



Tree-Structured Mixed-Effects Regression Modeling for Longitudinal Data

Author(s): Soo-Heang Eo and HyungJun Cho

Source: *Journal of Computational and Graphical Statistics*, September 2014, Vol. 23, No. 3 (September 2014), pp. 740-760

Published by: Taylor & Francis, Ltd. on behalf of the American Statistical Association, Institute of Mathematical Statistics, and Interface Foundation of America

Stable URL: <https://www.jstor.org/stable/43304920>

REFERENCES

Linked references are available on JSTOR for this article:

https://www.jstor.org/stable/43304920?seq=1&cid=pdf-reference#references_tab_contents

You may need to log in to JSTOR to access the linked references.

JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive. We use information technology and tools to increase productivity and facilitate new forms of scholarship. For more information about JSTOR, please contact support@jstor.org.

Your use of the JSTOR archive indicates your acceptance of the Terms & Conditions of Use, available at <https://about.jstor.org/terms>



JSTOR

Taylor & Francis, Ltd., American Statistical Association, and Institute of Mathematical Statistics are collaborating with JSTOR to digitize, preserve and extend access to *Journal of Computational and Graphical Statistics*

Tree-Structured Mixed-Effects Regression Modeling for Longitudinal Data

Soo-Heang EO and HyungJun CHO

Tree-structured models have been widely used because they function as interpretable prediction models that offer easy data visualization. A number of tree algorithms have been developed for univariate response data and can be extended to analyze multivariate response data. We propose a tree algorithm by combining the merits of a tree-based model and a mixed-effects model for longitudinal data. We alleviate variable selection bias through residual analysis, which is used to solve problems that exhaustive search approaches suffer from, such as undue preference to split variables with more possible splits, expensive computational cost, and end-cut preference. Most importantly, our tree algorithm discovers trends over time on each of the subspaces from recursive partitioning, while other tree algorithms predict responses. We investigate the performance of our algorithm with both simulation and real data studies. We also develop an R package `melt` that can be used conveniently and freely. Additional results are provided as online supplementary material.

Key words: Mixed-effects model; Recursive partitioning; Regression tree.

1. INTRODUCTION

Data visualization and model interpretation are as important as building an optimal predictive model. A tree-structured model has emerged as one of the solutions capable of achieving these purposes. Since the development of the automatic interaction detection (AID) algorithm by Morgan and Sonquist (1963) for a univariate response, the decision tree has become very popular in a variety of fields. The appearance of classification and regression tree (CART) (Breiman et al. 1984) and fast algorithm for classification tree (FACT) (Loh and Vanichsetakul 1988) algorithms subsequently had a strong effect on the field of decision trees, because of the pruning technique of CART and the variable selection approach of FACT. In particular, FACT and its offspring (e.g., generalized, unbiased interaction detection and estimation [GUIDE]; Loh 2002, 2009) were developed to reduce the computational cost and bias in selecting split variables by statistical tests. It is computationally efficient, retaining excellent model accuracy and interpretability. As a result, these

HyungJun Cho is Associate Professor, Department of Statistics, Korea University, Seoul, Korea (E-mail: hj4cho@korea.ac.kr). Soo-Heang Eo, Department of Statistics, Korea University, Seoul, Korea (E-mail: hanansh@korea.ac.kr).

© 2014 American Statistical Association, Institute of Mathematical Statistics,
and Interface Foundation of North America

Journal of Computational and Graphical Statistics, Volume 23, Number 3, Pages 740–760

DOI: 10.1080/10618600.2013.794732

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/r/jcgs.

algorithms have greatly influenced tree modeling methods. It follows that a number of extended versions have been developed, for example, the quantile regression tree (Chaudhuri et al. 1994), survival tree (Ahn and Loh 1994), regression impurity tree (Alexander and Grimshaw 1996), Bayesian tree (Chipman, George, and McCulloch 1998), multiway split tree called classification rule with unbiased interaction selection and estimation (CRUISE) (Kim and Loh 2001), unbiased regression tree called GUIDE (Loh 2002), unbiased logistic tree called logistic tree with unbiased selection (LOTUS) (Chan and Loh 2004), and so on. Hothorn and Lausen (2003) and Shih and Tsai (2004) also proposed unbiased split rules by using maximally selected splitting statistics and asymptotic distributions for changing points. However, these are computationally too demanding with large datasets.

Gillo and Shelly (1974) proposed an approach for multivariate responses by modifying the AID algorithm. About 20 years after reinventing tree models with a longitudinal study, Segal (1992) developed a regression tree method using the Mahalanobis distance as an impurity for continuous longitudinal data and Zhang (1998) proposed a classification tree method for multiple binary response data, both of which are inherited from CART. Segal's algorithm (Segal 1992) was implemented into an R package *mvpart* by De'Ath (2002). Focusing on a regression tree for longitudinal data, Siciliano and Mola (2000) developed a binary segmentation procedure based on a two-stage splitting criterion, which concentrates on reducing the computational cost. Abdoell et al. (2002) proposed a binary partitioning algorithm to find an optimal cutoff point based on a mixed-effects model with just one continuous predictor variable. These algorithms, however, may generate undue selection bias and have a huge computational cost due to the use of an exhaustive search (ES) in selecting the optimal splits. Lee et al. (2005) attempted to reduce variable selection bias for longitudinal data by adopting the splitting idea of smoothed and unsmoothed piecewise-polynomial regression trees (SUPPORT) (Chaudhuri et al. 1994) with the generalized estimating equation. Hsiao and Shih (2007) extended the GUIDE regression tree to multivariate continuous data. This algorithm controls selection probabilities by residual analysis (RA) using a three-way contingency table. Hothorn, Hornik, and Zeileis (2006) proposed an alternative approach to avoid variable selection bias by using the conditional distribution of linear statistics based on a permutation test.

More recently, based on the CART prediction algorithm, Hajjem, Bellavance, and Larocque (2011) and Sela and Simonoff (2012) independently proposed a mixed-effects regression tree for clustered data with unbalanced structures, where CART can be replaced with another algorithm such as GUIDE (Loh 2009). Based on the R package *rpart* (Therneau, Atkinson, and Ripley 2013) for CART, Sela and Simonoff (2012) implemented their algorithm into an R package *REEMtree* (Sela and Simonoff 2011). Loh and Zheng (2013) developed regression trees for longitudinal and multiresponse data that were included in the GUIDE program. These algorithms focused on the prediction of responses by minimizing the differences between observed and predicted responses in each node and showed the improved prediction power over linear modeling or other tree modeling.

In our study, we propose a tree algorithm for longitudinal data based on GUIDE, called mixed-effects longitudinal tree (MELT), which uses the merits of both decision tree and mixed-effects modeling approaches. The reason for using a mixed-effects model is that it enables the consideration of within-subject correlations and between-subject heterogeneity. Irregular data structures and any types of predictor variables are allowed.

Most importantly, our tree algorithm discovers trends over time on each of the subspaces from recursive partitioning, while other tree algorithms predict responses. It reveals changes of the responses over time within the same subject because it is natural to investigate the changes in longitudinal data analysis.

By combining the univariate and linear split algorithms, we attempt to reduce the computational cost while retaining good interpretability. Linear splits are used when neither main-effects nor local interaction tests work well. In addition, the two-level split search method is applied to enable interaction tests to be more useful. Finally, a multistep stopping rule, which determines an optimal tree size with less computing time than cost-complexity pruning is considered.

The rest of the article is organized as follows. In Section 2, our proposed algorithm is described by way of the new impurity functions and split selection algorithms for longitudinal data. In Section 3, a simulation study is performed to show the effectiveness of our proposed method, including a comparison with other algorithms for variable selection. In Section 4, MELT and its counterparts are demonstrated by using a real dataset. The detailed algorithm is summarized in the Appendix and the additional results are provided in the online supplementary material.

2. PROPOSED METHOD

We now describe a new piecewise mixed-effects modeling method, MELT, that combines the merits of tree-structured modeling and mixed-effects modeling for longitudinal data. The main merits of our proposal are the advantage gained by using RA for splitting with negligible selection bias and relatively low computational cost, and the use of random and fixed effects for detecting trend-over-time with the heterogeneity of subjects. We also retain the benefits of model interpretability and flexibility for the analysis of longitudinal data.

2.1 BASIC MODEL

With n independent subjects, often persons in practice, let (X_i, Y_i) be observations for subject i where Y_i is a vector on q_i responses, y_{i1}, \dots, y_{iq_i} , at the q_i follow-up time points and X_i is its corresponding covariate matrix with p predictors, $\{(x_{i11}, \dots, x_{iq_i1})^T, \dots, (x_{i1p}, \dots, x_{iq_ip})^T\}$, where $i = 1, \dots, n$. We consider a tree-structured regression model with q_i response values in a real space $Y_i \subset \mathbb{R}^{q_i}$, where $q_i \geq 2$.

Our goal is to provide a predictive and interpretable rule to discover various trends-over-time and the patterns of \mathbf{Y} from \mathbf{X} . Figure 1 provides a motivating example for our development. At first, there is no relationship between the response and the time variable. However, different trends clearly become evident over time after splitting by *gender*. As we see in this simple example, we hope to find a variety of meaningful patterns over time after recursive partitioning.

To build our tree-based model, we define the following mixed-effects model with a random intercept and a fixed time effect as a basic model

$$y_{ij} = \alpha_i + \beta(\text{time})_{ij} + \epsilon_{ij}, \quad (1)$$

where α_i is a random effect with $N(\alpha_0, \sigma_a^2)$, β is a fixed unknown parameter, and $(\text{time})_{ij}$ is corresponding times at which the measurements are taken. It is assumed that the error term

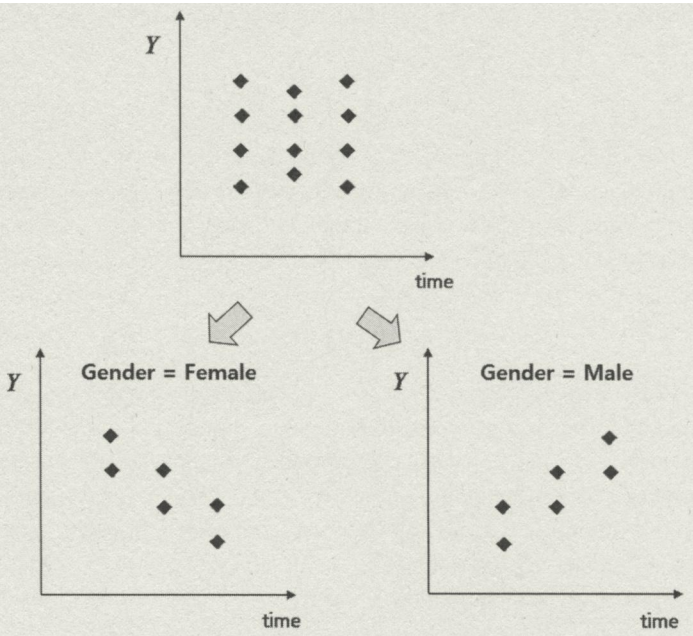


Figure 1. A motivating example of partitioning longitudinal data.

$\epsilon = (\epsilon_{11}, \dots, \epsilon_{1q_1}, \epsilon_{21}, \dots, \epsilon_{2q_2}, \epsilon_{n1}, \dots, \epsilon_{nq_n})' \sim N(0, \Lambda)$, where Λ is a certain covariance matrix. The basic model allows subjects to have the same slope at the same group (node), but they can have subject-specific intercepts even though they are in the same group. If it is not enough to explain trend-over-time by linearity, the model can be extended flexibly by

$$y_{ij} = \alpha_i + \sum_{k=1}^K \beta_k(\text{time})_{ij}^k + \epsilon_{ij}, \tag{2}$$

where K is determined by the complexity of the data, for example, $K = 2$ or 3 . A low-order polynomial is usually used because of simplicity and interpretability.

2.2 IMPURITY FUNCTION

An impurity measure is the most fundamental factor in constructing a tree. When constructing a regression tree for longitudinal data, the previous algorithms employed the Mahalanobis distance by Segal (1992), the generalized estimating equation by Lee et al. (2005), or the mean dependency on the response by Siciliano and Mola (2000) as impurity functions. We aim to divide subjects into several groups, each of which has its own trend over time, when the slopes are heterogeneous between terminal nodes and homogeneous within terminal nodes. For this purpose, the impurity measures are not proper. Thus, we introduce new impurity functions to discover the large heterogeneity bottom subnodes.

It is natural to consider the negative restricted log-likelihood as an impurity function. Let $y_i = (y_{i1} \dots, y_{iq_i})$ be the observed sequence of measurements on the i th subject and U be an $nq_i \times 1$ matrix of time variable with $\{n(i - 1) + j\}$ th row of $(\text{time})_{ij}$, where

$i = 1, \dots, n$ and $j = 1, \dots, q_i$. The negative restricted log-likelihood is defined as

$$R(t) = 0.5[\log |\Lambda_t| + \log |U' \Lambda_t^{-1} U| + (y_t - \mu_t)' \Lambda_t^{-1} (y_t - \mu_t)/2], \quad (3)$$

where $y_t = (y_1^T, \dots, y_n^T)$, $n_t = \sum_{i \in t} q_i$, and $E(y_t) = \mu_t$, $\text{var}(y_t) = \Lambda_t$ for node t (Diggle et al. 2002). The smaller the impurity, the better the fit. However, this function does not account for trends over time. A more intuitive measure of impurity is to use the distance of subject-specific slopes from the common slope of all the subjects at each node. Let β_{ti} be a slope of subject i and β_t be the common slope of all the subjects at node t . We define the new impurity as

$$R(t) = \sum_{i \in t} (\hat{\beta}_{ti} - \hat{\beta}_t)^2. \quad (4)$$

This measures the difference between the individual slopes and the common slope from fitting the basic model, and is small when the basic model, which is linear in time, is appropriate regardless of individuals' intercepts. Thus, this measure reflects the goodness of fit of the basic model with a random intercept and a fixed slope for time. We call the impurity function the sum of slopes residual squares (SSRS). For the more flexible complex model in Equation (2), we can define the impurity as

$$R(t) = \sum_{k=1}^K \sum_{i \in t} (\hat{\beta}_{t ki} - \hat{\beta}_{tk})^2, \quad (5)$$

where $\hat{\beta}_{t ki}$ and $\hat{\beta}_{tk}$ are the estimates of the k th-order individual and common slopes at node t .

2.3 FINDING SPLIT RULES

A natural and naive approach for finding splits is to select a split yielding the greatest reduction of impurity. The simplest way to find a split is to evaluate the reduction of impurity, $\Delta(t) = R(t) - [R(t_L) + R(t_R)]$, over all possible splits, so as to select the best cutoff point that generates the largest value of $\Delta(t)$, where $R(t)$, $R(t_L)$, and $R(t_R)$ are the loss functions of node t , its left branch t_L , and its right branch t_R , respectively. Because most existing tree algorithms such as CART and C4.5 (Quinlan 1993) use an exhaustive search (ES) algorithm for finding splits, they have issues over variable selection bias (Doyle 1973; Loh and Shih 1997). In addition, they suffer from impractical problems such as the substantial computational cost and end-cut preference. To solve these problems, Loh (2002) used the residual-based approach, and Hothorn and Lausen (2003) and Shih and Tsai (2004) used maximally selected splitting statistics and asymptotic distributions for changing points. However, the latter two approaches are computationally expensive. The manner in which the ES approach is especially vulnerable to both selection bias of split variable and computational cost is investigated for longitudinal data in Section 3.

For selecting optimal split rules, we use residuals arising from fitting the above basic model to longitudinal data, like the statistical approach in the GUIDE classification tree (Loh 2009). We use such an RA approach, to solve the well-known problems mostly found in the ES approaches. Unlike ES, RA selects split rules by investigating the randomness of residuals after fitting the basic model to the data at each node. Selecting the split variable and split point (or set) separately achieves the negligibility of selection bias as well as

the substantial reduction of the computing time. RA also reacts less sensitively to extreme cases and its split selection depends on the association between the response and predictor variables rather than the distributions of predictor variables, see Cho and Hong (2008).

To apply the RA approach to longitudinal data, we define new residuals as

$$r_i = \hat{\beta}_i - \hat{\beta}. \quad (6)$$

These are individual components of our new impurity residuals of SSRS. In general, if the fitted model is correct, the residuals should be randomly distributed over each predictor. The randomness of residuals means that the fitted model is sufficient to explain the data at the node so that no further split is needed. For Equation (2), we consider residuals as

$$r_{ki} = \hat{\beta}_{ki} - \hat{\beta}_k. \quad (7)$$

All the combinations of residual signs compose one variable and the RA approach is applied. For instance, consider a 2-degree polynomial and assume that positives (+) and negatives (−) are obtained from residuals minus the median of residuals for each k . Then, ++, +−, −+, and −− can be treated as four categories of a variable.

We next separate the task of split selection into two parts: selecting the split variable and then finding the split set or point. Such a separation approach was originally proposed by Loh and Vanichsetakul (1988), and several improved or extended algorithms have been developed: improved FACT (Kademan, Loh, and Vanichsetakul 1989), SUPPORT (Chaudhuri et al. 1994), Quick, Unbiased, Efficient, Statistical Tree (QUEST) (Loh and Shih 1997), CRUISE (Kim and Loh 2001), GUIDE (Loh 2002, 2009), LOTUS (Chan and Loh 2004), and STUDI (Cho and Hong 2008).

2.3.1 Selection of Split Variable Using Main-Effect Test. For selecting the most significant variable for splitting at each node, we adopt the selection algorithm of GUIDE, which is efficient computationally and free of selection bias. Its selection is divided into three major parts: the main effect, the local interaction, and the linear split tests.

The main effect test consists of conducting the Pearson chi-squared test for the independence of each predictor variable and its residuals. This procedure is segregated by the type of predictor variables. If a predictor is categorical, we construct a contingency table with the signs of the residuals as rows and the number of classes as columns. If a variable is noncategorical, we divide the data into three or four levels depending on the sample mean and deviation, and form a contingency table with the signs as rows and divide points (three or four) as columns. Then, the Wilson–Hilferty approximation (Wilson and Hilferty 1931), $W_M(X)$, is used to transform each value with a χ_p^2 to a standard normal. The most influential split variable is selected by the value of $W_M(X)$ using a Bonferroni-corrected significance threshold. The detailed algorithm is described in the Appendix.

2.3.2 Selection of Split Variable Using Interaction Test. The local interaction effect test is used to assess the local interaction of each pair of predictor variables. It makes a model easier to interpret by splitting on a predictor of the pair that interacts the most. Like the aforementioned main effect test, it also depends on the type of predictor variables. If a pair of predictors is noncategorical, we split its range into two or three intervals depending on the sample size in a node, and then divide the variable’s space into four to nine based on the calculated range. We form a contingency table with the signs as rows and the subset

as columns, and calculate its χ^2 statistic to obtain a 1-d.f. chi-squared value $W_I(X_1, X_2)$, for which X_1 and X_2 are a pair of predictor variables.

When the split variable is selected by the local interaction test, we need to decide which one is more significant. Since the ideal split point (or set) of X_1 and X_2 may strongly depend on each other, we consider the following two-level search approach. The algorithm calculates the total deviance hierarchically to choose the ideal split point (or set) on the split predictor variable. First, node t is divided into t_L and t_R on the basis of X_1 . Each of t_L and t_R is split into t_{LL} , t_{LR} , t_{RL} , and t_{RR} , where t_{LR} means the node generated by t_L going into the left child node to the split point of X_1 . Let $r(t_{LR})$ denote the deviance of node t_{LR} . Then, we seek the split point (or set) of X_1 : $\min \sum_i \sum_j r_{ij}$, where $(i, j) \in \{L, R\}$. The same method is used to gain the split point of X_2 . We finally select the split variable of node t , taking the smaller value of the total deviance in the pair.

2.3.3 Selection of Split Point or Set. After selecting a split variable at a node, we need to find a split set for a categorical variable, or a split point for a noncategorical variable. We can seek the split point or set that minimizes the total deviance using the greedy (G) search. For numeric X , we find a split point c of the form $X \leq c$. As in the variable selection, such an ES approach is computationally expensive and may even cause the end-cut preference problem, resulting in many additional splits, which are probably unnecessary. To avoid these problems, SUPPORT uses the sample mean, and the GUIDE regression tree uses the sample median as the split point. It is much faster and avoids an end-cut point; however, the sample median and the sample mean may be ineffective if the data are skewed or unbalanced. Rather, the weighted mean (WM) of positive and negative residuals as a split point, as described in the Appendix, can be more effective. As an intermediate strategy, Loh (2009) suggested the restricted greedy (RG) search, which is computationally less expensive than G and more expensive than WM. In addition, we suggest using a spline (S) method to find an optimal changing point effectively, as described in the Appendix.

For categorical X , we need to find a set A of the form $X \in A$, where A is a subset of X . If we use the ES, there are $(2^{L-1} - 1)$ candidates, which exponentially grow as L increases, where L is the number of categories. To reduce the computational burden, we consider the problem as one of classification, with two classes of residuals, as in Loh (2009). We compute the proportions of a class given each category, and then sort them in increasing order. The categories are ordered according to the degree proportions of one class of residuals. Then, the shortcut algorithm given in Breiman et al. (1984) can be applied as in the Appendix.

2.3.4 Selection of Split Variable Using Linear Splits. Main and interaction tests may not be significant. In such cases, we use linear splits, which can provide fewer terminal nodes and more powerful predictive accuracy. CART randomly searches for splits on linear combinations of variables. Following CART, oblique decision tree (Murthy, Kasif, and Salzberg 1994) combines deterministic hill-climbing with a randomized procedure to search for linear splits. Brodley and Utgoff (1995) used the concept of a linear machine called Perceptron. The linear discriminant analysis (LDA) approach is used by CRUISE, GUIDE, and the algorithm of Chen and Wang (2007).

Unlike the previous linear split approaches, we use linear splits when main-effect and local interaction tests are not significant. Our linear splits idea is based on the linear split tree

algorithm by Chen (2008). It selects the point satisfying $X^* \leq c_0$, where c_0 is the split point and X^* , defined by $X_1 \sin \theta + X_2 \cos \theta$, is used to find an ideal split. It changes the θ value from 0 to $11\pi/12$ by $\pi/12$ recursively. Along with X^* , we divide the dataset into three or four parts depending on the data size and then calculate the χ^2 value. For the comparison of main and interaction tests at the same level, the Wilson–Hilferty transformation is again approximated with a χ_1^2 value. Finally, we identify the linear combination of variables and X^* with the most significant value. The detailed algorithm of variable selection is described in the Appendix.

In addition, a Bonferroni-corrected significance threshold is used to select a split variable in which three selection algorithms are ordered. Due to the characteristic that catching a split variable in the interaction tests is more likely to be detected, we select a split variable in the interaction tests only if none of the main effects is significant. Similarly, linear tests are used when neither main or interaction tests is significant. Let $\chi_{\nu, \alpha}^2$ denote the upper- α quantile of the χ^2 distribution with ν degree of freedom, K be the number of nonconstant variables in a node, and $\alpha = 0.05/K$ and $\beta = 0.05/K(K - 1)$ are defined. We select the variable in the main effect tests for a split at the node when the maximum value of $W_M(X_i)$ is greater than $\chi_{1, \alpha}^2$. If no main effect tests are significant and the maximum value of $W_I(X_i, X_j)$ is greater than $\chi_{1, \beta}^2$ at the same time, we choose the pair with the largest value of it as a split. Otherwise, numeric X_i is selected by using linear splits and categorical X_i is selected by the largest value of $W_M(X_i)$. With the Bonferroni-corrected approach, we can reduce the huge computational cost generated in the linear splits step. Also, we can prevent the possibility of selecting a weak main effect as the result of the interaction tests by a two-level search algorithm.

2.4 DETERMINING TREE SIZE

How many splits should be used? The choice of the optimal number of splits, equivalently the determination of an optimal tree size, plays a crucial role in obtaining good prediction accuracy. Too few terminal nodes may lead to the generation of low prediction accuracy, while too many makes interpretation difficult and may cause overfitting.

To determine the optimal tree size, AID, chi-squared automatic interaction detection (CHAID), and FACT use a direct stopping rule to stop growing if either the error rate does not decrease in the node, or the sample size of the node is less than the user-specified minimum node size. However, using a direct stopping rule may give rise to an overfitting or underfitting problem. More recently, Hothorn, Hornik, and Zeileis (2006) used the permutation test stopping criteria based on a multiple test procedure. Breiman et al. (1984) developed the cost-complexity pruning algorithm, which was first implemented in CART. The algorithm consists of two steps: first, an overly large tree is constructed, and a series of candidate trees are obtained. Then, they are evaluated by independent test data or cross-validation (CV) to select the final tree. However, independent test data are often unavailable and CV requires substantial computational cost.

In addition to the cost-complexity pruning, we consider an M -step stopping rule to find an optimal tree size that reduces computing time. First, we stop splitting when a sample size at each node is too small. Second, we stop when there is no significant improvement at M consecutive later subnodes, for example, $M = 2, 3$, or ∞ . If $M = \infty$, we investigate all

subnodes until a sample size is too small and then prune the branches with no significant subnodes. This is comparable to the accuracy of the cost-complexity pruning and less computationally expensive. The M -step stopping algorithm is summarized in the Appendix and implemented into an R package, `melt`, as well as the cost-complexity pruning with cross-validation or independent test data.

2.5 TREATING TIME-VARYING VARIABLES

Predictor variables can often vary according to time in longitudinal data and they can play pivotal roles. Thus, it is important to use time-varying variables for longitudinal data analysis, but it is troublesome to incorporate them into tree construction. Segal (1992) replaced time-varying variables with low-order polynomial approximations and Sela and Simonoff (2012) proposed to use all periods of time-varying variables to predict every observation.

From the MELT's point of view, Segal's approach can be applicable to our algorithm. Each time-varying variable is regressed against time, and then its intercept and slope are treated as possible split variables. It results in the increase of possible split variables. If time-varying variables are categorical, they are transformed into dummy variables and logistic regression is applied rather than linear regression. If linearity is not enough, a low-order polynomial can be used. This approach is also implemented into our software program.

3. SIMULATION STUDY

We investigate the performance of the proposed method through simulation. It is essential to select split variables that are associated with the response. Thus, we compare the variable selection performance of residual analysis (RA) and exhaustive search (ES) approaches as univariate splits. We also evaluate the validity of linear splits and split point selections. In addition, the prediction power for splitting and size-determining rules discussed is investigated under several situations.

3.1 SIMULATION SETTING

In all the experiments, it is assumed that the response vector \mathbf{Y} is a repeated measured variable with five times and the error term is defined as $\varepsilon_i \sim N(0, \Lambda)$, where Λ has two types of variance-covariance structures: compound symmetry (CS) and autocorrelation AR(1) with $\rho = 0, 0.5$, and 0.9 . Three categorical variables (X_1 , X_2 , and X_3) and three numerical-valued ones (X_4 , X_5 , and X_6) are defined to be predictor variables.

As done by Loh (2002), we first generate random samples for the six predictor variables with the marginal distributions $X_1 \sim Ce_2$, $X_2 \sim Cd_3$, $X_3 \sim Ce_8$, $X_4 \sim K$, $X_5 \sim Z$, and $X_6 \sim U$ for the independence situation, where $X_1 \sim Ce_2$ and $X_2 \sim Cd_3$ with joint probability of X_3 ; $X_3 \sim Ce_8$ with joint probability of X_2 ; and $X_4 \sim KZ$, $X_5 \sim KZ$, and $X_6 \sim U$ for the dependence situation. The random variables Ce_2 , Cd_3 , Ce_8 , K , Z , and U are mutually independent, where Ce_L represents an L -category variable taking values $\{1, 2, \dots, L\}$ with equal probabilities and Cd_3 represents a 3-category variable with unequal probabilities, $\{1/6, 1/3, 1/2\}$. K is a chi-squared variable and Z is a standard normal variable. U is a

uniform variable over 0 and 1, and KZ is a bivariate normal variable with correlation 0.7. The joint distribution of categorical variables X_2 and X_3 is given in the online supplementary material.

The estimated probabilities of selecting each predictor are recorded for 1000 iterations with 200 subjects. The average CPU times are also recorded on a 3 GHz Xeon^(TM) workstation with 8 GB ECC-DDR2 memory. For the simulation study, we use R 2.15.2 (R Core Team 2012) with two well-known packages, *nlme* (Pinheiro and Bates 2000) and *mvtnorm* (Hothorn, Bretz, and Genz 2001), which are used to fit the basic model and to generate the simulated data, respectively. The R package *REEMtree* (Sela and Simonoff 2011) and FORTRAN-compiled binary *GUIDE* version 14.2 (<http://www.stat.wisc.edu/loh/guide>) are also used for the comparison.

3.2 SPLIT VARIABLE SELECTION

We first generate the simulated data under the following null model:

$$\text{Null: } y_i = \alpha_i + 0.3(\text{time})_i + \epsilon_i,$$

where $\alpha_i \sim N(1, 1)$ and $(\text{time})_i = (1, \dots, 5)^T$. The response depends on the time variable, but it is independent of all the other predictor variables. Therefore, the chance that each predictor is selected as the split variable should be the same, that is, the ideal selection probability is 1/6 for each. As explained in Section 3.1, we consider the following situations: (i) independence, such that the X 's are mutually independent and (ii) dependence, such that some of the X_i 's are not mutually independent. X_4 and X_5 have a bivariate normal distribution with correlation 0.7 and X_2 and X_3 have a joint distribution. In addition, we consider two types of data structures: regular and irregular. The irregular structure was generated by randomly selecting subjects' follow-ups from discrete uniform [2, 5].

In these scenarios, we investigate the performance of variable selection by the following approaches: RA and ES approaches by MELT (denoted by RA(MELT) and ES(MELT), respectively), RA approach by multivariate GUIDE of Loh and Zheng (2013) (denoted by RA(GUIDE)), ES approaches by MVPART of De'Ath (2002) and RE-EM of Sela and Simonoff (2012) (denoted by ES(MVPART) and ES(RE-EM), respectively).

Table 1 contains the estimated probabilities that each X_i is selected under the independence situation with both regular and irregular data structures using AR(1). Due to the limitation of space, the results are not shown for the other situation and covariance structure. Instead, they can be found in the supplements (available online). Because none of the predictor variables are associated with the response variable under the null model, the chance of selection as a split variable should be equal for all the six variables, i.e., the ideal selection probability of each variable is 1/6.

Under the independence situation with regular structure, the RA approach selects split variables regardless of the distribution of predictors because all the variables are equally likely to be selected. In contrast, the ES approaches tend to pick X_3 over the other variables. In particular, the numerical variables tend to be selected less than categorical variables, even though the numerical and categorical variables have a similar number of possible split points. Thus, selection bias can arise from the type of variables as well as the number of possible split points. This implies that the ES approaches are more vulnerable to bias

Table 1. Estimated selection probabilities with the null model

		Regular data structure					Irregular data structure				
ρ	X_i	RA	RA	ES	ES	ES	RA	RA	ES	ES	ES
		(MELT)	(GUIDE)	(MELT)	(MVPART)	(RE-EM)	(MELT)	(GUIDE)	(MELT)	(MVPART)	(RE-EM)
0.0	X_1	0.140	0.179	0.008	0.007	0.011	0.123	0.179	0.014	0.019	0.014
	X_2	0.157	0.140	0.027	0.024	0.027	0.174	0.140	0.025	0.023	0.029
	X_3	0.173	0.169	0.322	0.354	0.369	0.156	0.169	0.442	0.565	0.386
	X_4	0.181	0.164	0.231	0.220	0.205	0.181	0.164	0.178	0.113	0.221
	X_5	0.173	0.154	0.210	0.203	0.196	0.180	0.154	0.176	0.144	0.171
	X_6	0.176	0.194	0.202	0.192	0.192	0.186	0.194	0.165	0.136	0.179
0.5	X_1	0.141	0.185	0.008	0.005	0.014	0.144	0.190	0.007	0.012	0.016
	X_2	0.165	0.156	0.026	0.028	0.023	0.190	0.169	0.020	0.041	0.027
	X_3	0.174	0.145	0.352	0.353	0.365	0.167	0.162	0.361	0.495	0.391
	X_4	0.152	0.185	0.221	0.211	0.196	0.172	0.178	0.211	0.151	0.206
	X_5	0.186	0.173	0.198	0.202	0.207	0.140	0.151	0.197	0.154	0.179
	X_6	0.182	0.156	0.195	0.201	0.195	0.187	0.150	0.204	0.147	0.181
0.9	X_1	0.112	0.181	0.008	0.008	0.009	0.126	0.139	0.009	0.019	0.009
	X_2	0.188	0.171	0.030	0.032	0.025	0.164	0.156	0.020	0.037	0.023
	X_3	0.171	0.162	0.382	0.363	0.386	0.165	0.200	0.369	0.510	0.373
	X_4	0.169	0.146	0.200	0.212	0.186	0.179	0.153	0.204	0.126	0.175
	X_5	0.173	0.176	0.190	0.198	0.198	0.190	0.171	0.201	0.140	0.202
	X_6	0.187	0.164	0.190	0.187	0.196	0.176	0.181	0.193	0.168	0.218

toward categorical variables. There is no significant difference between CS and AR(1). The changes of correlations ρ have no signal effect on the results.

The results under the dependence situation are almost similar to those under the independence situation (the results in the supplements, available online). One difference is that the ES approaches have a tendency to select X_3 more often because of the joint distribution. In addition, similar results are obtained with the irregular data structure. There is no significant difference between RA(MELT) and RA(GUIDE) as well as among ES(MELT), ES(MVPART), and ES(RE-EM). This implies that selection probabilities are dependent on the choice of RA and ES. Thus, we can conclude that in contrast to the ES approaches, the RA approach generates negligible selection bias in selecting split variables under the null model.

In addition, we investigate the power of variable selection when the response is associated with some of the predictor variables. We consider various models such as jump, quadratic, cubic, cross, and step models. The investigation indicates that selection probabilities are highly dependent on the use of a mixed-effect model as well as the choice of RA and ES. It is found that RA(MELT) performs well. The detailed results can be found in the supplements (available online).

3.3 SPLIT POINT SELECTION

We also conduct a simulation experiment to investigate the performance of the split point selection approaches: greedy search (G), restricted greedy search (RG), weighted mean (WM), and spline (S). We assume the same settings in Sections 3.1 and 3.2 and generate the simulated data from each of the five alternative models. This procedure was repeated 1000 times independently and the distributions of split points that were selected are displayed in Figure 2. For the jump, cross, and step models, the ideal split point was

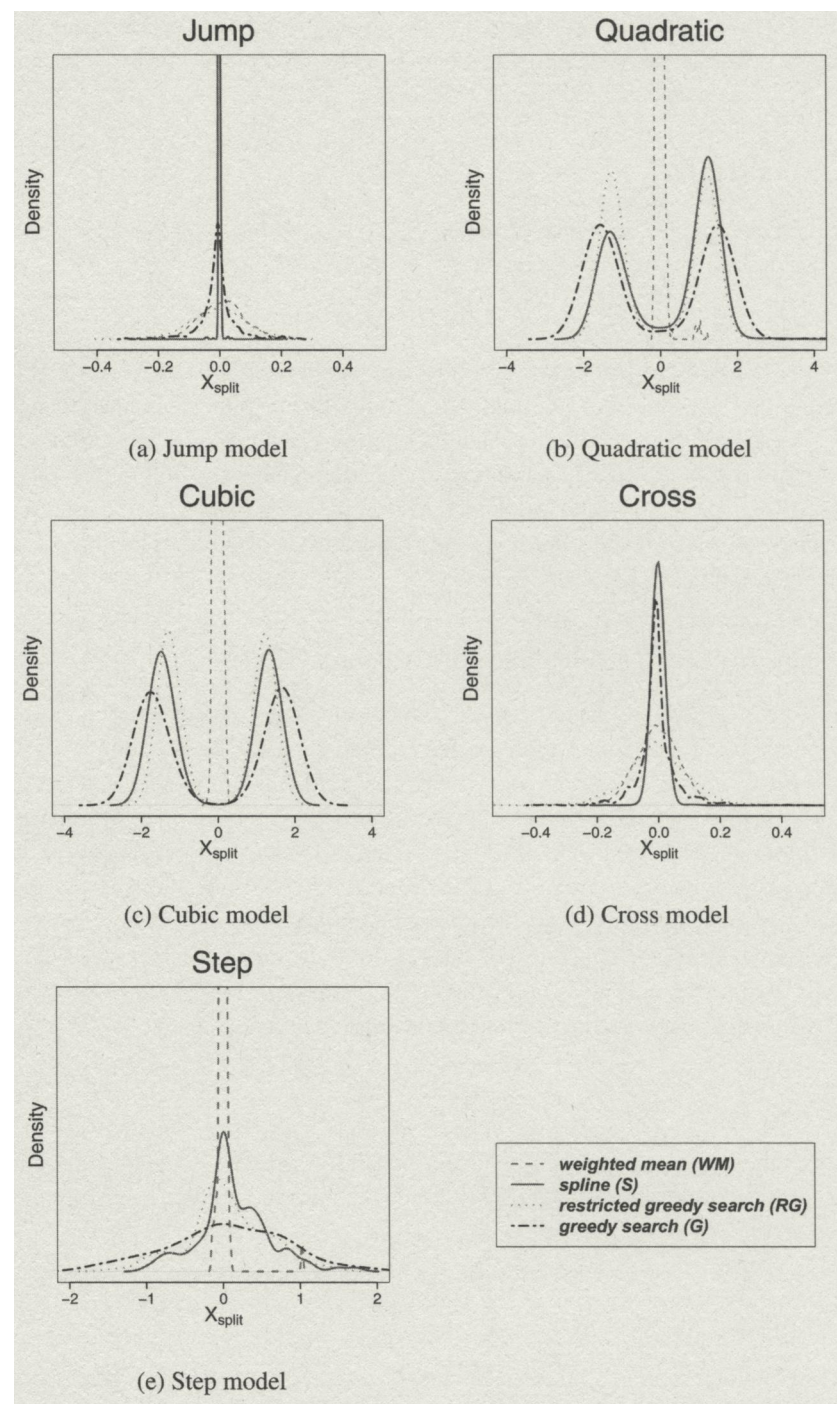


Figure 2. The distributions of split point selections under the alternative models by various approaches. The double-dashed black and dotted green lines indicate greedy search (G) and restricted greedy search (RG) and the solid blue and dashed red lines the spline method (S) and weighted mean method (WM), respectively.

well selected by S and WM particularly. For the quadratic and cubic models, one point was selected by WM, but two points by the others because the models change substantially at 0 or $\pm 1 \sim \pm 2$. In particular, WM tends to select the center.

3.4 LINEAR SPLITS

To investigate the performance of linear splits, we additionally consider the following model:

Linear: $y_i = \alpha_i + 0.1I(X_5 > X_6)(\text{time})_i + \epsilon_i$.

This setting is borrowed from Loh (2009) to emphasize the necessity of linear splits and more importantly to show that splitting with partially linear splits is practically unbiased.

Table 2 displays the estimated probabilities of predictor variables chosen over 10,000 simulation trials with RA(ME) as an impurity function. Under the null model, there is no preference to any predictor variable. Under the linear model, X_5 and X_6 make splits together, so the chance of selecting their linear split as well as their univariate splits increases in the independence and dependence situations.

3.5 PREDICTION POWER FOR SLOPES AND COMPUTATIONAL COST

In this section, we investigate the prediction power for slopes and computing time with various combinations of split variable selection and tree-size determination approaches based on MELT. Here, other tree algorithms and programs are not considered because they aim to predict responses rather than to discover trends over time. We generate the simulated data under the two models: three terminal nodes defined in Figure 3(a) and four terminal nodes defined in Figure 3(b). The difference between Models (a) and (b) is the complexity of tree structures. Model (b) has an additional split than Model (a), resulting in four terminal nodes. We generate 200 subjects for train and test sets, respectively, under each of the two models. To investigate the prediction power, we calculate the sum of the squared differences between the true and predicted slopes at terminal nodes \tilde{T} :

$$\sum_{t \in \tilde{T}} \sum_{i \in t} (\beta_i - \hat{\beta}_t)^2,$$

where β_i is the true slope for the i th subject of the test set and $\hat{\beta}_t$ denotes the predicted slope at terminal node t from the train set. Table 3 shows the prediction error, number of

Table 2. Estimated probabilities to select linear splits. X_1 , X_2 , and X_3 usually appear in univariate splits due to the characteristic of linear splits

Model	Situation	Univariate splits						Linear splits		
		X_1	X_2	X_3	X_4	X_5	X_6	X_4X_5	X_4X_6	X_5X_6
Null	<i>Indep.</i>	0.1163	0.1566	0.1552	0.1413	0.1400	0.1506	0.0478	0.0448	0.0443
	<i>Dep.</i>	0.1237	0.1631	0.1697	0.1356	0.1431	0.1440	0.0296	0.0336	0.0354
Linear	<i>Indep.</i>	0.0624	0.0867	0.0812	0.0701	0.2375	0.1150	0.0818	0.0445	0.2051
	<i>Dep.</i>	0.0651	0.0796	0.0805	0.0997	0.2046	0.1158	0.0762	0.0706	0.1952

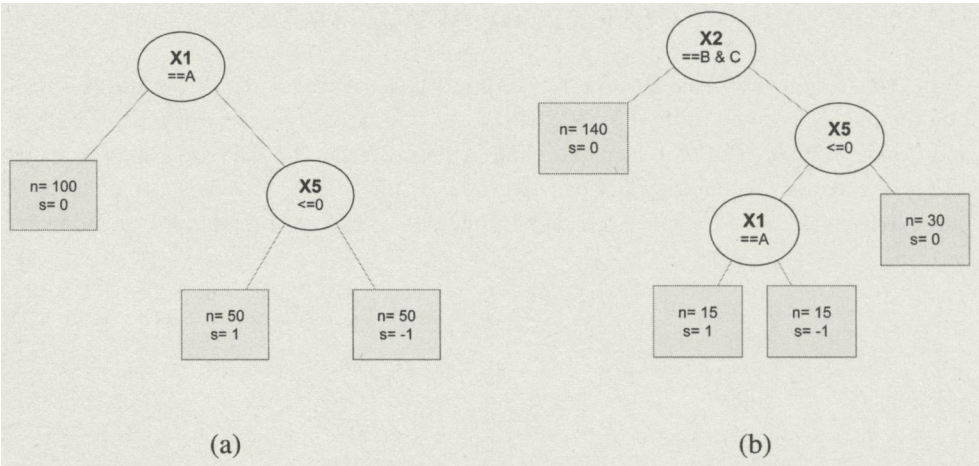


Figure 3. Tree models for testing the performance power. The split rules are given at each intermediate node. The sample size and true slope are given by n and s ($= \beta$), respectively, at each terminal node. An observation goes to the left node if the condition is satisfied, otherwise to the right.

terminal nodes, and computing time under four combinations of two split variable selections (RA = residual analysis and ES = exhaustive search) and two tree-size determination algorithms (M = M-step stopping rule, and CC = cost-complexity pruning with 10-fold cross-validation and 1-SE).

When the algorithms for split variable selection are compared, there are significant differences in prediction power between RA and ES. Table 3 shows that ES generates consistently larger errors than RA with any tree size determination approach because ES suffers from selecting incorrect splits, more seriously in Model (b). Model (b) has another split, which may be difficult to be found. Assuming RA, the algorithms for tree size determination generate the similar prediction errors under both models. The differences between M and CC are within the standard errors. Overall, RA has better prediction power than ES. Furthermore, RA requires much less computing time than ES, as expected. The accuracy of M is comparable to that of CC and M is much less computationally expensive.

Table 3. Prediction error of slopes, number of terminal nodes, and computing time (standard errors in parentheses)

Model	Split	Prediction error		# of nodes		Time (s)	
		M	CC	M	CC	M	CC
(a)	RA	23.23 (1.93)	25.97 (1.90)	5.93 (0.18)	5.11 (0.22)	3.48 (0.09)	32.76 (2.79)
	ES	57.28 (0.78)	59.21 (0.82)	5.05 (0.16)	3.43 (0.18)	290 (2.97)	1900 (13.03)
(b)	RA	45.71 (2.78)	47.30 (2.96)	5.68 (0.10)	4.88 (0.21)	5.18 (0.12)	45.30 (1.66)
	ES	92.53 (1.34)	90.07 (1.58)	4.87 (0.19)	3.72 (0.34)	481 (8.37)	3026 (35.1)

4. CASE STUDY: WAGES DATA

In this section, we consider a real example called the wages data from the National Longitudinal Survey of Youth (NLSY), which was originally studied by Murnane, Willett, and Boudett (1999), and then by Singer and Willett (2003). The data are available at the UCLA Academic Technology Services website (Bruin 2011). The data consist of 888 individuals, aged 14–17 years. The response variable, *wage*, is the hourly wage adjusted

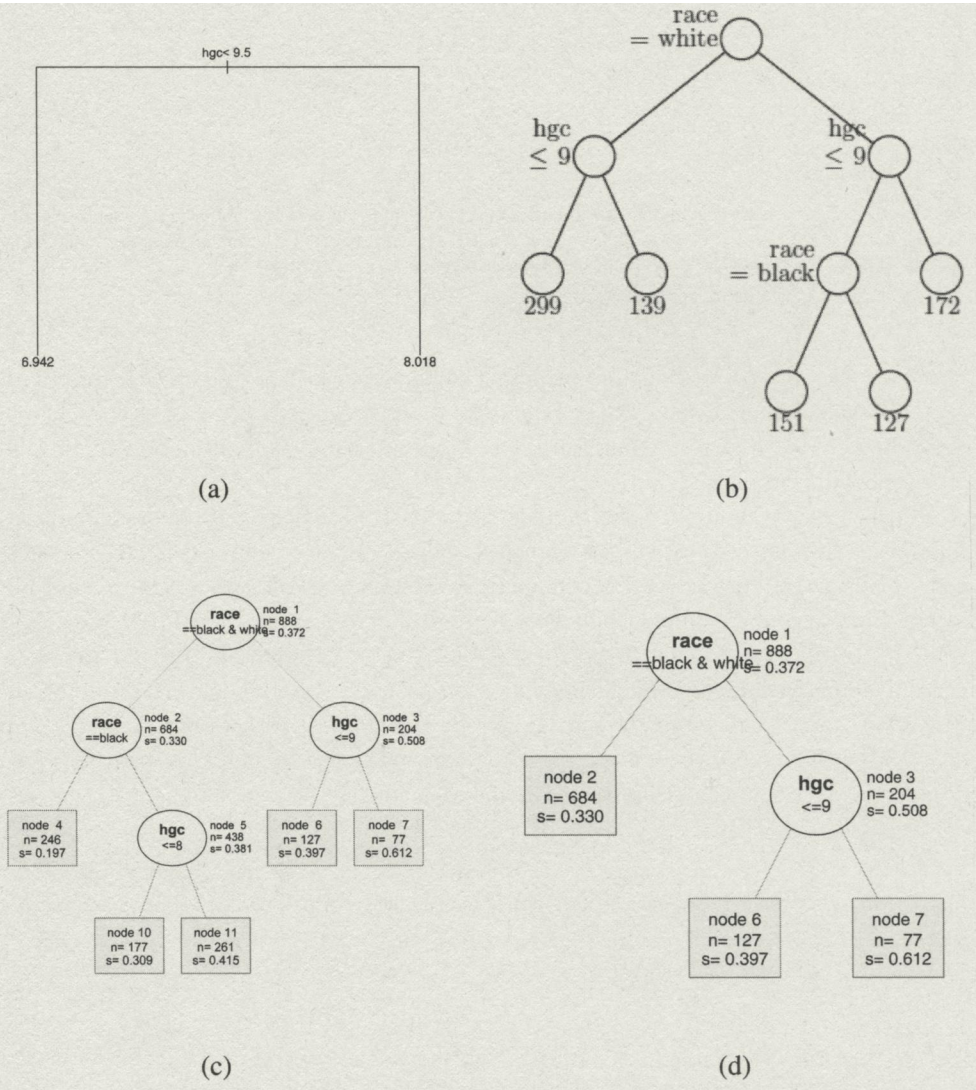


Figure 4. Regression trees for the wage data. (a) RE-EM tree. The mean estimate of the response is given at each terminal node. (b) Multivariate GUIDE tree (Loh and Zheng 2013). The sample size is given at each terminal node. (c) MELT trees with the M-step stopping rule. (d) MELT tree with the cost-complexity pruning with cross-validation and 1-SE rule, respectively. The sample size and slope at each terminal node are given by n and s , respectively. An observation goes to the left node if the condition is satisfied at the intermediate nodes; otherwise, it goes to the right.

for inflation in constant 1990 dollars and the time variable, *exper*, is the duration of the work experience (in years). There are two predictor variables, *race* and *hgc*. The former is the individual's race, namely, White, Black, and Hispanic; the latter is the highest grade completed by the individual. The numbers of subjects are 38, 39, 82, 166, 226, 240, and 97 at time points 1, 2, 3–4, 5–6, 9–10, and >10, respectively. This indicates that the data are highly unbalanced. The aim of this study is to determine what predictors, especially trends over time, have significant effects on the wage.

As the dataset is highly unbalanced, the number of subjects at the time points varies. Therefore, Segal's algorithm is not applicable. Figure 4 displays RE-EM, multivariate GUIDE, and MELT trees. For all experiments, the default options were used, and the minimum sample size was 40. The RE-EM tree in Figure 4(a) was obtained from the cost-complexity pruning with 1-SE rule (the default) in which the *hgc* is the primary factor influencing the increase of wages. The wage was the lowest when the highest grade completed (*hgc*) is less than 9.5. The multivariate GUIDE tree in Figure 4(b) was borrowed from Loh and Zheng (2013). Among split variables, *race* was the first followed by the *hgc*.

The MELT trees in Figure 4(c) and (d) were obtained with M-step stopping rule and the cost-complexity pruning with 1-SE rule. Overall, *race* is the primary factor influencing the increase in wages, followed by the *hgc*. The difference between the two trees occurs at node 2. The M-step stopping generates five terminal nodes in Figure 4(c), while the cost-complexity pruning cuts off the branches after node 2 in Figure 4(d). By closely examining the terminal nodes in Figure 4(c), we can find that regardless of the highest grade completed, the Black people have a slow increase in wage, and the White and Hispanic people have faster wage increases, depending on the highest grade completed. On the other hand, only Hispanic people have different wage increase rates as seen in Figure 4(d).

5. CONCLUSIONS

In this article, we proposed a tree algorithm by combining the merits of a tree-based model and a mixed-effects model for longitudinal data. We alleviated variable selection bias for longitudinal cases by RA as GUIDE does for univariate cases. The RA approach is used to solve problems that ES approaches suffer from, such as undue preference to split variables with more possible splits, expensive computational cost, and end-cut preference. Although longitudinal data are often unbalanced, the algorithm of Segal (1992) is only applicable to a regular structure. Our proposed algorithm does not suffer from such a limitation.

Most importantly, the major advantage of our tree algorithm is to discover trends over time from time-independent or time-dependent covariates, unlike the other tree algorithms such as in Segal (1992), Sela and Simonoff (2012), and Loh and Zheng (2013), which provide the predictions of responses. Therefore, our development is useful in finding common slopes on each of the subspaces from recursively partitioning, rather than predicting responses. We have implemented our algorithms into an R package *melt*, which can be used conveniently and freely in the R environment.

MELT has several limitations: (1) We focus on discovering trends over time. Response predictions are the main interest of most tree algorithms. By MELT, each subgroup is

homogeneous in trends, but may be heterogeneous in responses. Therefore, MELT is not suitable to predict responses. (2) We have suggested polynomial model fitting at each node for flexible extension; however, MELT is not invariant of time changes with polynomials although time centering is often taken in longitudinal data analysis. (3) Time-varying covariates can have significant effects on responses in longitudinal data analysis. For incorporating them into tree modeling, we have simply adopted the approach of Segal (1992); however, a better approach may yield a better result. We regard these as a scope for future research.

APPENDIX

Algorithm 1: Selection of split variable by RA approach

1. Obtain the residuals using model (1).
2. Obtain p -values by using a main-effect test. The main effect test is defined as the Pearson χ^2 test for a $2 \times c$ contingency table. Let t be a node and X a categorical predictor variable,
 - (a) Form a contingency table constructed by the signs of the residuals as rows and the categories of X as columns in which if the cell count is zero, they are deleted.
 - (b) Let ν be the degree of freedom of the table. Compute the chi-squared statistic χ_ν^2 for testing independence.
 - (c) If $\nu \geq 2$, use the Wilson–Hilferty approximation twice to convert χ_ν^2 to the 1-d.f. chi-squared

$$W_M(X) = \max \left(0, \left[\frac{7}{9} + \sqrt{\nu} \left\{ \left(\frac{\chi_\nu^2}{\nu} \right)^{1/3} - 1 + \frac{2}{9\nu} \right\} \right]^3 \right).$$

3. Obtain p -values for each noncategorical predictor variable X :
 - (a) Compute the sample mean \bar{x} and standard deviation s in node t .
 - (b) If $N(t) \geq 20$, divide the range of X into four intervals with boundary values $\bar{x} \pm s\sqrt{3}/2$. Otherwise, if $N(t) < 20$, divide the range of X into three intervals with boundary values $\bar{x} \pm s\sqrt{3}/3$, where $N(t)$ is the number of unique values in node t .
 - (c) Form a contingency table with the signs of the residuals as rows and the intervals as columns in which if the cell count is zero, they are deleted.
 - (d) Follow Step 2(c) to obtain $W_M(X)$.
4. Do the interaction test for each pair of predictor variables to obtain p -values, if $W_M(X)$ is not significant:
 - (a) If a pair of predictor variables, X_i and X_j , is noncategorical and $N(t) \geq 45$, split its range into three intervals (A_{i1} , A_{i2} , A_{i3}) at the points $\bar{x} \pm s\sqrt{3}/3$. If a pair is noncategorical and $N(t) < 45$, split its range into two intervals (A_{i1} , A_{i2}) at the sample median. Otherwise, if at least a variable is categorical, set A_{ik} denoted by its k th value.
 - (b) $B_{k,m}$ is constructed by A_{ik} , where $B_{k,m} = \{(x_i, x_j) : x_1 \in A_{1k}, x_2 \in A_{2m}\}$, for $k, m = 1, 2, \dots$

- (c) Form a contingency table with the signs of residuals as rows and $\{B_{k,m}\}$ as columns. Compute its chi-squared statistic and use the W-H approximation to transform it to a 1-d.f. chi-squared value $W_I(X_i, X_j)$.
5. Find the largest values of W_M from all tests using a Bonferroni-corrected significance threshold. Let K be the number of nonconstant predictor variables in the node. Define $\alpha = 0.05/K$ and $\beta = 0.05/\{K(K-1)\}$.
 - (a) If $\max_i W_M(X_i) > \chi_{1,\alpha}^2$, select the variable with the largest value of $W_M(X_i)$ and exit.
 - (b) If $\max_{i \neq j} W_I(X_i, X_j) > \chi_{1,\beta}^2$, select the pair with the largest value of $W_I(X_i, X_j)$ and exit. Otherwise, if a pair of noncategorical predictor variables exists, do the linear splits test. If it does not exist, select the X_i with the largest value of $W_M(X_i)$.
6. Do the linear splits test for each pair of noncategorical pair of predictor variables, X_i and X_j , to obtain p -values, if both $W_M(X)$ and $W_I(X_1, X_2)$ are not significant.
 - (a) Select the point satisfying $X_{ij\theta}^* \leq c_0$, where c_0 is the split point and $X_{ij\theta}^*$ is defined by $X_i \sin \theta + X_j \cos \theta$ to find an ideal split.
 - (b) Change the θ value from 0 to $11\pi/12$ by $\pi/12$ recursively.
 - (c) If $N^*(t) \geq 20$, divide the range of $X_{ij\theta}^*$ into four intervals with boundary values $\bar{x}_{ij\theta}^* \pm s_{ij\theta}^* \sqrt{3}/2$. Otherwise, divide the range of $X_{ij\theta}^*$ into three intervals with boundary values $\bar{x}_{ij\theta}^* \pm s_{ij\theta}^* \sqrt{3}/3$, where $N^*(t)$ is the number of unique $X_{ij\theta}^*$'s value in node t .
 - (d) Form a contingency table with the signs of the residuals as rows and the intervals as columns in which if the cell count is zero, they are deleted.
 - (e) Follow Step 2(c) to obtain $W_L(X_{ij\theta}^*)$ approximated with a χ_1^2 value.
 - (f) Select the variable with the largest $W_L(X_{ij\theta}^*)$.

Algorithm 2: Weighted Mean (WM) split point selection for noncategorical X

1. Calculate \bar{x}_+ and \bar{x}_- where $\bar{x}_+ = \sum_{i \in s_+} x_{+i} / N_+$ and $\bar{x}_- = \sum_{i \in s_-} x_{+i} / N_-$ for positive and negative index sets s_+, s_- and sample N_+ and N_- of positive and negative residuals.
2. Select $(\bar{x}_+ + \bar{x}_-) / 2$ as the split point.

Algorithm 3: Spline (S) split point selection for noncategorical X

1. Calculate fitted values using the spline model,

$$s(x) = \beta_0 + \beta_{-1}x + \sum_{j=1}^K \beta_j(x - t_j)_+.$$

2. Add knots achieving maximum reduction in residual sum of squares using a forward stepwise procedure.
3. Treat the knots on candidates for the split points.
4. Select the best split point of the candidate knots.

Algorithm 4: Split set selection for categorical X

1. Label each observation in the node as belonging to class 1 if the sign of residual is positive and class 2 otherwise.
2. Suppose there are L categories, b_1, \dots, b_L , in the selected X . Order the categories according to the class 1 proportions. That is, $\Pr(\text{class } 1 | X = b_{l_1}) \leq \dots \leq \Pr(\text{class } 1 | X = b_{l_L})$.
3. The best split set belongs to one of the $L - 1$ subsets $\{b_{l_1}, \dots, b_{l_k}\}, k = 1, \dots, L - 1$. Select the subset that minimizes the sum of the binomial variances in the left and right subnodes. The sum of binomial variances is $n_L p_L(1 - p_L) + n_R p_R(1 - p_R)$, where $p_L = \Pr(\text{class } 1 | X \in \{b_{l_1}, \dots, b_{l_k}\})$, $p_R = \Pr(\text{class } 1 | X \in \{b_{l_{k+1}}, \dots, b_{l_L}\})$, and the sample sizes of the left and right nodes are n_L and n_R , respectively.

Algorithm 5: Determine a tree size using M -step Stopping Rule (MSR), for example, $M = 2$.

1. Let t be any node and t_L and t_R be its left and right child node. Let $R(t)$ be an impurity function of node t and $\Delta(t) = R(t) - [R(t_L) + R(t_R)]$. We split node t if $\Delta(t) > \epsilon$ and go to Step 2 if $\Delta(t) \leq \epsilon$.
2. If $\Delta(t_L) > \epsilon$ (or $\Delta(t_R) > \epsilon$), then go to Step 1, letting t_L (or t_R) as node t .
3. Declare node t as a terminal node if $\Delta(t_L) \leq \epsilon$ and $\Delta(t_R) \leq \epsilon$. For $M > 2$, declare node t as a terminal node when there is no improvement at all M -step nodes.

SUPPLEMENTARY MATERIALS

Additional results: Additional simulation and case studies for tree-structured mixed-effects regression modeling for longitudinal data (Supplements.pdf)

R-package: R package `melt` to perform the tree-structured mixed-effects regression models described in the article. The package includes the wages data. (`melt_0.7-14.tar.gz`; GNU zipped tar file)

R code: R script for the simulation and case studies, and also creating figures described in the article. It is possible to download the files via our website at <http://statlab.korea.ac.kr/melt>. (`code_melt.zip`; zip file containing R script and example files)

ACKNOWLEDGMENTS

The authors would like to thank the editor, the associate editor, and two anonymous referees, whose insightful comments and constructive suggestions have greatly improved this article. In addition, the authors would like to thank Professor Wei-Yin Loh at the Department of Statistics, University of Wisconsin-Madison for his helpful comments on this article. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (2010-0007936).

[Received January 2012. Revised March 2013.]

REFERENCES

- Abdolell, M., LeBlanc, M., Stephens, D., and Harrison, R. (2002), "Binary Partitioning for Continuous Longitudinal Data: Categorizing a Prognostic Variable," *Statistics in Medicine*, 21, 3395–3409. [741]
- Ahn, H., and Loh, W. (1994), "Tree-Structured Proportional Hazards Regression Modeling," *Biometrics*, 50, 471–485. [741]
- Alexander, W., and Grimshaw, S. (1996), "Treed Regression," *Journal of Computational and Graphical Statistics*, 5, 156–175. [741]
- Breiman, L., Friedman, J., Stone, C., and Olshen, R. (1984), *Classification and Regression Trees*, Belmont, CA: Chapman & Hall/CRC. [740,746,747]
- Brodley, C., and Utgoff, P. (1995), "Multivariate Decision Trees," *Machine Learning*, 19, 45–77. [746]
- Bruin, J. (2011), "Newtest: Command to Compute New Test @ONLINE". [754]
- Chan, K., and Loh, W. (2004), "LOTUS: An Algorithm for Building Accurate and Comprehensible Logistic Regression Trees," *Journal of Computational and Graphical Statistics*, 13, 826–852. [741,745]
- Chaudhuri, P., Huang, M., Loh, W., and Yao, R. (1994), "Piecewise-Polynomial Regression Trees," *Statistica Sinica*, 4, 143–167. [741,745]
- Chen, C. (2008), "Enhancing the Prediction Accuracy of Regression Trees: Linear Splits and Variable Selection," Ph.D. thesis, University of Wisconsin-Madison. [747]
- Chen, L., and Wang, D. (2007), "Multivariate Decision Trees Based on Regression and Discriminant Analysis," in *IEEE International Conference on Convergence Information Technology*, pp. 1733–1741. [746]
- Chipman, H., George, E., and McCulloch, R. (1998), "Bayesian CART Model Search," *Journal of the American Statistical Association*, 93, 935–948. [741]
- Cho, H., and Hong, S. (2008), "Median Regression Tree for Analysis of Censored Survival Data," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38, 715–726. [745]
- De'Ath, G. (2002), "Multivariate Regression Trees: A New Technique for Modeling Species–Environment Relationships," *Ecology*, 83, 1105–1117. [741,749]
- Diggle, P., Heagerty, P., Liang, K., and Zeger, S. (2002), *Analysis of Longitudinal Data* (Vol. 25), Oxford, UK: Oxford University Press. [744]
- Doyle, P. (1973), "The Use of Automatic Interaction Detector and Similar Search Procedures," *Operational Research Quarterly*, 465–467. [744]
- Gillo, M., and Shelly, M. (1974), "Predictive Modeling of Multivariable and Multivariate Data," *Journal of the American Statistical Association*, 69, 646–653. [741]
- Hajjem, A., Bellavance, F., and Larocque, D. (2011), "Mixed Effects Regression Trees for Clustered Data," *Statistics & Probability Letters*, 81, 451–459. [741]
- Hothorn, T., Bretz, F., and Genz, A. (2001), "On Multivariate t and Gauß Probabilities in R," *R News*, 1, 27–29. [749]
- Hothorn, T., Hornik, K., and Zeileis, A. (2006), "Unbiased Recursive Partitioning: A Conditional Inference Framework," *Journal of Computational and Graphical Statistics*, 15, 651–674. [741,747]
- Hothorn, T., and Lausen, B. (2003), "On the Exact Distribution of Maximally Selected Rank Statistics," *Computational Statistics & Data Analysis*, 43, 121–137. [741,744]
- Hsiao, W.-C., and Shih, Y.-S. (2007), "Splitting Variable Selection for Multivariate Regression Trees," *Statistics and Probability Letters*, 77, 265–271. [741]
- Kademan, E., Loh, W., and Vanichsetakul, N. (1989), "An Improved Version of FACT With S Front-End," *The American Statistician*, 43, 273. [745]
- Kim, H., and Loh, W. (2001), "Classification Trees With Unbiased Multiway Splits," *Journal of the American Statistical Association*, 96, 589–604. [741,745]

- Lee, S., Kang, H., Han, S., and Kim, K. (2005), "Using Generalized Estimating Equation to Learn Decision Tree With Multivariate Responses," *Data Mining and Knowledge Discovery*, 11, 273–293. [741,743]
- Loh, W. (2002), "Regression Trees With Unbiased Variable Selection and Interaction Detection," *Statistica Sinica*, 12, 361–386. [740,744,745,748]
- (2009), "Improving the Precision of Classification Trees," *The Annals of Applied Statistics*, 3, 1710–1737. [740,741,744,745,752]
- Loh, W., and Shih, Y. (1997), "Split Selection Methods for Classification Trees," *Statistica Sinica*, 7, 815–840. [744,745]
- Loh, W., and Vanichsetakul, N. (1988), "Tree-Structured Classification via Generalized Discriminant Analysis" (with discussion), *Journal of the American Statistical Association*, 83, 715–728. [740,745]
- Loh, W., and Zheng, W. (2013), "Regression Trees for Longitudinal and Multiresponse Data," *Annals of Applied Statistics*, 7, 495–522. [741,749,755]
- Morgan, J., and Sonquist, J. (1963), "Problems in the Analysis of Survey Data and a Proposal," *Journal of the American Statistical Association*, 58, 415–434. [740]
- Murnane, R., Willett, J., and Boudett, K. (1999), "Do Male Dropouts Benefit From Obtaining a GED, Postsecondary Education, and Training?" *Evaluation Review*, 23, 475–503. [754]
- Murthy, S., Kasif, S., and Salzberg, S. (1994), "A System for Induction of Oblique Decision Trees," *Journal of Artificial Intelligence Research*, 2, 1–32. [746]
- Pinheiro, J., and Bates, D. (2000), *Mixed-Effects Models in S and S-PLUS*, Berlin: Springer Verlag. [749]
- Quinlan, J. (1993), *C4.5: Programs for Machine Learning* (Vol. 1), San Francisco, CA: Morgan Kaufmann. [744]
- R Core Team (2012), *R: A Language and Environment for Statistical Computing*, Vienna, Austria: R Foundation for Statistical Computing, ISBN 3-900051-07-0. [749]
- Segal, M. (1992), "Tree-Structured Methods for Longitudinal Data," *Journal of the American Statistical Association*, 87, 407–418. [741,743,748,755]
- Sela, R., and Simonoff, J. (2012), "RE-EM Trees: A Data Mining Approach for Longitudinal and Clustered Data," *Machine Learning*, 86, 169–207. [741,748,749,755]
- Sela, R. J., and Simonoff, J. S. (2011), *REEMtree: Regression Trees With Random Effects*, R package version 0.90.3. [741,749]
- Shih, Y., and Tsai, H. (2004), "Variable Selection Bias in Regression Trees With Constant Fits," *Computational Statistics & Data Analysis*, 45, 595–607. [741,744]
- Siciliano, R., and Mola, F. (2000), "Multivariate Data Analysis and Modeling Through Classification and Regression Trees," *Computational Statistics & Data Analysis*, 32, 285–301. [741,743]
- Singer, J., and Willett, J. (2003), *Applied Longitudinal Data Analysis: Modeling Change and Event Occurrence*, New York, NY: Oxford University Press. [754]
- Therneau, T., Atkinson, B., and Ripley, B. (2013), *rpart: Recursive Partitioning*, R package version 4.1-1. [741]
- Wilson, E., and Hilferty, M. (1931), "The Distribution of Chi-square," *Proceedings of the National Academy of Sciences of the United States of America*, 17, 684–688. [745]
- Zhang, H. (1998), "Classification Trees for Multiple Binary Responses," *Journal of the American Statistical Association*, 93, 180–193. [741]