Hindawi
Complexity
Volume 2020, Article ID 8941652, 13 pages
https://doi.org/10.1155/2020/8941652

Research Article

A Penalized h-Likelihood Variable Selection Algorithm for Generalized Linear Regression Models with Random Effects

Yanxi Xie, Yuewen Li, Zhijie Xia, Ruixia Yan, and Dongqing Luan

School of Management, Shanghai University of Engineering Science, Shanghai 201620, China

Correspondence should be addressed to Yuewen Li; sues0305@126.com

Received 31 July 2020; Revised 28 August 2020; Accepted 6 September 2020; Published 15 September 2020

Academic Editor: Shuping He

Copyright © 2020 Yanxi Xie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Reinforcement learning is one of the paradigms and methodologies of machine learning developed in the computational intelligence community. Reinforcement learning algorithms present a major challenge in complex dynamics recently. In the perspective of variable selection, we often come across situations where too many variables are included in the full model at the initial stage of modeling. Due to a high-dimensional and intractable integral of longitudinal data, likelihood inference is computationally challenging. It can be computationally difficult such as very slow convergence or even nonconvergence, for the computationally intensive methods. Recently, hierarchical likelihood (h-likelihood) plays an important role in inferences for models having unobservable or unobserved random variables. This paper focuses linear models with random effects in the mean structure and proposes a penalized h-likelihood algorithm which incorporates variable selection procedures in the setting of mean modeling via h-likelihood. The penalized h-likelihood method avoids the messy integration for the random effects and is computationally efficient. Furthermore, it demonstrates good performance in relevant-variable selection. Throughout theoretical analysis and simulations, it is confirmed that the penalized h-likelihood algorithm produces good fixed effect estimation results and can identify zero regression coefficients in modeling the mean structure.

1. Introduction

Reinforcement learning is specified as trial and error (variation and selection and search) plus learning (association and memory) in Sutton and Barto [1]. Traditional variable selection procedures, such as LASSO in Tibshirani [2] and OMP in Cai and Wang [3], only consider the fixed effect estimates in the linear models in the past literature. However, in real life, a lot of existing data have both the fixed effects and random effects involved. For example, in the clinic trials, several observations are taken for a period of time for one particular patient. After collecting the data needed for all the patients, it is natural to consider random effects for each individual patient in the model setting since a common error term for all the observations is not sufficient to capture the individual randomness. Moreover, random effects, which are not directly observable, are of interest in themselves if inference is focused on each individual's response. Therefore, to solve the problem of the random effects and to get good estimates, Lee and Nelder [4] proposed hierarchical generalized linear models (HGLMs). HGLMs are based on the idea of h-likelihood, a generalization of the classical likelihood to accommodate the random components coming through the model. It is preferable because it avoids the integration part for the marginal likelihood and uses the conditional distribution instead.

Inspired by the idea of reinforcement learning and hierarchical models, this paper proposes a method by adding a penalty term to the h-likelihood. This method considers not only the fixed effects but also the random effects in the linear model, and it produces good estimation results with the ability to identify zero regression coefficients in joint models of mean-covariance structures for high-dimensional multilevel data.

The rest of this paper is organized as follows: Section 2 provides the literature review on current variable selection methods based on partial linear models and h-likelihood. Section 3 explains a penalty-based h-likelihood variable selection algorithm and demonstrates via simulation that our proposed algorithm exhibits desired sample properties and can be useful in practical applications. Finally, Section 4 concludes the paper, and some future research directions are given.

2. Literature Review

2.1. Reinforcement Learning in the Perspective of Nonlinear Systems. Reinforcement learning, one of the most active research areas in artificial intelligence, is introduced and defined as a computational approach to learning whereby an agent tries to maximize the total amount of reward it receives when interacting with a complex, uncertain environment in Sutton and Barto [1]. In addition, in the paper of Sutton and Barto [5], reinforcement learning is specified to be trial and error (variation and selection and search) plus learning (association and memory). Furthermore, Barto and Mahadevan [6] propose hierarchical control architectures and associated learning algorithms. Approaches to temporal abstraction and hierarchical organization, which mainly rely on the theory of semi-Markov decision processes, are reviewed and discussed in Barto and Mahadevan's paper [6]. Recent works, such as Dietterich [7], have focused on the hierarchical methods that incorporate subroutines and state abstractions, instead of solving "flat" problem spaces.

Nonlinear control design has gained a lot of attention in the research area for a long time. In the industrial field, the controlled system usually has great nonlinearity. Various adaptive optimal control models have been applied to the identification of nonlinear systems in the past literature. In fact, the two important fundamental principles of controller design are optimality and veracity. He et al. [8] study a novel policy iterative scheme for the design of online H_∞ optimal laws for a class of nonlinear systems and establishes the convergence of the novel policy iterative scheme to the optimal control law. He et al. [9] investigate an online adaptive optimal control problem of a class of continuous-time Markov jump linear systems (MJLSs) by using a parallel reinforcement learning (RL) algorithm with completely unknown dynamics. A novel parallel RL algorithm is proposed, and the convergence of the proposed algorithm is shown. Wang et al. [10] study a new online adaptive optimal controller design scheme for a class of nonlinear systems with input time delays. An online policy iteration algorithm is proposed, and the effectiveness of the proposed method is verified. He et al. [11] propose the online adaptive optimal controller design for a class of nonlinear systems through a novel policy iteration (PI) algorithm. Cheng et al. [12] investigate the observer-based asynchronous fault detection problem for a class of nonlinear Markov jumping systems and introduces a hidden Markov model to ensure that the observer modes run synchronously with the system modes. Cheng et al. [13] propose the finite-time asynchronous output feedback control scheme for a class of Markov jump systems subject to external disturbances and nonlinearities.

2.2. Partial Linear Models. Linear models have been widely used and employed in the literature. One extension of linear models, which was introduced by Nelder and Wedderburn [14], is generalized linear models (GLMs). GLMs allow the class of distributions to be expanded from the normal distribution to that of one-parameter exponential families. In addition, GLMs generalize linear regression in the following two manners: first of all, GLMs allow the linear model to be related to the response variable via a link function, or equivalently a monotonic transform of the mean, rather than the mean itself. Second, GLMs allow the magnitude of the variance of each measurement to be a function of its predicted value.

On the contrary, Laird and Ware [15] propose linear mixed effect models (LMEs), which are widely used in the analysis of longitudinal and repeated measurement data. Linear mixed effect models have gained popular attention since they take into consideration within-cluster and between-cluster variations simultaneously. Vonesh and Chinchilli [16] have investigated and applied statistical estimation as well as inference for this class of LME models. However, it seems that model selection problem in LME models is ignored. This disregarded problem was noticed and pointed out by Vaida and Blanchard [17], stating that when the focus is on clusters instead of population, the traditional selection criteria such as AIC and BIC are not appropriate. In the paper of Vaida and Blanchard [17], the conditional AIC is proposed, for mixed effects models with detailed discussion on how to define degrees of freedom in the presence of random effects. Furthermore, Pu and Niu [18] study the asymptotic behavior of the proposed generalized information criterion method for selecting fixed effects. In addition, Rajaram and Castellani [19] use ordinary differential equations and the linear advection partial differential equations (PDEs) and introduce a case-based density approach to modeling big data longitudinally.

Recently, Fan and Li [20] develop a class of variable selection procedures for both fixed effects and random effects in linear mixed effect models by incorporating the penalized profile likelihood method. By this regularization method, both fixed effects and random effects can be selected and estimated. There are two outstanding aspects regarding Fan and Li's [20] method. First of all, the proposed procedures can estimate the fixed effects and random effects in a separate way. Or in other words, the fixed effects can be estimated without the random effects being estimated, and vice versa. In addition, the method works in the high-dimensional setting by allowing dimension of random effect to grow exponentially with sample size.

Combined with the idea of generalized linear models (GLMs) and linear mixed effect (LME) models, one extension, generalized linear mixed models (GLMMs), is developed. In the traditional GLMs, it is assumed that the observations are uncorrelated. To solve the constrained assumption, GLMMs allow for correlation between observations, which often happens in the longitudinal data and clustered designs. The advantages of GLMMs are presented as follows: first of all, GLMMs allow random effects to be included in the linear predictor. As a result, the correlations between observations can be explained through an explicit probability model. Second, when the focus is on estimating the fixed effects on a particular individual, GLMMs provide good subject-specific parameter estimates. However, since GLMMs are also called multilevel models, it is generally more computationally intensive when fitting the model.

So far, all those GLMs and GLMMs are well-established parametric regression models. A serious disadvantage of parametric modeling is that a parametric model may be too restrictive in some applications. To overcome this restrictive assumption difficulty in the parametric regression, nonparametric regression has gained popular attention in the literature. There are many nonparametric and smoothing methods, such as kernel smoothing, local polynomial fitting, and penalized splines. In this section, two often-used smoothing methods in estimating a nonparametric model are described in the following paragraphs since they are used later in simulations and applications.

The first type is called local linear kernel smoothing. The main idea of local linear kernel smoothing is to locally approximate the function f linearly. Local linear kernel smoothing uses Taylor expansion as a fundamental tool. In particular, Taylor expansion states that any smooth function can be locally approximated by a polynomial of some degree.

Suppose we have a simple nonparametric model

$$y_i = f(t_i) + \varepsilon_i, \tag{1}$$

for $i = 1, \ldots, n$. Let t_0 be an arbitrary fixed point where the function f is estimated. Assume $f(t)$ has a first-order continuous derivative at t_0. Then, by Taylor expansion, $f(t)$ can be locally approximated by

$$f(t) \approx f(t_0) + (t - t_0) f^{(1)}(t_0), \tag{2}$$

in a neighborhood of t_0 that allows the above expansion where $f^{(1)}(t_0)$ denotes the first derivative of $f(t)$ at t_0.

Let $\alpha_0 = f(t_0)$ and $\alpha_1 = f^{(1)}(t_0)$. The local linear smoother is obtained by fitting a data set locally with a linear function, to minimize the following weighted least squares criterion:

$$\sum_{i=1}^{n} [y_i - \alpha_0 - \alpha_1 (t - t_0)]^2 K_h(t_i - t_0), \tag{3}$$

where $K_h(.) = K(./h)/h$, which is obtained by rescaling a kernel function $K(.)$ with a positive constant bandwidth h. The primary objective of the bandwidth h is to specify the size of the local neighborhood $[t_0 - h, t_0 + h]$, where the local fitting is conducted. Moreover, the kernel function $K(.)$ determines how observations within the neighborhood contribute to the fit at t_0. A detailed introduction of the kernel function will be provided in the later paragraphs.

The local linear smoother $\widehat{f_h(t_0)} = \widehat{\alpha_0}$ can be simply expressed as

$$\widehat{f_h(t_0)} = \frac{\sum_{i=1}^{n} [s_2(t_0) - s_1(t_0)(t - t_0)] K_h(t_i - t_0) y_i}{s_2(t_0) s_0(t_0) - s_1^2(t_0)}, \tag{4}$$

where

$$\begin{aligned} s_0(t_0) &= \sum_{i=1}^{n} K_h(t_i - t_0), \\ s_1(t_0) &= \sum_{i=1}^{n} K_h(t_i - t_0)(t_i - t_0), \\ s_2(t_0) &= \sum_{i=1}^{n} K_h(t_i - t_0)(t_i - t_0)^2. \end{aligned} \tag{5}$$

A local linear smoother is often good enough for most problems if the kernel function $K(.)$ and the bandwidth h are adequately determined. Moreover, it enjoys many good properties that the other linear smoothers may lack. Fan [21], Fan and Gijbels [22], and Hastie and Loader [23] separately discussed those good properties in detail.

The kernel function $K(.)$ used in the local linear smoother is a symmetric probability density function. The kernel $K(.)$ specifies how the observations contribute to the local linear kernel fit at t_0, whereas the bandwidth h specifies the size of the local neighborhood $[t_0 - h, t_0 + h]$. Several widely used kernel functions include the following:

(i) Uniform $K(u) = (1/2)\mathbf{I}_{\{|u| \le 1\}}$

(ii) Epanechnikov $K(u) = (3/4)(1 - u^2)\mathbf{I}_{\{|u| \le 1\}}$

(iii) Biweight $K(u) = (15/16)(1 - u^2)^2\mathbf{I}_{\{|u| \le 1\}}$

(iv) Gaussian $K(u) = (1/\sqrt{2\phi})e^{-(1/2)u^2}$

Suppose, for instance, the uniform kernel is used. All the t_i's within the neighborhood $[t_0 - h, t_0 + h]$ contribute equally; or equivalently, the weights are the same, in the local linear kernel fit at t_0; on the contrary, all the t_i's outside the neighborhood $[t_0 - h, t_0 + h]$ contribute nothing. Suppose, for another example, the Gaussian kernel is used. The contribution of the t_i's is determined by the distance of t_i from t_0. In other words, smaller distance $(t - t_0)$ results in larger contribution since the Gaussian kernel is a bell-shaped curve, which peaks at the origin.

The second type of smoothing is called regression spline smoothing. In local linear kernel smoothing introduced above, local neighborhoods were defined by a bandwidth h and a fixed point t_0. On the contrary, in regression spline smoothing that will be introduced shortly, local neighborhoods are defined by a group of locations, known as knots, for example,

$$\tau_0, \tau_1, \ldots, \tau_K, \tau_{K+1}, \tag{6}$$

in an interval $[a, b]$, where $a = \tau_0 < \tau_1 < \cdots < \tau_k < \tau_{k+1} = b$. Moreover, τ_i, $i = 1, 2, \ldots, k$ are referred as interior knots or simple knots. Then, local neighborhoods are divided by these knots, i.e.,

$$[\tau_i, \tau_{i+1}), \quad i = 0, 1, \ldots, k, \tag{7}$$

and within any two neighboring knots, a Taylor's expansion up to some degree is applicable.

A regression spline can be constructed in terms of truncated power basis. As mentioned earlier, there are K knots $\tau_1, \ldots, \tau_K$, and the k-th degree truncated power basis can be expressed as

$$1, t, \ldots, t^k, (t-\tau_1)_+^k, \ldots, (t-\tau_K)_+^k, \quad (8)$$

where a_+^k denotes power k of the positive part of a with $a_+ = \max(0, a)$. In most of the literature, it is called "constant, linear, quadratic, and cubic" truncated power basis when $k = 0$, 1, 2, and 3 correspondingly. For the purpose of this chapter, cubic truncated power basis is used in subsequent sections of simulations and applications.

We still consider the abovementioned simple nonparametric model:

$$y_i = f(t_i) + \varepsilon_i, \quad (9)$$

for $i = 1, \ldots, n$. It is with conventional purpose to denote the truncated basis as

$$\Phi_p(t) = \left[1, t, \ldots, t^k, (t-\tau_1)_+^k, \ldots, (t-\tau_K)_+^k\right]^T, \quad (10)$$

where $p = K + k + 1$ is the number of the basis functions involved. Then, the regression fit of the function $f(t)$ in the nonparametric model can be expressed as

$$\widehat{f}_p(t) = \Phi_p(t)^T \left(X^T X\right)^{-1} X^T y, \quad (11)$$

where $y = (y_1, \ldots, y_n)^T$ and $X = (\Phi_p(t_1), \ldots, \Phi_p(t_n))^T$.

To sum up, parametric models are very useful for longitudinal data analysis since they provide a clear and easy description of the relationship between the response variable and its covariates. However, in most of data analysis, the parametric model does not fit the data well, resulting in biased estimates. To overcome the restricted assumptions on parametric forms, various nonparametric models such as nonparametric mixed effects models have been proposed for longitudinal data. Refer, for example, the study by Fan and Zhang [24] and Wu and Rice [25] among others. There is always a trade-off model assumption and model complexity. Parametric models are less robust against model assumptions, but they are efficient when the models are corrected assigned. On the contrary, nonparametric models are more robust against model assumptions, but they are less efficient and more complex. A trade-off between efficiency and complexity by the information measure is fully investigated and discussed in Caves and Schack [26]. Zhang et al. [27] propose an improved K-means clustering algorithm, which is called the covering K-means algorithm (C-K-means). There are two advantages for the C-K-means algorithm. First of all, it acquires efficient and accurate clustering results under both sequential and parallel conditions. Furthermore, it self-adaptively provides a reasonable number of clusters based on the data features.

Semiparametric models come across in the need to compromise and remain good features of both parametric and nonparametric models. In semiparametric models, parametric component and nonparametric component are the two essential components. More specifically, the parametric component is often used to model important factors that affect the responses parametrically, whereas the nonparametric component is often used for less important and nuisance factors. Various semiparametric models for longitudinal data include semiparametric population mean models proposed in Martinussen and Scheike [28] and Xu [29], among others, and semiparametric mixed effects models in the study by Zeger and Diggle [30], Groll and Tutz [31], and Heckman et al. [32]. For the purpose of this paper, we restrict our attention to partially linear regression models.

2.3. h-Likelihood.

In longitudinal studies, there are two types of models, marginal models, and conditional models. By definition, marginal models are usually referred as population-average models by ignoring the cluster random effects. In contrast, conditional models have random effect or are subject-specific models. The main difference between marginal and conditional models is whether the regression coefficients describe an individual's response or the marginal response to changing covariates. Or in other words, changing covariates does not attempt to control for unobserved subjects' random effects. Diggle et al. [33] suggested the random effect model for inferences about individual responses and the marginal model for inferences about margins.

The idea of h-likelihood was introduced by Lee and Nelder [4]. h-likelihood is an extension of Fisher likelihood to models of GLMs with additional random effects in the linear predictor. The concept of h-likelihood is for inferences of unobserved random variables. In fact, h-likelihood is a special kind of extended likelihood, where the random effect parameter is specified to satisfy certain conditions as we shall talk more in details later. In the meantime, with the idea of h-likelihood, hierarchical generalized linear models (HGLMs) were introduced as well in Lee and Nelder's [4] paper. This class of hierarchical GLMs allows various distributions of the random component. In addition, these distributions are conjugate to the distributions of the response y. Four conjugate HGLMs were introduced in [4], namely, normal-normal, Poisson-gamma, binomial-beta, and gamma-inverse gamma (Table 1). If we let y be the response and u be the unobserved random component, v is the scale on which the random effect u happens linearly in the linear predictor. In other words, u and v are linked via some strictly monotonic function.

Consider the hierarchical model where $y|v$ and v follow some arbitrary distributions listed in Table 1. The definition of h-likelihood, denoted by l_h, is presented in the following way:

$$l_h = l(\beta, \phi; y \mid v) + l(\alpha; v), \quad (12)$$

where $l(\alpha; v)$ is the log likelihood function of v given parameter α and $l(\beta, \phi; y \mid v)$ is that of $y|v$ given parameter β and ϕ. One point to note is that the h-likelihood is not a traditionally defined likelihood since v are not directly observable. In the traditional standard maximum likelihood estimation for models with random effects, the method is based on the marginal likelihood as the objective function. In this marginal likelihood approach, random effects v are integrated out and what remain in the maximized function are the fixed effects β and dispersion parameter ϕ. There are

Table 1: Conjugate HGLMs.

$y\|u$	u	Link
Normal	Normal	Identity
Poisson	Gamma	Log
Binomial	Beta	Logit
Gamma	Inverse gamma	Log

two disadvantages of the marginal likelihood approach. First of all, the intractable integration of v is with obvious difficulty. In addition, random effects are nonestimable after integration. In contrast, the h-likelihood approach avoids such intractable integration. In fact, as clearly stated by Lee and Nelder [4], "we can treat the h-likelihood as if it were an orthodox likelihood for the fixed effects β and random effects v, where the v are regarded as fixed parameters for realized but unobservable values of the random effects." Furthermore, the h-likelihood allows us to have a fixed effect estimator that is asymptotically efficient as the marginal maximum likelihood estimator. Last but not least, the maximized h-likelihood estimates are derived by solving the two equations simultaneously:

$$\begin{gathered}\frac{\partial l_h}{\partial \beta} = 0;\\ \frac{\partial l_h}{\partial v} = 0.\end{gathered} \tag{13}$$

People always expect an outstanding property of likelihood inference to be invariant with respect to transformations. As for maximum h-likelihood estimates, estimates for random effects are invariant with respect to the transformation of the random components of u.

Furthermore, Lee and Nelder [4] mentioned adjusted profile h-likelihood, which is defined in the following way:

$$l(\beta) \approx \left[l_h - \frac{1}{2} \log \det \left\{ \frac{D(l_h)}{2\pi} \right\} \right]_{v=\widehat{v}}, \tag{14}$$

where $D(l_h) = -\partial^2 l_h / \partial v \, \partial v^T$. It eliminates the nuisance effects v from the h-likelihood. Moreover, the $D(l_h)$ part is often referred as the adjusted term for such elimination. In fact, this adjusted profile h-likelihood, which is used for the estimation of dispersion components, acts as an approximation of the marginal likelihood, without integrating v out.

There are a few outstanding contributions in Lee and Nelder's [4] publication. First of all, it widens the choice of random effect distributions in mixed generalized linear models. In addition, it brings about the h-likelihood as a device for estimation and prediction in hierarchical generalized linear models. Compared to the traditional marginal likelihood, the h-likelihood avoids the messy integration for the random effects and hence is convenient to use. Furthermore, maximized h-likelihood estimates are obtained by iteratively solving equation (14). To conclude, the h-likelihood is used for inference about the fixed and random effects given dispersion parameter ϕ.

On the contrary, Lee and Nelder [34] demonstrated the use of an adjusted profile h-likelihood for inference about the dispersion components given fixed and random effects. In this paper, the focus is on the joint modeling of the mean and dispersion structure. Iterative weighted least squares (IWLS) algorithm is used for estimations of both the fixed and random effects by the extended likelihood and dispersion parameters by the adjusted profile likelihood. Later, in [35], the algorithm was adjusted by replacing the extended likelihood to the first-order adjusted profile likelihood, as to estimate fixed effects in the mean structure.

Lee and Nelder [36] proposed a class of double hierarchical generalized linear models in which random effects can be specified for both the mean and dispersion. Compared with HGLMs, double hierarchical generalized linear models allow heavy-tailed distributions to be present in the model. Random effects are introduced in the dispersion model to solve heteroscedasticity between clusters. Then, h-likelihood is applied for statistical references and efficient algorithm, as the synthesis of the inferential tool. In addition, Lee and Noh [37] proposed a class of double hierarchical generalized linear models in which random effects can be specified for both the mean and dispersion, allowing models with heavy-tailed distributions and providing robust estimation against outliers. Greenlaw and Kantabutra [38] address the parallel complexity of hierarchical clustering. Instead of the traditional sequential algorithms, the described top-down algorithm in Greenlaw and Kantabutra [38] is parallelized and the computational cost of the top-down algorithm is with $O(\log n)$ time.

In conclusion, for both hierarchical generalized linear models (HGLMs) and double hierarchical generalized linear models (DHGLMs), h-likelihood plays an important role in inferences for models having unobservable or unobserved random variables. Furthermore, numerical studies have been investigated and shown that h-likelihood gives statistically efficient estimates for HGLMs as well as DHGLMs. In addition, Noh and Lee [39] have shown that the h-likelihood procedure outperforms existing methods, including MCMC-type methods, in terms of bias. Last but not least, compared to the traditional marginal likelihood, the h-likelihood avoids the messy integration for the random effects and hence is convenient to use. Therefore, the h-likelihood method is worth attention.

3. Variable Selection via Penalized h-Likelihood

3.1. Model Setup. Suppose that we have k independent groups and each group contains m subjects. Let y_{ij} be the j^{th} subject of group i, where $i = 1, \ldots, k$ and $j = 1, \ldots, m$. Based on the idea of modeling the mean structure in the HGLM framework, we consider a partial linear model for modeling the conditional mean:

$$g(\mu_{ij}) = f(t_{ij}) + x_{ij}^T \beta + v_i, \tag{15}$$

where $f(.)$ is an unknown smooth function in t, t_{ij} is an univariate explanatory variable in $[0, 1]$ for simplicity, $g(.)$ is the canonical link function for the conditional distribution of y_{ij}, and x_{ij} is a $p \times 1$ covariate vector with β as the associated coefficients. In matrix representation,

$$y = f(t) + X\beta + Zv + \varepsilon. \tag{16}$$

We assume that conditional random variables u_i and y_{ij} are from an exponential family with mean and variance:

$$\begin{aligned} E\left(y_{ij} \middle| u_i\right) &= \mu_{ij}, \\ V\left(y_{ij} \middle| u_i\right) &= \phi V\left(\mu_{ij}\right). \end{aligned} \tag{17}$$

We also assume that $(X^T, t)^T$ and ε are independent. The random effects presented in the mean model v_i are linked to u_i via the relationship $v_i = v(u_i)$, where $u_i \sim \mathbf{N}(0, \sigma_u^2)$. This allows for the definition of h-likelihood given in Lee and Nelder [4]. In this paper, the identity link $v_i = u_i$ is used, and hence, this canonical scale corresponds to the case that the conditional distribution of the response y is normal, i.e., $y_{ij} \sim \mathbf{N}(\mu_{ij}, \phi)$.

For simplicity, random effects are considered in the form of a random intercept throughout this paper. If a random intercept is not sufficient to represent the variation exhibited in the data, then the model can be easily extended to a more general form by considering a more complex random effects structure.

3.2. Estimation Procedure via Penalized h-Likelihood

$$\begin{aligned} \mathrm{h-likelihood} &= \prod_{i=1}^{k} f(v_i) \prod_{j=1}^{m} f\left(y_{ij} \middle| v_i\right) \\ &= \prod_{i=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_u} \exp\left\{-\frac{(v_i - 0)^2}{2}\right\} \prod_{j=1}^{m} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{\left(y_{ij} - x_{ij}^T\beta - v_i - f\left(t_{ij}\right)\right)^2}{2}\right\} \\ &= \frac{1}{\left(\sqrt{2\pi}\,\sigma_u\right)^k} \frac{1}{\left(\sqrt{2\pi}\,\right)^{km}} \prod_{i=1}^{k} \exp\left\{-\frac{v_i^2}{2\sigma_u^2}\right\} \prod_{j=1}^{m} \exp\left\{-\frac{\left(y_{ij} - x_{ij}^T\beta - v_i - f\left(t_{ij}\right)\right)^2}{2}\right\} \\ &= \frac{1}{\left(\sqrt{2\pi}\,\sigma_u\right)^k} \frac{1}{\left(\sqrt{2\pi}\,\right)^{km}} \exp\left\{-\frac{\sigma_u^2 \sum_{i=1}^{k} \sum_{j=1}^{m} \left(y_{ij} - x_{ij}^T\beta - v_i - f\left(t_{ij}\right)\right)^2 + \sum_{i=1}^{k} v_i^2}{2\sigma_u^2}\right\}. \end{aligned} \tag{18}$$

Thus, the log of h-likelihood is

$$\begin{aligned} l_h(\beta, v) = &-k \log\{\sqrt{2\pi}\,\sigma_u\,(\sqrt{2\pi}\,)^m\} \\ &- \frac{\sum_{i=1}^{k} \sum_{j=1}^{m} \left(y_{ij} - x_{ij}^T\beta - v_i - f\left(t_{ij}\right)\right)^2}{2} - \frac{\sum_{i=1}^{k} v_i^2}{2\sigma_u^2} \\ = &-k \log\{\sqrt{2\pi}\,\sigma_u\,(\sqrt{2\pi}\,)^m\} \\ &- \frac{1}{2}\|y - X\beta - Zv - f(t)\|_2^2 - \frac{1}{2\sigma_u^2}\|v\|_2^2. \end{aligned} \tag{19}$$

For the purpose of this paper, the first and second derivatives of $l_h(\beta, v)$ with respect to β and v are derived and listed below:

$$\begin{aligned} \nabla l_h(\beta, v) &= \frac{\partial l_h(\beta, v)}{\partial \beta} = X^T(y - X\beta - Zv - f(t)); \\ \overset{2}{\nabla} l_h(\beta, v) &= \frac{\partial^2 l_h(\beta, v)}{\partial \beta\, \partial \beta^T} = -X^T X; \\ \frac{\partial l_h(\beta, v)}{\partial v} &= Z^T(y - X\beta - Zv - f(t)) - \frac{1}{\sigma_u^2} v; \\ \frac{\partial^2 l_h(\beta, v)}{\partial v\, \partial v^T} &= -Z^T Z - \frac{1}{\sigma_u^2} I. \end{aligned} \tag{20}$$

The maximum likelihood estimate for the random effects $\widehat{v}$ is obtained by setting $\partial l_h(\beta, v)/\partial v$ to zero. Then, an approximated likelihood for the fixed effects can be obtained by plugging the estimate $\widehat{v}$ in $l_h(\beta, v)$. In addition, the marginal likelihood is approximated by the adjusted profile likelihood:

$$l(\beta) \approx \left[l_h(\beta, v) - \frac{1}{2}\log\det\left\{\frac{D(l_h(\beta, v))}{2\pi}\right\}\right]_{v=\widehat{v}}, \tag{21}$$

where $D(l_h(\beta, v)) = -\partial^2 l_h(\beta, v)/\partial v\, \partial v^T$.

Now the problem of how to estimate the smooth function $f(t)$ rises. In this paper, we use two nonparametric approaches to estimate $f(t)$: local linear regression technique and spline technique.

In the framework of penalized variable selection, we apply a penalty on the approximated marginal likelihood so that

$$l_p(\beta) = l(\beta) - n\sum_{j=1}^{p} P_\lambda\left(\left|\beta_j\right|\right), \tag{22}$$

where $P_\lambda(.)$ is the penalty function with tuning parameter λ. Our aim is to maximize $l_p(\beta)$ and get the maximum likelihood estimates for the fixed effects β. We will give a brief theoretical support on how to derive the estimation in the following paragraphs.

First of all, the L_1 penalty functions are singular at the origin, and they do not have continuous second-order derivatives. However, they can be locally approximated by a quadratic function as follows. Assume that we are given an initial value β_0 that is close to the maximizer of $l_h(\beta)$. If β_{j0} is very close to 0, then set $\widehat{\beta}_j = 0$. Otherwise, they can be locally approximated by a quadratic function as

$$\begin{aligned}\left[P\lambda\left(\left|\beta_j\right|\right)\right]' &= P'_\lambda\left(\left|\beta_j\right|\right)\operatorname{sgn}\left(\beta_j\right)\\ &\approx \frac{P'_\lambda\left(\left|\beta_{j0}\right|\right)}{\left|\beta_{j0}\right|}\beta_j\\ &\approx \frac{P'_\lambda\left(\left|\beta_{j0}\right|\right)}{\left|\beta_{j0}\right|}\frac{\beta_j + \beta_{j0}}{2},\end{aligned} \tag{23}$$

when $\beta_j \neq 0$. In other words,

$$P_\lambda\left(\left|\beta_j\right|\right) \approx P_\lambda\left(\left|\beta_{j0}\right|\right) + \frac{1}{2}\frac{P'_\lambda\left(\left|\beta_{j0}\right|\right)}{\left|\beta_{j0}\right|}\left(\beta_j^2 - \beta_{j0}^2\right), \tag{24}$$

for $\beta_j \approx \beta_{j0}$. A drawback of this approximation is that once a coefficient is shrunk to zero, it will stay at zero.

Furthermore, note the first two derivatives of the log h-likelihood function $l_h(\beta, v)$ are continuous. Around a given point β_0, the log h-likelihood function can be approximated by

$$\begin{aligned}l_h(\beta) &\approx l_h(\beta_0) + \left[\frac{\partial l_h(\beta_0)}{\partial\beta}\right]^T(\beta - \beta_0)\\ &\quad + \frac{1}{2}(\beta - \beta_0)^T\left[\frac{\partial^2 l_h(\beta_0)}{\partial\beta\,\partial\beta^T}\right](\beta - \beta_0).\end{aligned} \tag{25}$$

Similarly, $l_p(\beta)$ can be locally approximated by the quadratic function

$$\begin{aligned}l_p(\beta) &= l(\beta_0) + \nabla l(\beta_0)^T(\beta - \beta_0)\\ &\quad + \frac{1}{2}(\beta - \beta_0)^T\overset{2}{\nabla} l(\beta_0)(\beta - \beta_0)\\ &\quad - \frac{1}{2}n\beta^T\sum_\lambda(\beta_0)\beta + C,\end{aligned} \tag{26}$$

where C is a constant term, $\nabla l(\beta_0) = \partial l(\beta_0)/\partial\beta$, $\nabla^2 l(\beta_0) = \partial^2 l(\beta_0)/\partial\beta\,\partial\beta^T$, and $\sum_\lambda(\beta_0) = \operatorname{diag}\{P'_\lambda(|\beta_{10}|)/|\beta_{10}|, \ldots, P'_\lambda(|\beta_{p0}|)/|\beta_{p0}|\}$. The quadratic maximization problem yields the solution iteratively by

$$\beta_1 = \beta_0 + \left\{\overset{2}{\nabla} l(\beta_0) - n\sum_\lambda(\beta_0)\right\}^{-1}\left\{n\sum_\lambda(\beta_0)\beta_0 - \nabla l(\beta_0)\right\}. \tag{27}$$

When the algorithm converges, the estimator satisfies the penalized likelihood equation condition

$$\frac{\partial l(\widehat{\beta_0})}{\partial\beta_j} - nP'_\lambda\left(\left|\widehat{\beta_{j0}}\right|\right)\operatorname{sgn}\left(\widehat{\beta_{j0}}\right) = 0, \tag{28}$$

for nonzero elements of $\widehat{\beta_0}$.

As stated in Fan and Li [20], in the maximum likelihood estimation (MLE) setting, with good initial value of β_0, the one-step procedure can be as efficient as the fully iterative procedure, when the Newton–Raphson algorithm is used. Thus, if we have a good initial value for β, the very next iteration can be regarded as a one-step procedure, and the resulting estimator can be as efficient as the fully iterative method.

3.3. Variable Selection via the Adaptive Lasso Penalty.

There are many penalized likelihood variable selection criteria available in the literature review on penalized approaches, such as lasso penalty and SCAD. In this paper, we focus on the adaptive lasso penalty, which was introduced by Zou [40]. The form of the penalty function for adaptive lasso is given by

$$P_\lambda\left(\left|\beta_j\right|\right) = \lambda w_j\left(\left|\beta_j\right|\right), \tag{29}$$

where w is a known weights vector and λ is the tuning parameter satisfying $\lambda > 0$. It has been shown if the weights are data-dependent and cleverly chosen, the weighted lasso can achieve the oracle properties, or in other words, it performs well as if the true underlying model was known in advance. This is the main reason for our choice of penalty function. In addition, the adaptive lasso is less complicated than the smoothly clipped absolute deviation (SCAD) penalty introduced by Fan and Li [20] and hence is easier to implement.

For the choice of the data-dependent weights vector w, we use the hierarchical generalized linear model to estimate $\widehat{\beta}_{\text{hglm}}$. To specify,

$$w = \frac{1}{\left|\widehat{\beta}_{\text{hglm}}\right|^{0.5}}. \tag{30}$$

As the sample size grows, the weights for zero-coefficient estimators get to infinity, whereas the weights for nonzero-coefficients converge to a finite constant.

A significant part of our proposed method is the process of variable selection by choosing an appropriate penalty function. As a result, the choice of the tuning parameter λ in the penalty function becomes important. The most popular methods for choosing such tuning parameters are K-fold cross-validation and generalized cross-validation procedures in the literature. In fact, the consistency of selection of various shrinkage methods relies on an appropriate choice of the tuning parameters, and the method of generalized cross-validation (GCV) method has been widely used in the past literature. Therefore, we adopt the traditional method and generalized cross-validation method, for the choice of the tuning parameter. In particular, suppose we have the fitted $\widehat{Y} = HY$ for a linear method under squared error, then the standard formula for the generalized cross-validation is

$$\mathrm{GCV}_\lambda = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \widehat{f}_\lambda(x_i)}{(1-\operatorname{tr}(H))/n}\right)^2. \tag{31}$$

Then, we obtain the tuning parameter λ with the minimized GCV.

3.4. Computational Algorithm. We propose the following h-likelihood algorithm (Algorithm 1) for developing the method discussed in this paper.

The computational cost of the proposed penalized h-likelihood algorithm is of order $O(np^2)$, where n is the sample size and p is the number of associated coefficients in equation (16). The efficient path algorithm makes the proposed penalized h-likelihood algorithm an attractive method for real applications. In particular, if we have a good initial value for β, the very next iteration can be regarded as a one-step procedure, and the resulting estimator can be as efficient as the fully iterative method.

3.5. Simulation Studies. To assess the finite sample performance of our proposed method, we conduct several simulation studies. All simulations are conducted using *R* codes. Our models have the form

$$y_{ij} = f\left(t_{ij}\right) + x_{ij}^T\beta + v_i + \varepsilon_{ij}, \tag{32}$$

with $v_i \sim \mathbf{N}(0,\sigma_u^2)$ and $\varepsilon_{ij} \sim \mathbf{N}(0,\phi)$. It has been assumed throughout this chapter $\sigma_u^2 = 0.2$ and $\phi = 1$. In addition, the distribution of the response y_{ij} conditional on the random components v_i is also assumed to be $\mathbf{N}(\mu_{ij},\phi)$, where $\mu_{ij} = f(t_{ij}) + x_{ij}^T\beta + v_i$. To form the covariates $x_{ij} = (x_{ij1},\dots,x_{ij10})^T$ for the model, we draw random samples from a multivariate normal distribution $\mathbf{N}(0,\Sigma)$, where the covariance matrix Σ is assumed to have an AR (1) structure with $\sigma^2 = 1$ and $\rho = 0.5$. The choice of the correlation parameter ρ is fixed here since the choice of the correlation has little impact on the resulting penalized estimates for β by trying several values for $\rho \in [0.1, 0.9]$. Furthermore, t_{ij} are simulated from a uniform [0, 1] distribution. We do the simulation studies through several examples. For each of the cases, we run a simulation study over 100 simulated datasets.

Furthermore, for the nonparametric part of the model, we use three different functions for simulation purposes: $f(t) = \exp(0.1t)$, $f(t) = \sin(0.1\pi t)$, and $f(t) = t^2$. Both $f(t) = \exp(0.1t)$ and $f(t) = t^2$ represent a nonlinear and increasing function, whereas $f(t) = \sin(0.1\pi t)$ represents a nonlinear and nonmonotonic function.

In order to examine the finite sample performance of our proposed method, we run simulations based on the following six examples.

Example 1. We generate a balanced dataset such that there are 10 subjects within each 100 groups. In other words, we have 100 clusters and 10 subjects within each cluster, denoted by $i = 1,\dots,100$ and $j = 1,\dots,10$. The size of the true model is $d_0 = 5$ with the true values of the parameters is set to be $\beta = (7.7, 4.6, 3.8, 2.9, 5.3, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = t^2$.

Example 2. Similar to Example 1 but with reduced number of within cluster subjects. We generate a balanced dataset, such that there are 5 subjects within each 100 groups. In other words, we have 100 clusters and 5 subjects within each cluster, denoted by $i = 1,\dots,100$ and $j = 1,\dots,5$. The size of the true model is $d_0 = 5$ with the true values of the parameters set to be $\beta = (7.7, 4.6, 3.8, 2.9, 5.3, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = t^2$.

Example 3. We generate a balanced dataset such that there are 10 subjects within each 100 groups. In other words, we have 100 clusters and 10 subjects within each cluster, denoted by $i = 1,\dots,100$ and $j = 1,\dots,10$. The size of the true model is $d_0 = 3$ with the true values of the parameters set to be $\beta = (2, 1, 3, 0, 0, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = \exp(0.1t)$.

Example 4. Similar to Example 3 but with reduced number of within cluster subjects. We generate a balanced dataset, such that there are 5 subjects within each 100 groups. In other words, we have 100 clusters and 5 subjects within each cluster, denoted by $i = 1,\dots,100$ and $j = 1,\dots,5$. The size of the true model is $d_0 = 3$ with the true values of the parameters set to be $\beta = (2, 1, 3, 0, 0, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = \exp(0.1t)$.

Example 5. We generate a balanced dataset, such that there are 10 subjects within each 100 groups. In other words, we have 100 clusters and 10 subjects within each cluster, denoted by $i = 1,\dots,100$ and $j = 1,\dots,10$. The size of the true model is $d_0 = 3$ with the true values of the parameters set to be $\beta = (2, 1, 3, 0, 0, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = \sin(0.1\pi t)$.

```
[(Step 1)] (initialization).
  (i) Assume a partial linear model excluding variable selection. Express $f(t_{ij})$ in a parametric way. For example, a cubic regression
      spline can be expressed by using the truncated power basis:
      $f(t_{ij}) = \sum_{k=0}^{s} \alpha_k t_{ij}^k + \sum_{l=1}^{r} \alpha_{l+3} (t_{ij} - \tau_l)_+^3$,
      where the 5 knots $\tau_1, \ldots, \tau_5$ are percentiles of $t$, $\alpha_0, \ldots, \alpha_8$ are the associated coefficients, and $s = 3, r = 5$, are the numbers
      corresponding to the cubic regression spline representation.
  (ii) Initialize the fixed effects $\widehat{\beta}^{(0)} = \widehat{\beta}_{\text{hglm}}$, where $\widehat{\beta}_{\text{hglm}}$ is the h-likelihood estimates by treating $f(t_{ij})$ in a parametrical way. Then, we
      have
      $w_j = 1/|\widehat{\beta}_j^{(0)}|^{0.5}$.
  (iii) Denote the estimates by $\widehat{f}(t_{ij})$:
      $\widehat{f}(t_{ij}) = \sum_{k=0}^{s} \widehat{\alpha}_k t_{ij}^k + \sum_{l=1}^{r} \widehat{\alpha}_{l+3} (t_{ij} - \tau_l)_+^3$,
      where $\widehat{\alpha} = \widehat{\alpha}_{\text{hglm}}$ are the h-likelihood estimates.
  (iv) Determine initial value for random effects using
      $\widehat{v}_i^{(0)} = (\sigma_u^2 / m\sigma_u^2 + \phi) \sum_{j=1}^{m} [y_{ij} - x_{ij}^T \widehat{\beta}^{(0)} - \widehat{f}(t_{ij})]$,
      with $\sigma_u^2 = 0.2$ and $\phi = 1$.
[(Step 2)] (loop).
  (i) Use $\widehat{\beta}^{(k)}$ and $\widehat{v}_i^{(k)}$ to get
      $\widehat{v}_i = (\sigma_u^2 / m\sigma_u^2 + \phi) \sum_{j=1}^{m} [y_{ij} - x_{ij}^T \widehat{\beta}^{(k)} - \widehat{f}(t_{ij})]$.
  (ii) For the $(k+1)^{\text{th}}$ iteration, set the estimator $\widehat{\beta}^{(k)}$ from the $k^{\text{th}}$ iteration and update $\beta$ by
      $\widehat{\beta}^{(k+1)} = \beta^{(k)} + \left\{ \overset{2}{\nabla} l(\beta^{(k)}) - n \sum_{\lambda} (\beta^{(k)}) \right\}^{-1} \left\{ n \sum_{\lambda} (\beta^{(k)}) \beta^{(k)} - \nabla l(\beta^{(k)}) \right\}$.
  (iii) For $s = 1, \ldots, p$, set $\widehat{\beta}^{(k+1)} = 0$ if $\widehat{\beta}^{(k+1)} < c \sum_{s=1}^{p} |\widehat{\beta}^{(k+1)}|$, for a small cutoff value $c$.
  (iv) Compute $|(\widehat{\beta}^{(k+1)} - \widehat{\beta}^{(k)}) / \widehat{\beta}^{(k)}|$ and compare to a small predetermined value $c'$. If $|(\widehat{\beta}^{(k+1)} - \widehat{\beta}^{(k)}) / \widehat{\beta}^{(k)}|$ is smaller than $c'$, stop the
      loop.
```

Algorithm 1

Example 6. Similar to Example 5 but with reduced number of within cluster subjects. We generate a balanced dataset, such that there are 5 subjects within each 100 groups. In other words, we have 100 clusters and 5 subjects within each cluster, denoted by $i = 1, \ldots, 100$ and $j = 1, \ldots, 5$. The size of the true model is $d_0 = 3$ with the true values of the parameters set to be $\beta = (2, 1, 3, 0, 0, 0, 0, 0, 0, 0)^T$. In addition to the linear component, the nonparametric component is $f(t) = \sin(0.1\pi t)$.

We simulate each random effect v_i from a normal distribution with 0 mean and $\sigma_u^2 = 0.2$. Moreover, we simulate t_{ij} from uniform distribution of $[0, 1]$. Then, we obtain the smoothing function $f(t)$ by plugging in the values of t_{ij}. Once we have the random effects and the nonparametric part of $f(t)$, we can simulate the response y_{ij} by computing its mean and variance through the model. In this case, $y_{ij} \sim \mathbf{N}(\mu_{ij}, \phi)$, where $\mu_{ij} = f(t_{ij}) + x_{ij}^T \beta + v_i$ and $\phi = 1$.

By default, we estimate the unknown smooth function $f(t)$ by two methods: local linear kernel smoothing method and cubic spine smoothing method. We denote the estimates with respective to those two methods by PHKernel and PHSpline. In addition, we also calculated the cubic spline smoothing method without the penalty term, i.e., $\lambda = 0$, and denote the estimates algorithm by HSpline. However, due to the computational complexity of the local linear kernel smoothing method, we only consider the comparison between local linear kernel smoothing method and cubic spine smoothing method for Examples 1 and 2. For the rest of the four examples, we only run the simulations in terms of HSpline and PHSpline.

Before we report the simulation performances of our proposed penalty-based procedure, several terms, which will be listed in the summary tables, are introduced. First of all, let percentage of correctly fitted and percentage of overfitted be the proportions of selected models that are correctly fitted and overfitted, respectively. In the case of overfitting, the columns "1," "2," and ">2" represent the proportions of selected models including one, two, and more than two irrelevant predictors, correspondingly.

Furthermore, to characterize the capability of a method in producing sparse solutions, we define

$$\text{percentage of correct zeros}\,(\%) = \frac{1}{d - d_0} \left\{ \frac{1}{100} \sum_{k=1}^{100} \sum_{j=1}^{d} I\left(\widehat{\beta}_{j(k)} = 0\right) \times I\left(\beta_j = 0\right) \right\}. \tag{33}$$

To characterize the method's underfitting effect, we further define

$$\text{percentage of incorrect zeros}\,(\%) = \frac{1}{d_0} \left\{ \frac{1}{100} \sum_{k=1}^{100} \sum_{j=1}^{d} I\left(\widehat{\beta}_{j(k)} = 0\right) \times I\left(\beta_j \neq 0\right) \right\}. \tag{34}$$

Table 2 presents a detailed summary of variable selection accuracy for all the six examples provided above. Several key

Table 2: Simulation summary of PHSpline for six examples.

Example	d_0	Underfitted (%)	Correctly fitted (%)	Overfitted (%)			Correct zeros (%)	Incorrect zeros (%)
				1	2	3		
1	5	0	100	0	0	0	100	0
2	5	0	94	2	2	0	98.8	0
3	3	0	74	14	8	4	93.3	0
4	3	0	69	16	10	5	84.4	0
5	3	0	71	20	2	7	93.1	0
6	3	0	64	21	6	9	88.3	0

findings can be observed from Table 2. First of all, all the six examples do not have the underfitting problem, which means all the relevant predictors can be discovered by the PHSpline method. Equivalently, results of zeros for percent of incorrect zeros column double confirm the above statement.

Furthermore, our proposed PHSpline method is in good performance in terms of variable selection consistency for Example 1, with 100% correctly fit. Similarly, simulation results of our proposed PHSpline method for Example 2 provides a 94% correct fit, a 2% of overfit with 1 irrelevant predictor included, and a 2% of overfit with 2 irrelevant predictors included. The overall performance of variable selection consistency for Example 2 is good with a 98.8% of correct zeros.

Thirdly, when we have a more sparse representation for the fixed effects β with smaller magnitudes, our proposed PHSpline tends to provide a little bit conservative result compared to Examples 1 and 2, in terms of variable selection accuracy. In particular, simulation results of our proposed PHSpline method for Example 3 provide a 74% of correct fit, a 14% of overfit with 1 irrelevant predictor included, a 8% of overfit with 2 irrelevant predictors included, and a 4% of overfit with more than 2 irrelevant predictors included. In fact, the overall performance of variable selection consistency for Example 3 is good with a 93.3% of correct zeros. On the contrary, when the number of within-cluster subjects decreases from 10 to 5 in Example 4, percent of correct zeros decreases to 84.4%, meaning that more irrelevant predictors are included in the model.

Last but not least, similar trends can be observed for Examples 5 and 6 compared to Examples 3 and 4. Example 5 returns a 71% of correct fit, a 20% of overfit with 1 irrelevant predictor included, a 2% of overfit with 2 irrelevant predictors included, and a 7% of overfit with more than 2 irrelevant predictors included. On the contrary, Example 6 returns a 64% of correct fit, a 21% of overfit with 1 irrelevant predictor included, a 6% of overfit with 2 irrelevant predictors included, and a 9% of overfit with more than 2 irrelevant predictors included. As a result, the 71% of correct fit for Example 5 outperforms the 64% of correctly fit for Example 6, in terms of the variable selection consistency. Hence, generally speaking, our proposed PHSpline method works better when the number of within cluster subjects increases.

Besides the variable selection accuracy summarized in Table 2, prediction accuracy for the fixed effects β for various examples is also with our interest. In the following paragraphs, results of prediction accuracy for the fixed effects β are discussed and interpreted, with Tables 3–8 presented.

Table 3: Simulation result of Example 1.

Coefficients	Truth	HSpline (s.e.)	PHKernel (s.e.)	PHSpline (s.e.)
β_1	7.7	7.741 (0.043)	7.724 (0.139)	7.704 (0.048)
β_2	4.6	4.529 (0.059)	4.562 (0.179)	4.588 (0.062)
β_3	3.8	3.930 (0.060)	3.830 (0.200)	3.806 (0.078)
β_4	2.9	2.800 (0.079)	2.878 (0.177)	2.883 (0.086)
β_5	5.3	5.363 (0.081)	5.311 (0.144)	5.298 (0.090)
β_6	0	−0.031 (0.048)	0 (—)	0 (—)
β_7	0	−0.001 (0.076)	0 (—)	0 (—)
β_8	0	−0.040 (0.109)	0 (—)	0 (—)
β_9	0	−0.004 (0.040)	0 (—)	0 (—)
β_{10}	0	−0.001 (0.063)	0 (—)	0 (—)

Table 4: Simulation result of Example 2.

Coefficients	Truth	HSpline (s.e.)	PHKernel (s.e.)	PHSpline (s.e.)
β_1	7.7	7.703 (0.050))	7.701 (0.051)	7.685 (0.067)
β_2	4.6	4.604 (0.064)	4.593 (0.065)	4.608 (0.108)
β_3	3.8	3.792 (0.075)	3.872 (0.078)	3.778 (0.115)
β_4	2.9	2.909 (0.078)	2.850 (0.091)	2.891 (0.125)
β_5	5.3	5.295 (0.079))	5.308 (0.087)	5.282 (0.133)
β_6	0	0.003 (0.072)	0.005 (0.083)	0.006 (0.058)
β_7	0	−0.003 (0.061)	0 (—)	0 (—)
β_8	0	0.001 (0.058)	0 (—)	0 (—)
β_9	0	−0.002 (0.059)	0 (—)	0 (—)
β_{10}	0	0.0004 (0.042)	0 (—)	0 (—)

Table 3 summarizes simulation result over 100 replications for Example 1. As we can see, both PHkernel and PHSpline can recover the relevant predictors accurately. In addition, the estimates of the fixed effects for both PHkernel and PHSpline are comparably making very little difference with the true values of β. However, in terms of speed of the algorithm, the PHSpline method is way fast than the PHKernel method, and hence, the PHSpline method is fast

Table 5: Simulation result of Example 3.

Coefficients	Truth	HSpline (s.e.)	PHSpline (s.e.)
β_1	2	2.001 (0.045)	1.995 (0.043)
β_2	1	1.002 (0.064)	0.995 (0.061)
β_3	3	2.994 (0.075)	2.992 (0.074)
β_4	0	0.006 (0.078)	0.005 (0.043)
β_5	0	0.004 (0.079)	0.004 (0.053)
β_6	0	0.003 (0.072)	−0.004 (0.060)
β_7	0	−0.003 (0.061)	0 (—)
β_8	0	0.001 (0.058)	0 (—)
β_9	0	−0.002 (0.059)	0.001 (0.023)
β_{10}	0	0.0004 (0.042)	0 (—)

Table 6: Simulation result of Example 4.

Coefficients	Truth	HSpline (s.e.)	PHSpline (s.e.)
β_1	2	1.930 (0.120)	1.977 (0.074)
β_2	1	0.951 (0.102)	0.997 (0.100)
β_3	3	2.943 (0.089)	2.979 (0.115)
β_4	0	0.041 (0.081)	0.012 (0.106)
β_5	0	−0.005 (0.072)	−0.009 (0.104)
β_6	0	0.011 (0.096)	0.008 (0.093)
β_7	0	0.022 (0.103)	0 (—)
β_8	0	−0.009 (0.085)	0.011 (0.088)
β_9	0	0.008 (0.084)	0 (—)
β_{10}	0	0.003 (0.077)	0 (—)

Table 7: Simulation result of Example 5.

Coefficients	Truth	HSpline (s.e.)	PHSpline (s.e.)
β_1	2	2.012 (0.064)	1.999 (0.046)
β_2	1	0.988 (0.055)	1.002 (0.062)
β_3	3	2.986 (0.070)	3.000 (0.067)
β_4	0	0.003 (0.048)	0 (—)
β_5	0	0.005 (0.050)	0 (—)
β_6	0	0.010 (0.062)	0.001 (0.040)
β_7	0	−0.007 (0.079)	−0.003 (0.058)
β_8	0	0.002 (0.070)	0 (—)
β_9	0	0.006 (0.061)	0 (—)
β_{10}	0	0.009 (0.069)	−0.001 (0.014)

Table 8: Simulation result of Example 6.

Coefficients	Truth	HSpline (s.e.)	PHSpline (s.e.)
β_1	2	1.990 (0.063)	1.982 (0.071)
β_2	1	0.992 (0.070)	0.984 (0.096)
β_3	3	2.973 (0.089)	2.972 (0.106)
β_4	0	0.022 (0.061)	0.017 (0.085)
β_5	0	0.019 (0.072)	−0.023 (0.090)
β_6	0	0.004 (0.089)	0.019 (0.079)
β_7	0	−0.041 (0.080)	0.002 (0.078)
β_8	0	0.031 (0.067)	0.001 (0.061)
β_9	0	−0.014 (0.071)	0.001 (0.073)
β_{10}	0	0.009 (0.087)	−0.001 (0.015)

to implement. On the contrary, the HSpline method returns the h-likelihood estimates of the fixed effects, without the penalty term. As we can observe from Table 3, the HSpline method gives nonzero estimates for all the β, resulting in bad variable selection performance compared with PHSpline, which involves a penalty term. Furthermore, PHSpline estimates tend to have relatively smaller standard deviations than those computed in HSpline estimates. Therefore, the PHSpline method outperforms the other two methods by either variable selection accuracy or efficiency of the implementation speed.

Simulation result over 100 replications for Example 2 is summarized in Table 4. Example 2 has a smaller number of within-cluster subjects than that in Example 1. In fact, similar to the results obtained in Example 1, both PHKernel and PHSpline methods return relatively good estimates of the fixed effects β in terms of variable selection accuracy and prediction accuracy. In particular, both PHKernel and PHSpline methods select one irrelevant covariate wrongly. In addition, the estimates of the fixed effects for both PHKernel and PHSpline methods are comparably making very little difference with the true values of β. On the contrary, as we can observe from Table 4, the HSpline method gives nonzero estimates for all the β, resulting in bad variable selection performance compared with PHSpline. Furthermore, PHSpline estimates tend to have relatively smaller standard deviations than those computed in HSpline estimates. In fact, it is not surprising to see that both PHkernel and PHSpline methods include X_6 as a relevant predictor in the model. Or in an equivalent way, both PHKernel and PHSpline methods return nonzero β_6. The reason is that we have a AR (1) model, which means there is a correlation of $\rho = 0.5$ between X_5 and X_6.

`As we compare simulation results of Examples 1 and 2, our proposed PHSpline method tends to perform better when the number of within-cluster subjects increases. In addition, a similar conclusion can be drawn for the PHKernel method. Furthermore, both PHKernel and PHSpline methods work well when the nonparametric component is $f(t) = t^2$.

Tables 5 and 6 present simulation results over 100 replications for Examples 3 and 4. In these two examples, we have a more sparse representation in terms of the fixed effects β than those in Examples 1 and 2. On top of that, the magnitudes of the fixed effects β are set to be smaller than those in Examples 1 and 2. For both of the results, the PHSpline method outperforms the HSpline method in terms of variable selection performance in two ways. First of all, the PHSpline method identifies some of the irrelevant predictors accurately, whereas the HSpline method gives nonzero estimates for all the β. Though PHSpline cannot guarantee 100% selection accuracy, it does improve the poor variable selection performance of HSpline by adding a penalty term. Furthermore, PHSpline estimates tend to have relatively smaller standard deviations than those computed in HSpline estimates. Therefore, the PHSpline method performs better than the HSpline method, even for the sparse fixed effects β situation.

Similarly, simulation results over 100 replications for Examples 5 and 6 are presented in Tables 7 and 8. Again, we have a more sparse representation in terms of the fixed effects β than those in Examples 1 and 2, with smaller magnitudes of the fixed effects β. The PHSpline method works pretty well in terms of variable selection for Example 5 even though it does not guarantee a 100% selection accuracy. On the contrary, the PHSpline method returns nonzero estimates for all the β, resulting in poor variable selection performance for Example 6, where the number of within cluster subjects reduces to 5.

Overall, the simulation results show that our proposed penalized h-likelihood approach performs good in terms of variable selection accuracy because of its ability to recover the true zeros, especially when the number of within-cluster subjects is not too small. Generally, our proposed PHSpline method works better when the number of within cluster subjects increases. In addition, even when the true model is sparse, our penalized estimator still does no worse than the h-likelihood estimator in terms of estimation accuracy.

4. Conclusion

To conclude, we have introduced a new penalized h-likelihood approach to identify nonzero relevant fixed effects in the partial linear model setting in this paper. This penalized h-likelihood incorporates variable selection procedures in the setting of mean modeling via h-likelihood. A few advantages of this newly proposed method are listed below. First of all, compared to the traditional marginal likelihood, the h-likelihood avoids the messy integration for the random effects and hence is convenient to use. In addition, h-likelihood plays an important role in inferences for models having unobserved random variables. Last but not least, it has been demonstrated by simulation studies that the proposed penalty-based method is able to identify zero regression coefficients in modeling the mean structure and produces good fixed effects estimation results.

As for future research, it would be interesting to apply the proposed penalized h-likelihood approach to be extended for more complicated circumstances for the partial linear models. In other words, the model in this paper assumes only a simple one-component structure for the random effects, such that only a random intercept is considered. For possible future research, we may consider a partial linear model for modeling the conditional mean with more than one random effect, i.e., the extended multi-component random effects model. Other future work, including variance components estimates of the random effects and study of penalized h-likelihood estimator's theoretical and asymptotical property such as convergence rate, would be investigated and discussed.

Data Availability

This is a theoretical study, and we do not have experimental data.

Disclosure

This work was part of the originally written Ph.D. thesis by the first author in 2013 [41].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded by the Ministry of Education of Humanities and Social Science Project (Grant no. 17YJCZH199). The authors gratefully acknowledge the Ministry of Education of Humanities and Social Science for the technical and financial support.

References

[1] R. S. Sutton and A. G. Barto, "Reinforcement learning: an introduction," *IEEE Transactions on Neural Networks*, vol. 9, no. 5, p. 1054, 1998.

[2] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[3] T. T. Cai and L. Wang, "Orthogonal matching pursuit for sparse signal recovery with noise," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4680–4688, 2011.

[4] Y. Lee and J. A. Nelder, "Hierarchical generalized linear models," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 4, pp. 619–678, 1996.

[5] R. Sutton and G. Barto, "Reinforcement learning," *A Bradford Book*, vol. 15, no. 7, pp. 665–685, 1998.

[6] A. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning," *Discrete Event Dynamic Systems*, vol. 13, no. 1-2, pp. 341–379, 2003.

[7] T. Dietterich, "An overview of MAXQ hierarchical reinforcement learning," in *Proceedings of the International Symposium on Abstraction*, Horseshoe Bay, TX, USA, July 2000.

[8] S. He, H. Fang, M. Zhang, F. Liu, X. Luan, and Z. Ding, "Online policy iterative-based H_∞ optimization algorithm for a class of nonlinear systems," *Information Sciences*, vol. 495, pp. 1–13, 2019.

[9] S. He, M. Zhang, H. Fang, F. Liu, X. Luan, and Z. Ding, "Reinforcement learning and adaptive optimization of a class of Markov jump systems with completely unknown dynamic information," *Neural Computing and Applications*, 2019.

[10] C. Wang, H. Fang, and S. He, "Adaptive optimal controller design for A class of LDI-based neural network systems with input time-delays," *Neurocomputing*, vol. 385, pp. 292–299, 2019.

[11] S. He, H. Fang, M. Zhang, F. Liu, and Z. Ding, "Adaptive optimal control for a class of nonlinear systems: the online policy iteration approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 2, pp. 549–558, 2020.

[12] P. Cheng, J. Wang, S. He, X. Luan, and F. Liu, "Observer-based asynchronous fault detection for conic-type nonlinear jumping systems and its application to separately excited DC motor," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 3, 2020.

[13] P. Cheng, S. He, J. Cheng, X. Luan, and F. Liu, "Asynchronous output feedback control for a class of conic-type nonlinear hidden Markov jump systems within a finite-time interval," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 99, pp. 1–8, 2020.

[14] J. A. Nelder and R. W. M. Wedderburn, "Generalized linear models," *Journal of the Royal Statistical Society. Series A (General)*, vol. 135, no. 3, pp. 370–384, 1972.

[15] N. M. Laird and J. H. Ware, "Random-effects models for longitudinal data," *Biometrics*, vol. 38, no. 4, pp. 963–974, 1982.

[16] F. Vonesh, C. Edward, and M. Vernon, "Linear and nonlinear models for the analysis of repeated measurements," *Journal of Biopharmaceutical Statistics*, vol. 18, no. 4, pp. 595–610, 1996.

[17] F. Vaida and S. Blanchard, "Conditional Akaike information for mixed-effects models," *Biometrika*, vol. 92, no. 2, pp. 351–370, 2005.

[18] W. Pu and X. Niu, "Selecting mixed-effects models based on a generalized information criterion," *Journal of Multivariate Analysis*, vol. 97, no. 3, pp. 733–758, 2008.

[19] R. Rajaram and B. Castellani, "The utility of nonequilibrium statistical mechanics, specifically transport theory, for modeling cohort data," *Complexity*, vol. 20, no. 4, pp. 45–57, 2015.

[20] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[21] J. Fan and I. Gijbels, "Variable bandwidth and local linear regression smoothers," *The Annals of Statistics*, vol. 20, no. 4, pp. 2008–2036, 1992.

[22] J. Fan, "Design-adaptive nonparametric regression," *Journal of the American Statistical Association*, vol. 87, no. 420, pp. 998–1004, 1992.

[23] T. Hastie and C. Loader, "Local regression: automatic kernel carpentry (with discussion)," *Statistical Science*, vol. 8, pp. 120–143, 1993.

[24] J. Fan and J.-T. Zhang, "Two-step estimation of functional linear models with applications to longitudinal data," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 2, pp. 303–322, 2000.

[25] Wu and Rice, "Nonparametric mixed effects models for unequally sampled noisy curves," *Biometrics*, vol. 57, no. 1, pp. 253–259, 2001.

[26] C. Caves and R. Schack, "Unpredictability, information, and chaos," *Complexity*, vol. 3, no. 1, pp. 46–57, 2015.

[27] Y. Zhang, Y. Zhou, X. Guo et al., "Self-adaptive K-means based on a covering algorithm," *Complexity*, vol. 2018, Article ID 7698274, 16 pages, 2018.

[28] T. Martinussen and T. Scheike, "Sampling corrected analysis of dynamic additive regression models for longitudinal data," *University of Copenhagen*, vol. 28, no. 2, pp. 303–323, 2001.

[29] R. Xu, "Measuring explained variation in linear mixed effects models," *Statistics in Medicine*, vol. 22, no. 22, pp. 3527–4354, 2003.

[30] S. L. Zeger and P. J. Diggle, "Semiparametric models for longitudinal data with application to CD4 cell numbers in HIV seroconverters," *Biometrics*, vol. 50, no. 3, pp. 689–699, 1994.

[31] A. Groll and G. Tutz, "Variable selection for generalized linear mixed models by L 1-penalized estimation," *Statistics and Computing*, vol. 24, no. 2, pp. 137–154, 2014.

[32] N. Heckman, R. Lockhart, and J. D. Nielsen, "Penalized regression, mixed effects models and appropriate modelling," *Electronic Journal of Statistics*, vol. 7, pp. 1517–1552, 2013.

[33] P. Diggle, K. Liang, and S. Zeger, *Analysis of Longitudinal Data*, Clarendon Press, Oxford, UK, 1994.

[34] Y. Lee and J. A. Nelder, "Modelling and analysing correlated non-normal data," *Statistical Modelling*, vol. 1, no. 1, pp. 3–16, 2001.

[35] M. Noh and Y. Lee, "Hierarchical-likelihood approach for nonlinear mixed-effects models," *Computational Statistics & Data Analysis*, vol. 52, no. 7, pp. 3517–3527, 2008.

[36] Y. Lee and J. Nelder, "Double hierarchical generalized linear models (with discussion)," *Journal of the Royal Statistical Society*, vol. 55, no. 4, pp. 139–185, 2010.

[37] M. Noh and Y. Lee, "Double hierarchical generalized linear models," *Journal of the Royal Statistical Society*, vol. 55, no. 2, pp. 139–185, 2017.

[38] R. Greenlaw and S. Kantabutra, "On the parallel complexity of hierarchical clustering and CC-complete problems," *Complexity*, vol. 14, no. 2, pp. 18–28, 2010.

[39] M. Noh and Y. Lee, "REML estimation for binary data in GLMMs," *Journal of Multivariate Analysis*, vol. 98, no. 5, pp. 896–915, 2007.

[40] H. Zou, "The adaptive Lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, 2006.

[41] Y. Xie, "Variable selection procedures in linear regression models," Ph.D. dissertation, Stats Department, NUS, Singapore, 2013.