SN Computer Science (2022) 3:186
https://doi.org/10.1007/s42979-022-01072-6

ORIGINAL RESEARCH

Boosting for Multivariate Longitudinal Responses

Amol Pande[1] · Hemant Ishwaran[2] · Eugene Blackstone[1]

Received: 24 January 2022 / Accepted: 13 February 2022

Abstract

Boosting, a machine learning approach, has gained popularity over the years in its application to various types of data, including longitudinal data. However, its application to data involving multivariate responses is limited. In this article, we present a new approach where we apply gradient boosting, a generic form of boosting, to model multivariate longitudinal responses. Our approach can handle time-varying covariates as well as high dimensionality of covariates and responses when some of the covariates and responses are pure noise. A key feature of our approach is that it is designed to select covariates that affect responses differently at different time intervals; thereby, an overall effect of any covariate can be dissected and represented as a function of time. A novel feature of our approach is that, in addition to covariate selection, we also perform response selection for different time intervals. This helps to identify and order responses based on their importance for a given time interval. Simulation results show that the prediction performance of our approach does not deteriorate in high dimensionality and can approximate the true model. Application of our approach to a clinical laboratory data evaluates the behavior of bilirubin and creatinine for the heart failure patients before and after the heart transplant, and identifies important risk factors that affect their behavior. Our approach can be implemented using the R package `BoostMLR`

Keywords Multivariate longitudinal responses · Gradient boosting · *B*-spline · Variable importance · Response selection

Introduction

Longitudinal data is a special type of data in which, for a given subject, response is measured repeatedly over a period of time. Some covariates are measured only at baseline (i.e., at the beginning of the study), while others are measured along with the response [1]. Covariates measured at baseline are referred to as time-invariant covariates, whereas covariates measured along with the response, over a period of time, are referred to as time-varying covariates. Studies with time-varying covariates are more informative because they provide concurrent effects of covariates on the response.

In some longitudinal studies, investigator collects multiple responses, collectively referred as multivariate longitudinal responses, to model them jointly. In such studies, the aims are as follows:

1. Jointly model multiple responses over time.
2. Find covariates that affect most of the responses. This happens in situations where responses collectively measure an underlying characteristic. For example, in type 2 diabetic patients, the measurements of glucose and insulin can be used to understand the progression of the disease.
3. Study how covariates affect one response in the presence of other responses.
4. Predict trajectories of multiple responses for new subjects based on their covariate data.

As an example consider the laboratory data for heart failure (HF) patients. HF is a serious condition in which heart cannot pump enough blood to various body parts to meet their requirement. Risk factors for HF include older age,

Hemant Ishwaran and Eugene Blackstone contributed equally to this work.

✉ Amol Pande
amoljpande@gmail.com

Hemant Ishwaran
hemant.ishwaran@gmail.com

Eugene Blackstone
blackse@ccf.org

[1] Heart,Vascular and Thoracic Institute, Cleveland Clinic, 9500 Euclid Avenue, Cleveland, OH 44016, USA

[2] Division of Biostatistics, University of Miami, 1120 NW 14th St, Miami, FL 33136, USA

Published online: 09 March 2022

hypertension, coronary heart disease, and diabetes. Treatment of HF depends on type, stage and cause of HF. Patients with advanced condition of HF, heart transplant is the most effective treatment option. Patients who require heart transplant are put on mechanical circulatory support (MCS) through a device implantation while they wait for availability of donors. In such patients, it is important to monitor their heart condition periodically. Impaired liver and renal functions provide indications of advanced stage of heart failure [2–4]. Therefore, monitoring patient's liver and renal functions before and after the heart transplant is a crucial part of patient management. Bilirubin and creatinine, respectively, represent good indicators of liver and renal functions. The laboratory data consists of patients' bilirubin and creatinine levels measured after the device implantation repeatedly over a period of time. Joint modeling of bilirubin and creatinine allows the investigator to estimate their trajectories over time to identify critical levels of bilirubin and creatinine in these high risk patients. It also allows the investigator to evaluate their risk factors.

Earlier work of modeling longitudinal data was focused on parametric models. Two such parametric models often used in the literature are marginal model [5] and mixed effect model [6]. Extensions of these models are available for multivariate longitudinal responses. For example, an extension of the marginal model for multivariate longitudinal responses is provided by [7, 8] and an extension of the mixed effect model is provided by [9]. Typically, these models are limited in their applicability because they assume a linear functional form of covariate and covariate-time interaction. To address non-linearity, a non-linear model is available [10]. However, in such a model, the user needs to explicitly specify a non-linear functional form for each covariate, which may be unknown. There are some non-parametric models available for modeling longitudinal data where the relationships of covariate and time with response are defined using an unspecified functions, which are estimated from the data (referred to as data-adaptive functions) [11, 12]. Estimation of data-adaptive function is computationally intensive even in low dimensional covariate situation, and often impossible when the dimension of covariates is large. Some of the estimation procedures proposed in the literature can be found in [13–15]; also, see a review by [16]. Tree-based approaches, for example RE-EM trees [17], and their extension to ensemble approaches such as bagging and random forest, and modern approach such as generalized neural network mixed model [18] can be applied to high dimensional longitudinal data. However, these approaches are not yet generalized to multivariate longitudinal responses. One of the well recognized non-parametric approaches is the generalized additive mixed model (GAMM) [19]. This approach can be implemented using the R package `mgcv`, which can handle multivariate longitudinal responses.

There exist a special case of non-parametric model that has drawn a substantial interest because of the flexibility of interpretation of its coefficients. This model is referred to as a time-varying coefficient model [20–22]. As the name suggests, in this model, the coefficient, which represents the relationship between response and covariate, itself is a function of time; hence it provides a time-varying relationship. Such model is preferred over the fixed-coefficient model discussed above because, instead of providing an overall relationship, it provides a relationship between response and covariate which is varying over time. For example, in case of the laboratory data, the investigator might be interested to know if the baseline bilirubin and creatinine have any effect on bilirubin and creatinine levels after the device implantation (i.e., whether patients with impaired baseline levels have different trajectories from those with normal baseline levels), and if so, does that effect last for longer duration or only at the beginning of the follow-up?

In this article, we present a new approach for joint modeling of multivariate longitudinal responses. Our approach uses data-adaptive functions for modeling non-linearities among covariates and time. Below we describe some salient features of our approach and briefly describe our contribution to these features.

1. In our approach, covariates can be time-invariant or time-varying. Although time-varying covariates can also be handled by some of the well known methods described above, it is worth mentioning here because it is relevant when we combine it with the next feature.
2. Our approach is designed to identify covariates that affect responses differently at different time intervals. This idea is helpful to dissect an overall effect of covariate on the response into different time intervals. For example, some covariates affect response at the beginning of the follow-up, whereas others at a later stage [23]. This feature can work for time-invariant and time-varying covariates, however, this is more effective for time-varying covariates because it can provide a concurrent effect of covariate on the response. For example, when we use time-invariant covariate (say covariate measured at baseline) and plot the relationship of covariate and response across time, this relationship is represented as a function of time. On the other hand, for time-varying covariate, the same relationship can be represented as a function of covariate and time. Such relationship can be depicted using 3-dimensional partial predicted plot [24]. This feature can be handled by the time-varying coefficient model. The time-varying coefficient model described earlier includes a non-linear term for time, which is modeled non-parametrically, and a linear term for each covariate. Our contributions with respect to this feature are: (1) we extended the time-var-

ying coefficient model by modeling multiple responses jointly, and (2) we extend the time-varying coefficient model by introducing a data-adaptive function for the covariate as well. Such an extension is useful in situation when, in the true model, covariate enters into the model non-linearly, and the aim of the analysis is to improve the model's prediction performance.

3. Our approach can handle high dimensionalities of covariates and responses when some of the covariates and responses are pure noise, and having them in the model do not unduly influence results for other decisive covariates and responses. In literature, issue of high dimensionality of covariates is often arises for cross-sectional data. There is a limited literature that talks about the high dimensionality of covariates in case of longitudinal data, and certainly no literature that we come across that talks about high dimensionality of the responses, and therefore our approach is important in providing a framework for handling high dimensionalities for both covariates and responses in longitudinal data.
4. Our approach performs response selection separately for each time interval. In literature, covariate selection is often implemented for longitudinal data. For example, see article by Wang and group that talks about covariate selection in case of time-varying coefficient model [25]. The novelty of our approach is that, in addition to covariate selection, we also perform response selection and we do that for different time intervals. This helps to identify and order the responses based on their importance for a given time interval. This approach can guide an investigator in allocating resources while collecting relevant information.
5. In our approach, the parameters in the model are estimated using gradient boosting, a generic form of boosting. Using boosting for parameter estimation is very crucial for our approach because it allows to estimate parameters accurately even in high dimension without breaking or substantially compromising the performance.

Combining above features, we believe, make our approach powerful in addressing various aspects of longitudinal data analysis with multiple responses. Use of boosting for parameter estimation adds robustness to our approach. Boosting is a powerful machine learning approach introduced in the statistical community by Friedman [26] for handling classification problems. Following this work, Friedman [24] provided a generic gradient boosting algorithm. Initially, most of the applications of boosting were focused on classification and regression problems. However, as boosting started to gain popularity, applications of boosting to new problems, including longitudinal data, have increased [27–30]. Model based boosting, implemented using the R package `mboost`, provides an application of boosting to a general data problem including the longitudinal data problem [31]. However, the package does not implement joint modeling of multivariate longitudinal responses. Applications of boosting for joint modeling of multiple responses are still rare; one such example is the work by Lutz and Buhlmann [32] which is focused on cross-sectional and time series data. Therefore, we believe that our work provides an important gateway for utilizing boosting for joint modeling of multivariate longitudinal responses. Due to the use of boosting in our nonparametric approach for modeling multivariate longitudinal responses, we refer to our approach as BoostMLR.

BoostMLR has some similarities with the component-wise ℓ_2 boosting [33], which is a special case of gradient boosting. Component-wise ℓ_2 boosting has been used in the literature for modeling linear models with high dimensional covariates [34]. Section S1 of the supplementary material provides brief overview of component-wise ℓ_2 boosting. To handle high dimensionality of data that includes redundant covariate-response pairs, we use an ℓ_2 loss function with an ℓ_1 penalization. Penalization helps to shrink the effect of redundant covariate-response pairs to zero. This has multiple advantages:

1. It provides a parsimonious model.
2. It helps to improve the prediction performance.
3. It allows for early termination of boosting procedure, thereby prevents it from overfitting, and also reduces significant amount of computation.

Our article is arranged as follows. In "Model", we describe our model. In "Boosting Procedure", we describe the estimation procedure for our model. As a part of estimation, we estimate flexible non-linear data-adaptive functions to model covariates and their interactions with time. In "Identifying Important Variables", we describe our approach for selection of important variables, which include selection of covariates and responses. Covariate selection is performed using variable importance (VIMP) approach. Estimates from VIMP can separate the effect of a covariate into covariate main effect and covariate-time interaction effect. The covariate-time interaction VIMP effect is further separated into various time intervals to identity covariates that are associated with the response for a specific time interval. This answers whether the covariate has any effect on the response, and if so, whether this effect is constant or varying over time, and identify the time interval where the effect is maximum. Response selection is performed using a new metric that we derived, referred to as likelihood of response selection. This metric answers how likely a particular response is selected among the multiple competing responses for a given time interval. This is used to identify importance of one response over others for a given time. For example, using the laboratory data, we showed that, as a metric for identifying advanced stage of heart failure, bilirubin is more important than creatinine before the transplant, whereas, creatinine is more important than bilirubin after the transplant. In "Simulation", we compare the performance of BoostMLR with other comparative approaches. In "Clinical Laboratory Data Analysis", we provide an application of BoostMLR to the laboratory data. "Conclusion" provides conclusion.

Model

Let $\mathbf{Y}_i^{(l)} \in \mathbb{R}^{n_i}$ represents the lth response vector of n_i dimension, where $l = 1, 2, \ldots, L$, and $\mathbf{x}_i^{(k)}$ represents an n_i dimensional observations of kth covariate where $k = 1, 2, \ldots, K$, measured at observed time $\mathbf{t}_i$ for the ith subject, where $i = 1, 2, \ldots, n$. It is possible that some covariates are time-varying and others are measured only at baseline (in such case, baseline values are replicated to maintain their dimension consistent with time-varying covariates and responses). We assume that covariates and responses are standardized such that for $l = 1, 2, \ldots, L$ and for $k = 1, 2, \ldots, K$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} Y_{ij}^{(l)} = 0,$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} Y_{ij}^{(l)^2} = 1,$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} x_{ij}^{(k)} = 0,$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n_i} x_{ij}^{(k)^2} = 1.$$

Remark 1 Note that if some of the covariates are nominal with two or more non-numeric levels, those needs to be converted into the numerical binary covariates so they can be standardized. This is a standard procedure which is followed to create design matrix before model fitting.

Consider the following model

$$\mathbf{Y}^{(l)} = \boldsymbol{\mu}^{(l)} + \epsilon^{(l)}, l = 1, 2, \ldots L, \tag{1}$$

where $\mathbf{Y}^{(l)} = \left(\mathbf{Y}_1^{(l)}, \mathbf{Y}_2^{(l)}, \ldots, \mathbf{Y}_n^{(l)}\right)^T$ represents a vector of lth response of dimension $N = \sum_{i=1}^{n} n_i$ and $\boldsymbol{\mu}^{(l)}$ represents an expectation of $\mathbf{Y}^{(l)}$, and let $\mathbf{t} = \left(\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_n\right)^T$ represents an N dimensional vector of time and the matrix $\mathbf{x} = \left[\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(K)}\right]$ represents an $N \times K$ dimensional matrix of covariates where the kth column of $\mathbf{x}$ is represented by $\mathbf{x}^{(k)} = \left(\mathbf{x}_1^{(k)}, \mathbf{x}_2^{(k)}, \ldots, \mathbf{x}_n^{(k)}\right)^T$. Term $\boldsymbol{\epsilon}^{(l)}$ represents the measurement error for response $\mathbf{Y}^{(l)}$. We assume that $\boldsymbol{\epsilon}^{(l)}$ has zero mean and a block diagonal variance matrix $\mathbf{V}^{(l)}$ of dimension $N \times N$ where the ith diagonal element of $\mathbf{V}^{(l)}$, given by $\mathbf{V}_i^{(l)}$, which represents the variance-covariance matrix of $\boldsymbol{\epsilon}_i^{(l)}$, where $i = 1, 2, \ldots, n$. We represent $\mathbf{V}_i^{(l)}$ by $\mathbf{V}_i^{(l)} = \phi^{(l)}\mathbf{R}_i(\rho^{(l)})$, where $\phi^{(l)}$ represents the dispersion parameter and $\mathbf{R}_i(\rho^{(l)})$ represents an exchangeable correlation matrix, parameterized by correlation parameter $\rho^{(l)}$. Henceforth, we represent the correlation matrix using $\mathbf{R}_i^{(l)}$. Additionally, we assume that $\text{Corr}\left(Y_{i,j}^{(l)}, Y_{i,j}^{(l')}\right) = 0$ where $l \neq l'$. This is a strong assumption, specifically in clinical studies. However, there are two main reasons to make such assumption:

1. The primary focus of our approach is to model the mean response $\boldsymbol{\mu}^{(l)}$ in the presence of other responses, and not so much on modeling correlations among multiple responses.
2. Modeling correlation among multiple responses can be computationally challenging, especially when L is large.

It is important to note that even with the above assumption, our approach can still be treated as an approach for multivariate responses. This is because, as you will see in "Boosting Procedure", we are modeling the responses simultaneously and only one response is updated at any boosting iteration, and hence all the responses are competing with each other to update their respective mean responses.

We consider the following form for $\boldsymbol{\mu}^{(l)}$

$$\boldsymbol{\mu}^{(l)} = \sum_{k=1}^{K} F^{(k,l)}\left(\mathbf{x}^{(k)}\right) G^{(k,l)}(\mathbf{t}), \quad \text{for } l = 1, 2, \ldots, L, \tag{2}$$

where $F^{(k,l)}\left(\mathbf{x}^{(k)}\right)$ represents an $N \times N$ dimensional function of the kth covariate and lth response and $G^{(k,l)}(\mathbf{t})$ represents an N dimensional function of time $\mathbf{t}$ corresponding to the kth covariate and lth response. Both $F(\cdot)$ and $G(\cdot)$ are unspecified functions (unspecified by the user) that we estimate from the data. We represent function $F^{(k,l)}\left(\mathbf{x}^{(k)}\right)$ as

$$F^{(k,l)}\left(\mathbf{x}^{(k)}\right) = \sum_{d=1}^{D_k} \mathbf{B}_{\mathbf{X}_d}^{(k)} \alpha_d^{(k,l)}, \tag{3}$$

where $\alpha_d^{(k,l)}$ represents an unknown scalar parameter corresponding to the kth covariate and lth response and $\mathbf{B}_{\mathbf{X}_d}^{(k)}$ represents a known diagonal matrix of $N \times N$ dimension corresponding to the kth covariate. This matrix is obtained as follows. First, map $\mathbf{x}^{(k)}$ using B-spline [35]; this generates an $N \times D_k$ dimensional matrix, represented by $\mathbf{B}_{\mathbf{x}}^{(k)} = \left[\mathbf{b}_{\mathbf{x},1}^{(k)}, \mathbf{b}_{\mathbf{x},2}^{(k)}, \ldots, \mathbf{b}_{\mathbf{x},D_k}^{(k)}\right]$. Second, write $\mathbf{b}_{\mathbf{x},d}^{(k)}$, which is the dth column of $\mathbf{B}_{\mathbf{x}}^{(k)}$, as an $N \times N$ diagonal matrix, represented by $\mathbf{B}_{\mathbf{X}_d}^{(k)}$. The form of $\mathbf{B}_{\mathbf{X}_d}^{(k)}$ is given by

$$\mathbf{B}_{\mathbf{X}_d}^{(k)} = \begin{pmatrix} b_{\mathbf{x},d_1}^{(k)} & 0 & \dots & 0 \\ 0 & b_{\mathbf{x},d_2}^{(k)} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & b_{\mathbf{x},d_N}^{(k)} \end{pmatrix}. \tag{4}$$

We represent $G^{(k,l)}(\mathbf{t})$ as

$$G^{(k,l)}(\mathbf{t}) = \mathbf{B}_{\mathbf{T}} \boldsymbol{\gamma}^{(k,l)}, \tag{5}$$

where $\mathbf{B}_{\mathbf{T}}$ represents an $N \times H$ dimensional matrix obtained using B-spline of time $\mathbf{t}$, and $\boldsymbol{\gamma}^{(k,l)}$ represents an H dimensional unknown parameter. To understand the form of $\mathbf{B}_{\mathbf{T}}$, assume that $\mathbf{t} = \left(t_{(1)}, t_{(2)}, \dots, t_{(N)}\right)^T$ such that $t_{(1)} \leq t_{(2)} \leq \cdots \leq t_{(N)}$, then the form of $\mathbf{B}_{\mathbf{T}}$ is given by

$$\mathbf{B}_{\mathbf{T}} = \begin{pmatrix} b_{1,1} & 0 & \dots & 0 \\ . & . & \dots & . \\ . & . & \dots & . \\ b_{1,s_1-1} & 0 & \dots & . \\ b_{1,s_1} & b_{2,1} & \dots & 0 \\ 0 & . & \dots & . \\ . & . & \dots & . \\ . & b_{2,s_2-1} & \dots & . \\ . & b_{2,s_2} & \dots & 0 \\ 0 & 0 & \dots & b_{H,1} \\ . & . & \dots & . \\ . & . & \dots & . \\ . & . & \dots & b_{H,s_H-1} \\ 0 & 0 & \dots & b_{H,s_H} \end{pmatrix}, \tag{6}$$

where s_h, for $h = 1, 2, \dots, H$, represents an integer value that depends on the degree of local polynomial and the B-spline knots, whereas $b_{i',j'} > 0$ for $i' = 1, 2, \dots, H$ and $j' = 1, 2, \dots, s_h$. Rows of $\mathbf{B}_{\mathbf{T}}$ correspond to $\mathbf{t} = \left(t_{(1)}, t_{(2)}, \dots, t_{(N)}\right)^T$ and columns of $\mathbf{B}_{\mathbf{T}}$ represent H time intervals generated using the B-spline knots. Notice that the initial values of $\mathbf{t}$ correspond to non-zero values for the initial columns of $\mathbf{B}_{\mathbf{T}}$ whereas later values of $\mathbf{t}$ correspond to non-zero values for the later columns of $\mathbf{B}_{\mathbf{T}}$. Note that, for illustrating the form of $\mathbf{B}_{\mathbf{T}}$, we use $\mathbf{t} = \left(t_{(1)}, t_{(2)}, \dots, t_{(N)}\right)^T$, i.e., time is in increasing order. However, $\mathbf{B}_{\mathbf{T}}$ can be obtained even when time is not ordered, i.e., using $\mathbf{t} = \left(t_1, t_2, \dots, t_N\right)^T$. In that case, the rows of $\mathbf{B}_{\mathbf{T}}$ are in the order of $\mathbf{t} = \left(t_1, t_2, \dots, t_N\right)^T$. In "Identifying Important Variables", we use the columns of $\mathbf{B}_{\mathbf{T}}$ to find an association of response with covariate for various time intervals. Equation (5) can be rewritten as

$$G^{(k,l)}(\mathbf{t}) = \sum_{h=1}^{H} \mathbf{B}_{\mathbf{T}h} \gamma_h^{(k,l)}, \tag{7}$$

where $\mathbf{B}_{\mathbf{T}h}$ represents the hth column of $\mathbf{B}_{\mathbf{T}}$ and $\gamma_h^{(k,l)}$ represents an unknown parameter corresponding to kth covariate and lth response. Using the form of $F^{(k,l)}\left(\mathbf{x}^{(k)}\right)$ and $G^{(k,l)}(\mathbf{t})$, (2) can be rewritten as

$$\begin{aligned} \boldsymbol{\mu}^{(l)} &= \sum_{k=1}^{K} \sum_{d=1}^{D_k} \mathbf{B}_{\mathbf{X}_d}^{(k)} \alpha_d^{(k,l)} \sum_{h=1}^{H} \mathbf{B}_{\mathbf{T}h} \gamma_h^{(k,l)} \\ &= \sum_{k=1}^{K} \sum_{d=1}^{D_k} \sum_{h=1}^{H} \mathbf{B}_{d,h}^{(k)} \beta_{d,h}^{(k,l)}, \end{aligned} \tag{8}$$

where $\mathbf{B}_{d,h}^{(k)} = \mathbf{B}_{\mathbf{X}_d}^{(k)} \mathbf{B}_{\mathbf{T}h}$ and $\beta_{d,h}^{(k,l)} = \alpha_d^{(k,l)} \gamma_h^{(k,l)}$. Equation (8) is cluttered with subscripts and superscripts. To reduce some of the cluttering, we combine subscripts d and h, and superscript k using κ. Thus, we replace $\mathbf{B}_{d,h}^{(k)}$ and $\beta_{d,h}^{(k,l)}$, respectively, with $\mathbf{B}_\kappa$ and $\beta_\kappa^{(l)}$. Additionally, triple summations are replaced by the single summation. Using the new notations (8) can be written as

$$\boldsymbol{\mu}^{(l)} = \sum_{\kappa} \mathbf{B}_\kappa \beta_\kappa^{(l)} \tag{9}$$

Remark 2 Note that, although we use B-spline for mapping covariates and time, other local polynomial approaches can be used as long as the form of $\mathbf{B}_{\mathbf{T}}$ is similar to the one describe in (6).

Remark 3 Note that in typical non-parametric model using B-spline (not in the context of boosting), number and positioning of knots can have high impact as they control model fitting. However, in our approach, B-spline is used within the boosting framework, and in the terminology of boosting, it is represented as a part of the base learner (described in "Boosting Procedure"). In boosting, the base learners are often simple functions which may have weak performance when fitted individually, however, when use within the boosting framework, they demonstrate strong performance. Therefore, in our boosting approach, we use relatively small number of knots which are distributed uniformly. We then use tuning parameters such as M and ν which are built within boosting to control the model fitting.

Boosting Procedure

As a part of the boosting procedure, we define our base learner as follows

$$\mathbf{h}(\mathbf{x}, \mathbf{t}, \mathbf{a}^{(l)}) = \sum_{\kappa} \mathbf{B}_\kappa \beta_\kappa^{(l)} 1_{(k=k_h, l=l_h)}, \quad \text{for } l = 1, 2, \dots, L,$$

where $\mathbf{a}^{(l)} = \{\beta_\kappa^{(l)}, k_h, l_h, \forall \kappa\}$ represents a set of parameters for the base learner and $1_{(\cdot)}$ represents an indicator function that takes value 1 if the condition in the parenthesis is satisfied, else 0. Parameter $\beta_\kappa^{(l)}$ is already introduced in (9). Parameters k_h and l_h represent functions of time and suggest the most informative covariate-response pair for the hth time interval. Thus, rather than fitting each response separately, all the responses are fitted simultaneously so they could compete with one another and our method chooses the most informative response for a given time interval. This is a crucial step of our approach where multiple responses are analysed together. To estimate $\mathbf{a}^{(l)}$, we use procedure which is similar to component-wise ℓ_2 boosting. In literature, ℓ_2 boosting is used primarily for estimating parameters from a linear model where each covariate has a unique scalar parameter, and for a given boosting iteration, one of the coordinates, for example the coordinate corresponding to the kth covariate, is updated. In our case, we follow a similar procedure except we update the estimate for the function corresponding to kth covariate and lth response.

The loss function we consider is given by

$$\begin{aligned}\mathscr{L}^{(l)} &= \mathscr{L}\big(\mathbf{Y}^{(l)}, \boldsymbol{\mu}^{(l)}\big)\\ &= \frac{1}{2}\big(\mathbf{Y}^{(l)} - \boldsymbol{\mu}^{(l)}\big)^T \mathbf{V}^{(l)^{-1}}\big(\mathbf{Y}^{(l)} - \boldsymbol{\mu}^{(l)}\big).\end{aligned} \tag{10}$$

Once we define the loss function, we estimate $\mathbf{a}^{(l)}$ by finding the base learner closest to the negative gradient. The negative gradient for the mth boosting iteration is given by

$$\begin{aligned}\mathbf{g}_{m,\mathbf{V}}^{(l)} &= -\frac{\partial \mathscr{L}\big(\mathbf{Y}^{(l)}, \boldsymbol{\mu}^{(l)}\big)}{\partial \boldsymbol{\mu}^{(l)}}\Bigg|_{\boldsymbol{\mu}^{(l)}=\boldsymbol{\mu}_{m-1}^{(l)}}\\ &= \mathbf{V}^{(l)^{-1}}\big(\mathbf{Y}^{(l)} - \boldsymbol{\mu}_{m-1}^{(l)}\big),\end{aligned}$$

where $\boldsymbol{\mu}_{m-1}^{(l)}$ represents an estimate of $\boldsymbol{\mu}^{(l)}$ from the $(m-1)$th boosting iteration. We assume that $\boldsymbol{\mu}_0^{(l)} = \mathbf{0}$ for $l = 1, 2, \dots, L$. We studied a similar loss function in our earlier work and observed that the performance of gradient boosting improves when we replace variance matrix by an identify matrix in the expression of negative gradient such that it is represented by residual (see Chapter 5 from [36]). The performance gain is specially notable for high dimensional situation. Thus, rather than using the form describe above, we use the following form for the negative gradient

$$\mathbf{g}_m^{(l)} = \mathbf{Y}^{(l)} - \boldsymbol{\mu}_{m-1}^{(l)}.$$

As a validation, we compare the performance of two types of negative gradients $\big(\mathbf{g}_{m,\mathbf{V}}^{(l)}$ vs $\mathbf{g}_m^{(l)}\big)$ using simulation approach. The simulation approach is described in "Simulation" and the results are provided in the supplementary material. Using this new form of negative gradient has two advantages. First, the negative gradient is the same as residual, described in the component-wise ℓ_2 boosting, and thus easy to interpret; second, it allows us to get rid of the line search optimization step. Once we define the negative gradient, the estimate of $\mathbf{a}^{(l)}$ can be obtained by solving the following loss function

$$\begin{aligned}\mathscr{L}^{(l)} &= \mathscr{L}\big(\mathbf{g}_m^{(l)}, \mathbf{h}(\mathbf{x}, \mathbf{t}, \mathbf{a}^{(l)})\big)\\ &= \mathscr{L}\left(\mathbf{g}_m^{(l)}, \sum_{\kappa} \mathbf{B}_\kappa \beta_\kappa^{(l)} 1_{(k=k_h, l=l_h)}\right),\end{aligned} \tag{11}$$

where the form of $\mathscr{L}(\cdot,\cdot)$ is given in (10). Estimation of $\mathbf{a}^{(l)}$ is performed by estimating each component separately (a feature of component-wise ℓ_2 boosting). In addition to estimating $\mathbf{a}^{(l)}$, our aim is also to address the high dimensionality of responses and covariates. To do that we modify the above loss function by adding an ℓ_1 penalization. Details involving parameter estimation are described in Section S2 of the supplementary material.

Following the procedure from Section S2 of the supplementary material, we obtain the estimate of $\beta_\kappa^{(l)}$, denoted by $\hat{\beta}_\kappa^{(l)}$. Once we estimate $\hat{\beta}_\kappa^{(l)}$, we find $\{k_{m,h}, l_{m,h}\}$, for $h = 1, 2, \dots, H$, meaning, we find the covariate-response pair that satisfies the following condition,

$$\{k_{m,h}, l_{m,h}\} = \operatorname*{argmax}_{1\le k\le K, 1\le l\le L} \sum_{d=1}^{D_k} \big(\hat{\beta}_\kappa^{(l)}\big)^2, \tag{12}$$

where $\{k_{m,h}, l_{m,h}\}$ represents an estimate of $\{k_h, l_h\}$ for the mth boosting iteration. Using $\{k_{m,h}, l_{m,h}\}$, we update $\boldsymbol{\mu}^{(l)}$ for the mth boosting iteration using

$$\boldsymbol{\mu}_m^{(l)} = \boldsymbol{\mu}_{m-1}^{(l)} + \nu \sum_{\kappa} \mathbf{B}_\kappa \hat{\beta}_\kappa^{(l)} 1_{(k=k_{m,h}, l=l_{m,h})}, \tag{13}$$

where ν represents the learning rate (see Algorithm provided in Section S1 of the supplementary material).

Remark 4 In (13) we update $\boldsymbol{\mu}^{(l)}$ using $\hat{\beta}_\kappa^{(l)}$. In "Identifying Important Variables", we observe that it is convenient if we first update $\beta_\kappa^{(l)}$ for the mth boosting iteration and then update $\boldsymbol{\mu}^{(l)}$. We update $\beta_\kappa^{(l)}$ for the mth boosting iteration using

$$\beta_{\kappa,m}^{(l)} = \beta_{\kappa,m-1}^{(l)} + \nu \hat{\beta}_\kappa^{(l)} 1_{(k=k_{m,h}, l=l_{m,h})}. \tag{14}$$

Update $\boldsymbol{\mu}^{(l)}$ for $l = 1, 2, \dots, L$ corresponding to the mth boosting iteration using

$$\boldsymbol{\mu}_m^{(l)} = \sum_{\kappa} \mathbf{B}_\kappa \beta_{\kappa,m}^{(l)}. \tag{15}$$

Remark 5 One advantage of an estimate $\beta_{\kappa,m}^{(l)}$ is that, for the mth boosting iteration, it can be estimated independently for different combinations of k and l. This way if there are missing values between kth covariate and lth response, they do not affect estimation for k'th covariate and l'th response where $k \neq k'$ and $l \neq l'$.

Remark 6 Estimation of $\beta_\kappa^{(l)}$ requires estimate of $\mathbf{V}^{(l)}$. Details about the estimation are provided in Section S3 of the supplementary material.

Following algorithm provides key steps of implementation of our BoostMLR approach. This is a concise version of the detailed version provided in Section S4 of the supplementary material. We use the algorithm from the supplementary material for modeling of simulated and real data.

Algorithm *BoostMLR algorithm for modeling multivariate longitudinal responses*

```
Initialize $\boldsymbol{\mu}_0^{(l)} = \mathbf{0}$ and $\mathbf{V}_0^{(l)} = \mathbf{I}$ for $l = 1, 2, \ldots, \mathrm{L}$, where $\mathbf{I}$ represents the identity matrix.
for $m = 1, \ldots, M$ do
    Find the negative gradient $\mathbf{g}_m^{(l)}$ for $l = 1, 2, \ldots, \mathrm{L}$
        $\mathbf{g}_m^{(l)} = \mathbf{Y}^{(l)} - \boldsymbol{\mu}_{m-1}^{(l)},$
    where $\boldsymbol{\mu}_{m-1}^{(l)}$ represents an estimates of $\boldsymbol{\mu}^{(l)}$ from the $(m-1)$th boosting iteration.
    Solve the following loss function to estimate parameters from the base learner
        $\mathscr{L}^{(l)} = \mathscr{L}\left(\mathbf{g}_m^{(l)}, \sum_{\kappa} \mathbf{B}_{\kappa} \beta_{\kappa}^{(l)} 1_{(\mathrm{k}=\mathrm{k}_{\mathrm{h}}, l=l_{\mathrm{h}})}\right).$
    Use the estimate of $\beta_{\kappa}^{(l)}$ and $\{\mathrm{k}_{\mathrm{h}}, l_{\mathrm{h}}\}$ to update $\boldsymbol{\mu}^{(l)}$ for the $m$th boosting iteration using
        $\boldsymbol{\mu}_m^{(l)} = \boldsymbol{\mu}_{m-1}^{(l)} + \nu \sum_{\kappa} \mathbf{B}_{\kappa} \hat{\beta}_{\kappa}^{(l)} 1_{(\mathrm{k}=\mathrm{k}_{m,\mathrm{h}}, l=l_{m,\mathrm{h}})}.$
    Estimate $\mathbf{V}^{(l)}$ for the $m$th boosting iteration using the procedure described in Section S3 of the supplementary material.
end for
Return $\left(\boldsymbol{\mu}_M^{(l)}\right)_{l=1,\ldots,\mathrm{L}}$.
```

Identifying Important Variables

In our method, we identify important covariates and responses, and to do that, we use different approaches. In this section, we discuss approaches for identifying covariates and responses separately.

Identifying Important Covariates

To identify covariates that influence the response, we use standardized variable importance (VIMP) approach [24]. VIMP measures the importance of a covariate using its effect on model's prediction performance. Our VIMP approach can separate the effect of covariate into covariate main effect and covariate-time interaction effect. Covariate main effect represents an effect of a covariate on the response without involving the time component, whereas covariate-time interaction effect represents an effect of covariate and time on the response. We further separate covariate-time interaction effect into various time intervals. This allows us to evaluate how the effects are varying across time. The standardized VIMP for covariate main effect and covariate-time interaction effect are denoted by $\mathrm{sVIMP}_{\mathrm{main}}$ and $\mathrm{sVIMP}_{\mathrm{int}}$ respectively.

In this section, we differentiate two sources of data. Data that we described in "Model" is referred here as training data and the other data is referred as test data. We build the model using training data, using the procedure described earlier, and use test data to calculate VIMP. Let $\tilde{\mathbf{X}} = \left[\tilde{\mathbf{x}}^{(1)}, \tilde{\mathbf{x}}^{(2)}, \ldots, \tilde{\mathbf{x}}^{(K)}\right]$, $\tilde{\mathbf{t}}$ and $\{\tilde{\mathbf{Y}}^{(l)}\}_{1 \le l \le L}$ represent test data for K covariates, time, and L responses respectively. We assume that both training and test data are generated from the same distribution. To find $\mathrm{sVIMP}_{\mathrm{main}}$ and $\mathrm{sVIMP}_{\mathrm{int}}$, we need to find the predicted response corresponding to $\{\tilde{\mathbf{Y}}^{(l)}\}_{1 \le l \le L}$ as follows. Let $\tilde{x}_{i,j}^{(k)}$ represents an observation for the (i, j)th component of covariate $\tilde{\mathbf{x}}^{(k)}$. Find a value for the kth covariate from the training data that is closest to $\tilde{x}_{i,j}^{(k)}$, denoted by $x_{i',j'}^{(k)}$. For a given d, where $d = 1, 2, \ldots, D_k$, we use $x_{i',j'}^{(k)}$ to extract a corresponding value from the set $\left\{b_{\mathbf{x},d_1}^{(k)}, b_{\mathbf{x},d_2}^{(k)}, \ldots, b_{\mathbf{x},d_N}^{(k)}\right\}$. (Elements from the set are the diagonal elements from $\mathbf{B}_{\mathbf{X}_d}^{(k)}$, described in (4).) This procedure is repeated for all possible (i, j)th components of $\tilde{\mathbf{x}}^{(k)}$. The extracted values from the set are used to generate a new diagonal matrix $\tilde{\mathbf{B}}_{\mathbf{X}_d}^{(k)}$ for the test data where $d = 1, 2, \ldots, D_k$. Repeat this procedure for $k = 1, 2, \ldots, K$. Similar approach is used to generate $\tilde{\mathbf{B}}_{\mathbf{T}_h}$, where $h = 1, 2, \ldots, H$, for the test data $\tilde{\mathbf{t}}$ as follows. Let $\tilde{t}_{i,j}$ represents an observation for the (i, j)th component of $\tilde{\mathbf{t}}$. Find a value from $\mathbf{t}$ (where $\mathbf{t}$ represents time from the training data) that is closest to $\tilde{t}_{i,j}$ and use this value to extract a corresponding value from the hth column of $\mathbf{B}_{\mathbf{T}}$, described in (6). This procedure is repeated for all possible (i,j)th components of $\tilde{\mathbf{t}}$. The extracted values are used to generate a new column vector $\tilde{\mathbf{B}}_{\mathbf{T}h}$ for $h = 1, 2, \dots, H$. Calculate $\tilde{\mathbf{B}}^{(k)}_{d,h}$ using $\tilde{\mathbf{B}}^{(k)}_{d,h} = \tilde{\mathbf{B}}^{(k)}_{\mathbf{X}_d}\tilde{\mathbf{B}}_{\mathbf{T}h}$ for $k = 1, 2, \dots, K$, $d = 1, 2, \dots, D_k$ and $h = 1, 2, \dots, H$. The estimate of the predicted response for $\tilde{\mathbf{Y}}^{(l)}$ is given by

$$\tilde{\boldsymbol{\mu}}^{(l)} = \sum_{k=1}^{K}\sum_{d=1}^{D_k}\sum_{h=1}^{H}\tilde{\mathbf{B}}^{(k)}_{d,h}\beta^{(k,l)}_{d,h,M} \quad \text{for } l = 1, 2, \dots, L,$$

where $\beta^{(k,l)}_{d,h,M}$ is obtained from (14), corresponding to the Mth boosting iteration. Note that in our new notations, described in (9) and (14), the above expression can be written as

$$\tilde{\boldsymbol{\mu}}^{(l)} = \sum_{\kappa}\tilde{\mathbf{B}}_{\kappa}\beta^{(l)}_{\kappa,M} \quad \text{for } l = 1, 2, \dots, L.$$

The prediction error for the test data is calculated using standardized root mean square error (sRMSE)

$$\text{sRMSE}^{(l)} = \frac{\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left(\tilde{Y}^{(l)}_{i,j} - \tilde{\mu}^{(l)}_{i,j}\right)^2\right]^{1/2}}{\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{n_i}\sum_{j=1}^{n_i}\left(\tilde{Y}^{(l)}_{i,j}\right)^2\right]^{1/2}} \qquad (16)$$
$$\text{for } l = 1, 2, \dots, L,$$

where $\tilde{Y}^{(l)}_{i,j}$ and $\tilde{\mu}^{(l)}_{i,j}$ represent the (i,j)th components of $\tilde{\mathbf{Y}}^{(l)}$ and $\tilde{\boldsymbol{\mu}}^{(l)}$ respectively. In “Results”, we use sRMSE to compare the performance of our approach with other comparative methods.

To calculate $\text{sVIMP}_{\text{main}}$ for kth covariate, we proceed as follows. As mentioned earlier, we represented $\tilde{\mathbf{x}}^{(k)}$ as the kth covariate from the test data $\tilde{\mathbf{X}}$. We use $\tilde{\mathbf{x}}^{(k)}$ to generate a noised-up covariate, denoted by $\tilde{\mathbf{x}}^{(k)}_{\text{Noise}}$. The noised-up $\tilde{\mathbf{x}}^{(k)}_{\text{Noise}}$ is obtained by randomly permuting the coordinates of $\tilde{\mathbf{x}}^{(k)}$. We perform the same procedure that we performed earlier, except instead of using $\tilde{\mathbf{x}}^{(k)}$, we use $\tilde{\mathbf{x}}^{(k)}_{\text{Noise}}$ to generate a diagonal matrix, denoted here by $\tilde{\mathbf{B}}^{(k)}_{\mathbf{X}_{d,\text{Noise}}}$, for $d = 1, 2, \dots, D_k$. Calculate $\tilde{\mathbf{B}}^{(k)}_{d,\text{Noise},h} = \tilde{\mathbf{B}}^{(k)}_{\mathbf{X}_{d,\text{Noise}}}\tilde{\mathbf{B}}_{\mathbf{T}h}$. Using $\tilde{\mathbf{B}}^{(k)}_{d,\text{Noise},h}$, we obtain an estimate of predicted response, denoted by $\tilde{\boldsymbol{\mu}}^{(l)}_{k}$ using

$$\tilde{\boldsymbol{\mu}}^{(l)}_{k} = \sum_{k'=1,k'\neq k}^{K}\sum_{d=1}^{D_{k'}}\sum_{h=1}^{H}\tilde{\mathbf{B}}^{(k')}_{d,h}\beta^{(k',l)}_{d,h,M}$$
$$+ \sum_{d=1}^{D_k}\sum_{h=1}^{H}\tilde{\mathbf{B}}^{(k)}_{d,\text{Noise},h}\beta^{(k,l)}_{d,h,M} \quad \text{for } l = 1, 2, \dots, L.$$

Note that to calculate $\tilde{\boldsymbol{\mu}}^{(l)}_{k}$ we restricted the noising specific to kth covariate; everything else remains same as described earlier. Once $\tilde{\boldsymbol{\mu}}^{(l)}_{k}$, for $l = 1, 2, \dots, L$, are estimated, we calculate $\text{sRMSE}^{(l)}_{\text{main},k}$ by replacing $\tilde{\mu}^{(l)}_{i,j}$ by $\tilde{\mu}^{(l)}_{i,j,k}$ in (16), where $\tilde{\mu}^{(l)}_{i,j,k}$ represents the (i,j)th component of $\tilde{\boldsymbol{\mu}}^{(l)}_{k}$. The $\text{sVIMP}_{\text{main}}$ for the kth covariate and lth response is calculated using

$$\text{sVIMP}^{(l)}_{\text{main},k} = \frac{\text{sRMSE}^{(l)}_{\text{main},k} - \text{sRMSE}^{(l)}}{\text{sRMSE}^{(l)}} \times 100 \quad \text{for } l = 1, 2, \dots, L.$$

This procedure is repeated for $k = 1, 2, \dots, K$ to find $\{\text{sVIMP}^{(l)}_{\text{main},k}\}_{1\leq k\leq K}$.

To calculate $\text{sVIMP}_{\text{int}}$ for kth covariate and lth response, corresponding to hth time interval, we use $\tilde{\mathbf{B}}_{\mathbf{T}h}$, a column vector that we described earlier in this section. Let $\tilde{\mathbf{B}}_{\mathbf{T}h,\text{Noise}}$ represents a new column vector obtain by permuting the coordinates of $\tilde{\mathbf{B}}_{\mathbf{T}h}$. Using $\tilde{\mathbf{B}}^{(k)}_{\mathbf{X}_d}$, calculate $\tilde{\mathbf{B}}^{(k)}_{d,h,\text{Noise}} = \tilde{\mathbf{B}}^{(k)}_{\mathbf{X}_d}\tilde{\mathbf{B}}_{\mathbf{T}h,\text{Noise}}$. Using $\tilde{\mathbf{B}}^{(k)}_{d,h,\text{Noise}}$, we obtain an estimate of predicted response, denoted by $\tilde{\boldsymbol{\mu}}^{(l)}_{k,h}$ using

$$\tilde{\boldsymbol{\mu}}^{(l)}_{k,h} = \sum_{k'=1,k'\neq k}^{K}\sum_{d=1}^{D_{k'}}\sum_{h'=1}^{H}\tilde{\mathbf{B}}^{(k')}_{d,h'}\beta^{(k',l)}_{d,h',M}$$
$$+ \sum_{d=1}^{D_k}\left[\sum_{h'=1,h'\neq h}^{H}\tilde{\mathbf{B}}^{(k)}_{d,h'}\beta^{(k,l)}_{d,h',M} + \tilde{\mathbf{B}}^{(k)}_{d,h,\text{Noise}}\beta^{(k,l)}_{d,h,M}\right].$$

Note that to calculate $\tilde{\boldsymbol{\mu}}^{(l)}_{k,h}$ we restricted noising specific to kth covariate and hth time interval; everything else remains same. Once $\tilde{\boldsymbol{\mu}}^{(l)}_{k,h}$, for $l = 1, 2, \dots, L$, are estimated, we calculate $\text{sRMSE}^{(l)}_{\text{int},k,h}$ by replacing $\tilde{\mu}^{(l)}_{i,j}$ by $\tilde{\mu}^{(l)}_{i,j,k,h}$ in (16), where $\tilde{\mu}^{(l)}_{i,j,k,h}$ represents the (i,j)th component of $\tilde{\boldsymbol{\mu}}^{(l)}_{k,h}$. The $\text{sVIMP}_{\text{int}}$ for the kth covariate and the lth response corresponding to the hth time interval is calculated using

$$\text{sVIMP}^{(l)}_{\text{int},k,h} = \frac{\text{sRMSE}^{(l)}_{\text{int},k,h} - \text{sRMSE}^{(l)}}{\text{sRMSE}^{(l)}}$$
$$\times 100 \quad \text{for } l = 1, 2, \dots, L.$$

This procedure is repeated for $k = 1, 2, \dots, K$ and $h = 1, 2, \dots, H$ to find $\{\text{sVIMP}^{(l)}_{\text{int},k,h}\}_{1\leq k\leq K,1\leq h\leq H}$. In “Results” and “Clinical Laboratory Data Analysis”, we provide results for $\text{sVIMP}_{\text{main}}$ and $\text{sVIMP}_{\text{int}}$ for simulated and real data, respectively.

Identifying Important Responses

There is no literature that we came across that talks about identifying important responses from the model that handles multivariate responses. However, identifying important responses is equally important especially when the number of responses is large. Obviously, the VIMP approach used for identifying important covariates will not work for identifying important responses. Here we use one of the results from our boosting method to identify important responses. Notice that after we estimate $\beta^{(l)}_{\kappa}$ in (12), we identify the covariate-response pair with the highest magnitude. The estimate of the covariate-response pair for the mth boosting iteration and for the hth time interval is denoted by

$\{k_{m,h}, l_{m,h}\}$. To find the important responses, we use the estimate $l_{m,h}$ as follows. Note that boosting is a sequential procedure where we use negative gradient as our response. We represented negative gradient using residual, i.e., $\mathbf{Y}^{(l)} - \boldsymbol{\mu}^{(l)}_{m-1}$. If we observe the above algorithm carefully, we find that, as the boosting increases, we move from modeling the original responses $\mathbf{Y}^{(l)}$ to modeling the residuals. This means that initial part of boosting is more important than the later part, such that after certain boosting iteration, the residual part is mostly noise. Therefore, one way to identify important responses is to give more weights to the initial part of the boosting results compared to the later part. We do this, to identify important responses, by generating new estimator, which we referred as likelihood of response selection, denoted by $\mathcal{L}(l,h)$. This estimator provides the chance that lth response is selected at the hth time interval; higher the chance, higher the importance of that response. We define $\mathcal{L}(l,h)$ by

$$\mathcal{L}(l,h) = \frac{1}{M}\sum_{m=1}^{M}\left(\frac{1}{m}\sum_{m'=1}^{m} 1_{(l_{m',h}=l)}\right),$$

where $1_{(l_{m',h}=l)}$ represents an indicator function that takes value 1 if the estimate $l_{m',h}$ is the lth response, else 0. As a toy example to understand the metric, assume that there are 3 responses, y_1, y_2 and y_3, and $h = 1$, and we fit our model with $M = 3$, and assume that the sequence in which these three responses are selected is $\{y_3, y_2, y_1\}$. Then using the index $l = 1, 2, 3$ for y_1, y_2 and y_3, the result we get is $\mathcal{L}(l = 1, h = 1) = 2/18$, $\mathcal{L}(l = 2, h = 1) = 5/18$ and $\mathcal{L}(l = 3, h = 1) = 11/18$. Thus, $\mathcal{L}(l,h)$ is highest for the y_3 because it was the first response which got selected in the boosting iteration, followed by y_2 and y_1. In "Results" and "Clinical Laboratory Data Analysis", we provide results of our response selection approach for simulated and real data, respectively.

Remark 7 Note that the same approach can be applicable for identifying important covariates as well. However, VIMP is a well-known approach in machine learning literature and thus we stick with VIMP approach for identifying important covariates.

Simulation

In this section, we compare our approach with other methods available in the literature. To do that, we use the following three experiments. In these experiments, our goal is to mimic the real world longitudinal data where some covariates are associated with the response for a portion of the study period (e.g., $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ in all three experiments), some are associated with the response for the entire study period (e.g., $\mathbf{x}^{(4)}$ in Experiment II), and others have no association with any of the responses.

Experiment Description

Experiment I

In this experiment, we generate $(4 + q_x)$ time-invariant covariates. Some combinations of covariates $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ are associated with responses $\mathbf{Y}^{(1)}$, $\mathbf{Y}^{(2)}$ and $\mathbf{Y}^{(3)}$. Covariate $\mathbf{x}^{(4)}$ and remaining q_x covariates are non-informative, meaning they do not relate with any responses. Additionally, we generate q_y non-informative responses. The model we consider is given by

$$\begin{aligned}
\mathbf{Y}^{(1)} &= \beta_0 + \beta_1\mathbf{x}^{(1)}1_{(\mathbf{t}<1.5)} + \beta_2\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \beta_3\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \gamma\mathbf{x}^{(4)} + \epsilon^{(1)}, \\
\mathbf{Y}^{(2)} &= \beta_0 + \beta_1\mathbf{x}^{(1)}1_{(\mathbf{t}<1.5)} + \gamma\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \beta_3\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \gamma\mathbf{x}^{(4)} + \epsilon^{(2)}, \\
\mathbf{Y}^{(3)} &= \beta_0 + \beta_1\mathbf{x}^{(1)}1_{(\mathbf{t}<1.5)} + \beta_2\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \gamma\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \gamma\mathbf{x}^{(4)} + \epsilon^{(3)},
\end{aligned}$$

where $\beta_0 = 1.5$, $\beta_1 = 1.5$, $\beta_2 = 1.2$, $\beta_3 = 1$ and $\gamma = 0$. The form of the model is such that, for a given time interval, say $1.5 \leq \mathbf{t} \leq 4.5$, and for a non-zero coefficient, the relationship between covariate and response is linear, and thus relatively simple. The time measurement $\mathbf{t}$ for a given subject is generated from uniform [0, 6] and arranged in an ascending order. All $(4 + q_x)$ covariates and q_y non-informative responses are generated from the standard normal distribution. The measurement error terms $\epsilon^{(l)}$ for $l = 1, 2, 3$ are generated from the normal distribution with mean zero and variance-covariance matrix $\mathbf{V}^{(l)} = \phi^{(l)}\mathbf{R}^{(l)}(\rho)$, where $\phi^{(l)} = 1$ and $\rho^{(l)} = 0.8$.

Experiment II

In this experiment, except for $\mathbf{x}^{(4)}$, all other covariates, time and q_y non-informative responses are generated in the same way as described in Experiment I. Covariate $\mathbf{x}^{(4)}$ represents a time-varying covariate and is generated using

$$\mathbf{x}^{(4)} = \exp(\mathbf{t} \times 0.2) + \epsilon_{\mathbf{x}},$$

where $\epsilon_{\mathbf{x}}$ is generated from the normal distribution with mean 0 and standard deviation 0.5. The model we consider is given by

$$\begin{aligned}
\mathbf{Y}^{(1)} &= \beta_0 + \beta_1\left(\mathbf{x}^{(1)}\right)^2 1_{(\mathbf{t}<1.5)} + \beta_2\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \beta_3\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \beta_4\mathbf{x}^{(4)} + \epsilon^{(1)}, \\
\mathbf{Y}^{(2)} &= \beta_0 + \beta_1\left(\mathbf{x}^{(1)}\right)^2 1_{(\mathbf{t}<1.5)} + \gamma\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \beta_3\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \beta_4\mathbf{x}^{(4)} + \epsilon^{(2)}, \\
\mathbf{Y}^{(3)} &= \beta_0 + \beta_1\left(\mathbf{x}^{(1)}\right)^2 1_{(\mathbf{t}<1.5)} + \beta_2\mathbf{x}^{(2)}1_{(1.5\leq\mathbf{t}\leq4.5)} \\
&\quad + \gamma\mathbf{x}^{(3)}1_{(\mathbf{t}>4.5)} + \beta_4\mathbf{x}^{(4)} + \epsilon^{(3)},
\end{aligned}$$

where $\beta_4 = 1$. Values for other coefficients are the same as described in Experiment I. In comparison to Experiment I, Experiment II has a quadratic term for $\mathbf{x}^{(1)}$ and a time-varying covariate $\mathbf{x}^{(4)}$ is associated with all 3 informative responses with non-zero coefficient. Thus, the model in this experiment is relatively more complex than Experiment I.

Experiment III

In this experiment, all covariates, time and q_y non-informative responses are generated in the same way as describe in Experiment I. The model we consider is given by

$$\begin{aligned}
\mathbf{Y}^{(1)} &= \beta_0 + \tilde{\beta}_1 \mathbf{x}^{(1)} 1_{(\mathbf{t}<1.5)} + \tilde{\beta}_2 \mathbf{x}^{(2)} 1_{(1.5\le\mathbf{t}\le4.5)} \\
&\quad + \tilde{\beta}_3 \mathbf{x}^{(3)} 1_{(\mathbf{t}>4.5)} + \gamma \mathbf{x}^{(4)} + \epsilon^{(1)}, \\
\mathbf{Y}^{(2)} &= \beta_0 + \tilde{\beta}_1 \mathbf{x}^{(1)} 1_{(\mathbf{t}<1.5)} \\
&\quad + \gamma \mathbf{x}^{(2)} 1_{(1.5\le\mathbf{t}\le4.5)} + \tilde{\beta}_3 \mathbf{x}^{(3)} 1_{(\mathbf{t}>4.5)} + \gamma \mathbf{x}^{(4)} + \epsilon^{(2)}, \\
\mathbf{Y}^{(3)} &= \beta_0 + \tilde{\beta}_1 \mathbf{x}^{(1)} 1_{(\mathbf{t}<1.5)} + \tilde{\beta}_2 \mathbf{x}^{(2)} 1_{(1.5\le\mathbf{t}\le4.5)} \\
&\quad + \gamma \mathbf{x}^{(3)} 1_{(\mathbf{t}>4.5)} + \gamma \mathbf{x}^{(4)} + \epsilon^{(3)},
\end{aligned}$$

where

$$\begin{aligned}
\tilde{\beta}_1 &= \exp\left(\log\left(\beta_1 + 1\right) \times \frac{\mathbf{t}}{1.5} \times 1_{(\mathbf{t}\le1.5)}\right) - 1 \\
\tilde{\beta}_2 &= \exp\left(\log\left(\beta_2 + 1\right) \times \left(\frac{\mathbf{t} - 1.5}{4.5 - 1.5}\right)^2 \times 1_{(1.5\le\mathbf{t}\le4.5)}\right) - 1 \\
\tilde{\beta}_3 &= \exp\left(\log\left(\beta_3 + 1\right) \times \left(\frac{\mathbf{t} - 4.5}{6 - 4.5}\right)^3 \times 1_{(\mathbf{t}\ge4.5)}\right) - 1
\end{aligned}$$

Values for coefficients β_0, β_1, β_2, β_3 and γ are the same as describe in Experiment I. In comparison to Experiment I, Experiment III has non-linear time-varying coefficients such that, for a given time interval, the value of coefficients vary between 0 to coefficient values from experiment I. For example, $\mathbf{x}^{(2)}$ is associated with $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(3)}$ with a non-zero coefficient between $1.5 \le \mathbf{t} \le 4.5$. In experiment I, the coefficient value is $\beta_2 = 1.2$ and is constant for $1.5 \le \mathbf{t} \le 4.5$; on the other hand, in Experiment III, the coefficient value increases non-linearly with time such that for $\mathbf{t} = 1.5$, $\tilde{\beta}_2 = 0$ and for $\mathbf{t} = 4.5$, $\tilde{\beta}_2 = \beta_2$ (see Fig. 1)

Experimental Settings

We use $n = 100$ subjects separately for training and test data. For each subject, the number of repeated observations is generated using a discrete uniform distribution from the interval [1, 20]. We consider low and high dimensional covariate settings with $q_x = 5$ and $q_x = 50$ respectively in all 3 experiments. While comparing the performance of our approach with other methods and for VIMP analysis, we use $q_y = 0$, however, in a separate simulation, we compare performance of our approach with increasing dimensionality of non-informative responses using $q_y = \{0, 5, 50\}$. These experimental settings are used to independently generate 100 datasets, and the results, average across 100 datasets, are recorded. All computations are performed on CentOS Linux server.

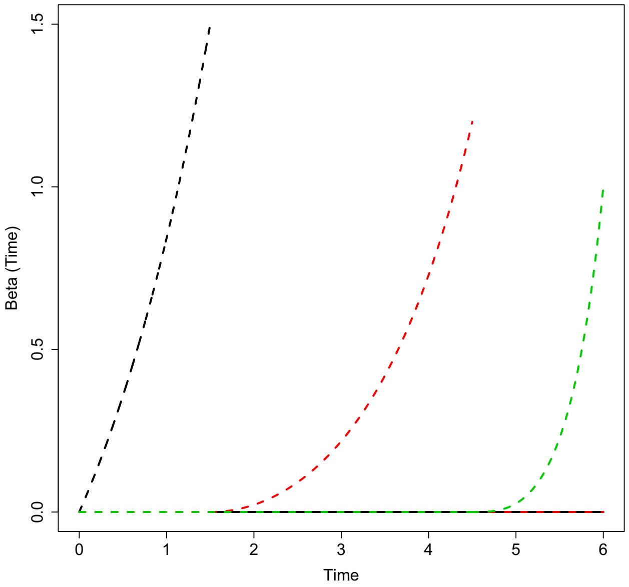

Fig. 1 Time varying coefficient $\tilde{\beta}_1$, $\tilde{\beta}_2$ and $\tilde{\beta}_3$ in black, red and green respectively plotted across time

Remark 8 We refer high dimensional setting for $q_x = 50$ and $q_y = 50$, however, this is highly subjective. In case of $q_x = 50$, one can view that $q_x = 50$ is not really a high dimensional covariate setting (even for the longitudinal data), and we agree with this viewpoint. The broader point we wanted to make through this simulation is that our method could be easily implemented even when $q_x = 500$ or $q_x = 5000$ without breaking, and we use $q_x = 50$ because of the limited computational resources. In case of $q_y = 50$, we strongly believe that $q_y = 50$ is a high dimensional response setting for longitudinal studies. We have not come across any literature that would treat modeling of 50 responses simultaneously as a common scenario.

Implementing Boosting for Multivariate Longitudinal Response

Our approach is implemented using the R package `Boost-MLR` [37], which implements boosting for multivariate longitudinal responses as described in the algorithm. We use cubic B-spline for mapping of covariates and time, respectively, using 5 and 25 equally spaced knots. The ℓ_1 penalization procedure is repeated 100 times to generate the distribution of noised-up estimate and 15th and 85th percentiles of that distribution are used to set limits for lower and upper bound respectively (see Section S2 of the supplementary material for details). We use the learning rate $\nu = 0.1$. We set the total number of boosting iterations $M = 500$, however M is estimated as a part of the boosting iteration (see Algorithm 2 in Section S4 of the supplementary material), which allows for early termination to avoid model overfitting.

Remark 9 Notice that in "Introduction", we described two forms of negative gradient, denoted by $\mathbf{g}^{(l)}_{m,\mathbf{V}}$ and $\mathbf{g}^{(l)}_{m}$. We mentioned that we prefer to use $\mathbf{g}^{(l)}_{m}$. In the supplementary section S5, we have provided a comparison of performance of two forms of negative gradient. From the result, it is clear that our approach has a superior performance when we use $\mathbf{g}^{(l)}_{m}$ as a negative gradient and thus use of this form is justified.

Comparative Methods

Multivariate Marginal and Mixed Effect Model

As a comparison to our BoostMLR approach, among other methods, we use multivariate marginal model (MMM) [8, 38] and multivariate mixed effect model (MMEM) [9]. MMM and MMEM can be implemented using the R packages `mmm` and `mixAK` respectively. The R package `mixAK` is primarily used for cluster analysis, which is based on multivariate longitudinal responses, however, in our case, we use the `GLMM_MCMC` function from this package, which provides an implementation of Bayesian estimation to multivariate linear mixed effect models. MMM includes, for each covariate, a linear term and an interaction term for covariate and time. For parameter estimation, we assume an exchangeable correlation structure. MMEM includes, for the random effect part, a random intercept, and for the fixed effect part, for each covariate, a linear term and an interaction term for covariate and time. For MMEM, we use Markov chain Monte Carlo approach for parameter estimation with a total of 6000 samples from which first 5000 are discarded as a burn-in sample, and remaining samples are used for parameter estimation. For other parameters, we use the default setting.

Multivariate Generalized Additive Mixed Model

Multivariate generalized additive mixed model (MGAMM) can be implemented using the R package `mgcv`. We fit the following model

$$\mathbf{Y}^{(l)} \leftarrow \alpha\mathbf{1} + \sum_{k=1}^{K} s(\mathbf{x}^{(k)}) + \sum_{k=1}^{K} s(\mathbf{x}^{(k)} \star \mathbf{t}), \quad \text{for } l = 1, 2, 3,$$

where the first term corresponds to the random intercept; second and third terms include, for each covariate, a term for covariate and for covariate-time interaction respectively. Function s represents an unknown function that needs to be estimated. We use B-spline to specify this function with 5 equally spaced knots. (We could not use higher number of knots because of the longer computational time.) Smoothing parameter for each function is estimated using restricted maximum likelihood. For other parameters, we use the default setting.

Model-Based Boosting

To provide comparison with other boosting procedures, we use model-based boosting from the R package `mboost`. We fit the following random intercept models

$$\mathbf{Y}^{(l)} \leftarrow \alpha\mathbf{1} + \sum_{k=1}^{K} \text{bbs}(\mathbf{x}^{(k)}) + \sum_{k=1}^{K} \text{bbs}(\mathbf{x}^{(k)} \star \mathbf{t}),$$
$$\mathbf{Y}^{(l)} \leftarrow \alpha\mathbf{1} + \text{btree}(\mathbf{x} : \mathbf{t}), \quad \text{for } l = 1, 2, 3,$$

where $\mathbf{x} : \mathbf{t}$ represents a dataset that consist of all K covariates as well as all covariate-time interactions. Unlike other comparative methods, `mboost` does not fit multivariate responses jointly. Therefore we fit each response separately and compare its prediction performance. The first term from each model represents a random intercept. Second and third terms from the first model represent a cubic B-spline base learner for the covariate and the covariate-time interaction respectively. Second term from the second model represents a regression tree base learner. First and second models are denoted by $\text{mboost}_{\text{BS}}$ and $\text{mboost}_{\text{Tree}}$ respectively. For $\text{mboost}_{\text{BS}}$, we use 5 and 25 equally spaced knots for second and third terms respectively, and for $\text{mboost}_{\text{Tree}}$, we grow a maximum of 5 terminal nodes regression tree. We use the learning rate $\nu = 0.1$. We set the total number of boosting iterations $M = 500$, however we estimate the value of M using 10 folds cross-validation to avoid model overfitting. For other parameters, we use the default setting.

Results

Prediction Performance

Methods described in "Comparative Methods" are compared for their prediction performance on the test data using sRMSE, described in (16). Table 1 provides sRMSE values averaged across 100 independently generated data.

In all three experiments, the overall performance of BoostMLR is considerably better than other procedures in both low and high dimensional covariate settings.

In Experiment I, for a given time interval, covariates are related with responses linearly. Linear models such as MMM and MMEM, each with linear term for covariates can better approximate the true model compared to

Table 1 Test set performance using simulations

	$q_x = 5$			$q_x = 50$		
	$\mathbf{Y}^{(1)}$	$\mathbf{Y}^{(2)}$	$\mathbf{Y}^{(3)}$	$\mathbf{Y}^{(1)}$	$\mathbf{Y}^{(2)}$	$\mathbf{Y}^{(3)}$
Experiment I						
MMM	0.611	0.582	0.613	0.828	0.816	0.831
MMEM	0.608	0.578	0.609	0.805	0.780	0.808
MGAMM	0.643	0.600	0.643	NA	NA	NA
$\text{mboost}_{\text{BS}}$	0.621	0.594	0.619	0.637	0.610	0.634
$\text{mboost}_{\text{Tree}}$	0.634	0.616	0.632	0.650	0.617	0.651
BoostMLR	**0.518**	**0.517**	**0.511**	**0.534**	**0.533**	**0.526**
Experiment II						
MMM	0.418	0.398	0.413	0.581	0.564	0.570
MMEM	0.417	0.397	0.412	0.579	0.557	0.565
MGAMM	0.365	0.336	0.360	NA	NA	NA
$\text{mboost}_{\text{BS}}$	**0.360**	0.333	0.352	**0.368**	0.340	0.358
$\text{mboost}_{\text{Tree}}$	0.365	0.348	0.364	0.399	0.385	0.400
BoostMLR	**0.331**	**0.288**	**0.314**	**0.343**	**0.296**	**0.325**
Experiment III						
MMM	0.608	0.601	0.605	0.849	0.846	0.849
MMEM	0.602	**0.594**	0.599	0.817	0.805	0.813
MGAMM	0.616	0.608	0.613	NA	NA	NA
$\text{mboost}_{\text{BS}}$	0.602	**0.591**	0.600	0.622	**0.610**	0.618
$\text{mboost}_{\text{Tree}}$	0.653	0.646	0.650	0.645	0.635	0.647
BoostMLR	**0.561**	**0.564**	**0.561**	**0.576**	**0.580**	**0.577**

Values reported are test set standardized RMSE (sRMSE) averaged over 100 independently generated data with $q_y = 0$. Values displayed in bold identify the winning method for an experiment and any other method within one standard deviation of its sRMSE. NA represents that the approach failed to execute due to high dimensionality

the other non-linear models. Thus, we observe that MMM and MMEM have better prediction performance compared to MGAMM, $\text{mboost}_{\text{BS}}$ and $\text{mboost}_{\text{Tree}}$ in low dimensional settings. However, despite being a non-linear model, performance of BoostMLR is far better, which suggests that it can approximate a simple linear model. It is not surprising that a model without any mechanism for penalizing the dimensionality suffers in high dimensional settings. This explains the poor prediction performance of MMM and MMEM in high dimensional settings.

Experiment II has a non-linear functional form for $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(4)}$. Thus, performance of MMM and MMEM is poor compare to non-linear models in low dimensional settings. Among non-linear models, performance of BoostMLR is better than all other methods in low and high dimensional settings; the only method that comes close to BoostMLR is $\text{mboost}_{\text{BS}}$.

Experiment III has a unique situation where covariates enter into the model linearly but the functional forms of coefficients are non-linear and vary with time. We observe that overall, the performance of MMM and MMEM is comparable to MGAMM, $\text{mboost}_{\text{BS}}$ and $\text{mboost}_{\text{Tree}}$ in low dimensional setting. In both low and high dimensional settings, performance of BoostMLR is better; other methods that come close to BoostMLR are $\text{mboost}_{\text{BS}}$ and MMEM in low dimensional settings and $\text{mboost}_{\text{BS}}$ in high dimensional settings.

It is surprising to observe that performance of $\text{mboost}_{\text{Tree}}$ is not at par with $\text{mboost}_{\text{BS}}$. Trees are well-known for modeling multivariable interactions but not so much for modeling non-linearity. We believe this might be the reason for its relatively low performance compare to $\text{mboost}_{\text{BS}}$. High dimensionality can deteriorate performance of any approach. We observe that MGAMM fails to execute in high dimensionality. However, boosting is generally robust to high dimensionality, and we observe that the overall performance of boosting methods, including BoostMLR, does not deteriorate notably.

We compare the prediction performance of BoostMLR with an increasing number of non-informative responses using $q_y = \{0, 5, 50\}$. Results from this analysis are provided in Table 2. Comparing results for $q_y = 5$ with $q_y = 0$, we observe that having a moderate number of non-informative responses does not impair prediction performance of BoostMLR. In fact, in Experiments I and III, having a moderate number of non-informative responses improve the predictive performance, whereas in Experiment II, it impairs

Table 2 Test set performance using simulations

	$q_x = 5$			$q_x = 50$		
	$\mathbf{Y}^{(1)}$	$\mathbf{Y}^{(2)}$	$\mathbf{Y}^{(3)}$	$\mathbf{Y}^{(1)}$	$\mathbf{Y}^{(2)}$	$\mathbf{Y}^{(3)}$
Experiment I						
BoostMLR ($q_y = 0$)	0.518	0.517	0.511	0.534	0.533	0.526
BoostMLR ($q_y = 5$)	0.516	0.515	0.511	0.529	0.522	0.515
BoostMLR ($q_y = 50$)	0.532	0.512	0.527	0.547	0.522	0.531
Experiment II						
BoostMLR ($q_y = 0$)	0.331	0.288	0.314	0.343	0.296	0.325
BoostMLR ($q_y = 5$)	0.332	0.290	0.316	0.345	0.298	0.327
BoostMLR ($q_y = 50$)	0.343	0.293	0.323	0.375	0.315	0.351
Experiment III						
BoostMLR ($q_y = 0$)	0.561	0.564	0.561	0.576	0.580	0.577
BoostMLR ($q_y = 5$)	0.555	0.558	0.556	0.558	0.562	0.559
BoostMLR ($q_y = 50$)	0.558	0.557	0.556	0.554	0.556	0.554

Values reported are test set standardized RMSE (sRMSE) averaged over 100 independently generated data

prediction performance only marginally. Comparing results for $q_y = 50$ with $q_y = 0$, we observe that having a large number of non-informative responses marginally affects prediction performance of BoostMLR. However, despite including 50 additional non-informative responses in the model, the performance of BoostMLR is considerably better than other available methods from Table 1, which are evaluated without any non-informative responses.

Results for Identifying Important Covariates

Our approach can identify important covariates that are related with responses using standardized VIMP, denoted by sVIMP. sVIMP values can separate the effect of covariate into covariate main effect and covariate-time interaction effect. sVIMP values for covariate main effect (i.e., sVIMP$_{\text{main}}$) are provided in Table 3. Overall, the sVIMP approach is able to find the important covariates that affect responses in all three experiments. We observe that non-informative covariates have sVIMP values close to zero, as shown under the column noise. Thus, the sVIMP approach correctly distinguishes informative and non-informative covariates. sVIMP values are marginally smaller for the high dimensional covariate setting, which is expected due to some deterioration in the prediction performance, yet, it clearly identifies important covariates with ease. This suggests that our method is robust in identifying important covariates in high dimensional settings.

sVIMP values describing covariate-time interaction effects (i.e., sVIMP$_{\text{int}}$) for low dimensional covariate settings (i.e., $q_x = 5$) are shown in Fig. 2. The sVIMP approach clearly identifies covariates that are related with different responses at different time intervals. The top three plots

Table 3 Standardized VIMP main effect (sVIMP$_{\text{main}}$) averaged over 100 independent replications

	$q_x = 5$					$q_x = 50$				
	$\mathbf{x}^{(1)}$	$\mathbf{x}^{(2)}$	$\mathbf{x}^{(3)}$	$\mathbf{x}^{(4)}$	Noise	$\mathbf{x}^{(1)}$	$\mathbf{x}^{(2)}$	$\mathbf{x}^{(3)}$	$\mathbf{x}^{(4)}$	noise
Experiment I										
$\mathbf{Y}^{(1)}$	31.41	32.72	13.97	0.02	0.00	27.90	28.70	11.96	0.01	0.00
$\mathbf{Y}^{(2)}$	37.47	– 0.01	17.43	0.00	0.02	35.11	– 0.01	15.81	– 0.01	0.00
$\mathbf{Y}^{(3)}$	33.73	37.16	0.03	0.00	0.02	30.83	33.87	0.01	0.01	0.00
Experiment II										
$\mathbf{Y}^{(1)}$	26.31	12.83	5.79	5.69	0.01	22.99	10.48	4.82	4.57	0.00
$\mathbf{Y}^{(2)}$	41.33	0.02	10.57	14.27	0.01	36.44	0.02	9.41	11.25	0.00
$\mathbf{Y}^{(3)}$	31.99	16.96	0.02	9.67	0.02	28.06	15.02	0.01	7.27	0.00
Experiment III										
$\mathbf{Y}^{(1)}$	9.72	6.65	1.56	– 0.02	0.02	8.75	6.02	1.38	0.00	0.00
$\mathbf{Y}^{(2)}$	10.22	0.07	1.81	0.02	0.02	9.07	0.01	1.49	0.00	0.00
$\mathbf{Y}^{(3)}$	9.56	6.85	0.03	0.03	0.01	9.03	5.98	0.00	0.00	0.00

Values in the table correspond to first 4 covariates and the average VIMP value from q_x non-informative covariates provided under noise

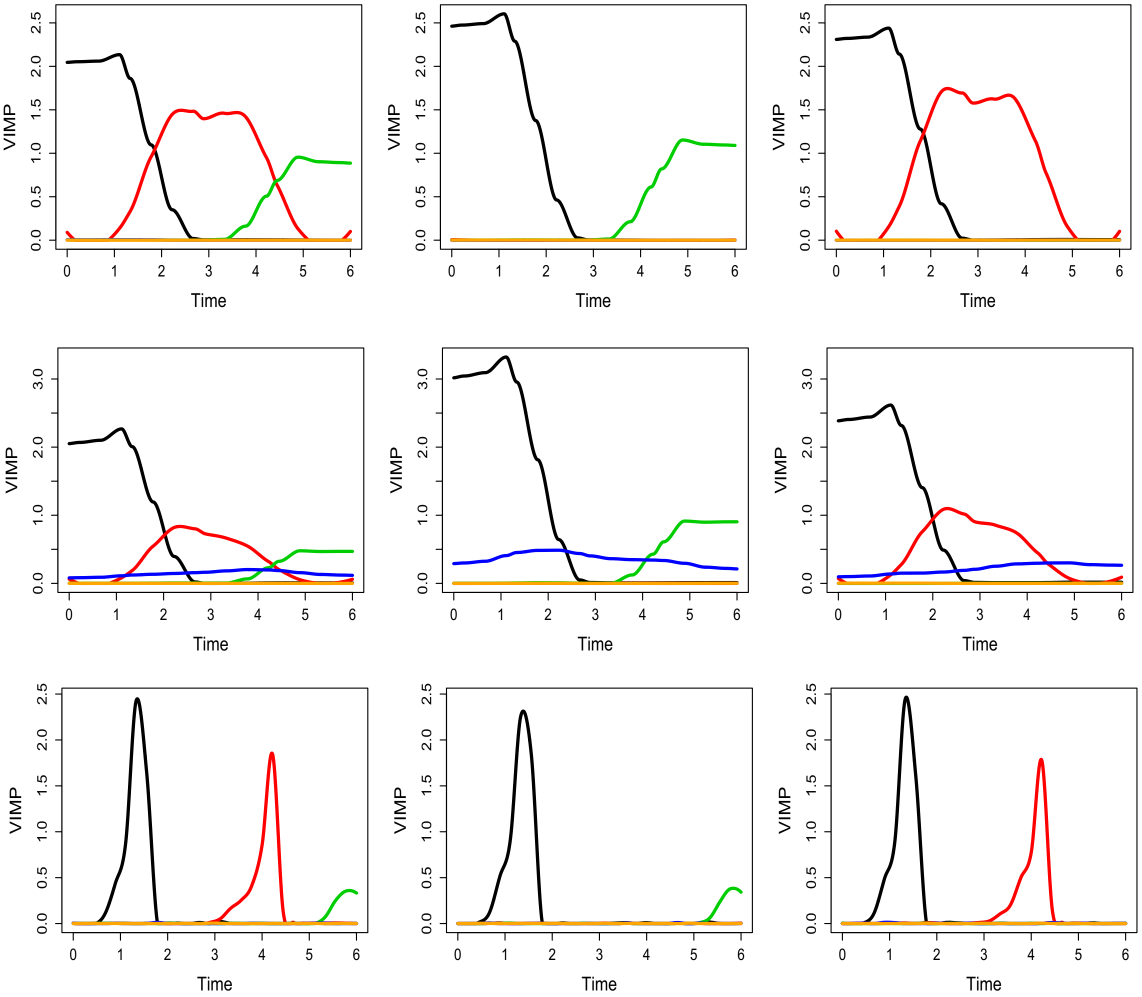

Fig. 2 Standardized VIMP for covariate-time interaction $\left(\mathrm{sVIMP}_{\mathrm{int}}\right)$ for Experiment I (top), Experiment II (middle) and Experiment III (bottom) for $q_x = 5$. Left, middle and right plots correspond to $\mathbf{Y}^{(1)}$, $\mathbf{Y}^{(2)}$ and $\mathbf{Y}^{(3)}$ respectively. Black, red, green and blue colors correspond to $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$, $\mathbf{x}^{(3)}$ and $\mathbf{x}^{(4)}$ respectively, and orange corresponds to the average VIMP values across q_x non-informative covariates

from Fig. 2 correspond to Experiment I. They show that $\mathbf{x}^{(1)}$ is related with all 3 responses at the beginning of the study, $\mathbf{x}^{(2)}$ is related with $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(3)}$ during the middle portion of the follow-up, and $\mathbf{x}^{(3)}$ is related with $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$ towards the end of the follow-up. The middle three plots from Fig. 2 correspond to Experiment II. They provide similar findings to those observed for Experiment I, except an additional constant effect of $\mathbf{x}^{(4)}$. The effect of $\mathbf{x}^{(4)}$ is constant because, although $\mathbf{x}^{(4)}$ is time-varying, the associated coefficient is constant during the entire follow-up period. Lastly, the bottom three plots from Fig. 2 correspond to Experiment III. Covariates $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ for Experiment III are time-invariant but the corresponding coefficients are functions of time as described in Fig. 1. We observe that sVIMP values approximate the relationship described in Fig. 1; for example, $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ are associated with $\mathbf{Y}^{(1)}$ and has sVIMP value zero at the beginning of their time intervals and as they reach towards the end of their time intervals, sVIMP value increases rapidly. Similar trends continue for $\mathbf{Y}^{(2)}$ and $\mathbf{Y}^{(3)}$. Throughout, we observe that non-informative covariates have their sVIMP values close to zero. Similar results are observed for high dimensional covariate setting (i.e., $q_x = 50$) and they are shown in Section S6 of the supplementary material.

Results for Identifying Important Responses

As mentioned earlier, our approach can identify not only important covariates but also important responses. This is specially useful in high dimensional response situation. Our response selection criteria can identify important responses at different time points. Figure 3 identifies important responses at different time points. To get a sense of how to identify important responses and to order them in our experiments, we need to consider the magnitude of the coefficient for each covariate corresponding to each response as well as

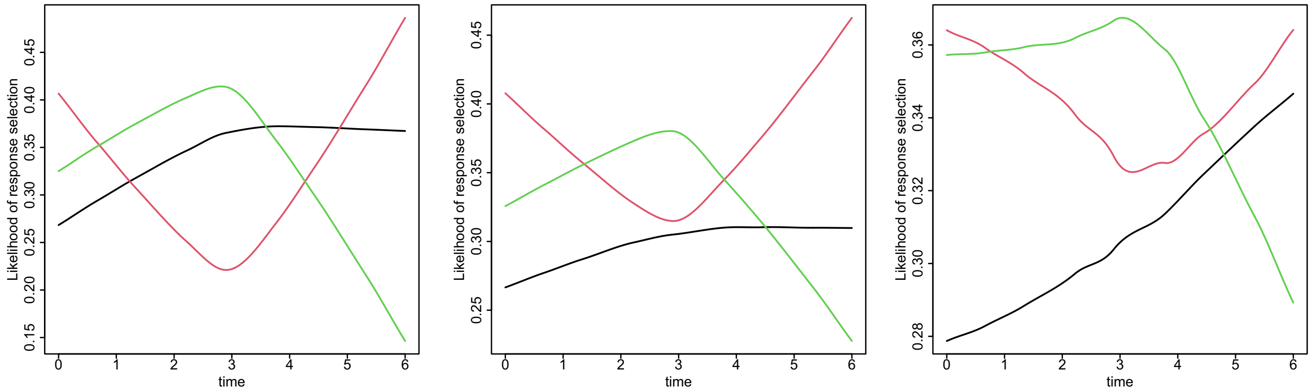

Fig. 3 Figure plots the values of likelihood of response selection across time for $\mathbf{Y}^{(1)}$ (black), $\mathbf{Y}^{(2)}$ (red) and $\mathbf{Y}^{(3)}$ (green). Left, middle and right plots correspond to Experiments I, II and III respectively, each for $q_x = 5$

the variance of each response. For example in Experiment I, we can observe that the variance of $\mathbf{Y}^{(2)}$ is lowest, followed by $\mathbf{Y}^{(3)}$ and then $\mathbf{Y}^{(1)}$. We then consider the magnitude of coefficient for each covariate. We observe that for $\mathbf{t} < 1.5$, covariate $\mathbf{x}^{(1)}$ affects all three responses and has the same magnitude of coefficient. In this situation, the important response is the one which has the lowest variance, which is $\mathbf{Y}^{(2)}$, followed by $\mathbf{Y}^{(3)}$ and $\mathbf{Y}^{(1)}$, and thus we observe that, in Fig. 3, for $\mathbf{t} < 1.5$, the responses are selected in the order of $\mathbf{Y}^{(2)}$, $\mathbf{Y}^{(3)}$ and $\mathbf{Y}^{(1)}$. Again in Experiment I, for $1.5 \le \mathbf{t} \le 4.5$, coefficient corresponding to the model for $\mathbf{Y}^{(2)}$ is zero, and thus it has the lowest likelihood, whereas coefficients corresponding to models for $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(3)}$ are same, and thus $\mathbf{Y}^{(3)}$ has a higher likelihood of getting selected followed by $\mathbf{Y}^{(1)}$. Similarly for $\mathbf{t} > 4.5$, coefficient for $\mathbf{Y}^{(3)}$ is zero, and thus it has the lowest likelihood, whereas coefficients for $\mathbf{Y}^{(1)}$ and $\mathbf{Y}^{(2)}$ are same, thus $\mathbf{Y}^{(2)}$ has a higher likelihood of getting selected followed by $\mathbf{Y}^{(1)}$. Note that even though Experiments II and III are different from Experiment I, the overall form is not very different in this context, and thus the explanation used for Experiment I can be applied for Experiments II and III. Similar results are observed for high dimensional covariate setting (i.e., $q_x = 50$) and they are shown in Section S6 of the supplementary material.

Remark 10 In addition to the above results we have also extracted estimates for correlation and dispersion parameters. These results are described in Section S6 of the supplementary material.

Clinical Laboratory Data Analysis

As an application of BoostMLR to the real data, we consider the laboratory data for the HF patients that we described in "Introduction". We use our BoostMLR approach for joint modeling of bilirubin and creatinine, and to find temporal trends for bilirubin and creatinine. (Note that the bilirubin we refer here indicates total bilirubin.) Studying the temporal trend allows the investigator to evaluate the behavior of bilirubin and creatinine and identify their critical levels for the high risk patients before and after the heart transplant. Also, by joint modeling of bilirubin and creatinine, we can identify the risk factors that influence their trajectories. Using the risk factor information, investigator can predict the future risk for new patients and control the risk factors to minimize this risk.

The laboratory data is based on $n = 459$ patients who were listed for heart transplant and were put on MCS through device implantation from December 1991 to July 2009 at Cleveland Clinic. These patients had periodic measurements of their bilirubin and creatinine. A total of 18285 measurements of bilirubin and creatinine were available following device implantation with an average of 39 measurements per patient. Tables 4 and 5 provide patients' characteristics. By the end of the study follow-up, 312 patients had heart transplant. Time of heart transplant varied among patients, and thus, to identify levels of bilirubin and creatinine before and after the transplant, instead of using the observed time, we created a new time variable, referred as centered time, such that, if the patient had heart transplant, we subtract his/her time of the transplant from the observed time (at which bilirubin and creatinine were measured), and if the patient didn't receive heart transplant, we subtract his/her maximum observed time from the observed time. By doing that, for the heart transplant patient, the centered time takes negative values before the transplant, zero at the time of transplant, and positive values after the transplant, whereas, for the non-transplant patient, centered time takes non-positive values.

The laboratory data is analyzed using our BoostMLR approach, implemented using the R package `BoostMLR`. The package includes the laboratory data as well as sample

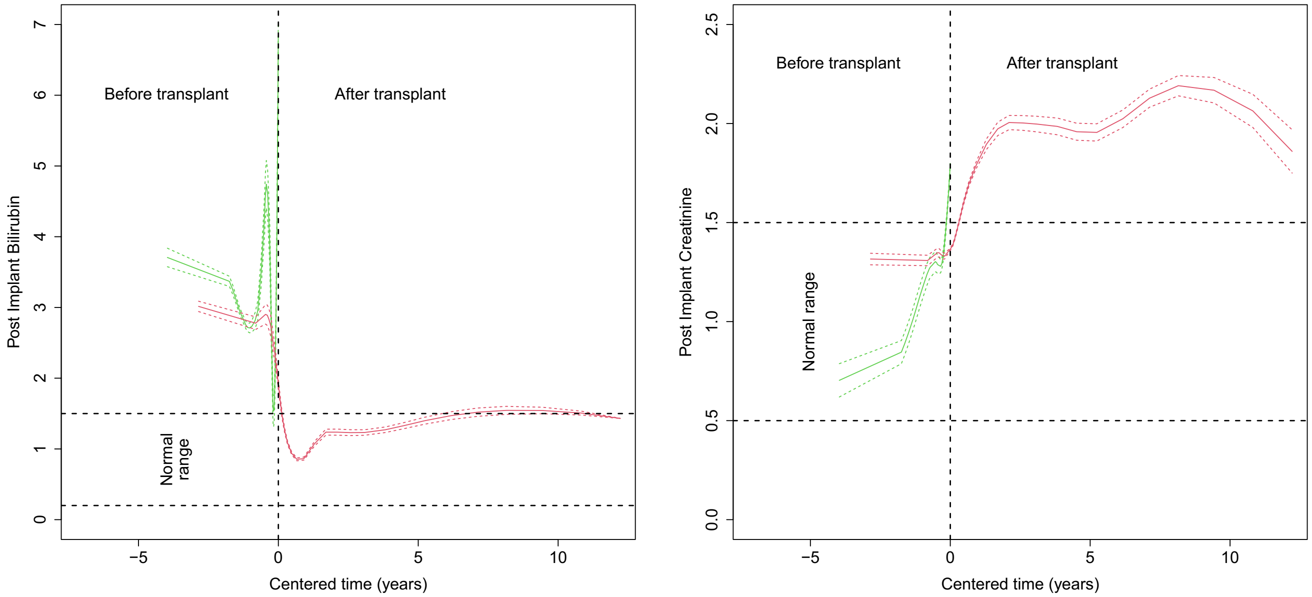

Fig. 4 Left and right plots respectively correspond to the mean predicted levels of post implant bilirubin and creatinine. In both plots, red lines corresponds to patients who received heart transplant, and green lines corresponds to patients who didn't receive heart transplant

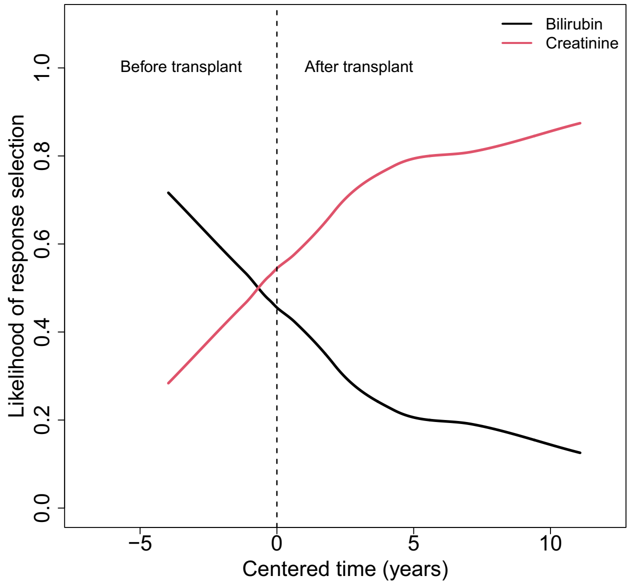

Fig. 5 Likelihood of selection of bilirubin and creatinine

codes used for analysis of this and the simulated data from "Simulation". We used cubic B-spline for mapping of continuous covariates and time respectively using 5 and 25 equally spaced knots. The ℓ_1 penalization procedure is repeated 100 times to generate the distribution of noised-up estimate and 15th and 85th percentiles of that distribution are used to set limits for lower and upper bound respectively (see Section S2 of the supplementary material for details). We use the learning rate $\nu = 0.1$. We set the total number of boosting iterations $M = 500$, however M is estimated as a part of the boosting iteration to avoid model overfitting. The original data was randomly split into the training (350 patients) and test (109 patients) data. The training data was used to build the model and the test data was used to calculate VIMP.

Figure 4 shows the mean predicted levels of bilirubin and creatinine, stratified by transplant status. Figure shows that levels of bilirubin and creatinine are different among patients who received transplant compared to patients who didn't receive transplant. There are multiple factors that were considered in order for patient to qualify for the transplant and we observed that patients who didn't receive heart transplant have very high bilirubin levels compared to patients who received transplant. (We observed a spike at zero with bilirubin value around 7.) It is possible that impaired levels of bilirubin could have played a role in deciding whether to proceed with transplant or not. This is because impaired levels of bilirubin at the time of transplant is a risk factor for mortality [39]. We observed different effects of transplant on bilirubin and creatinine. Bilirubin level normalized after the transplant, whereas, creatinine level risen beyond the normal range. In Fig. 5 we plotted the likelihood of selection of bilirubin and creatinine at different time points. Plot shows that bilirubin is most likely to be selected as an important response before the transplant, whereas, creatinine is most likely to be selected as an important response after the transplant. Findings from Figs. 4 and 5 suggest that investigator should focus more on the levels of bilirubin before the transplant and levels of creatinine after the transplant. This allows investigator to allocate resources in collecting

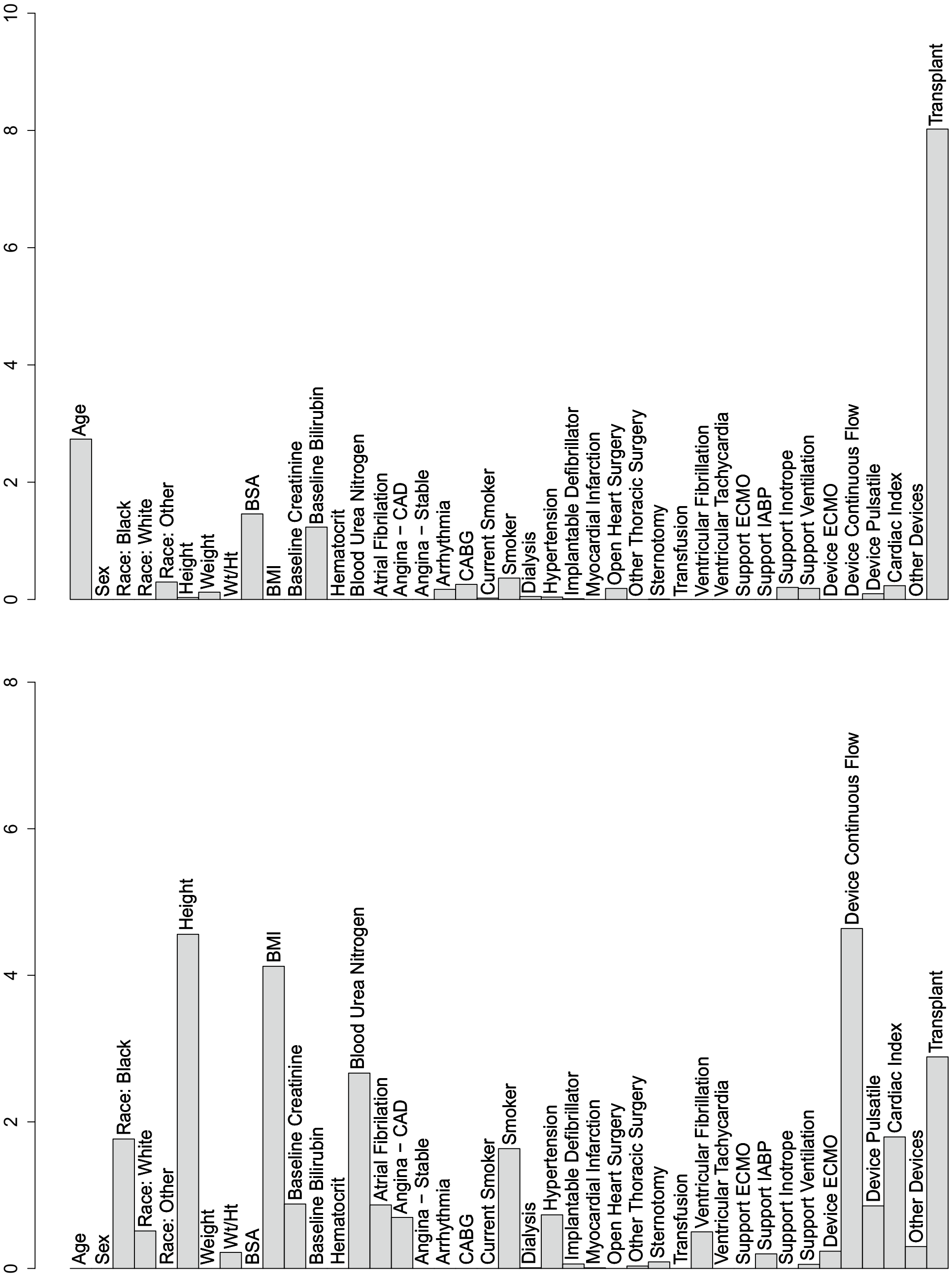

Fig. 6 Top and bottom plots respectively represent the standardized VIMP main effect (sVIMP$_{\text{main}}$) for each covariate corresponding to the post implant bilirubin and creatinine

bilirubin and creatinine at different phases of management of high risk HF patients.

To understand factors that contributed to the trajectories of bilirubin and creatinine, we consider the covariate main effects and covariate-time interaction effects using VIMP approach. The standardized VIMP main effect (sVIMP$_{\text{main}}$) for each covariate is shown in Fig. 6, whereas the standardized VIMP covariate-time interaction effect (sVIMP$_{\text{int}}$) is shown in Fig. 7. In case of bilirubin, transplant status, age, BSA and baseline bilirubin are covariates that have high covariate main effects and covariate-time interaction effects. In case of creatinine, height, device continuous flow (which is a type of MCS device), BMI, blood urea nitrogen, history of smoking and hypertension have high covariate main effects and covariate-time interaction effects. Other covariates which have high covariate-time interaction effects with bilirubin are BMI and inotrope (a type of life support). When we dissect the covariate-time interaction effects, we observed that, in case of bilirubin, age and BMI affected the bilirubin levels before transplant, and the bilirubin levels were very different between patients who didn't receive transplant and patients who received transplant. On the other hand, in case of creatinine, BMI and blood urea nitrogen affected creatinine levels before transplant, whereas height,

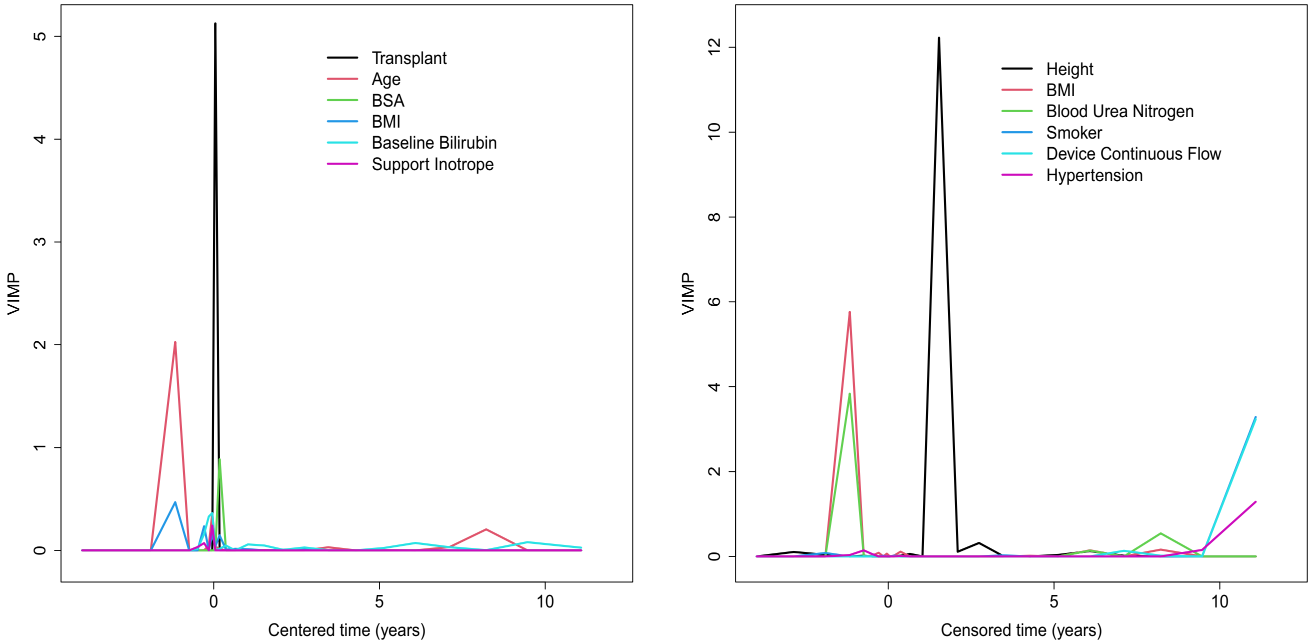

Fig. 7 Left and right plots respectively represent the standardized VIMP of covariate-time interaction effect (sVIMP$_{\text{int}}$) for the post implant bilirubin and creatinine

Table 4 Summary statistics from laboratory data

	N	Summary		*N*	Summary
Demography					
Age	459	52.71± 12.25	Race (black)	456	54(12)
Sex (female)	459	83(18)	Race (other)	456	13(3)
Race (white)	456	389(85)			
Body size					
Height	451	174.50 ± 9.33	BSA	391	2.03 ± 0.27
Weight	393	83.88 ± 18.68	BMI	391	27.40 ± 5.09
Weight/height	391	0.48 ± 0.09			
Lab values prior to implantation					
Baseline creatinine	382	1.72 ± 1.43	Blood urea nitrogen	378	37.73 ± 21.91
Baseline bilirubin	371	1.73 ± 1.39	Hematocrit	352	32.73 ± 5.05
Hemodynamics prior to implantation					
Cardiac Index	288	1.93 ± 0.58			
Patient history					
Atrial fibrillation	459	79 (17)	Implantable defibrillator	459	111 (24)
Angina-CAD	459	22 (5)	Myocardial infarction	459	147 (32)
Angina-stable	459	76 (17)	Open heart surgery	459	146 (32)
Arrhythmia	459	93 (20)	Other thoracic surgery	459	116 (25)
CABG	459	92 (20)	Sternotomy	459	109 (24)
Current smoker	459	41 (9)	Transfusion	459	142 (31)
Smoking	459	106 (23)	Ventricular fibrillation	459	21 (5)
Dialysis	459	19 (4)	Ventricular tachycardia	459	127 (28)
Hypertension	459	116 (25)			

Values in the table are mean ± standard deviation or n(%)

Table 5 Summary statistics from laboratory data

	N	Summary		*N*	Summary
Life support					
ECMO	459	27 (6)	Inotrope	459	312 (68)
IABP	459	257 (56)	Ventilation	459	147 (32)
Device type and placement					
ECMO	459	82 (18)	Pulsatile	459	256 (56)
Other devices[a]	459	374 (81)	Continuous flow	459	75 (16)

Values in the table are *n* (%)

[a]Other devices include LVAD, TAH and Bivad

device continuous flow, history of smoking (both device continuous flow and history of smoking have similar VIMP values) and hypertension affected creatinine levels after the transplant.

Conclusion

In this article, we presented new gradient boosting approach for joint modeling of multivariate longitudinal responses. Our approach is equipped to handle time-varying covariates. Use of gradient boosting allows our approach to handle high dimensionality of covariates with ease and maintain high prediction performance, contrary to other non-boosting approaches that either break or suffer in terms of prediction performance.

One of the unique features of our approach is the handling of high dimensionality of responses. Literature on joint modeling of multiple responses and particularly high dimensionality of responses, is rare. Simulation results show that prediction performance of our approach does not deteriorate even in high dimensional response situations. Another unique feature of our approach is the ability to identify covariates that affect responses differently at different time intervals. Such feature has an important applications in observational studies where certain covariates are known to affect response trajectory at different time points. Additionally, our approach has the ability to identify important responses at different time points. This allows investigator to allocate resources in collecting different information at different time points. For example, in case of laboratory data, we observed that bilirubin was an important response before transplant whereas creatinine was an important response after transplant. Thus, in summary, our approach addresses important aspects of analysis of longitudinal study. This is specifically relevant to investigator whose objective is to identify important covariates and responses in the data, and once identified, learn about how the relationships between them are varying across time.

Our approach has some limitations. Our approach will not be effective in situations when the true model includes interactions among multiple covariates. Interactions among covariates can be incorporated in our approach by explicitly creating such variables and adding them into the model. However, more efficient way to incorporate interaction is to use regression tree as a base learner. We developed a tree-based gradient boosting method for modeling longitudinal response when covariates are time-invariant [27]. Tree-based boosting with time-varying covariates is challenging and the literature on this do not provide satisfactory results. The challenging part is in the derivation of tree splitting rule which should provide an optimal separation of parent node into the daughter nodes and yet maintains all observations for the specific subject undivided. Another limitation of our approach is that we assume an exchangeable correlation structure of the correlation matrix. This is our attempt to keep the structure of the correlation matrix simpler. In a future update of `BoostMLR` package, we will provide some additional structural forms of the correlation matrix.

In this article, we focus on situation where the function $G(\cdot)$ is a function of single variable, namely $\mathbf{t}$. In future, we will extend our approach where function $G(\cdot)$ can accommodate multiple time-varying covariates simultaneously. Another possible extension is when function $G(\cdot)$ simultaneously accommodates $\mathbf{t}$ and $\mathbf{t}_k$, where $\mathbf{t}_k$ represents time points at which kth time-varying covariate is measured and it need not coincide with time $\mathbf{t}$ when the responses are measured. Additional possible extension of our approach is the extension to categorical responses.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s42979-022-01072-6.

Acknowledgements Heart, Vascular and Thoracic Institute, Cleveland Clinic provided funding for this research.

Declarations

Conflict of interest All authors declare that they have no conflict of interest.

References

1. Fitzmaurice GM, Laird NM, Ware JH. Applied Longitudinal Analysis. 2nd ed. Hoboken: Wiley Press; 2011.
2. Majid M, Farveh V, Ahmad A. Liver diseases in heart failure. Heart Asia;143–149;2011.
3. Mark Sarnak. A patient with heart failure and worsening kidney function. Clin J Am Soc Nephrol. 2014;9(10):1790–8.
4. Rajeswaran J, Blackstone EH, Bernard J. Evolution of association between renal and liver function while awaiting for the heart transplant: an application using bivariate multiphase nonlinear mixed effect model. Stat Methods Med Res. 2018;27(7):2216–30.
5. Liang KY, Zeger SL. Longitudinal data analysis using generalized linear models. Biometrika. 1986;73:13–22.
6. Laird NM, Ware JH. Random-effect models for longitudinal data. Biometrics. 1982;38:963–74.

7. Cho H. The analysis of multivariate longitudinal data using multivariate marginal models. J Multivar Anal. 2016;143:481–91.
8. Asar O. On multivariate binary longitudinal data models and their application in forecasting. MS Thesis, Middle East Technical University; 2012.
9. Komarek A, Komarkova L. Capabilities of R package mixAK for clustering based on multivariate continuous and discrete longitudinal data. J Stat Softw. 2014;59(12):1–38.
10. Giltinan D, Davidian M. Nonlinear models for repeated measurement data. London: Chapman & Hall; 1995.
11. Staniswalis JG, Lee JJ. Nonparametric regression analysis of longitudinal data. J Am Stat Assoc. 1998;93(444):1403–18.
12. Lin X, Carroll RJ. Nonparametric function estimation for cluster data when the predictor is measured without/with error. J Am Stat Assoc. 2000;95(450):520–34.
13. Welsh AH, Lin X, Carroll RJ. Marginal longitudinal nonparametric regression: locality and efficiency of spline and kernel methods. J Am Stat Assoc. 2002;97(458):482–93.
14. Fan J, Zhang W. Statistical estimation in varying coefficient models. Ann Stat. 1999;27(5):1491–518.
15. Cai Z, Fan J, Li R. Efficient estimation and inferences for varying-coefficient models. J Am Stat Assoc. 2000;95(451):888–902.
16. Fan J, Zhang W. Statistical methods for varying coefficient models. Stat Infer. 2008;1:179–95.
17. Sela RJ, Simonoff JS. RE-EM trees: a data mining approach for longitudinal and clustered data. Mach Learn. 2012;86:169–207.
18. Mandel F, Ghosh RP, Barnett I. Neural networks for clustered and longitudinal data using mixed effects models. Biometrics. https://doi.org/10.1111/biom.13615.
19. Wood SN. Low rank scale invariant tensor product smooths for generalized additive mixed models. Biometrics. 2006;62(4):1025–36.
20. Hoover DR, Rice JA, Wu CO, Yang L-P. Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data. Biometrika. 1998;85(4):809–22.
21. Huang JZ, Wu CO, Zhou L. Varying coefficient models and basis function approximations for the analysis of repreated measurements. Biometrika. 2002;89(1):111–28.
22. Chiang CT, Rice JA, Wu CO. Smoothing splines estimation for varying coefficient models with repeatedly measured dependent variables. J Am Stat Assoc. 2001;96(454):605–19.
23. Blackstone EH, Naftel DC, Turner ME Jr. The decomposition of time-varying hazard into phases, each incorporating a separate stream of concomitant information. J Am Stat Assoc. 1986;81:615–24.
24. Friedman JH. Greedy function approximation: a gradient boosting machine. Ann Stat. 2001;29:1189–232.
25. Wang L, Li H, Huang JZ. Variable selection in nonparametric varying coefficient models for analysis of repeated measurements. J Am Stat Assoc. 2008;103(484):1556–69.
26. Friedman J, Hastie T, Tibshirani R. Additive logistic regression: a statistical view of boosting (with discussion). Ann Stat. 2000;28(2):337–74.
27. Pande A, Li L, Rajeswaran J, Ehrlinger J, Kogalur UB, Blackstone Eugene H, Ishwaran H. Boosted multivariate trees for longitudinal data. Mach Learn. 2017;106(2):277–305.
28. Tutz G, Reithinger F. A boosting approach to flexible semi parametric mixed models. Stat Med:26(14),2872–2900;2007.
29. Tutz G, Groll A. Generalized linear mixed models based on boosting. Stat Model Regress Struct:197–215;2010.
30. Yue M, Li J, Cheng MY. Two-step sparse boosting for high dimensional longitudinal data with varying coefficients. Comput Stat Data Anal. 2019;131:222–34.
31. Hothorn T, Buhlmann P, Kneib T, Schmid M, Hofner B. Model-based boosting 2.0. J Mach Learn Res. 2010;11:2109–13.
32. Lutz RW, Buhlmann P. Boosting for high multivariate responses in high dimensional linear regression. Stat Sin. 2006;16:471–94.
33. Buhlmann P, Yu B. Boosting with L_2 loss: regression and classification. J Am Stat Assoc. 2003;98(462):324–39.
34. Buhlmann P. Boosting for high-dimensional linear models. Ann Stat. 2006;34(2):559–83.
35. De Boor C. A practical guide to splines. Berlin: Springer; 1978.
36. Pande A. Boosting model for longitudinal data. Ph.D. dissertation, University of Miami; 2017.
37. Pande A, Ishwaran H. BoostMLR: boosting for multivariate longitudinal response, 2021. R package version 1.0.3.
38. Asar O, Ilk O. mmm: an R package for analyzing multivariate longitudinal data with multivariate marginal models. Comput Methods Programs Biomed. 2013;112:649–54.
39. Hunt SA, Abraham WT, Chin MH, et al. American College of Cardiology, American Heart Association,. guideline update for the diagnosis and management of chronic heart failure in the adult. Circulation. 2005;112:1824–1852.

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