Index Models for Sparsely Sampled Functional Data

Peter Radchenko, Xinghao Qiao, and Gareth M. James [*]

May 21, 2014

Abstract

The regression problem involving functional predictors has many important applications and a number of functional regression methods have been developed. However, a common complication in functional data analysis is one of sparsely observed curves, that is predictors that are observed, with error, on a small subset of the possible time points. Such sparsely observed data induces an *errors-in-variables* model, where one must account for measurement error in the functional predictors. Faced with sparsely observed data, most current functional regression methods simply estimate the unobserved predictors and treat them as fully observed; thus failing to account for the extra uncertainty from the measurement error. We propose a new functional errors-in-variables approach, *Sparse Index Model Functional Estimation* (SIMFE), which uses a functional index model formulation to deal with sparsely observed predictors. SIMFE has several advantages over more traditional methods. First, the index model implements a non-linear regression and uses an accurate supervised method to estimate the lower dimensional space into which the predictors should be projected. Second, SIMFE can be applied to both scalar and functional responses and multiple predictors. Finally, SIMFE uses a mixed effects model to effectively deal with very sparsely observed functional predictors and to correctly model the measurement error.

Some key words: Index Model; Functional Regression; Non-linear Regression; Sparsely Sampled Functional Data; Error-In-Variables

[*] Marshall School of Business, University of Southern California. This work was partially supported by NSF Grant DMS-1209057.

1 Introduction

In a *Functional Data Analysis* (FDA) setting the regression problem involving one or more functional predictors, $X_1(t), \ldots, X_p(t)$, and either a functional or scalar response, has recently received a great deal of attention. A few examples include Hastie and Mallows (1993); Hall *et al.* (2000); Alter *et al.* (2000); Hall *et al.* (2001); James (2002); Cardot *et al.* (2003); Ferraty and Vieu (2003); James and Silverman (2005); Muller and Stadtmuller (2005); Chen *et al.* (2011), and Jiang and Wang (2011). See Chapter 15 of Ramsay and Silverman (2005) for a thorough discussion of the issues involved with fitting such data. For examples of recent research on multivariate functional data see Hall *et al.* (2006), Li and Hsing (2010), Li and Chiou (2011), Chiou and Muller (2014) and the references therein.

Given a scalar response Y_i and a functional predictor $X_i(t)$, the standard classical functional regression model is of the form

$$Y_i = \beta_0 + \int \beta(t) X_i(t) dt + \epsilon_i, \tag{1}$$

which implies a scalar response, a single densely observed functional predictor and a linear relationship. Since functional predictors are infinite dimensional, fitting (1) also requires some form of dimension reduction. Most approaches use an unsupervised method, such as functional principal components analysis, to represent the predictors and then regress Y against the lower dimensional representation of $X(t)$.

More recently there has been some work on extending (1) using supervised dimension reduction methods to represent the predictors and non-linear models for the response surface. One of the most natural ways to approach this problem is to use an index model:

$$Y_i = m\left(\int \beta(t) X_i(t) dt\right) + \epsilon_i, \tag{2}$$

where the index function, $\beta(t)$, projects the predictor into a lower dimensional space and $m(\cdot)$ is a low dimensional non-linear function. James and Silverman (2005) proposed a functional index model similar to (2), and Chen *et al.* (2011) extended this work to a fully non-parametric setting and provided further theoretical motivation.

In this article we consider the common situation where one only observes a noisy version of $X_i(t)$ over a handful of time points. In this setting, computing the integral in (2), and hence fitting the index model, becomes considerably more complicated. We propose a new errors-in-variables approach (Carroll *et al.*, 2006), named *Sparse Index Model Functional*

Estimation (SIMFE), which also implements an index model, but offers advantages over the previously discussed approaches. In particular SIMFE uses a mixed effects model to utilize information from all the predictors, and hence provide an accurate reconstruction of $X_i(t)$. Further, we prove that the SIMFE estimate, $\hat{\beta}(t)$, still has good convergence properties, even for sparsely observed predictors. Finally, SIMFE can be applied to data with multiple functional predictors and either scalar or functional responses.

The remainder of this article is structured as follows. In Section 2 we present the SIMFE model and develop fitting procedures for both scalar and functional responses. We also present an extension of SIMFE which adjusts for possible bias in the estimate for the non-linear function $m(\cdot)$ in situations where the predictors are observed at different time points for each individual. Theoretical results are presented in Section 3, which demonstrate that even for sparsely observed predictors SIMFE will be consistent in estimating the space spanned by the set of index functions and has a faster rate of convergence than other potential approaches. Section 4 illustrates the performance of SIMFE on an extensive set of simulations covering both scalar and functional responses. Section 5 applies SIMFE to an online auction data set containing sparsely observed predictors.

2 SIMFE

In Section 2.1 we present the SIMFE model for scalar responses and develop a fitting procedure in Section 2.2. SIMFE is extended to functional responses in Section 2.3. Section 2.4 presents a bias corrected version of SIMFE for situations where predictors are observed at differing sets of time points. Finally, we discuss selection of tuning parameters in Section 2.5.

2.1 Scalar Response

In the scalar response setting we observe p functional predictors, $X_{i1}(t), \ldots, X_{ip}(t)$, and a scalar response, Y_i, where $i = 1, \ldots, n$. For concreteness, we will assume that the domain for each predictor is $[0, 1]$, however, all of the presented methods and conclusions are still valid in the situation where the predictors have different domains. Without loss of generality we can model

$$Y_i = m_0(\mathbf{X}_i) + \varepsilon_i \tag{3}$$

where $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})$ and $E(\varepsilon_i|\mathbf{X}_i) = 0$. However, (3) is too general to fit in practice because m_0 is a function of p infinite dimensional objects. A particularly natural approach to deal with the infinite dimension of the predictors involves restricting m_0 to be a function of linear projections of $\mathbf{X}_i$:

$$m_0(\mathbf{X}_i) = m(\mathbf{P}_i), \tag{4}$$

where $m(\mathbf{P}_i) \equiv m(\mathbf{P}_{i1}, \ldots, \mathbf{P}_{ip})$,

$$\mathbf{P}_{ij} = \left(\int \beta_{j1}(t) X_{ij}(t) dt, \ldots, \int \beta_{jd_j}(t) X_{ij}(t) dt \right) \tag{5}$$

and d_j represents the dimension of the space into which $X_{ij}(t)$ is projected. Hence, $\mathbf{P}_{ij}$ is the d_j-dimensional linear projection of $X_{ij}(t)$ formed from the index functions, $\beta_{j1}(t), \ldots, \beta_{jd_j}(t)$. We can now rewrite equation (3) as follows:

$$Y_i = m(\mathbf{P}_i) + \varepsilon_i. \tag{6}$$

Equation (6) is a functional version of a multi-index model, which has been considered previously in James and Silverman (2005) and Chen *et al.* (2011). However, these previous methods mainly dealt with the situation involving a single densely sampled predictor. In this paper we consider the common situation involving multiple very sparsely observed predictors.

To deal with the sparsely observed predictors we build strength across all observations by modeling $\mathbf{X}_i(t)$ as coming from a multivariate Gaussian process with mean $\boldsymbol{\mu}(t)$ and covariance $\Gamma(t, u)$:

$$\mathbf{X}_i \sim G(\boldsymbol{\mu}, \Gamma). \tag{7}$$

Further, we assume that $X_{ij}(t)$ is observed, with measurement error, at T_{ij} time points, $\mathbf{t}_{ij} = (t_{ij1}, \ldots, t_{ijT_{ij}})$. In particular, let W_{ijk} represent the observed value of $X_{ijk} = X_{ij}(t_{ijk})$. Then,

$$W_{ijk} = X_{ijk} + e_{ijk}, \tag{8}$$

where $i = 1, \ldots, n$, $j = 1, \ldots, p$, $k = 1, \ldots, T_{ij}$ and the e_{ijk}'s are modeled as iid $N(0, \sigma^2)$. We will see shortly that (7) and (8) allow us to fit a mixed effects model to infer the entire predictor trajectory based on only a small number of observations. The SIMFE model is specified by equations (6), (7) and (8).

2.2 The SIMFE Fitting Method

For scalar responses the SIMFE model states that $Y_i = m(\mathbf{P}_i) + \varepsilon_i$, where $E(\varepsilon_i|\mathbf{X}_i) = 0$. Hence, the fitting methods of James and Silverman (2005) or Chen *et al.* (2011) could be used to implement SIMFE, provided that the predictors were fully observed. Unfortunately, for sparse functional predictors, rather than observing $\mathbf{X}_i(t)$ we observe only $\mathbf{W}_i = (\mathbf{W}_{i1}, \dots, \mathbf{W}_{ip})$ where $\mathbf{W}_{ij} = (W_{ij1}, \dots, W_{ijT_{ij}})$.

However, Theorem 1 shows that by replacing $X_{ij}(t)$ with its conditional expectation, the response can still be represented using an index model with exactly the same index as if $X_{ij}(t)$ has been fully observed.

Theorem 1 *Let* $\tilde{X}_{ij}(t) = E\left\{X_{ij}(t)|\mathbf{W}_i\right\}$ *and*

$$\tilde{\mathbf{P}}_{ij} = \left(\int \beta_{j1}(t)\tilde{X}_{ij}(t)dt, \dots, \int \beta_{jd_j}(t)\tilde{X}_{ij}(t)dt\right). \tag{9}$$

Then, under the SIMFE model,

$$Y_i = \tilde{m}_{\mathbf{t}_i}(\tilde{\mathbf{P}}_i) + \varepsilon_i^*, \tag{10}$$

where $\mathbf{t}_i = (\mathbf{t}_{i1}, \dots, \mathbf{t}_{ip})$ *and* $E(\varepsilon_i^*|\mathbf{W}_i) = 0$.

The proof of Theorem 1 is provided in the appendix. Notice that the definitions of $\mathbf{P}_{ij}$ and $\tilde{\mathbf{P}}_{ij}$ are identical except that $X_{ij}(t)$ in (5) is replaced by $\tilde{X}_{ij}(t)$ in (9). In particular, the same index functions, $\beta_{jk}(t)$, are used in both definitions, thus fitting an index model using $\tilde{X}_{ij}(t)$ will produce estimates for the same index functions as one would obtain when using the original $X_{ij}(t)$ predictors. Theorem 1 is an important result because it suggests a two step approach for fitting SIMFE; first estimate $\tilde{X}_{ij}(t)$ and second fit the resulting multi-index model given by (10).

Theorem 1 does imply one significant difficulty in fitting (10); the function $\tilde{m}$ is related to both $\tilde{\mathbf{P}}_i$ and the time points $\mathbf{t}_i$. If the time points are common for all individuals, i.e. $\mathbf{t}_i = \mathbf{t}$ for all i, then (10) reduces to

$$Y_i = \tilde{m}(\tilde{\mathbf{P}}_i) + \varepsilon_i^*, \tag{11}$$

and fitting $\tilde{m}(\cdot)$ is feasible. However, if the time points differ among individuals, then we are potentially faced with the highly challenging problem of estimating n separate $\tilde{m}_{\mathbf{t}_i}(\cdot)$ functions from only n observations.

In the following two sections we first develop an approach for estimating $\tilde{X}_{ij}(t)$ and then propose a method for fitting (10) in the easier setting where (11) holds. Then in Section 2.4 we consider the more challenging setting where $\mathbf{t}_i$ differ across individuals.

2.2.1 Estimating $\tilde{X}_{ij}(t)$

In order to fit the SIMFE model some form of smoothness constraint is required for $X_{ij}(t)$ and $\beta_{jk}(t)$. We take the standard approach of modeling these functions using q_j-dimensional orthogonal basis functions, $\mathbf{s}_j(t)$, such that $\int \mathbf{s}_j(t)\mathbf{s}_j(t)^T dt = I$. Hence,

$$X_{ij}(t) = \mathbf{s}_j(t)^T \boldsymbol{\delta}_{ij} \quad \text{and} \quad \beta_{jk}(t) = \mathbf{s}_j(t)^T \boldsymbol{\eta}_{jk}, \tag{12}$$

where $\boldsymbol{\delta}_{ij}$ and $\boldsymbol{\eta}_{jk}$ are respectively the basis coefficients for the predictors and index functions.

Using (12) the projection, $\mathbf{P}_{ij}$, becomes

$$\mathbf{P}_{ij} = (\boldsymbol{\eta}_{j1}^T \boldsymbol{\delta}_{ij}, \ldots, \boldsymbol{\eta}_{jd_j}^T \boldsymbol{\delta}_{ij}),$$

and the Gaussian process model for $\mathbf{X}_i(t)$ implies that

$$\boldsymbol{\delta}_i = (\boldsymbol{\delta}_{i1}^T, \ldots, \boldsymbol{\delta}_{ip}^T)^T \sim N(\boldsymbol{\mu}_\delta, \Delta), \tag{13}$$

Let $\mathbf{s}(\cdot)$ be a matrix valued function, such that for each t matrix $\mathbf{s}(t)$ is block diagonal, with the j-th block given by the column vector $s_j(t)$. Then, $E(\mathbf{X}_i(t)) = \boldsymbol{\mu}(t) = \boldsymbol{\mu}_\delta^T \mathbf{s}(t)$ and $cov(\mathbf{X}_i(t), \mathbf{X}_i(u)) = \Gamma(t,u) = \mathbf{s}(t)^T \Delta \mathbf{s}(u)$. Let S_j be the T_j by q_j-dimensional basis matrix with kth row $\mathbf{s}_j(t_{jk})$, and S be the block diagonal matrix with jth block S_j. Standard calculations show that (8), (12) and (13) imply

$$\mathbf{X}_i(t)|\mathbf{W}_i \sim N\left(\tilde{\boldsymbol{\mu}}_i^T \mathbf{s}(t), \mathbf{s}(t)^T \tilde{\Delta} \mathbf{s}(t)\right),$$

where

$$\tilde{\boldsymbol{\mu}}_i = (\tilde{\boldsymbol{\mu}}_{i1}^T, ..., \tilde{\boldsymbol{\mu}}_{ip}^T)^T = \tilde{\Delta}\left(\Delta^{-1}\boldsymbol{\mu}_\delta + \frac{1}{\sigma^2} S^T \mathbf{W}_i^T\right) \tag{14}$$

and

$$\tilde{\Delta} = \left(\Delta^{-1} + \frac{1}{\sigma^2} S^T S\right)^{-1}. \tag{15}$$

Hence,

$$\tilde{X}_{ij}(t) = E(X_{ij}(t)|\mathbf{W}_i) = \mathbf{s}_j(t)^T \tilde{\boldsymbol{\mu}}_{ij}. \tag{16}$$

In practice $\Delta, \boldsymbol{\mu}_\delta$ and σ^2 are unknown, so we need to estimate $\tilde{X}_{ij}(t)$ using

$$\hat{X}_{ij}(t) = \mathbf{s}_j(t)^T \hat{\boldsymbol{\mu}}_{ij},$$

where $\hat{\boldsymbol{\mu}}_i = (\hat{\boldsymbol{\mu}}_{i1}^T, ..., \hat{\boldsymbol{\mu}}_{ip}^T)^T$ is computed by inserting appropriate parameter estimates into (14). We can estimate Δ, $\boldsymbol{\mu}_\delta$ and σ^2 using an EM algorithm, the details of which are provided in Appendix A.

Note that formulas (14) and (15) can be significantly simplified under the assumption that the predictors are independent. In our experience this simplified version of SIMFE often performs competitively relative to the more general version, even in the settings with correlated predictors.

2.2.2 Fitting the Multi-Index Model

To fit the multi-index model a natural population criterion to minimize is

$$E_{Y,\mathbf{X}} \left(Y - \tilde{m}(\tilde{\mathbf{P}}) \right)^2 = E_{\mathbf{X}} \left[E_Y \left([Y - \tilde{m}(\tilde{\mathbf{P}})]^2 | \mathbf{X} \right) \right],$$

which can be approximated for a finite sample by

$$E_{\mathbf{X}} \left[E_Y \left([Y - \tilde{m}(\tilde{\mathbf{P}})]^2 | \mathbf{X} \right) \right] \approx \frac{1}{n} \sum_{i=1}^n E_Y \left[\left(Y - \tilde{m}(\tilde{\mathbf{P}}) \right)^2 | \mathbf{X} = \mathbf{X}_i \right]. \tag{17}$$

We approximate $E_Y \left[\left(Y - \tilde{m}(\tilde{\mathbf{P}}) \right)^2 | \mathbf{X} = \mathbf{X}_i \right]$ using local linear regression,

$$\sum_{l=1}^n \left(Y_l - a_i - \sum_{j=1}^p \hat{\mathbf{P}}_{lj} \mathbf{c}_{ij} \right)^2 K_{il}, \tag{18}$$

where

$$K_{il} = K_h(\hat{\mathbf{P}}_l - \hat{\mathbf{P}}_i) \tag{19}$$

is an appropriate kernel function with bandwidth h and $\hat{\mathbf{P}}_k = (\hat{\mathbf{P}}_{k1}, \ldots, \hat{\mathbf{P}}_{kp})$. Note that $\hat{\mathbf{P}}_{kj}$ is identical to $\tilde{\mathbf{P}}_{kj}$, except that $\tilde{X}_{ij}(t)$ is replaced by $\hat{X}_{ij}(t)$. We use a Gaussian kernel with the optimal bandwidth

$$h_{opt} = (4/(d+2))^{1/(d+4)} n^{-1/(d+4)}, \tag{20}$$

where $d = \sum_{j=1}^p d_j$ is the dimension of the kernel (Silverman, 1999).

Combining (17) and (18) gives the finite sample criterion that SIMFE minimizes:

$$\frac{1}{n} \sum_{i=1}^n \sum_{l=1}^n \left(Y_l - a_i - \sum_{j=1}^p \hat{\mathbf{P}}_{lj} \mathbf{c}_{ij} \right)^2 K_{il}. \tag{21}$$

Note that the resulting estimator does not change if we replace $\hat{\mathbf{P}}_{lj}$ with $\hat{\mathbf{P}}_{lj} - \hat{\mathbf{P}}_{ij}$ in the above display, as is often done in the literature. To minimize (21) we use a two step iteration, where we first estimate $\tilde{m}(\cdot)$, and second compute the $\beta_{jk}(t)$ functions given $\tilde{m}(\cdot)$. We describe these two steps below.

Step One

Let $\boldsymbol{\gamma}_i = \left(a_i, \mathbf{c}_{i1}^T, \ldots, \mathbf{c}_{ip}^T\right)^T$ and $\mathbf{R}_l = (1, \hat{\mathbf{P}}_{l1}, \ldots, \hat{\mathbf{P}}_{lp})^T$ then (21) can be written as

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{l=1}^{n} K_{il}\left(Y_l - \mathbf{R}_l^T\boldsymbol{\gamma}_i\right)^2,$$

whose minimum is obtained by setting

$$\hat{\boldsymbol{\gamma}}_i = \left(\sum_{l=1}^{n} K_{il}\mathbf{R}_l\mathbf{R}_l^T\right)^{-1}\sum_{l=1}^{n} K_{il}Y_l\mathbf{R}_l, \quad i = 1, \ldots n. \tag{22}$$

Step Two

Next, we estimate $\beta_{jk}(t)$ for $j = 1, \ldots, p$ and $k = 1, \ldots, d_j$, given the current estimate for $\tilde{m}(\cdot)$. Let

$$Q_{il} = \begin{pmatrix} \mathbf{c}_{i1} \otimes \int \hat{X}_{l1}(t)\mathbf{s}_1(t)dt \\ \vdots \\ \mathbf{c}_{ip} \otimes \int \hat{X}_{lp}(t)\mathbf{s}_p(t)dt \end{pmatrix} = \begin{pmatrix} \mathbf{c}_{i1} \otimes \hat{\boldsymbol{\mu}}_{l1} \\ \vdots \\ \mathbf{c}_{ip} \otimes \hat{\boldsymbol{\mu}}_{lp} \end{pmatrix}$$

and $\boldsymbol{\eta} = (\boldsymbol{\eta}_{11}^T, \ldots, \boldsymbol{\eta}_{1d_1}^T, \ldots, \boldsymbol{\eta}_{p1}^T, \ldots, \boldsymbol{\eta}_{pd_p}^T)^T$, where $\otimes$ is the Kronecker product. It is not hard to show that the estimate for $\boldsymbol{\eta}$ that minimizes (21) is given by

$$\hat{\boldsymbol{\eta}} = \left(\sum_{i=1}^{n}\sum_{l=1}^{n} K_{il}Q_{il}Q_{il}^T\right)^{-1}\sum_{i=1}^{n}\sum_{l=1}^{n} K_{il}Q_{il}(Y_l - a_i). \tag{23}$$

Hence,

$$\hat{\beta}_{jk}(t) = \mathbf{s}_j(t)^T\hat{\boldsymbol{\eta}}_{jk}, \quad j = 1, \ldots, p, \quad k = 1, \ldots, d_j, \tag{24}$$

where $\hat{\boldsymbol{\eta}}_{jk}$ represent the corresponding elements of $\hat{\boldsymbol{\eta}}$.

2.2.3 Scalar Response SIMFE Algorithm

Our theoretical results in Section 3 show that initially fitting SIMFE using a bandwith larger than h_{opt} (20) and then iteratively reducing h provides a good rate of convergence. Hence, the scalar response version of the SIMFE algorithm is summarized in Algorithm 1 (next page).

Algorithm 1 Scalar Response SIMFE Algorithm

```
1. Compute the $\hat{X}_{ij}(t)$'s by plugging the parameter estimates from the EM algorithm into (16).
2. Initialize $\hat{\beta}_{jk}(t)$ and $h \propto n^{-1/(\tilde{p}+4)}$ where $\tilde{p} = \sum_{j=1}^{p} q_j$.
3. (a) Estimate $\tilde{m}(\hat{\mathbf{P}})$ using the linear approximation given by (22).
   (b) Compute the $\hat{\beta}_{jk}(t)$'s via (23) and (24).
   (c) Repeat Steps (a) and (b) until convergence.
4. Set $h \leftarrow ch$ for some $c < 1$.
5. Iterate Steps 3. and 4. until $h \leq h_{opt}$.
```

We initialize $\hat{\beta}_{jk}(t)$ in Step 2 by fitting the groupwise Outer Product of Gradients (gOPG) estimator (Li *et al.*, 2010) to the $\hat{\boldsymbol{\mu}}_{ij}$'s from Step 1, using a bandwidth of $h \propto n^{-1/(\tilde{p}+4)}$. Empirically, we have found that the SIMFE algorithm is stable and typically converges fast.

2.2.4 Predicting the Response

Given a new observation $\mathbf{W}^*$ we will often wish to form a prediction for the corresponding response Y^*. Once SIMFE has been fitted, we can compute $\hat{\mathbf{X}}^*(t)$, and hence the $\hat{\mathbf{P}}^*$ corresponding to the new observation. Because $\tilde{m}(\cdot)$ is computed using a linear approximation, we have n different predictions for $\hat{Y}^*$, i.e. $\tilde{m}_i(\hat{\mathbf{P}}^*) = a_i + \sum_j \hat{\mathbf{P}}_j^* \mathbf{c}_{ij}$. An obvious approach to produce a single $\hat{Y}$ is to weight the individual predictions according to the difference between $\mathbf{P}_i$ and $\mathbf{P}^*$. Hence, we use the following weighted average:

$$\hat{Y}^* = \tilde{m}(\hat{\mathbf{P}}^*) = \sum_{i=1}^{n} \tilde{m}_i(\hat{\mathbf{P}}^*) w_{*i} = \sum_{i=1}^{n} (a_i + \sum_j \hat{\mathbf{P}}_j^* \mathbf{c}_{ij}) w_{*i}, \tag{25}$$

where $w_{*i} = K_{*i} / \sum_l K_{*l}$ and K_{*i} is computed using (19).

2.3 Functional Response

In the functional response setting we observe p functional predictors, $X_{i1}(t), \ldots, X_{ip}(t)$, and a functional response, $Y_i(s)$, where $i = 1, \ldots, n$ and Y_i is observed at points $s_{i1}, \ldots, s_{in_i}$. Let $Y_{ik} = Y_i(s_{ik})$. A natural approach to extend (6) to functional responses is to model

$E(Y(s)|\mathbf{X}) = \mu_Y(s)$ as a function of both $\mathbf{P}_1, \ldots, \mathbf{P}_p$ and s:

$$\mu_Y(s) = m(s, \mathbf{P}).$$

To fit the multi-index model we aim to minimize the population criterion

$$\int E_{Y,\mathbf{X}} \left\{ \left(Y(s) - m(s, \mathbf{P})\right)^2 \right\} ds = \int E_{\mathbf{X}} \left\{ E_Y \Big([Y(s) - m(s, \mathbf{P})]^2 \big| \mathbf{X} \Big) \right\} ds,$$

which can be approximated for a finite sample by

$$\frac{1}{\sum_{i=1}^n n_i} \sum_{i,k} E_Y \left[\left(Y(s) - m(s, \mathbf{P})\right)^2 | \mathbf{X} = \mathbf{X}_i, s = s_{ik} \right]. \tag{26}$$

Again we approximate $E_Y \left[\left(Y(s) - m(s, \mathbf{P})\right)^2 | \mathbf{X} = \mathbf{X}_i, s = s_{ik} \right]$ using a local linear regression:

$$\sum_{l,k'} \left(Y_{lk'} - a_{ik} - c_{ik}^s s_{lk'} - \sum_{j=1}^p \hat{\mathbf{P}}_{lj} \mathbf{c}_{ikj} \right)^2 K_{iklk'}, \tag{27}$$

where $K_{iklk'} = K_h(s_{lk'} - s_{ik}, \hat{\mathbf{P}}_l - \hat{\mathbf{P}}_i)$ is an appropriate kernel function with bandwidth h. Note that the d in (20) is replaced by $1 + \sum_{j=1}^p d_j$, because the kernel is a function of one additional parameter, s.

Combining (26) and (27) gives the functional response finite sample criterion that SIMFE minimizes:

$$\frac{1}{\sum_{i=1}^n n_i} \sum_{i,k} \sum_{l,k'} \left(Y_{lk'} - a_{ik} - c_{ik}^s s_{lk'} - \sum_{j=1}^p \hat{\mathbf{P}}_{lj} \mathbf{c}_{ikj} \right)^2 K_{iklk'}. \tag{28}$$

As in the scalar response setting, we can minimize (28) using a two step iteration; first estimate $m(s, \hat{\mathbf{P}})$, and second compute the $\beta_{jk}(t)$ functions given m. Each step, and the final algorithm, are similar to that for the scalar response SIMFE. The details are provided in Appendix B.

2.4 Bias Corrected SIMFE

We now return to the situation where $\mathbf{t}_i$ may differ among individuals. There is in principle nothing preventing us from still fitting the previously described approach. The only change that is required is to replace matrix S_j, introduced in Section 2.2.1 with a T_{ij} by q_j matrix, S_{ij}, whose kth row is given by $\mathbf{s}_j(t_{ijk})$. The corresponding block-diagonal matrix S is then replaced in formulas (14), (15) and (41) by S_i, which is constructed using matrices S_{ij}. Indeed, we successfully implement this version of SIMFE in Sections 4 and 5. We refer to this implementation of our method as Base SIMFE for the remainder of the paper. However, in the setting with differing time points this version of SIMFE suffers from a potential biased estimation problem.

To better understand the difficulty with differing time points, consider the expected value of the response, Y_i, conditional on our noisy observation of the predictors, $\mathbf{W}_i$. If the predictors are observed at the set of time points $\mathbf{t}_i$, then we have:

$$\begin{aligned} E(Y_i|\mathbf{W}_i) &= E_{\mathbf{X}}\left(E_Y(Y_i|\mathbf{X}_i, \mathbf{W}_i)|\mathbf{W}_i\right) \\ &= E_{\mathbf{X}}\left(m\left(\mathbf{P}_i\right)|\mathbf{W}_i\right) \\ &= E_U\left(m\left(\tilde{\mathbf{P}}_i + \mathbf{U}_i\right)|\mathbf{W}_i\right) \\ &= E_U\left(m\left(\tilde{\mathbf{P}}_i + \mathbf{U}_i\right)\right) = \tilde{m}_{\mathbf{t}_i}\left(\tilde{\mathbf{P}}_i\right), \end{aligned} \tag{29}$$

where $\mathbf{U}_i = \mathbf{P}_i - \tilde{\mathbf{P}}_i$ and $\tilde{m}_{\mathbf{t}_i}(\cdot) = E_U\left(m\left(\cdot + \mathbf{U}_i\right)\right)$. Equation (29) follows because $\mathbf{U}_i$ is statistically independent of $\mathbf{W}_i$; a fact which we demonstrate in the proof of Theorem 1. However, while $\mathbf{U}_i$ is statistically independent of $\mathbf{W}_i$, the distribution of $\mathbf{U}_i$ is a function of the time points over which $\mathbf{W}_i$ is observed. If $\mathbf{W}_i$ is observed at the same set of points for all i then the distribution of $\mathbf{U}_i$ is the same for all i, hence $\tilde{m}_{\mathbf{t}_i}(\cdot) = \tilde{m}(\cdot)$. In other words, we only need estimate a single link function.

Unfortunately, when the time points differ among observations, function $\tilde{m}_{\mathbf{t}_i}(\cdot)$ depends on the locations of the points. Hence, Base SIMFE, which estimates a single $\tilde{m}(\cdot)$, essentially takes a weighted average of all the functions $\tilde{m}_{\mathbf{t}_i}(\cdot)$, which has the potential to produce biased parameter estimates. In this section we explore the *bias corrected SIMFE*, simply referred to as SIMFE from here on, which estimates $\tilde{m}_{\mathbf{t}_i}(\cdot)$ separately for each observation.

Recall from (18) that Base SIMFE approximates

$$E_Y\left[\left(Y - \tilde{m}(\tilde{\mathbf{P}})\right)^2 |\mathbf{X} = \mathbf{X}_i\right] \tag{30}$$

using a weighted sum of squared errors between Y_l and $\tilde{m}(\hat{\mathbf{P}}_l)$, where the weight is a decreasing function of the distance between $\hat{\mathbf{P}}_i$ and $\hat{\mathbf{P}}_l$. This weighting scheme makes sense if $\tilde{m}(\cdot)$ is the same for all observations but may be inappropriate if $\mathbf{X}_i(t)$ and $\mathbf{X}_l(t)$ are measured over a very different set of time points, because in that case $\tilde{m}_{\mathbf{t}_i}(\cdot) \neq \tilde{m}_{\mathbf{t}_l}(\cdot)$. The bias corrected implementation of SIMFE overcomes this deficiency by observing that, for a smooth $m(\cdot)$, if the distributions of $\mathbf{U}_i$ and $\mathbf{U}_l$ are similar then $\tilde{m}_{\mathbf{t}_i}(\cdot) \approx \tilde{m}_{\mathbf{t}_l}(\cdot)$. Hence, a more appropriate approximation to (30) is given by

$$\sum_{l=1}^{n}\left(Y_l - a_i - \sum_{j=1}^{p}\hat{\mathbf{P}}_{lj}\mathbf{c}_{ij}\right)^2 \tilde{K}_{il}, \tag{31}$$

where $\tilde{K}_{il}$ is a decreasing function of *both* the distance between $\hat{\mathbf{P}}_i$ and $\hat{\mathbf{P}}_l$ and the difference in the distributions between $\mathbf{U}_i$ and $\mathbf{U}_l$. This is achieved by setting

$$\tilde{K}_{il} = K_{il}^{(1)} \times K_{il}^{(2)}, \tag{32}$$

with $K_{il}^{(1)} = K_{h_1}(\hat{\mathbf{P}}_i - \hat{\mathbf{P}}_l)$ and $K_{il}^{(2)} = K_{h_2}(\hat{D}_{il})$, where $\hat{D}_{il}$ is an estimate of $D_{KL}(\mathbf{U}_i, \mathbf{U}_l)$, the Kullback-Leibler divergence between the probability distributions of $\mathbf{U}_i$ and $\mathbf{U}_l$.

Let $\boldsymbol{\eta}_j$ be a q_j by d_j matrix, whose k-th column is $\boldsymbol{\eta}_{jk}$, and define $[\boldsymbol{\eta}]$ as a block diagonal matrix, with the j-th block given by $\boldsymbol{\eta}_j$. One can show that $\mathbf{U}_i = (\mathbf{U}_{i1}, \cdots, \mathbf{U}_{ip}) \sim N(\mathbf{0}, \Sigma_i)$ where

$$\Sigma_i = [\boldsymbol{\eta}]^T (I - \Omega_i S_i)\Delta(I - \Omega_i S_i)^T[\boldsymbol{\eta}] + \sigma^2[\boldsymbol{\eta}]^T\Omega_i\Omega_i^T[\boldsymbol{\eta}], \tag{33}$$

and $\Omega_i = (\Delta^{-1} + \frac{1}{\sigma^2}S_i^T S_i)^{-1}S_i^T/\sigma^2$. Let $d = \sum_{j=1}^{p} d_j$. The Kullback-Leibler divergence between the multivariate normal distributions of $\mathbf{U}_i$ and $\mathbf{U}_l$ is given by

$$D_{KL}(\mathbf{U}_i, \mathbf{U}_l) = \frac{1}{2}\left(\mathrm{tr}(\Sigma_l^{-1}\Sigma_i) - \log\left(\frac{\det \Sigma_i}{\det \Sigma_l}\right) - d\right),$$

and $\hat{D}_{il}$ is obtained by replacing the parameters Δ and σ^2 and $[\boldsymbol{\eta}]$ in the above formula with their estimates. Thus, given a set of time points, one can estimate the Kullback-Leibler divergence between $\mathbf{U}_i$ and $\mathbf{U}_l$ for any i and l, and hence obtain $K_{il}^{(2)}$ and $\tilde{K}_{il}$.

The new implementation of SIMFE still uses the same algorithm to minimize (21) in the scalar response setting, or to minimize (28) for functional responses, with the only change being that the kernel function K is replaced by the kernel $\tilde{K}$, defined in (32). Notice that if all predictors are observed at the same set of time points, then $S_i = S_l$ for all i and l. In this case $\hat{D}_{il} = 0$, so that $\tilde{K}_{il} \propto K_{h_1}(\hat{\mathbf{P}}_i - \hat{\mathbf{P}}_l)$ and the new implementation reduces to that of Base SIMFE. Alternatively, in situations where time points differ among observations, the new approach will place most weight on observations where i and l are similar both in terms of $\hat{\mathbf{P}}_i$ versus $\hat{\mathbf{P}}_l$ and in terms of the distributions of $\mathbf{U}_i$ and $\mathbf{U}_l$. The functional response version of SIMFE can be extended to implement the bias corrected approach using an analogous expansion of the kernel.

While the two SIMFE approaches are similar, there is a major distinction between the methods in terms of their estimates for $\tilde{m}(\cdot)$. Consider for example an extreme situation where the predictors are all observed over one of two possible time point configurations, $\mathbf{t}_1$ and $\mathbf{t}_2$. Then Base SIMFE would ignore the difference in the time points and produce a single estimate for $\tilde{m}(\cdot)$. However, the new implementation of SIMFE would essentially take the observations from one configuration to estimate $\tilde{m}_{\mathbf{t}_1}(\cdot)$ and use the remaining observations to estimate $\tilde{m}_{\mathbf{t}_2}(\cdot)$. If the two $\tilde{m}(\cdot)$ functions were sufficiently different from each other, we would expect the new SIMFE estimate to be superior to that of the Base SIMFE.

We will predict a response Y^* based on a new observation $\mathbf{W}^*$ using the same approach as the Base SIMFE (25). The only difference is that the weight w_{*i} is computed using $\tilde{K}$ rather than K.

2.5 Selection of tuning parameters

Implementing SIMFE requires selecting q_j (the dimension of the basis function $\mathbf{s}_j(t)$) and d_j (the dimension into which $X_{ij}(t)$ will be projected). We use K-fold cross-validation to obtain q_j. In particular, for a given candidate value of q_j we remove $1/K$th of the observed time points for each $X_{ij}(t)$ as a validation set, fit a random effects model to the remaining observations, calculate the squared error between $\mathbf{W}_{ij}$ and $\hat{X}_{ij}(t)$ on the validation set, and repeat the procedure K times until each time point has been held out once. This procedure is repeated over a grid of q_j values, and the dimension corresponding to the lowest cross-validated squared error is selected. We repeat this approach for each of the p predictors.

In the implementation of the Base SIMFE approach the final bandwidth, h_{opt}, is set proportional to $n^{-1/(d+4)}$, where d is the total reduced dimension for the predictors. In the bias corrected implementation the reduced dimension d is increased to account for the fact that the link $\tilde{m}$ in equation (29) is a function not only of the projection $\tilde{\mathbf{P}}_i$, but also of the matrix Σ_i. Thus, the new reduced dimension, $\tilde{d}$, is set equal to the sum of d and the number of unique nonzero elements in Σ_i, i.e. $\tilde{d} = d + d(d+1)/2$. The final bandwidth is then set proportional to $n^{-1/(\tilde{d}+4)}$. Because we use the Kullback-Leibler divergence in the definition of the kernel, the choice of the proportionality constant in equation (20) is no longer justified. For the empirical work in this paper, we set this constant to 2, which gave good results in all the settings we considered.

Cross-validation could also be used to select d_j, but this approach suffers from a heavy computational burden. Instead, we extend the criterion that Li *et al.* (2010) developed for g-MAVE. Let $L(d_{c1}, \ldots, d_{cp})$ be the minimum value of the SIMFE criterion function for a candidate set of dimensions $\{d_{c1}, \ldots, d_{cp}\}$. Define $d_c = d_{c1}+\cdots+d_{cp}$, set $\tilde{d}_c = d_c+d_c(d_c+1)/2$, and let $h_n = h_n(\tilde{d}_c)$ be the bandwidth chosen as described in the paragraph above. We estimate dimensions $d_1, \ldots, d_p$ as $\widehat{d}_1, \ldots, \widehat{d}_p$, which are selected to minimize the following finite sample criterion:

$$\log\left(L(d_{c1}, \ldots, d_{cp})\right) + (d_c \log n)/(n{h_n}^{\tilde{d}_c}). \tag{34}$$

The next result, which is proved in the Supplementary Material, demonstrates that this criterion leads to consistent estimation of the dimensions d_j.

Theorem 2 *Under assumptions A1, A2, C3-C5, given in the Appendix,*

$$P\{(\widehat{d}_1, \ldots, \widehat{d}_p) = (d_1, \ldots, d_p)\} \to 1 \ \ as \ \ n \to \infty.$$

3 Theoretical Results

In the standard non-functional setting there exist index model fitting methods for which one can prove a fast rate of convergence to the true subspace as $n \to \infty$ and $h \to 0$. However, these results are not appropriate for our situation, because they assume a standard fully observed non-functional predictor and a scalar response. The SIMFE setting is considerably more complicated, because it involves functional data which, for sparsely sampled predictors, are observed subject to measurement error. In addition, the response may also be functional. Our main theoretical contribution is to establish that despite these added complications SIMFE still attains a fast convergence rate. In Theorem 3 we consider scalar responses and then, in Theorem 4, extend the result to functional responses.

As with existing results in the non-functional setting (Xia, 2008; Li *et al.*, 2010), throughout this section we assume that the true values of the reduced dimensions are known for each predictor. Our first result corresponds to the estimator obtained by the scalar response SIMFE algorithm. Note that in the standard non-functional setting the rate of convergence for the gOPG estimator, which we use to initialize SIMFE, is $n^{-2/(\tilde{p}+4)}$ (Li *et al.*, 2010). Theorem 3 below shows that the rate for the SIMFE estimator is significantly better.

In order to state our results we need to define a distance measure between a true index function, $\boldsymbol{\beta}_j(t) = (\beta_{j1}(t), \ldots, \beta_{jd_j}(t))^T$, and the corresponding SIMFE estimate, $\hat{\boldsymbol{\beta}}_j(t) = (\hat{\beta}_{j1}(t), \ldots, \hat{\beta}_{jd_j}(t))^T$. Since $\hat{\boldsymbol{\beta}}_j(t)$ defines a linear projection of the predictor, $X_{ij}(t)$, it is non-unique up to a linear transformation. In other words, $\hat{\boldsymbol{\beta}}_j(t)$ and $\tilde{\boldsymbol{\beta}}_j(t) = A_j\hat{\boldsymbol{\beta}}_j(t)$ both project $X_{ij}(t)$ into the same lower dimensional space for any invertible d_j by d_j matrix, A_j. Hence, we define the distance between $\hat{\boldsymbol{\beta}}_j(t)$ and $\boldsymbol{\beta}_j(t)$ as

$$r\left(\hat{\boldsymbol{\beta}}_j(t), \boldsymbol{\beta}_j(t)\right) = \min_{A_j \in R^{d_j} \times R^{d_j}} \|A_j\hat{\boldsymbol{\beta}}_j(t) - \boldsymbol{\beta}_j(t)\|,$$

where $\|\tilde{\boldsymbol{\beta}}_j(t) - \boldsymbol{\beta}_j(t)\|^2 = \sum_{k=1}^{d_j} \int \left(\tilde{\beta}_{jk}(t) - \beta_{jk}(t)\right)^2 dt$. Theorem 3 establishes the rate at which $r\left(\hat{\boldsymbol{\beta}}_j(t), \boldsymbol{\beta}_j(t)\right)$ converges to zero as n tends to infinity.

To prove Theorem 3 we impose smoothness and regularity conditions A1-A5, listed in the Appendix. We also suppose that we are given a finite, but possibly large, collection of potential time points, $\mathcal{T} = \{T_1, ..., T_L\}$. We assume that time points $t_{ij1}, ..., t_{ijT_{ij}}$ are randomly generated from the set $\mathcal{T}$, where the number of time points, T_{ij}, is also randomly generated.

Theorem 3 *Suppose that assumptions A1-A5, stated in the Appendix, are satisfied. Then, the SIMFE estimator satisfies*

$$r\left(\hat{\boldsymbol{\beta}}_j(t), \boldsymbol{\beta}_j(t)\right) = O_p\left(h_{opt}^4 + \frac{\log n}{nh_{opt}^d} + n^{-1/2}\right) = O_p\left(n^{-\frac{4}{d+4}}\log n + n^{-1/2}\right),$$

for $j = 1, \ldots, p$.

To understand the resulting rate of convergence, let us consider some special cases. For $d \leq 3$ the SIMFE estimator achieves the parametric rate of convergence, $n^{-1/2}$, while for $d = 4$ the parametric rate is attained up to a $\log n$ factor. For $d \geq 5$ the rate of convergence is $n^{-4/(d+4)} \log n$.

The following result provides the rate of convergence in the functional response case. We assume that the time points at which the response is observed are generated from a continuous distribution; the details are given in the Appendix.

Theorem 4 *Suppose that assumptions A1-A3, B4 and B5, stated in the Appendix, are satisfied. Then, the estimator obtained from the functional response SIMFE algorithm satisfies*

$$r\left(\hat{\boldsymbol{\beta}}_j(t), \boldsymbol{\beta}_j(t)\right) = O_p\left(n^{-\frac{4}{d+5}}\log n + n^{-1/2}\right),$$

for $j = 1, \ldots, p$.

Note that for $d \leq 2$ the functional SIMFE estimator achieves the parametric rate of convergence, $n^{-1/2}$. For $d = 3$ the rate of convergence is $n^{-1/2} \log n$, and for $d \geq 4$ it is $n^{-4/(d+5)} \log n$. The rate of convergence in the functional response setting is generally slower than that in the scalar response case because the dependence of the link function on the time parameter needs to be estimated.

Theorems 3 and 4 assume that the time points are generated from a finite collection. The next result corresponds to the case where time point locations come from a continuous distribution. Let $\hat{\Sigma}_i$ be the estimate of Σ_i in (33) and define $\hat{\boldsymbol{\xi}}_i$ as a vectorized version of the unique elements of $\hat{\Sigma}_i$, i.e. those on and above the diagonal. Let $\boldsymbol{\xi}_i$ be the corresponding vectorized form of Σ_i. Note that because $\boldsymbol{\xi}_i$ uniquely defines the distribution of $\mathbf{U}_i$, the link $m(\cdot)$ is a function of both $\mathbf{P}_i$ and $\boldsymbol{\xi}_i$, i.e. $m(\mathbf{P}_i, \boldsymbol{\xi}_i)$. Hence, in order to achieve a faster rate of convergence we slightly modify approximation (31) by including an additional term corresponding to $\hat{\boldsymbol{\xi}}_i$:

$$\sum_{l=1}^{n} \left(Y_l - a_i - \sum_{j=1}^{p} \hat{\mathbf{P}}_{lj} \mathbf{c}_{ij} - \hat{\boldsymbol{\xi}}_i \tilde{\mathbf{c}}_i \right)^2 \tilde{K}_{il}. \tag{35}$$

This approach uses local linear smoothing not only with respect to $\hat{\mathbf{P}}_i$, but also with respect to $\hat{\boldsymbol{\xi}}_i$. We also replace h_{opt} with $\tilde{h}_{opt} = n^{-1/(\tilde{d}+4)}$, where $\tilde{d} = d + d(d+1)/2$, as the final bandwidth.

Theorem 5 *Suppose that the number of time points and their locations are randomly generated for each observation. Under assumptions A1, A2, C3-C5, given in the Appendix, the SIMFE estimator satisfies*

$$r\left(\hat{\boldsymbol{\beta}}_j(t), \boldsymbol{\beta}_j(t)\right) = O_p\left(n^{-4/(\tilde{d}+4)} \log n + n^{-1/2}\right),$$

for $j = 1, \dots, p$.

In the setting of Theorem 5, the SIMFE estimator generally converges at a slower rate than the one in Theorem 3, where the time points are generated from a finite collection. This is due to the increased dimensionality of the problem. However, note that when $d = 1$ the estimator still achieves the parametric rate of convergence, $n^{-1/2}$. When the dimension d is large, the convergence can be slow. This is the case when, for example, the number of available predictor functions is large. In this situation, a more reasonable approach would be to add a group penalty on the estimated index coefficients to the SIMFE estimation criterion, which would force some of the predictors to be excluded from the estimated model.

However, we believe that in a number of settings it would be appropriate to assume the time points are sampled from a finite set of possibilities, in which case the faster rate of convergence in Theorem 3 would apply. For example in an experimental design setting, one would often sample patients at a finite (and common) set of time points. Our results suggest that SIMFE would perform better in this setting relative to generating the time points from a continuous distribution.

The proofs of all the results in this section are provided in the Supplementary Material.

4 Simulation Analysis

To evaluate the finite sample performance of our methods we test SIMFE out over two general scenarios; scalar responses and functional responses. In each setting we perform 100 simulation runs and compare the two versions of SIMFE with three competing methods: groupwise Outer Product of Gradients (gOPG) (Li *et al.*, 2010), groupwise Sliced Inverse Regression (gSIR) (Li, 2009) and groupwise Principal Components Analysis (gPCA). The three competing approaches are implemented as follows. We start by computing $\hat{X}_{ij}(t)$, as described in Step 1 of both the scalar and functional response SIMFE algorithms. This step is exactly the same for all the methods, including SIMFE. We then use the estimated basis coefficients, $\hat{\boldsymbol{\mu}}_{ij}$, to produce estimates for the index functions. All the methods proceed as they would in the non-functional setting by treating $\hat{\boldsymbol{\mu}}_{ij}$ as the observed predictors. For all methods we use local linear smoothing to estimate the link function, which we use to calculate the prediction error. Note that the gOPG implementation is equivalent to applying the first step of SIMFE without iterating. As a consequence, the SIMFE results provide a measure of the relative improvement from applying our iterative fitting procedure.

We use the cross-validation approach described in Section 2.5 to select the basis dimension q_j for all the methods. We also use the criterion given by (34) to select $d_1, \ldots, d_p$ for gOPG and gSIR. However, we found that gPCA performed poorly using (34). Hence, we generated a separate validation data set for each simulation run and selected the number of principal components that provides the best fit to the predictors on the validation data. While this provided a slight advantage for gPCA relative to the other methods, the performance of gPCA improved significantly.

4.1 Scalar Gaussian Responses

In the first scenario we generate scalar Gaussian responses from a model involving three predictors, $X_1(t), X_2(t), X_3(t)$. The observed values of the predictors, $\mathbf{W}_{ij}$, are generated using equation (8) with $\sigma_j = 0.1$ for $j = 1, 2, 3$ and

$$X_{ij}(t) = a_{ij0} + \sum_{k=1}^{2} b_{ijk} \sin(k\pi t) + \sum_{k=1}^{2} c_{ijk} \cos(k\pi t). \tag{36}$$

The corresponding index functions are similarly generated by

$$\beta_j(t) = \tilde{a}_{j0} + \sum_{k=1}^{2} \tilde{b}_{jk} \sin(k\pi t) + \sum_{k=1}^{2} \tilde{c}_{jk} \cos(k\pi t),$$

where $a, b, c, \tilde{a}, \tilde{b}$ and $\tilde{c}$ are sampled from a standard normal. Finally, the responses come from the non-linear model

$$Y_i = m_i + \varepsilon_i = P_{i1} + \exp(0.8 P_{i2}) + \sin(0.5\pi P_{i3}) + \varepsilon_i, \tag{37}$$

where $\varepsilon_i \sim N(0, \sigma_y^2)$, $P_{ij} = \int X_{ij}(t)\beta_j(t)dt$ and $\sigma_y = 0.1$. This model corresponds to projecting each predictor down into a one-dimensional space i.e. $d_1 = d_2 = d_3 = 1$. To simulate a real world setting where the functional form of the $\beta_{jk}(t)$'s would be unknown, we fit SIMFE using a spline basis rather than the true Fourier basis from which the data was generated.

Seven separate simulation settings are considered. In the first five a, b and c are generated independently in (36) so the predictors are uncorrelated. These settings correspond to two different sample sizes ($n = 100$ and $n = 200$) and three different levels of sparsity for sampled predictors ($T_{ij} = 5$, $T_{ij} = 8$ and T_{ij} randomly sampled from the set $\{5, 6, 7, 8\}$). Different time points are randomly selected for each observation from a very dense grid of points. Thus, for example, the observed points for $i = 1$ differ from those for $i = 2$. In the last two simulation settings we take $n = 100$ and $T_{ij} = 5$. However, we generate correlated functional predictors by sampling the coefficients in (36) from a zero mean multivariate normal distribution with the diagonal elements of the covariance matrix set to one and off-diagonal elements all equal to either $\rho = 0.25$ or $\rho = 0.5$.

Figure 1 provides a graphical illustration of the results for the $n = 200$ and $T_{ij} = 8$ simulation setting. The black solid lines correspond to the three true $\beta_j(t)$'s from which the data were generated. The median most accurate estimate, over the 100 simulation runs, is also plotted for each of the five competing methods. Both SIMFE (blue) and Base SIMFE (red) provide the highest levels of accuracy.

Setting	Method	MSPE	$\beta_1(t)$	$\beta_2(t)$	$\beta_3(t)$
$T_{ij} = 5$ $n = 100$ $\rho = 0$	SIMFE	0.388 (0.010)	0.818(0.025)	0.776(0.030)	0.812(0.024)
	Base SIMFE	0.429(0.011)	0.794(0.027)	0.754(0.031)	0.769(0.027)
	gOPG	1.914(0.099)	0.446(0.036)	0.498(0.033)	0.408(0.029)
	gSIR	0.786(0.024)	0.719(0.029)	0.664(0.033)	0.603(0.028)
	gPCA	0.908(0.019)	0.480(0.028)	0.575(0.027)	0.396(0.024)
$T_{ij} = 5$ $n = 200$ $\rho = 0$	SIMFE	0.284(0.010)	0.721(0.032)	0.757(0.034)	0.762(0.029)
	Base SIMFE	0.314(0.011)	0.702(0.031)	0.727(0.033)	0.732(0.030)
	gOPG	1.745(0.100)	0.325(0.039)	0.522(0.038)	0.323(0.030)
	gSIR	0.461(0.014)	0.598(0.037)	0.656(0.038)	0.594(0.032)
	gPCA	0.712(0.015)	0.455(0.030)	0.623(0.026)	0.329(0.020)
$T_{ij} = 8$ $n = 100$ $\rho = 0$	SIMFE	0.174(0.006)	0.729(0.023)	0.694(0.024)	0.728(0.023)
	Base SIMFE	0.174(0.006)	0.711(0.025)	0.695(0.025)	0.697(0.025)
	gOPG	1.873(0.085)	0.059(0.006)	0.062(0.007)	0.056(0.005)
	gSIR	0.654(0.025)	0.366(0.027)	0.265(0.024)	0.246(0.025)
	gPCA	0.493(0.011)	0.508(0.030)	0.639(0.025)	0.320(0.019)
$T_{ij} = 8$ $n = 200$ $\rho = 0$	SIMFE	0.124(0.006)	0.896(0.015)	0.883(0.016)	0.864(0.016)
	Base SIMFE	0.121(0.006)	0.856(0.018)	0.862(0.017)	0.844(0.017)
	gOPG	2.017(0.077)	0.046(0.003)	0.048(0.003)	0.040(0.003)
	gSIR	0.304(0.011)	0.443(0.026)	0.444(0.028)	0.370(0.027)
	gPCA	0.384(0.009)	0.468(0.029)	0.653(0.025)	0.283(0.018)
$T_{ij} = 5, 6, 7, 8$ $n = 100$ $\rho = 0$	SIMFE	0.280(0.010)	0.759(0.028)	0.775(0.029)	0.772(0.026)
	Base SIMFE	0.298(0.010)	0.708(0.031)	0.742(0.031)	0.745(0.028)
	gOPG	1.701(0.108)	0.364(0.041)	0.447(0.042)	0.388(0.038)
	gSIR	0.721(0.028)	0.536(0.033)	0.531(0.036)	0.460(0.032)
	gPCA	0.673(0.016)	0.483(0.030)	0.598(0.027)	0.357(0.021)
$T_{ij} = 5$ $n = 100$ $\rho = 0.25$	SIMFE	0.662(0.024)	0.787(0.026)	0.787(0.027)	0.766(0.027)
	Base SIMFE	0.701(0.026)	0.763(0.027)	0.783(0.026)	0.756(0.028)
	gOPG	4.059(0.286)	0.234(0.025)	0.479(0.029)	0.232(0.020)
	gSIR	1.368(0.055)	0.689(0.030)	0.610(0.029)	0.607(0.029)
	gPCA	1.459(0.034)	0.205(0.018)	0.873(0.009)	0.148(0.011)
$T_{ij} = 5$ $n = 100$ $\rho = 0.50$	SIMFE	0.742(0.038)	0.773(0.027)	0.761(0.029)	0.756(0.026)
	Base SIMFE	0.780(0.036)	0.736(0.029)	0.752(0.029)	0.725(0.027)
	gOPG	4.336(0.493)	0.287(0.025)	0.504(0.027)	0.260(0.023)
	gSIR	1.689(0.048)	0.783(0.023)	0.614(0.023)	0.706(0.024)
	gPCA	1.419(0.037)	0.192(0.016)	0.862(0.008)	0.145(0.012)

Table 1: Scalar Gaussian Response for $d_1 = d_2 = d_3 = 1$: The mean squared prediction error, and correlation coefficients. Standard errors are shown in parentheses.

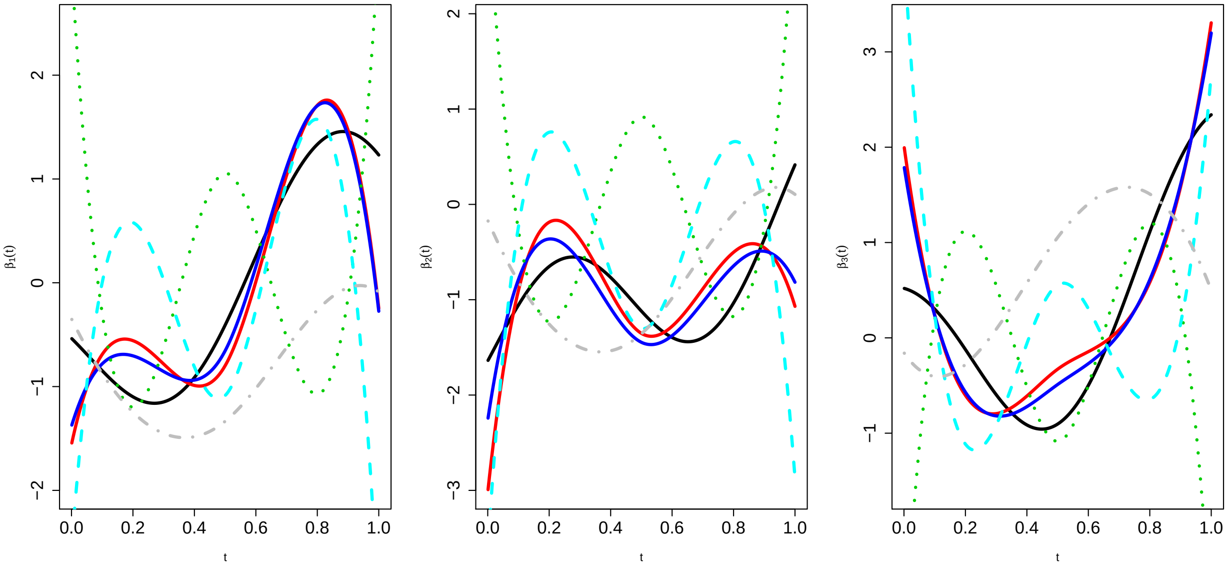

Figure 1: *Scalar Gaussian simulation with* $T_{ij} = 8$ *and* $n = 200$*: Comparison of true* $\beta(t)$ *functions (black solid) with median estimates over* 100 *simulation runs; SIMFE (blue solid), Base SIMFE (red solid), gOPG (green dotted), gSIR (cyan dashed) and gPCA (grey dash dotted).*

Table 1 compares SIMFE with gOPG, gSIR and gPCA over all seven simulations with mean prediction errors computed on a separate test set of size $n = 500$. We also provide the mean correlations between the estimated and true $\beta_j(t)$ curves, with numbers close to one demonstrating a good estimate. The correlations are calculated from the vector correlation (Hotelling, 1936) between discretized versions of the true and estimated curves. In this calculation, the curves are evaluated on the same dense grid that is used to generate the time points. All results are averaged over 100 simulation runs. As one would expect, the best results are obtained for the more densely sampled case, with larger sample size. In all seven simulations, the SIMFE approaches outperform the competing methods. In most cases the difference is large, and generally statistically significant. The SIMFE methods perform particularly well in terms of prediction error. As we would expect, SIMFE generally outperforms Base SIMFE in the sparsest settings, but the two methods give similar results in the denser scenario. It is also worth noting that gOPG, which is used as the initialization for SIMFE, provides significantly worse results, highlighting the improvement that is possible from the iterative SIMFE algorithm. The correct reduced dimensions are selected in all of

Setting	Method	MSPE	$\beta_1(t)$	$\beta_2(t)$
	SIMFE	0.280(0.009)	0.840(0.016)	0.832(0.027)
$T_{ij} = 5, n = 100$	Base SIMFE	0.520(0.020)	0.858(0.016)	0.809(0.028)
$\rho = 0$	gOPG	1.548(0.073)	0.647(0.021)	0.417(0.030)
	gSIR	0.736(0.027)	0.773(0.015)	0.771(0.026)
	gPCA	0.684(0.016)	0.868(0.007)	0.410(0.024)
	SIMFE	0.197(0.005)	0.853(0.014)	0.783(0.030)
$T_{ij} = 5, n = 200$	Base SIMFE	0.367(0.010)	0.867(0.013)	0.745(0.033)
$\rho = 0$	gOPG	1.414(0.062)	0.631(0.019)	0.349(0.032)
	gSIR	0.464(0.011)	0.789(0.016)	0.711(0.034)
	gPCA	0.559(0.011)	0.885(0.005)	0.323(0.020)
	SIMFE	0.137(0.005)	0.757(0.008)	0.716(0.025)
$T_{ij} = 8, n = 100$	Base SIMFE	0.253(0.009)	0.772(0.009)	0.654(0.026)
$\rho = 0$	gOPG	1.907(0.116)	0.567(0.014)	0.054(0.012)
	gSIR	0.646(0.035)	0.580(0.013)	0.372(0.025)
	gPCA	0.395(0.009)	0.915(0.003)	0.329(0.020)
	SIMFE	0.107(0.003)	0.787(0.012)	0.874(0.014)
$T_{ij} = 8, n = 200$	Base SIMFE	0.203(0.012)	0.811(0.012)	0.822(0.018)
$\rho = 0$	gOPG	1.611(0.035)	0.599(0.018)	0.045(0.010)
	gSIR	0.358(0.012)	0.629(0.012)	0.572(0.026)
	gPCA	0.306(0.006)	0.916(0.002)	0.275(0.018)

Table 2: Scalar Gaussian Response for $d_1 = 2$, $d_2 = 1$: The mean squared prediction error, and correlation coefficients. Standard errors are shown in parentheses.

the simulation runs using criterion (34).

In the next simulation, we consider an example with two predictors, where the responses come from a model with the following nonlinear function:

$$m_i = \frac{P_{i11}}{0.5 + (1.5 + P_{i12})^2} + P_{i21}, \tag{38}$$

where $P_{ijk} = \int X_{ij}(t)\beta_{jk}(t)dt$. This model projects the two predictors, respectively, into a two-dimensional and a one-dimensional space, i.e. $d_1 = 2$ and $d_2 = 1$. In all other respects the simulation setup is identical to our first simulation.

Table 2 reports numerical summaries for all four simulations. SIMFE does better than both gOPG and gSIR in estimating $\beta_1(t) = (\beta_{11}(t), \beta_{12}(t))$. gPCA gives a slightly better estimate for $\beta_1(t)$, but SIMFE is substantially superior to all three competing methods in estimating $\beta_2(t)$. Moreover, SIMFE significantly outperforms all of the competitors, includ-

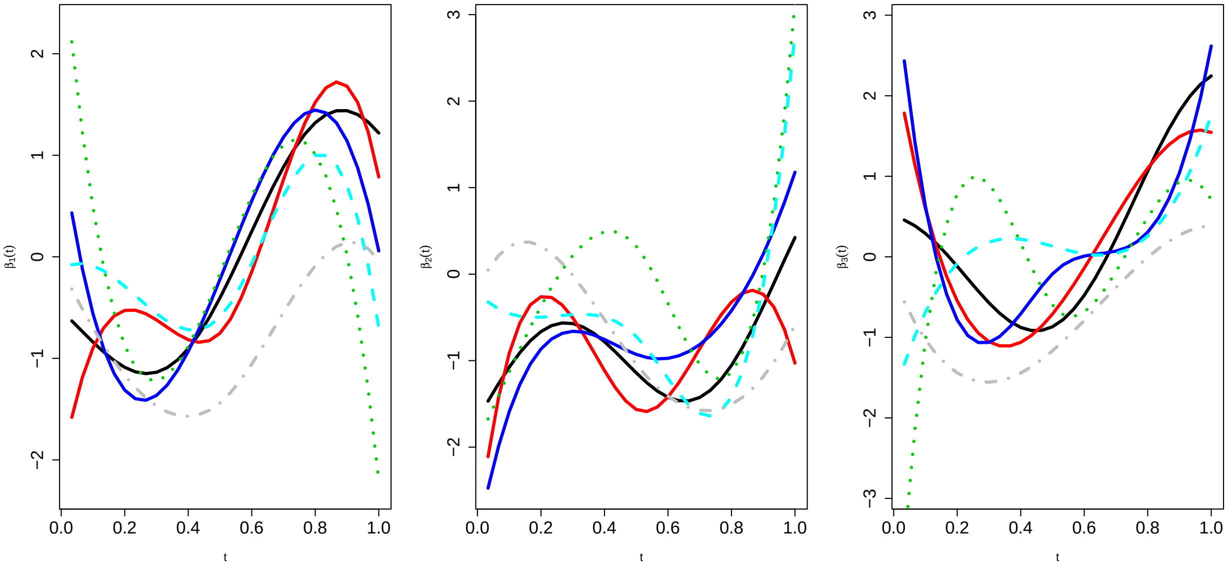

Figure 2: *Functional Gaussian Response for $T_{ij} = n_i = 5, n = 100$: Comparison of true $\beta_j(t)$ functions (black solid) with SIMFE (blue solid), Base SIMFE (red solid), gOPG(green dotted), gSIR(cyan dashed) and gPCA estimates(grey dash dotted).*

ing Base SIMFE, in terms of the prediction error. SIMFE choose the correct dimension, $d = (2, 1)^T$, on more than 90% of simulations, and selected $d = (1, 1)^T$ in the remainder.

4.2 Functional Gaussian Responses

In our third scenario the responses are generated as functions rather than scalars. In particular, each response function is randomly sampled at $n_i = 5$ or $n_i = 8$ different time points, $s_1, \ldots, s_{n_i}$, with

$$Y_i(s_{ik}) = m(s_{ik}, \mathbf{P}_i) + \varepsilon_{ik}, \quad \varepsilon_{ik} \sim N(0, \sigma_y^2),$$

where $m(s_{ik}, \mathbf{P}_i) = s_{ik} m_i$ and m_i is equal to the response function from (37) with $p = 3$ predictors. We only consider the $n = 100$ setting. The predictors and index functions are generated in the same fashion as for the scalar response simulations. To implement gOPG we again initialize $\beta_j(t)$ by applying gOPG to the estimated basis coefficients, $\hat{\boldsymbol{\mu}}_{ij}$, of the predictor functions and then use (42) to estimate $\tilde{m}(\cdot)$, i.e. a single iteration of the functional response version of SIMFE. gSIR and gPCA are computed in an analogous fashion.

Figure 2 illustrates the graphical results for the more sparsely sampled $n_i = T_{ij} = 5$ case, while Table 3 gives numerical summaries for both sparsity levels. The results in Figure 2 and

Setting	Method	MSPE	$\beta_1(t)$	$\beta_2(t)$	$\beta_3(t)$
$T_{ij} = 5, n = 100$	SIMFE	0.137 (0.007)	0.843(0.023)	0.793(0.030)	0.735(0.026)
	Base SIMFE	0.148(0.008)	0.838(0.025)	0.781(0.031)	0.726(0.033)
	gOPG	0.907(0.042)	0.409(0.031)	0.429(0.037)	0.285(0.028)
	gSIR	0.435(0.012)	0.737(0.032)	0.643(0.034)	0.545(0.031)
	gPCA	0.414(0.014)	0.483(0.031)	0.593(0.028)	0.384(0.027)
$T_{ij} = 8, n = 100$	SIMFE	0.073(0.006)	0.793(0.027)	0.754(0.024)	0.763(0.026)
	Base SIMFE	0.074(0.006)	0.786(0.028)	0.744(0.024)	0.752(0.026)
	gOPG	0.797(0.039)	0.180(0.016)	0.079(0.015)	0.123(0.020)
	gSIR	0.277(0.010)	0.681(0.031)	0.521(0.028)	0.248(0.025)
	gPCA	0.197(0.009)	0.346(0.028)	0.589(0.027)	0.401(0.022)

Table 3: Functional Gaussian Response: The mean squared prediction error, and correlation coefficients. Standard errors are shown in parentheses.

Table 3 are generally consistent with those in the scalar case. In every simulation setting SIMFE provides improved estimates of the index functions, and lower prediction errors, relative to gOPG, gSIR and gPCA. In most cases the differences are highly statistically significant. The correct reduced dimensions are selected in all of the simulation runs using criterion (34). When compared to its Base version across all the simulation settings, SIMFE performs better with respect to the prediction error. The advantage of SIMFE is especially prominent in the sparsest simulation scenarios, corresponding to $T_{ij} = 5$.

5 Auction Data

Online auctions have attracted considerable attention in recent years. EBay, the world's largest consumer-to-consumer electronic commerce company, provides a convenient auction site for global sellers and buyers to trade with each other through the internet. The majority of eBay's revenue is generated through fees for listing and selling auctions. EBay makes the historical auction data for millions of different types of products publicly accessible. Here we consider the most common single-item online auctions, where bidders place their orders during a fixed auction duration. Those who submit the highest bid before the closing time win the auction but only need to pay the second highest bid. eBay does not display the highest bid, but rather the second highest bid (plus a small bid increment). This is called the *live* bid. The price histories of these live bids from auctions of similar products can be viewed as i.i.d realizations of bid trajectories. Functional data analysis (FDA) provides a powerful tool to deal with such online auction data, see Jank and Shmueli (2005); Shmueli and Jank (2005); Reddy and Dass (2006); Reithinger *et al.* (2008); Wang and Shmueli (2008) and Liu and Muller (2008). The sequence of observed bids differs from the values of the smooth underlying bid trajectory. These differences can be viewed as random aberrations of bids, but we will treat them as "measurement error".

Here we examine 156 eBay online 7-day second price auctions of Palm M515 Personal Digital Assistants (PDA) that took place between March and May, 2003 (Liu and Muller, 2008). Our interest is in predicting Y = "closing price" based on observing the auction trajectory, $X(t)$, up to time $0 \leq t \leq T$. Traditional functional data analysis methods require regular and dense data, but auction data are generally sparsely and irregularly observed with measurement errors. Examination of the PDA data confirms the presence of "bid sniping" i.e. where bids are very sparse in the middle of the auction, a little denser in the beginning and much denser at the end of the auction. Each auction contains 9 to 52 sparse and irregular observations of live bids. We converted the bid time to hours, $t \in [0, 168)$, and applied a log transformation to the bid values.

Figure 3 plots the estimated time varying $\beta_1(t)$'s using SIMFE, Base SIMFE, gOPG, gSIR and gPCA. Each plot represents a different subset of the data for $T = 138, 144, \ldots, 168$. For example, the top left figure corresponds to $T = 138$ where we only considered auction trajectories up to $t \leq 138$ i.e. 30 hours prior to the end of the auction. A few trends are apparent. First, SIMFE (red dashed) and Base SIMFE (black solid) give very similar estimates and both methods place most of the weight on the bid trajectories after $t = 100$ hours. These results are consist across different values of T and seem reasonable given that the most recent bids contain the most information about the final auction price. Second, for the larger values of T, gPCA (cyan long dash) places roughly equal weight on all sections of the bid trajectory, while gOPG (green dotted) and gSIR (blue dash-dot) produce wildly varying results that are hard to justify based on the context of the problem.

To judge which projection produced superior predictions, we computed the 5-fold cross-validated prediction errors for SIMFE and Base SIMFE. We also implement a hybrid approach, gSIMFE, which uses SIMFE to compute P_i, but then predicts the response by applying generalized additive models (GAM) with P_i as the predictor. We compared the three SIMFE methods to similar implementations of gOPG, gSIR and gPCA, where each method was used to estimate P_i, and then GAM was fitted to give a predicted response.

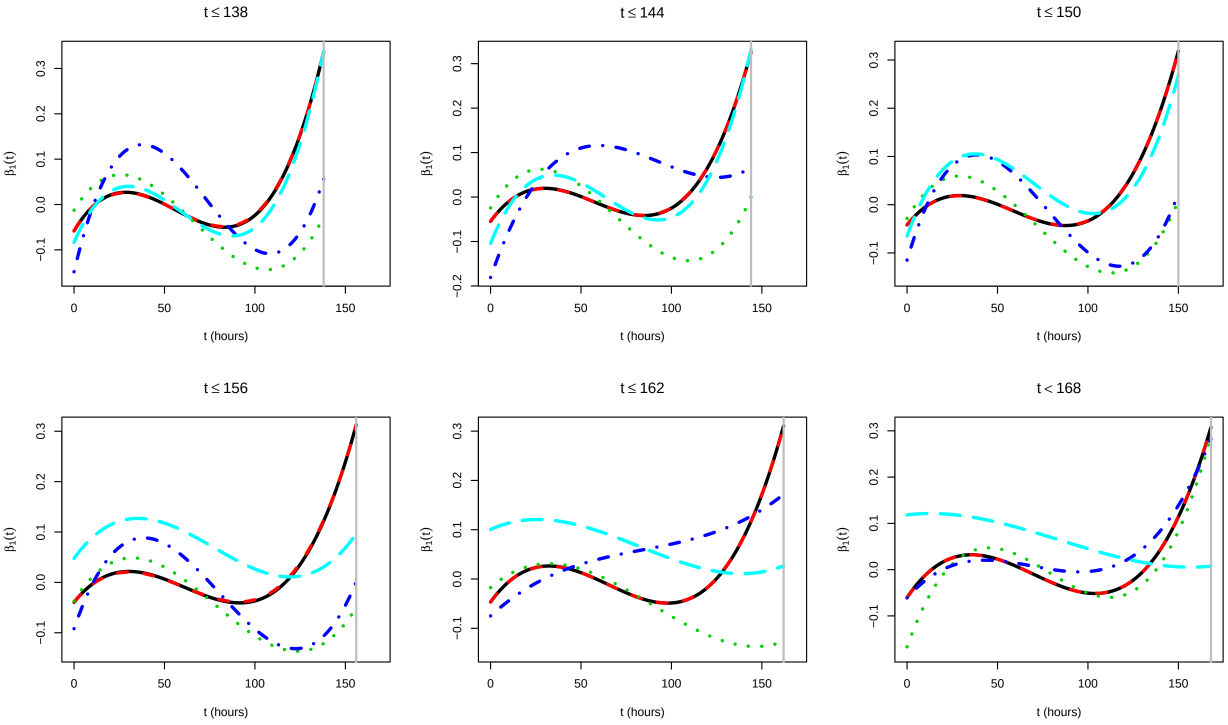

Figure 3: *Estimated* $\beta_{jk}(t)$ *curves for eBay online Auction Data. The red dashed, black solid, green dotted, blue dash-dot and cyan long dashed lines correspond to SIMFE, Base SIMFE, gOPG, gSIR and gPCA, respectively, up to different current times,* $T = 138, 144, 150, 156, 162$ *and* 168 *hours.*

Note that for gPCA we fitted GAM to the first 3 principal components, which explained more than 95% of the variation in the bid trajectory. The resulting cross-validated errors, for various values of T, are provided in Table 4. We have also included the error rates from the null model, using the mean of the training response to predict the test response. gSIMFE gave the best results, though SIMFE and Base SIMFE were only slightly inferior. For larger values of T all three SIMFE methods outperformed the competitors.

Acknowledgments

We would like to thank the Editor, the Associate Editor and two referees for their useful comments and suggestions.

Method	$t \leq 138$	$t \leq 144$	$t \leq 150$	$t \leq 156$	$t \leq 162$	$t < 168$
SIMFE	6.502	6.536	6.452	6.281	6.202	4.435
Base SIMFE	6.481	6.533	6.471	6.286	6.194	4.552
gSIMFE	**6.390**	**6.381**	**6.263**	**6.146**	**6.191**	**4.283**
gOPG	7.139	6.601	7.056	6.926	6.444	5.541
gSIR	6.413	6.543	6.380	6.387	6.253	5.023
gPCA	6.996	6.486	6.721	6.584	6.240	6.109
Mean	7.211	7.211	7.211	7.211	7.211	7.211

Table 4: Cross-validated mean squared prediction errors ($\times 10^{-3}$) for the three versions of SIMFE and four competing methods. The lowest MSPE for each value of T is bolded.

A EM Algorithm for Estimating $\tilde{X}_{ij}(t)$

Standard calculations show that the expected value of the joint likelihood for $\mathbf{X}$ and $\mathbf{W}$ is maximized by setting

$$\hat{\boldsymbol{\mu}}_\delta = \frac{1}{n}\sum_{i=1}^n E\left[\boldsymbol{\delta}_i | \mathbf{W}_i, \hat{\boldsymbol{\mu}}_i\right] = \frac{1}{n}\sum_{i=1}^n \hat{\boldsymbol{\mu}}_i, \tag{39}$$

$$\begin{aligned} \hat{\Delta} &= \frac{1}{n}\sum_{i=1}^n E\left[(\boldsymbol{\delta}_i - \boldsymbol{\mu}_\delta)(\boldsymbol{\delta}_i - \boldsymbol{\mu}_\delta)^T | \mathbf{W}_i, \hat{\boldsymbol{\mu}}_i, \tilde{\Delta}, \hat{\boldsymbol{\mu}}_\delta\right] \\ &= \frac{1}{n}\sum_{i=1}^n \left((\hat{\boldsymbol{\mu}}_i - \hat{\boldsymbol{\mu}}_\delta)(\hat{\boldsymbol{\mu}}_i - \hat{\boldsymbol{\mu}}_\delta)^T + \tilde{\Delta}\right) \end{aligned} \tag{40}$$

and

$$\begin{aligned} \hat{\sigma}^2 &= \frac{1}{\sum_{i=1}^n \sum_{j=1}^p T_{ij}} \sum_{i=1}^n E\left[(\mathbf{W}_i^T - S\boldsymbol{\delta}_i)^T(\mathbf{W}_i^T - S\boldsymbol{\delta}_i) | \mathbf{W}_i, \hat{\boldsymbol{\mu}}_i, \tilde{\Delta}\right] \\ &= \frac{1}{\sum_{i=1}^n \sum_{j=1}^p T_{ij}} \sum_{i=1}^n \left((\mathbf{W}_i^T - S\hat{\boldsymbol{\mu}}_i)^T(\mathbf{W}_i^T - S\hat{\boldsymbol{\mu}}_i) + \operatorname{trace}\left(S\tilde{\Delta}S^T\right)\right). \end{aligned} \tag{41}$$

Hence, the following EM algorithm can be applied to efficiently estimate the required parameters. Note that when $\mathbf{t}_i$ differ among individuals, we replace matrix S in the formulas above with S_i.

B Fitting Functional Response SIMFE

We use a similar two step iteration for the functional response version of SIMFE. The steps are summarized in Algorithm 3.

Algorithm 2 EM Algorithm

1. Compute $\hat{\boldsymbol{\mu}}_\delta, \hat{\Delta}$ and $\hat{\sigma}^2$ via Equations (39), (40) and (41).

2. Use the current parameter estimates to update $\hat{\boldsymbol{\mu}}_i$ and $\tilde{\Delta}$ via Equations (14) and (15).

3. Repeat Steps 1. and 2. until convergence.

Step One

Let $\boldsymbol{\gamma}_{ik} = \left(a_{ik}, c^s_{ik}, \mathbf{c}^T_{ik1}, \ldots, \mathbf{c}^T_{ikp}\right)^T$ and $\mathbf{R}_{lk'} = (1, s_{lk'}, \hat{\mathbf{P}}_{l1}, \ldots, \hat{\mathbf{P}}_{lp})^T$ then (28) can be written as,

$$\frac{1}{\sum_{i=1}^n n_i} \sum_{i,k} \sum_{l,k'} K_{iklk'} \left(Y_{lk'} - \mathbf{R}^T_{lk'} \boldsymbol{\gamma}_{ik}\right)^2,$$

whose minimum is obtained by setting,

$$\hat{\boldsymbol{\gamma}}_{ik} = \left(\sum_{l,k'} K_{iklk'} \mathbf{R}_{lk'} \mathbf{R}^T_{lk'}\right)^{-1} \sum_{l,k'} K_{iklk'} Y_{lk'} \mathbf{R}_{lk'}, \tag{42}$$

for $i = 1, \ldots, n$ and $k = 1, \ldots, n_i$.

Step Two

Next, we estimate $\beta_{jk}(t)$ for $j = 1, \ldots, p$ and $k = 1, \ldots, d_j$, given the current estimate for $m(s, \hat{\mathbf{P}}_1, \ldots, \hat{\mathbf{P}}_p)$. Let

$$Q_{ikl} = \begin{pmatrix} \mathbf{c}_{ik1} \otimes \int \hat{X}_{l1}(t) \mathbf{s}_j(t) dt \\ \vdots \\ \mathbf{c}_{ikp} \otimes \int \hat{X}_{lp}(t) \mathbf{s}_j(t) dt \end{pmatrix} = \begin{pmatrix} \mathbf{c}_{ik1} \otimes \hat{\boldsymbol{\mu}}_{l1} \\ \vdots \\ \mathbf{c}_{ikp} \otimes \hat{\boldsymbol{\mu}}_{lp} \end{pmatrix}$$

and $\boldsymbol{\eta} = (\boldsymbol{\eta}_{11}, \ldots, \boldsymbol{\eta}_{1d_1}, \ldots, \boldsymbol{\eta}_{p1}, \ldots, \boldsymbol{\eta}_{pd_p})^T$, where $\otimes$ is the Kronecker product. It is not hard to show that the estimate for $\boldsymbol{\eta}$ that minimizes (28) is given by,

$$\hat{\boldsymbol{\eta}} = \left(\sum_{i,k} \sum_{l,k'} K_{iklk'} Q_{ikl} Q^T_{ikl}\right)^{-1} \sum_{i,k} \sum_{l,k'} K_{iklk'} Q_{ikl} (Y_{lk'} - a_{ik} - c^s_{ik} s_{lk'}). \tag{43}$$

Hence,

$$\hat{\beta}_{jk}(t) = \mathbf{s}_j(t)^T \hat{\boldsymbol{\eta}}_{jk}, \quad j = 1, \ldots, p, \quad k = 1, \ldots, d_j, \tag{44}$$

where $\hat{\boldsymbol{\eta}}_{jk}$ represent the corresponding elements of $\hat{\boldsymbol{\eta}}$.

Algorithm 3 Functional Response SIMFE Algorithm

```
1. Compute $\hat{X}_{ij}(t)$'s by plugging parameter estimates from the EM algorithm into (16).
2. Initialize $\hat{\beta}_{jk}(t)$ and $h \propto n^{-1/(\tilde{p}+4)}$ where $\tilde{p} = 1 + \sum_{j=1}^{p} q_j$.
3. (a) Estimate $m(s, \hat{\mathbf{P}})$ using the linear approximation given by (42).
   (b) Compute the $\hat{\beta}_{jk}(t)$'s via (43) and (44).
   (c) Repeat Steps (a) and (b) until convergence.
4. Set $h \leftarrow ch$ for some $c < 1$.
5. Iterate Steps 3 and 4 until $h \leq h_{opt}$, which is defined using $d + 1$ instead of $d$.
```

C Proof of Theorem 1

As a consequence of the SIMFE model, we have the following simple fact,

$$E(\varepsilon_i|\mathbf{W}_i) = E(E(\varepsilon_i|\mathbf{X}_i, \mathbf{W}_i)|\mathbf{W}_i) = 0.$$

We also have,

$$\begin{aligned} E(Y_i|\mathbf{W}_i) &= E\left(m\left(\mathbf{P}_{i1}, \ldots, \mathbf{P}_{ip}\right)|\mathbf{W}_i\right) + E\left(\varepsilon_i|\mathbf{W}_i\right) \\ &= E\left(m(\tilde{\mathbf{P}}_{i1} + \mathbf{U}_{i1}, \ldots, \tilde{\mathbf{P}}_{ip} + \mathbf{U}_{ip})|\mathbf{W}_i\right), \end{aligned}$$

where $\mathbf{U}_{ij} = \mathbf{P}_{ij} - \tilde{\mathbf{P}}_{ij}$. Under some mild integrability assumptions, $\mathbf{U}_{ij}$ are mean zero Gaussian random vectors, and $\tilde{\mathbf{P}}_{ij} = E(\mathbf{P}_{ij}|\mathbf{W}_i)$. Note that

$$E(W_{ijk}U_{ilm}) = E(E(W_{ijk}U_{ilm}|\mathbf{W}_i)) = E(W_{ijk}E(P_{ilm} - \tilde{P}_{ilm}|\mathbf{W}_i)) = 0,$$

for all i, j, k, l and m. Consequently, vectors $\mathbf{W}_i$ and $(\mathbf{U}_{i1}, \ldots, \mathbf{U}_{ip})$ are independent. Let $f_{U_i}(u)$ be the density of the second vector. Note that, according to the derivations in Section 2.4, the dependence of f_{U_i} on the index i is completely determined by $\mathbf{t}_i$, the time point configuration for the i-th observation. Write row vector u as $(u_1, ..., u_p)$, where $u_j \in \mathbb{R}^{d_j}$, and define $\tilde{m}_{\mathbf{t}_i}(s_1, ..., s_p)$, with $s_j \in \mathbb{R}^{d_j}$, as $\int m\left(s_1 + u_1, \ldots, s_p + u_p\right) f_{U_i}(u) du$. Then,

$$E\left(m(\tilde{\mathbf{P}}_{i1} + \mathbf{U}_{i1}, \ldots, \tilde{\mathbf{P}}_{ip} + \mathbf{U}_{ip})|\mathbf{W}_i\right) = \tilde{m}_{\mathbf{t}_i}(\tilde{\mathbf{P}}_{i1}, \ldots, \tilde{\mathbf{P}}_{ip}).$$

Hence,

$$Y_i = E(Y_i|\mathbf{W}_i) + \varepsilon_i^* = \tilde{m}_{\mathbf{t}_i}\left(\tilde{\mathbf{P}}_{i1}, \ldots, \tilde{\mathbf{P}}_{ip}\right) + \varepsilon_i^*,$$

where, by construction, $E(\varepsilon_i^*|\mathbf{W}_i) = 0$.

D Theoretical Assumptions for the Results in Section 3

Recall that $\boldsymbol{\eta}$ consists of components $\boldsymbol{\eta}_{jk}$, which are the vectors of basis coefficients for the true projection functions in the SIMFE model. We will use $\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i$ to denote the vector $(\boldsymbol{\eta}_{11}^T\tilde{\boldsymbol{\mu}}_{i1}, \dots, \boldsymbol{\eta}_{1d_1}^T\tilde{\boldsymbol{\mu}}_{i1}, \dots, \boldsymbol{\eta}_{p1}^T\tilde{\boldsymbol{\mu}}_{ip}, \dots, \boldsymbol{\eta}_{pd_p}^T\tilde{\boldsymbol{\mu}}_{ip})^T$ and define $\hat{\boldsymbol{\eta}} * \tilde{\boldsymbol{\mu}}_i$ and $\tilde{\boldsymbol{\eta}} * \tilde{\boldsymbol{\mu}}_i$ analogously. For technical reasons we will introduce trimming functions into the SIMFE optimization problem (21). This is a standard technical device, which serves the purpose of handling the notorious boundary points. Write $\tilde{\boldsymbol{\mu}}_i$ for $(\tilde{\boldsymbol{\mu}}_{i1}^T, ..., \tilde{\boldsymbol{\mu}}_{ip}^T)^T$ and define $I_{ni} = 1\{||\tilde{\boldsymbol{\mu}}_i|| \leq n\}$. Let ρ be some bounded function with a bounded second derivative, such that $\rho(v) > 0$ if $v > w_0$ and $\rho(v) = 0$ if $v \leq w_0$, for some small positive w_0. Define $\rho_{\hat{\boldsymbol{\eta}}i} = \rho(\hat{f}_{\hat{\boldsymbol{\eta}}}(\tilde{\boldsymbol{\mu}}_i))/\hat{f}_{\hat{\boldsymbol{\eta}}}(\tilde{\boldsymbol{\mu}}_i)$, with $\hat{f}_{\hat{\boldsymbol{\eta}}}(\tilde{\boldsymbol{\mu}}_i) = n^{-1}\sum_{k=1}^n K_h(\hat{\boldsymbol{\eta}} * (\tilde{\boldsymbol{\mu}}_k - \tilde{\boldsymbol{\mu}}_i))$. The only difference between the new objective function,

$$\frac{1}{n}\sum_{i=1}^n\sum_{l=1}^n I_{ni}\rho_{\hat{\boldsymbol{\eta}}i}\left(Y_l - a_i - \sum_{j=1}^p \mathbf{c}_{ij}^T\hat{P}_{lj}\right)^2 K_{il}, \tag{45}$$

and the one given by display (21) is the presence of the trimming term $I_{ni}\rho_{\hat{\boldsymbol{\eta}}i}$. The corresponding change in the SIMFE algorithm is that the trimming term also appears in formula (23).

The following assumptions are used in the proof of Theorem 3.

A1. The response has a finite fifth absolute moment, $E|Y|^5 < \infty$.

A2. Kernel function $\tilde{K}$ is a multivariate density function with bounded second-order derivatives and a compact support.

A3. For each k, i and j, time point T_k has a positive probability of being included in the time point configuration generated for the predictor curve $X_{ij}(t)$. For each j, the L by q_j dimensional basis matrix with k-th row $s_j(T_k)$ has rank q_j.

A4. For each i, function $E(Y_i|\tilde{\boldsymbol{\eta}} * \tilde{\boldsymbol{\mu}}_i = u)$ has bounded fourth-order derivatives with respect to u and $\tilde{\boldsymbol{\eta}}$, for $\tilde{\boldsymbol{\eta}}$ in a small neighborhood of $\boldsymbol{\eta}$.

A5. For each i, matrix $E[\rho(f_i(\tilde{\boldsymbol{\mu}}_i))\{\nabla\tilde{m}_i(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i) - \bar{\nabla}\}\{\nabla\tilde{m}_i(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i) - \bar{\nabla}\}^T]$ has full rank, where $\bar{\nabla} = E[\rho(f_i(\tilde{\boldsymbol{\mu}}_i))\{\nabla\tilde{m}_i(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i)]/E\rho(f_i(\tilde{\boldsymbol{\mu}}_i))$, and f_i is the density of $\tilde{\boldsymbol{\mu}}_i$.

For the case of functional response, considered in Theorem 4, we need to modify the last two assumptions.

B4. Time points at which the response is observed, s_{ik}, are independent realizations of a random variable S, and sequence $\{n_i\}$ is bounded. For each i, function $E(Y_i|S = s, \tilde{\boldsymbol{\eta}} * \tilde{\boldsymbol{\mu}}_i = u)$ has bounded fourth-order derivatives with respect to s, u and $\tilde{\boldsymbol{\eta}}$, for $\tilde{\boldsymbol{\eta}}$ in a small neighborhood of $\boldsymbol{\eta}$.

B5. For each i, matrix $E[\rho(f(S, \tilde{\boldsymbol{\mu}}_i))\{\nabla\check{m}(S, \boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i) - \bar{\nabla}\}\{\nabla\check{m}(S, \boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i) - \bar{\nabla}\}^T]$ has full rank, where $\bar{\nabla} = E[\rho(f(S, \tilde{\boldsymbol{\mu}}_i))\{\nabla\check{m}(S, \boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i)]/E\rho(f(S, \tilde{\boldsymbol{\mu}}_i))$, and f is the density of $(S, \tilde{\boldsymbol{\mu}}_i)$.

Now consider the setting of Theorem 5. In the Supplementary Material we show that the response satisfies the following model,

$$Y_i = \check{m}\left(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i, \boldsymbol{\gamma} * \mathbf{v}_i\right) + \varepsilon_i^*,$$

for some function $\check{m}$ and vector $\boldsymbol{\gamma}^*$, where vectors $\mathbf{v}_i$ consist of elements of the matrix $(I - \Omega_i S_i)\Delta(I - \Omega_i S_i)^T + \sigma^2 \Omega_i \Omega_i^T$. We will need the following assumptions on the components of this model.

C3. For each j, time points t_{ijk} are generated from the same continuous distribution with a density that is bounded away from zero on the domain of the j-th predictor.

C4. For each i, function $E(Y_i|\tilde{\boldsymbol{\eta}} * \tilde{\boldsymbol{\mu}}_i = u, \tilde{\boldsymbol{\gamma}} * \mathbf{v}_i = w)$ has bounded fourth-order derivatives with respect to u, w, $\tilde{\boldsymbol{\eta}}$ and $\tilde{\boldsymbol{\gamma}}$, for $(\tilde{\boldsymbol{\eta}}, \tilde{\boldsymbol{\gamma}})$ in a small neighborhood of $(\boldsymbol{\eta}, \boldsymbol{\gamma})$.

C5. For each i, matrix $E[\rho(f(\tilde{\boldsymbol{\mu}}_i, \mathbf{v}_i))\{\nabla\check{m}(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i, \boldsymbol{\gamma} * \mathbf{v}_i) - \bar{\nabla}\}\{\nabla\check{m}(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i, \boldsymbol{\gamma} * \mathbf{v}_i) - \bar{\nabla}\}^T]$ has full rank, where $\bar{\nabla} = E[\rho(f(\tilde{\boldsymbol{\mu}}_i, \mathbf{v}_i))\{\nabla\check{m}(\boldsymbol{\eta} * \tilde{\boldsymbol{\mu}}_i, \boldsymbol{\gamma} * \mathbf{v}_i)]/E\rho(f(\tilde{\boldsymbol{\mu}}_i, \mathbf{v}_i))$, and f is the density of $(\tilde{\boldsymbol{\mu}}_i, \mathbf{v}_i)$.

References

Alter, O., Brown, P. O., and Botstein, D. (2000). Singular value decomposition for genome-wide expression data processing and modeling. *Proc. Nat. Acad. Sci. USA* **97**, 10101–10106.

Cardot, H., Ferraty, F., and Sarda, P. (2003). Spline estimators for the functional linear model. *Statistica Sinica* **13**, 571–591.

Carroll, R. J., Ruppert, D., and Stefanski, L.A. andCrainiceanu, C. (2006). *Measurement Error in Nonlinear Models: A Modern Perspective.* Chapman and Hall/CRC, 2nd edn.

Chen, D., Hall, P., and Muller, H. G. (2011). Single and multiple index functional regression models with nonparametric link. *Annals of Statistics* **39**, 1720–1747.

Chiou, J.-M. and Muller, H.-G. (2014). Linear manifold modelling of multivariate functional data. *Journal of the Royal Statistical Society, Ser. B* .

Ferraty, F. and Vieu, P. (2003). Curves discrimination: a nonparametric functional approach. *Computational Statistics and Data Analysis* **44**, 161–173.

Hall, P., Muller, H.-G., and Wang, J.-L. (2006). Properties of principal component methods for functional and longitudinal data analysis. *The Annals of Statistics* **34**, 1493-1517,.

Hall, P., Poskitt, D. S., and Presnell, B. (2001). A functional data-analytic approach to signal discrimination. *Technometrics* **43**, 1–9.

Hall, P., Reimann, J., and Rice, J. (2000). Nonparametric estimation of a periodic function. *Biometrika* **87**, 545–557.

Hastie, T. and Mallows, C. (1993). Comment on "a statistical view of some chemometrics regression tools". *Technometrics* **35**, 140–143.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika* **28**, 321–377.

James, G. M. (2002). Generalized linear models with functional predictors. *Journal of the Royal Statistical Society, Series B* **64**, 411–432.

James, G. M. and Silverman, B. W. (2005). Functional adaptive model estimation. *Journal of the American Statistical Association* **100**, 565–576.

Jank, W. and Shmueli, G. (2005). Profiling price dynamics in online auctions using curve clustering. *Working Paper* .

Jiang, C. J. and Wang, J. L. (2011). Functional single index models for longitudinal data. *The Annals of Statistics* **39**, 362–388.

Li, L. (2009). Exploiting predictor domain information in sufficient dimension reduction. *Computational Statistics and Data Analysis* **53**, 2665–2672.

Li, L., Li, B., and Zhu, L.-X. (2010). Groupwise dimension reduction. *Journal of the American Statistical Association* **105**, 1188–1201.

Li, P. and Chiou, J. (2011). Identifying cluster number for subspace projected functional data clustering. *Computational Statistics and Data Analysis* **55**, 20902103.

Li, Y. and Hsing, T. (2010). Deciding the dimension of effective dimension reduction space for functional and high-dimensional data. *Annals of Statistics* **38**, 2587–3216.

Liu, B. and Muller, H.-G. (2008). Functional data analysis for sparse auction data. In W. Jank and G. Shmueli, eds., *Statistical Methods in e-Commerce Research*. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Muller, H. G. and Stadtmuller, U. (2005). Generalized functional linear models. *Annals of Statistics* **33**, 774–805.

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*. Springer, 2nd edn.

Reddy, S. K. and Dass, M. (2006). Modeling on-line art auction dynamics using functional data analysis. *Statistical Science* **21**, 179–193.

Reithinger, F., Jank, W., Tutz, G., and Shmueli, G. (2008). Smooothing sparse and unevenly sampled curves using semiparametric mixed models: An application to online auctions. *Journal of the Royal Statistical Society, Series C* **57**, 127–148.

Shmueli, G. and Jank, W. (2005). Visualizing online auctions. *Journal of Computational and Graphical Statistics* **14**, 299–319.

Silverman, B. W. (1999). *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London, 2nd edn.

Wang, S., J. W. and Shmueli, G. (2008). Explaning and forecasting online auction prices and their dynamics using functional data analysis. *Journal of Business and Economics Statistics* **26**, 144–160.

Xia, Y. (2008). A multiple-index model and dimension reduction. *Journal of the American Statistical Association* **103**, 1631–1640.