Variable Selection with Bayesian Additive Regression Trees

Carlos Carvalho, Edward George, Richard Hahn, and Robert McCulloch *

October, 2020

Abstract

Bayesian Additive Regression Trees (BART) has emerged as a highly effective Bayesian approach to ensemble modeling with many binary trees. The BART Markov Chain Monte Carlo (MCMC) algorithm provides effective stochastic search in a complex model space and Bayesian uncertainty. As is the case with many modern approaches, the overall complexity of the model makes interpretation difficult. In practice, investigators often wish to know what predictor variables are important or, more generally, what roles variables play in the model. In this paper we review some approaches for understanding how variables enter the BART model. We present simple ways to find out what variables are important, what pairs of variables interact in the model, and what subsets of variables allow us to approximate the full information information inference according to a user defined metric. In all cases, our approach is based on post processing of the output from basic BART modeling and uncertainty is captured naturally by the usual MCMC variation in a straightforward way.

*Carlos M. Carvalho is Professor of Statistics, McCombs School of Business, The University of Texas at Austin. Edward I. George is Professor of Statistics, Department of Statistics, The Wharton School, University of Pennsylvania, 3730 Walnut St, 400 JMHH, Philadelphia, PA 19104-6304, edgeorge@wharton.upenn.edu. Richard Hahn is Associate Professor of Statistics, The School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ, 85281, prhahn@asu.edu. Robert McCulloch is Professor of Statistics, The School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ, 85281, robert.mcculloch@asu.edu.

Contents

1 Introduction

Modern statistical methods have, to a remarkable extent, advanced our ability to uncover complex, high dimensional relationships. In particular, for directed problems in which we predict y from $x = (x_1, \ldots, x_p)$, methods based on ensembles of trees have performed amazingly well in practice. Although the inner workings of these "black box" models are necessarily complex and somewhat intimidating, these ensembles are particularly well suited for variable selection because of their flexible nonparametric nature. In contrast to popular parametric approaches such as normal linear regression, where variable selection is tantamount to submodel selection, the unrestricted nature of ensembles of trees allows for a richer set of possibilities for discovering the potential relatedness of y and a subset of the predictors.

Bayesian Additive Regression Trees (BART, [3]) is a Bayesian approach to ensemble tree modeling which has proven to be remarkably effective for prediction and inference, often with minimal tuning. In this chapter, we show how the output of BART is particularly congenial for the problem of variable selection and the problem of interaction detection, itself a form of structured variable selection. The essential idea is to simply to assess the relative frequency with which variables and their interactions appear in the tree components of the ensemble.

2 BART Overview

In this section we review the essentials of the BART proposed by [3], hereafter GGM10. In a nutshell, BART is a fully Bayesian approach for modeling the regression y on $x = (x_1, \ldots, x_p)$ with a flexible sum-of-trees model of the form

$$Y = \sum_{j=1}^{m} g(x; T_j, M_j) + \epsilon, \qquad \epsilon \sim N(0, \sigma^2). \tag{1}$$

Here, each T_j is a recursive binary regression tree associated with a set M_j of terminal node constants μ_{ij}, for which $g(x; T_j, M_j)$ is the step function which assigns $\mu_{ij} \in M_j$ to x according to the sequence of decision rules in T_j. These decision rules are binary splits of the predictor space of the form $\{x \in A\}$ vs $\{x \notin A\}$ where A is a subset of the range of x. When the number of trees $m = 1$, (1) reduces to the single tree model used by [4], hereafter CGM98, for Bayesian CART.

For each value of x, under (1), $E(Y \mid x)$ is equal to the sum of all the terminal node μ_{ij}'s assigned to x by the $g(x; T_j, M_j)$'s. Thus, the sum-of-trees function is flexibly capable of approximating a wide class of functions from R^n to R, especially when the number of trees m is large. Note also that the sum-of-trees representation is simply the sum of many simple multidimensional step functions from R^n to R, namely the $g(x; T_j, M_j)$, rendering it much more manageable than a basis expansion with more complicated elements such as multidimensional wavelets or multidimensional splines.

The BART model specification is completed by introducing a prior distribution over all the parameters of the sum-of-trees model, namely $(T_1, M_1), \ldots, (T_m, M_m)$ and σ. Note that

$(T_1, M_1), \ldots, (T_m, M_m)$ entail all the bottom node parameters as well as the tree structures and decision rules, a very large number of parameters, especially when m is large. To cope with this parameter explosion, we use a "regularization" prior that effectively constrains the fit by keeping each of the individual tree effects from being unduly influential. Without such a regularizing influence, large tree components would overwhelm the rich structure of (1), thereby limiting its scope of fine structure approximation.

2.1 Specification of the BART Regularization Prior

To simplify the specification of this regularization prior, we restrict attention to independence priors of the form

$$p((T_1, M_1), \ldots, (T_m, M_m), \sigma) = \left[\prod_j \left(\prod_i p(\mu_{ij} \mid T_j) \right) p(T_j) \right] p(\sigma), \tag{2}$$

where $\mu_{ij} \in M_j$, there by reducing prior specification to the choice of prior forms for $p(T_j), p(\mu_{ij} \mid T_j)$ and $p(\sigma)$. To simplify matters further we use identical prior forms for every $p(T_j)$ and for every $p(\mu_{ij} \mid T_j)$. As detailed below, each of these prior forms are controlled by just a few interpretable hyperparameters that can be calibrated to yield surprisingly effective default specifications for regularization of the sum-of-trees model.

For $p(T_j)$, we use the sequential tree-generating process which is specified by three aspects: (i) the probability that a node at depth d $(= 0, 1, 2, \ldots)$ is nonterminal, given by

$$\alpha(1 + d)^{-\beta}, \qquad \alpha \in (0, 1), \beta \in [0, \infty), \tag{3}$$

(ii) the distribution on the splitting variable assignments at each interior node, and (iii) the distribution on the splitting rule assignment in each interior node, conditional on the selected splitting variable. As default choices, CGGM 98 and CGM10 recommend $\alpha = 0.95$ and $\beta = 2$ for (1), and a uniform priors on each set of possibilities for (ii) and (iii). Interesting alternative enhancements of these choice for $p(T_j)$ have been proposed by Linero [8], Rockova and van der Paas [10] and Rockova and Saha [9].

For $p(\mu_{ij} \mid T_j)$, we use the conjugate normal distribution $N(\mu_\mu, \sigma_\mu^2)$ which allows μ_{ij} to be margined out, vastly simplifying MCMC posterior calculations. To guide the specification of the hyperparameters μ_μ and σ_μ, we note that under (1), it is highly probable that $E(Y \mid x)$ lies between y_{min} and y_{max}, the observed minimum and maximum of y in the data, and that the prior distribution of $E(Y \mid x)$ is $N(m\,\mu_\mu, m\,\sigma_\mu^2)$, (because $E(Y \mid x)$ is the sum of m independent μ_{ij}'s under the sum-of-trees model). Based on these facts, we use the informal empirical Bayes strategy of choosing μ_μ and σ_μ so that $N(m\,\mu_\mu, m\,\sigma_\mu^2)$ assigns substantial probability to the interval (y_{min}, y_{max}). This is conveniently done by choosing μ_μ and σ_μ so that $m\,\mu_\mu - k\,\sqrt{m}\,\sigma_\mu = y_{min}$ and $m\,\mu_\mu + k\,\sqrt{m}\,\sigma_\mu = y_{max}$ for some preselected value of k such 1,2 or 3. For example, $k = 2$ would yield a 95% prior probability that $E(Y \mid x)$ is in the interval (y_{min}, y_{max}). The goal of this specification strategy for μ_μ and σ_μ is to ensure that the implicit prior for $E(Y \mid x)$ is in the right "ballpark" in the sense of assigning

substantial probability to the entire region of plausible values of $E(Y \mid x)$ while avoiding overconcentration and overdispersion of the prior with respect to the likelihood. As long as this goal is met, BART seems to be very robust to the variations of these specifications.

For $p(\sigma)$, we also use a conjugate prior, here the inverse chi-square distribution $\sigma^2 \sim \nu\,\lambda/\chi^2_\nu$. Here again, we use a informal empirical Bayes approach, to guide the specification of the hyperparameters ν and λ, in this case to assign substantial probability to the entire region of plausible values of σ while avoiding overconcentration and overdispersion of the prior. Essentially, we calibrate the prior df ν and scale λ using a "rough data-based overestimate" $\hat{\sigma}$ of σ. Two natural choices of $\hat{\sigma}$ are 1) a "naive" specification, the sample standard deviation of Y, or 2) a "linear model" specification, the residual standard deviation from a least squares linear regression of Y on all the predictors. We then pick a value of ν between 3 and 10 to get an appropriate shape, and a value of λ so that the qth quantile of the prior on σ is located at $\hat{\sigma}$, that is $P(\sigma < \hat{\sigma}) = q$. We consider values of q such as 0.75, 0.90 or 0.99 to center the distribution below $\hat{\sigma}$.

2.2 Posterior Calculation and Information Extraction

Combing the regulation prior with the likelihood, $L((T_1, M_1), \ldots, (T_m, M_m), \sigma \mid y)$ induces a posterior distribution

$$p((T_1, M_1), \ldots, (T_m, M_m), \sigma \mid\ y) \tag{4}$$

over the full sum-of-trees model parameter space. Although analytically intractable, the following backfitting MCMC algorithm can be used to very effectively simulate samples from this posterior.

The algorithm is a Gibbs sampler at the outer level. Let $T_{(j)}$ be the set of all trees in the sum *except* T_j, and similarly define $M_{(j)}$, so that $T_{(j)}$ will be a set of $m-1$ trees, and $M_{(j)}$ the associated terminal node parameters. A Gibbs sampling strategy for sampling from (4) is obtained by m successive draws of (T_j, M_j) conditionally on $(T_{(j)}, M_{(j)}, \sigma)$:

$$(T_j, M_j) \mid T_{(j)}, M_{(j)}, \sigma, y, \tag{5}$$

$j = 1, \ldots, m$, followed by a draw of σ from the full conditional:

$$\sigma \mid T_1, \ldots T_m, M_1, \ldots, M_m, y. \tag{6}$$

The draw of σ in (6) is simply a draw from an inverse gamma distribution and so can be easily obtained by routine methods. More subtle is the implementation of the m draws of (T_j, M_j) in (5). This can be done by taking advantage of the following reductions. First, observe that the conditional distribution $p(T_j, M_j \mid T_{(j)}, M_{(j)}, \sigma, y)$ depends on $(T_{(j)}, M_{(j)}, y)$ only through

$$R_j \equiv y - \sum_{k \neq j} g(x; T_k, M_k), \tag{7}$$

the $n-$vector of partial residuals based on a fit that excludes the jth tree. Thus, the m draws of (T_j, M_j) given $(T_{(j)}, M_{(j)}, \sigma, y)$ in (5) are equivalent to m draws from

$$(T_j, M_j) | R_j, \sigma, \tag{8}$$

$j = 1, \ldots, m$. Because we have used a conjugate prior for M_j, $p(T_j|R_j, \sigma)$ can be obtained in closed form up to a norming constant. This allows us to carry out each draw from (8) in two successive steps as

$$T_j|R_j, \sigma \tag{9}$$

$$M_j|T_j, R_j, \sigma. \tag{10}$$

The draw of T_j in (9), although somewhat elaborate, can be obtained using the Metropolis-Hastings (MH) algorithm of CGM98. The draw of M_j in (10) is simply a set of independent draws of the terminal node μ_{ij}'s from a normal distribution. The draw of M_j enables the calculation of the subsequent residual R_{j+1} which is critical for the next draw of T_j.

We initialize the chain with m simple single node trees, and then iterations are repeated until satisfactory convergence is obtained. Fortunately, this backfitting MCMC algorithm appears to mix very well as we have found that different restarts give remarkably similar results even in difficult problems. At each iteration, each tree may increase or decrease the number of terminal nodes by one, or change one or two decision rules. The sum-of-trees model, with its abundance of unidentified parameters, allows for "fit" to be freely reallocated from one tree to another. Because each move makes only small incremental changes to the fit, we can imagine the algorithm as analogous to sculpting a complex figure by adding and subtracting small dabs of clay.

For inference based on our MCMC sample, we rely on the fact the our backfitting algorithm is ergotic. Thus, the induced sequence of sum-of-trees functions

$$f^*(\cdot) = \sum_{j=1}^{m} g(\cdot; T_j^*, M_j^*), \tag{11}$$

for the sequence of draws $(T_1^*, M_1^*), \ldots, (T_m^*, M_m^*)$, is converging to $p(f \,|\, y)$, the posterior distribution on the "true" $f(\cdot)$. Thus, by running the algorithm long enough after a suitable burn-in period, the sequence of f^* draws, say $f_1^*, \ldots, f_K^*$, may be regarded as an approximate, dependent sample of size K from $p(f \,|\, y)$. Bayesian inferential quantities of interest can then be approximated with this sample as indicated below.

To estimate $f(x)$ or predict Y at a particular x, in-sample or out-of-sample, a natural choice is the average of the after burn-in sample $f_1^*, \ldots, f_K^*$,

$$\frac{1}{K} \sum_{k=1}^{K} f_k^*(x), \tag{12}$$

which approximates the posterior mean $E(f(x) \,|\, y)$. Posterior uncertainty about $f(x)$ may be gauged by the variation of $f_1^*(x), \ldots, f_K^*(x)$. For example, a natural and convenient $(1-\alpha)\%$ posterior interval for $f(x)$ is obtained as the interval between the upper and lower $\alpha/2$ quantiles of $f_1^*(x), \ldots, f_K^*(x)$.

3 Model-Free Variable Selection with BART

In this section we describe and illustrate how variable selection may be accomplished by direct inspection of the basic BART output. The essential idea to select those components of x that tend to be used most often in the MCMC sequence of fitted sum-of-trees models, $f_1^*, \ldots, f_K^*$. To measure this tendency, we simply record for each model f_k^*, the proportion of splitting rules using each component of x, and then average these proportions across the whole sequence. Those components with the largest average usage proportions are then selected. The attribution of classical statistical significance levels to these proportions can be obtained with the permutation based methods proposed by Bleich et.al. [1].

While it makes intuitive sense that relevant predictors will tend to be used frequently in the fitted sum-of-trees ensemble, it turns out that when the number of trees m is large, the redundancy of the overly rich basis of trees will also allow for many irrelevant predictors to be mixed in with the relevant ones. Fortunately, as m is decreased, this redundancy is diminished, thereby forcing the BART output tends to more heavily favor relevant predictors for its fit. In a sense, when m is small the predictors compete with each other to improve the fit.

This direct BART approach to variable selection is thus accomplished by observing what happens to the x component usage frequencies in a sequence of MCMC samples $f_1^*, \ldots, f_K^*$ as the number of trees m is set smaller and smaller. More precisely, for each simulated sum-of-trees model f_k^*, let z_{ik} be the proportion of all splitting rules that use the ith component of x. Then

$$v_i \equiv \frac{1}{K} \sum_{k=1}^{K} z_{ik} \tag{13}$$

is the average use per splitting rule for the ith component of x. As m is set smaller and smaller, the sum-of-trees models tend to more strongly favor inclusion of those x components which improve prediction of y and exclusion of those x components that are unrelated to y. In effect, smaller m seems to create a bottleneck that forces the x components to compete for entry into the sum-of-trees model. As we illustrate below, the x components with the larger v_i's will then be those that provide the most information for predicting y.

3.1 Variable Selection with the Boston Housing Data

We illustrate this approach with the well known Boston housing data which is available in many data science environments (library MASS in R). There are $n = 506$ observations each of which corresponds to a neighborhood in the Boston area. The response of interest is $y =$ the median value of a house in thousands of dollars. For each neighborhood, thirteen characteristics have been measured. We refer the reader the R documentation for details but a few variables which will be important are (i) lstat: lower status of the population (percent), (ii) rm: average number of rooms per dwelling, and (iii) nox: nitrogen oxides concentration (parts per 10 million).

We ran BART for 10,000 burn in iterations and then 10,000 post burn in iterations. We used 20 trees in the BART ensemble. We used the implementation in the R library `BART` and all other BART parameters were left at default values. Figure 1 displays the draws of σ for all 20,000 BART MCMC iterations. While it does take all 10,000 initial draws to burn in, it does look like things have stabilized from that point on.

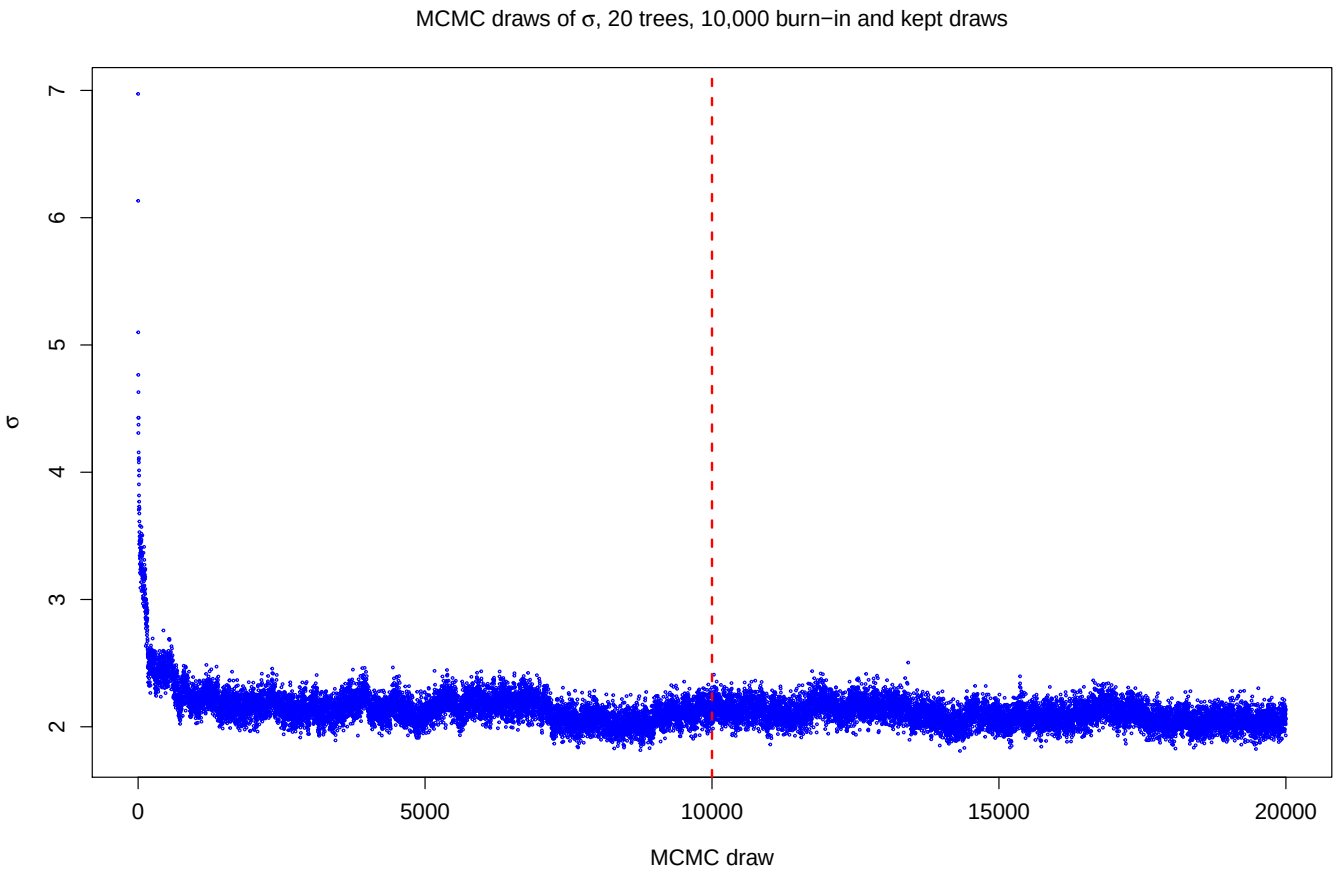

Figure 1: MCMC draws of σ. 20 trees. Vertical line draw at iteration 10,000 to indicate burn-in period.

Figure 2 displays the posterior distribution of percent usage for each variable. The red dot is at the posterior mean (average over MCMC draws) and the vertical solid (blue) line is a 90% posterior interval (5% and 95% quantiles from the MCMC draws). The results are quite clear. With high posterior probability, the variables nox, rm, and lstat are important. For all other variables there is considerable uncertainty.

In Figure 2 are results are based on BART modeling using 20 trees in the ensemble. The default value for the number of trees in the R package BART is 200. For predicive accurracy, a large number of trees often works best. However, for the simple variable selection approach, fewer tree may give a better understanding of the important variables. This is BART is

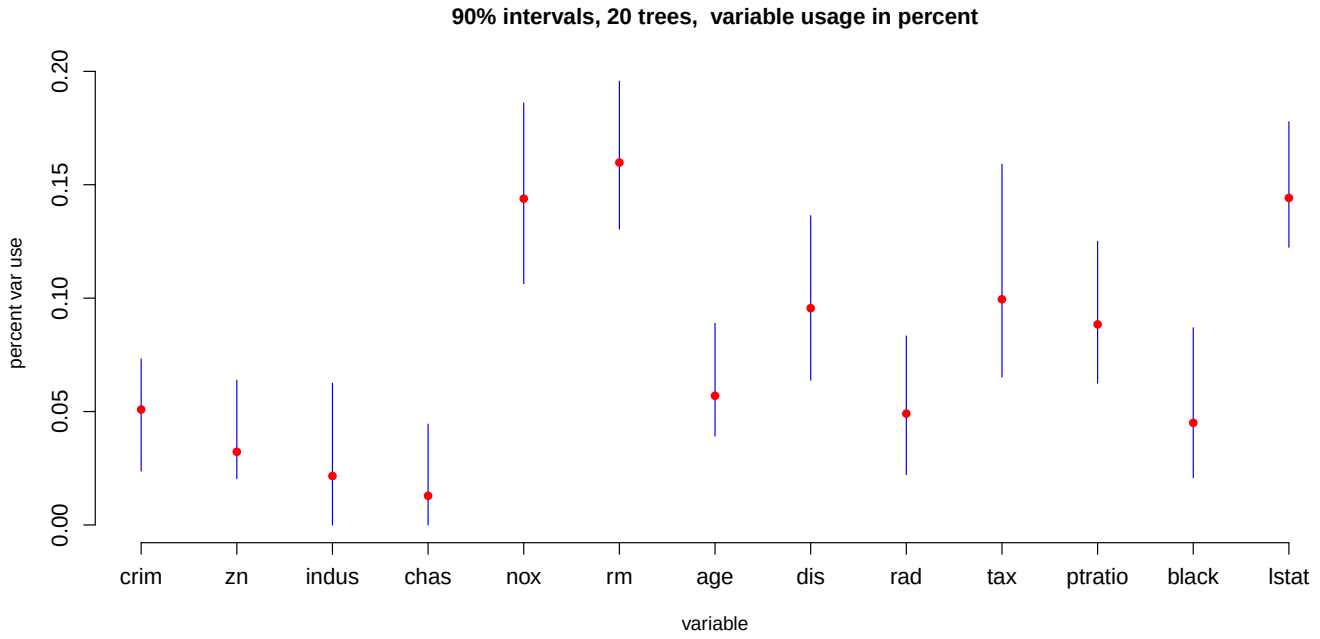

Figure 2: Posterior distributions of usage percentage for each variable. 20 trees. 90% intervals.

designed to only let a variable enter with "a little bit of fit" and the amount of fit gets smaller as we add trees. With a large number of trees, a variable may be used in several tree decision rules without really doing anything. With fewer trees, the focus is narrowed to variable that make an important contribution.

Figure 3 plots the posterior mean of the variable percent usage from our previous BART run with 20 trees and a BART run (same number burn-in and kept draws) with the default 200 trees. We can see that it would be very hard to pick out a promising variable subset using the 200 tree run.

The appeal of the simple percent usage approach is its simplicity. In some case, the use of fewer trees may degrade the fit. Figure 4 plots the response y=median price, the BART fit with 200 trees, the BART fit with 20 trees, and the fit from a linear model. In this case the BART default fit is very good and very similar to the 20 tree fit. The linear fit is dramatically worse. Of course, we could improve the fit by doing simple data analysis (e.g. log y) but the BART default immediately give appealing results.

In practice, as in the Boston example, it is often straightforward to find a number of trees which gives a fit similar to the default, but identifying variables. It is noteworthy that the percent usage approach can be effective without the explicit use of variable selection prior and by relying on just a simple measure of variable usage as opposed to how much its usage improves the loss (as in CART). Linero ([8]) discusses a very nice approach to adding prior information about sparcity to BART which works very well. The cost is another layer of

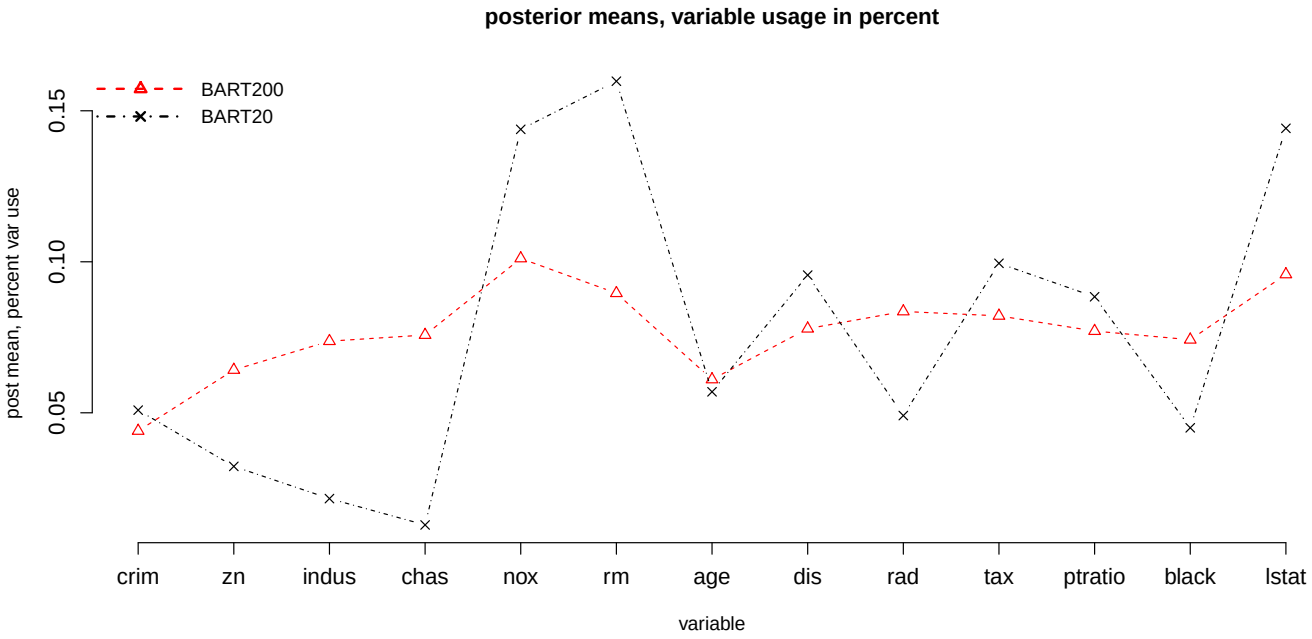

Figure 3: Posterior means of usage percentage for each variable. 20 and 200 trees.

prior comprehension and specification.

4 Model-Free Interaction Detection with BART

In a regression of y on x, an interaction between two predictors x_i and x_j is said to exist when the effect of x_i on y depends on the value of x_j, and vice-versa. Such an interaction is manifested in a sum-of-trees model when the x_i and x_j components of x tend to both be used as splitting rules in common trees. Thus, analogously to the previously described percentage usage approach for variable selection, interaction detection can be similarly obtained by identifying those interactions which occur most frequently over the MCMC sequence of fitted sum-of-trees models $f_1^*, \ldots, f_K^*$. More precisely, for each simulated sum-of-trees model f_k^*, let z_{ijk} be the proportion of trees in f_k^* where both x_i and x_j are used for splitting rules. Then

$$v_{ij} \equiv \frac{1}{K} \sum_{k=1}^{K} z_{ijk} \tag{14}$$

is the average proportion of x_i,x_j interactions across $f_1^*, \ldots, f_K^*$. Those interactions with the largest average proportions are deemed most important. Note that this approach can be straightforwardly extended to three-way and higher order interaction detection.

In contrast, to our BART variable selection approach, it turns out to be less critical to use a small number of trees m for our interaction detection approach. Because the

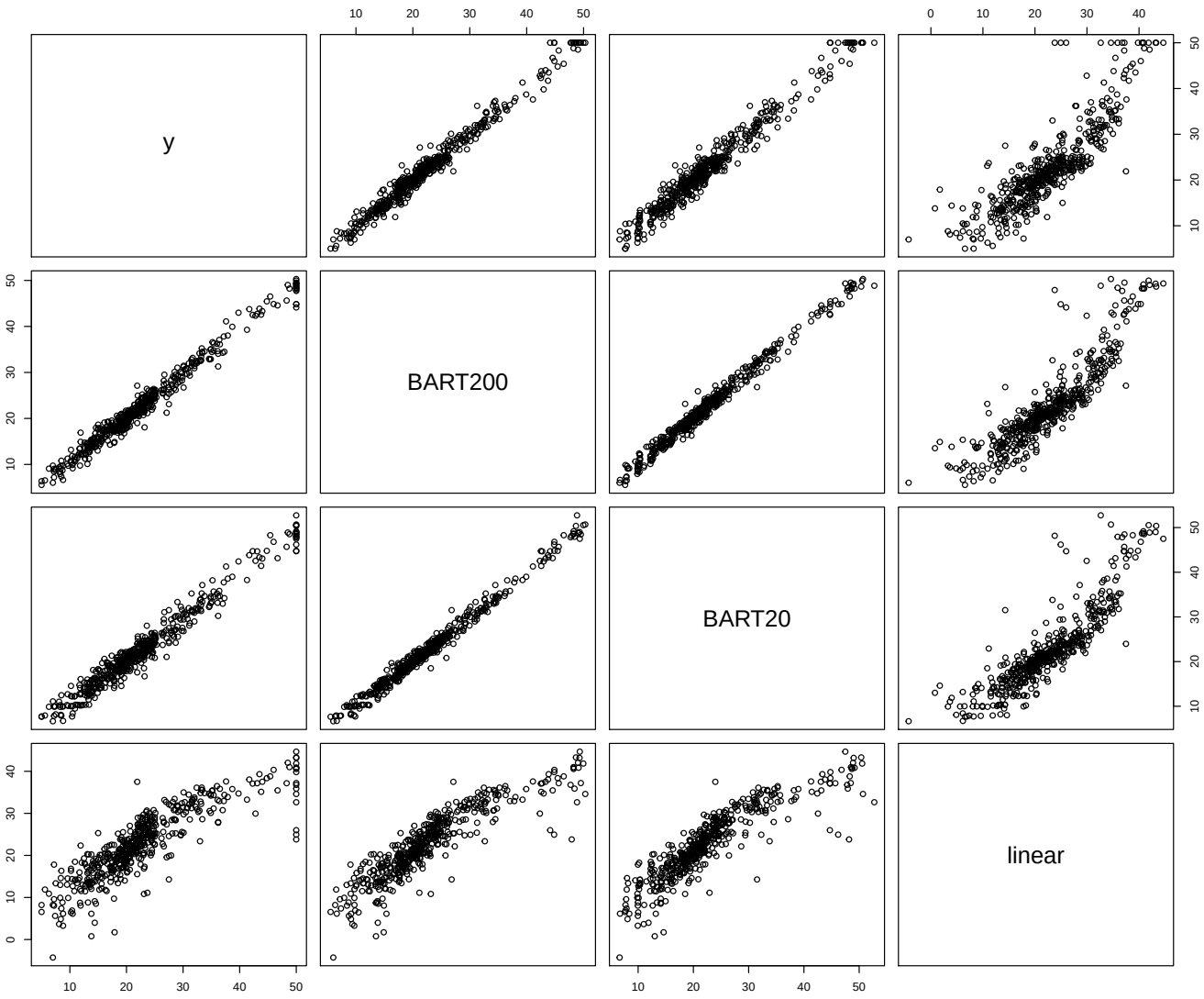

Figure 4: Pairs plot of y=median value, BART fits using 20 and 200 trees, and the fitted values from a linear model.

number of potential interactions grows rapidly with the number of variables, the chances of accumulation for irrelevant interaction becomes much smaller as well. Nevertheless, it is still illuminating to carry out the procedure with both large and small values of m.

4.1 Variable Selection and Interaction Detection with the Friedman Simulation Setup

Let us first illustrate our BART approaches to both variable selection and interaction detection with the well-known Friedman simulation setup. We here simulate $n = 500$ observations from the basic model

$$y = f(x) + \sigma Z, \quad Z \sim N(0, 1),$$

with x ten dimensional and

$$f(x_1, x_2, \ldots, x_{10}) = 10 \sin(\pi x_1 x_2) + 20 (x_3 - .5)^2 + x_4 + x_5.$$

The x_i are iid uniform on $(0, 1)$ and $\sigma = 1$.

This simulation setup was devised by [6] to study the efficacy of nonlinear regression techniques. However, the setup also turns out to be perfect for illustrating both variable selection and interaction discovery. Only the first five of the ten x components matter. With ten x's there are 45 possible interaction pairs, for the underlying model only one of these possibilities is present: only x_1 and x_2 interact. In a real application it would be of tremendous interest to know that only these two variables interact, even without having further knowledge of the functional form.

Results for one simulated data set are displayed in Figure (5). In panel (a) we have variable selection results. This panel corresponds closely to Figure (5) of [5]. For each variable, we plot the posterior mean of the percentage of rules (across all m tree) which use that variable. With $m = 20$, we very clearly identify the first five variables as being important.

Panel (b) gives the interaction detection results. With ten variables, there are 10*9/2=45 possible variable pairs. For each pair, we plot the posterior mean of the percent of trees (out of m) which use both of the variables in splitting rules. We normalize the $m = 20$ and $m = 200$ results by dividing by each set of 45 posterior means by the maximum. Thus, the largest value displayed in each case is one. With both $m = 20$ and $m = 200$ we clearly identify the first pair (x_1 and x_2) as being of interest. With two variables involved, a pair is less likely to come in inconsequentially, so that the identification of interesting pairs is less sensitive to the choice of m than in the case of variable selection.

4.2 Interaction Detection with the Boston Housing Data

For our second example, we return to the Boston housing data for which we illustrated variable selection in Section 3. Recall that each of the $n = 506$ observations corresponds to neighborhood. The response is the median house price in the neighborhood. There are 13

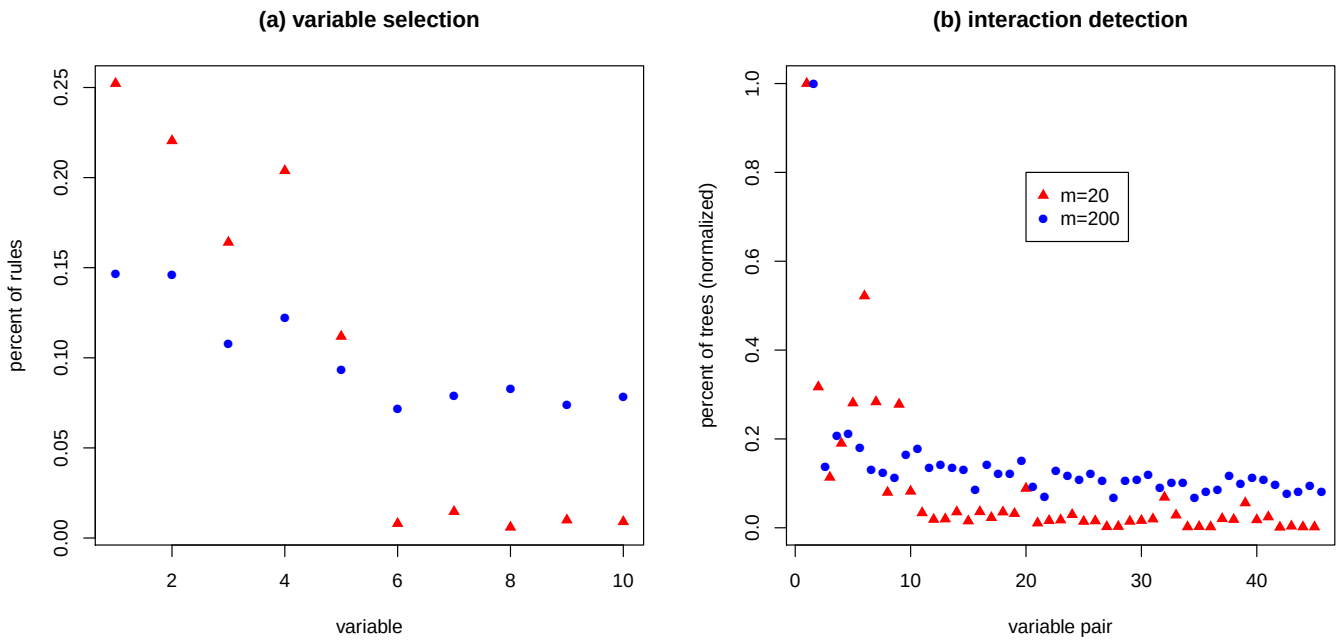

Figure 5: In panel (a) we correctly identify the first five variables as being important. In panel(b) we correctly identify the first interaction, which corresponds to variables x_1 and x_2.

explanatory variables measuring characteristics of the neighborhoods. We did a preliminary variable selection (using the approach illustrated in the previous section) and tossed out three of the x's. Fitted values (from BART) with and without the three x's are very similar.

Figure (6) displays the results of the interaction detection. The format is the same as in panel (b) of Figure (5). Several pairs of interest are identified. Our real data has more interesting structure than our simulated data! We will investigate the pair `dis` and `lstat` simply because these variables are more easily understood. `dis` is the "weighted distances to five Boston employment centres". `lstat` is the "percentage of lower status of the population".

In Figure (7) we attempt to graphically see the interaction between `dis` and `lstat` suggested by Figure (6). In panel (a) we plot `dis` vs. `lstat`. Four subsets of points are identified depending on whether `dis` and `lstat` are "low" or "high". In the (b) panel we plot the fitted values from the BART run with $m = 200$. Before fitting we subtracted off the average response so the vertical axis is actually the amount the median value for a neighborhood is above the average. The four boxplots correspond to the four data subsets indicated in panel (a).

So, for example, the first boxplot displays the fitted prices when both `ds` and `lstat` are low. The observations included here correspond to those highlighted in the bottom left corner of panel (a). The label "`dL_lL`" indicates that `ds` is Low and `lstat` is Low. Similarly, the third boxplot is labelled "`dH_lL`", indicating that `ds` is High and `lstat` is Low.

The first pair of boxplots indicate the effect of increasing `lstat` when `ds` is low. The second pair of boxplots indicate the effect of increasing `lstat` when `ds` is high. Clearly, the boxplots indicate a strong interaction. For low `dl`, the effect of a the change in `lstat` is much more pronounced. A nice neighborhood close to the city center is highly desirable whereas a bad neighborhood close to the city center may be very bad.

5 A Utility Based Approach to Variable Selection using BART Inference

In this section we discuss an approach to variable selection developed by Carvalho, Hahn, and McCulloch (CHM henceforth, [2]). While the CHM approach may be used with any model, CHM emphasize the use of BART to do the nonlinear modeling. BART is appealing for the CHM methodology because of it's ability to get reasonable results from default settings and assessment of uncertainty.

CHM cast variable selection as a *decision* to use a subset of variables rather than a search for the "true" subset. Generally we can think of a Bayesian decision as having three components, the prior, likelihood, and the utility. CHM emphasize the utility component rather than the prior component. Rather than developing a prior specification that believes that there is some small subset of active variables (see Linero, [8]) CHM run BART at its default setting (as discussed in Section 2) and then look for variable subsets which can approximate the simple BART inference *as a practical matter*.

Infamously, the ubiquitous p-value fails to distinguish between "practical significance"

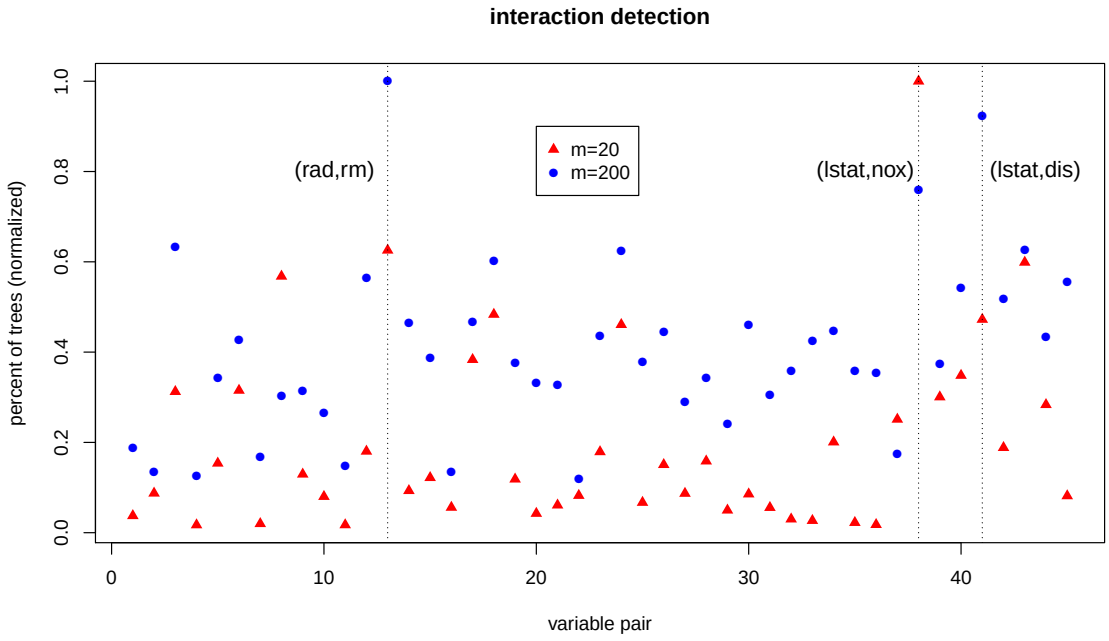

Figure 6: Interaction detection for the Boston housing data with ten explanatory variables.

and "statistical significance". Recent appreciation of this severe deficiency has lead to many futile attempts to fix a fundamentally flawed approach. The CHM approach first looks for subsets of variables that can approximate the unrestricted inference and then assesses uncertainty using the posterior distribution of the approximation error. While not strictly following the correct Bayesian decision theoretic approach, by remaining true to its spirit, CHM develop a simple approach which captures the elements of the problem practitioners really care about. This approach was first developed for the linear model case in [7] and is extended to the nonlinear case using BART in CHM. See also [11] for a general discussion of the "projection strategy" in which look for suggested simplifying structure suggested by unrestricted inferences.

Let S denote a subset of the variables. Let $\hat{f}(x)$ be the the posterior mean of $f(x)$ obtained from BART. Let X_P denote as set of x vectors at which we want to learn $E(Y \mid x) = f(x)$. The CHM approach has three basic steps:

1. Run a simple BART inference,

2. Find a subset S of the variables such there is a nonlinear function $\gamma_S(x)$ such that γ_S only depends on the subset of variables indicated by S and $\gamma_S(x) \approx \hat{f}(x)$, for $x \in X_P$.

3. Use the BART draws to assess our uncertainty about the approximation error.

Note the choice of X_P above. To select variables, we need to specify how we going to use the model in practice! Often a simple default choice is to let X_P be the values observed in

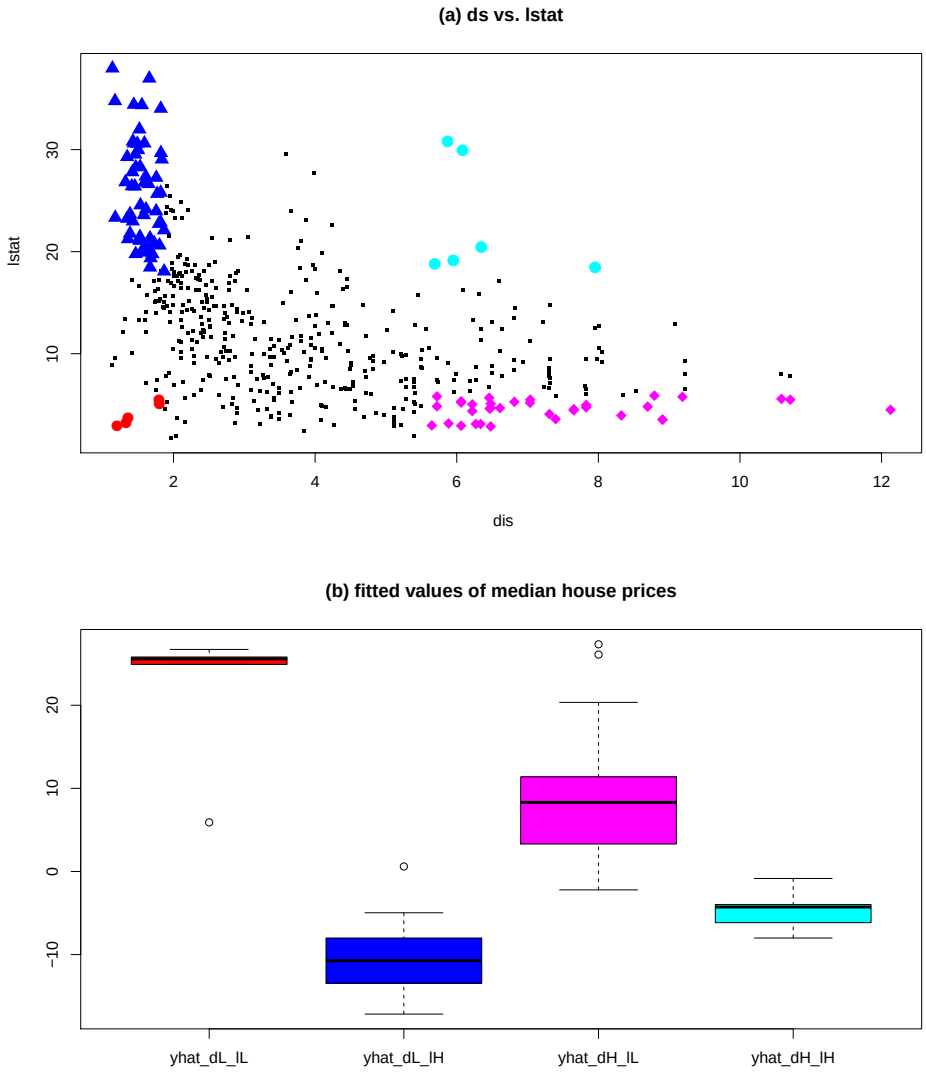

Figure 7: In panel (a) we identify four subsets of our data by whether each of `ds` and `lstat` are low or high. In panel (b) the boxplots display the fitted values (median house values) for the observations in the four subsets. The average of the dependent variable was subtracted off so that the vertical axis is the amount the median value of a neighborhood is above average. The first pair of boxplots both have low values of `dis`. The first box has low values of `lstat` and the second box has high values of `lstat`. The second pair of boxplots again compare low and high `lstat` but now `ds` is high.

the training data. It is however, not uncommon for an application to consider a much richer set of x. This consideration, which seems fundamental, is clearly missing from most variable selection approaches.

5.1 Step 1: BART Inference

Figure 8 displays results from step 1 for the Boston housing data. BART was run with default settings and 10,000 burn and kept MCMC iterations. The full data set of $n = 506$ observations were used for training Figure 8 reproduces some of the information in Figure 4 in Section 3, but here the intervals for $f(x)$ are included. The crucial point here is that we are just running standard BART.

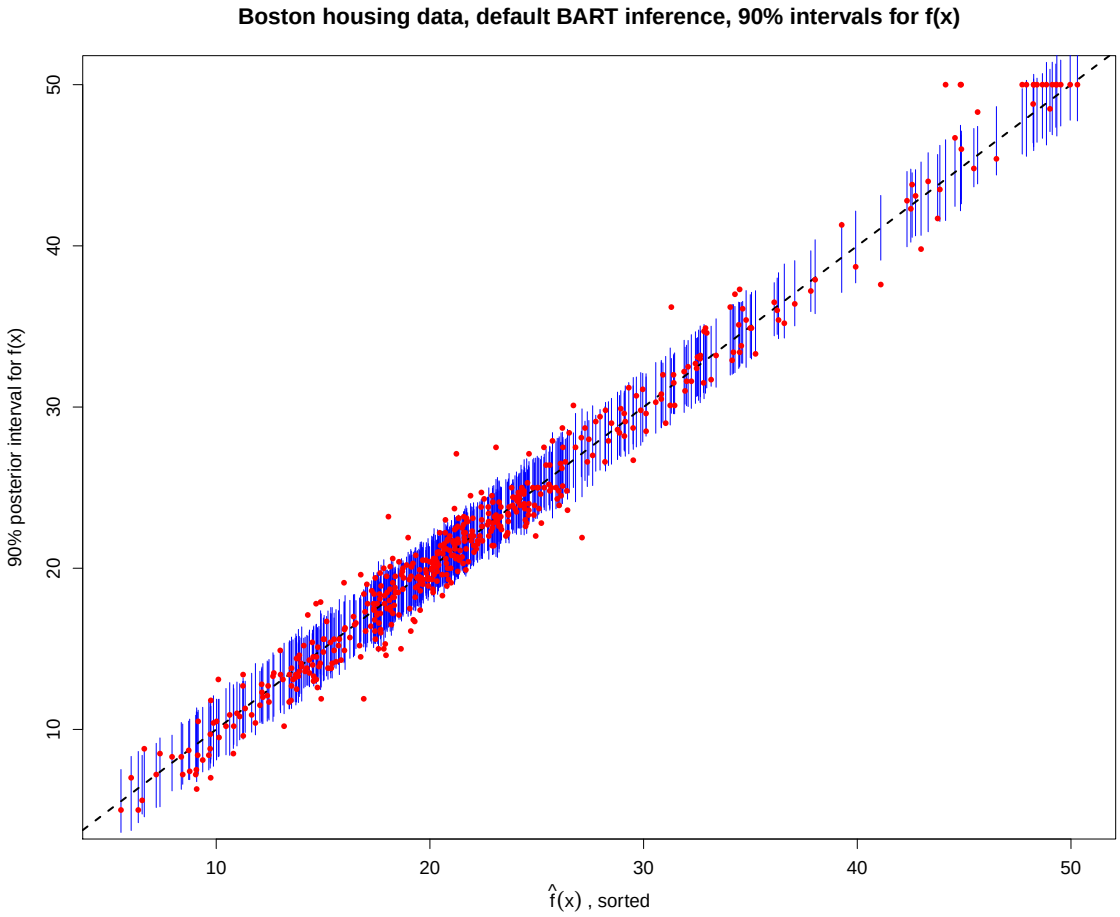

Figure 8: The short vertical lines are 90% posterior intervals for $f(x)$, for x in the training data. Sorted by the values of $\hat{f}(x)$, the posterior mean. Points plotted at $(\hat{f}(x), y)$, for (x, y) in the training data. The dashed line as intercept 0 and slope 1.

5.2 Step 2: Subset Search

We now tackle the problem of finding subsets S and nonlinear functions $\gamma_S(x)$ which depend on x only through the subset of variables indicated by S, such that γ_S approximates $\hat{f}$ for $x \in X_P$. Again, $\hat{f}$ is obtained from step 1.

To search for subsets, we use the time honored forward selection and backwards selection approaches. When the number of variables is small (as in our Boston example) we can simply try all possible subsets. The tricky part is finding γ_S given S. We do not want to rerun BART at each candidate subset since this would be too time consuming. In addition, we are seeking an arbitrarily good approximation and BART is engineered to deal with the noisy y and this problem has different characteristics so that simple default BART may not be effective.

The key idea in CHM is to use a large decision tree for γ_S. Given S, we fit a large tree to the data $(x_S, \hat{f}(x))$, where x_S is the subset of variables indicated by S. When fitting a decision tree to data, we have to worry about the bias variance trade-off when choosing the tree size. Here, since we are simply seeking an approximation to already smoothed data, we find that a simple default large tree size works very well. When we have sufficient information in x_S we cannot overfit. If x_S does not capture the relevant information in x, our big tree could, in principle overfit, but we have not found this to be a problem (see Figure 10). Informally, CHM refer to this approach as "fit-the-fit" as we use the big tree to explain the fit $\hat{f}(x)$ rather than y.

Figure 9 shows the results from forward search (top panel), backward search (middle panel) and all subsets search (bottom panel). So, for example, if we try to fit-the-fit with just one of the variables x_j, $j = 1, 2, \ldots, p$, by fitting a big tree to the data $(x_j, \hat{f}(x))$ we find that `lstat` gives us the best fit. In Figure 9, the square of the correlation between $\gamma_S(x)$ and $\hat{f}(x)$ for $x \in X_P$ is reported on the vertical axis (labeled R-squared). In the top panel we can see how variables come into our search one by one, and in the middle panel we can see how they go out one by one. From the bottom panel we see that is this example, all three search methods give very similar results for the fit. With as few as three variables, (lstat, rm, nox) we can approximated the fit obtaining all the information in x very well.

Interestingly, forward and backwards to not give exactly the same variable subsets. In the top panel, the first three coming in are lstat, rm, and nox. These three variables are the three left in the middle panel. But, the fourth variable is rad in the forward search and dis in the backward search. Indeed, rad is the first variable out in the backward search. The variable subsets of sizes 1-5 found by the exhaustive search are:

```
1: "lstat"
2: "rm"    "lstat"
3: "nox"   "rm"    "lstat"
4: "nox"   "rm"    "rad"   "lstat"
5  "nox"     "rm"      "dis"     "ptratio" "lstat"
```

Note that while both the forward and backward searches have the property that a smaller set of variables is always a subset of a larger set, this need not be the case for the all subsets

search. We see that, for all subsets search, the best subset of size four has rad and not dis, while the best subset of size 5 has dis and not rad. These two variables, rad and dis, have a strong nonlinear relationship which explains the different results from the different searches. CHM are careful not to claim they know the "truth" about rad and dis. They only know the certain subsets approximate well.

Given found subsets S_j of size $j = 1, 2, \ldots, p$, CHM sometimes find that we can improve the approximation by rerunning BART using the training data and the indicated subsets. Figure 10 plots $\hat{f}(x)$ vs the approximating function found by the big tree and re-running BART for $j = 1, 2, 3, 4$. BART fits are in the left column and big tree fits are in the right column. We see that after three or four variables we have an excellent fit and the results from refitting BART and the big tree and sufficiently comparable to give us faith in our search mechanism. It does not seem that the big tree is overfitting for small j.

5.3 Step 3: Uncertainty Assessment

In step 2, we only used the posterior means from the BART inference to provide $\hat{f}$. In this section we use the BART MCMC draws to assess our uncertainty about the approximation error.

Crucially, at this stage we need to choose a metric to quantify the practical consequences of a certain level of approximation error. We must choose a distance

$$D(f, g) \equiv D((f(x_1), f(x_2), \ldots, f(x_p)), (g(x_1), g(x_2), \ldots, g(x_p)), x_i \in X_P)$$

to capture how different f is from g for $x \in X_P$. We then report the posterior distribution of

$$D(f, \gamma_S)$$

Where f is the random variable representing our posterior uncertainty about $f(x) = E(Y \mid x)$ and γ_S is a candidate function using the subset S. The posterior distribution is estimated by the set of values $D(f_d, \gamma)$ where f_d are BART MCMC draws of f, $d = 1, 2, \ldots, N$. In our Step 1, N was 10,000. The idea is that each f_d is a plausible candidate for a good function and we want to see how small the approximation error tends to be. We are reporting the posterior distribution of the approximation error for various subsets of the variables in x.

The top panel of Figure 11 reports the posterior for the subsets found using all possible subsets and γ_S found using the BART refit. The distance D is the root mean squared error. To use Figure 11, we actually have to understand something about y. Here is the usual R summary of the y values (median house prices) in the training data.

```
Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
5.00   17.02   21.20   22.53   25.00   50.00
```

Clearly, with only the three variables (lstat,nox,rm) the approximation error is highly likely to be close to 2. This may be deemed sufficiently accurate as a practical matter given the range of y. With seven variables, the error is highly likely to be comparable to that with all the variables.

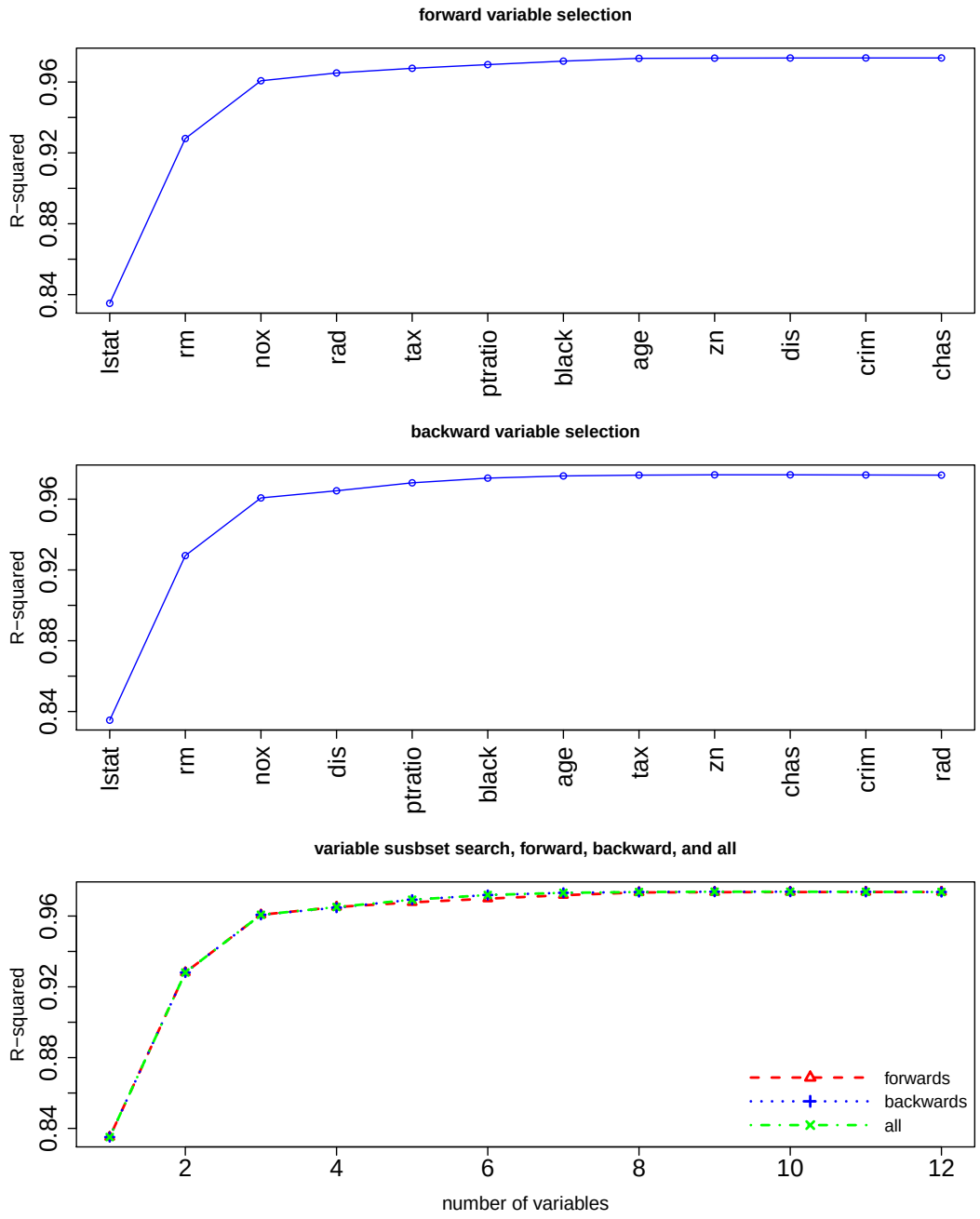

Figure 9: Search for functions which approximate $\hat{f}(x) \approx E(Y|x)$ obtained from initial BART run using all the variables. Approximating functions use subsets of the variables. In each panel the vertical axis is the square of the correlation between $\hat{f}(x)$ and the approximation. The top panel shows results from the forward search in which variables enter one at a time. The middle panel shows results from the backward search in which variables exit one at a time. In both the top and middle panel the horizontal axis is labeled with the names of the variables as they go in or out. The plot on the bottom panel plots the fit from forwards, backwards, and all subsets search. In the bottom panel, the number of variables used is on the horizontal axis.

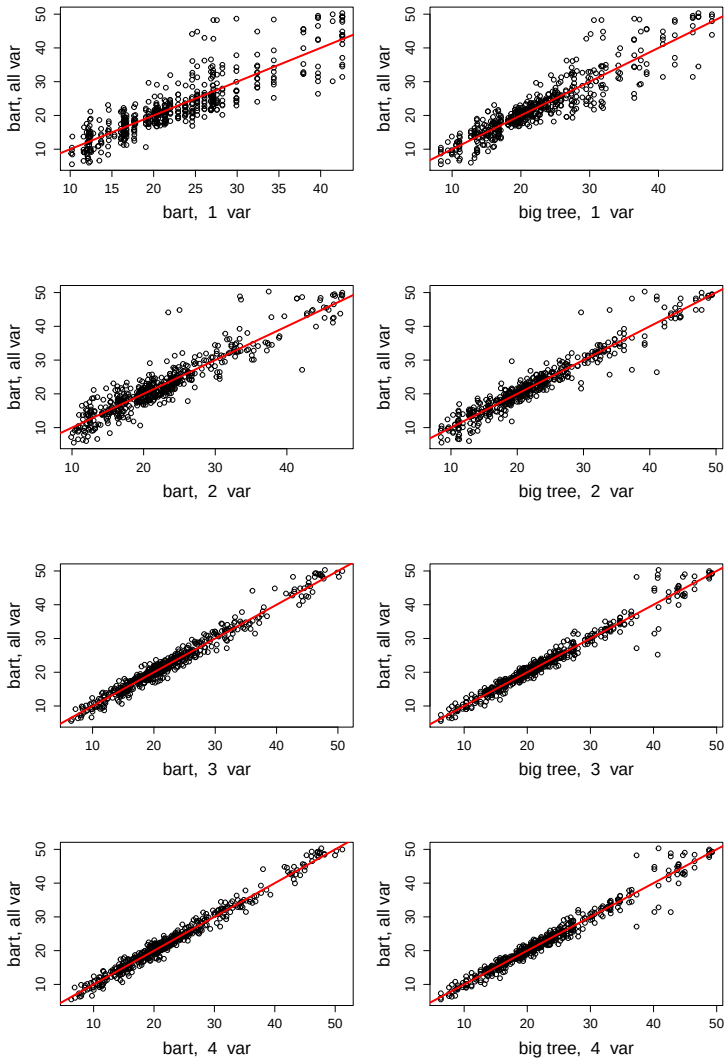

Figure 10: Comparing approximations to $\hat{f}(x)$ using big trees are running BART on found subsets. In each graph the posterior mean estimates $\hat{f}(x)$ is on the vertical axis. The four rows correspond to using the first 1,2,3 and 4 variables as found by the all subsets search. In both the all subset and forward search the first four variables variables enter in the order lstat, rm, nox, and rad. In the left column of plots, the horizontal axis is the posterior mean estimate of $f(x)$ obtained by running BART using only the subset of variables. In the right column is the big-tree fit to $\hat{f}(x)$.

The bottom panel of Figure 11 reports the posterior distribution of

$$D(f, \gamma_S) - D(f, \hat{f}).$$

How much bigger is the approximation error when using γ_S than when using $\hat{f}$? With our top three variables, this difference is highly likely to be less than one, and with seven variables it is known to be negligible as a practical matter!

In Figure 11 we have use root mean squared error. Of course, in any application it would be straightforward to replace this with a more meaningful metric motivated the the problem at hand.

While steps 2 and 3 are motivated by the Bayesian decision theory framework, they do not stick to it fully. In particular, in step 2, $\hat{f}$, the posterior mean (or median) is the holy grail. In step 3 our attitude is that any draw f_d from the MCMC *could* be the function we want to approximate. These compromises are made to build a simple approach that will give meaningful answers to applied investigators.

6 Conclusion

Modern statistical methodology provides us with remarkable effective tools for uncovering potentially nonlinear relationships between an outcome y and a complex, high dimensional predictor x. Understanding the nature of an uncovered relationship is inherently difficult given the necessarily complex mathematical representation of the relationship.

A fundamental aspect of a relationship which investigators quite commonly wish to learn about is variable selection. Out of all the variables in x, which ones are *the important ones*? The development of tools for variable selection in the context of a linear model has been an important area of research for many years. The development of methods for variable selection for modern nonlinear models is a crucial for applied use. Remarkably, key `R` packages such as `rpart` (for classification and regression trees) and `randomForest` include "variable importance measures".

Any approach to variable selection involves some difficult choices given the complexity of the problem. In this paper we describe some relative simple, but effective, approaches using Bayesian Additive Regression trees (BART). BART has several advantages. Some key aspects of BART are the relatively simple default choices for the prior and the Markov Chain Monte Carlo (MCMC) stochastic search and uncertainty representation. All of the methods described in this paper allow for use of the MCMC draws to develop a representation of the uncertainty (see Figures 2 and 11).

Section 3 describes an extremely simple yet effective approach to variable selection using BART inference. We simply count the number of times a particular component of x is used in a decision rule in a tree in the BART ensemble of trees. This may be done for each MCMC draw. A drawback is that it may be necessary to use fewer trees in the BART ensemble that might be optimal for predictive inference. However, it is a simple matter find a number of trees that identities important variables without drastically compromising the fit (Figure 4).

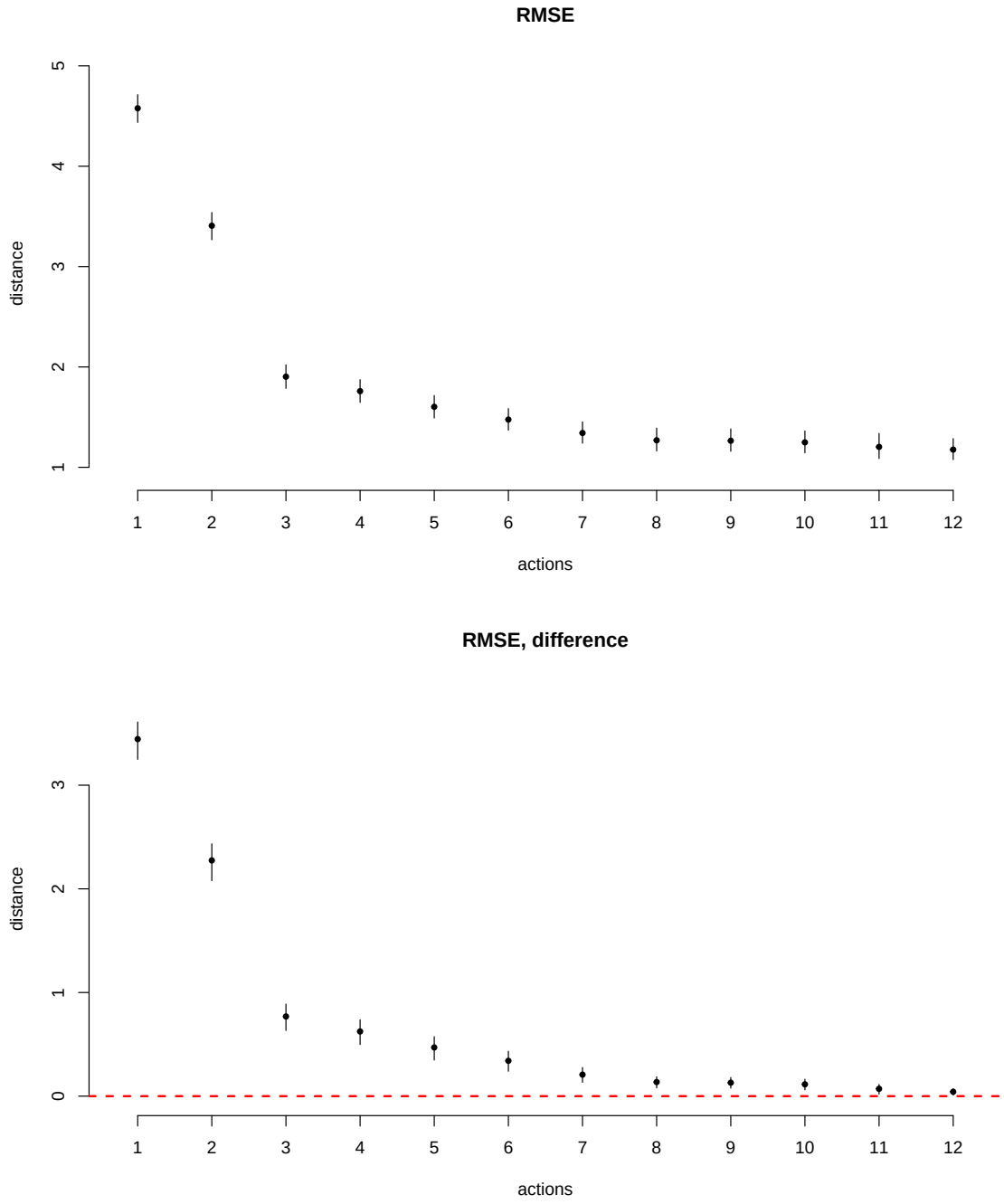

Figure 11: Posterior distribution of the approximation error. Approximations are found by running BART with the subsets found by the all subsets search. Top panel shows the posterior distribution of the approximation error. Bottom panel show the posterior distribution of the between the error using the subset and the error using all of the variables.

In practice this may be much easier than altering the basic model to incorporate prior choices about variable sparsity.

Rather than finding important variables, Section 4 identifies pairs of variables that interaction. The key simple ideas is that a single tree can capture interactions be including different variables in the tree. Rather than count how often a variable is used in a tree decision rule, we count how often a pair of variables occur in the same tree.

Section 5 presents a completely different approach. Given the fit from a basic BART inference, we search for an approximation to the fit that uses only a subset of the variables. A simple yet effective search algorithm is presented. Given a potentially useful subset, we assess the subset and our uncertainty through the posterior distribution of the approximation error. By focusing on the approximation error, we focus on the practical importance of variable subsets.

The unifying principle underlying all the approaches in this paper is that we rely and simple versions of BART inference. Effective post processing of the BART MCMC draws allows us to uncover variables of interest.

References

[1] Justin Bleich, Adam Kapelner, Edward I. George, and Shane T. Jensen. Variable selection for bart: An application to gene regulation. *Ann. Appl. Stat.*, 8(3):1750–1781, 09 2014.

[2] Carlos Carvalho, Richard Hahn, and Robert McCulloch. Fitting the fit, variable selection using surrogate models and decision analysis, a brief introduction and tutorial. *arxiv*, 2020.

[3] H.A. Chipman, E.I. George, and R.E. McCulloch. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, 2010.

[4] Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. Bayesian cart model search. *Journal of the American Statistical Association*, 93(443):935–948, 1998.

[5] Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. BART: Bayesian additive regression trees. *Annals of Applied Statistics*, 4(1):266–298, 2010.

[6] Jerome H. Friedman. Multivariate adaptive regression splines (Disc: P67-141). *The Annals of Statistics*, 19:1–67, 1991.

[7] P Richard Hahn and Carlos M Carvalho. Decoupling shrinkage and selection in bayesian linear models: a posterior summary perspective. *Journal of the American Statistical Association*, 110(509):435–448, 2015.

[8] A. Linero. Bayesian regression trees for high dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113(522):626–36, 2018.

[9] Veronika Ročková and Enakshi Saha. On theory for bart. volume 89 of *Proceedings of Machine Learning Research*, pages 2839–2848. PMLR, 16–18 Apr 2019.

[10] Veronika Ročková and Stephanie van der Pas. Posterior concentration for bayesian regression trees and forests. *Ann. Statist.*, 48(4):2108–2131, 08 2020.

[11] Spencer Woody, Carlos M. Carvalho, and Jared S. Murray. Model interpretation through lower-dimensional posterior summarization. *Journal of Computational and Graphical Statistics*, 0(0):1–9, 2020.