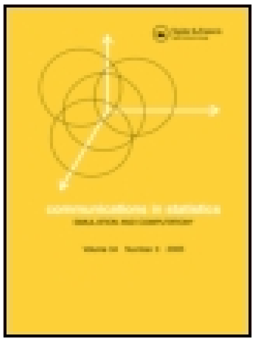

Communications in Statistics - Simulation and Computation

ISSN: 0361-0918 (Print) 1532-4141 (Online) Journal homepage: http://www.tandfonline.com/loi/lssp20

BiMM tree: a decision tree method for modeling clustered and longitudinal binary outcomes

Jaime Lynn Speiser, Bethany J. Wolf, Dongjun Chung, Constantine J. Karvellas, David G. Koch & Valerie L. Durkalski

To cite this article: Jaime Lynn Speiser, Bethany J. Wolf, Dongjun Chung, Constantine J. Karvellas, David G. Koch & Valerie L. Durkalski (2018): BiMM tree: a decision tree method for modeling clustered and longitudinal binary outcomes, Communications in Statistics - Simulation and Computation, DOI: 10.1080/03610918.2018.1490429

To link to this article: https://doi.org/10.1080/03610918.2018.1490429

Published online: 12 Sep 2018.

COMMUNICATIONS IN STATISTICS—SIMULATION AND COMPUTATION®
https://doi.org/10.1080/03610918.2018.1490429

Taylor & Francis
Taylor & Francis Group

BiMM tree: a decision tree method for modeling clustered and longitudinal binary outcomes

Jaime Lynn Speiser[a], Bethany J. Wolf[b], Dongjun Chung[b], Constantine J. Karvellas[c], David G. Koch[d], and Valerie L. Durkalski[b]

[a]Department of Biostatistical Sciences, Wake Forest School of Medicine, Winston-Salem, North Carolina, USA; [b]Department of Public Health Sciences, Medical University of South Carolina, Charleston, South Carolina, USA; [c]Divisions of Hepatology and Critical Care Medicine, University of Alberta, Edmonton, Canada; [d]Division of Gastroenterology and Hepatology, Department of Medicine, Medical University of South Carolina, Charleston, South Carolina, USA

ABSTRACT
Clustered binary outcomes are frequently encountered in clinical research (e.g. longitudinal studies). Generalized linear mixed models (GLMMs) for clustered endpoints have challenges for some scenarios (e.g. data with multi-way interactions and nonlinear predictors unknown *a priori*). We develop an alternative, data-driven method called Binary Mixed Model (BiMM) tree, which combines decision tree and GLMM within a unified framework. Simulation studies show that BiMM tree achieves slightly higher or similar accuracy compared to standard methods. The method is applied to a real dataset from the Acute Liver Failure Study Group.

ARTICLE HISTORY
Received 18 January 2018
Accepted 13 June 2018

KEYWORDS
Classification and regression tree; clustered data; decision tree; longitudinal data; mixed effects

MATHEMATICS SUBJECT CLASSIFICATION
62-04 Statistics - Machine Computation

1. Introduction

Clustered binary outcomes are frequently encountered in clinical research. Correlation within datasets may result from variables representing subject clusters, such as medical centers or family groups. Another type of clustered outcome results from longitudinal or repeated measures studies, where each patient represents a cluster. For example, a longitudinal study may collect repeated measurements of outcomes to evaluate disease prognosis (e.g. poor versus good outcome), diagnosis or disease relapse (e.g. disease versus disease-free), or other endpoints (e.g. re-admitted or not re-admitted to the hospital). Outcomes collected on the same patient at multiple time points are almost always dependent on one another. This within-subject correlation should be considered because failing to account for correlation may result in a loss of prediction efficiency.

Generalized linear mixed models (GLMMs) are typically employed for modeling clustered and longitudinal outcomes, but suffer limitations for some datasets. Interactions between predictor variables should be selected *a priori* to be included in GLMM modeling. However, knowledge about interactions between predictors is often lacking in

CONTACT Jaime Lynn Speiser ✉ jspeiser@wakehealth.edu Department of Biostatistical Sciences, Wake Forest School of Medicine, Winston-Salem, North Carolina 27157, USA

Supplemental data for this article can be accessed on the publisher's website.

practice, especially in complex clinical settings considering many personal, familial, and environmental factors. The GLMM framework also requires users to specify if there is a nonlinear relationship between predictors and outcome through the link function. Though specification of nonlinear relationships and interaction terms is not impossible, it often presents a challenge in the GLMM framework since there is not a universal method for making these decisions about modeling.

In this paper, we propose an alternative method that may provide greater flexibility for complex datasets called Binary Mixed Model (BiMM) tree, which combines decision tree methodology with mixed models. Decision tree methodology can be implemented to develop prediction models without the assumption of a linear relationship between predictor variables and outcome. Interactions between predictor variables are also naturally modeled within the decision tree framework without prior knowledge. In the BiMM tree method, we incorporate results from decision trees within mixed models to adjust for clustered and longitudinal outcomes. A Bayesian implementation of GLMM is used to avoid issues with convergence and quasi- or completely separated datasets with binary outcomes.

A specific motivating example dataset for the novel methodology in this paper is a longitudinal registry dataset of acute liver failure (ALF) patients (clinicaltrials.gov ID: NCT00518440). ALF is a rare and devastating condition characterized by rapid onset of severe liver damage, encephalopathy (altered mental status) and coagulopathy (impaired blood clotting), with approximately 25% of patients requiring a liver transplant and approximately 30% of patients dying during the acute phase (Lee et al. 2008). Complexities of the ALF registry data, including skewed distributions of predictors with many extreme values, nonlinear predictors of outcome, and a multitude of possible interactions among predictors, make it difficult to employ GLMMs for predicting outcomes.

The paper is structured as follows. In Section 2, we present background information about decision tree modeling in general and tree models for longitudinal and clustered continuous outcomes. In Section 3, we introduce the BiMM tree method for predicting longitudinal and clustered binary outcomes and in Section 4 we describe the motivating ALF registry in detail. We compare the BiMM tree method performance to several other methods with a simulation study in Section 5 and 6. An application to data from the ALF study is then presented as an example for the BiMM tree method in Section 7. Finally, in Section 8 we discuss implications of our study, limitations, and avenues for further research.

2. Background

A decision tree framework is utilized for the novel BiMM tree method because it offers several potential advantages compared to traditional models such as GLMMs. There are many different decision tree methods available, and we implement our BiMM tree method with the classification and regression tree (CART) framework, a commonly used methodology developed by Breiman et al. (1984). CART does not require specification of non-linear relationships or interaction terms, and offers simple and intuitive interpretation of predictor variables. Moreover, CART provides an alternative method for developing prediction models when traditional models are not feasible (e.g. if the number of predictor variables is greater than the number of observations). For these reasons, CART can sometimes better predict outcomes compared to other procedures such as discriminant analysis and logistic regression for data captured at a single time point (Hastie, Tibshirani, and Friedman 2001).

In spite of this flexibility, few decision tree methods exist for modeling clustered categorical endpoints. The R package *party* can be used to implement CART models if two predictor variables are correlated, but it does not adjust for longitudinal and clustered measurements of the same outcome variable (Hothorn, Hornik, and Zeileis 2011). There are some techniques which circumvent the issue of adjusting for longitudinal and clustered outcomes, such as summarizing variables (e.g. using averages or most frequent categorical values) or using data from only a single time point (e.g. admission values); however, these methods have a marked loss of information since available data is summarized or partially used.

Several methods have been proposed to modify CART models for longitudinal and clustered continuous outcomes (Segal 1992; Yu and Lambert 1999; Abdolell et al. 2002; De'Ath 2002; Keon Lee 2005; Dine, Larocque, and Bellavance 2009; Hajjem, Bellavance, and Larocque 2011, 2014; Sela and Simonoff 2012; Loh and Zheng 2013). Hajjem (2011) and Sela and Simonoff (2012) develop similar methods for implementing CART models for longitudinal and clustered data with continuous outcomes. These methods incorporate mixed effects within the tree framework to account for the clustered structure within the data, using an algorithm analogous to expectation-maximization described by Wu and Zhang (2006). The main idea in the Sela RE-EM tree (2012) and Hajjem mixed effects regression tree (2011) algorithms is to dissociate the fixed and cluster-level components within the modeling framework. First, a CART with all predictors as fixed effects is fitted with the assumption that the random effects for the clusters are known. Next, a linear mixed model is fitted using the estimated fixed effects from the CART and the random cluster effects are estimated which account for correlation induced by clustered variables with the assumption that the fixed effects are known. Finally, the continuous outcome is updated based on the linear mixed model using an additive effect in which the estimated random cluster effect is added to the original continuous outcome. The algorithm continues to iterate between CART, linear mixed models, and updating the outcome in a framework similar to the expectation-maximization algorithm (Wu and Zhang 2006). The algorithms continue iteratively until convergence is satisfied, which is based on the change in the likelihood from the mixed model being less than a specified value.

While the framework for clustered CART modeling has been developed for continuous outcomes, adjusting the algorithm for clustered categorical outcomes is non-trivial. For continuous endpoints, the outcomes are updated based on random effects from the linear mixed model using an additive effect. For categorical outcomes, the optimal method for adjusting outcomes is unclear because a random effect cannot simply be added.

3. BiMM tree method

The BiMM tree method iterates between developing CART models using all predictors and then using information from the CART model within a Bayesian GLMM to adjust for the clustered structure of the outcome. Consistent with the continuous methods for clustered decision trees, we implement an algorithm similar to the expectation-maximization algorithm, in which the fixed (decision tree) effects are dissociated from the random (cluster-level) effects. The BiMM tree method may be considered as an extension of GLMMs where the fixed covariates are not assumed to be linearly associated with the link function of the outcome and interactions do not need to be pre-specified. The traditional GLMM for binary outcomes has the form

$$\text{logit}(y_{it}) = \boldsymbol{X_{it}\beta} + Z_{it}b_{it},$$

where y_{it} is the binary outcome for cluster $i = 1, \ldots, M$ for longitudinal measurements $t = 1, \ldots, T_i$, logit() is the logistic link function, $\boldsymbol{X_{it}}$ is a matrix of fixed covariates for cluster i for longitudinal measurement t, $\boldsymbol{\beta}$ is a vector of fitted coefficients for the intercept and fixed covariates, Z_{it} is the clustered covariate for cluster i for longitudinal measurement t, and b_{it} is the fitted random effect for cluster i for longitudinal measurement t. Note that GLMMs may be fitted when the cluster sizes differ (e.g. if there are different numbers of longitudinal measurements for each cluster).

The GLMM portion of the BiMM method has the form

$$\text{logit}(y_{it}) = (1 \;\; \text{CART}(\boldsymbol{X_{it}}))\boldsymbol{\beta} + Z_{it}b_{it}.$$

CART($\boldsymbol{X_{it}}$) is a row vector represented within the GLMM as indicator variables reflecting membership of each longitudinal observation t for cluster i in terminal nodes in numerical order from the CART model. Terminal nodes are at the bottom of CART models and provide an outcome prediction for each subject's observation. Figure 1 provides an example CART model with terminal Nodes 1, 3, 5 and 6. Thus, the terminal nodes of CART provide a method for determining similar groups of observations which may be included within the Bayesian GLMM portion of the BiMM method. In this example, CART($\boldsymbol{X_{it}}$) would contain indicator variables for being in Node 1, Node 3, and Node 5, in that order. It is not necessary to include the indicator variable for the last terminal node, Node 6, because this would be redundant information within the GLMM. This is consistent with traditional models, where one includes one less indicator variables than the number of categories in the regression framework. Using Figure 1 as an example, CART($\boldsymbol{X_{it}}$) = (0 0 1) for an observation t for cluster i contained within Node 5. Thus, for this example,

$$\text{logit}(y_{it}) = (1 \;\; 0 \;\; 0 \;\; 1)\boldsymbol{\beta} + Z_{it}b_{it}.$$

This simplifies to

$$\text{logit}(y_{it}) = \beta_0 + \beta_3 + Z_{it}b_{it}.$$

Implementation of GLMMs is more challenging compared to standard linear mixed models employed for continuous outcomes. A consideration within the generalized model setting for categorical outcomes is that an iterative procedure (e.g. iterative reweighted least squares or Newton Raphson) must be used to compute random effects of clustered variables for GLMMs. GLMMs can have computational issues with model convergence or with inversion of large matrices, particularly for large datasets, which makes GLMM fitting challenging (Bates 2009). Also, if data are quasi-separated or completely separated, meaning that one or a combination of variables perfectly predicts the

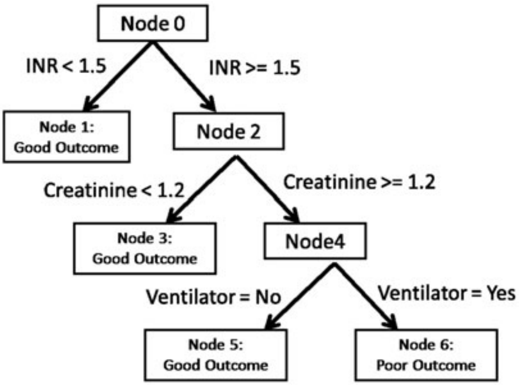

Figure 1. The decision tree data generating process for the simulation study is depicted in the figure. There are three variables: INR, creatinine, and ventilator use (yes/no). The tree contains four terminal nodes: 1, 3, and 5 represent good outcomes and 6 represents poor outcomes.

outcome, traditional implementations of GLMMs cannot be used (Zorn 2005; Gelman et al. 2008).

To address these challenges, we propose an algorithm that integrates CART and a Bayesian implementation of GLMM. There are several benefits to employing a Bayesian implementation of the GLMM instead of the traditional GLMM in our algorithm. First, Bayesian computation of GLMMs produce similar parameter estimates to that of frequentist GLMMs when uninformative prior distributions are used; however, weakly informative prior distributions can be used as a solution to separated or quasi-separated datasets (Gelman et al. 2008). Therefore, Bayesian implementation of the GLMM in the BiMM tree method offers more flexibility compared to frequentist GLMMs. Second, there are efficient methods for applying Bayesian GLMMs [e.g. integrated nested Laplace approximation implemented in the R package *INLA* (Fong, Rue, and Wakefield 2010) and maximum a posteriori estimation implemented in the R package *blme* (Dorie 2013, 2014)] easily applied on open source software which offer similar computation time to frequentist GLMMs. Finally, employing the Bayesian GLMM avoids convergence issues with traditional GLMMs [e.g. using the R package *lme4* (Bates 2009; Bates et al. 2015)].

The Bayesian GLMM within the BiMM tree method considers uninformative priors for the fixed effects and random effect covariance parameters using Normal and Wishart distributions respectively. An unstructured covariance matrix is employed within the Bayesian GLMM. After the random effects for subjects are fitted with the Bayesian GLMM, the original outcome variable is updated using results from the Bayesian GLMM, which we define as the target outcome variable. A split function which divides the observations into two groups is used to create a binary target outcome variable for each iteration since a simple additive effect does not result in a binary measure.

Specifically, the BiMM tree algorithm is as follows:

1. Initialize the CART and Bayesian GLMM:
   a. Fit a CART using y_{it} as the outcome for fixed predictors ($\boldsymbol{X}_{it}$) and develop J-1 indicator variables for the $j = 1, \ldots, J$ terminal nodes of clusters $i = 1, \ldots, M$

for longitudinal measurements $t = 1, \ldots, T_i$:

$$I(y_{it} \in node_j) = \begin{cases} 1 \text{ if } y_{it} \text{ is in terminal node } j \\ 0 \text{ otherwise} \end{cases}$$

Define $\mathrm{CART}(\boldsymbol{X_{it}})$ as the row vector of the J-1 indicator variables for cluster i at longitudinal measure t.

b. Fit a Bayesian GLMM using y_{it} as the outcome, including $CART(\boldsymbol{X_{it}})$ and clustered variable (Z_{it}) to obtain fitted values for the random effect (b_{it}):

$$\mathrm{logit}(y_{it}) = (1\ \mathrm{CART}(\boldsymbol{X_{it}}))\boldsymbol{\beta} + Z_{it}b_{it}.$$

c. Extract the predicted probabilities from Bayesian GLMM (denoted $\mathrm{pr}_{BGLMM}(\boldsymbol{X_{it}}, Z_{it})$) for each measurement t within cluster i:

$$q_{it} = \mathrm{pr}_{BGLMM}(\boldsymbol{X_{it}}, Z_{it})$$

2. Iterate through the following steps until convergence is satisfied:

a. Determine the target outcome (y^*_{it}) by adding the predicted probability (q_{it}) from the original outcome (y_{it}) and applying a split function h() to make y^*_{it} a binary value:

$$y^*_{it} = \mathrm{h}(y_{it} + q_{it})$$

b. Repeat steps 1a-c using y^*_{it} as the outcome until the change in the posterior log likelihood from the Bayesian GLMM is less than a specified tolerance value

Predictions for observations included within the model development dataset are made using the CART (population-level) and random (observation-level) components. For observations not included within the model development dataset, predictions are made using the CART (population-level) component only.

There are several different split functions (denoted $\mathrm{h}(y_{it} + q_{it})$) which may be used to create the new iteration of the binary target outcome (y^*_{it}) . We use a function of $y_{it} + q_{it}$ to update the target outcome to account for both the original outcome and the average predicted probability from the CART and Bayesian GLMM models for the specific observation t within the cluster i. Before introducing the split functions, it is necessary to understand the distribution of $y_{it} + q_{it}$. Since y_{it} is a binary value, it is either 0 or 1, and q_{it} is a probability which is between 0 and 1. Therefore, value of $y_{it} + q_{it}$ is between 0 and 2. We present three options for the split function which may be employed based on the overall goal of the prediction model. The first split function maximizes model sensitivity, the second split function maximizes model specificity, and the third split function equally weights sensitivity and specificity for updating the target outcome vector. Now, the split function which maximizes sensitivity uses a threshold $(0<k_1 < 1)$ to update the target outcome:

$$\mathrm{h}_1(y_{it} + q_{it}) = \begin{cases} 1 \text{ if } y_{it} + q_{it} > k_1 \\ 0 \text{ otherwise} \end{cases}$$

Thus, using $\mathrm{h}_1(y_{it}+q_{it})$, binary outcomes of 0 can be updated to be 1, but outcomes of 1 cannot be updated to be 0. This provides a mechanism for maximizing the sensitivity. Similarly, a split function which maximizes specificity may be employed using a threshold $(1 < k_2 < 2)$ to update the target outcome:

$$\mathrm{h}_2(y_{it}+q_{it}) = \begin{cases} 0 \text{ if } y_{it}+q_{it}<k_2 \\ 1 \text{ otherwise} \end{cases}$$

Using $\mathrm{h}_2(y_{it}+q_{it})$, binary outcomes of 1 can be updated to be 0, but outcomes of 0 cannot be updated to be 1. This provides a mechanism for maximizing the specificity. A final, more general, split function which does not favor sensitivity or specificity updates the target outcome using the following method:

$$\mathrm{h}_3(y_{it}+q_{it}) = \begin{cases} 0 \text{ if } y_{it}+q_{it}<0.5 \\ 1 \text{ if } y_{it}+q_{it}>1.5 \\ 1 \text{ with probability } q_{it} \\ 0 \text{ otherwise} \end{cases}$$

Using $\mathrm{h}_3(y_{it}+q_{it})$, if the prediction from the current iteration of the BiMM method agrees with the original binary outcome (i.e. if $y_{it}+q_{it} < 0.5$ or if $y_{it}+q_{it} > 1.5$) then the target outcome is the same as the original binary outcome. Otherwise, the target outcome is updated to be 1 with probability q_{it}, and 0 with probability $1-q_{it}$. Therefore, original values of 0 can be updated to 1 and original values of 1 can be updated to 0.

An example of the four possible scenarios of an iteration within the BiMM method is depicted within Table 1, with $k_1 = 0.5$ and $k_2 = 1.5$ for observation t within cluster i. Using the split function $\mathrm{h}_1(y_{it}+q_{it})$, the original binary outcome (y_{it}) changes from a 0 to a 1 in Scenario B, which will increase the sensitivity since the next iteration of the BiMM method will contain more values of 1 within the target outcome. Likewise, using the split function $\mathrm{h}_2(y_{it}+q_{it})$, the original binary outcome (y_{it}) changes from a 1 to a 0 in Scenario C, which will increase the specificity since the next iteration of the BiMM method will contain more values of 0 within the target outcome. Using $\mathrm{h}_3(y_{it}+q_{it})$, the target outcomes are updated in Scenarios B and C based on the strength of the predicted probability from the BiMM iteration. In all split functions, if the original binary outcome agrees with the predicted probability from the BiMM iteration (i.e. in Scenarios A and D), then the target outcome is the original outcome.

BiMM trees for this study are computed using R software version 3.1.2 (R Development Core Team 2008). CART models are implemented using the R package *rpart* (Therneau and Atkinson 1997). Default settings are used within the CART models, but we require that the minimum terminal node size be at least 10% of the development dataset so that node indicators within the Bayesian GLMM contain adequate data for fitting fixed effects. Bayesian GLMMs within the BiMM method are implemented using the R package *blme* (Dorie 2013, 2014), again with all default settings.

4. Data description

ALF occurs in approximately 2,000 patients in the United States each year, with about half of the cases attributed to acetaminophen overdose (Lee et al. 2008). A critical goal

Table 1. Example scenarios for split functions within the BiMM tree method.

Scenario	y_{it}	q_{it}	$y_{it} + q_{it}$	$y^*_{it} = \mathrm{h}_1(y_{it} + q_{it})$	$y^*_{it} = \mathrm{h}_2(y_{it} + q_{it})$	$y^*_{it} = \mathrm{h}_3(y_{it} + q_{it})$
A	0	$0 < q_{it} < 0.5$	$0 < y_{it} + q_{it} < 0.5$	0	0	0
B	0	$0.5 < q_{it} < 1$	$0.5 < y_{it} + q_{it} < 1$	1	0	1 with probability q_{it}, 0 otherwise
C	1	$0 < q_{it} < 0.5$	$1 < y_{it} + q_{it} < 1.5$	1	0	1 with probability q_{it}, 0 otherwise
D	1	$0.5 < q_{it} < 1$	$1.5 < y_{it} + q_{it} < 1$	1	1	1

of the ALF Study Group is to predict the likelihood of poor outcomes of acetaminophen-induced ALF patients which may be used both on hospital admission and post-hospital admission (Speiser, Lee, and Karvellas 2015). The ALF Study Group registry consists of over 2,700 patients with a multitude of clinical data (e.g. laboratory values, treatments, complications, etc.) collected daily for up to seven days following enrollment unless a patient is transplanted, discharged from the hospital or dies. To date, most prognosis prediction models for ALF patients use variables collected at a single baseline time point [e.g. King's College Criteria and Clichy Criteria (O'Grady et al. 1989; Bernuau 1990)]. Many patients may remain alive for longer periods beyond the initial insult because of advances in intensive care unit management (Stravitz et al. 2007; Antoniades et al. 2008). Thus, there is a need for a prediction model which may be used to determine prognosis of acetaminophen-induced ALF patients (poor or favorable outcome) each day which can aid clinicians in management of patients during the first week of hospitalization. We define a poor outcome as having coma grade of III or IV and favorable outcome as having a coma grade of 0, I or II.

The ALF registry dataset contains many clinical predictor variables which may be used in modeling outcome. A few fixed predictor variables included within the registry are gender, ethnicity, and age. Some examples of continuous predictor variables collected daily for the first week in the hospital include aspartate aminotransferase (AST), alanine aminotransferase (ALT), creatinine, bilirubin and international normalized ratio (INR). Categorical variables collected daily include treatments and clinical measurements such as mechanical ventilation, pressor use, and renal replacement therapy.

5. Simulation study design

To assess the predictive performance of the proposed BiMM tree method, we conduct a simulation study based on the real motivating dataset, the ALF Study Group registry. We simulate data from the ALF registry for several reasons. First, the complexity of the ALF dataset allows comparing of novel and traditional methodologies in realistic settings. Additionally, the ALF dataset contains multiple continuous predictors which are not normally distributed and several categorical variables, so our simulated ALF data provides various types of predictors which are consistent with data that arises from many real-world scenarios. A final reason we simulate data based on the real ALF dataset is that a correlation structure between repeated measures on the same person is not imposed, so we can evaluate the performance of proposed methodology with a real observed correlation structure within the ALF data.

We construct a dataset from which we sample simulation data by selecting all data from acetaminophen-induced ALF patients within the registry ($N = 1{,}064$) and imputing all missing predictor data using an imputation method (Mistler n.d.) for multilevel data to preserve the original correlation structure between predictor variables within the dataset. Thus, the simulated datasets contain 1,064 patients with complete data for seven days (three fixed predictors and eight longitudinal predictors). We use two data generating processes for the fixed portion of the outcome: a tree structure and a linear structure. For both processes, variables related to the outcome include INR, creatinine, and ventilator use, which is consistent with clinical literature (Speiser, Lee, and Karvellas

2015; Koch et al. 2016). The other five longitudinal variables and the three fixed predictors are included within the simulation datasets as noise variables. The tree data generating process is depicted within Figure 1, which is read like a CART (i.e. begin at Node 0 and follow the arrow corresponding to the predictor variable values until a terminal node is reached). Nodes 1, 3 and 5 represent favorable outcome for the subject on the specific day, whereas Node 6 represents poor outcome for the subject on the specific day. The equation for the linear data generating process is:

$$\text{logit}(\text{poor outcome}) = \; -2.3 + 1.4 * \ln(\text{INR}) + 0.6 * \text{Creatinine} + 2.1 * \text{I}(\text{Ventilator})$$

where $I(Ventilator)$ is 1 if the patient was on a ventilator on the specific day, and is 0 otherwise. Thus, high INR and creatinine and being on a ventilator are associated with higher likelihood of poor outcomes, consistent with clinical literature (Koch et al. 2016).

Small and large random effects are added to the fixed portion of the outcome to create a within-subject correlation structure. The small random effect is generated for each subject from a normal distribution centered at zero with standard deviation of 0.1, whereas the large random effect is generated for each subject from a normal distribution centered at zero with standard deviation of 0.5. To derive the outcome of observations at every time point, the fixed portion (from the tree or linear data generating process) is added to the random effect, and a cut point is used to create the binary outcome.

Using the simulated datasets described in the previous paragraphs based on the ALF registry, we compare the performance of several models. We use $\text{h}_1(y_{it} + q_{it})$ as our split function for updating the target outcome with a threshold (k_1) of 0.5 because clinicians often prefer to develop prediction models maximized for sensitivity to identify patients at highest risk of poor outcomes. All models are fit using all predictors in the data (i.e. both those associated with outcome and those that were noise variables). We produce models for BiMM trees with one iteration (denoted BiMM tree 1) and with multiple iterations (denoted BiMM tree H1 and BiMM tree H3 for the respective split functions $\text{h}_1(y_{it} + q_{it})$ and $\text{h}_3(y_{it} + q_{it})$) to assess if iterating between fixed and random effects results in increased prediction accuracy. Models are compiled for 1000 simulation runs. Sample sizes (number of subjects) for training datasets used in model development are 100, 250 and 500. All test datasets consist of 500 new subjects not included within the training dataset. The numbers of repeated measurements of outcomes in our simulation study are 2, 4 and 7. Since the main objective in this study is to develop methodology for predicting new observations, we assess the prediction (test set) accuracy of the models, defined as the number of correct predictions divided by the total number of predictions made.

6. Simulation study results

Prediction accuracy is presented within Figure 2 for the sample size of 100. Overall, the BiMM trees with one iteration or more than one iteration have higher accuracy compared to CART and Bayesian GLMM when the random effect is large, regardless of whether the data are generated using a tree or linear structure. When the random effect is small, the accuracy distributions overlap, with the CART models generally having

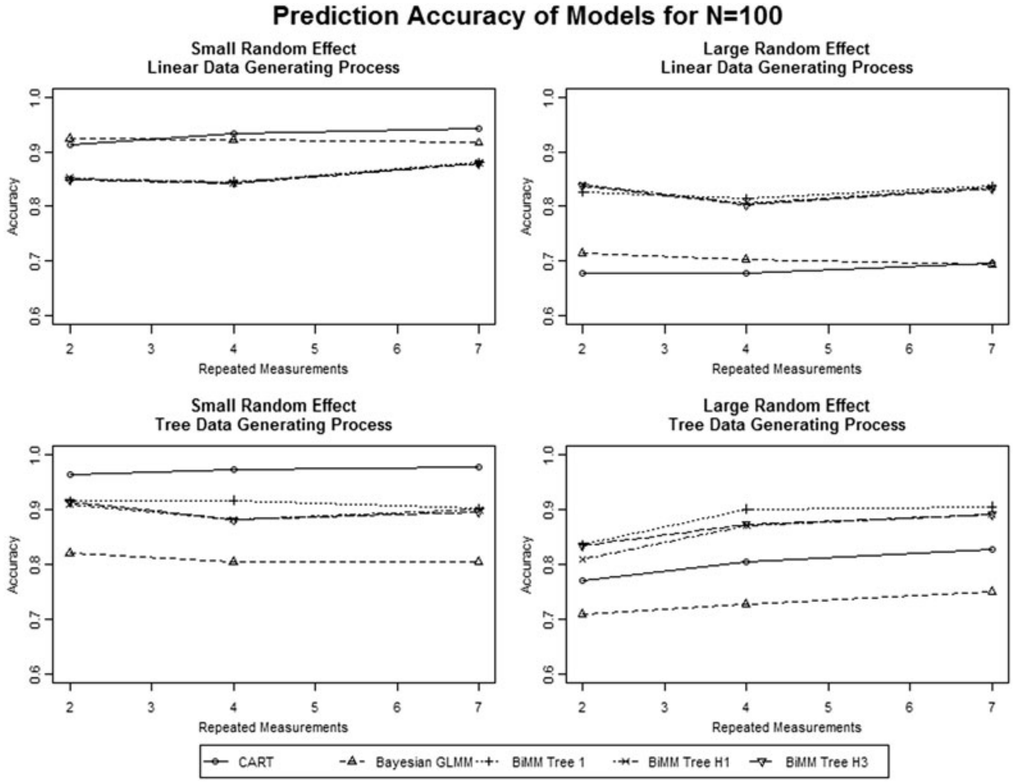

Figure 2. The simulated prediction (test set) accuracy of models for $N = 100$ patients are displayed within the figure for small and large random effects, for linear and tree data generating processes, and for 2, 4 and 7 repeated measurements per patient. Traditional CART, Bayesian GLMM, BiMM tree with one iteration (denoted BiMM tree 1) and BiMM Tree with multiple iterations for the split function maximizing sensitivity and the general split function (denoted BiMM Tree H1 and BiMM Tree H3) are compared.

slightly higher accuracy compared to the BiMM trees. With a linear data generating process and small random effect, the CART and Bayesian GLMM have similar predictive accuracy, whereas with a tree data generating process and small random effect, the Bayesian GLMM has the lowest prediction accuracy. The Bayesian GLMM also has the lowest prediction accuracy for the tree data generating process with a large random effect. The BiMM tree models with one iteration and with multiple iterations generally have similar predictive accuracies for each of the scenarios. Similar results were obtained for the sample sizes of 250 and 500 (Supplementary Figures 1 and 2).

Most BiMM iterative trees converge in two iterations regardless of the split function, and in rare cases convergence is reached in three or four iterations. Table 2 contains the median (interquartile range) estimates of prediction accuracy of the test dataset for each simulation scenario for CART, Bayesian GLMM, BiMM tree with one iteration, and BiMM tree algorithm with more than one iteration. Interquartile ranges of prediction accuracy for the models in the different scenarios are relatively tight around the median estimates, indicating that the distribution of prediction accuracy for models does not vary greatly over the simulation runs. Across 2, 4 and 7 repeated

Table 2. Median prediction (test set) accuracy (interquartile range) for simulated datasets.

		$N = 100$				$N = 250$				$N = 500$			
		Linear DGP		Tree DGP		Linear DGP		Tree DGP		Linear DGP		Tree DGP	
Model	Repeated outcomes	Small RE	Large RE	Small RE	Large RE	Small RE	Large RE	Small RE	Large RE	Small RE	Large RE	Small RE	Large RE
CART	2	0.913	0.678	0.963	0.769	0.929	0.706	0.970	0.796	0.942	0.732	0.971	0.821
		(0.890,0.925)	(0.651.0.699)	(0.950,0.971)	(0.745,0.791)	(0.920,0.937)	(0.690,0.722)	(0.965,0.975)	(0.780,0.812)	(0.936,0.948)	(0.718,0.745)	(0.965,0.975)	(0.806,0.832)
	4	0.933	0.679	0.973	0.805	0.942	0.721	0.975	0.830	0.945	0.735	0.975	0.844
		(0.922,0.940)	(0.659,0.699)	(0.966,0.978)	(0.788,0.821)	(0.937,0.947)	(0.707,0.733)	(0.970,0.979)	(0.818,0.843)	(0.941,0.950)	(0.724,0.745)	(0.971,0.980)	(0.833,0.853)
	7	0.941	0.697	0.977	0.828	0.946	0.732	0.979	0.853	0.947	0.740	0.979	0.859
		(0.935,0.947)	(0.680,0.714)	(0.972,0.981)	(0.814,0.842)	(0.942,0.950)	(0.720,0.743)	(0.975,0.982)	(0.840,0.863)	(0.944,0.951)	(0.729,0.750)	(0.975,0.982)	(0.849,0.867)
Bayesian GLMM	2	0.924	0.715	0.820	0.709	0.939	0.731	0.836	0.722	0.943	0.737	0.842	0.725
		(0.914,0.933)	(0.700,0.730)	(0.805,0.835)	(0.691,0.723)	(0.933,0.944)	(0.718,0.745)	(0.826,0.847)	(0.708,0.733)	(0.938,0.948)	(0.723,0.748)	(0.833,0.851)	(0.714,0.737)
	4	0.922	0.704	0.805	0.728	0.930,	0.717	0.812	0.735	0.932	0.721	0.814	0.736
		(0.915,0.928)	(0.689,0.716)	(0.795,0.814)	(0.717,0.738)	(0.926,0.934)	(0.704,0.729)	(0.804,0.820)	(0.724,0.744)	(0.928,0.936)	(0.710,0.732)	(0.808,0.821)	(0.727,0.746)
	7	0.917	0.695	0.805	0.750	0.921	0.707	0.808	0.755	0.923	0.710	0.809	0.756
		(0.912,0.921)	(0.682,0.707)	(0.796,0.812)	(0.741,0.760)	(0.917,0.925)	(0.696,0.717)	(0.801,0.814)	(0.746,0.763)	(0.919,0.926)	(0.699,0.721)	(0.803,0.814)	(0.748,0.765)
BiMM tree	2	0.849	0.827	0.916	0.836	0.850	0.850	0.921	0.911	0.850	0.856	0.920	0.917
1 iteration		(0.833,0.868)	(0.776,0.852)	(0.897,0.927)	(0.782,0.902)	(0.838,0.866)	(0.830,0.873)	(0.914,0.929)	(0.880,0.923)	(0.840,0.863)	(0.840,0.886)	(0.915,0.926)	(0.907,0.925)
	4	0.845	0.815	0.917	0.901	0.850	0.837	0.942	0.947	0.849	0.841	0.952	0.953
		(0.822,0.860)	(0.780,0.849)	(0.881,0.952)	(0.854,0.956)	(0.835,0.862)	(0.807,0.862)	(0.888,0.959)	(0.894,0.964)	(0.838,0.862)	(0.813,0.865)	(0.891,0.960)	(0.900,0.963)
	7	0.881	0.837	0.902	0.905	0.887	0.860	0.905	0.907	0.890	0.867	0.905	0.906
		(0.869,0.891)	(0.806,0.864)	(0.892,0.913)	(0.862,0.920)	(0.878,0.894)	(0.837,0.889)	(0.895,0.914)	(0.864,0.914)	(0.884,0.895)	(0.843,0.890)	(0.897,0.913)	(0.858,0.913)
BiMM tree	2	0.852	0.840	0.910	0.809	0.856	0.847	0.918	0.858	0.859	0.851	0.918	0.852
H1 algorithm		(0.832,0.876)	(0.813,0.854)	(0.842,0.924)	(0.757,0.870)	(0.842,0.874)	(0.836,0.861)	(0.904,0.925)	(0.819,0.916)	(0.845,0.874)	(0.841,0.866)	(0.909,0.924)	(0.822,0.917)
	4	0.844	0.806	0.882	0.871	0.847	0.820	0.887	0.882	0.847	0.816	0.886	0.878
		(0.823,0.857)	(0.790,0.840)	(0.868,0.925)	(0.817,0.934)	(0.830,0.860)	(0.800,0.850)	(0.873,0.947)	(0.855,0.952)	(0.832,0.860)	(0.796,0.852)	(0.874,0.901)	(0.851,0.904)
	7	0.879	0.835	0.899	0.891	0.887	0.842	0.905	0.897	0.890	0.843	0.905	0.896
		(0.862,0.891)	(0.814,0.847)	(0.890,0.911)	(0.801,0.910)	(0.876,0.894)	(0.832,0.852)	(0.894,0.913)	(0.853,0.909)	(0.884,0.895)	(0.836,0.851)	(0.896,0.912)	(0.725,0.907)
BiMM tree	2	0.849	0.838	0.913	0.834	0.850	0.846	0.920	0.908	0.850	0.848	0.920	0.915
H3 algorithm		(0.834,0.867)	(0.804,0.852)	(0.850,0.924)	(0.775,0.909)	(0.839,0.865)	(0.835,0.857)	(0.912,0.928)	(0.830,0.923)	(0.840,0.863)	(0.838,0.859)	(0.914,0.926)	(0.852,0.923)
	4	0.843	0.803	0.881	0.874	0.848	0.807	0.881	0.879	0.848	0.805	0.879	0.877
		(0.822,0.857)	(0.788,0.833)	(0.868,0.938)	(0.848,0.943)	(0.832,0.861)	(0.793,0.842)	(0.871,0.949)	(0.860,0.953)	(0.836,0.861)	(0.793,0.839)	(0.869,0.891)	(0.862,0.894)
	7	0.879	0.833	0.895	0.891	0.886	0.840	0.899	0.895	0.890	0.842	0.890	0.895
		(0.852,0.891)	(0.805,0.845)	(0.887,0.905)	(0.805,0.909)	(0.877,0.894)	(0.831,0.849)	(0.890,0.909)	(0.785,0.907)	(0.884,0.895)	(0.835,0.849)	(0.891,0.910)	(0.776,0.904)

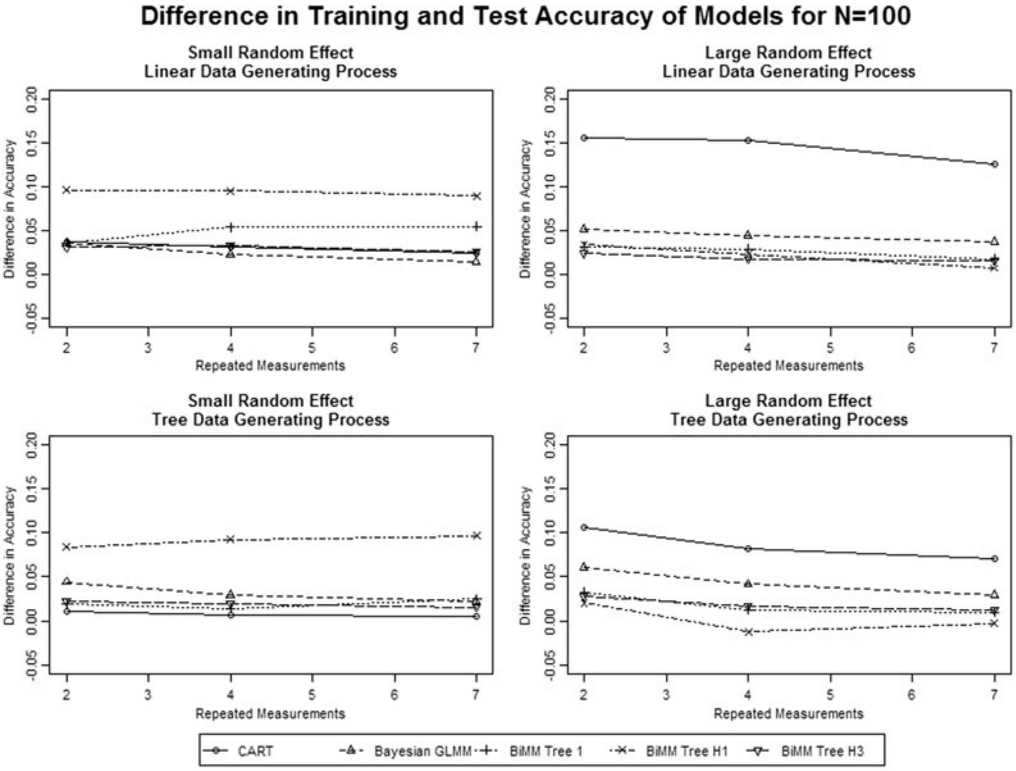

Figure 3. The simulated difference in training and test set accuracy of models for $N = 100$ patients are displayed within the figure for small and large random effects, for linear and tree data generating processes, and for 2, 4 and 7 repeated measurements per patient. Traditional CART, Bayesian GLMM, BiMM tree with one iteration (denoted BiMM tree 1) and BiMM tree with multiple iterations for the split function maximizing sensitivity and the general split function (denoted BiMM tree H1 and BiMM tree H3) are compared.

measurements for the models and scenarios, prediction accuracy is similar, except for the tree data generating process with a large random effect, where slight gains in accuracy are achieved with increasing number of repeated measurements. In general, the prediction accuracy estimates are similar for sample sizes of 100, 250 and 500, with slight improvements in accuracy for BiMM models with larger sample sizes.

In addition to assessing the predictive accuracy of models, we present the difference between training and test accuracy for models in the simulated scenarios to measure the amount of overfitting in models for sample size of 100 (Figure 3). Within this plot, large values of the difference between the training and test datasets indicate that the accuracy of the training dataset is larger than the accuracy of the test dataset. For small random effects, CARTs, Bayesian GLMMs, BiMM trees with one iteration, and BiMM trees updated with $\mathrm{h}_3(y_{it} + q_{it})$ have minimal overfitting, since the difference between training and test set accuracy is small. However, for small random effects, BiMM trees with multiple iterations updated with $\mathrm{h}_1(y_{it} + q_{it})$ have larger differences in accuracy, suggesting the models may have overfit the training data. When random effects are large, the CART models tend to overfit the training data the most for both data generating processes. For the tree data generating process with a large random effect, the

Bayesian GLMM overfits the data more than the BiMM trees, but this is only a slight difference. Regardless of the data generating process, the BiMM trees overfit the training data the least. As the number of repeated measurements increase, model overfitting slightly decreases for large random effect datasets, whereas model overfitting remains similar for datasets with small random effects. The performance of each model in terms of overfitting is similar for the sample sizes of 250 and 500; however, the amount of overfitting is slightly less with the larger sample sizes for datasets with large random effects (Supplementary Figures 3 and 4).

7. Data application

To illustrate the use of the BiMM tree method, we develop prediction models for acetaminophen-induced ALF patients enrolled in the ALF registry from January 1998 to February 2016 ($N = 1{,}082$). The primary endpoint is poor versus favorable outcome, defined by a daily measurement of coma grade. High coma grade (III or IV) represents poor outcome on a particular day for a patient, whereas low coma grade (0, I or II) represents favorable outcome on a particular day for a patient. We define sensitivity as the proportion of correct predictions for patients within the poor outcome group and specificity as the proportion of correct predictions for patients within the favorable outcome group.

We include three fixed predictors as possible variables for the models: sex, ethnicity and age. Several continuous laboratory predictors which are repeatedly collected for up to seven days when patients are in the hospital are included as well: ALT, AST, bilirubin, creatinine, phosphate, lactate, pH, platelets, ammonia and INR. Many categorical (binary for yes/no) variables collected daily about treatments and complications experienced by patients are considered as possible predictors. Some patients have missing predictor data, and we use the default CART method of surrogate splits to handle this within the BiMM framework. Surrogate splitting uses non-missing predictor variables for patients who have missing variables to run down the CART model so that predictions can be made regardless of missing values (Breiman et al. 1984). We randomly split the dataset into a training ($N = 541$ subjects) and test ($N = 541$ subjects) dataset so that we can assess the predictive accuracy of the BiMM tree. Patients within both the training and test datasets have on average four days of data collected.

The BiMM tree with multiple iterations using $\mathrm{h}_1(y_{it} + q_{it})$ as the split function for predicting poor versus favorable outcomes for acetaminophen-ALF patients is illustrated in Figure 4. We use this split function because it is desirable to develop a tree prediction model which maximizes sensitivity. The model includes four clinical predictors of outcome: pressor use (binary, yes/no), bilirubin, creatinine, and pH. The BiMM tree with one iteration contains these variables in addition to sex. Patients requiring pressors or with high values of bilirubin, creatinine and pH are associated with higher likelihoods of poor daily outcomes.

BiMM tree performance statistics and 95% binomial confidence intervals are presented within Table 3. Exact binomial confidence intervals are calculated using the base R function *binom.test()*. BiMM tree models with one iteration and with split function $\mathrm{h}_1(y_{it} + q_{it})$ have similar training dataset accuracies of 0.9, sensitivities of 1, and

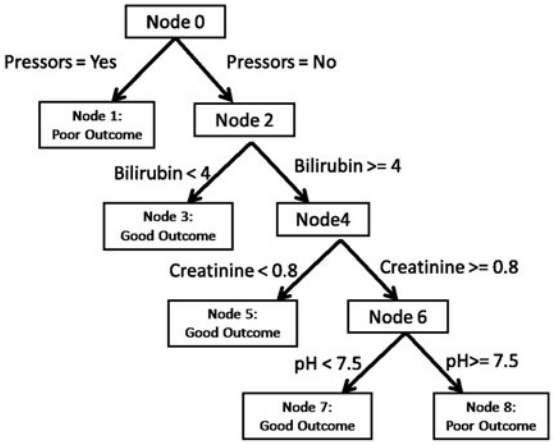

Figure 4. The BiMM tree algorithm with multiple iterations using the split function that maximizes sensitivity to predict daily prognosis of ALF patients is represented within the figure. The decision tree uses four clinical predictors of outcome and contains five terminal nodes: 3, 5 and 7 represent good outcomes and 1 and 8 represent poor outcomes.

specificities of 0.8, which indicates overall good model predictions for the training dataset. However, the accuracies for the test dataset for the all BiMM trees are lower (between 0.63 and 0.65). The test dataset sensitivity for the BiMM tree with one iteration of 0.67 is lower compared to the sensitivity for the BiMM tree with multiple iterations with $\mathrm{h}_1(y_{it} + q_{it})$ (0.81), indicating that the splitting function maximizes sensitivity as expected. The BiMM tree with one iteration has a slightly higher specificity compared to the BiMM tree with multiple iterations with $\mathrm{h}_1(y_{it} + q_{it})$ for the test dataset. The posterior log likelihood of the BiMM tree model with one iteration is larger than that of the BiMM tree models with multiple iterations. This indicates that the BiMM tree produced by multiple iterations provides better fit compared to the tree with one iteration.

A frequentist GLMM is implemented using the *lme4* R package, which produces error messages and warnings for the ALF data application; therefore, its results are omitted. Because the Bayesian GLMM method requires complete datasets with no predictor values missing, a substantial amount of observations could not be included for model development (training data) or model predictions (test data). When all variables are included in the GLMM, only 78 (3.5%) observations in the training dataset and 101 (4.6%) observations in the test dataset could be used (i.e. had no missing values). Overall, methods which included a clustering variable (i.e. Bayesian GLMM and BiMM trees) have higher training dataset accuracy compared to CART. Test dataset accuracy is similar for all methods except for the Bayesian GLMM, which had substantially lower prediction accuracy. CART and BiMM tree with one iteration offer similar values of sensitivity and specificity, whereas the BiMM tree algorithm with multiple iterations with $\mathrm{h}_1(y_{it} + q_{it})$ has slightly higher sensitivity compared to specificity, as desired.

Also included in Table 3 are the computational run times for developing the models using the training dataset. In general, computation time will vary based on the size of the dataset and the number of clusters. For the ALF data application, the CART has the quickest run time, followed by the BiMM tree methods (around 1–3 s). The Bayesian GLMM has the highest run time, around 10 s.

Table 3. Data application performance and run time statistics for ALF prediction models with 95% confidence intervals.

Method	Training data					Test data			
	# Obs (M)	Run time (s)	Accuracy	Sensitivity	Specificity	# Obs (M)	Accuracy	Sensitivity	Specificity
CART	2253	0.109	0.702 (0.683,0.721)	0.683 (0.656,0.710)	0.723 (0.695,0.749)	2208	0.639 (0.619,0.660)	0.613 (0.584,0.641)	0.669 (0.640,0.698)
Bayesian GLMM	78	10.057	0.859 (0.762,0.927)	0.881 (0.744,0.960)	0.833 (0.672,0.936)	101	0.554 (0.452,0.653)	0.629 (0.449,0.785)	0.515 (0.389,0.640)
BiMM tree 1 Iteration	2253	1.338	0.893 (0.880,0.905)	1.000 (0.996,1.000)	0.818 (0.797,0.839)	2208	0.645 (0.625,0.665)	0.667 (0.638,0.694)	0.623 (0.593,0.652)
BiMM tree H1 algorithm	2253	1.361	0.885 (0.871,0.898)	1.000 (0.996,1.000)	0.807 (0.785,0.828)	2208	0.649 (0.629,0.669)	0.812 (0.779,0.842)	0.585 (0.560,0.609)
BiMM tree H3 algorithm	2253	2.711	0.666 (0.646,0.686)	0.684 (0.656,0.712)	0.650 (0.621,0.677)	2208	0.627 (0.607,0.647)	0.669 (0.640,0.698)	0.590 (0.561,0.618)

8. Discussion

Overall, the BiMM tree framework may offer advantages compared to CARTs and Bayesian GLMMs. The main benefit of BiMM compared to CART is that it can account for clustered outcomes in modeling so that the assumption of independent observations is not violated. BiMM trees do not require specification of nonlinear relationships or interaction terms and can be implemented for high dimensional datasets. A strength of BiMM tree is that nonlinear forms of predictors and interactions between predictors are developed by the method based on the data. The computation time of GLMMs may be higher compared to BiMM trees, yet BiMM trees may potentially offer higher prediction accuracy for certain situations: for example, when the underlying structure of the data contains interactions between predictors and nonlinear predictors of outcome or when there is a large clustered effect for the outcomes. A final potential strength of BiMM tree methodology is that missing values in predictor data is naturally handled using surrogate splits (using other non-missing variables) within the CART portion of the algorithm; thus, observations with missing predictor data can still be included within BiMM models. GLMMs only use complete cases within datasets, so missing values would need to be imputed (filled in) within the GLMM setting in order to use the entire dataset. The BiMM tree method does not require missing data to be imputed prior to model development.

A distinction of the BiMM tree framework compared to other decision tree methods for longitudinal and clustered outcomes within the literature (Segal 1992; Yu and Lambert 1999; Abdolell et al. 2002; De'Ath 2002; Keon Lee 2005; Dine, Larocque, and Bellavance 2009; Hajjem, Bellavance, and Larocque 2011, 2014; Sela and Simonoff 2012; Loh and Zheng 2013) is the Bayesian implementation of GLMMs. For continuous outcomes, there are fewer issues with GLMM convergence because estimates may be computed directly; however, with categorical outcomes complete or quasi-separation may pose a challenge to GLMM fitting. The default priors specified in the BiMM tree method are uninformative, but if convergence issues arise, weakly informative priors may be used for estimating the random effects (Gelman et al. 2008).

BiMM tree provides a flexible, data-driven predictive modeling framework for longitudinal and clustered binary outcomes. Our simulation study demonstrates that BiMM tree may potentially be advantageous compared to CART which ignores clustered outcomes and Bayesian GLMM when predictors are not linearly related to the outcome through the link function and when the random effect of the clustered variable is large. Though standard CART models can have high predictive accuracy if random effects are small, failing to account for large clustering effects causes a sizeable decrease in prediction accuracy in our simulations. While Bayesian GLMM can be used to adjust for clustering within the data, model misspecification may reduce prediction accuracy (e.g. not including a significant interaction term or specifying an incorrect nonlinear relationship between predictor and outcome). This is evident in our simulation study, where BiMM tree models have higher prediction accuracy compared to Bayesian GLMMs if the data has a tree structure or if there is a large clustering effect between outcomes. One possible reason that the Bayesian GLMMs did not perform well for simulated data in this study is that some of the continuous predictor variables have skewed distributions, and extreme values may have adversely affected parameter estimates.

The BiMM trees with one iteration generally have similar prediction accuracy compared to BiMM trees with more than one iteration within our simulation study. While the training dataset accuracies for the BiMM trees with more than one iteration are higher than the BiMM trees with only one iteration, the multiple iteration method with the split function which maximizes sensitivity produces overfitted models which do not predict well for test datasets if the effect of clustering within subjects is small. BiMM trees which iterate between fixed and random effects have slightly higher computation time and offer minimal increases in prediction accuracy, suggesting that BiMM trees with one iteration may be sufficient. It is possible that real-world datasets are more complex than our simulated datasets, though, so multiple iterations may be necessary in some situations. However, one may assess this by compiling both BiMM tree models with one iteration and with multiple iterations and comparing the posterior log likelihoods.

Another interesting result from the simulation study is that the prediction accuracy of models remained similar whether models were developed using 2, 4 or 7 repeated measurements. We expected to see increases in prediction accuracy with increases in the number of repeated measurements. However, the simulated dataset for our study is created based on the real ALF Study Group registry, so this result may be because the clustering effect for repeated outcomes does not change whether 2, 4 or 7 measurements are included. Though a simulated dataset could have been constructed to induce a specific correlation structure for repeated observations (e.g. autoregressive structure), we wanted the data simulation to resemble our motivating dataset as closely as possible. Our simulated dataset based on the real ALF registry also allows us to assess how the models performed when certain aspects of the data make modeling challenging (e.g. collinear predictors, predictors with skewed distributions with extreme values, and complex interactions between predictors). A future study could assess the performance of BiMM tree methodology for more complex simulated scenarios, such as a high dimensional dataset or a dataset containing nonlinear predictors and high-order interactions.

An application of the novel BiMM tree methodology using the split function which maximizes sensitivity provides a prediction model for acetaminophen-induced ALF patients for daily use for the first week of hospitalization. The model offers predictive (test set) accuracy of 65% and provides clinicians with a simple, interpretable model which can be easily used in practice. Compared to other prediction models in the literature, the BiMM tree uses similar predictors [e.g. King’s College Criteria also uses pH and creatinine, and the model for end stage liver disease uses bilirubin and creatinine (Bernuau 1990; Wiesner et al. 2003)].

The main objective of this study is to develop a flexible framework for constructing prediction models for binary outcomes. BiMM tree methodology may offer comparable or slightly higher prediction accuracy to other models and may be considered an alternative to using GLMM for complex datasets. Future work could investigate the use of alternative implementations of decision tree algorithms within the BiMM tree framework for modeling longitudinal and clustered binary outcomes [e.g. C4.5, GUIDE, QUEST, CRUISE, BART and bartMachine (Chipman, George, and McCulloch 2010; Kapelner and Bleich 2013; Loh 2014)].

An R package for implementing BiMM tree methodology is being developed and will be available on the Comprehensive R Archive Network. An R program implementing BiMM tree methodology is available within Supplemental File 1.

Acknowledgments

This study was funded by the National Institute of Diabetes and Digestive and Kidney Diseases (DK U01-58369). This work was partially supported by the South Carolina Clinical and Translational Research Institute NIH/NCATS Grant (UL1TR001450 and TL1TR001451), NIH/NIAMS Grant (P60 AR062755), NIH/NIGMS Grant (R01 GM122078), and NIH/NCI Grant (R21 CA209848). Authors would like to thank Dr. Paul Nietert for his critical review of this manuscript.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Abdolell, M., M. LeBlanc, D. Stephens, and R. V. Harrison. 2002. Binary partitioning for continuous longitudinal data: Categorizing a prognostic variable. *Statistics in Medicine* 21 (22):3395–409.

Antoniades, C. G., P. A. Berry, J. A. Wendon, and D. Vergani. 2008. The importance of immune dysfunction in determining outcome in acute liver failure. *Journal of Hepatology* 49 (5):845–61. doi:10.1016/j.jhep.2008.08.009.

Bates, D. 2009. Online response to convergence issues in CRAN R LME4 package. http://permalink.gmane.org/gmane.comp.lang.r.lme4.devel/2956

Bates, D., M. Maechler, B. Bolker, S. Walker, R. H. B. Christensen, H. Singmann, B. Dai, G. Grothendieck, C + Eigen, and LinkingTo Rcpp. 2015. Package 'lme4'. *Convergence* 12:1.

Bernuau, J. 1990. Fulminant and subfulminant viral hepatitis. *La Revue du Praticien* 40 (18):1652–5.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and regression trees*. Monterrey, CA: Wadsworth and Brooks.

Chipman, H. A., E. I. George, and R. E. McCulloch. 2010. BART: Bayesian additive regression trees. *The Annals of Applied Statistics* 4 (1):266–98.

De'Ath, G. 2002. Multivariate regression trees: A new technique for modeling species-environment relationships. *Ecology* 83 (4):1105–17.

Dine, A., D. Larocque, and F. Bellavance. 2009. Multivariate trees for mixed outcomes. *Computational Statistics & Data Analysis* 53 (11):3795–804.

Dorie, V. 2013. blme: Bayesian linear mixed-effects models. R package.

Dorie, V. 2014. Mixed methods for mixed models. Columbia University.

Fong, Y., H. Rue, and J. Wakefield. 2010. Bayesian inference for generalized linear mixed models. *Biostatistics* 11 (3):397–412. doi:10.1093/biostatistics/kxp053.

Gelman, A., A. Jakulin, M. G. Pittau, and Y.-S. Su. 2008. A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics* 2 (4):1360–83.

Hajjem, A., F. Bellavance, and D. Larocque. 2011. Mixed effects regression trees for clustered data. *Statistics & Probability Letters* 81 (4):451–9.

Hajjem, A., F. Bellavance, and D. Larocque. 2014. Mixed effects random forest for clustered data. *Journal of Statistical Computation and Simulation* 84 (6):1313–28.

Hastie, T., R. Tibshirani, and J. Friedman. 2001. *The elements of statistical learning*. 2nd ed. New York: Springer.

Hothorn, T., K. Hornik, and A. Zeileis. 2011. party: A Laboratory for Recursive Part (y) itioning. R package version 0.9-9999. http://cran. r-project. org/package=party (accessed December 1, 2010).

Kapelner, A., and J. Bleich. 2013. bartMachine: Machine learning with Bayesian additive regression trees. arXiv Preprint arXiv 1312:2171.

Keon Lee, S. 2005. On generalized multivariate decision tree by using GEE. *Computational Statistics & Data Analysis* 49 (4):1105–19.

Koch, D. G., H. Tillman, V. Durkalski, W. M. Lee, and A. Reuben. 2016. Development of a model to predict transplant-free survival of patients with acute liver failure. *Clinical Gastroenterology and Hepatology* 14 (8):1199–206.

Lee, W. M., R. H. Squires, S. L. Nyberg, E. Doo, and J. H. Hoofnagle. 2008. Acute liver failure: Summary of a workshop. *Hepatology* 47 (4):1401–15.

Loh, W.-Y., and W. Zheng. 2013. Regression trees for longitudinal and multiresponse data. *The Annals of Applied Statistics* 7 (1):495–522.

Loh, W.-Y. 2014. Fifty years of classification and regression trees. *International Statistical Review* 82 (3):329–348.

Mistler, S. A. 2013. A SAS macro for applying multiple imputation to multilevel data. Proceedings of the SAS Global Forum. San Francisco, CA: Contributed Paper (Statistics and Data Analysis).

O'Grady, J. G., G. J. Alexander, K. M. Hayllar, and R. Williams. 1989. Early indicators of prognosis in fulminant hepatic failure. *Gastroenterology* 97 (2):439–45.

R Development Core Team. 2008. *R: a language and environment for statistical computing*. Vienna, Austria: R Development Core Team.

Segal, M. R. 1992. Tree-structured methods for longitudinal data. *Journal of the American Statistical Association* 87 (418):407–18.

Sela, R. J., and J. S. Simonoff. 2012. RE-EM trees: A data mining approach for longitudinal and clustered data. *Machine Learning* 86 (2):169–207.

Speiser, J. L., W. M. Lee, and C. J. Karvellas, for the US Acute Liver Failure Study Group. 2015. Predicting outcome on admission and post-admission for acetaminophen-induced acute liver failure using classification and regression tree models. *PLoS ONE* 10 (4): e0122929.

Stravitz, R. T., Kramer, A. H. T. Davern, A. O. Shaikh, S. H. Caldwell, R. L. Mehta, A. T. Blei, R. J. Fontana, B. M. McGuire, L. Rossaro, et al. 2007. Intensive care of patients with acute liver failure: Recommendations of the U.S. Acute liver failure study group. *Crit Care Med* 35 (11):2498–508. doi:10.1097/01.CCM.0000287592.94554.5F

Therneau, T. M., and E. J. Atkinson. 1997. An introduction to recursive partitioning using the Rpart routines. Mayo Foundation.

Wiesner, R., E. Edwards, R. Freeman, A. Harper, R. Kim, P. Kamath, W. Kremers, J. Lake, T. Howard, R. M. Merion, et al. 2003. Model for end-stage liver disease (MELD) and allocation of donor livers. *Gastroenterology* 124 (1):91–6.

Wu, H., and J.-T. Zhang. 2006. *Nonparametric regression methods for longitudinal data analysis: mixed-effects modeling approaches*, vol. 515. Hoboken, NJ: Wiley.

Yu, Y., and D. Lambert. 1999. Fitting trees to functional data, with an application to time-of-day patterns. *Journal of Computational and Graphical Statistics* 8 (4):749–62.

Zorn, C. 2005. A solution to separation in binary response models. *Political Analysis* 13(02):157–70. 2.