Published in final edited form as:

Stat Med. 2013 March 15; 32(6): 978–994. doi:10.1002/sim.5574.

Recursive Partitioning for Monotone Missing at Random Longitudinal Markers

Shannon Stock^{1,2} and Victor DeGruttola²

¹Department of Biostatistics and Computational Biology, Dana-Farber Cancer Institute, CLS11007, 450 Brookline Ave, Boston, MA 02215

²Department of Biostatistics, Harvard School of Public Health, 655 Huntington Avenue, Boston, MA 02115, U.S.A.

Abstract

The development of HIV resistance mutations reduces the efficacy of specific antiretroviral drugs used to treat HIV infection, and cross-resistance within classes of drugs is common. Recursive partitioning has been extensively used to identify resistance mutations associated with a reduced virologic response measured at a single time point; here we describe a statistical method that accommodates a large set of genetic or other covariates and a longitudinal response. This recursive partitioning approach for continuous longitudinal data uses the kernel of a U-statistic as the splitting criterion, and avoids the need for parametric assumptions regarding the relationship between observed response trajectories and covariates. We propose an extension of this approach that allows longitudinal measurements to be monotone missing at random by making use of inverse probability weights. We assess the performance of our method using extensive simulation studies, and apply them to data collected by the Forum for Collaborative HIV Research as part of an investigation of the viral genetic mutations associated with reduced clinical efficacy of the drug abacavir.

Keywords

Inverse Probability Weighting; Recursive Partitioning; U-statistics

1 Introduction

Over the past 15 years Highly Active Anti-Retroviral Therapy (HAART), which consists of three or more antiretroviral (ARV) drugs, has greatly improved the length and quality of life of persons infected with human immunodeficiency virus (HIV). Patients who adhere to fully active regimens exhibit marked declines in plasma HIV RNA within a few weeks and generally drop to very low levels within 6 months. Because resistance to these drugs is common, however, it is important to understand the relationship between the viral genome and reduced drug sensitivity, both to guide therapy for individual patients and to preserve treatment options for communities. To date, 21 drugs from 5 classes have been approved by the FDA, and considerable research has focused on determining genetic mutations associated with in-vitro measures of drug resistance. Deciphering the relationship between the viral genome and reduced clinical efficacy is more challenging, however, because drugs are used in combination and patients often discontinue treatment for a wide variety of reasons. Particularly challenging has been determining the viral genetic predictors of reduced clinical response to the drug abacavir. To address this challenge, the Forum for Collaborative HIV Research (FCHR) embarked on an investigation of this issue by collecting data from 12 completed studies. Below we describe the development of new methods to aid in this research and their application to these data.

Although antiretroviral drugs from 5 different classes are currently available, only drugs from three classes were used in the studies targeted by the Forum. Two of the classes target reverse transcriptase (RT) and the third, protease (PR). We therefore focus on these regions of the HIV genome. The PR region consists of 99 amino acids, and the RT region consists of over 400; hence statistical methods that can accommodate both high-dimensional and complex data structures are required.

Standard non-parametric, tree-structured methods such as *Classification and Regression Trees* (CART), introduced by Breiman et al. [1], accommodate a large set of covariates, allow for the exploration of complex interactions between covariates, and provide easily interpretable results for a categorical (classification trees) or continuous (regression trees) response. These methods have been used extensively to identify resistance mutations and other baseline covariates associated with a univariate response. For specific examples in the HIV literature, please see Mellors et al. [2], Doherty et al. [3], and Daszykowski et al. [4].

Recursive partitioning methods have also been developed for repeated outcomes data. A tree-structured method for the analysis of longitudinal data was proposed by Segal [5], and further illustrated by Larsen and Speckman [6]. This method uses Mahalanobis distance to measure node homogeneity, and requires equally spaced outcome measurements, as well as specification of a covariance structure. Zhang [7] developed a method for multiple binary responses, whose split function is based on a generalized entropy criterion. Lee [8] presented a method that uses generalized estimating equation (GEE) techniques in the tree construction.

Here we propose an extension of a recursive partitioning method for continuous longitudinal data that uses the kernel of a U-statistic as the split criterion (described by Hu and DeGruttola [9]) to settings where observations may be monotone missing at random (MAR). The use of the U-statistic reduces the dimension of longitudinal outcome measurements by summarizing pairs of subjects' response trajectories and avoids the need for parametric assumptions regarding the relationship between observed outcome trajectories and covariates. Because patients in longitudinal studies often dropout, and dropping out may be related to health status, the restriction of the methods of Hu and DeGruttola to settings where the data are missing completely at random (MCAR) limits its usefulness. Section 2 reviews the recursive partitioning approach for balanced times of measurement described by Hu and DeGruttola, and proposes an extension of their method for pruning the resulting trees. Section 3 presents the adjustment that allows this method to accommodate monotone MAR outcome measurements, Section 4 summarizes simulation results, and Section 5 utilizes this method in the analysis of a motivating data set involving HIV-1 RNA viral load measurements. Finally, Section 6 discusses key features, limitations, and further possible extensions of this work.

As mentioned above, we use these methods to analyze data from several different randomized and observational studies of the drug abacavir, which is in the nucleoside reverse transcriptase inhibitor (NNRTI) class. The FCHR inclusion criteria required that enrolled patients have a failed treatment history, and start a new regimen containing abacavir for the first time. The Forum launched this investigation because of uncertainties regarding the viral genetic factors that most reduced the clinical efficacy of abacavir. Combining data from different sources greatly increased power for these investigations and the times of measurement (baseline, week 8 and week 24) were common to all studies. Nonetheless, there were a large number of patients who missed the week 24 visit; hence the need for new methods.

2 A Recursive Partitioning Method for Longitudinal Data

The formation of a recursively partitioned tree relies on sequential binary splits of the data that, for a given node, maximize some objective function. The objective function, often referred to as a split function, is used to determine the covariate that maximizes the within-node homogeneity or between-node separation of the daughter nodes that would result from a split based on the value of that covariate. Following Hu and DeGruttola, we describe a split function whose structure is the same as the kernel of a U-statistic. It reduces the dimension of longitudinal outcome measurements by using a scoring function to summarize the difference between a pair of subjects' outcome trajectories.

2.1 A Scoring Function to Compare Pairs of Subjects' Outcome Trajectories

We assume each of n enrolled subjects has a complete set of R binary baseline covariates, $(X_{i1}, X_{i2}, \dots, X_{iR}; i = 1, 2, \dots, n)$. Simplifying to the case of balanced data, we further assume each subject i has a trajectory of $P+1$ outcome measurements, $(Y_{i0}, Y_{i1}, \dots, Y_{iP}; i = 1, 2, \dots, n)$, obtained at the corresponding fixed time points $t_0 < t_1 < \dots < t_P$. As in May and DeGruttola [10] and Hu and DeGruttola, we define the scoring function comparing the k^{th} outcome measurement of subject i with the l^{th} outcome measurement of subject j as:

$$D\{(Y_{ik}, t_k), (Y_{jl}, t_l)\} = \begin{cases} 1 & \text{if } Y_{ik} < Y_{jl} \text{ and } t_k \leq t_l, \text{ or } Y_{ik} = Y_{jl} \text{ and } t_k < t_l \\ -1 & \text{if } Y_{ik} > Y_{jl} \text{ and } t_k \geq t_l, \text{ or } Y_{ik} = Y_{jl} \text{ and } t_k > t_l \\ 0 & \text{otherwise} \end{cases}$$

for $i, j = 1, 2, \dots, n$ and $k, l = 1, 2, \dots, P$.

Here we focus on applications involving HIV-1 RNA viral load measurements, for which steeper declines indicate more favorable responses. The kernel therefore assigns a score of 1 if the k^{th} viral load measurement of subject i is less than the l^{th} viral load measurement of subject j and is obtained at the same or an earlier time. Similarly, a score of -1 is assigned to comparisons in which the k^{th} viral load measurement of subject i is greater than the l^{th} viral load measurement of subject j and occurs at the same or a later time. When subject i 's viral load measurement is lower and at a later time, or higher and at an earlier time than subject j 's, we are unable to judge whose performance is better, and consequently a score of 0 is assigned to such comparisons.

This scoring function, which is the kernel of a U-statistic, can be altered to make clinically relevant comparisons based on the expected form of the outcome trajectories under study. For example, if the outcome trajectories consist of CD4 cell counts, for which higher cell counts are more favorable, the scoring function above could be used, but with the inequalities corresponding to Y reversed. The scoring function can also be altered to be sensitive to any feature of the trajectory; for example, using change in viral load makes it sensitive to differences in slopes.

A summary of the comparisons between subjects i and j can be obtained by summing over all pairwise comparisons between the two subjects:

$$D(i, j) = \sum_{k=0}^P \sum_{l=0}^P D\{(Y_{ik}, t_k), (Y_{jl}, t_l)\}.$$

In the context of HIV-1 RNA viral load measurements previously discussed, a positive value of $D(i, j)$ indicates that subject i had a more favorable response trajectory than subject j .

2.2 Tree Construction

Beginning with the root node consisting of all n subjects, let $S_{X_r=1}$ denote the subset of subjects in node S having $X_r = 1$. Similarly, let $S_{X_r=0}$ denote the subset of all subjects in node S with $X_r = 0$. For each covariate X_r , define:

$$G(X_r, S) = \left| \sum_{i \in S_{X_r=1}} \sum_{j \in S_{X_r=0}} D(i, j) \right|.$$

Larger values of this quantity reflect a greater degree of separation between the two subsets of subjects being compared. It therefore provides a measure of the degree of dissimilarity between the outcome trajectories for the pair of daughter nodes that would result from splitting node S based on the value of the covariate X_r . We choose the best binary split of node S to occur at the covariate which maximizes this quantity. More specifically, node S is split according to the value of the covariate X_{r^*} , where

$$G(X_{r^*}, S) = \max_{1 \leq r \leq R} G(X_r, S).$$

Note that this function inherently favors splits based on covariates with a more balanced distribution. In such cases, the values of $G(X_r, S)$ are based on a greater number of comparisons, reflecting the greater amount of information available in the data. This reduces the potential for chance variation in outcome measurements to have undue influence on the recursive partitioning method. We consider alternative splitting functions in the discussion for settings where this property is not desirable.

Once an initial split of the root node is made, recursive partitioning can be used on each resulting daughter node; the full tree is denoted T_0 . The splitting of a node should cease when it is either homogenous with respect to the outcome, a minimum node size has been reached, or the tree is saturated.

2.3 Selecting an Optimally Sized Tree

Pruning algorithms can be based on cost-complexity pruning discussed in Brieman et al. and make use of sequences of optimally pruned nested subtrees. From these sequences, we select the trees that correspond to the maximum value of an accuracy measure, which is based on and calibrated using resampling methods.

2.3.1 Forming a Sequence of Nested Subtrees—

Following Hu and DeGruttola we let $\overset{\circ}{T}$ denote the set of all internal nodes of a tree T , and $|\overset{\circ}{T}|$ the number of such internal nodes. We define a split-complexity measure for T as:

$$H_\alpha(T) = H(T) - \alpha |\overset{\circ}{T}|.$$

Where,

$$H(T) = \sum_{t \in \overset{\circ}{T}} G(., t),$$

is the sum of all goodness-of-split measures corresponding to the internal nodes of T . This quantity provides a summary of the overall performance of the tree T .

Here α is a non-negative pruning parameter that adjusts the penalty for tree complexity, and can be adjusted to favor trees of varying size. For a fixed value of α , we let $T(\alpha)$ denote the smallest optimally pruned subtree of T corresponding to α . That is,

$$H_{\alpha}(T(\alpha)) = \max_{T^* \leq T} H_{\alpha}(T^*).$$

The variation of $T(\alpha)$ with α results in a sequence of nested, optimally pruned subtrees, $(T_0, T_1, T_2, \dots, T_M)$, and a corresponding sequence of intervals of α , $([\alpha_0 = 0, \alpha_1], [\alpha_1, \alpha_2], [\alpha_2, \alpha_3], \dots, [\alpha_M, \alpha_{M+1} = \infty])$, such that $T_m = T_0(\alpha)$ is the optimally pruned subtree for $\alpha_m < \alpha_{m+1}$. Details regarding the order branches are pruned are provided by Hu and DeGruttola.

2.3.2 Selecting a Properly Sized Subtree—Using the same data to grow and prune a tree as calculate its cost-complexity measure tends to overestimate the true cost-complexity and overfit the data. If available, additional observations (independent of the data used to construct the sequence of nested subtrees) can be used to calculate $H_{\alpha}(T)$ and select the subtree corresponding to its maximum value as the optimally sized tree. In this case, problems related to overfitting are avoided because independent observations are used for training and testing potential subtrees.

When there aren't enough observations to justify using an independent sample to calculate the cost-complexity of the subtrees, resampling methods provide a viable alternative. We implement an adaptation of the bootstrap method proposed by Fan et al. [11], which subtracts from $H_{\alpha}(T)$ an estimate of the difference between the goodness-of-split measures calculated when the same versus independent data sets are used to construct a tree and recalculate its goodness-of-split measure. If L_1 and L_2 are two independent data sets, we use the notation $H_{\alpha L_2}(L_1, T)$ to denote the overall goodness-of-split measure based on a tree T constructed using the observations in L_1 , and calculated using the independent observations in L_2 .

The resampling method uses B bootstrap samples, $L^{(b)}$ ($b = 1, 2, \dots, B$), from the original data set, L , and constructs a sequence of optimally pruned subtrees,

$T^{(b)}(\alpha'_m)$ ($m=0, 1, \dots, M$), for each of the B bootstrap samples. Here, the α'_m values correspond to the geometric mean of the α 's from the nested subtrees constructed using the complete data set L . That is, $(\alpha'_m = \sqrt{\alpha_m \alpha_{m+1}})$. For a given α'_m , the bootstrap estimator of $H_{\alpha'_m}(T_m)$, denoted by $H_{\alpha'_m}^{(b)}(T_m)$, is given by:

$$H_{\alpha'_m}^{(b)}(T_m) = H_{\alpha'_m L}(L, T(\alpha'_m)) - \frac{1}{B} \sum_{b=1}^B \left\{ H_{\alpha'_m L^{(b)}}(L^{(b)}, T^{(b)}(\alpha'_m)) - H_{\alpha'_m L}(L^{(b)}, T^{(b)}(\alpha'_m)) \right\}$$

The first term in this estimator is the overall goodness-of-split measure based on the tree constructed using the original data set, L , and pruned to the size corresponding to α'_m . The second term subtracts the mean over the B bootstrap samples of the difference between the goodness-of-split measures calculated when the same versus different samples are used for constructing/pruning a tree and calculating its goodness-of-split summary measures.

True splits, those for which the splitting covariate is associated with the response, will be accompanied by relatively large increases in $H_{\alpha_m}^{(b)}(T_m)$, but small increases may occur even when the covariate is not associated with the response. Hu and DeGruttola suggest selecting the tree with the largest value of $H_{\alpha_m}^{(b)}(T_m)$. To reduce further the possibility of selecting false splits, we propose calibrating this value at each potential split by comparing observed increases to those obtained from random noise.

As an example, to assess the legitimacy of the first split, we simulate random noise by shuffling the covariates to form 1000 new data sets:

Original Data						Resampled Data				
ID	X_1	X_2	X_3	X_4		ID	X_1	X_2	X_3	X_4
1	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	$\xrightarrow{\text{ShuffleCovariates}}$	1	$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$
2	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$		2	$x_{11,1}$	$x_{11,2}$	$x_{11,3}$	$x_{11,4}$
3	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$		3	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$
...	...	...	...	...		...	...	...	...	...
n	$x_{n,1}$	$x_{n,2}$	$x_{n,3}$	$x_{n,4}$		n	$x_{n,1}$	$x_{n,2}$	$x_{n,3}$	$x_{n,4}$

For each resampled data set, we calculate the difference in the bootstrap estimate of the goodness-of-split measure corresponding to one split versus no splits,

$$H_{\alpha_{M-1}}^{(b)}(T_{M-1}) - H_{\alpha_M}^{(b)}(T_M).$$

Using the resulting distribution of the $H_{\alpha_{M-1}}^{(b)}(T_{M-1}) - H_{\alpha_M}^{(b)}(T_M)$'s,

we consider a split plausible if the observed value of $H_{\alpha_{M-1}}^{(b)}(T_{M-1}) - H_{\alpha_M}^{(b)}(T_M)$ is greater than some threshold, e.g. the ninetieth percentile of the distribution obtained from random noise.

The plausibility of the second split is evaluated by comparing the increase in the bootstrap estimate for a tree with two splits versus one split, $H_{\alpha_{M-2}}^{(b)}(T_{M-2}) - H_{\alpha_{M-1}}^{(b)}(T_{M-1})$, the distribution of such differences generated by trees with only one true split. This distribution of differences is obtained by resampling, with replacement, the covariates not used in the first split to form 1000 new data sets:

Original Data (First split based on X_1)						Resampled Data (Values of X_1 are fixed)				
ID	X_1	X_2	X_3	X_4		ID	X_1	X_2	X_3	X_4
1	$x_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$\xrightarrow{\text{ShuffleCovariates}}$	1	$x_{1,1}$	$X_{17,2}$	$X_{17,3}$	$X_{17,4}$
2	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$		2	$X_{2,1}$	$X_{2,2}$	$X_{3,3}$	$X_{3,4}$
3	$X_{3,1}$	$X_{3,2}$	$X_{3,3}$	$X_{3,4}$		3	$X_{3,1}$	$X_{12,2}$	$X_{12,3}$	$X_{12,4}$
...	...	...	...	...		...	...	...	...	...
n	$x_{n,1}$	$x_{n,2}$	$x_{n,3}$	$x_{n,4}$		n	$x_{n,1}$	$x_{24,2}$	$x_{24,3}$	$x_{24,4}$

Again, the observed difference in the bootstrap goodness-of-split measures corresponding to the tree with two splits and the tree with only one split can be compared to the desired percentile of the distribution obtained from random noise. Calibration of subsequent splits can be done analogously.

2.4 Continuous Covariates

The implementation of our recursive partitioning method discussed above focused on the use of binary covariates, but continuous covariates can be accommodated in a manner analogous to that used for CART. The continuous covariate is replaced by a series of binary variables that are indicators of whether the covariate is less than each distinct value in the

observed data set. Each observed value of the continuous covariate is therefore evaluated as a potential cutoff for a binary split of a given node. One resulting daughter node would include subjects with values of that covariate less than or equal to the cutoff, and the other, subjects with values greater than the cutoff. Unordered categorical covariates can be accommodated by evaluating each combination of levels of the categorical covariate as a potential split from the remaining levels of the covariate.

If a split is selected based on one of these indicators, then the calibration method discussed in the previous subsection needs to be modified. Shuffling all covariates after fixing an indicator that the continuous covariate is less than a particular cut-point will lead to illogical covariate combinations, such as simultaneously being less than one cutoff (fixed) and not being less than a larger cutoff (due to shuffling). To prevent this, calibration of splits occurring after a split based on a continuous covariate should be shuffled among subjects within each child node. This will preserve the logical structure of the indicator variables used to represent the covariate.

2.5 Illustration

To investigate the performance of the method that makes use of the pruning algorithm described above, we simulated a data set consisting of 1000 subjects with outcome measurements obtained at three time points ($t = 0, 1, 2$), and twenty binary covariates ($X_1, X_2, \dots, X_{20}$). The correlation between the covariates ranged from 0.5 to 0.85. Outcome measurements were simulated according to a tree that had a first split based on the covariate X_1 , a second split based on X_2 , a third split based on X_3 , and a fourth split based on X_4 . A different rate of decline was used to simulate the outcome profile for subjects within each terminal node (leaf). The tree as well as the simulated mean outcome profile used for each terminal node are described in Figure 1 and Table 1, respectively. Each subject had complete outcome and covariate data.

Implementation of the recursive partitioning method resulted in the correct selection of the first four splits. Figure 2 displays a plot of the bootstrap goodness-of-split measure,

$H_{\alpha_m}^{(b)}(T_m)$, versus the corresponding number of internal nodes. There is a decrease in the improvement of $H_{\alpha_m}^{(b)}(T_m)$ for trees including more than four internal nodes. Calibration of the increases correctly indicated that the appropriately sized tree includes the first four splits (resampled p-values = 0.001, 0.001, 0.094, and 0.001 for the splits based on $X_1 - X_4$, respectively).

3 Accommodating Missing Data

When outcome measurements are not missing completely at random (MCAR), but are instead monotone missing at random (MAR), the method above must be modified accordingly. A method for accommodating such patterns of missing data through the use of inverse probability weighting (IPW) is discussed in this section.

3.1 Adjustment to the Scoring Function

When a given outcome measure predicts the probability of observing subsequent measures, outcome measurements are monotone missing at random (MAR). A weighted summary measure of the rate of decline for two subjects is given by:

$$\tilde{D}(i, j) = \sum_{k=0}^P \sum_{l=0}^P \frac{D(i, j)^{kl}}{\pi_i(t_k) \pi_j(t_l)}.$$

Here $\pi_i(t_k)$ is the probability of observing subject i 's k^{th} outcome measurement. The quantity $D(i, j)^{kl}$ is shorthand notation for the score comparing subject i 's k^{th} outcome measurement with subject j 's l^{th} outcome measurement, $D((Y_{ik}, t_k), (Y_{jl}, t_l))$.

When the probability of observing an outcome measurement depends on outcome history, previous measurements must be observed in order to estimate the weights. Therefore, all outcome measurements must be monotone missing at random up until the last time point that is predictive of observing a subsequent response. Under the assumptions: 1) The probability of observing subject i 's k^{th} outcome measurement and the probability of observing subject j 's l^{th} outcome measurement are independent of the observed values of those quantities, conditional on observed baseline covariates and outcome histories, and 2) The probability of observing the k^{th} outcome measurement of subject i is independent of the probability of observing the l^{th} outcome measurement of subject j , conditional on observed baseline covariates and outcome histories, $\tilde{D}(i, j)$ is an unbiased estimator for $E\{D(i, j)\}$. A proof is given in the Appendix.

3.2 Estimation of the Inverse Probability Weights

A variety of methods can be used to estimate the inverse probability weights. The choice of approach can be guided by the size of the set of covariates that needs to be explored and whether or not the probability of observing an outcome may depend on previous values of that outcome. If the probability of observing an outcome measurement depends on a small set of known covariates, independent logistic regression models can be used at each time point to estimate the $\pi_i(t_k)$. The covariates for these models can include all of the previous history, including previous values of the response.

When a large set of baseline covariates needs to be explored, a recursive partitioning method based on a U-statistic can be used to jointly model the missing data mechanism across all $P + 1$ time points. This requires creating a scoring function, analogous to that discussed in section 2.1 for the observed outcome trajectories, to compare the amount of missing data between pairs of subjects. For example, a scoring function that treats missing data at each measurement time as equally important is given by:

$$M\{R_i(t_k), R_j(t_l)\} = R_i(t_k) - R_j(t_l)$$

Here we assign a positive score of 1 to comparisons in which the k^{th} outcome measurement of subject i is observed and the l^{th} outcome measurement of subject j is not. Similarly, we assign a score of -1 to comparisons in which the k^{th} outcome measurement of subject i is not observed and the l^{th} outcome measurement of subject j is observed. When both subject i 's k^{th} observation and subject j 's l^{th} observation are either observed or not observed, the two subjects contain the same amount of information and we therefore assign a score of 0 to such comparisons. If outcome measurements at certain time points are of greater interest than others, the scoring function above can be altered to incorporate weights as appropriate.

The trees constructed in this way identify the baseline covariate patterns associated with different missingness patterns. The probability that a subject with a given set of covariates is observed at each time point can be estimated as the observed proportion of subjects with an observation at that time point within each terminal node. We can combine this type of RP with logistic regression to accommodate dependence of the probability of observing an outcome on previous values of that outcome. Within each terminal node, we will have different patterns of missingness, although they should be more homogeneous than those across nodes. To allow for dependence on a subject's outcome history within a node, logistic

regressions using outcome history as a covariate can be used to estimate the probability of observing an outcome measurement at each follow-up time point.

4 Simulation Studies

An extensive simulation study was undertaken to assess the performance of our recursive partitioning method. Specifically, we investigated its ability to select the correct splits from simulated data, the effectiveness of IPW in correcting biases resulting from missing observations, and the accuracy of the recursive partitioning method described in subsection 3.2 for modeling the missing data process.

Simulated data sets included n subjects and 10 correlated binary baseline covariates ($X_1, X_2, \dots, X_{10}$), as well as a response trajectory with four outcome measurements (Y_0, Y_1, Y_2, Y_3). The response trajectories were simulated to reflect HIV-1 RNA viral load data motivated by the FCHR data. Subjects' outcome trajectories were simulated based on a tree that split on the baseline covariates X_1 and X_2 . As summarized in Figure 3 and Table 2, subjects falling into the first terminal node ($X_1 = 1$) had the least favorable trajectories, followed by those in the second terminal node ($X_1 = 0, X_2 = 1$), and subjects in the third terminal node ($X_1 = 0, X_2 = 0$) had the best outcome trajectories. Data sets were simulated for sample sizes of $n = 250$ and 500 , and outcome measurements were normally distributed with standard deviations of $0.10, 0.15, 0.20$, and 0.25 .

Missing data were generated for the third and fourth observations in each data set in order to assess the usefulness of the recursive partitioning method in modeling the underlying missing data process, and to evaluate the performance of recursive partitioning using IPW.

For the first two sets of simulations, missing observations were simulated based on the value of the covariate X_1 and the second outcome measurement, as illustrated in Figure 4 and Table 3. In the first set of simulations, approximately 20 percent of the last two outcome measurements were missing; in the second and third, approximately 40 percent were missing. For the third set of simulations, the probability of observing the fourth outcome measurement also depended on the observed value of the third outcome measurement (Figure 5 and Table 3).

4.1 Simulation Results

The results of the simulation studies are summarized in Table 4. The number of simulated data sets that correctly selected the first two splits (unpruned), (X_1 and X_2), is given for each simulation set. The 'Complete' data tree corresponds to the tree constructed using complete simulated data sets. The recursive partitioning methods selected the correct first two splits in at least 83% of the data sets used in our simulation studies. As would be expected, the data sets with a smaller sample size and larger variability in outcome measurements ($n = 250$ and $\sigma = 0.25$) had the lowest percentage of correct splits selected (83%), and the data sets with the largest sample size and least variability ($n = 500$ and $\sigma = 0.10$) had the highest percentage of correctly selected splits (99%).

When the un-weighted recursive partitioning method was applied to data sets with missing data, its performance was poor. As summarized in the 'Missing' column of Table 4, for simulation sets 2 and 3 (with approximately 40% of the last two observations missing), in no more than 61% of data sets were the first two splits correctly identified across all sample size and standard deviation combinations. A comparison of the performance of the recursive partitioning method using complete versus missing data illustrates the potential biases that can result when observations are monotone MAR.

The 'IPW Observed', 'IPW Theoretical', and 'IPW Estimated' trees correspond to those obtained using inverse probability weights based on the observed probability of observing an outcome measurement obtained from generating missing observations for a given data set (the empirical probabilities), the theoretical weights used in the simulations (summarized in Table 3), and the estimated weights obtained via the recursive partitioning method described in subsection 3.2, respectively. The use of IPW was highly effective in correcting the biases due to MAR outcome measurements. The simulation studies show that for all parameter combinations, the IPW trees resulted in the proper selection of the first two splits almost as often as trees constructed using the complete data. Furthermore the results in columns 'IPWObs.' and 'IPWEst.', demonstrate that using weights estimated from recursive partitioning (IPW estimated) performs almost equally as well as using weights calculated using the true probabilities observed from the simulated data sets (IPW Observed).

The 'Missingness' column corresponds to the percent of data sets that correctly identified the splits involved in the trees used to generate the missing data (Figures 4 and 5). The recursive partitioning process accurately identified the correct splits for more than 89% of data sets in each simulation. In other simulations involving covariates with high correlation (not summarized here), there were some instances where the tree used to generate the missing data was not accurately selected. In these situations, however, there was still a marked improvement in the performance of the recursive partitioning method resulting from the use of IPW. Even though the correct covariates were not always selected in estimating the weights, the probability of observing an outcome measurement was nonetheless reasonably well estimated, mostly as a result of the high correlation among the covariates.

5 Application: Forum for Collaborative HIV Research Data

We illustrate these methods using data from the FCHR described in the introduction. Our analyses included 1133 patients with complete baseline sequencing of the reverse transcriptase (RT) and protease (PR) regions of the HIV genome. We restricted the number of genotypic patterns by coding each codon as 1 (mutant) or 0 (wild-type).

Our interest was in exploring baseline mutation patterns associated with virologic response, so we also required all subjects to have a baseline viral load measurement as well as a follow-up at week 8. An additional measurement at week 24 was obtained for approximately 86% of subjects.

We implemented the recursive partitioning method using change in viral load from baseline to week 8, and baseline to week 24. The mutation status at codons RT 1 - RT 230 and PR 1 - PR 99, baseline log₁₀ viral load (continuous variable), and the number of active drugs (Stanford Genetic Susceptibility Score) were included as covariates.

Since analyses involved changes in viral load measurements, and some week 8 and week 24 measurements fell below the lower level of assay detection, we used a slightly altered scoring function. For comparisons involving differences in viral load measurements that were above the lower limit of detection for both subjects, the score from section 2.1 is defined as before. If a follow-up measurement for subject i , but not for subject j , was below the lower limit we assigned a score of 1 when subject i 's change in viral load was greater than subject j 's. The justification is that despite the censoring of subject i 's measurement i is known to have had a larger decline. Similarly, if a measurement was below the lower limit for subject j , but not for subject i , we assigned a score of -1 when subject j 's change in viral load was greater than subject i 's. For all other comparisons, such as when a censored change for subject i was less than an uncensored change for subject j , or a censored change for subject j was less than an uncensored change for subject i , or differences for both subjects

were based on censored observations, a score of zero was assigned because in such circumstances we were unable to determine whose response was more favorable.

Implementation of the recursive partitioning method using an un-weighted version of the score described above resulted in a tree with five splits. The splits were made on an indicator that the Stanford Genetic Susceptibility Score was 1.0 or less, as well as the mutation status at codons PR-13, 37, and RT-41, 215. The un-weighted tree, along with summary statistics of the viral load measurements within each leaf, are provided in Figure 6 and Table 5, respectively. We discuss the interpretation of tree results below but first consider the impact of missing data.

Approximately 15% of subjects were missing the viral load measurement at week 24; to adjust for this, we first used the score described in section 3.2 to model the missing data process. This analysis demonstrated a fairly complex relationship between the probability of observing a week 24 measurement and the genetic data; the resulting tree is summarized in Figure 7. It includes splits based on indicators that the Stanford Genetic Susceptibility Score was less than or equal to 0.5 as the first split and, at a later split, that this score was less than or equal to 2.5. The tree also implied the importance of the mutation status at codons RT-67, 207, 215, and PR-3. The observed proportion of subjects with a week 24 viral load measurement is given for each leaf in Table 6. A logistic regression model including baseline and week 8 viral load measurements as covariates was used within each leaf to estimate the probability a week 24 outcome measurement was observed for each subject.

Adjustment for missing data as described in subsection 3.2 resulted in a tree with six splits, summarized in Figure 8 and Table 7; this tree has some differences from the unadjusted tree. For both trees, the first split was based on an indicator that the Stanford Genetic Susceptibility Score was greater than 1; for patients with more than one active drug in their regimen, no other aspect of the genotype was important. However for patients receiving at most one fully effective drug, there were a number of mutations that were associated with response. For both trees these included RT-41, 215, and PR-37, but the adjusted tree also included RT-181. Both RT-41 and RT-215 have been shown to be associated with in-vitro abacavir resistance and appear to be important for clinical response as well. RT-181 is associated with resistance to drugs of the NNRTI class, and PR-37, to drugs of the PI class. Their presence in the adjusted tree implies that the Stanford Genetic Susceptibility Score does not fully capture the impact of resistance to other drugs in patients' regimens besides abacavir. The reason may be that these mutations are of lower importance in the Stanford Genetic Susceptibility Score than other drugs in these classes. Nonetheless, in our analyses, they do appear to have clinical relevance. Finally we note a later split on the Stanford Genetic Susceptibility Score in the adjusted tree; this split implies that for patients with the RT-41 mutation (but wildtype at PR-37), there is a difference between having at most one partially-active drug in the regimen and having one fully active drug. In other words, for patients with reduced sensitivity to abacavir (because of RT-41) but some potential sensitivity to drugs from the PI or NNRTI class, the distinction between full or partial sensitivity is important.

Cui [1412] also considered methods for variable selection to predict outcome using the Forum data. She used regularized regression via penalized likelihood-based methods including the LASSO, adaptive LASSO, and SCAD. She modeled log-transformed viral load measurements at weeks 8 and 24 using as covariates, baseline viral load, the Stanford genetic susceptibility score, gender, survey type, and the mutation status at codons in the RT region; thus the choice of covariates for consideration overlapped with ours but was not exactly identical. As did our methods, all three of hers selected the Sanford susceptibility score and mutation status at codons RT 41 and 215 as important predictor variables. She

also found mutations at RT 74 and 118 important, whereas we found mutations at RT 181 and PR 37 to be important.

6 Discussion

Recursive partitioning methods provide useful non-parametric techniques for exploring a potentially large set of covariates. Here we extend the regression tree method for univariate outcome measurements proposed by Breiman et al. to accommodate continuous longitudinal data with observations that are potentially monotone MAR. Our method incorporates a split function that summarizes pairs of subjects' outcome trajectories into a univariate score, and then weights these scores by the inverse probability of observation at specific times.

We intentionally designed our split function to favor covariates with a balanced distribution. In such cases, more comparisons are made between subjects, so more information is available for assessing the authenticity of a potential split. Hothorn et al. [13] and Loh [14] point out that a greater number of splits also entail increased false positive rates. An alternative approach would normalize $G(X_r, S)$ to obtain the average effect of a covariate; this is obtained by dividing $G(X_r, S)$ by the total number of comparisons that contribute to the split function. When complete data are available for all subjects, this quantity is given by the product of the number of subjects with $X_r = 1$ and the number of subjects with $X_r = 0$. When the split function is based on the score using IPW, an analogous quantity is given by the sum of the inverse weights.

The selection of a proper tree size relied on an application of the bootstrap method proposed by Fan et al. When standard CART is implemented for continuous outcome measurements, a sum of squares loss function is used and replicates within a bootstrap sample contribute to an increase in the split function for a node unless they are exactly equal to the node's sample mean. In our context involving longitudinal outcome measurements, replicate trajectories within a node don't contribute to the split function because the comparison of any trajectory with itself is zero. For this reason, the summary measure provided by the bootstrap procedure doesn't necessarily peak at the proper tree size. For this reason we suggest that observed increases corresponding to an additional split be calibrated by comparing it to the increase obtained with random noise. As this approach can be computationally intensive, guidance can also be provided by noting the tree size at which the summary measure levels off. Such an approach is especially effective when the ratio of the number of covariates involved in the true tree to the number of candidate covariates is small. Using this criterion is simple, convenient, and most appropriate for exploratory analyses. Simplified resampling methods (similar to random forest/bagging techniques) that provide information regarding tree variability and that aide in the selection of the proper tree size is the subject of future research.

Our recursive partitioning method was developed as an exploratory method for applications involving continuous longitudinal data, but a predicted trajectory for a new observation can be obtained using the same random forest/bagging algorithm that is used for a univariate response. The new observation is pushed down each bootstrapped tree, and a predicted trajectory for each bootstrapped tree is given by the mean trajectory of observations within the terminal node in which it falls. The predictions of the bootstrapped trees are then aggregated by taking their mean to obtain the predicted trajectory for the new observation.

A measure of error that can be used in the variable importance algorithm is obtained by aggregating comparisons of 'Out of Bag' (OOB-not in the bootstrap training sample) observations with observations from the training sample in the corresponding terminal node. More specifically, for each bootstrap iteration, the OOB data is pushed down the tree grown

with the bootstrap sample. For an OOB observation, the absolute value of the mean of all comparisons between the OOB observation and the training data in the same terminal node will be small if the trajectory of the OOB observation is similar to those within the terminal node. The mean of this quantity over all OOB observations for a bootstrapped tree provided a summary measure. The difference between this quantity and that analogously obtained after randomly permuting the values of a covariate for OOB observations will be large if the permuted covariate is important. The mean of these differences over all trees gives an importance score for the covariate.

Our method provides an extremely flexible approach for summarizing outcome trajectories that can also be tailored to accommodate the underlying clinical or biological application being investigated. Here, our clinical motivation relied on HIV-1 RNA viral load data, but the scoring function discussed in section 2.1 can easily be altered to satisfy other clinical settings. A related example includes studies that wish to jointly model bivariate longitudinal outcomes such as HIV-1 RNA viral load measurements and CD4 cell counts. In fields that deal with chronic illness, such as dermatology or neurology, disease progression tends to exhibit flares and therefore results in highly non-parametric trajectories that pose data analytic challenges. The flexibility of our method accommodates a wide variety of settings.

Appendix

Here we show that under the assumptions 1) The probability of observing subject i 's k^{th} outcome measurement and the probability of observing subject j 's l^{th} outcome measurement are independent of the observed values of those quantities, conditional on observed baseline covariates and outcome histories, and 2) The probability of observing the k^{th} outcome measurement of subject i is independent of the probability of observing the l^{th} outcome measurement of subject j , conditional on observed baseline covariates and outcome histories, then $\tilde{D}(i, j)$ is an unbiased estimator of $E\{D(i, j)\}$.

More explicitly, assumptions 1 and 2 require that, for all pairs of subjects i and j :

$$R_i^k, R_j^l \perp\!\!\!\perp y_{ik}, y_{jl} \mid \{x_i\}, \{x_j\}, \{y_{i,hist_{k-1}}\}, \{y_{j,hist_{l-1}}\} \quad 1$$

$$R_i^k \perp\!\!\!\perp R_j^l \mid \{x_i\}, \{x_j\}, \{y_{i,hist_{k-1}}\}, \{y_{j,hist_{l-1}}\} \quad 2$$

Here R_i^k is the indicator that the k^{th} outcome measurement of subject i is observed, $\{x_i\}$ denotes subject i 's baseline covariates, and $\{y_{i,hist_{k-1}}\}$ represents subject i 's observed outcome history until time k (that is, $\{y_{i,hist_{k-1}}\} = \{y_{i1}, \dots, y_{i,k-1}, R_i^1, \dots, R_i^{k-1}\}$).

First, note that:

$$E\{\tilde{D}(i, j)\} = \sum_{k=0}^P \sum_{l=0}^P E\left\{\frac{D(i, j)^{kl} R_i^k R_j^l}{\pi_i^k \pi_j^l}\right\}.$$

So to show that $\tilde{D}(i, j)$ is unbiased for $D(i, j)$, it suffices to show that

$$E \left\{ \frac{D(i, j)^{kl} R_i^k R_j^l}{\pi_i^k \pi_j^l} \right\} = E \{ D(i, j)^{kl} \}.$$

Second, note that assumption 1 implies:

$$R_i^k, R_j^l \perp\!\!\!\perp D(i, j)^{kl} | \{x_i\}, \{x_j\}, \{y_{i, hist_{k-1}}\}, \{y_{j, hist_{l-1}}\}.$$

This follows since,

$$\begin{aligned} & P(D(i, j)^{kl} = 1, R_i^k = r_i^k, R_j^l = r_j^l | \{c\}) \\ &= P(y_{ik} < y_{jl}, R_i^k = r_i^k, R_j^l = r_j^l | \{c\}) \\ &= P(y_{ik} < y_{jl} | \{c\}) P(R_i^k = r_i^k, R_j^l = r_j^l | \{c\}), \end{aligned}$$

Where we let $\{c\} = (\{x_i\}, \{x_j\}, \{y_{i, hist_{k-1}}\}, \{y_{j, hist_{l-1}}\})$. We can similarly show this also holds for $D(i, j)^{kl} = -1$, and $D(i, j)^{kl} = 0$.

Repeated use of iterative conditional expectation gives:

$$\begin{aligned} & E \left\{ \frac{D(i, j)^{kl} R_i^k R_j^l}{\pi_i^k \pi_j^l} \right\} = E_{(c), R_i^k, R_j^l} \left\{ E \left(\frac{D(i, j)^{kl} R_i^k R_j^l}{\pi_i^k \pi_j^l} | \{c\}, R_i^k, R_j^l \right) \right\} \\ &= E_{(c), R_i^k, R_j^l} \left\{ \frac{R_i^k R_j^l}{\pi_i^k \pi_j^l} E(D(i, j)^{kl} | \{c\}, R_i^k, R_j^l) \right\} \\ &= E_{(c), R_i^k, R_j^l} \left\{ \frac{R_i^k R_j^l}{\pi_i^k \pi_j^l} E(D(i, j)^{kl} | \{c\}) \right\} \end{aligned}$$

(by assumption 1)

$$\begin{aligned} &= E_{\{c\}} \left\{ E_{R_i^k, R_j^l} \left[\frac{R_i^k R_j^l}{\pi_i^k \pi_j^l} E(D(i, j)^{kl} | \{c\}) | \{c\} \right] \right\} \\ &= E_{\{c\}} \left\{ \frac{E(D(i, j)^{kl} | \{c\})}{\pi_i^k \pi_j^l} E_{R_i^k, R_j^l} (R_i^k R_j^l | \{c\}) \right\} \\ &= E_{\{c\}} \left\{ \frac{E(D(i, j)^{kl} | \{c\})}{\pi_i^k \pi_j^l} E_{R_i^k} (R_i^k | \{c\}) E_{R_j^l} (R_j^l | \{c\}) \right\} \end{aligned}$$

(by assumption 2)

$$\begin{aligned} &= E_{\{c\}} \left\{ \frac{E(D(i, j)^{kl} | \{c\})}{\pi_i^k \pi_j^l} \pi_i^k \pi_j^l \right\} \\ &= E_{\{c\}} \{ E(D(i, j)^{kl} | \{c\}) \} \\ &= E \{ D(i, j)^{kl} \}. \end{aligned}$$

References

1. Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. Classification and Regression Trees. Wadsworth, Inc.; 1984.
2. Mellors J, Munoz A, Giorgi J, Margolick J, Tassoni C, Gupta P, Kingsley L, Todd J, Saah A, Detels R, Phair J, Rinaldo C. Plasma viral load and cd4 lymphocytes as prognostic markers of HIV-1 infection. *Annals of Internal Medicine*. 1997; 126:946–954. [PubMed: 9182471]
3. Doherty M, Garfein R, Monterroso E, Brown D, Vlahov D. Correlates of HIV infection among young adult short term injection drug users. *AIDS*. 2000; 14:717–726. [PubMed: 10807195]
4. Daszykowski M, Walczak B, Xu Q, Daeyaert F, DeJonge M, Heeres J, Koymans L, Lewi P, Vinkers H, Janssen P, Massart D. Classification and regression trees-studies of HIV reverse transcriptase inhibitors. *Journal of Chemical Information and Computer Sciences*. 2004; 44:716–726. [PubMed: 15032554]
5. Segal M. Tree-structured methods for longitudinal data. *Journal of the American Statistical Association*. 1992; 87(418):407–418.
6. Larsen D, Speckman P. Multivariate regression trees for analysis of abundance data. *Biometrics*. 2004; 60(2):543–549. [PubMed: 15180683]
7. Zhang H. Classification trees for multiple binary responses. *Journal of the American Statistical Association*. 1998; 93(441):180–193.
8. Lee S. On classification and regression trees for multiple responses and its application. *Journal of Classification*. 2006; 23(1):123–141.
9. Hu C, DeGruttola V. Recursive partitioning of resistance mutations for longitudinal markers based on a U-type score. *Biostatistics*. 2011; 0(0):1–13.
10. May S, DeGruttola V. Nonparametric tests for dependent observations obtained at varying time points. *Biometrics*. 2007; 63(1):194–200. [PubMed: 17447945]
11. Fan J, Su X, Levine R, Nunn M, LeBlanc M. Trees for correlated survival data by goodness of split, with applications to tooth prognosis. *Journal of the American Statistical Association*. 2006; 101(475):959–967.
12. Cui, Rain. Doctoral dissertation. 2011. Variable Selection Methods for Longitudinal Data. Retrieved from ProQuest Dissertations and Theses Database.(AAT 3491937)
13. Hothorn T, Hornik K, Zeileis A. Unbiased Recursive Partitioning: A Conditional Inference Framework. *Journal of Computational and Graphical Statistics*. 2006; 15(3):651–674.
14. Loh W. Classification and Regression Trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*. 2011; 1:14–24.

Figure 1.
Structure of the tree used to simulate outcome trajectories for the illustrative example in Section 2.4

Figure 2. Plot of the bootstrap goodness-of-split estimator versus the number of internal nodes for the simulated example in Section 2.4 (includes up to 20 internal nodes)

Figure 3.
Structure of the tree used to simulate outcome trajectories for the simulation studies

Figure 4.
Structure of the tree used to simulate missing observations for the first and second simulation studies

Figure 5.
Structure of the tree used to simulate missing observations for the third simulation study

Figure 6. Covariates selected via recursive partitioning for the un-weighted Forum data analysis, where 1 = mutant and 0 = wild-type for splits based on codons, or 1 = true and 0 = false for splits based on the Stanford variable

Figure 7.

Covariates selected via recursive partitioning for the Forum missing data analysis, here 1 = mutant and 0 = wild-type for splits based on codons, or 1 = true and 0 = false for splits based on the Stanford variable

Figure 8. Covariates selected via recursive partitioning for the Forum IPW weighted data analysis, here 1 = mutant and 0 = wild-type for splits based on codons, or 1 = true and 0 = false for splits based on the Stanford variable

Table 1

Summary of the simulated outcome profile (Y_0 , Y_1 , Y_2) corresponding to each leaf in Figure 1, where $N(\mu, \sigma^2)$ refers to a normally distributed random variable with mean μ and variance σ^2

Leaf	$Y_0 =$	$Y_1 =$	$Y_2 =$
1	$N(100, 10)$	$Y_0 - N(5, 3)$	$Y_1 - N(5, 3)$
2	$N(100, 10)$	$Y_0 - N(10, 3)$	$Y_1 - N(5, 3)$
3	$N(100, 10)$	$Y_0 - N(25, 5)$	$Y_1 - N(10, 5)$
4	$N(100, 10)$	$Y_0 - N(35, 5)$	$Y_1 - N(25, 5)$
5	$N(100, 10)$	$Y_0 - N(40, 5)$	$Y_1 - N(35, 5)$

Table 2

Summary of the simulated outcome profiles (Y_0, Y_1, Y_2, Y_3) corresponding to the leaves in Figure 3, where $N(\mu, \sigma^2)$ refers to a normally distributed random variable with mean μ and variance σ^2

Leaf	$Y_0 =$	$Y_1 =$	$Y_2 =$	$Y_3 =$
1	$N(4.75, \sigma^2)$	$Y_0 - N(0.55, \sigma^2)$	$Y_1 - N(0.50, \sigma^2)$	$Y_2 - N(0.15, \sigma^2)$
2	$N(4.75, \sigma^2)$	$Y_0 - N(0.65, \sigma^2)$	$Y_1 - N(0.60, \sigma^2)$	$Y_2 - N(0.55, \sigma^2)$
3	$N(4.75, \sigma^2)$	$Y_0 - N(0.75, \sigma^2)$	$Y_1 - N(0.65, \sigma^2)$	$Y_2 - N(0.65, \sigma^2)$

Table 3

Probability of observing a third and fourth outcome measurement for the leaves corresponding to the trees displayed in Figures 4 and 5, where P_k is the probability of observing a response at time k

Leaf	Set1			Set2			Set3		
	P_3	P_4	P_3	P_3	P_4	P_3	P_3	P_4	P_4
1	0.5	0.4	0.3	0.2	0.3	0.2	0.3	0.2	0.2
2	0.9	0.8	0.7	0.6	0.6	0.7	0.7	0.6	0.6
3	1.0	1.0	0.9	0.8	0.8	0.9	0.9	0.78	0.78
4						0.9	0.9	1.0	1.0

Table 4

Simulation results-Percent of simulated data sets that correctly identified the first two splits (unpruned), here 'Complete'= tree constructed from the complete data, 'Missing' = tree constructed with missing data, 'IPWObs.' = IPW tree constructed using weights estimated from the observed probability of observing a response obtained from generating missing observations for a given data set (empirical weights), 'IPWTheo.' = IPW tree constructed using the theoretical weights used in the simulations, 'IPWEst.' = IPW tree constructed using weights estimated via recursive partitioning, and 'Missingness' = tree constructed for the missing data process

Set	n	σ	Complete	Missing	IPWObs.	IPWTheo.	IPWEst. (Missingness)
Set1	250	0.1	0.96	0.73	0.95	0.95	0.95(0.99)
	500	0.1	0.99	0.83	0.99	0.99	0.99(1.00)
	250	0.15	0.86	0.66	0.86	0.85	0.86(1.00)
	500	0.15	0.95	0.79	0.95	0.95	0.95(1.00)
	250	0.2	0.85	0.55	0.84	0.84	0.84(0.91)
	500	0.2	0.95	0.60	0.94	0.95	0.94(0.99)
	250	0.25	0.83	0.44	0.81	0.80	0.81(0.95)
	500	0.25	0.95	0.54	0.93	0.93	0.92(1.00)
Set2	250	0.1	0.96	0.55	0.95	0.94	0.94(0.93)
	500	0.1	0.99	0.61	0.98	0.99	0.98(0.99)
	250	0.15	0.86	0.45	0.83	0.82	0.83(0.94)
	500	0.15	0.95	0.54	0.93	0.94	0.93(1.00)
	250	0.2	0.85	0.36	0.84	0.84	0.83(0.90)
	500	0.2	0.95	0.42	0.94	0.94	0.93(0.97)
	250	0.25	0.83	0.31	0.80	0.78	0.79(0.90)
	500	0.25	0.95	0.36	0.92	0.92	0.92(0.91)
Set3	250	0.1	0.96	0.54	0.95	0.94	0.94(0.91)
	500	0.1	0.99	0.60	0.98	0.99	0.98(0.99)
	250	0.15	0.86	0.45	0.83	0.83	0.82(0.95)
	500	0.15	0.95	0.54	0.93	0.93	0.92(1.00)
	250	0.2	0.85	0.36	0.83	0.83	0.83(0.89)
	500	0.2	0.95	0.41	0.94	0.93	0.91(0.97)
	250	0.25	0.83	0.29	0.80	0.80	0.78(0.90)
	500	0.25	0.95	0.35	0.91	0.90	0.90(0.93)

Table 5
Summary statistics of the outcome measurements corresponding to the leaves in Figure 6

Leaf	n	Y ₀			Y ₈			Y ₂₄		
		mean	(Range)	n	mean	(Range)	n	mean	(Range)	n
1	22	4.77	(3.15,6.24)	22	4.12	(2.60,5.55)	20	4.33	(2.60,5.77)	20
2	73	4.57	(2.97,6.60)	73	3.35	(2.60,6.21)	69	3.54	(2.60,6.38)	69
3	225	4.62	(2.71,6.46)	225	3.86	(2.60,6.73)	204	4.09	(2.60,6.46)	204
4	108	4.58	(2.90,6.18)	108	3.56	(2.60,6.44)	93	3.81	(2.60,5.89)	93
5	221	4.37	(2.74,6.01)	221	3.12	(2.60,6.32)	169	3.35	(2.60,6.62)	169
6	484	4.36	(2.70,6.88)	484	3.09	(2.60,6.55)	424	3.22	(2.60,6.37)	424

Table 6

Observed proportion of subjects with a week 24 follow-up measurement for each leaf in Figure 7

Leaf	n	$P(R_i(\text{week24}) = 1)$
1	111	65.77%
2	159	91.19%
3	107	76.64%
4	68	77.94%
5	13	38.46%
6	290	95.86%
7	385	89.09%

Table 7
Summary statistics of the outcome measurements corresponding to the leaves in Figure 8

Leaf	n	Y ₀			Y ₈			Y ₂₄		
		mean	(Range)	n	mean	(Range)	n	mean	(Range)	n
1	95	4.62	(2.97,6.60)	95	3.53	(2.60,6.21)	89	3.72	(2.60,6.38)	
2	126	4.63	(2.71,6.46)	126	4.02	(2.60,6.73)	110	4.30	(2.60,6.46)	
3	99	4.61	(2.80,6.21)	99	3.67	(2.60,6.36)	94	3.84	(2.60,6.24)	
4	54	4.61	(3.00,6.01)	54	3.82	(2.60,6.44)	42	4.06	(2.60,6.19)	
5	84	4.47	(2.90,6.18)	84	3.38	(2.60,5.88)	73	3.65	(2.60,5.89)	
6	191	4.38	(2.74,5.84)	191	3.60	(2.60,6.13)	147	3.29	(2.60,6.62)	
7	484	4.36	(2.70,6.88)	484	3.09	(2.60,6.55)	424	3.22	(2.60,6.37)	