



Locally sparse and robust partial least squares in scalar-on-function regression

Sude Gurer¹ · Han Lin Shang² · Abhijit Mandal³ · Ufuk Beyaztas¹

Received: 6 February 2024 / Accepted: 24 June 2024
© The Author(s) 2024

Abstract

We present a novel approach for estimating a scalar-on-function regression model, leveraging a functional partial least squares methodology. Our proposed method involves computing the functional partial least squares components through sparse partial robust M regression, facilitating robust and locally sparse estimations of the regression coefficient function. This strategy delivers a robust decomposition for the functional predictor and regression coefficient functions. After the decomposition, model parameters are estimated using a weighted loss function, incorporating robustness through iterative reweighting of the partial least squares components. The robust decomposition feature of our proposed method enables the robust estimation of model parameters in the scalar-on-function regression model, ensuring reliable predictions in the presence of outliers and leverage points. Moreover, it accurately identifies zero and nonzero sub-regions where the slope function is estimated, even in the presence of outliers and leverage points. We assess our proposed method's estimation and predictive performance through a series of Monte Carlo experiments and an empirical dataset—that is, data collected in relation to oriented strand board. Compared to existing methods our proposed method performs favorably. Notably, our robust procedure exhibits superior performance in the presence of outliers while maintaining competitiveness in their absence. Our method has been implemented in the `robsfpls` package in

Keywords Dimension reduction · Functional linear regression · Robustness · Sparse estimation

1 Introduction

In the realm of statistical modeling, scalar-on-function regression has emerged as a powerful framework for scrutinizing the relationship between a scalar response variable and one or more functional predictors (see, e.g., Hastie and Mallows 1993). It is particularly useful for data structures featuring predictors that manifest as curves, surfaces, or, more broadly, functions evolving over a continuous domain. For comprehensive theoretical developments in and applications of scalar-on-function regression models, readers are referred to the works of Cardot et al. (1999), Preda and

Saporta (2005), Cai and Hall (2006), Ramsay and Silverman (2006), Hall and Horowitz (2007), Morris (2015), Reiss et al. (2017). However, as researchers seek to uncover localized patterns and sparse structures within these functional relationships, a demand arises for flexible and interpretable methodologies.

This study explores locally sparse estimation in the context of scalar-on-function regression models. Locally sparse models are designed to identify and exploit regions within the functional domain where the relationship between the response variable and functional predictors exhibits sparsity, allowing for more precise and interpretable insights. This localized approach is valuable when dealing with data where the functional predictors influence the response variable in a localized and non-uniform manner.

Consider a stochastic process denoted by $\{Y_i, \mathcal{X}_i(t) : i = 1, \dots, n\}$, extracted as a random sample from the pair (Y, \mathcal{X}) , where $Y \in \mathbb{R}$ is a scalar random variable. The process $\mathcal{X} = \{\mathcal{X}(t)\}_{t \in \mathcal{I}}$, satisfying $\int_{\mathcal{I}} E(\mathcal{X}^2) < \infty$, is characterized by curves in the \mathcal{L}_2 Hilbert space denoted by \mathcal{H} , with the constraint that t belongs to a bounded and closed interval $\mathcal{I} \subseteq \mathbb{R}$.

✉ Ufuk Beyaztas
ufuk.beyaztas@marmara.edu.tr

¹ Department of Statistics, Marmara University, 34722 Kadikoy, Istanbul, Turkey

² Department of Actuarial Studies and Business Analytics, Macquarie University, Sydney, Australia

³ Department of Mathematical Sciences, University of Texas at El Paso, El Paso, USA

It is assumed that $\mathcal{X}(t)$ represents a \mathcal{L}_2 continuous stochastic process. Then, we consider the following scalar-on-function regression model:

$$Y_i = \beta_0 + \int_{\mathcal{I}} \mathcal{X}_i(t)\beta(t)dt + \epsilon_i, \quad (1.1)$$

where $\beta_0 \in \mathbb{R}$ denotes a constant scalar intercept in the model, $\beta(t) \in \mathcal{L}_2[\mathcal{I}]$ represents a square-integrable function from \mathcal{I} to the real line, and $\epsilon_i \in \mathbb{R}$ is the scalar error term. The error term is commonly assumed to follow an independent and identically distributed Gaussian distribution with a mean of zero and a variance σ_ϵ^2 , that is, $\epsilon_i \sim N(0, \sigma_\epsilon^2)$, for $i = 1, \dots, n$.

Model (1.1) can be applied to a range of diverse practical scenarios. The primary emphasis typically centers on estimating the functional component, $\beta(t)$. As $\beta(t)$ is a function rather than a scalar, discerning the regions where it assumes significant or minimal values becomes crucial for comprehending the complex interplay between the functional explanatory variable and the outcome variable. This insight enhances our understanding of the model's dynamics in empirical data analyses.

Estimating the slope function $\beta(t)$ from observed functional data presents a formidable challenge due to its inherently infinite-dimensional nature. Without constraints on $\beta(t)$, arbitrary choices of $\hat{\beta}_0$ and $\hat{\beta}(t)$ could be made to drive the residual sum of squares to zero, resulting in a perfect prediction of the response variable. However, this unconstrained approach often yields a highly irregular and difficult-to-interpret slope function. Thus, a certain level of smoothing is imperative to regulate the slope estimator. Furthermore, the estimation of $\beta(t)$ poses an ill-posed inverse problem. To reframe Model (1.1), consider the expression:

$$Y_i - \mu_y = \int_{\mathcal{I}} [\mathcal{X}_i(t) - \mu_x]\beta(t)dt + \epsilon_i,$$

where $\mu_x = E(\mathcal{X}_i)$ and $\mu_y = E(Y_i) = \beta_0 + \int_{\mathcal{I}} \mathcal{X}_i(t)\beta(t)dt$. Additionally, define $g(v) = E\{[Y(v) - \mu_y(v)]\{\mathcal{X}(v) - \mu_x(v)\}\}$. Utilizing Fubini's theorem, it follows that $\zeta\beta(t) = g$, requiring the inversion of the operator ζ . However, as ζ is a compact linear operator in the infinite-dimensional space $\mathcal{L}_2[\mathcal{I}]$, it lacks a bounded inverse.

The regression model outlined in (1.1), featuring a scalar-on-function relationship, can be conceptualized as a classical multiple linear regression model with an extensive array of predictors. In this context, treating data points on the curves as individual variables allows for the application of dimension reduction-based regression techniques, such as principal component regression and partial least squares (PLS) regression, to Model (1.1). Nevertheless, these methodologies overlook the continuity, smoothness, and ordering inherent in the measurements on the curves, as pointed out by

Guan et al. (2022). Diverse strategies, including general basis expansion-based regressions (see, e.g., Marx and Eilers 1999; Cardot et al. 2003; Cai and Hall 2006; Goldsmith et al. 2011; Zhao et al. 2012), functional principal component (FPC) regression (see, e.g., Yao 2007; Hall and Horowitz 2007; Lee and Park 2012; Reiss et al. 2017; Wang et al. 2016; Goldsmith and Scheipl 2014), and functional partial least squares (FPLS) regression (see, e.g., Preda and Saporta 2005; Reiss and Odgen 2007; Aguilera et al. 2010; Delaigle and Hall 2012; Febrero-Bande et al. 2017; Yu et al. 2016), have been proposed to proficiently estimate the model parameters in scalar-on-function regression models.

All of the techniques mentioned above assume that the impact of the functional predictor extends across the entire time interval \mathcal{I} or a continuous region thereof. However, this assumption may not hold universally. Specifically, in cases where, on a subregion $\mathcal{U} \subset \mathcal{I}$, $\beta(t) = 0$ for every $t \in \mathcal{U}$, the contribution of $\mathcal{X}_i(t)$ to Y_i is null within the interval \mathcal{U} . Recognizing this scenario, an estimation of $\beta(t)$ becomes more interpretable and practically appealing if it not only provides weights for the contribution of $\mathcal{X}_i(t)$ across the entire domain, but also identifies subregions where $\mathcal{X}_i(t)$ lacks statistically significant influence on Y_i . This type of $\beta(t)$ estimate is referred to as the locally sparse estimate (Tu et al. 2012; Wang and Kai 2015; Lin et al. 2017).

A range of methodologies has been advanced to address the issue of locally sparse modeling for the regression coefficient function in (1.1). Pioneering this realm, James et al. (2009) introduce a method for locally sparse estimation of the slope function, wherein they employ a straightforward grid basis for expanding the slope function, yielding empirical results. Hall and Hooker (2016), employing general functional regression, explore truncated functional linear regression models. In a different approach, Zhou et al. (2013) and Lin et al. (2017) discuss the locally sparse slope function, representing it as a linear combination of B-spline basis functions. Guan et al. (2020) propose a novel approach employing a nested group bridge penalty in conjunction with B-spline basis expansion and penalized least squares to examine the scalar-on-function truncated linear regression model. To pursue localized sparsity, Guan et al. (2022) introduce a sparse variant of FPLS regression to obtain a locally sparse estimate for the regression coefficient function in Model (1.1).

The existing methodologies employed to derive locally sparse estimations of regression coefficient functions in scalar-on-function regression models rely on non-robust estimation strategies. The finite-sample performance of these methods may be significantly compromised in the presence of outliers emanating from a stochastic process with a distribution distinct from that of the majority of the remaining observations. In instances involving outliers, the current non-robust estimation techniques may yield biased parameter

estimates for the model, resulting in suboptimal model fitting and predictive accuracy.

The current research addresses this gap by introducing a robust approach to effectively obtain the locally sparse estimator of the regression coefficient function $\beta(t)$ in the scalar-on-function regression model amidst outliers. Our methodology is founded on the robust decomposition of the functional entities outlined in Model (1.1). In the literature, various robust PLS algorithms have been introduced to mitigate the impact of outliers. For instance, Wakelin and Macfie (1992) introduce a robust PLS algorithm by incorporating the M estimate into the PLS framework. Griep et al. (1995) propose three distinct robust PLS algorithms based on the least median of squares, Siegel's repeated median, and iterative reweighted least squares methods. However, these techniques are vulnerable to outliers in the predictor space (see, e.g., Gonzalez et al. 2009). Gil and Romera (1998) introduce a robust PLS algorithm by robustifying the sample covariance matrix of the predictor and the sample cross-covariance matrix between the predictor and response variables. Nevertheless, this algorithm is unsuitable for high-dimensional regressors due to the inappropriate subsampling scheme employed. Hubert and Branden (2003) propose two robust PLS algorithms resilient to outliers in both the response and predictor variables, utilizing a robust covariance matrix for high-dimensional data and robust linear regression. Serneels et al. (2005) propose a robust PLS algorithm where the PLS components and corresponding scores are obtained via robust M regression. Gonzalez et al. (2009) present a robust PLS algorithm by robustifying the sample covariance matrix, demonstrating that proper robustification of the sample covariance matrix renders further robustification of the linear regression steps of the PLS algorithm unnecessary. Alin and Agostinelli (2017) extend the weighted likelihood estimation principle to the PLS method, proposing a robust SIMPLS algorithm. Polat and Gunay (2019) propose a robust PLS algorithm by robustly estimating the sample covariance matrix using the minimum covariance determinant approach. Some of these approaches have been successfully extended to functional data for robust estimation of regression parameters in functional regression models (see, e.g., Delaigle and Hall 2012; Beyaztas and Shang 2022a, b; Beyaztas et al. 2023).

While many of the aforementioned robust PLS methods facilitate robust dimension reduction and parameter estimation in regression models, they do not simultaneously yield sparse estimates of regression parameters. To obtain robust and sparse estimates of Model (1.1), we extend the sparse partial robust M regression algorithm proposed by Hoffmann et al. (2015) to functional data. This algorithm is grounded in the partially robust M regression method introduced by Serneels et al. (2005) and the sparse PLS algorithm presented by Chun and Keleş (2010), adapted explicitly for functional data. The proposed method, namely RSFPLS (Robust Sparse

Functional Partial Least Squares), simultaneously identifies null subregions of $\beta(t)$ while generating a robust and smooth estimate in nonnull subregions. In the RSFPLS approach, the functional predictor undergoes RSFPLS decomposition, approximating the infinite-dimensional scalar-on-function regression model within the finite-dimensional space defined by RSFPLS basis expansion coefficients. This transformation results in a finite-dimensional PLS regression model based on these coefficients for scalar response. Subsequently, we employ an M-estimator to estimate the parameters of this model. The basis expansion coefficients generated by the RSFPLS decomposition represent the projection of the functional predictor in the finite-dimensional space. Thus, the sparse partial robust M regression algorithm robustly and effectively identifies null subregions of the regression coefficient within the regression problem involving the scalar response and basis expansion coefficients, which is used to approximate the regression coefficient function $\beta(t)$.

In the domain of regression analysis, two distinct categories of anomalies manifest: 1) leverage points, signifying atypical observations within the predictor space; and 2) vertical outliers, denoting uncommon observations within the response variable. The presented approach integrates mechanisms to mitigate the influence of leverage points by utilizing the RSFPLS technique. The M-estimator is also harnessed to alleviate the impact of vertical outliers in estimating regression parameters within the finite-dimensional space. As a result, the proposed methodology is resilient against the influence of leverage points in the predictor variable and the presence of vertical outliers in the response variable.

The subsequent sections of this manuscript are organized as follows. In Sect. 2, we provide an overview of FPLS and its sparse variant and introduce our proposed method along with the underlying principles employed for estimating model parameters. Section 3 details a set of Monte-Carlo experiments conducted to assess the performance of the proposed method in terms of estimation and prediction. Results from the oriented strand board (OSB) data analysis are presented in Sect. 4. Finally, Sect. 5 offers concluding remarks, accompanied by insights into potential extensions of the methodology.

2 Methodology

2.1 Functional partial least squares regression

For the scalar-on-function regression model presented in (1.1), the FPLS procedure, encompassing both sparse and non-sparse variants, starts by computing orthonormal FPLS basis functions denoted as $w(t)$. In this study, we assume that the scalar response and functional predictor are mean-zero processes so that $E[Y] = E[\mathcal{X}(t)] = 0$. As in the discrete scenario, the FPLS technique yields orthogonal latent

components, representing linear combinations of the functional predictors, by maximizing the squared covariance between the response and these orthogonal latent components (refer to (2.1)). In other words, the FPLS basis functions are obtained as solutions to Tucker’s criterion, extended to functional data (see, for instance, Tenenhaus 1998; Stone and Brooks 1990; Preda and Saporta 2005). The criterion is expressed through optimization problems, such as maximizing the squared covariance between Y and the integral of $\mathcal{X}(t)$ weighted by $w(t)$ as follows:

$$\operatorname{argmax}_{\substack{w(t) \in \mathcal{L}_2[\mathcal{I}] \\ \|w(t)\|=1}} \operatorname{Cov}^2 \left(Y, \int_{\mathcal{I}} \mathcal{X}(t)w(t)dt \right). \tag{2.1}$$

The cross-covariance operators $\mathcal{C}_{Y\mathcal{X}}$ and $\mathcal{C}_{\mathcal{X}Y}$ are introduced to evaluate the contribution of $\mathcal{X}(t)$ to Y and its adjoint as follows:

$$\begin{aligned} \mathcal{C}_{Y\mathcal{X}} : \mathcal{L}_2[\mathcal{I}] &\rightarrow \mathbb{R}, & f &\xrightarrow{\mathcal{C}_{Y\mathcal{X}}} x = \int_{\mathcal{I}} \operatorname{Cov}(\mathcal{X}(t), Y) f(t)dt, \\ \mathcal{C}_{\mathcal{X}Y} : \mathbb{R} &\rightarrow \mathcal{L}_2[\mathcal{I}], & x &\xrightarrow{\mathcal{C}_{\mathcal{X}Y}} f(t) = x \operatorname{Cov}(\mathcal{X}(t), Y). \end{aligned}$$

The optimization problem presented in (2.1) can be rephrased as the maximization of the ratio of inner products, expressed as:

$$\max_{w(t) \in \mathcal{L}_2[\mathcal{I}]} \frac{\langle \mathcal{V}w, w \rangle_{\mathcal{H}}}{\langle w, w \rangle_{\mathcal{H}}}, \tag{2.2}$$

where the operator $\mathcal{V} = \mathcal{C}_{\mathcal{X}Y} \circ \mathcal{C}_{Y\mathcal{X}}$ is self-adjoint, positive, and compact. Its spectral analysis provides a countable set of positive eigenvalues λ along with corresponding orthonormal basis functions w . The solution to (2.1) corresponds to the eigenfunction of \mathcal{V} associated with its largest eigenvalue λ_1 , as denoted by $\mathcal{V}w_1 = \lambda_1 w_1$ (see, for instance, Preda and Saporta 2005). Consequently, the first FPLS component is defined as $\xi_1 = \int_{\mathcal{I}} \mathcal{X}(t)w_1(t)dt$.

The FPLS procedure unfolds iteratively, with each iteration incorporating information from the previous one. Let $h = 1, \dots, H$ represent the iteration index, where H stands for the maximum number of PLS components we aim to obtain. Let $Y^{(0)} = Y$ and $\mathcal{X}^{(0)}(t) = \mathcal{X}(t)$. Then, at each step h , the FPLS component is derived by computing residuals from regression problems involving $Y^{(h)}$ and $\mathcal{X}^{(h)}(t)$. These residuals are defined as $Y^{(h)} = Y^{(h-1)} - c^{(h)}\xi^{(h)}$ and $\mathcal{X}^{(h)}(t) = \mathcal{X}^{(h-1)}(t) - p^{(h)}(t)\xi^{(h)}$. The coefficients $c^{(h)}$ and $p^{(h)}(t)$ are computed at each iteration using conditional expectations, that is,

$$\begin{aligned} c^{(h)} &= \frac{\mathbb{E}[Y^{(h-1)}\xi^{(h)}]}{\mathbb{E}[(\xi^{(h)})^2]}, \\ p^{(h)}(t) &= \frac{\mathbb{E}[\mathcal{X}^{(h-1)}(t)\xi^{(h)}]}{\mathbb{E}[(\xi^{(h)})^2]}. \end{aligned}$$

In each iteration, the h^{th} FPLS component is determined as the random variable that maximizes the Tucker criterion (2.1) using the residuals $Y^{(h-1)}$ and $\mathcal{X}^{(h-1)}(t)$. Thus, $\xi^{(h)} = \int_{\mathcal{I}} \mathcal{X}^{(h-1)}(t)w^{(h)}(t)dt$, where the basis function $w^{(h)}(t)$ is selected as the solution that maximizes the squared covariance between $Y^{(h-1)}$ and the integral of $\mathcal{X}^{(h-1)}(t)$ weighted by $w(t)$:

$$w^{(h)}(t) = \operatorname{argmax}_{\substack{w(t) \in \mathcal{L}_2[\mathcal{I}] \\ \|w_m(t)\|=1}} \operatorname{Cov}^2 \left(Y^{(h-1)}, \int_{\mathcal{I}} \mathcal{X}^{(h-1)}(t)w(t)dt \right),$$

which corresponds to the largest eigenvalue of $\mathcal{V}_{h-1} = \mathcal{C}_{\mathcal{X}Y}^{h-1} \circ \mathcal{C}_{Y\mathcal{X}}^{h-1}$. Here, $\mathcal{C}_{\mathcal{X}Y}^{h-1}$ and $\mathcal{C}_{Y\mathcal{X}}^{h-1}$ denote the cross-covariance operators of $Y^{(h-1)}$ and $\mathcal{X}^{(h-1)}$, respectively, and the following equation holds:

$$\mathcal{V}_{h-1}(w_h) = \lambda_h w_h. \tag{2.3}$$

Incorporating sparsity into the FPLS method involves introducing a penalty term, such as L_1 penalization, into the optimization problem (2.2). However, solving this formulation results in a solution that needs more sparsity, and the problem is non-convex (see, e.g., Jolliffe et al. 2003; Chun and Keleş 2010, for further details). To achieve a more sparse solution, akin to Chun and Keleş (2010), we propose to reframe the problem (2.2) by extending the regression framework of sparse principal component analysis as introduced by Zou et al. (2006). Specifically, we introduce a surrogate for the basis function $w(t)$, denoted by $\psi(t)$, and integrate an L_1 penalty term into the orthonormal basis functions. As a result, we re-express the optimization problem in (2.2) as follows:

$$\begin{aligned} \min_{w(t), \psi(t) \in \mathcal{L}_2[\mathcal{I}]} & -\kappa \frac{\langle \mathcal{V}w, w \rangle_{\mathcal{H}}}{\langle w, w \rangle_{\mathcal{H}}} \\ & + (1 - \kappa) \frac{\langle \mathcal{V}(\psi - w), (\psi - w) \rangle_{\mathcal{H}}}{\langle (\psi - w), (\psi - w) \rangle_{\mathcal{H}}} + \eta_1 \mathcal{P}(\psi), \end{aligned} \tag{2.4}$$

where κ and η_1 are non-negative parameters and $\mathcal{P}(\psi)$ is a linear form that acts as a global penalty on $\psi(t)$. The proposed approach enhances sparsity by applying an L_1 penalty to a surrogate direction vector $\psi(t)$, rather than directly to the original direction vector $w(t)$. This strategy aims to enforce exact zeros in the solution, while ensuring that $w(t)$ and $\psi(t)$ remain closely aligned. Following the methodology outlined by Chun and Keleş (2010), the optimization problem (2.4) is addressed through an iterative process, alternating between optimizing $w(t)$ with fixed $\psi(t)$ and optimizing $\psi(t)$ with fixed $w(t)$. Then, the sparse functional partial least squares (SFPLS) basis function is obtained as $w(t) = \psi(t)/\|\psi(t)\|$. However, due to the infinite-dimensional nature of the functional predictors,

direct computation of the sparse/non-sparse FPLS basis functions and, consequently, the corresponding components is not attainable. To address this challenge, a pragmatic approach involves approximating the FPLS orthonormal basis functions within a finite-dimensional space, achieved through an appropriate basis expansion function. In this context, we adopt the cubic *B*-spline basis, where each basis function is associated with four knots for this purpose (de Boor 2001).

2.2 The FPLS via basis function expansion

Suppose the functional predictor $\mathcal{X}(t)$ is formed by a K -dimensional basis of functions denoted as $\{\phi_1(t), \dots, \phi_K(t)\} \in \mathcal{L}_2[\mathcal{I}]$. Consequently, $\mathcal{X}(t)$ can be represented using the basis expansion as $\mathcal{X}(t) = \sum_{k=1}^K a_k \phi_k(t) = \mathbf{a}^\top \boldsymbol{\phi}(t)$, where $a_k = \int_{\mathcal{I}} \mathcal{X}(t) \phi_k(t) dt$ is the random coefficient, $\mathbf{a} = [a_1, \dots, a_K]^\top$, and $\boldsymbol{\phi}(t) = [\phi_1(t), \dots, \phi_K(t)]^\top$. Through the spectral analysis of \mathcal{V} and (2.3), the FPLS basis functions and the corresponding FPLS components are approximated within the basis $\boldsymbol{\phi}(t)$, expressed as $w(t) = \sum_{k=1}^K w_k \phi_k(t) = \mathbf{w}^\top \boldsymbol{\phi}(t)$, where $w_k = \int_{\mathcal{I}} w(t) \phi_k(t) dt$ and $\mathbf{w} = [w_1, \dots, w_K]^\top$. Therefore, the regression coefficient function $\beta(t)$ can be represented in the same basis $\boldsymbol{\phi}(t)$ as $\beta(t) = \sum_{k=1}^K \beta_k \phi_k(t) = \boldsymbol{\beta}^\top \boldsymbol{\phi}(t)$, where $\beta_k = \int_{\mathcal{I}} \beta(t) \phi_k(t) dt$ and $\boldsymbol{\beta} = [\beta_1, \dots, \beta_K]^\top$.

Let $\Phi = \int_{\mathcal{I}} \boldsymbol{\phi}(t) \boldsymbol{\phi}^\top(t) dt$ represent the matrix of inner products between the basis functions. Additionally, let $\Phi^{1/2}$ denote the square root of Φ . Then, as per Aguilera et al. (2010) and Beyaztas and Shang (2022a), the FPLS regression of Y on $\mathcal{X}(t)$ is equivalent to the PLS regression of Y on $\mathbf{A} = \mathbf{a}(\Phi^{1/2})^\top$. Consequently, both models yield the same PLS components at each step of the PLS algorithm. Let \mathbf{W} denote the H -dimensional matrix of eigenvectors obtained from the PLS regression of Y on \mathbf{A} ; then, the PLS scores are computed as $\boldsymbol{\xi} = \mathbf{A}\mathbf{W}$. Subsequently, the scalar-on-function regression model is approximated through the regression problem $Y = \gamma_0 + \boldsymbol{\xi}\boldsymbol{\gamma}$, where γ_0 is the intercept, and $\boldsymbol{\gamma}$ is the vector of regression coefficients of Y on \mathbf{A} . Let $\widehat{\boldsymbol{\gamma}}$ represent the least-squares estimation of $\boldsymbol{\gamma}$. The random coefficient associated with the regression coefficient function is derived as $\widehat{\boldsymbol{\beta}} = (\Phi^{-1/2})^\top \mathbf{W}\widehat{\boldsymbol{\gamma}}$. Finally, the regression coefficient function is estimated as $\widehat{\beta}(t) = \boldsymbol{\phi}(t)\widehat{\boldsymbol{\beta}}$.

2.3 Robust sparse functional partial least squares regression

As shown by Hoffmann et al. (2015), the sparse partial robust M estimator can be implemented straightforwardly in the classical PLS algorithm. Thus, it can be extended to the FPLS algorithm proposed by Beyaztas and Shang (2022a). In the proposed methodology, our initial step involves enhancing the robustness of the PLS estimator through the incorporation of case weights, denoted as $\omega_i \in [0, 1]$, assigned to

the rows of \mathbf{a} and Y . In this context, the case weights are derived as the product of two individual weights, denoted as $\omega^2 = \omega_r \omega_\xi$. Here, ω_r represents the weight assigned to the residuals, while ω_ξ signifies the weight assigned to the predictors. In the conventional M estimator, only ω_r is employed to mitigate the impact of outliers, explicitly focusing on vertical outliers. Consequently, when applying the classical M estimator within the PLS algorithm, only the influence of vertical outliers is dampened in the PLS components. Conversely, ω_ξ is tailored to attenuate the effects of outliers in the predictor domain, particularly leverage points. Consequently, if case weights $\omega^2 = \omega_r \omega_\xi$ are employed, the effects of both vertical outliers and leverage points are mitigated in the PLS components (see, e.g., Serneels et al. 2005, for further insights on case weights).

Introducing weighted basis expansion coefficients and a response variable, we define $\mathbf{a}^\dagger = \boldsymbol{\Omega}\mathbf{a}$ and $Y^\dagger = \boldsymbol{\Omega}Y$, where $\boldsymbol{\Omega}$ is a diagonal matrix with diagonal elements ω_i . Remarkably, observations deemed as outliers—those exhibiting either substantial residuals or elevated predictor values in the regression model—are allocated a weight lower than unity. Residuals, denoted as r_i for $i = 1, \dots, n$, are obtained from the PLS approximation of the scalar-on-function regression model, specifically $r_i = Y_i - \gamma_0 - \boldsymbol{\xi}_i^\top \widehat{\boldsymbol{\gamma}}$. Let $\widehat{\sigma}$ represent the residuals' median absolute deviation (MAD). The case weights are then defined as per Hoffmann et al. (2015):

$$\omega_i^2 = \omega_r \left(\frac{r_i}{\widehat{\sigma}} \right) \omega_\xi \left(\frac{\|\boldsymbol{\xi}_i - \text{med}_{L_1}(\boldsymbol{\xi})\|}{1.4826 \times \text{med}_i \|\boldsymbol{\xi}_i - \text{med}_{L_1}(\boldsymbol{\xi})\|} \right), \tag{2.5}$$

where $\text{med}_i(\mathbf{X}_i)$ is the median of i^{th} row of a matrix \mathbf{A} , $\text{med}_{L_1}(\cdot)$ is the L_1 -median (i.e., the multivariate version of the sample median), and the constant 1.4826 is used for the consistency of the MAD. In the proposed method, Hampel's weight function is used to compute the case weights as suggested by Hoffmann et al. (2015):

$$\omega(u) = \begin{cases} 1, & |u| \leq c_1 \\ \frac{c_1}{|u|}, & c_1 < |u| \leq c_2 \\ \frac{c_3 - |u|}{c_3 - c_2} \frac{c_1}{|u|}, & c_2 < |u| \leq c_3 \\ 0, & c_3 < |u| \end{cases}$$

Here, c_1 , c_2 , and c_3 represent chosen distribution quantiles. The component ω_r conforms to the standard normal distribution, while the component ω_ξ adheres to a chi-square distribution characterized by one degree of freedom. Within Hampel's weight function, a gentle transition occurs from the 95th to the 99.9th percentiles; however, beyond this range, the weight function effectively eliminates outliers (see, for instance, Hampel et al. 1986, for detailed elucidation). Thus, for the residual weight function ω_r , we use the 0.95, 0.975,

and 0.999 quantiles of the standard normal, corresponding to $c_1 = 1.65$, $c_2 = 1.96$, and $c_3 = 3.09$. As for ω_ξ , we employ the respective quantiles of the chi-square distribution with a degree of one, that is, $c_1 = 3.84$, $c_2 = 5.02$, and $c_3 = 10.83$.

Conducting a spectral analysis of \mathcal{V} allows us to represent the cross-covariance operators in terms of the basis $\phi(t)$ as follows:

$$\begin{aligned} \mathcal{C}_{Y\mathcal{X}} : \mathcal{L}_2[\mathcal{I}] &\rightarrow \mathbb{R}, & f &\xrightarrow{\mathcal{C}_{Y\mathcal{X}}} x = \Sigma_{a^\dagger Y^\dagger}^\top \Phi f, \\ \mathcal{C}_{\mathcal{X}Y} : \mathbb{R} &\rightarrow \mathcal{L}_2[\mathcal{I}], & x &\xrightarrow{\mathcal{C}_{\mathcal{X}Y}} f(t) = x \Sigma_{a^\dagger Y^\dagger}, \end{aligned}$$

where $\Sigma_{a^\dagger Y^\dagger} = (\sigma_k)_{K \times 1}$ with $\sigma_k = E[a_k^\dagger Y^\dagger]$ for $k = 1, \dots, K$ represents the cross-covariance matrix between a^\dagger and Y^\dagger . Then, the optimization problem (2.1) can be reformulated in terms of the weighted basis expansion coefficients and the response variable:

$$w^\dagger = \underset{w}{\operatorname{argmax}} \quad w^\top \Phi \Sigma_{a^\dagger Y^\dagger} \Sigma_{a^\dagger Y^\dagger}^\top \Phi w. \tag{2.6}$$

Subsequently, we propose a surrogate for the eigenvector w^\dagger , denoted by ψ^\dagger , and introduce an L_1 penalty on this surrogate to induce sparsity in the PLS estimator:

$$\begin{aligned} \min_{\substack{\psi^\dagger, w^\dagger \\ (w^\dagger)^\top \Phi w^\dagger = 1}} & -\kappa (w^\dagger)^\top \Phi \Sigma_{a^\dagger Y^\dagger} \Sigma_{a^\dagger Y^\dagger}^\top \Phi w^\dagger \\ & + (1 - \kappa) (\psi^\dagger - w^\dagger)^\top \Phi \Sigma_{a^\dagger Y^\dagger} \Sigma_{a^\dagger Y^\dagger}^\top \Phi (\psi^\dagger - w^\dagger) \\ & + \eta_1 \|\psi^\dagger\|_1, \end{aligned} \tag{2.7}$$

where κ and η_1 are non-negative parameters. The final estimation of the eigenvector is then derived as $w^\dagger = \frac{\hat{\psi}^\dagger}{\|\hat{\psi}^\dagger\|}$, ensuring robust sparsity in the eigenvectors. Consequently, we obtain the matrix of robustly estimated sparse eigenvectors denoted by W^\dagger through the solution of (2.7), and let $\xi^\dagger = A^\dagger W^\dagger$, where $A^\dagger = a^\dagger (\Phi^{1/2})^\top$. To achieve a fully robust estimate for the regression coefficient function, we formulate the optimization problem:

$$\hat{\gamma}^\dagger = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^n \rho(Y_i^\dagger - \gamma_0 - \xi^\dagger \gamma), \tag{2.8}$$

where $\rho(\cdot)$ is a symmetric, non-decreasing, and continuously differentiable function with a bounded derivative, ensuring partial robust M regression estimation (Serneels et al. 2005). It's worth noting that when $\rho(u) = u^2$, solving (2.8) yields the least squares estimate of γ , which is known to be sensitive to outliers. To mitigate this sensitivity and obtain a robust estimate of γ , commonly employed bounded loss functions include those introduced by Huber (1964, 1981), Beaton and

Tukey (1974), and Hampel (1974). If we denote the objective function (2.8) as follows:

$$\hat{\gamma}^\dagger = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^n \omega_i (Y_i^\dagger - \gamma_0 - \xi^\dagger \gamma)^2,$$

where ω_i s for $i = 1, 2, \dots, n$ are the case weights. Then, the M estimator transforms into a weighted least squares estimator, albeit with weights contingent upon γ . Employing this formulation in our computational procedures enables the M estimator's computation through an iteratively reweighted least squares algorithm (see, for example, Serneels et al. 2005). It's essential to highlight that the case weights mentioned earlier are dynamic and undergo iterative updates within each PLS iteration.

In more detail, from the optimization problem (2.6), when weighted basis expansion coefficients are employed in the PLS regression, the first eigenvector, denoted as $w^{*(1)}$, is determined by solving

$$\Sigma_{a^\dagger Y^\dagger} \Sigma_{a^\dagger Y^\dagger}^\top \Phi w^{*(1)} = \lambda_1 w^{*(1)}. \tag{2.9}$$

Let us denote $\Phi = \Phi^{1/2} (\Phi^{1/2})^\top$. Consequently, we have:

$$\begin{aligned} (w^{*(1)})^\top \Phi w^{*(1)} &= (w^{*(1)})^\top \Phi^{1/2} (\Phi^{1/2})^\top w^{*(1)}, \\ &= (\tilde{w}^{*(1)})^\top \tilde{w}^{*(1)}, \end{aligned}$$

where $\tilde{w}^{*(1)} = (\Phi^{1/2})^\top w^{*(1)}$ and $w^{*(1)} = (\Phi^{-1/2})^\top \tilde{w}^{*(1)}$. Thus, problem (2.9) can be reformulated as $(\Phi^{1/2})^\top \Sigma_{a^\dagger Y^\dagger} \Sigma_{a^\dagger Y^\dagger}^\top \Phi^{1/2} \tilde{w}^{*(1)} = \lambda_1 \tilde{w}^{*(1)}$. Subsequently, the first robust PLS component is expressed as $\xi^{\dagger(1)} = A^\dagger \tilde{w}^{*(1)}$. The ensuing robust PLS components are iteratively computed using the basis expansion coefficients.

Following the methodologies proposed by Chun and Keleş (2010) and Hoffmann et al. (2015), imposing the sparsity on FPLS estimates according to the constraints in (2.7) yields analytically exact solutions. We update $A^{\dagger(h)}$ through the deflation process, represented as

$$A^{\dagger(h+1)} = A^{\dagger(h)} - \xi^{*(h)} (\xi^{*(1)})^\top A^{\dagger(h)} / \|\xi^{*(h)}\|^2,$$

where $\xi^{*(h)}$ is the matrix of robust PLS scores obtained in the previous iteration and $A^{\dagger(1)} = A^\dagger$ denotes the initial deflation to satisfy the constraints in (2.7). Then, the precise solution for the robust sparse PLS eigenvector, that is, the solution of problem (2.7), is computed as:

$$\begin{aligned} w^{\dagger(h)} &= (|w^{*(h)}| - \eta_1/2) \odot \mathbb{I}(|w^{*(h)}| \\ &\quad - \eta_1/2 > 0) \odot \operatorname{sgn}(w^{*(h)}), \end{aligned} \tag{2.10}$$

where $\mathbb{I}(\cdot)$ signifies the indicator function, \odot denotes the Hadamard product, and $\operatorname{sgn}(\cdot)$ represents the vector of

signs for each component. Consequently, the robust sparse eigenvectors, in terms of the non-deflated basis expansion coefficients, are determined by $\mathbf{W}^\dagger[(\mathbf{W}^\dagger)^\top(\mathbf{A}^\dagger)^\top\mathbf{A}^\dagger\mathbf{W}^\dagger]^{-1}$.

From the results given above, the sparse partial M estimator iteratively reweights the sparse PLS estimate, and the sparsity inherent in the estimated eigenvectors extends to the vector of coefficients acquired through the partial robust M regression, as outlined in (2.8). Subsequently, the robust and sparse estimation of the regression coefficient function within the scalar-on-function regression model is computed as:

$$\widehat{\beta}^\dagger(t) = \phi(t)\widehat{\beta}^\dagger,$$

where $\widehat{\beta}^\dagger = (\Phi^{-1/2})^\top \mathbf{W}^\dagger \widehat{\gamma}^\dagger$.

2.4 Computation details

The determination of $\widehat{\beta}^\dagger(t)$ involves the optimization of four parameters: the number of B-spline basis functions K , the maximum number of PLS components H , sparsity parameter η_1 , and κ . It is noteworthy that, as demonstrated by Chun and Keleş (2010), the optimization problem remains independent of κ for any $\kappa \in (0, 1/2]$ in the context of univariate response variables, which applies to our study. Consequently, the four-parameter search is streamlined to optimize K , H , and η_1 . We employ a three-dimensional grid search algorithm to identify the optimal values for these parameters. Let $\eta \in [0, 1]$ be a sparsity tuning parameter. Consistent with the approach introduced by Hoffmann et al. (2015), we redefine (2.10) as expressed below:

$$\begin{aligned} \mathbf{w}^{\dagger(h)} &= (|\mathbf{w}^{*(h)}| - \eta \max_j |w_j^{*(h)}|) \odot \mathbb{I}(|\mathbf{w}^{*(h)}| \\ &\quad - \eta \max_j |w_j^{*(h)}| > 0) \odot \text{sgn}(\mathbf{w}^{*(h)}), \end{aligned} \tag{2.11}$$

where $w_j^{*(h)}$ represents the j^{th} element of $\mathbf{w}^{*(h)}$. The parameter η governs the threshold size as a fraction of the maximum of $\mathbf{w}^{\dagger(h)}$, below which all elements of the vector $\mathbf{w}^{\dagger(h)}$ are zeroed. Given the known range of η in this definition, it facilitates the tuning parameter selection through cross-validation.

To identify the optimal values of the tuning parameters, namely K , H , and η , we employ a robust k -fold cross-validation, with k set to $k = 10$ in our numerical evaluations. Throughout the cross-validation process, we consider grid values for $\eta \in [0, 0.1, 0.3, 0.5, 0.7, 0.9]$, $H \in [1, 2, 3, 4, 5]$, and $K = [4, 5, 8, 10, 20]$. A 10-fold cross-validation is executed for each combination of these parameters, and the mean squared prediction error on the test sets is computed. Subsequently, the one-sided 20% trimmed mean (20% trimming proportion is commonly used in robust statistics, e.g., Wilcox 2012) squared prediction error is applied as the decision criterion. The combination of parameters yielding the smallest

trimmed mean squared prediction error is identified as the optimal tuning parameter.

We present the general structure of the proposed method in Algorithm 1. In addition, the method is implemented in R package `robsfpls` (the package is available at <https://github.com/sudegurer/robsfpls>), which also includes functions to perform SFPLS, robust and non-robust FPLS, and the functional principal component regression.

Algorithm 1: RSFPLS algorithm

- 1 Given a scalar response Y , functional predictor $\mathcal{X}(t)$, and the tuning parameter K , compute B-spline basis expansion coefficients $\mathbf{A} = \mathbf{a}\Phi^{1/2}$. Then, robustly center both Y and \mathbf{A} using the column-wise median.
- 2 Define initial weights for Ω as $\omega_i = \sqrt{\omega_r(r_i)\omega_\xi(\delta_i)}$, where

$$\begin{aligned} r_i &= \frac{|Y_i|}{1.4826 \text{ med}_j |Y_j|}, \\ \delta_i &= \frac{\|\mathbf{A}_i\|}{\text{med}_j \|\mathbf{A}_j\|}, \quad i = 1, \dots, n. \end{aligned}$$

Here, 1.4826 ensures the consistency of the MAD.

- 3 Iteratively reweighting:
 - 3.1 Weight the response and basis expansion coefficients as $Y^\dagger = \Omega Y$ and $\mathbf{A}^\dagger = \Omega \mathbf{a}\Phi^{1/2}$.
 - 3.2 For a given tuning parameter η and $h = 1, \dots, H$, compute H -dimensional eigenvectors \mathbf{W}^\dagger , scores ξ^\dagger , coefficients γ^\dagger , and the fitted response \widehat{Y}^\dagger by applying the iterative procedure introduced in Sect. 2.3 on Y^\dagger and \mathbf{A}^\dagger .
 - 3.3 Center and scale ξ^\dagger using median and Q_n estimator of Rousseeuw and Croux (1993). Then, compute weights for scores and responses as follows:

$$\begin{aligned} r_i &= \frac{|Y_i^\dagger - \widehat{Y}_i^\dagger - \text{med}_l (Y_l^\dagger - \widehat{Y}_l^\dagger)|}{1.4826 \text{ med}_j |Y_j^\dagger - \widehat{Y}_j^\dagger - \text{med}_l (Y_l^\dagger - \widehat{Y}_l^\dagger)|}, \\ \delta_i &= \frac{\|\xi_i^\dagger\|}{\text{med}_j \|\xi_j^\dagger\|}, \quad i = 1, \dots, n \end{aligned}$$

- 3.4 Update the weight matrix Ω with diagonal elements $\omega_i = \sqrt{\omega_r(r_i)\omega_\xi(\delta_i)}$.

- 4 Repeat Step 3 until convergence, where convergence occurs when the norm between two consecutive approximations of the parameters for the regression problem of Y^\dagger and \mathbf{A}^\dagger is smaller than a specified threshold, e.g., 10^{-2} .
 - 5 Finally, obtain the robust and sparse estimate of the regression coefficient function as $\widehat{\beta}^\dagger(t) = \phi(t)\widehat{\beta}^\dagger$, where $\widehat{\beta}^\dagger = (\Phi^{-1/2})^\top \mathbf{W}^\dagger \widehat{\gamma}^\dagger$.
-

The robust sparse eigenvectors are derived in Step 3.2 of Algorithm 1. This process entails obtaining eigenvectors utilizing the sparse NIPALS algorithm proposed by Chun and Keleş (2010), with further details available in works such as Lee et al. (2011) and Hoffmann et al. (2015). In essence, this algorithm adapts the eigenvector of the deflated \mathbf{A}^\dagger , obtained through the standard NIPALS algorithm of Wold (1974), as outlined in (2.11), to induce sparsity. Within Steps 3.1–3.4

of the algorithm, the iteratively reweighted least squares algorithm is executed, wherein the H -step NIPALS algorithm is conducted during each iteration of the reweighting process.

3 Monte Carlo experiments

A comprehensive assessment of the estimation and predictive capabilities of the proposed RSFPLS method is conducted through a series of Monte Carlo simulations. These simulations encompass two distinct data generation processes (DGPs), and the proposed method’s performance is compared to various counterparts. The methodologies considered for comparison include SFPLS (the non-robust version of RSFPLS), the SFPLS approach introduced by Guan et al. (2022) (referred to as “SFPLS*”, the code for this method is available at <https://github.com/caojiguu/SparFunPLS>), both robust and non-robust FPLS methods proposed by previous research (Beyaztas and Shang 2022a), and functional principal component regression (FPCR). In FPCR, the model relies on the functional principal component scores derived from the functional predictors, as detailed in works such as Ramsay and Silverman (2006) and Beyaztas and Shang (2023).

The DGPs employed, denoted as DGP-I and DGP-II, are modified versions of those utilized by Guan et al. (2022). DGP-I employs a continuous regression coefficient function devoid of a zero sub-region, while DGP-II involves a continuous regression coefficient function featuring a zero sub-region. Both DGPs are examined under two scenarios: the first DGP assumes datasets generated from a smooth process without clear outliers, and the second DGP introduces contamination by random outliers in 5% and 10% of the generated data. The first scenario assesses the correctness of the proposed method, while the second evaluates its robustness.

The functional predictor variable realizations are generated at 501 equally spaced points within the interval $[0, 1]$. The process involves $\mathcal{X}_i(t) = \sum_{j=1}^{50} a_{ij} B_j(t)$, where $B_j(t)$ represents cubic B-spline basis functions defined on 50 equally spaced knots over $[0, 1]$, and the coefficients a_{ij} are sampled from a standard normal distribution. For DGP-I, the regression coefficient function is specified as $\beta(t) = 3t + \exp(t^2) \cos(3\pi t) + 1$. For DGP-II, it is set to $\beta(t) = \sin(4\pi t) \exp(-10t^2)$ with a zero sub-region occurring after the discrete point 368 within the 501 equally spaced points in the interval $[0, 1]$. Subsequently, the functional response is generated according to

$$Y_i = \int_0^1 \mathcal{X}_i(t)\beta(t)dt + \epsilon_i, \tag{3.1}$$

where the ϵ_i are independently generated from normal distributions, ensuring a signal-to-noise ratio of 5 (that is, the

variance of ϵ_i is set to $\sigma_1/5$ where σ_1 is the variance of integral part of (3.1), i.e., $\int_0^1 \mathcal{X}_i(t)\beta(t)dt$).

To introduce outlier-contaminated data in both DGPs, we substitute 5% and 10% of the randomly generated data with outlying observations derived from the error terms and functional predictor, specifically vertical outliers and leverage points. The outlying observations in the functional predictor, denoted by $\tilde{\mathcal{X}}_i(t)$, are generated as $\tilde{\mathcal{X}}_i(t) = \sum a_{ij} B_j(t)$, where a_{ij} is sampled from a chi-square distribution with two degrees of freedom, and $B_j(t)$ represents the B-spline basis functions, consistent with the approach used in the generation of outlier-free data. Subsequently, the contaminated response values are computed using (3.1), but using $\epsilon_i \sim N(\mu = 0.75, \sigma^2 = 1)$.

In the outlier generation process, vertical outliers and leverage points are generated separately. After generating outlier-free data, $n \times [5\%, 10\%]$ of the randomly selected observations of the scalar response are replaced by vertical outliers generated using (3.1) and $\epsilon_i \sim N(\mu = 0.75, \sigma^2 = 1)$. Then, the elements of the functional predictor corresponding to vertical outliers are replaced by leverage points generated by $\tilde{\mathcal{X}}_i(t)$. In this way, the same $n \times [5\%, 10\%]$ portion of the generated data is replaced by vertical outliers and leverage points. To verify whether the generated vertical outliers are indeed outliers, that is, whether they lie at an abnormal distance from the bulk of the data, we generate 100 independent outlier-contaminated datasets with sizes $n = 100$ and $n = 250$. Then, for each dataset, we employ the interquartile range (IQR) method. Specifically, observations below $Q_1 - 1.5 \times \text{IQR}$ or above $Q_3 + 1.5 \times \text{IQR}$, where Q_1 and Q_3 represent the first and third quartiles of the data, and $\text{IQR} = Q_3 - Q_1$, are considered outliers. Utilizing the IQR criterion, 100% of the deliberately inserted vertical outliers are detected as outliers. For the generated leverage points, we utilize the functional boxplot method proposed by Sun and Genton (2011), available in the `fdaoutlier` package for R (Ojo et al. 2023). The results from functional boxplots reveal that, on average, 85% of the deliberately inserted leverage points are detected as outliers. A visual representation of the generated data for both DGP-I and DGP-II is presented in Fig. 1.

For both DGPs and scenarios, we explore three distinct training sample sizes, namely $n_{\text{train}} = [100, 250, 500]$. Utilizing the generated training data, the models are constructed, and we assess the estimation performance of the methods by computing the integrated squared errors in both zero and nonzero sub-regions, defined as follows:

$$\text{ISE}_0 = \frac{1}{\ell_0} \int_{\mathcal{I}_0} [\beta(t) - \hat{\beta}(t)]^2 dt,$$

$$\text{ISE}_1 = \frac{1}{\ell_1} \int_{\mathcal{I}_1} [\beta(t) - \hat{\beta}(t)]^2 dt,$$

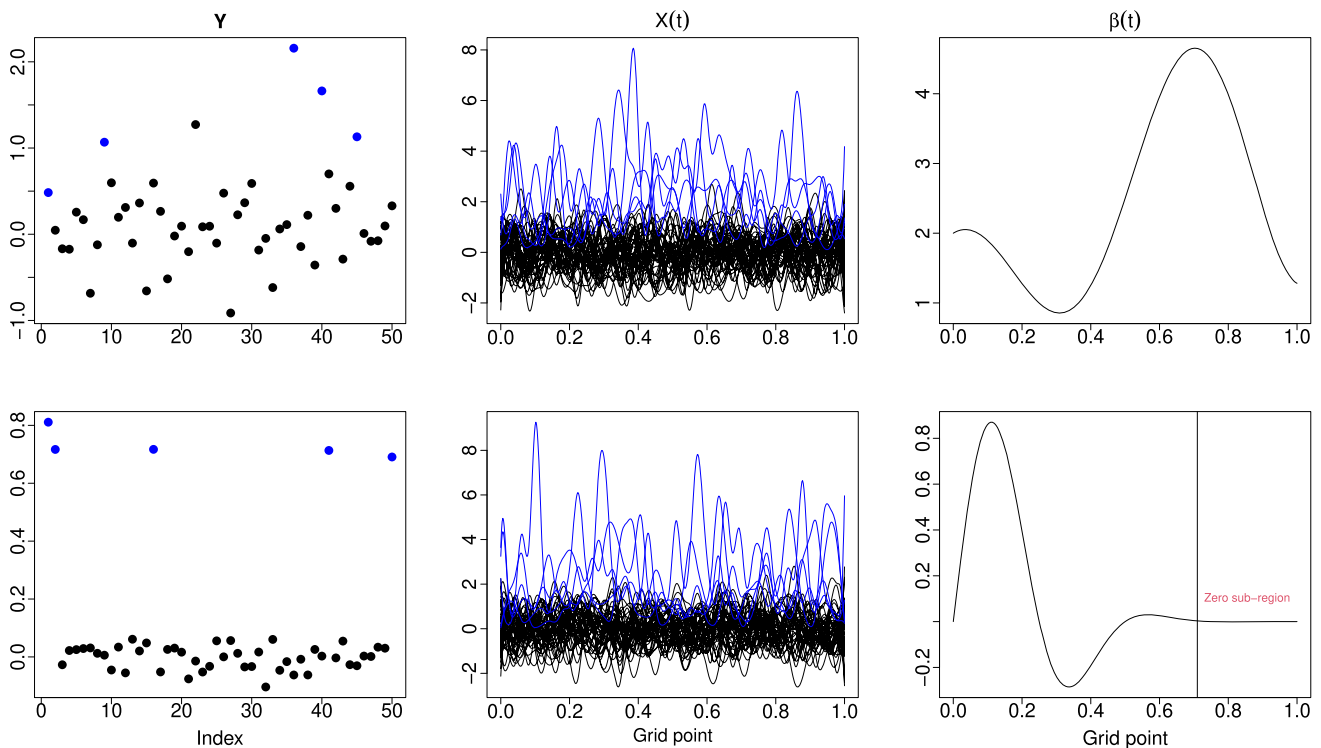


Fig. 1 Graphical display of the generated 50 scalar response (first column), functional predictor (second column), and the regression parameter function (third column) under DGP-I (first row) and DGP-

II (second row). For the scalar response and functional predictor, the observations generated by the DGP are shown as black, while the blue points and curves denote the outlying observations

where \mathcal{I}_0 denotes the zero sub-regions of $\beta(t)$, \mathcal{I}_1 denotes the nonzero sub-regions of $\beta(t)$, and ℓ_0 and ℓ_1 represent the lengths of the zero and nonzero sub-regions, respectively. For each training sample size, a test sample of size $n_{\text{test}} = 5000$ independent samples is generated. The predictive performance of the methods is evaluated by applying the fitted models based on the training sets to the test samples, and the mean squared prediction error (MSPE) is computed as follows:

$$\text{MSPE} = \frac{1}{5000} \sum_{i=1}^{5000} (Y_i - \widehat{Y}_i)^2,$$

where Y_i is the true response in the test sample, and \widehat{Y}_i is its predicted value obtained by the methods. It is important to note that, for all methods, the optimal values of the tuning parameters are determined through a $k = 10$ -fold cross-validation.

Figure 2 displays boxplots representing the logarithmic values of MSPE and ISE_1 recorded from 500 Monte-Carlo simulations conducted under DGP-I. In the absence of outliers, that is, at a contamination level of 0%, the FPLS and SFPLS* methods exhibit slightly superior performance in terms of MSPE and ISE_1 compared to other methods, exclud-

ing FPCR. Notably, FPCR shows the poorest estimation and predictive performance among all methods, as FPLS-based methods generate more informative components with fewer terms, contrasting with the richer information produced by FPCR (see, e.g., Delaigle and Hall 2012). The enhanced performance of SFPLS* over the proposed SFPLS might be attributed to the adaptive roughness penalty applied in the former, leading to improved control over the smoothness of function estimates and yielding superior results compared to the proposed SFPLS.

When the contamination level is 0%, all methods exhibit decreasing MSPE and ISE_1 values with increasing sample size, as expected. In the presence of outliers, non-robust methods (FPCR, FPLS, SFPLS*, and SFPLS) are significantly impacted, resulting in considerably larger MSPE and ISE_1 values than scenarios without outliers. In contrast, RFPLS and the proposed RSFPLS are less affected by outliers, producing notably smaller MSPE and ISE_1 values. The proposed RSFPLS, particularly for large sample sizes, remains resilient to outliers, yielding MSPE and ISE_1 values comparable to those observed in scenarios without outliers. In comparison, RFPLS experiences an increase in error metrics when outliers are present. This is attributed to the robust procedure used in tuning parameter selection for the proposed method. The robust procedure improves tun-

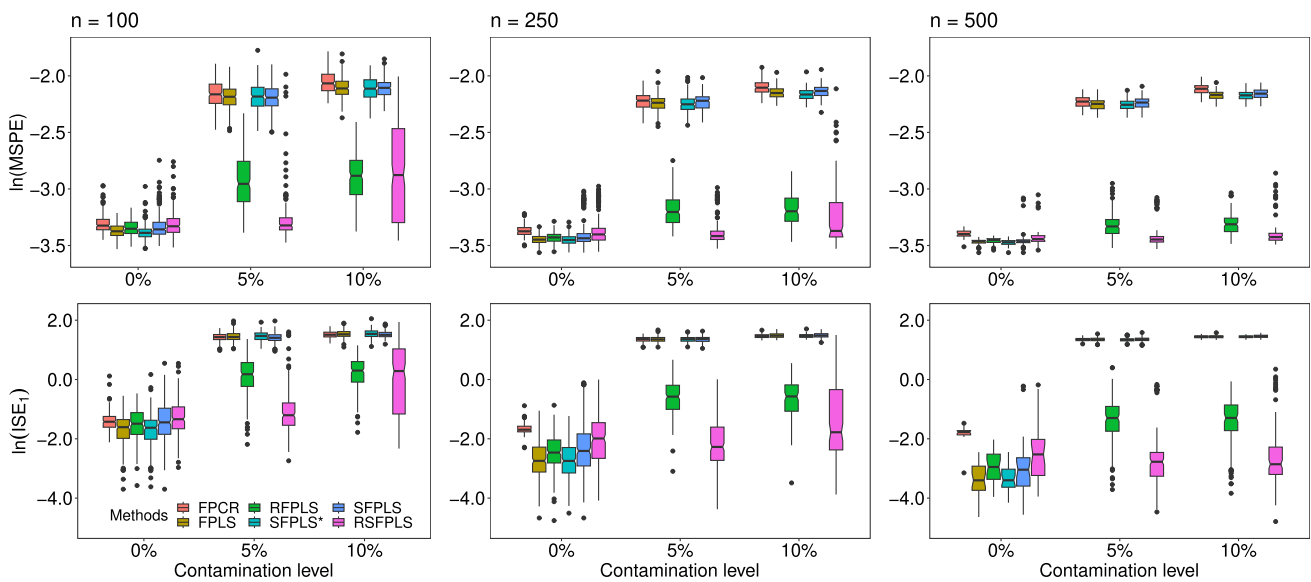


Fig. 2 Boxplots of the logarithmic MSPEs (first row) and logarithmic ISE_1 s (second row) under DGP-I when the training sample size is 100 (left panels), 250 (middle panels), and 500 (right panels). The MSPE

and ISE_1 values are computed for the FPCR, FPLS, RFPLS, SFPLS*, SFPLS, and RSFPLS methods when the contamination level is 0%, 5%, and 10%

ing parameter determination in the k -fold cross-validation for RSFPLS, leading to enhanced parameter estimates and predictions compared to RFPLS.

Figure 3 presents boxplots illustrating the logarithmic values of MSPE, logarithmic ISE_1 (defined on the nonzero sub-region), and logarithmic ISE_0 (defined on the zero sub-region) derived from Monte-Carlo simulations conducted under DGP-II. When the contamination level is 0%, all methods, excluding FPCR, exhibit similar MSPE and ISE_1 values for the nonzero sub-region. Consistent with the findings in Fig. 2, FPCR demonstrates the largest error metrics in this scenario. Notably, non-sparse methods, including FPCR, FPLS, and RFPLS, yield unsatisfactory ISE_0 values, close to their ISE_1 values. In contrast, sparse methods such as SFPLS*, SFPLS, and RSFPLS consistently produce small ISE_0 values, closely aligned. This indicates that all sparse methods accurately identify the zero sub-region in the absence of outliers. In the presence of outliers, all methods, excluding the proposed RSFPLS, are influenced by outlying observations, resulting in significantly larger error metrics compared to the scenario without outliers.

The proposed RSFPLS effectively mitigates the impact of outlier observations, leading to substantially smaller MSPE, ISE_1 , and ISE_0 values than other methods. Irrespective of the contamination level, the proposed RSFPLS consistently produces similar error metrics to those observed in scenarios without outliers, particularly for large sample sizes. This implies that the proposed RSFPLS is adept at defining both nonzero and zero sub-regions regardless of outlier presence. Conversely, the other sparse methods, SFPLS* and SFPLS,

yield nearly identical results for ISE_1 and ISE_0 in the presence of outliers, indicating a loss of sparse characteristics due to outlier influence.

Note that the proposed method does not produce a stable logarithmic ISE_0 when the generated data are contaminated by outliers and $n_{\text{train}} = [100, 500]$ (i.e., the left and right panels of the last row of Fig. 3). In these cases, the logarithmic ISE_0 values computed by the proposed method have a larger variance than those generated by other methods. Our detailed analyses show that the proposed method produces the smallest logarithmic ISE_0 (with values of -15) than the other methods in 90% of 500 Monte Carlo experiments. In contrast, it produces logarithmic ISE_0 ranging between -9 and -3 in the remaining 10% of Monte Carlo experiments. Additionally, from the bottom-left plot in Fig. 3, because the variance of the logarithmic ISE_0 values generated by the proposed method is larger when the contamination level is 10% compared to other scenarios, it might be thought that the proposed method yields better results when the contamination level is 10% than when the contamination level is 0%. However, in both scenarios, the proposed method has generated values around -12 on average. In other words, the proposed method tends to produce similar results in both 0% and 10% contamination levels.

Figures 4 and 5 depict the plots of the true regression coefficient functions, their estimates by the methods, and the mean estimated regression coefficient functions under DGP-I and DGP-II, respectively, when the sample size $n_{\text{train}} = 500$. These figures provide additional support to the findings presented in Figs. 2 and 3, confirming that our proposed method

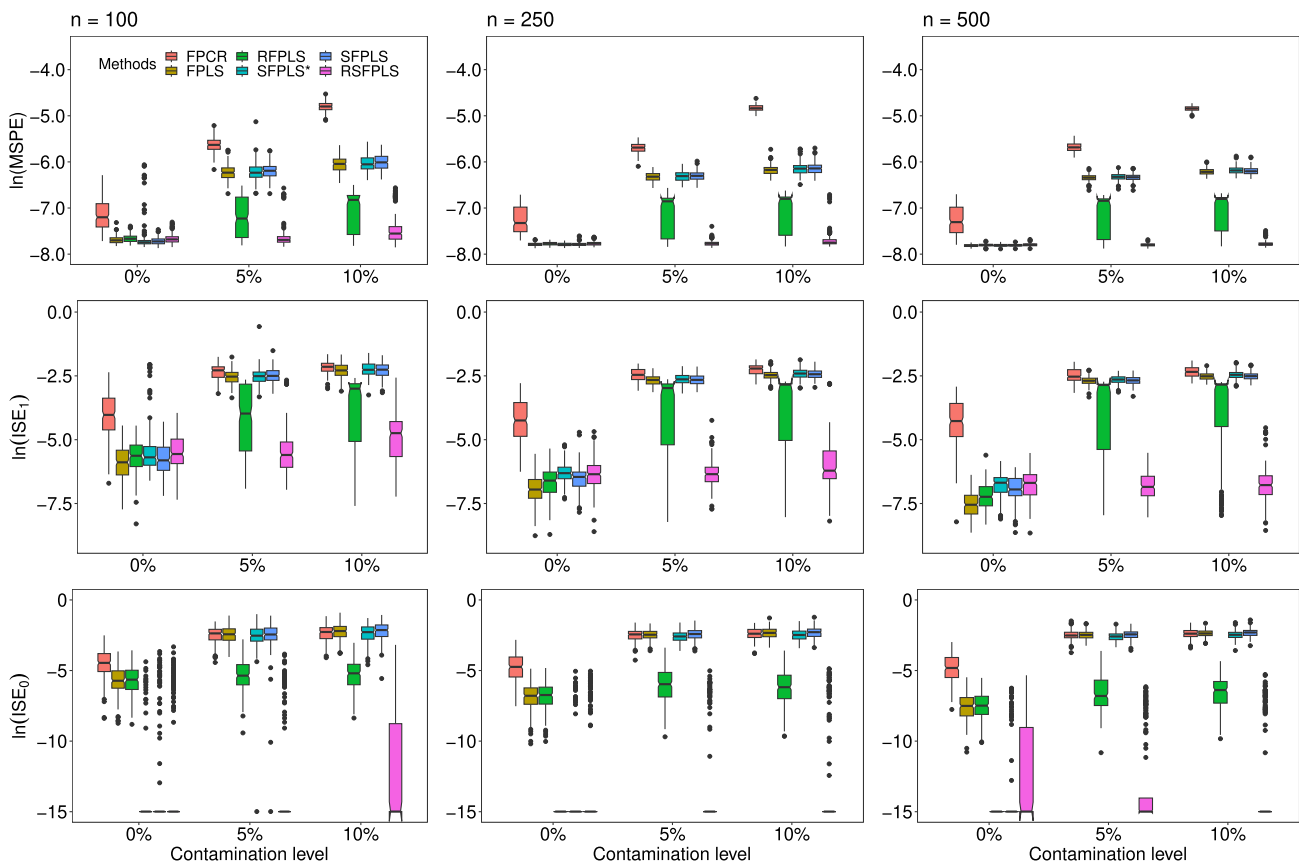


Fig. 3 Boxplots of the logarithmic MSPEs (first row), logarithmic ISE_1 s (second row), and logarithmic ISE_0 s under DGP-II when the training sample size is 100 (left panels) 250 (middle panels), and 500 (right panels). The MSPE, ISE_1 , and ISE_0 s values are computed for the FPCR, FPLS, RFPLS, SFPLS*, SFPLS, and RSFPLS methods when the contamination level is 0%, 5%, and 10%

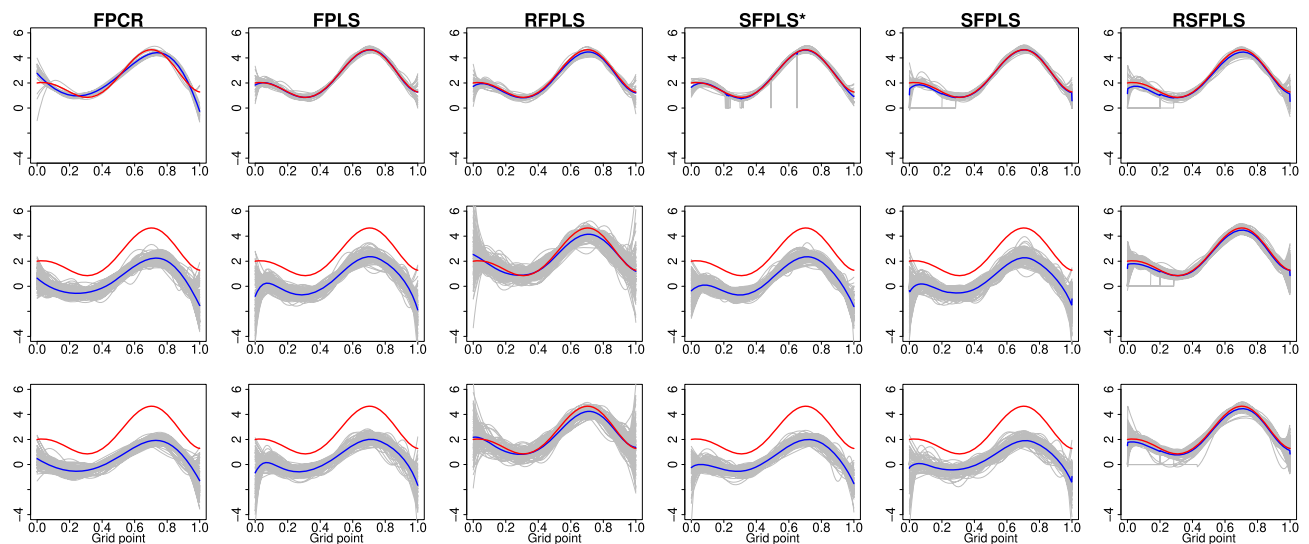


Fig. 4 Plots of the generated true parameter functions (red lines), estimated parameter functions (gray lines), and mean functions of the estimated parameter functions (blue lines). Parameter estimates are obtained under DGP-I when no outlier is present in the data (first row), 5% and 10% of the generated data are contaminated by outliers (second and third rows, respectively)

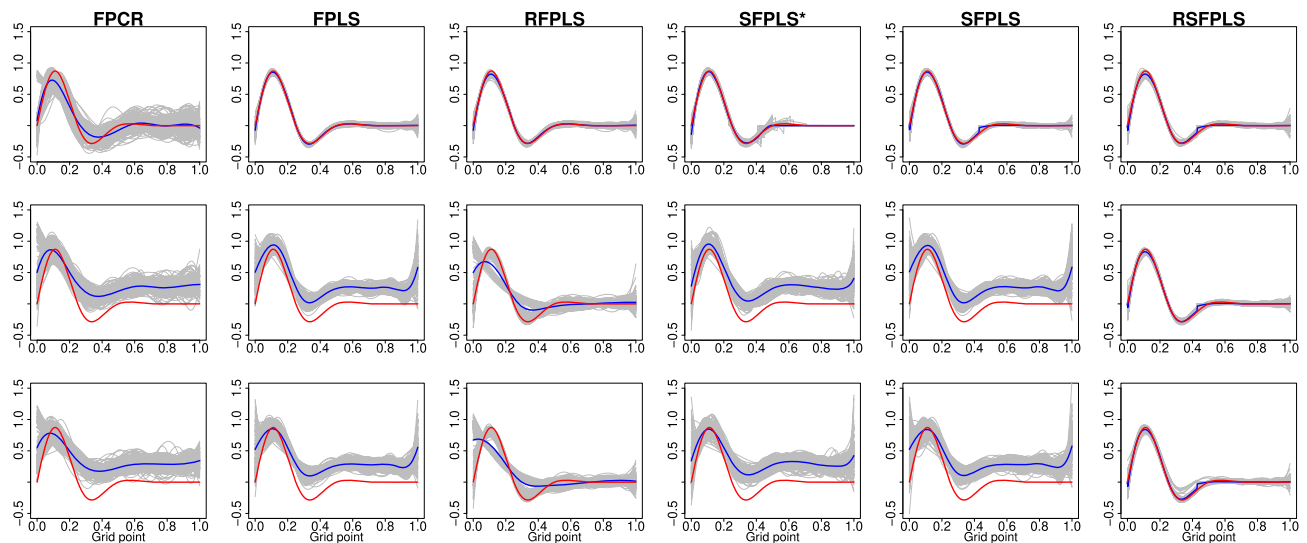


Fig. 5 Plots of the generated true parameter functions (red lines), estimated parameter functions (gray lines), and mean functions of the estimated parameter functions (blue lines). Parameter estimates are

obtained under DGP-II when no outlier is present in the data (first row), 5% and 10% of the generated data are contaminated by outliers (second and third rows, respectively)

yields improved parameter estimates for both nonzero and zero sub-regions in comparison to existing sparse and non-sparse methods.

Examining Fig. 4, where only the nonzero region exists for the regression parameter function, all methods demonstrate the ability to produce consistent estimates when no outliers are present. However, in the presence of outliers, the RFPLS and proposed RSFPLS methods consistently generate stable parameter estimates, closely aligning with the true parameter function across various contamination levels. In contrast, other methods exhibit unsatisfactory parameter estimates, and, notably, the estimates derived from the proposed RSFPLS method exhibit less variability than those obtained by the RFPLS method.

Moving to Fig. 5, which encompasses zero and nonzero sub-regions for the regression coefficient function, all methods, except for FPCR, accurately define both sub-regions and produce consistent estimates when no outliers are present. Among these methods, SFPLS* stands out by producing

slightly better estimates for the zero sub-region. However, in the presence of outliers, the proposed RSFPLS maintains its ability to generate consistent estimates for both the zero and nonzero sub-regions, while all other methods yield unsatisfactory parameter estimates.

We also thoroughly examine the computational efficiencies of the methods through a series of Monte-Carlo experiments. To this end, we perform a single Monte-Carlo simulation under both DGP-I and DGP-II in the absence of outliers, considering three distinct training sample sizes: $n_{\text{train}} = [100, 250, 500]$. The elapsed computing time for all methods is then recorded. These computations are executed on a desktop PC with an Intel(R) Core(TM) i5-9500 CPU @ 3.00GHz and 8 GB RAM. Notably, a 10-fold cross-validation is employed for all methods to ascertain the optimum values of the tuning parameters. The resulting computing times for both DGPs are detailed in Table 1.

Table 1 shows that non-sparse methods exhibit shorter computing times than sparse methods, given that sparse

Table 1 The elapsed computing times for the FPCR, FPLS, RFPLS, SFPLS*, SFPLS, and RSFPLS methods (in seconds)

DGP	n	FPCR	FPLS	RFPLS	SFPLS*	SFPLS	RSFPLS
I	100	7.525	2.551	5.557	99.380	15.313	34.106
	250	8.588	3.140	6.057	101.224	19.219	34.800
	500	12.460	5.056	8.100	108.306	30.610	63.008
II	100	7.005	2.282	4.150	47.754	14.375	26.464
	250	8.935	3.316	7.970	72.406	20.227	35.416
	500	12.544	5.269	8.803	98.421	30.535	56.732

The computing times are obtained from a single Monte-Carlo experiment when $n_{\text{train}} = [100, 250, 500]$ using $k = 10$ -fold cross-validation

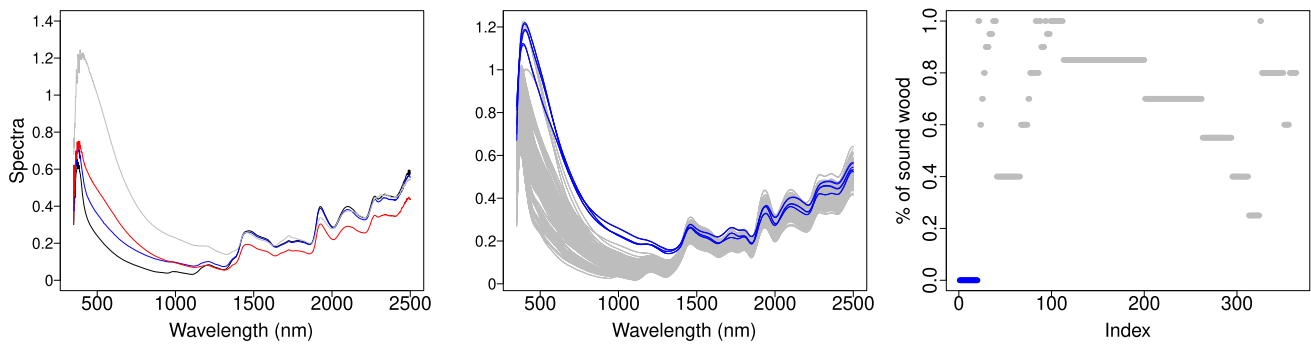


Fig. 6 Left panel: selected spectra traces of OSB samples with different proportions of sound wood (—, 100%; —, 80%; —, 40%; —, 0%); middle panel: spectra traces of OSB samples; right panel: sound wood proportions. Normal observations are given a gray color, while outliers are marked blue

methods involve tuning more parameters. Within the sparse category, SFPLS* demands considerably more computing time than SFPLS and RSFPLS due to the additional inclusion of a smoothing parameter. The proposed RSFPLS procedure, utilizing an M-estimation procedure involving iterative methods, needs approximately twice the computing time required by the SFPLS procedure.

4 Empirical data analysis: OSB data

In this section, an empirical dataset, OSB data (available at <https://github.com/caojiguo/SparFunPLS>, also see Guan et al. 2022) is analyzed to evaluate the empirical performance of the proposed method. With the data analysis, we aim to compare the predictive performance of our RFPLS method with FPCR, FPLS, RFPLS, SFPLS*, and SFPLS. In addition, we examine the functional relationship between the functional predictor and scalar response variables.

FPInnovations, a non-profit organization specializing in solutions for the Canadian forest sector’s global competitiveness, conducted a research project to obtain data about OSB. The research employed a pioneering laboratory spectroscopy technique to ascertain species identification of OSB strands and determine the relative proportions of sound wood, rot, and bark in OSB samples. Log section samples were sourced from Canadian OSB mills, separated into sound wood, rot, and bark groups, and processed into a coarse powder using a laboratory disk strander and grinder. Visible and near-infrared (Vis-NIR) spectroscopy with 2150 wavelengths (350 to 2500 nm) was employed to measure spectra traces.

Traditionally, the crucial raw material components (sound wood, rot, and bark) influencing production efficiency and final product attributes are not systematically monitored in mills. Periodic monitoring is used to identify issues related to log rot, debarking inefficiency, and species, and the information is used for process adjustments, production planning, and budgeting.

In the research, 60 cookies from different logs were collected, representing various log types. Samples underwent processing, drying, and segregation, resulting in 182 mixtures with different sound wood proportions. Benchtop Vis-NIR spectrometer ASD 5000 scanned powder samples, providing spectra with 2150 wavelengths (350 to 2500 nm). As is standard in the industry, Vis-NIR spectroscopy records color composition in the visible range and interactions between light and molecular structures in the NIR range. Each sample had three spectra replicates, contributing to a spectral file of log inverse reflectance versus wavelength.

In the left panel of Fig. 6, diverse spectral traces correspond to varying proportions of sound wood, forming the foundation for predicting sound wood proportions in OSB samples through spectral analysis. The OSB dataset comprises $n = 364$ spectra curves and wood proportions, displayed graphically in the middle and right panels of Fig. 6. To identify outliers in the scalar response (wood proportions in OSB samples) for this dataset, we employ the IQR method. Utilizing the IQR criterion, we identify 20 clear outliers (those with 0% wood proportions) in the response variable, as depicted in the right panel of Fig. 6. For the functional predictor, we employ the functional boxplot, the results of which reveal the presence of six outlying curves in the data, as illustrated in the middle panel of Fig. 6. Previous findings reported in Guan et al. (2022) indicated that the SFPLS* estimate for the slope function is zero over wavelengths 587–2285 nm, indicating a clear zero sub-region where the spectra are unrelated to sound wood proportions. The culmination of these analyses motivates our application of the RSFPLS method to robustly model wood proportions and the associated spectra. It is essential to acknowledge that the curves flagged as outliers by the functional boxplot (or any other suitable methods) may not definitively be outliers. Nevertheless, within this context, we regard these curves as potential outliers.

To assess and contrast the predictive efficacy of both the proposed and existing methods for the OSB data, the subsequent process is iterated 100 times. Initially, we randomly

Table 2 Upper 20% MSPE values ($\times 10^{-3}$) and their standard errors ($\times 10^{-3}$) given in brackets for the FPCR, FPLS, RFPLS, SFPLS*, SFPLS, and RSFPLS methods computed from 100 experiments from the OSB data

FPCR	FPLS	RFPLS	SFPLS*	SFPLS	RSFPLS
4.095	3.323	3.054	3.879	3.472	2.451
(0.776)	(0.503)	(0.463)	(0.373)	(0.718)	(0.410)

partition the complete dataset into a training set comprising $n_{\text{train}} = 200$ observations and a test set comprising $n_{\text{test}} = 164$ observations. Subsequently, using the training set, we formulate models to forecast wood proportions in the test set. The upper 20% MSPE values are then computed for each iteration, allowing for a comparative analysis of the methods. This trimming is crucial for gauging the predictive accuracy of the standard data, as certain excluded observations may be outliers. Tuning parameters for all approaches in this dataset are determined through $k = 10$ -fold cross-validation, mirroring the procedure employed in the Monte Carlo experiments. Note that for this dataset, both the scalar response and functional predictor are centered before fitting the model as proposed by Guan et al. (2022).

Our findings are presented in Table 2, revealing that our proposed RSFPLS technique outperforms existing methods, whether sparse/non-sparse or robust/non-robust. In the table, the second-best MSPE value, which is 20% larger than those of the proposed method, is produced by the RFPLS. This discrepancy arises because our method delivers superior estimates for tuning parameters and effectively accommodates sparsity, a capability lacking in the RFPLS. Compared to other sparse techniques like SFPLS* and SFPLS, the enhanced performance of our proposed method stems from its ability to mitigate the impact of outliers in the data. In contrast, non-robust sparse methods are adversely affected by outlier observations, resulting in less accurate predictions when compared to our proposed method.

All methods considered estimate the regression parameter function using the entire dataset, as illustrated in Fig. 7. In this figure, non-sparse models, namely FPCR, FPLS, and RFPLS, yield slope function estimates without distinct sub-regions where the spectra are unrelated to sound wood proportions. Conversely, sparse methods, namely SFPLS*, SFPLS, and RSFPLS, provide slope function estimates that exhibit zero values over specific wavelength ranges: 587–2285 nm, 1068–2500 nm, and 888–2319 nm, respectively.

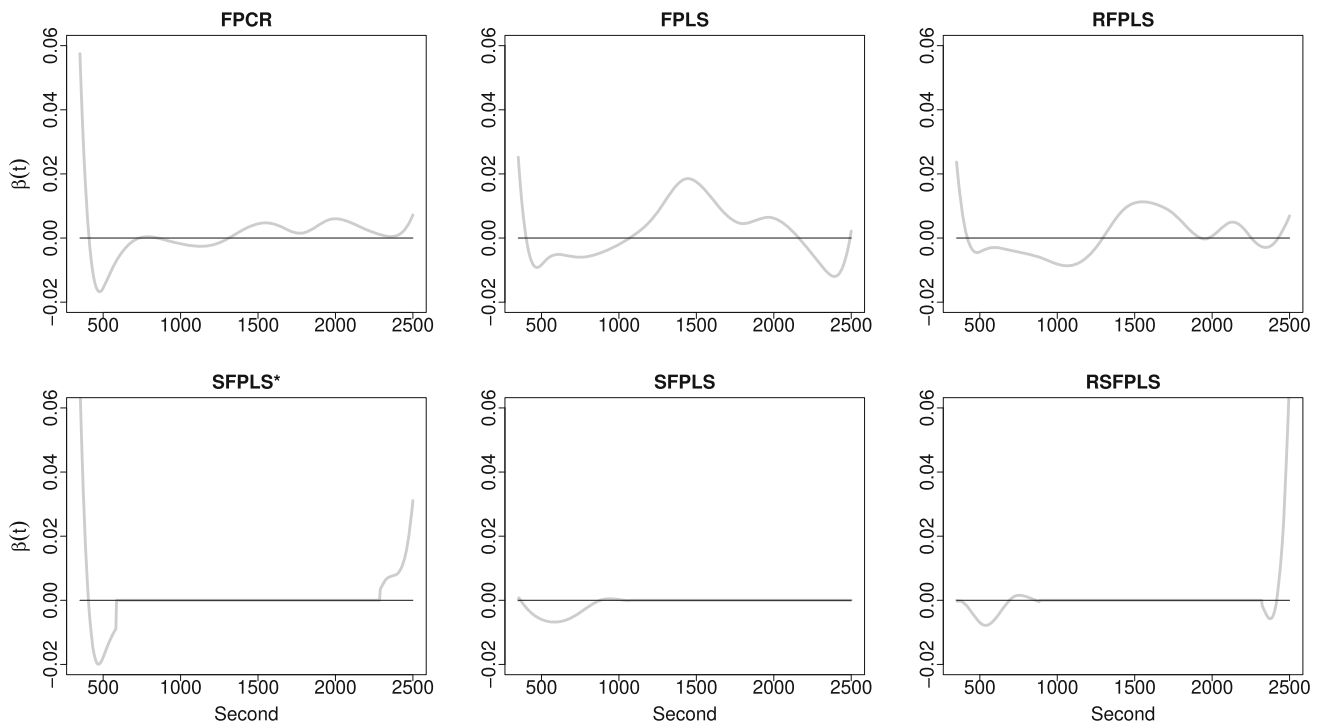


Fig. 7 Regression parameter function for the OSB data was estimated by FPCR, FPLS, RFPLS, SFPLS*, SFPLS, and RSFPLS

5 Conclusion

We introduce an innovative RSFPLS approach for estimating regression coefficients within the context of a scalar-on-function regression model. Our method leverages sparse partial robust M regression to derive the FPLS components. Subsequently, the M-estimator, combined with Hampel’s loss function, is employed to obtain the final estimates. In our approach, incorporating sparse partial robust M regression ensures robust parameter estimation in the scalar-on-function regression model. This facilitates reliable predictions even in the presence of outliers and leverage points. Furthermore, it accurately discerns zero and nonzero sub-regions, where the slope function is estimated, even in the presence of outliers and leverage points. We evaluate our approach’s finite-sample estimation and predictive performance through a series of Monte Carlo experiments and an empirical dataset, benchmarking against FPCR, FPLS, RSFPLS, SFPLS*, and SFPLS methods. Our numerical findings reveal that the proposed method yields comparable estimation and predictive performance to existing methods when outliers are absent. Remarkably, it surpasses existing methods when outliers influence the predictor and response variables.

While our proposed RSFPLS method demonstrates improved performance over the existing SFPLS* method of Guan et al. (2020) in the presence of outliers, the SFPLS* method excels in producing better estimates for the zero sub-region when no outliers are present in the data, as illustrated in the first row of Fig. 5. This may be attributed to two reasons. First, the choice of penalty function used to recover the sparsity profile of the true coefficient function differs between the two methods. Our proposed method employs an L_1 penalty, which is convex (Tibshirani 1996), whereas the SFPLS* method utilizes a non-concave smooth clipped absolute deviation (SCAD) penalty. Under certain conditions, variable selection with L_1 penalization can be consistent and capable of recovering the sparsity profile of the true coefficient (see, e.g., Meinshausen and Bullman 2006; Donoho and Huo 2001; Tropp 2006). However, as noted by Fan and Li (2001), an ideal penalty function should yield an estimator with three properties: unbiasedness, sparsity, and continuity. While the SCAD penalty satisfies all these conditions, the L_1 penalty does not fulfill the unbiasedness condition, meaning the SCAD penalty improves upon the L_1 penalty by reducing estimation bias. Generally, SCAD behaves similarly to the L_1 penalty for small values but levels off for larger values. Therefore, the SFPLS* method based on the SCAD penalty may recover the sparsity profile of the true coefficient better than our proposed method. Second, the algorithms employed to obtain the sparse FPLS basis functions differ between the two methods. In the SFPLS* method, the FPLS basis functions are computed from the defined active regions (zero and nonzero active regions), where all coefficients are initially set to zero.

These active regions are then updated at each PLS iteration, and the regression coefficient function is expanded by the FPLS basis functions, being nonzero only in the active regions. Since all coefficients start as zero, and those with small magnitudes in the zero active region are set to zero at each iteration, the algorithm of Guan et al. (2020) ensures the sparsity of the regression coefficient function. Conversely, in our algorithm, the FPLS basis functions are computed from the B-spline basis expansion coefficients, and these coefficients are not initialized to zero. The eigenvectors computed from the sparse NIPALS algorithm, used in our method, may not have the same zero sub-regions (i.e., estimated coefficients with small magnitudes in different non-overlapping eigenvectors are not set to zero). Thus, the estimated regression coefficient function may not be fully sparse in some cases. This discrepancy is illustrated in Fig. 5. However, the estimated mean regression coefficient function by our proposed method matches exactly with the true coefficient function in the zero sub-region.

Furthermore, in addition to the aforementioned factors, the SFPLS* method incorporates an additional penalization parameter that governs the smoothness of the regression coefficient function, a feature absent in our approach. This supplementary penalization could also exert a minor influence on the sparse estimation process.

The L_1 penalization employed in our approach to uncover the sparsity pattern of the true coefficient is effective in many scenarios, but it comes with its limitations. For example, when the number of predictor variables exceeds the number of observations, the L_1 penalization tends to select a maximum of n (sample size) variables before reaching saturation due to its convex nature. Moreover, in cases where there exists a high pairwise correlation among a set of variables, the L_1 penalization tends to choose only one variable from the group without regard for which one is selected (see, e.g. Zou and Hastie 2005). Consequently, our proposed method may not yield improved outcomes in such situations.

Our proposed methodology presents several avenues for expansion and exploration, as follows.

- 1) The present study considers a single functional predictor variable in the scalar-on-function regression model. Our proposed method can be extended to improve further estimation and predictive performance by using multiple functional predictors to estimate the model.
- 2) We consider only functional predictors. However, one may prefer a mixed data type consisting of functional and scalar predictors, such as a partial functional linear model, to explain the scalar response variation. Our proposed method can be extended to this model when the scalar predictors are included.

- 3) The RSFPLS method can also be extended to the function-on-function-regression model. The response and predictor variables are random curves, as an alternative to Bernardi et al. (2022).
- 4) The estimation performance of the proposed method can be improved by utilizing a penalization term in the model to control the smoothness of the functional variables as in Aguilera et al. (2016); that is, a penalized version of our proposed method can be developed to obtain improved estimation results for the scalar-on-function regression.
- 5) As discussed above, the estimation algorithm used in the SFPLS* method may lead to better estimation of the zero sub-region compared to the one utilized in the proposed method. Extending the estimation algorithm of the SFPLS* method to the proposed method may result in improved outcomes.

Acknowledgements We thank two reviewers for their careful reading of our manuscript and valuable suggestions and comments, which have helped us produce an improved version of our article. The second author acknowledges the funding of an Australian Research Council Discovery Project DP230102250.

Author Contributions All authors contributed to the study's conception and design. Material preparation, data collection, and analysis were performed by Sude Gurer, Han Lin Shang, Abhijit Mandal, and Ufuk Beyaztas. The first draft of the manuscript was written by Sude Gurer and Ufuk Beyaztas and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK).

Declarations

Conflict of interest The authors declare no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aguilera, A.M., Aguilera-Morillo, M.C., Preda, C.: Penalized versions of functional PLS regression. *Chemom. Intell. Lab. Syst.* **154**, 80–92 (2016)
- Aguilera, A.M., Escabias, M., Preda, C., Saporta, G.: Using basis expansions for estimating functional PLS regression applications with chemometric data. *Chemom. Intell. Lab. Syst.* **104**, 289–305 (2010)
- Alin, A., Agostinelli, C.: Robust iteratively reweighted SIMPLS. *J. Chemom.* **31**(3), e2881 (2017)
- Beaton, A.E., Tukey, J.W.: The fitting power series, meaning polynomials, illustrated on band-spectroscopic data. *Technometrics* **16**(2), 147–185 (1974)
- Bernardi, M., Canale, A., Stefanucci, M.: Locally sparse function-on-function regression. *J. Comput. Graph. Stat.* **32**(3), 985–999 (2022)
- Beyaztas, U., Shang, H.L.: A robust functional partial least squares for scalar-on-multiple-function regression. *J. Chemom.* **36**(4), e3394 (2022)
- Beyaztas, U., Shang, H.L.: A robust partial least squares approach for function-on-function regression. *Braz. J. Probab. Stat.* **36**(2), 199–219 (2022)
- Beyaztas, U., Shang, H.L.: Robust functional linear regression models. *The R J.* **15**(1), 212–233 (2023)
- Beyaztas, U., Tez, M., Hang, H.L.: Robust scalar-on-function partial quantile regression. *J. Appl. Stat.* **51**(7), 359–1377 (2023)
- Cai, T.T., Hall, P.: Prediction in functional linear regression. *Ann. Stat.* **34**(5), 2159–2179 (2006)
- Cardot, H., Ferraty, F., Sarda, P.: Functional linear model. *Stat. Probab. Lett.* **45**(1), 11–22 (1999)
- Cardot, H., Ferraty, F., Sarda, P.: Spline estimators for the functional linear model. *Stat. Sin.* **13**(3), 2159–2179 (2003)
- Chun, H., Keleş, S.: Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *J. R. Stat. Soc. Ser. B* **72**(1), 3–25 (2010)
- de Boor, C.: *A Practical Guide to Splines*. Springer-Verlag, New York (2001)
- Delaigle, A., Hall, P.: Methodology and theory for partial least squares applied to functional data. *Ann. Stat.* **40**(1), 322–352 (2012)
- Donoho, D.L., Huo, X.: Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inf. Theory* **47**(7), 2845–2862 (2001)
- Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Stat. Assoc.* **96**(456), 1348–1360 (2001)
- Febrero-Bande, M., Galeano, P., Gonzalez-Manteiga, W.: Functional principal component regression and functional partial least-squares regression: an overview and a comparative study. *Int. Stat. Rev.* **85**(1), 61–83 (2017)
- Gil, J.A., Romera, R.: On robust partial least squares (PLS) methods. *J. Chemom.* **12**(6), 365–378 (1998)
- Goldsmith, J., Bobb, J., Crainiceanu, C.M., Caffo, B., Reich, D.: Penalized functional regression. *J. Comput. Graph. Stat.* **20**(4), 830–851 (2011)
- Goldsmith, J., Scheipl, F.: Estimator selection and combination in scalar-on-function regression. *Comput. Stat. Data Anal.* **70**, 362–372 (2014)
- Gonzalez, J., Pena, D., Romera, R.: A robust partial least squares regression method with applications. *J. Chemom.* **23**(2), 37–50 (2009)
- Griep, M.L., Wakeling, I.N., Vankeerberghen, P., Massart, D.L.: Comparison of semirobust and robust partial least squares procedures. *Chemom. Intell. Lab. Syst.* **29**(1), 37–50 (1995)
- Guan, T., Lin, Z., Cao, J.: Estimating truncated functional linear models with a nested group bridge approach. *J. Comput. Graph. Stat.* **29**(3), 620–628 (2020)
- Guan, T., Lin, Z., Groves, K., Cao, J.: Sparse functional partial least squares regression with a locally sparse slope function. *Stat. Comput.* **32**(30) (2022)
- Hall, P., Hooker, G.: Truncated linear models for functional data. *J. R. Stat. Soc. Ser. B* **78**(3), 637–653 (2016)
- Hall, P., Horowitz, J.L.: Methodology and convergence rates for functional linear regression. *Ann. Stat.* **35**(1), 70–91 (2007)
- Hampel, F.R.: The influence curve and its role in robust estimation. *J. Am. Stat. Assoc. Theory Methods* **69**(346), 383–393 (1974)

- Hampel, F.R., Ronchetti, E.M., Rousseeuw, P.J., Stahel, W.A.: Robust Statistics. The Approach Based on Influence Functions/ John Wiley & Sons, New York (1986)
- Hastie, T., Mallows, C.: A statistical view of some chemometrics regression tools: discussion. *Technometrics* **35**(2), 140–143 (1993)
- Hoffmann, I., Serneels, S., Filzmoser, P., Croux, C.: Sparse partial robust M regression. *Chemom. Intell. Lab. Syst.* **149**, 50–59 (2015)
- Huber, P.J.: Robust estimation of a location parameter. *Ann. Math. Stat.* **35**(1), 73–101 (1964)
- Huber, P.J.: Robust Statistics. John Wiley & Sons, New York (1981)
- Hubert, M., Branden, K.V.: Robust methods for partial least squares regression. *J. Chemom.* **17**(10), 537–549 (2003)
- James, G.M., Wang, J., Zhu, J.: Functional linear regression that’s interpretable. *Ann. Stat.* **37**(5A), 2083–2018 (2009)
- Jolliffe, I.T., Trendafilov, N.T., Uddin, M.: A modified principal component technique based on LASSO. *J. Comput. Graph. Stat.* **12**(3), 531–547 (2003)
- Lee, D., Lee, W., Lee, Y., Pawitan, Y.: Sparse partial least-squares regression and its applications to high-throughput data analysis. *Chemom. Intell. Lab. Syst.* **109**(1), 1–8 (2011)
- Lee, E.R., Park, B.U.: Sparse estimation in functional linear regression. *J. Multivar. Anal.* **105**(1), 1–17 (2012)
- Lin, Z., Cao, J., Wang, L., Wang, H.: Locally sparse estimator for functional linear regression models. *J. Comput. Graph. Stat.* **26**(2), 306–318 (2017)
- Marx, B.D., Eilers, P.H.C.: Generalized linear regression on sampled signals and curves: a p-spline approach. *Technometrics* **41**(1), 1–13 (1999)
- Meinshausen, N., Bühlman, P.: High-dimensional graphs and variable selection with the Lasso. *Ann. Stat.* **34**(3), 1436–1462 (2006)
- Morris, J.S.: Functional regression. *Annu. Rev. Stat. Appl.* **2**(1), 321–359 (2015)
- Ojo, O.T., Lillo, R.E., Anta, A.F.: fdaoutlier: outlier detection tools for functional data analysis. R package version 0.2.1. <https://CRAN.R-project.org/package=fdaoutlier> (2023)
- Polat, E., Gunay, S.: A new robust partial least squares regression method based on a robust and efficient adaptive reweighted estimator of covariance. *Comput. Stat. Data Anal.* **17**(4), 449–474 (2019)
- Preda, C., Saporta, G.: PLS regression on a stochastic process. *Comput. Stat. Data Anal.* **48**(1), 149–158 (2005)
- Ramsay, J.O., Silverman, B.W.: Functional Data Analysis. Springer, New York (2006)
- Reiss, P.T., Goldsmith, J., Shang, H.L., Odgen, R.T.: Methods for scalar-on-function regression. *Int. Stat. Rev.* **85**(2), 228–249 (2017)
- Reiss, P.T., Odgen, R.T.: Functional principal component regression and functional partial least squares. *J. Am. Stat. Assoc. Theory Methods* **102**(479), 984–996 (2007)
- Rousseeuw, P.J., Croux, C.: Alternatives to the median absolute deviation. *J. Am. Stat. Assoc.* **88**(424), 1273–1283 (1993)
- Serneels, S., Croux, C., Filzmoser, P., Espen, P.J.V.: Partial robust M-regression. *Chemom. Intell. Lab. Syst.* **79**(1–2), 55–64 (2005)
- Stone, M., Brooks, R.J.: Continuum regression: cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal components regression. *J. R. Stat. Soc. B* **52**(2), 237–258 (1990)
- Sun, Y., Genton, M.G.: Functional boxplots. *J. Comput. Graph. Stat.* **20**(2), 316–334 (2011)
- Tenenhaus, M.: La régression PLS, théorie et pratique, PhD thesis, Editions Technip, Paris (1998)
- Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **58**(1), 267–288 (1996)
- Tropp, J.A.: Just relax: convex programming methods for identifying sparse signals in noise. *IEEE Trans. Inf. Theory* **52**(3), 1030–1051 (2006)
- Tu, C.Y., Song, D., Breidt, F.J., Berger, T.W., Wang, H.: Functional model selection for sparse binary time series with multiple inputs. In: Bell, W.R., Holan, S.H., McElroy, T.S. (eds.) *Economic Time Series*, pp. 477–497. Chapman and Hall/CRC, Oxford (2012)
- Wakelinc, I.N., Macfie, H.J.H.: A robust PLS procedure. *J. Chemom.* **6**(4), 189–198 (1992)
- Wang, H., Kai, B.: Functional sparsity: global versus local. *Stat. Sin.* **25**(4), 1337–1354 (2015)
- Wang, J.-L., Chiou, J.-M., Müller, H.-G.: Functional data analysis. *Annu. Rev. Stat. Appl.* **3**, 257–295 (2016)
- Wilcox, R.: Introduction to Robust Estimation and Hypothesis Testing. Elsevier, Waltham (2012)
- Wold, H.: Causal flows with latent variables: Partings of the ways in the light of NIPALS modelling. *Eur. Econ. Rev.* **5**(1), 67–86 (1974)
- Yao, F.: Functional principal component analysis for longitudinal and survival data. *Stat. Sin.* **17**(3), 965–983 (2007)
- Yu, D., Kong, L., Mizera, I.: Partial functional linear quantile regression for neuroimaging data analysis. *Neurocomputing* **195**, 74–87 (2016)
- Zhao, Y., Odgen, R.T., Reiss, P.T.: Wavelet-based LASSO in functional linear regression. *J. Comput. Graph. Stat.* **21**(3), 600–617 (2012)
- Zhou, J., Wang, N.-Y., Wang, N.: Functional linear model with zero-value coefficient function at sub-regions. *Stat. Sin.* **23**(1), 25–50 (2013)
- Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. *J. R. Stat. Soc. B* **67**(2), 301–320 (2005)
- Zou, H., Hastie, T., Tibshirani, R.: Sparse principal component analysis. *J. Comput. Graph. Stat.* **15**(2), 265–286 (2006)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.