The Canadian Journal of Statistics
Vol. 50, No. 1, 2022, Pages 122–141
La revue canadienne de statistique

Recovering the underlying trajectory from sparse and irregular longitudinal data

Yunlong NIE[1], Yuping YANG[1], Liangliang WANG[1], and Jiguo CAO[1*]

[1]*Department of Statistics and Actuarial Science, Simon Fraser University, Burnaby, BC, Canada*

Key words and phrases: Empirical basis functions; functional data analysis; functional principal component analysis; PACE.

MSC 2020: Primary 62G99; secondary 62G08.

Abstract: In this article, we consider the problem of recovering the underlying trajectory when the longitudinal data are sparsely and irregularly observed and noise-contaminated. Such data are popularly analyzed with functional principal component analysis via the principal analysis by conditional estimation (PACE) method. The PACE method may sometimes be numerically unstable because it involves the inverse of the covariance matrix. We propose a sparse orthonormal approximation (SOAP) method as an alternative. It estimates the optimal empirical basis functions in the best approximation framework rather than eigen-decomposing the covariance function. The SOAP method avoids estimating the mean and covariance function, which is challenging when the assembled time points with observations for all subjects are not sufficiently dense. The method also avoids the inverse of the covariance matrix, hence the computation is more stable. It does not require the functional principal component scores to follow the Gaussian distribution. We show that the SOAP estimate for the optimal empirical basis function is asymptotically consistent. The finite-sample performance of the SOAP method is investigated in simulation studies in comparison with the PACE method. Our method is demonstrated by recovering the CD4 percentage curves from sparse and irregular data in a multi-centre AIDS cohort study. *The Canadian Journal of Statistics* 50: 122–141; 2022 © 2021 Statistical Society of Canada

Résumé: Cet article traite de l'estimation (ou récupération) d'une trajectoire sous-jacente à des données longitudinales clairsemées, irrégulières et contaminées par du bruit. L'analyse de ce type de données fait usuellement appel aux techniques de l'Analyse en Composantes Principales Fonctionnelles, plus spécifiquement l'Analyse Principale par Espérance Conditionnelle (PACE). La méthode PACE peut parfois être numériquement instable car elle implique l'inverse de la matrice de covariance. Pour parer à cette éventualité, les auteurs du présent travail proposent une méthode d'approximation orthonormale clairsemée (SOAP). Cette dernière adopte le meilleure cadre d'approximation pour estimer les fonctions de base empiriques optimales plutôt que de recourir à une décomposition en valeurs propres de la matrice de covariance. Ainsi, la méthode SOAP contourne la difficulté d'estimer la moyenne et la covariance lorsque l'ensemble des observations dans le temps n'est pas dense. En plus d'assurer une stabilité numérique en évitant l'inversion de la matrice de covariance, cette méthode n'impose pas la normalité des scores des composantes principales fonctionnelles. Les auteurs prouvent la convergence asymptotique de l'estimateur SOAP de la fonction de base empirique optimale et présentent des études de simulation pour comparer sa performance avec celle de la méthode PACE sur des échantillons finis. Ensuite, ils illustrent la mise en œuvre de l'approche proposée en utilisant des données éparses et irrégulières provenant d'une étude de cohorte multicentrique sur le SIDA pour estimer des courbes de CD4. *La revue canadienne de statistique* 50: 122–141; 2022 © 2021 Société statistique du Canada

Additional Supporting Information may be found in the online version of this article at the publisher's website.

* *Corresponding author: jiguo_cao@sfu.ca*

1. INTRODUCTION

Functional principal component analysis (FPCA) is a key dimension reduction tool in functional data analysis. FPCA explores major sources of variability in a sample of random curves by finding functional principal components (FPCs) that maximize the curve variation. Consequently, the top few FPCs explain most of the variability in the random curves. In addition, each random curve can be approximated by a linear combination of the top FPCs. Therefore, the infinite-dimensional curves are projected to a low-dimensional space defined by the top FPCs. This powerful dimensional reduction feature also contributes to the popularity of FPCA.

The theoretical properties of FPCA have been carefully studied at length. For example, Dauxois, Pousse & Romain (1982) first studied the asymptotic properties of PCA estimators for the infinite-dimensional data from a linear operator perspective. Following this point of view, Bosq (2000) and Mas (2002) utilized functional analysis to study FPCA theoretically. On the other hand, Yao, Müller & Wang (2005), Hall, Müller & Wang (2006), and Hall & Horowitz (2007) studied FPCA from the kernel perspective. The smooth version of FPCA was carefully studied by Rice & Silverman (1991), Pezzulli (1993), Silverman (1996), and Yao, Müller & Wang (2005). There are mainly three methods to achieve smoothness. The first method smooths the functional data in the first step and conducts the regular FPCA on the sample covariance function. The second method smooths the covariance function first and then eigen-decomposes the resulting smoothed covariance function to estimate the smoothed FPCs. The last method directly adds a roughness penalty to the optimization criterion for estimating FPCs. Moreover, various extensions of FPCA have been proposed to suit different goals. For instance, Lin, Wang & Cao (2016) and Nie & Cao (2020) proposed adding a penalty function on the non-zero regions of FPCs, which led to better visualization, as their estimated FPCs become non-zero only in the intervals with major variation. Sang, Wang & Cao (2017) proposed conducting FPCA from a parametric perspective to improve the interpretability of the FPCs. To enhance the predictiveness of FPCs, Nie et al. (2018) developed a supervised version of FPCA that accommodates the correlation between FPCs and a response variable of interest. Shi et al. (2021) proposed FPCA for longitudinal data with informative dropout.

FPCA has been widely and successfully applied in many applications such as functional linear regression (Cardot, Ferraty & Sarda, 1999) and classification and clustering of functional data (Müller, 2005; Müller & Stadtmüller, 2005; Ramsay & Silverman, 2005; Peng & Müller, 2008; Dong et al., 2018). All these applications assume the functional data to be densely and regularly observed. When the functional data are sparsely or irregularly observed, it is challenging to obtain a good estimate of FPCs and the corresponding FPC scores. Yao, Müller & Wang (2005) proposed the PACE method to analyze sparse functional data. The PACE method estimates the covariance function by the local polynomial regression method and then eigen-decomposes the estimated covariance function to obtain the eigenfunctions as the estimates of FPCs. The corresponding FPC score is estimated using conditional expectation, which requires that FPC scores follow a Gaussian distribution. The asymptotic properties were established in Hall, Müller & Wang (2006).

The PACE method is very successful. It is now popularly used to analyze sparse functional data. On the other hand, the PACE method also has two major assumptions, which may limit its applications. The first assumption of PACE is that the observed time points over all subjects are sufficiently dense. Otherwise, PACE cannot estimate the mean and covariance function by pooling data for all subjects together. The second assumption of PACE is that the FPC scores follow a Gaussian distribution. Otherwise, the conditional expectation formula is invalid. In addition, the PACE method involves the inverse of the estimated covariance matrix when estimating individual trajectories, which may be computationally unstable. This problem will be demonstrated in our simulation studies. Peng & Paul (2009) proposed a restricted maximum likelihood approach to estimate FPCs and apply a Newton–Raphson procedure on a Stiefel

manifold to guarantee that the resulting FPCs satisfy the orthonormality constraints. They also used conditional expectation to obtain FPC scores in order to recover individual trajectories. Therefore, their method also involves the inverse of the estimated covariance matrix and requires the FPC scores to have a Gaussian distribution.

The main objective of this article is to recover the underlying trajectory given sparse and irregular longitudinal observations. Note that this objective is different from exploring the major variation patterns of the functional data, which is the central goal for the conventional FPCA. We propose a new sparse orthonormal approximation (SOAP) method to recover the underlying trajectory. This method directly estimates the optimal empirical basis functions and the corresponding coefficients in the best approximation framework. The SOAP method has three main advantages. First, it avoids the inverse of the covariance matrix, and the computation is stable and efficient. Second, it does not require that the scores follow a Gaussian distribution. Therefore, it can be applied in non-Gaussian cases. Lastly, the method does not need to estimate the mean and covariance function, which might be challenging in the case of sparse observation times. The computing scripts for the simulation studies and the application to real data are available at https://github.com/caojiguo/SOAP.

The rest of the article is organized as follows. Section 2 introduces the best approximation framework for recovering the underlying trajectory given sparse and irregular longitudinal observations. Section 3 describes the SOAP method for estimating the optimal empirical basis functions and the corresponding coefficients. The asymptotic consistency results for the estimated functional empirical components (FECs) are provided in Section 4. Our proposed method is demonstrated in Section 5 by recovering the longitudinal CD4 percentage trajectories. In Section 6, we compare the finite-sample performance of our method with that of the PACE method using simulation studies. Section 7 provides concluding remarks.

2. FUNCTIONAL EMPIRICAL COMPONENT ANALYSIS

Consider n independent realizations, $x_1(t), \ldots, x_n(t)$, of an L^2 stochastic process $X(t) : t \in [0, T]$ at a sequence of random points on $[0, T]$ with measurement errors. That is, the observed data $y_{ij}, i = 1, \ldots, n, j = 1, \ldots, n_i$, is

$$y_{ij} = x_i\left(t_{ij}\right) + \epsilon_{ij},$$

where $\left\{\epsilon_{ij}\right\}$ are independent and identically distributed random errors with mean zero and variance σ^2. The number of measurements n_i for each curve is random and small. The observed time points t_{ij} can also be different for each curve. Using the Karhunen–Loève expansion (Fukunaga & Koontz, 1970), each $x_i(t)$ can be expressed as

$$x_i(t) = \mu(t) + \sum_{k=1}^{\infty} \tau_{ik}\phi_k(t),$$

where $\mu(t) = \mathrm{E}(X(t))$ is the mean function, and $\phi_k(t), k = 1, 2, \ldots$, are the eigenfunctions of the covariance function $C(s, t) = \mathrm{E}[(X(s) - \mu(s))(X(t) - \mu(t))], t, s \in [0, T]$. We call the $\phi_k(t)$'s and the corresponding τ_{ik}'s the FPCs and FPC scores, respectively. The above estimation procedure is called the FPCA.

A main advantage of FPCA is that $x_i(t)$ is projected onto orthogonal basis functions. This projection allows us to approximate $x_i(t)$ using the first K leading FPCs:

$$x_i(t) \approx \mu(t) + \sum_{k=1}^{K} \tau_{ik}\phi_k(t).$$

There are many other basis functions onto which $x_i(t)$ can be projected. However, the eigenfunctions of the covariance functions have been proved to be the optimal basis functions in the sense that they minimize the mean L^2 errors (see Tran, 2008). Formally speaking, for any fixed $K \in \{1, 2, \dots\}$, the first K leading empirical FPCs minimize

$$\frac{1}{n}\left(\sum_{i=1}^{n}\int_0^{\mathrm{T}}\left[x_i(t)-\mu(t)-\sum_{k=1}^{K}\langle x_i-\mu,\phi_k\rangle\phi_k(t)\right]^2 dt\right),$$

subject to $\langle \phi_k, \phi_l\rangle \equiv \int \phi_k(t)\phi_l(t)dt = \delta_{kl}$, where δ_{kl} is the Kronecker delta. We will omit the interval of integration $[0, T]$ for the rest of the article for the sake of notational simplicity. From the above criterion, we can see that the eigenfunctions $\phi_k(t), k = 1, \dots, K$, are essentially the optimal empirical basis functions for approximating the centred stochastic process $X(t) - \mu(t)$.

For the original uncentred stochastic process $X(t)$, the optimal empirical basis functions are the eigenfunctions of $K(s, t) = \mathrm{E}[X(s)X(t)]$, as shown in Theorem 1. Note that though $K(s, t)$ is not a covariance function, it is still a Mercer kernel. By Mercer's theorem, there exists an orthonormal basis $\psi_m(t)$ such that $K(s, t)$ has the following representation:

$$K(s,t)=\sum_{m=1}^{\infty}\lambda_m\psi_m(s)\psi_m(t),$$

in which the eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$ and the eigenfunctions satisfy $\langle \psi_m, \psi_{\ell}\rangle = \delta_{m\ell}$. Correspondingly, $x_i(t)$ can be represented as

$$x_i(t)=\sum_{m=1}^{\infty}\alpha_{im}\psi_m(t).$$

Now we will show that the empirical basis functions, $\psi_m(t), m = 1, \dots, M$, optimal in the sense of minimizing the approximation error (defined later in Theorem 1), are the eigenfunctions of the estimate $\widehat{K}(s, t) = \frac{1}{n}\sum_{i=1}^{n}[x_i(s)x_i(t)]$.

Theorem 1. *For any given value of M, the optimal empirical basis functions* $\psi_m(t), m = 1, \dots, M$, *which minimize*

$$\frac{1}{n}\sum_{i=1}^{n}\left(\int\left[x_i(t)-\sum_{m=1}^{M}\alpha_{im}\psi_m(t)\right]^2 dt\right), \tag{1}$$

subject to $\langle \psi_m, \psi_{\ell}\rangle = \delta_{m\ell}$, *are the first M eigenfunctions of* $\widehat{K}(s,t) = \frac{1}{n}\sum_{i=1}^{n}\left[x_i(s)x_i(t)\right]$ *and* $\alpha_{im} = \langle x_i, \psi_m\rangle$.

The detailed proof for Theorem 1 is given in the Supplementary Material. Theorem 1 not only shows that those eigenfunctions of $\widehat{K}(s, t)$ are the optimal empirical basis functions for approximating the original functional data but also provides an alternative way to estimate these optimal empirical basis functions in the best approximation framework other than eigen-decomposing the uncentred sample covariance function $\widehat{K}(s, t)$. Note that estimating the sample covariance function may become challenging when the data are sparsely observed.

Moreover, this best approximation framework also allows the coefficients of the optimal empirical basis functions to be estimated without inverting the sample covariance matrix. Furthermore, Theorem 1 shows that estimating the mean function $\mu(t)$ is not necessary if the goal

is recovering or approximating the original trajectory. In practice, when the observed data are very sparsely observed, it may be challenging to estimate the mean function $\mu(t)$. Alternatively, we can simply estimate those optimal empirical basis functions and represent each trajectory using the estimated optimal empirical basis functions.

In this article, the optimal empirical basis functions $\psi_m(t), m = 1, 2, \ldots$, are called the FECs, and α_{im} is the corresponding FEC score. Note that when the mean function of the stochastic process $X(t)$, $\mu(t) = \mathrm{E}(X(t)) = 0$, the FECs are equivalent to the FPCs.

We propose the SOAP method to estimate the first M FECs $\psi_m(t), m = 1, \ldots, M$, by minimizing the observed loss function:

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \sum_{m=1}^{M}\alpha_{im}\psi_m\left(t_{ij}\right)\right]^2, \tag{2}$$

subject to $\langle \psi_m, \psi_\ell \rangle = \delta_{m\ell}$, where $m, \ell = 1, \ldots, M$. We minimize the objective function (2) in a sequential manner. That is, we first obtain the first FEC. Then conditional on the estimated first FEC, we estimate the second FEC, and so on. When estimating each FEC, we estimate the mth component ψ_m and the corresponding FEC score $\boldsymbol{\alpha}_m = (\alpha_{1m}, \ldots, \alpha_{nm})^\top$ in an iterative fashion. We first estimate $\boldsymbol{\alpha}_m$ based on the given FEC $\psi_m(t)$ and the observations $y_{ij}, i = 1, \ldots, n, j = 1, \ldots, n_i$. Then, given the estimated $\widehat{\boldsymbol{\alpha}}_m$, we obtain the corresponding FEC $\psi_m(t)$ by minimizing (2). In each iteration, the loss function (2) is guaranteed to decrease.

3. SPARSE ORTHONORMAL APPROXIMATION METHOD

We first describe the SOAP method to estimate the first FEC in Section 3.1. Then our method is expanded to estimate the first M FECs in Section 3.2.

3.1. Estimating the First FEC

The first FEC $\psi_1(t)$ is obtained by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \alpha_{i1}\psi_1\left(t_{ij}\right)\right]^2, \tag{3}$$

subject to $\|\psi_1\|^2 = 1$. We first express $\psi_1(t)$ as a linear combination of basis functions: $\psi_1(t) = \boldsymbol{\beta}_1^\top \mathbf{b}(t)$, where $\mathbf{b}(t) = \left(b_1(t), \ldots, b_L(t)\right)^\top$ is a vector of basis functions chosen beforehand, and $\boldsymbol{\beta}_1 = \left(\beta_{11}, \ldots, \beta_{1L}\right)^\top$ is the corresponding vector of coefficients. We choose cubic B-spline basis functions in our numerical studies. The choice of basis functions will not affect the performance of our method. Users can also apply other basis functions such as the Fourier basis functions. We propose to minimize (3) in an iterative fashion. That is, for a given $\psi_1(t)$, we find the corresponding α_{i1} that minimizes (3). Then, given the value of α_{i1}, we minimize (3) with respect to $\psi_1(t)$. In every iteration step, the value of the loss function (3) decreases. The detailed algorithm is outlined as follows:

Step I Set the initial value of $\psi_1(t)$ as $\psi_1^{(0)}(t)$, which satisfies $\left|\left|\psi_1^{(0)}\right|\right|^2 = 1$;

Step II Given the current value of $\psi_1^{(\ell)}(t), \ell = 0, 1, 2, \ldots$, we can obtain the value of $\boldsymbol{\alpha}_1^{(\ell)} = (\alpha_{11}, \ldots, \alpha_{n1})^\top$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \alpha_{i1}\psi_1^{(\ell)}\left(t_{ij}\right)\right]^2.$$

 DOI: 10.1002/cjs.11677

In fact, this is simply a least-squares problem. The ith element of $\boldsymbol{\alpha}_1^{(\ell)}$ can be expressed as

$$\alpha_{i1} = \left(\boldsymbol{\psi}_{1i}^\top \boldsymbol{\psi}_{1i}\right)^{-1} \boldsymbol{\psi}_{1i}^\top \mathbf{y}_i,$$

where $\boldsymbol{\psi}_{1i} = \left(\psi_1(t_{i1}), \ldots, \psi_1(t_{in_i})\right)^\top$ is an $n_i \times 1$ vector, and $\mathbf{y}_i = \left(y_{i1}, \ldots, y_{in_i}\right)^\top$.

Step III Given the current value of $\boldsymbol{\alpha}_1^{(\ell)}$, update $\psi_1^{(\ell)}(t)$ to $\psi_1^{(\ell+1)}(t)$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \alpha_{i1}^{(\ell)}\psi_1(t_{ij})\right]^2,$$

subject to $||\psi_1||^2 = 1$.
We recast the above criterion into

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \alpha_{i1}^{(\ell)}\psi_1(t_{ij})\right]^2 &= \frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \alpha_{i1}^{(\ell)}\boldsymbol{\beta}_1^\top \mathbf{b}(t_{ij})\right]^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n_i}\left[\frac{1}{\sqrt{n_i}}y_{ij} - \boldsymbol{\beta}_1^\top \frac{1}{\sqrt{n_i}}\alpha_{i1}^{(\ell)}\mathbf{b}(t_{ij})\right]^2,\end{aligned}$$

subject to $\boldsymbol{\beta}_1^\top \mathbf{G}\boldsymbol{\beta}_1 = 1$, in which $\mathbf{G}$ is an $L \times L$ matrix with the (i,j)th element $\langle b_i, b_j\rangle$. This is a constrained least-squares problem. Fortunately, we can ignore the norm constraint and obtain the unconstrained least-squares minimizer first, and then scale it such that its norm is 1. More specifically, the solution can be written as $\boldsymbol{\beta}_1^{(\ell+1)} = \tilde{\boldsymbol{\beta}}_1^{(\ell+1)} \Big/ \sqrt{\{\tilde{\boldsymbol{\beta}}_1^{(\ell+1)}\}^\top \mathbf{G}\tilde{\boldsymbol{\beta}}_1^{(\ell+1)}}$, in which $\tilde{\boldsymbol{\beta}}_1^{(\ell+1)} = (\mathbf{a}^{(\ell)\top}\mathbf{a}^{(\ell)})^{-1}(\mathbf{a}^{(\ell)})^\top \mathbf{y}_w$, $\mathbf{y}_w = \left(\mathbf{y}_1^\top \Big/ \sqrt{n_1}, \ldots, \mathbf{y}_n^\top \Big/ \sqrt{n_n}\right)^\top$ and $\mathbf{a}^{(\ell)} = \left(\mathbf{a}_1^{(\ell)\top}, \ldots, \mathbf{a}_n^{(\ell)\top}\right)^\top$ is a $\left(\sum_{i=1}^n n_i\right) \times L$ matrix, in which $\mathbf{a}_i^{(\ell)}$ is an $n_i \times L$ matrix with (p,q) elements being $\alpha_{i1}^{(\ell)} b_q(t_{ip}) / \sqrt{n_i}$. It can be checked that the minimizer obtained from the least squares will satisfy the Karush–Kuhn–Tucker (KKT) condition, and thus it is the global minimizer of the loss function (3).

Step IV Repeat Steps II and III until the change in value falls below a given threshold.

3.2. Estimating the Second FEC Given the First Estimated FEC

Once the first estimated FEC, $\hat{\psi}_1(t)$, is obtained, the second FEC is estimated by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[\hat{r}_{ij}^{(1)} - \alpha_{i2}\psi_2(t_{ij})\right]^2,$$

subject to $\langle\hat{\psi}_1, \psi_2\rangle = 0$ and $||\psi_2||^2 = 1$. Here, $\hat{r}_{ij}^{(1)}$ denotes the estimated residuals after estimating the first FEC, which is expressed as

$$\hat{r}_{ij}^{(1)} = y_{ij} - \hat{\alpha}_{i1}\hat{\psi}_1(t_{ij}).$$

We can apply a similar iterative optimization procedure as described in the previous subsection to obtain $\hat{\psi}_2(t)$:

Step I: Set an initial value of $\psi_2^{(0)}(t)$, satisfying $\langle \hat{\psi}_1, \psi_2 \rangle = 0$ and $||\psi_2||_2^2 = 1$.

Step II: Given the current value of $\psi_2^{(\ell)}(t)$, we can obtain the current value of $\boldsymbol{\alpha}_2^{(\ell)} = (\alpha_{12}, \ldots, \alpha_{n2})^\top$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[\hat{r}_{ij}^{(1)} - \alpha_{i2}\psi_2^{(\ell)}\left(t_{ij}\right)\right]^2.$$

The ith element of $\boldsymbol{\alpha}_2^{(\ell)}$ can be expressed as

$$\alpha_{i2}^{(\ell)} = \left(\boldsymbol{\psi}_{2i}^{(\ell)\top}\boldsymbol{\psi}_{2i}^{(\ell)}\right)^{-1}\boldsymbol{\psi}_{2i}^{(\ell)\top}\hat{\mathbf{r}}_i^{(1)},$$

where $\boldsymbol{\psi}_{2i}^{(\ell)} = \left(\psi_2^{(\ell)}\left(t_{i1}\right), \ldots, \psi_2^{(\ell)}\left(t_{in_i}\right)\right)^\top$ and $\hat{\mathbf{r}}_i^{(1)} = \left(\hat{r}_{i1}^{(1)}, \ldots, \hat{r}_{in_i}^{(1)}\right)^\top$.

Step III: Given the value of $\boldsymbol{\alpha}_2^{(\ell)}$, update the value of $\psi_2^{(\ell+1)}(t)$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}[\hat{r}_{ij} - \alpha_{i2}^{(\ell)}\psi_2(t_{ij})]^2,$$

subject to $\langle \psi_2^{(\ell+1)}, \hat{\psi}_1 \rangle = 0$ and $\|\psi_2^{(\ell+1)}\|^2 = 1$. Because the norm of $\psi_2^{(\ell+1)}(t)$ will not affect the KKT conditions, we can first ignore the norm constraint, and the minimization becomes a problem of least squares with equality constraints. This problem can also be solved efficiently using the Least Squares with Equalities and Inequalities (LSEI) algorithm proposed by Lawson & Hanson (1974). Then, we normalize the resulting solution such that the norm of $\psi_2^{(\ell+1)}(t)$ is 1.

Step IV: Repeat Steps II and III until convergence is reached.

3.3. Estimating the Mth FEC Given the First $M-1$ Estimated FECs

Given the first $M-1$ estimated FECs, $\widehat{\psi}_m(t), m = 1, \ldots, M-1$ and the residuals $\mathbf{r}_i^{(M)}, i = 1, \ldots, n$, we can obtain the estimate for $\psi_M(t)$ using a similar strategy to that described in Section 3.2, by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[\hat{r}_{ij}^{(M-1)} - \alpha_{iM}\psi_M\left(t_{ij}\right)\right]^2,$$

subject to $\langle \hat{\psi}_m, \psi_M \rangle = 0, m = 1, \ldots, M-1$ and $\|\psi_M\|^2 = 1$. An algorithm similar to that described in Section 3.2 can be applied here to obtain the estimated $\hat{\psi}_M$.

3.4. Smoothness Regulation

In order to control the smoothness of the estimated FECs $\psi_m(t), m = 1, \ldots, M$, we can add a roughness penalty in (2). That is, for any fixed M, we estimate $\psi_1(t), \ldots, \psi_M(t)$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij} - \sum_{m=1}^{M}\alpha_{im}\psi_m\left(t_j\right)\right]^2 + \sum_{m=1}^{M}\gamma_m\int\left[\frac{d^2\psi_m(t)}{dt^2}\right]^2 dt, \tag{4}$$

subject to $\langle \psi_m, \psi_\ell \rangle = \delta_{m\ell}$, where $m, \ell = 1, \ldots, M$. The algorithm introduced in Sections 3.1–3.3 can be modified accordingly. For instance, we can estimate the first FEC by modifying Step III in Section 3.1 as follows:

Step III (b) Given the current value of $\boldsymbol{\alpha}_1^{(\ell)}$, update the estimate of $\psi_1^{(\ell)}(t)$ to $\psi_1^{(\ell+1)}(t)$ by minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[y_{ij}-\alpha_{i1}^{(\ell)}\psi_1\left(t_{ij}\right)\right]^2+\gamma_1\int\left[\frac{d^2\psi_1(t)}{dt^2}\right]^2dt,$$

subject to $\|\psi_1\|^2=1$.

The above minimization is essentially a quadratically constrained quadratic programming (QCQP) problem. We use the R package `Rsolnp` (Ghalanos & Theussl, 2015) based on the SOLNP algorithm proposed by Ye (1987) to solve it numerically. We will demonstrate the performance of this method in our simulation studies.

When estimating each FEC, there is only one tuning parameter involved, i.e., the smoothing parameter γ_m. The value of γ_m controls the amount of smoothness imposed on the mth FEC. We propose to select the tuning parameter based on the leave-one-curve-out cross-validation strategy. To be more specific, we treat one curve's observations as the test dataset and the data for all other curves as the training dataset. For instance, when we estimate the first FEC $\psi_1(t)$, we can first obtain the estimate for the first FEC, $\widehat{\psi}_1^{(-i)}(t)$, using all the training data for a given value of γ_1 but omitting the ith curve, and then using this estimate to predict it. Then, the score for the test curve can be calculated by minimizing

$$\sum_{j=1}^{n_i}\left(y_{ij}-\alpha_{i1}\widehat{\psi}_1^{(-i)}\left(t_{ij}\right)\right)^2.$$

Then the prediction for y_{ij} is $\hat{y}_{ij}^{(-i)}=\widehat{\alpha}_{i1}^{(-i)}\widehat{\psi}_1^{(-i)}\left(t_{ij}\right)$, and the mean-squared prediction error for the ith curve is

$$\frac{1}{n_i}\sum_{j}\left(\hat{y}_{ij}^{(-i)}-y_{ij}\right)^2.$$

We choose γ_1 to minimize the cross-validation error

$$\mathrm{CV}(\gamma_1)=\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left(\hat{y}_{ij}^{(-i)}-y_{ij}\right)^2.$$

For the following FECs, we propose selecting the smoothing parameter after treating the previous estimated FECs as fixed.

3.5. Selecting the Number of FECs

We use the AIC criterion proposed by Li, Wang & Carroll (2013) to select the number of FECs:

$$\mathrm{AIC}(M)=N\log\left(\sigma_M^2\right)+N+2nM, \tag{5}$$

in which M denotes the number of FECs, n denotes the number of individual curves, and $N=\sum_{i=1}^{n}n_i$ is the total number of observations. We can estimate the noise variance, σ_M^2, by using the average square of the residuals. That is

$$\hat{\sigma}_M^2=\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\left(\mathbf{y}_i-\widehat{\mathbf{y}}_{i,M}\right)^{\top}\left(\mathbf{y}_i-\widehat{\mathbf{y}}_{i,M}\right),$$

where $\hat{\mathbf{y}}_{i,M} = \left(\hat{y}_{i1}, \ldots, \hat{y}_{in_i}\right)^\top$ represents the fitted ith individual's observations when the number of FECs is selected to be M.

3.6. Recovering the Individual Trajectory

Given the first M estimated eigenfunctions $\hat{\psi}_1(t), \ldots, \hat{\psi}_M(t)$, and the observations from a new individual denoted by $\mathbf{y}^* = \left(y_1^*, \ldots, y_{n^*}^*\right)^\top$, at time points $\left(t_1^*, \ldots, t_{n^*}^*\right)$, we describe the method to estimate the score vector $\boldsymbol{\alpha}^* = \left(\alpha_1^*, \ldots, \alpha_M^*\right)^\top$ in this section. Following the formulation before, we have

$$y_j^* = \sum_{m=1}^{M} \alpha_m^* \psi_m\left(t_j^*\right) + \epsilon_j^*,$$

in which $\epsilon_j^* \sim N(0, \sigma^2)$. We also consider the mth score α_m^* as a realization of a random variable with density $f_m(\boldsymbol{\theta}_m)$. We further assume that the scores are a priori mutually independent of each other. The posterior distribution of $\boldsymbol{\alpha}^*$ can be expressed as

$$\Pr\left(\boldsymbol{\alpha}^* | \mathbf{y}^*, \psi_1, \ldots, \psi_m, \sigma\right) \propto \Pr\left(\mathbf{y}^* | \boldsymbol{\alpha}^*, \psi_1, \ldots, \psi_m, \sigma\right) f(\boldsymbol{\alpha}^*),$$

in which

$$f(\boldsymbol{\alpha}^*) = \prod_{m=1}^{M} f_m\left(\alpha_m^*\right)$$

$$\Pr\left(\mathbf{y}^* | \boldsymbol{\alpha}^*, \psi_1, \ldots, \psi_m, \sigma\right) = \prod_{j=1}^{n^*} g\left(y_j^*; (\boldsymbol{\alpha}^*)^\top \boldsymbol{\psi}\left(t_j\right), \sigma\right),$$

and $g\left(y_j^*; (\boldsymbol{\alpha}^*)^\top \boldsymbol{\psi}\left(t_j\right), \sigma\right)$ denotes the Gaussian density function evaluated at y_j^* with mean $(\boldsymbol{\alpha}^*)^\top \boldsymbol{\psi}\left(t_j\right)$ and standard derivation σ.

We can substitute ψ_m and σ with the corresponding estimated $\hat{\psi}_m$ and $\hat{\sigma}_M$ (Eq. (5)) in all the equations above. Then we apply the Metropolis–Hastings algorithm to draw Markov chain Monte Carlo (MCMC) samples from the corresponding posterior distribution $\Pr\left(\boldsymbol{\alpha}'^* | \mathbf{y}^*, \hat{\psi}_1, \ldots, \hat{\psi}_m, \hat{\sigma}_M\right)$. The distribution $f_m\left(\boldsymbol{\theta}_m\right)$ can be estimated using the estimated scores $\hat{\alpha}_{im}, i = 1, \ldots, n$ described in Sections 3.1–3.3. If the parametric form of $f_m(\boldsymbol{\theta}_m)$ is known, we can always estimate the associated unknown parameter $\boldsymbol{\theta}_m$ by treating $\hat{\alpha}_{im}, i = 1, \ldots, n$ as the realizations from this distribution; otherwise, we can use the empirical density of $\hat{\alpha}_{im}$ as the estimated $\hat{f}_m(\boldsymbol{\theta})$.

4. THEORETICAL RESULTS

Theorem 3 shows that our first estimated FEC will asymptotically converge to the true FEC as the number of subjects increases. Similar results are shown in Theorem 4 for the other estimated FECs.

Let the Mercer expansion with kernel $k(s,t) = \mathrm{E}X(s)X(t)$ of the stochastic process $X(t)$ be $\left\{\lambda_k, \psi_k^0 : k = 1, 2, \ldots\right\}$. That is, there are random variables a_k and square-integrable functions ψ_k^0 on $[0,1]$ such that $X(t) = \sum a_k \psi_k^0(t)$. We observe that $\mathrm{E}a_k a_l = \lambda_k \delta_{kl}$ with the λ_i's positive and strictly decreasing.

Assumption A0. $\sum_k \mathrm{E}a_k^4 < \infty$, and for some M, $\psi_k^0(t) < M$ for all $t \in [0,1]$ and each $k = 1, 2, \ldots$.

Assumption A1. The parameter set $\Theta = \{((\alpha_i), (\beta_k)) \in C_{00} \oplus B_{\ell_2}\}$ is manageable (Pollard, 1990), where $C_{00} = \{(c_i) : |c_i| < M$ for some constant M, and $c_i = 0$ for all but finitely many $i\}$ and $B_{\ell_2} = \{(b_i) : \sum |b_i|^2 \leq 1\}$.

Theorem 2 (Pollard, 1990, Pollard's ULLN). *Let $\{f_i(\omega, t) : t \in T, \omega \in \Omega\}$ be a sequence of independent processes that are manageable for their envelopes $\{F_i(\omega) = \sup_t |f_i(\omega, t)|\}$. If $\sum_i \left(EF_i/i^2\right) < \infty$, then*

$$P\left\{\omega : n^{-1} \sup_t |S_n(w, t) - ES_n(w, t)| \to 0 \text{ as } n \to \infty\right\} = 1.$$

Lemma 1. Let $A = [a_{ij}]$ be an $n \times p$ matrix, $r \leq \min(n, p)$, and define $\tilde{A} = \sum_{i=1}^{r} \boldsymbol{\alpha}_i \otimes \boldsymbol{\beta}_i$, where $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ are the r left and r right singular eigenvectors of A, i.e., the eigenvectors of $AA^\top$ and $A^\top A$, respectively. Then the Frobenius norm of $A - B$, where B is an $n \times p$ matrix of rank r, is minimized when $B = \tilde{A}$.

Remark 1. If A has orthogonal columns, $A^\top A$ has unit basis vectors as eigenvectors; i.e., the $\boldsymbol{\beta}_i$'s are $\mathbf{e}_i \in \mathbb{R}^p$, where $\mathbf{e}_i$ has only the ith entry non-zero, with value 1.

Given sparse observations of functional data $y_{ij} = x_i\left(t_{ij}\right) + \epsilon_{ij}$, where the observation times $t_{ij}, j = 1, \ldots, n_i$, for subject i are uniformly drawn from [0, 1], recall the objective function in (3)

$$L_n(\boldsymbol{\alpha}, \psi) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} \left[y_{ij} - \alpha_i \psi\left(t_{ij}\right)\right]^2,$$

where $\boldsymbol{\alpha} = \left(\alpha_i\right) \in \mathbb{R}^n$ and $\psi(t)$ is a function on [0, 1] with constraint $\int_0^1 \psi^2(t) dt = 1$.

Theorem 3. *Under Assumptions A0 and A1, if $\left(\hat{\psi}(t), \left[\hat{\alpha}_i, i = 1, \ldots, n\right]\right)$ jointly minimize L_n, then as $n \to \infty$, $\|\|\hat{\psi} - \psi_1^0\|\| \to 0$ in probability, and $n^{-1} \sum_{i=1}^{n} \hat{\alpha}_i \to < \psi_1^0, E(X) >$.*

Theorem 4. *The estimators $\left\{\hat{\psi}_k : k = 2, 3, \ldots\right\}$ as obtained in Section 3.2 are consistent in $L_2(0, 1)$.*

5. APPLICATION: LONGITUDINAL CD4 PERCENTAGES

To demonstrate our proposed method, we analyzed the longitudinal CD4-count dataset. The dataset considered here is from the multi-centre AIDS cohort study, which includes repeated measurements of CD4 percentages for 283 homosexual men who became HIV positive between 1984 and 1991. The CD4 percentage, defined as 100 times the count of CD4 cells divided by the total number of lymphocytes, is a commonly used marker to describe the health status of HIV-infected persons. All subjects were scheduled to be measured at semi-annual visits. The trajectories of 10 randomly selected subjects are shown in Figure 1. It shows that the data are sparse with unequal numbers of repeated measurements and different visit times for individual subjects, because many of them missed scheduled visits and the HIV infections could occur randomly during the study. For all 283 subjects, the number of observations per subject ranged between 1 and 14, with a median of 6 measurements.

The objective of our analysis is to recover individual longitudinal trajectories from the sparse and irregular observations. The smoothing parameters were selected from $\{0, 10^2, 10^4, 10^8\}$ using leave-one-curve-out cross-validation, and the selected smoothing parameters for the first five estimated eigenfunctions were $10^4, 10^2, 10^4, 10^2$, and 10^4, respectively. Table 1 displays the values of AIC defined in (5) varying with the number of FECs. It shows that AIC is minimized when the number of FECs is 3.

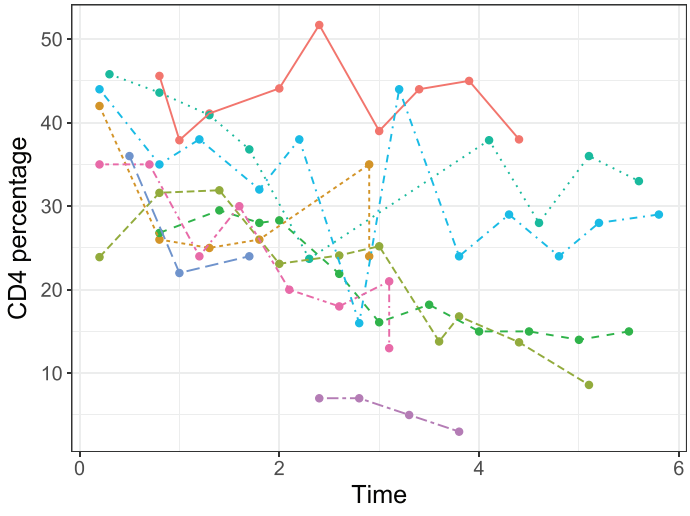

FIGURE 1: Longitudinal CD4 percentage for 10 randomly selected subjects. Each curve represents the measurements for one single subject.

TABLE 1: Values of AIC for various numbers of FECs.

# FECs	1	2	3	4	5	6
AIC	8493.44	7632.86	7626.01	7720.19	7913.83	8059.46

Figure 2 shows the three estimated FECs and the estimated mean function. The first estimated FEC, $\hat{\psi}_1(t)$, is decreasing and positive over the whole time interval. The first FEC score can be interpreted as the weighted average of the longitudinal trajectory across time. The second estimated FEC, $\hat{\psi}_2(t)$, changes its sign at time 3. The second FEC score can be interpreted as the change of the longitudinal trajectory between $[0, 2.78]$ and $[2.78, 6]$. Similarly, the third estimated FEC, $\hat{\psi}_3(t)$, is positive in $(1.6, 4.3)$ and negative elsewhere. So the third FEC score represents the change of the longitudinal trajectory between $[1.6, 4.3]$ and the other periods. The mean function was obtained by taking the average of all the individual predicted trajectories. The mean function shows an overall decreasing trend across individuals.

Figure 3 shows the predicted individual trajectories along with the corresponding pointwise confidence intervals for four different individuals with various numbers of observations. It shows that all the estimated CD4 trajectories fit the observations well. An estimated individual trajectory generally displays the overall decreasing trend when the number of observations is small, as is the case for individual 47. On the other hand, when an individual has sufficiently many observations, such as individual 3 or 90 shown in Figure 3, the estimated individual trajectory is able to capture the individual trend. Both these subjects' trajectories gradually increase between 0 and 2 years and then decrease afterwards. The pointwise confidence intervals, marked by the dashed line in Figure 3, were constructed using the parametric bootstrap method. More specifically, we first estimated the variance of the residuals, denoted by $\hat{\sigma}^2$, by pooling all the residuals from each subject at each observed time point together; then we generated bootstrap observations $\tilde{y}_{ij}$. The pointwise interval exhibits a widening effect in those regions where no individual observation is available.

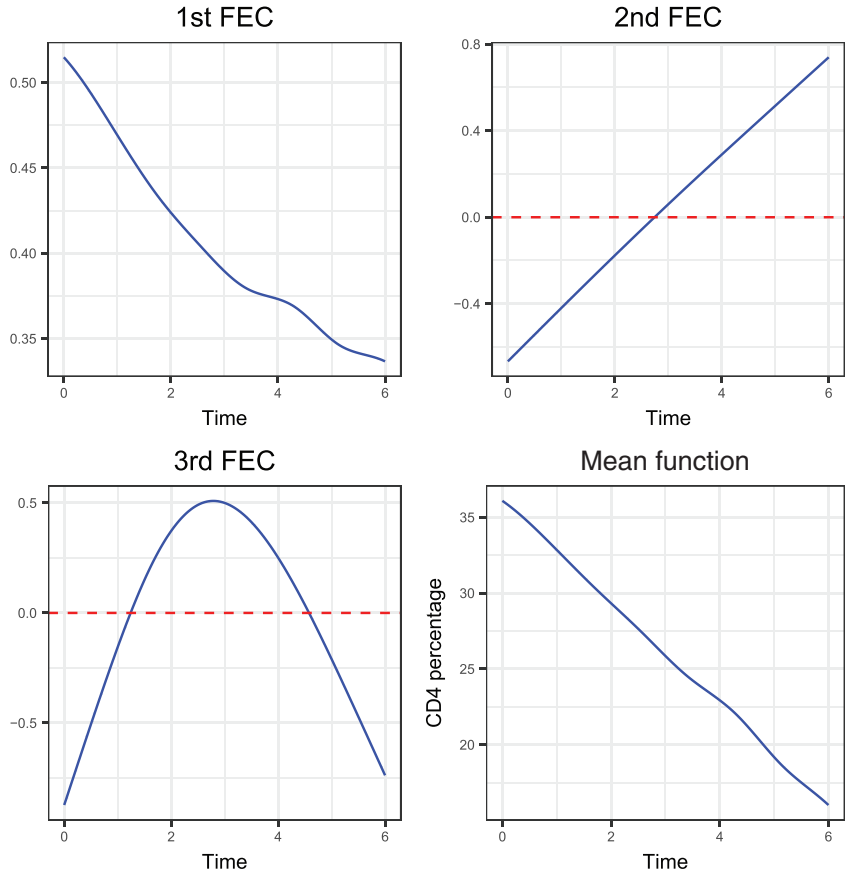

FIGURE 2: Three estimated FECs along with the estimated mean function for the CD4 data.

To assess the prediction power of the SOAP method, we first selected those 224 subjects who had at least two observations, and then treated the last observation of each subject as unknown. We applied the SOAP method to estimate the underlying eigenfunctions and scores using all the remaining observations from each subject. In the end, we obtained the predicted value at the last observation's time point with the observed value at that time point for each subject. The mean squared error (MSE) using the SOAP method was 54.37. In comparison, the MSE using the PACE method was 60.54.

6. SIMULATION STUDIES

6.1. Simulation I

To evaluate the performance of our proposed method, we conducted one simulation study in comparison with the PACE method. In order to make our proposed method and the PACE method comparable, we simulated the curves $X_i(t)$ such that $\mathrm{E}\big(X_i(t)\big) = 0$. Then, in this simulation setting, the FPCs in PACE are equivalent to our proposed FECs. Therefore, for the rest of this section, we refer to both of them as eigenfunctions.

The underlying true trajectories were simulated as $X_i(t) = \alpha_{i1}\psi_1(t) + \alpha_{i2}\psi_2(t), i = 1, \ldots, n$, where the true eigenfunctions $\psi_1(t)$ and $\psi_2(t)$ are shown in Figure 4, satisfying $\big\langle \psi_i, \psi_j \big\rangle = \delta_{ij}, i, j = 1, 2$. The corresponding scores α_{i1} and α_{i2} were generated in both Gaussian and

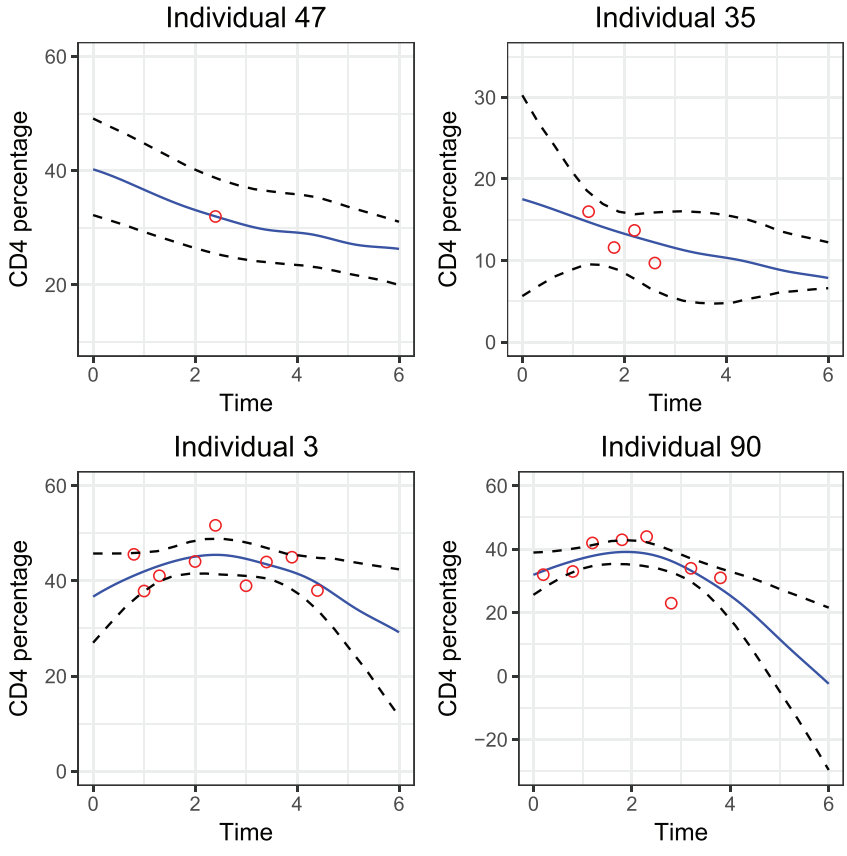

FIGURE 3: Estimated individual trajectories using the SOAP method (solid line) and the corresponding observations (circles) for four individuals (47, 35, 3, and 90) with various numbers of observations. The dashed line represents the 95% pointwise confidence interval via bootstrapping.

non-Gaussian distributions. For the Gaussian scenario, the scores are generated from two independent Gaussian distributions. That is, $\alpha_{i1} \overset{i.i.d.}{\sim} N(0, 30)$ and $\alpha_{i2} \overset{i.i.d.}{\sim} N(0, 10)$. For the non-Gaussian scenario, the scores were first generated from two independent gamma distributions and then were centred by subtracting the sample mean. That is, $\alpha_{i1} = \alpha'_{i1} - \bar{\alpha}'_{i1}$, where $\alpha'_{i1} \overset{i.i.d}{\sim}$ Gamma$(1, 0.03)$ and $\alpha_{i2} = \alpha'_{i2} - \bar{\alpha}'_{i2}$, where $\alpha'_{i2} \overset{i.i.d}{\sim}$ Gamma$(1, 0.1)$. We chose the parameters of these two gamma distributions such that the standard deviations were roughly the same as in the Gaussian scenario. The corresponding observed data for each trajectory were generated as $y_{ij} = X_i(t_{ij}) + \epsilon_{ij}$, in which $\epsilon_{ij} \sim N(0, 0.05^2)$. To achieve the sparseness, the number of time points, n_i, for each trajectory was chosen randomly from a discrete uniform distribution on $\{2, 3, 4, 5\}$. Two types of designs were considered: the regular-grid design and the uniformly distributed design. For the regular-grid design, each curve $X_i(t)$ was sampled from an equally spaced grid $\{c_0, \ldots, c_{50}\}$ on $[0, 365]$. For the uniformly distributed design, each curve $X_i(t)$ was sampled uniformly from $[0, 365]$.

To evaluate the performance of the SOAP method, we generated n sample curves in each simulation replication. We first used our proposed method to estimate the eigenfunctions using the training dataset, and then estimated the trajectories of the sample curves. The PACE method

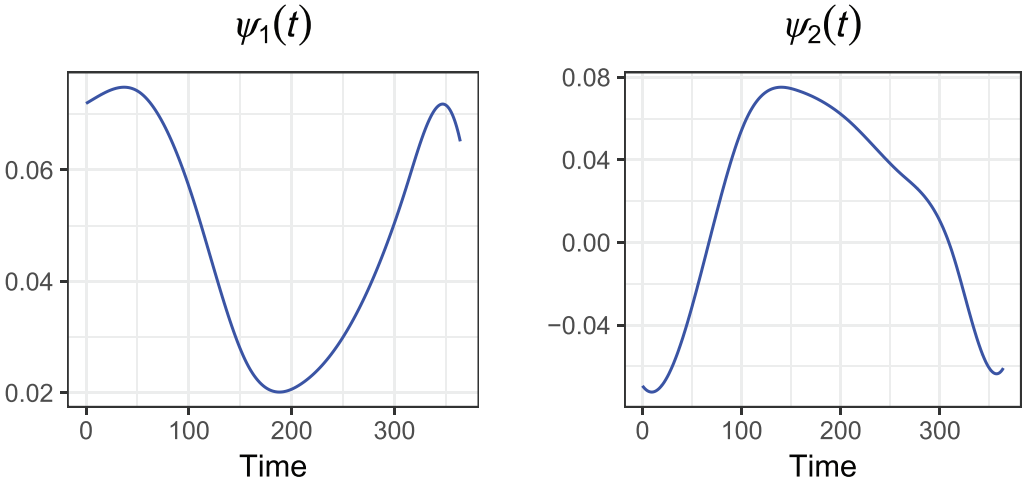

FIGURE 4: True eigenfunctions used to generate the true underlying individual trajectories. We obtain these two FECs by conducting conventional FPCA on the Canadian temperature data (Ramsay & Silverman, 2002).

was also applied to estimate the eigenfunctions from the training data and estimate the trajectories. These two methods were compared by defining the integrated mean prediction error (IMPE) as

$$\text{IMPE} = \frac{1}{n}\sum_{i=1}^{n}\int \left[\widehat{x}_i(t) - x_i(t)\right]^2 dt, \tag{6}$$

in which $x_i(t)$ represents the true ith curve, and $\widehat{x}_i(t)$ is the corresponding estimated trajectory. We repeated the above procedure for 100 Monte Carlo runs.

The results are shown in Table 2. To formally assess the performance of these two methods, a paired t-test was conducted to compare the average IMPEs under each setting. The observed mean difference and the P-value are provided in the last two columns in Table 2. First of all, we notice that the PACE method outperforms the SOAP method only when the underlying FPC scores are Gaussian-distributed and the number of curves is 300. However, the average difference between IMPEs is only 16 for both uniformly distributed and regular-grid designs. With fewer curves or non-Gaussian scores, the SOAP method yields on average smaller IMPEs compared to the PACE method. For instance, when the number of curves is 30 and the FPC scores are non-Gaussian-distributed, the reduction in IMPE from using the SOAP method instead of the PACE method is 54% for regular-grid designs and 66% for uniformly distributed designs. In addition, we notice that the performance of the PACE significantly dropped from Gaussian to non-Gaussian cases, whereas the SOAP method performs more robustly. For example, with a regular-grid design and $n = 300$, the average IMPE of the non-Gaussian cases using the PACE method is 197, which is about 1.68 times as large as the average IMPE, 117, of the Gaussian case. By contrast, under the same setting, the average IMPEs using the SOAP method are 138 and 133, respectively. Furthermore, both methods' performance improves when the number of curves increases and when the design switches from the uniformly distributed to the regular-grid design.

We also compared the performance of these two methods for irregular but more densely simulated data, where the number of observations for each curve was randomly chosen from $\{10, 11, 12, 13, 14, 15\}$. The results are shown in Table 3. First, we can see that both SOAP and PACE's average IMPEs are significantly reduced on switching from the sparse case to the dense

TABLE 2: Summary results for recovering the individual trajectory using the SOAP and PACE methods for 100 Monte Carlo runs for the sparse case.

Distribution	n	Design	SOAP	PACE	MeanDiff	P-value
Gaussian	30	Regular	187 (125)	334 (128)	−146 (10)	1.4e−33
		Uniform	194 (114)	352 (156)	−158 (13)	1.7e−24
	300	Regular	133 (34)	117 (24)	16 (2)	1.6e−13
		Uniform	149 (44)	133 (28)	16 (3)	3.8e−06
Non-Gaussian	30	Regular	193 (157)	427 (302)	−234 (23)	2.1e−20
		Uniform	200 (152)	592 (801)	−392 (47)	5.3e−15
	300	Regular	138 (58)	197 (57)	−59 (5)	2.7e−29
		Uniform	147 (52)	223 (70)	−76 (5)	5.0e−34

Note: Shown are the average and the corresponding standard deviation in parentheses for IMPE (6) under different simulation settings. A paired t-test was conducted to compare the IMPEs between the SOAP and PACE methods. "MeanDiff" represents average difference of IMSEs between the SOAP and PACE methods with the corresponding standard deviation in parentheses.

TABLE 3: Summary results for recovering the individual trajectory using the SOAP and PACE methods for 100 Monte Carlo runs for the dense case.

Distribution	n	Design	SOAP	PACE	MeanDiff	P-value
Gaussian	30	Regular	9.49 (42.5)	82.44 (39.86)	−73 (1.20)	4.1e−81
		Uniform	11.23 (49.4)	80.62 (35.9)	−69 (1.97))	3.6e−58
	300	Regular	0.46 (0.33)	6.63 (2.2)	−6 (0.04)	2.9e−114
		Uniform	1.94 (15.02)	10.76 (3.67)	−9 (0.93)	2.4e−16
Non-Gaussian	30	Regular	9.33 (31.83)	129.92 (137.13)	−121 (3.01)	8.4e−64
		Uniform	12.68 (61.3)	127.37 (115.9)	−115 (0.26)	2.9e−166
	300	Regular	0.53 (0.9)	14.18 (7.44)	−14 (−0.22)	7.1e−82
		Uniform	0.96 (1.87)	23.98 (13.33)	−23 (−0.55)	4.2e−65

Note: Shown are the average and corresponding standard deviation in parentheses for IMPE (6) under different simulation settings. A paired t-test was conducted to compare the IMPEs between the SOAP and PACE methods. "MeanDiff" represents average difference of IMSEs between the SOAP and PACE methods with the corresponding standard deviation in parentheses.

case. For instance, the average IMPEs for the SOAP method drop from 187 to 9.49 under the regular-grid design with Gaussian scores and 30 simulated curves. Second, compared to the PACE method, the SOAP method yields lower IMPEs on average across various settings for the dense cases. For example, the largest difference of the average IMPEs between the PACE method and the SOAP method is 121 under the regular-grid design with non-Gaussian scores and 30 simulated curves. Even for PACE's most ideal setting, that is, Gaussian-distributed scores with 300 simulated curves under the regular-grid design, the SOAP method still outperforms the PACE method and the average IMPE is reduced by 90% from 6.63 to 0.46. Third, we again notice

that the PACE method performs worse on switching from the Gaussian case to the corresponding non-Gaussian case, whereas the SOAP method is not sensitive to the scores' distribution.

Besides recovering the individual trend, we also compared the estimated eigenfunctions with the true eigenfunctions using the following integrated IMSE:

$$\text{IMSE}(\hat{\psi}_i) = \int \left[\psi_i(t) - \hat{\psi}_i(t)\right]^2 dt, \quad i = 1, 2. \tag{7}$$

The results are summarized in Table 4. First, the SOAP method tends to estimate the eigenfunctions well across all simulation settings. More specifically, the average $\text{IMSE}(\hat{\psi}_1)$ using the SOAP method ranges between 2×10^{-3} and 33×10^{-3} with various settings and the average $\text{IMSE}(\hat{\psi}_2)$ ranges between 104×10^{-3} and 218×10^{-3}. Second, compared to the PACE method, the IMSEs yielded by the SOAP method are much smaller. For instance, the average $\text{IMSE}(\hat{\psi}_1)$ of the PACE method is 84×10^{-3} for the Gaussian case under regular-grid design with $n = 30$, which is almost three times as large as the average $\text{IMSE}(\hat{\psi}_1)$ using the SOAP method. We also conducted a paired t-test to compare IMSEs for both ψ_1 and ψ_2, and the results suggest that the SOAP method produces more accurate estimates for both ψ_1 and ψ_2. For instance, for the Gaussian cases under a regular-grid design with 30 simulated curves, the $\text{IMSE}(\hat{\psi}_1)$ of the SOAP method is 58×10^{-3} smaller than that of the PACE method. Last but not least, we noticed that the average IMSE for the non-Gaussian scenarios is smaller than those of the corresponding Gaussian scenarios. For example, the average $\text{IMSE}(\hat{\psi}_1)$ of the PACE method under the uniformly distributed design with 300 simulated curves was 70% smaller in the non-Gaussian scenario in comparison with the Gaussian scenario. This difference is due to the effect that the non-Gaussian distribution is more likely to generate scores with large absolute values, making the signal within each eigenfunction easier to estimate. Similar conclusions can be reached for the dense case as shown in Table 5. Both methods estimate the eigenfunctions significantly better in the dense scenarios than the sparse scenarios. Overall, the SOAP method outperforms the PACE method for the dense scenarios as well.

TABLE 4: Summary results for estimating the underlying eigenfunctions using the SOAP and PACE methods for 100 Monte Carlo runs for the sparse case.

			$\text{IMSE}(\hat{\psi}_1) \times 10^{-3}$			$\text{IMSE}(\hat{\psi}_2) \times 10^{-3}$		
Distribution	n	Design	PACE	SOAP	MeanDiff	PACE	SOAP	MeanDiff
Gaussian	30	Regular	84 (57)	26 (26)	−58 (4)	705 (598)	197 (213)	−509 (45)
		Uniform	91 (70)	29 (26)	−62 (6)	738 (838)	210 (220)	−528 (78)
	300	Regular	20 (13)	2 (1)	−17 (1)	557 (453)	104 (201)	−453 (29)
		Uniform	22 (15)	3 (2)	−19 (1)	676 (489)	118 (221)	−559 (41)
Non-Gaussian	30	Regular	144 (154)	30 (45)	−114 (12)	798 (581)	212 (216)	−586 (54)
		Uniform	140 (117)	33 (45)	−107 (8)	909 (633)	218 (209)	−690 (46)
	300	Regular	61 (59)	2 (2)	−59 (4)	930 (542)	114 (200)	−815 (40)
		Uniform	73 (68)	3 (2)	−71(5)	1064 (551)	125 (229)	−938 (44)

Note: Shown are the average and the corresponding standard deviation in parentheses for IMSEs (7) under different simulation settings. "MeanDiff" represents average difference between the SOAP and PACE methods with the corresponding standard deviation in parentheses.

TABLE 5: Summary results for estimating the underlying eigenfunctions using the SOAP and PACE methods for 100 Monte Carlo runs for the dense case.

			$\text{IMSE}(\hat{\psi}_1) \times 10^{-4}$			$\text{IMSE}(\hat{\psi}_2) \times 10^{-4}$		
Distribution	n	Design	PACE	SOAP	MeanDiff	PACE	SOAP	MeanDiff
Gaussian	30	Regular	37 (21)	7 (7)	−30 (1)	321 (204)	5 (7)	−316 (13)
		Uniform	50 (30)	8 (7)	−42 (2)	581 (862)	5 (6)	−576 (55)
	300	Regular	461 (305)	85 (75)	−376 (15)	4936 (4186)	88 (250)	−4848 (216)
		Uniform	479 (324)	85 (85)	−394 (18)	4972 (4649)	73 (84)	−4899 (271)
Non-Gaussian	30	Regular	112 (89)	6 (6)	−107 (6)	1777 (2745)	4 (6)	−1773 (172)
		Uniform	157 (117)	8 (7)	−149 (8)	3178 (3965)	5 (7)	−3173 (273)
	300	Regular	924 (726)	119 (252)	−805 (31)	7415 (5599)	156 (566)	−7259 (253)
		Uniform	923 (717)	102 (249)	−822 (20)	7445 (5534)	129 (517)	−7316 (159)

Note: Shown are the average and the corresponding standard deviation in parentheses for IMSEs (7) under different simulation settings. "MeanDiff" represents average difference of IMSEs between the SOAP and PACE methods with the corresponding standard deviation in parentheses.

6.2. Simulation II

In the second simulation study, the true underlying curves were simulated with five eigenfunctions: $X_i(t) = \sum_{m=1}^{5} \alpha_{im}\psi_m(t), i = 1, 2, \ldots, n$. Similar to the eigenfunctions in the first simulation, these five true eigenfunctions were obtained by conducting conventional FPCA on the Canadian temperature data. For each individual, the scores $\alpha_{im}, m = 1, \ldots, 5$, were simulated independently with either Gaussian and non-Gaussian distributions. More specifically, for the Gaussian scenario, $\boldsymbol{\alpha}_i = \left(\alpha_{i1}, \ldots, \alpha_{i5}\right)^\top \sim MVN((1, 1, 1, 1, 1)^\top, \Sigma)$, in which $\Sigma = \text{diag}(800, 160, 32, 6.4, 1.6)$. For the non-Gaussian scenario, five Gamma distributions with shape parameter 1 and rate parameters $0.03, 0.1, 0.2, 0.35$, and 1.1 were used to generate the five scores for individual i. The rate parameters were chosen such that the standard deviations were roughly the same as in the Gaussian scenario. Note that the mean function is non-zero in both the Gaussian and non-Gaussian cases. To achieve sparseness, the number of time points, n_i, for each trajectory was chosen randomly from a discrete uniform distribution on $\{2, 3, 4, 5\}$. We consider only the uniformly distributed design here. That is, each curve $X_i(t)$ was sampled uniformly in $[0, 365]$.

In each simulation replication, we first generated n sample curves and then applied the SOAP method and the PACE method to estimate each sampled curve's trajectory. We repeated this procedure for 100 Monte Carlo runs. Table 6 summarizes the IMPEs defined in Equation (6) for recovering the individual trajectory using the SOAP method and the PACE method for 100 Monte Carlo runs. First of all, we can see that the SOAP method yields significantly smaller IMPEs than the PACE method in the non-Gaussian scenario as well as in the Gaussian scenario when the number of sample curves is small ($n = 30$). Furthermore, even in the Gaussian scenario with a large number of sample curves ($n = 300$), the SOAP method's performance still appears better than that of the PACE method, although the paired t-test, with a P-value of 0.14, fails to show a significant difference between the average IMPEs in this scenario.

Table 7 shows the IMSEs defined in (7) for estimating the five underlying eigenfunctions using the SOAP method and the PACE method. It shows that the SOAP method provided a smaller average IMSE for all five eigenfunctions in all simulation scenarios except for the first

TABLE 6: Average of the integrated mean prediction errors (IMPEs) defined in (6) for recovering the individual trajectory using the SOAP method and the PACE method for 100 Monte Carlo runs in the second simulation with five true eigenfunctions and non-zero mean function.

Distribution	n	SOAP	PACE	MeanDiff	P-value
Gaussian	30	258 (177)	304 (124)	−46 (15)	1.6e−03
	300	172 (103)	158 (28)	14 (13)	1.4e−01
Non-Gaussian	30	205 (129)	337 (171)	−131 (10)	1.4e−23
	300	152 (51)	197 (55)	−45 (10)	1.1e−05

Note: The corresponding standard deviations of IMPE are given in parentheses. A paired t-test was conducted to compare the IMPEs between the SOAP and PACE method. "MeanDiff" represents the average difference of IMSEs between the SOAP method and the PACE method with the corresponding standard deviation in parentheses.

TABLE 7: Average of the integrated mean square errors (IMSEs) ($\times 10^{-3}$) defined in (7) for estimating the underlying eigenfunctions using the SOAP method and the PACE method for 100 Monte Carlo runs.

		$n = 30$		$n = 300$	
Distribution	Eigenfunction	PACE	SOAP	PACE	SOAP
Gaussian	The first	117 (99)	106 (158)	23 (15)	48 (118)
	The second	659 (565)	208 (182)	227 (336)	76 (98)
	The third	1412 (527)	407 (144)	768 (470)	334 (112)
	The fourth	1545 (485)	356 (193)	1331 (442)	279 (121)
	The fifth	1775 (742)	671 (117)	1532 (395)	647 (66)
Non-Gaussian	The first	164 (172)	78 (142)	36 (29)	3 (1)
	The second	882 (592)	237 (210)	698 (580)	109 (158)
	The third	1565 (482)	512 (160)	1157 (479)	338 (232)
	The fourth	1573 (493)	518 (187)	1322 (452)	546 (245)
	The fifth	1863 (749)	712 (135)	1576 (338)	537 (258)

Note: The corresponding standard deviations of IMSE ($\times 10^{-3}$) are given in parentheses.

eigenfunction in the case where the scores followed a Gaussian distribution and the number of curves was large ($n = 300$). The improvement from using the SOAP method is greater when the scores are non-Gaussian.

7. CONCLUSIONS

In this article, we proposed a novel SOAP method for recovering the underlying individual trajectories as well as the major variation patterns from sparse and irregularly longitudinal observations. The SOAP method directly estimates the empirical functional components from the best approximation perspective. This perspective is different from most conventional methods, such as PACE, which first estimates the covariance function from the centred data and then eigen-decomposes the resulting covariance function to obtain the estimated FPCs. This new

best-approximation perspective enables the SOAP method to recover the individual trajectories without estimating the mean and covariance functions and without requiring that the underlying FPC scores have a Gaussian distribution.

We illustrated the SOAP method by analyzing a dataset of CD4-cell percentages, in which the longitudinal measurements for each individual were sparsely and irregularly observed. Our SOAP method was able to recover the individual CD4 trajectories and explore the major variational sources across all subjects. We also compared the prediction performance of the SOAP method with the PACE method by treating the last observation of each individual as unknown and found that the SOAP method produced better predictions than the PACE method.

Furthermore, we evaluated the performance of the SOAP method and the PACE method in a simulation study. Generally speaking, the SOAP method outperforms the PACE method in both predicting the individual trajectory and recovering the optimal empirical basis functions.

ACKNOWLEDGEMENTS

The authors would like to thank the editor, the associate editor, and two anonymous referees for many insightful comments. These comments were very helpful for us to improve our work. This research was supported by a Postgraduate Scholarship-Doctor (PGS-D) for Y. Nie from the Natural Sciences and Engineering Research Council of Canada (NSERC) and NSERC Discovery grants to L. Wang and J. Cao.

DATA AVAILABILITY STATEMENT

The computing codes of the simulation studies and real data analysis are also provided at https://github.com/caojiguo/SOAP.

REFERENCES

Bosq, D. (2000). *Linear Processes in Function Spaces: Theory and Applications*. Springer-Verlag, New York.

Cardot, H., Ferraty, F., & Sarda, P. (1999). Functional linear model. *Statistics & Probability Letters*, 45, 11–22.

Dauxois, J., Pousse, A., & Romain, Y. (1982). Asymptotic theory for the principal component analysis of a vector random function: Some applications to statistical inference. *Journal of Multivariate Analysis*, 12, 136–154.

Dong, J., Wang, L., Gill, J., & Cao, J. (2018). Functional principal component analysis of GFR curves after kidney transplant. *Statistical Methods in Medical Research*, 27, 3785–3796.

Fukunaga, K. & Koontz, W. L. (1970). Representation of random processes using the finite Karhunen–Loeve expansion. *Information and Control*, 16, 85–101.

Ghalanos, A. & Theussl, S. (2015). *Rsolnp: General non-linear optimization using augmented Lagrange multiplier method*. R package version 1.16.

Hall, P. & Horowitz, J. L. (2007). Methodology and convergence rates for functional linear regression. *The Annals of Statistics*, 35, 70–91.

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

Lawson, C. L. & Hanson, R. J. (1974). *Solving Least Squares Problems*. Prentice Hall, Englewood Cliffs.

Li, Y., Wang, N., & Carroll, R. J. (2013). Selecting the number of principal components in functional data. *Journal of the American Statistical Association*, 108, 1284–1294.

Lin, Z., Wang, L., & Cao, J. (2016). Interpretable functional principal component analysis. *Biometrics*, 72, 846–854.

Mas, A. (2002). Weak convergence for the covariance operators of a Hilbertian linear process. *Stochastic Processes and their Applications*, 99, 117–135.

Müller, H.-G. (2005). Functional modelling and classification of longitudinal data. *Scandinavian Journal of Statistics*, 32, 223–240.

Müller, H.-G. & Stadtmüller, U. (2005). Generalized functional linear models. *The Annals of Statistics*, 33, 774–805.

Nie, Y. & Cao, J. (2020). Sparse functional principal component analysis in a new regression framework. *Computational Statistics & Data Analysis*, 152, 107016.

Nie, Y., Wang, L., Liu, B., & Cao, J. (2018). Supervised functional principal component analysis. *Statistics and Computing*, 28, 713–723.

Peng, J. & Müller, H.-G. (2008). Distance-based clustering of sparsely observed stochastic processes, with applications to online auctions. *The Annals of Applied Statistics*, 2, 1056–1077.

Peng, J. & Paul, D. (2009). A geometric approach to maximum likelihood estimation of the functional principal components from sparse longitudinal data. *Journal of Computational and Graphical Statistics*, 18, 995–1015.

Pezzulli, S. (1993). Some properties of smoothed principal components analysis for functional data. *Computational Statistics*, 8, 1–16.

Pollard, D. (1990). Empirical processes: Theory and applications. *NSF-CBMS Regional Conference Series in Probability and Statistics 2*.

Ramsay, J. & Silverman, B. (2005). *Functional Data Analysis*. Springer-Verlag, New York.

Ramsay, J. O. & Silverman, B. W. (2002). *Applied Functional Data Analysis: Methods and Case Studies*. Springer-Verlag, New York.

Rice, J. A. & Silverman, B. W. (1991). Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society. Series B*, 53, 233–243.

Sang, P., Wang, L., & Cao, J. (2017). Parametric functional principal component analysis. *Biometrics*, 73, 802–810.

Shi, H., Dong, J., Wang, L., & Cao, J. (2021). Functional principal component analysis for longitudinal data with informative dropout. *Statistics in Medicine*, 40, 712–724.

Silverman, B. W. (1996). Smoothed functional principal components analysis by choice of norm. *The Annals of Statistics*, 24, 1–24.

Tran, N. M. (2008). *An Introduction to Theoretical Properties of Functional Principal Component Analysis*. Department of Mathematics and Statistics, The University of Melbourne.

Yao, F., Müller, H.-G., & Wang, J.-L. (2005). Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, 100, 577–590.

Ye, Y. (1987). *Interior Algorithms for Linear, Quadratic, and Linearly Constrained Non-linear Programming*. Department of Earth System Science, Stanford University.

Received 10 September 2020
Accepted 11 June 2021