HHS Public Access

Author manuscript

Comput Stat Data Anal. Author manuscript; available in PMC 2017 January 01.

Published in final edited form as:

Comput Stat Data Anal. 2016 January 1; 93: 86–96. doi:10.1016/j.csda.2014.11.017.

Wavelet-Based Scalar-on-Function Finite Mixture Regression Models

Adam Ciarleglio[a,*] and **R. Todd Ogden**[b]

[a]Department of Child and Adolescent Psychiatry, Division of Biostatistics, New York University, United States

[b]Department of Biostatistics, Columbia University Mailman School of Public Health, United States

Abstract

Classical finite mixture regression is useful for modeling the relationship between scalar predictors and scalar responses arising from subpopulations defined by the di ering associations between those predictors and responses. The classical finite mixture regression model is extended to incorporate functional predictors by taking a wavelet-based approach in which both the functional predictors and the component-specific coefficient functions are represented in terms of an appropriate wavelet basis. By using the wavelet representation of the model, the coefficients corresponding to the functional covariates become the predictors. In this setting, there are typically many more predictors than observations. Hence a lasso-type penalization is employed to simultaneously perform feature selection and estimation. Specification of the model is discussed and a fitting algorithm is provided. The wavelet-based approach is evaluated on synthetic data as well as applied to a real data set from a study of the relationship between cognitive ability and di usion tensor imaging measures in subjects with multiple sclerosis.[1]

Keywords

EM algorithm; Functional data analysis; Lasso; Wavelets

1. Introduction

Let $Y_i \in \mathbb{R}$ be the scalar response of interest for observation i, $i = 1, \ldots, n$ and let X_i be a random predictor process that is square integrable on a compact support $I \subset \mathbb{R}$ (i.e,

[1]Supplementary materials, including R code for implementing the proposed method and conducting analyses, as well as additional simulation results can be found in the electronic version of this paper.

[*]Corresponding author Telephone: +1 646 754 5463, Adam.Ciarleglio@nyumc.org (Adam Ciarleglio).

Supplementary Materials

Appendix: Additional results from simulations discussed in the article as well as two additional simulations. (Appendix.pdf)

R Code: A set of files consisting of a "README" file and R code for implementing the WBFMR method and for replicating the analysis of the DTI data. Code for constructing the true component coefficient functions used in the simulations is also available.

$\int_I X_i^2(t)\,dt<\infty$). It is increasingly common to model the relationship between Y_i and X_i via a functional linear model (FLM) given by:

$$Y_i=\alpha+\int_I X_i(t)\,\omega(t)\,dt+\varepsilon_i,\quad i=1,\ldots,n, \quad (1.1)$$

where α is the scalar intercept and ε_i is the error term such that $\varepsilon_i \sim N(0, \sigma^2)$. ω is a square integrable coefficient function that relates the predictor process to the response. The magnitude of $\omega(t)$ indicates the relative importance of the predictor X_i at a given value of t. If $|\omega(t_0)|$ is large, this means that changes in the predictor process at t_0 are important in predicting the response. A variety of approaches have been developed for estimating the coefficient function in (1.1) (James, 2002; Cardot et al., 2003; Cardot and Sarda, 2005; Ramsay and Silverman, 2005; James and Silverman, 2005; Cai and Hall, 2006; Reiss and Ogden, 2007; Müller and Yao, 2008; Zhao et al., 2012).

Although (1.1) is adequate for modeling the relationship between a scalar response and a functional predictor when the association between the response and predictor is the same for all observations, it is not appropriate for settings in which the coefficient function di ers across subgroups of the observations. If there are C different associations corresponding to C different coefficient functions then we can think of each observation as coming from one of C distinct subpopulations/components and would need C distinct FLMs to adequately describe the relationship between the response and the predictor. We are concerned with settings in which subpopulation membership is not observed and will need to be estimated along with the component-specific coefficient functions. In order to appropriately model the relationship between X_i and Y_i in this context, we combine finite mixture and functional linear modeling strategies.

Although the underlying theory of finite mixture regression models and methods for estimating those models have been well-studied when the predictors are scalars (McLachlan and Peel, 2000; Schlattmann, 2009), methods for finite mixture regression remain relatively undeveloped when the predictors are functions. To our knowledge, Yao et al. (2011) are the only ones to investigate such an extension. In their approach to functional mixture regression (FMR) models, they first represent each functional predictor in terms of some suitably chosen number of functional principal components and apply standard mixture regression techniques in the new coordinate space.

The FMR model is given by

$$Y_i=\alpha_r+\int_I X_i(t)\,\omega_r(t)\,dt+\varepsilon_i \quad \text{if subject } i \text{ belongs to the } rth \text{ group}, \quad (1.2)$$

where C is the number of components or distinct subpopulations, α_r is the rth component-specific intercept, and ω_r is the regression function for the rth group, $r = 1, \ldots, C$.

In contrast to Yao et al. (2011), we propose to take a wavelet-based approach to FMR models. Our approach is distinct from that of Yao et al. (2011) in several ways. First, just as with using functional principal components, using a wavelet basis initially provides no dimension reduction. However, wavelets are useful for providing sparse representations of

functions so that most of the information about the function is contained in relatively few wavelet coefficients. The method that we use to achieve dimension reduction relies on this sparsity property and is very different from simply choosing a small number of principal components that explain some specified proportion of the variance in the predictors. We take a fully functional approach which performs dimension reduction and model estimation simultaneously so that dimension reduction is driven by the relationship between the functional predictor and the response of interest. This is not the case with the functional principal components-based approach. Furthermore, the approach that we present here is flexible in that we can choose from a host of wavelet families to represent the functional predictors and component coefficient functions. Our approach comes with the added cost of needing to select more tuning parameters, but we propose methods for tuning parameter selection that perform well in simulations and we show that we can make substantial gains in estimation accuracy over using functional principal components when the true component coefficient functions are characterized by local features. Finally, it is computationally trivial to extend our approach to higher-dimensional predictors such as 2- or 3-dimensional images. This is due to the fact that there are several software packages available for performing wavelet decompositions for 2- or 3-dimensional data. To our knowledge, there is no readily available software for performing FPCA for such data.

The rest of the paper is organized as follows. In Section 2, we provide a brief discussion of wavelets and the wavelet-based functional linear model followed by specification of the wavelet-based (WB) functional finite mixture regression model. In Section 3, we outline an algorithm for fitting the WB model. Section 4 discusses the various tuning parameters in the WB model. Section 5 presents simulation results showing the performance of the WB method and an application of our method to a real data set. We conclude with a brief discussion in Section 6.

2. Methodology

2.1. Wavelets, Wavelet Decomposition, and Wavelet Representation of the FLM

We focus here on the wavelet basis for several reasons. Wavelets are particularly well suited to handle many types of functional data, especially functional data that contain features on multiple scales. They have the ability to adequately represent global and local attributes of functions and can handle discontinuities and rapid changes. Furthermore a large class of functions can be well represented by a wavelet expansion with relatively few non-zero coefficients. This is a desirable property from a computational point of view as it aids in achieving the goal of dimension reduction.

In $L^2(\mathbb{R})$, a wavelet basis is generated by two kinds of functions: a father wavelet, $\phi(t)$, and a mother wavelet, $\psi(t)$, satisfying $\int \phi(t)\,dt=1$ and $\int \psi(t)\,dt=0$. Here we restrict ourselves to orthonormal wavelet basis families (Daubechies, 1988).

Any particular wavelet basis consists of translated and dilated versions of its father and mother wavelets given by $\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k)$ and $\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k)$ where the integer j is the dilation index referring to the scale and k is an integer that serves as a translation index. These functions can be adapted via implementation of one of several boundary

handling schemes that represent a given function on a specified interval. Without loss of generality, we take that interval to be [0,1]. Hence if we assume that $\omega \in L^2([0, 1])$ then we can represent ω in the wavelet domain by

$$\omega(t) = \sum_{k=0}^{2^{j_0}-1} \beta'_{j_0,k}\phi_{j_0,k}(t) + \sum_{j=j_0}^{\infty}\sum_{k=0}^{2^j-1} \beta_{j,k}\psi_{j,k}(t), \quad (2.1)$$

where j_0 is an integer that determines the number of scaling functions used in the lowest scale representation. The values $\beta'_{j_0,k}$ and $\beta_{j,k}$ are the corresponding scaling and wavelet coefficients for the functions $\phi_{j0,k}$ and $\psi_{j,k}$ respectively and are given by $\beta'_{j_0,k} = \int \omega(t)\,\phi_{j_0,k}(t)\,dt$ and $\beta_{j,k} = \int \omega(t)\,\psi_{j,k}(t)\,dt$.

In practice, the functional predictors are discretely sampled. We assume that we observe a dyadic length ($N = 2^J$) vector of function values $X_i = (X_i(t_1), \ldots, X_i(t_N))^T$ where the arguments $t_1, \ldots, t_N$, are equally spaced and the same for all observations. If the observed functional data are not dyadic and/or not regularly spaced then there are several procedures that one may employ to obtain dyadic and regularly spaced predictors from the original data. One commonly used option is interpolation. To obtain the wavelet and scaling coefficients corresponding to the functional predictors we use the discrete wavelet transform (DWT). The inverse DWT (IDWT) can be used to reconstruct a vector of functional observations from its corresponding wavelet and scaling coefficients. Both the DWT and IDWT can be performed using a computationally fast pyramid algorithm (Mallat, 1989). Comprehensive treatment of wavelets and their applications in statistics can be found in Ogden (1997); Nason (2008); and Vidakovic (1999).

Before moving on to our WB model we first make note of an important consequence of using a wavelet-space representation. Note that $\omega(t)$ can be expressed as in (2.1) and $X_i(t)$ can be expressed as

$$X_i(t) = \sum_{k=0}^{2^{j_0}-1} z'_{i,j_0,k}\phi_{j_0,k}(t) + \sum_{j=j_0}^{\infty}\sum_{k=0}^{2^j-1} z_{i,j,k}\psi_{j,k}(t),$$

where the scaling and wavelet coefficients are given respectively by $z'_{i,j_0,k} = \int X_i(t)\,\phi_{j_0,k}(t)\,dt$ and $z_{i,j,k} = \int X_i(t)\,\psi_{j,k}(t)\,dt$.

Applying the DWT to the N equally-spaced observations of X_i produces wavelet and scaling coefficients that can be put into an $(N + 1) \times 1$ vector denoted by Z_i:

$$Z_i = \left(1, z'_{i,j_0,0}, \ldots, z'_{i,j_0,k_{j_0}}, z_{i,j_0,0}, \ldots, z_{i,j_0,k_{j_0}}, \ldots, z_{i,J,0}, \ldots, z_{i,J,k_J}\right), \quad (2.2)$$

where $J = \log_2(N) - 1$, $k_j = 2^j - 1$, and the first element, 1, corresponds to the intercept.

Because of the orthonormality of the wavelet basis, (1.1) can be simply written as

$$Y_i=\alpha+\sum_{k=0}^{2^{j_0}-1} z'_{i,j_0,k}\beta'_{j_0,k}+\sum_{j=j_0}^{J}\sum_{k=0}^{2^j-1} z_{i,j,k}\beta_{j,k}+\varepsilon_i, \quad i=1,\ldots,n,$$

(Zhao et al., 2012), or, in matrix notation the linear model is given by

$$Y=Z\beta+\varepsilon, \quad (2.3)$$

where $Y = (Y_1, \ldots, Y_n)^T$, Z is an $n \times (N + 1)$ matrix with ith row Z_i, β is an $(N + 1) \times 1$ vector containing the intercept followed by the coefficients arranged in the same order as the vector Z_i, and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^T$. The fact that a wavelet-space representation results in a linear model has often been exploited in the context of functional regression (Vannucci et al., 2005; Amato et al., 2006; Malloy et al., 2010; Zhao et al., 2012; Reiss et al., 2013).

2.2. Specification of the WB Functional Mixture Regression Model

If the pairs of functional predictors and scalar responses come from a heterogeneous population, where the subpopulations (or components) are determined by C distinct associations between the predictors and the response, then there is a unique coefficient function, ω_r, $r = 1, \ldots, C$ corresponding to each subpopulation. Since, as noted above, the predictors are typically discretely sampled at N points, the model we consider is

$$Y_i=\alpha_r+\sum_{k=0}^{2^{j_0}-1} z'_{i,j_0,k}\beta'_{r,j_0,k}+\sum_{j=j_0}^{J}\sum_{k=0}^{2^j-1} z_{i,j,k}\beta_{r,j,k}+\varepsilon_i,$$

where $\varepsilon_i \sim N\left(0, \sigma_r^2\right)$ given that observation i belongs to component r and σ_r^2 is the error variance in the rth component. (In the appendix, we investigate similar models where the error terms are not normally distributed.) Thus, the coefficient functions of interest are given by

$$\omega_r(t)=\sum_{k=0}^{2^{j_0}-1}\beta'_{r,j_0,k}\phi_{j_0,k}(t)+\sum_{j=j_0}^{J}\sum_{k=0}^{2^j-1}\beta_{r,j,k}\psi_{j,k}(t),$$

and our goal is to find estimates for the $\beta'_{r,j_0,k}$'s and the $\beta_{r,j,k}$'s.

In this setting, the model of interest is similar to that seen in classical finite mixture regression. We have that $Y_i|Z_i$ for $i = 1, \ldots, n$ are independent and

$$Y_i|Z_i=z \sim \sum_{r=1}^{C}\pi_r\frac{1}{\sqrt{2\pi}\sigma_r}exp\left(-\frac{(y-z\beta_r)^2}{2\sigma_r^2}\right) \quad \text{for} \quad i=1,\ldots,n, \quad (2.4)$$

where β_r is the component-specific coefficient vector for component r with the same form as β in the model given by (2.3) and π_r is the probability that observation i belongs to component r. Let $\xi=\left(\beta_1,\ldots,\beta_C,\sigma_1,\ldots,\sigma_C,\pi_1,\ldots,\pi_{C-1}\right) \in \mathbb{R}^{C(N+1)} \times \mathbb{R}_+^C \times \Pi$ be the

$((N + 3) \cdot C - 1) \times 1$ vector of free parameters to be estimated from (2.4), where is the space of vectors of the form $(\pi_1, \ldots, \pi_{C-1})$ such that $\pi_r > 0$ for $r = 1, \ldots, C - 1$, $\sum_{r=1}^{C-1} \pi_r < 1$, and $\pi_C = 1 - \sum_{r=1}^{C-1} \pi_r$.

In practice, X_i may be densely sampled and so we may have $N >> n$. In this case, maximum likelihood estimation will provide inaccurate and unstable estimates for each β_r and consequently poor estimates for each ω_r. Since wavelets allow for sparse representation of each ω_r, most of the information about the coefficient functions to be estimated will be contained in a relatively small number of wavelet coefficients. Those wavelet coefficients corresponding to unimportant features in the wavelet space will comprise most of the elements of β_r and will be of negligible magnitude. Thus we consider a lasso-type procedure for estimating the C component-specific vectors of wavelet and scaling coefficient values.

Städler et al. (2010) proposed an ℓ_1-penalized mixture regression procedure for model fitting with general high-dimensional predictors. We make use of this procedure here. We begin by first reparameterizing model (2.4) using the following:

$$\varphi_r = \beta_r / \sigma_r, \quad \rho_r = \sigma_r^{-1}, \quad r = 1, \ldots, C.$$

Based on this new parameterization, model (2.4) can be written as:

$$Y_i | Z_i = z \sim \sum_{r=1}^{C} \pi_r \frac{\rho_r}{\sqrt{2\pi}} exp\left(-\frac{1}{2}(\rho_r y - z\varphi_r)^2\right) \text{ for } \quad i = 1, \ldots, n. \quad (2.5)$$

There is a one-to-one mapping from ξ to a new parameter vector

$$\theta = (\varphi_1, \ldots, \varphi_C, \rho_1, \ldots, \rho_C, \pi_1, \ldots, \pi_{C-1}) \in \mathbb{R}^{C(N+1)} \times \mathbb{R}_+^C \times \Pi.$$

The corresponding log-likelihood for model (2.5) is

$$\ell(\theta; Y) = \sum_{i=1}^{n} log\left(\sum_{r=1}^{C} \pi_r \frac{\rho_r}{\sqrt{2\pi}} exp\left(-\frac{1}{2}(\rho_r Y_i - Z_i \varphi_r)^2\right)\right). \quad (2.6)$$

To estimate the parameter vector θ in model (2.5), we propose to use $\hat{\theta}_\lambda \in \mathbb{R}^{C(N+1)} \times \mathbb{R}_+^C \times \Pi$ that minimizes

$$-n^{-1}\ell_\lambda(\theta) = -n^{-1}\ell(\theta; Y) + \lambda \sum_{r=1}^{C} \pi_r ||\varphi_r||_1, \quad (2.7)$$

where $\|\phi_r\|_1$ is the ℓ_1-norm of the vector ϕ_r. Note that the penalty on each wavelet and scaling component coefficient vector ϕ_r is proportional to the mixing probability π_r. Including the mixing proportion in this manner corresponds to the common practice of relating the amount of penalty to the sample size, where, in the context of mixture

regression, Khalili and Chen (2007) note that the "virtual" sample size from the rth component is proportional to π_r. Further discussion of the tuning parameters is given in Section 4.

Estimation of ϕ_r and ρ_r rather than the direct estimation of β_r and σ_r is considered primarily for two reasons. The reparametrization, along with a lasso-type penalty allows for penalization of both the coefficient vectors of interest and the error variances within each component (Städler et al., 2010) while maintaining convexity of the optimization problem to be solved.

The identifiability of finite mixture models is an important consideration. In short, when no two distinct sets of parameter values (up to component label switching) define the same mixture distribution, the model is identifiable. As noted in Khalili and Chen (2007), identifiability depends on the component densities, the maximum possible number of components, and the design matrix. Hennig (2000) discusses conditions under which finite mixture models are identifiable. In what follows, we assume that the model under consideration is identifiable.

3. Fitting WB Functional Mixture Models

WB functional mixture regression models can be fit in three main steps outlined below.

Step 1

Use the DWT to decompose the functional predictors and obtain the corresponding wavelet and scaling coefficients for each predictor. Here we must choose the wavelet family (e.g., Daubechies' least asymmetric wavelets), number of vanishing moments, lowest level of decomposition ($j_0 \in \{0, \ldots, \log_2(N) - 1\}$), and method for handling the boundaries (e.g., symmetric boundary handling).

The empirical wavelet and scaling coefficients for each predictor curve can be arranged into $(N + 1) \times 1$ vectors, denoted Z_i, $i = 1, \ldots, n$, which have the same structure as (2.2). We then form Z, an $n \times (N + 1)$ matrix with ith row Z_i.

Step 2

We carry out an EM-type algorithm for our setting in a manner similar to that described in Städler et al. (2010). Details of this step are provided in the Appendix.

Step 3

Use the IDWT to obtain estimates $\hat{\omega}_1, \ldots, \hat{\omega}_C$ from the estimates $\hat{\sigma}_1\hat{\varphi}_1, \ldots, \hat{\sigma}_1\hat{\varphi}_C$ respectively.

The EM procedure discussed in **Step 2** above requires that we provide initial values for the parameters being estimated. We use the following scheme for obtaining these initial values. We first assign a weight to each observation corresponding to each of the C distinct components. To do this, we randomly assign to each observation i a class, κ, from the set $\{1, \ldots, C\}$. For observation i and its randomly selected class κ we assign $\tilde{\Delta}_{i,\kappa}=0.9$ and for each

of the other classes we assign $\tilde{\Delta}_{i,r}=0.1, \quad r \in \{1,\ldots,C\}/\kappa$. We then normalize the vector of $\tilde{\Delta}_{i,r}$ values, $r = 1, \ldots, C$ to sum to 1. Note that this process can be thought of as an initialization of the E-step. This is followed by updating all of the coordinates involved in the optimizations in the M-step with initial values of $\varphi_{r,q}^{(0)}=0$, $\rho_r^{(0)}=2$, and $\pi_r^{(0)}=1/C$, $r = 1, \ldots, C$, $q = 1, \ldots, N + 1$. Convergence properties of the algorithm implemented in **Step 2**. are discussed in detail in Städler et al. (2010).

To speed up the EM procedure, we restrict ourselves to updating only the non-zero coordinates (active set elements) for 10 out of every 11 iterations of **Step 2**. This type of active set algorithm is used in Meier et al. (2008); Friedman et al. (2010); and Städler et al. (2010). After 10 iterations on the active set, we expand to consider all coordinates, both the active and non-active, for updating in the 11th iteration. We obtain a possibly new active set and continue in this manner until the convergence criteria are satisfied.

4. Tuning Parameters and Their Selection

If the number of components is known *a priori* or exploratory data analysis suggests a particular number of components, then C can be specified outright. However, we often employ the mixture modeling approach when the number of components is unknown or knowledge of component membership is unavailable.

The value of j_0 corresponds to the lowest level of decomposition and can range from 0 to $\log_2(N) - 1$. Since the predictors are sampled at N points, the DWT provides a decomposition that uses a total of N wavelet and scaling functions. Among this set of N basis functions, 2^{j0} will be scaling functions and $N - 2^{j0}$ will be wavelet functions. Hence, setting j_0 close or equal to 0 results in using fewer scaling functions to represent large-scale features and more wavelet functions to represent local details of the function of interest. Conversely, setting j_0 close or equal to $\log_2(N) - 1$ results in using more scaling functions.

The value of λ directly determines the role that the penalty function will have in both estimating and selecting variables in the model. Large values of λ force elements of the estimated component coefficient vectors to zero while small values result in many non-zero estimates.

We will employ two methods for tuning parameter selection. First we consider selecting the parameters that minimize the cross-validated value

$$-2\ell\left(\hat{\theta}_{j_0,\lambda,C};Y\right). \quad (4.1)$$

Here, $\ell(\cdot;\cdot)$ denotes the log-likelihood from (2.6) and the estimate $\hat{\theta}_{j_0,\lambda,C}$ depends on the values of the tuning parameters as indexed by the subscripts. We will refer to (4.1) as the "predictive loss". We also consider selecting the parameters that minimize a modified BIC criterion. We use the modified BIC measure, proposed by Pan and Shen (2007), which is given by

$$BIC = -2\ell\left(\hat{\theta}_{j_0,\lambda,C}; Y\right) + log\,(n)\,d_e, \quad (4.2)$$

where $d_e = (N + 3) \cdot C - 1 - q_0$ is the e ective number of parameters with q_0 being the number of coefficients estimated to be zero in all of the components.

The cross-validation procedure generally puts more emphasis on predictive ability and chooses a model that performs well in this regard. On the other hand, BIC focuses more on finding the "true" model and often chooses a simpler one. In the numerical investigations discussed below, we found that the cross-validation procedure is sometimes prone to over-fitting while use of BIC tends to avoid this issue. Compared to BIC, cross-validation is computationally demanding and can be prohibitive for large and/or high-dimensional data. Based on these characteristics, we are inclined to recommend the use of the modified BIC when selecting tuning parameters. Further discussion of these two approaches for tuning parameter selection is provided in Section 5.2 and in the appendix.

5. Simulations and Application

We present simulation results that demonstrate various aspects of the WBFMR procedure and that draw comparisons to a functional principal components-based (FPC) method similar to that proposed by Yao et al. (2011). For each simulation discussed below, we generated observations consisting of a discretely sampled one-dimensional functional predictor signal, X_i, and a scalar response, Y_i whose association with X_i depends on some known group membership.

Each functional predictor is a Brownian bridge stochastic process for $t \in (0, 1)$ with an expected value of 0, covariance given by $cov(X_i(t), X_i(s)) = s(1-t)$ for $s < t$, and with $X_i(0) = X_i(1) = 0$. We consider various sampling densities for the functional predictors. Specifically, we consider data sets where the functional predictors are sampled at $N = 64$, 128, 256, or 512 equally-spaced points. A sample of three of these predictors is given in the left panel of Figure 1.

The scalar outcomes corresponding to each functional predictor were generated using two distinct settings for the component-specific coefficient functions. The first pair of component-specific coefficient functions are given by $\omega_{s1}(t) = -sin(2\pi t)$ for the first component and $\omega_{s2}(t) = sin(\pi t)$ for the second component. The second pair of component-specific coefficient functions are given by $\omega_{b1}(t) = -3.257e^{-a(t-0.15)^2} + 4.886e^{-a(t-0.25)^2} - 3.257e^{-a(t-0.5)^2} + 2.606e^{-a(t-0.9)^2}$ for the first component and $\omega_{b2}(t) = 3.257e^{-a(t-0.1)^2} - 4.886e^{-a(t-0.35)^2} + 3.257e^{-a(t-0.7)^2}$ for the second component where $a = 20000/9$. The middle panel of Figure 1 shows ω_{s1} and ω_{s2} which we will refer to as the "smooth" functions while the right panel shows ω_{b1} and ω_{b2} which we will refer to as the "bumpy" functions.

Equal proportions of observations were generated in each component. In the first component, random error terms were drawn from $N\left(0, \sigma_1^2\right)$ and in the second component they were drawn from $N\left(0, \sigma_2^2\right)$ where $\sigma_1 = \sigma_2$. Different values for σ_1 and σ_2

corresponding to R^2 values of 0.9, 0.7, and 0.5 were used. Here R^2 is the the discrete approximation to $\sum_{r=1}^{2}\pi_r var\left[\int X(t)\,\omega_r(t)\,dt\right] / \sum_{r=1}^{2}\pi_r\left\{var\left[\int X(t)\,\omega_r(t)\,dt\right]+\sigma_r^2\right\}$, which measures the proportion of variation in the response attributed to the functional predictor.

Daubechies' least asymmetric wavelets with eight vanishing moments were used in all simulations, as we found these to provide a good balance between smoothness and compact support in prior numerical investigations (not shown here). The `WaveThresh` package in `R` (Nason, 1998) was used to perform the DWT and IDWT with the periodic boundary handling option. Additional results from the simulations discussed below, as well as results from additional simulations are available in the online appendix.

5.1. Simulation 1: Comparison of FMR Methods

In the first set of simulations we compare the wavelet-based (WB) and functional principal components-based (FPC) mixture regression methods. In all, we consider 24 different settings: two types of component coefficient functions (smooth or bumpy), three possible R^2 values (0.9, 0.7, or 0.5), and four possible sampling densities (N = 64, 128, 256, or 512).

For a given simulation run, we generate a training set, a validation set, and a test set all from the same setting. Each set is made up of 100 observation pairs consisting of a functional predictor and its corresponding scalar response. The training set is used to fit a model for each combination of the tuning parameters. The validation set is then used to select the combination of tuning parameters that minimizes (4.1) among all combinations of tuning parameters. Finally, the model chosen based on the validation set is applied to the test set and the corresponding predictive loss is computed. We repeat this procedure 100 times for each setting.

Here we treat the number of components as known (i.e., C = 2). For the wavelet-based method, we fix the lowest level of decomposition to $j_0 = 0$ in the smooth setting and to $j_0 = 5$ in the bumpy setting. In extensive prior simulations (not shown here), these decomposition levels tended to consistently minimize the predictive loss for each of the settings that we consider. The optimal value of λ is chosen from a grid of 100 candidate values. The grids for each setting were also selected based on prior simulations. (In these simulations, we plotted cross-validated log-likelihood loss values against λ values from very wide and very fine grids for several data sets from each of the generative models to see which λ values resulted in the minimum log-likelihood loss for each data set. For a given generative model, we then constructed the grid of 100 λ values wide enough to span those lambda values that gave the minimum log-likelihood losses such that we were confident that the selected λs would fall well within the lowest and highest grid values.)

For the functional principal components-based procedure, based on the procedure proposed in Yao et al. (2011), the tuning parameters consisted of the number of order four B-spline basis functions used in representing the predictor signals and the number of principal components to serve as the predictors in the FMR model. The optimal set of tuning parameters was selected by first fitting a model for each combination of the number of B-

spline basis functions and the number of principal components using the training data and then picking the pair that minimized (4.1) in the corresponding validation set. The fitted model was then applied to the corresponding test data and the predictive loss was obtained. We use the FlexMix packge (Leisch, 2004) in R to fit the functional principal components-based models.

We first consider how the three methods compare with respect to predictive loss based on the test sets. Table 1 shows the average predictive loss and standard deviation from the test sets in the 100 simulation runs. Lower loss values are preferred. In the smooth setting, we note that the wavelet-based method performs comparably to the functional principal components-based method while in the bumpy setting, the wavelet-based method appears to do better, especially for higher values of R^2.

Average estimation performance is illustrated in Figures 2 and 3. The solid and dashed thick curves correspond to the point-wise mean estimated component coefficient functions over the 100 simulation runs at the specified setting. The solid and dashed thin curves correspond to the true component coefficient functions used to generate the scalar responses. (Performance for $R^2 = 0.5$ is not shown.) In settings for which the true component coefficient functions are smooth, we note that the functional principal components-based method appears to do best while the wavelet-based method performs similarly well when the functional predictors are densely sampled. Substantial gains in estimation performance by the wavelet-based method are evident in the bumpy settings. We note that the wavelet-based method does very well in capturing the local features of the component coefficient functions and in estimating regions where there is no association between the functional predictor and the response while the functional principal components-based method struggles with both of these tasks. Additional information regarding Simulation 1, including how we handled label-switching, is available in the online appendix.

5.2. Simulation 2: Tuning Parameter Selection Methods

In the second set of simulations, we investigate selection methods for the tuning parameters in the wavelet-based model. We compare selection based on minimizing the 5-fold cross-validated log-likelihood loss to that based on minimizing the modified BIC criteria given in (4.2). We consider three different scenarios for tuning parameter selection:

Scenario 1. Set $C = 2$ and $j_0 = 0$ (smooth setting) or 5 (bumpy setting); select λ.

Scenario 2. Set $j_0 = 0$ (smooth setting) or 5 (bumpy setting); select $C \in \{1, 2, 3\}$ and λ.

Scenario 3. Set $C = 2$; select $j_0 \in \{0, \ldots, \log_2(N) - 1\}$ and λ.

We restrict ourselves to a subset of four of the 24 settings from the first group of simulations discussed above. Specifically, we compare the three tuning parameter selection scenarios in the smooth and bumpy settings when the sampling density of the functional predictors is either 128 or 256. In all four settings we have $R^2 = 0.9$. For each of 100 simulation runs at each setting, the training set was used to determine the optimal tuning parameters that either minimized the 5-fold cross-validated predictive log-likelihood loss or minimized the modified BIC criteria. The corresponding test set was used to estimate the test loss in each scenario for both selection methods.

Table 2 shows the mean and standard deviation of the log-likelihood loss values from the test sets in each setting and for each scenario. The three tuning parameter selection scenarios appear to be comparable with respect to predictive log-likelihood loss. This suggests that it is possible to achieve similar performance with respect to predictive loss when estimating C or j_0 from the data as when C and j_0 are known.

In Scenario 2, we allowed the data to select the number of components, C. Table 3 shows the proportions of simulation runs at each of the three settings for which the number of components was chosen to be 1, 2, or 3. The table suggests that 5-fold cross validation has greater tendency than BIC to over-fit by estimating more components than truly exist. Hence we are inclined to recommend using BIC when selecting the number of components. Additional information on Simulation 2 regarding estimation performance and computational considerations are provided in the online appendix.

5.3. Application to DTI Data for Subjects with Multiple Sclerosis

We now analyze data from a di usion tensor imaging (DTI) study, discussed in Goldsmith et al. (2012), using our wavelet-based functional mixture regression approach. The data are from a longitudinal study investigating the cerebral white matter tracts of subjects with multiple sclerosis (MS). Here we focus on the baseline observations for the 100 MS subjects. In particular, we are interested in the relationship between the fractional anisotropy profile (FAP) from the corpus callosum (functional predictor) and the Paced Auditory Serial Addition Test (PASAT) score (scalar response).

The PASAT is an assessment tool that measures a subject's cognitive ability with respect to auditory information processing speed and flexibility and also provides information on calculation ability (Rosti et al., 2006). The PASAT score ranges from 0 to 60 where lower scores indicate some level of dysfunction. The FAP from the corpus callosum is derived from DTI, a magnetic resonance imaging modality that is commonly used to track the di usion of water in biological tissue. The FAP is a continuous summary of water di usivity that is parametrized by the arc length along a curve. The tract profiles are estimated via an automated tract-probability-mapping scheme described in Reich et al. (2010). In the data set, the FAP predictors are recorded at 93 locations along the corpus callosum. In our analysis, we linearly interpolate the FAP curves at 128 equally spaced points before projecting them onto a wavelet basis. We used data from 99 of the 100 MS subjects since one subject had missing FAP values at several locations along the tract. Figure 4 shows the FAPs for all 99 MS subjects that we considered as well as those for three subjects with the lowest, median, and highest PASAT scores.

We were interested in conducting an analysis that inspects whether the regression relationship between corpus callosum FAP and PASAT score varies due to some unknown mechanism. We apply our WB functional mixture regression approach in which we used the BIC from (4.2) to select the optimal tuning parameters. This approach suggests that there are two distinct groups with different coefficient functions describing the association between corpus callosum FAP and PASAT score. Figure 5 shows the estimated coefficient functions, $\hat{\omega}_1$ and $\hat{\omega}_2$, for each of the two groups. For illustration, Figure 5 also shows the FAPs that belong to the groups associated with those functions. To determine which group a subject's

FAP belongs to, we use the estimated group membership indicators from the last iteration of the EM algorithm. The indicator of group membership with the the highest value was taken to correspond to the group from which the observation came. Using this assignment method, there are 52 subjects in Group 1 and 47 in Group 2.

We note that the estimated coefficient function corresponding to Group 2 is identically zero at all locations along the profile suggesting no association between FAP and PASAT score among MS subjects belonging to this group. The estimated coefficient function for Group 1 suggests that higher fractional anisotropy values between profile locations of about 0.2 and 0.7 are associated with higher PASAT scores while higher values between profile locations of about 0.7 and 0.9 are associated with lower PASAT scores for those MS subjects belonging to Group 1.

Figure 5 also shows the PASAT scores corresponding to the two groups. This plot illustrates a distinctive split between the two groups with respect to PASAT score. Overall the model may suggest that, among MS subjects with better cognitive function, there is no association between corpus callosum FAP and PASAT score whereas among those with worse cognitive function, fractional anisotropy values in the middle region of the tract can discriminate among the PASAT scores and that greater fractional anisotropy corresponds to higher scores.

Finally, we compare the chosen wavelet-based function mixture regression model to the wavelet-based FLM (selected to minimize BIC) with respect to leave-one-out cross-validated relative prediction errors $CVRPE = \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i^{(-i)} \right)^2 / \sum_{i=1}^{n} Y_i^2$ where $\hat{Y}_i^{(-i)}$ is the predicted PASAT score for subject *i* from a model fit on data with subject *i* removed. To determine which estimated coefficient function to use to obtain the predicted PASAT score for subject *i*, we use the following ad hoc method similar to that used in Yao et al. (2011): if the observed PASAT score Y_i is less than 50 then we use the coefficient function that is not identically zero at each profile location and if Y_i is 50 or larger then we use the zero function. For our model with 2 groups, the CVRPE is 0.0315 and for the wavelet-based FLM the CVRPE is 0.0723.

6. Discussion

In this article we present a general wavelet-based approach to functional mixture regression which is appropriate to use when modeling the association between a continuous scalar response and a functional predictor where the association is not homogeneous across the population. We provide a fitting algorithm and demonstrate some properties of the corresponding estimators using simulations. Although our approach may be more computationally demanding, due in large part to the need to select several tuning parameters, when compared with a functional principal components-based approach to functional mixture regression, evidence suggests that our method performs better with respect to prediction and estimation accuracy when the component coefficient functions defining the association between the predictors and responses possess relatively small scale features. Furthermore, our approach can be directly and easily extended to handle functional

predictors of higher dimensionality including 2- and 3-dimensional images. Existing software can be used to obtain a discrete wavelet decomposition of the image and once we have the corresponding wavelet coefficients, those coefficients can be organized into a vector in which the ordering of the coefficients does not matter. Then model fitting is carried out in the the same manner as presented above.

Zhao et al. (2012) note that there are many factors that may be important to the performance of a wavelet-based approach like the one we present here. For one, selection of a particular wavelet basis for the DWT has an impact on the sparsity of the representation of the functional predictor. This is not an issue when using a functional principal components-based approach since the basis representation is determined by the functional predictors.

Another factor that plays an important role in the performance of our method is tuning parameter selection. We looked at two criteria for selecting tuning parameters: minimizing the 5-fold cross-validated predictive loss and minimizing a modified BIC value. We found that both methods were generally comparable with the BIC method perhaps slightly under-performing with respect to predictive loss. However, simulations showed that BIC tended to select the correct number of components more often and 5-fold cross-validation tended to over fit. Estimation performance was generally comparable (see supplementary material) between the cross-validation and BIC procedures, and so we recommend using BIC to select tuning parameters in practice.

We noted in Section 2.2 that it is common to relate the amount of penalty on the covariates to the sample size as is done in (2.7) by including π_r in the penalty function. In their ℓ_1-penalized mixture approach, Städler et al. (2010) suggest including an additional tuning parameter,, in the form of an exponent on the mixing probability π_r. They consider using only the values of $\gamma \in \{0, 1/2, 1\}$. They suggest using the value of 0 when the true mixing proportions are not very different from each other and using 1/2 or 1 when the mixing proportions are unbalanced. Our method corresponds to the case where $\gamma = 1$. In other simulations (not presented here) we compared models that resulted from using different values of γ in both balanced and unbalanced settings but generally saw little di erence with respect to predictive loss.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

The authors would like to thank the reviewers of an earlier version of this article for their constructive comments. This work was partially supported by NIBIB grant 5 R01 EB009744. We would also like to thank Nicolas Städler for providing his R code for high-dimensional mixture regression model fitting as well as Lan Huo and Phil Reiss for the code that they have provided.

References

Amato U, Antoniadis A, De Feis I. Dimension reduction in dimension reduction in functional regression with applications. Computational Statistics and Data Analysis. 2006; 50:2422–2446.

Cai T, Hall P. Prediction in functional linear regression. Annals of Statistics. 2006; 34:2159–2179.

Cardot H, Ferraty F, Sarda P. Spline estimators for the functional linear model. Statistica Sinica. 2003; 13:571–591.

Cardot H, Sarda P. Estimation in generalized linear models for functional data via penalized likelihood. Journal of Multivariate Analysis. 2005; 92:24–41.

Daubechies I. Orthonormal bases of compactly supported wavelets. Communications on Pure and Applied Mathematics. 1988; 41:909–996.

Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. Journal of Statistical Software. 2010; 33:1–22. [PubMed: 20808728]

Goldsmith J, Crainiceanu C, Ca o B, Reich D. Longitudinal penalized functional regression for cognitive outcomes on neuronal tract measurements. Journal of the Royal Statistical Society, Series C. 2012; 12:453–469.

Hennig C. Identifiability of models for clusterwise linear regression. Journal of Classification. 2000; 17:273–296.

James GM. Generalized linear models with functional predictors. Journal of the Royal Statistical Society Series B. 2002; 64:411–432.

James GM, Silverman BW. Functional adaptive model estimation. Journal of the American Statistical Association. 2005; 100:565–576.

Khalili A, Chen J. Variable selection in finite mixture of regression models. Journal of the American Statistical Association. 2007; 102:1025–1038.

Leisch F. FlexMix: A general framework for finite mixture models and latent class regression in R. Journal of Statistical Software. 2004; 11:1–18.

Mallat SG. A theory for multiresolution signal decomposition: The wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1989; 11:674–693.

Malloy E, Morris J, Adar S, Suh H, Gold D, Coull B. Wavelet-based functional linear mixed models: An application to measurement error-corrected distributed lag models. Biostatistics. 2010; 11:432–452. [PubMed: 20156988]

McLachlan, G.; Peel, D. Finite Mixture Models. Wiley-Interscience; New York: 2000.

Meier L, van de Geer S, Bühlmann P. The group lasso for logistic regression. Journal of the Royal Statistical Society, Series B. 2008; 70:53–71.

Müller H, Yao F. Functional additive models. Journal of the American Statistical Association. 2008; 103:1534–1544.

Nason, G. Wavethresh software. Department of Mathematics, University of Bristol; Bristol, UK: 1998.

Nason, G. Wavelet Methods in Statistics, with R. Springer; New York: 2008.

Ogden, RT. Essential Wavelets for Statistical Applications and Data Analysis. Birkẗauser; Boston: 1997.

Pan W, Shen X. Penalized model-based clustering with application to variable selection. Journal of Machine Learning Research. 2007; 8:1145–1164.

Ramsay, JO.; Silverman, BW. Functional Data Analysis. Second Edition. Springer; New York: 2005.

Reich D, Ozturk A, Calabresi P, Mori S. Automated vs. conventional tractography in multiple sclerosis: Variability and correlation with disability. NeuroImage. 2010; 49:3047–3056. [PubMed: 19944769]

Reiss, PT.; Huo, L.; Ogden, RT.; Zhao, Y.; Kelly, C. Wavelet-domain regression with image predictors, and a surprising (non-)result in psychiatric neuroimaging. 2013. (preprint)

Reiss PT, Ogden RT. Functional principal component regression and functional partial least squares. Journal of the American Statistical Association. 2007; 102:984–996.

Rosti E, Ẅamäläinen P, Koivisto K, Hokkanen L. The PASAT performance among patients with multiple sclerosis: Analyses of responding patterns using different scoring methods. Multiple Sclerosis. 2006; 12:586–593. [PubMed: 17086904]

Schlattmann, P. Medical Applications of Finite Mixture Models. Springer-Verlag; Berlin: 2009.

Städler N, Bühlmann P, van de Geer S. ℓ_1-penalization for mixture regression models. Test. 2010; 19:209–256.

Vannucci M, Sha N, Brown P. Nir and mass spectra classification: Bayesian methods for wavelet-based feature selection. Chemometrics and Intelligent Laboratory Systems. 2005; 77:139–148.

Vidakovic, B. Statistical Modeling by Wavelets. Wiley; New York: 1999.

Yao F, Fu Y, Lee T. Functional mixture regression. Biostatistics. 2011; 12:341–353. [PubMed: 21030384]

Zhao Y, Ogden RT, Reiss PT. Wavelet-based lasso in functional linear regression. Journal of Computational and Graphical Statistics. 2012; 21:600–617. [PubMed: 23794794]

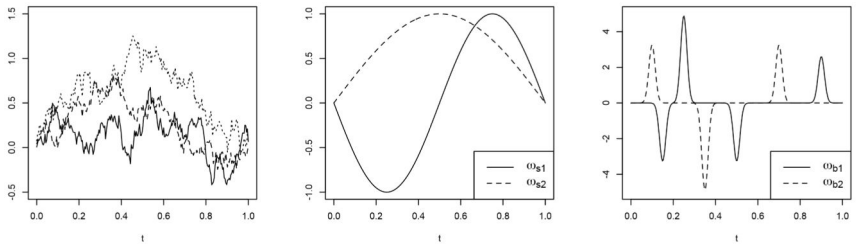

Figure 1.

Left: Sample of three predictor curves ($N = 256$). Center: Component coefficient functions in smooth setting (ω_{s1} and ω_{s2}). Right: Component coefficient functions in bumpy setting (ω_{b1} and ω_{b2}).

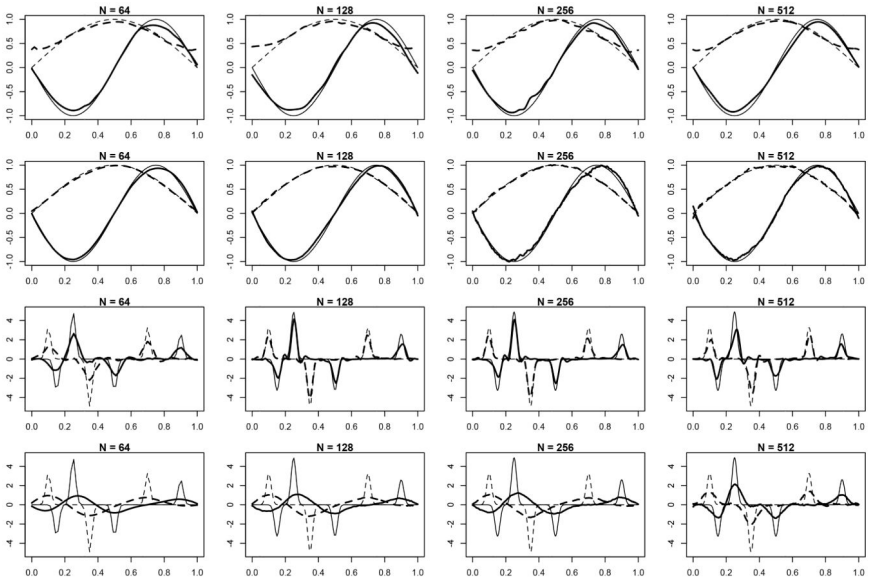

Figure 2.

$R^2 = 0.9$; solid and dashed thin curves correspond to the truth; solid and dashed thick curves correspond to the point-wise mean estimated component coefficient functions; rows 1 and 3 depict WB method; rows 2 and 4 depict FPC method.

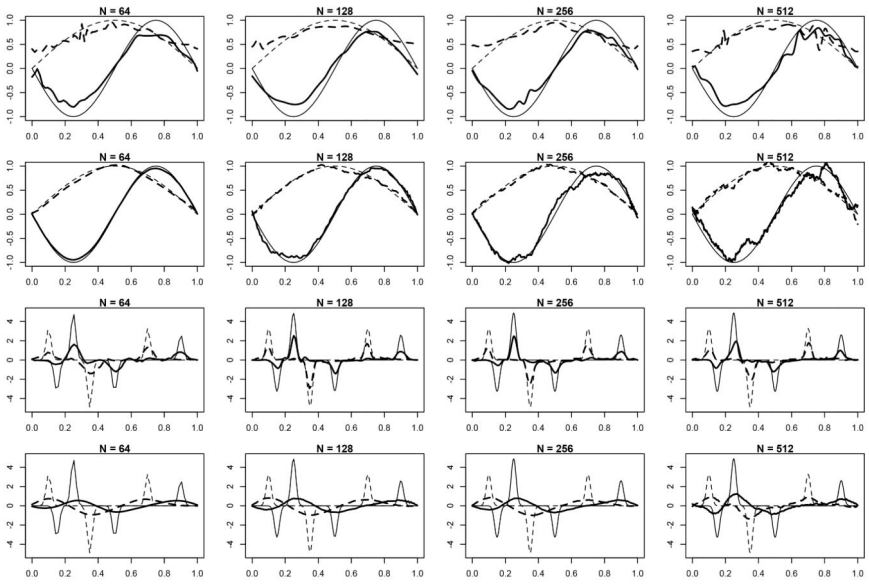

Figure 3.

$R^2 = 0.7$; solid and dashed thin curves correspond to the truth; solid and dashed thick curves correspond to the point-wise mean estimated component coefficient functions; rows 1 and 3 depict WB method; rows 2 and 4 depict FPC method.

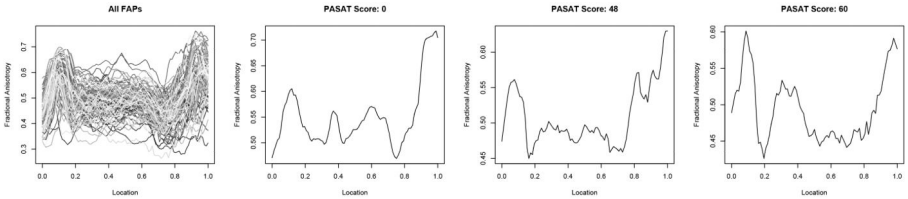

Figure 4.

FAPs for all subjects (first panel), subject with lowest PASAT score (second panel), subject with median PASAT score (third panel), and subject with highest (fourth panel) PASAT score.

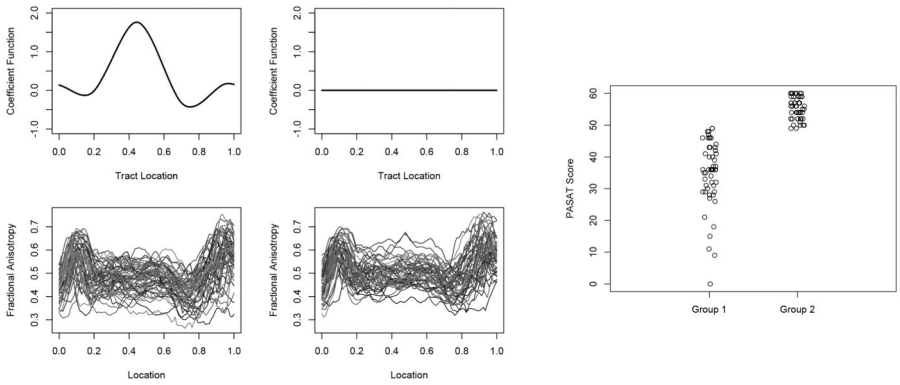

Figure 5.

Top left panels: estimated coefficient functions $\hat{\omega}_1$ (left) and $\hat{\omega}_2$ (right) determined by the WB functional mixture regression approach. Bottom left panels: FAP curves for MS subjects that correspond to $\hat{\omega}_1$ (left) and for those that correspond to $\hat{\omega}_2$ (right). Right panel: PASAT Scores for those in Group 1 (corresponding to $\hat{\omega}_1$) and in Group 2 (corresponding to $\hat{\omega}_2$).

Table 1

Mean (standard deviation) log-likelihood test loss comparing Wavelet-Based (WB) and Functional Principal Components-Based (FPC) methods in different settings.

		Smooth		
		***R2* = 0.9**	**R^2 = 0.7**	***R^2* = 0.5**
N = 64	WB	644.31 (16.97)	745.71 (17.50)	810.13 (18.54)
	FPC	646.28 (19.74)	745.09 (18.95)	811.03 (18.32)
N = 128	WB	784.20 (18.84)	885.43 (20.79)	950.61 (22.96)
	FPC	786.58 (21.06)	890.02 (22.99)	955.90 (24.31)
N = 256	WB	923.14 (19.19)	1024.70 (21.45)	1089.17 (21.04)
	FPC	926.73 (22.19)	1029.43 (23.96)	1093.04 (23.72)
N = 512	WB	1061.71 (20.74)	1162.09 (21.08)	1225.62 (21.19)
	FPC	1064.83 (22.02)	1166.00 (21.48)	1229.20 (21.03)
		Bumpy		
		R^2 = 0.9	R^2 = 0.7	R^2 = 0.5
N = 64	WB	455.61 (19.86)	552.03 (21.09)	611.86 (22.63)
	FPC	476.01 (23.40)	562.25 (23.88)	618.21 (22.25)
N = 128	WB	592.84 (21.07)	693.23 (20.46)	756.26 (24.57)
	FPC	616.40 (25.35)	703.38 (21.18)	761.40 (24.53)
N = 256	WB	734.67 (18.00)	836.91 (20.46)	897.84 (18.17)
	FPC	758.41 (23.01)	849.79 (25.24)	904.03 (23.13)
N = 512	WB	874.30 (20.95)	974.10 (20.55)	1034.70 (21.46)
	FPC	895.89 (25.30)	983.39 (22.94)	1040.91 (22.98)

Table 2

Mean (standard deviation) log-likelihood test loss for different tuning parameter selection scenarios.

		Smooth		
		Scenario 1	**Scenario 2**	**Scenario 3**
$N = 128$	CV	762.64 (20.22)	760.06 (20.29)	758.71 (20.43)
	BIC	764.20 (20.78)	764.06 (20.59)	764.00 (20.25)
$N = 256$	CV	904.45 (17.51)	902.26 (19.25)	900.38 (19.78)
	BIC	906.59 (18.13)	906.67 (18.16)	906.42 (18.23)
		Bumpy		
		Scenario 1	Scenario 2	Scenario 3
$N = 128$	CV	559.52 (31.15)	558.16 (33.11)	562.25 (30.25)
	BIC	554.72 (37.74)	556.90 (38.89)	561.98 (33.42)
$N = 256$	CV	704.18 (34.34)	705.44 (39.56)	699.25 (28.63)
	BIC	706.43 (40.35)	705.50 (44.10)	706.77 (30.76)

Table 3

Proportion of simulation runs at each setting such that the indicated number of components (C) is chosen by either 5-fold cross-validation or modified BIC.

		Smooth			Bumpy		
		$C = 1$	$C = 2$	$C = 3$	$C = 1$	$C = 2$	$C = 3$
$N = 128$	CV	0.00	0.62	0.38	0.04	0.76	0.20
	BIC	0.00	1.00	0.00	0.05	0.94	0.01
$N = 256$	CV	0.00	0.59	0.41	0.07	0.72	0.21
	BIC	0.00	1.00	0.00	0.05	0.94	0.01