Journal of Computational and Graphical Statistics

ISSN: 1061-8600 (Print) 1537-2715 (Online) Journal homepage: http://www.tandfonline.com/loi/ucgs20

Locally Sparse Estimator for Functional Linear Regression Models

Zhenhua Lin, Jiguo Cao, Liangliang Wang & Haonan Wang

To cite this article: Zhenhua Lin, Jiguo Cao, Liangliang Wang & Haonan Wang (2017) Locally Sparse Estimator for Functional Linear Regression Models, Journal of Computational and Graphical Statistics, 26:2, 306-318, DOI: 10.1080/10618600.2016.1195273

To link to this article: http://dx.doi.org/10.1080/10618600.2016.1195273

JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS
2017, VOL. 26, NO. 2, 306–318
http://dx.doi.org/10.1080/10618600.2016.1195273

Taylor & Francis
Taylor & Francis Group

Locally Sparse Estimator for Functional Linear Regression Models

Zhenhua Lin[a], Jiguo Cao[b], Liangliang Wang[b], and Haonan Wang[c]

[a]Department of Statistical Sciences, University of Toronto, Toronto, ON, Canada; [b]Department of Statistics and Actuarial Science, Simon Fraser University, Burnaby, BC, Canada; [c]Department of Statistics, Colorado State University, Fort Collins, Colorado

ABSTRACT

A new locally sparse (i.e., zero on some subregions) estimator for coefficient functions in functional linear regression models is developed based on a novel functional regularization technique called "fSCAD." The nice shrinkage property of fSCAD allows the proposed estimator to locate null subregions of coefficient functions without over shrinking nonzero values of coefficient functions. Additionally, a roughness penalty is incorporated to control the roughness of the locally sparse estimator. Our method is theoretically sounder and computationally simpler than existing methods. Asymptotic analysis reveals that the proposed estimator is consistent and can identify null subregions with probability tending to one. Extensive simulations confirm the theoretical analysis and show excellent numerical performance of the proposed method. Practical merit of locally sparse modeling is demonstrated by two real applications. Supplemental materials for the article are available online.

ARTICLE HISTORY
Received December 2015
Revised May 2016

KEYWORDS
B-spline basis; Functional data analysis; Null region; SCAD; Smoothing spline

1. Introduction

Functional linear regression (FLR) is a popular technique in data analysis when predictors themselves are functions. In classic FLR models, a scalar response Y_i of the ith experimental subject, $i = 1, 2, \ldots, n$, is related to a functional predictor $X_i(t)$ by a coefficient function $\beta(t)$ in the way that

$$Y_i = \mu + \int_0^T X_i(t)\beta(t)dt + \varepsilon_i, \tag{1}$$

where μ is the grand mean of all Y_i, and ε_i represents measurement error. Further assumptions on the model are that $X_1(t), \ldots, X_n(t)$ are independent realizations of an unknown centered stochastic process $X(t)$ defined on domain $[0, T]$, and that $\varepsilon_i \sim N(0, \sigma_\varepsilon^2)$, $i = 1, \ldots, n$. The coefficient function $\beta(t)$ is also called the regression weight function, since $\beta(t)$ gives rise to weights of the contribution of the functional predictor $X_i(t)$ to the corresponding response Y_i at each point $t \in [0, T]$.

In particular, if on a subregion $I \subset [0, T]$, $\beta(t) = 0$ for every $t \in I$, then X_i has no contribution to Y_i on the interval I. In light of this observation, an estimate of $\beta(t)$ improves the interpretability of the model and is practically appealing, if it not only yields weights of the contribution of $X_i(t)$ over the entire domain, but also locates subregions where $X_i(t)$ has no statistically significant contribution to Y_i. This estimate of $\beta(t)$ is called the locally sparse estimate, where the term "locally sparse" appears in Tu et al. (2012) and Wang and Kai (2015) to mean a curve is zero on some subregions of its domain. The focus of this article is to develop a locally sparse estimator for the coefficient function $\beta(t)$.

Historically, the functional linear regression originates from the ordinary linear regression with a large number of predictors. When the number of predictors is very large, estimators produced by the ordinary least squares exhibit excessive variability, and hence perform poorly on prediction. Statistical approaches have been proposed to rectify the issue, such as partial least squares (PLS), principal components regression (PCR), ridge regression, etc. Unfortunately, when predictors are actually discrete observations of a continuous process, none of these methods directly make use of the spatial information, that is, the correlation and the order between predictors (Hastie and Mallows 1993). More promising are techniques of regularizing the variability by penalized least squares or smoothing splines to directly restrict the coefficient vector to be smooth, as prototyped in Hastie and Mallows (1993) and explored more thoroughly in Ramsay and Silverman (1997).

Since then, more theoretical and practical treatments of the functional linear regression have emerged. For example, identifiability, existence, and unicity of estimate of the coefficient function $\beta(t)$ were studied in Cardot, Ferraty, and Sarda (2003). Smooth estimators of $\beta(t)$ based on a direct roughness penalty on $\beta(t)$ were studied in Cardot, Ferraty, and Sarda (2003), Li and Hsing (2007), and Crambes, Kneip, and Sarda (2009). Alternative smooth estimators that are based on spectral decomposition of the covariance function of $X(t)$ were developed by Cardot, Ferraty, and Sarda (2003), Cai and Hall (2006), and Hall and Horowitz (2007). Convergence rates of some estimators in the weak topology of $L^2([0, T])$ were treated in Cardot, Mas, and Sarda (2007). Preda (2007) and Yuan and Cai (2010) explored FLR from the point of view of reproducing kernel Hilbert space. Extensions of classic FLR include generalized functional linear models studied by Müller and Stadtmüller (2005), PACE proposed by Yao, Müller, and Wang (2005) for sparse longitudinal

CONTACT Jiguo Cao jiguo_cao@sfu.ca Department of Statistics and Actuarial Science, Simon Fraser University, Burnaby, BC V5A 1S6, Canada.
Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/r/JCGS.
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JCGS.

data, and FLR with functional response explored by Fan and Zhang (2000) and Lian (2012).

Although the literature on FLR is abundant, little has been done on interpretability and locally sparse modeling, besides "FLiRTI" proposed by James, Wang, and Zhu (2009) and a two-stage method developed by Zhou, Wang, and Wang (2013). FLiRTI achieves local sparseness by employing L_1 penalty on the function $\beta(t)$ and its first several derivatives at some discrete grid points. While the idea is quite intuitive and neat, careful analysis reveals that the method is not theoretically sound and might fail due to the following reasons. On one hand, for an estimate $\hat{\beta}(t)$ being zero on some subregion I, the grid points that fall into I are required to be zero simultaneously. However, the L_1 regularization does not warrant this requirement. On the other hand, the condition that a function and its first several derivatives are zero at a point is not sufficient for the function to be zero on a subregion around the point. Instead, it requires a much stronger condition that the function and *all* of its derivatives are zero at that point. Another drawback of FLiRTI method, as noted by Zhou, Wang, and Wang (2013), is that the produced estimate $\hat{\beta}(t)$ possesses large variation, because when the grid size is small, the numerical solution is unstable, while when the grid size is large, FLiRTI method tends to overparameterize the model. To overcome the conundrum, Zhou, Wang, and Wang (2013) proposed an alternative locally sparse estimator of $\beta(t)$ obtained in two stages. In the first stage, Dantzig selector is used to obtain an initial estimate of null subregions. In the second stage, the initial null subregions are refined via the group SCAD penalty proposed by Wang, Chen, and Li (2007). However, the two-stage procedure requires selection of several tuning parameters in each stage, which not only increases the estimation variability and computational complexity, but also makes the method difficult to implement, and therefore limits applications of this method.

To address the above problems of existing approaches, in this article we propose a simple one-stage procedure that yields a smooth and locally sparse estimator of the coefficient function $\beta(t)$. The proposed estimator is called *SLoS* (smooth and locally *S*parse), which simultaneously identifies null subregions of $\beta(t)$ and produces a smooth estimate in nonnull subregions. The procedure relies on a novel functional regularization technique that we call "functional smoothly clipped absolute deviation" (fSCAD for short). The fSCAD can be viewed as a functional generalization of ordinary SCAD proposed by Fan and Li (2001), and inherently has a nice shrinkage property that enables the SLoS estimator to identify null subregions of $\beta(t)$ without over shrinking nonzero values of $\beta(t)$. In addition, a roughness penalty is employed to regularize the smoothness of the SLoS estimator. Compared to existing methods in the literature, the proposed SLoS method has two distinct features. First, unlike the pointwise penalization of FLiRTI method, the fSCAD penalty regularizes the overall magnitude of the estimated coefficient function $\hat{\beta}(t)$ in small subintervals. Second, unlike the two-stage procedure in Zhou, Wang, and Wang (2013), the SLoS method combines fSCAD regularization and smoothing splines together in a single optimization objective function. By solving a single optimization problem, we are able to produce an estimate of null subregions, as well as a smooth estimate of $\beta(t)$ in its nonnull subregions simultaneously in one single stage. These two features make SLoS estimator theoretically sounder, computationally simpler, and statistically stabler. An R package "slos" is developed to implement the proposed SLoS estimator for the functional linear regression model. This package can be downloaded at *http://people.stat.sfu.ca/~cao/Research/FLR/*.

The rest of the article is organized as follows. In Section 2, the fSCAD regularization technique and the SLoS estimator are developed. Asymptotic properties of SLoS estimator are discussed in Section 3. Present in Section 4 is evaluation of numerical performance of the proposed estimator via simulation studies. Applications of the developed method to two real datasets are explored in Section 5. Section 6 concludes the article.

2. Methodology

2.1 Functional SCAD

To motivate fSCAD, we first briefly review ordinary SCAD penalty in multivariate linear regression model

$$y_i = \mu + \sum_{j=1}^{J} b_j z_{ij} + \varepsilon_i, \tag{2}$$

where y_i is response variable, $z_{i1}, z_{i2}, \ldots, z_{iJ}$ are scalar predictors, μ represents the grand mean, $b_1, b_2, \ldots, b_J$ are regression coefficients, and measurement errors are $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$, $i = 1, \ldots, n$. Using least squares as the loss function, the SCAD estimate of $\mathbf{b} = (b_1, b_2, \ldots, b_J)$ is defined as

$$\hat{\mathbf{b}}^{\text{scad}} = \arg\min_{\mathbf{b}\in\mathbb{R}^J} \left\{ \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \mu - \sum_{j=1}^{J} b_j z_{ij}\right)^2 + \sum_{j=1}^{J} p_\lambda(|b_j|) \right\}, \tag{3}$$

where $p_\lambda(\cdot)$ is the SCAD penalty function of Fan and Li (2001). It is defined on $[0, +\infty]$ as

$$p_\lambda(u) = \begin{cases} \lambda u & \text{if } 0 \le u \le \lambda \\ -\dfrac{u^2 - 2a\lambda u + \lambda^2}{2(a-1)} & \text{if } \lambda < u < a\lambda \\ \dfrac{(a+1)\lambda^2}{2} & \text{if } u \ge a\lambda, \end{cases}$$

where a suggested value for a is 3.7 according to Fan and Li (2001), and λ is a tuning parameter varying with the sample size. The SCAD estimator $\hat{\mathbf{b}}^{\text{scad}}$ enjoys the so-called oracle property: it is able to identify the true sub-model with probability tending to one, and meanwhile produce an asymptotically normal estimate for each nonzero variable (Fan and Li 2001; Fan and Peng 2004; Fan and Lv 2011). In other words, the estimator $\hat{\mathbf{b}}^{\text{scad}}$ performs as well as if the true sub-model was known, that is, it was known that which coefficients b_j's were zero in advance.

The ordinary linear model (2) changes to the functional linear model (1) if the summation $\sum_{j=1}^{J} b_j z_{ij}$ is replaced by the integral $\int_0^T X_i(t)\beta(t)dt$. Similarly, the SCAD penalty term $\sum_{j=1}^{J} p_\lambda(|b_j|)$ in (3), if normalized by $\frac{1}{J}$, generalizes to the integral that we call fSCAD penalty:

$$\mathcal{L}(\beta) \overset{\text{def}}{=} \frac{1}{T}\int_0^T p_\lambda(|\beta(t)|)\, dt. \tag{4}$$

While fSCAD is introduced as a functional generalization of ordinary SCAD, an alternative introduction in the setting of locally sparse modelling provides insights into the shrinkage nature of fSCAD, as follows. Suppose $\beta(t)$ is locally sparse and a locally sparse estimate of $\beta(t)$ is preferred. A natural idea is to divide the domain into many small disjoint subintervals and then penalize the overall magnitude of $\beta(t)$ on each subinterval to shrink the estimated $\hat{\beta}(t)$ toward zero on those subintervals where the true $\beta(t)$ is zero. A possible measure of overall magnitude could be normalized L^q ($q \geq 1$) norm of $\beta(t)$ on each subinterval, and a possible penalty function could be the ordinary SCAD penalty. The following remarkable result shows that this idea of locally sparse modeling actually leads to our fSCAD regularization.

Theorem 1. Let $0 = t_0 < t_1 < \cdots < t_M = T$ be an equally spaced sequence in the domain $[0, T]$, and $\beta_{[j]}(t)$ denote the restriction of $\beta(t)$ on the subinterval $[t_{j-1}, t_j]$, that is, $\beta_{[j]}(t) = \beta(t)$ for $t \in [t_{j-1}, t_j]$ and zero elsewhere. If $\beta(t)$ is continuous, then for any $q \geq 1$, \hspace*{-6pt}

$$\frac{1}{T}\int_0^T p_\lambda(|\beta(t)|)\,dt = \lim_{M\to\infty}\frac{1}{M}\sum_{j=1}^{M} p_\lambda\left(M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q\right), \quad (5)$$

where $\|\beta_{[j]}\|_q \stackrel{\text{def}}{=} (\int_0^T |\beta_{[j]}(t)|^q dt)^{1/q}$.

In the above theorem, the normalized L^q norm $M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q$ measures "overall magnitude" of $\beta(t)$ over the subinterval $[t_{j-1}, t_j]$. The identity (5) shows that, the fSCAD of $\beta(t)$ is indeed the limit of the average SCAD penalty on the "overall magnitude" of $\beta(t)$ over each small subinterval $[t_{j-1}, t_j]$. The theorem sheds light on the shrinkage nature of fSCAD. On one hand, when $\beta(t)$ is overall very small on $[t_{j-1}, t_j]$ (i.e., $M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q$ is very small), then the penalty $p_\lambda(M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q)$ will shrink $\beta(t)$ toward zero identically for every $t \in [t_{j-1}, t_j]$ and hence produce a locally sparse estimate. By this way, fSCAD regularization avoids the aforementioned theoretical issue of FLiRTI method. On the other hand, when $\beta(t)$ has significant overall magnitude on $[t_{j-1}, t_j]$ (i.e., $M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q$ is big enough), then $\beta(t)$ does not get penalized on the subinterval $[t_{j-1}, t_j]$. In other words, fSCAD does not over shrink $\beta(t)$.

Now we provide a more precise description of the shrinkage nature of fSCAD regularization. First, since

$$\left(\min_{t\in[t_{j-1},t_j]}|\beta_{[j]}(t)|^q\right)\frac{T}{M} \leq \int_{t_{j-1}}^{t_j}|\beta_{[j]}(t)|^q dt = \|\beta_{[j]}\|_q^q \leq \|\beta_{[j]}\|_\infty^q\frac{T}{M},$$

where $\|\cdot\|_\infty$ denotes the supremum norm, immediately, we have

$$\min_{t\in[t_{j-1},t_j]}|\beta_{[j]}(t)| \leq M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q \leq \|\beta_{[j]}\|_\infty \leq \|\beta\|_\infty. \quad (6)$$

Let $\hat{\beta}(t)$ denote a uniformly consistent estimator of $\beta(t)$ and $H(\beta_{[j]}) = M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q$. For any fixed $t \in [0, T]$, if $\beta(t) \neq 0$, then by the continuity of $\beta(t)$ we have $|\beta(t)| > \epsilon$ on a small neighborhood $\mathcal{N}_\epsilon(t)$ of t for some $\epsilon > 0$. When M is sufficiently large, the subinterval $[t_{j-1}, t_j]$ containing t is inside $\mathcal{N}_\epsilon(t)$. Then with the probability tending to one, we have $|\hat{\beta}_{[j]}(s)| \geq \epsilon/2$ for all $s \in \mathcal{N}_\epsilon(t)$ and hence $H(\hat{\beta}_{[j]}) \geq \epsilon/2$. As $\lambda_n \to 0$, $H(\hat{\beta}_{[j]}) > a\lambda_n$ with the probability tending to one. This indicates that the consistent estimator $\hat{\beta}(t)$ does not get penalized for its values at t. On the other hand, if $\beta(t) = 0$ on a small interval $\mathcal{N}_0$, then $|\hat{\beta}(t)| \to 0$ on $\mathcal{N}_0$ with probability tending to one. When M is sufficiently large, one or more subintervals $[t_{j-1}, t_j]$ are inside $\mathcal{N}_0$. Then $H(\hat{\beta}_{[j]}) \to 0$ in probability. By choosing an appropriate λ_n, the penalty $p_{\lambda_n}(H(\hat{\beta}_{[j]}))$ grows with $H(\hat{\beta}_{[j]})$ in the rate λ_n, and hence forces $\hat{\beta}(t)$ to become identically zero on $[t_{j-1}, t_j]$.

Theorem 1 also provides a practical way to approximately compute fSCAD penalty by choosing a large M and then approximating $\frac{1}{T}\int_0^T p_\lambda(|\beta(t)|)dt$ by $\frac{1}{M}\sum_{j=1}^{M} p_\lambda(M^{\frac{1}{q}}T^{-\frac{1}{q}}\|\beta_{[j]}\|_q)$. Since L^2 is relatively easier to compute for most functional bases than other norms, $q = 2$ is a typical choice for q. The shrinkage feature and easy computation of fSCAD penalty bring it numerous potential applications in locally sparse modeling. For example, in Section 2.3, we use it to derive our SLoS estimator of the coefficient function in functional linear regression models.

2.2 Smoothing Spline Method

To prepare for the introduction of our SLoS estimator, here we briefly review the smoothing spline method for estimating $\beta(t)$. Recall that B-spline basis functions are defined by their order, as well as a sequence of knots. Suppose we set the order to $d + 1$ and place $M + 1$ equally spaced knots $0 = t_0 < t_1 < \cdots < t_M = T$ in the domain $[0, T]$ to define a set of B-spline basis functions. Note that in our method, these knots are also those grid points t_j for approximating the fSCAD penalty discussed in the previous subsection. Over each subinterval, each B-spline basis function $B_j(t)$ is a polynomial of the degree d. Moreover, each B-spline basis function is nonzero over no more than $d + 1$ consecutive subintervals. When M is large, each B-spline basis function is only nonzero on a small subregion. This property is called compact support property, which is critical for efficient computation and makes B-spline bases very popular in functional data analysis. For example, Figure 1 shows one of 23 cubic B-spline basis functions defined with 21 equally spaced knots. This basis function is only nonzero on the interval [0.1,0.3]. For more details about B-spline basis functions, readers are referred to de Boor (2001).

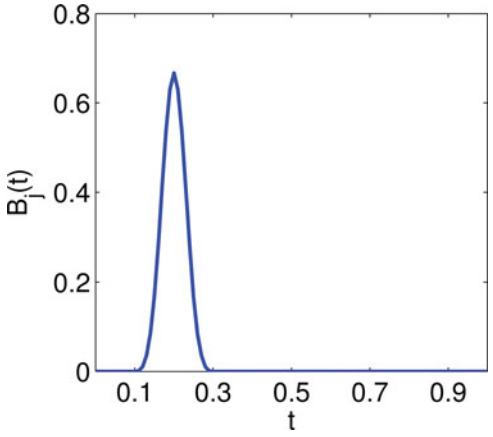

Figure 1. One example of cubic B-spline basis functions, which is only nonzero in [0.1,0.3].

Let $\mathcal{S}_{dM}$ denote the linear space spanned by the B-spline basis functions $\{B_j(t) : j = 1, 2, \ldots, M + d\}$ defined above. A spline estimator $\hat{\beta}^{\text{spline}}(t)$ of coefficient function $\beta(t)$ in model (1) is the one in $\mathcal{S}_{dM}$ that minimizes the sum of squared errors. However, this estimator usually exhibits excessive variability when M is relatively large. A popular approach to rectify the variability is to add a roughness penalty on $\beta(t)$. For example, a smooth estimator of $\beta(t)$, proposed in Cardot, Ferraty, and Sarda (2003), is defined as

$$\hat{\beta}^{\text{smooth}} = \arg\min_{\beta \in \mathcal{S}_{dM}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left[Y_i - \mu - \int_0^T X_i(t)\beta(t)dt \right]^2 + \gamma \|\mathcal{D}^m \beta\|^2 \right\},$$

where $\mathcal{D}^m$ is the mth-order differential operator with $m \leq d$, and $\|\cdot\|$ denotes L^2 norm of a function. The tuning parameter $\gamma \geq 0$ varies with the sample size n. When $\gamma = 0$, we obtain the spline estimator $\hat{\beta}^{\text{spline}}(t)$ of $\beta(t)$, while as $\gamma \to \infty$, the estimator $\hat{\beta}^{\text{smooth}}$ converges to a polynomial function of degree $m - 1$. Thus, γ serves as a smoothing parameter to control the degree of smoothing of $\hat{\beta}^{\text{smooth}}$. Note that $\hat{\beta}^{\text{smooth}}$ in general is not locally sparse. In other words, it is not able to identify null subregions of $\beta(t)$.

2.3 SLoS Estimator

To obtain a locally sparse estimator of $\beta(t)$ in model (1) using fSCAD and smoothing splines techniques, we propose to estimate $\beta(t)$ by $\hat{\beta} \in \mathcal{S}_{dM}$ that, together with an estimate $\hat{\mu}$ for μ, minimizes the penalized mean squared error $Q(\beta, \mu)$ as defined by

$$Q(\beta, \mu) = \frac{1}{n} \sum_{i=1}^{n} \left[Y_i - \mu - \int_0^T X_i(t)\beta(t)dt \right]^2 + \gamma \|\mathcal{D}^m \beta\|^2 + \frac{M}{T} \int_0^T p_\lambda(|\beta(t)|)dt. \quad (7)$$

In the loss function $Q(\beta, \mu)$, we combine two penalty terms in a single optimization criterion, namely, the roughness penalty $\gamma \|\mathcal{D}^m \beta\|$ that controls the smoothness of $\beta(t)$, and the fSCAD penalty term $\frac{M}{T} \int_0^T p_\lambda(|\beta(t)|)dt$ that regularizes the local sparseness of $\beta(t)$. The $\hat{\beta}$ defined above is called SLoS estimator of β.

In Equation (7), the fSCAD penalty is scaled up by a factor M, where M is the number of subintervals segmented by the knots that define B-spline basis functions. This is necessary for the SLoS estimator $\hat{\beta}(t)$ to enjoy theoretical properties presented in Section 3. Also, according to Theorem 1, the fSCAD penalty term can be approximated by

$$\frac{M}{T} \int_0^T p_\lambda(|\beta(t)|)dt \approx \sum_{j=1}^{M} p_\lambda \left(\frac{\|\beta_{[j]}\|_2}{\sqrt{T/M}} \right) = \sum_{j=1}^{M} p_\lambda \left(\sqrt{\frac{M}{T} \int_{t_{j-1}}^{t_j} \beta^2(t)dt} \right). \quad (8)$$

Thus, the term $\frac{M}{T} \int_0^T p_\lambda(|\beta(t)|)dt$ actually represents a penalty on the overall magnitude of $\beta(t)$ of all M subintervals. In light of this observation, we emphasize that fSCAD penalty regularizes $\beta(t)$ in the way that it penalizes the overall magnitude of $\beta(t)$ on each subinterval, rather than the pointwise penalization fashion adopted in FLiRTI (James, Wang, and Zhu 2009). Consequently, if the fSCAD penalty shrinks the overall magnitude of continuous function $\hat{\beta}(t)$ on a subinterval to zero, then $\hat{\beta}(t)$ is identically zero on that subinterval. Therefore, the proposed SLoS estimator overcomes the shortcoming of FLiRTI (James, Wang, and Zhu 2009). Moreover, the method is a one-stage fitting procedure and hence computationally simpler than the two-stage method proposed by Zhou, Wang, and Wang (2013).

Below we shall show how to compute the SLoS estimator $\hat{\beta}(t)$ in practice. To simplify notation and manifest our main idea, we assume $\mu = 0$, and redefine the loss function $Q(\beta) = Q(\beta, \mu = 0)$. The case that $\mu \neq 0$ is considered at the end of this section. As the minimization of $Q(\beta)$ is over all $\beta \in \mathcal{S}_{dM}$, we represent $\beta(t)$ as a linear combination of B-spline basis functions,

$$\beta(t) = \sum_{k=1}^{M+d} b_k B_k(t) = \mathbf{B}^T(t)\mathbf{b},$$

where $\mathbf{B}(t) = (B_1(t), B_2(t), \ldots, B_{M+d}(t))^T$ is the vector of B-spline basis functions, and $\mathbf{b} = (b_1, b_2, \ldots, b_{M+d})^T$ is the corresponding vector of coefficients. Let $\mathbf{U}$ be an $n \times (M + d)$ matrix with entries $u_{ij} = \int_0^T X_i(t)B_j(t)dt$ for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, (M + d)$. Let $\mathbf{y} = (Y_1, Y_2, \ldots, Y_n)^T$. Then the first term of $Q(\beta)$ in (7) is written in matrix notation as

$$\frac{1}{n} \sum_{i=1}^{n} \left[Y_i - \int_0^T X_i(t)\beta(t)dt \right]^2 = \frac{1}{n}(\mathbf{y} - \mathbf{U}\mathbf{b})^T(\mathbf{y} - \mathbf{U}\mathbf{b}). \quad (9)$$

Let $\mathbf{V}$ be an $(M + d) \times (M + d)$ matrix with entries $v_{ij} = \int_0^T \left(\frac{d^m B_i(t)}{dt^m} \frac{d^m B_j(t)}{dt^m} \right) dt$ for $1 \leq i, j \leq M + d$. Then the second term of $Q(\beta)$ in (7) is

$$\gamma \|\mathcal{D}^m \beta\|^2 = \gamma \mathbf{b}^T \mathbf{V} \mathbf{b}. \quad (10)$$

Next, we turn to the third term of $Q(\beta)$ in (7). According to Theorem 1, we approximate the fSCAD penalty term $\frac{M}{T} \int_0^T p_\lambda(|\beta(t)|)dt$ as (8), and

$$\int_{t_{j-1}}^{t_j} \beta^2(t)dt = \mathbf{b}^T \mathbf{W}_j \mathbf{b},$$

where $\mathbf{W}_j$ is an $(M + d)$-by-$(M + d)$ matrix with entries $w_{uv} = \int_{t_{j-1}}^{t_j} B_u(t)B_v(t)dt$ if $j \leq u, v \leq j + d$ and zero otherwise. The SCAD penalty function $p_\lambda(\cdot)$ might be approximated by the local quadratic approximation (LQA; Fan and Li 2001) or the one-step local linear approximation (LLA; Zou and Li 2008; Noh and Park 2010). It has been shown that LLA is able to produce a local minimizer that enjoys the oracle property (Zou and Li 2008; Noh and Park 2010), while it remains unclear whether the local minimizer found by LQA has the oracle property or not. However, computationally, LLA does not interact with L^q norm of functions well. In fact, we find that it is difficult to optimize (7) with LLA for SCAD. In contrast, LQA with L^2 norm of β yields a quadratic representation of the fSCAD penalty. This feature not only makes optimization with the fSCAD penalty feasible, but also dramatically simplifies the computation. In simulation studies presented in Section 4, we will also show that LQA indeed yields solutions of high quality in practice. Thus, instead

of LLA, we adopt LQA for the SCAD penalty in our algorithm as follows.

When $u \approx u^{(0)}$, the LQA of SCAD function $p_\lambda(u)$ is

$$p_\lambda(|u|) \approx p_\lambda(|u^{(0)}|) + \frac{1}{2}\frac{p'_\lambda(|u^{(0)}|)}{|u^{(0)}|}(u^2 - u^{(0)2}).$$

Then given some initial estimate $\beta^{(0)}$, for $\beta \approx \beta^{(0)}$, we have

$$\sum_{j=1}^{M} p_\lambda\left(\frac{\|\beta_{[j]}\|_2}{\sqrt{T/M}}\right) \approx \frac{1}{2}\sum_{j=1}^{M} \frac{p'_\lambda\left(\frac{\|\beta^{(0)}_{[j]}\|_2}{\sqrt{T/M}}\right)}{\frac{\|\beta^{(0)}_{[j]}\|_2}{\sqrt{T/M}}}\frac{\|\beta_{[j]}\|_2^2}{T/M} + G(\beta^{(0)}), \tag{11}$$

where $G(\beta^{(0)})$ is a term defined as

$$G(\beta^{(0)}) \stackrel{\text{def}}{=} \sum_{j=1}^{M} p_\lambda\left(\frac{\|\beta^{(0)}_{[j]}\|_2}{\sqrt{T/M}}\right) - \frac{1}{2}\sum_{j=1}^{M} p'_\lambda\left(\frac{\|\beta^{(0)}_{[j]}\|_2}{\sqrt{T/M}}\right)\frac{\|\beta^{(0)}_{[j]}\|_2}{\sqrt{T/M}}.$$

Let

$$\mathbf{W}^{(0)} = \frac{1}{2}\sum_{j=1}^{M}\left(\frac{p'_\lambda(\|\beta^{(0)}_{[j]}\|_2\sqrt{M/T})}{\|\beta^{(0)}_{[j]}\|_2\sqrt{T/M}}\mathbf{W}_j\right). \tag{12}$$

Then we have

$$\frac{M}{T}\int_0^T p_\lambda(|\beta(t)|)dt \approx \mathbf{b}^T\mathbf{W}^{(0)}\mathbf{b} + G(\beta^{(0)}). \tag{13}$$

Now, we combine (9), (10), and (13) together to express $Q(\beta)$ as

$$R(\mathbf{b}) \stackrel{\text{def}}{=} \frac{1}{n}(\mathbf{y} - \mathbf{U}\mathbf{b})^T(\mathbf{y} - \mathbf{U}\mathbf{b}) + \gamma\mathbf{b}^T\mathbf{V}\mathbf{b} + \mathbf{b}^T\mathbf{W}^{(0)}\mathbf{b} + G(\beta^{(0)}). \tag{14}$$

Thus, optimizing $Q(\beta)$ with respect to $\beta(t)$ is equivalent to optimizing $R(\mathbf{b})$ with respect to $\mathbf{b}$ in (14). Note that the term $G(\beta^{(0)})$ does not depend on β and hence has no impact on optimizing $R(\mathbf{b})$. Differentiating $R(\mathbf{b})$ with respect to $\mathbf{b}$ and setting it to zero, we have the following equation:

$$\mathbf{U}^T\mathbf{U}\mathbf{b} + n\gamma\mathbf{V}\mathbf{b} + n\mathbf{W}^{(0)}\mathbf{b} = \mathbf{U}^T\mathbf{y},$$

with solution

$$\hat{\mathbf{b}} = \left(\mathbf{U}^T\mathbf{U} + n\gamma\mathbf{V} + n\mathbf{W}^{(0)}\right)^{-1}\mathbf{U}^T\mathbf{y}.$$

We repeat the above computation steps until $\hat{\mathbf{b}}$ converges. It is known that LQA with finite iterations does not yield sparse estimates. To address this numerical issue, as suggested by Fan and Li (2001), we cut off variables whose absolute values are smaller than a predefined threshold τ after convergence is declared. Recommended value for τ is $\tau = 10^{-4}\|\tilde{\beta}\|_2/T$, that is, the relative order of τ to the average L^2 norm of β on $[0, T]$ is 10^{-4}, where $\tilde{\beta}$ is some consistent estimator of β, such as $\hat{\beta}^{\text{smooth}}$. In general, the result is not quite sensitive to τ in a rather large range. Detailed discussion of the effect of τ is provided in the supplementary file.

In summary, we have the following algorithm to compute $\hat{\mathbf{b}}$ and obtain the estimator $\hat{\beta}(t) = \mathbf{B}^T(t)\hat{\mathbf{b}}$.

Step 1: Compute the initial estimate $\hat{\mathbf{b}}^{(0)} = \left(\mathbf{U}^T\mathbf{U} + n\gamma\mathbf{V}\right)^{-1}\mathbf{U}^T\mathbf{y}$.

Step 2: Given $\hat{\mathbf{b}}^{(i)}$, compute $\mathbf{W}^{(i)}$ and $\hat{\mathbf{b}}^{(i+1)} = \left(\mathbf{U}^T\mathbf{U} + n\gamma\mathbf{V} + n\mathbf{W}^{(i)}\right)^{-1}\mathbf{U}^T\mathbf{y}$. In practice, if a variable is too small in magnitude and makes $\mathbf{U}^T\mathbf{U} + n\gamma\mathbf{V} + n\mathbf{W}^{(i)}$ almost singular or badly scaled so that inverting $\mathbf{U}^T\mathbf{U} + n\gamma\mathbf{V} + n\mathbf{W}^{(i)}$ is numerically unstable, then the variable is shrunk to zero.

Step 3: Repeat Step 2 until the convergence of $\hat{\mathbf{b}}$ is reached. The final estimate of $\mathbf{b}$ is obtained by setting elements in $\hat{\mathbf{b}}$ whose absolute values are smaller than τ to zero.

To carry out the above algorithm, one needs to evaluate the matrix $\mathbf{U}$ with elements $u_{ij} = \int_0^T X_i(t)B_j(t)dt$. When covariate functions $X_i(t)$ are not fully observed, approximation of elements u_{ij} is mandatory. Provided that covariate functions $X_i(t)$ are observed at a regular and dense grid of time points $t_1, t_2, \ldots, t_K$ over $[0, T]$, we can compute $u_{ij} \approx \frac{1}{K}\sum_{k=1}^{K} X_i(t_k)B_j(t_k)$. An alternative approach is to find a fitted smooth curve $\hat{X}_i(t)$ for each $X_i(t)$ using spline regression method based on discrete observations $X_i(t_1), X_i(t_2), \ldots, X_i(t_K)$, and then compute $u_{ij} = \int_0^T \hat{X}_i(t)B_j(t)dt$. When the grid is regular and dense, these two approaches yield almost identical approximates of u_{ij}. When $X_i(t)$ is observed on a sparse and irregular grid, $X_i(t)$ itself can be estimated from sparse measurements using PACE procedure developed by Yao, Müller, and Wang (2005).

In penalized spline method, the choice of M is not crucial (Cardot, Ferraty, and Sarda 2003), as the roughness of estimator is controlled by a roughness penalty, rather than the number of knots. In practice, M is chosen to be large enough so that local features of $\beta(t)$ can be captured. This rule applies to our SLoS estimator. In particular, we require M to be large enough to be able to capture null subregions of $\beta(t)$. A simulation study detailed in Section C in the supplementary file provides a practical guideline on choosing M: estimators with a larger M perform better in identifying null region of $\beta(t)$, while those with a smaller M perform slightly better in prediction. For determining tuning parameters M, γ, and λ, one can simply fix a large M and then search for optimal γ and λ based on some popular selection criteria such as cross-validation, generalized cross-validation, Bayesian information criterion (BIC), Akaike information criterion (AIC), or risk inflation criterion (RIC). More sophisticatedly, one can search for optimal M, γ, and λ simultaneously over a grid of candidate values based on one of the aforementioned selection criteria.

When $\mu \neq 0$, we modify the matrices $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}^{(i)}$ as follows to produce an estimate of $\hat{\mu}$ simultaneously: (1) add one more parameter μ to the topmost of the column vector $\mathbf{b}$; (2) add one more column of all ones to the leftmost of $\mathbf{U}$; (3) add one column of all zeros to the leftmost of both $\mathbf{V}$ and $\mathbf{W}^{(i)}$, and then add one more row of all zeros to the topmost of both $\mathbf{V}$ and $\mathbf{W}^{(i)}$. With these changes, the algorithm above can be carried out to estimate $\beta(t)$ and μ simultaneously.

2.4 Extension to Multiple Regressors

Our method can be extended to estimate the following functional linear model with multiple functional covariates:

$$Y_i = \mu + \sum_{k=1}^{K}\int_0^T X_{ki}(t)\beta_k(t)dt + \varepsilon_i, \tag{15}$$

where Y_i is scalar response, μ is the grand mean, $X_{k1}(t), \ldots, X_{kn}(t)$ are independent realizations of an unknown centered stochastic process $X_k(t)$ defined on the domain $[0, T]$, and $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma_\varepsilon^2)$, $i = 1, \ldots, n$. The SLoS estimates of coefficient functions $\beta_k(t), k = 1, \ldots, K$, and μ are obtained by minimizing

$$Q(\boldsymbol{\beta}, \mu) = \frac{1}{n}\sum_{i=1}^{n}\left[Y_i - \mu - \sum_{k=1}^{K}\int_0^T X_{ki}(t)\beta_k(t)dt\right]^2 + \sum_{k=1}^{K}\gamma_k\|\mathcal{D}^m\beta_k\|^2 + \sum_{k=1}^{K}\frac{M}{T}\int_0^T p_{\lambda_k}(|\beta_k(t)|)dt, \quad (16)$$

where $\boldsymbol{\beta}(t) = (\beta_1(t), \beta_2(t), \ldots, \beta_K(t))^T$. The computational steps described in Section (2.3) are modified as follows to accommodate estimating multiple coefficient functions. Corresponding to each regressor $X_{k\cdot}$, we compute a matrix $\mathbf{U}_k$, which has $n \times (M + d)$ entries $u_{ij} = \int_0^T X_{ki}(t)B_j(t)dt$ for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, (M + d)$. Let $\mathbf{U} = (\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_K)$ be the column catenation of $\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_K$, and correspondingly set $\mathbf{V} = \text{diag}(\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_K)$, that is, set $\mathbf{V}$ to be the matrix with blocks $\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_K$ in its main diagonal and zeros elsewhere. For each k, we also have a matrix $\mathbf{W}^{(0,k)}$, which corresponds to the matrix $\mathbf{W}^{(0)}$ in (12). Then $\mathbf{W}^{(0)}$ in (14) is replaced by $\mathbf{W}^{(0)} = \text{diag}(\mathbf{W}^{(0,1)}, \mathbf{W}^{(0,2)}, \ldots, \mathbf{W}^{(0,K)})$. After these modifications, the iterative algorithm described in Section 2.3 can be carried out to estimate $\beta_1, \beta_2, \ldots, \beta_K$ simultaneously.

3. Theoretical Properties

Let $N(\beta)$ denote the null region of $\beta(t)$ and $S(\beta)$ denote the nonnull region of $\beta(t)$, that is, $N(\beta) = \{t \in [0, T] : \beta(t) = 0\}$ and $S(\beta) = \{t \in [0, T] : \beta(t) \neq 0\}$. We show that, under some regularity conditions, our SLoS estimator $\hat{\beta}(t)$ enjoys the oracle property for identifying the null region $N(\beta)$ and at the same time estimating $\beta(t)$ on $S(\beta)$ with an optimal pointwise convergence rate. In other words, our estimator performs as well as if the true null region $N(\beta)$ was known.

The following regularity conditions are assumed:

(C1) $\|X\|_2$ is almost surely bounded, that is, $\|X\|_2 \le c_1 < \infty$ a.s for some constant $c_1 > 0$. Here, $\|X\|_2$ denotes $(\int_0^T X^2(t)dt)^{1/2}$.

(C2) $\beta(t)$ is in the Hölder space $C^{p',\nu}[S(\beta)]$. That is, $|\beta^{(p')}(u_1) - \beta^{(p')}(u_2)| \le c_1|u_1 - u_2|^\nu$ for some constant $c_2 > 0$, integer p' and $\nu \in [0, 1]$, and for all $u_1, u_2 \in S(\beta)$. Let $p \overset{\text{def}}{=} p' + \nu$. Assume $3/2 < p \le d$, where d is the degree of the B-spline basis.

(C3) There exists a sequence of λ_n depending on sample size n such that $\lambda_n \to 0$, $\sqrt{\int_{S(\beta)} p'_\lambda(|\beta(t)|)^2\,dt} = O(n^{-1/2}M^{-3/2})$ and $\sqrt{\int_{S(\beta)} p''_{\lambda_n}(|\beta(t)|)^2 dt} = o(M^{-3/2})$.

In the above, (C1) and (C2) are the same as (H1) and (H3) of Cardot, Ferraty, and Sarda (2003). Additionally, assumption (C3) is analogous to regularity conditions (B′) and (C′) in Fan and Peng (2004) to ensure the unbiasedness and guarantee that the penalty does not dominate the least squares. Essentially, (C3) implies that λ_n should not decay too lowly, since quantities $\sqrt{\int_{S(\beta)} p'_\lambda(|\beta(t)|)^2\,dt}$ and $\sqrt{\int_{S(\beta)} p''_{\lambda_n}(|\beta(t)|)^2 dt}$ decreases to zero as λ_n goes to zero. For example, if the domain is $[0, 10]$ and $\beta(t) = (\max\{x - 5, 0\})^{1/4}$, then $\lambda_n = o(\min\{M^{-3/4}, n^{-1/5}M^{-3/5}\})$ satisfies (C3).

To state the next regularity condition, we define Γ as the covariance operator of the random process X, that is, $(\Gamma x)(t) = \int_0^T \mathbf{E}(X(s)X(t))x(s)ds$. Let Γ_n denote the empirical version of the covariance operator Γ of X. That is,

$$(\Gamma_n x)(v) \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n}\int_0^T X_i(v)X_i(u)x(u)du.$$

Let $\mathbf{H}$ be the $n \times M$ matrix with element $h_{i,j} = \langle \Gamma_n B_i, B_j\rangle$. Let $\lambda_{\min}(\mathbf{H})$ and $\lambda_{\max}(\mathbf{H})$ denote the minimum and maximum eigenvalues of $\mathbf{H}$, respectively. The following condition is the same as the condition A_8 in Zhou, Wang, and Wang (2013). It is essential for establishing the oracle property of the estimator proposed in Zhou, Wang, and Wang (2013).

(C4) $\lambda_{\min}(\mathbf{H})/M^{-1}$ and $\lambda_{\max}(\mathbf{H})/M^{-1}$ are bounded away from 0 and ∞ with probability tending to one as $n \to \infty$.

Recall that γ and λ are tuning parameters varying with n. To emphasize their dependency on the sample size n, we denote them by γ_n and λ_n, respectively. In addition, we assume the following conditions on choosing values of M, γ_n, and λ_n:

(C5) $M = o(n^{1/2})$, $M = \omega(n^{1/(2p-1)})$, $\gamma_n = o(n^{-1/2})$, $\lambda_n = o(1)$, and $\lambda_n n^{1/2}M^{-1/2} \to \infty$, where $M = \omega(n^{1/(2p-1)})$ means $M/n^{1/(2p-1)} \to \infty$ as $n \to \infty$.

Our first theoretical result, which is stated in Theorem 2, shows that with probability tending to one, there exists a local solution of (7) that converges to the "best" B-spline function from the family $\mathcal{S}_{dM}$ in approximating $\beta(t)$. More precisely, according to Theorem XII(6) of de Boor (2001), there exists some $\alpha(t) \overset{\text{def}}{=} \sum_{j=1}^{M+d} b_{\alpha j}B_j(t) \overset{\text{def}}{=} \mathbf{B}^T(t)\mathbf{b}_\alpha \in \mathcal{S}_{dM}$ such that $\|\alpha - \beta\|_\infty \le C_1M^{-p}$ for some constant $C_1 \ge 0$. Moreover, it is clear that the coefficient $b_{\alpha j}$ can be chosen to be zero if the support of the corresponding basis function $B_j(t)$ is contained in the null region $N(\beta)$. A less obvious fact is that we can also choose $\alpha(t)$ such that $\|\mathbf{b}_\alpha\|_\infty \le \|\beta\|_\infty$. A detailed discussion of this fact is provided in Section B of the supplementary file. The following theorem exactly assures the existence of a local solution $\hat{\beta}(t) = \mathbf{B}^T(t)\hat{\mathbf{b}}$ of (7) with probability tending to one, such that $\hat{\mathbf{b}}$ converges to $\mathbf{b}_\alpha$ in a rate bounded by $O_P(n^{-1/2}M^{1/2})$. This result also leads to convergence rates of $\hat{\beta}$ in terms of both supremum norm and seminorm induced by Γ. The seminorm $\|\beta\|_\Gamma$ of β is defined by $\|\beta\|_\Gamma = \{\int_0^T (\Gamma\beta)(t)\beta(t)dt\}^{1/2}$.

Theorem 2 (Existence of SLoS Estimator). Under conditions (C1)–(C5), with probability tending to one, there exists a local minimizer $(\hat{\beta}, \hat{\mu})$ of (7) such that $\hat{\beta}(t) = \mathbf{B}^T(t)\hat{\mathbf{b}}$, $\|\hat{\mathbf{b}} - \mathbf{b}_\alpha\|_2 = O_P(n^{-1/2}M^{1/2})$, and $|\hat{\mu} - \mu| = O_P(n^{-1/2})$. Therefore, $\|\hat{\beta} - \beta\|_\infty = O_P(n^{-1/2}M^{1/2})$ and $\|\hat{\beta} - \beta\|_\Gamma = O_P(n^{-1/2}M^{1/2})$.

According to the above theorem, we have a uniformly root-n/γ_nM consistent local minimizer $\hat{\beta}(t)$ and a root-n consistent

local minimizer $\hat{\mu}$. This local minimizer $(\hat{\beta}, \hat{\mu})$ is taken as our SLoS estimator for $\beta(t)$ and the intercept μ. The estimator $\hat{\beta}(t)$ has an asymptotic property that corresponds to the oracle property reported in Zhou, Wang, and Wang (2013).

We now introduce some notations for stating the oracle property. First, recall that the support of $\beta(t)$ is defined as the closure of $S(\beta)$. For any $\epsilon > 0$ and a subset of the real line A, the ϵ-neighborhood of A, defined by $\{t \in [0, T] : \inf_{u \in A} |t - u| < \epsilon\}$, is denoted by A^{ϵ}. Also, when we say a sequence of subsets A_n of the real line converges to a subset F, we mean the Lebesgue measure of the symmetric difference of A_n and F converges to zero. Let D denote the interval $[0, T]$. Given a λ_n and M, we divide D into two parts: the first part $D_1 = \{t \in D : |\beta(t)| \geq aC_4(\lambda_n + M^{-p})\}$ for some constant $C_4 > 0$ that is determined later in the supplementary file, and the second part $D - D_1$. Let $\mathbf{B}_1$ be the sub-vector of $\mathbf{B}$ such that each $B_j(t)$ in $\mathbf{B}_1$ has a support inside D_1. Correspondingly, let $\mathbf{U}_1$ be the columns of $\mathbf{U}$ associated with $\mathbf{B}_1$, where $\mathbf{U}$ is defined in Section 2.3.

Theorem 3 (Oracle Property). Under conditions (C1)–(C5), as $n \to \infty$:

(i) For every t not in the support of $\beta(t)$, we have $\hat{\beta}(t) = 0$ with probability tending to one. Moreover, for every $\epsilon > 0$, we have $N^{\circ}(\beta) \subset N^{\epsilon}(\hat{\beta})$ with probability tending to one, where $N^{\circ}(\beta)$ denotes the interior of $N(\beta)$. Thus, $N(\hat{\beta}) \to N(\beta)$ and $S(\hat{\beta}) \to S(\beta)$ in probability.

(ii) For every t such that $\beta(t) \neq 0$, we have

$$\left(\frac{n}{M}\right)^{1/2} \left[\hat{\beta}(t) - \beta(t)\right] \xrightarrow{d} N(0, \sigma^2(t)),$$

where

$$\sigma^2(t) = \lim_{n \to \infty} \frac{n}{M} \mathbf{B}_1^T(t) \left(\mathbf{U}_1^T \mathbf{U}_1\right)^{-1} \mathbf{B}_1(t).$$

(iii) For the estimated intercept $\hat{\mu}$, we have

$$\sqrt{n}(\hat{\mu} - \mu) \xrightarrow{d} N\left(0, \sigma_{\varepsilon}^2\right).$$

In the above theorem, the first statement of (i) says that with probability tending to one, $\hat{\beta}(t)$ is pointwise zero on the null region except for those points on the boundaries. The second statement of (i) effectively states that the null region identified by our SLoS estimator is consistent with the true null region $N(\beta)$. This conclusion is much stronger than the pointwise consistency in the first statement.

4. Simulation Studies

To investigate numerical performance of the proposed SLoS estimator, we conduct a simulation study on the following functional linear model

$$Y_i = \mu + \int_0^1 X_i(t)\beta(t)dt + \varepsilon_i, \tag{17}$$

where $\varepsilon_i \sim N(0, \sigma_{\varepsilon}^2)$. Four different types of coefficient functions $\beta(t)$ are considered:

Case I: There is no signal, that is, $\beta(t) \equiv 0$.

Case II: There is a flat region of $\beta(t)$. We consider the following function

$$\beta(t) = \begin{cases} 2(1-t)\sin(2\pi(t+0.2)) & 0 \leq t \leq 0.3, \\ 0 & 0.3 < t < 0.7, \\ 2t\sin(2\pi(t-0.2)) & 0.7 \leq t \leq 1, \end{cases}$$

which vanishes on $[0.3, 0.7]$. The function is plotted in Figure 2(a).

Case III: There is no flat region of $\beta(t)$. We consider $\beta(t) = 7x^3 + 2\sin(4\pi t + 0.2)$. It is plotted in Figure 2(b). It is designed so that it has no flat region, but crosses zero twice. Therefore, this case does not favor the SLoS method.

Case IV: The true $\beta(t)$ is never close to zero. We consider $\beta(t) = 4\sqrt{t} + e^{x^2}\cos(3\pi t)$. It is plotted in Figure 2(c). It is designed so that it does not cross zero, and hence shrinkage is unnecessary. As case III, this case does not favor the SLoS method.

The measurement error σ_{ε} is set to 1 in Case I, and chosen so that signal-to-noise ratio equals to 4 in the other cases. The true μ is set to 1. The covariate functions $X_i(t)$ are generated based on equation $X_i(t) = \sum a_{ij}B_j(t)$, where the coefficients a_{ij} are generated from the standard normal distribution, and each $B_j(t)$ is a B-spline basis function defined by order 5 and 71 equally spaced knots (the number 71 is randomly selected between 50 and 100). Using this setup, for each sample size $n = 150, 450, 1000$, we independently generate 100 datasets. For each dataset, we also generate a separate test dataset with sample size equal to 5000.

The quality of estimates is measured by the integrated squared error (ISE), which is defined on null subregions and nonnull subregions, respectively, by

$$\text{ISE}_0 = \frac{1}{\ell_0} \int_{N(\beta)} \left(\hat{\beta}(t) - \beta(t)\right)^2 dt, \text{ and}$$
$$\text{ISE}_1 = \frac{1}{\ell_1} \int_{S(\beta)} \left(\hat{\beta}(t) - \beta(t)\right)^2 dt, \tag{18}$$

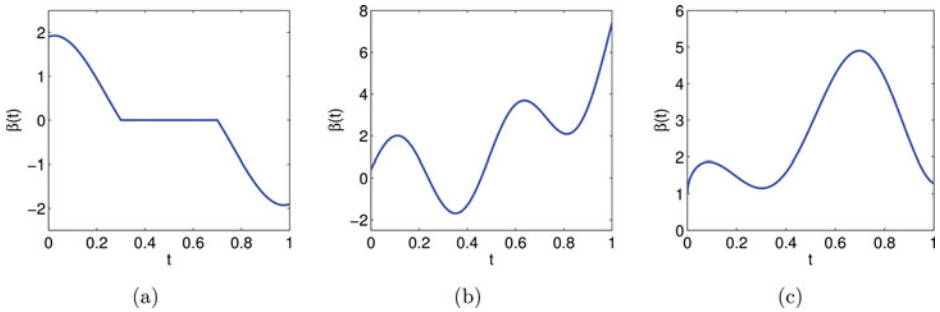

Figure 2. (a) The true $\beta(t)$ in Case II. (b) The true $\beta(t)$ in Case III. (c) The true $\beta(t)$ in Case IV.

where ℓ_0 is the length of null subregions and ℓ_1 is the length of nonnull subregions of $\beta(t)$. ISE_0 and ISE_1 measure integrated squared errors between an estimated coefficient function $\hat{\beta}(t)$ and the true function $\beta(t)$ on null subregions and nonnull subregions, respectively. In addition, performance of estimators on prediction is assessed by prediction mean squared errors (PMSE) on a test dataset that is independent of training data. The PMSE is computed as follows:

$$\text{PMSE} = \frac{1}{N} \sum_{(X,y)\in \text{test}} \left(y - \hat{\mu} - \int_0^T X(t)\hat{\beta}(t)dt \right)^2, \quad (19)$$

where test denotes the test dataset, N is the size of test dataset, and $(\hat{\mu}, \hat{\beta}(t))$ is the estimated intercept μ and coefficient function $\beta(t)$ from training data.

To gain insight into numerical performance of the proposed method, we compare it to an oracle estimator that assumes true null subregions of $\beta(t)$ are known in advance. Taking advantage of practically inaccessible knowledge of null subregions of $\beta(t)$, the oracle estimator ought to substantially outperform any other practical estimators that do not have access to the oracle knowledge. On the other hand, we will demonstrate that the proposed SLoS estimator achieves a performance competitive to the oracle estimator in terms of ISE_0, ISE_1, and PMSE in all simulation setups. As comparison in the studies, nonsparse estimators such as ordinary least-square estimator (OLS), smoothing spline estimator, and estimator produced by the popular principal component regression method (PCR) are significantly inferior in terms of at least one of the measures ISE_0, ISE_1, and PMSE. In comparison with the aforementioned competing methods FLiRTI in James, Wang, and Zhu (2009) and the two-stage method in Zhou, Wang, and Wang (2013), the proposed SLoS estimator is numerically advantageous in Cases II, III, and IV, while is comparable in Case I.

Estimators except FLiRTI are computed by cubic B-spline bases. The oracle estimator is computed by the smoothing spline method, where knots to define B-spline basis functions are evenly placed only on *nonnull* subregions of $S(\beta)$, to respect the fact that the null subregions of $\beta(t)$ are not known in advance. Therefore, the oracle estimator is always identically zero over null subregions of the true $\beta(t)$. In contrast, knots to define B-spline basis functions for OLS, smoothing spline, and SLoS are evenly placed on the *entire* domain $[0, 1]$, to respect the fact that they do not know the null subregions of $\beta(t)$ in advance. The OLS estimator is equivalent to the estimator obtained by setting $\gamma = 0$ and $\lambda = 0$ in (7), while the smoothing spline estimator proposed in Cardot, Ferraty, and Sarda (2003), can be obtained by setting $\lambda = 0$ in (7). The tuning parameters, such as M, γ, and λ of SLoS estimator are chosen by the procedure reported in Section 2.3, with BIC as selection criterion because BIC encourages sparse models. For the OLS estimator, the smoothing spline estimator and the oracle estimator, tuning parameters are chosen by AIC, AIC and cross-validation (CV), respectively. This is because, our experiments show that AIC and CV yield slightly better performance than BIC for these estimators in terms of ISE_1 and PMSE, while the performance of these selection criteria on ISE_0 is indistinguishable. Both the FLiRTI and two-stage estimators are implemented and tuned according to James, Wang, and Zhu (2009) and Zhou, Wang, and Wang (2013), respectively.

Table 1 displays a summary of ISE_0 of the estimators. Note that the ISE_0 of the oracle estimator is always zero, since it assumes that null subregions of $\beta(t)$ are known. It is not surprising that the SLoS estimator has a substantially smaller error than nonsparse estimators on null subregions, since fSCAD shrinks estimates of SLoS toward zero. It is remarkable that the ISE_0 of SLoS estimator is quite close to zero, particularly when the sample size is relatively large. This shows that the SLoS estimator is competitive to the oracle estimator for estimating $\beta(t)$ over null subregions. On the other hand, the performance of the FLiRTI estimator on null subregions is significantly inferior to the SLoS and two-stage estimators. Moreover, the FLiRTI estimator exhibits larger variability. This is because, as pointed out by Zhou, Wang, and Wang (2013), FLiRTI requires a large number of parameters (coefficients to basis functions) to estimate $\beta(t)$, which leads to large variation of estimation. For the two-stage estimator, it performs slightly better than the proposed SLoS estimator in Case I where there is no signal, but much worse than SLoS in Case II where there is a mix of nonnull and null subregions.

Table 2 summarizes ISE_1 of the estimators. It is observed that when the sample size is small, the performance of SLoS is indeed better than nonsparse estimators and comparable to the performance of the oracle estimator, even in Case III where the setup of $\beta(t)$ does not favor the proposed estimator. One explanation for this phenomenon is that, the shrinkage effect of fSCAD penalty reduces the variability of SLoS estimator without causing a significant bias. This reduction of variability is significant when the sample size is relatively small, and diminishes as the sample size grows, as observed by comparing ISE_1 of the smoothing spline estimator and SLoS estimator side by side in Table 2. When the true $\beta(t)$ is never close to zero as in Case IV, because of the roughness penalty term in (7), the SLoS estimator behaves like the smoothing spline estimator that is equivalent to the SLoS estimator when $\lambda = 0$. In contrast, without the roughness penalty, the errors of the FLiRTI and two-stage estimators on nonnull subregions are similar to the OLS estimator and larger than the SLoS estimator.

Table 3 presents PMSE of each method. In Cases I and II where locally sparse estimators are in favor, the SLoS estimator reduces PMSE substantially over nonsparse estimators, and achieves a competitive performance as the oracle estimator. A more remarkable observation is that, in Case III that does not favor locally sparse estimators, the SLoS estimator slightly outperforms nonsparse estimators. In particular, when the sample size is relatively small, the advantage of SLoS on PMSE is substantial. As the explanation we suggest for a similar observation on ISE_1 in Table 2, the nice shrinkage property of fSCAD penalty is credited for this phenomenon. In contrast, FLiRTI outperforms nonsparse estimators only in Case I where there is no signal, but hindered by excessive variability, it is significantly inferior to all other estimators in most other cases, particularly Cases III and IV where the true $\beta(t)$ has no flat subregion. For the two-stage estimator, although it does not suffer from excessive variability, due to a lack of regularization on roughness, it incurs larger PMSE than SLoS in Cases II and III, and in Case IV when the sample size is relatively small.

We also investigate the ability of the SLoS method to identify null subregions. Figure 3(a) displays the estimated $\hat{\beta}(t)$

Table 1. The integrated squared error, ISE_0, defined on the null region for estimators using six methods: the ordinary least-square method (abbreviated as OLS), the smoothing spline method (abbreviated as Smooth), the principal component regression method (abbreviated as PCR), the FLiRTI method, the two-stage method proposed by Zhou, Wang, and Wang (2013) (abbreviated as Two-Stage), and our SLoS method. ISE_0 is defined in (18). Each entry is the Monte Carlo average of 100 simulation replicates. The corresponding Monte Carlo standard deviation is included in parentheses.

	OLS	Smooth	PCR	FLiRTI	Two-stage	SLoS
Case I						
$n = 150$	1.65 (1.67)	0.57 (0.72)	0.16 (0.13)	0.15 (0.12)	0.05 (0.35)	0.06 (0.31)
$n = 450$	0.42 (0.41)	0.18 (0.18)	0.09 (0.69)	0.04 (0.02)	0.00 (0.00)	0.01 (0.10)
$n = 1000$	0.20 (0.22)	0.08 (0.08)	0.04 (0.03)	0.00 (0.01)	0.00 (0.00)	0.00 (0.00)
Case II ($\times 10^{-3}$)						
$n = 150$	19.75 (18.15)	20.09 (18.20)	18.34 (11.69)	2.29 (3.75)	1.71 (1.29)	0.15 (0.32)
$n = 450$	7.34 (4.62)	6.37 (3.31)	7.67 (4.70)	0.88 (1.16)	0.43 (1.28)	0.04 (0.09)
$n = 1000$	4.23 (3.10)	4.00 (2.58)	4.92 (2.63)	0.50 (0.50)	0.33 (0.32)	0.01 (0.03)

Table 2. The integrated squared error, ISE_1, defined on the nonnull region for estimators using seven methods: the oracle method, the ordinary least-square method (abbreviated as OLS), the smoothing spline method (abbreviated as Smooth), the principal component regression method (abbreviated as PCR), the FLiRTI method, the two-stage method proposed by Zhou, Wang, and Wang (2013) (abbreviated as Two-stage), and our SLoS method. ISE_1 is defined in (18). Each entry is the Monte Carlo average of 100 simulation replicates. The corresponding Monte Carlo standard deviation is included in parentheses.

	Oracle	OLS	Smooth	PCR	FLiRTI	Two-stage	SLoS
Case II ($\times 10^{-2}$)							
$n = 150$	1.93 (1.18)	4.13 (3.56)	3.11 (2.68)	4.10 (2.70)	4.59 (2.32)	4.32 (2.18)	2.51 (1.60)
$n = 450$	0.73 (0.44)	1.23 (0.88)	0.88 (0.47)	1.50 (0.78)	1.41 (0.73)	1.24 (0.72)	0.86 (0.44)
$n = 1000$	0.37 (0.18)	0.66 (0.39)	0.46 (0.22)	0.86 (0.43)	0.63 (0.28)	0.57 (0.35)	0.46 (0.19)
Case III ($\times 10^{-1}$)							
$n = 150$	—	1.88 (1.55)	1.64 (1.35)	2.50 (1.31)	2.11 (1.09)	1.92 (0.95)	1.32 (0.61)
$n = 450$	—	0.52 (0.36)	0.46 (0.25)	0.65 (0.38)	0.79 (0.42)	0.65 (0.27)	0.41 (0.17)
$n = 1000$	—	0.28 (0.20)	0.22 (0.12)	0.38 (0.14)	0.41 (0.19)	0.31 (0.13)	0.20 (0.08)
Case IV ($\times 10^{-1}$)							
$n = 150$	—	2.23 (2.01)	1.76 (1.81)	2.78 (1.43)	2.32 (1.34)	2.46 (1.28)	1.71 (0.82)
$n = 450$	—	0.51 (0.29)	0.45 (0.24)	0.90 (0.39)	0.89 (0.54)	0.82 (0.28)	0.43 (0.18)
$n = 1000$	—	0.31 (0.26)	0.27 (0.16)	0.45 (0.20)	0.49 (0.21)	0.41 (0.17)	0.24 (0.09)

Table 3. The prediction mean squared error (PMSE) on test data using seven methods: the oracle method, the ordinary least-square method (abbreviated as OLS), the smoothing spline method (abbreviated as Smooth), the principal component regression method (abbreviated as PCR), the FLiRTI method, the two-stage method proposed by Zhou, Wang, and Wang (2013) (abbreviated as Two-stage), and our SLoS method. PMSE is defined in (19). Each entry is the Monte Carlo average of 100 simulation replicates. The corresponding Monte Carlo standard deviation is included in parentheses.

	Oracle	OLS	Smooth	PCR	FLiRTI	Two-stage	SLoS
Case I ($\times 10^{-2}$)							
$n = 150$	—	2.10 (1.61)	0.91 (0.92)	0.39 (0.27)	0.95 (0.76)	0.19 (0.77)	0.23 (0.43)
$n = 450$	—	0.57 (0.46)	0.29 (0.24)	0.17 (0.12)	0.21 (0.33)	0.04 (0.06)	0.05 (0.11)
$n = 1000$	—	0.27 (0.22)	0.12 (0.10)	0.08 (0.05)	0.07 (0.12)	0.02 (0.03)	0.02 (0.03)
Case II ($\times 10^{-4}$)							
$n = 150$	1.62 (0.89)	3.89 (2.39)	3.47 (2.18)	4.42 (2.93)	3.85 (3.72)	3.61 (2.34)	2.11 (1.25)
$n = 450$	0.58 (0.32)	1.28 (0.58)	1.06 (0.40)	1.53 (0.89)	1.30 (1.01)	1.10 (0.90)	0.72 (0.36)
$n = 1000$	0.28 (0.14)	0.67 (0.31)	0.57 (0.24)	1.02 (0.38)	0.66 (0.73)	0.45 (0.48)	0.37 (0.16)
Case III ($\times 10^{-3}$)							
$n = 150$	—	2.25 (1.42)	2.12 (1.36)	2.99 (1.62)	2.42 (1.72)	2.17 (1.04)	1.83 (0.78)
$n = 450$	—	0.64 (0.35)	0.61 (0.28)	0.86 (0.59)	0.78 (0.61)	0.71 (0.28)	0.56 (0.21)
$n = 1000$	—	0.35 (0.20)	0.30 (0.15)	0.46 (0.22)	0.42 (0.26)	0.35 (0.14)	0.28 (0.11)
Case IV ($\times 10^{-3}$)							
$n = 150$	—	2.74 (1.88)	2.31 (1.83)	3.19 (1.81)	2.45 (2.03)	2.80 (1.41)	2.39 (1.07)
$n = 450$	—	0.68 (0.34)	0.63 (0.31)	0.98 (0.82)	0.80(0.76)	0.67 (0.32)	0.62 (0.26)
$n = 1000$	—	0.40 (0.26)	0.36 (0.20)	0.65 (0.30)	0.53 (0.31)	0.43 (0.18)	0.35 (0.13)

by the SLoS, OLS, and smoothing spline method in one random simulation replicate, as well as the true $\beta(t)$. The estimated $\hat{\beta}(t)$ by the oracle procedure is not displayed, as it is almost identical to the true $\beta(t)$. Figure 3(a) shows that all three methods produce a very good estimate of $\beta(t)$. However, if we take a close look at estimates over the null subregion [0.3, 0.7], as displayed in Figure 3(b), we see that the SLoS estimate is identically zero, while the other two estimates are not. Similar results are observed in most other simulation replicates. This suggests that the SLoS estimator can locate null subregions accurately.

To further quantify the ability of identifying null subregions, we calculate the average of proportions of null subregions that are *correctly* identified by the SLoS estimator. The quantity is computed as follows. First, for each run, we compute the value of $\hat{\beta}(t)$ on a sequence of dense and equally spaced points in the null subregion. For example, in Case I, the sequence is taken to be 0, 0.001, 0.002, . . . , 0.999, 1, and in Case II, it is taken to be 0.3, 0.301, 0.302, . . . , 0.7. Then we compute the proportion of the points at which $\hat{\beta}(t)$ is zero among all points in the sequence. Finally, we average the calculated proportions from 100 runs. The average of proportions of *falsely* identified null subregions

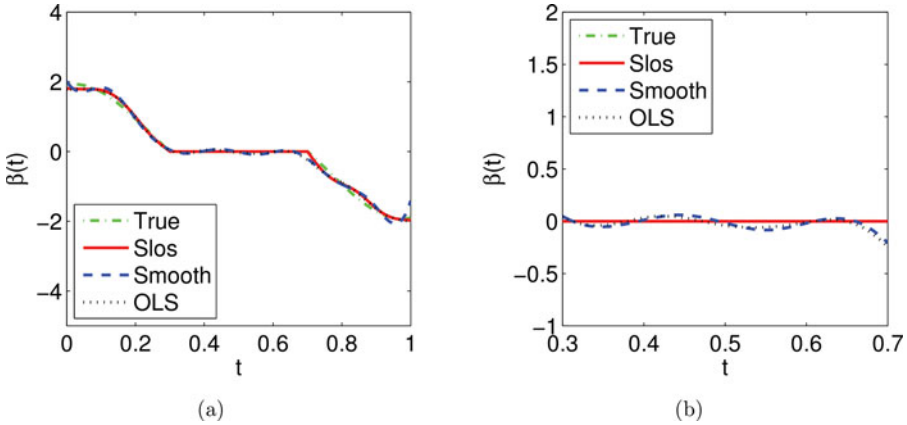

Figure 3. (a) Estimates of $\hat{\beta}(t)$ in a simulation replicate chosen randomly in the simulation. (b) The estimates of $\hat{\beta}(t)$ in (c) over the null region [0.3, 0.7] of $\beta(t)$.

is also computed in a similar fashion, except that a sequence of dense and equally spaced points are placed in the nonnull subregions. The results are summarized in Tables 4 and 5, respectively. It shows that the performance of our estimator on identifying null subregions is quite impressive: in Cases I and II, on average more than 95% and 92% of the null subregion is correctly identified when $n = 150$, respectively, and this number increases to 100% and 95% when $n = 1000$, respectively. Moreover, the proportion of falsely identified null subregions is quite marginal. For example, it is less than 1% in Case II and zero in Cases III and IV. This numerically confirms our claim that the SLoS estimator is free from over shrinkage. The close-to-one average true-null proportion and the negligible average false-null proportion together numerically verify the conclusion in Theorem 3. For the other two sparse estimators, FLiRTI misses a significant amount of true null subregions when the sample size is relatively small or there is a mix of null and nonnull subregions, while the two-stage estimator seems to experience a mild problem of over shrinkage when both null and nonnull subregions are present in the coefficient function $\beta(t)$, as indicated by Case II in Table 5.

Table 4. The proportions of null region that is *correctly* identified by locally sparse estimators for Case I and Case II. Each proportion is computed as the proportion of points where $\hat{\beta}(t)$ is zero on a sequence of dense and equally spaced points on null region. Each entry is the Monte Carlo average of 100 simulation replicates. The corresponding Monte Carlo standard deviation is included in parentheses.

	FLiRTI	Two-stage	SLoS
Case I (%)			
$n = 150$	78.91 (34.4)	94.40 (11.36)	98.40 (6.14)
$n = 450$	92.55 (22.1)	100.00 (0.00)	99.70 (1.75)
$n = 1000$	99.56 (15.8)	100.00 (0.00)	100.00 (0.00)
Case II (%)			
$n = 150$	53.72 (28.5)	88.02 (26.31)	92.20 (5.47)
$n = 450$	66.57 (24.5)	90.81 (10.21)	93.41 (3.27)
$n = 1000$	84.30 (15.4)	94.43 (4.53)	95.01 (2.04)

Table 5. The proportions of null region that is *falsely* identified by locally sparse estimators for Cases II–IV. Each proportion is computed as the proportion of points where $\hat{\beta}(t)$ is zero on a sequence of dense and equally spaced points on nonnull region. Each entry is the Monte Carlo average of 100 simulation replicates. The corresponding Monte Carlo standard deviation is included in parentheses.

	FLiRTI	Two-stage	SLoS
Case II (%)			
$n = 150$	0.34 (0.79)	2.72 (2.33)	0.69 (1.92)
$n = 450$	0.39 (0.76)	1.81 (1.97)	0.28 (0.64)
$n = 1000$	0.09 (0.40)	1.06 (1.32)	0.05 (0.22)
Case III (%)			
$n = 150$	0.21 (0.51)	0.05 (0.34)	0.00 (0.00)
$n = 450$	0.21 (0.44)	0.00 (0.00)	0.00 (0.00)
$n = 1000$	0.16 (0.45)	0.00 (0.00)	0.00 (0.00)
Case IV (%)			
$n = 150$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
$n = 450$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)
$n = 1000$	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)

In summary, the SLoS estimator has a competitive performance to the oracle estimator in terms of estimation of $\beta(t)$ on null and nonnull subregions, as well as prediction on new data. However, one should keep in mind that the oracle estimator assumes the true null subregions of $\beta(t)$ are known in advance, but this assumption is not valid in most practical scenarios. Moreover, even when the true $\beta(t)$ is not locally sparse, the SLoS estimator improves both estimation of $\beta(t)$ and prediction on new data over its nonsparse counterparts, particularly when the sample size is relatively small. Finally, SLoS outperforms existing competing methods by significant margins when $\beta(t)$ has both null and nonnull subregions or when $\beta(t)$ has no null subregion but crosses zero. In the case that $\beta(t)$ is constantly zero or never close to zero, SLoS and the two-stage method proposed in Zhou, Wang, and Wang (2013) have very similar performance, except that the estimation quality of SLoS on nonnull subregions is significantly better.

5. Applications

5.1 Spectrometric Data

Originating from Tecator dataset (*http://lib.stat.cmu.edu/datasets/tecator*), the data are available at *http://www.math.univ-toulouse.fr/ferraty/SOFTWARES/NPFDA/npfda-datasets.html*. There are in total 215 samples. Each sample contains finely chopped pure meat, and a spectrometric curve of spectra of absorbances measured at 100 wavelengths between 850 nm and 1050 nm is recorded. Figure 4 displays the 215 curves in the dataset. At the same time, the fat content, measured in percent, is also determined by analytic chemistry. The task is to predict the fat content based on the spectrometric curve. Our interest is to investigate what range of spectra that has *no* predicting power on fat content under the model (1). Once the range is

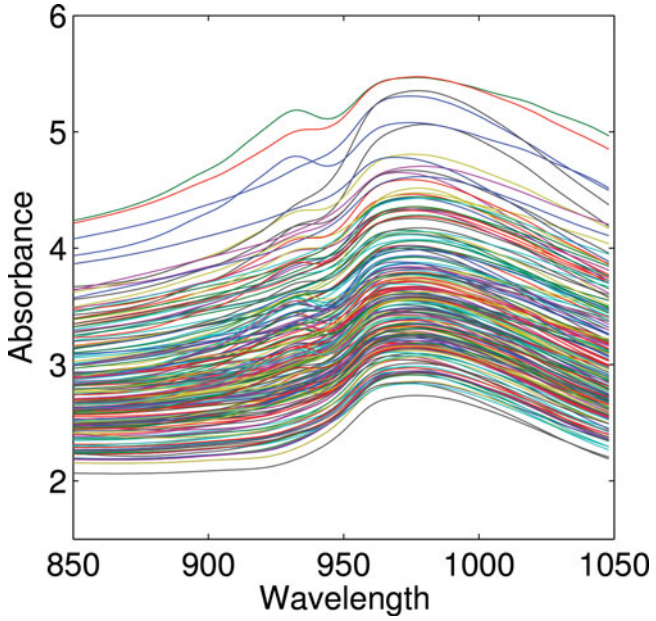

Figure 4. Spectrometric curves measured at 100 wavelengths between 850 nm and 1050 nm.

identified, when we use the model to predict fat content for a new piece of meat, there is no need to record spectra on that range. This might results in potential saving on energy, time, and money in practice.

Figure 5 shows the coefficient function estimated by our SLoS method and smoothing spline method (Cardot, Ferraty, and Sarda 2003) that both use cubic B-spline basis functions. It shows that the SLoS estimate $\hat{\beta}(t)$ is zero roughly on [970,1050]. This suggests the high end of spectrum has no contribution on predicting fat content. Our method also produces a smooth estimate of $\beta(t)$ on the nonnull subregions. It indicates spectrum channels with wavelength between 850 nm and 970 nm have significant contribution to the prediction power of the fat content.

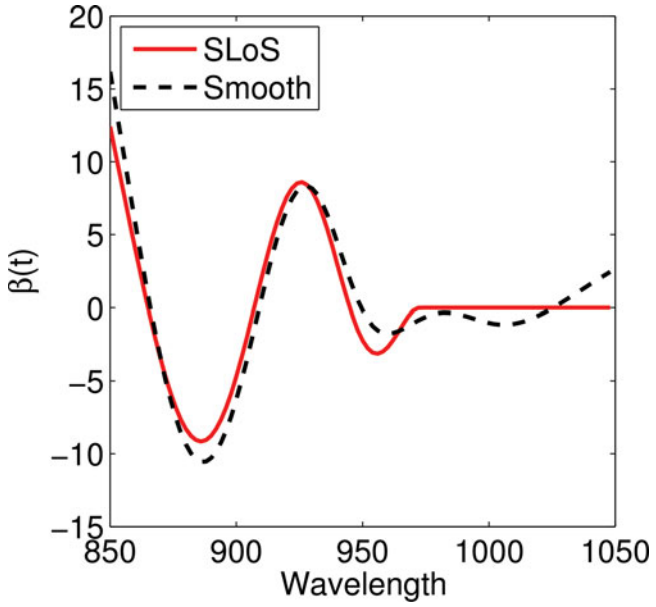

Figure 5. The estimated coefficient function $\hat{\beta}(t)$ for the functional linear model (1) from the spectrometric data using our SLoS method (solid line) and the smoothing spline method (dashed line).

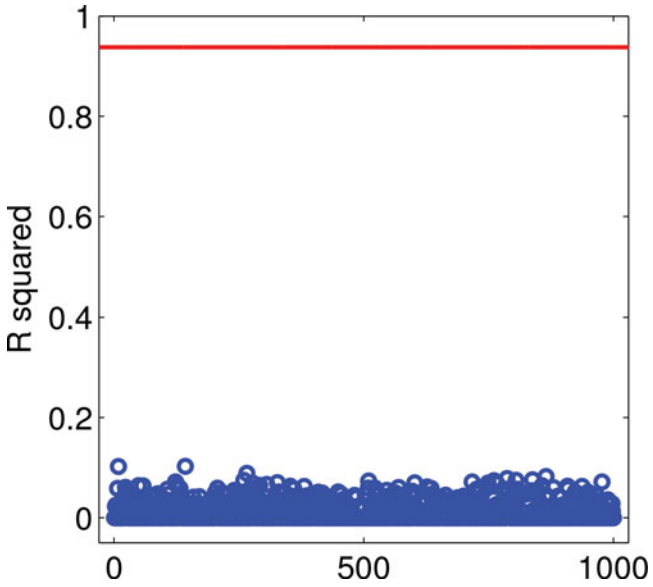

Figure 6. R^2 from permuting the response variable 1000 times in spectrometric data. The solid line represents the observed R^2 of SLoS estimate from the true data.

Figure 6 shows the results of a permutation test for SLoS estimator on this data. The solid line represents the R^2 for the SLoS estimator, which is 0.94. The responses are randomly permuted 1000 times and the new R^2 are plotted in Figure 6, indicated by circles. All 1000 permuted R^2 are under 0.2. This provides strong evidence of the relationship between spectrum and fat content discovered by our SLoS method. Since the SLoS estimate of $\beta(t)$ is zero over [970, 1050], to predict fat content according to the model (1), only spectral channels between [850,970] are required.

5.2 Canadian Weather Data

The Canadian weather data have been studied in Ramsay and Silverman (2005) and James, Wang, and Zhu (2009) with the aim to predict annual rainfall from daily mean temperature over the

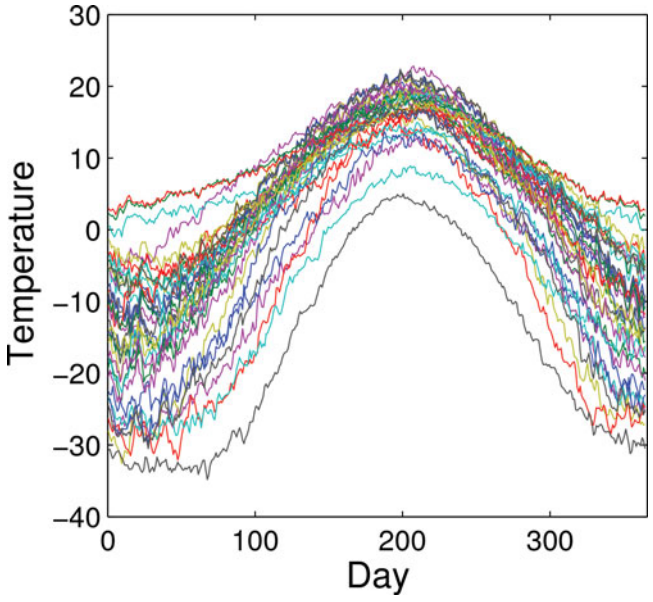

Figure 7. Daily mean temperature recorded at 35 Canadian cities in 1 year.

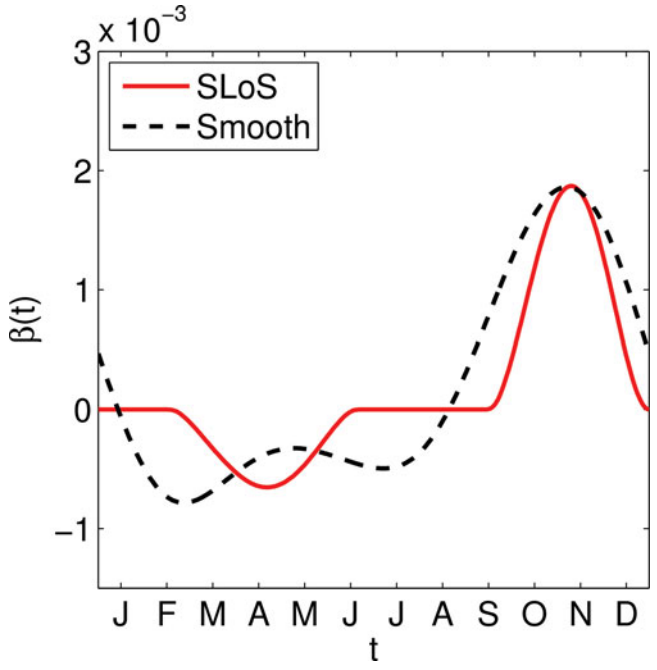

Figure 8. The estimated coefficient function $\hat{\beta}(t)$ for the functional linear model (1) from the Canadian weather data using our SLoS method (solid line) and the smoothing spline method (dashed line).

year using model (1). The dataset contains daily mean temperature curves of 35 Canadian cities in a year, as plotted in Figure 7. Along with these curves, the logarithm of the annual rainfall for each city is recorded.

Figure 8 displays the temperature coefficient functions estimated by the SLoS method and smoothing spline method. It is observed that the SLoS estimate $\hat{\beta}(t)$ is zero roughly in January, February, and summer months (late June, July, August, and early September). This suggests that the temperature in summer has no significant effect on the annual rainfall. Our method also produces a smooth estimate of $\beta(t)$ on the nonnull subregions. It indicates that the temperature in fall months (late October, November, and early December) has a significant contribution to the annual rainfall. These results are consistent with the results discovered in the previous research on this data (James, Wang, and Zhu 2009).

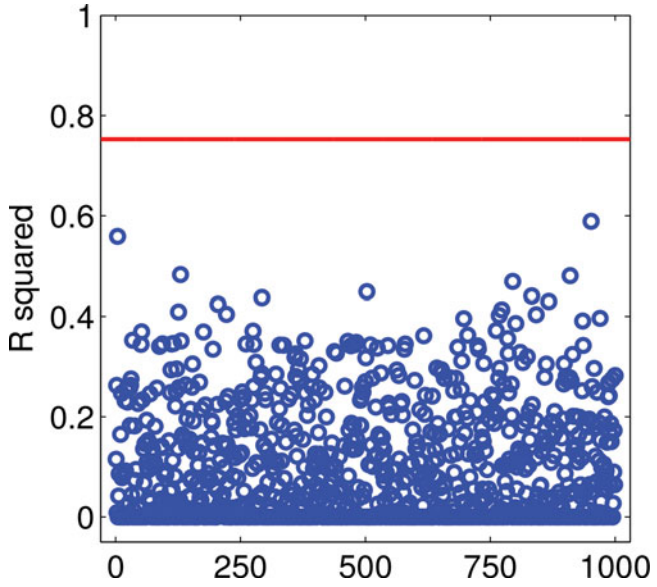

Figure 9. R^2 from permuting the response variable 1000 times. The solid line represents the observed R^2 of SLoS estimate from the true data.

Figure 9 shows the results of a permutation test for SLoS estimator on this data. The solid line represents the R^2 for the SLoS estimator, which is 0.73. The responses are random permuted 1000 times and the new R^2 values are plotted in Figure 9, indicated by circles. All 1000 permuted R^2 values are under 0.73. This is a strong evidence for the relationship between temperature and annual rainfall that is uncovered by our SLoS method.

6. Concluding Remarks

Parsimonious models via SCAD or other shrinkage regularization methods have been proven to have less variability and better interpretability. Similarly, locally sparse modeling in functional linear regression models enjoys reduction of variability and improvement of interpretability. In this article, we have proposed a smooth and locally sparse (SLoS) estimator of the coefficient function based on a novel functional regularization technique "fSCAD" that extends the ordinary SCAD to the functional setting.

The SLoS procedure is a combination of three techniques: (1) the fSCAD that is responsible for identifying the null subregions of the coefficient function while at the same time avoiding over shrinking the nonzero values, (2) the B-spline basis expansion that is used to practically compute the SLoS estimator efficiently thanks to its compact support property, and (3) the roughness regularization that assures the smoothness of our estimator even when a large number of knots are used to define the B-spline basis. Therefore, our method is able to accurately identify the null region and simultaneously produce a smooth estimator on the nonnull region. Comparing to existing methods in the literature, our estimation procedure is theoretically sounder, computationally simpler, and numerically superior. Simulation studies and applications on real datasets have demonstrated that the SLoS estimator is practically appealing, as it helps not only improve interpretability of the model but also reduce both estimation and prediction errors. Moreover, it might result in further practical impact, such as a saving on energy, time, money, and/or other resources.

While this work focuses on functional linear regression, it is important to recognize that the framework of SLoS method and the developed fSCAD regularization can be applied in many other domains of functional data analysis. For example, it may be used in spline smoothing problems to obtain a smooth and locally sparse estimator of an unknown curve. It can also be used in functional principle component analysis to produce smooth and locally sparse functional principal components. Both of these problems are investigated in our ongoing research.

Supplementary Materials

Supplementary File: This file contains additional simulation studies and proofs of theorems stated in the article. (supplementary.pdf, PDF file)

Acknowledgments

The authors are grateful for the invaluable comments and suggestions from the editor, Dr. Dianne Cook, an associate editor, and two reviewers, which are very helpful for improving this article. The authors thank Dr. Jianhua Huang from Texas A&M University for his great suggestions and help in the theorem proofs. This research was supported by discovery grants of J. Cao and L. Wang from the Natural Sciences and Engineering Research Council of Canada (NSERC), H. Wang's NSF grants DMS-1106975 and DMS-1521746, and Z. Lin's Alexander Graham Bell Canada Graduate Scholarship from NSERC.

References

Cai, T. T., and Hall, P. (2006), "Prediction in Functional Linear Regression," *The Annals of Statistics*, 34, 2159–2179. [306]

Cardot, H., Ferraty, F., and Sarda, P. (2003), "Spline Estimators for the Functional Linear Model," *Statistica Sinica*, 13, 571–591. [306,309,310,311,313,316]

Cardot, H., Mas, A., and Sarda, P. (2007), "CLT in Functional Linear Regression Models," *Probability Theory and Related Fields*, 138, 325–361. [306]

Crambes, C., Kneip, A., and Sarda, P. (2009), "Smoothing Splines Estimators for Functional Linear Regression," *The Annals of Statistics*, 37, 35–72. [306]

de Boor, C. (2001), *A Practical Guide to Splines*, New York: Springer-Verlag. [308,311]

Fan, J., and Li, R. (2001), "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties," *Journal of the American Statistical Association*, 96, 1348–1360. [307,309,310]

Fan, J., and Lv, J. (2011), "Non-Concave Penalized Likelihood with NP-Dimensionality," *IEEE Transactions on Information Theory*, 57, 5467–5484. [307]

Fan, J., and Peng, H. (2004), "Nonconcave Penalized Likelihood with a Diverging Number of Parameters," *The Annals of Statistics*, 32, 928–961. [307,311]

Fan, J., and Zhang, J.-T. (2000), "Two-Step Estimation of Functional Linear Models with Applications to Longitudinal Data," *Journal of the Royal Statistical Society*, Series B, 62, 303–322. [307]

Hall, P., and Horowitz, J. L. (2007), "Methodology and Convergence rates for Functional Linear Regression," *The Annals of Statistics*, 35, 70–91. [306]

Hastie, T., and Mallows, C. (1993, May), "A Statistical View of Some Chemometrics Regression Tools: Discussion," *Technometrics*, 35, 140–143. [306]

James, G. M., Wang, J., and Zhu, J. (2009), "Functional Linear Regression that's Interpretable," *The Annals of Statistics*, 37, 2083–2108. [307,309,313,316,317]

Li, Y., and Hsing, T. (2007), "On Rates of Convergence in Functional Linear Regression," *Journal of Multivariate Analysis*, 98, 1782–1804. [306]

Lian, H. (2012), "Convergence of Nonparametric Functional Regression Estimates with Functional Responses," *Electronic Journal of Statistics*, 6, 1373–1391. [307]

Müller, H.-G., and Stadtmüller, U. (2005), "Generalized Functional Linear Models," *The Annals of Statistics*, 33, 774–805. [306]

Noh, H. S., and Park, B. U. (2010), "Sparse Varying Coefficient Models for Longitudinal Data," *Statistica Sinica*, 20, 1183–1202. [309]

Preda, C. (2007), "Regression Models for Functional Data by Reproducing Kernel Hilbert Spaces Methods," *Journal of Statistical Planning and Inference*, 137, 829–840. [306]

Ramsay, J. O., and Silverman, B. W. (1997), *Functional Data Analysis*, New York: Springer-Verlag. [306]

——— (2005), *Functional Data Analysis* (Springer Series in Statistics, 2nd ed.), New York: Springer. [316]

Tu, C. Y., Song, D., Breidt, F. J., Berger, T. W., and Wang, H. (2012), "Functional Model Selection for Sparse Binary Time Series With Multiple Inputs," *Economic Time Series: Modeling and Seasonality*, 477–497. [306]

Wang, H., and Kai, B. (2015), "Functional Sparsity: Global Versus Local," *Statistica Sinica*, 25, 1337–1354. [306]

Wang, L., Chen, G., and Li, H. (2007), "Group Scad Regression Analysis for Microarray Time Course Gene Expression Data," *Bioinformatics*, 23, 1486–1494. [307]

Yao, F., Müller, H.-G., and Wang, J.-L. (2005), "Functional Linear Regression Analysis for Longitudinal Data," *The Annals of Statistics*, 33, 2873–2903. [306,310]

Yuan, M., and Cai, T. T. (2010), "A Reproducing Kernel Hilbert Space Approach to Functional Linear Regression," *The Annals of Statistics*, 38, 3412–3444. [306]

Zhou, J., Wang, N.-Y., and Wang, N. (2013), "Functional Linear Model with Zero-Value Coefficient Function at Sub-Regions," *Statistica Sinica*, 23, 25–50. [307,309,311,313,315]

Zou, H., and Li, R. (2008), "One-Step Sparse Estimates in Nonconcave Penalized Likelihood Models," *The Annals of Statistics*, 36, 1509–1533. [309]