Research article

Estimation and diagnostic for single-index partially functional linear regression model with p-order autoregressive skew-normal errors

Lijie Zhou[1]**, Liucang Wu**[1] **and Bin Yang**[2,*]

[1] Faculty of Science, Kunming University of Science and Technology, Kunming 650500, China

[2] City College, Kunming University of Science and Technology, Kunming 650500, China

* **Correspondence:** Email: byang@kust.edu.cn.

Abstract: This paper introduced a novel single-index partially functional linear regression model with p-order autoregressive skew-normal errors, addressing the dual challenges of autocorrelation and skewness in high-dimensional functional data. We proposed an innovative EM-CALS algorithm, which synergizes the expectation-maximization (EM) framework with conditional adaptive least squares to estimate constrained parameters and functional components efficiently. The algorithm uniquely addressed the unit-norm constraint of single-index vectors through reparameterization, thereby overcoming the limitations of conventional EM in nonlinear optimization. Comprehensive simulations demonstrated EM-CALS's superiority over the two-step iterative least squares (TSILS) algorithm, achieving reductions of 1.19% in root mean square error (RMSE), 2.45% in mean square error (MSE), 38.00% in root average squared errors (RASE$_1$), and 51.35% in RASE$_2$, respectively, demonstrating a clear advantage in enhancing prediction accuracy. Additionally, we conducted residual analysis based on conditional quantiles, considering the skew-normal distribution and autocorrelation of the residuals, and performed local influence analysis using the Q-function in the EM algorithm. The efficiency of the EM-CALS algorithm was validated through simulation studies. Finally, the methodology was applied to an empirical dataset from photovoltaic power forecasting, underscoring its practical applicability. This work bridged significant gaps in functional data analysis by simultaneously addressing dimension reduction, temporal dependence, and distributional asymmetry, which are crucial challenges in modern energy analytics and biomedical studies.

Keywords: single-index partially functional linear regression; B-spline; EM-CALS algorithm; autoregressive errors; skew-normal distribution; adaptive least squares; local influence
Mathematics Subject Classification: 62J05, 62M10

1. Introduction

With advances in information technology, functional data collected as curves or images is common in econometrics and biomedicine. However, the classical statistical method does not perform well when applied directly to such data, which has driven the development of functional data analysis (FDA). Researchers have extensively studied the functional regression model as an essential part of the FDA. For example, Cardot et al. [1] introduced the functional linear model, and due to the flexibility of semi-parametric models, significant research on functional semi-parametric regression has been conducted since 2008. For instance, Şentürk et al. [2] extended the traditional varying-coefficient model to functional data, proposing the functional varying-coefficient model. Aneiros-Perez et al. [3] proposed the semi-functional partially linear model, combining the advantages of semi-linear modeling and the nonparametric form of functional data. Zhou et al. [4] introduced the semi-functional linear model, studied the spline estimation of functional coefficients and nonparametric functions, and derived the convergence rate of these spline estimates.

When the dimensionality of multivariate covariates becomes excessively high, the curse of dimensionality inevitably emerges. The single-index model captures the key features of high-dimensional data by searching for a univariate index of multivariate covariates, modeling the effects of the covariates in a nonparametric manner. Härdle et al. [5] applied the single-index model to discrete choice analysis in econometrics and dose-response modeling in biostatistics. Zou et al. [6] studied the M-estimators for the single-index model, approximated the unknown link function with B-splines, and obtained the M-estimators for both the parametric and nonparametric components in a single step. They also proved the asymptotic normality of the estimators. In functional data analysis, to avoid the curse of dimensionality while preserving the advantages of nonparametric smoothing, Yu et al. [7] combined the single-index model with functional linear regression models and proposed the single-index partially functional linear regression model.

However, the above functional regression models assume that the errors are independent. In reality, data often exhibits autocorrelation, such as in financial series or photovoltaic system outputs. For regression analysis of such data, it is necessary to assume that the errors are autocorrelated. To address this issue, Dabo-Niang et al. [8] proposed the functional semi-parametric partially linear model with autoregressive errors and obtained estimates of these coefficients using generalized least squares, proving the consistency and asymptotic normality of the estimators. Yang et al. [9] proposed a robust estimation method for the semi-functional linear model with autoregressive errors. Xiao et al. [10] introduced the partial functional linear model with autoregressive errors, estimated multivariate regression parameters, and functional regression coefficients using generalized least squares, and proved the asymptotic normality of multivariate coefficients and the optimal convergence rate of the functional regression parameters. When the dimensionality of multivariate covariates in a partially functional linear model becomes excessively high, the curse of dimensionality often arises. To address this issue, this paper introduces a single-index partially functional linear model with autoregressive errors for analyzing autocorrelated data.

The research above assumes that the error terms follow a normal distribution. However, in practice, the data may show skewness. To address this skewness issue, Azzalini et al. [11] proposed the skew-normal (SN) distribution, which accommodates skewness and includes the normal distribution as a particular case. Xiao et al. [12] proposed a new asymmetric Huber regression (AHR) estimation

method for analyzing the partially functional linear models with skewed data and derived the asymptotic properties of the proposed estimator. In multivariate data analysis, Ferreira et al. [13] proposed an estimation method for the partially linear model with first-order autoregressive (AR(1)) skew-normal errors and conducted local influence analysis. Liu et al. [14] introduced a Bayesian local influence method for detecting influential observations in the partially linear model with first-order autoregressive skew-normal errors and performed local influence analysis. Ferreira et al. [15] extended the autoregressive order to p-order, proposing the partially linear model with p-order autoregressive (AR) skew-normal errors and deriving the maximum likelihood estimator using the EM algorithm. We extend this autoregressive error distribution to models involving functional data. We investigate the parameter estimation problem for the single-index partially functional linear model with p-order autoregressive skew-normal errors.

In the study of parameter estimation for functional data models with autoregressive errors, Chen et al. [16] investigated a functional partial linear model with autoregressive errors, used weighted least squares to estimate spline coefficients, and derived theoretical properties such as the convergence rates of the function regression parameters and nonparametric function estimates for scalar predictor variables. Wang et al. [17] studied a multivariate functional linear model with autoregressive errors, employed least squares to estimate the function coefficients and autoregressive coefficients, and proved the asymptotic properties of the proposed estimators. Inspired by the study of Ferreira et al. [15], this paper proposes a parameter estimation method based on the EM algorithm for the single-index partially functional linear model with p-order autoregressive skew-normal errors. Due to the constraints on the single-index vector, the standard EM algorithm cannot be directly applied. Therefore, a new EM algorithm is proposed, which introduces a conditional adaptive least squares (CALS) step after the conventional EM steps. This step handles the constraint issue through reparameterization and uses the adaptive least squares method to estimate the single-index vector.

The main objective of sensitivity analysis is to assess the impact of perturbations in the model or data on parameter estimates. A commonly used method is case deletion, which evaluates the influence of each observation on the parameter estimates. However, this method does not directly capture the impact of other perturbations in the model. To address this limitation, Cook [18] proposed the local influence method, which studies the sensitivity of the log-likelihood to small perturbations in parts of the model and is computationally simpler. Building upon Cook's pioneering work, Zou et al. [19] extended this method to partially linear single-index models. In the study of autocorrelated data, Ferreira et al. [13] used the EM algorithm to estimate the unknown parameters in partially linear models with first-order autoregressive AR(1) skew-normal errors and conducted a local influence analysis using the conditional expectation of the complete-data log-likelihood function. Inspired by these works, we extend the local influence method to the single-index partially functional linear regression models with p-order autoregressive skew-normal errors.

The structure of this paper is as follows. Section 2 describes in detail the single-index partially functional linear model with p-order autoregressive skew-normal errors. Section 3 introduces the proposed EM-CALS algorithm, compares it with the TSILS, and presents a residual analysis based on conditional quantiles. Section 4 conducts local influence analysis using the Q-function in the EM algorithm. Section 5 evaluates the efficiency of the EM-CALS algorithm through simulation studies. Section 6 demonstrates the application of the proposed method using actual data from grid-connected photovoltaic systems. Section 7 provides some discussions and summarizes the contributions of

this paper.

2. Model and estimation

2.1. The skew-normal distribution

Common statistical regression models often assume that the error terms follow normal or other symmetric distributions. However, in practice, data often exhibits skewness. If a symmetric distribution assumption is maintained in such cases, it may lead to inefficient estimates. To better reflect the characteristics of the data, it is more reasonable to assume a skewed distribution, such as the skew-normal distribution, for the error terms, which can improve the model's estimation performance. The following section provides an introduction to skew-normal distribution.

The SN distribution was proposed by Azzalini [11], which is an effective extension of the normal distribution. The skewness coefficient of the skew-normal distribution ranges from -0.995 to 0.995, and the maximum kurtosis can reach 3.869. By introducing the skewness parameter, this distribution can flexibly accommodate skewed data, making it suitable for a wider range of data modeling scenarios. In the parameter estimation below, we refer to the method of Ferreira et al. [15] and propose the EM-CALS algorithm for estimating unknown parameters and functions. We adopt the parameterized form of the skew-normal distribution proposed by Sahu et al. [20] to facilitate the direct use of analytical expressions in the E and M steps.

We will briefly introduce the Sahu-type skew-normal distribution and discuss some of its properties, which will be used in the EM-CALS algorithm.

The probability density function (pdf) and cumulative distribution function (cdf) of the Sahu-type skew-normal distribution are as follows:

$$f(y \mid \mu, \sigma^2, \delta) = \frac{2}{\sqrt{\sigma^2 + \delta^2}} \phi\left(\frac{y - \mu}{\sqrt{\sigma^2 + \delta^2}}\right) \Phi\left(\frac{\delta}{\sigma} \frac{y - \mu}{\sqrt{\sigma^2 + \delta^2}}\right), \tag{2.1}$$

$$F_Y(y; \mu, \sigma^2, \delta) = 2\Phi\left(\left(\frac{y - \mu}{\sqrt{\sigma^2 + \delta^2}}, 0\right); \mathbf{0}, \mathbf{\Omega}\right), \tag{2.2}$$

where μ is the location parameter, σ^2 is the scale parameter, and δ is the skewness parameter. ϕ and Φ are the pdf and cdf of the standard normal distribution $N(0, 1)$, respectively, $\mathbf{\Omega} = \begin{pmatrix} 1 & -\delta_1 \\ -\delta_1 & 1 \end{pmatrix}$, where $\delta_1 = \frac{\delta}{\sqrt{\sigma^2 + \delta^2}}$. If the random variable Y follows a skew-normal distribution with parameters μ, σ^2, δ, we denote it as $Y \sim SN(\mu, \sigma^2, \delta)$. The skew-normal distribution simplifies to the normal distribution when $\delta = 0$.

If $Y \sim SN(\mu, \sigma^2, \delta)$, the expectation and variance of Y are as follows:

$$\mathbb{E}[Y] = \mu + b\delta, \quad \mathrm{Var}(Y) = \sigma^2 + (1 - b^2)\delta^2, \tag{2.3}$$

where $b = \sqrt{\frac{2}{\pi}}$.

Additionally, the stochastic representation of Y is given by $Y \stackrel{d}{=} \mu + \delta|X_0| + \sigma X_1$, where X_0 and X_1 are independent random variables $N(0, 1)$.

2.2. The single-index partially functional linear model with AR(p) skew-normal errors

In regression analysis of autocorrelated data, autoregressive error structures are commonly used for modeling. Traditional studies often assume that the errors follow a normal distribution. However, when the data exhibits skewness, setting the error distribution to a skewed distribution, such as the skew-normal distribution, can effectively improve the efficiency of parameter estimation.

The single-index model, as a dimensionality reduction method, was combined with the functional linear model by Yu et al. [7] to propose the single-index partially functional linear model. In this model, the error structure is based on autoregressive errors under the skew-normal distribution when modeling autocorrelated data with skewness.

The single-index partially functional linear model with $AR(p)$ skew-normal errors, denoted as SIPFLM-SNAR(p), is defined as follows:

$$y_i = g(\mathbf{Z}_i^\top \boldsymbol{\alpha}) + \int_T \beta(t) X_i(t) dt + \varepsilon_i,$$

$$\varepsilon_i = e_i + \sum_{j=1}^p \psi_j \varepsilon_{i-j}, \tag{2.4}$$

$$e_i \sim SN(-b\delta, \sigma^2, \delta),$$

let y_i represent the observed response values for $i = 1, 2, \ldots, n$, and $\mathbf{Z_i}$ denote an l-dimensional vector of covariates. The parameter vector $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_l)^\top$ is an unknown vector satisfying the constraint $\|\boldsymbol{\alpha}\| = 1$. For identification purposes, we assume that the first component of α_1 is positive. The function $g(\cdot)$ is an unknown univariate link function, and $\{X(t) : t \in T\}$ represents a zero-mean random element in the Hilbert space $H = L^2(T)$, where H denotes the space of all square-integrable functions on T. The inner product in H is defined as $\langle x, y \rangle = \int_T x(t) y(t) dt$ for any $x, y \in H$, with the corresponding norm $\|x\| = \langle x, x \rangle^{1/2}$. The autoregressive parameters are denoted as $\psi_1, \ldots, \psi_p$, and the error terms e_i are independent random variables, each following a skew-normal distribution with zero mean and constant variance, as described in Eq (2.3). We assume that $\varepsilon_0 = \varepsilon_{-1} = \cdots = \varepsilon_{-(p-1)} = 0$. When $\psi_1 = \cdots = \psi_p = 0$ and $\delta = 0$, model (2.4) reduces to the single-index partially functional linear model as discussed by Yu et al. [7]. To define the function $g(\cdot)$, we let its domain be $[a_1, b_1]$, where a_1 and b_1 are the infimum and supremum of the set $\{\mathbf{Z}^\top \boldsymbol{\alpha}\}$, respectively. For simplicity, we assume that $T = [0, 1]$.

Following the approach of Ferreira et al. [15], for $i = 1, \ldots, n$, we assume that

$$Y_i \mid (y_{i-1}, \ldots, y_{i-p}) \sim SN\left(u_i + \sum_{j=1}^p \psi_j(y_{i-j} - u_{i-j}) - b\delta, \sigma^2, \delta\right),$$

where $u_i = g(\mathbf{Z}_i^\top \boldsymbol{\alpha}) + \int_0^1 \beta(t) X_i(t) dt$.

Given the flexibility and local control of the B-spline, we choose to represent the smooth functions $g(u)$ and $\beta(t)$ using B-splines (deBoor [21]). First, consider $g(u)$, which is expressed as

$$g(u) \approx \sum_{j=1}^{N_1} B_{1j,l_1}(u) \eta_j,$$

where $B_{1j,l_1}(u)$ denotes the normalized B-spline basis function of degree l_1, and η_j are the coefficients to be estimated.

$$B_{1j,0}(u) = \begin{cases} 1, & \kappa_j \le u < \kappa_{j+1}, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$B_{1j,l_1}(u) = w_{jl_1}B_{1j,l_1-1}(u) + (1 - w_{j+1,l_1})B_{1j+1,l_1-1}(u), \quad l_1 > 0,$$

where $w_{jl_1}(u) = \frac{u-\kappa_j}{\kappa_{j+l_1}-\kappa_j}$. To approximate the link function $g(\cdot)$, we partition the interval $[a_1, b_1]$ as $a_1 = \kappa_0 < \kappa_1 < \cdots < \kappa_{k_1+1} = b_1$, and use κ_i as the knots. We use $N_1 = k_1 + l_1 + 1$ as the normalized B-spline basis functions of degree l_1 to approximate $g(\cdot)$, forming a linear spline space. Following the idea of Yu et al. [7], the basis functions are defined using a cubic B-spline with evenly distributed knots, where $l_1 = 3$. In this case, we have the approximation: $g(u) \approx \sum_{j=1}^{N_1} B_{1j,3}(u)\eta_j$. To simplify, we let $B_{1j} = B_{1j,3}(u)$, and, thus, the approximation becomes: $g(u) \approx \sum_{j=1}^{N_1} B_{1j}(u)\eta_j$.

We place these basis functions into the vector $\boldsymbol{B}_1(u) = (B_{11}(u), \ldots, B_{1N_1}(u))^\top$. Then, we approximate $g(\cdot)$ on the interval $[a_1, b_1]$ using $\boldsymbol{B}_1(\cdot)$. Similarly, we approximate slope function $\beta(t)$ using the same method. Let $\boldsymbol{B}_2(t) = (B_{21}(t), \ldots, B_{2N_2}(t))^\top$ be the vector of normalized B-spline basis functions of degree l_2 on the interval $[0, 1]$, containing k_2 internal knots, where $N_2 = k_2 + l_2 + 1$. Thus, we have $\beta(t) \approx \sum_{j=1}^{N_2} B_{2j,l_2}(u)\gamma_j$. This can be rewritten in matrix form as

$$g(u) \approx \boldsymbol{B}_1^\top(u)\boldsymbol{\eta}, \quad \beta(t) \approx \boldsymbol{B}_2^\top(t)\boldsymbol{\gamma},$$

where $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_{N_1})^\top$ and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_{N_2})^\top$.

To estimate the slope function $\beta(\cdot)$ and the link function $g(\cdot)$, we employ B-spline approximation. First, the degrees of spline functions must be determined. Inspired by the idea of Huang et al. [22], cubic splines with equally spaced knots are selected. The choice of the number of knots and basis functions is also crucial. Too many knots and basis functions may increase the complexity of the model, and although the fitting error may decrease, it could lead to overfitting by capturing noise, resulting in unreliable parameter estimates. On the other hand, too few knots and basis functions may fail to capture the complexity of the data, leading to underfitting. We need to select the numbers of basis functions, N_1 and N_2. For this purpose, the Akaike information criterion (AIC) and Bayesian information criterion (BIC) are commonly used to select the truncation parameters. In this study, the optimal values of N_1 and N_2 are determined by minimizing the BIC criterion, which is defined as follows:

$$BIC(N_1, N_2) = -2L(\hat{\boldsymbol{\theta}}, N_1, N_2) + (N_1 + N_2 + l + p + 1)\ln(n),$$

where $L(\hat{\boldsymbol{\theta}}, N_1, N_2)$ denotes the log-likelihood function evaluated at $\hat{\boldsymbol{\theta}}$ for fixed N_1 and N_2.

We define $\boldsymbol{W} = \langle X(t), \boldsymbol{B}_2(t) \rangle = \left(\int_0^1 X(t)B_{21}(t)\,dt, \ldots, \int_0^1 X(t)B_{2N_2}(t)\,dt \right)^\top$, and $\boldsymbol{W}_i = \langle X_i(t), \boldsymbol{B}_2(t) \rangle$, $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_p)^\top$. The model (2.4) simplifies to

$$y_i \approx \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top\boldsymbol{\alpha})\boldsymbol{\eta} + \boldsymbol{W}_i^\top\boldsymbol{\gamma} + \varepsilon_i,$$

$$\varepsilon_i = e_i + \sum_{j=1}^{p} \psi_j \varepsilon_{i-j}, \tag{2.5}$$

$$e_i \sim SN(-b\delta, \sigma^2, \delta), i = 1, \ldots, n.$$

The unknown parameters in this model are $\tilde{\boldsymbol{\theta}} = (\boldsymbol{\alpha}^\top, \boldsymbol{\eta}^\top, \boldsymbol{\gamma}^\top, \boldsymbol{\psi}^\top, \delta, \sigma^2)^\top$, where the l-dimensional single-index vector $\boldsymbol{\alpha}$ must satisfy the constraints $\|\boldsymbol{\alpha}\| = 1$ and $\alpha_1 > 0$. Inspired by Yu and Ruppert [26], we also handle the constraint on the single-index parameter through reparameterization.

Inspired by the work of Yu et al. [7], let $\boldsymbol{\phi}$ represent a parameter vector of dimension $l - 1$, and define $\boldsymbol{\alpha} = \boldsymbol{\alpha}(\boldsymbol{\phi}) = \left(\sqrt{1 - \|\boldsymbol{\phi}\|^2}, \boldsymbol{\phi}^\top \right)^\top$. Given that the true parameter vector $\boldsymbol{\phi}_0$ must satisfy the constraint $\|\boldsymbol{\phi}_0\| \leq 1$, we assume the strict inequality $\|\boldsymbol{\phi}_0\| < 1$. As a result, the function $\boldsymbol{\alpha}(\boldsymbol{\phi})$ is infinitely differentiable with respect to $\boldsymbol{\phi}$. The Jacobian matrix of $\boldsymbol{\alpha}(\boldsymbol{\phi})$ with respect to $\boldsymbol{\phi}$ is expressed as

$$
\boldsymbol{J}_\phi \equiv \begin{pmatrix} -\left(1 - \|\boldsymbol{\phi}\|^2\right)^{-1/2} \boldsymbol{\phi}^\top \\ \boldsymbol{I}_{l-1} \end{pmatrix},
$$

where $\boldsymbol{I}_l$ is the $l \times l$ identity matrix. After reparameterization, the unknown parameters are $\boldsymbol{\theta} = (\boldsymbol{\phi}^\top, \boldsymbol{\eta}^\top, \boldsymbol{\gamma}^\top, \boldsymbol{\psi}^\top, \delta, \sigma^2)^\top$.

3. Parameter estimation via the EM-CALS algorithm

According to Eq (2.5), for the parameter $\boldsymbol{\theta} = (\boldsymbol{\phi}^\top, \boldsymbol{\eta}^\top, \boldsymbol{\gamma}^\top, \boldsymbol{\psi}^\top, \delta, \sigma^2)^\top \in \mathbb{R}^{p^*}$, where $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_p)$ and $p^* = N_1 + N_2 + l + p + 1$, the log-likelihood function of the observed data can be expressed as

$$
\ell(\boldsymbol{\theta}) = n \log\left(\frac{2}{\sqrt{2\pi}}\right) - \frac{n}{2} \log(\sigma^2 + \delta^2) - \frac{1}{2(\sigma^2 + \delta^2)} \sum_{i=1}^n (y_i - \xi_i + b\delta)^2 + \sum_{i=1}^n \log\{\Phi(B_i)\}, \qquad (3.1)
$$

where $B_i = \frac{\delta}{\sigma(\sigma^2+\delta^2)^{1/2}}(y_i - \xi_i + b\delta)$, $\xi_i = \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top\boldsymbol{\alpha})\boldsymbol{\eta} + \boldsymbol{W}_i^\top\boldsymbol{\gamma} + \sum_{j=1}^p \psi_j(y_{i-j} - \boldsymbol{B}_1^\top(\boldsymbol{Z}_{i-j}^\top\boldsymbol{\alpha})\boldsymbol{\eta} - \boldsymbol{W}_{i-j}^\top\boldsymbol{\gamma})$, $\boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top\boldsymbol{\alpha})$ and $\boldsymbol{W}_i^\top$ represent the i-th row of $\boldsymbol{B}_1(\boldsymbol{Z}^\top\boldsymbol{\alpha})$ and $\boldsymbol{W}$, respectively.

To estimate the parameter $\boldsymbol{\theta}$, we need to maximize the log-likelihood function. However, directly maximizing this objective function can be challenging, so numerical methods are necessary. Given the stochastic representation of the skew-normal distribution, the EM algorithm is particularly useful for addressing this problem.

3.1. EM-CALS algorithm

The EM algorithm, introduced by Dempster et al. [23], is widely utilized for solving maximum likelihood estimation problems. One of the key advantages of the EM algorithm is its ability to efficiently handle maximum likelihood estimation issues that involve missing data or latent variables through an iterative approach. This capability makes it particularly effective for managing complex models, especially when the data is incomplete or the underlying structure is not well-defined.

Let $TN(\mu, \sigma^2; 0, +\infty)$ represent the truncated normal distribution with parameters μ and σ^2, which is supported on the interval $(0, +\infty)$ (Johnson et al. [24]). By utilizing the stochastic representation of the Sahu-type skew-normal distribution, we can derive:

$$
\begin{aligned}
Y_i \mid Z_i = z_i, y_{i-1}, \ldots, y_{i-p} &\stackrel{\text{ind}}{\sim} N(\xi_i - b\delta + \delta z_i, \sigma^2), \\
Z_i &\stackrel{\text{iid}}{\sim} TN(0, 1; (0, +\infty)), \quad i = 1, \ldots, n.
\end{aligned} \qquad (3.2)
$$

The EM algorithm treats latent variables $z = (z_1, \ldots, z_n)^\top$ as unobserved data and $y = (y_1, \ldots, y_n)^\top$ as observed data. Using Eq (3.2), we obtain the joint distribution of (Y_i, Z_i) as

$$
\begin{aligned}
f(y_i, z_i; y_{i-1}, \ldots, y_{i-p}, \boldsymbol{\theta}) &= 2\phi(y_i \mid \xi_i + \delta z_i - b\delta, \sigma^2)\phi(z_i)\mathbb{I}_{(0,+\infty)}(z_i) \\
&= 2\phi(y_i \mid \xi_i - b\delta, \sigma^2 + \delta^2)\phi(z_i \mid \mu_{iz}, \sigma_z^2)\mathbb{I}_{(0,+\infty)}(z_i) \\
&= f(y_i \mid y_{i-1}, \ldots, y_{i-p})f(z_i \mid y_i, y_{i-1}, \ldots, y_{i-p}),
\end{aligned}
\tag{3.3}
$$

where $\mu_{iz} = \frac{\delta}{\sigma^2+\delta^2}(y_i - \xi_i + b\delta), \sigma_z^2 = \frac{\sigma^2}{\sigma^2+\delta^2}$; thus, using Eq (3.3), we obtain $Z_i \mid y_i, \boldsymbol{\theta} \sim TN(\mu_{iz}, \sigma_z^2; (0, +\infty))$.

Using the properties of the truncated normal distribution, we can obtain the following conditional expectation:

$$
\begin{aligned}
E[Z_i \mid y_i] &= \mu_{iz} + \sigma_z W_\Phi\left(\frac{\mu_{iz}}{\sigma_z}\right), \\
E[Z_i^2 \mid y_i] &= \mu_{iz}^2 + \sigma_z^2 + \sigma_z \mu_{iz} W_\Phi\left(\frac{\mu_{iz}}{\sigma_z}\right),
\end{aligned}
\tag{3.4}
$$

where $W_\Phi(u) = \frac{\phi(u)}{\Phi(u)}$.

Using Eq (3.3), the joint distribution of $y_c = (y, z)$, we obtain the logarithmic likelihood function for the complete data:

$$
\ell_c(\boldsymbol{\theta}|y_c) = C - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[(y_i - \xi_i)^2 - 2\delta(y_i - \xi_i)(z_i - b) + \delta^2(b^2 - 2bz_i + z_i^2)\right],
$$

where C is a constant independent of the unknown parameters.

In the EM algorithm, the Q-function is the expectation of the complete logarithmic likelihood function of the data, given the observed data y and the current parameter estimates $\hat{\boldsymbol{\theta}}^{(k)}$. Specifically, it is defined as

$$
Q(\boldsymbol{\theta} \mid \hat{\boldsymbol{\theta}}^{(k)}) = \mathbb{E}[\ell_c(\boldsymbol{\theta} \mid y_c) \mid y, \hat{\boldsymbol{\theta}}^{(k)}] \propto -\frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}[(y_i-\xi_i)^2 - 2\delta(y_i-\xi_i)(\hat{z}_i^{(k)}-b) + \delta^2(b^2 - 2b\hat{z}_i^{(k)} + \hat{z^2}_i^{(k)})],
$$

where $\hat{z}_i^{(k)} = \mathbb{E}[Z_i \mid y_i, \hat{\boldsymbol{\theta}}^{(k)}]$ and $\hat{z^2}_i^{(k)} = \mathbb{E}[Z_i^2 \mid y_i, \hat{\boldsymbol{\theta}}^{(k)}]$.

Based on Eq (3.4), the expectation of the latent variable z given the observed data y and the current parameter estimates $\hat{\boldsymbol{\theta}}^{(k)}$ can be computed as follows:

$$
\hat{z}_i^{(k)} = \hat{\mu}_{iz}^{(k)} + \hat{\sigma}_z^{(k)} W_\Phi\left(\frac{\hat{\mu}_{iz}^{(k)}}{\hat{\sigma}_z^{(k)}}\right),
\tag{3.5}
$$

$$
\hat{z^2}_i^{(k)} = \left[\hat{\mu}_{iz}^{(k)}\right]^2 + \hat{\sigma^2}_z^{(k)} + \hat{\sigma}_z^{(k)}\hat{\mu}_{iz}^{(k)} W_\Phi\left(\frac{\hat{\mu}_{iz}^{(k)}}{\hat{\sigma}_z^{(k)}}\right),
\tag{3.6}
$$

for any $i = 1, \ldots, n$, where $\hat{\mu}_{iz}^{(k)} = \frac{\hat{\delta}^{(k)}}{\hat{\sigma}^{2(k)}+\hat{\delta}^{(k)2}}\left(y_i - \hat{\xi}_i^{(k)} + b\hat{\delta}^{(k)}\right)$, $\hat{\sigma^2}_z^{(k)} = \frac{\hat{\sigma}^{2(k)}}{\hat{\sigma}^{2(k)}+\hat{\delta}^{(k)2}}$, $\hat{\xi}_i^{(k)} = \hat{u}_i^{(k)} + \sum_{j=1}^{p}\hat{\psi}_j^{(k)}\left(y_{i-j} - \hat{u}_{i-j}^{(k)}\right)$, $\hat{u}_i^{(k)} = \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top\boldsymbol{\alpha}^{(k)})\boldsymbol{\eta}^{(k)} + \boldsymbol{W}_i^\top\boldsymbol{\gamma}^{(k)}$.

Let $\boldsymbol{A} = \boldsymbol{A}(\boldsymbol{\psi})$ be an $n \times n$ matrix, given by

$$
\boldsymbol{A} = \begin{pmatrix}
1 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\
-\psi_1 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\
-\psi_2 & -\psi_1 & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
-\psi_p & -\psi_{p-1} & -\psi_{p-2} & \cdots & -\psi_1 & 1 & \cdots & 0 & 0 & 0 \\
0 & -\psi_p & -\psi_{p-1} & \cdots & -\psi_2 & -\psi_1 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & -\psi_2 & -\psi_1 & 1
\end{pmatrix}.
$$

Let $\overline{\boldsymbol{\theta}} = (\boldsymbol{\eta}^\top, \boldsymbol{\gamma}^\top, \boldsymbol{\psi}^\top, \delta, \sigma^2)^\top$, where the M-step updates the parameter $\overline{\boldsymbol{\theta}}$ by maximizing $Q(\boldsymbol{\theta} \mid \hat{\boldsymbol{\theta}}^{(k)})$, yielding a new estimate $\overline{\boldsymbol{\theta}}^{(k+1)}$, and the ith row of $\boldsymbol{A}$ is $\boldsymbol{A}_i^\top$.

The EM algorithm is preferred for its simplicity and stability. However, when parameters are subject to constraints, maximizing the Q-function can become extremely difficult, complicating the M-step. Keiji Takai [25] proposed a constrained EM algorithm that utilizes a projection method, but challenges still arise when addressing nonlinear constraints in single-index models. In particular, the unit norm constraint $\|\boldsymbol{\alpha}\| = 1$ requires solving nonlinear equations, calculating gradients, and inverting matrices, which becomes increasingly complex in high-dimensional parameter spaces and can lead to numerical instability, thereby complicating optimization. Furthermore, selecting the appropriate step size in the P-step of the constrained EM algorithm is challenging, especially under high-dimensional nonlinear constraints, where finding a suitable step size is often difficult. This frequently results in slow convergence and reduced efficiency.

For the constraint $\|\boldsymbol{\alpha}\| = 1$ and $\alpha_1 > 0$, Yu and Ruppert [26] addressed this issue using a reparameterization approach. Additionally, Yu et al. [7] applied the least squares method to estimate the single-index vector in the single-index partially functional linear model. Inspired by these works, we also use the least squares method to compute the single-index vector. Furthermore, since the least squares method converges faster than the EM algorithm and is simpler for handling nonlinear constraints, we choose to use the least squares method as a substitute for maximum likelihood estimation (MLE). Furthermore, given that the error terms exhibit an autoregressive structure and do not meet the assumption of normality in the error distribution, we employ the ALS to estimate the single-index vector $\boldsymbol{\alpha}$. The ALS method effectively addresses autocorrelation among the error terms, providing robust parameter estimates by accounting for dependencies among errors. Additionally, ALS can adapt to skewed distributions and nonzero expectations of errors, leading to more accurate and stable parameter estimates. Within the framework of the EM algorithm for this model, the E-step and M-step are utilized to estimate other complex parameters. In contrast, the single-index vector $\boldsymbol{\alpha}$ is estimated by introducing a CALS step. In this CALS step, we estimate $\boldsymbol{\phi}$ using ALS while holding the current estimates of the other parameters fixed, and then we obtain the estimate of the single-index vector $\boldsymbol{\alpha}(\boldsymbol{\phi})$ through a transformation.

Similar to Ferreira et al. [15], for $i = 1, \ldots, n$, we assume that

$$
Y_i \mid (y_{i-1}, \ldots, y_{i-p}) \sim \mathrm{SN}\left(u_i + \sum_{j=1}^{p} \psi_j(y_{i-j} - u_{i-j}) - b\delta, \sigma^2, \delta\right), \tag{3.7}
$$

where $u_i = \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \boldsymbol{\alpha}(\boldsymbol{\phi}))\boldsymbol{\eta} + \boldsymbol{W}_i^\top \boldsymbol{\gamma}$.

Given the current estimates of $\bar{\boldsymbol{\theta}}$, namely, $\boldsymbol{\eta} = \boldsymbol{\eta}^{(t)}$, $\boldsymbol{\gamma} = \boldsymbol{\gamma}^{(t)}$, $\boldsymbol{\psi} = \boldsymbol{\psi}^{(t)}$, $\delta = \delta^{(t)}$, and $\sigma^2 = \sigma^{2(t)}$, and the observed data Y, the ALS method can be employed to estimate the single-index vector $\boldsymbol{\alpha}$. Using the conditional expectation formula (2.3) for the truncated normal distribution, we can compute its conditional expectation as follows:

$$\hat{Y}_i = E\left(Y_i \mid y_{i-1}, \dots, y_{i-p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \delta, \sigma^2\right) = u_i + \sum_{j=1}^{p} \psi_j(y_{i-j} - u_{i-j}).$$

In ALS, the parameter $\boldsymbol{\phi}$ is obtained by minimizing the sum of squared residuals

$$\hat{\boldsymbol{\phi}}_{ALS} = \arg\min_{\boldsymbol{\phi}} \sum_{i=1}^{n} U_i^2,$$

where the residual U_i is defined as: $U_i = Y_i - E\left(Y_i \mid y_{i-1}, \dots, y_{i-p}, \boldsymbol{\eta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \delta, \sigma^2\right)$. Finally, the estimate of the single-index vector $\boldsymbol{\alpha}$ is obtained by transforming $\hat{\boldsymbol{\phi}}_{ALS}$ into $\boldsymbol{\alpha}(\hat{\boldsymbol{\phi}}_{ALS})$.

3.2. Algorithm implementation

Step 0. Start from $\hat{\boldsymbol{\theta}}^{(0)} = \left(\hat{\boldsymbol{\alpha}}^{(0)\top}, \hat{\boldsymbol{\eta}}^{(0)\top}, \hat{\boldsymbol{\gamma}}^{(0)\top}, \hat{\boldsymbol{\psi}}^{(0)\top}, \hat{\delta}^{(0)}, \hat{\sigma}^{2(0)}\right)^\top$, for example, by minimizing $\sum_{i=1}^{n}(Y_i - \boldsymbol{Z}_i^\top \boldsymbol{\alpha} - \boldsymbol{W}_i^\top \boldsymbol{\gamma})^2$ and normalizing $\hat{\boldsymbol{\alpha}}^{(0)}$ such that $\|\hat{\boldsymbol{\alpha}}^{(0)}\| = 1$ and $\hat{\alpha}_1^{(0)} > 0$, to obtain the estimate of $\hat{\boldsymbol{\alpha}}^{(0)}$, where $\hat{\alpha}_1^{(0)}$ is the first component of $\hat{\boldsymbol{\alpha}}^{(0)}$. Given the initial single-index vector $\hat{\boldsymbol{\alpha}}^{(0)}$, we compute $\{u_i = \boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}}^{(0)}, i = 1, \dots, n\}$. Then, minimize the error sum of squares function:

$$\sum_{i=1}^{n} \left(Y_i - \boldsymbol{B}_1^\top(u_i)\boldsymbol{\eta} - \boldsymbol{W}_i^\top \boldsymbol{\gamma}\right)^2,$$

to optimize $(\boldsymbol{\eta}^\top, \boldsymbol{\gamma}^\top)^\top$, yielding $\hat{\boldsymbol{\eta}}^{(0)}$ and $\hat{\boldsymbol{\gamma}}^{(0)}$. In $\boldsymbol{B}_1(u)$, the basis functions use k_1 equidistant points as nodes within the domain of $g(\cdot)$. Once $\hat{\boldsymbol{\eta}}^{(0)}$ and $\hat{\boldsymbol{\gamma}}^{(0)}$ are obtained, fit $\boldsymbol{r} = \boldsymbol{Y} - \boldsymbol{B}_1(u)\hat{\boldsymbol{\eta}}^{(0)} - \boldsymbol{W}\hat{\boldsymbol{\gamma}}^{(0)}$ using the R function `selm`, extract the shape parameter α and scale parameter ω from the skew-normal distribution, and use the relationships $\sigma^2 = \frac{\omega^2}{1+\alpha^2}$ and $\delta = \frac{\alpha^2 \omega^2}{1+\alpha^2}$ to obtain $\hat{\delta}^{(0)}$ and $\hat{\sigma}^{2(0)}$. Assume $\hat{\boldsymbol{\psi}}^{(0)\top} = \boldsymbol{0}$. After completing the above steps, the initial estimate is $\hat{\boldsymbol{\theta}}^{(0)} = \left(\hat{\boldsymbol{\alpha}}^{(0)\top}, \hat{\boldsymbol{\eta}}^{(0)\top}, \hat{\boldsymbol{\gamma}}^{(0)\top}, \hat{\boldsymbol{\psi}}^{(0)\top}, \hat{\delta}^{(0)}, \hat{\sigma}^{2(0)}\right)^\top$.

Step 1. In the E-step, calculate $z_i^{(k)}$ and $z_i^{2(k)}$ for $i = 1, \dots, n$ using formulas (3.5) and (3.6).

Step 2. In the M-step, update $\left(\hat{\boldsymbol{\gamma}}^{(k)\top}, \hat{\boldsymbol{\eta}}^{(k)\top}, \hat{\delta}^{(k)}, \hat{\boldsymbol{\psi}}^{(k)\top}, \hat{\sigma}^{2(k)}\right)^\top$. When $k > 1$: $\hat{\boldsymbol{\alpha}}^{(k)} = \boldsymbol{\alpha}(\hat{\boldsymbol{\phi}}^{(k)})$, and when $k = 0$: $\hat{\boldsymbol{\alpha}}^{(0)} = \hat{\boldsymbol{\alpha}}^{(0)}$. The iteration formulas are as follows:

$$\hat{\boldsymbol{\gamma}}^{(k+1)} = \left[(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{W})^\top(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{W})\right]^{-1} \left[(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{W})^\top\left(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{Y} - \hat{\boldsymbol{A}}^{(k)}\boldsymbol{B}_1(\boldsymbol{Z}^\top\hat{\boldsymbol{\alpha}}^{(k)})\hat{\boldsymbol{\eta}}^{(k)} - \hat{\delta}^{(k)}(\hat{\boldsymbol{z}}^{(k)} - b)\right)\right],$$

$$\hat{\boldsymbol{\eta}}^{(k+1)} = \left[(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{B}_1(\boldsymbol{Z}^\top\hat{\boldsymbol{\alpha}}^{(k)}))^\top(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{B}_1(\boldsymbol{Z}^\top\hat{\boldsymbol{\alpha}}^{(k)}))\right]^{-1} \left[(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{B}_1(\boldsymbol{Z}^\top\hat{\boldsymbol{\alpha}}^{(k)}))^\top\left(\hat{\boldsymbol{A}}^{(k)}\boldsymbol{Y} - \hat{\boldsymbol{A}}^{(k)}\boldsymbol{W}\hat{\boldsymbol{\gamma}}^{(k)} - \hat{\delta}^{(k)}(\hat{\boldsymbol{z}}^{(k)} - b)\right)\right],$$

$$\hat{\delta}^{(k+1)} = \frac{\sum_{i=1}^{n}\left(y_i - \hat{\xi}_i^{(k)}\right)\left(\hat{z}_i^{(k)} - b\right)}{\sum_{i=1}^{n}\left(b^2 - 2b\hat{z}_i^{(k)} + \hat{z}_i^{2(k)}\right)},$$

$$\hat{\psi}_j^{(k+1)} = \frac{1}{\sum_{i=1}^n r_{i-j}^2} \sum_{i=1}^n \left[r_i - \sum_{l=1,l\neq j}^p \hat{\psi}_l^{(k)} r_{i-l} - \hat{\delta}^{(k)}(\hat{z}_i^{(k)} - b) \right] r_{i-j}, \quad j = 1, \ldots, p,$$

$$\hat{\sigma}^{2^{(k+1)}} = \frac{1}{n} \sum_{i=1}^n \left[(y_i - \hat{\xi}_i^{(k)})^2 - 2\hat{\delta}^{(k)}(y_i - \hat{\xi}_i^{(k)})(\hat{z}_i^{(k)} - b) + \hat{\delta}^{(k)2}\left(b^2 - 2b\hat{z}_i^{(k)} + \hat{z}_i^{2^{(k)}} \right) \right],$$

where $r_i = y_i - u_i^{(k)} = y_i - \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}}^{(k)})\hat{\boldsymbol{\eta}}^{(k)} - \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}}^{(k)}$, $i = 1, \ldots, n$, and $r_0 = r_{-1} = \cdots = r_{-(p-1)} = 0$, $\hat{\boldsymbol{z}}^{(k)} = (\hat{z}_1^{(k)}, \hat{z}_2^{(k)}, \ldots, \hat{z}_n^{(k)})^\top$, for $k = 0, 1, 2, \ldots$. Finally, the matrix $\hat{\boldsymbol{A}}^{(k)}$ is generated by the following formula: $\hat{\boldsymbol{A}}^{(k)} = \boldsymbol{A}(\hat{\boldsymbol{\psi}}^{(k)})$.

The basis function $\boldsymbol{B}_1(u)$ uses k_1 equidistant points within the interval $[a_1, b_1]$, but in practice, the interval $[a_1, b_1]$ is unknown. We generate the B-spline basis function for each given $\hat{\boldsymbol{\alpha}}^{(k)}$ using the minimum and maximum values of $\boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}}^{(k)}$ as boundary points. Beyond this interval, $g(\cdot)$ may be defined in any reasonable manner without altering the results.

Step 3. In the CALS step, apply the ALS method to estimate the parameter vector $\boldsymbol{\phi}$. Fix $\left(\hat{\boldsymbol{\eta}}^{(k)\top}, \hat{\boldsymbol{\gamma}}^{(k)\top}, \hat{\boldsymbol{\psi}}^{(k)\top}, \hat{\delta}^{(k)}, \hat{\sigma}^{2(k)} \right)^\top$ and minimize $\sum_{i=1}^n (Y_i - u_i^{(k)} - \sum_{j=1}^p \psi_j^{(k)}(y_{i-j} - u_{i-j}^{(k)}))^2$, where $u_i^{(k)} = \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \boldsymbol{\alpha}(\hat{\boldsymbol{\phi}}^{(k)}))\hat{\boldsymbol{\eta}}^{(k)} + \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}}^{(k)}$, to obtain $\hat{\boldsymbol{\phi}}^{(k+1)}$. This step can be optimized using the `optim` function in R, and then the transformed $\hat{\boldsymbol{\alpha}}^{(k+1)}$ is obtained.

Step 4. Repeat Steps 1 to 3 until the convergence criterion is satisfied, where $\|\overline{\boldsymbol{\theta}}^{(k+1)} - \overline{\boldsymbol{\theta}}^{(k)}\|$ is less than 10^{-5}, and represent the final estimates of $\boldsymbol{\alpha}, \boldsymbol{\eta}, \boldsymbol{\gamma}, \boldsymbol{\psi}, \delta, \sigma^2$ as $\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\psi}}, \hat{\delta}, \hat{\sigma}^2$, respectively.

The following is the flowchart of the EM-CALS algorithm (see Figure 1):

Initial value

The initial estimate $\hat{\theta}^{(0)}$ is obtained by minimizing the error function $\sum_{i=1}^n (Y_i - Z_i^T \hat{\alpha} - W_i^T \hat{\gamma})^2$, normalizing $\hat{\alpha}^{(0)}$ such that $\|\hat{\alpha}^{(0)}\| = 1$ and $\hat{\alpha}_1^{(0)} > 0$. Then, compute $u_i = Z_i^T \hat{\alpha}^{(0)}$ for each i, and minimize the sum of squares $\sum_{i=1}^n (Y_i - B_1(u_i)^T \hat{\eta} - W_i^T \hat{\gamma})^2$ to obtain $\hat{\eta}^{(0)}$ and $\hat{\gamma}^{(0)}$. Using the R function 'selm', extract the shape parameter α and scale parameter ω from the skew-normal distribution, and compute $\hat{\sigma}^2$ and $\hat{\delta}$ using $\hat{\sigma}^2 = \frac{\omega^2}{1+\alpha^2}$ and $\hat{\delta} = \frac{\alpha^2 \omega^2}{1+\alpha^2}$. Finally, assume $\hat{\psi}^{(0)T} = 0$.

Step E

The expectation of the latent variable $\hat{z}_i^{(k)}$ given the observed data y_i and the current parameter estimates $\hat{\theta}^{(k)}$ can be computed as:

$$\hat{z}_i^{(k)} = \hat{\mu}_{iz}^{(k)} + \hat{\sigma}_z^{(k)} W_\Phi \left(\frac{\hat{\mu}_{iz}^{(k)}}{\hat{\sigma}_z^{(k)}} \right),$$

$$\hat{z}_i^{2(k)} = \left[\hat{\mu}_{iz}^{(k)} \right]^2 + \hat{\sigma}_z^{2(k)} + \hat{\sigma}_z^{(k)} \hat{\mu}_{iz}^{(k)} W_\Phi \left(\frac{\hat{\mu}_{iz}^{(k)}}{\hat{\sigma}_z^{(k)}} \right),$$

In the E-step, compute $\hat{z}_i^{(k)}$ and $\hat{z}_i^{2(k)}$ for $i = 1, \ldots, n$ using the above formulas.

Step M

According to the formula in Step 2 of Section 3.2, update the following parameters:

$$\hat{\gamma}^{(k)}, \hat{\eta}^{(k)}, \hat{\delta}^{(k)}, \hat{\psi}^{(k)}, \hat{\sigma}^2.$$

Step C-ALS

Step 3. In the C-ALS step, apply the ALS method to estimate the parameter vector ϕ. Fix the following parameters:

$$\hat{\eta}^{(k)}, \hat{\gamma}^{(k)}, \hat{\psi}^{(k)}, \hat{\delta}^{(k)}, \hat{\sigma}^2.$$

Minimize the following expression:

$$\sum_{i=1}^n \left(Y_i - u_i^{(k)} - \sum_{j=1}^p \psi_j^{(k)} \left(y_{i-j} - u_{i-j}^{(k)} \right) \right)^2$$

where

$$u_i^{(k)} = B_1^T (Z_i^T \alpha(\hat{\phi}^{(k)})) \hat{\eta}^{(k)} + W_i^T \hat{\gamma}^{(k)}.$$

The optimization can be performed using the optim function in R to obtain the updated parameter vector $\hat{\phi}^{(k+1)}$. After optimization, the transformed $\hat{\alpha}^{(k+1)}$ is obtained.

Condition checking

For $k = 0, 1, 2, \ldots$, the iterations are repeated until a suitable convergence rule is satisfied, for example, when the Euclidean norm of $\theta^{(k+1)} - \theta^{(k)}$ is less than 10^{-5}.

Stop

Figure 1. The flowchart of the EM-CALS algorithm.

3.3. Comparison of algorithms

Inspired by the work of Yang et al. [27], we use the TSILS to estimate the unknown parameters in the single-index partially functional linear regression model with p-order autoregressive skew-normal errors. This section compares the proposed EM-CALS algorithm with the TSILS.

The two-step iterative least squares estimation is as follows:

Step 0. Start with the initial estimate $\hat{\overset{\frown}{\theta}}^{(0)} = \left(\hat{\alpha}^{(0)\top}, \hat{\eta}^{(0)\top}, \hat{\gamma}^{(0)\top}, \hat{\psi}^{(0)\top}, \hat{\delta}^{(0)}, \hat{\sigma}^{2(0)} \right)^{\top}$, for example, by minimizing the objective function: $\sum_{i=1}^{n} \left(Y_i - Z_i^{\top}\alpha - W_i^{\top}\gamma \right)^2$, and normalizing $\hat{\alpha}^{(0)}$ such that $\|\hat{\alpha}^{(0)}\| = 1$ and $\hat{\alpha}_1^{(0)} > 0$, where $\hat{\alpha}_1^{(0)}$ is the first component of $\hat{\alpha}^{(0)}$, to obtain the estimate of $\hat{\alpha}^{(0)}$. Given the initial single-index vector $\hat{\alpha}^{(0)}$, we compute $\{u_i = Z_i^{\top}\hat{\alpha}^{(0)}, i = 1, \ldots, n\}$. Then, obtain $\widehat{U}_i = \left(\widehat{W}_i^{\top}, B_i^{\top} \right)^{\top}$, and compute $\hat{\theta}^{(0)} = \left(\hat{\gamma}^{(0)\top}, \hat{\eta}^{(0)\top} \right)^{\top}$ based on the least squares method: $\theta^{(0)} = \left(\widehat{U}^{\top}\widehat{U} \right)^{-1} \widehat{U}^{\top} Y$, where $Y = (Y_1, Y_2, \ldots, Y_n)^{\top}$ and $\widehat{U}^{\top} = \left(\widehat{U}_1^{\top}, \widehat{U}_2^{\top}, \ldots, \widehat{U}_n^{\top} \right)$. Once $\hat{\eta}^{(0)}$ and $\hat{\gamma}^{(0)}$ are obtained, fit $r = Y - B_1(u)\hat{\eta}^{(0)} - W\hat{\gamma}^{(0)}$ using the R function `selm`. Extract the shape parameter α and scale parameter ω from the skew-normal distribution, and use the relationships: $\sigma^2 = \frac{\omega^2}{1+\alpha^2}, \quad \delta = \frac{\alpha^2 \omega^2}{1+\alpha^2}$ to obtain $\hat{\delta}^{(0)}$ and $\hat{\sigma}^{2(0)}$. Assume $\hat{\psi}^{(0)\top} = 0$. After completing the above steps, the initial estimate is:

$$\hat{\overset{\frown}{\theta}}^{(0)} = \left(\hat{\alpha}^{(0)\top}, \hat{\eta}^{(0)\top}, \hat{\gamma}^{(0)\top}, \hat{\psi}^{(0)\top}, \hat{\delta}^{(0)}, \hat{\sigma}^{2(0)} \right)^{\top}.$$

Step 1. Update $\tilde{V}^{(k)}, \tilde{H}^{(k)}$:

$$\tilde{V}^{(k)} = \begin{pmatrix} \hat{\varepsilon}_{p+1}^{(k)} \\ \vdots \\ \hat{\varepsilon}_n^{(k)} \end{pmatrix}, \quad \tilde{H}^{(k)} = \begin{pmatrix} \hat{\varepsilon}_p^{(k)} & \cdots & \hat{\varepsilon}_1^{(k)} \\ \vdots & & \vdots \\ \hat{\varepsilon}_{n-1}^{(k)} & \cdots & \hat{\varepsilon}_{n-p}^{(k)} \end{pmatrix},$$

where $\hat{\varepsilon}_t^{(k)} = Y_t - \widehat{U}_t^{\top}\hat{\theta}^{(k-1)}$, $t = 1, 2, \cdots, n$.

Step 2. Compute $\hat{\psi}^{(k)}$:

$$\hat{\psi}^{(k)} = (\tilde{H}^{(k)\top}\tilde{H}^{(k)})^{-1}\tilde{H}^{(k)\top}\tilde{V}^{(k)}.$$

Step 3. Update $V_t^{(k)}, H_t^{(k)}$:

$$V_t^{(k)} = Y_t - \sum_{l=1}^{p} \psi_l^{(k)} Y_{t-l}, \quad H_t^{(k)} = \widehat{U}_t - \sum_{l=1}^{p} \psi_l^{(k)} \widehat{U}_{t-l}.$$

Step 4. Compute $\theta^{(k)}$:

$$\hat{\theta}^{(k)} = (H^{(k)\top}H^{(k)})^{-1}H^{(k)\top}V^{(k)},$$

where

$$V^{(k)} = \begin{pmatrix} V_{p+1}^{(k)} \\ \vdots \\ V_n^{(k)} \end{pmatrix}, \quad H^{(k)} = \begin{pmatrix} H_{p+1}^{(k)} \\ \vdots \\ H_n^{(k)} \end{pmatrix}.$$

Step 5. In this step, apply the ALS method to estimate the parameter vector ϕ. Fix $\left(\hat{\boldsymbol{\eta}}^{(k)\top}, \hat{\boldsymbol{\gamma}}^{(k)\top}, \hat{\boldsymbol{\psi}}^{(k)\top}, \hat{\delta}^{(k)}, \hat{\sigma}^{2(k)}\right)^\top$ and minimize $\sum_{i=1}^n (Y_i - u_i^{(k)} - \sum_{j=1}^p \psi_j^{(k)}(y_{i-j} - u_{i-j}^{(k)}))^2$, where $u_i^{(k)} = \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \alpha(\hat{\boldsymbol{\phi}}^{(k)}))\hat{\boldsymbol{\eta}}^{(k)} + \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}}^{(k)}$, to obtain $\hat{\boldsymbol{\phi}}^{(k+1)}$. This step can be optimized using the `optim` function in R, and then the transformed $\hat{\alpha}^{(k+1)}$ is obtained.

Step 6. In $\boldsymbol{B}_1(u)$, the basis functions use k_1 equidistant points as nodes within the domain of $g(\cdot)$. Once $\hat{\boldsymbol{\eta}}^{(k)}$ and $\hat{\boldsymbol{\gamma}}^{(k)}$ are obtained, fit $r = \boldsymbol{Y} - \boldsymbol{B}_1(u)\hat{\boldsymbol{\eta}}^{(k)} - \boldsymbol{W}\hat{\boldsymbol{\gamma}}^{(k)}$ using the R function `selm`, extract the shape parameter α and scale parameter ω from the skew-normal distribution, and use the relationships $\sigma^2 = \frac{\omega^2}{1+\alpha^2}$ and $\delta = \frac{\alpha^2 \omega^2}{1+\alpha^2}$ to obtain $\hat{\delta}^{(k+1)}$ and $\hat{\sigma}^{2^{(k+1)}}$.

Step 7. Repeat Steps 1 to 6 until the convergence criterion is satisfied, where $\|\overline{\boldsymbol{\theta}}^{(k+1)} - \overline{\boldsymbol{\theta}}^{(k)}\|$ is smaller than 10^{-5}.

This section verifies the statistical performance of the proposed algorithm through simulation experiments. The experimental setup is as follows: We generate simulated data with the sample size $n = 1000$ and conduct 100 independent repeated experiments to eliminate randomness. For objective performance evaluation, all comparative experiments are implemented under identical simulated data conditions, with estimation accuracy and computational efficiency quantitatively analyzed through mean square error (MSE) and root MSE (RMSE) metrics.

To evaluate the performance of the proposed algorithm, we compute both the RMSE and the MSE, which are defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^n \left(Y_i - \hat{Y}_i\right)^2},$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^n \left(Y_i - \hat{Y}_i\right)^2,$$

where Y_i is the true value of the i-th observation, $\hat{Y}_i$ is the estimated value for the i-th observation, and n is the total number of samples.

To assess the performance of the estimated link function $g(\cdot)$ and slope function $\beta(\cdot)$, we adopt the root of the average squared errors (RASE), as introduced by Peng et al. [28], which is defined as follows:

$$\text{RASE}_1 = \left(\frac{1}{K_1}\sum_{k=1}^{K_1} (\hat{g}(u_k) - g(u_k))^2\right)^{1/2},$$

$$\text{RASE}_2 = \left(\frac{1}{K_2}\sum_{k=1}^{K_2} \left(\hat{\beta}(t_k) - \beta(t_k)\right)^2\right)^{1/2},$$

where $\{u_k, k = 1, \ldots, K_1\}$ and $\{t_k, k = 1, \ldots, K_2\}$ are the grid points uniformly distributed on the domains of $g(\cdot)$ and $\beta(\cdot)$, respectively. In the two examples below, we choose $K_1 = K_2 = 200$.

The dataset in the simulation experiments is generated according to the following model:

$$Y = \sin(u) + \int_0^1 \beta(t)X(t)\,dt + \varepsilon.$$

In the single-index component, we use the design of Lin et al. [29], $g(u) = \sin(u)$, where $\boldsymbol{u} = \boldsymbol{Z}^\top \boldsymbol{\alpha}$, $\boldsymbol{\alpha} = \frac{(0.2, -0.7)^\top}{\sqrt{0.53}}$, and let $\boldsymbol{Z} = (Z_1, Z_2)^\top$, where $Z_i \overset{iid.}{\sim} \text{Unif}(0, 1)$, for $i = 1, 2$. In the functional linear component, we follow the design of Yu et al. [30], with the slope function set as $\beta(t) = \sqrt{2} \sin\left(\frac{\pi t}{2}\right) + 3\sqrt{2} \sin\left(\frac{3\pi t}{2}\right)$, and $X(t) = \sum_{j=1}^{50} \xi_j v_j(t)$. Here, ξ_j follows a normal distribution with mean 0 and variance $\lambda_j = ((j - 0.5)\pi)^{-2}$ and $v_j(t) = \sqrt{2} \sin((j - 0.5)\pi t)$. The random error ε satisfies the model $\varepsilon_i = e_i + \sum_{l=1}^{p} \psi_l \varepsilon_{i-l}$, where $e_i \sim \text{SN}(-b\delta, \sigma^2, \delta)$, with $\delta = 1$ and $\sigma^2 = 0.2$. This part studies the 1-order autoregressive error structure ($AR(1)$), where we set $\psi = 0.5$.

The computational efficiency of the two algorithms was quantitatively evaluated through average user central processing unit (CPU) time per iteration monitoring. The EM-CALS algorithm demonstrated a mean user time of 80.765 s per iteration, compared to 75.486 s for the TSILS implementation. This observed time difference aligns with the established computational complexity characteristics of EM algorithms versus least squares methods, where EM-based approaches typically require more intensive computations due to their iterative latent variable estimation process.

As shown in Table 1, EM-CALS outperforms TSILS across all metrics, particularly excelling in RASE$_1$ and RASE$_2$. The EM-CALS algorithm shows significant improvement over the TSILS algorithm in terms of the mean values of RMSE, MSE, RASE$_1$, and RASE$_2$, with decreases of 1.19%, 2.45%, 38.00%, and 51.35%, respectively, demonstrating its significant advantage in enhancing prediction accuracy. However, in terms of computational time, TSILS is found to be faster than EM-CALS. Therefore, EM-CALS is better suited for tasks that prioritize accuracy, while TSILS may remain relevant for scenarios where computational efficiency is the primary concern.

Table 1. Algorithm results.

	RMSE		MSE		RASE$_1$		RASE$_2$	
Algorithms	Mean	Var	Mean	Var	Mean	Var	Mean	Var
EM-CALS	0.747	0.000	0.558	0.001	0.498	0.073	0.072	0.001
TSILS	0.756	0.000	0.572	0.001	0.801	0.151	0.148	0.006

3.4. Residual analysis

If the data is correlated, Dunn and Smyth [31] suggested using conditional residuals to ensure the independence and asymptotic normality of the quantile residuals. These residuals can be derived from the conditional residuals defined in model (2.5) as follows:

$$
\begin{aligned}
r_{qi} &= y_i - \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}})\hat{\boldsymbol{\eta}} - \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}} - \sum_{j=1}^{p} \hat{\psi}_j \left(y_{i-j} - \boldsymbol{B}_1^\top(\boldsymbol{Z}_{i-j}^\top \hat{\boldsymbol{\alpha}})\hat{\boldsymbol{\eta}} - \boldsymbol{W}_{i-j}^\top \hat{\boldsymbol{\gamma}} \right) \\
&= \begin{cases} y_i - \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}})\hat{\boldsymbol{\eta}} - \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}}, & i = 1, \ldots, p, \\ y_i - \boldsymbol{B}_1^\top(\boldsymbol{Z}_i^\top \hat{\boldsymbol{\alpha}})\hat{\boldsymbol{\eta}} - \boldsymbol{W}_i^\top \hat{\boldsymbol{\gamma}} - \sum_{j=1}^{p} \hat{\psi}_j \left(y_{i-j} - \boldsymbol{B}_1^\top(\boldsymbol{Z}_{i-j}^\top \hat{\boldsymbol{\alpha}})\hat{\boldsymbol{\eta}} - \boldsymbol{W}_{i-j}^\top \hat{\boldsymbol{\gamma}} \right), & i = p + 1, \ldots, n, \end{cases}
\end{aligned}
$$

where $\boldsymbol{W}_i = \langle X_i(t), \boldsymbol{B}_2(t) \rangle$.

Using the expression for the cdf of a skew-normal distribution as presented in Eq (2.2), the conditional quantile residual can be defined as follows:

$$t_{qi} = \Phi^{-1}\left(F_Y\left(r_{qi}; -b\hat{\delta}, \hat{\sigma}^2, \hat{\delta}\right)\right), \quad i = 1, \ldots, n.$$

According to the research by Dunn and Smyth [31], if the parameter θ can be consistently estimated, the distribution of t_{qi} will asymptotically approach a standard normal distribution. Therefore, the conditional quantile residual can be used to analyze deviations from the error assumptions and identify potential outliers.

4. The local influence approach

Inspired by Ferreira's work [13], we use the conditional expectation of the complete-data log-likelihood function to conduct a local influence analysis.

The perturbation model is defined as $M = \{f(\mathbf{y}_c, \theta, \omega) : \omega \in \Omega\}$, where $\omega = (\omega_1, \ldots, \omega_n)$ represents a perturbation vector that varies within an open region $\Omega \subset \mathbb{R}^n$. The function $f(\mathbf{y}_c, \theta, \omega)$ is the pdf of the complete data $\mathbf{y}_c$ perturbed by ω and $\ell_c(\theta, \omega \mid \mathbf{y}_c) = \log f(\mathbf{y}_c, \theta, \omega)$. Let $\hat{\theta}(\omega)$ be the maximum of the function $Q(\theta, \omega \mid \hat{\theta}) = E\left[\ell_c(\theta, \omega \mid Y_c) \mid \mathbf{y}, \hat{\theta}\right]$. It is assumed that there exists a ω_0 such that $\ell_c(\theta, \omega_0 | Y_c) = \ell_c(\theta | Y_c)$ for all θ. The influence graph is defined as $\alpha(\omega) = (\omega^\top, f_Q(\omega))^\top$, where $f_Q(\omega)$ is the Q-displacement function, defined as: $f_Q(\omega) = 2\left[Q(\hat{\theta}|\hat{\theta}) - Q(\hat{\theta}(\omega)|\hat{\theta})\right]$.

Zhu et al. [33] pointed out that, at the parameter point ω_0, the normal curvature $\mathbf{C}_{f_Q,d}$ of $\alpha(\mathbf{w})$ along the direction of the unit vector $\mathbf{d}$ effectively characterizes the local behavior of the Q-displacement function.

The normal curvature $\mathbf{C}_{f_Q,d}$ of $\alpha(\mathbf{w})$ is defined as:

$$C_{f_Q,d} = -2\mathbf{d}^\top \ddot{Q}_{\omega_0} \mathbf{d} \quad \text{and} \quad -\ddot{Q}_{\omega_0} = \Delta_{\omega_0}^\top \left(-\ddot{Q}_\theta(\hat{\theta})\right)^{-1} \Delta_{\omega_0},$$

where $\ddot{Q}_\theta(\hat{\theta}) = \left.\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial\theta\partial\theta^\top}\right|_{\theta=\hat{\theta}}$ and $\Delta_\omega = \left.\frac{\partial^2 Q(\theta,\omega|\hat{\theta})}{\partial\theta\partial\omega^\top}\right|_{\theta=\hat{\theta}(\omega)}$.

Following Cook's approach, we construct a measure of influence by using the spectral decomposition of the symmetric matrix $-2\ddot{Q}_{\omega_0} = \sum_{k=1}^n \xi_k \mathbf{e}_k \mathbf{e}_k^\top$. Let $\{(\xi_1, \mathbf{e}_1), \ldots, (\xi_n, \mathbf{e}_n)\}$ be the eigenvalue-eigenvector pairs of the matrix $-2\ddot{Q}_{\omega_0}$, where the first q eigenvalues satisfy $\xi_1 \geq \cdots \geq \xi_q > 0$ and $\xi_{q+1} = \cdots = \xi_n = 0$, and $\{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$ forms an orthonormal basis. Based on the methods of Zhu et al. [33], the aggregated contribution vector $M(0)_l$ is defined as the normalized weighted sum of the components corresponding to all nonzero eigenvalues:

$$M(0)_l = \sum_{k=1}^q \tilde{\xi}_k e_{kl}^2,$$

where $\tilde{\xi}_k = \frac{\xi_k}{\sqrt{\sum_{j=1}^q \xi_j^2}}$ represents the normalized eigenvalue weights, and $\mathbf{e}_k^2 = (e_{k1}^2, \ldots, e_{kn}^2)$ is the square of the components of the eigenvector $\mathbf{e}_k$.

To identify influential cases, we perform a preliminary evaluation by inspecting the graph of $\{M(0)_l, l = 1, \ldots, n\}$. Following Ferreira's approach [13], we use $\frac{1}{n} + c^*S$ as a benchmark, and consider the l-th case as influential if $M(0)_l$ exceeds this benchmark. Here, c^* is a constant selected based on the specific application, and S is the standard deviation of the vector $\{M(0)_l, l = 1, \ldots, n\}$.

Let A be an $n \times n$ matrix. The result of $\frac{\partial A}{\partial \psi_i}$ is that the elements on the i-th lower off-diagonal are all -1, and the other elements are 0. The result of $\frac{\partial A^T}{\partial \psi_i}$ is that the elements on the i-th upper off-diagonal are all -1, and the other elements are 0. The k-th row of $\frac{\partial A}{\partial \psi_i}$ is $\frac{\partial A_k^\top}{\partial \psi_i}$.

When $i = 2$, the results are shown below:

$$\frac{\partial A}{\partial \psi_2} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ -1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & -1 & 0 & 0 \end{bmatrix}_{n \times n},$$

$$\frac{\partial A^\top}{\partial \psi_2} = \begin{bmatrix} 0 & 0 & -1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{n \times n}.$$

4.1. The Hessian matrix, $\ddot{Q}(\hat{\theta})$

To obtain the diagnostic measures of the SIPFLM-SNAR(P), based on the approach of Zhu and Lee [33], it is necessary to compute $\ddot{Q}_\theta(\hat{\theta}) = \frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \theta \partial \theta^\top}$, where $\theta = (\eta^T, \gamma^T, \sigma^2, \delta, \psi^T)^T$. $\ddot{Q}_\theta(\hat{\theta})$ has elements given by the following expression:

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \eta \partial \eta^\top} = -\frac{1}{\sigma^2} \left(\mathbf{AB}_1(\mathbf{Z}^\top \alpha)\right)^\top \mathbf{AB}_1(\mathbf{Z}^\top \alpha),$$

$$\frac{\partial^2 Q(\theta \mid \hat{\theta})}{\partial \gamma \partial \eta^\top} = -\frac{1}{\sigma^2} \left(\mathbf{AW}\right)^\top \mathbf{AB}_1(\mathbf{Z}^\top \alpha),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \sigma^2 \partial \eta} = -\frac{1}{\sigma^4} \left(\mathbf{AB}_1(\mathbf{Z}^\top \alpha)\right)^\top \left(\mathbf{y} - \xi - \delta(\hat{\mathbf{z}} - b\mathbf{1}_n)\right),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \delta \partial \eta} = -\frac{1}{\sigma^2} \left(\mathbf{AB}_1(\mathbf{Z}^\mathbf{T} \alpha)\right)^\top (\hat{\mathbf{z}} - b\mathbf{1}_n),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_i \partial \eta} = \frac{1}{\sigma^2} \left(\mathbf{AB}_1(\mathbf{Z}^\top \alpha)\right)^\top \cdot \frac{\partial \mathbf{A}}{\partial \psi_i} \mathbf{r} + \frac{1}{\sigma^2} \mathbf{B}_1^\top(\mathbf{Z}^\top \alpha) \cdot \frac{\partial \mathbf{A}^\top}{\partial \psi_i} (\mathbf{y} - \xi - \delta(\hat{\mathbf{z}} - b\mathbf{1}_n)), i = 1, \ldots, p,$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \gamma \partial \gamma^\top} = -\frac{1}{\sigma^2} \left(\mathbf{AW}\right)^\top \mathbf{AW},$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \sigma^2 \, \partial \gamma} = -\frac{1}{\sigma^4} \, (\mathbf{AW})^\top \left(\mathbf{y} - \boldsymbol{\xi} - \delta(\hat{\mathbf{z}} - b\mathbf{1}_n) \right),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \delta \, \partial \gamma} = -\frac{1}{\sigma^2} \, (\mathbf{AW})^\top (\hat{\mathbf{z}} - b\mathbf{1}_n),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_i \partial \gamma} = \frac{1}{\sigma^2} \, (\mathbf{AW})^\top \cdot \frac{\partial \mathbf{A}}{\partial \psi_i} \mathbf{r} + \frac{1}{\sigma^2} \left(\mathbf{W}^\top \frac{\partial \mathbf{A}^\top}{\partial \psi_i} \right) (\mathbf{y} - \boldsymbol{\xi} - \delta \, (\hat{\mathbf{z}} - b\mathbf{1}_n)), i = 1, \ldots, p,$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \sigma^4} = \frac{n}{2\sigma^4} - \frac{1}{\sigma^6} \sum_{i=1}^{n} \left[(y_i - \xi_i)^2 - 2\delta(y_i - \xi_i)(\hat{z}_i - b) + \delta^2(b^2 - 2b\hat{z}_i + \hat{z_i^2}) \right],$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \delta \partial \sigma^2} = \frac{1}{\sigma^4} \sum_{i=1}^{n} \left[-(y_i - \xi_i)(\hat{z}_i - b) + \delta(b^2 - 2b\hat{z}_i + \hat{z_i^2}) \right],$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_i \partial \sigma^2} = \frac{1}{\sigma^4} \boldsymbol{r}^\top \left(\frac{\partial \mathbf{A}^\top}{\partial \psi_i} \right) \mathbf{A} \mathbf{r} - \frac{\delta}{\sigma^4} \mathbf{r}^\top \frac{\partial \mathbf{A}^\top}{\partial \psi_i} (\hat{\mathbf{z}} - b\mathbf{1}_n), i = 1, \ldots, p,$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \delta^2} = -\frac{1}{\sigma^2} \sum_{i=1}^{n} \left(b^2 - 2b\hat{z}_i + \hat{z_i^2} \right),$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_i \partial \delta} = \frac{1}{\sigma^2} \mathbf{r}^\top \frac{\partial \mathbf{A}^\top}{\partial \psi_i} (\hat{\mathbf{z}} - b\mathbf{1}_n), i = 1, \ldots, p,$$

$$\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_j \partial \psi_i} = -\frac{1}{2\sigma^2} \mathbf{r}^\top \left(\frac{\partial \mathbf{A}^\top}{\partial \psi_i} \cdot \frac{\partial \mathbf{A}}{\partial \psi_j} + \frac{\partial \mathbf{A}^\top}{\partial \psi_j} \cdot \frac{\partial \mathbf{A}}{\partial \psi_i} \right) \mathbf{r}, i = 1, \ldots, p, j = 1, \ldots, p,$$

where $\mathbf{1}_n$ is a vector of ones, and $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)$, $\boldsymbol{r} = \boldsymbol{y} - \mathbf{B}_1(\mathbf{Z}^\top \alpha)\boldsymbol{\eta} - W\gamma$, $\hat{\mathbf{z}} = (\hat{z}_1, \ldots, \hat{z}_n)^\top$.

In the above formula, $\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_i \partial \eta}$ represents the i-th column of $\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi^\top \partial \eta}$, and $\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi_j \partial \psi_i}$ corresponds to the (i, j)-th element of $\frac{\partial^2 Q(\theta|\hat{\theta})}{\partial \psi \partial \psi^\top}$; other symbols follow the same pattern.

4.2. Perturbation schemes

In this section, we consider the three usual perturbation schemes in local influence for the SIPFLM-SNAR(P) proposed in this work.

4.2.1. Case-weight perturbation

This section examines whether differentially weighted observations influence the maximum likelihood estimation of θ. The perturbed Q-function is written as: $Q(\theta, \omega|\hat{\theta}) = \sum_{i=1}^{n} \omega_i Q_i(\theta|\hat{\theta})$. In this case, $\omega_0 = (1, \ldots, 1) = \mathbf{1}_n$ and $\frac{\partial Q(\theta, \omega|\hat{\theta})}{\partial \omega_i} = Q_i(\theta|\hat{\theta})$, and Δ_{ω_0} has elements $\frac{\partial Q_i(\theta|\hat{\theta})}{\partial \theta}$, $i = 1, \ldots, n$.

$$\frac{\partial Q_i(\theta|\hat{\theta})}{\partial \eta} = \frac{1}{\sigma^2} \left(\mathbf{B}_1^\top (\mathbf{Z}^\top \alpha) A_i \right) (y_i - \xi_i - \delta(\hat{z}_i - b)),$$

$$\frac{\partial Q_i(\theta|\hat{\theta})}{\partial \gamma} = \frac{1}{\sigma^2} \left(W^\top A_i \right) (y_i - \xi_i - \delta(\hat{z}_i - b)),$$

$$\frac{\partial Q_i(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})}{\partial \sigma^2} = -\frac{1}{2\sigma^2} + \frac{1}{2\sigma^4}\left[(y_i - \xi_i)^2 - 2\delta(y_i - \xi_i)(\hat{z}_i - b) + \delta^2(b^2 - 2b\hat{z}_i + \hat{z}_i^2)\right],$$

$$\frac{\partial Q_i(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})}{\partial \delta} = \frac{1}{\sigma^2}\left[(y_i - \xi_i)(\hat{z}_i - b) - \delta(b^2 - 2b\hat{z}_i + \hat{z}_i^2)\right],$$

$$\frac{\partial Q_i(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})}{\partial \psi_k} = -\frac{1}{\sigma^2}(A_i^T\mathbf{r} - \delta(\hat{z}_i - b))\left(\frac{\partial \mathbf{A_i}^\top}{\partial \psi_k}\right)\mathbf{r}, k = 1, \ldots, p.$$

4.2.2. Response variable perturbation

Inspired by Ferreira's work [13], we consider an additive perturbation given by

$$y_{i\omega} = y_i + S_y\omega_i, \quad i = 1, \ldots, n,$$

where S_y is the standard deviation of $\mathbf{y}$. In this case, $\omega_0 = \mathbf{0} : n \times 1$, and by replacing y_i with $y_{i\omega}$ in the Q-function, we obtain $Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})$.

It follows that the matrix

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega}|\hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\theta}\partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}},$$

where

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\eta}\partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}} = \frac{S_y}{\sigma^2}\left(\mathbf{AB_1}(\mathbf{Z^T}\boldsymbol{\alpha})\right)^\top \mathbf{A},$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\gamma}\partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}} = \frac{S_y}{\sigma^2}\left(\mathbf{AW}\right)^\top \mathbf{A},$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \sigma^2\partial \boldsymbol{\omega}}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}} = \frac{S_y}{\sigma^4}\left(\mathbf{A}^\top\mathbf{A}\mathbf{r} - \delta\mathbf{A}^\top(\hat{\mathbf{z}} - b\mathbf{1}_n)\right),$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \delta\partial \boldsymbol{\omega}}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}} = \frac{S_y}{\sigma^2}\mathbf{A}^\top(\hat{\mathbf{z}} - b\mathbf{1}_n),$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \psi_i\partial \boldsymbol{\omega}}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega_0}} = -\frac{S_y}{\sigma^2}\left[\left(\frac{\partial \mathbf{A}^\top}{\partial \psi_i} \cdot A + A^\top \cdot \frac{\partial \mathbf{A}}{\partial \psi_i}\right)r - \left(\delta\frac{\partial \mathbf{A}^\top}{\partial \psi_i}\right)(\hat{\mathbf{z}} - b\mathbf{1}_n)\right].$$

4.2.3. Explanatory variable perturbation

Inspired by the research of Zou et al. [19], we consider the following perturbation.

We perturb $\mathbf{Z}_i$ as follows: $\mathbf{Z}_i + l_j\boldsymbol{\omega}^\top\mathbf{k}_i$, where $\boldsymbol{\omega} \in \mathbb{R}^n$, $l_j \in \mathbb{R}^l$, $\mathbf{k}_i \in \mathbb{R}^n$, where l_j and $\mathbf{k}_i$ are unit vectors with their j-th and i-th elements equal to 1, respectively. It means that only the j-th covariate is being perturbed. In this case, $\omega_0 = \mathbf{0} : n \times 1$. By replacing $\mathbf{Z}_i$ in the Q-function with $\mathbf{Z}_i + l_j\boldsymbol{\omega}^\top\mathbf{k}_i$, we obtain: $Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})$.

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\eta} \partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0} = \frac{1}{\sigma^2}\Bigg[\sum_{i=1}^{n}\Bigg[\Bigg(-\boldsymbol{B}_1(\boldsymbol{Z}_i^\top\boldsymbol{\alpha}) + \sum_{l=1}^{p}\psi_l \boldsymbol{B}_1(\boldsymbol{Z}_{i-l}^\top\boldsymbol{\alpha})\Bigg)$$
$$\cdot\Bigg(c_i\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l(c_{i-l}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top)\Bigg)$$
$$+ (y_i - \xi_i - \delta(\hat{z}_i - b))$$
$$\cdot\Bigg(\dot{\boldsymbol{B}}_1(\boldsymbol{Z}_i^\top\boldsymbol{\alpha})\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l\dot{\boldsymbol{B}}_1(\boldsymbol{Z}_{i-l}^\top\boldsymbol{\alpha})\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top\Bigg)\Bigg]\Bigg],$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \boldsymbol{\gamma} \partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0} = \frac{1}{\sigma^2}\Bigg[\sum_{i=1}^{n}\Bigg(-\boldsymbol{W}_i + \sum_{l=1}^{p}\psi_l\boldsymbol{W}_{i-l}\Bigg)\Bigg(c_i\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l\Big(c_{i-l}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top\Big)\Bigg)\Bigg],$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \delta \partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0} = \frac{1}{\sigma^2}\Bigg[\sum_{i=1}^{n}-(\hat{z}_i - b)\Bigg(c_i\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l\Big(c_{i-l}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top\Big)\Bigg)\Bigg],$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \sigma^2 \partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0} = -\frac{1}{\sigma^4}\Bigg[\sum_{i=1}^{n}(y_i - \xi_i - \delta(\hat{z}_i - b))\Bigg(c_i\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l\Big(c_{i-l}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top\Big)\Bigg)\Bigg],$$

$$\left.\frac{\partial^2 Q(\boldsymbol{\theta}, \boldsymbol{\omega} \mid \hat{\boldsymbol{\theta}})}{\partial \psi_k \partial \boldsymbol{\omega}^\top}\right|_{\boldsymbol{\omega}=\boldsymbol{\omega}_0} = -\frac{1}{\sigma^2}\Bigg[\sum_{i=1}^{n}\Bigg[r_{i-k}\Bigg(c_i\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_i^\top - \sum_{l=1}^{p}\psi_l\Big(c_{i-l}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-l}^\top\Big)\Bigg)+(y_i-\xi_i-\delta(\hat{z}_i-b))(c_{i-k}\boldsymbol{\alpha}^\top\boldsymbol{l}_j\boldsymbol{k}_{i-k}^\top)\Bigg]\Bigg],$$

where c_i is given by $c_i = \dot{\boldsymbol{B}}_1^\top(\boldsymbol{Z}_i^\top\boldsymbol{\alpha})\boldsymbol{\eta}$, and it is important to note that all terms must have indices greater than 0; otherwise, the term will be equal to 0.

5. Simulation study

In this section, we examine the properties of the proposed estimator using two simulation examples. We consider three different sample sizes: $n = 100, 300,$ and 700. Each example is repeated 300 times.

Additionally, we utilize MSE, bias, and variance (var) values to assess the performance of parameter estimation. For instance, in the case of δ, the MSE, bias, and var values are calculated using the following equations:

$$\text{MSE}(\hat{\delta}) = \frac{1}{300}\sum_{j=1}^{300}\left(\hat{\delta}_j - \delta\right)^2, \tag{5.1}$$

$$\text{Bias}(\hat{\delta}) = \frac{1}{300}\sum_{j=1}^{300}\left(\hat{\delta}_j - \delta\right), \tag{5.2}$$

$$\text{Var}(\hat{\delta}) = \frac{1}{300}\sum_{j=1}^{300}\left(\hat{\delta}_j - \bar{\delta}\right)^2, \tag{5.3}$$

where δ is the true value, and $\bar{\delta}$ is the mean of $\{\hat{\delta}_j\}_{j=1,\dots,300}$. To assess the performance of the estimated link function $g(\cdot)$ and slope function $\beta(\cdot)$, we adopt the RASE as a metric.

Following the idea of Yu et al. [7], the simulation experiment uses a cubic B-spline with evenly distributed knots. To simplify the computational complexity and ensure numerical stability, the numbers of spline basis functions, N_1 and N_2, are selected by minimizing the BIC.

5.1. Example 1

The dataset for the first example is generated according to the following model:

$$Y = \sin(u) + \int_0^1 \beta(t)X(t)\,dt + \varepsilon.$$

In the single-index component, we use the design of Lin et al. [29], $g(u) = \sin(u)$, where $\boldsymbol{u} = \boldsymbol{Z}^\top \boldsymbol{\alpha}$, $\boldsymbol{\alpha} = \frac{(0.2, -0.7)^\top}{\sqrt{0.53}}$, and let $\boldsymbol{Z} = (Z_1, Z_2)^\top$, where $Z_i \overset{iid}{\sim} \text{Unif}(0,1)$, for $i = 1, 2$. In the functional linear component, we follow the design of Yu et al. [30], with the slope function set as $\beta(t) = \sqrt{2}\sin\left(\frac{\pi t}{2}\right) + 3\sqrt{2}\sin\left(\frac{3\pi t}{2}\right)$, and $X(t) = \sum_{j=1}^{50} \xi_j v_j(t)$. Here, ξ_j follows a normal distribution with mean 0 and variance $\lambda_j = ((j - 0.5)\pi)^{-2}$, and $v_j(t) = \sqrt{2}\sin((j - 0.5)\pi t)$. The random error ε satisfies the model $\varepsilon_i = e_i + \sum_{l=1}^p \psi_l \varepsilon_{i-l}$, where $e_i \sim \text{SN}(-b\delta, \sigma^2, \delta)$, with $\sigma^2 = 0.2$.

To assess the algorithm's robustness, we examine its performance under various error structures by adjusting the autoregressive coefficient, order, and skewness. We focus on three cases: Case 1, where we have a 1-order autoregressive structure ($AR(1)$) with $\delta = 0.7$; Case 2, where we have a 2-order autoregressive structure ($AR(2)$) with $\delta = 0.7$; and Case 3, where we again use a 1-order autoregressive structure ($AR(1)$) but with $\delta = 1$. For the autoregressive structures, we set $\psi = 0.5$ for $AR(1)$, and $\psi_1 = -0.7$, $\psi_2 = 0.2$ for $AR(2)$.

Tables 2 and 3 present the MSE, Var, and bias for the parameters $\boldsymbol{\alpha}$, $\boldsymbol{\psi}$, σ^2, and δ, along with the sample mean, median, and variance of RASE_j (for $j = 1, 2$) at different sample sizes. It is evident that as the sample size n increases from 100 to 300 and further to 700, both the MSE and the sample statistics (mean, median, and variance) of RASE_j show a decreasing trend. Based on the above results, it can be seen that the B-splines provide asymptotically unbiased estimates for the nonparametric components. Although the bias of α_1 and ψ fluctuates slightly, it shows a decreasing trend as the sample size increases, while the bias of the other parameters consistently decreases as the sample size grows. Figures 2 and 3 present the true curves and the fitted curves based on the estimates. It is evident that as the sample size increases, the estimated curves gradually approach the true curves, indicating that the algorithm's fitting performance for the nonparametric components improves with larger sample sizes. Overall, the simulation results demonstrate that the proposed estimation method is effective.

Table 2. Simulation results for nonparametric components of Case 1.

n	Criterion	Mean	Median	Var
100	$RASE_1$	1.266	1.037	1.106
	$RASE_2$	0.297	0.188	0.092
300	$RASE_1$	0.695	0.646	0.104
	$RASE_2$	0.131	0.110	0.017
700	$RASE_1$	0.486	0.433	0.044
	$RASE_2$	0.073	0.066	0.001

Table 3. Simulation results for parametric components of Case 1.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{0.2}{\sqrt{0.53}}$	0.025	0.025	0.009	0.013	0.013	0.017	0.006	0.006	0.005
α_2	$\frac{-0.7}{\sqrt{0.53}}$	0.471	0.403	0.261	0.065	0.063	0.045	0.001	0.001	0.005
ψ	0.5	0.009	0.009	0.001	0.002	0.002	-0.002	0.001	0.001	0
σ^2	0.2	0.011	0.011	-0.024	0.004	0.004	0.005	0.001	0.011	-0.002
δ	0.7	0.112	0.104	-0.091	0.031	0.028	-0.053	0.008	0.008	-0.013
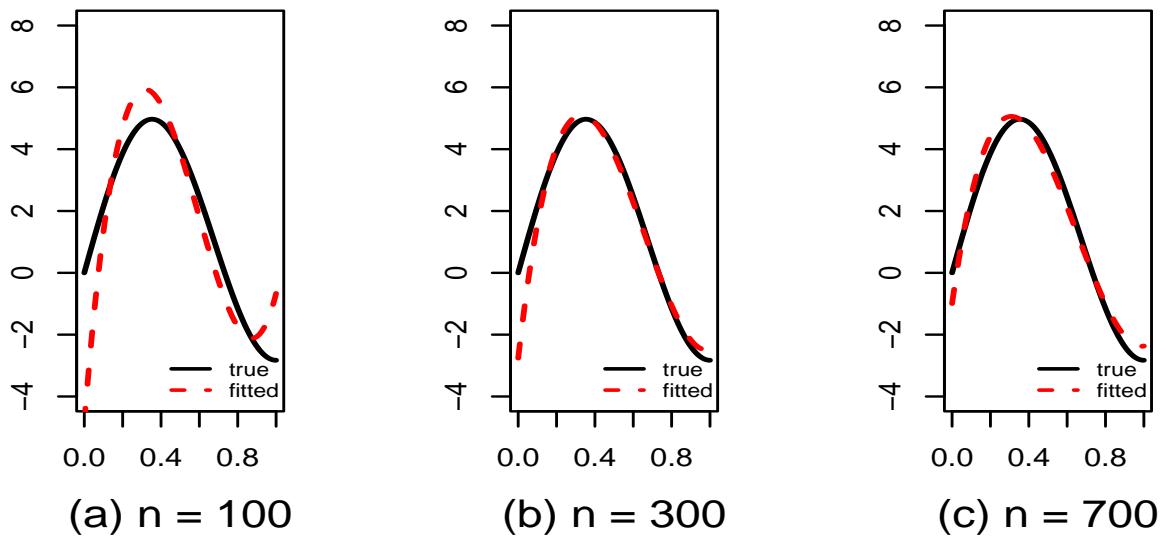
**Figure 2.** The estimated curves of $\beta(t)$ in Case 1.
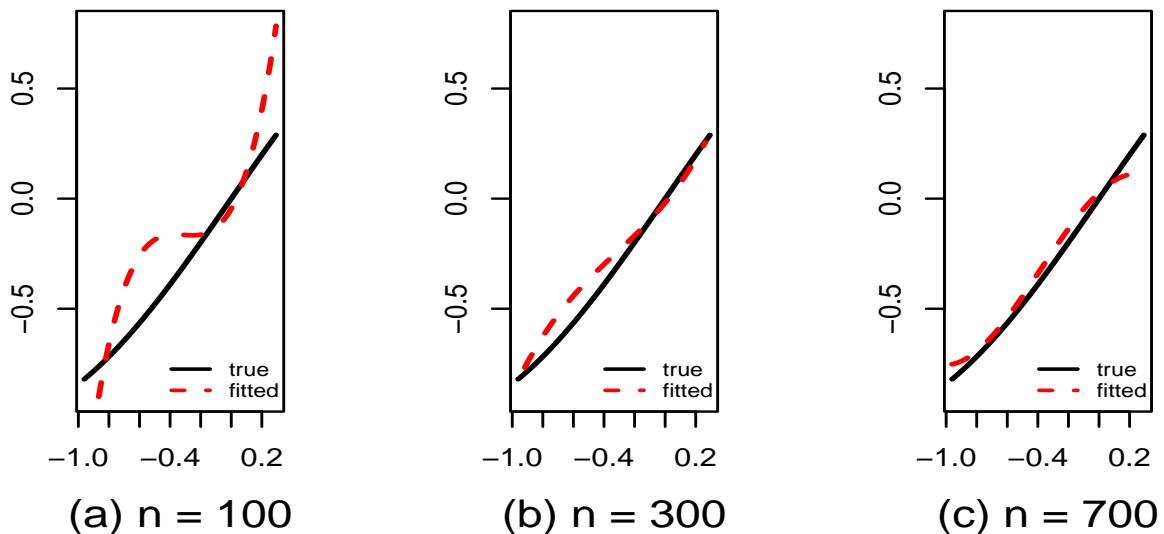
Figure 3. The estimated curves of $g(t)$ in Case 1.

To validate the stability of the algorithm, Case 2 extends the error structure from 1-order autoregressive (AR(1)) to 2-order autoregressive (AR(2)), increasing the complexity of the error model. As shown in Tables 4 and 5, the bias and MSE of the parameters α, ψ, and δ, as well as the sample

mean, median, and variance of RASE_j (for $j = 1, 2$), decrease as the sample size increases. Although the bias of σ^2 fluctuates slightly, its MSE still decreases with increasing sample size. This indicates that as the sample size grows, the accuracy and stability of the parameter estimates improve. From Figures 4 and 5, it can be observed that the estimated curves gradually approach the true curves.

Table 4. Simulation results for nonparametric components of Case 2.

n	Criterion	Mean	Median	Var
100	RASE_1	1.149	0.923	0.591
	RASE_2	0.336	0.170	0.120
300	RASE_1	0.620	0.553	0.097
	RASE_2	0.149	0.086	0.053
700	RASE_1	0.436	0.397	0.025
	RASE_2	0.083	0.055	0.022

Table 5. Simulation results for parametric components of Case 2.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{0.2}{\sqrt{0.53}}$	0.031	0.031	0.027	0.010	0.010	0.007	0.005	0.005	-0.006
α_2	$\frac{-0.7}{\sqrt{0.53}}$	0.800	0.606	0.440	0.268	0.246	0.147	0.102	0.099	0.054
ψ_1	-0.7	0.018	0.016	-0.050	0.004	0.003	-0.012	0.001	0.001	-0.003
ψ_2	0.2	0.020	0.015	-0.066	0.004	0.003	-0.017	0.001	0.001	-0.007
σ^2	0.2	0.014	0.014	-0.007	0.005	0.005	0.011	0.003	0.003	0.009
δ	0.7	0.156	0.128	-0.167	0.057	0.050	-0.084	0.040	0.037	-0.055
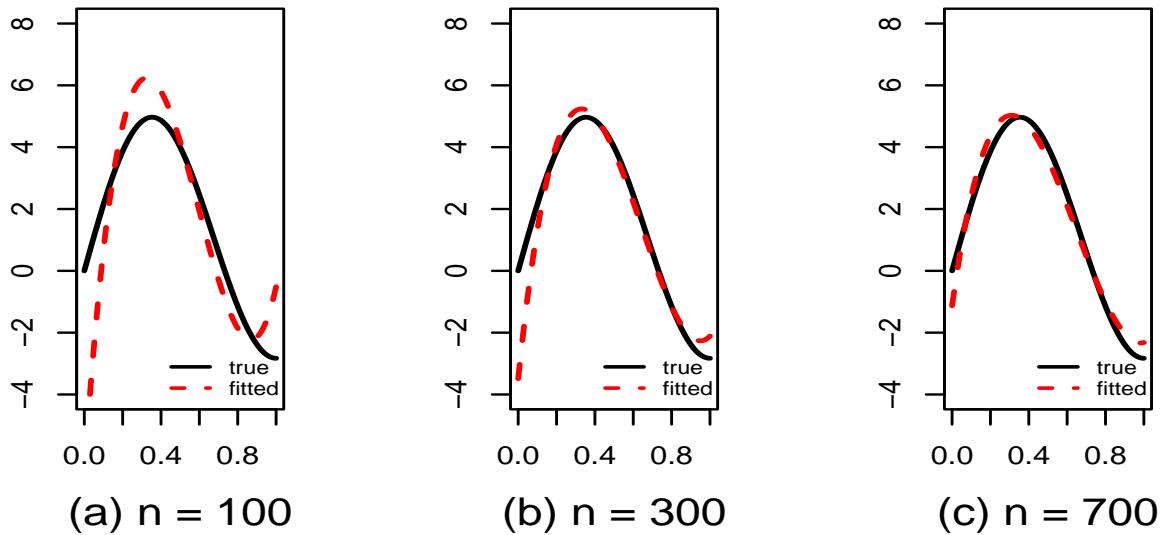
Figure 4. The estimated curves of $\beta(t)$ in Case 2.
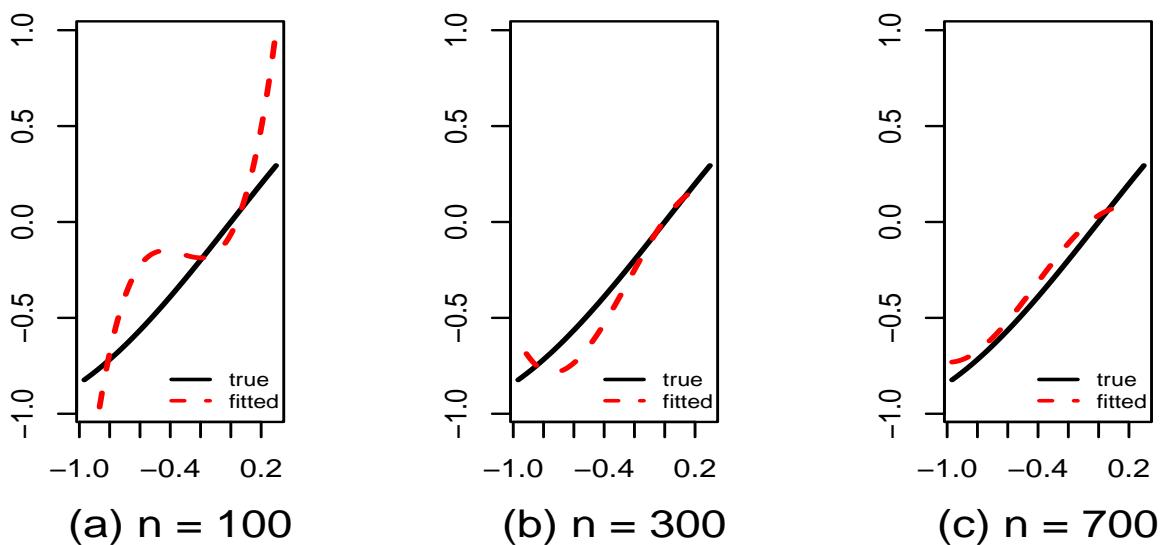
Figure 5. The estimated curves of $g(t)$ in Case 2.

To evaluate the performance of the proposed algorithm under data with varying skewness, Case 3 systematically increased the skewness parameter δ from 0.7 to 1. As shown in Tables 6 and 7, the bias and mean squared error (MSE) of the parameters α, ψ, and δ, as well as the sample mean, median, and variance of RASE_j (for $j = 1, 2$), decrease with increasing sample size across different sample

sizes. Although the bias of σ^2 fluctuates slightly, it remains at a low level (ranging from -0.001 to -0.004). Additionally, as seen in Figures 6 and 7, the estimated functions progressively approach the true functions as the sample size increases.

From the algorithm's performance under $AR(1)$ and $AR(2)$ error structures, as well as varying levels of skewness, it can be observed that the proposed estimation method demonstrates good consistency and effectiveness in complex error structures. Furthermore, the estimation accuracy improves significantly as the sample size increases.

To systematically evaluate the time complexity and efficiency of the algorithm, this paper measures the average runtime of the EM-CALS algorithm under Cases 1–3 scenarios with sample sizes of $n = 100$, 300, and 700. As shown in Table 8, the experiment execution time does not directly increase or decrease with the increase in sample size. This is related to the more flexible convergence criteria applied in the EM-CALS algorithm.

Table 6. Simulation results for nonparametric components of Case 3.

n	Criterion	Mean	Median	Var
100	RASE$_1$	1.149	0.923	0.591
	RASE$_2$	0.336	0.170	0.120
300	RASE$_1$	0.620	0.553	0.097
	RASE$_2$	0.149	0.086	0.053
700	RASE$_1$	0.436	0.397	0.025
	RASE$_2$	0.083	0.055	0.022

Table 7. Simulation results for parametric components of Case 3.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{0.2}{\sqrt{0.53}}$	0.038	0.037	0.030	0.018	0.017	0.021	0.009	0.009	0.007
α_2	$\frac{-0.7}{\sqrt{0.53}}$	0.674	0.530	0.378	0.180	0.168	0.109	0.001	0.001	0.007
ψ	0.5	0.009	0.009	0.009	0.002	0.002	-0.001	0.001	0.001	0
σ^2	0.2	0.018	0.017	-0.034	0.004	0.004	-0.001	0.001	0.001	-0.004
δ	1	0.108	0.104	-0.060	0.014	0.013	-0.028	0.005	0.004	-0.007
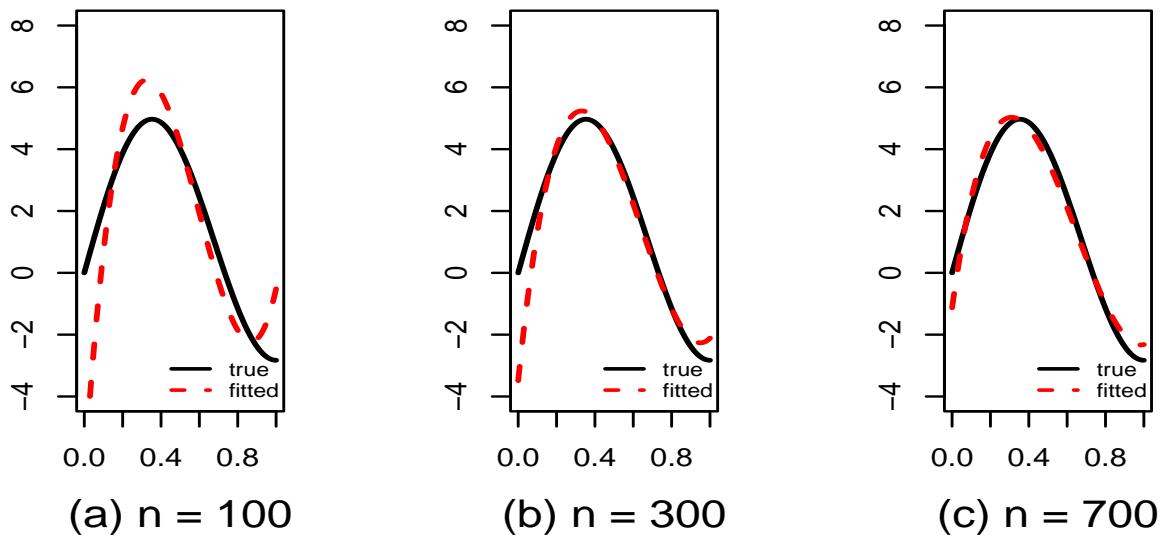
Figure 6. The estimated curves of $\beta(t)$ in Case 3.
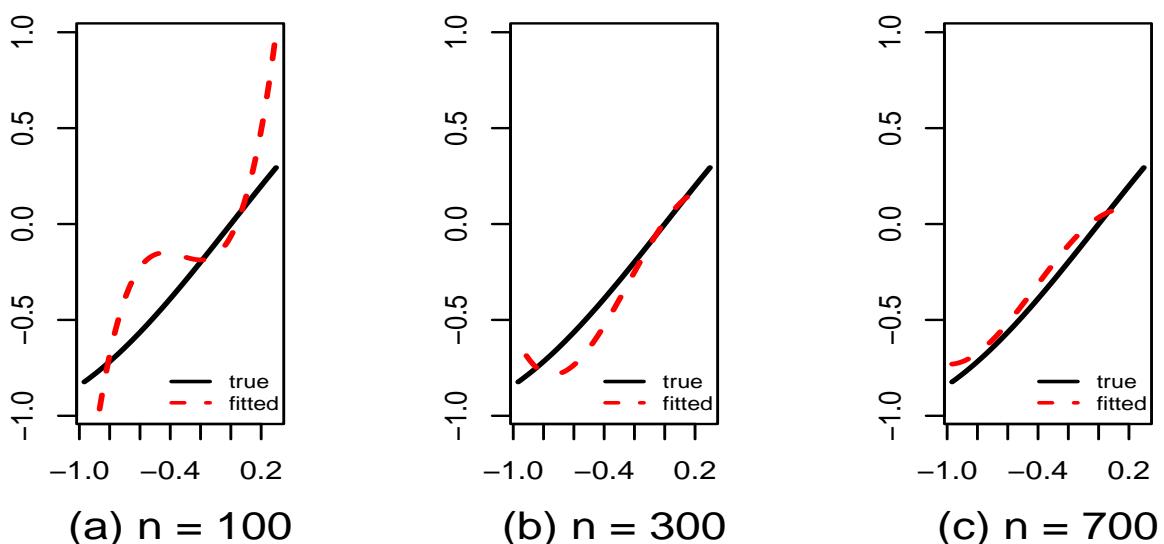
Figure 7. The estimated curves of $g(t)$ in Case 3.

Table 8. Experiment Execution Time(s).

n		User	System	Elapsed
100	Case 1	146.328	2.723	151.925
	Case 2	189.914	2.816	196.018
	Caee 3	143.048	2.206	148.332
300	Case 1	123.117	2.297	127.611
	Case 2	256.243	3.896	263.757
	Case 3	50.093	0.829	51.795
700	Case 1	135.031	2.781	140.028
	Case 2	203.583	3.370	207.119
	Case 3	82.259	1.514	84.915

5.2. *Example 2*

The dataset is generated based on the following model:

$$Y = -2(u-1)^2 + 1 + \int_0^1 \beta(t)X(t)\,dt + \varepsilon,$$

where $\boldsymbol{u} = \boldsymbol{Z}^\top \boldsymbol{\alpha}$, $\boldsymbol{Z} = (Z_1, Z_2)^\top$, and $Z_i \overset{iid}{\sim}$ Unif$(0, 1)$, $i = 1, 2$. The single-index vector is $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)^\top = \left(\frac{\sqrt{3}}{3}, \frac{\sqrt{6}}{3}\right)^\top$. For the slope function, the generation of $X(t)$ and ε follows the same design as in Example 1.

To evaluate the robustness of the algorithm, we adopt the same design as in Example 1. By varying the skewness parameter, autoregressive coefficients, and orders, the algorithm's performance is assessed under different error structures. Specifically, we consider three cases: Case 4, which involves a 1-order autoregressive structure ($AR(1)$) with $\delta = 0.7$; Case 5, which uses a 2-order autoregressive structure ($AR(2)$) with $\delta = 0.7$; and Case 6, which again employs a 1-order autoregressive structure ($AR(1)$), but with $\delta = 1$. For the autoregressive structures, we set $\psi = 0.5$ for $AR(1)$, and $\psi_1 = -0.7$, $\psi_2 = 0.2$ for $AR(2)$.

From Tables 9 and 10, it is evident that as the sample size n increases, the sample mean, median, and variance of RASE$_1$ and RASE$_2$ gradually decrease. This trend indicates that the algorithm's performance in fitting the nonparametric components improves with larger sample sizes. Additionally, the MSE values for all parameters also decline as n increases. The Var consistently decreases with larger n, indicating improved stability of the estimators. Although the bias of some parameters, such as α_1, shows slight fluctuations across different sample sizes, these variations remain small and are generally kept at low levels. These fluctuations may be linked to the complexity of the error terms.

Table 9. Simulation results for nonparametric components of Case 4.

n	Criterion	Mean	Median	Var
100	$RASE_1$	1.269	1.073	0.644
	$RASE_2$	0.199	0.160	0.113
300	$RASE_1$	0.706	0.625	0.182
	$RASE_2$	0.104	0.099	0.002
700	$RASE_1$	0.478	0.429	0.041
	$RASE_2$	0.070	0.067	0.001

Table 10. Simulation results for parametric components of Case 4.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{\sqrt{3}}{3}$	0.017	0.017	-0.010	0.005	0.005	0.001	0.002	0.002	0.004
α_2	$\frac{\sqrt{6}}{3}$	0.030	0.030	-0.022	0.002	0.002	-0.005	0.001	0.001	-0.004
ψ	0.5	0.010	0.010	0.002	0.002	0.002	-0.001	0.001	0.001	0
σ^2	0.2	0.011	0.011	-0.029	0.004	0.004	0.006	0.001	0.001	-0.002
δ	0.7	0.112	0.106	-0.079	0.033	0.030	-0.057	0.008	0.008	-0.014

Figures 8 and 9, it is evident that as the sample size n increases, the performance of the fitted functions $\beta(t)$ and $g(t)$ using B-splines improves significantly. When $n = 100$, a noticeable deviation exists between the fitted curves and the true curves. However, as n increases to 300 and 700, the fitted curves progressively converge to the true curves. This indicates that larger sample sizes allow the algorithm to more effectively capture the characteristics of the true curves, leading to improved estimation accuracy and convergence.
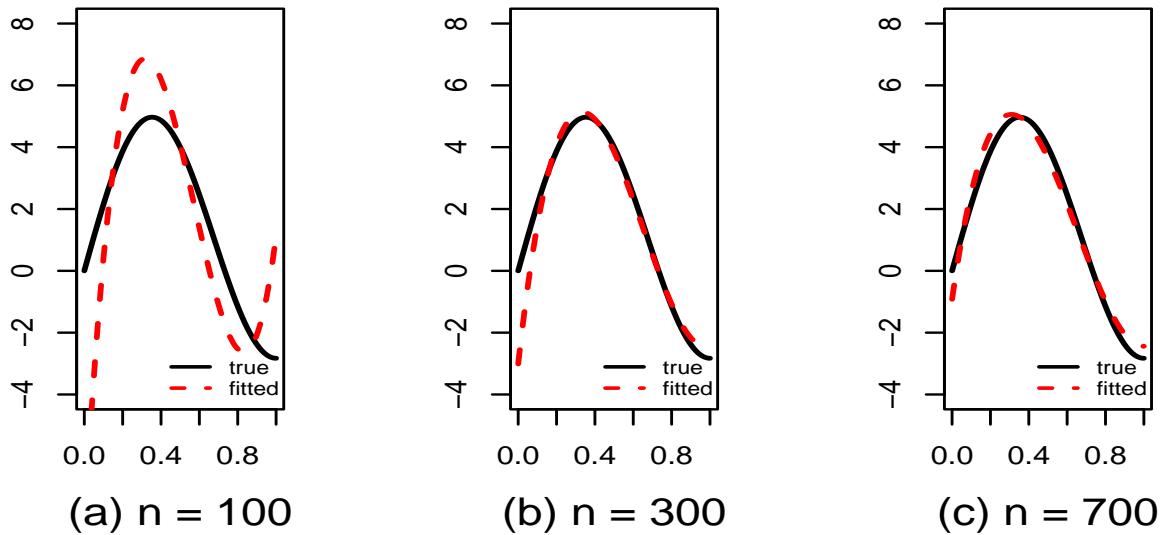
Figure 8. The estimated curves of $\beta(t)$ in Case 4.
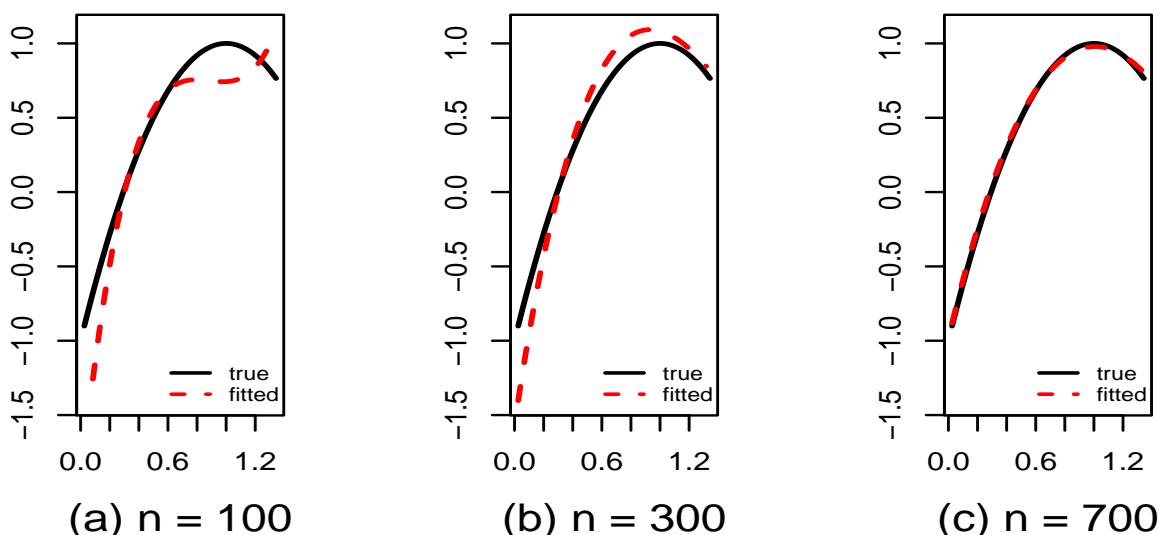
Figure 9. The estimated curves of $g(t)$ in Case 4.

The analysis of Tables 11 and 12 reveals that as the sample size n increases, the algorithm's performance in fitting the nonparametric components improves consistently. This improvement is reflected in reducing the sample mean, median, and variance of $RASE_1$ and $RASE_2$. Moreover, as the sample size grows, the MSE and the variance of the parameters also decrease. For the bias of the

parameters α_1 and σ^2, although there are some fluctuations, the values remain within a relatively small range overall. Increasing the sample size results in more precise and stable estimation outcomes.

Table 11. Simulation results for nonparametric components of Case 5.

n	Criterion	Mean	Median	Var
100	$RASE_1$	1.097	0.908	0.559
	$RASE_2$	0.333	0.139	0.766
300	$RASE_1$	0.618	0.540	0.095
	$RASE_2$	0.085	0.064	0.062
700	$RASE_1$	0.438	0.403	0.024
	$RASE_2$	0.049	0.047	0

Table 12. Simulation results for parametric components of Case 5.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{\sqrt{3}}{3}$	0.022	0.021	-0.026	0.004	0.004	-0.003	0.001	0.001	-0.004
α_2	$\frac{\sqrt{6}}{3}$	0.157	0.149	-0.091	0.012	0.012	-0.008	0.001	0.001	0.001
ψ_1	-0.7	0.019	0.016	-0.054	0.004	0.004	-0.012	0.001	0.001	-0.004
ψ_2	0.2	0.020	0.016	-0.068	0.004	0.003	-0.017	0.001	0.001	-0.007
σ^2	0.2	0.013	0.013	-0.011	0.006	0.005	0.015	0.003	0.003	0.007
δ	0.7	0.153	0.127	-0.159	0.068	0.058	-0.101	0.036	0.033	-0.050

In Figures 10 and 11, it is evident that as the sample size n increases, the algorithm's performance to fit $\beta(t)$ and $g(t)$ improves significantly. When the sample size n increases to 300 and 700, the fitted curves approach the accurate curves more closely, indicating that the accuracy of the estimation improves with larger sample sizes.
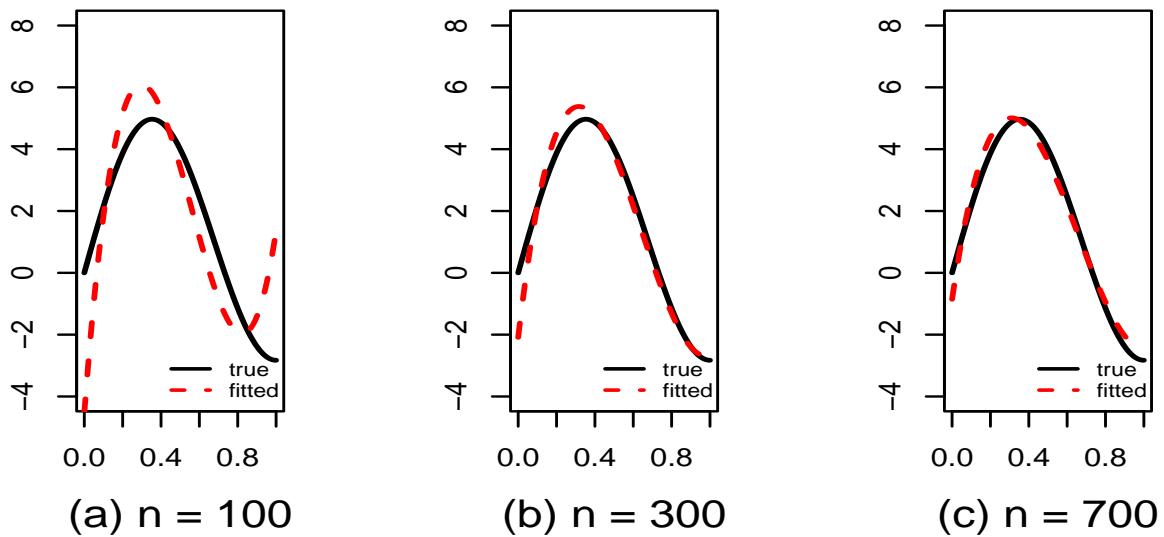
Figure 10. The estimated curves of $\beta(t)$ in Case 5.
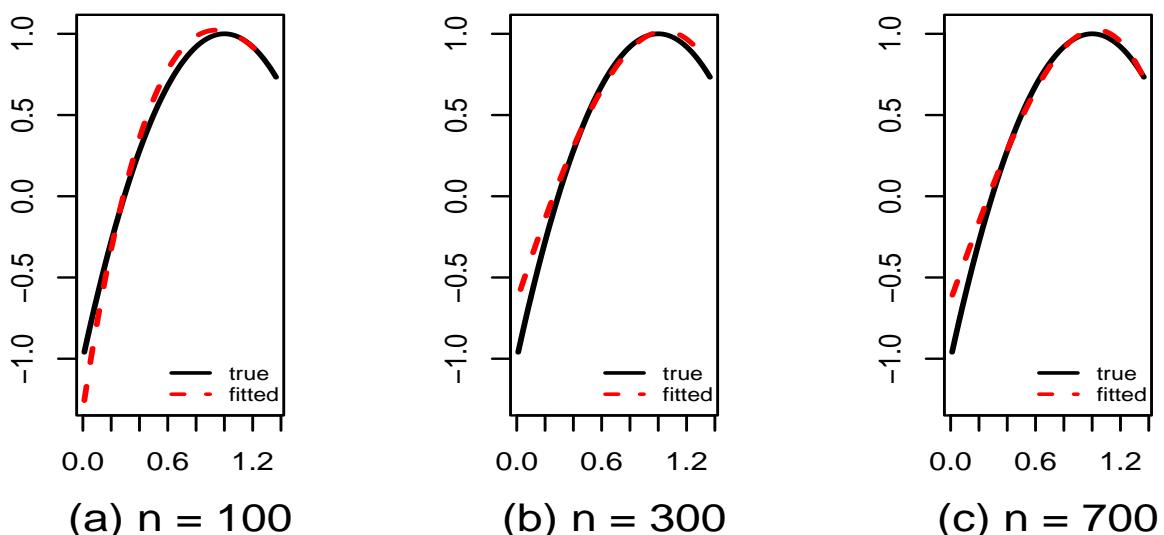
Figure 11. The estimated curves of $g(t)$ in Case 5.

Tables 13 and 14 present the MSE, Var, and bias for the parameters α, ψ, σ^2, and δ, along with the sample mean, median, and variance of RASE_j (for $j = 1, 2$) at different sample sizes. These results demonstrate conclusions consistent with Case 5. Simulation experiments using three distinct parameter configurations in Example 2 indicate that the proposed algorithm demonstrates strong adaptability

in the single-index partially linear model, even with complex autoregressive error structures. As illustrated in Figures 12 and 13, the estimated curves gradually approach the true curve as the sample size increases.

Table 13. Simulation results for nonparametric components of Case 6.

n	Criterion	Mean	Median	Var
100	$RASE_1$	1.538	1.238	1.284
	$RASE_2$	0.319	0.211	0.408
300	$RASE_1$	0.771	0.701	0.232
	$RASE_2$	0.125	0.117	0.003
700	$RASE_1$	0.527	0.467	0.069
	$RASE_2$	0.083	0.080	0.001

Table 14. Simulation results for parametric components of Case 6.

Parameter	True Value	$n = 100$			$n = 300$			$n = 700$		
		MSE	Var	Bias	MSE	Var	Bias	MSE	Var	Bias
α_1	$\frac{\sqrt{3}}{3}$	0.030	0.029	-0.016	0.007	0.007	-0.001	0.003	0.003	0.005
α_2	$\frac{\sqrt{6}}{3}$	0.104	0.099	-0.071	0.003	0.003	-0.006	0.001	0.001	-0.006
ψ	0.5	0.010	0.010	0.003	0.002	0.002	0	0.001	0.001	0
σ^2	0.2	0.019	0.018	-0.035	0.004	0.004	-0.001	0.001	0.001	-0.005
δ	1	0.107	0.104	-0.053	0.015	0.014	-0.028	0.004	0.004	-0.008
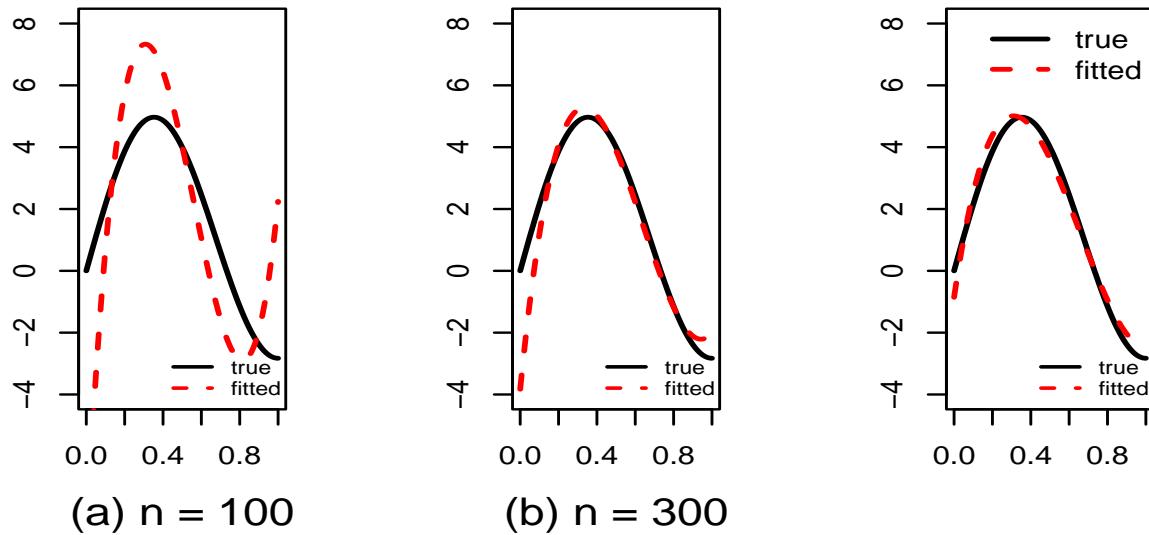
Figure 12. The estimated curves of $\beta(t)$ in Case 6.
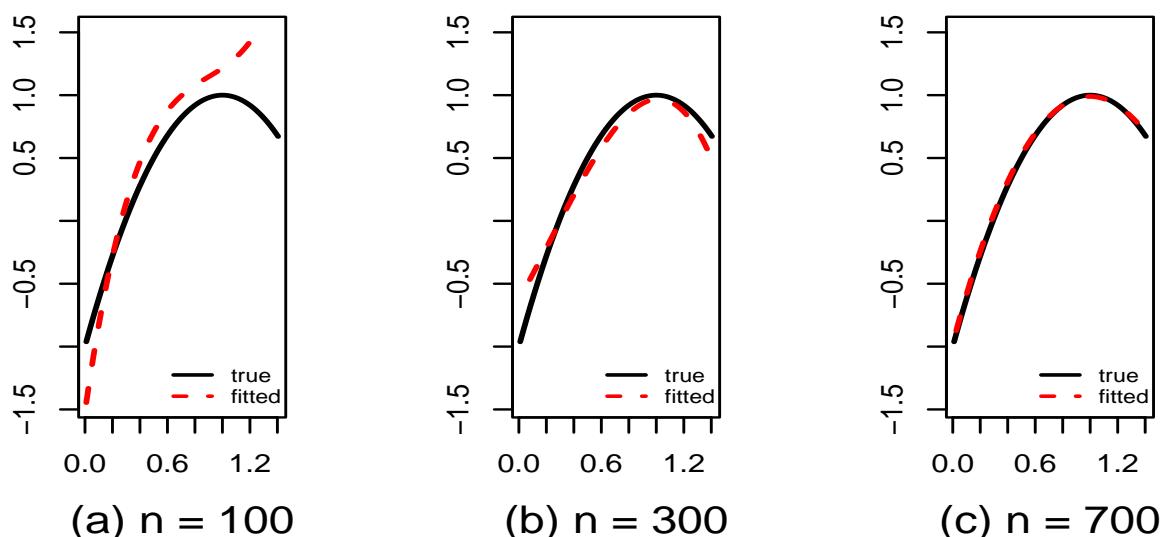
Figure 13. The estimated curves of $g(t)$ in Case 6.

To systematically evaluate the time complexity and efficiency of the algorithm, the average runtime of the EM-CALS algorithm is measured under Cases 4–6 scenarios in Example 2, with sample sizes of $n = 100$, 300, and 700, following the same approach as in Example 1. As shown in Table 15, the experiment execution time does not directly increase or decrease with the increase in sample size. This

is related to the more flexible convergence criteria applied in the EM-CALS algorithm.

Table 15. Experiment Execution Time(s).

n		User	System	Elapsed
100	Case 4	126.892	2.917	194.251
	Case 5	193.034	2.961	199.880
	Caee 6	111.520	2.566	173.009
300	Case 4	75.457	1.741	109.605
	Case 5	168.304	2.418	172.591
	Case 6	32.806	0.786	48.882
700	Case 4	65.384	1.678	68.649
	Case 5	177.791	2.418	172.591
	Case 6	48.716	1.256	56.743

6. Application

With the rapid advancement of photovoltaic (PV) technology, it has gained significant popularity in grid-connected applications. The power output of PV systems is affected by various factors, including solar irradiance, ambient temperature, sunlight intensity, and installation angle. Given the fluctuations in solar irradiance and environmental conditions, the power output of PV systems inherently exhibits temporal variability. Wang, Su, and Shu [32] analyzed grid-connected power generation data from Macau in 2011 using partial functional linear regression under the assumption of independent errors. Xiao and Wang [12] extended this analysis by employing a partial functional linear model with autoregressive errors on the same dataset.

In this study, we analyzed the Macau 2018 PV power generation dataset provided by Qiu [34]. This dataset comprises solar power generation data collected from a 3-kilowatt rooftop PV plant at the University of Macau throughout 2018, with measurements taken at 30-second intervals. Additionally, it includes public weather report data from Macau, with weather variables recorded hourly. The system is located on Coloane Island in the Macau Special Administrative Region (SAR) (latitude 22° 100000'N, longitude 113° 330000'E). The data was recorded from January 1, 2018 to December 31, 2018. Due to various factors, such as meteorological conditions, maintenance, or instrument malfunctions, some data entries were missing. After employing standard preprocessing techniques to eliminate missing values and outliers, we retained 356 days of data. The total output power for the following day was selected as the response variable Y(kW), while the hourly output power from the previous day served as the functional predictor. Several meteorological variables were also considered as multivariate predictors. Specifically, Z_1 represents daily average cloud cover, Z_2 refers to daily precipitation, Z_3 denotes the daily average temperature, and Z_4 refers to total solar radiation . Figure 14 illustrates the behavior of the functional predictor variables $X_i(t)$, where all functional predictors exhibit similar patterns. Before modeling, Figures 15–18 depict the relationships between daily average cloud cover, daily

precipitation, daily average temperature, and daily solar radiation with daily output power, respectively. The fitted curves were derived using the lasso method.

To evaluate model performance, we split the dataset into two subsamples: A training subsample $\{Y_t, Z_t, X_t(u)\}_{t=1}^{300}$ for parameter estimation and a test subsample $\{Y_t, Z_t, X_t(u)\}_{t=301}^{356}$ for validating prediction quality. We quantify forecasting accuracy through two metrics: the MSE and mean relative squared error (MRSE), defined, respectively, as

$$\text{MSE} = \frac{1}{56} \sum_{t=301}^{356} \left(Y_t - \hat{Y}_t\right)^2, \tag{6.1}$$

$$\text{MRSE} = \frac{1}{56} \sum_{t=301}^{356} \frac{\left(Y_t - \hat{Y}_t\right)^2}{\text{Var}(Y)}, \tag{6.2}$$

where $\text{Var}(Y)$ denotes the variance of the response variable over the test set, Y_i represents the true value of the i-th observation, and $\hat{Y}_i$ is the predicted value for the i-th observation from the model.

As shown in Figures 15–18, the daily average cloud cover (Z_1) negatively impacts output power, as seen in Figure 15. Clouds obstruct sunlight, reducing effective radiation reaching PV panels and significantly lowering generation efficiency. Daily precipitation (Z_2) demonstrates a negative nonlinear relationship with output power. As precipitation increases, the power output decreases. Heavier rainfall is typically accompanied by cloudy conditions, which reduce solar radiation. This attenuation of solar radiation, caused by both cloud cover and rain, leads to a decrease in power generation. The daily average temperature (Z_3) has a positive but insignificant effect on output power. While higher temperatures are generally associated with ample sunlight that aids power generation, the efficiency of PV panels actually decreases at elevated temperatures. This results in only a weak positive impact of temperature on output power. In contrast, total solar radiation (Z_4) shows a strong positive correlation with output power. Higher radiation levels enhance photon absorption, leading to a direct increase in power generation, in accordance with the fundamental principles of PV energy conversion.
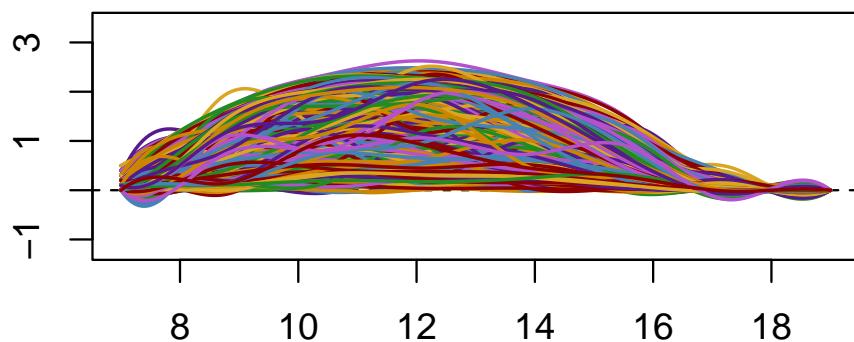

**Figure 14.** The plot of the functional predictor.
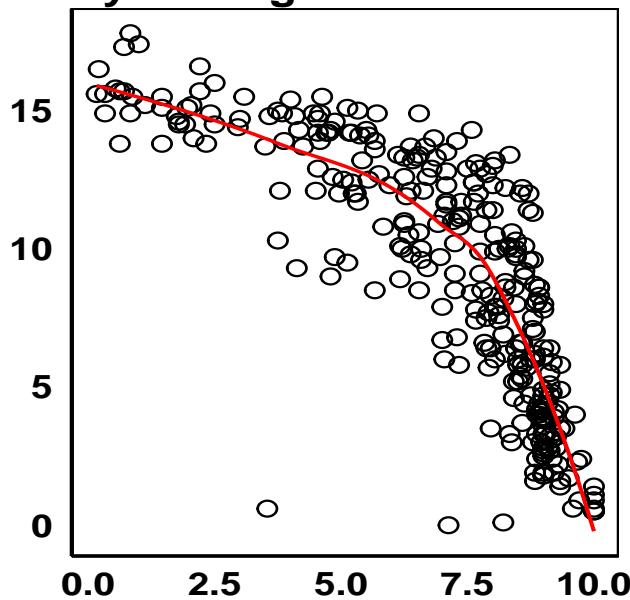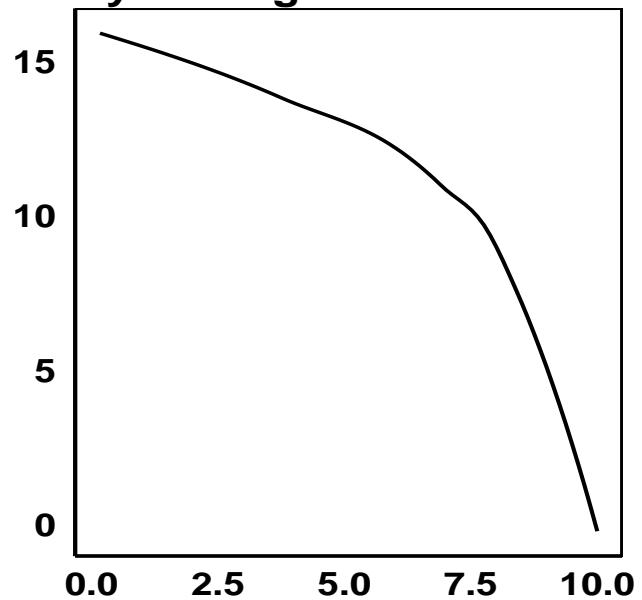
**Figure 15.** Scatter plot and fitted curve of daily average cloud cover and daily output power.
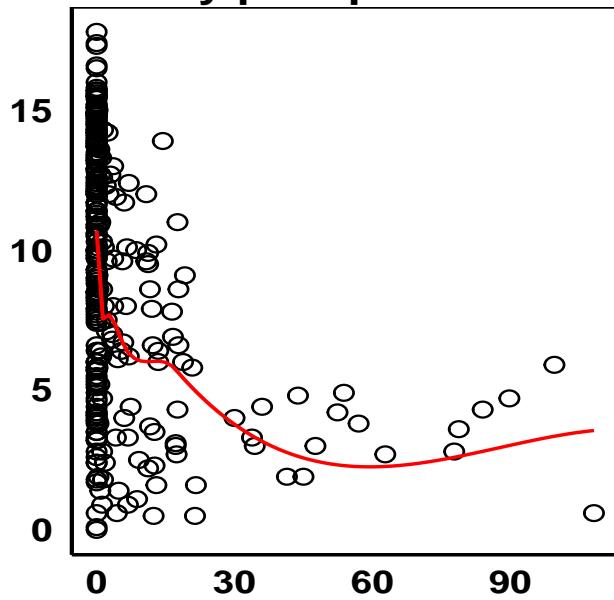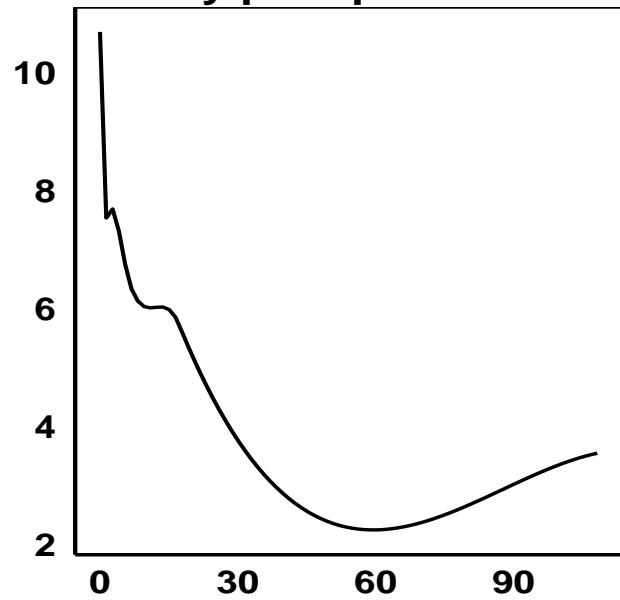
**Figure 16.** Scatter plot and fitted curve of daily precipitation and daily output power.

Daily average temperature

Daily average temperature
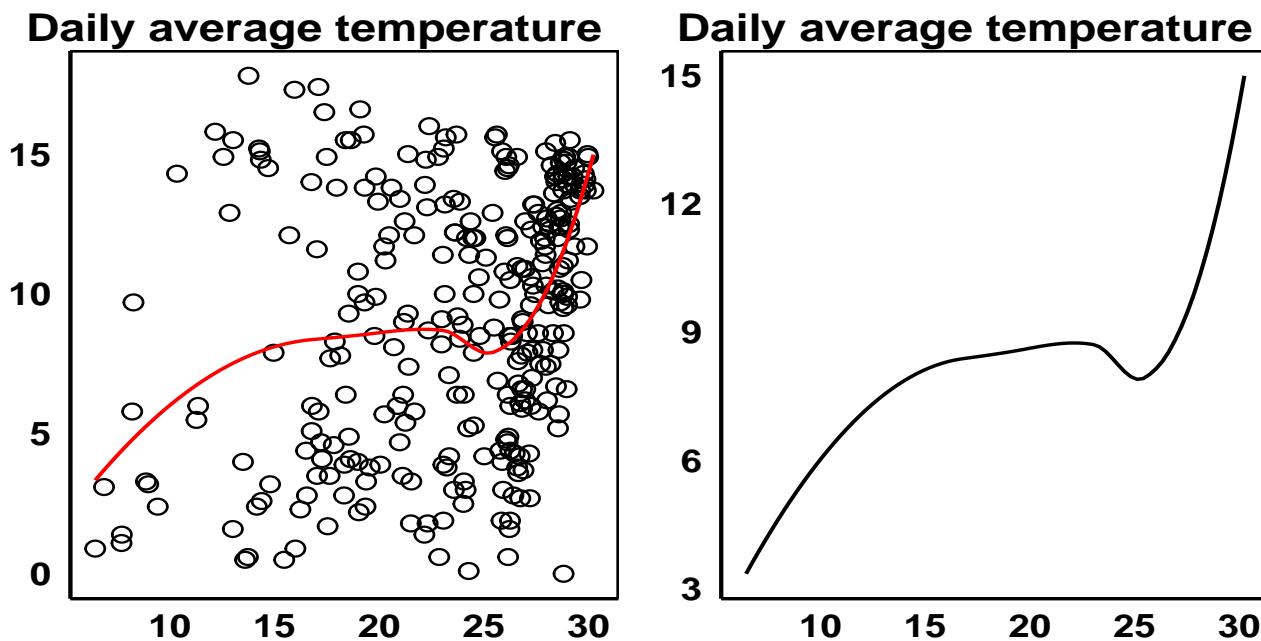
Figure 17. Scatter plot and fitted curve of daily average temperature and daily output power.

Daily Solar Radiation

Daily Solar Radiation
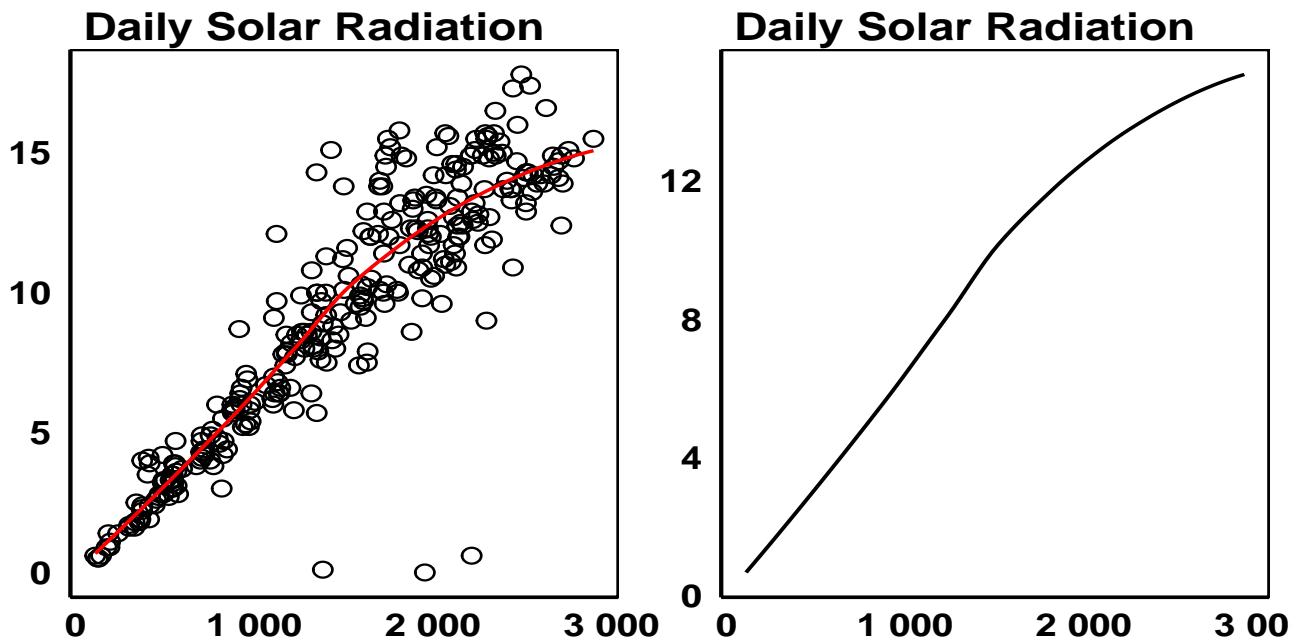
Figure 18. Scatter plot and fitted curve of solar radiation and daily output power.

Figures 15–18 illustrate that the relationships between daily average cloud cover, daily precipitation, daily average temperature, and daily solar radiation with output power are nonlinear. Consequently, traditional linear models are insufficient for accurately capturing these complex interactions. The

single-index model provides the necessary flexibility to account for nonlinear relationships among multiple covariates. By integrating the effects of several covariates into a single index, this model is particularly effective for modeling the nonlinear dynamics between output power and the relevant variables. Therefore, we utilize the single-index partially functional linear regression model for our analysis.

The residual analysis was performed using the single-index partially functional linear regression model introduced by Yu et al. [7]. The residuals were then evaluated with the Shapiro-Wilk test to check for normality. The results revealed a W statistic of 0.77008 and a p-value less than 2.2×10^{-16}, which led to the rejection of the null hypothesis. This indicates that the residuals do not follow a normal distribution. Furthermore, the Q-Q plot shown in Figure 19 suggests that while the residuals approximate a normal distribution in the central region, they exhibit significant deviations in the tails, indicating heavy-tailed characteristics and a departure from normality. Given the observed skewness and heavy tails in the residuals, the SN distribution, as proposed by Azzalini [11], was adopted to more accurately model errors. The SN distribution introduces a skewness parameter, enabling it to effectively capture asymmetry and accommodate data with moderate skewness and kurtosis. As a result, the assumption of normal errors was replaced with the skew-normal error distribution.
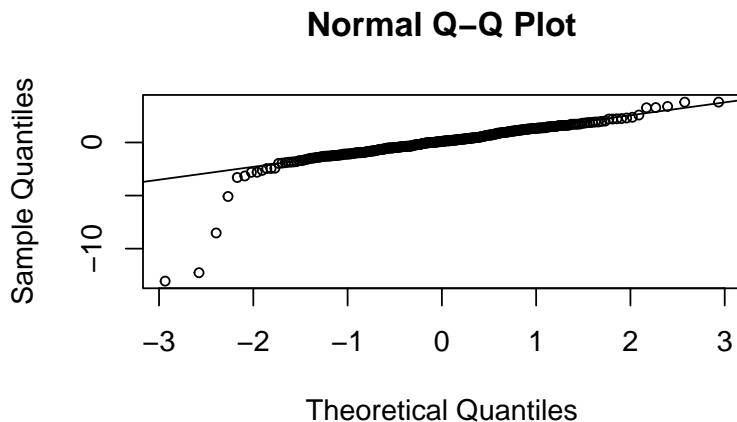

Figure 19. The Q-Q plot of the residuals.

Based on the model and algorithm proposed by Yu et al. [7], the residual sequence is obtained. Subsequently, the Ljung-Box (LB) test statistic, along with the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots, are utilized to test for the presence of 1-order to 5-order serial correlation.

Figures 20 and 21 illustrate the PACF and the ACF of the residuals, respectively. Both functions reveal significant deviations from zero at a lag of 2, indicating that the residuals display an $AR(2)$ structure. As shown in Table 16, when $h = 2$, the p-value of the LB test statistic is the smallest and is less than the given significance level of 0.05, leading to the rejection of the null hypothesis and indicating that the errors follow an AR(2) structure. Considering the $AR(2)$ structure and skewness of the errors, we propose a single-index partially functional linear model with AR(2) skew-normal errors, referred to as SIPFLM-SNAR(2). For comparative analysis, we also created two additional models:

one that includes normal *AR*(2) errors, called SIPFLM-NAR(2), and another that assumes independent errors.

Series r
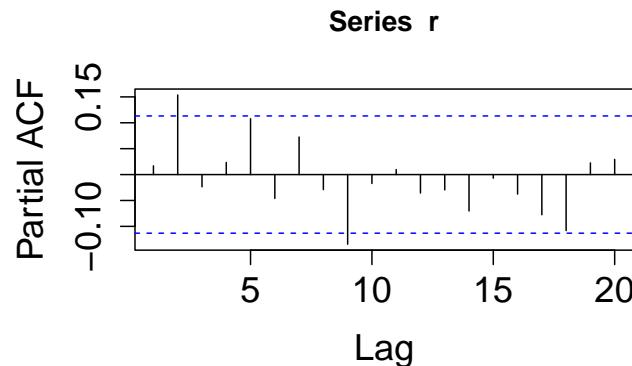

Figure 20. The PACF plot of the residuals.

reps 1
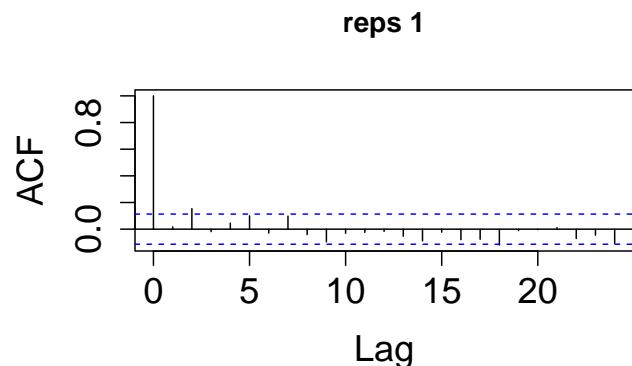
Figure 21. The ACF plot of the residuals.

Table 16. The p-value of the LB test statistic.

	h-order				
	h=1	h=2	h=3	h=4	h=5
p values	0.768	0.026	0.061	0.091	0.050

Table 17 summarizes the estimated values of the relevant parameters and MSE and MRSE. The results indicate that the proposed SIPFLM-SNAR(2) model achieves a lower MSE and MRSE compared to the SIPFLM-NAR(2) and SIPFLM models, demonstrating its superior predictive accuracy.

Table 17. The estimated parameters and the MSE and MRSE for the power output data.

Parameter	Indep. Normal	$AR(2)$-Normal	$AR(2)$-Skew-Normal
α_1	0.001	0.649	0.646
α_2	0.074	0.049	0.051
α_3	-0.996	-0.757	-0.760
α_4	0.048	0.053	0.054
γ_1	-1.095	-0.687	-0.688
γ_2	2.212	1.563	1.567
γ_3	-1.917	-1.490	-1.497
γ_4	0.625	0.573	0.582
η_1	0.548	0.642	0.646
η_2	5.524	5.751	5.741
η_3	14.573	14.609	14.608
η_4	15.032	15.018	15.007
ψ_1	–	0.115	0.114
ψ_2	–	0.194	0.194
σ^2	2.061	2.916	2.958
δ	–	–	0.086
MSE	5.168	3.190	3.161
MRSE	0.314	0.194	0.192
Total Sample Size	356	356	356
Training Set Size	300	300	300
Test Set Size	56	56	56

7. Conclusions and discussion

This paper addresses parameter estimation for the single-index partially functional linear models with p-order autoregressive skew-normal errors. We propose an EM-CALS algorithm to estimate the model's parametric and nonparametric components. The method includes analytical expressions for the E-step and M-step. To handle the nonlinear constraint imposed on the single-index coefficient, specifically the unit norm constraint, we incorporate a CALS step into the algorithm. This modification reduces the parameter estimation complexity and improves the algorithm's stability. To demonstrate the performance advantages of the EM-CALS algorithm, we compare it with the TSILS algorithm. Compared to the TSILS algorithm, the EM-CALS algorithm shows significant improvements in the mean values of RMSE, MSE, RASE$_1$, and RASE$_2$, with decreases of 1.19%, 2.45%, 38%, and 51.35%, respectively, demonstrating its clear advantage in enhancing prediction accuracy. Additionally, we perform a residual analysis based on conditional residuals, following the methodology described by Dunn and Smyth [31].

The proposed model performs local influence analysis using the Q-function in the EM algorithm. The impact of case-weight perturbation, response variable perturbation, and explanatory variable perturbation on the model is examined, and the analytical expression of the Hessian matrix is derived.

The performance of the EM-CALS algorithm is validated through simulation studies that assess

its behavior under various scenarios, including changes in the autoregressive order of the errors and the nonparametric function in the single-index component. The results indicate that while some parameters show fluctuations in bias, their MSE decreases with larger sample sizes. This suggests an improvement in model fitting performance. The proposed estimation method demonstrates good stability and accuracy, particularly when larger sample sizes are used.

Furthermore, the model is applied to a dataset from a PV system for power prediction. The findings reveal that the SIPFLM-SNAR(2) model effectively captures asymmetry and temporal dependence in the response variable, making it highly useful in scenarios involving functional data and multidimensional scalar predictors.

Despite the positive results of this study, several limitations remain. First, the asymptotic properties and convergence rate of the EM-CALS estimators have yet to be proven. Future research could explore the asymptotic normality and convergence speed of this algorithm. Second, the simulations in this study mainly focus on low-dimensional cases, and the performance of the algorithm in high-dimensional single-index models has not been fully examined. As the model's dimensionality increases, ensuring the accuracy of estimations and improving the computational efficiency of the algorithm will present significant challenges. Future research can expand the scope of simulations to further validate the applicability of the algorithm in high-dimensional settings while considering additional potential influencing factors, thereby enhancing the robustness and broader applicability of the algorithm.

Author contributions

Lijie Zhou: Investigation, formal analysis, validation, software, data curation, writing-original draft, writing-review and editing; Liucang Wu: methodology, validation; Bin Yang: Conceptualization, funding acquisition, investigation, resources, validation, data curation, writing-original draft, writing-review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in creating this article.

Acknowledgments

Conflict of interest

The authors declare that there is no conflict of interest.
Prof. Liucang Wu is the Guest Editor of special issue "Advanced in Engineering Statistics,

Technology, and Applications" for AIMS Mathematics. Prof. Liucang Wu was not involved in the editorial review and the decision to publish this article.

References

1. H. Cardot, F. Ferraty, P. Sarda, Spline estimators for the functional linear model, *Stat. Sin.,* **13** (2003), 571–591. Available from: http://www.jstor.org/stable/24307112.

2. D. Şentürk, H. G. Müller, Functional varying coefficient models for longitudinal data, *J. Am. Stat. Assoc.,* **105** (2010), 1256–1264. https://doi.org/10.1198/jasa.2010.tm09228

3. G. Aneiros-Pérez, P. Vieu, Semi-functional partial linear regression, *Stat. Probab. Lett.,* **76** (2006), 1102–1110. https://doi.org/10.1016/j.spl.2005.12.007

4. J. Zhou, M. Chen, Spline estimators for semi-functional linear model, *Stat. Probab. Lett.,* **82** (2012), 505–513. https://doi.org/10.1016/j.spl.2011.11.027

5. W. Härdle, P. Hall, H. Ichimura, Optimal smoothing in single-index Models, *J. Comput. Graph. Stat.,* **21** (1993), 157–178. https://doi.org/10.1214/aos/1176349020

6. Q. M. Zou, Z. Y. Zhu, M-estimators for single-index model using B-spline, *Metrika,* **77** (2014), 225–246. https://doi.org/10.1007/s00184-013-0434-z

7. P. Yu, J. Du, Z. Zhang, Single-index partially functional linear regression model, *Stat. Pap.,* **61** (2020), 1107–1123. https://doi.org/10.1007/s00362-018-0980-6

8. S. Dabo-Niang, S. Guillas, Functional semiparametric partially linear model with autoregressive errors, *J. Multivar. Anal.,* **101** (2010), 307–315. https://doi.org/10.1016/j.jmva.2008.06.008

9. B. Yang, M. Chen, T. Su, J. Zhou, Robust estimation for semi-functional linear model with autoregressive errors, *Mathematics,* **11** (2023), 277. https://doi.org/10.3390/math11020277

10. P. Xiao, G. Wang, Partial functional linear regression with autoregressive errors, *Commun. Stat. Theor.-M.,* **51** (2020), 4515–4536. https://doi.org/10.1080/03610926.2020.1818097

11. A. Azzalini, A class of distributions which includes the normal ones, *Scand. J. Stat.,* **12** (1985), 171–178. Available from: http://www.jstor.org/stable/4615982.

12. J. Xiao, P. Yu, Z. Zhang, Weighted composite asymmetric Huber estimation for partial functional linear models, *AIMS Math.,* **7** (2022), 7657–7684. https://doi.org/10.3934/math.2022430

13. C. da Silva Ferreira, G. A. Paula, G. C. Lana, Estimation and diagnostic for partially linear models with first-order autoregressive skew-normal errors, *Comput. Stat.,* **37** (2022), 445–468. https://doi.org/10.1007/s00180-021-01130-2

14. Y. Liu, J. Lu, G. A. Paula, S. Liu, Bayesian diagnostics in a partially linear model with first-order autoregressive skew-normal errors, *Comput. Stat.,* **46** (2024), 1–31. https://doi.org/10.1007/s00180-024-01504-2

15. C. da Silva Ferreira, M. H. Montoril, G. A. Paula, Partially linear models with p-order autoregressive skew-normal errors, *Braz. J. Probab. Stat.,* **36** (2022), 792–806. https://doi.org/10.1214/22-bjps556

16. L. Z. Chen, G. C. Wang, B. X. Zhang, Functional partial linear regression with autoregressive errors, *Commun. Math. Stat.,* 2025, 225–246. https://doi.org/10.1007/s40304-024-00409-2

17. M. Wang, M. L. Shu, J. J. Zhou, S. X. Wu, M. Chen, Least square estimation for multiple functional linear model with autoregressive errors, *Acta Math. Appl. Sin. Engl. Ser.,* **41** (2025), 84–98. https://doi.org/10.1007/s10255-024-1143-2

18. R. D. Cook, Assessment of local influence, *J. R. Stat. Soc. Ser. B,* **48** (1986), 133–155. https://doi.org/10.1111/j.2517-6161.1986.tb01398.x

19. Q. M. Zou, Z. Y. Zhu, J. L. Wang, Local influence analysis for penalized Gaussian likelihood estimation in partially linear single-index models, *Ann. Inst. Stat. Math.,* **61** (2009), 905–918. https://doi.org/10.1007/s10463-007-0158-9

20. S. K. Sahu, D. K. Dey, M. D. Branco, A new class of multivariate skew distributions with applications to bayesian regression models, *Can. J. Stat.,* **31** (2003), 129–150. https://doi.org/10.2307/3316064

21. C. de Boor, *A practical guide to splines*, New York: Springer, 2001. https://doi.org/10.2307/2006241

22. J. Huang, J. L. Horowitz, F. Wei, Variable selection in nonparametric additive models, *Ann. Stat.,* **38** (2010), 2282–2313. https://doi.org/10.1214/09-aos781

23. A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, *J. Roy. Stat. Soc. B,* **39** (1977), 1–38. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

24. N. L. Johnson, A. W. Kemp, S. Kotz, *Univariate discrete distributions*, New York: John Wiley, 2005. https://doi.org/10.1002/0471715816

25. K. Takai, Constrained EM algorithm with projection method, *Comput. Stat.,* **27** (2012), 701–714. https://doi.org/10.1007/s00180-011-0285-x

26. Y. Yu, D. Ruppert, Root-n consistency of penalized spline estimator for partially linear single-index models under general Euclidean space, *Stat. Sin.,* **14** (2004), 449–456. Available from: https://api.semanticscholar.org/CorpusID:14712886.

27. B. Yang, M. Chen, J. J. Zhou, Forecasting the monthly retail sales of electricity based on the semi-functional linear model with autoregressive errors, *AIMS Math.,* **10** (2025), 1602–1627. https://doi.org/10.3934/math.2025074

28. Q. Y. Peng, J. J. Zhou, N. S. Tang, Varying coefficient partially functional linear regression models, *Stat. Pap.,* **57** (2015), 827–841. https://doi.org/10.1007/s00362-015-0681-3

29. H. Lin, J. Shi, T. Tong, R. Zhang, Partially linear single-index model in the presence of measurement error, *J. Syst. Sci. Complex.,* **35** (2022), 2361–2380. https://doi.org/10.1007/s11424-022-1112-x

30. P. Yu, Z. Zhang, J. Du, A test of linearity in partial functional linear regression, *Metrika,* **79** (2016), 953–969. https://doi.org/10.1007/s00184-016-0584-x

31. P. K. Dunn, G. K. Smyth, Randomized quantile residuals, *J. Comput. Graph. Stat.,* **5** (1996), 236–244. https://doi.org/10.2307/1390802

32. G. Wang, Y. Su, L. Shu, One-day-ahead daily power forecasting of photovoltaic systems based on partial functional linear regression models, *Renew. Energy,* **96** (2016), 469–478. https://doi.org/10.1016/j.renene.2016.04.089

33. H. T. Zhu, S. Lee, Local influence for incomplete data models, *J. R. Stat. Soc. Ser. B*, **63** (2001), 111–126. https://doi.org/10.1111/1467-9868.00279

34. Y. Qiu, Z. Zhou, *Macao meteorological and geophysical Bureau*, In: 2018-2019 PV Generation of a Rooftop Plant in the University of Macau and Weather Report, IEEE Dataport, 2020. https://dx.doi.org/10.21227/1khg-8t55