Biometrics 000, 1–20 DOI: 000
000 0000

Multilevel Sparse Functional Principal Component Analysis

Chong-Zhi Di

Division of Public Health Sciences, Fred Hutchinson Cancer Research Center

1100 Fairview Avenue North, M2-B500, Seattle, WA 98109, U.S.A.

email: cdi@fhcrc.org

and

Ciprian M. Crainiceanu

Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health

615 North Wolfe Stree, Baltimore, MD 21205, U.S.A.

email: ccrainic@jhsph.edu

Summary: The basic observational unit in this paper is a function. Data are assumed to have a natural hierarchy of basic units. A simple example is when functions are recorded at multiple visits for the same subject. Di et al. (2009) proposed Multilevel Functional Principal Component Analysis (MFPCA) for this type of data structure when functions are densely sampled. Here we consider the case when functions are sparsely sampled and may contain as few as 2 or 3 observations per function. As with MFPCA, we exploit the multilevel structure of covariance operators and data reduction induced by the use of principal component bases. However, we address inherent methodological differences in the sparse sampling context to: 1) estimate the covariance operators; 2) estimate the functional scores and predict the underlying curves. We show that in the sparse context 1) is harder and propose an algorithm to circumvent the problem. Surprisingly, we show that 2) is easier via new BLUP calculations. Using simulations and real data analysis we show that the ability of our method to reconstruct underlying curves with few observations is stunning. This approach is illustrated by an application to the Sleep Heart Health Study, which contains two electroencephalographic (EEG) series at two visits for each subject.

Key words: Functional principal component analysis; multilevel models; smoothing

1. Introduction

The basic observational unit in this paper is a function. Data are assumed to have a natural hierarchy of basic units. A simple example is when functions are recorded at multiple visits for the same subject. Di et al. (2009) proposed Multilevel Functional Principal Component Analysis (MFPCA) for this type of data structure when functions are densely sampled. Here we consider the case when functions are sparsely sampled and may contain as few as 2 or 3 observations per function. As with MFPCA, we exploit the multilevel structure of covariance operators and data reduction induced by the use of principal component bases. However, we address inherent methodological differences in the sparse sampling context to: 1) estimate the covariance operators; 2) estimate the functional scores and predict the underlying curves. We show that in the sparse context 1) is harder and propose an algorithm to circumvent the problem. Surprisingly, we show that 2) is easier via new BLUP calculations. Using simulations and real data analysis we show that the ability of our method to reconstruct underlying curves with few observations is stunning. This approach is illustrated by an application to the Sleep Heart Health Study, which contains two electroencephalographic (EEG) series at two visits for each subject.

Traditionally, sparsely sampled data have been treated as longitudinal data (see Diggle et al., 2002). Here we take a different view and treat them as sparse and possibly noisy observations of an underlying smooth signal; see Zhao et al. (2004) and Yao et al. (2005) for excellent discussions on the advantages or disadvantages of this approach. Our methods differ fundamentally from these work because our data contain a natural hierarchy of sparsely sampled functions.

Motivated by modern scientific studies, functional data analysis (FDA; Ramsay and Silverman, 2005) is an increasingly popular area of research. We briefly review the most recent developments in functional principal component analysis (FPCA), which plays a central role

in FDA. The fundamental aims of FPCA are to capture the principal directions of variation and to reduce dimensionality. Besides discussion in Ramsay and Silverman (2005), other relevant research in FPCA includes Ramsay and Dalzell (1991), Silverman (1996), James et al. (2000), and Yao et al. (2005), while important theoretical results can be found in Hall and Hosseini-Nasab (2006).

The original FPCA methodology was designed for a sample of *densely* recorded *independent* functions. The scope was extend in two main directions. First, it was extended to *sparse* functional/longitudinal data (Yao et al., 2005; Müller, 2005). Sparsity is the characteristic of the sampling algorithm that leads to a small number of observations per function and a dense collection of sampling points across all curves. Second, it was extended to functions with a *multilevel* structure, which led to multilevel functional principal component analysis (MFPCA; Di et al., 2009). In this paper, we propose methods for a sample of *sparsely* recorded functions at *multiple levels*.

Our research was motivated by the Sleep Heart Health Study (SHHS), a landmark study of sleep and its impacts on health outcomes. A detailed description of the SHHS can be found in Quan et al. (1997), Di et al. (2009) and Crainiceanu et al. (2009). The SHHS is a multi-center cohort study that utilized the resources of existing, well characterized, epidemiologic cohorts, and conducted further data collection, including measurements of sleep and breathing. Between 1995 and 1997, a sample of 6,441 participants was recruited from the parent studies. Subjects underwent in-home polysomnograms (PSGs). A PSG is a quasi-continuous multi-channel recording of physiological signals acquired during sleep that include two surface electroencephalograms (EEG). After the baseline visit, a second SHHS follow-up visit was undertaken between 1999 and 2003 and included all of the measurements collected at the baseline visit along with a repeat PSG. A total of 3,201 participants (47.8% of baseline cohort) completed a repeat home PSG.

We consider the sleep EEG percent δ-power series of the SHHS data. For each subject at each visit this is a function of time calculated in adjacent 30-second intervals and has 960 observations in a 8-hour interval of sleep. We compare the full data analysis with the analysis of data where each function is sub-sampled at a random set of 30-second intervals. Our findings indicate: 1) stunning ability to predict subject-specific curves even when the number of observations sampled is very small, say 3 or 6; 2) remarkable consistency between the analyses of the full and reduced data sets. Our results do not advocate throwing away data. Instead, they indicate that sparse data analysis of multilevel functions is a powerful inferential tool when data *can only be or was sparsely collected*.

The remainder of this paper is organized as follows. Section 2 introduces Multilevel Functional Principal Component Analysis (MFPCA) for sparse data. Section 3 provides mathematical details for predicting the principal component scores and curves. Section 4 describes extensive simulation studies for realistic settings. Section 5 describes the application of our methodology to the SHHS data set. Section 6 presents our discussion.

2. MFPCA for sparsely sampled functions

In this section, we briefly review the MFPCA technique proposed by Di et al. (2009), and then discuss statistical issues to deal with sparsity.

The MFPCA was designed to capture dominant modes of variations and reduce dimensions for multilevel functional data. This method decomposes the total functional variation into between subject and within subject variations via functional analysis of variance (FANOVA), and conducts FPCA at both levels. More precisely, let $Y_{ij}(t)$ denote the observed function for subject i at visit j, the two way FANOVA decomposes the total variation as

$$\begin{aligned} Y_{ij}(t) &= \mu(t) + \eta_j(t) + Z_i(t) + W_{ij}(t) + \epsilon_{ij}(t), \qquad (1) \\ & i \in \{1, 2, \cdots, n\}, \;\; j \in \{1, 2, \cdots, n_i\}, \;\; t \in \{t_{ijs} : s = 1, 2, \cdots, T_{ij}\} \subset \mathcal{T}, \end{aligned}$$

where $\mu(t)$ and $\eta_j(t)$ are fixed functional effects that represent the overall mean function and visit-specific shifts, respectively, $Z_i(t)$ and $W_{ij}(t)$ are the subject-specific and visit-specific deviations, respectively, and $\epsilon_{ij}(t)$ is measurement error with mean 0 and variance σ^2. The level 1 and 2 processes, $Z_i(t)$ and $W_{ij}(t)$, are assumed to be uncorrelated mean 0 stochastic processes. The idea of MFPCA is to decompose both $Z_i(t)$ and $W_{ij}(t)$ using the the Karhunen-Loève (KL) expansion (Karhunen, 1947; Loève, 1945), i.e.,

$$Z_i(t) = \sum_{k=1}^{N_1} \xi_{ik}\, \phi_k^{(1)}(t)\ , \quad W_{ij}(t) = \sum_{l=1}^{N_2} \zeta_{ijl}\, \phi_l^{(2)}(t)\,, \tag{2}$$

where $\phi_k^{(1)}(t)$ and $\phi_l^{(2)}(t)$ are level 1 and level 2 eigenfunctions, respectively, and ξ_{ik} and ζ_{ijl} are mean zero random variables called principal component scores. The variances of ξ_{ik} and ζ_{ijl}, $\lambda_k^{(1)}$ and $\lambda_l^{(2)}$, respectively, are the level 1 and 2 eigenvalues that characterize the magnitude of variation in the direction of the corresponding eigenfunctions. The number of principal components, N_1 and N_2, could be either finite integers or ∞. Combining model (1) with the KL expansions (2), one obtains the MFPCA model

$$Y_{ij}(t) = \mu(t) + \eta_j(t) + \sum_{k=1}^{N_1} \xi_{ik}\, \phi_k^{(1)}(t) + \sum_{l=1}^{N_2} \zeta_{ijl}\, \phi_l^{(2)}(t) + \epsilon_{ij}(t). \tag{3}$$

The MFPCA reduces high dimensional hierarchical functional data $\{\, Y_{i1}(t), \cdots, Y_{in_i}(t)\,\}$ into the low dimensional principal component score vectors, including subject level (level 1) scores $\boldsymbol{\xi}_i = (\xi_{i1}, \cdots, \xi_{iN_1})$ and subject/visit level (level 2) scores $\boldsymbol{\zeta}_{ij} = (\zeta_{ij1}, \cdots, \zeta_{ijN_2})$, while retaining most information contained in the data.

Equation (1) introduced FANOVA in full generality without details about the sampling design for t_{ijs}. The set of sampling points $\{t_{ijs} : s = 1, 2, \cdots, T_{ij}\}$ for subject i at visit j could be dense or sparse, regular or irregular, depending on the application. Although Di et al. (2009) discussed potential difficulties with sparse designs, they focused on densely and regularly recorded functional data. In the following, we will discuss and address new

problems raised by the sparse sampling design. This will lead to methods that are related to but markedly different from MFPCA.

2.1 *Estimating eigenvalues and eigenfunctions*

Throughout the paper, we assume sparse and irregular grid points. More precisely, for each subject and visit, the number of grid points T_{ij} is relatively small, and the set of grid points $\{t_{ijs} : s = 1, 2, \cdots, T_{ij}\}$ is a random sample of $\mathcal{T}$. We also assume that the set of grid points are different across subjects and visits.

The first step of MFPCA is to estimate the eigenvalues and eigenfunctions. This can be done by the method of moments and eigen-analysis for dense functional data, but smoothing is needed for sparse functional data. Let $K_B(s,t) = \text{cov}\{\, Z_i(s), Z_i(t)\,\}$ be the covariance function for level 1 processes ("between" covariance), $K_W(s,t) = \text{cov}\{\, W_{ij}(s), W_{ij}(t)\,\}$ be the covariance function for level 2 processes ("within"). The total covariance function $K_T(s,t)$ contains three sources of variation, that is, $K_T(s,t) = K_B(s,t) + K_W(s,t) + \sigma^2\, I(t = s)$. One can easily verify that $\text{E}\{\, Y_{ij}(t)\,\} = \mu(t) + \eta_j(t)$, $\text{cov}\{\, Y_{ij}(s), Y_{ij}(t)\,\} = K_B(s,t) + K_W(s,t) + \sigma^2\, I(t = s)$, and $\text{cov}\{\, Y_{ij}(s), Y_{ik}(t)\,\} = K_B(s,t)$. These results suggest the following convenient algorithm to estimate the eigenvalues and eigenfunctions. Because functions were sparsely sampled over irregular grid points, smoothing will be used repeatedly to estimate the underlying means and covariances.

Sparse MFPCA Algorithm

Step 1. Use scatter plot smoothing using all pairs $\{(t_{ijs}, Y_{ij}(t_{ijs})\,) : i = 1, ..., n; j = 1, ..., n_i; s = 1, ..., T_{ij}\}$ to obtain an estimate of $\mu(t)$, $\hat{\mu}(t)$;

Step 2. Use scatter plot smoothing using all pairs $\{(t_{ijs}, Y_{ij}(t_{ijs}) - \hat{\mu}(t_{ijs})\,) : i = 1, ..., n; s = 1, ..., T_{ij}\}$ to obtain an estimate of $\eta_j(t)$, $\hat{\eta}_j(t)$;

Step 3. Estimate $\hat{K}_B(s,t)$ by bivariate smoothing of all products $\{Y_{ij_1}(t_{ij_1s}) - \hat{\mu}(t_{ij_1s}) - \hat{\eta}_{j_1}(t_{ij_1s})\}\{Y_{ij_2}(t_{ij_2r}) - \hat{\mu}(t_{ij_2r}) - \hat{\eta}_{j_2}(t_{ij_2r})\}$ with respect to (t_{ij_1s}, t_{ij_2r}) for all i, j_1, j_2, r and s;

Step 4. Estimate $\hat{K}_T(s,t)$ by bivariate smoothing of all products $\{Y_{ij}(t_{ijs}) - \hat{\mu}(t_{ijs}) - \hat{\eta}_j(t_{ijs})\}\{Y_{ij}(t_{ijr}) - \hat{\mu}(t_{ijr}) - \hat{\eta}_j(t_{ij_2r})\}$ with respect to (t_{ijs}, t_{ijr}) for all i, j, r, s with $r \neq s$, and set $\hat{K}_W(s,t) = \hat{K}_T(s,t) - \hat{K}_B(s,t)$;

Step 5. Use eigen-analysis on $\hat{K}_B(s,t)$ to obtain $\hat{\lambda}_k^{(1)}$, $\hat{\phi}_k^{(1)}(t)$; use eigen-analysis on $\hat{K}_W(s,t)$ to obtain $\hat{\lambda}_l^{(2)}$, $\hat{\phi}_l^{(2)}(t)$.

Step 6. Estimate the nugget variance σ^2 by smoothing $\{Y_{ij}(t_{ijs}) - \hat{\mu}(t_{ijs}) - \hat{\eta}_j(t_{ijs})\}^2 - \hat{K}_T(t_{ijs}, t_{ijs})$ with respect to t_{ijs} for all possible i, j, s.

For univariate and bivariate smoothing we use penalized spline smoothing (Ruppert et al., 2003) with the smoothing parameter estimated via restricted maximum likelihood (REML). Cross validation (CV) or generalized cross validation (GCV) could also be used. Alternatively, one can also use local polynomial smoothing Fan and Gijbels (1996) with cross validation to choose the smoothing parameter. In *Step 3*, the diagonal elements (when $s = r$) are dropped when estimating the total covariance $K_T(s,t)$, because they are contaminated by measurement error. In contrast, diagonal elements are included when estimating the between covariance $K_B(s,t)$.

In *Step 4*, one needs to determine the dimensions of level 1 and 2 spaces, namely, N_1 and N_2, respectively. Although they are allowed to be ∞ in theory, in practice a low dimensional principal component space suffices to approximate the functional space. We will discuss this issue in more details later.

2.2 *Principal component scores*

Once the fixed functional effects $\mu(t)$, $\eta_j(t)$, the eigenvalues $\lambda_k^{(1)}$, $\lambda_l^{(2)}$ and the eigenfunctions $\phi_k^{(1)}(t)$, $\phi_l^{(2)}(t)$ are estimated, the MFPCA model can be re-written as a linear mixed model

$$\begin{cases} Y_{ijs} &= \mu(t_{ijs}) + \eta_j(t_{ijs}) + \sum_{k=1}^{N_1} \xi_{ik}\phi_k^{(1)}(t_{ijs}) + \sum_{l=1}^{N_2} \zeta_{ijl}\phi_l^{(2)}(t_{ijs}) + \epsilon_{ijs} \\ \xi_{ik} &\sim N\{0, \lambda_k^{(1)}\},\ \zeta_{ijl} \sim N\{0, \lambda_l^{(2)}\},\ \epsilon_{ijs} \sim N(0, \sigma^2), \end{cases} \tag{4}$$

where $Y_{ijs} := Y_{ij}(t_{ijs})$ and $\epsilon_{ijs} = \epsilon_{ij}(t_{ijs})$. The random effects ξ_{ik} and ζ_{ijl} are principal component scores that we are trying to estimate. Thus, one could use the mixed model inferential machinery to estimate the scores, for example, using the best linear unbiased prediction (BLUP). The BLUP gives point estimates of the scores, and one could also construct their 95% confidence intervals.

Note that the subject-specific effect, $Z_i(t)$, and the visit-specific effect, $W_{ij}(t)$, are linear functions of the random effects ξ_{ik} and ζ_{ijl}, respectively. Thus, $Z_i(t)$, $W_{ij}(t)$ and their variability can be estimated directly from the BLUP formulas for the random effects. The BLUPs of $Z_i(t)$ and $W_{ij}(t)$ are shrinkage estimators, which automatically combine information from different visits of the same subject and across subjects. More information is borrowed when the measurement error variance, σ, is large and when the number of observations at the subject level is small.

2.3 *Choosing the dimensions N_1 and N_2*

Two popular methods for estimating the dimension of the functional space in the single level case are cross validation (Rice and Silverman, 1991) and Akaike's Information Criterion (or AIC, as in Yao et al., 2005). These methods can be generalized to the multilevel setting. For example, one could use the leave-one-subject-out cross validation criterion to select the number of dimensions. Define the cross validation score as

$$CV(N_1, N_2) = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \sum_{s=1}^{T_{ij}} \{ Y_{ij}(t_{ijs}) - \hat{Y}_{ij}^{(-i),N_1,N_2}(t_{ijs}) \}^2,$$

where $\hat{Y}_{ij}^{(-i),N_1,N_2}(t_{ijs})$ is the predicted curve for subject i at visit j, computed after removing the data from subject i. The estimated number of dimensions N_1 and N_2 are the arguments that minimize $CV(N_1, N_2)$. In practice, the leave-one-subject-out cross validation method may be too computationally intensive, and an m-fold cross validation can serve as a fast alternative. This method divides the subjects into m groups, and the prediction error for the

m^{th} group is calculated by fitting a model using the data from other groups. The number of dimensions are chosen as those that minimize the total prediction error. Similar criteria could be designed for leave-visits-out.

One could also use a fast method proposed by Di et al. (2009). More precisely, let P_1 and P_2 be two thresholds and define

$$N_1 = \min\{k : \rho_k^{(1)} \geqslant P_1, \lambda_k^{(1)} < P_2\},\ N_2 = \min\{k : \rho_k^{(2)} \geqslant P_1, \lambda_k^{(2)} < P_2\},$$

where $\rho_k^{(j)} = (\lambda_1^{(j)} + \ldots + \lambda_k^{(j)})/(\lambda_1^{(j)} + \ldots + \lambda_k^{(j)} + \cdots)$, $j = 1, 2$, is the proportion of variation explained by the first k principal components at level j. Intuitively, this method chooses the number of dimension at each level to be the smallest integer k such that the first k components explain more than P_1 of the total variation while any component after the kth explains less than P_2 of the variation. To use this method, the thresholds P_1 and P_2 need to be carefully tuned using simulation studies or cross validations. In practice, we found that $P_1 = 90\%$ and $P_2 = 5\%$ are often good choices.

2.4 *Iterative procedure to improve accuracy*

Initial estimates of the mean functions and eigenfunctions via smoothing are typically accurate in the dense functional data, and less accurate for the sparse data. To improve estimation accuracy, one may adopt an iterative procedure as follows.

Iterative Sparse MFPCA Algorithm

Step 1. Obtain initial estimates, $\hat{\mu}^0(t)$, $\hat{\eta}_j^0(t)$, $\hat{\lambda}_k^{(1),0}$, $\hat{\lambda}_l^{(2),0}$, $\hat{\phi}_k^{(1),0}(t)$, $\hat{\phi}_l^{(2),0}(t)$ and $\hat{\sigma}^{2,0}$ for all j, k, l, using the sparse MFPCA algorithm described in Section 2.2; estimate principal component scores, $\hat{\boldsymbol{\xi}}_i^0$ and $\hat{\boldsymbol{\zeta}}_i^0$, using formulas that will be described in Section 3;

Step 2. Apply *Steps 1–2* of the sparse MFPCA algorithm on $Y_{ij}(t_{ijs}) - \sum_{k=1}^{N_1} \hat{\xi}_{ik}^0 \hat{\phi}_k^{(1),0}(t_{ijs}) - \sum_{l=1}^{N_2} \hat{\zeta}_{ijl}^0 \hat{\phi}_l^{(2),0}(t_{ijs})$, and obtain updated mean functions $\hat{\mu}^1(t)$ and $\hat{\eta}_j^1(t)$;

Step 3. Apply *Steps 3–6* of the sparse MFPCA algorithm on $Y_{ij}(t_{ijs}) - \hat{\mu}^1(t_{ijs}) - \hat{\eta}_j^1(t_{ijs})$,

and obtain updated eigenvalues $\hat{\lambda}_k^{(1),1}$, $\hat{\lambda}_l^{(2),1}$, eigenfunctions $\hat{\phi}_k^{(1),1}(t)$, $\hat{\phi}_l^{(2),1}(t)$ and $\hat{\sigma}^{2,1}$ for all j, k, l;

Step 4. Update principal component scores, $\hat{\boldsymbol{\xi}}_i^1$ and $\hat{\boldsymbol{\zeta}}_i^1$, based on the new estimates from *Steps 2–3*;

Step 5. Stop if certain criteria are met. Otherwise, set $\hat{\mu}^1(t)$, $\hat{\eta}_j^1(t)$, $\hat{\lambda}_k^{(1),1}$, $\hat{\lambda}_l^{(2),1}$, $\hat{\phi}_k^{(1),1}(t)$, $\hat{\phi}_l^{(2),1}(t)$, $\hat{\boldsymbol{\xi}}_i^1$ and $\hat{\boldsymbol{\zeta}}_i^1$ as initial estimates and repeat *Step 2–5* until the algorithm converges.

One reasonable set of the stopping criteria could be $\|\hat{\mu}^1(t) - \hat{\mu}^0(t)\| < \varepsilon_1$, $\|\hat{\eta}_j^1(t) - \hat{\eta}_j^0(t)\| < \varepsilon_1$, $\|\hat{\phi}_k^{(1),1}(t) - \hat{\phi}_k^{(1),0}(t)\| < \varepsilon_1$, $\|\hat{\phi}_l^{(2),1}(t) - \hat{\phi}_l^{(2),0}(t)\| < \varepsilon_1$, $|\hat{\lambda}_k^{(1),1} - \hat{\lambda}_k^{(1),0}| < \varepsilon_2$ and $|\hat{\lambda}_l^{(2),1} - \hat{\lambda}_l^{(2),0}| < \varepsilon_2$ for all j, k and l, where $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ are small pre-specified thresholds.

This algorithm iterates between updating the mean functions, the eigenfunctions, eigenvalues, and principal component scores. One advantage of the iterative procedure is, as pointed out by Yao and Lee (2006), that the working data in *Step 2* is asymptotically independent. Thus, theoretical results for local polynomial smoothing or penalized splines would ensure that smoothing estimates of the mean functions, $\mu(t)$ and $\eta_j(t)$, have good asymptotic behavior.

3. Prediction of principal component scores and various curves

This section provides BLUP calculation results for principal component scores and function prediction at various levels. Some heavy notation is unavoidable, but results are crucial for the implementation of our quick algorithms.

3.1 *Prediction of principal component scores*

We introduce some notations before presenting the formulas for the principal component scores. Let $\boldsymbol{\xi}_i = (\xi_{i1}, \xi_{i2}, \ldots, \xi_{iN_1})^T$ be an $N_1 \times 1$ vector, $\boldsymbol{\zeta}_{ij} = (\zeta_{ij1}, \zeta_{ij2}, \ldots, \zeta_{ijN_2})^T$ be an $N_2 \times 1$ vector, $\boldsymbol{\zeta}_i = (\boldsymbol{\zeta}_{i1}^T, \boldsymbol{\zeta}_{i2}^T, \cdots, \boldsymbol{\zeta}_{in_i}^T)^T$ be an $(N_2 n_i) \times 1$ vector, $\boldsymbol{Y}_{ij} = (Y_{ij1}, \cdots, Y_{ijT_{ij}})$ be a $T_{ij} \times 1$ vector, $\boldsymbol{t}_{ij} = (t_{ij1}, \cdots, t_{ijT_{ij}})$ be a $T_{ij} \times 1$ vector, $\boldsymbol{Y}_i = (\boldsymbol{Y}_{i1}^T, \cdots, \boldsymbol{Y}_{in_i}^T)^T$ be a $(\sum_j T_{ij}) \times 1$

vector. Calculations will be done conditionally on $\mu(t)$, $\eta_j(t)$, $\lambda_k^{(1)}$, $\lambda_l^{(2)}$, $\phi_k^{(1)}(t)$ and $\phi_l^{(2)}(t)$, which can be estimated using methods described in the previous sections. It is straightforward to evaluate these functions at observed grid points, i.e, $\boldsymbol{\mu}_{ij} = \{\mu(t_{ij1}), \cdots, \mu(t_{ijT_{ij}})\}^T$, $\boldsymbol{\mu}_i = (\boldsymbol{\mu}_{ij}^T, \cdots, \boldsymbol{\mu}_{in_i}^T)^T$, $\boldsymbol{\eta}_{ij} = \{\eta_j(t_{ij1}), \cdots, \eta_j(t_{ijT_{ij}})\}^T$, $\boldsymbol{\eta}_i = (\boldsymbol{\eta}_{ij}^T, \cdots, \boldsymbol{\eta}_{in_i}^T)^T$, $\boldsymbol{\phi}_{k,ij}^{(1)} = \{\phi_k^{(1)}(t_{ij1}), \cdots, \phi_k^{(1)}(t_{ijT_{ij}})\}^T$ and $\boldsymbol{\phi}_{l,ij}^{(2)} = \{\phi_l^{(2)}(t_{ij1}), \cdots, \phi_l^{(2)}(t_{ijT_{ij}})\}^T$. Let $\boldsymbol{\Phi}_{ij}^{(1)}$ denote a $T_{ij} \times N_1$ matrix whose k^{th} column is given by $\boldsymbol{\phi}_{k,ij}^{(1)}$, $\boldsymbol{\Phi}_{ij}^{(2)}$ denote a $T_{ij} \times N_2$ matrix whose l^{th} column is given by $\boldsymbol{\phi}_{l,ij}^{(2)}$, $\boldsymbol{\Lambda}^{(1)}$ denote an $N_1 \times N_1$ diagonal matrix with diagonal elements $(\lambda_1^{(1)}, \cdots, \lambda_{N_1}^{(1)})$ and $\boldsymbol{\Lambda}^{(2)}$ denote an $N_2 \times N_2$ diagonal matrix with diagonal elements $(\lambda_1^{(2)}, \cdots, \lambda_{N_2}^{(2)})$. The following proposition gives the point estimates and variance for the principal component scores.

Proposition 1. Under the MFPCA model (4), the best linear unbiased prediction for principal component scores $(\boldsymbol{\xi}_i^T, \boldsymbol{\zeta}_i^T)$ has the following form,

$$\begin{pmatrix} \hat{\boldsymbol{\xi}}_i \\ \hat{\boldsymbol{\zeta}}_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_i \\ \boldsymbol{B}_i \end{pmatrix} \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{Y}_i - \boldsymbol{\mu}_i - \boldsymbol{\eta}_i), \tag{5}$$

and their covariance matrix, $\text{cov}\{(\hat{\boldsymbol{\xi}}_i^T - \boldsymbol{\xi}_i^T, \hat{\boldsymbol{\zeta}}_i^T - \boldsymbol{\zeta}_i^T) \,|\, \boldsymbol{Y}_i\}$, is given by

$$\begin{pmatrix} \boldsymbol{\Lambda}^{(1)} & 0 \\ 0 & \boldsymbol{\Lambda}^{(2)} \bigotimes \boldsymbol{I}_{n_i \times n_i} \end{pmatrix} - \begin{pmatrix} \boldsymbol{A}_i \\ \boldsymbol{B}_i \end{pmatrix} \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{A}_i^T, \boldsymbol{B}_i^T), \tag{6}$$

where $\bigotimes$ denotes the Kronecker product, $\boldsymbol{A}_i := \text{cov}(\boldsymbol{\xi}_i, \boldsymbol{Y}_i)$ is an $N_1 \times (\sum_j T_{ij})$ matrix, $\boldsymbol{B}_i := \text{cov}(\boldsymbol{\zeta}_i, \boldsymbol{Y}_i)$ is an $(N_2 n_i) \times (\sum_j T_{ij})$ matrix and $\boldsymbol{\Sigma}_i := cov(\boldsymbol{Y}_i)$ is a $(\sum_j T_{ij}) \times (\sum_j T_{ij})$ matrix. The matrix $\boldsymbol{A}_i$ has the form $\boldsymbol{A}_i = (\boldsymbol{\Lambda}^{(1)}\boldsymbol{\Phi}_{i1}^{(1)^T}, \boldsymbol{\Lambda}^{(1)}\boldsymbol{\Phi}_{i2}^{(1)^T}, \cdots, \boldsymbol{\Lambda}^{(1)}\boldsymbol{\Phi}_{in_i}^{(1)^T})$, and $\boldsymbol{B}_i$ is a block diagonal matrix with diagonal elements $\{\boldsymbol{\Lambda}^{(2)}\boldsymbol{\Phi}_{i1}^{(2)^T}, \boldsymbol{\Lambda}^{(2)}\boldsymbol{\Phi}_{i2}^{(2)^T}, \cdots, \boldsymbol{\Lambda}^{(2)}\boldsymbol{\Phi}_{in_i}^{(2)^T}\}$. Let $\boldsymbol{\Sigma}_{i,jk} := \text{cov}(\boldsymbol{Y}_{ij}, \boldsymbol{Y}_{ik})$ be the (j,k) block of $\boldsymbol{\Sigma}_i$ with size $T_{ij} \times T_{ik}$. When $j = k$,

$$\boldsymbol{\Sigma}_{i,jj} = \boldsymbol{\Phi}_{ij}^{(1)}\boldsymbol{\Lambda}^{(1)}\boldsymbol{\Phi}_{ij}^{(1)^T} + \boldsymbol{\Phi}_{ij}^{(2)}\boldsymbol{\Lambda}^{(2)}\boldsymbol{\Phi}_{ij}^{(2)^T} + \sigma^2 \boldsymbol{I}_{T_{ij} \times T_{ij}},$$

where $\boldsymbol{I}_{T_{ij} \times T_{ij}}$ is an identity matrix, and when $j \neq k$, $\boldsymbol{\Sigma}_{i,jk} = \boldsymbol{\Phi}_{ij}^{(1)}\boldsymbol{\Lambda}^{(1)}\boldsymbol{\Phi}_{ik}^{(1)^T}$.

Equation (5) provides the best prediction of the principal component scores under the Gaussian assumptions, and the best linear prediction otherwise. Thus, the Gaussian assumptions in model (4) can be relaxed. Crainiceanu et al. (2009) also provides formulae for the BLUPs of the principal component scores. Their results are applicable to balanced and dense designs only, i.e. to cases when each function is measured at exactly the same set of grid points; the formulae in Theorem 1 are applicable both for balanced and unbalanced designs. When data are balanced and dense, the results in Crainiceanu et al. (2009) are preferable because they avoid inverting large matrixes. Otherwise, one should use results in Theorem 1.

Once estimates of principal component scores are obtained, they can be used in further analysis either as outcome or predictor variables. For example, Di et al. (2009) explored the distribution of subject specific principal component scores in different sex and age groups. Di et al. (2009) and Crainiceanu et al. (2009) considered generalized multilevel functional regression, which modeled principal component scores as predictors for health outcomes, such as hypertension. These analyses can be extended to the sparse case.

3.2 *Prediction of functional effects*

Our approach for sparsely recorded functions allows estimation of the covariance structure of the population of functions at various levels; it also provides a simple algorithm for predicting subject-level or subject/visit-level curves. This process intrinsically borrows information both across subjects and across visits within-subjects, and yields surprisingly accurate predictions; see our simulation results in Section 4 and data analysis in Section 5 for demonstration. The following theorem provides the formulas for prediction of various functions and their confidence intervals. Results are derived based on the MFPCA model (4) and Proposition 1.

Proposition 2. Under the MFPCA model (4), the best linear unbiased prediction for the subject-specific curve, $Z_i(t)$, is given by

$$\hat{Z}_i(t) = \boldsymbol{\Phi}^{(1)}(t)^T \, \hat{\boldsymbol{\xi}}_i = \boldsymbol{\Phi}^{(1)}(t)^T \, \boldsymbol{A}_i \, \boldsymbol{\Sigma}_i^{-1} \, (\boldsymbol{Y}_i - \boldsymbol{\mu}_i - \boldsymbol{\eta}_i),$$

with variance, $\mathrm{var}\{\hat{Z}_i(t) - Z_i(t) \,|\, \boldsymbol{Y}_i\}$, provided by

$$\boldsymbol{\Phi}^{(1)}(t)^T (\, \boldsymbol{\Lambda}^{(1)} - \boldsymbol{A}_i \boldsymbol{\Sigma}_i^{-1} \boldsymbol{A}_i^T \,) \, \boldsymbol{\Phi}^{(1)}(t),$$

where $\boldsymbol{\Phi}^{(1)}(t) := \{ \, \phi_1^{(1)}(t), \phi_2^{(1)}(t), \cdots, \phi_{N_1}^{(1)}(t) \, \}^T$. The best linear unbiased prediction for the visit-specific curve, $W_{ij}(t)$, is given by

$$\hat{W}_{ij}(t) = \boldsymbol{\Phi}^{(2)}(t)^T \, \hat{\boldsymbol{\zeta}}_{ij} = \boldsymbol{\Phi}^{(2)}(t)^T \, \boldsymbol{H}_j \boldsymbol{B}_i \, \boldsymbol{\Sigma}_i^{-1} \, (\boldsymbol{Y}_i - \boldsymbol{\mu}_i - \boldsymbol{\eta}_i),$$

with variance, $\mathrm{var}\{\hat{W}_{ij}(t) - W_{ij}(t) \,|\, \boldsymbol{Y}_i\}$, provided by

$$\boldsymbol{\Phi}^{(2)}(t)^T (\, \boldsymbol{\Lambda}^{(2)} \bigotimes \boldsymbol{I}_{n_i \times n_i} - \boldsymbol{H}_j \boldsymbol{B}_i \boldsymbol{\Sigma}_i^{-1} \boldsymbol{A}_i^T \, \boldsymbol{H}_j^T \,) \, \boldsymbol{\Phi}^{(2)}(t),$$

where $\boldsymbol{\Phi}^{(2)}(t) := \{ \, \phi_1^{(2)}(t), \phi_2^{(2)}(t), \cdots, \phi_{N_2}^{(2)}(t) \, \}^T$, and $\boldsymbol{H}_j = (\boldsymbol{H}_{j,1}, \cdots, \boldsymbol{H}_{j,n_i})$ is an $N_2 \times (N_2 n_i)$ matrix with $\boldsymbol{H}_{j,k} = \boldsymbol{I}_{N_2 \times N_2}$ if $j = k$ and $\boldsymbol{H}_{j,k} = \boldsymbol{0}$ otherwise. The individual curve $Y_{ij}(t)$ can be predicted by

$$\hat{Y}_{ij}(t) = \hat{\mu}(t) + \hat{\eta}_j(t) + \hat{Z}_i(t) + \hat{W}_{ij}(t),$$

with variance, $\mathrm{var}\{ \, \hat{Y}_{ij}(t) - Y_{ij}(t) \,|\, \boldsymbol{Y}_i \, \}$, estimated by

$$\boldsymbol{\Phi}(t)^T \boldsymbol{G}_j \left\{ \begin{pmatrix} \boldsymbol{\Lambda}^{(1)} & 0 \\ 0 & \boldsymbol{\Lambda}^{(2)} \bigotimes \boldsymbol{I}_{n_i \times n_i} \end{pmatrix} - \begin{pmatrix} \boldsymbol{A}_i \\ \boldsymbol{B}_i \end{pmatrix} \boldsymbol{\Sigma}_i^{-1} \, (\boldsymbol{A}_i^T, \boldsymbol{B}_i^T) \right\} \boldsymbol{G}_j^T \boldsymbol{\Phi}(t),$$

where $\boldsymbol{G}_j = (\boldsymbol{I}_{N_1 \times N_1}, \boldsymbol{H}_j)$ and $\boldsymbol{\Phi}(t) = (\boldsymbol{\Phi}^{(1)}(t)^T, \boldsymbol{\Phi}^{(2)}(t)^T)^T$.

Proofs of Proposition 1 and 2 can be obtained by direct BLUP calculations (see, e.g., Ruppert et al., 2003) for general linear mixed models, and are omitted. The estimators above require plugging in consistent estimates of the fixed functional effects $\mu(t)$ and $\eta_j(t)$, eigenvalues $\lambda_k^{(1)}$ and $\lambda_l^{(2)}$, eigenfunctions $\phi_k^{(1)}(t)$ and $\phi_k^{(2)}(t)$, as well as variance σ^2. Thus, the

variance estimators in Theorem 2 do not account for the uncertainty associated to estimating these quantities. This is a common practice in functional data analysis (Yao et al., 2005) and produces satisfactory results.

4. Simulations

To evaluate finite sample performance, we conducted simulation studies under a variety of settings. The data was generated from a true model used in Di et al. (2009), except that the curves are sampled on a sparse set of grid points. More precisely, the true model is

$$Y_{ij}(t_{ijm}) = \mu(t_{ijm}) + \sum_{k=1}^{4} \xi_{ik}\, \phi_k^{(1)}(t_{ijm}) + \sum_{l=1}^{4} \zeta_{ijl}\, \phi_l^{(2)}(t_{ijm}) + \epsilon_{ij}(t_{ijm})\,,$$

where $\xi_{ik} \sim N(0, \lambda_k^{(1)})$, $\zeta_{ijl} \sim N(0, \lambda_l^{(2)})$, $\epsilon_{ij}(t_{ijm}) \sim N(0, \sigma^2)$ and $(t_{ijm} : m = 1, 2, \cdots, T_{ij})$ is a set of grid points in the interval $[0, 1]$. The set of grid points are generated uniformly in the interval $[0, 1]$, and are different across subjects and visits. Let J denotes the maximum number of visits per subject, i.e., $J = \max\{n_i : i = 1, \cdots, n\}$ and N denote the maximum number of grid points per curve, i.e., $N = \max\{T_{ij} : i = 1, \cdots, n; j = 1, \cdots, n_i\}$. The true mean function is $\mu(t) = 8t(1 - t)$. The true eigenvalues are $\lambda_k^{(1)} = 0.5^{k-1}, k = 1, 2, 3, 4$, and $\lambda_l^{(2)} = 0.5^{l-1}, l = 1, 2, 3, 4$, and the eigenfunctions are given as follows.

Level 1: $\phi_k^{(1)}(t) = \{\sqrt{2}\sin(2\pi t),\ \sqrt{2}\cos(2\pi t),\ \sqrt{2}\sin(4\pi t),\ \sqrt{2}\cos(4\pi t)\}$.

Level 2: $\phi_1^{(2)}(t) = 1$, $\phi_2^{(2)}(t) = \sqrt{3}(2t - 1)$, $\phi_3^{(2)}(t) = \sqrt{5}(6t^2 - 6t + 1)$,

$\phi_4^{(2)}(t) = \sqrt{7}(20t^3 - 30t^2 + 12t - 1)$.

We considered several scenarios corresponding to various choices of n, N and σ^2, and simulated $1,000$ data sets for each scenario. We considered $n = 100, 200, 300$ subjects, $J = 2$ visits per subject, $N = 3, 6, 9, 12$ measurements per function, and magnitude of noise $\sigma = 0.01, 0.5, 1, 2$. Due space limitations, we report results for a few scenarios, and present the others in the supplementary file.

We found that the estimation accuracy of eigenvalues and eigenfunctions increases with

the number of subjects n, the number of visits J, and the number of grid points N. With $n = 100$ subjects and $J = 2$ visits, the first two components at both levels can be recovered well when $N = 3$; more precisely, the shapes of estimated eigenfunctions approximate well the true eigenfunctions and the estimated eigenvalues are close to their true values. When the number of grid points increases to $N = 9$, all four principal components at both levels can be estimated well. Table 1 reports the root mean square errors (RMSE) for eigenvalues and root integrated mean square errors for eigenfunctions in various simulation settings. More details on these results can be found in the supplementary file.

[Table 1 about here.]

We also evaluated the finite sample performance of predictions of subject and subject/visit level curves. We discuss our results for the case when $n = 200$, $J = 2$ with different levels of noise level, σ, and number of observations per visit, N. Figure 1 shows predictions for subject- and visit-specific curves for the first subject under various scenarios. In the sparsest case, $N = 3$, the BLUPs can still capture the rough shape of the curve, the confidence bands are relatively wide because of the large amount of uncertainty, but cover the true curve in most cases. For example, the predicted curve corresponding to subject 1 visit 2 (first panel in the middle row) identifies a local minimum at $t = 0.2$, even though there are no observations around the area. This is probably due to the additional information provided by the data for the same subject at visit 1 (first panel in the top row). The bottom row shows results for subject-specific curve, $Z_1(t)$, indicating that the BLUP estimates captures the major trend, though misses some of the details. When the number of grid points increases, predictions of both individual curves and subject specific curves improve. When $N = 9$, these predictions are already very close to the true curves.

[Figure 1 about here.]

In summary, the sparse MFPCA algorithm is able to capture dominating modes of varia-

tions at both levels for sparse data, in a typical setting with hundreds of subjects and a few visits per subject. Predictions of curves via BLUPs also perform very well in finite samples.

5. Application

We now illustrate the use of sparse MFPCA on the SHHS data. The data contains dense EEG series for 3201 subjects at two visits per subject. Di et al. (2009) analyzed the full SHHS data using the MFPCA methodology and extracted dominant modes of variations at both between- and within-subject levels. In this paper, we take a sparse random sample of the SHHS data. More precisely, for each subject and each visit, we take a random sample of size N without replacement from the sleep EEG percent δ-power time series. We perform analysis with N=3, 6, 12, 24 observations per visit and compare the results using the proposed sparse MFPCA method on the sub-sampled, now sparse, data with the MFPCA method on the entire data set.

One might argue that it is somewhat artificial and unnecessary to analyze a sparse subset of the data while the full data set is available. However, this will provide additional insight into the performance of our method and build confidence into using these methods even when N is small. The analysis based on the full data (henceforth denoted "full analysis") provides a "golden standard", which can be compared with the analysis based on sparse data. The sparse MFPCA analyses with increased number of observations, N, will further illustrate how much information is lost/gained at different level of sparsity, in realistic settings.

Figure 2 displays the estimated mean functions, including the overall mean function and visit specific mean functions, in the dense case and four different sparse cases. Even with N=3, the estimated mean functions are similar to those from the full analysis. By borrowing information from 3201 subject, the sparse MFPCA captures the trend of mean functions well and correctly identifies peaks and valleys, compared to those from the full analysis. When

the number of grid points increases to 6, 12 and 24, the estimated mean functions become indistinguishable from those from the full analysis.

[Table 2 about here.]

Table 2 displays the estimated eigenvalues, and Figure 3 shows the estimated eigenfunctions at the subject level (level 1) in the dense case and four different sparse cases. The dashed lines represent eigenfunctions estimated from the dense analysis, while solid lines correspond to estimated eigenfunctions from the sparse analysis. The first principal components were recovered very well in each case, even when $N = 3$. The shapes of the second and third components are roughly captured with 3 grid points, and get closer to those from full analysis as N increases. When $N = 24$, all three components agree well with their full analysis counterparts. For the visit level (level 2), similar results are observed and the details are reported in the supplementary file. Similar patterns can be observed for eigenvalues and percent variance explained. The surprisingly good performance on estimation mean functions and eigenvalues is due to the ability of sparse MFPCA to borrow information over all subjects and visits.

[Figure 2 about here.]

[Figure 3 about here.]

Next, we evaluate the performance on predictions in each of the four sparse scenarios, and illustrate the results on the first two subjects in Figure 4. The thick solid lines are predictions using sparse MFPCA methods on sparse data, while the thin solid lines are smooth estimates of the functions using the full data set. The dots are the actual sampled points. With $N = 3$ (first row of panels), predictions of curves can only capture the rough trend and miss the detailes features such as peaks and valleys, even though mean functions and eigenfunctions can be recovered well. This is not surprising, because a lot of subject-specific information

loss should be expected. When the number of sampled points increases (next rows of panels), the sparse MFPCA method estimates more detailed features.

[Figure 4 about here.]

The sparse MFPCA methods provides accurate estimates of the mean and principal components functions even when the sampled points are sparse, provided that there are many subjects. In terms of prediction for specific curves, the accuracy greatly depends on the amount of available information, or equivalently, the level of sparsity. One can expect to recover a rough trend with few grid points, but more observations are needed to estimate detailed features, if those features exist. In practice, it is often the case that only the sparse data are available, and the BLUPs from the MFPCA model provide the best linear predictions.

6. Discussion

We considered sparsely sampled multilevel functional data, and proposed a sparse MFPCA methodology for such data. We incorporated smoothing to deal with sparsity. Simulation studies show that our methods perform well. In an application to the SHHS, we compared the full analysis with sparse analysis under different levels of sparsity. The results show that the sparse MFPCA methodology works well in extracting the principal components, while prediction accuracy of functions depends on the level of sparsity. The sparse MFPCA methodology developed in this paper is generally applicable to many scientific studies that generate multilevel functional outcomes. We encounter this type of data sets more and more often in our scientific studies. Moreover, more researchers recognize or have the intuition that the data they collect are functional and are looking for way to extract it.

Other functional approaches for multilevel functions are available. For example, Morris et al. (2003) and Morris and Carroll (2006) proposed Bayesian hierarchical models based

on wavelets. However, Di et al. (2009) is the first attempt to generalize FPCA to multilevel functional data, and the current paper further extends its scope to sparsely sampled hierarchical functions. To the best of our knowledge this is the first functional approach to multilevel functional data analysis where functions are sparsely sampled.

Multilevel functional research is a rich research area motivated by an explosion of studies that generate functional data sets. The explosion is mainly due to improved technologies that generate new type of data sets. Here are two examples of interesting methodological developments: 1) extend the generalized multilevel functional regression Crainiceanu et al. (2009) to the sparse case; and 2) develop methods that are more efficient to estimating eigenvalues and eigenfunctions extending, for example, ideas in James et al. (2000) and Peng and Paul (2009) to the multilevel context.

Acknowledgements

Research was supported by Award Number R01NS060910 from the National Institute Of Neurological Disorders And Stroke. The content is solely the responsibility of the author and does not necessarily represent the official views of the National Institute Of Neurological Disorders And Stroke or the National Institutes of Health.

References

Crainiceanu, C. M., Caffo, B. S., Di, C.-Z., and Punjabi, N. M. (2009). Nonparametric signal extraction and measurement error in the analysis of electroencephalographic activity during sleep. *Journal of the American Statistical Association* **104,** 541–555.

Crainiceanu, C. M., Staicu, A. M., and Di, C.-Z. (2009). Generalized multilevel functional regression. *Journal of the American Statistical Association, to appear* .

Di, C.-Z., Crainiceanu, C. M., Caffo, B. S., and Punjabi, N. M. (2009). Multilevel functional principal component analysis. *Annals of Applied Statistics* **3,** 458–488.

Diggle, P. J., Heagerty, P., Liang, K.-Y., and Zeger, S. L. (2002). *Analysis of longitudinal data*, volume 25 of *Oxford Statistical Science Series*. Oxford University Press, Oxford, second edition.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications*. Monographs on Statistics and Applied Probability. Chapman & Hall, London.

Hall, P. and Hosseini-Nasab, M. (2006). On properties of functional principal components analysis. *J. Roy. Statist. Soc. Ser. B* **68,** 109–126.

James, G. M., Hastie, T. J., and Sugar, C. A. (2000). Principal component models for sparse functional data. *Biometrika* **87,** 587–602.

Karhunen, K. (1947). *Über lineare Methoden in der Wahrscheinlichkeitsrechnung*. Suomalainen Tiedeakatemia.

Loève, M. (1945). Fonctions aléatoires de second ordre. *Comptes Rendus Académie des Sciences* **220,** 469.

Morris, J. S. and Carroll, R. J. (2006). Wavelet-based functional mixed models. *J. Roy. Statist. Soc. Ser. B* **68,** 179–199.

Morris, J. S., Vannucci, M., Brown, P. J., and Carroll, R. J. (2003). Wavelet-based nonparametric modeling of hierarchical functions in colon carcinogenesis. *J. Amer. Statist. Assoc.* **98,** 573–584.

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

Peng, J. and 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, to appear* .

Quan, S. F., Howard, B. V., Iber, C., Kiley, J. P., Nieto, F. J., O'Connor, G. T., Rapoport, D. M., Redline, S., Robbins, J., Samet, J. M., and Wahl, P. W. (1997). The Sleep Heart

Health Study: design, rationale, and methods. *Sleep* **20,** 1077–1085.

Ramsay, J. O. and Dalzell, C. J. (1991). Some tools for functional data analysis. *J. Roy. Statist. Soc. Ser. B* **53,** 539–572.

Ramsay, J. O. and Silverman, B. W. (2005). *Functional Data Analysis.* Springer Series in Statistics. Springer Verlag, New York, second edition.

Rice, J. A. and Silverman, B. W. (1991). Estimating the mean and covariance structure nonparametrically when the data are curves. *J. Roy. Statist. Soc. Ser. B* **53,** 233–243.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003). *Semiparametric Regression.* Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge.

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

Yao, F. and Lee, T. (2006). Penalized spline models for functional principal component analysis. *Journal of the Royal Statistical Society Series B* **68,** 3–25.

Yao, F., Müller, H.-G., and Wang, J. L. (2005). Functional data analysis for sparse longitudinal data. *J. Amer. Statist. Assoc.* **100,** 577–591.

Zhao, X., Marron, J., and Wells, M. (2004). The functional data analysis view of longitudinal data. *Statistica Sinica* **14,** 789–808.

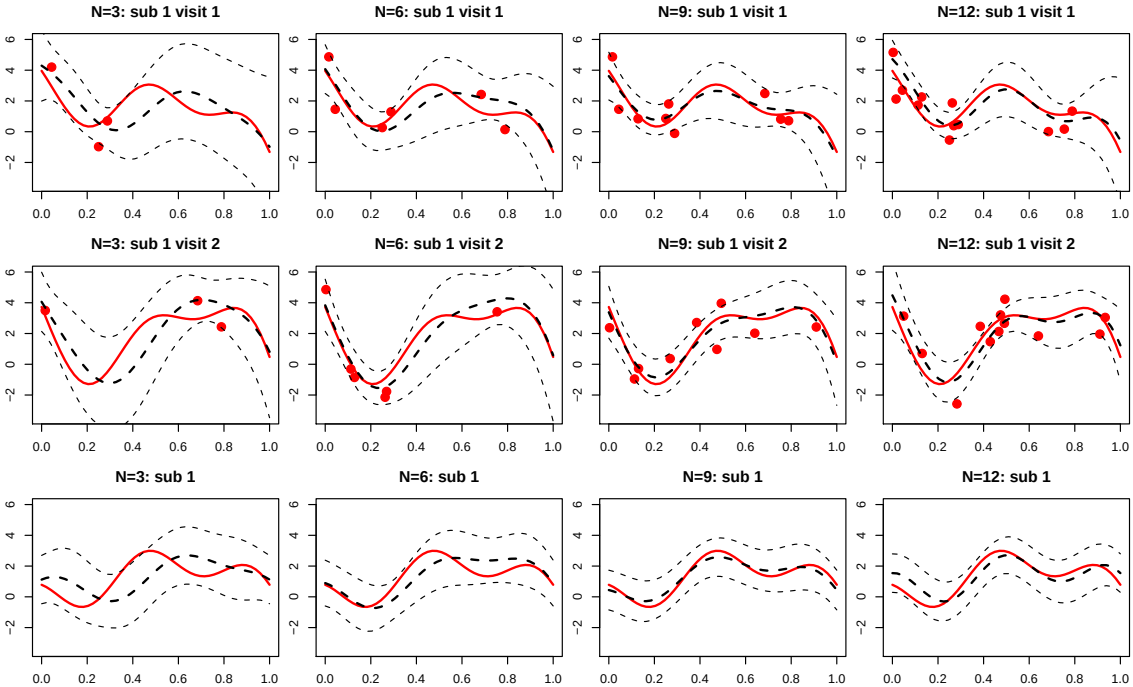

Figure 1. Prediction of curves for the first subject, in simulation setting with $n = 200$ subjects, $J = 2$ visits per subject, $N = 3, 6, 9, 12$ grid points per curve and noise variance $\sigma^2 = 1$. Different columns correspond to different levels of sparsity, with the number of grid points varying from 3 to 12. The first and second rows show predictions of subject/visit specific curves, at visit 1 and 2, respectively. In these subfigures, red solid lines correspond to the true underlying curves, $Y_{ij}(t)$, and red dots are observed sparse data, $Y_{ij}(t_{ijs})$. The thick black dashed lines are the predictions of curves, $\hat{Y}_{ij}(t)$, and thin black dashed lines give their 95% pointwise confidence bands. The third row display subject level curves, $Z_i(t)$, and their predictions $\hat{Z}_i(t)$.

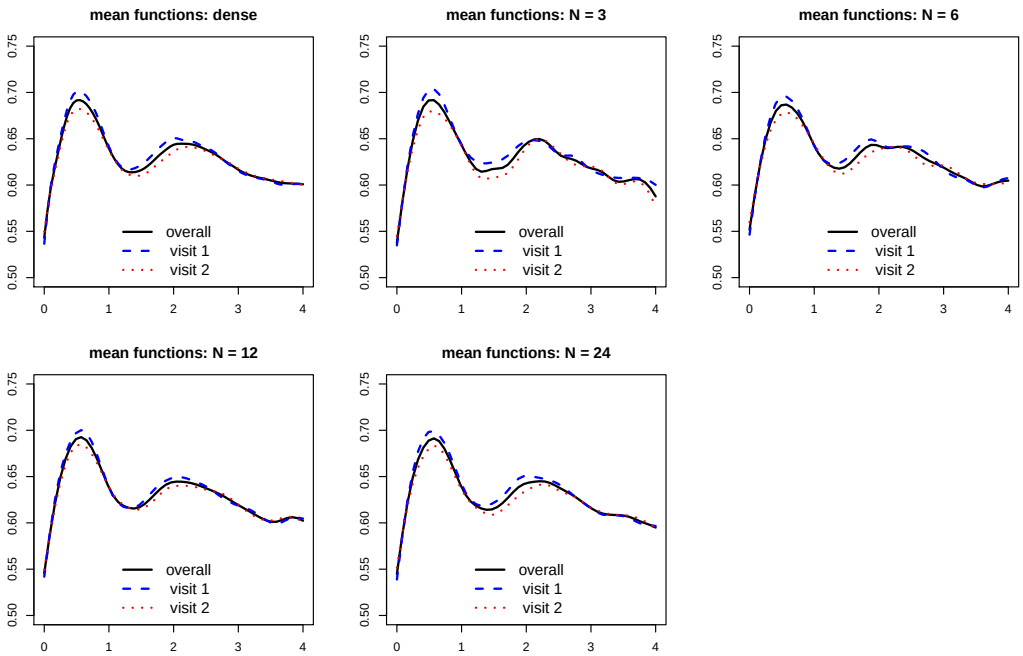

Figure 2. Estimated mean functions from MFPCA for the SHHS data: dense and sparse cases. The upper left panel shows estimated mean functions from the full analysis using dense data. The four remaining panels correspond to results from sparse cases, with the number of grid points per curve $N = 3, 6, 12, 24$, respectively. In each subfigure, solid lines correspond to overall mean functions, $\hat{\mu}(t)$, while dashed and dotted lines represent visit specific mean functions, $\hat{\mu}(t) + \hat{\eta}_j(t)$, at visit 1 and 2, respectively.

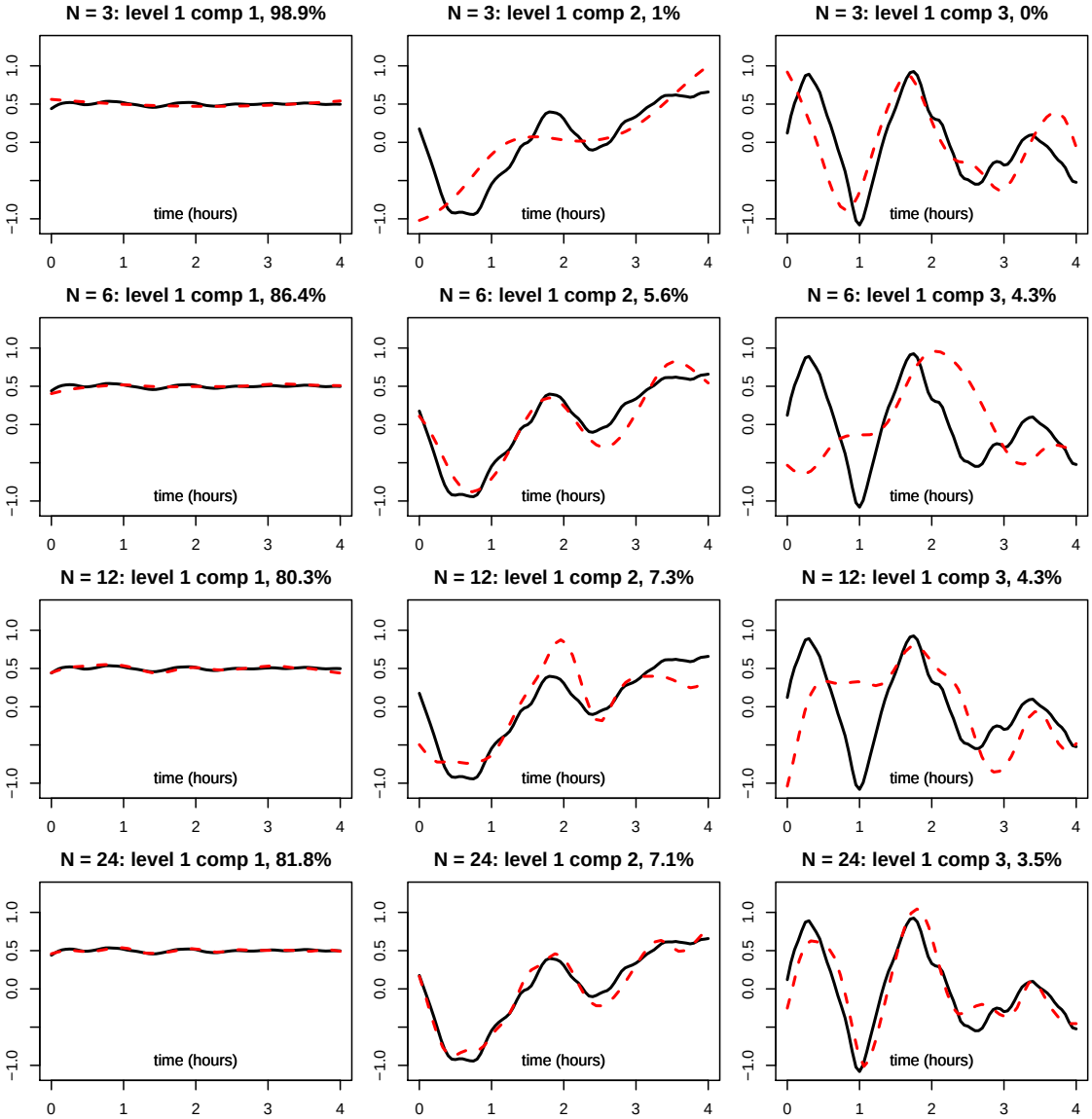

Figure 3. Estimated eigenfunctions (first three components at level 1) from MFPCA for the SHHS data: dense and sparse cases. The four rows correspond to different levels of sparsity, with the number of grid points per function $N = 3, 6, 12, 24$, respectively. The three columns represent the first three principal components, respectively. In each subfigure, solid black lines correspond to estimates from the dense case, while dashed red lines represent estimates from sparse cases.

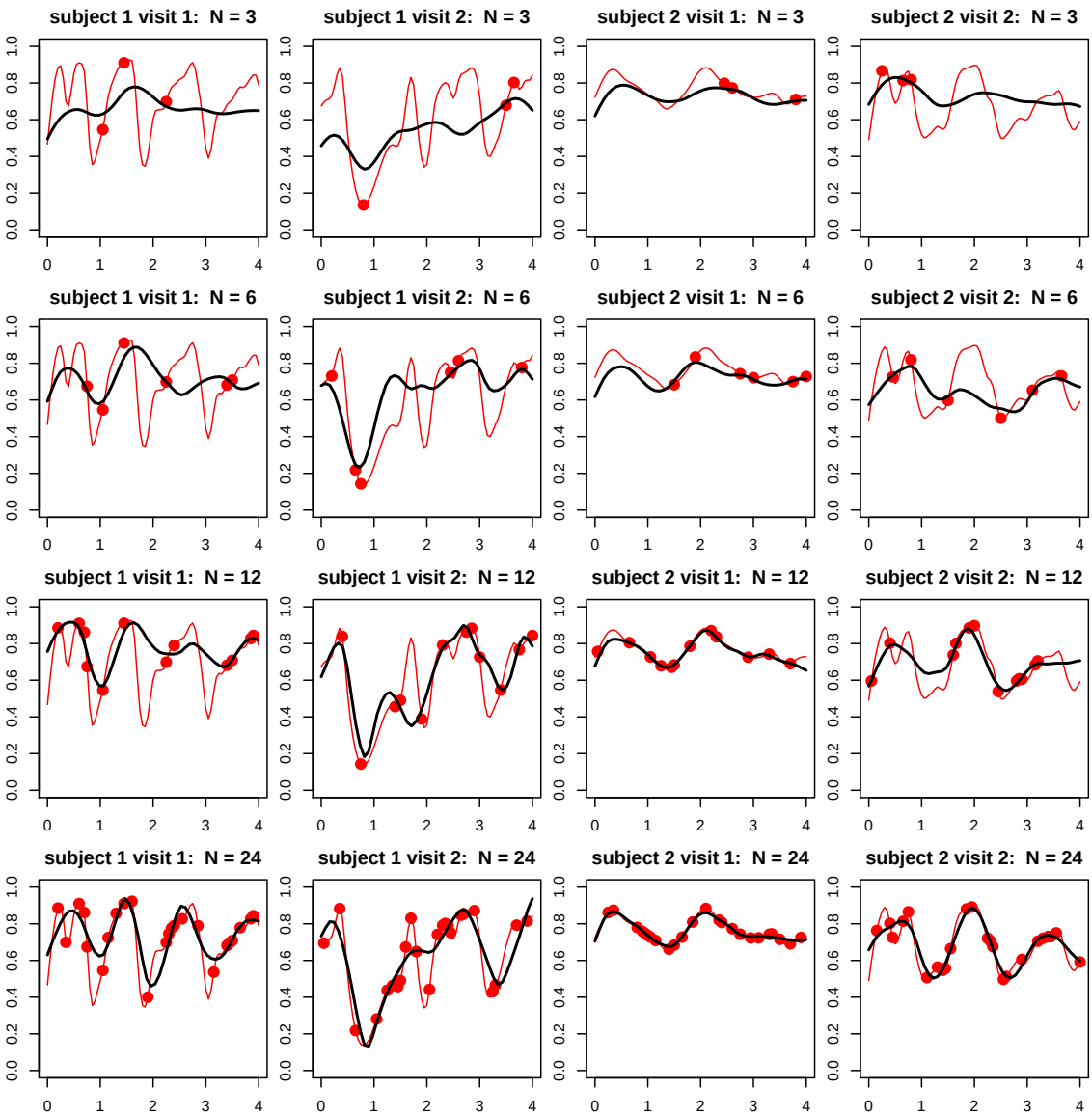

Figure 4. Prediction of sleep EEG curves for the first two subjects from the SHHS. The first and second columns correspond to visit 1 and 2 for the first subject, respectively. The third and fourth columns correspond to visit 1 and 2 for the second subject, respectively. Different rows represent different levels of sparsity, with the number of grid points N varying from 3 to 12. In each subfigure, red lines represent smoothed sleep EEG curves, while red dots are sparsified data at certain level of sparsity. Black lines are predictions of sleep EEG curves from the MFPCA model.

Table 1

Root (integrated) mean square errors for eigenvalues and eigenfunctions in simulations. Simulation settings vary according to sample size ($n = 100, 200, 300$) and the number of grid points per function ($N = 3, 6, 9, 12$). In simulations, the first four principal components (PC) at both levels were compared to their underlying true counterparts. Root mean square errors and root integrated mean square errors are used to measure estimation accuracy for eigenvalues and eigenfunctions, respectively.

n	N	Eigenvalues				Eigenfunctions			
		PC 1	PC 2	PC 3	PC 4	PC 1	PC 2	PC 3	PC 4
					Level 1				
100	3	0.25	0.39	0.69	1.16	0.45	0.66	1.03	1.07
100	6	0.29	0.36	0.76	1.26	0.56	0.81	1.00	1.21
100	9	0.19	0.25	0.35	0.48	0.38	0.54	0.83	0.98
100	12	0.21	0.26	0.36	0.54	0.42	0.66	0.85	1.08
200	3	0.18	0.22	0.26	0.36	0.34	0.48	0.73	0.92
200	6	0.19	0.23	0.30	0.41	0.35	0.56	0.76	0.97
300	3	0.17	0.20	0.23	0.31	0.32	0.46	0.66	0.87
					Level 2				
100	3	0.14	0.18	0.28	0.36	0.25	0.37	0.67	0.90
100	6	0.15	0.21	0.30	0.42	0.31	0.51	0.71	0.95
100	9	0.15	0.23	0.45	0.64	0.27	0.39	0.81	0.98
100	12	0.17	0.25	0.37	0.64	0.36	0.62	0.83	1.06
200	3	0.12	0.16	0.39	0.50	0.21	0.30	0.67	0.90
200	6	0.14	0.22	0.32	0.51	0.30	0.53	0.74	0.97
300	3	0.09	0.10	0.16	0.20	0.15	0.21	0.33	0.51

Table 2

Estimated eigenvalues (first three components at level 1 and first five components at level 2) for SHHS, from dense and four sparse cases. "Percent" means percentage of variation explained by the corresponding to the principal component, relative to the total variation at the corresponding level. "N" is the number of grid points per function, and reflects levels of sparsity.

		Level 1			Level 2				
		PC 1	PC 2	PC 3	PC 1	PC 2	PC 3	PC 4	PC 5
dense	eigenvalue	1.30	0.10	0.10	1.30	0.80	0.70	0.60	0.60
	percent	80.80	7.60	3.30	21.80	12.80	12.50	10.90	9.60
$N = 3$	eigenvalue	1.20	0.00	0.00	1.30	0.70	0.50	0.50	0.40
	percent	98.90	1.10	0.00	34.00	18.50	14.40	13.20	10.30
$N = 6$	eigenvalue	1.30	0.10	0.10	1.30	0.80	0.60	0.60	0.40
	percent	86.40	5.60	4.30	27.40	15.60	13.50	11.80	9.10
$N = 12$	eigenvalue	1.30	0.10	0.10	1.30	0.80	0.70	0.60	0.50
	percent	80.30	7.30	4.30	24.60	14.70	13.80	11.50	9.40
$N = 24$	eigenvalue	1.30	0.10	0.10	1.30	0.80	0.70	0.70	0.50
	percent	81.80	7.10	3.50	23.60	13.80	13.40	12.10	9.70