University of California, Berkeley

U.C. Berkeley Division of Biostatistics Working Paper Series

Year 2005 *Paper* 185

Cross-Validating and Bagging Partitioning Algorithms with Variable Importance

Annette M. Molinaro* Mark J. van der Laan†

*Division of Biostatistics, Yale University School of Medicine, annette.molinaro@yale.edu

†Division of Biostatistics, School of Public Health, University of California, Berkeley, laan@berkeley.edu

Cross-Validating and Bagging Partitioning Algorithms with Variable Importance

Annette M. Molinaro and Mark J. van der Laan

Abstract

We present a cross-validated bagging scheme in the context of partitioning algorithms. To explore the benefits of the various bagging scheme, we compare via simulations the predictive ability of single Classification and Regression (CART) Tree with several previously suggested bagging schemes and with our proposed approach. Additionally, a variable importance measure is explained and illustrated.

1 Introduction

Clinicians aim toward a more preventative model of attacking cancer by pinpointing and targeting specific early events in disease development. These early events can be measured as genomic, proteomic, epidemiologic, and/or clinical variables, using expression or Comparative Genomic Hybridization (CGH) microarrays, SELDI-TOF/mass spectra, patient histories, and pathology and histology reports. These measurements are then used to predict clinical outcomes such as time to primary occurrence, recurrence, metastasis, or mortality.

In such analyses, the primary goal is to unearth biologically driven associations between variables and clinical outcomes. Additionally, statisticians must be able to quantify the interactions between different types of variables and the effects of those interactions on the clinical outcome. The subsequent challenge is to assess how well a selected model will predict outcomes in an independent validation sample, i.e., in future data sets.

Recursive partitioning seeks to explain the individual contributions of various covariates as well as their interactions for the purposes of predicting outcomes, either continuous or categorical. As such, one might choose to use Classification and Regression Trees (CART), a binary recursive partitioning algorithm, to generate said sieves (Breiman *et al.*, 1984). Another option is the Partitioning Deletion/Substitution/Addition (Part-DSA) algorithm which builds *'and'* and *'or'* statements (Molinaro and van der Laan, 2004). This algorithm not only splits regions (nodes in tree estimation) it also combines and substitutes regions. These additional moves allow us to unearth intricate correlation patterns and further elucidate interactions in addition to main effects.

An important consideration when using recursive binary partitioning is the stability of the resulting predictor. Algorithms such as CART are sensitive to data fluctuations and, thus, given a perturbation will potentially build a different predictor than that built on the original data. This calls into question the generalizability of these predictors to independent data sets.

Various methods have been suggested to improve the prediction and classification accuracy of a single recursive partitioning tree. Breiman [1994, 1996] suggested bootstrap aggregating, or *bagging*, for stabilizing predictors. In bagging, numerous trees are grown each with a random selection (with replacement), i.e. a bootstrap sample, from the learning set. The

resulting *aggregated* predictor for a continuous outcome is the average prediction over all trees, while the resulting classification is that class which receives the majority vote.

Aggregated predictors are a promising approach to the motivating problem of predicting outcomes based on hundreds or thousands of variables many of which are measured on different scales. In addition to stabilizing a single predictor, aggregated predictors accumulate much more information. For example, a single regression tree represents a collection of estimators indexed by the number of terminal nodes k, where $k = 1, \ldots, K$. In CART, the best k, or level of the tree, is chosen via cross-validation. The chosen level then represents the minimizer of the sum of squared regression specific residuals over all regressions which are linear combinations of a maximum number of levels K.

As a single tree this results in a linear combination of very few partitions. In most applications one expects that the true regression equals a linear combination of thousands (or an infinite number) of partitions with many very small coefficients. Different from the single tree, the bagged trees average regression estimators based on multiple partitions of size k. As a consequence, the aggregated estimator equals a linear combination of possibly thousands of partitions with many very small coefficients. Thus, we believe that these bagged estimators correspond with sensible fits in numerous applications.

There have been several suggestions for bagging. In Breiman's initial approach, for each bootstrap sample, v-fold cross-validation is employed to prune, i.e. select the best level of the tree (Breiman [1994]). We will refer to this initial approach as **Breiman VFOLD**. Subsequently, Breiman suggested using a test set to pick the level of the tree [Breiman, 1996]. In this approach a bootstrap sample from the learning set is used to grow a tree and then the prediction error is estimated with the entire learning set. The chosen 'best' level of the tree corresponds with the one with minimal prediction error. We will refer to this approach as **Test Tree**.

In a later paper, Breiman suggested growing a full tree for each bootstrap sample avoiding any model selection [Breiman, 1999]. As such, the full tree is assumed to be the best model, or level of the tree, for each bootstrap sample. We will refer to this approach as **Full Tree**.

While cross-validation may not be entirely tolerant of perturbations in the data (e.g. Breiman VFOLD), we maintain that it is the best method for estimator selection given a collection of plausible estimators. Here, we

propose to first build candidate predictors and then chose the 'best' via v-fold cross-validation. Thus, our class of estimators is the collection of aggregate predictors spanning the possible number of partitions. We then choose the best number of partitions (or level of the tree) by minimizing the cross-validated error. In comparison to Breiman's approaches, this can be viewed as an 'external' cross-validation where the level of the tree is selected subsequent to aggregating the predictors. As a result our class of candidate estimators differs from those produced by Breiman's methods. Where we use cross-validation as an estimator selection tool he uses it to build estimators and then averages over said estimators. As a result his class of estimators includes a sole estimator.

Although Breiman's VFOLD approach may provide the right selection among the estimators by trading off bias and variance it may not perform well in terms of the bagged estimator's prediction accuracy. The reason for this is that the bagged estimator should be less variable as a result of the averaging, although more biased. This increase in bias is due to two (possibly cumulative) sources: first, the bias introduced by applying an estimator to a bootstrap sample relative to the empirical sample; and second, the bias introduced by applying the estimator to the empirical sample relative to the truth.

We contend that cross-validation is immensely important for the proper selection of estimators. Although the VFOLD approach employs cross-validation it occurs within each bootstrap sample. The resulting increase in bias will tend toward underfitting. On the other hand, the Full Tree and Test Tree methods and lack of honest cross-validation will tend toward overfitting. The Test Tree approach provides an interesting mix between a resubstitution and an independent test set estimate of prediction error. The resubstitution estimate pertains to the ≈ 0.628 unique observations in the learning set which comprise the bootstrap sample, while the $\approx .368$ observations not included in the bootstrap sample offsets the overfitting and acts as the independent test set. We anticipate that neither of these methods will perform well in the situation where a small or medium sized tree is most appropriate. Their inherent nature will always overfit and lacks the ability for correction.

Our proposed cross-validated bagged estimator also corresponds to overfitting; however, we expect that the 'external' cross-validation will prove to be more flexible and, if not improve the generalizability error, it will be equivalent to the best of the previously suggested schemes.

A large number of available variables will be included in this bagged linear regression fit, and, as such, a measure of variable importance can be assessed. One such measure is the partial derivative with respect to each variable, averaged over all observed covariate values. This variable importance measure provides a relevant and useful summary measure of the actual regression fit.

The next section elaborates on the setting and methods for our proposed cross-validated bagged estimators. The main steps for our approach are detailed and contrasted to Breiman's bagging approach and in Section 2.4 the suggested variable importance measures are explained. In Section 3 we compare our approach to Breiman's via simulations and in Section 4 via a publicly available data set. On the latter we also illustrate the variable importance measure.

2 Methods

This section elaborates on the setting and methods for our proposed cross-validated bagged estimators. We begin by detailing the data structure, emphasizing the choice of a loss function, and describing piecewise constant estimators, both CART and Part-DSA. The main steps for our approach are detailed and contrasted to Breiman's bagging approaches. Our suggested variable importance measures are motivated and explained as well as how to accommodate censored data.

The *observed data structure* can be written as an i. i. d. sample $O_1, \ldots, O_n$, where O includes an outcome T and baseline covariates $\mathbf{W}$, i.e., $O = (T, \mathbf{W})$. For the time being we will consider the observed data to be complete and thus equivalent to the full data structure, i.e., no missingness on covariates nor outcomes. See Section 2.3 for references with consideration of censored outcomes. The data generating distribution of O is denoted with P_0 and the empirical probability distribution with P_n. Assume that $P_0 \in \mathcal{M}$ for some model $\mathcal{M}$, and let $\Psi : \mathcal{M} \to (D(\mathcal{S}))$ be the parameter of interest. The parameter ψ_0 is defined in terms of a *loss function*, $L(O, \psi)$, as the minimizer of the expected loss, or *risk*. We can write ψ_0 as:

$$\psi_0 = \Psi(P_0) = \arg\min_{\psi \in \mathbf{\Psi}} \int L(o, \psi) dP_0(o)$$

over a parameter space $\mathbf{\Psi} \subset D(\mathcal{S})$. The purpose of the loss function L is to

quantify performance. Thus, depending on the parameter of interest, there could be numerous loss functions from which to choose. In regression trees with a continuous outcome, a common loss function is the squared error loss, $L(O,\psi) = (T - \psi(W))^2$, corresponding to the conditional mean $\psi_0(W) = E_0[T \mid W]$. In classification trees with a categorical outcome, the indicator loss function, $L(O,\psi) = I(T \neq \psi(W))$ is frequently used. This loss function corresponds to choosing the class with maximum probability given covariates W, i.e., $\psi_0(W) = \text{argmax}_t Pr_0(t \mid W)$.

2.1 Piecewise Constant Regression Estimators

In tree-based estimation procedures such as CART [Breiman et al., 1984], the candidate estimators are generated by recursive binary partitioning of a suitably defined covariate space. We have also recently introduced a new method for forming piecewise constant estimators, the Partitioning Deletion Substitution Addition algorithm (Part-DSA) [Molinaro and van der Laan, 2004a,b]. Part-DSA exhaustively searches the covariate space forming 'and' and 'or' statements.

Both of these approaches define a countable set of *basis functions*, $\{\phi_j : j \in I\!N\}$, indexed by the non-negative integers $I\!N$. These basis functions are simply set indicators $\{\mathcal{R}_j : j \in I\}$ which form a partition of the covariate space $\mathcal{S}$, where I is an index set, $I \in \mathcal{I}$, and $\mathcal{I}$ is a collection of subsets of $I\!N$.

Here $\mathcal{R}_j$ denotes regions of $\mathcal{S}$ which are disjoint ($\mathcal{R}_j \cap \mathcal{R}_{j'} = \emptyset$, $j \neq j'$) and exhaustive ($\mathcal{S} = \cup_{j \in I} \mathcal{R}_j$). Now every parameter $\psi \in \Psi$ can be written (and approximated) as a finite linear combination of the basis functions:

$$\psi_{I,\beta}(\cdot) \equiv \sum_{j \in I} \beta_j \phi_j(\cdot),$$

where for a given index set $I \in \mathcal{I}$, the coefficients $\beta = (\beta_1, \ldots, \beta_{|I|})$ belong to $B_I \equiv \{\beta : \psi_{I,\beta} \in \Psi\} \subseteq I\!R^{|I|}$. These are of the form referred to as *piecewise constant regression models* [Härdle, 1989].

The complete parameter space $\mathbf{\Psi}$ can be written as the collection of basis functions $\{\phi_j : j \in I\!N\}$ and represented by

$$\mathbf{\Psi} \equiv \{\psi_{I,\beta}(\cdot) = \sum_{j \in I} \beta_j \phi_j(\cdot) : \beta,\ I \in \mathcal{I}\}.$$

Define a *sieve*, $\{\Psi_k\}$, of subspaces $\Psi_k \subset \mathbf{\Psi}$, of increasing dimension approximating the complete parameter space $\mathbf{\Psi}$, such as,

$$\Psi_k \equiv \left\{ \psi_{I,\beta}(\cdot) = \sum_{j \in I} \beta_j \phi_j(\cdot) : \beta,\ I,\ |I| \leq k \right\},$$

where k denotes the index set size (i.e., how many basis functions). Now for every k we want to find the estimator which minimizes the empirical risk over the subspace Ψ_k. That can be done by initially optimizing over the regression coefficients $\beta \in B_I$ for a given index set I and then optimizing over the index sets I.
Given index sets $I \in \mathcal{I}$, define I-specific subspaces

$$\Psi_I \equiv \{\psi_I, \beta : \beta \in B_I\}.$$

For each subspace Ψ_I, the regression coefficients β are estimated by minimizing the empirical risk, i.e.,

$$\begin{aligned} \hat{\beta}_I = \beta_I(P_n) &\equiv \text{argmin}_{\beta \in B_I} \int L(o, \psi_{I,\beta}) dP_n(o) \\ &= \text{argmin}_{\beta \in B_I} \sum_{i=1}^{n} L(O_i, \psi_{I,\beta}), \end{aligned}$$

It is possible to write the I-specific estimators as $\hat{\psi}_I = \Psi_I(P_n) \equiv \psi_{I,\beta_{I(P_n)}}, I \in \mathcal{I}$. For example, with the squared error loss function $\hat{\psi}_I$ is the least squares linear regression estimator corresponding with the variables identified by the index set I.

CART and Part-DSA can be used to construct a sequence of candidate estimators, $\hat{\psi}_k = \psi_k(P_n)$, $k \in \{1, \ldots, K(n)\}$, up to a maximal size, $\psi_{max} = \psi_{K(n)}$. Here, k indexes the number of partitions, measured by the number of terminal nodes in CART and by the number of basis functions in Part-DSA. The maximum size $K(n)$ of the partitioning is typically determined by criteria such as the complexity parameter (cp) as defined in Breiman et al. [1984], the minimal number of observations need to further split a partition, and/or homogeneity (purity) for categorical outcomes. Given this sequence of candidate estimators, the goal is to select a data adaptive $\hat{k} = k(P_n) \in \{1, \ldots, K(n)\}$, such that the risk for $\psi_{\hat{k}}(P_n)$ converges to that for the parameter ψ_0 in an optimal manner. In order to address this

selection problem, define the *conditional risk* of the estimator $\psi_k(P_n)$ based on the loss function as

$$\tilde{\theta}_n(k) \equiv \int L(o, \psi_k(P_n)) dP_0(o).$$

Also define the *optimal risk*, θ_{opt}, as the risk of the parameter of interest,

$$\theta_{opt} = \min_{\psi \in \Psi} \int L(o, \psi) dP_0(o).$$

Let

$$\tilde{k}_n \equiv \text{argmin}_k \, \tilde{\theta}_n(k) = \text{argmin}_k \int L(o, \psi_k(P_n)) dP_0(o)$$

be the *optimal benchmark selector* which chooses the estimator with minimal conditional risk $\tilde{\theta}_n(k)$, for *each given data set*. If the minimum is not unique, then the argmin is defined as the smallest k achieving the minimum. A selector $\hat{k} = k(P_n)$ is said to be *asymptotically equivalent* with the optimal benchmark $\tilde{k}_n$ if

$$\frac{\tilde{\theta}_n(\hat{k}) - \theta_{opt}}{\tilde{\theta}_n(\tilde{k}_n) - \theta_{opt}} \to 1 \text{ in probability.} \tag{1}$$

In particular, then it is *asymptotically optimal* [van der Laan and Dudoit, 2003].

Note that the optimal benchmark selector $\tilde{k}_n$ depends on the unknown data generating distribution P_0. The selection problem therefore involves estimating the unknown conditional risk $\tilde{\theta}_n(k)$ for each candidate estimator $\hat{\psi}_k = \psi_k(P_n)$. Cross-validation provides a general approach for estimating the conditional risk and producing a data adaptive selector $\hat{k}$ which is asymptotically equivalent to the oracle selector $\tilde{k}_n$ based on the true data generating distribution P_0. The default in both CART and Part-DSA is 10-fold cross validation for estimator selection.

2.2 Bagging Estimators

We will write the collection of regression estimators based on a recursive partitioning (described in Section 2.1) as $\hat{\Psi}^1_s(P_n)$ indexed by $s \in \mathcal{A}_n$, where s represents a vector of fine-tuning parameters. For example in CART, s contains the size of the tree k which ranges from the root node to the maximal tree, $k = 1, \ldots, K(n)$, the complexity parameter setting (default is

$cp = .01$ in `rpart`), and the minimal size of a terminal node t_n (default is minbucket/3 in `rpart`). In this example, the collection of single tree estimators would range over k with the complexity parameter and terminal node size held constant. In Part-DSA, s contains, k as the range of partitions ($k = 1, \ldots, K(n)$), as well as $k_0 = K(n)$ the maximum number of basis functions, k_1 the allowed complexity of the basis functions, and $minsplit$ the minimal number of observations to split a partition. Again, the collection of partitionings would range over k with $k_0, k_1,$ and $minsplit$ held constant.

Given $\hat{\Psi}_s^1(P_n)$, the collection of corresponding bagged estimators is defined as:

$$\hat{\Psi}_s(P_n) \equiv E_{P_n^{\#}|P_n} \hat{\Psi}_s^1(P_n),$$

where $P_n^{\#}$ given P_n is the empirical distribution of a bootstrap sample $O_1^{\#}, \ldots, O_n^{\#}$ of the empirical distribution P_n. And thus, $E_{P_{n\#}|P_n}$ denotes the expectation over many draws of bootstrap samples $P_{n\#}$ given P_n. For each of these draws, $P_{n1}^{\#}$, ..., $P_{nB}^{\#}$, (of size n), the estimators $\hat{\Psi}_s^1(P_{nb}^{\#})$, $b = 1, \ldots, B$, are calculated and averaged. Such that $\hat{\Psi}_s(P_n)$ can be written as:

$$\hat{\Psi}_s(P_n) = \lim_{B \to \infty} \frac{1}{B} \sum_{b=1}^{B} \hat{\Psi}_s^1(P_{nb}^{\#}).$$

This results in a sequence of candidate estimators $\hat{\Psi}_s(P_n)$ indexed over s. Our goal is to data adaptively select the s which minimizes the risk of $\hat{\Psi}_s(P_n)$ over $\mathcal{A}_n$. As such, we propose the *cross-validated bagged estimator* defined as:

$$\hat{\Psi}(P_n) = \hat{\Psi}_{\hat{S}(P_n)}(P_n),$$

where $\hat{S}(P_n)$ is the cross-validation selector corresponding with a cross-validation scheme defined by a random n vector $\mathcal{B}_n \in \{0,1\}^n$. A realization of $\mathcal{B}_n = (\mathcal{B}_{n,1}, \ldots, \mathcal{B}_{n,n})$ defines a particular split of the learning sample of n observations into a training set, $\{i \in \{1, \ldots, n\} : \mathcal{B}_{n,i} = 0\}$, and a validation set, $\{i \in \{1, \ldots, n\} : \mathcal{B}_{n,i} = 1\}$. The proportion of observations in the validation set is p. The empirical distributions of the training and validation sets are denoted by $P_{n,\mathcal{B}_n}^0$ and $P_{n,\mathcal{B}_n}^1$, respectively. The cross-validation selector is written as:

$$\hat{S}(P_n) = \arg\min_{s \in \mathcal{A}_n} E_{\mathcal{B}_n P_{n,\mathcal{B}_n}^1} L[\cdot, \hat{\Psi}_s(P_{n,\mathcal{B}_n}^0)]. \quad (2)$$

To calculate this selector of s, for each possible realization of $\mathcal{B}_n$ and $s \in \mathcal{A}_n$, B bootstrap samples of size $n(1-p)$ are drawn from the training sample $P^0_{n,\mathcal{B}_n}$. For each the B corresponding s-specific estimators $\hat{\Psi}^1_s(P^{0,\#}_{n,\mathcal{B}_n,b})$ are calculated and averaged to obtain:

$$\hat{\Psi}_s(P^0_{n,\mathcal{B}_n}) = \lim_{B\to\infty} \frac{1}{B}\sum_{b=1}^{B} \hat{\Psi}^1_s(P^{0\#}_{n\mathcal{B}_n b}).$$

The *cross-validated bagged estimator* is defined as the one which minimizes the risk as evaluated by the validation sample,$P^1_{n,\mathcal{B}_n}$, as in Equation 2.

In Breiman's VFOLD approach, the selection of s via cross-validation is performed within each bootstrap sample. Subsequently the B bootstrap specific estimators are averaged to arrive at the final estimator. As such, the cross-validation selector is defined as

$$\hat{S}_{br_V}(P^{\#}_n) = \arg\min_{s\in\mathcal{A}_n} E^{1,\#}_{\mathcal{B}_n,P_{n,\mathcal{B}_n}} L[\cdot, \hat{\Psi}^1_s(P^{0,\#}_{n,\mathcal{B}_n})].$$

The B bootstrap specific estimators are written as $\hat{\Psi}_{CV}(P^{\#}_n) = \hat{\Psi}_{\hat{S}^1_{br}(P^{\#}_n)}(P^{\#}_n)$. While the final estimator is the average of these B bootstrap specific estimators:

$$\tilde{\Psi}_{br_V}(P_n) = E_{P^{\#}_n|P_n} \hat{\Psi}_{CV}(P^{\#}_n).$$

It is our belief that Breiman's VFOLD approach provides the right selection among the estimators $\hat{\Psi}^1_s$, $s \in \mathcal{A}_n$ by trading off bias and variance. However, it may result in bagged estimators $\hat{\Psi}_s$, $s \in \mathcal{A}_n$, with poor performance. This is due to the decreased variance and the possibly increased bias of the bagged estimators. The increase in bias is due to two, possibly cumulative, sources: the bias introduced by applying an estimator to a bootstrap sample relative to the empirical sample; and the bias introduced by applying the estimator to the empirical sample relative to the truth.

In Breiman's Full Tree approach there is no selection of s within each bootstrap sample, instead a full tree $\hat{\Psi}^1_s(P^{\#}_n)$, where $k = K(n) \in s$, is grown on each of the B bootstrap samples. The B bootstrap specific estimators can be written as $\hat{\Psi}_{s_{k=K(n)}}(P^{\#}_n)$. The final estimator is the average of these B bootstrap specific estimators:

$$\tilde{\Psi}_{br_{FT}}(P_n) = E_{P^{\#}_n|P_n} \hat{\Psi}_{s_{k=K(n)}}(P^{\#}_n).$$

In Breiman's Test Tree approach the selection of s is determined by minimizing the learning set's prediction error on the bootstrap sample's tree. Subsequently, the B bootstrap specific estimators are averaged to construct the final estimator. The Test Tree selector is defined as

$$\hat{S}_{br_{TT}}(P_n^{\#}) = \arg\min_{s \in \mathcal{A}_n} E_{P_n} L[\cdot, \hat{\Psi}_s^1(P_n^{\#})].$$

The B bootstrap specific estimators are written as $\hat{\Psi}_{TT}(P_n^{\#}) = \hat{\Psi}_{\hat{S}_{br_{TT}}(P_n^{\#})}(P_n^{\#})$. While the final estimator is the average of these B bootstrap specific estimators:

$$\tilde{\Psi}_{br_{TT}}(P_n) = E_{P_n^{\#}|P_n} \hat{\Psi}_{TT}(P_n^{\#}).$$

Let $d(\psi, \psi_0) = E_0 L(o, \psi) dP_0(o) - E_0 L(o, \psi_0) dP_0(o)$ denote the risk dissimilarity. The results on the cross-validation selector (see van der Laan et al. [2003]) imply that under reasonable general conditions the cross-validated bagged estimator performs as well as the oracle selected bagged estimator $\hat{\Psi}_{\tilde{S}_{n(1-p)}(P_n)}(P_n)$, where $\tilde{S}_{n(1-p)}(P_n) = \arg\min_s E_{\mathcal{B}} E_0 L(o, \hat{\Psi}_s(P^0_{n,\mathcal{B}_n})) dP_0(o)$. $\tilde{S}_{n(1-p)}(P_n)$ selects the bagged estimator (based on $n(1-p)$ observations) closest to the truth w.r.t. to the risk dissimilarity. As a consequence, it is of interest to understand the relation between risk dissimilarity of $\hat{\Psi}_s^1(P_n)$ and the risk dissimilarity of the corresponding bagged estimator $\hat{\Psi}_s(P_n)$. This would immediately imply asymptotic consistency results for our proposed cross-validated bagged estimator.

2.3 Censored Data

In Section 2, the observed data structure resembles the full data structure in that neither the outcome nor the covariates are missing. In both Molinaro et al. [2004] and Molinaro and van der Laan [2004a] we have addressed how to accommodate censored observations using loss-based piecewise constant estimation. The exact same approach is applicable here using either the inverse probability censoring weighted (IPCW) or doubly robust IPCW.

2.4 Variable Importance

As most available variables are included in the bagged linear regression fit, we can comprehensively assess variable importance. One such measure

is the partial derivative with respect to each variable, averaged over all observed covariate values.

Given the collection of d baseline covariates $\mathbf{W}$, we can define the following function of $(\mathbf{W}, w, j)$:

$$\mathbf{W}_{-j}(w) \equiv (W_1, \ldots, W_{j-1}, W_j = w, W_{j+1}, \ldots, W_d).$$

Then for an ordered categorical or binary variable $W_j \in \{W_0, \ldots, W_{k_j}\}$, we can define the variable importance of W_j at each value of W_j as:

$$\nu_{j,k}(P_0) = E_{W_{-j}}[\Psi_0(W_{-j}(w_k)) - \Psi_0(W_{-j}(w_{k-1}))], \qquad \text{for } k = 1, \ldots, k_j.$$

The variable importance measure of each value of W_j can be evaluated, i.e., $(\nu_{j,1}(P_0), \ldots, \nu_{j,k_j}(P_0))$, and graphed to illustrate any fluctuations in the importance of W_j at the different values. An overall measure of importance for each variable can also be given by averaging over the observed covariate values, that is:

$$\nu_j(P_0) = \frac{1}{k_j} \sum_{k=1}^{k_j} E_{W_{-j}}[\Psi_0(W_{-j}(w_k)) - \Psi_0(W_{-j}(w_{k-1}))]$$

This can be estimated with the empirical distribution by calculating:

$$\hat{\nu}_j = \frac{1}{k_j} \sum_{k=1}^{k_j} \left[\frac{1}{n} \sum_{i=1}^{n} \hat{\Psi}(P_n)(W_{-j}(w_k)) - \hat{\Psi}(P_n)(W_{-j}(w_{k-1})) \right].$$

Given a continuous variable we can dichotomize it into k_j bins and use this approach. Alternatively, with a continuous variable we can take the derivative of $\Psi_0(W)$ evaluated at $W_j = w$:

$$\Psi_0^{(j)}(W_{-j}(w)) \equiv \frac{d}{dW_j} \Psi_0(w) \mid_{W_j = w}$$

And define the variable importance measure over the observed covariate values as:

$$\nu_j(P_0) = E_{W_{-j}} \Psi_0^{(j)}(W_{-j}(w)).$$

The values of $\nu_j(\cdot)$ can be plotted over the observed values of W_j to observe any fluctuations in importance. Pairwise variable importance can be measured by evaluating ν_j within ν_i, where $i \neq j$.

3 Simulations

To compare our proposed cross-validated bagged estimator to that of Breiman's we performed similar simulations to those in the original manuscript [Breiman, 1996]. For the simulations we used CART as implemented in the recursive partitioning algorithm `rpart` [Therneau and Atkinson, 1997] in the statistical package `R` [Ihaka and Gentlemen, 1996].

For each simulation, we evaluated estimators for a single tree, Breiman's suggested schemes, and our proposed method. For each, the same learning set L_{set} was used for building and choosing the estimator. The same independent test set T_{set} evaluated the fit and reflects the empirical risk estimated in the following tables.

For the single tree, $\hat{\Psi}_s^1(L_{set})$ was estimated with L_{set}, where $s = (k = 1, \ldots, K(n), minbucket = 7, cp = .01)$. The best level of the tree $\hat{k}$ was chosen via 10-fold cross-validation. The empirical risk was evaluated with the independent test set at level $\hat{k}$. For Breiman's VFOLD bagging scheme, the learning set L_{set} was used to generate B bootstrap samples and build B bootstrap specific estimators $\hat{\Psi}_{CV}(L_{set}^{\#})$, where 10-fold cross validation was implemented to select $\hat{S}_{br}(L_{set}^{\#})$. The predicted test set T_{set} values $\tilde{\Psi}_{br_V}(T_{set})$ were calculated by averaging the B bootstrap specific estimators built on L_{set}. For Breiman's Full Tree bagging scheme, the learning set L_{set} was used to generate B bootstrap samples and build B bootstrap specific estimators $\hat{\Psi}_{s_{k=K(n)}}(P_n^{\#})$. The predicted test set T_{set} values $\tilde{\Psi}_{br_{FT}}(T_{set})$ were calculated by averaging the B bootstrap specific estimators built on L_{set}. In Breiman's Test Tree approach the learning set L_{set} was used to generate B bootstrap samples and build B bootstrap specific estimators $\hat{\Psi}_{CV}(L_{set}^{\#})$, where the entire learning set was used to select $\hat{S}_{br_{TT}}(L_{set}^{\#})$. The predicted test set T_{set} values $\tilde{\Psi}_{br_{TT}}(T_{set})$ were calculated by averaging the B bootstrap specific estimators built on L_{set}.

For our proposed cross-validated bagging scheme, L_{set} was used to generate B bootstrap samples and 10-fold cross-validation to select $\hat{k}$. The empirical risk is calculated with the averaged bootstrap specific estimators, $\hat{\Psi}_s(T_{set})$.

This entire procedure was repeated 100 times for each of the methods and the empirical risk averaged over the 100 repetitions.

Table 1: Simulation F1

Method	Bootstrap Samples	Emp Risk Mean	Emp Risk Std.Dev.	% improvement in Risk
Single Tree	0	21.39	3.19	
Breiman VFOLD	25	15.67	1.34	27%
Test Tree	25	13.95	0.92	35%
Full Tree	25	13.89	0.91	35%
Our^T Bagging	25	14.02	1.02	34%
Single Tree	0	20.83	3.07	
Breiman VFOLD	100	15.63	1.37	25%
Test Tree	100	13.74	0.95	34%
Full Tree	100	13.71	0.95	34%
Our^T Bagging	100	13.75	1.01	34%
Single Tree	0	21.18	3.37	
Breiman VFOLD	1000	15.54	1.32	27%
Test Tree	1000	13.68	0.95	35%
Full Tree	1000	13.61	.94	36%
Our^T Bagging	1000	13.64	.97	36%

3.1 Simulation 1

This simulation is from Friedman's simulations in the MARS paper [Friedman, 1991] and also implemented in Breiman [1996]. There are 10 independent predictor variables $x_1, \ldots, x_{10}$ each of which is uniformly distributed over $(0, 1)$. The response is given by

$$y = 10sin(\pi x_1 x_2) + 20(x_3 - .5)^2 + 10x_4 + 5x_5 + error,$$

where the $error \sim N(0, 1)$. For this the learning set L_{SET} has 200 observations and the test set T_{SET} has 1000.

The procedures for the single tree, Breiman's VFOLD, Test Set, and Full Tree, and our proposed bagging (details above) were repeated 100 times. The results are shown in Table 1.

3.2 Simulation 2

This simulation is from Friedman's simulations in the MARS paper [Friedman, 1991] and also implemented in Breiman [1996]. There are 4 independent predictor variables $x_1, \ldots, x_4$ each of which is uniformly distributed over different ranges:

$$\begin{aligned} 0 &\leq x_1 \leq 100 \\ 20 &\leq (x_2/2\pi) \leq 280 \\ 0 &\leq x_3 \leq 1 \\ 1 &\leq x_4 \leq 11 \end{aligned}$$

The response is given by

$$y = (x_1^2 + (x_2 x_3 - (1/x_2 x_4))^2)^{1/2} + error,$$

where the $error \sim N(0, \sigma)$. We compared 0.35 and $.862$ as values for σ. For this the learning set L_{SET} has 200 observations and the test set T_{SET} has 1000.

The procedures for the single tree, Breiman's VFOLD, Test Set, and Full Tree, and our proposed bagging (details above) were repeated 100 times. The results are shown in Table 2.

3.3 Simulation 3

This simulation is from Friedman's simulations in the MARS paper [Friedman, 1991] and also implemented in Breiman [1996]. There are 4 independent predictor variables $x_1, \ldots, x_4$ each of which is uniformly distributed over different ranges:

$$\begin{aligned} 0 &\leq x_1 \leq 100 \\ 20 &\leq (x_2/2\pi) \leq 280 \\ 0 &\leq x_3 \leq 1 \\ 1 &\leq x_4 \leq 11 \end{aligned}$$

The response is given by

$$y = \arctan\left(\frac{x_2 x_3 - (1/x_2 x_4)}{x_1}\right) + error,$$

Table 2: Simulation F2

σ	Method	Bootstrap Samples	Emp Risk Mean	Emp Risk Std.Dev.	% improvement over single tree
	Single Tree	0	14529.2	3546.97	
	Breiman VFOLD	25	6638.60	1278.63	56%
	Test Tree	25	5250.65	1013.96	65%
	Full Tree	25	5188.31	1020.34	65%
	Our^T Bagging	25	5185.77	1040.5	64%
	Single Tree	0	15044	4697.57	
	Breiman VFOLD	100	6248.02	1196.94	58%
.35	Test Tree	100	4957.21	958.82	67%
	Full Tree	100	4871.85	1011.65	68%
	Our^T Bagging	100	4870.32	954.91	68%
	Single Tree	0	15015.85	4562.50	
	Breiman VFOLD	1000	6145.56	1134.76	59%
	Test Tree	1000	4835.71	965.72	68%
	Full Tree	1000	4787.36	952.69	68%
	Our^T Bagging	1000	4771.76	948.49	68%
	Single Tree	0	14787.81	4488.39	
	Breiman VFOLD	25	6650.93	1269.7	55%
	Test Tree	25	5254.50	1016.85	64%
	Full Tree	25	5191.34	1023.36	65%
	Our^T Bagging	25	5180.04	1035.8	65%
	Single Tree	0	15055.41	4709.78	
	Breiman VFOLD	100	6283.21	1177.67	58%
.62	Test Tree	100	4944.95	975.27	67%
	Full Tree	100	4883.19	978.73	67%
	Our^T Bagging	100	4871.83	964.94	68%
	Single Tree	0	15071.4	4569.87	
	Breiman VFOLD	1000	6127.12	1144.38	59%
	Test Tree	1000	4836.26	965.55	68%
	Full Tree	1000	4774.79	966.83	68%
	Our^T Bagging	1000	4770.82	949.19	68%

where the $error \sim N(0, \sigma)$. We compared 0.35 and $.86$ as values for σ. For this the learning set L_{SET} has 200 observations and the test set T_{SET} has 1000.

The procedures for the single tree, Breiman's VFOLD, Breiman's Test Set, Full Tree, and our proposed bagging (as explained above) were repeated 100 times. The results are shown in Table 3.

3.4 Simulation 4

To understand the ramifications of not using honest cross-validation or any cross-validation at all, a simulation study with only one covariate was studied. Histogram regression necessitates a medium to small tree for adequate prediction. As such, overfitting will not be appropriate. In Table 4, the full data distribution was simulated from $y = x^2 + er$, where $x \sim N(0, 1)$ and $er \sim N(0, .25)$. For each of 100 repetitions a training set of size 200 was used to build a classifier and an independent test sample of 1000 to assess the empirical risk. The mean of the risk for each of the 100 repetitions is reported in the table along with the standard deviation.

4 Data Analysis

4.1 Boston Housing Data

In the Boston Housing dataset (available from the MASS library in `R` (Venables and Ripley [2002])) socio-economic variables are used to predict median value of houses in Boston housing tracts. There are 506 observations and 14 variables. For the following an independent test set T_{SET} of size 25 was randomly selected. This is a replication of simulations in Breiman [1994]. The learning set L_{SET} contains the 481 remaining observations. The results of 100 simulations are shown in Table 5.

Using the variable importance overall measure as described in Section 2.4, the socio-economic predictors are ranked in increasing importance in Table 6. From this ranking, the number of rooms (`rm`) is decidedly the most important variable. To further investigate any fluctuations in importance of this variable we can plot the values of $\nu_{\texttt{rm}}(\cdot)$ over the observed values of `rm` (Figure 1). In this exercise `rm` split was into 10 bins.

Table 3: Simulation F3

σ	Method	Bootstrap Samples	Emp Risk Mean	Emp Risk Std.Dev.	% improvement over single tree
	Single Tree	0	0.1903	0.0137	
	Breiman VFOLD	25	0.1616	0.0092	15%
	Test Tree	25	0.1534	0.008	19%
	Full Tree	25	0.1543	0.008	19%
	Our^T Bagging	25	0.1516	0.0091	20%
	Single Tree	0	0.1891	0.0135	
	Breiman VFOLD	100	0.1 603	0.009	15%
.35	Test Tree	100	0.1513	0.008	20%
	Full Tree	100	0.1521	0.0079	20%
	Our^T Bagging	100	0.1511	0.0084	20%
	Single Tree	0	0.1876	0.0123	
	Breiman VFOLD	1000	0.1599	0.0088	15%
	Test Tree	1000	0.1507	0.0076	20%
	Full Tree	1000	0.1514	0.0076	19%
	Our^T Bagging	1000	0.1504	0.0080	20%
	Single Tree	0	0.8374	0.0353	
	Breiman VFOLD	25	0.8037	0.0349	4%
	Test Tree	25	0.8209	0.0385	2%
	Full Tree	25	0.8328	0.0397	0.6%
	Our^T Bagging	25	0.7987	0.0379	5%
	Single Tree	0	0.8377	0.0365	
	Breiman VFOLD	100	0.8005	0.0354	4%
.86	Test Tree	100	0.8122	0.0381	3%
	Breiman Full Tree	100	0.8217	0.0391	2%
	Our^T Bagging	100	0.7951	0.0357	5%
	Single Tree	0	0.8386	0.0365	
	Breiman VFOLD	1000	0.7995	0.0354	5%
	Test Tree	1000	0.8091	0.0375	3.5%
	Full Tree	1000	0.8185	0.03863	2%
	Our^T Bagging	1000	0.7941	0.0351	5%

Table 4: Histogram Regression Simulation

Method	Bootstrap Samples	Emp Risk Mean	Emp Risk Std.Dev.	% improvement over single tree
Single Tree	0	0.2767	0.0144	
Breiman VFOLD	25	0.2625	0.0123	5%
Test Tree	25	0.353	0.0238	-28%
Full Tree	25	0.3625	0.0228	-31%
Our^T Bagging	25	0.2612	0.0122	6%
Single Tree	0	0.2773	0.0159	
Breiman VFOLD	100	0.2613	0.0125	6%
Test Tree	100	0.3476	0.0214	-25%
Full Tree	100	0.3576	0.0207	-29%
Our^T Bagging	100	0.2606	0.0127	6%
Single Tree	0	0.2767	0.0149	
Breiman VFOLD	1000	0.2617	0.0123	6%
Test Tree	1000	0.3465	0.0215	-25%
Full Tree	1000	0.3566	0.0208	-28%
Our^T Bagging	1000	0.2606	0.0127	6%

Table 5: Boston Housing Data

Method	Bootstrap Samples	Emp Risk Mean	Emp Risk Std.Dev.	% improvement over single tree
Single Tree	0	23.61	15.77	
Breiman VFOLD	25	15.93	11.45	31%
Test Tree	25	13.14	10.28	43%
Full Tree	25	13.13	10.31	43%
Our^T Bagging	25	12.92	10.15	44%
Single Tree	100	22.93	14.78	
Breiman VFOLD	100	15.67	11.29	31.7%
Test Tree	100	12.91	10.07	43.8%
Full Tree	100	12.9	10.1	43.7%
Our^T Bagging	100	12.81	9.999	44.2%

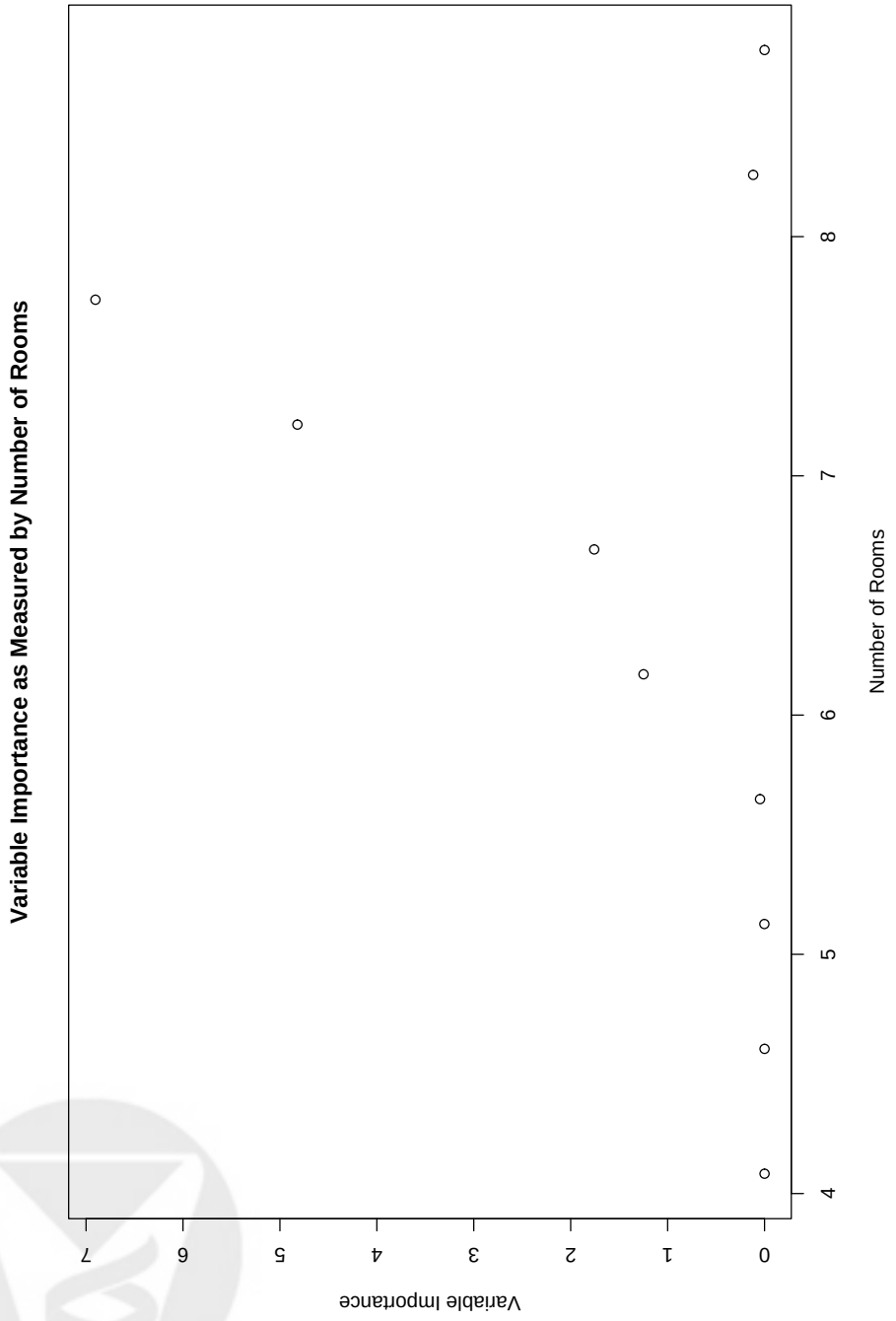

Figure 1: *Variable importance over value of RM.*

Table 6: Variable Overall Importance Measure for Boston Housing Data

Variable	Importance Measure
zn	0.0073
rad	0.0337
indus	0.0506
black	0.0572
chas	0.0702
age	0.1181
ptratio	0.1378
tax	0.1634
nox	0.1754
crim	0.3648
dis	0.4299
lstat	1.5615
rm	2.2151

5 Summary

We have presented a cross-validated bagging scheme in the context of partitioning algorithms. This method differs from previously suggested methods by implementing an 'external' cross-validation for estimator selection. We compared our method to the previous ones and a single regression tree via simulations and a data analysis. In the first three simulations over-fitting is appropriate, as such, ours, the Full Tree and Test Tree methods do equivalently well. This illustrates that VFOLD is restricted by the use of cross-validation within the bootstrap samples. This is true in all but one case where we increased the variance in Simulation 3.3. There a more conservative tree is beneficial and we note that our method, closely followed by the VFOLD method, acts appropriately.

In Simulation 3.4 we investigated the different approaches in the context of histogram regression. Histogram regression necessitates a medium to small tree for adequate prediction. As such, over-fitting is not appropriate. This can best be seen in Table 4, where the Full Tree and Test Set methods perform very poorly. Our method and VFOLD do well in restricting the tree size. This simulation illustrates the fact that dishonest or

no cross-validation can limit the user to only over-fitting.

In Section 4, the Boston Housing data was explored with all four methods as well as a single regression tree. We see in Table 5 that over-fitting is favored and as such our method, Test Set and Full Tree do well. The variable importance measure introduced in Section 2.4, is illustrated with the Boston Housing data. In this example we see that 'rm', or number of rooms, is designated as the most important variable. In Figure 1, the values of 'rm' are shown with their corresponding variable importance measure. Here it is apparent that between six and eight rooms contributes most to the prediction ability of the bagged trees.

References

L. Breiman, J. H. Friedman, R.A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth and Brooks/Cole, Monterey, CA, 1984.

Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996. URL citeseer.ist.psu.edu/breiman96bagging.html.

Leo Breiman. Bagging predictors. Technical Report 421, 1994. URL www.stat.berkeley.edu/~breiman.

Leo Breiman. Using adaptive bagging to debias regressions. Technical Report 547, 1999. URL www.stat.berkeley.edu/~breiman.

J.H. Friedman. Multivariate adaptive regression splines. *Annals of Statistics*, 19(1):1–141, 1991.

W. Härdle. *Applied nonparametric regression*. Number 17 in Econometric Society Monographs. Cambridge University Press, 1989.

R. Ihaka and R. Gentlemen. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5:299–314, 1996.

A. M. Molinaro and M. J. van der Laan. Deletion/substitution/addition algorithm for partitioning the covariate space in prediction. Technical Report 162, Division of Biostatistics, University of California, Berkeley, 2004a. URL www.bepress.com/ucbbiostat/paper162/.

A. M. Molinaro and M. J. van der Laan. A new partitioning algorithm for prediction of survival outcomes: Illustration with histogram regression. Statistical Computing Section[CD-ROM], Alexandria, VA, 2004b. Proceedings of the American Statistical Association.

A. M. Molinaro, S. Dudoit, and M. J. van der Laan. Tree-based multivariate regression and density estimation based on right-censored data. *Journal of Multivariate Analysis*, 90:154–177, 2004.

T. Therneau and E. Atkinson. An introduction to recursive partitioning using the rpart routine. Technical Report 61, Section of Biostatistics, Mayo Clinic, Rochester, 1997.

M. J. van der Laan and S. Dudoit. Unified cross-validation methods for selection among estimators: Finite sample results, asymptotic optimality, and applications. Technical Report 130, Division of Biostatistics, University of California, Berkeley, 2003. URL www.bepress.com/ucbbiostat/paper130/.

M. J. van der Laan, S. Dudoit, and S. Keleş. Asymptotic optimality of likelihood based cross-validation. Technical Report 125, Division of Biostatistics, University of California, Berkeley, 2003. URL www.bepress.com/ucbbiostat/paper125/.

W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, fourth edition, 2002.