Chapter

Sparse Boosting Based Machine Learning Methods for High-Dimensional Data

Mu Yue

Abstract

In high-dimensional data, penalized regression is often used for variable selection and parameter estimation. However, these methods typically require time-consuming cross-validation methods to select tuning parameters and retain more false positives under high dimensionality. This chapter discusses sparse boosting based machine learning methods in the following high-dimensional problems. First, a sparse boosting method to select important biomarkers is studied for the right censored survival data with high-dimensional biomarkers. Then, a two-step sparse boosting method to carry out the variable selection and the model-based prediction is studied for the high-dimensional longitudinal observations measured repeatedly over time. Finally, a multi-step sparse boosting method to identify patient sub-groups that exhibit different treatment effects is studied for the high-dimensional dense longitudinal observations. This chapter intends to solve the problem of how to improve the accuracy and calculation speed of variable selection and parameter estimation in high-dimensional data. It aims to expand the application scope of sparse boosting and develop new methods of high-dimensional survival analysis, longitudinal data analysis, and subgroup analysis, which has great application prospects.

Keywords: sparse boosting, high-dimensional data, machine learning, variable selection, data analysis

1. Introduction

High-dimensional model has become very popular in statistical literature and many new machine learning techniques have been developed to deal with data with very large number of features. In the past decades, researchers have done a great deal of high-dimensional data analysis where the sample size n is relatively small but the number of features p under consideration is extremely large. It is widely known that including irrelevant predictors in the statistical model may result in unstable estimation and dreadful computing issues. Thus, variable selection is crucial to address the challenges. Among all developments, regularization procedures such as LASSO [1], smoothly clipped absolute deviation (SCAD) [2], MCP [3] and their various extensions [4–6] have been thoroughly studied and widely used to perform variable selection and estimation simultaneously in order to improve the prediction accuracy and interpretability of the statistical model. However, those penalized

estimation approaches all have some tuning parameters required to be selected by computationally expensive methods like cross-validation.

In recent years, machine learning methods such as boosting have become very prominent for high-dimensional data settings since they can improve the selection accuracy substantially and reduce the chance of including irrelevant features. The original boosting algorithms were proposed by Schapire [7] which is an ensemble method that iteratively combines weaker learners to minimize the expected loss. The major difference among different boosting algorithms is the loss function. For example, AdaBoost [8] has the exponential loss function, L2 boosting [9] has the squared error loss function, sparse boosting [10] has the penalized loss function and HingeBoost [11] has the weighted hinge loss function. Recently, more various versions of boosting algorithms have been proposed. See, for example, Bühlmann and Hothorn [12] for the twin boosting; Komori and Eguchi [13] for the pAUCBoost; Wang [14] for the twin HingeBoost; Zhao [15] for the GSBoosting and Yang and Zou [16] for the ER-Boost. Besides these extensions, much effort has been made in understanding the advantages of boosting such as relatively lower over-fitting risk, smaller computational cost, and simpler adjustment to include additional constraints.

In this chapter we review some sparse boosting based methods for the following high-dimensional problems based on three research papers. First, a sparse boosting method to select important biomarkers is studied for the right censored survival data with high-dimensional biomarkers [17]. Then, a two-step sparse boosting to carry out the variable selection and the model-based prediction is studied for the high-dimensional longitudinal observations measured repeated over time [18]. Finally, a multi-step sparse boosting method to identify patient subgroups that exhibit different treatment effects is studied for the high-dimensional dense longitudinal observations [19]. This chapter intends to solve the problem of how to improve the accuracy and calculation speed of variable selection and parameter estimation in high-dimensional data. It aims to expand the application scope of sparse boosting and develop new methods of high-dimensional survival analysis, longitudinal data analysis, and subgroup analysis, which has great application prospects.

The rest of the chapter is arranged as follows. In Section 2, a sparse boosting method to fit high-dimensional survival data is studied. In Section 3, a two-step sparse boosting approach to carry out variable selection and model-based prediction by fitting high-dimensional models with longitudinal data is studied. In Section 4, a subgroup identification method incorporating multi-step sparse boosting algorithm for high-dimensional dense longitudinal data is studied. Finally, Section 5 provides concluding remarks.

2. Sparse boosting for survival data

Survival time data are usually referred to time-to-event data and they are usually censored. Predicting survival time and identifying the risk factors can be very helpful for patient treatment selection, disease prevention strategy or disease management in evidence-based medicine. A well-known model in survival analysis is the Cox proportional hazards (PH) model [20] which assumes multiplicative covariate effects in the hazards function. Another popular model is the accelerated failure time (AFT) model [21] which assumes that the covariate effect is to accelerate or decelerate the life time of a disease. The coefficients in the regression model have the direct interpretation of the covariate effects on the mean survival time. Recently, researchers developed boosting methods to analyze survival data. For

example, Schmid and Hothorn [22] proposed a flexible boosting method for parametric AFT models, and Wang and Wang [23] proposed Buckley-James boosting for survival data with right censoring and high dimensionality.

In this section, a sparse boosting method to fit high-dimensional varying-coefficient AFT models is presented. In particular, the sparse boosting techniques for right censored survival data is studied. In Section 2.1, the varying-coefficient AFT model for survival data is formulated and a detailed sparse boosting algorithm to fit the model is proposed. In Section 2.2, the proposed sparse boosting techniques through simulation studies is evaluated. In Section 2.3, the performance of sparse boosting via a lung cancer data example is examined.

2.1 Methodology

2.1.1 Model and estimation

Let T_i and C_i be the logarithm of survival time and censoring time for the ith subject in a random sample of size n respectively. In reality $Y_i = \min\{T_i, C_i\}$ and the censoring indicator $\delta_i = I(T_i \le C_i)$ [24] are observed. Denote $\mathbf{X}_i = \left(X_{i,1}, \cdots, X_{i,p-1}\right)$ to be the corresponding $(p\text{-}1)$-dimensional predictors such as gene expressions or biomarkers for the ith subject and U_i to be the univariate index variable. Our observed data set $\left\{(\mathbf{X}_i, \delta_i, Y_i, U_i) : \mathbf{X}_i \in \mathbb{R}^{p-1}, \delta_i \in \{0,1\}, Y_i \in \mathbb{R}, U_i \in \mathbb{R}, i = 1, 2, \cdots, n\right\}$ is an independently and identically distributed random sample from $(\mathbf{X}, \delta, Y, U)$. The varying-coefficient AFT model is:

$$T_i = \beta_0(U_i) + \sum_{j=1}^{p-1} X_{i,j}\beta_j(U_i) + \varepsilon_i, \quad i = 1, \ldots, n, \tag{1}$$

where $\beta_0(.), \beta_1(.), \cdots, \beta_{p-1}(.)$ are the unknown varying-coefficient functions of confounder U and ε_i is the random error with $E(\varepsilon_i|\mathbf{X}_i, U_i) = 0$.

A weighted least squares estimation approach is adopted. Let w_i's be the Kaplan–Meier weights [25], which are the jumps in the Kaplan–Meier estimator computed as $w_1 = \frac{\delta_{(1)}}{n}$ and $w_i = \frac{\delta_{(i)}}{n-i+1}\prod_{j=1}^{i-1}\left(\frac{n-j}{n-j+1}\right)^{\delta_{(j)}}, i = 2, \ldots, n$. Let $Y_{(1)} \le \cdots \le Y_{(n)}$ be the order statistics of $Y_{i'}s$, $\delta_{(1)}, \cdots, \delta_{(n)}$ be the corresponding censoring indicators of the ordered $Y_{i'}s$, and $X_{(1),j}, \cdots, X_{(n),j}, j = 1, \cdots, p-1$ and $U_{(1)}, \cdots, U_{(n)}$ are defined similarly. Then the weighed least squares loss function is

$$\sum_{i=1}^{n} w_i \left(Y_{(i)} - \beta_0\left(U_{(i)}\right) - \sum_{j=1}^{p-1} X_{(i),j}\beta_j\left(U_{(i)}\right)\right)^2. \tag{2}$$

Let $B(.) = (B_1(.), \ldots, B_L(.))^T$ be an equal-spaced B-spline basis, where L is the dimension of the basis. Under certain smoothness conditions, the Curry-Schonberg theorem [26] implies that for every smooth function $\beta_j(.)$, it can be approximated by

$$\beta_j(.) \approx B^T(.)\gamma_j, \quad j = 0, \cdots, p-1, \tag{3}$$

where γ_j is a vector of length L. Then the weighted least squares loss function Eq. (2) can be approximated by

$$\sum_{i=1}^{n} w_i \left(Y_{(i)} - B^T\left(U_{(i)}\right)\gamma_0 - \sum_{j=1}^{p-1} X_{(i),j} B^T\left(U_{(i)}\right)\gamma_j \right)^2. \tag{4}$$

Denote by $\tilde{Y} = \left(Y_{(1)}, \cdots, Y_{(n)}\right)^T$, $X_{(i),0} = 1$ for $i = 1, \cdots, n$, $\tilde{\mathbf{X}}_j = \left(B\left(U_{(1)}\right)X_{(1),j}, \cdots, B\left(U_{(n)}\right)X_{(n),j}\right)^T$, $\tilde{\mathbf{X}} = \left(\tilde{\mathbf{X}}_0, \cdots, \tilde{\mathbf{X}}_{p-1}\right)$, $W = \text{diag}(w_1, \cdots, w_n)$ and $\gamma = \left(\gamma_0^T, \cdots, \gamma_{p-1}^T\right)^T$. Then the objective function Eq. (4) may be written in the following matrix form:

$$\left(\tilde{Y} - \tilde{\mathbf{X}}\gamma\right)^T W \left(\tilde{Y} - \tilde{\mathbf{X}}\gamma\right). \tag{5}$$

The estimation may yield close-form solution for the coefficients when dimensionality p is small or moderate. With high dimensionality the solution cannot be easily achieved. Let $\gamma^{[\hat{K}]} = \left(\left(\gamma_0^{[\hat{K}]}\right)^T, \cdots, \left(\gamma_{p-1}^{[\hat{K}]}\right)^T\right)^T$ be the estimator of γ from sparse boosting approach with weighted square loss function Eq. (5), and $\hat{K}$ is the estimated number of stopping iterations. Then the estimates of coefficient function are given by

$$\hat{\beta}_j(u) = B^T(u)\gamma_j^{[\hat{K}]}, \quad j = 0, \cdots, p-1. \tag{6}$$

Instead of using the regularized estimation approaches, a sparse boosting method to estimate $\gamma^{[\hat{K}]}$ is presented in the following subsection.

2.1.2 Sparse boosting techniques

The key idea of sparse boosting is to replace the empirical risk function in L2 boosting with the penalized empirical risk function which is a combination of squared loss and the trace of boosting operator as a measure of boosting complexity, and then perform gradient descent in a function space iteratively. Thus sparse boosting produces sparser models compared to L2 boosting. The g-prior minimum description length (gMDL) proposed by [27] can be used as the penalized empirical risk function to estimate the update criterion in each iteration and the stopping criterion. The gMDL takes the form:

$$\begin{aligned} \text{gMDL}(RSS, \text{trace}(\mathcal{B})) &= \log(S) + \frac{\text{trace}(\mathcal{B})}{n} \log\left(\frac{\tilde{Y}^T\tilde{Y} - RSS}{\text{trace}(\mathcal{B}) \times S}\right), \\ S &= \frac{RSS}{n - \text{trace}(\mathcal{B})}. \end{aligned} \tag{7}$$

Here RSS is the residual sum of squares and $\mathcal{B}$ is the boosting operator. The model that achieves the shortest description of data will be selected. The advantage is that it has a data-dependent penalty for each dimension since it is explicitly given as a function of data only, thus the selection of the tuning parameter can be avoided.

The sparse boosting procedure is described in details. The initial value of γ is set to be a zero vector, i.e. $\gamma^{[k]} = \mathbf{0}$ for $k = 0$, while in each of the kth iteration ($1 \le k \le K$ for K being the total number of iterations) only the current residual $R^{[k]} = \tilde{Y} - \tilde{\mathbf{X}}\gamma^{[k-1]}$ is used

to regress every jth working element $\tilde{\mathbf{X}}_j, j = 0, \cdots, p-1$. The fit denoted by $\hat{\lambda}_j^{[k]}$ can be obtained by minimizing the weighted squared loss function $\left(R^{[k]} - \tilde{\mathbf{X}}_j\lambda\right)^T W\left(R^{[k]} - \tilde{\mathbf{X}}_j\lambda\right)$ with respect to λ. Hence the weighted least squared estimate is $\hat{\lambda}_j^{[k]} = \left[\left(\tilde{\mathbf{X}}_j\right)^T W\left(\tilde{\mathbf{X}}_j\right)\right]^{-1}\left(\tilde{\mathbf{X}}_j\right)^T WR^{[k]}$, the corresponding hat matrix is $\mathcal{H}_j = \left(\tilde{\mathbf{X}}_j\right)\left[\left(\tilde{\mathbf{X}}_j\right)^T W\left(\tilde{\mathbf{X}}_j\right)\right]^{-1}\left(\tilde{\mathbf{X}}_j\right)^T W$ and the weighted residual sum of squares is $RSS_j^{[k]} = \left(R^{[k]} - \tilde{\mathbf{X}}_j\hat{\lambda}_j^{[k]}\right)^T W\left(R^{[k]} - \tilde{\mathbf{X}}_j\hat{\lambda}_j^{[k]}\right)$. The selected component $\hat{s}_k$ can be obtained by:

$$\hat{s}_k = \text{argmin}_{0\le j\le p-1}\text{gMDL}\left(RSS_j^{[k]}, \text{trace}\left(\mathcal{B}_j^{[k]}\right)\right), \tag{8}$$

where $\mathcal{B}_j^{[1]} = \mathcal{H}_j$ and $\mathcal{B}_j^{[k]} = I - \left(I - \mathcal{H}_j\right)\left(I - \nu\mathcal{H}_{\hat{s}_{k-1}}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$ for $k > 1$ is the boosting operator for selecting jth component in the kth iteration. Therefore, at each iteration there is only one working component $\tilde{\mathbf{X}}_{\hat{s}_k}$ to be chosen, and only the corresponding coefficient vector $\gamma_{\hat{s}_k}^{[k]}$ changes, i.e. $\gamma_{\hat{s}_k}^{[k]} = \gamma_{\hat{s}_k}^{[k-1]} + \nu\hat{\lambda}_{\hat{s}_k}^{[k]}$, where ν is the step size, while all the other $\gamma_j^{[k]}$ for $j \neq \hat{s}_k$ remain the same. This process is repeated for K iterations and estimate the stopping iteration K by.

$$\hat{K} = \text{argmin}_{1\le k\le K}\text{gMDL}\left(RSS_{\hat{s}_k}^{[k]}, \text{trace}\left(\mathcal{B}^{[k]}\right)\right), \tag{9}$$

where $\mathcal{B}^{[k]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_k}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$.

From this sparse boosting procedure, the estimator of γ is obtained as $\gamma^{[\hat{K}]} = \left(\left(\gamma_0^{[\hat{K}]}\right)^T, \cdots, \left(\gamma_{p-1}^{[\hat{K}]}\right)^T\right)^T$. The sparse boosting algorithm for the varying-coefficient AFT model can be summarized as follows:

Sparse Boosting Algorithm for Varying-Coefficient AFT Model.

a. Initialization. Set $k = 0$ and $\gamma_0^{[k]} = \mathbf{0}, \cdots, \gamma_{p-1}^{[k]} = \mathbf{0}$ (component-wise).

b. Iteration. $k = k + 1$. Compute $\hat{s}_k = \text{argmin}_{0\le j\le p-1}\text{gMDL}\left(RSS_j^{[k]}, \text{trace}\left(\mathcal{B}_j^{[k]}\right)\right)$, where $\mathcal{B}_j^{[1]} = \mathcal{H}_j$ and $\mathcal{B}_j^{[k]} = I - \left(I - \mathcal{H}_j\right)\left(I - \nu\mathcal{H}_{\hat{s}_{k-1}}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$ for $k > 1$.

c. Update. $\gamma_{\hat{s}_k}^{[k]} = \gamma_{\hat{s}_k}^{[k-1]}$ for $j \neq \hat{s}_k$ and $\gamma_{\hat{s}_k}^{[k]} = \gamma_{\hat{s}_k}^{[k-1]} + \nu\hat{\lambda}_{\hat{s}_k}^{[k]}$, where ν is the step size.

d. Iteration. Repeat step (b)-(c) for K iterations.

e. Stopping. Estimate $\hat{K} = \text{argmin}_{1\le k\le K}\text{gMDL}\left(RSS_{\hat{s}_k}^{[k]}, \text{trace}\left(\mathcal{B}^{[k]}\right)\right)$, where $\mathcal{B}^{[k]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_k}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$.Thus, $\gamma^{[\hat{K}]} = \left(\left(\gamma_0^{[\hat{K}]}\right)^T, \cdots, \left(\gamma_{p-1}^{[\hat{K}]}\right)^T\right)^T$ is the estimate for γ and $\hat{\beta}_j(u) = B^T(u)\gamma_j^{[\hat{K}]}, j = 0, \cdots, p-1$ are the estimators for varying coefficients. The final estimator of $\tilde{Y}$ is $\tilde{Y}^{[\hat{K}]} = \tilde{\mathbf{X}}\gamma^{[\hat{K}]}$.

According to [10] and references therein, the selection of step size ν is of minor importance as long as it is small. A smaller value of ν achieves higher prediction

accuracy while requires a larger number of boosting iterations and more computing time. A typical value used in literature is $\nu = 0.1$.

2.2 Simulation

The performance of the above sparse boosting algorithm is evaluated by studying their performance on simulated data. L2 boosting and sparse boosting methods are compared in their performance of variable selection and function estimation. Sparse boosting method is what we present in this section while L2 boosting method is a relatively simpler version and may not achieve sparse solution in general.

The simulation results from [17] show that both boosting methods can identify important variables while sparse boosting selects much fewer irrelevant variables than L2 boosting. Although in-sample prediction errors (defined as $\sum_{i=1}^{n}\delta_i\left(Y_i - Y_i^{[\hat{K}]}\right)^2/\sum_{i=1}^{n}\delta_i$) using L2 boosting is a little bit smaller than using sparse boosting since the former has larger model sizes, the average of root mean integrated squared errors (defined as $\sqrt{\frac{1}{n}\sum_{j=0}^{5}\sum_{i=1}^{n}\left(\beta_j(u_i) - \hat{\beta}_j(u_i)\right)^2}$) using sparse boosting is much smaller than that using L2 boosting. Furthermore, when the smoothness assumption in Curry-Schonberg theorem is violated for the coefficient functions, the performance of variable selection remains good. In summary, sparse boosting outperforms L2 boosting in terms of parameter estimation and variable selection.

2.3 Lung cancer data analysis

Lung cancer is the top cancer killer for people in the U.S. Identifying relevant gene expressions in lung cancer is important for treatment and prevention. Our data is from a large multi-site blinded validation study [28] with 442 lung adenocarcinomas. Age is treated as the potential confounder in this analysis, since it is usually strongly correlated with survival time [29]. After removing missing measurements and predictors in overall survival, a total of 439 patients are left in the analysis. For each patient, 22,283 gene expressions are available. The median follow-up time is 46 months (range: 0.03 to 204 months) with the overall censoring rate 46.47 %. The median age at diagnosis is 65 years (range: 33 to 87 years). After adopting a marginal screening procedure to screen out irrelevant genes, variable selection approaches are used to identify important genes associated with lung cancer. With the aim of comparison, except L2 boosting and the proposed sparse boosting, the following existing variable selection approaches for constant-coefficient AFT models are also considered: Buckley-James boosting with linear least squares [23], Buckley-James twin boosting with linear least squares [23], Buckley-James regression with elastic net penalty [30] and SCAD penalty respectively.

The results from [17] show that L2 boosting and sparse boosting for varying-coefficient AFT model not only produce relatively sparser model, but also have smaller in-sample and out-of-sample prediction error compared to the four methods for constant-coefficient AFT model. Again, sparse boosting produce even sparser model than L2 boosting. In conclusion, including age in the varying-coefficient AFT model could lead to more accurate estimate than constant-coefficient AFT model and the proposed sparse boosting method for varying-coefficient AFT model has good performance in terms of estimation, prediction as well as sparsity.

3. Two-step sparse boosting for longitudinal data

Longitudinal data contain repeated measurements collected from the same respondents over time. The assumption that all measurements are independent does not hold for such data. One important question in longitudinal analysis is how to make efficient inference by taking into account of the within subjects correlation. This question has been investigated in depth by many researchers [31, 32] for parametric models. Semiparametric and nonparametric models for longitudinal data are also presented in the literature, see [33, 34]. Recently, there are some development on longitudinal data with high-dimensionalilty using varying-coefficient models [35, 36]. All previous studies adopted the penalty methods.

In this section, a two-step sparse boosting approach is presented to preform the variable selection and the model-based prediction. Specifically, high-dimensional varying-coefficient models with longitudinal data will be considered. In the first step, the sparse boosting approach is utilized to obtain an estimate of the correlation structure. In the second step, the within-subject correlation structure is considered and variable selection and coefficients estimation are achieved by sparse boosting again. The rest of this section is arranged as follows. In Section 3.1, the varying-coefficient model for longitudinal data is formulated and a two-step sparse boosting algorithm is presented. In Section 3.2, simulation studies are conducted to illustrate the validity of the two-step sparse boosting method. In Section 3.3, the performance of two-stage method is assessed by studying yeast cell cycle gene expression data.

3.1 Methodology

3.1.1 Model and estimation

Let Y_{ij} be the continuous outcome for the jth measurement of individual i taken at time $t_{ij} \in T$, where T is the time interval on which the measurements are taken. Denote $\mathbf{X}_{ij} = \left(X_{ij,1}, \cdots, X_{ij,p-1}\right)$ to be the corresponding (p-1)-dimensional covariate vector. The varying-coefficient model which can capture the dynamical impacts of the covariates on the response variable is considered:

$$Y_{ij} = \beta_0\left(t_{ij}\right) + \sum_{d=1}^{p-1} X_{ij,d}\beta_d\left(t_{ij}\right) + \varepsilon_{ij}, \quad i = 1, \cdots, n, j = 1, \cdots, n_i, \tag{10}$$

where $\beta_0(.), \beta_1(.), \cdots, \beta_{p-1}(.)$ are the unknown smooth coefficient functions of time and $\varepsilon_i = (\varepsilon_{i1}, \cdots, \varepsilon_{in_i})^T, i = 1, \cdots, n$ are multivariate error terms with mean zero. Errors are assumed to be uncorrelated for different i, but components of ε_i are correlated with each other. Without loss of generality, the balanced longitudinal study is considered in the following implementation, i.e., $t_{ij} = t_{kj}$, and $n_i = m$ for all i.

The estimation procedure is presented below. In the first step, the within-subject correlation is ignored first and the coefficients are estimated by minimizing the following least squares loss function:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left(Y_{ij} - \beta_0\left(t_{ij}\right) - \sum_{d=1}^{p-1} X_{ij,d}\beta_d\left(t_{ij}\right)\right)^2. \tag{11}$$

The B-spline basis is used to estimate the coefficient functions $\beta_0(.), \beta_1(.), \cdots, \beta_{p-1}(.)$. Denote $B(.) = (B_1(.), \ldots, B_L(.))^T$ to be an equal-spaced

B-spline basis of dimension L. Under certain smoothness assumptions, function $\beta_d(.)$ can be approximated by

$$\beta_d(.) \approx B^T(.)\gamma_d, \quad d = 0, \cdots, p-1, \tag{12}$$

where γ_d is a loading vector of length L. Then the least squares loss function Eq. (11) is close to

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left(Y_{ij} - B^T(t_{ij})\gamma_0 - \sum_{d=1}^{p-1} X_{ij,d}B^T(t_{ij})\gamma_d\right)^2. \tag{13}$$

Further denote $Y_i = (Y_{i1}, \cdots, Y_{im})^T, Y = \left(Y_1^T, \cdots, Y_n^T\right)^T, X_{ij,0} = 1, \tilde{\mathbf{X}}_{i,d} = (B(t_{i1})X_{i1,d}, \cdots, B(t_{im})X_{im,d})^T, \tilde{\mathbf{X}}_i = \left(\tilde{\mathbf{X}}_{i,0}, \cdots, \tilde{\mathbf{X}}_{i,p-1}\right), \tilde{\mathbf{X}} = \left(\tilde{\mathbf{X}}_1^T, \cdots, \tilde{\mathbf{X}}_n^T\right)^T$ and $\gamma = \left(\gamma_0^T, \cdots, \gamma_{p-1}^T\right)^T$. Then the target function Eq. (13) can be expressed in the matrix format:

$$\sum_{i=1}^{n}\left(Y_i - \tilde{\mathbf{X}}_i\gamma\right)^T\left(Y_i - \tilde{\mathbf{X}}_i\gamma\right) \equiv \left(Y - \tilde{\mathbf{X}}\gamma\right)^T\left(Y - \tilde{\mathbf{X}}\gamma\right). \tag{14}$$

Denote $\gamma^{\left[\widehat{K_1}\right]}$ to be the estimator of γ by sparse boosting with squared loss function Eq. (14) being loss function, where $\widehat{K_1}$ is the estimated stopping iterations in this step. There is no exact closed form for $\gamma^{\left[\widehat{K_1}\right]}$ since it is derived from an iterative algorithm. However it can be evaluated very fast in a computer implementation. The detailed algorithm will be presented in the next subsection.

The first step coefficient estimates are given by

$$\tilde{\beta}_d(t) = B^T(t)\gamma_d^{\left[\widehat{K_1}\right]}, \quad d = 0, \cdots, p-1. \tag{15}$$

Write $\hat{\varepsilon}_i = Y_i - \tilde{\mathbf{X}}_i\gamma^{\left[\widehat{K_1}\right]}, i = 1, \cdots, n$. The $m \times m$ covariance matrix $Cov(Y_i) \equiv \Sigma$ can be estimated by the following empirical estimator

$$\widehat{\sum} = \frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i\hat{\varepsilon}_i^T. \tag{16}$$

In the second step, the estimated correlation structure within repeated measurements is taken into account to form the weighted least squares loss function as follows:

$$\sum_{i=1}^{n}\left(Y_i - \tilde{\mathbf{X}}_i\gamma^\star\right)^T\hat{\Sigma}^{-1}\left(Y_i - \tilde{\mathbf{X}}_i\gamma^\star\right) \equiv \left(Y - \tilde{\mathbf{X}}\gamma^\star\right)^T W\left(Y - \tilde{\mathbf{X}}\gamma^\star\right), \tag{17}$$

where $W = \text{diag}\left(\hat{\Sigma}^{-1}, \cdots, \hat{\Sigma}^{-1}\right)$ is the estimated $(n \times m) \times (n \times m)$ weight matrix. Denote $\gamma^{\star\left[\widehat{K_2}\right]}$ to be the estimator of $\gamma^\star$ by sparse boosting with weighted loss function Eq. (17) being the loss function, where $\widehat{K_2}$ is the estimated stopping iterations in the second step. Then the coefficient estimates from the second step are given by

$$\hat{\beta}_d(t) = B^T(t)\gamma_d^{\star\left[\widehat{K_2}\right]}, \quad d = 0, \cdots, p-1. \tag{18}$$

The reliable estimates for the coefficient functions could then be obtained. More details about how to use sparse boosting to get $\gamma^{\left[\widehat{K_1}\right]}$ and $\gamma^{\star\left[\widehat{K_2}\right]}$ are provided in the following subsection.

3.1.2 Two-step sparse boosting techniques

gMDL can be adopted as the penalized empirical risk function to estimate the update criterion in each iteration and the stopping criterion. gMDL can be expressed in the following form:

$$\begin{aligned} \text{gMDL}(RSS, \text{trace}(\mathcal{B})) &= \log(F) + \frac{\text{trace}(\mathcal{B})}{n \times m} \log\left(\frac{Y^T Y - RSS}{\text{trace}(\mathcal{B}) \times F}\right), \\ F &= \frac{RSS}{n \times m - \text{trace}(\mathcal{B})}, \end{aligned} \tag{19}$$

where $\mathcal{B}$ is the boosting operator and RSS is the residual sum of squares.

The two-step sparse boosting approach is presented more specifically. In the first step, the start value of γ is set to zero vector, i.e. $\gamma^{[0]} = \mathbf{0}$, and in each of the k_1th iteration ($0 < k_1 \leq K_1$, and K_1 is the maximum number of iterations considered in the first step), the residual $R^{[k_1]} = Y - \tilde{\mathbf{X}}\gamma^{[k_1-1]}$ in present iteration is used to fit each of the dth component $\tilde{\mathbf{X}}_{,d} = \left(\tilde{\mathbf{X}}_{1,d}^T, \cdots, \tilde{\mathbf{X}}_{n,d}^T\right)^T, d = 0, \cdots, p-1$ by treating all the within-subject observations uncorrelated. Then the fit denoted by $\hat{\lambda}_d^{[k_1]}$ can be calculated by minimizing the squared loss function $\left(R^{[k_1]} - \tilde{\mathbf{X}}_{,d}\lambda\right)^T\left(R^{[k_1]} - \tilde{\mathbf{X}}_{,d}\lambda\right)$ with respect to λ. Therefore, the least squares estimate is $\hat{\lambda}_d^{[k_1]} = \left[\left(\tilde{\mathbf{X}}_{,d}\right)^T\left(\tilde{\mathbf{X}}_{,d}\right)\right]^{-1}\left(\tilde{\mathbf{X}}_{,d}\right)^T R^{[k_1]}$, the corresponding hat matrix is $\mathcal{H}_d = \left(\tilde{\mathbf{X}}_{,d}\right)\left[\left(\tilde{\mathbf{X}}_{,d}\right)^T\left(\tilde{\mathbf{X}}_{,d}\right)\right]^{-1}\left(\tilde{\mathbf{X}}_{,d}\right)^T$ and the residual sum of squares is $RSS_d^{[k_1]} = \left(R^{[k_1]} - \tilde{\mathbf{X}}_{,d}\hat{\lambda}_d^{[k_1]}\right)^T\left(R^{[k_1]} - \tilde{\mathbf{X}}_{,d}\hat{\lambda}_d^{[k_1]}\right)$. The chosen element $\hat{s}_{k_1}$ is attained by:

$$\hat{s}_{k_1} = \text{argmin}_{0 \leq d \leq p-1}\text{gMDL}\left(RSS_d^{[k_1]}, \text{trace}\left(\mathcal{B}_d^{[k_1]}\right)\right), \tag{20}$$

where $\mathcal{B}_d^{[1]} = \mathcal{H}_d$ and $\mathcal{B}_d^{[k_1]} = I - (I - \mathcal{H}_d)\left(I - \nu\mathcal{H}_{\hat{s}_{k_1-1}}\right).\cdots.(I - \nu\mathcal{H}_{\hat{s}_1})$ for $k_1 > 1$ is the first step boosting operator for choosing dth element in the k_1th iteration. Hence, there is an unique element $\tilde{\mathbf{X}}_{,\hat{s}_{k_1}}$ to be selected at each iteration, and only the corresponding coefficient vector $\gamma_{\hat{s}_{k_1}}^{[k_1]}$ changes, i.e., $\gamma_{\hat{s}_{k_1}}^{[k_1]} = \gamma_{\hat{s}_{k_1}}^{[k_1-1]} + \nu\hat{\lambda}_{\hat{s}_{k_1}}^{[k_1]}$, where ν is the pre-specified step-size parameter. All the other $\gamma_d^{[k_1]}$ for $d \neq \hat{s}_{k_1}$ keep unchanged. This procedure is repeated for K_1 times and the number of iterations K_1 can be estimated by

$$\widehat{K_1} = \text{argmin}_{1 \leq k_1 \leq K_1}\text{gMDL}\left(RSS_{\hat{s}_{k_1}}^{[k_1]}, \text{trace}\left(\mathcal{B}^{[k_1]}\right)\right), \tag{21}$$

where $\mathcal{B}^{[k_1]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_{k_1}}\right).\cdots.(I - \nu\mathcal{H}_{\hat{s}_1})$.

From the first step of sparse boosting, the estimator of γ is obtained by $\gamma^{\left[\widehat{K_1}\right]} = \left(\left(\gamma_0^{\left[\widehat{K_1}\right]}\right)^T, \cdots, \left(\gamma_{p-1}^{\left[\widehat{K_1}\right]}\right)^T\right)^T$. Then the weight matrix W can be easily obtained too.

In the second step, sparse boosting is used again by taking into account of the correlation structure estimator for the repeated measurements estimated in the first step. The initial value of $\gamma^{\star}$ is set to be the coefficient estimator from the first step of sparse boosting, i.e. $\gamma^{\star[0]} = \gamma^{\left[\widehat{K_1}\right]}$, and in each of the k_2th iteration ($0 < k_2 \le K_2$, and K_2 is the maximum number of iterations under consideration in the second step), the residual $R^{\star[k_2]} = Y - \tilde{\mathbf{X}}\gamma^{\star[k_2-1]}$ in current iteration is used to fit each of the dth working element $\tilde{\mathbf{X}}_{,d}, d = 0, \cdots, p-1$ by incorporating the within-subject correlation estimator from the first step. Then the fit denoted by $\hat{\lambda}_d^{\star[k_2]}$ can be obtained by minimizing the weighted squared loss function $\left(R^{\star[k_2]} - \tilde{\mathbf{X}}_{,d}\lambda\right)^T W\left(R^{\star[k_2]} - \tilde{\mathbf{X}}_{,d}\lambda\right)$ with respect to λ. Thus, the weighted least squares estimate is $\hat{\lambda}_d^{\star[k_2]} = \left[\left(\tilde{\mathbf{X}}_{,d}\right)^T W\left(\tilde{\mathbf{X}}_{,d}\right)\right]^{-1}\left(\tilde{\mathbf{X}}_{,d}\right)^T W R^{\star[k_2]}$, the corresponding hat matrix is $\mathcal{H}_d^{\star} = \left(\tilde{\mathbf{X}}_{,d}\right)\left[\left(\tilde{\mathbf{X}}_{,d}\right)^T W\left(\tilde{\mathbf{X}}_{,d}\right)\right]^{-1}\left(\tilde{\mathbf{X}}_{,d}\right)^T W$ and the weighted residual sum of squares is $RSS_d^{\star[k_2]} = \left(R^{\star[k_2]} - \tilde{\mathbf{X}}_{,d}\hat{\lambda}_d^{\star[k_2]}\right)^T W\left(R^{\star[k_2]} - \tilde{\mathbf{X}}_{,d}\hat{\lambda}_d^{\star[k_2]}\right)$. The chosen element $\hat{s}_{k_2}$ can be obtained by:

$$\hat{s}_{k_2} = \text{argmin}_{0 \le d \le p-1}\text{gMDL}\left(RSS_d^{\star[k_2]}, \text{trace}\left(\mathcal{B}_d^{\star[k_2]}\right)\right), \tag{22}$$

where $\mathcal{B}_d^{\star[1]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \mathcal{H}_d^{\star}\right)$ and $\mathcal{B}_d^{\star[k_2]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \mathcal{H}_d^{\star}\right)\left(I - \nu\mathcal{H}^{\star}_{\hat{s}_{k_2-1}}\right).\cdots.\left(I - \nu\mathcal{H}^{\star}_{\hat{s}_1}\right)$ for $k_2 > 1$ is the second step boosting operator for choosing dth element in the k_2th iteration. Thus, there is an unique element $\tilde{\mathbf{X}}_{,\hat{s}_{k_2}}$ to be selected at each time, and only the corresponding coefficient vector $\gamma_{\hat{s}_{k_2}}^{\star[k_2]}$ change, i.e., $\gamma_{\hat{s}_{k_2}}^{\star[k_2]} = \gamma_{\hat{s}_{k_2}}^{\star[k_2-1]} + \nu\hat{\lambda}_{\hat{s}_{k_2}}^{\star[k_2]}$. While all the other $\gamma_d^{\star[k_2]}$ for $d \ne \hat{s}_{k_2}$ remain the same. This procedure is repeated for K_2 times and the estimated stopping iterations $\widehat{K_2}$ is

$$\widehat{K_2} = \text{argmin}_{1 \le k_2 \le K_2}\text{gMDL}\left(RSS_{\hat{s}_{k_2}}^{\star[k_2]}, \text{trace}\left(\mathcal{B}^{\star[k_2]}\right)\right), \tag{23}$$

where $\mathcal{B}^{\star[k_2]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \nu\mathcal{H}^{\star}_{\hat{s}_{k_2}}\right).\cdots.\left(I - \nu\mathcal{H}^{\star}_{\hat{s}_1}\right)$.

From the second step of sparse boosting, the estimator of $\gamma^{\star}$ is arrived by $\gamma^{\star\left[\widehat{K_2}\right]} = \left(\left(\gamma_0^{\star\left[\widehat{K_2}\right]}\right)^T, \cdots, \left(\gamma_{p-1}^{\star\left[\widehat{K_2}\right]}\right)^T\right)^T$. The two-step sparse boosting algorithm for varying-coefficient model with longitudinal data can be summarized in the following form:

Two-step Sparse Boosting Algorithm with Longitudinal Data.

Step I: Use sparse boosting to estimate covariance matrix.

a. Initialization. Let $k_1 = 0$ and $\gamma_0^{[k_1]} = \mathbf{0}, \cdots, \gamma_{p-1}^{[k_1]} = \mathbf{0}$.

b. Increase k_1 by 1. Calculate $\hat{s}_{k_1} = \text{argmin}_{0 \le d \le p-1}\text{gMDL}\left(RSS_d^{[k_1]}, \text{trace}\left(\mathcal{B}_d^{[k_1]}\right)\right)$, where $\mathcal{B}_d^{[1]} = \mathcal{H}_d$ and $\mathcal{B}_d^{[k_1]} = I - (I - \mathcal{H}_d)\left(I - \nu\mathcal{H}_{\hat{s}_{k_1-1}}\right).\cdots.(I - \nu\mathcal{H}_{\hat{s}_1})$ for $k_1 > 1$.

c. Update. $\gamma_{\hat{s}_{k_1}}^{[k_1]} = \gamma_{\hat{s}_{k_1}}^{[k_1-1]}$ for $d \neq \hat{s}_{k_1}$ and $\gamma_{\hat{s}_{k_1}}^{[k_1]} = \gamma_{\hat{s}_{k_1}}^{[k_1-1]} + \nu\hat{\lambda}_{\hat{s}_{k_1}}^{[k_1]}$, where ν is the step-size parameter.

d. Iteration. Repeat step (b)-(c) for some large iteration number K_1.

e. Stopping. The optimal iteration number can be taken as $\widehat{K_1} = \text{argmin}_{1\le k_1\le K_1}\text{gMDL}\left(RSS_{\hat{s}_{k_1}}^{[k_1]}, \text{trace}\left(\mathcal{B}^{[k_1]}\right)\right)$, where $\mathcal{B}^{[k_1]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_{k_1}}\right)\cdot\cdots\cdot(I - \nu\mathcal{H}_{\hat{s}_1})$.

Thus, $\gamma^{\left[\widehat{K_1}\right]} = \left(\left(\gamma_0^{\left[\widehat{K_1}\right]}\right)^T, \cdots, \left(\gamma_{p-1}^{\left[\widehat{K_1}\right]}\right)^T\right)^T$ is the first step estimator for γ from sparse boosting and $\tilde{\beta}_d(t) = B^T(t)\gamma_d^{\left[\widehat{K_1}\right]}$, $d = 0, \cdots, p-1$ are the varying coefficient estimates ignoring the within-subject correlation. $Cov(Y_i)$ can be estimated by

$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \tilde{\mathbf{X}}_i\gamma^{\left[\widehat{K_1}\right]}\right)\left(Y_i - \tilde{\mathbf{X}}_i\gamma^{\left[\widehat{K_1}\right]}\right)^T.$$

Step II: Use sparse boosting again by incorporating covariance matrix estimator.

a. Initialization. Let $k_2 = 0$ and $\gamma^{\star[k_2]} = \gamma^{\left[\widehat{K_1}\right]}$.

b. Increase k_2 by 1. Calculate $\hat{s}_{k_2} = \text{argmin}_{0\le d\le p-1}\text{g MDL}\left(RSS_d^{\star[k_2]}, \text{trace}\left(\mathcal{B}_d^{\star[k_2]}\right)\right)$, where $\mathcal{B}_d^{\star[1]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \mathcal{H}_d^{\star}\right)$ and $\mathcal{B}_d^{\star[k_2]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \mathcal{H}_d^{\star}\right)\left(I - \nu\mathcal{H}_{\hat{s}_{k_2-1}}^{\star}\right)\cdot\cdots\cdot\left(I - \nu\mathcal{H}_{\hat{s}_1}^{\star}\right)$ for $k_2 > 1$.

c. Update. $\gamma_{\hat{s}_{k_2}}^{\star[k_2]} = \gamma_{\hat{s}_{k_2}}^{\star[k_2-1]}$ for $d \neq \hat{s}_{k_2}$ and $\gamma_{\hat{s}_{k_2}}^{\star[k_2]} = \gamma_{\hat{s}_{k_2}}^{\star[k_2-1]} + \nu\hat{\lambda}_{\hat{s}_{k_2}}^{\star[k_2]}$.

d. Iteration. Repeat step (b)-(c) for some large iteration number K_2.

e. Stopping. The optimal iteration number can be taken as $\widehat{K_2} = \text{argmin}_{1\le k_2\le K_2}\text{gMDL}\left(RSS_{\hat{s}_{k_2}}^{\star[k_2]}, \text{trace}\left(\mathcal{B}^{\star[k_2]}\right)\right)$, where $\mathcal{B}^{\star[k_2]} = I - \left(I - \mathcal{B}^{\left[\widehat{K_1}\right]}\right)\left(I - \nu\mathcal{H}_{\hat{s}_{k_2}}^{\star}\right)\cdot\cdots\cdot\left(I - \nu\mathcal{H}_{\hat{s}_1}^{\star}\right)$.

Therefore, $\gamma^{\star\left[\widehat{K_2}\right]} = \left(\left(\gamma_0^{\star\left[\widehat{K_2}\right]}\right)^T, \cdots, \left(\gamma_{p-1}^{\star\left[\widehat{K_2}\right]}\right)^T\right)^T$ and $\hat{\beta}_d(t) = B^T(t)\gamma_d^{\star\left[\widehat{K_2}\right]}$, $d = 0, \cdots, p-1$ are the final estimator for $\gamma^\star$ and varying coefficient estimates by the two-step sparse boosting. The final estimate for Y is $\hat{Y} = \tilde{\mathbf{X}}\gamma^{\star\left[\widehat{K_2}\right]}$.

3.2 Simulation

Simulation studies are conducted to evaluate the performance of the above two-step sparse boosting algorithm. The following four methods are compared in terms

of variable selection and function estimation performance. M1: two-step L2 boosting (use squared loss for update criterion and gMDL for stopping criterion); M2: two-step sparse boosting; M3: two-step lasso (performs lasso regression in the first step to calculate the estimated within-subject correlation structure using Eq. (14), and use lasso regression in the second step by taking into account of the estimated correlation structure) and M4: two-step elastic net regression (similar as M3 with the elastic net mixing parameter 0.5).

The simulation results from [18] show that all methods are able to identify important variables. However, in terms of sparsity, the two-step sparse boosting method preforms best with smallest number of false positives. Both penalization methods select much more irrelevant variables than boosting methods, with elastic net selects the most. For two-step sparse boosting, results of variable selection are quite stable from step I to step II but for the other approaches, the false positives and thus the sizes of model from step I to step II are expanding. Two-step sparse boosting yields smallest bias for the coefficients estimation among the competing methods. The refined estimates after incorporating the within-subject correlation generally perform better than the initial estimates without taking into account of the within-subject correlation since the two-step methods gain reduction of bias, especially when the within-subject correlation is high. In other words, the reduction of bias from step I to step II are much larger when the within-subject correlation is higher. This is intuitive as in the second step, the within-subject correlation structure estimated from the first step have been taken into account. The similar results obtained for the bias of the estimated covariance matrix. The bias under smaller within-subject correlation is smaller than under larger within-subject correlation. The two-step sparse boosting yields smaller bias of the estimated covariance matrix than other competing methods when the within-subject correlation is high. In summary, the performance of variable selection and functional coefficients estimation for two-step sparse boosting is quite satisfactory.

3.3 Yeast cell cycle gene expression data analysis

The cell cycle is one of the most important activities in life by which cells grow, replicate their chromosomes, undergo mitosis, and split into daughter cells. Thus, identifying cell cycle-regulated genes becomes very important. Adopting a model-based approach, Luan and Li [37] identified $n = 297$ cell cycle-regulated genes based on the α-factor synchronization experiments. All gene expression levels were measured at $m = 18$ different time points covering two cell-cycle periods. Using the same subset of the original data as in [38], a total $p = 96$ transcriptional factors (TFs) are included as predictors in the downstream analysis. Wei, Huang and Li [39] proved that the effects of the TFs on gene expression levels are time-dependent. After the independence screening by l^2-norm [40] to screen out the irrelevant predictors at first step, several methods can be used to identify the key TFs involved in gene regulation. Except two-step L2 boosting and two-step sparse boosting which take into account of the within-subject correlation in the second step, one-step L2 boosting and one-step sparse boosting which ignore the within-subject correlation are also considered for better comparison. Besides, some two-step penalized approaches are also considered: two-step lasso, two-step adaptive lasso and two-step elastic net (the elastic net mixing parameter 0.5).

The results from [18] show that boosting approaches yield sparser model than the penalized methods. Sparse boosting yields even sparser model and smaller errors in terms of estimation and prediction than L2 boosting. Two-step boosting achieves better performance than one-step boosting with smaller estimation and

prediction errors. Two-step sparse boosting method yields the most sparse model, with the smallest in-sample and out-of-sample prediction errors compared to other methods. In terms of the selected TFs, there is a significant overlap between two-step sparse boosting and each of the other methods. In conclusion, the two-step sparse boosting approach performs quite well in terms of variable selection, coefficients estimation and prediction and can provide useful information in identifying the important TFs that take part in the network of regulations.

4. Multi-step sparse boosting for subgroup identification

As personalized medicine is gaining popularity, identification of subgroups of the patients that can gain a higher efficacy from the treatment becomes greatly important. Recently, significant statistical approaches have been proposed to identify subgroups of patients who may be suitable for different treatments. Traditionally, subgroup identification is achieved by parametric partitioning approaches such as Bayesian approaches [41] or classification and regression tree (CART) [42]. Recently, recursive partitioning methods gain popularity since they achieve greater generalizability and efficiency. Such methods include MOB [43], PRIM [44], sequential-BATTing [45] and other non-parametric methods. For a detailed literature review of subgroup identification refer to Lipkovich et al. [46]. In this section, a sparse boosting based subgroup identification method is presented in the context of dense longitudinal data.

In particular, a formal subgroup identification method for high-dimensional dense longitudinal data is presented. It incorporates multi-step sparse boosting into the homogeneous pursuit via change point detection. Firstly, sparse boosting algorithm for individual modeling is first performed to obtain initial estimates. Then, change point detection via binary segmentation is used to identify the subgroup structure of patients. Lastly, the model on each identified subgroups is refitted and again sparse boosting is utilized to remove irrelevant predictors and yield reliable final estimates. The rest of the section is organized as follows. In Section 4.1, the subgroup model is formulated and a detailed method for subgroup identification and estimation is presented. In Section 4.2, the subgroup identification technique is evaluated through simulation studies. In Section 4.3, the feasibility and applicability of the approach is validated by studying a wallaby growth dataset.

4.1 Methodology

4.1.1 Patients model

Denote Y_{it} be the continuous measurement of the tth follow-up for patient i, where $i = 1, \cdots, n, t = 1, \cdots, T_i$. Let $\mathbf{X}_{it} = \left(X_{it,1}, \cdots, X_{it,p}\right)$ be the corresponding p-dimensional predictors. Assume n patients are independent. The following longitudinal model for the patients is considered:

$$Y_{it} = \tilde{\beta}_{i,0} + \sum_{j=1}^{p} X_{it,j}\tilde{\beta}_{i,j} + \varepsilon_{it}, \quad i = 1, \cdots, \quad n, \quad t = 1, \quad \cdots, \quad T_i. \tag{24}$$

where $\varepsilon_i = (\varepsilon_{i1}, \quad \cdots, \quad \varepsilon_{iT_i})^T, i = 1, \cdots, n$ are multivariate error terms with mean zero. Errors are assumed to be uncorrelated for different i, but components of ε_i are correlated with each other.

Moreover, the model is further assumed to have the following subgroup structure:

$$\tilde{\beta}_{i,j} = \begin{cases} \beta_{1,j} & when \quad i \in \Omega_{1,j} \\ \beta_{2,j} & when \quad i \in \Omega_{2,j} \\ \vdots & \vdots \\ \beta_{\mathcal{N}_j+1,j} & when \quad i \in \Omega_{\mathcal{N}_j+1,j} \end{cases} \tag{25}$$

The partition for regression coefficient $\left\{\tilde{\beta}_{i,j} : \quad 1 \le i \le n\right\}$ is $\left\{\Omega_{k,j} : \quad 1 \le k \le \mathcal{N}_j + 1\right\}$, which is unknown, and thus there are $\mathcal{N}_j + 1$ subgroups for the jth predictor. All patients are divided into at least $\max_j\left(\mathcal{N}_j + 1\right)$ and at most $\prod_{j=0}^{p}\left(\mathcal{N}_j + 1\right)$ subgroups by the model. The patients in the same subgroup share a similar relationship between the response and the predictors and have the same set of regression coefficients while different subgroups have different overall relationship between response and covariates. The main aim is to investigate the effects of the predictors on the response for different subgroups.

However, if the number of predictors under consideration is much larger than the number of patients and the number of follow-ups, a serious challenge may arise to estimate regression coefficients. Therefore, instead of adopting traditional methods (eg, MLE), sparse boosting method can be used to estimate the regression coefficients. With this, the dimensionality of features can be reduced and the coefficients of parameters can be obtained simultaneously.

4.1.2 Subgroup identification and estimation

Denote $\tilde{\beta}_i = \left(\tilde{\beta}_{i,0}, \ \cdots, \ \tilde{\beta}_{i,p}\right)^T$ and $\tilde{\beta} = \left(\tilde{\beta}_1^T, \ \cdots, \ \tilde{\beta}_n^T\right)^T$. Firstly, an initial estimator for $\tilde{\beta}_i$ is calculated for each subject i through sparse boosting approach using his or her own repeated measurements data; then, homogeneity pursuit via change point detection can be used to identify the change points among $\beta_{k,j}$s; lastly, the $\tilde{\beta}_i$s can be replaced by the identified subgroup structure, and the final estimator of regression coefficients can be obtained by the sparse boosting algorithm again. The steps for estimating $\tilde{\beta}_i$ is outlined as below.

In the first step, individualized modeling via sparse boosting is performed. For each of the ith individual, the initial coefficients $\tilde{\beta}_i$ can be estimated by minimizing the following least squares loss function:

$$\sum_{t=1}^{T_i}\left(Y_{it} - \tilde{\beta}_{i,0} - \sum_{j=1}^{p} X_{it,j}\tilde{\beta}_{i,j}\right)^2. \tag{26}$$

Let $Y_i = (Y_{i1}, \ \cdots, \ Y_{iT_i})^T, X_{it,0} = 1, X_{i,j} = \left(X_{i1,j}, \ \cdots, \ X_{iT_i,j}\right)^T, \mathbf{X}_i = \left(X_{i,0}, \ \cdots, \ X_{i,p}\right)$. Then the function Eq. (26) can be written in the matrix form:

$$\left(Y_i - \mathbf{X}_i\tilde{\beta}_i\right)^T\left(Y_i - \mathbf{X}_i\tilde{\beta}_i\right). \tag{27}$$

Denote $\tilde{\beta}_i^{\left[\hat{L}_i\right]} = \left(\tilde{\beta}_{i,0}^{\left[\hat{L}_i\right]} \ \cdots, \ \tilde{\beta}_{i,p}^{\left[\hat{L}_i\right]}\right)^T$ to be the estimator of $\tilde{\beta}_i$ by sparse boosting with Eq. (27) being loss function, where $\hat{L}_i$ is the estimated stopping iterations in

this step. This is the initial estimator of $\tilde{\beta}_i$. The detailed sparse boosting algorithm will be presented in the next subsection.

In the second step, homogeneity pursuit via change point detection is performed. Binary segmentation algorithm [47] is used to detect the change points among $\tilde{\beta}_{i,j}$, $i = 1, \cdots, n$ and to identify the subgroup structure. Let $\tilde{\beta}_{i,j}^{[\hat{L}_i]}$ be the $(j+1)$th component of $\tilde{\beta}_i^{[\hat{L}_i]}$. For the jth covariate, $\tilde{\beta}_{i,j}^{[\hat{L}_i]}$, $i = 1, \cdots, n$, are sorted in ascending order, and denoted by $b_{(1)} \le \cdots \le b_{(n)}$. Denote $r_{i,j}$ be the rank of $\tilde{\beta}_{i,j}^{[\hat{L}_i]}$.

For any $1 \le l_1 < l_2 \le n$, denote the scaled difference between the partial means of the first $\tau - l_1 + 1$ observations and the last $l_2 - \tau$ observations to be

$$H_{l_1 l_2}(\tau) = \sqrt{\frac{(l_2 - \tau)(\tau - l_1 + 1)}{l_2 - l_1 + 1}} \left(\frac{\sum_{i=\tau+1}^{l_2} b_{(l)}}{l_2 - \tau} - \frac{\sum_{i=l_1}^{\tau} b_{(i)}}{\tau - l_1 + 1} \right). \tag{28}$$

Denote δ to be the threshold, which is a tuning parameter and can be selected by AIC or BIC, then the binary segmentation algorithm is as follows:

1. Find $\hat{t}_1$ such that

$$H_{1,n}(\hat{t}_1) = \max_{1 \le \tau < n} H_{1,n}(\tau). \tag{29}$$

   If $H_{1,n}(\hat{t}_1) \le \delta$, there is no change points among $b_{(l)}$, $l = 1, \cdots, n$, and the change point detection process terminates. Otherwise, $\hat{t}_1$ is added to the set of change points and the region $\{\tau : 1 \le \tau \le n\}$ is divided into two subregions: $\{\tau : 1 \le \tau \le \hat{t}_1\}$ and $\{\tau : \hat{t}_1 + 1 \le \tau \le n\}$.

2. Find the change points in the two subregions derived in part (1), respectively. Consider the region $\{\tau : 1 \le \tau \le \hat{t}_1\}$ first. Find $\hat{t}_2$ such that

$$H_{1,\hat{t}_1}(\hat{t}_2) = \max_{1 \le \tau < \hat{t}_1} H_{1,\hat{t}_1}(\tau). \tag{30}$$

   If $H_{1,\hat{t}_1}(\hat{t}_2) \le \delta$, there is no change point in the region $\{\tau : 1 \le \tau \le \hat{t}_1\}$. Otherwise, add $\hat{t}_2$ to the set of change points and divide the region $\{\tau : 1 \le \tau \le \hat{t}_1\}$ into two subregions: $\{\tau : 1 \le \tau \le \hat{t}_2\}$ and $\{\tau : \hat{t}_2 + 1 \le \tau \le \hat{t}_1\}$. Similarly, for the region $\{\tau : \hat{t}_1 + 1 \le \tau \le n\}$, $\hat{t}_3$ can be found such that

$$H_{\hat{t}_1+1,n}(\hat{t}_3) = \max_{\hat{t}_1+1 \le \tau < n} H_{\hat{t}_1+1,n}(\tau). \tag{31}$$

   If $H_{\hat{t}_1+1,n}(\hat{t}_3) \le \delta$, there is no change point in the region $\{\tau : \hat{t}_1 + 1 \le \tau \le n\}$. Otherwise, add $\hat{t}_3$ to the set of change points and divide the region $\{\tau : \hat{t}_1 + 1 \le \tau \le n\}$ into two subregions: $\{\tau : \hat{t}_1 + 1 \le \tau \le \hat{t}_3\}$ and $\{\tau : \hat{t}_3 + 1 \le \tau \le n\}$.

3. For each subregion derived in part (2), the above algorithm is repeated for the subregion $\{\tau : 1 \le \tau \le \hat{t}_1\}$ or $\{\tau : \hat{t}_1 + 1 \le \tau \le n\}$ in part (2) until no change point is detected in any subregions.

The estimated locations for change points are sorted in increasing order and denoted by

$$\hat{t}_{(1)} < \hat{t}_{(2)} < \cdots < \hat{t}_{\left(\hat{\mathcal{N}}_j\right)}, \tag{32}$$

where $\hat{\mathcal{N}}_j$ is the number of detected change points and could be used to estimate $\mathcal{N}_j$. Further denote $\hat{t}_{(0)} = 0$, and $\hat{t}_{\left(\hat{\mathcal{N}}_j+1\right)} = n$. Let. $\hat{R}_{i,j} = \left\{\ell : \ \hat{t}_{(\ell-1)} < r_{i,j} \le \hat{t}_{(\ell)}\right\}$, $1 \le \ell \le \hat{\mathcal{N}}_{j+1}$, where $\left\{\hat{R}_{i,j} : \ 1 \le i \le n\right\}$ can be used to estimate the grouping index $\left\{R_{i,j} : \ 1 \le i \le n\right\}$. The above algorithm can be used to identify the change points for all $j = 0, \ \cdots, \ p$ and correspondingly obtain $\left\{\hat{R}_{i,j} : \ 1 \le i \le n, 0 \le j \le p\right\}$. Let $\left\{\hat{R}^{\star}_{\ell,j} : \ 1 \le \ell \le \hat{\mathcal{N}}, 0 \le j \le p\right\}$ = unique rows of $\left\{\hat{R}_{i,j} : \ 1 \le i \le n, 0 \le j \le p\right\}$, then $\hat{\mathcal{N}}$ is the estimated total number of subgroups for patients and the patients index in group ℓ is.

$$\hat{\Omega}_{\ell} = \left\{i : \ \hat{R}_{i,j} = \hat{R}^{\star}_{\ell,j}\right\}, \quad 1 \le \ell \le \hat{\mathcal{N}}. \tag{33}$$

All the coefficients $\tilde{\beta}_{i,j}$s in the same estimated subgroup $\hat{\Omega}_{\ell}$ are treated to be equal.

In the third step, subgroup modeling is performed by sparse boosting. Incorporating the patients structure identified in step 2, the model is refitted to each of the subgroups via sparse boosting with the following least squares loss function

$$\sum_{i \in \hat{\Omega}_{\ell}} \sum_{t=1}^{T_i} \left(Y_{it} - \tilde{\beta}_{i,0} - \sum_{j=1}^{p} X_{it,j} \tilde{\beta}_{i,j}\right)^2, \quad 1 \le \ell \le \hat{\mathcal{N}}. \tag{34}$$

Further denote $Y^{\star}_{\ell} = \left(Y^T_{\hat{\Omega}_{\ell}[1]}, \ \cdots, \ Y^T_{\hat{\Omega}_{\ell}\left[|\hat{\Omega}_{\ell}|\right]}\right)^T$, $X^{\star}_{\ell,j} = \left(X^T_{\hat{\Omega}_{\ell}[1],j}, \ \cdots, \ X^T_{\hat{\Omega}_{\ell}\left[|\hat{\Omega}_{\ell}|\right],j}\right)^T$, $\mathbf{X}^{\star}_{\ell} = \left(X^{\star}_{\ell,0}, \ \cdots, \ X^{\star}_{\ell,p}\right)$ and $\tilde{\beta}^{\star}_{\ell} = \left(\tilde{\beta}^T_{\hat{\Omega}_{\ell}[1]}, \ \cdots, \ \tilde{\beta}^T_{\hat{\Omega}_{\ell}\left[|\hat{\Omega}_{\ell}|\right]}\right)^T$ for $\ell = 1, \ \cdots, \ \hat{\mathcal{N}}$, where $\hat{\Omega}_{\ell}[i]$ is the ith element of $\hat{\Omega}_{\ell}$ and $|\hat{\Omega}_{\ell}|$ is the number of elements in $\hat{\Omega}_{\ell}$. The function Eq. (34) can be written in the matrix form:

$$\left(Y^{\star}_{\ell} - \mathbf{X}^{\star}_{\ell} \tilde{\beta}^{\star}_{\ell}\right)^T \left(Y^{\star}_{\ell} - \mathbf{X}^{\star}_{\ell} \tilde{\beta}^{\star}_{\ell}\right), \quad 1 \le \ell \le \hat{\mathcal{N}}. \tag{35}$$

Denote $\tilde{\beta}^{\star\left[\hat{L}^{\star}_{\ell}\right]}_{\ell}$ to be the estimate for $\tilde{\beta}^{\star}_{\ell}$ by sparse boosting with Eq. (35) being the loss function, where $\hat{L}^{\star}_{\ell}$ is the estimated number of stopping iterations in this step. The estimator for coefficient $\tilde{\beta}_i$ is

$$\hat{\beta}_i = \left\{\tilde{\beta}^{\star\left[\hat{L}^{\star}_{\ell}\right]}_{\ell} \ \text{for} \, i \in \hat{\Omega}_{\ell}\right\}, \quad 1 \le i \le n. \tag{36}$$

More details about how to use sparse boosting to obtain $\left\{\tilde{\beta}^{\left[\hat{L}_i\right]}_i, 1 \le i \le n\right\}$ and $\left\{\tilde{\beta}^{\star\left[\hat{L}^{\star}_{\ell}\right]}_{\ell}, 1 \le \ell \le \hat{\mathcal{N}}\right\}$ are given in the following subsection.

4.1.3 Multi-step sparse boosting techniques

gMDL can be used as the penalized empirical risk function to estimate the update criterion in each iteration and the stopping criterion to avoid the selection of the tuning parameter. gMDL can be expressed in the following form:

$$\begin{aligned} \text{gMDL}(Y, RSS, \text{trace}(\mathcal{B})) = & \log(F) + \frac{\text{trace}(\mathcal{B})}{|Y|} \log\left(\frac{Y^T Y - RSS}{\text{trace}(\mathcal{B}) \times F}\right), \\ F = & \frac{RSS}{|Y| - \text{trace}(\mathcal{B})}, \end{aligned} \tag{37}$$

where Y is the vector of response variable, $|Y|$ is the length of Y, $\mathcal{B}$ is the boosting operator and RSS is the residual sum of squares.

The sparse boosting procedure is described in details. The starting value of $\tilde{\beta}_i$ is set to zero vector, i.e. $\tilde{\beta}_i^{[0]} = 0$, and in each of the l_ith iteration ($0 < l_i \leq L_i$, and L_i is the maximum number of iterations considered in this step), the residual $R^{[l_i]} = Y_i - \mathbf{X}_i \tilde{\beta}_i^{[l_i-1]}$ in present iteration is used to fit each of the jth element $X_{i,j}, j = 0, \cdots, p$. The fit denoted by $\hat{\lambda}_j^{[l_i]}$ can be obtained by minimizing the squared loss function $\left(R^{[l_i]} - X_{i,j}\lambda\right)^T \left(R^{[l_i]} - X_{i,j}\lambda\right)$ with respect to λ. Thus, the least squares estimate is $\hat{\lambda}_j^{[l_i]} = \left[\left(X_{i,j}\right)^T \left(X_{i,j}\right)\right]^{-1} \left(X_{i,j}\right)^T R^{[l_i]}$, the corresponding hat matrix is $\mathcal{H}_j = \left(X_{i,j}\right)\left[\left(X_{i,j}\right)^T \left(X_{i,j}\right)\right]^{-1} \left(X_{i,j}\right)^T$ and the residual sum of squares is $RSS_j^{[l_i]} = \left(R^{[l_i]} - X_{i,j}\hat{\lambda}_j^{[l_i]}\right)^T \left(R^{[l_i]} - X_{i,j}\hat{\lambda}_j^{[l_i]}\right)$. The selected entry $\hat{s}_{l_i}$ is obtained by:

$$\hat{s}_{l_i} = \text{argmin}_{0 \leq j \leq p} \text{gMDL}\left(Y_i, RSS_j^{[l_i]}, \text{trace}\left(\mathcal{B}_j^{[l_i]}\right)\right), \tag{38}$$

where $\mathcal{B}_j^{[1]} = \mathcal{H}_j$ and $\mathcal{B}_j^{[l_i]} = I - \left(I - \mathcal{H}_j\right)\left(I - \nu\mathcal{H}_{\hat{s}_{l_i-1}}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$ for $l_i > 1$ is the boosting operator for choosing jth entry in the l_ith iteration in this step. Hence, there is an unique element $X_{i,\hat{s}_{l_i}}$ to be selected at each iteration, and only the corresponding coefficient vector $\tilde{\beta}_{i,\hat{s}_{l_i}}^{[l_i]}$ changes, i.e., $\tilde{\beta}_{i,\hat{s}_{l_i}}^{[l_i]} = \tilde{\beta}_{i,\hat{s}_{l_i}}^{[l_i-1]} + \nu\hat{\lambda}_{\hat{s}_{l_i}}^{[l_i]}$, where ν is the pre-specified step-size parameter. All the other $\tilde{\beta}_{i,j}^{[l_i]}$ for $j \neq \hat{s}_{l_i}$ keep unchanged. This procedure is repeated for L_i times and the number of iterations L_i can be estimated by

$$\hat{L}_i = \text{argmin}_{1 \leq l_i \leq L_i} \text{gMDL}\left(Y_i, RSS_{\hat{s}_{l_i}}^{[l_i]}, \text{trace}\left(\mathcal{B}^{[l_i]}\right)\right), \tag{39}$$

where $\mathcal{B}^{[l_i]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_{l_i}}\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}\right)$.

From the above sparse boosting approach, the estimator of $\tilde{\beta}_i$ is $\tilde{\beta}_i^{\left[\hat{L}_i\right]} = \left(\tilde{\beta}_{i,0}^{\left[\hat{L}_i\right]}, \cdots, \tilde{\beta}_{i,p}^{\left[\hat{L}_i\right]}\right)^T, i = 1, \cdots, n$. Then the subgroup structure can be obtained by homogeneity pursuit via change point detection.

Next, sparse boosting is used again for each estimated subgroups. The starting value of $\tilde{\beta}_\ell^\star$ is set to zero vector, i.e. $\tilde{\beta}_\ell^{\star[0]} = 0$, and in each of the $l_\ell^\star$th iteration ($0 < l_\ell^\star \leq L_\ell^\star$, and $L_\ell^\star$ is the maximum number of iterations considered in this stage), the residual $R^{\star\left[l_\ell^\star\right]} = Y_\ell^\star - \mathbf{X}_\ell^\star \tilde{\beta}_\ell^{\star\left[l_\ell^\star - 1\right]}$ in present iteration is used to fit each of the jth component $X_{\ell,j}^\star, j = 0, \cdots, p$. Then the fit denoted by $\hat{\lambda}_j^{\star\left[l_i^\star\right]}$ can be calculated by minimizing the squared loss function $\left(R^{\star\left[l_\ell^\star\right]} - X_{\ell,j}^\star\lambda\right)^T \left(R^{\star\left[l_\ell^\star\right]} - X_{i,j}^\star\lambda\right)$ with respect

to λ. Therefore, the least squares estimate is $\hat{\lambda}_j^{\star[l_\ell^\star]} = \left[\left(X_{\ell,j}^\star\right)^T\left(X_{\ell,j}^\star\right)\right]^{-1}\left(X_{\ell,j}^\star\right)^T R^{\star[l_\ell^\star]}$, the corresponding hat matrix is $\mathcal{H}_j^\star = \left(X_{\ell,j}^\star\right)\left[\left(X_{\ell,j}^\star\right)^T\left(X_{\ell,j}^\star\right)\right]^{-1}\left(X_{\ell,j}^\star\right)^T$ and the residual sum of squares is $RSS_j^{\star[l_\ell^\star]} = \left(R^{\star[l_\ell^\star]} - X_{\ell,j}^\star\hat{\lambda}_j^{[l_\ell^\star]}\right)^T\left(R^{\star[l_\ell^\star]} - X_{\ell,j}^\star\hat{\lambda}_j^{[l_\ell^\star]}\right)$. The chosen element $\hat{s}_{l_\ell^\star}^\star$ is attained by:

$$\hat{s}_{l_\ell^\star}^\star = \text{argmin}_{0\le j\le p}\text{gMDL}\left(Y_\ell^\star, RSS_j^{\star[l_\ell^\star]}, \text{trace}\left(\mathcal{B}_j^{\star[l_\ell^\star]}\right)\right), \tag{40}$$

where $\mathcal{B}_j^{\star[1]} = \mathcal{H}_j^\star$ and $\mathcal{B}_j^{\star[l_\ell]} = I - \left(I - \mathcal{H}_j^\star\right)\left(I - \nu\mathcal{H}_{\hat{s}_{l_\ell^\star-1}}^\star\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}^\star\right)$ for $l_\ell^\star > 1$ is the boosting operator for choosing jth element in the $l_\ell^\star$th iteration in this stage. Hence, there is an unique element $X_{\ell,\hat{s}_{l_\ell^\star}}^\star$ to be selected at each iteration, and only the corresponding coefficient vector $\tilde{\beta}_{\ell,\hat{s}_{l_\ell^\star}}^{\star[l_\ell^\star]}$ changes, i.e., $\tilde{\beta}_{\ell,\hat{s}_{l_\ell^\star}}^{[l_\ell]} = \tilde{\beta}_{\ell,\hat{s}_{l_\ell^\star}}^{\star[l_\ell^\star-1]} + \nu\hat{\lambda}_{\hat{s}_{l_\ell^\star}}^{[l_\ell^\star]}$, where ν is the pre-specified step-size parameter. All the other $\tilde{\beta}_{\ell,j}^{\star[l_\ell^\star]}$ for $j \ne \hat{s}_{l_\ell^\star}$ keep unchanged. This procedure is repeated for $L_\ell^\star$ times and the number of iterations $L_\ell^\star$ can be estimated by

$$\hat{L}_i^\star = \text{argmin}_{1\le l_\ell\le L_i^\star}\text{gMDL}\left(Y_\ell^\star, RSS_{\hat{s}_{l_\ell^\star}}^{\star[l_\ell^\star]}, \text{trace}\left(\mathcal{B}^{\star[l_\ell^\star]}\right)\right), \tag{41}$$

where $\mathcal{B}^{\star[l_\ell^\star]} = I - \left(I - \nu\mathcal{H}_{\hat{s}_{l_\ell^\star}}^\star\right).\cdots.\left(I - \nu\mathcal{H}_{\hat{s}_1}^\star\right)$.

From the second step of sparse boosting, the estimator of $\tilde{\beta}_\ell$ is $\tilde{\beta}_\ell^{\star\left[\hat{L}_\ell^\star\right]} = \left(\tilde{\beta}_{\ell,0}^{\star\left[\hat{L}_\ell^\star\right]}, \cdots, \tilde{\beta}_{\ell,p}^{\star\left[\hat{L}_\ell^\star\right]}\right)^T, \ell = 1, \cdots, \hat{\mathcal{N}}$.

4.2 Simulation

Extensive simulations are conducted to evaluate the performance of the proposed procedure. The accuracy of subgrouping, feature selection, coefficients estimation and prediction are assessed in the setting of different number of patients and repeated measurements. To understand the advantage of the proposed method better, the following four approaches are also considered. M1: the homogeneous model fitting method which treats all patients as one group and use sparse boosting for the single model to estimate $\tilde{\beta}$; M2: the heterogeneous model fitting method which uses initial pre-grouping estimate $\tilde{\beta}_i^{\left[\hat{L}_i\right]}$ as the final estimate of $\tilde{\beta}_i$; M3: same as the proposed method but in step 2, instead of detecting the change points for coefficients of each covariate $\tilde{\beta}_{i,j}^{\left[\hat{L}_i\right]}, i = 1, \cdots, n$ for $j = 0, \cdots, p$, it detects the change points among $\left(\tilde{\beta}_1^{T\left[\hat{L}_1\right]}, \cdots, \tilde{\beta}_n^{T\left[\hat{L}_n\right]}\right)^T$ similarly to Ke et al. [48]; M4: the proposed method.

The results from [19] show that the naive homogeneous model fitting method M1 can rarely identify the important covariates while the over-parameterized model fitting method M2 and other two methods (M3 & M4) which identify subgroup structures consistently yield true positives equal to the true number of important covariates. Compared these three methods which can identify the important covariates, the proposed method produces smallest false positives. In addition, the number of false positives is decreasing when there is an increase in cluster size. Neither the homogeneous model fitting method nor heterogeneous model fitting method is able to identify the true structure among patients. The method M3 produces much more subgroups than it really has, while the proposed method M4 identified the number of subgroups closest to the actual number of subgroups. Furthermore, the probability of identifying the true subgroups becomes larger when the number of repeated measurements increases. For in-sample prediction, the over-parameterized model M2 performs the best while the methods M3 & M4 performs very competitively. However, for out-of-sample prediction, method M4 is the best. M1 is inferior to M4, yielding poor results of estimation and prediction. In summary, the proposed method preforms pretty well in terms of subgroup identification, variable selection, estimation as well as prediction.

4.3 Wallaby growth data analysis

The proposed subgroup identification method is applied to wallaby growth data, which is from the Australian Dataset and Story Library (OzDASL) and can be found at http://www.statsci.org/data/oz/wallaby.html. The data set has 77 Tammar wallabies' growth measurements which were taken longitudinally. The response variable is the weight of wallabies (tenths of a gram). The predictors involve length of head, ear, body, arm, leg, tail, foot and their second order interactions. Therefore, a total of 35 predictors are included in the analysis. After removing the missing data, 43 Tammar wallabies are kept in our dataset. The number of repeated measurements ranges from 9 to 34 (median: 23). To have a better understanding of the wallabies' growth trend, the questions of which parts of body would affect the weight and whether the length of each body parts have the same effects on the weight for all wallabies are investigated, i.e. is there any subgroups among wallabies. Except the above subgroup identification method (SB-CPD1), the other 3 methods studied in simulation are also considered, i.e. homogeneous model fitting method (SB-Homogeneous), heterogeneous model fitting method (SB-Heterogeneous) and the method similar to SB-CPD1 but identifying subgroups via other method in Ke et al. [48] (SB-CPD2). In addition, the following subgroup identification methods incorporating penalized methods are also investigated: similar to our proposed method but instead of using sparse boosting, lasso (Lasso-CPD1), elastic net (ElasticNet-CPD1), SCAD (SCAD-CPD1) or MCP (MCP-CPD1) is used.

The results from [19] show that although Lasso-CPD1 and ElasticNet-CPD1 yield smaller in-sample prediction error by keeping all 35 covariates, they have relatively large out-of-sample prediction errors due to over-fitting problem. The subgroup identification method via sparse boosting keeps smaller number of predictors, achieves sparser model than penalized methods. The proposed method SB-CPD1 identifies smaller number of subgroups and predictors than alternative competing methods while produces smallest out-of-sample prediction errors. In conclusion, the proposed subgroup identification method provides a more precise definition for various subgroups. It may also result in a more accurate medical decision making for these subjects.

5. Conclusions

In this chapter, we discussed various sparse boosting based machine learning methods in the context of high-dimensional data problems. Specifically, we presented the sparse boosting procedure and two-step sparse boosting procedure for nonparametric varying-coefficient models with survival data and repeatedly measured longitudinal data respectively to simultaneously perform variable selection and estimation of functional coefficients. We further presented the multi-step sparse boosting based subgroup identification method with longitudinal patient data to identify subgroups that exhibit different treatment effects. The extensive numerical studies show the validity and effectiveness of our proposed methods and the real data analysis further demonstrate their usefulness and advantages.

Author details

Mu Yue
Singapore University of Technology and Design, Singapore

*Address all correspondence to: yuemu.moon@gmail.com

References

[1] Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological). 1996 Jan;58(1):267–288.

[2] Fan J, Li R. Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American statistical Association. 2001 Dec 1;96(456):1348–1360.

[3] Zhang CH. Nearly unbiased variable selection under minimax concave penalty. The Annals of statistics. 2010; 38(2):894–942.

[4] Zou H. The adaptive lasso and its oracle properties. Journal of the American statistical association. 2006 Dec 1;101(476):1418–1429.

[5] Zou H, Hastie T. Regularization and variable selection via the elastic net. Journal of the royal statistical society: series B (statistical methodology). 2005 Apr 1;67(2):301–320.

[6] Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2006 Feb;68(1):49–67.

[7] Schapire RE. The strength of weak learnability. Machine learning. 1990 Jun 1;5(2):197–227.

[8] Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences. 1997 Aug 1;55(1):119–139.

[9] Bühlmann P, Yu B. Boosting with the L 2 loss: regression and classification. Journal of the American Statistical Association. 2003 Jun 1;98(462):324–339.

[10] Bühlmann P, Yu B, Singer Y, Wasserman L. Sparse Boosting. Journal of Machine Learning Research. 2006 Jun 1;7(6).

[11] Wang Z. HingeBoost: ROC-based boost for classification and variable selection. The International Journal of Biostatistics. 2011 Feb 4;7(1).

[12] Bühlmann P, Hothorn T. Twin boosting: improved feature selection and prediction. Statistics and Computing. 2010 Apr;20(2):119–138.

[13] Komori O, Eguchi S. A boosting method for maximizing the partial area under the ROC curve. BMC bioinformatics. 2010 Dec;11(1):1–7.

[14] Wang Z. Multi-class hingeboost. Methods of information in medicine. 2012;51(02):162–167.

[15] Zhao J. General sparse boosting: improving feature selection of l2 boosting by correlation-based penalty family. Communications in Statistics-Simulation and Computation. 2015 Jul 3; 44(6):1612–1640.

[16] Yang Y, Zou H. Nonparametric multiple expectile regression via ER-Boost. Journal of Statistical Computation and Simulation. 2015 May 3;85(7):1442–1458.

[17] Yue M, Li J, Ma S. Sparse boosting for high-dimensional survival data with varying coefficients. Statistics in medicine. 2018 Feb 28;37(5):789–800.

[18] Yue M, Li J, Cheng MY. Two-step sparse boosting for high-dimensional longitudinal data with varying coefficients. Computational Statistics Data Analysis. 2019 Mar 1;131:222–234.

[19] Yue M, Huang L. A new approach of subgroup identification for high-dimensional longitudinal data. Journal of Statistical Computation and

Simulation. 2020 Jul 23;90(11):2098–2116.

[20] David CR. Regression models and life tables (with discussion). Journal of the Royal Statistical Society. 1972;34(2): 187–220.

[21] Wei LJ. The accelerated failure time model: a useful alternative to the Cox regression model in survival analysis. Statistics in medicine. 1992;11(14–15): 1871–1879.

[22] Schmid M, Hothorn T. Flexible boosting of accelerated failure time models. BMC bioinformatics. 2008 Dec; 9(1):1–3.

[23] Wang Z, Wang CY. Buckley-James boosting for survival analysis with high-dimensional biomarker data. Statistical Applications in Genetics and Molecular Biology. 2010 Jun 8;9(1).

[24] Li J, Ma S. Survival analysis in medicine and genetics. CRC Press; 2013 Jun 4.

[25] Stute W. Consistent estimation under random censorship when covariables are present. Journal of Multivariate Analysis. 1993 Apr 1;45(1): 89–103.

[26] Curry HB, Schoenberg IJ. On Pólya frequency functions IV: the fundamental spline functions and their limits. InIJ Schoenberg Selected Papers 1988 (pp. 347–383). Birkhäuser, Boston, MA.

[27] Hansen MH, Yu B. Model selection and the principle of minimum description length. Journal of the American Statistical Association. 2001 Jun 1;96(454):746–774.

[28] Shedden K, Taylor JM, Enkemann SA, Tsao MS, Yeatman TJ, Gerald WL, Eschrich S, Jurisica I, Giordano TJ, Misek DE, Chang AC. Gene expression–based survival prediction in lung adenocarcinoma: a multi-site, blinded validation study. Nature medicine. 2008 Aug;14(8):822.

[29] Consonni D, Bertazzi PA, Zocchetti C. Why and how to control for age in occupational epidemiology. Occupational and environmental medicine. 1997 Nov 1;54(11):772–776.

[30] Wang S, Nan B, Zhu J, Beer DG. Doubly penalized Buckley–James method for survival data with high-dimensional covariates. Biometrics. 2008 Mar;64(1):132–140.

[31] Diggle P, Heagerty P, Liang KY, Zeger S. Analysis of Longitudinal Data: Oxford University Press. 2002.

[32] Fitzmaurice GM, Laird NM, Ware JH. Applied longitudinal analysis, vol. 998 John Wiley & Sons. Hoboken NJ. 2012.

[33] Lin X, Carroll RJ. Semiparametric regression for clustered data using generalized estimating equations. Journal of the American Statistical Association. 2001 Sep 1;96(455):1045–1056.

[34] Fan J, Huang T, Li R. Analysis of longitudinal data with semiparametric estimation of covariance function. Journal of the American Statistical Association. 2007 Jun 1;102(478):632–641.

[35] Cheng MY, Honda T, Li J, Peng H. Nonparametric independence screening and structure identification for ultra-high dimensional longitudinal data. Annals of Statistics. 2014;42(5):1819–1849.

[36] Cheng MY, Honda T, Li J. Efficient estimation in semivarying coefficient models for longitudinal/clustered data. The Annals of Statistics. 2016;44(5): 1988–2017.

[37] Luan Y, Li H. Clustering of time-course gene expression data using a mixed-effects model with B-splines.

Bioinformatics. 2003 Mar 1;19(4): 474–482.

[38] Wang L, Chen G, Li H. Group SCAD regression analysis for microarray time course gene expression data. Bioinformatics. 2007 Jun 15;23(12): 1486–1494.

[39] Wei F, Huang J, Li H. Variable selection and estimation in high-dimensional varying-coefficient models. Statistica Sinica. 2011 Oct 1;21(4):1515.

[40] Yue M, Li J. Improvement screening for ultra-high dimensional data with censored survival outcomes and varying coefficients. The international journal of biostatistics. 2017 May 18;13(1).

[41] Sivaganesan S, MÃ¼ller P, Huang B. Subgroup finding via Bayesian additive regression trees. Statistics in medicine. 2017 Jul 10;36(15):2391–2403.

[42] Zhang H, Singer BH. Recursive partitioning and applications. Springer Science & Business Media; 2010 Jul 1.

[43] Zeileis A, Hothorn T, Hornik K. Model-based recursive partitioning. Journal of Computational and Graphical Statistics. 2008 Jun 1;17(2):492–514.

[44] Chen G, Zhong H, Belousov A, Devanarayan V. A PRIM approach to predictive-signature development for patient stratification. Statistics in medicine. 2015 Jan 30;34(2):317–342.

[45] Huang X, Sun Y, Trow P, Chatterjee S, Chakravartty A, Tian L, Devanarayan V. Patient subgroup identification for clinical drug development. Statistics in medicine. 2017 Apr 30;36(9):1414–1428.

[46] Lipkovich I, Dmitrienko A, B D'Agostino Sr R. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. Statistics in medicine. 2017 Jan 15; 36(1):136–196.

[47] Bai J. Estimating multiple breaks one at a time. Econometric theory. 1997 Jun 1:315–352.

[48] Ke Y, Li J, Zhang W. Structure identification in panel data analysis. Annals of Statistics. 2016;44(3):1193–1233.