05)\cdot10^{-5}$	$1.23(0.19)\cdot10^{-2}$
	1	1	1	$4.89(0.35)\cdot10^{-2}$	$1.83(0.37)\cdot10^{-1}$	$5.28(0.39)\cdot10^{-2}$	1.54(0.04)
			2	$1.01(0.07)\cdot10^{-1}$	$1.44(0.15)\cdot10^{-1}$	$1.00(0.06)\cdot10^{-1}$	1.53(0.03)
			3	$2.05(0.19)\cdot10^{-2}$	$2.26(0.18)\cdot10^{-2}$	$2.18(1.96)\cdot10^{-2}$	1.53(0.03)
		2	1	$2.73(0.28)\cdot10^{-2}$	$1.13(0.08)\cdot10^{-1}$	$3.04(0.22)\cdot10^{-2}$	$9.01(0.40)\cdot10^{-1}$
			2	$7.01(0.46)\cdot10^{-2}$	$1.20(0.11)\cdot10^{-1}$	$7.16(0.32)\cdot10^{-2}$	$8.84(0.40)\cdot10^{-1}$
			3	$6.64(2.16)\cdot10^{-4}$	$6.61(2.42)\cdot10^{-4}$	$1.10(0.26)\cdot10^{-3}$	$8.82(0.34)\cdot10^{-1}$
2	1	1	1	$2.44(0.47)\cdot10^{-4}$	$3.57(0.11)\cdot10^{-3}$	$4.67(2.50)\cdot10^{-3}$	$6.54(0.07)\cdot10^{-1}$
			2	$5.64(1.97)\cdot10^{-4}$	$4.50(0.42)\cdot10^{-3}$	$1.72(0.13)\cdot10^{-3}$	$8.94(0.73)\cdot10^{-1}$
			3	$7.57(0.21)\cdot10^{-6}$	$3.11(0.00)\cdot10^{-3}$	$9.63(0.44)\cdot10^{-4}$	$6.43(0.00)\cdot10^{-1}$
		2	1	$2.46(0.64)\cdot10^{-4}$	$4.64(1.20)\cdot10^{-4}$	$3.80(2.84)\cdot10^{-3}$	$1.29(0.58)\cdot10^{-2}$
			2	$5.93(1.72)\cdot10^{-4}$	$1.36(0.33)\cdot10^{-3}$	$1.06(1.78)\cdot10^{-3}$	$2.53(0.62)\cdot10^{-1}$
			3	$1.57(0.27)\cdot10^{-7}$	$1.68(0.28)\cdot10^{-7}$	$2.82(0.59)\cdot10^{-7}$	$1.19(0.03)\cdot10^{-3}$
	1	1	1	$1.25(0.06)\cdot10^{-3}$	$4.51(0.11)\cdot10^{-3}$	$5.90(2.63)\cdot10^{-3}$	$6.57(0.09)\cdot10^{-1}$
			2	$9.57(1.49)\cdot10^{-4}$	$5.00(0.34)\cdot10^{-3}$	$2.52(2.26)\cdot10^{-3}$	$8.82(0.69)\cdot10^{-1}$
			3	$2.09(0.20)\cdot10^{-4}$	$3.30(0.34)\cdot10^{-3}$	$1.18(0.17)\cdot10^{-3}$	$6.45(0.00)\cdot10^{-1}$
		2	1	$1.23(0.06)\cdot10^{-3}$	$1.39(0.08)\cdot10^{-3}$	$4.53(2.36)\cdot10^{-3}$	$1.41(0.76)\cdot10^{-2}$
			2	$9.68(1.41)\cdot10^{-4}$	$1.87(0.28)\cdot10^{-3}$	$1.30(0.83)\cdot10^{-3}$	$2.47(0.68)\cdot10^{-1}$
			3	$1.33(0.27)\cdot10^{-5}$	$1.15(0.24)\cdot10^{-5}$	$1.48(3.73)\cdot10^{-5}$	$3.09(0.00)\cdot10^{-3}$
	0.1	1	1	$7.16(0.42)\cdot10^{-2}$	$7.27(0.41)\cdot10^{-2}$	$7.37(0.43)\cdot10^{-2}$	$0.85(0.02)\cdot10^{-1}$
			2	$3.30(0.25)\cdot10^{-2}$	$4.94(0.70)\cdot10^{-2}$	$3.46(0.38)\cdot10^{-2}$	1.08(0.07)
			3	$2.02(0.17)\cdot10^{-2}$	$2.23(0.17)\cdot10^{-2}$	$2.13(0.17)\cdot10^{-2}$	$8.35(0.11)\cdot10^{-1}$
		2	1	$2.82(0.16)\cdot10^{-2}$	$3.27(0.16)\cdot10^{-2}$	$2.83(0.21)\cdot10^{-2}$	$2.06(0.14)\cdot10^{-1}$
			2	$2.44(0.16)\cdot10^{-2}$	$3.83(0.44)\cdot10^{-2}$	$2.84(0.36)\cdot10^{-2}$	$4.40(0.67)\cdot10^{-1}$
			3	$6.13(1.93)\cdot10^{-4}$	$6.26(2.28)\cdot10^{-4}$	$8.09(2.33)\cdot10^{-4}$	$1.93(0.13)\cdot10^{-1}$

curves and makes it difficult to estimate the complex local variations in $\mathfrak{B}(s,t)$.

(4). When $\mathfrak{B}(s,t)$ is smooth (Type 3) and $\mathfrak{U}(t)$ is smooth (Type 2), both d_1 and d_2 are chosen to be zero in most iterations regardless of the noise level and the smoothness level of $X(s)$. When $\mathfrak{B}(s,t)$ is smooth and $\mathfrak{U}(t)$ is spiky, d_1 is zero in most iterations, but d_2, the order of derivative on t, is always chosen as 1 to capture the local variations caused by the spiky $\mathfrak{U}(t)$.

(5). As shown in Figure S.5 in supplementary material, the selected number of components usually focuses on a single value or two consecutive values. More components are chosen for a spiky $\mathfrak{B}(s,t)$ than for a smooth one. Given $\mathfrak{U}(t)$ and $X(s)$, for a spiky $\mathfrak{B}(s,t)$, less components are selected when the noise level is high ($\sigma = 1$).

(6). As shown in Figure S.6 in supplementary material, the selection of κ for tuning the smoothness of $\mu(t)$ and the $\xi_k(t)$ is quite stable, with a single value selected in most settings. A smaller κ is usually selected when both $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are spiky and σ is smaller. For the tuning parameters λ and τ that respectively controls the magnitude and smoothness of the $\phi_k(s)$, we observe that only one or two values are selected for each of them when both $X(s)$ and $\mathfrak{B}(s,t)$ are spiky. When $X(s)$ or $\mathfrak{B}(s,t)$ is smooth, more variations in the selection of these two tuning parameters are observed. We have also used denser grids $\{10^{-12}, 10^{-11}, \ldots, 10^{2}\}$ for λ, τ and κ simultaneously, and do not observe obvious improvement in prediction.

(7). The average running time is summarized in Table S.4 in supplementary material for the settings with Type 1 $\mathfrak{U}(s)$ (similar for Type 2 $\mathfrak{U}(s)$). The *fof.deriv* is slower than *sSigComp* and *fdapace*, but faster than the wavelet-based method *wSigComp* which involves sparse penalty.

(8). When there are less observation points ($T = 2^7$), as shown in Table S.1 of supplementary material, for $\sigma < 1$, the *fof.deriv* has the lowest MISSEs (when $\mathfrak{U}(t)$ or $\mathfrak{B}(s,t)$ is spiky) or is among the methods with the lowest MISSEs (when both $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are smooth). When

$\sigma = 1$, *fof.deriv* performs similarly with *sSigComp* and/or *wSigComp*, whose MISSEs are lower than other methods when $\mathfrak{U}(t)$ or $\mathfrak{B}(s,t)$ is spiky, and similar with *pffr* and *FDboost* when both $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are smooth.

We also consider predictor curves with observation errors. Let $\widetilde{X}(s) = X(s) + \epsilon_X(s)$ denote the observed predictor, where the observation noise $\epsilon_X(s)$ independently follows $N(0, \sigma_X^2)$ for all $0 \le s \le 1$, and is independent of the true predictor curve $X(s)$. We consider two noise levels: σ_X is equal to 1% or 10% of the square root of the integrated variance of $X(s)$ in $[0,1]$, for all the settings with Type 1 $X(s)$ in Table 1. The results are summarized in Table S.5 of supplementary material. The *fof.deriv* has the lowest MISSEs in all cases except when both $\mathfrak{U}(t)$ and $\mathfrak{B}(s,t)$ are smooth (Type 2 $\mathfrak{U}(t)$ and Type 3 $\mathfrak{B}(s,t)$), where *sSigComp* performs the best and *fof.deriv* has error slightly higher than *sSigComp* but lower than other methods. When the observation error $\epsilon_X(s)$ becomes larger, the *fof.deriv*, *sSigComp* and *wSigComp* tend to have larger MISEEs.

4.2 Simulation 2

We consider the model (1.2) where $\mathfrak{B}(s,t)$ has relatively large values only in a narrow region around the diagonal line $s = t$. This type of $\mathfrak{B}(s,t)$ implies that the association between $X(s)$ and $Y(t)$ quickly declines as $|s-t|$ increases. We generate data as follows.

(1). We consider wiggly sample curves $X(s)$ generated from a Gaussian process with covariance function $\exp\{-2500(s-s')^2\}$. This is the first type of predictor curve in Simulation 1.

(2). We consider two types of $\mathfrak{B}(s,t)$, denoted by $\mathfrak{B}_4(s,t)$ and $\mathfrak{B}_5(s,t)$ and shown in Figure S.9 of supplementary material, where $\mathfrak{B}_4(s,t) = \exp\{-400(s-t)^2\}\cos\{20\pi(s-t)\}$ has a high and narrow ridge along the diagonal line $s = t$ and exponentially decays as $|s-t|$ increases, and $\mathfrak{B}_5(s,t) = \sum_{i=1}^{3} \exp\{-1600(s-c_i)^2 - 1600(t-c_i)^2\}$, where $c_1 = 0.2$, $c_2 = 0.5$ and $c_3 = 0.8$, has

three narrow peaks centered at (0.2,0.2), (0.5,0.5), and (0.8,0.8), along the diagonal line.

(3). We set $\mathfrak{U}(t) = 0$ and generate $\varepsilon(t)$ in the same way as in Simulation 1 with three noise levels, $\sigma = 0.01$, 0.1, and 1. In each simulation, we scale the coefficient function by a scalar factor such that when $\sigma = 1$, the signal to noise ratio is equal to 1.

(4). We consider $T = 2^9$ and 2^{10} equally spaced observation points on each sample curve.

As in Simulation 1, for each setting, we conduct 100 iterations. The MISEEs in 100 iterations are summarized in Table 2, from which we have the following observations.

(1). The new method *fof.deriv* has the lowest average MISEEs in all settings except two cases of $\sigma = 1$, where its average MISEE is slightly larger than the smallest ones. The smooth method *sSigComp* generally has lower error than the wavelet-based method *wSigComp* for the ridge-shaped $\mathfrak{B}_4(s,t)$, whereas for $\mathfrak{B}_5(s,t)$, *wSigComp* is better than *sSigComp*. This is because $\mathfrak{B}_5(s,t)$ only has three isolated peaks, which satisfies the sparsity assumption in the wavelet domain required by the wavelet-based method *wSigComp*. The *fdapace* has slightly higher MISSEs than *fof.deriv* for $\mathfrak{B}_4(s,t)$ when $\sigma = 0.01$, but much higher errors in other settings.

(2). When the observation points get denser (T changes from 2^9 to 2^{10}), the *fof.deriv* has decreased MISEEs in all settings, the *wSigComp* has decreased MISEEs for $\mathfrak{B}_5(s,t)$ where the assumption of sparse wavelet coefficients is satisfied, and the *sSigComp* has obvious reduction in MISEE for large variance ($\sigma = 1$).

(3). Frequencies of the selected order of derivatives (d_1, d_2) are provided in Table S.6 of Section S.3.2 in supplementary material. For $\mathfrak{B}_4(s,t)$, $d_2 = 0$ is selected in all iterations of all settings, and the most likely selected value of d_1 decreases from 1 (when $\sigma = 0.01$) to 0 (when $\sigma = 0.1$ or 1). For $\mathfrak{B}_5(s,t)$, the selected values for d_1 and d_2 are 1 when $\sigma = 0.01$ and decrease to 0 when $\sigma = 1$, with 100% frequency, indicating that large noise can mask complex local features.

Table 2: The average (and standard deviation) of MISEEs for Simulation 2.

T	σ	$\mathfrak{B}$	*fof.deriv*	*sSigComp*	*wSigComp*	*fdapace*
2^9	0.01	4	$4.27(1.15)\cdot10^{-3}$	$1.56(0.34)\cdot10^{-2}$	$9.43(2.04)\cdot10^{-2}$	$5.42(0.98)\cdot10^{-3}$
		5	$6.16(0.49)\cdot10^{-6}$	$4.62(2.41)\cdot10^{-3}$	$1.50(8.05)\cdot10^{-4}$	$5.59(2.07)\cdot10^{-3}$
	0.1	4	$5.20(0.99)\cdot10^{-3}$	$1.58(0.30)\cdot10^{-2}$	$9.80(1.66)\cdot10^{-3}$	$1.42(0.11)\cdot10^{-2}$
		5	$1.81(0.24)\cdot10^{-4}$	$4.34(2.12)\cdot10^{-3}$	$2.00(0.22)\cdot10^{-4}$	$1.43(0.20)\cdot10^{-2}$
	1	4	$7.15(0.30)\cdot10^{-2}$	$1.03(0.11)\cdot10^{-1}$	$1.02(0.07)\cdot10^{-1}$	$8.76(0.38)\cdot10^{-1}$
		5	$1.25(0.13)\cdot10^{-2}$	$2.09(0.30)\cdot10^{-2}$	$1.15(0.17)\cdot10^{-2}$	$8.76(0.35)\cdot10^{-1}$
2^{10}	0.01	4	$4.21(1.10)\cdot10^{-3}$	$1.60(0.34)\cdot10^{-2}$	$9.69(2.27)\cdot10^{-2}$	$5.55(1.36)\cdot10^{-3}$
		5	$4.96(0.43)\cdot10^{-6}$	$4.53(2.44)\cdot10^{-3}$	$4.88(21.2)\cdot10^{-5}$	$5.49(1.77)\cdot10^{-3}$
	0.1	4	$4.51(1.08)\cdot10^{-3}$	$1.60(0.38)\cdot10^{-2}$	$9.94(1.78)\cdot10^{-2}$	$1.40(0.12)\cdot10^{-2}$
		5	$1.20(0.18)\cdot10^{-4}$	$4.17(2.29)\cdot10^{-3}$	$1.33(0.22)\cdot10^{-4}$	$1.46(0.23)\cdot10^{-2}$
	1	4	$5.50(0.17)\cdot10^{-2}$	$5.41(0.17)\cdot10^{-2}$	$7.69(0.65)\cdot10^{-2}$	$8.81(0.41)\cdot10^{-1}$
		5	$6.16(0.53)\cdot10^{-3}$	$1.72(0.21)\cdot10^{-2}$	$7.00(0.91)\cdot10^{-3}$	$8.81(0.36)\cdot10^{-1}$

5. Application to the HPLC-PDA data

To illustrate the performance of our proposed method, we analyze the *HPLC-PDA data*. This is a metabolite profiling dataset (http://www.models.life.ku.dk/Bonnie) containing HPLC (high performance liquid chromatography) measurements of commercial extracts of St. John's wort, a plant that grows in the wild and is used for the treatment of mild to moderate depression. HPLC is a technique in analytical chemistry used to separate, identify, and quantify components in a mixture. It relies on pumps to pass a pressurized liquid and a sample mixture through a column filled with adsorbent, leading to the separation of the sample components. The time taken for a solute to pass through a chromatography column is called the retention time and is an identifying characteristic of a given analyte under particular conditions. It depends on the chemical nature of the component and its interaction with the column. The stronger the interaction, the more will

be the interaction time. The separated components are monitored and expressed electronically via detectors, such as PDA detector (photodiode array) that measures the amount of light of variable wavelengths absorbed by components of the mixture. PDA detects an entire spectrum simultaneously and the recorder (computer based data processor) generates a chromatogram at each wavelength. A chromatogram curve is a function of the retention time and its value gives concentration. As compounds have different absorbance sensitivity at different wavelength, it is helpful to study chromatograms across wavelengths in discerning between analytes with dissimilar absorbance spectra and determining an unknown peak in the chromatograms. However, due to economic reasons, not all wavelengths are used in practice. It is beneficial if we can accurately estimate the chromatograms at unused wavelength based on those generated.

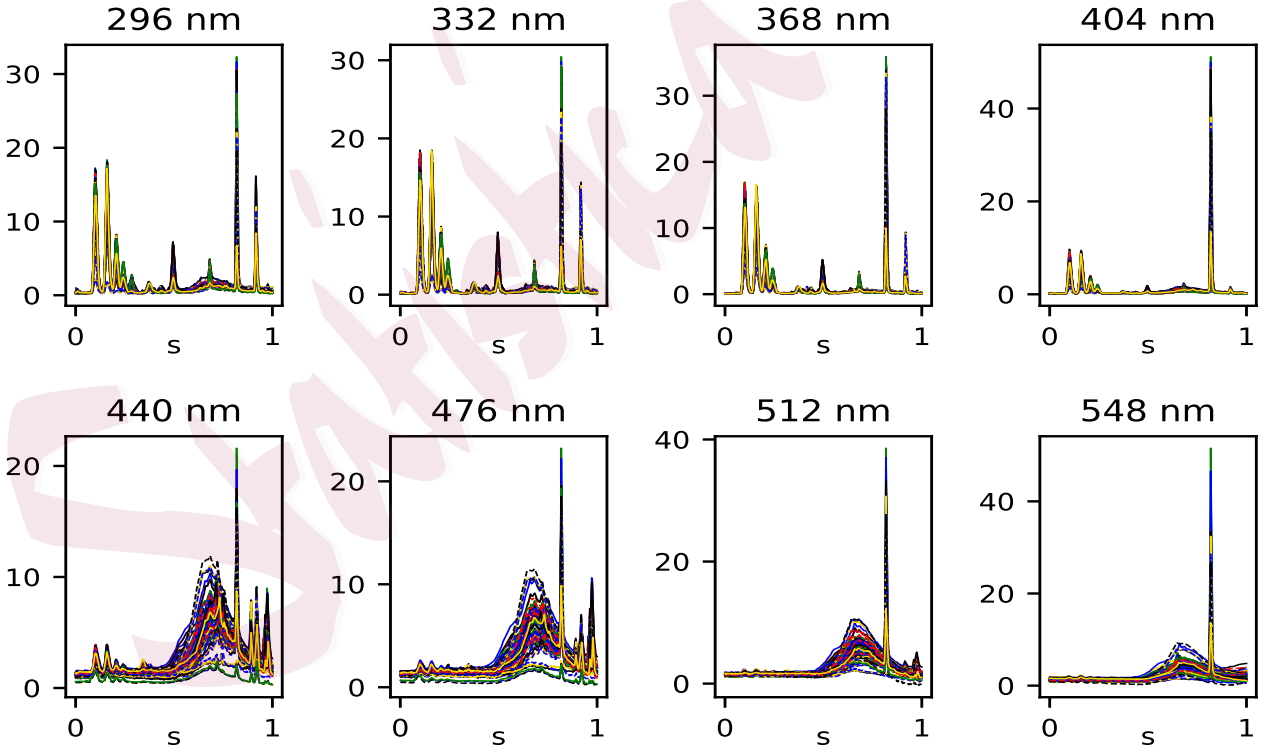

Figure 1: Chromatogram curves for all samples at eight equally spaced wavelengths (296, 332, 368, 404, 440, 476, 512, 548 nm) in the HPLC-PDA data. The x-axis is the retention time s which is scaled to $[0, 1]$, and the y-axis is the signal intensity.

The dataset in this study was obtained at every 3 nm wavelength from 260 nm to 550 nm. In Figure 1, we show the chromatogram curves for all samples at eight equally spaced wavelengths, 296, 332, 368, 404, 440, 476, 512, 548 nm, which almost spread the whole range of wavelengths in this data set. The retention time s has been scaled to $[0, 1]$ from the original range [12, 22.9] minutes. They show various smoothness/spikiness patterns. The chromatogram curves at lower wavelengths (296, 332, 368 nm) have more spiky peaks, while the curves at higher wavelengths gradually include smoother components and have less peaks. Fitting FOF models using these curves, we can evaluate the performance of our proposed method for functional data of various smoothness or spikiness patterns. To show by example, we fit seven FOF models using seven datasets formed by the chromatogram curves at the neighboring aforementioned wavelengths, each of which has the curves at the lower wavelength as the predictor and the curves at the higher wavelength as response. For example, the first model takes the chromatogram curves at wavelengths 296 and 332 nm as $X(s)$ and $Y(t)$, respectively. Table 3 lists the wavelengths of curves used as functional predictor and response, respectively, in each of the seven models. The first three models have spiky predictive and response curves (296$\sim$404 nm), the last two models have smooth response curves (512 nm and 548 nm) except a spike, and the fourth model has the greatest difference in the smoothness/spikiness patterns of the predictor (404 nm) and response curves (440 nm).

To compare all methods, we repeat the following procedure 100 times for each model. In each repeat, we randomly split the total 89 observations into a training data with $N_{\text{train}} = 60$ observations, and a test data with $N_{\text{test}} = 29$ observations. For each method, we choose tuning parameters and estimate the final model using the training data, and apply the final model to the test data. For each method, we calculate the mean integrated squared prediction error

Table 3: Average (and standard deviation) of MISPEs from 100 replicates for the HPLC-PDA data in seven models. The Y and X columns specify the wavelengths (nm) at which the chromatogram curves are used as the functional response and predictor, respectively, in each model.

Model	Y	X	*fof.deriv*	*sSigComp*	*wSigComp*	*fdapace*	*pffr*	*FDboost*
1	332	296	0.012(0.006)	0.049(0.011)	0.035(0.024)	1.228(0.056)	3.945(0.148)	2.551(0.114)
2	368	332	0.017(0.020)	0.039(0.019)	0.054(0.082)	1.267(0.140)	3.774(0.191)	2.580(0.199)
3	404	368	0.021(0.020)	0.106(0.033)	0.055(0.039)	1.388(0.456)	2.853(0.407)	2.242(0.364)
4	440	404	0.077(0.073)	0.166(0.032)	0.109(0.075)	0.594(0.097)	0.581(0.105)	0.524(0.055)
5	476	440	0.047(0.009)	0.058(0.009)	0.061(0.012)	0.099(0.020)	0.467(0.069)	0.405(0.063)
6	512	476	0.084(0.029)	0.083(0.021)	0.082(0.023)	0.199(0.035)	0.972(0.169)	0.878(0.154)
7	548	512	0.087(0.023)	0.110(0.023)	0.081(0.022)	0.206(0.054)	1.060(0.231)	0.959(0.220)

MISPE $= \frac{1}{TN_{\text{test}}} \sum_{l=1}^{N_{\text{test}}} \sum_{m=1}^{T} \left(\widehat{Y}_l^{\text{pred}}(t_m) - Y_l^{\text{test}}(t_m) \right)^2$, where $0 = t_1 < t_2 < \cdots < t_T = 1$ denote the $T = 2^9$ equally spaced observation points, $\{Y_l^{\text{test}}(t) : 1 \le l \le N_{\text{test}}\}$ denote the response curves in the test set and $\{\widehat{Y}_l^{\text{pred}}(t) : 1 \le l \le N_{\text{test}}\}$ are the corresponding predicted curves.

The MISPEs for the seven models are summarized in Table 3. The new method *fof.deriv* has significantly lower averaged MISPEs than all other methods in the first five models, where there are at least half a dozen peaks in both response and predictive curves. In Models $1 \sim 3$, where both the response and the predictor are spiky, the averaged MISPEs of all other methods are $2.3 \sim 328$ times as high as those of *fof.deriv*. In Models 4 and 5 where the response curve has both smooth and spiky parts, the average MISPEs of other methods are $1.3 \sim 10$ times as high as that of *fof.deriv*. In Models 6 and 7, the new method has slightly higher MISPE than *wSigComp* which has the smallest averaged MISPEs. In these two models, both the predictive and response curves are smooth except a few peaks. This implies that the wavelet coefficient vectors of these curves are sparse, hence the sparsity assumption for the wavelet-based method *wSigComp* is well satisfied. In Models $4 \sim 7$, with smooth components appeared in response or

in both the response and predictive curves, the smooth methods *fdapace*, *pffr* and *FDboost* have obvious improvement compared to their performance in the first three models. But they still have much higher error than the *fof.deriv*, *sSigComp* and *wSigComp*.

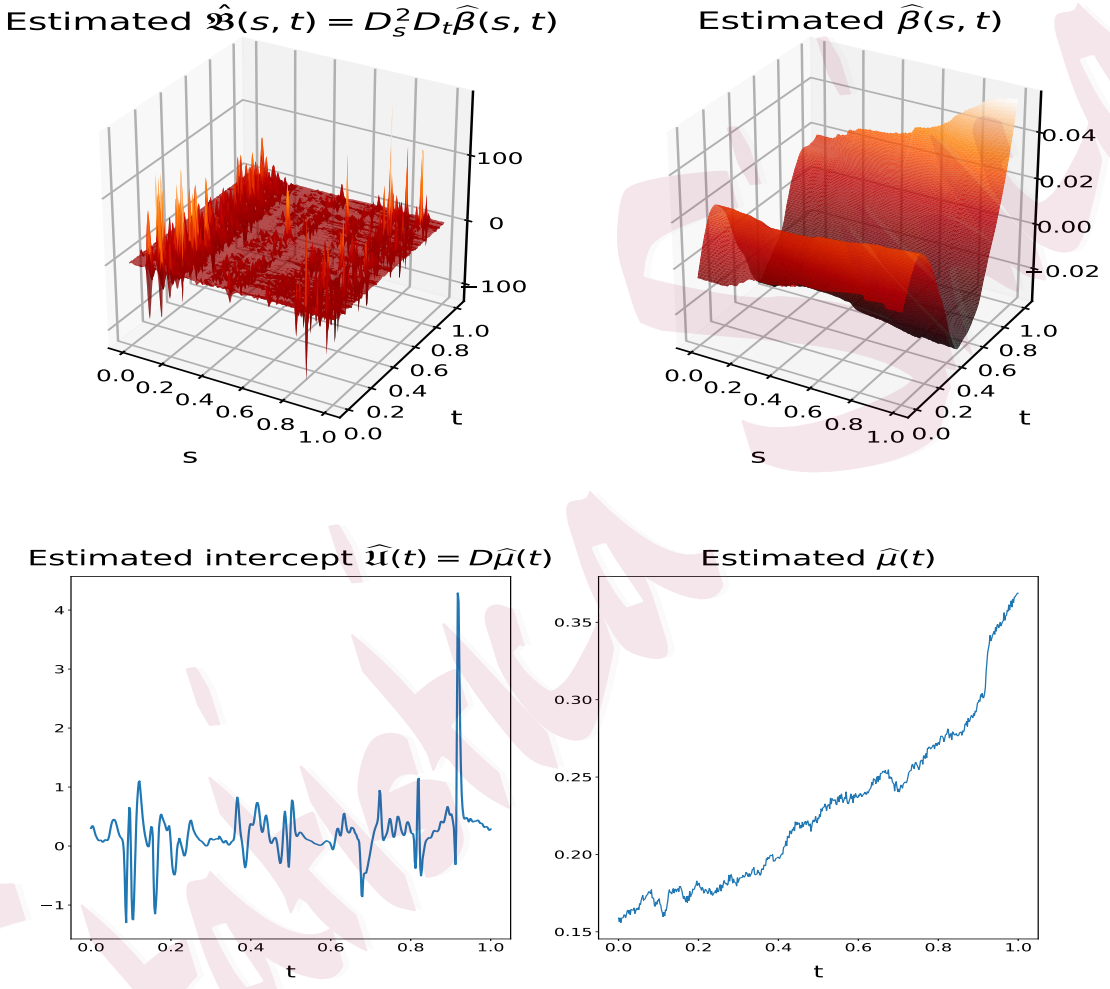

Figure 2: Estimated functions from the **first** model of the HPLC-PDA data, where $X(s)$ and $Y(t)$ are the chromatogram curves at wavelengths 296 and 332 nm, respectively. Top: the estimated coefficient surface $\widehat{\mathfrak{B}}(s,t)$ (left) and its corresponding auxiliary smooth function $\widehat{\beta}(s,t)$ (right) where $\widehat{\mathfrak{B}}(s,t) = D_s^2 D_t \widehat{\beta}(s,t)$; *Bottom:* the estimated intercept function $\widehat{\mathfrak{U}}(t)$ (left) and the corresponding auxiliary function $\widehat{\mu}(t)$ (right) with $\widehat{\mathfrak{U}}(t) = D\widehat{\mu}(t)$.

We next apply the new method *fof.deriv* to all 89 observations and fit the seven models, separately. In Figure 2, we provide the estimated functions in the first model with spiky predictor (296nm) and response (332nm) curves. The selected orders of partial derivatives are $d_1 = 2$ and $d_2 = 1$ in this model. The top panel of Figure 2 shows the estimated coefficient surface $\widehat{\mathfrak{B}}(s,t)$ (left) and the corresponding auxiliary function $\widehat{\beta}(s,t)$ (right) with $\widehat{\mathfrak{B}}(s,t) = D_s^2 D_t \widehat{\beta}(s,t)$. The $\widehat{\beta}(s,t)$ is smooth. By taking partial derivatives, we obtain the estimate $\widehat{\mathfrak{B}}(s,t)$ of the coefficient surface which is spiky, especially when $s \leq 0.2$ or $s \geq 0.8$, together with an isolated peak around $(0.5, 0.5)$. This corresponds to the large spikes in $Y(t)$ and $X(s)$ and indicates their associations. Similarly, the estimated intercept function $\widehat{\mathfrak{U}}(t)$ and its corresponding auxiliary function $\widehat{\mu}(t)$ with $\widehat{\mathfrak{U}}(t) = D\widehat{\mu}(t)$ are shown in the bottom panel of Figure 2. The $\widehat{\mathfrak{U}}(t)$ is wiggle in the whole range of t, with deep valleys and large peaks corresponding to the main spikes in the samples curves at wavelengths 296nm and 332nm of Figure 1. We show the estimated functions for the other six models in Figures S.13∼S.18 in Section S.3.4 of supplementary material. All these figures show that we can efficiently get spiky coefficient estimates via smooth auxiliary functions. Compared to the estimate $\widehat{\mathfrak{B}}(s,t)$ in Figure 2 for Model 1, the figures for Models $2 \sim 7$ do not have the isolated peak around $(0.5, 0.5)$, the peaks at $s \leq 0.2$ in $\widehat{\mathfrak{B}}(s,t)$ gradually weaken (Models $2 \sim 4$) and then completely disappear (Models $5 \sim 7$), and bulks show up around $s = 0.6$ in Model 4, get smoother in Models 5 and 6, and finally dampen in Model 7. All figures of $\widehat{\mathfrak{B}}(s,t)$ have spikes for $s \geq 0.8$, but the spikes get much fewer and weaker in Models 6 and 7. All these observations match the gradually changed patterns shown in the sample curves in Figure 1.

6. Discussion

By introducing a novel perspective for spiky estimates, we propose a new method to fit the FOF regression model for spiky functional data observed on a dense grid. We view the coefficient functions as the derivatives of smooth auxiliary functions. By imposing smoothing penalties on such auxiliary functions, we do not need the smoothness assumption on the coefficient functions which is common in FDA, and can produce unsmooth estimates by taking derivatives of the smooth auxiliary functions. Compared to the existing methods which directly estimate the coefficient function, our new approach is more efficient in dimension reduction and overcomes over-smoothing. Simulation and real data analysis show that the new method has better performance than existing methods for spiky coefficient functions, and has comparable prediction accuracy with the competing smooth method when both the intercept and slope coefficient functions are smooth. The asymptotic theory is applicable to models with coefficient functions in a larger space than the usual Sobolev space, and can provide smaller upper bounds for spiky functional data than the method designed for smooth functional data.

We used the CV to selected tuning parameters. Other methods can be explored, such as the GCV and the population based training (PBT) (Jaderberg *et al.*, 2017). The PBT is an adaptive method for hyperparameter search used in neural network. It starts with an initial set of parameter combinations and repeatedly updates the set by exploitation and exploration. There are various exploration and exploitation strategies, whose performance in our model deserves further investigation.

We empirically explored the performance of the proposed method when the predictor curves contain observation errors in one simulation, and the results show that the proposed method still has good performance. However, it is not trivial to extend our theoretical results to the

general cases of predictor curves with noises. The proof of our theoretical results relies on the fact that $\widehat{\mathbf{B}}(s,s')$ and $\widehat{\mathbf{\Sigma}}(s,s')$ respectively are consistent estimates of $\mathbf{B}(s,s')$ and $\mathbf{\Sigma}(s,s')$ in the generalized eigenvalue problem in Theorem 1. However, when the observed predictor curves have noises, $\widehat{\mathbf{\Sigma}}(s,s')$ defined in (2.6) cannot be calculated directly, and the sample covariance function of the noisy observations may not be good estimate of $\mathbf{\Sigma}(s,s')$. Hence, the current proof cannot be directly extended to the situation of predictor curves with observation errors. Further investigation is needed.

We use the relationship between integration and differentiation and consider smooth auxiliary functions whose derivative gives the original coefficient functions. It is possible to consider other operators to get smooth auxiliary functions for coefficient functions. The idea of regularizing smooth auxiliary functions can also be applied to other analysis involving spiky functional data.

References

Adams, R. A. and Fournier, J. J. (2003) *Sobolev spaces*, vol. 140. Academic press.

Besse, P. C. and Cardot, H. (1996) Approximation spline de la prévision d'un processus fonctionnel autorégressif d'ordre 1. *Canadian Journal of Statistics*, **24**, 467–487.

Brockhaus, S., Rügamer, D. and Greven, S. (2017) Boosting functional regression models with fdboost. *arXiv preprint arXiv:1705.10662*.

Chiou, J.-M., Yang, Y.-F. and Chen, Y.-T. (2016) Multivariate functional linear regression and prediction. *Journal of Multivariate Analysis*, **146**, 301–312.

Delaigle, A. and Hall, P. (2012) Methodology and theory for partial least squares applied to functional data. *The Annals of Statistics*, **40**, 322–352.

Ivanescu, A. E., Staicu, A.-M., Scheipl, F. and Greven, S. (2014) Penalized function-on-function regression. *Computational Statistics*, 1–30.

Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K. *et al.* (2017) Population based training of neural networks. *arXiv preprint arXiv:1711.09846*.

Luo, R. and Qi, X. (2017) Function-on-function linear regression by signal compression. *Journal of the American Statistical Association*, **112**, 690–705.

Luo, R., Qi, X. and Wang, Y. (2016) Functional wavelet regression for linear function-on-function models. *Electronic Journal of Statistics*, **10**, 3179–3216.

Nason, G. (2010) *Wavelet methods in statistics with R*. Springer.

Ramsay, J. O. and Dalzell, C. (1991) Some tools for functional data analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, 539–572.

Ramsay, J. O. and Silverman, B. W. (2005) *Functional Data Analysis. 2nd Edition.* New York: Springer.

Reiss, P. T., Huo, L., Zhao, Y., Kelly, C. and Ogden, R. T. (2015) Wavelet-domain regression and predictive inference in psychiatric neuroimaging. *The annals of applied statistics*, **9**, 1076.

Scheipl, F., Staicu, A.-M. and Greven, S. (2015) Functional additive mixed models. *Journal of Computational and Graphical Statistics*, **24**, 477–501.

Yao, F., Müller, H.-G. and Wang, J.-L. (2005) Functional linear regression analysis for longitudinal data. *The Annals of Statistics*, **33**, 2873–2903.

Zhao, Y., Ogden, R. T. and Reiss, P. T. (2012) Wavelet-based lasso in functional linear regression. *Journal of Computational and Graphical Statistics*, **21**, 600–617.