Statistics and Its Interface Volume 14 (2021) 295–308

Residual-based tree for clustered binary data

Rong Xia*, Christopher R. Friese, and Mousumi Banerjee

Tree-based methods are widely used for classification in health sciences research, where data are often clustered. In this paper, we propose a variant of the standard classification and regression tree paradigm (CART) to handle clustered binary outcomes. Using residuals from a null generalized linear mixed model as the response, we build a regression tree to partition the covariate space into rectangles. This circumvents modeling the correlation structure explicitly while still accounting for the cluster-correlated design, thereby allowing us to adopt the standard CART machinery in tree growing, pruning, and cross-validation. Class predictions for each terminal node in the final tree are estimated based on the success probabilities within the specific node. Our method also allows easy extension to ensemble of trees and random forest. Using extensive simulations, we compare our residual-based trees to the standard classification tree. Finally, the methods are illustrated using data from a study of kidney cancer and a study of surgical mortality after colectomy.

Keywords and phrases: Clustered data, Classification, Tree-based methods, Residuals, Kidney cancer, Colectomy surgical mortality.

1. INTRODUCTION

Tree-based methods have become one of the most flexible, intuitive, and powerful data analytic tools for exploring complex data structures. The applications of these methods are far reaching. The best documented, and arguably most popular uses of these methods are in health sciences research where classification is a central issue. The fitted tree identifies subgroups with common covariate values and homogeneous outcome, which is often used for decision making in the clinical setting. Some interesting applications of tree-based methods in the health sciences literature are described by Zhang and Singer [40], Banerjee et al. [4] and Segal et al. [31].

Tree-based methods were originally introduced by Morgan and Sonquist [29], and further advanced by Breiman et al. [7] in their monograph on Classification and Regression Trees (CART). In the CART paradigm, the covariate space is recursively partitioned into disjoint rectangle regions and the corresponding data is split into groups (nodes). The partitions are intended to increase within-node homogeneity in the response distribution. For each node, extent of homogeneity is measured quantitatively using an impurity function, *e.g.*, Gini or entropy for binary outcomes. At each step of the splitting process, a parent node gives rise to two daughter nodes (binary partitioning). Goodness of a split is assessed by the reduction in impurity going from the parent node to the two daughter nodes. All possible splits for each covariate are evaluated, and the covariate with the corresponding split point that results in the maximum impurity reduction is chosen. This splitting procedure is applied recursively until each node is pure in response or all predictor variables, or only contains a few observations. After a large tree is grown, there are rules for pruning and readjusting the size of the tree. The final result can be represented as a binary tree. Terminal nodes in the final tree represent subgroups characterized by common covariate values and homogeneous outcomes.

Clustered data frequently arise in the social, behavioral, and health sciences since individuals can be grouped in many different ways. For example, in studies of health services and outcomes, patients are clustered within physicians and/or hospitals (Haymart et al. [28], Miller et al. [20]). Such data are referred to as hierarchical/multilevel, with patients referred to as level 1 units and physicians/hospitals as level 2 units. The clustering induces correlation among individuals within the same cluster, and this intra-cluster correlation has to be accounted for in order to obtain valid statistical inferences.

Several authors have studied extensions of the original CART method to clustered outcomes. Historically, the attempt was to treat the clustered responses as a multivariate outcome and modify the tree splitting criterion accordingly. For continuous outcomes, Segal [30] proposed splitting functions that focus on either the mean vector (while treating the covariance as nuisance) or the covariance heterogeneity. Abdolell et al. [1] assumed a multivariate normal distribution and employed the likelihood ratio statistic to evaluate splits. Zhang [39] developed classification trees for clustered binary outcomes by employing three different splitting criteria, the generalized entropy, logarithm of the determinant of sample covariance, and a Wald-type statistic. However, these approaches require the cluster size to be equal and cannot handle individual-level covariates, such that individuals within a cluster always end up in the same node of the tree. Missing data is also a challenge for these methods.

Recently, multiple efforts have been made to build (generalized) mixed effects trees for clustered data. Hajjem et al.

*Corresponding author.

[16] and Sela and Simonoff [32] independently developed mixed effects regression trees for continuous outcomes. Hajjem et al. [17] also developed mixed effects random forests for regression. Hajjem et al. [18] further extended their work to non-continuous outcomes and proposed a generalized mixed effects tree model (GMERT). These authors used CART to model the fixed effect and global (node-invariant) random effects to capture the within cluster correlations. Since neither the fixed effect tree nor the random effects were known, Expectation-maximization (EM) algorithm was employed to iteratively estimate the model. Despite the flexibility of these models to handle unbalanced clusters with missing data, both individual- and cluster- level covariates, they have not been widely used in the literature. This may be largely due to the computational challenges in implementing the EM algorithm and lack of easily accessible software.

A few authors have also explored applying other types of tree models to clustered or longitudinal data. Lee [25] proposed to use generalized estimating equations (GEE) to build trees for general types of responses. Unlike CART, this method fits a GEE at each node and then splits based on the residuals. Loh and Zheng [27] and Eo and Cho [10] extended the Generalized, Unbiased, Interaction Detection and Estimation (GUIDE) [26] to longitudinal continuous outcomes. Fu and Simonoff [14]) developed a mixed effects regression tree using the conditional inference tree [22] as the fixed effect structure. Speiser et al. [34] combined Bayesian Mixed Models with CART to create a Binary Mixed Model (BiMM) tree for longitudinal binary outcomes. This addressed the model convergence issue in fitting a generalized mixed effects model and could incorporate prior knowledge. Instead of employing the EM algorithm in the estimation process, this approach applied three different functions to convert the fitted probabilities to binary outcomes, based on sensitivity, specificity and both. Speiser et al. [35] further extended this work and created Binary Mixed Model (BiMM) random forests. With the main goal of studying moderation effects between predictors, Bürgin and Ritschard [9] developed a varying coefficient regression model for longitudinal ordinal responses by extending multivariate generalized linear mixed models such that the fixed coefficients could vary as nonparametric functions of some variables, which were approximated by model-based recursive partitioning (MOB) [38]. This approach updated the splitting procedure, *i.e.*, the "coefficient constancy test" of MOB, so that global random effects were allowed in each split. Finally, Fokkema et al. [11] combined MOB with generalized linear mixed effects models and developed the generalized linear mixed-effects model (GLMM) trees. Unlike GMERT which assumes a constant value in each terminal node, this method allows fitting different parametric models in each partition.

Among these methods, (generalized) mixed effects type models generally include random effects to model the correlation structure explicitly, *e.g.*, [9], [11], [16], [17], [18], [34], [35]. Sela and Simonoff [32] and Fu and Simonoff [14] further allow the use of different residual covariance structures. This is particularly helpful when the correlation structure is complicated but well-understood, for example, in longitudinal studies, where both random intercept and random slopes could be used.

To address the challenges of iterative estimation usually required by the generalized mixed effects tree models, we propose an alternative solution to extend CART for cluster-correlated binary data. Our approach uses Pearson or deviance residuals from a null generalized linear mixed model as the outcome to partition the covariate space into rectangles. This circumvents modeling the correlation structure explicitly while still accounting for the cluster-correlated design, thereby allowing us to adopt the original CART machinery in tree growing, pruning and cross-validation. Our method can be viewed as a one-step approximation of the generalized mixed effects tree. It can handle both individual- and cluster- level covariates, does not require balance in cluster sizes, is able to handle missing data, and can be easily implemented in standard statistical softwares. Furthermore, our residualization approach can be easily combined with other types of tree methods. Lastly, our method lends itself to a natural extension to ensembles of trees, which can often give more accurate predictions and address the instability in a single tree.

This paper is organized as follows. In Section 2, we introduce the methodology for growing trees for clustered binary outcomes using residuals from a null generalized linear mixed model. Section 3 compares our residual-based trees to standard classification trees via simulations. We illustrate our methodology in Section 4 using data from a health services research study to investigate determinant of kidney cancer treatment receipt. Section 5 applies our methodology to investigate determinants of surgical mortality after receiving colectomy surgery. Finally, Section 6 contains some concluding remarks.

2. RESIDUAL-BASED TREE FOR CLUSTERED DATA

For clustered data, individuals within the same cluster are usually correlated. In familial segregation studies, family members are usually alike as they share the same genetic factors. In clinical studies, patients treated by the same provider are usually more similar in terms of treatment received. Popularized by Breslow and Clayton [8], generalized linear mixed effects models (GLMMs) have become a standard framework for modeling such clustered non-normal data, where the inclusion of cluster-specific random effects induces correlation among individuals within the same cluster. Consider a two-level hierarchical data structure: let y_{ij} be the binary response of the jth individual (level-one unit) in the ith cluster (level-two unit), where 0 stands for 'failure' and 1 stands for 'success', $i = 1, ..., m$, $j = 1, ..., n_i$, $N = \sum_{i=1}^{m} n_i$. The GLMM with logit link can be written as

$$g(\mu_{ij}) = log(\frac{\mu_{ij}}{1-\mu_{ij}}) = \mathbf{x}_{ij}\boldsymbol{\beta} + \mathbf{z}_{ij}\mathbf{b}_i, \quad (1)$$

where $\boldsymbol{\beta} = (\beta_0, ..., \beta_p)$ are the population level fixed effects coefficients, and $\mathbf{b}_i = (b_{i0}, ..., b_{iq})'$ are the random effects for cluster i. The $\mathbf{x}_{ij} = (1, x_{ij1}, ..., x_{ijp})'$ and $\mathbf{z}_{ij} = (1, z_{ij1}, ..., z_{ijq})'$ are the fixed effects covariates and random effects covariates, respectively, for the jth individual in cluster i. The random effects $\mathbf{b}_i$ are assumed to follow a multivariate normal distribution with mean $\mathbf{0}$ and covariance matrix $\boldsymbol{\Sigma}$. The $\mu_{ij} = E(y_{ij}|\mathbf{b}_i) = P(y_{ij} = 1|\mathbf{b}_i)$ is the conditional expectation of y_{ij} given random effects $\mathbf{b}_i$. Note that given random effects $\mathbf{b}_i$, all n_i individuals y_{ij} from cluster i are conditionally independent.

Parameter estimation in GLMMs typically uses maximum likelihood (ML) or variants of ML. This involves integration over the random effects that cannot be done analytically. Instead, numerical approximate algorithms such as penalized quasi-likelihood (PQL) or Gauss–Hermite quadrature (GHQ) are often employed. One limitation with PQL is that it tends to under-estimate the regression coefficients and variance components, especially when the random effects are heterogeneous (Jang and Lim [24]). And the biases usually decrease as cluster size increases. In contrast, GHQ uses numerical integration to approximate the true likelihood. The accuracy of GHQ increases as the number of integration points increase, at the cost of higher computations. The random effects $\mathbf{b}_i$ are estimated using empirical Bayes method.

(Generalized) linear mixed effects models can be extended by replacing the fixed effect linear part $\mathbf{x}_{ij}\boldsymbol{\beta}$ with tree $f(\mathbf{x}_{ij})$, *i.e.*,

$$g(\mu_{ij}) = f(\mathbf{x}_{ij}) + \mathbf{z}_{ij}\mathbf{b}_i. \tag{2}$$

The main challenge of fitting a (generalized) mixed effects tree lies in the model estimation, since neither the random effects nor the fixed effects are known. Several authors have proposed to solve this problem by essentially approaching it like an EM algorithm, *i.e.*, alternating between estimating the tree, assuming the estimated random effects are correct, estimating the random effects, assuming the estimated tree is correct, and cycle until convergence. Predictions from the fitted model are obtained using both estimated fixed effect tree and estimated random effects. When growing the tree, several versions of residualization have been employed. For example, for continuous outcomes, Hajjem et al. [16] and Sela and Simonoff [32] explicitly used conditional partial residuals $\mathbf{y}_{ij} - \mathbf{z}_{ij}\mathbf{b}_i$. A first-order Taylor series (*i.e.* linear) approximation of the conditional partial residuals were used when the outcomes were non-normal (Hajjem et al. [18]). Fokkema et al. [11], in contrast, included $\mathbf{z}_{ij}\mathbf{b}_i$ as an offset, which is equivalent to using conditional partial residuals when the outcomes are continuous. However, iterative estimations are not easily achievable.

2.1 Residuals from the null model

When modeling survival data, Therneau et al. [37] had advocated using null martingale residuals from a Cox proportional hazards model as the response to grow trees. Following this approach, we propose to grow trees for clustered binary data using classic residuals from the null GLMM as the new response.

The null GLMM $g(\mu_{ij}) = \beta_0 + b_{i0}$ includes one fixed effect β_0, which is the population-level intercept, and random effects b_{i0}, which are cluster-level intercepts. Random intercept b_{i0} captures the shared correlation among all individuals within cluster i. The prediction from this null model is $\hat{\mu}_{ij} = g^{-1}(\hat{\beta}_0 + \hat{b}_{i0})$, where $g^{-1}(.)$ is the inverse of the logit link. It is easily seen from this model that all n_i individuals from cluster i have the same predicted value, which is the estimated success probability for cluster i after accounting for the hierarchical structure.

Two types of residuals are commonly used for binary responses: the Pearson residual and the deviance residual [12]. For individual j in cluster i, given its prediction $\hat{\mu}_{ij}$ from the fitted null GLMM, the Pearson residual pr_{ij} can be defined as

$$pr_{ij} = \frac{y_{ij} - \hat{\mu}_{ij}}{\sqrt{\hat{\mu}_{ij}(1 - \hat{\mu}_{ij})}}. \tag{3}$$

The deviance residual dr_{ij} is defined as

$$dr_{ij} = sign(y_{ij} - \hat{\mu}_{ij})\sqrt{2y_{ij}log(\frac{y_{ij}}{\hat{\mu}_{ij}}) + 2(1 - y_{ij})log(\frac{1 - y_{ij}}{1 - \hat{\mu}_{ij}})}, \tag{4}$$

where $sign(y_{ij} - \hat{\mu}_{ij})$ is the sign of $y_{ij} - \hat{\mu}_{ij}$.

2.2 Residual-based trees

In the CART paradigm, the covariate space is recursively partitioned into disjoint rectangular regions and the data is divided into subgroups (nodes). Here we denote the rectangular region formed by terminal node l as R_l. Suppose tree T has L terminal nodes, then we can envision this tree as an additive model f with L terms, where each term corresponds to a terminal node

$$f(\mathbf{x}_{ij}) = \sum_{l=1}^{L} a_l I(\mathbf{x}_{ij} \in R_l), \tag{5}$$

and a_l is the predicted value of observations falling in terminal node l. Indicator function $I(\mathbf{x}_{ij} \in R_l) = 1$ if individual j of cluster i falling in terminal node l. Fitting a tree involves greedy searching for the optimized combination of L, R_l's and a_l's.

We propose to build the tree architecture by growing a regression tree using residuals from the fitted null GLMM as the response (*i.e.* in a transformed scale). At each stage, we search for the best split that maximizes the node impurity reduction, which is measured as the mean squared error of the residuals, *i.e.*, $\sum_{ij \in node}(r_{ij} - \bar{r})^2$, where r_{ij} is the residual of individual j in cluster i from the fitted null GLMM. After an overly complicated initial tree is grown, we prune it

back by applying cost complexity pruning on the residuals as well. Therefore, L and R_l's are both obtained by optimizing the within-node homogeneity based on the transformed outcomes.

We recommend applying 10-fold cross-validation for cost complexity pruning. Depending on the application, two criteria could be used for selecting the optimal tree (Therneau and Atkinson [36]). First, look for the tree with final splits corresponding to the *cp* value that minimizes the 10-fold cross-validation error. This tree should have the best prediction performance and the maximum power of detecting meaningful subgroups, which could be valuable for exploratory data analysis. However, there is a potential risk that this tree is still overly complicated. Another choice is the "one standard error (SE)" rule. Standard errors are also calculated with the cross-validations. After the minimal cross-validation error is found, choose the tree with final splits corresponding to the largest *cp* value that has 10-fold cross-validation error less than one standard error above the achieved minimal error. In this way, we choose the simplest tree that should have prediction performance "tied" with the best model. This conservative approach could further trim off false splits and provide only subgroups that are statistically meaningful. It is worth mentioning that through simulation studies, we have observed that "one SE" rule might not pick the most predictive residual-based tree, especially when the sample size is small. This is because when sample size is small, the cross-validation standard error tends to be large, hence an overly trimmed tree is selected.

Once the tree architecture (L and R_l's) has been selected, success probability for the lth terminal node is predicted by the proportion of success observations falling in that terminal node *i.e.*,

$$\hat{a}_l = \frac{\sum_{i,j} y_{ij} I(\mathbf{x}_{ij} \in R_l)}{\sum_{i,j} I(\mathbf{x}_{ij} \in R_l)}, \tag{6}$$

where y_{ij} is the raw binary outcome of individual j in cluster i. The predicted probabilities range from 0 to 1. If needed, classification prediction could be obtained by dichotomizing the estimated probabilities. Since it is individuals that are being split, not clusters, one terminal node might have individuals from different clusters. The clustering effect is accounted for in the residualization step, it does not complicate the selection of cut-offs. Default cut-off of 0.5 could be used; or overall success rate in the full training set, *e.g.*, $\sum_{i=1}^{m}\sum_{j=1}^{n_i} y_{ij}/N$ could be adopted to balance type I and type II error rates when the observed classes are skewed.

Our method can be applied to unbalanced data with varying cluster sizes. Furthermore, our method can split on both individual- and cluster- level covariates, since it is individuals that are being split, not clusters. Finally, our method inherits CART's flexibility to handle missing covariates via the surrogate variables approach.

The residual-based tree algorithm is as follows:

1. Fit a null GLMM with population-level intercept as the fixed effect and cluster-level random intercepts.
2. Calculate residuals r_{ij} from the fitted null GLMM model.
3. Grow a full regression tree using residuals r_{ij} as the response.
4. Prune the full tree back by applying cost-complex pruning on the residuals r_{ij} as well.
5. Convert the pruned tree terminal nodes into a set of indicators $I(\mathbf{x}_{ij} \in R_l)$.
6. Estimate class probability as the proportion of successes (raw outcomes) in each terminal node,

$$\frac{\sum_{i,j} y_{ij} I(\mathbf{x}_{ij} \in R_l)}{\sum_{i,j} I(\mathbf{x}_{ij} \in R_l)}. \tag{7}$$

One limitation of a single tree is that it is usually unstable, where a small change in data may largely affect the tree architecture. Another shortcoming is its modest prediction performance. Ensemble methods such as bagging (Breiman [5]) and random forests (Breiman [6]) can greatly improve these deficiencies. The framework of our residual-based approach can be easily extended to generate residual-based ensemble of trees, where the architecture of each tree in the ensemble is built using residualized responses, and the predictions are estimated from the raw binary outcomes.

2.3 Software implementation

Our residual-based tree algorithm can be easily implemented in R. The null GLMM could be fitted using either the "glmmPQL" function from the "MASS" package (default), which utilizes the penalized quasi-likelihood algorithm; or the "glmer" function from the "lme4" package, which employs adaptive Gauss-Hermite quadrature. The later can correct for the potential bias in the PQL estimation at a higher computational cost. In our applications, we have seen that these two functions give almost identical null residuals. The regression tree with residuals as the new response is grown and pruned using the standard "rpart" package. We create an R function that extracts the architecture of the regression tree and gives the class prediction for every terminal node.

3. SIMULATION STUDIES

In this section, we compared our residual-based tree to standard classification tree via simulations. Our comparisons focused on the architecture of the fitted trees as well as their prediction performance.

3.1 Simulation design

We first generated data from a two-level hierarchical design with 75 clusters, and equal cluster size of 5, 10, 50 to 100. We also considered the situation in which the number of individuals per cluster vary from 50 to 100. For the jth

individual in cluster i, its binary response y_{ij} was generated from a two-level random intercept model via a latent variable formulation

$$y_{ij}^* = log(\frac{p(\mathbf{x}_{ij})}{1-p(\mathbf{x}_{ij})}) + b_i + \epsilon_{ij}, \tag{8}$$

where the random intercept b_i was generated from normal distribution $N(0, \sigma_b^2)$, and the level-one error ϵ_{ij} was generated from a logistic distribution with mean 0 and variance $\pi^2/3$. We defined $y_{ij} = 1$ if $y_{ij}^* > 0$, and $y_{ij} = 0$ otherwise.

For every observation, eight independent covariates were generated 1) from two distributions, 2) at the cluster- and individual- levels, *i.e.*, $\mathbf{x}_{ij} = (x_{ij1}, ..., x_{ij8})'$. Covariates x_{ij1}, x_{ij2}, x_{ij5} and x_{ij6} followed standard normal distributions, while x_{ij3}, x_{ij4}, x_{ij7} and x_{ij8} had Bernoulli distributions with mean 0.5. The x_{ij1}, x_{ij3}, x_{ij5} and x_{ij7} were individual-level covariates, *i.e.*, varied by individual j, while the others were at the cluster-level, *i.e.*, stayed the same for all individuals within cluster i.

The fixed effect $p(\mathbf{x}_{ij})$ depended on covariates x_{ij1} to x_{ij4} (x_{ij5} to x_{ij8} were noise variables) via an underlying tree illustrated in Figure 1, where $(p_1, p_2, p_3, p_4, p_5)' = (0.9, 0.4, 0.8, 0.3, 0.7)'$ were the marginal probabilities of success in terminal nodes:

- Terminal Node I: If $x_{ij3} = 0$ and $x_{ij1} \leq 0$, $p(\mathbf{x}_{ij}) = p_1$;
- Terminal Node II: If $x_{ij3} = 0$ and $x_{ij1} > 0$ and $x_{ij4} = 0$, $p(\mathbf{x}_{ij}) = p_2$;
- Terminal Node III: If $x_{ij3} = 0$ and $x_{ij1} > 0$ and $x_{ij4} = 1$, $p(\mathbf{x}_{ij}) = p_3$;
- Terminal Node IV: If $x_{ij3} = 1$ and $x_{ij2} \leq 0$, $p(\mathbf{x}_{ij}) = p_4$;
- Terminal Node V: If $x_{ij3} = 1$ and $x_{ij2} > 0$, $p(\mathbf{x}_{ij}) = p_5$.

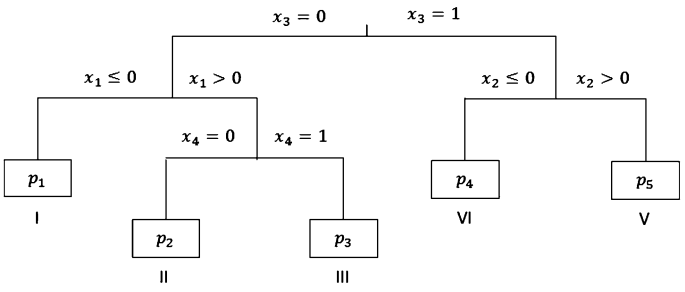

Figure 1. Underlying tree structure of the two-level random intercept model used for generating data, where $(p_1, p_2, p_3, p_4, p_5)'$ were the marginal probabilities of success in terminal nodes.

Under this simulation design, individuals from different clusters were independent while individuals within the same cluster were correlated. The strength of correlation could be expressed by the intra-cluster correlation coefficient (ICC), which was defined as the ratio of between-cluster variance to total variance, *i.e.*,

$$ICC = \frac{\sigma_b^2}{\sigma_b^2 + (\pi^2/3)}.$$

To study the effect of different within cluster correlations, we varied σ_b^2 between 0, 0.5^2, 1^2 and 2^2, which corresponded to estimated ICC of 0, 0.07, 0.23 and 0.55.

The simulated data were randomly divided into a training set and a test set by clusters. The training set contained 50 clusters and was used for building the tree models. The remaining 25 clusters were employed as the test set for evaluating predictions. The simulations were repeated 1,000 times.

3.2 Simulation results

We compared how similar the fitted residual-based tree and the standard classification tree were to the underlying true tree model. Two choices of similarity metrics were used, namely, tree size, *i.e.*, the number of terminal nodes, and the number of times each covariate was used as a splitter in the tree.

In addition, we argued that the fitted tree and the underlying true model were similar if they place the same observations together in a terminal node and separate the same observations in different terminal nodes (*i.e.*, if observations g and h were in two different terminal nodes under the true model T_0, then these two observations should also be placed in two different terminal nodes by the fitted tree T_1 for it to be similar to the underlying true model). As introduced in Banerjee et al. [2], we employed a metric d to quantify how observations were clustered in tree terminal nodes. For all $\binom{N}{2}$ pairs of observations, if observations g and h were in the same terminal node by tree T, then $I_T(g,h) = 1$, otherwise $I_T(g,h) = 0$. The difference of terminal nodes (over all distinct pairs of observations) between the fitted tree T_1 and underlying true model T_0 was then measured as

$$d(T_0, T_1) = \frac{\sum_{g>h} \sum_h |I_{T_0}(g,h) - I_{T_1}(g,h)|}{\binom{N}{2}}, \tag{9}$$

where the factor $\binom{N}{2}$ scaled this metric to range from 0 to 1 such that $d = 0$ when the terminal nodes of fitted tree T_1 were exactly the same as the terminal nodes of underlying true model T_0, and $d = 1$ when they were completely different.

Lastly, we compared the fitted residual-based trees and the standard classification tree in terms of prediction accuracy using AUC evaluated on the test data. AUC was known as the area under the Receiver Operating Characteristic (ROC) curve formed by a binary classifier, which estimated the probability that the classification predictions and the actual binary responses were concordant, *i.e.*, observations with the larger y also had the larger $\hat{y}$. It is

considered the standard method to assess the accuracy of predictive classification models. As it avoids the subjectivity in the threshold selection when converting probability scores to binary outcomes, by summarizing overall model performance over all possible thresholds. The values of AUC should range from 0.5 to 1, where 0.5 indicated that the predictions were no better than random guessing, and 1 indicated that the classifier had perfectly identified all binary responses.

Penalized quasi-likelihood estimation via "glmmPQL" and "rpart" package were used in this simulation. The default values for most "rpart" parameters were adopted, expect for "minsplit" and "cp". "minsplit" controls the minimum number of observations that must exist in a node in order for a split to be attempted, and it was set to 10. "cp" is the complexity parameter, and only splits that reduce the impurity by a factor of "cp" will be attempted. "cp" was updated to 0. We first grew an overly complicated tree. Then we pruned it back via 10-fold cross-validation cost complexity pruning. "one SE" rule was used for choosing the optimal tree size. By doing so, we removed the potential impact of arbitrary parameter values, and let data decide the statistically meaningful tree size.

Table 1 contains the averages of the above mentioned metrics over 1,000 simulations, under different combinations of cluster size (n) and random effect (σ_b^2). To examine the variation over different runs, in Figure 2–4 we show the boxplots of fitted tree sizes, terminal nodes similarity metrics d, and AUC on the test set. Within each figure, the top, second, third, fourth and bottom panel correspond to equal-sized clusters of size 5, 10, 50, 100 and unequal clusters with sizes varying from 50 to 100, respectively.

When intra-cluster correlation is zero or small, *e.g.*, $ICC = 0$ or 0.07, the architectures of the fitted standard classification tree (RPART), Pearson residual-based tree (PR) and deviance residual-based tree (DR) are all similar to the underlying true tree model, for equal size clusters 50, 100 and unequal clusters with sizes varying from 50 to 100. The average fitted tree sizes are all around 5; the four signal covariates X_1 to X_4 are correctly selected for splitting, and each covariate is split once on average; and the terminal nodes similarity metric d are all near 0. The prediction performances of the three fitted trees are also similar based on their AUCs. The boxplots further indicate that these statistics have little variations over the 1,000 simulations.

When intra-cluster correlation is moderate to strong, *e.g.*, $ICC = 0.23$ or 0.55, and cluster size is reasonably large, RPART fits overly complicated trees, with average tree sizes much larger than 5. This is primarily because RPART fails to distinguish between signal and noise variables, and frequently splits on the noise variables X_5, X_6 and X_8. One possible explanation is that without properly accounting for the clustering correlation, random effects give rise to spurious subgroup detection, which is represented by noise

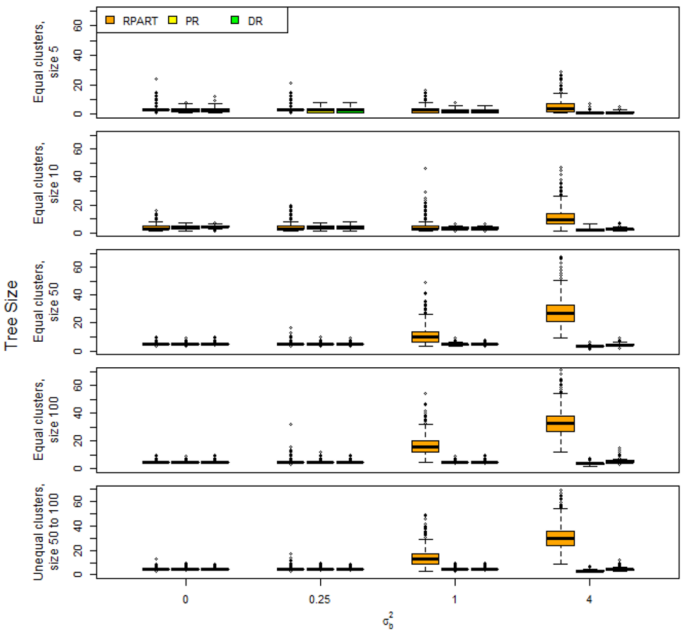

Figure 2. Boxplot of tree sizes over 1,000 *simulations, under different random intercept variance* σ_b^2*. The top panel is for equal cluster size* 5*, the second panel is for equal cluster size* 10*, the third panel is for equal cluster size* 50*, the fourth panel is for equal cluster size* 100*, and the bottom panel is for unequal clusters with sizes varying from* 50 *to* 100 *(RPART: standard classification tree; PR: Pearson residual-based tree; DR: Deviance residual-based tree).*

variables. Furthermore, it shows a propensity to over-split individual-level continuous covariates. This agrees with the well-known selection bias issue of RPART (Hastie et al. [19]). In contrast, PR and DR, particularly DR, fit trees with sizes close to the true underlying tree. Both PR and DR are unaffected by noise variables, and in general, make correct splits on the signal covariates only. The average of metric d over 1,000 simulations is consistently smallest for DR, indicating that the DR tree is most similar to the true underlying tree in terms of terminal nodes. Furthermore, the AUC of the DR tree is consistently larger than the PR or RPART tree, demonstrating its superior prediction performance.

The performance of both residual-based trees and the standard classification tree decrease as cluster size (hence sample size) gets smaller. When cluster size is 5 or 10, all three methods generate overly pruned trees than the truth. This is partly because we have chosen the "one SE" rule in deciding the optimal tree size, which prefers conservative trees over plausible splits. The small sample size also makes the fitted trees highly variable. The improvements of PR and DR trees over RPART are less evident when cluster size is small. In fact, for $ICC = 0.55$ and cluster size 5, residual-based trees appear to be slightly worse

Table 1. Average of comparison metrics over $1{,}000$ *simulations (RPART: standard classification tree; PR: Pearson residual-based tree; DR: Deviance residual-based tree; ICC: intra-cluster correlation coefficient;* d*: terminal nodes similarity metric; AUC: Area under the receiver operating characteristic curve evaluated on test data)*

Cluster Size	σ_b^2	ICC	Tree Type	Tree Size	Selection Frequency								d	AUC
					X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8		
Equal, 5	0^2	0.00	RPART	3.4	0.3	0.9	0.8	0.1	0.1	0.1	0.0	0.0	0.279	0.651
			PR	2.9	0.4	0.6	0.7	0.1	0.0	0.0	0.0	0.0	0.317	0.641
			DR	2.9	0.4	0.6	0.7	0.1	0.0	0.0	0.0	0.0	0.304	0.647
	0.5^2	0.07	RPART	3.3	0.4	0.8	0.8	0.1	0.1	0.1	0.0	0.0	0.316	0.631
			PR	2.6	0.4	0.4	0.7	0.1	0.0	0.0	0.0	0.0	0.359	0.619
			DR	2.7	0.4	0.5	0.7	0.1	0.0	0.0	0.0	0.0	0.348	0.623
	1^2	0.23	RPART	3.4	0.4	0.8	0.7	0.1	0.1	0.2	0.0	0.0	0.372	0.600
			PR	2.1	0.3	0.2	0.5	0.1	0.0	0.0	0.0	0.0	0.480	0.576
			DR	2.2	0.3	0.3	0.5	0.1	0.0	0.0	0.0	0.0	0.458	0.582
	2^2	0.55	RPART	5.2	0.6	1.3	0.5	0.2	0.4	1.0	0.1	0.1	0.398	0.551
			PR	1.4	0.1	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.652	0.524
			DR	1.5	0.1	0.1	0.3	0.0	0.0	0.0	0.0	0.0	0.618	0.530
Equal, 10	0^2	0.00	RPART	4.0	0.5	1.1	1.0	0.4	0.1	0.0	0.0	0.0	0.156	0.703
			PR	4.1	0.8	0.9	1.0	0.4	0.0	0.0	0.0	0.0	0.093	0.726
			DR	4.2	0.8	0.9	1.0	0.4	0.0	0.0	0.0	0.0	0.088	0.731
	0.5^2	0.07	RPART	4.2	0.5	1.1	1.0	0.4	0.1	0.1	0.0	0.0	0.173	0.686
			PR	3.8	0.8	0.7	1.0	0.3	0.0	0.0	0.0	0.0	0.123	0.701
			DR	4.0	0.8	0.8	1.0	0.4	0.0	0.0	0.0	0.0	0.107	0.710
	1^2	0.23	RPART	4.8	0.6	1.3	1.0	0.3	0.2	0.3	0.0	0.0	0.205	0.657
			PR	3.2	0.7	0.3	0.9	0.2	0.0	0.0	0.0	0.0	0.201	0.654
			DR	3.5	0.8	0.5	1.0	0.2	0.0	0.0	0.0	0.0	0.157	0.671
	2^2	0.55	RPART	10.7	1.3	3.4	0.9	0.5	0.7	2.7	0.1	0.3	0.247	0.574
			PR	2.0	0.4	0.1	0.6	0.0	0.0	0.0	0.0	0.0	0.443	0.566
			DR	2.5	0.5	0.2	0.7	0.1	0.0	0.0	0.0	0.0	0.334	0.588
Equal, 50	0^2	0.00	RPART	5.2	1.0	1.0	1.1	1.0	0.0	0.0	0.0	0.0	0.030	0.769
			PR	5.0	1.0	1.0	1.0	0.9	0.0	0.0	0.0	0.0	0.010	0.773
			DR	5.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.006	0.774
	0.5^2	0.07	RPART	5.2	1.0	1.1	1.1	1.0	0.0	0.1	0.0	0.0	0.048	0.752
			PR	4.9	1.0	1.0	1.0	0.9	0.0	0.0	0.0	0.0	0.016	0.759
			DR	5.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.007	0.764
	1^2	0.23	RPART	11.2	1.5	3.4	1.1	1.2	0.4	2.3	0.0	0.2	0.100	0.709
			PR	4.6	1.0	0.7	1.0	0.8	0.0	0.0	0.0	0.0	0.049	0.721
			DR	4.9	1.0	1.0	1.0	0.8	0.0	0.0	0.0	0.0	0.020	0.734
	2^2	0.55	RPART	27.8	2.5	10.7	1.6	1.5	0.9	8.6	0.1	0.9	0.192	0.586
			PR	3.2	0.9	0.1	1.0	0.1	0.0	0.0	0.0	0.0	0.154	0.629
			DR	4.4	1.0	0.8	1.0	0.6	0.0	0.0	0.0	0.0	0.068	0.664
Equal, 100	0^2	0.00	RPART	5.2	1.1	1.0	1.1	1.1	0.0	0.0	0.0	0.0	0.015	0.774
			PR	5.3	1.0	1.2	1.0	1.0	0.0	0.0	0.0	0.0	0.008	0.776
			DR	5.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.002	0.776
	0.5^2	0.07	RPART	5.6	1.1	1.2	1.1	1.1	0.0	0.1	0.0	0.0	0.029	0.761
			PR	5.3	1.0	1.3	1.0	1.0	0.0	0.0	0.0	0.0	0.009	0.766
			DR	5.1	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.004	0.766
	1^2	0.23	RPART	16.7	1.7	5.8	1.3	1.5	0.5	4.4	0.0	0.5	0.108	0.713
			PR	5.1	1.0	1.1	1.0	1.0	0.0	0.0	0.0	0.0	0.013	0.740
			DR	5.1	1.0	1.1	1.0	1.0	0.0	0.0	0.0	0.0	0.008	0.742
	2^2	0.55	RPART	33.3	2.6	13.7	2.0	1.7	0.8	10.3	0.1	1.2	0.196	0.585
			PR	3.7	1.0	0.3	1.0	0.4	0.0	0.0	0.0	0.0	0.116	0.645
			DR	5.6	1.0	1.4	1.0	1.0	0.0	0.1	0.0	0.0	0.037	0.676
Unequal, varying from 50 to 100	0^2	0.00	RPART	5.3	1.1	1.0	1.1	1.1	0.0	0.0	0.0	0.0	0.022	0.772
			PR	5.1	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.006	0.774
			DR	5.1	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.004	0.775
	0.5^2	0.07	RPART	5.5	1.1	1.1	1.1	1.1	0.0	0.1	0.0	0.0	0.038	0.758
			PR	5.1	1.0	1.1	1.0	1.0	0.0	0.0	0.0	0.0	0.009	0.765
			DR	5.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.005	0.766
	1^2	0.23	RPART	13.9	1.6	4.7	1.2	1.3	0.4	3.3	0.0	0.4	0.101	0.710
			PR	4.9	1.0	0.9	1.0	1.0	0.0	0.0	0.0	0.0	0.025	0.731
			DR	5.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.010	0.738
	2^2	0.55	RPART	30.8	2.6	12.3	1.8	1.6	0.9	9.6	0.1	1.1	0.196	0.584
			PR	3.4	1.0	0.2	1.0	0.3	0.0	0.0	0.0	0.0	0.134	0.637
			DR	4.9	1.0	1.0	1.0	0.8	0.0	0.0	0.0	0.0	0.044	0.671

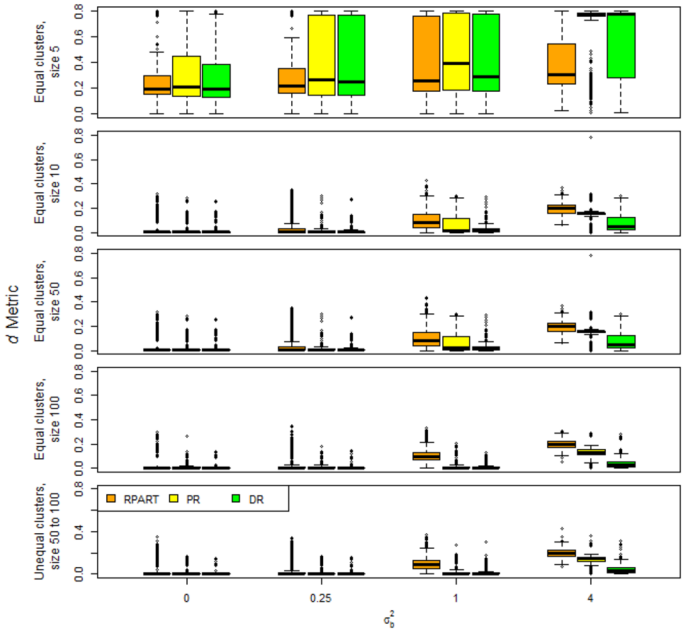

Figure 3. Boxplot of terminal nodes similarity metrics d over 1,000 *simulations, under different random intercept variance* σ_b^2. *The top panel is for equal cluster size* 5, *the second panel is for equal cluster size* 10, *the third panel is for equal cluster size* 50, *the fourth panel is for equal cluster size* 100, *and the bottom panel is for unequal clusters with sizes varying from* 50 *to* 100 *(RPART: standard classification tree; PR: Pearson residual-based tree; DR: Deviance residual-based tree).*

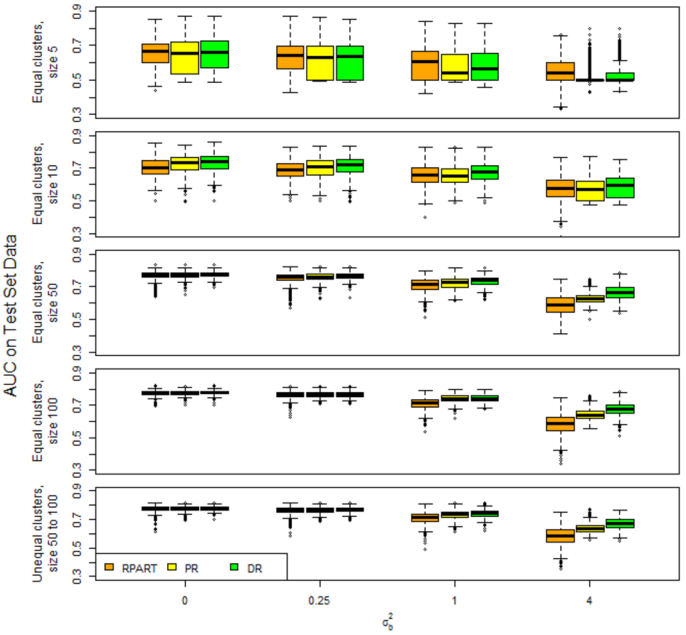

Figure 4. Boxplot of AUC (area under the receiver operating characteristic curve) evaluated on test data over 1,000 *simulations, under different random intercept variance* σ_b^2. *The top panel is for equal cluster size* 5, *the second panel is for equal cluster size* 10, *the third panel is for equal cluster size* 50, *the fourth panel is for equal cluster size* 100, *and the bottom panel is for unequal clusters with sizes varying from* 50 *to* 100 *(RPART: standard classification tree; PR: Pearson residual-based tree; DR: Deviance residual-based tree).*

than the standard classification tree. This is possibly due to the bias in the empirical Bayes estimation of the cluster effect ($\hat{b}_i$) when the cluster size is small (Skrondal and Rabe-Hesketh [33]), which could affect the residuals of the fitted null GLMM. It is worth mentioning that in addition to oversimplification, RPART also tends to mis-split on noise variables.

One heuristic explanation for the better performance of deviance residual-based tree over Pearson residual-based tree is that for extreme observations, *e.g.*, $y_{ij} = 0$ and $\hat{\mu}_{ij}$ from the null GLMM is around 1, or $y_{ij} = 1$ and $\hat{\mu}_{ij}$ is around 0, deviance residuals are smaller than Pearson residuals in absolute value, thereby preventing the deviance residual-based tree from being overly influenced by these extreme observations.

In summary, based on the simulations, we conclude that the residual-based trees outperform the standard classification tree for modeling clustered binary data. Specifically, deviance residual-based trees can recover the true underlying tree model, and provide accurate predictions. The improvements are substantial when the intra-cluster correlations are strong and the cluster sizes are moderate to large. Even when random effects are small or negligible, residual-based trees still show better or equal performance.

4. APPLICATION OF DEVIANCE RESIDUAL-BASED TREE AND RANDOM FOREST TO KIDNEY CANCER TREATMENT STUDY

To illustrate our method, we present an analysis of data from a population-based study of kidney cancer where the outcome of interest is receipt of treatment. Radical nephrectomy is the traditional gold standard for treating patients with organ-confined kidney cancer. During the last two decades, however, the introduction of a nephron-sparing alternative (*i.e.*, partial nephrectomy) to radical excision has appreciably modified the therapeutic options for patients with kidney cancer. Partial nephrectomy yields oncologic outcomes that are indistinguishable from radical excision, and it preserves long-term renal function while reducing overtreatment of patients with benign tumors. Despite these potential benefits, population-based data suggest that the adoption of partial nephrectomy has been slow, and radical nephrectomy remains the predominant surgical therapy for patients with kidney cancer (Hollenbeck et al. [21], Banerjee et al. [3]). The goal of our study was to apply the residual-based tree to understand the pattern of utilizing partial

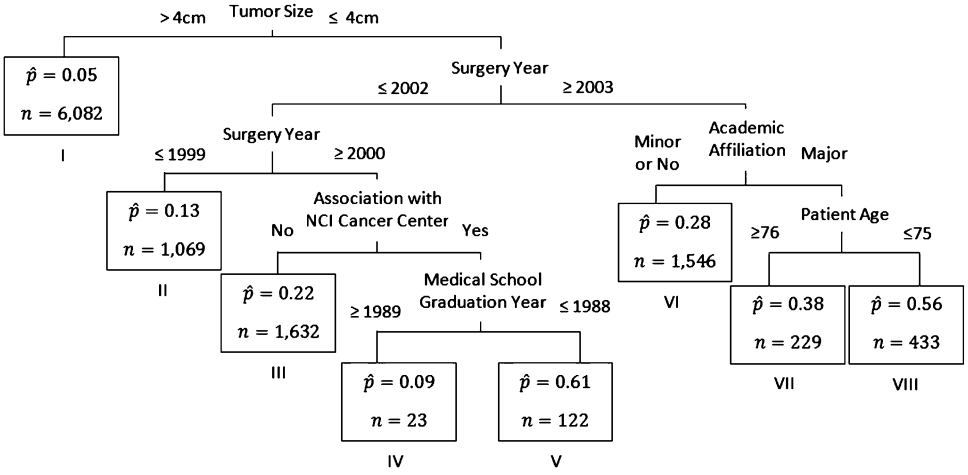

Figure 5. Deviance residual-based tree applied to kidney cancer data. In each terminal node, we list the estimated probability of receiving partial nephrectomy (PAR) ($\hat{p}$) and the number of patients (n) falling in that node.

nephrectomy in the population.

Our analysis cohort was comprised of 11,136 Medicare beneficiaries treated by 2,031 urologists for kidney cancer diagnosed between year 1995 and 2006. This data set exhibited a two-level hierarchical structure with patients nested within surgeons. The median number of patients treated by a surgeon was 4 (min = 1, Q1 = 2, mean = 5.5, Q3 = 7, max = 96). The outcome of interest was receipt of partial versus radical nephrectomy (*i.e.*, binary outcome). Among the 11,136 patients, 1,667 underwent partial nephrectomy, and the event rate is 0.15. The ICC is around 0.09. A total of sixteen covariates were considered for analysis, which included eight patient characteristics such as socio-demographic variables (age, race/ethnicity, gender, marital status and socioeconomic status), year of surgery, tumor size and the number of preexisting comorbid conditions (using a modification of the Charlson index based on claims submitted during the 12 months before kidney cancer surgery). On the basis of standard clinical guidelines, we categorized tumor size as $\leq$ 4 cm, 4.1–7 cm and $>$ 7 cm. We also considered eight surgeon-level covariates including the surgeon's age, gender, year of medical school graduation, practice size (solo or two-person, group practice, HMO or hospital-based, medical school, or other/unclassified), practice location (rural vs. urban), academic affiliation (major, minor, or no academic affiliation), surgeon's association with a National Cancer Institute (NCI)-designated Cancer Center, and surgeon's average annual nephrectomy volume during the study period. Among these sixteen covariates, patient's age, number of preexisting comorbid conditions, surgeon's age and annual nephrectomy volume were treated as continuous variables, while others were treated as nominal variables.

We implemented our residual-based tree approach on the entire cohort of 11,136 Medicare beneficiaries. After a full tree was grown on the transformed outcome, we performed cost complexity pruning. The final tree was chosen based on 10-fold cross-validation, and the tree with minimum error in the residualized outcome was selected. Here we have chosen the tree with the minimal cross-validation error because we would like to retain the maximum power in detecting clinical meaningful subgroups. And based on the simulation study, we have found that "one SE" rule might give overly simplified trees when cluster size is small. The deviance and Pearson residual-based trees were very similar. In Figure 5 we present the deviance residual-based tree. At each level of the tree, we show the best split (covariate with cut-point). The numbers in the terminal nodes denote the estimated probability of receiving partial nephrectomy (PAR) ($\hat{p}$) and the number of patients (n) falling in that node.

The deviance residual-based tree identified patient's tumor size and the year of performing surgery as the primary factors of receiving PAR. For patients with tumor size $>$ 4 cm, the estimated probability of receiving PAR was only 5%. As a relatively new technique, utilization of PAR gradually increased over the study period. Surgeons' association with NCI-designated cancer centers seemed to be another factor of utilizing PAR. Between year 2000 and 2003, among surgeons associated with NCI-designated cancer centers who graduated from medical school before 1989, the estimated probability of performing PAR on patients with tumor size $\leq$ 4 cm was as high as 61%! In contrast, during this time, for surgeons who were not associated with NCI-designated cancer centers, the estimated probability of performing PAR on patients with tumor size $\leq$ 4 cm was only 22%. It was also noticeable that even among surgeons associated with NCI-designated cancer centers, the estimated probability of utilizing PAR was merely 9% if they graduated from medical school after 1989. One possible explanation was that these surgeons were at their early career edge of independently performing surgeries, which might have restricted them from performing the new technique PAR. Since 2003, surgeons' academic affiliations became a factor of utilizing PAR. Sur-

geons with minor or no academic affiliations performed PAR on about 28% of their patients with tumor size ≤ 4 cm. On the other hand, the estimated probability of utilizing PAR was much higher for surgeons with major academic affiliations. Among these surgeons, the estimated probability of utilizing PAR was 56% if their patients were younger than 75 years with tumor size ≤ 4 cm; and the estimated probability was 38% if the patients were older than 76 years with tumor size ≤ 4 cm.

We also analyzed this data by growing a deviance residual-based random forests. Individual tree structures were lost in growing the forest, therefore, we evaluated the effect of covariates by examining their permutation variable importance. For each covariate, its permutation importance was calculated as the average percentage increase in mean squared error (MSE) of the predicted responses (in residualized outcome) from the forest, after randomly permuting the values of this variable. The permutation variable importance plot is displayed in Figure 6. This plot again confirms that tumor size is the most important determinant of PAR usage. The second and fourth most important factors according to the ranked variable importance, *i.e.*, year of medical school graduation and year of surgery also align with our findings from the single deviance residual-based tree. Surgeon age was also deemed important in the residual-based forest.

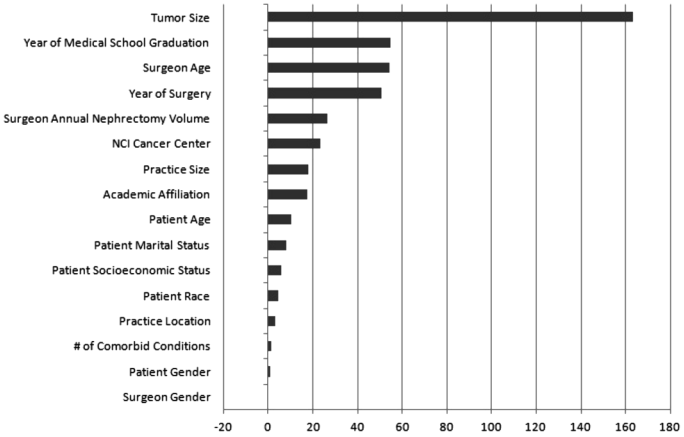

Figure 6. Variable importance plot of deviance residual-based random forest applied to kidney cancer data. Variable importance is defined as the percentage of increase in mean squared errors (MSE) of the predicted residualized responses, after randomly permuting a variable.

5. APPLICATION OF DEVIANCE RESIDUAL-BASED TREE AND RANDOM FOREST TO SURGICAL MORTALITY STUDY

Understanding the relationship between hospital/patient characteristics and patient outcomes is important for improving health care quality. In this analysis, we were interested in identifying hospital characteristics and patient risk factors that might be associated with patient outcomes after receiving colon resection surgeries (Friese et al. [13]).

We extracted data from nationwide Medicare inpatient claims files between year 2009 and 2010 on patients hospitalized for colon resection. A total of 58,816 patients 65 years or older, enrolled in fee-for-service Medicare were included in our analysis, and these patients were treated in 3,189 hospitals. The median number of colectomy patients treated in each hospital is 12 (min = 1, Q1 = 4, mean = 18.4, Q3 = 26, max = 173). We measured patient outcomes using failure to rescue (FTR), which is defined as death within 30 days of hospital admission for patients who have experienced a postoperative complication. FTR focuses on a hospital's capability to recognize and address a complication and is less affected by the severity of patients' illness, therefore, it is considered as a better measure for comparing hospital quality (Ghaferi et al. [15]). Of the 58,816 patients, 14,340 experienced a FTR, and the event rate is 0.24. The ICC is about 0.02. Seven hospital characteristics were considered, including a hospital's recognition of Magnet status by the American Nurses' Credentialing Center, which was a voluntary program reflecting a hospital's nursing care quality; the geographic location (rural vs. urban); whether a hospital had an active organ and/or tissue transplant program; whether a hospital had full-time equivalent medical residents or fellows; the number of staffed beds; a hospital's cost to charge ratio; and a hospital's registered nurse hours per patient day (RNHPPD). Patient risk factors included age (categorized into an ordinal variable as 65–69, 70–74, 75–79, 80–84, 85 years and older), gender, race/ethnicity, and the number of comorbid conditions reported on their insurance claims. Among all considered covariates, hospital's number of staffed beds, cost to charge ratio, RNHPPD, and patient's number of comorbid conditions were treated as continuous variables, patient's age was treated as ordinal, and others were treated as nominal variables.

This data set exhibited a two-level hierarchical structure as patients were nested within hospitals. We accounted for this hierarchical structure by fitting a hospital-specific random effect in the null GLMM. Deviance residuals from the fitted null model were used as response in growing the tree and random forest. The final tree is selected minimizing 10-fold cross-validation error, which is presented in Figure 7. At each level of the tree, we show the best split covariate along with the cut-point of the best split. For each terminal node, we present the estimated failure to rescue rate ($\hat{p}$) and the number of patients (n) in that node.

The deviance residual-based tree first split by patients' age and divided into three cohorts with age 65–74, 75–84, and 85 or older. As expected, FTR rates increased with patients' age. Patients aged 65–74 were further split by their comorbid conditions: Terminal node I contained the 6,312 patients with 3 or more comorbid conditions, who had the lowest FTR on average, which was 16%; terminal node II

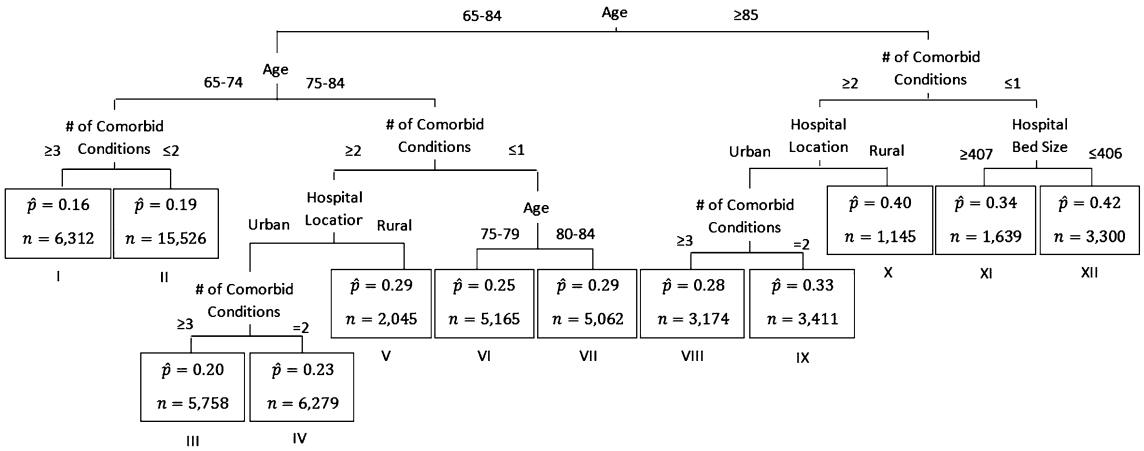

Figure 7. Deviance residual-based tree applied to surgical mortality data. In each terminal node, we list the estimated probability of failure to rescue rate (FTR) ($\hat{p}$) and the number of patients (n) falling in that node.

contained the 15,526 patients with no more than 2 comorbid conditions, and their average FTR was 19%. For patients in this age group, hospital characteristics did not show an association with FTR. Among patients aged 75–84 with 2 or more comorbid conditions, FTR was higher in rural hospitals than in urban hospitals, as indicated by terminal nodes III, V and IV. In addition, terminal nodes I and VII suggested that among patients with no more than 1 comorbid condition, the average FTR was 5% higher for age 80–84 than age 75–79. Patients older than 85 years were further divided by their comorbid conditions, as well as location and bed size of the hospitals they were treated in: Terminal nodes VIII, IX and X indicated that among patients with 2 or more comorbid conditions, the average FTR in rural hospitals was 40%, which was much higher than urban hospitals; for patients with no more than 1 comorbid condition, the average FTR was 42% in hospitals with less than 406 staffed beds, comparing to 34% in hospitals with more than 407 staffed beds, as illustrated by terminal nodes XI and XII.

In summary, through this deviance residual-based tree, we found that failure to rescue exhibited an increasing trend with patients' age. The effects of hospital characteristics were more evident among older patients, who were commonly considered frailer. Older patients treated in bigger and/or urban hospitals tended to have lower FTR. Our findings on patients' comorbid conditions were confusing, since patients with more comorbid conditions appeared to have lower FTR. However, the frequency table demonstrated that for patients with 0, 1, 2, and 3 or more comorbid conditions, the crude FTR was 26%, 27%, 24%, and 21%, respectively. This observed pattern, although apparently counterintuitive, was accurately identified by our residual-based tree. One possible explanation for this phenomenon is the bias in coding comorbidities, also known as "DRG Creep" (Iezzoni [23]). The number of comorbid conditions is collected from a patient's insurance claim, rather than the medical records. Thus, it is not a precise reflection of a patient's illness condition. Furthermore, hospitals with more resources are likely to identify and report more comorbidities in their patients' insurance claims, in order to receive higher reimbursements. These better resourced hospitals usually provide better health care service as well. Therefore, patients' comorbid conditions could be a confounder of hospitals' service quality.

The permutation variable importance based on deviance residual-based random forest is plotted in Figure 8. The two most important variables, patients' age and hospitals' bed size matched with our findings from the single deviance residual-based tree. This confirmed our conclusion that FTR is primarily associated with patients' age, and larger hospitals have lower FTR in general. Interestingly, the importance of hospital location was relatively low, which is possibly due to its confounding with other hospital characteristics such as bed size and teaching program, as urban hospitals are usually bigger and more likely to have teaching program.

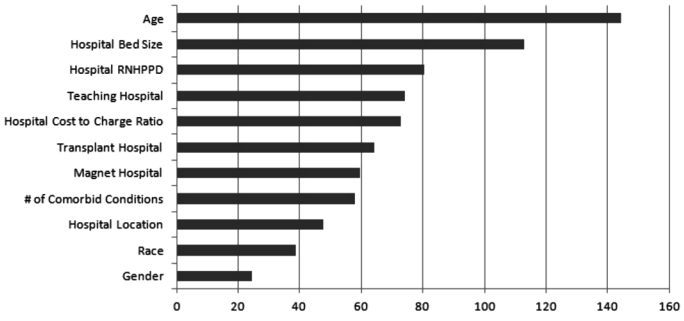

Figure 8. Variable importance plot of deviance residual-based random forest applied to surgical mortality data. Variable importance is defined as the percentage of increase in mean squared errors (MSE) of the predicted residualized responses, after randomly permuting a variable.

We also performed the standard classification tree analysis to this data. However, the standard classification tree model was unable to find any split (using minimal 10-fold cross-validation error rule), and simply returned a root node. Therefore, this surgical mortality example illustrates a real scenario where our residual-based trees uncovered patterns and structures in the data that the standard classification tree failed to identify.

6. CONCLUSION AND DISCUSSION

In this paper, we proposed a variant of the standard CART for modeling clustered binary outcomes. Our approach was based on using classic residuals from a null generalized linear mixed model as the response. This circumvents modeling the correlation structure explicitly while still accounting for the cluster-correlated design, thereby allowing us to adopt the original CART machinery in tree growing, pruning and cross-validation. Class predictions were estimated as the success probability, *i.e.*, average of the raw binary outcomes within each terminal node. The main advantage of our residual-based tree is that it is easy to implement in R. It allows for unbalanced clusters with different sizes, splits by any attribute, and can naturally handle missing covariate data. We also provide an extension of our residual-based approach to random forest.

Through extensive simulations, we have shown that our residual-based trees, especially the deviance residual-based trees, outperform the standard classification tree. The residual-based trees are better at identifying the true tree structure in the data, and provide more accurate predictions. The improvements over the standard classification tree are substantial when tree is the correct model, under strong intra-cluster correlations and moderate cluster sizes. We also applied our residual-based approach to studies of kidney cancer and surgical mortality after colectomy, where the data exhibited cluster correlated structures. In both studies, residual-based tree and random forests identified clinically meaningful subgroups. For the surgical mortality data, the standard classification tree failed to split at all, which further demonstrated the advantage of our residual-based approach.

We would like to point out that the predictions of our residual-based trees do not include the estimated random effects $\hat{\mathbf{b}}_i$. Instead, we are using $\hat{f}(\mathbf{x}_{ij})$, and the estimated random effects are folded in by taking average of all observations in a terminal node. One potential approach to further include random effects is to refit a generalized mixed effects model by converting the fitted residual-based tree structure to a set of indicator variables and treating them as the fixed effect. However, we argue that one should really perform a full iterative estimation (*e.g.*, Hajjem et al. [18], Sela and Simonoff [32]) if they are interested in specifically adopting certain random effects and/or correlation structures.

This brings up an interesting difference between the mixed effects trees and our residual-based trees. In mixed effects tree models, random effects and correlation structures are introduced to capture the correlations between individuals within a cluster. Both random intercept and random coefficients can be included, allowing the flexibility of modeling complicated correlation structures. Since the final predictions are based on of the estimated random effects and the fixed effect tree, it is important to correctly specify both parts of the model, which could be challenging in real applications. Additionally, the iterative estimation is not easily accessible.

The motivation of this paper was to offer an easily accessible approach that could handle clustered structure data reasonably well, a structure that has been typically ignored (or underutilized). Unlike mixed effects tree models, residual-based trees do not attempt to specify random effects and explicitly model all potential complicated correlation structures. Instead, they are designed for the common scenario when individuals within a cluster are correlated through a shared clustering effect. This amounts to a compound symmetry structure. Since predictions are based on the proportion of success in each terminal node, it is not critical to precisely estimate the random effect values, as long as the tree structure is correct. Our simulations demonstrated that the fitted tree structures are close to the truth when using Pearson or deviance residuals from the null GLMM. It is worth mentioning that when applying the mixed effects tree to longitudinal continuous data, Sela and Simonoff [32] suggested that including autoregressive correlation structure (AR(1)) had minimal impact on the structure of the fitted tree.

Instead of iterating between estimating the random effects and the fixed effect tree, our method can be viewed as a one-step approximation of the generalized mixed effect tree. Through simulations, we have shown that this approximation works reasonably well at recovering the truth. A heuristic explanation is that the null GLMM adequately accounts for the within-cluster correlation, and the residual-based trees estimate the truth quite well. While additional iterations might improve the random effects estimation, the benefit to the fitted tree is minimal. Interestingly, Sela and Simonoff [32] showed that one-step approximation of the mixed effect regression tree had a decent performance in modeling correlated continuous data. Speiser et al. [34] also showed that most BiMM trees converged in two iterations, and BiMM tree with one iteration generally had similar prediction accuracy than BiMM tree with multiple iterations in simulations, unless when there was a large random effect in the data generating process. Speiser et al. [35] further pointed out that multiple iterations could cause overfitting in BiMM forest.

There is one important consideration on the iterative estimation that has not received much attention in previous literature. Sela and Simonoff [32], Fokkema et al. [11] and Speiser et al. [34] recommend starting with zero random effect to grow the tree first. In contrast, Hajjem et al. [16] [18] start with a null model and estimate the random effects first. When only individual-level covariates exist or the

fixed effects model is wrong, this difference in the initialization does not seem to make much difference. However, when the fixed effects model is a tree and there are cluster-level covariates and strong within-cluster correlations (large random effects), our simulations (see supplement) suggested that initializing with a null model and estimating the random effects first work much better. An intuitive explanation for this is that without the estimated random effects, cluster-level covariates could be misused for splitting in the first tree to account for the within-cluster correlation. Subsequent iterations are needed to overcome this issue.

Another issue is the variable selection bias of CART and its potential implication on our residual-based trees. In selecting variables for partitioning, CART is known to be biased towards variables with more possible splits (Hastie et al. [19]). However, this selection bias is usually minor, unless among noise variables or variables that are only weakly correlated with the response. As called out in Fu and Simonoff [14], "Note that in all of the discussion on bias, it is considered as a property under the null hypothesis; that is, CART prefers to split on variables with more possible splits when the variables have no predictive power." For clustered data, it is not surprising that individual-level covariates tend to offer more possible splits than cluster-level covariates and hence could be overly selected. In simulations, we observed that cluster-level covariates were less likely to be split on when the cluster sizes were small (size 5 or 10) or when the clustering effect was very strong ($ICC = 0.55$), which coincided with situations when the null GLMM random effects estimations were unstable. One possible alternative is to employ other type of tree models. Our residualization approach can be easily combined with other tree models such as Generalized, Unbiased, Interaction Detection and Estimation (GUIDE) [26], conditional inference tree [22] or model-based recursive partitioning (MOB) [38]. MOB builds segmented parametric models and applies score-based fluctuation tests on parameter instabilities to find splits. In contrast to CART which fits a constant in each terminal node, MOB fits a local model in each partition. Function "glmtree" from R package "partykit" fits the MOB tree. The readily available "offset" parameter allows for incorporating the estimated random effects from the null GLMM in growing the tree, which can be viewed as a special form of residualization. Function "glmertree" from package "glmertree" further implements the iteratively estimated mixed effects MOB tree model. It is interesting to point out that "glmtree" with null GLMM random effects as "offset" can be treated as a one-step approximation of "glmertree". In experiments (see supplement), we have seen similarly good performances from both functions when used with a null GLMM, which demonstrates the validity and generalizability of our residualization idea.

SUPPLEMENTARY MATERIALS

An R program implementing the residual-based tree/random forests algorithm is attached. Additional simulation results, and the comprehensive analyses of the kidney cancer and the colectomy surgical mortality studies are also presented in the supplement (http://intlpress.com/site/pub/files/_supp/sii/2021/0014/0003/SII-2021-0014-0003-s002.zip).

ACKNOWLEDGEMENTS

We would like to thank the associate editor and the reviewers for their insightful comments which have greatly improved this manuscript.

Received 2 December 2019

REFERENCES

[1] Abdolell, M., LeBlanc, M., Stephens, D. and Harrison R. (2002). Binary partitioning for continuous longitudinal data: categorizing a prognostic variable. *Statistics in Medicine* **21**(22):3395–3409. MR3203208

[2] Banerjee, M., Ding, Y. and Noone, A.M. (2012). Identifying representative trees from ensembles. *Statistics in Medicine* **31**(15):1601–1616. MR2101453

[3] Banerjee, M., Filson, C., Xia, R. and Miller, D.C. (2014). Logic regression for provider effects on kidney cancer treatment delivery. *Computational and Mathematical Methods in Medicine*, 2014. MR2892116

[4] Banerjee, M., George, J., Song E.Y., Roy, A. and Hryniuk W. (2004). Tree-based model for breast cancer prognostication. *Journal of Clinical Oncology* **22**(13):2567–2575. MR2751671

[5] Breiman, L. (1996). Bagging predictors. *Machine Learning* **24**(2):123–140. MR4079933

[6] Breiman, L. Random forests. (2001). *Machine Learning* **45**(1):5–32. MR4079933

[7] Breiman, L., Friedman, J., Stone, C.J. and Olshen, R.A. (1984). *Classification and Regression Trees*. CRC Press.

[8] Breslow, N.E. and Clayton, D.G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* **88**(421):9–25. MR1049416

[9] Bürgin, R. and Ritschard, G. (2015). Tree-Based Varying Coefficient Regression for Longitudinal Ordinal Responses. *Computational Statistics & Data Analysis* **86**:65–80. MR2439970

[10] Eo, S.-H. and Cho, H. (2014). Tree-structured Mixed-effects Regression Modeling for Longitudinal Data. *Journal of Computational and Graphical Statistics* **23**:740–760.

[11] Fokkema, M., Smits, N., Zeileis, A., Hothorn, T. and Kelderman, H. (2018). Detecting treatment-subgroup interactions in clustered data with generalized linear mixed-effects model trees. *Behavior Research Methods*. **50**(5):2016–2034. MR2985421

[12] Forthover, R.N., Lee, S.L. and Hernandez, M. (2007). *Biostatistics: A Guide to Design, Analysis, and Discovery*. 2nd ed. San Diego: Academic Press.

[13] Friese, C.R., Xia, R., Ghaferi, A.A., Birkmeyer, J.D. and Banerjee, M. (2015). Hospitals in 'magnet' program show better patient outcomes on mortality measures compared to non-'magnet' hospitals. *Health Affairs* **34**(6):986–992.

[14] Fu, W. and Simonoff, J.S. (2015). Unbiased Regression Trees for Longitudinal and Clustered Data. *Computational Statistics and Data Analysis* **88**:53–74.

[15] Ghaferi, A.A., Birkmeyer, J.D. and Dimick J.B. (2009). Variation in hospital mortality associated with inpatient surgery. *New England Journal of Medicine* **361**(14):1368–1375.

[16] Hajjem, A., Bellavance, F. and Larocque, D. (2011). Mixed effects regression trees for clustered data. *Statistics & Probability Letters* **81**(4):451–459.

[17] Hajjem, A., Bellavance, F. and Larocque, D. (2014). Mixed Effects Random Forest for Clustered Data. *Journal of Statistical Computation and Simulation* **84**, 1313–1328.
[18] Hajjem, A., Larocque, D. and Bellavance, F. (2017). Generalized mixed effects regression trees. *Statistics & Probability Letters* **126**:114–118.
[19] Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning*, Vol. 2. Springer.
[20] Haymart, M.R., Banerjee, M., Stewart, A.K., Koenig, R.J., Birkmeyer, J.D. and Griggs, J.J. (2011). Use of radioactive iodine for thyroid cancer. *JAMA* **306**(7):721–728.
[21] Hollenbeck, B.K., Taub, D.A., Miller, D.C., Dunn, R.L. and Wei, J.T. (2006). National utilization trends of partial nephrectomy for renal cell carcinoma: a case of underutilization? *Urology* **67**(2):254–259.
[22] Hothorn, T., Hornik, K. and Zeileis, A. (2006). Unbiased Recursive Partitioning: A Conditional Inference Framework. *Journal of Computational and Graphical Statistics* **15**(3):651–674.
[23] Iezzoni, L.I. (2012). *Risk Adjustment for Measuring Healthcare Outcomes*. Health Administration Press.
[24] Jang, W. and Lim, J. (2009). A numerical study of PQL estimation biases in generalized linear mixed models under heterogeneity of random effects. *Communication in Statistics – Simulation and Computation* **38**:692–702.
[25] Lee, S.K. (2005). On generalized multivariate decision tree by using GEE. *Computational Statistics & Data Analysis* **49**(4):1105–1119.
[26] Loh, W.Y. (2002). Regression trees with unbiased variable selection and interaction detection. *Statistica Sinica* **12**:361–386.
[27] Loh, W.Y. and Zheng, W. (2013). Regression Trees for Longitudinal and Multiresponse Data. *Annals of Applied Statistics* **7**:495–522.
[28] Miller, D.C., Saigal, C.S., Banerjee, M., Hanley, J. and Litwin, M.S. (2008). Diffusion of surgical innovation among patients with kidney cancer. *Cancer* **112**(8):1708–1717.
[29] Morgan, J.N. and Sonquist, J.A. (1963). Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association* **58**(302):415–434.
[30] Segal, M.R. (1992). Tree-structured methods for longitudinal data. *Journal of the American Statistical Association* **87**(418):407–418.
[31] Segal, M.R., Barbour, J.D. and Grant, R.M. (2004). Relating hiv-1 sequence variation to replication capacity via trees and forests. *Statistical Applications in Genetics and Molecular Biology* **3**(1).
[32] Sela, R.J. and Simonoff, J.S. (2012). Re-em trees: a data mining approach for longitudinal and clustered data. *Machine Learning* **86**(2):169–207.
[33] Skrondal, A. and Rabe-Hesketh, S. (2009). Prediction in multilevel generalized linear models. *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **172**(3):659–687.
[34] Speiser, J., Wolf, B., Chung, D., Karvellas, C., Koch, D. and Durkalski, V. (2018). BiMM tree: A decision tree method for modeling clustered and longitudinal binary outcomes. *Communications in Statistics – Simulation and Computation* **49**:1–20.
[35] Speiser, J., Wolf, B., Chung, D., Karvellas, C., Koch, D. and Durkalski, V. (2019). BiMM forest: A random forest method for modeling clustered and longitudinal binary outcomes. *Chemometrics and Intelligent Laboratory Systems* **185**:122–134.
[36] Therneau, T.M. and Atkinson, E.J. (1997). An introduction to recursive partitioning using the rpart routines. Technical Report **61**, Department of Health Science Research, Mayo Clinic, Rochester.
[37] Therneau, T.M., Grambsch, P.M. and Fleming T.R. (1990). Martingale-based residuals for survival models. *Biometrika* **77**(1):147–160.
[38] Zeileis, A., Hothorn, T. and Hornik, K. (2008). Model-Based Recursive Partitioning. *Journal of Computational and Graphical Statistics* **17**(2):492.
[39] Zhang, H. (1998). Classification trees for multiple binary responses. *Journal of the American Statistical Association* **93**(441):180–193.
[40] Zhang, H. and Singer, B.H (2013). *Recursive Partitioning in the Health Sciences*. Springer Science & Business Media.

Rong Xia
Department of Biostatistics
University of Michigan
Ann Arbor, Michigan
USA
E-mail address: rongxia@umich.edu

Christopher R. Friese
Department of Systems, Populations and Leadership
University of Michigan
Ann Arbor, Michigan
USA
E-mail address: cfriese@umich.edu

Mousumi Banerjee
Department of Biostatistics
University of Michigan
Ann Arbor, Michigan
USA
E-mail address: mousumib@umich.edu