Statistical Inference on Random Forests

A Thesis
Presented to
The Division of Mathematics and Natural Sciences
Reed College

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Arts

Aurora Owens

May 2017

Approved for the Division
(Mathematics)

Andrew Bray

Acknowledgements

I want to thank a few people.

Preface

This is an example of a thesis setup to use the reed thesis document class.

Table of Contents

List of Tables

List of Figures

Abstract

The preface pretty much says it all.

Second paragraph of abstract starts here.

Dedication

You can have a dedication here if you wish.

Chapter 1

Introduction

1.1 Trees and Random Forests

1.1.1 Trees

Decision trees may be familiar to many with a background in the social or medical sciences as convenient ways to represent data and can assist in decision making. Morgan and Sonquist (1963) derived a way for constructing trees motivated by the specific feature space of data collected from interviews and surveys. Unlike, say agricultural data which involves mostly numerical variables like rainfall, the data collected from interviews is mostly categorical. On top of this issue, the datasets Morgan and Sonquist dealt with had few participants, n, and much data collected on them, p. To continue with their list of difficulties, there was reason to believe that there were lurking errors in the variables that would be hard identify and quantify. Lastly, many of the predictors were correlated and Morgan and Sonquist doubted that the additive assumptions of many models would be appropriate for this data. Morgan and Sonquist noted that while many statistical methods would have difficulty accurately parsing this data, a clever researcher with quite a lot of time could create a suitable model simply by grouping values in the feature space and predicting that the response corresponding to these values would be the mean of the observed responses given the grouped conditions. Their formalization of this procedure in terms of “decision rules” laid the ground work for future research on decision trees.

Later researchers proposed new methods for creating trees that improved upon the Morgan and Sonquist model. Leo Breiman et al 1984 proposed an algorithm called CART, *classification and regression trees*, that would allow trees to be fit on various types of data. An alternative to this method is conditional inference trees. Torsten Hothorn, Kurt Hornik, Achim Zeileis argue in their 2006 paper *Unbiased Recursive Partitioning: A Conditional Inference Framework*, CART has a selection bias toward variables with either missing values or a great number of possible splits. This bias can effect the interpretability of all tree models fit using this method. As an alternative to CART and other algorithms, Hothorn et al propose a new method, conditional inference trees.

There is a limit to the predictive capabilities of a single tree as they suffer from high variance. To alleviate this, random forests are often used instead. They function by enlisting the help of many trees, and then by aggregating the responses over all of them but with a subtle trick that ensures the trees will be independent of each other. At each split only m variables are considered as possible candidates. Random forests and their algorithms will be discussed at length in Chapter 2.

1.2 What We Mean When We Talk About Inference

1.2.1 Inferential vs Descriptive Statistics

A note should be made of the difference between inferential and descriptive statistics. This paper's aim is to describe a process of making inferential claims using random forests, not descriptive ones. Descriptive statistics describe the data at hand without making any reference to a larger data generating system that they come from. It follows that inferential statistics then make claims about the data generating system given the data at hand.

—Frequentist vs Bayesian—

—There is some debate about interpreting inferential statistics. On one hand, we have the Bayesian model—

Need a better way to discuss inference than Bayes/frequentist

1.3 Permutations and Populations

As stated in the introduction of the *Chronical of Permutations Statistical Methods* by KJ Berry et al, 2014, there are two models of statistical inference. One is the population model, where we assume that the data was randomly sampled from one (or more) populations. Under this model, we assume that the data generated follows some known distribution. "Under the population model, the level of statistical significance that results from applying a statistical test to the results of an experiment or a survey corresponds to the frequency with which the null hypothesis would be rejected in repeated random samplings from the same specified population(s)", (Berry et al, 2014).

The permutation family of methods, on the other hand, only assumes that the observed result was caused by experimental variability. The test statistics is first calculated for the observed data, then the data is permuted a number of times. The statistic is calculated after each permutation to dervive a distribution of possible values. Then the original test statistic is tested against this distribution. If it is exceptionally rare, then there is evidence that our observation was not simply experimental variability.

1.4 Inference on Random Forests

1.4.1 The Problem

Random forests create models with great predictive-, but poor inferential capabilities. After Morgan and Sonquist initial development of decision trees, they quickly moved to the domain of machine learning and away from statistics, thus, researchers focused on bettering predictions and improving run times and less on the statistics behind them. Inferential statistics with random forests is usually treated as a variable selection problem, and generally falls behind the predictions in importance. This has limited the applications of random forests in certain fields, as to many the question of "why" the data is the way it is, is just, if not more, important as the predictions. There are several means of performing descriptive statistics with random forests that could be interpreted incorrectly as attempting to answer this, namely base variable importance, but without a statistically backed method for performing variable importance, the use of random forest is limited to prediction-only settings.

1.4.2 Proposed solutions to this problem

Statisticians Breiman and Cutler proposed a method of permuted variable importance to answer this problem. Their method compares the variable importance for each variable in a tree-wise manner. For each tree, the permuted variable importance of the variable X_j is:

$$PV^t(x_j) = \frac{\sum_{i\in|B|} y - \hat{y}^t}{|B|} - \frac{\sum_{i\in|*B|} y - \hat{*y}^t}{|*B|}$$

Where B is the matrix representing the feature space, $|B|$ is the number of observations, $*B$ is the matrix of predictors but with X_j permuted, $\hat{y}$ is the predicted outcome, and $\hat{*y}^t$ is the predicted outcomes after variable X_j has been permuted. This value is averaged over all the trees. It's important to note that if the variable X_j is not split on in the tree t, the tree-wise variable importance will be 0.

Creating a permutation-based method is certainly an attractive solution to our problem. One, it allows us to estimate the distribution of variable importance and generate a Z score under the null hypothesis that $PV = 0$.

$$PV(x_j) = \frac{\sum_1^n treePV^t(x_j)}{\frac{\hat{\sigma}}{\sqrt{ntree}}}$$

Strobl et al from the University of Munich criticize this method in their 2008 technical report, **Danger: High Power! – Exploring the Statistical Properties of a Test for Random Forest Variable Importance**. One, this method has the downside of increasing power with increasing numbers of trees in the forest. This is a more or less arbitrary parameter which we would hope would not affect our importance estimates. Secondly, the null hypothesis under Breiman and Cutler's strategy is that the variable importance V for any variable X_j is not equal to zero given Y, the response. Because random forests are most often used in situations with multicolinearity that would make other methods like the linear model difficult, Strobl argues that any variable importance measure worth its salt should not be mislead by correlation within the predictors.

The researchers at the University of Munich published a fully fleshed response to the Breiman and Cutler method in 2008, titled *Conditional Variable Importance for Random Forests* that address these issues. Strobl et al propose restructuring the Breiman and Cutler algorithm to account for conditional dependence among the predictors. Their algorithm looks like this:

Algorithm 1 Conditional Variable Importance for Random Forests

```
Fit a random forest to the model, $R_0$, and calculate base variable importance for each variable $V$
for every predictor $X_j \in X_1, ..., X_n$ do
    Conditionally permute $X_j$ given the splits found in $R_0$
    Fit a new random forest $R_j$ with the permuted data
    Calculate a new variable importance $\hat{V}_j$
end for
For every variable $X_1, ..., X_n$,
    $$CV(X_j) = \hat{V}_j - V_j$$
```

The null hypothesis is that $CV(X_j) = 0$ given the predictor Y *and all other predictors* $X_1, ..X_n$.This accounts for interactions between X_j and the other predictors. Using the simulated data from the previous example, here's an implementation of the algorithm outlined here as it is in the `party` package.

This paper aims to provide a response to this method. One the conditional permutation algorithm is notoriously slow with any dataset of a size that is appropriate for a random forest. Two, the partitions are made from the random forest corresponding to the formula of $Y\ X_1, ..., X_n$ instead of a model of $X_j\ X_1, ..., X_n$.

T as a piece-wise constant function from the predictors to Y

Chapter 2

Simulations and Comparisons

2.1 Simulated Data

Tree-based methods shine in situations with correlated predictors, although these situations can pose problems for inference. In a situation with correlated predictors X_1 and X_2, and the model we are considering is $Y \sim X_1 + X_2$, it is difficult to say how much of the modeled effect on Y is due to X_1 or X_2. To illustrate this idea, compare a few existing methods, and explore methods of inference on tree based models three datasets will be simulated with different correlation structures. We will be focused more on the correlation structure between the predictors than on their relationships with the response and this will be reflected in the simulations.

To aid in comparisons between the methods, one of the simulated datasets considered in this paper will be generated from the same method as used in (Strobl et al, 2008???). Under this method, the 13 x 1000 data set, D_1, has 12 predictors, $V_1, .., V_{12}$, where $V_j \sim N(0,1)$. The first four are, however, block correlated to each other with $\rho = .9$. They are related to Y by the linear equation:

$$Y = 5 \cdot V_1 + 5 \cdot V_2 + 2 \cdot V_3 + 0 \cdot V_4 + -5 \cdot V_5 + -5 \cdot V_6 + 0 \cdot V_7 + 0 \cdot ..... + E, E \sim N(0, \frac{1}{2})$$

Note that the coefficients for $V_7, ..., V_{12}$ are all zero.

Table 1: Correlation of $V_1, ..., V_7$ and Y

	V1	V2	V3	V4	V5	V6	V7	y	beta
V1	1.000	0.915	0.908	0.907	-0.034	0.006	0.012	0.829	5
V2	0.915	1.000	0.914	0.914	-0.020	-0.001	-0.001	0.830	5
V3	0.908	0.914	1.000	0.903	-0.017	-0.007	0.007	0.808	2
V4	0.907	0.914	0.903	1.000	-0.002	-0.015	0.023	0.789	0
V5	-0.034	-0.020	-0.017	-0.002	1.000	0.044	0.005	-0.388	-5
V6	0.006	-0.001	-0.007	-0.015	0.044	1.000	-0.005	-0.364	-5
V7	0.012	-0.001	0.007	0.023	0.005	-0.005	1.000	-0.141	-2

As can be seen from the last column in the table, “beta”, although $V4$ was not included in the model $Y \sim V1, ..V_{12}$, its’ strong correlation with more influential predictors $V_1, ..., V_3$ insures that it still shows a strong linear correlation with Y. A linear model would likely *overstate* the effect of V_4 on Y.[1][2]

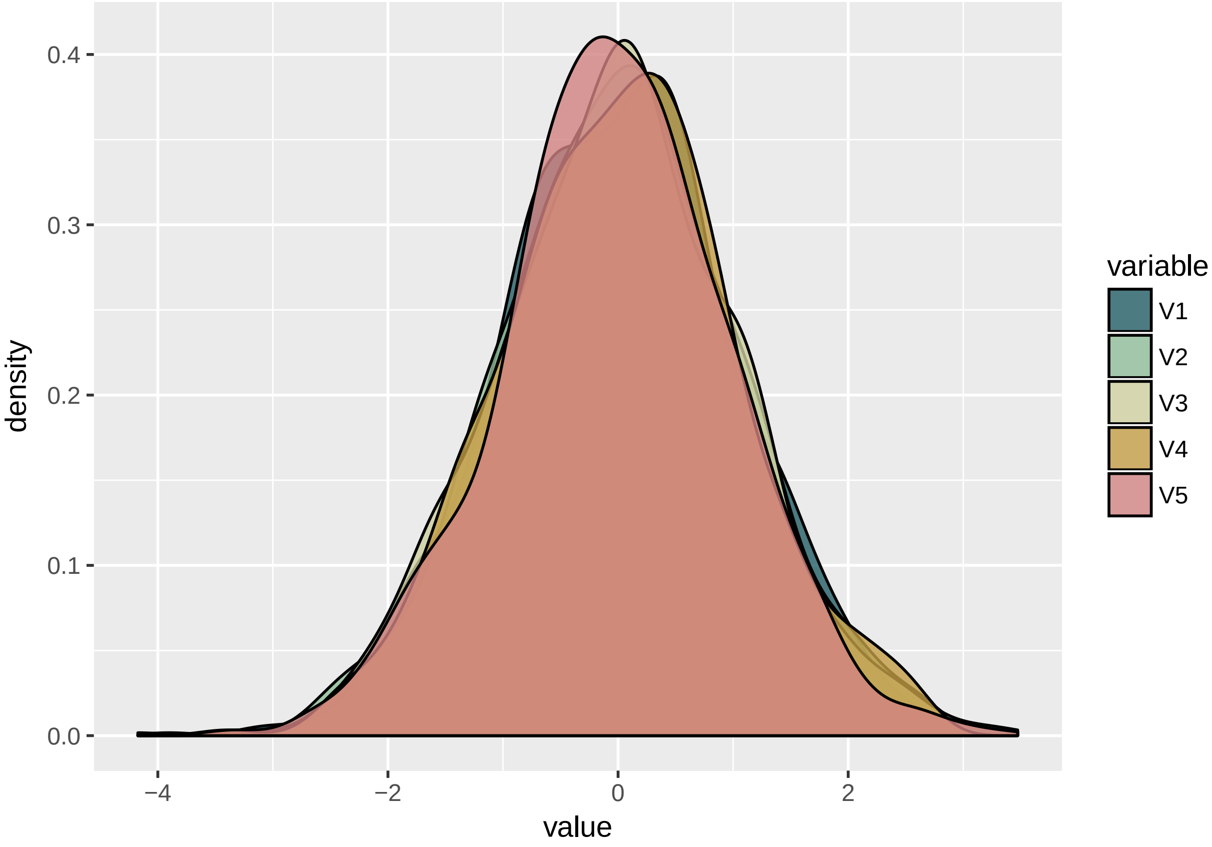


Figure 2.1: Density Graphs for V1 through V5

As can be seen above in Figure 1 the densities of $V_1, ..., V_5$ are all very similar due to the way they were generated.

Figure 2 is an illustration of the relationship between $Y \sim V_4$ with linear correlation of .789.

While D_1 represents a situation with linear correlation between the predictors, D_2 does not. Here, the model is the same, $Y\ X_1, ..., X_12$ where Y is generated according to the equation:

$$Y = 5 \cdot X_1 + 5 \cdot X_2 + 2 \cdot X_3 + 0 \cdot X_4 + -5 \cdot X_5 + -5 \cdot X_6 + 0 \cdot X_7 + 0 \cdot ..... + E, E \sim N(0, \frac{1}{2})$$

[1] A great deal of effort was undertaken by the author to find the defenative, authentic CART algorithm. This implementation follows the rough strokes set out in the 1984 text *Classification and Regression Trees* to the best of the author's ability and may not be exactly the algorithm found in R packages like 'tree()'

[2] Another point that could be said is that, no V_4 is not unimportant, V_1, V_2, and V_3 are just stand ins for the real star, V_4, as they are nearly the same ($\rho \sim 1$). Then the real relationship represented here is $Y \sim (5 + 5 + 2) \cdot V_4 + -5 \cdot V_5 + -5 \cdot V_6 + -2 \cdot V_7$. This model is not unsuccessful in capturing the structure of the data, and this is typically the practice used to model data with highly correlated predictors. If this seems philosophically satisfying to you, the rest of this thesis may seem a bit inconsequential. I apologize.

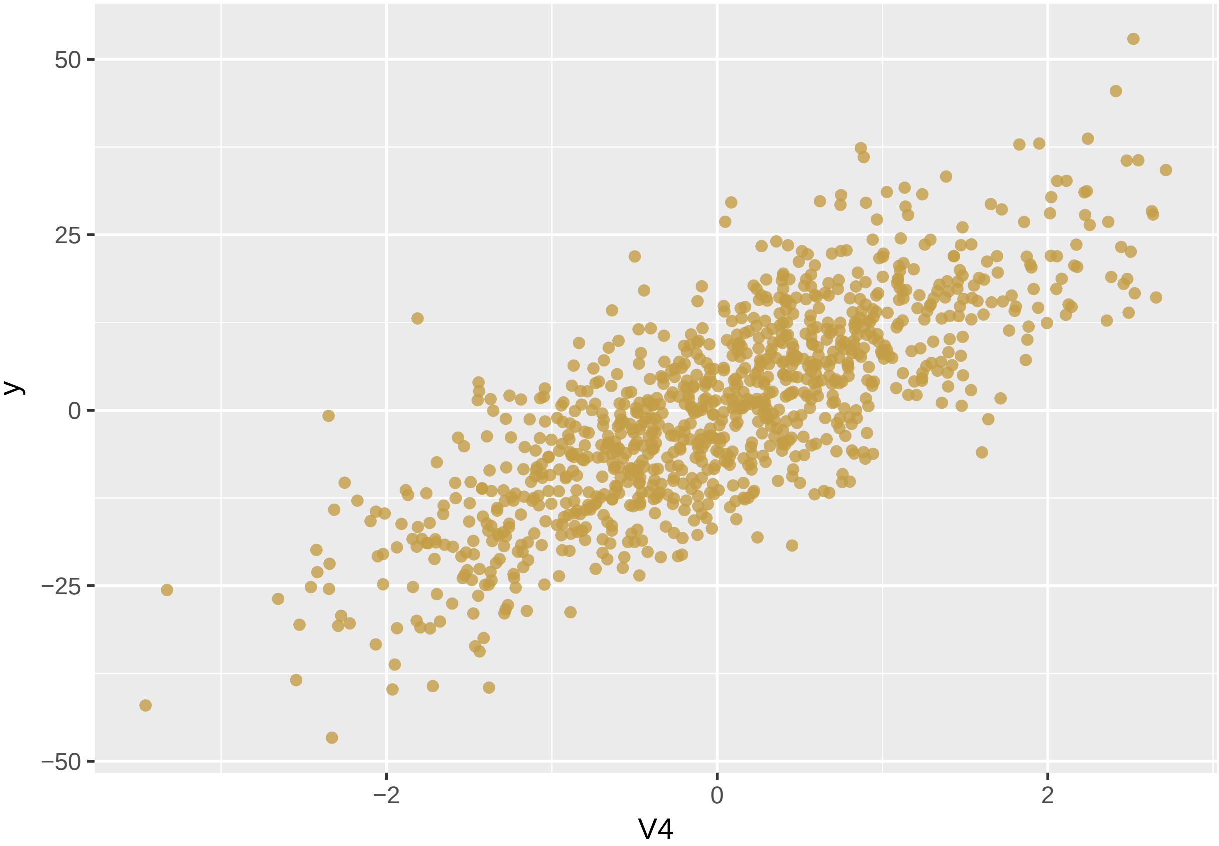


Figure 2.2: Plot of Y ~ V4, Correlation = .789

However, instead of block correlation with $\rho = .9$, four variables are related to each other by the equations below. Note that $X_1, X_5, ..., X_{12}\ N(0, 1)$

$$X_2 = X_1 + E, E \sim Exponential(1)$$

$$X_3 = X_2 + E, E \sim Exponential(1)$$

$$X_4 = X_3 + E, E \sim Exponential(1)$$

Table 2: Correlation of $X_1, ..., X_7$ and Y

	X1	X2	X3	X4	X5	X6	X7	y	beta
X1	1.000	0.693	0.605	0.552	-0.043	0.009	-0.006	0.760	5
X2	0.693	1.000	0.847	0.745	0.004	0.006	-0.018	0.845	5
X3	0.605	0.847	1.000	0.877	0.007	0.005	-0.024	0.785	2
X4	0.552	0.745	0.877	1.000	0.011	0.006	-0.032	0.696	0
X5	-0.043	0.004	0.007	0.011	1.000	-0.008	0.020	-0.318	-5
X6	0.009	0.006	0.005	0.006	-0.008	1.000	-0.046	-0.310	-5
X7	-0.006	-0.018	-0.024	-0.032	0.020	-0.046	1.000	-0.133	-2

As one can see, Table 2 mirrors Table 1. For this dataset, however, the correlation structure is more complicated. X_1 and X_2 are highly correlated with $\rho = .7$.

As seen in Figure 4, the pattern observed between X_1 and X_2 does not carry over to the other correlated predictors.

Figure 5 demonstrate how the correlation between a few of the predictors and Y may be effected by slope. Scale is much more a factor in this dataset, with some variables like X_3 having a larger range than the variables $X_1 \sim N(0, 1)$ or $X_5, ..., X_{12} \sim MVN()$.

The last dataset we'll consider is D_3, a data set with even more non-linear relationships between the first four variables. Otherwise it is very similar to both D_1 and D_2. The first four variables are generated as follows:

$$\omega_1 \sim N(1, 0)$$

$$\omega_2 = log(\omega_1) + E, E \sim N(1, 0)$$

$$\omega_3 = log(\omega_2) + E, E \sim N(1, 0)$$

$$\omega_4 = log(\omega_4) + E, E \sim N(1, 0)$$

	W1	W2	W3	W4	W5	W6	W7	y	beta
W1	1.000	-0.056	-0.040	0.041	0.002	-0.034	-0.028	0.322	5
W2	-0.056	1.000	-0.533	-0.279	-0.002	0.049	-0.003	0.668	5
W3	-0.040	-0.533	1.000	-0.002	-0.019	-0.031	-0.010	-0.096	2
W4	0.041	-0.279	-0.002	1.000	-0.007	-0.008	-0.079	-0.223	0
W5	0.002	-0.002	-0.019	-0.007	1.000	-0.012	-0.019	-0.382	-5
W6	-0.034	0.049	-0.031	-0.008	-0.012	1.000	0.004	-0.358	-5
W7	-0.028	-0.003	-0.010	-0.079	-0.019	0.004	1.000	-0.159	-2

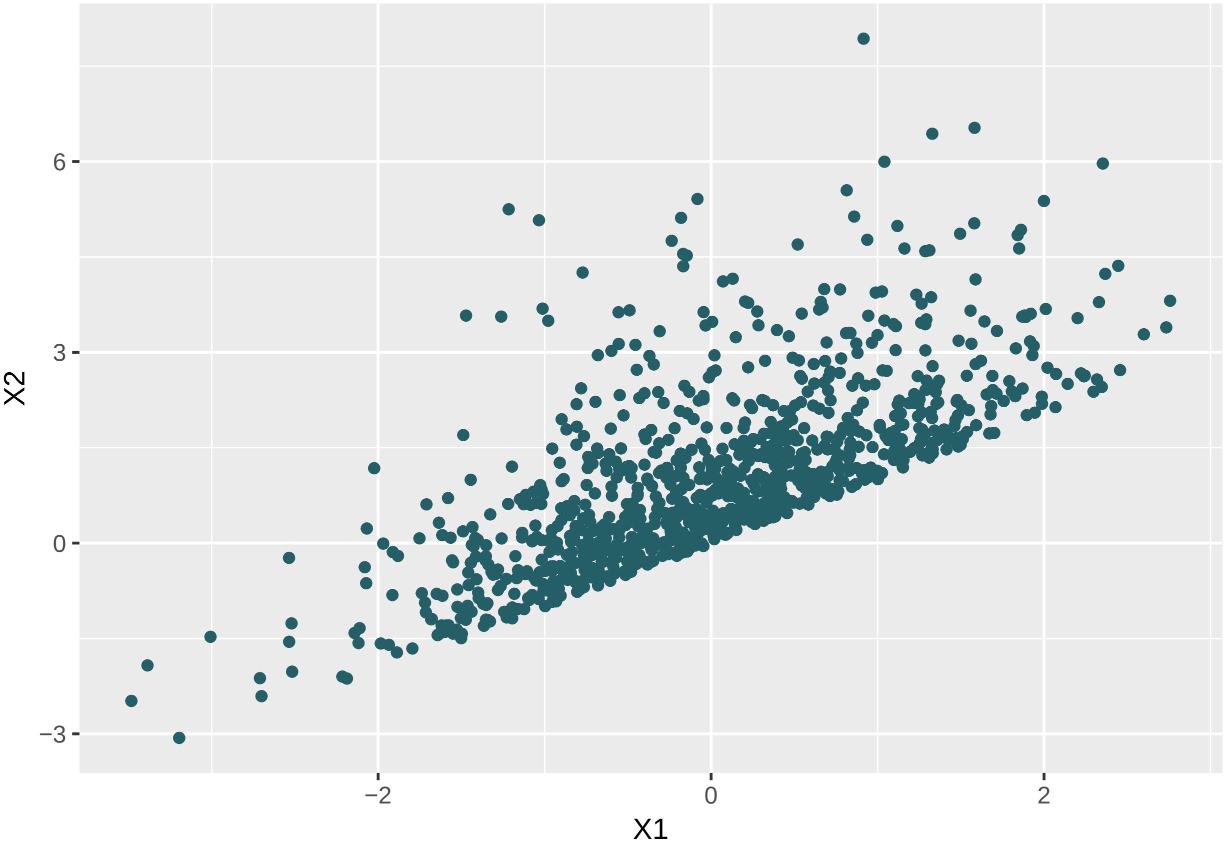


Figure 2.3: Plot of X2~X1, Correlation = .7

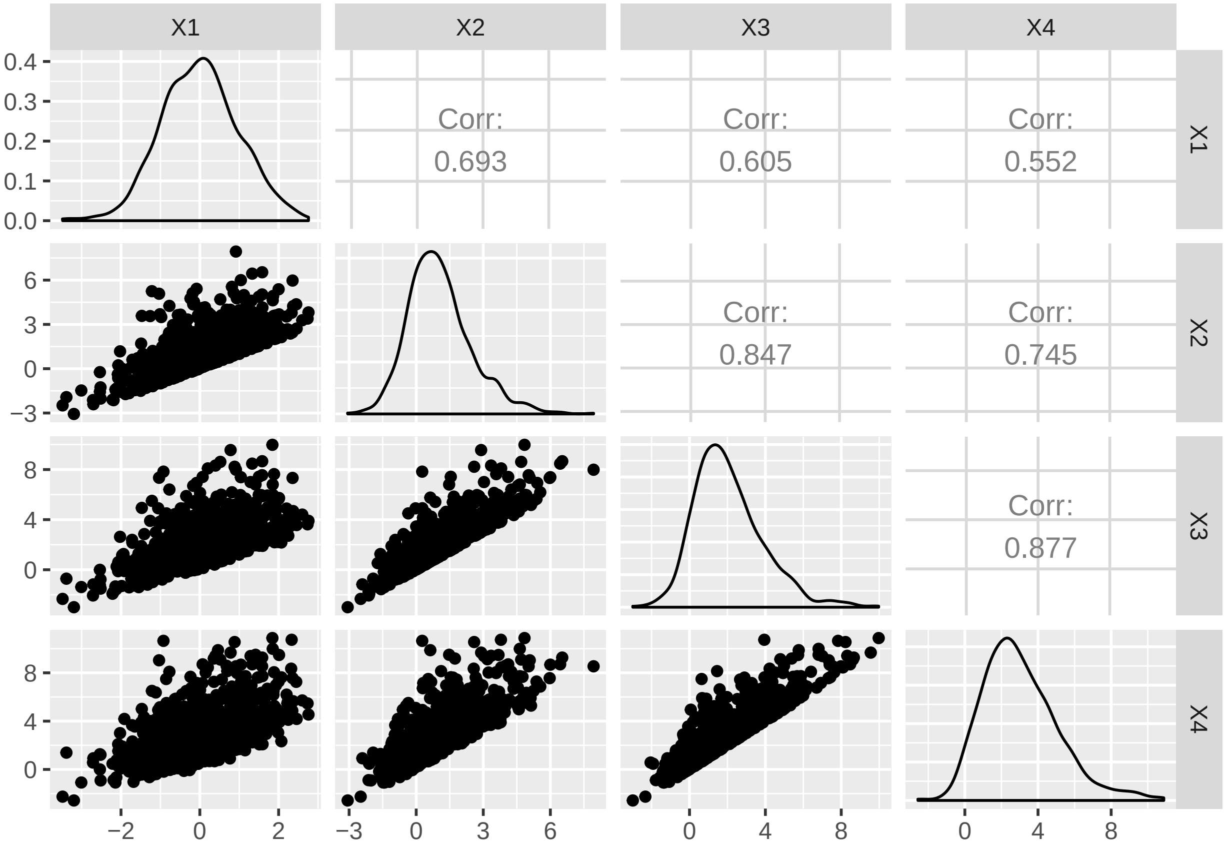


Figure 2.4: Correlation Structure of the First Four Variables in D2

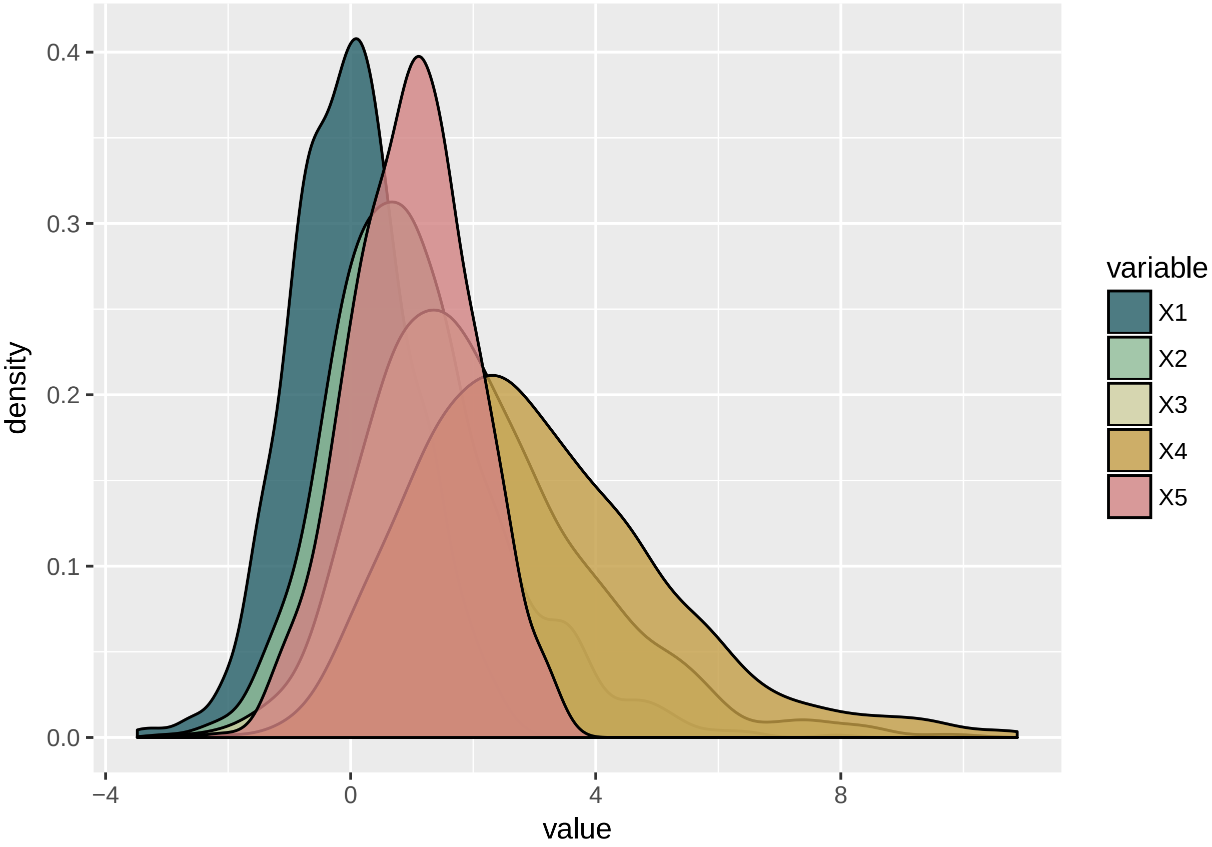


Figure 2.5: Comparisons of the Density Graphs for X1 through X5

The linear correlation structure in D_3 is not as striking as in D_1. The two strongest linear relationships are between ω_2 and ω_3 with $\rho = -.534$ and between Y and ω_2 with $\rho = .700$.

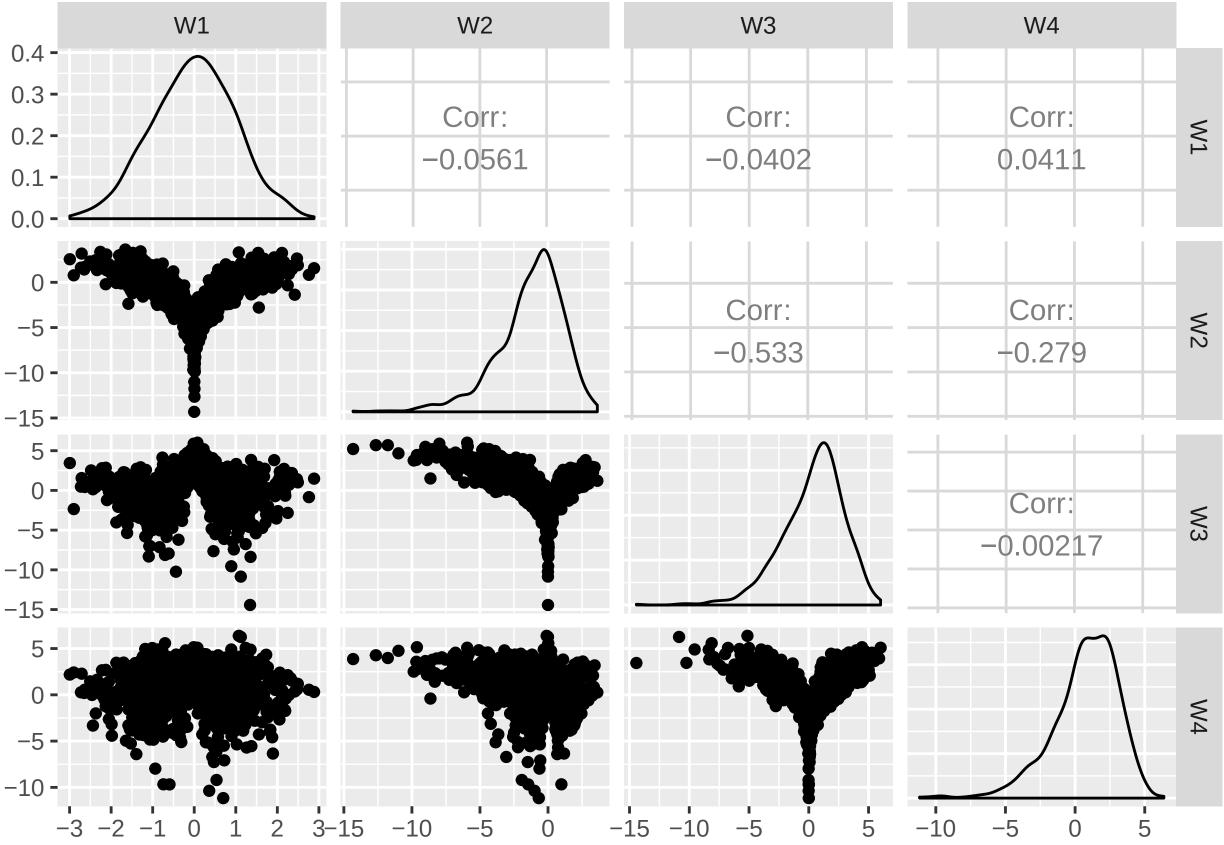


Figure 2.6: Correlation Structure of the First Four Variables in D3

Figure 6 provides another way of visualizing some of the information given in Table 3. Here we can see the densities as well as the paired correlations of the first four variables in D_3.

There is more variation between the densities of $\omega_1, ..., \omega_5$ then we have seen in the other data sets. ω_2, ω_3, and ω_4 have greater spread than their counterparts that are generated under the normal distribution.

As the relationship between Y and ω_2 was so striking, it is nice to see a scatter plot that represents it.

2.2 Models and Comparisons

CART: Regression Trees

As outlined in the 1984 textbook, *Classification and Regression Trees*, Brieman, Friedman, Olshen, and Stone described their method for creating, pruning, and testing regression trees. There are essentially three steps: one, decide on a variable to split

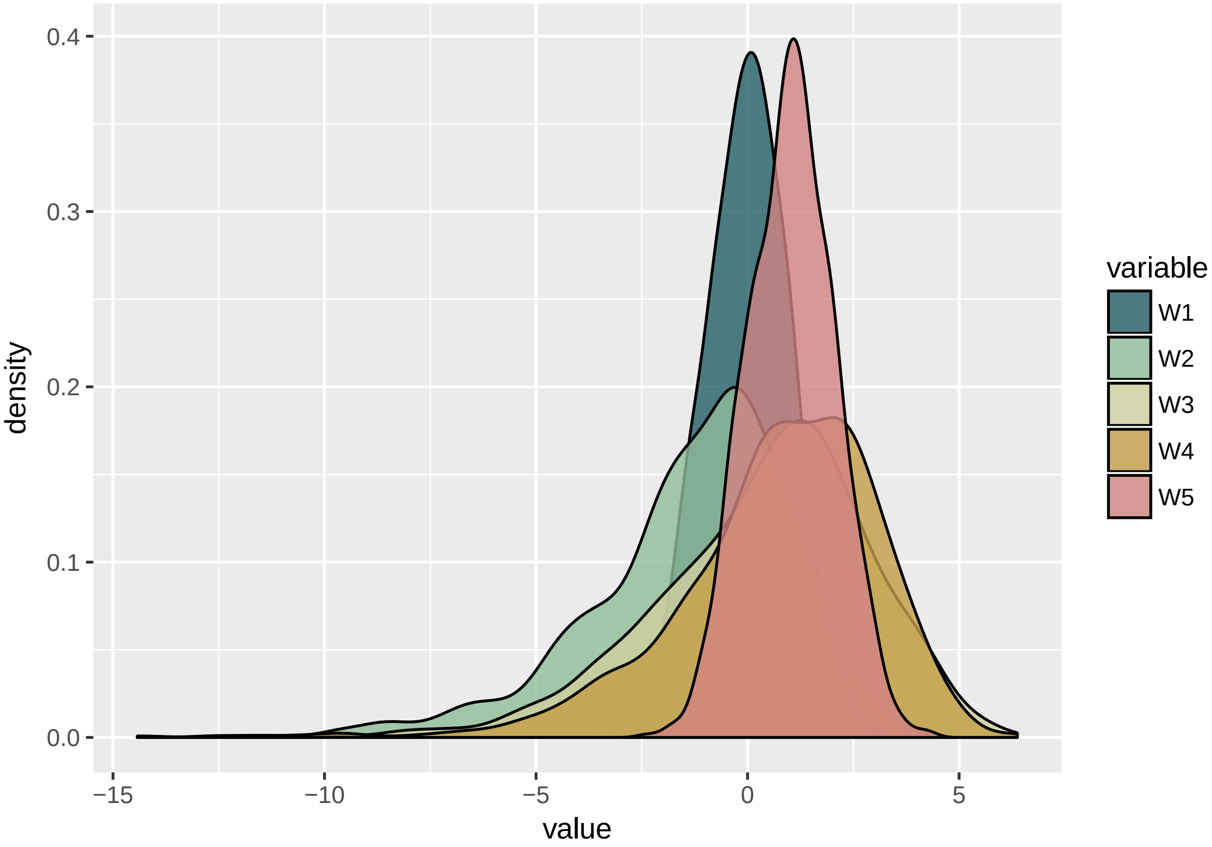


Figure 2.7: Comparisons of the Density Graphs for W1 through W5

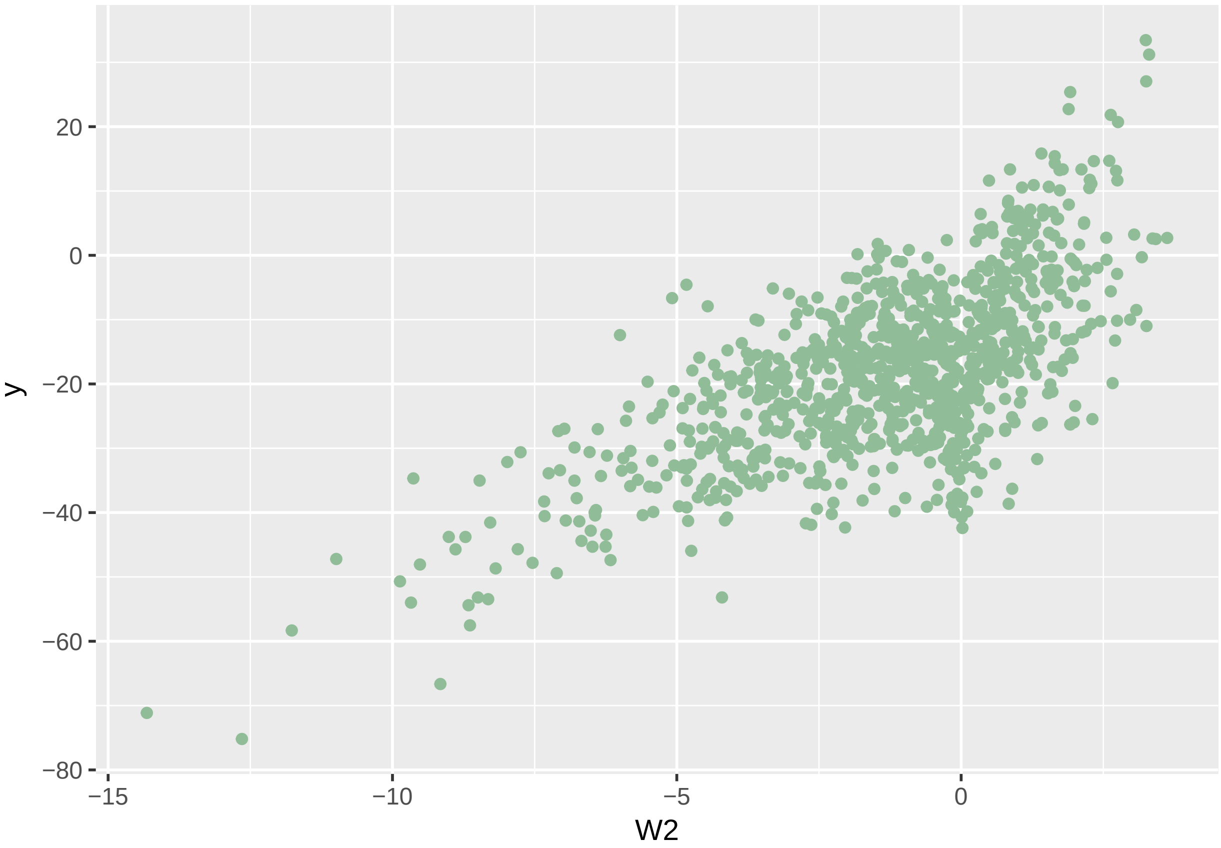


Figure 2.8: Plot of Y~W2, Correlation = .7

over, two, partition that variable space in two distinct partitions, and three, set our initial predictions for each partition to be mean value of the response according to the observed responses corresponding to the values in the partitions. Recursively, this process is repeated for each new partition until some stopping condition is reached.This is a top down, greedy algorithm that functions by creating as large a tree as possible and then is pruned down to prevent over fitting.

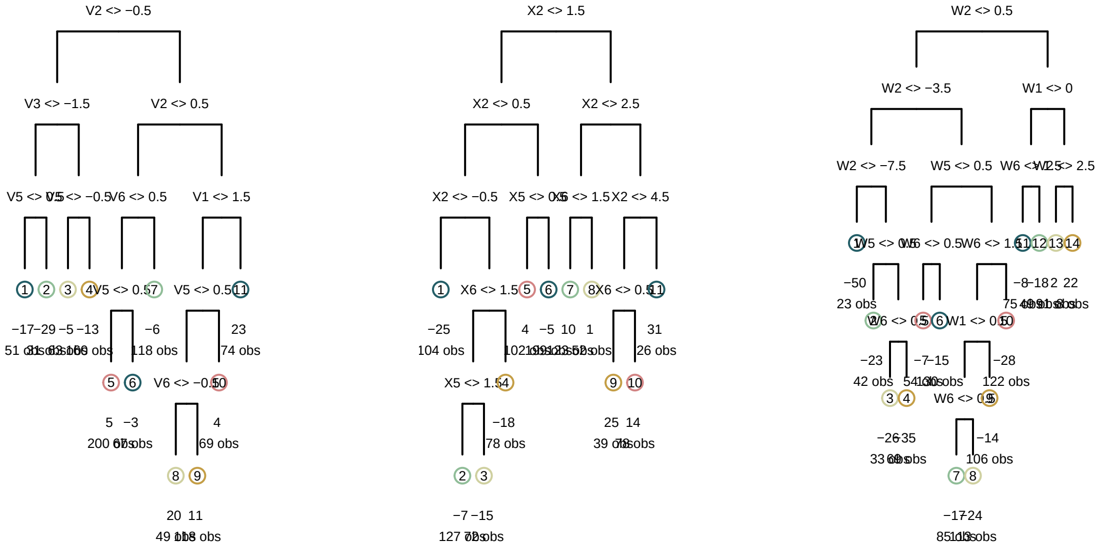


Figure 2.9: CART for the Model Y~, from D1,D2, and D3

Trees can be quite variable, so to get a better idea of the differences between the methods let's run a simulation.

Algorithm 2 Simulation Scheme 2.1

```
for $i \leq 1000$ do
    Randomly sample $\frac{2}{3}$ of the observations in $D_2$ to a training set, $D^i_{2,train}$. The other observations, $x \in D_2, x \notin D^i_{2,train}$ form the testing set $D^i_{2,test}$
    Fit a tree, $T^i$, to the data under the model $Y \sim X_1, ..., X_2$ using the observations in $D^i_2$
    Calculate the $MSE_{test}$ of the model using the equation: $MSE_{test} = \frac{1}{n}\sum(y_j - \hat{y}_j)^2$
end for
```

Where n is the number of observations in $D^i_{2,test}$, $y_j \in D^i_{2,test}$, $\hat{y}_j \in T^i(D^i_{2,test})$ for $1 \leq j \leq n$ This produces two distributions of MSE_{test}, one for CART and one for CTree, conditional inference trees.

The distribution of 1000 CART trees' MSE_{test} is roughly normal with a variance of `var(testmseC)`.

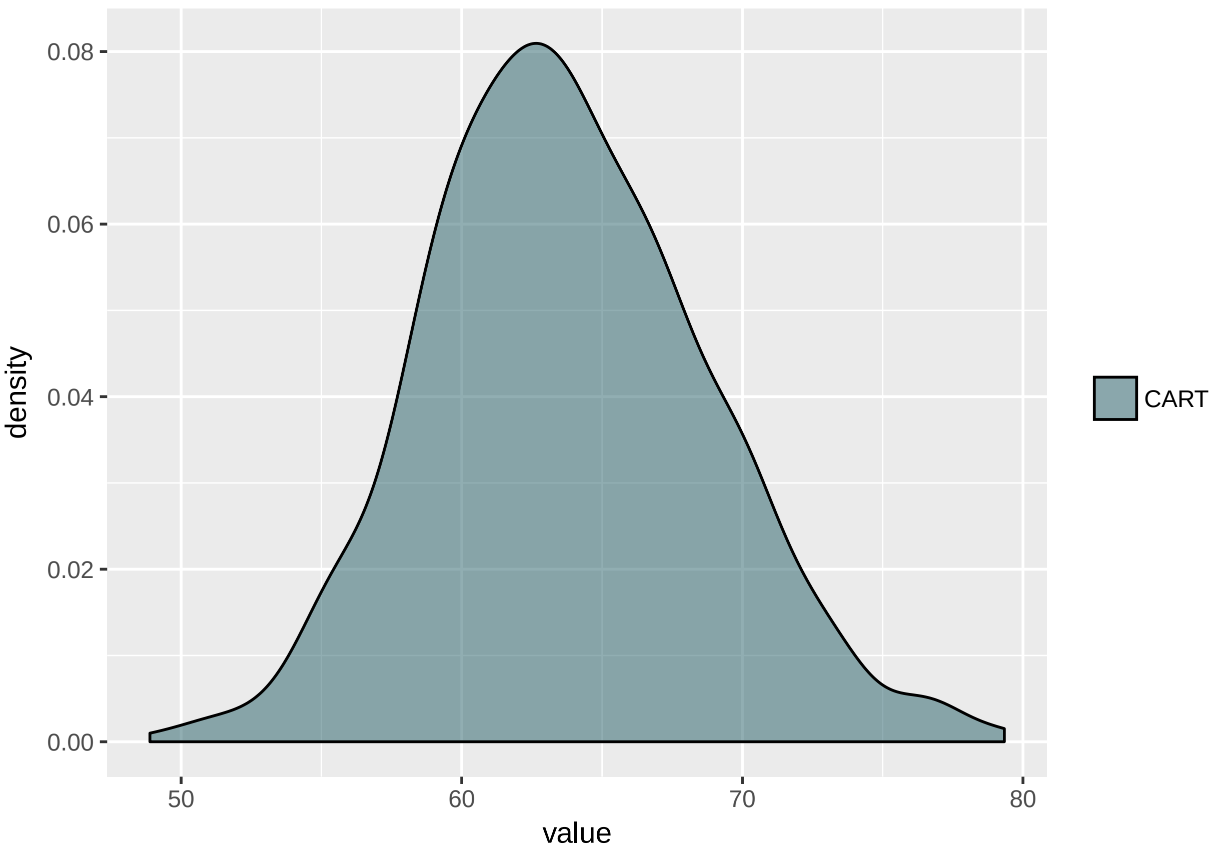


Figure 2.10: Comparison of the Simulated MSEtest Distributions of CART

2.3 Bagged Forests

As one can see in the Figure 10, there is a fair amount of variability in a single tree, they are heavily dependent on fluctuations in the starting data set. As mention briefly in the introduction, bagged forests present one solution to this problem. To create a bagged forest, as outlined in *An Introduction to Statistical Learning* by James, Witten, Hastie and Tibshirani, 2013, many bootstrapped samples are taken from the initial dataset and trees are fitted to them. The final predictions are, then, averaged over all of the trees. This ensures that while each tree has high variance, when they are aggregated the variance will decrease.

Let's put that to the test here using our dataset $D3$ again. We'll build 100 forests of 100 trees each and compare the variability of the MSE distributions.

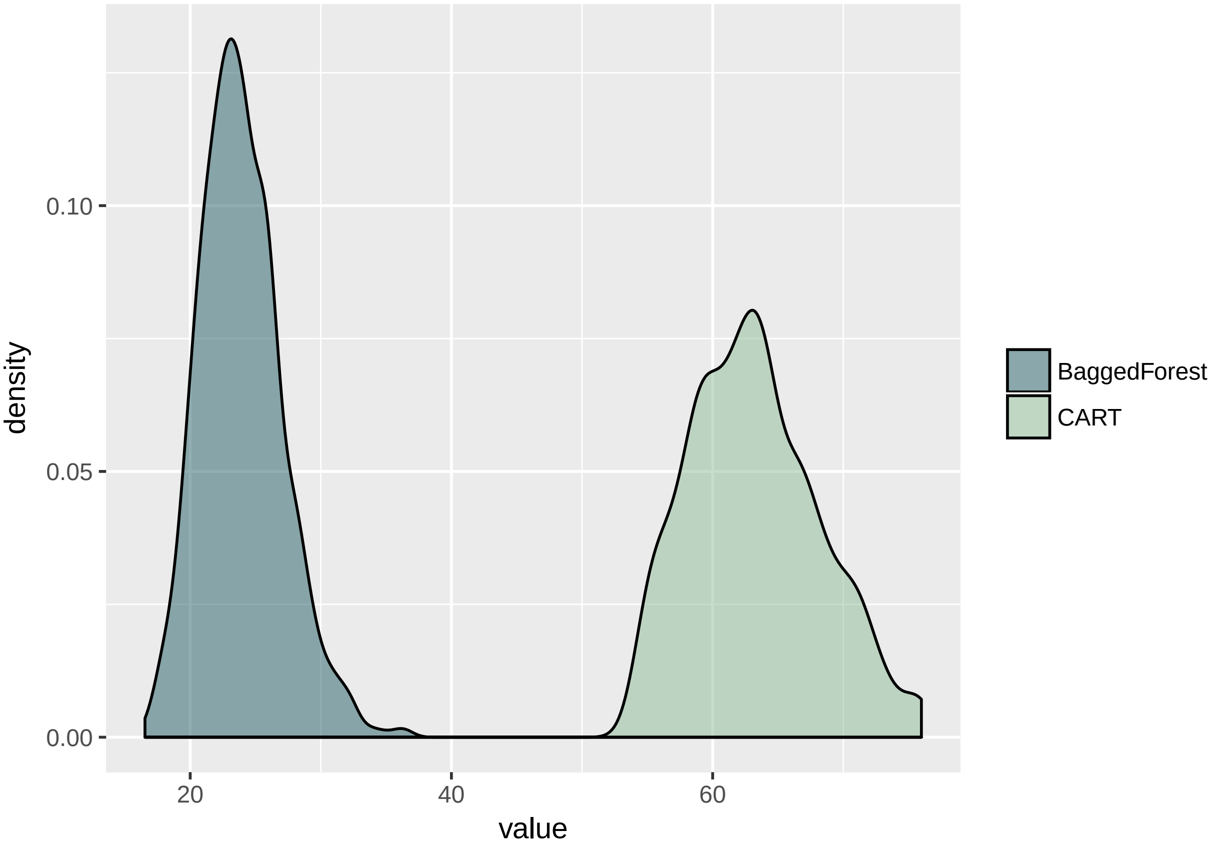


Figure 2.11: the Simulated MSEtest Distributions of Bagged Forests and CART

As one can see, the values of MSE_{test} for the bagged forest were entirely below the MSE_test for the trees and the variance was much smaller.

2.4 Random Forests

EXPAND AND BEGIN NOTATION OF R AS RANDOM FOREST

As the number of trees grown in each forest increases, the MSE_{test} decreases (cite). Still, this can become computationally intensive on larger data sets where we would like very accurate models. Random forests are often seen as a solution to this problem. In a bagged forest, every variable is considered when each split is made but in a random forest only $mtry, mtry \leq p$ are considered. This allows us to assume that the trees have a level of independence not found in bagged forests, and that a small random forest will often out perform the bagged forest.

For an illustration, let's build a random forest on $D3$ and compare the MSE.

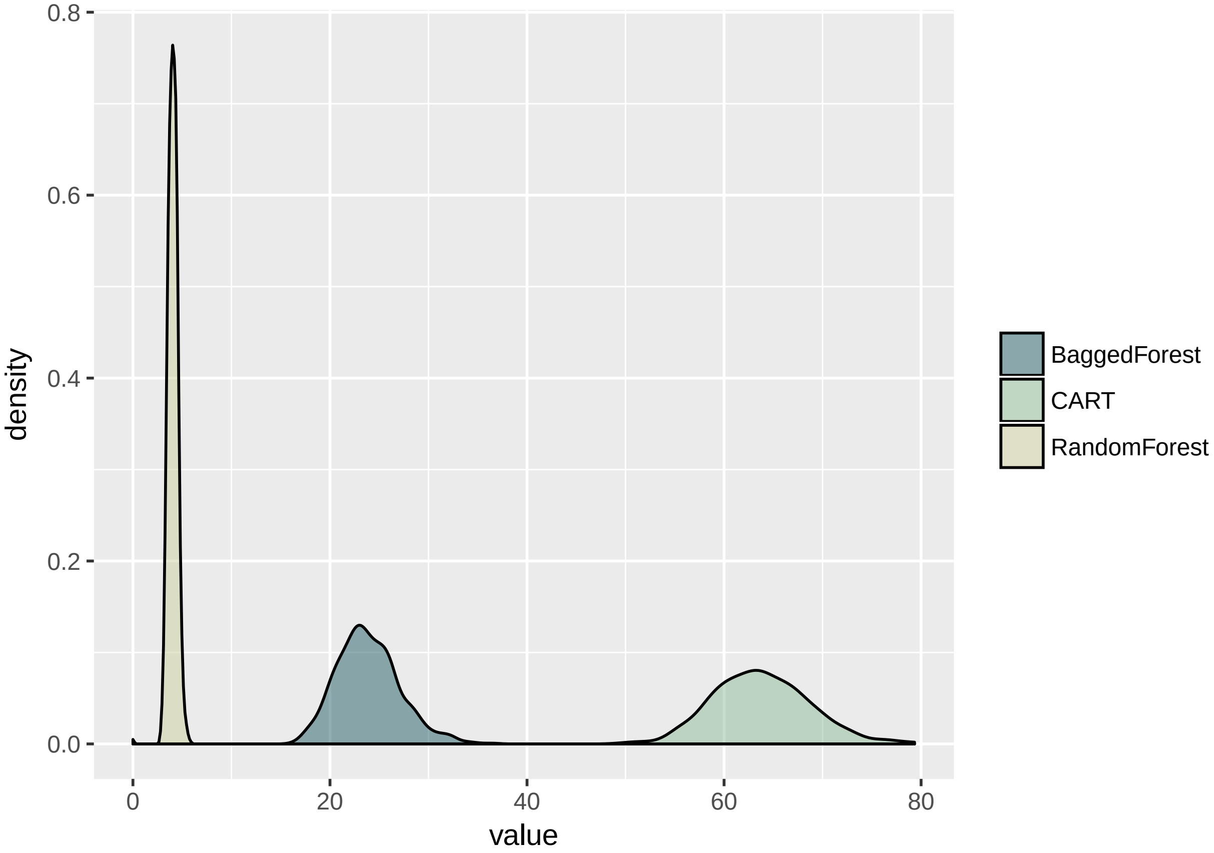


Figure 2.12: Simulated MSEtest Distributions of CART, Random, and Bagged Forests

Chapter 3

Random Forest Variable Importance

3.1 Breiman et al Introduce Permuted Variable Importance (1984)

3.1.1 Variable Importance on a Single Tree

Breiman et al in *Classification and Regression Trees* (1984) propose a method for variable importance for individual trees that stems from their definition of $\tilde{s}$, a surrogate split. Surrogate splits help Brieman et al deal with several common problems one may have: modeling with missing data, diagnosing masking, and variable importance. They are defined using logic that resembles that behind random forests.

Definitions

Assume the standard structure for tree models. Let D be the dataset composed of $D = Y, X_1, ...X_p$, where the model we would like to estimate is of the form $T : Y \sim X_1, ...X_p$. For any node $t \in T(D)$, $s*$ is the best split of the node into daughters t_r and t_l. Take $X_i \in D$ and let S_i be the set of all of the splits on X_i in T. Then set $\bar{S}_i$ equal to the complement of S_i, $\bar{S}_i = S_i^c$. For any possible split $s_i \in S_i \cup \bar{S}_i$, s_i will split the node t into two daughters, $t_{i,l}$ and $t_{i,r}$. Count the number of times that $s*$ and s_i, while splitting differently, generate the same left daughter t_l as N_{LL} and the number of times they generate the same right daughter as N_{RR}. Then the probability that a case falls within $t_L \cap t'_L$ is $P(t_L \cap t'_L) = \sum_j \frac{\pi(j)N_j(LL)}{N_j}$ and the probability that a case falls within $t_R \cap t'_R$ is $P(t_R \cap t'_R) = \sum_j \frac{\pi(j)N_j(RR)}{N_j}$. Where $\pi(j)$ is the prior assumed for the the jth variable.Finally, the probability that a surrogate split predicts $s*$ is $P(s*, s_M) = (t_R \cap t'_R) + P(t_L \cap t'_L)$. Then the surrogate split is the value of $s*$ that maximizes this probability. It is denoted $\tilde{s}$

A surrogate split $\tilde{s}$,is one that estimates the best possible univariate split $s*$ on node t.

Defintion: Variable Importance, Single Tree

$$VI_{tree}(X_i, T) = \sum_{t \in T} \Delta RSS(\tilde{s_i}, t)$$

Or the decrease of RSS attributable to X_i across the tree T. In *Classification and Regression Trees*, Brieman et al, outline several potential problems with this method that the do not attempt to solve. First, that this is only one of a number of reasonable ways to define variable importance. Second, the variable importances for variables $X_1, .., X_p$ can be effected by outliers or random fluctuations within the data. (Ch 5.3)

3.1.2 Variable Importance for a Random Forest

One way to define variable importance for a random forest follows directly from Breiman et al's definition for a single tree. Recall that each tree in a random forest is fit to a bootstrapped sample of the original observations. To estimate the test error, therefor, no cross validation is needed - each tree is simply tested against the test set of observations that were not in that tree's initial training set. To determine variable importance for a predictor X_j, we look at the RSS of the each tree's prediction that did not split on X_j. These values are then averaged over the subset forest that did not include X_j. A large value would imply that in trees that included X_j, the predictive capabilities were increased.

To formalize that idea, let's refer to the set of trees that did not consider X_j, $T^c_{x_j}$. Now, $T^c_{x_j} \subset R$, the random forest. The subset of the original data that will be tested on each tree, t, is $\bar{B}^t$. The dimensions of $\bar{B}^t$ are ν_t x p, where p is the number of predictors and $\nu \leq n$. The number of trees in $T^c_{x_j}$ is μ where $\mu \leq ntree$

Now, base variable importance is:

$$VI_\alpha(X_j, R) = \sum_{t \in T^c_{x_j}} \frac{1}{\nu_t} RSS(t, \bar{B}_t)$$

However, this method poses some problems. Namely, while variable importance for random forests is more stable than for the variable importance values for CART, (this is because the model is less variable in general), it is lacking the traditional inferential capabilities of other regression models. In an effort to derive a p-value for variable importance values, Breiman 2001b, describes a *permuted variable importance* or VI_β that does not utilize $T^c_{x_j}$.

Again, a large variable importance value suggests that X_j is a valuable predictor for the model.

3.2 Strobl et al Respond (2008)

Strobl et al (2008) respond to Breiman's method with one main argument: the null hypothesis implied by the permutation distribution utilized in permuted variable importance is that X_i is independent of Y **and** $X_j \notin X_1, ..., X_p$ so the null hypothesis will be rejected in the case where X_j is independent of Y but not some subset of the

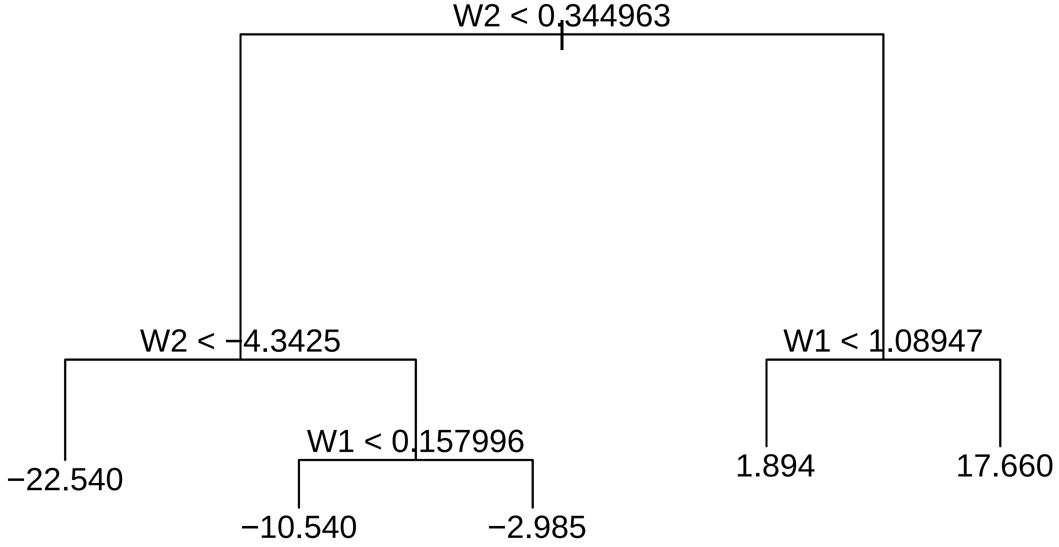
Algorithm 3 Permuted Variable Importance for Random Forests, VI_β

```
Fit a random forest, $R$ on the dataset $D$ fitting the model $Y \sim X_1, ..., X_p$.
for each $X_i \in X_1, ..., X_p$ do
    for each $t \in R$ do
        Calculate: $\Phi_o = \frac{1}{\nu_t} RSS(t, \bar{B}^t)$
        Permute $X_i$. Now find $\Phi^* = \frac{1}{\nu_t} RSS(t, \bar{B}^*_t)$
        The difference between these values, $\Phi^* - \Phi_o$, is the variable importance for $X_j$ on $t$,
    end for
    Average over all $t \in R$
        $$VI_\beta(X_j) = \frac{1}{ntree} \sum^{ntree} \Phi^* - \Phi_o$$
        $$VI_\beta(X_j) = \frac{1}{ntree} \sum^{ntree} \frac{1}{\nu_t} RSS(t, \bar{B}^*_t) - \frac{1}{\nu_t} RSS(t, \bar{B}^t)$$
end for
```

other predictors. As correlation among the predictors is very common in data sets that are used for random forests, this is a large problem for Breiman's method.

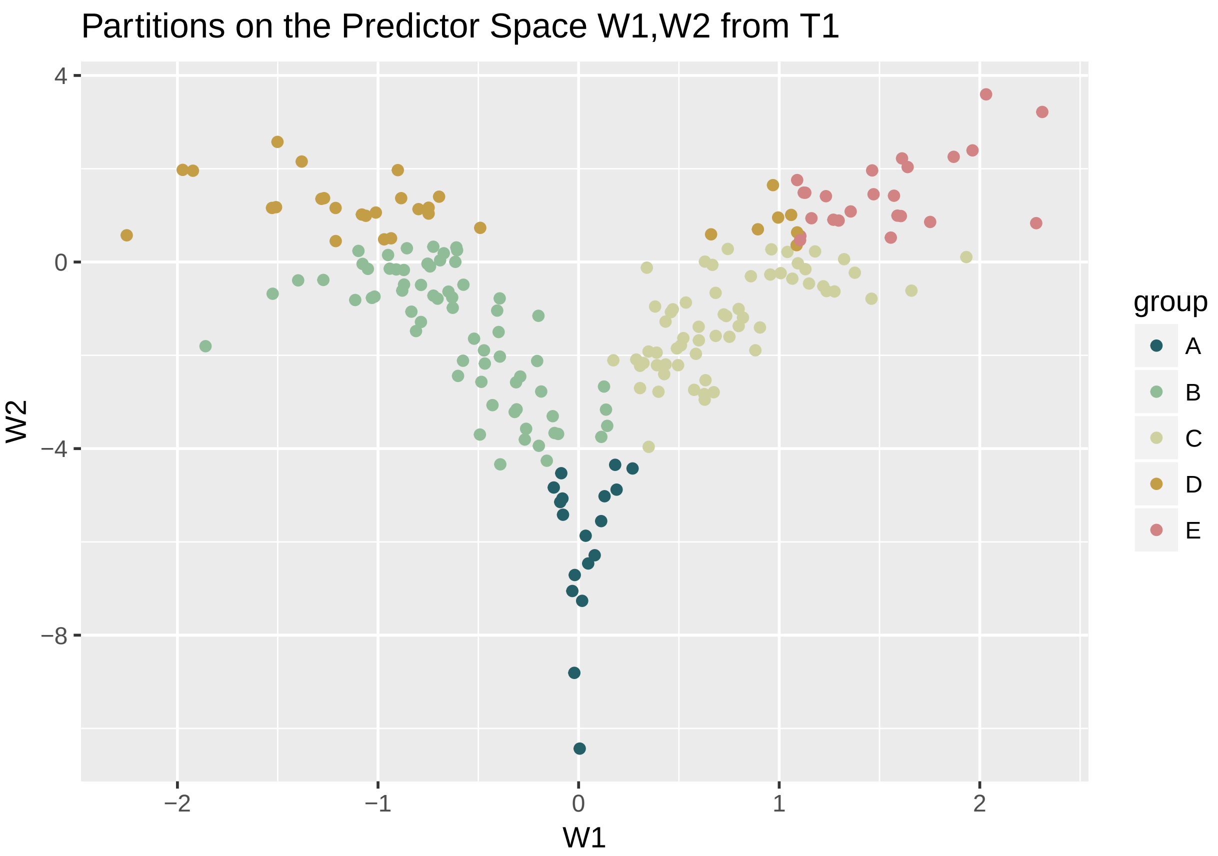
To alleviate this difficulty, Strobl et al propose a permutation scheme under the null hypothesis that X_j given it's relationship with the other predictors is independent of Y.

As an example for 5 in the algorithm above, consider the dataset $D_{3,lite}$, where $D_{3,lite} = \omega_1, \omega_2, y$ where $D_{3,lite}$ has dimensions $n = 100$, $p = 2$. A simple CART tree with 5 splits on the model Y ~ω_1, ω_2



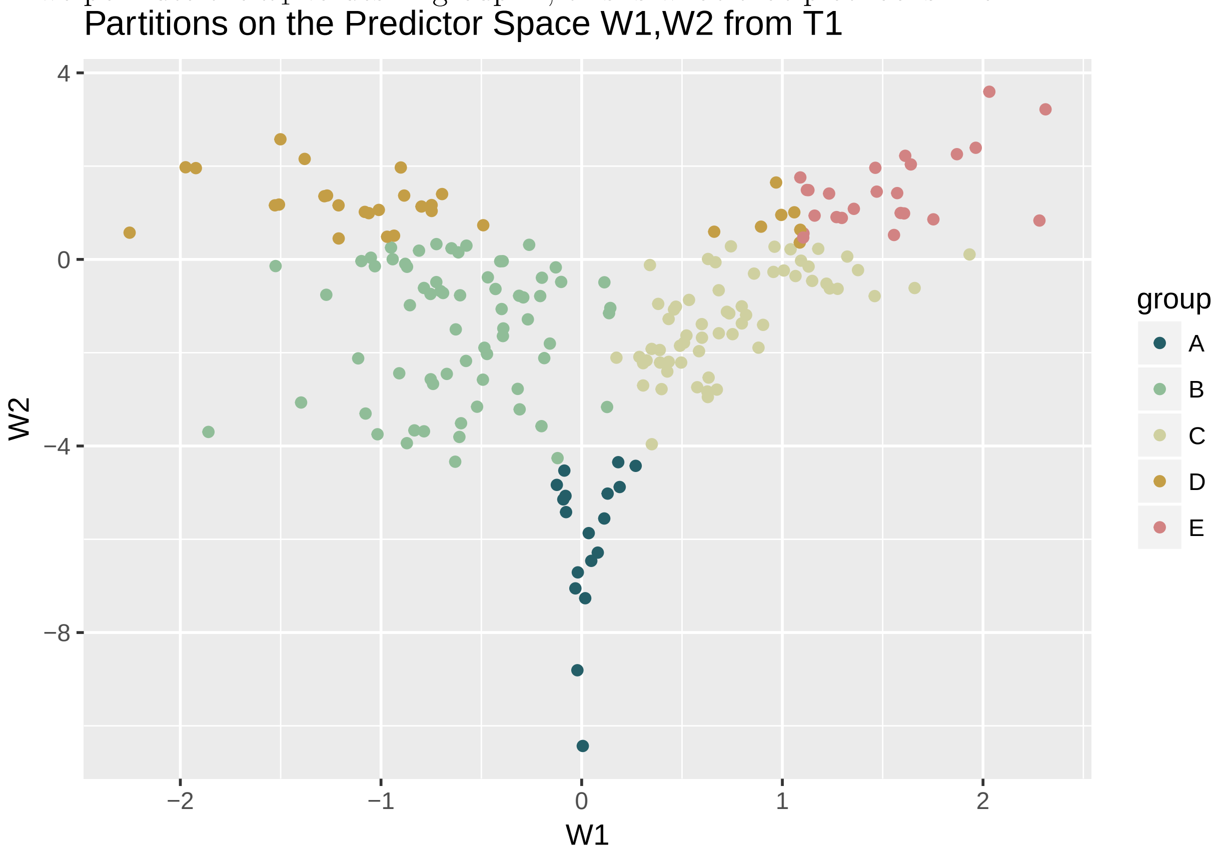
Algorithm 4 Conditional Variable Importance for Random Forests, VI_γ

```
Fit a random forest, $R$ on the dataset $D$ fitting the model $Y \sim X_1, ..., X_p$.
for each $X_i \in X_1, ..., X_p$ do
    for each $t \in R$ do
        Calculate: $\Psi_o = \frac{1}{\nu_t} RSS(t, \bar{B}^t)$
        Permute $X_i$ according to the partitions on $X_j$ from $t$ (see notes below on this step) Now find $\Psi^* = \frac{1}{\nu_t} RSS(t, \bar{B}_t^*)$
        The difference between these values, $\Psi^* - \Psi_o$, is the variable importance for $X_j$ on $t$,
    end for
    Average over all $t \in R$
        $$VI_\gamma(X_i, R) = \frac{1}{ntree} \sum^{ntree} \Psi^* - \Psi_o$$
        $$VI_\gamma(X_i, R) = \frac{1}{ntree} \sum^{ntree} \frac{1}{\nu_t} RSS(t, \bar{B}_t^*) - \frac{1}{\nu_t} RSS(t, \bar{B}^t)$$
end for
```



Group	Predicted Value of Y	Min(W2)	Max(W2)	Min(W1)	Max(W1)
A	-22.5362734865312	-10.43	-4.35	-0.12	0.27
B	-10.5421878802014	-4.34	0.33	-1.86	0.14
C	-2.98460142826064	-3.96	0.28	0.17	1.93
D	1.89361038304057	0.36	2.58	-2.25	1.09
E	17.661072262451	0.47	3.59	1.09	2.31

If we permute the ω_1 values in group B, this is what that plot looks like:



3.3 Inferential Variable Importance

This thesis hopes to be a response to conditional variable importance as outlined by Strobl et al 2008. First is that the practice of permuting given the partitions from the model $Y \sim X_1, ..., X_p$ instead of $X_j \sim X_1, .., X_p$. This procedure is reminisent of Breiman et al's notions of grouped predictors in the book *Classification and Regression Trees.*

Chapter 4

INFTrees and INFFOREST Variable Importance

4.1 Theory

While conditional variable importance (Strobl et al) conditionally permutes each variable given the structure signified by the model that predicts the response, $Y \sim X_1, ..., X_i, ..., X_p$, our method conditionally permutes each variable given the structure outlined in a new model with the variable of interest as the response, $X_i \sim X_1, ...X_{i-1}, X_{i+1}, ...X_p$. This is not the most straightforward process, as trees partition the sample space, however, in INFTrees these partitions on the variables $X_1, ...X_{i-1}, X_{i+1}, ...X_p$ are treated as psuedo partitions on the variable of interest, X_i. This is accomplished by first partitioning on the sample predictors $X_1, ...X_{i-1}, X_{i+1}, ...X_p$ and then infering the partitions on X_i. As a visualizaiton of this, lets return to the D_3 dataset discussed in chapter 2.

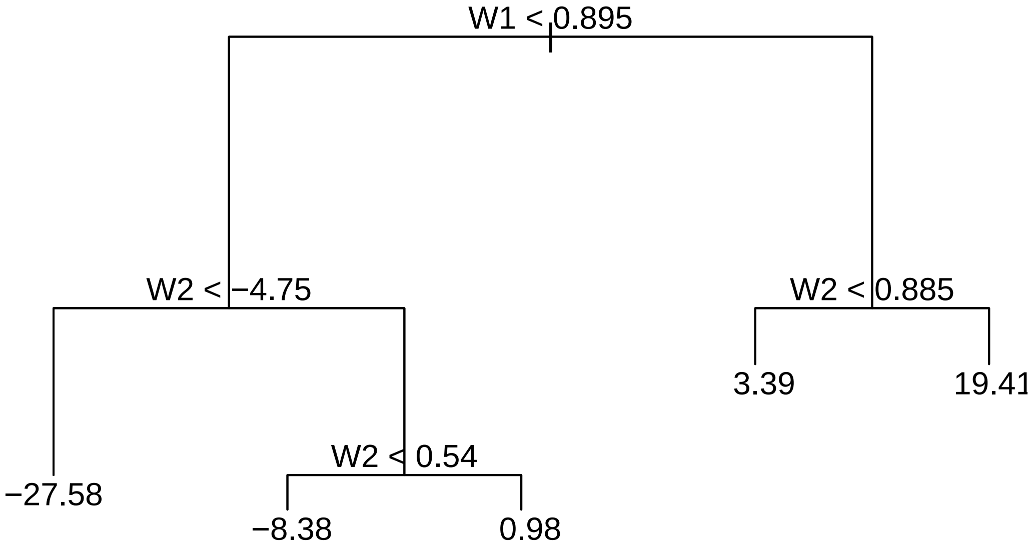


Figure 4.1: A Tree of the Model Y~W1,W2

Lets say we are interested in the variable importance of ω_2. Then using the conditional variable importance (Strobl et al)'s permutation scheme, we would first look at the partitions on ω_2 from this tree.

Clearly, the values of ω_2 are less important to the patitioning structure than the interations of ω_2 and the other variables.

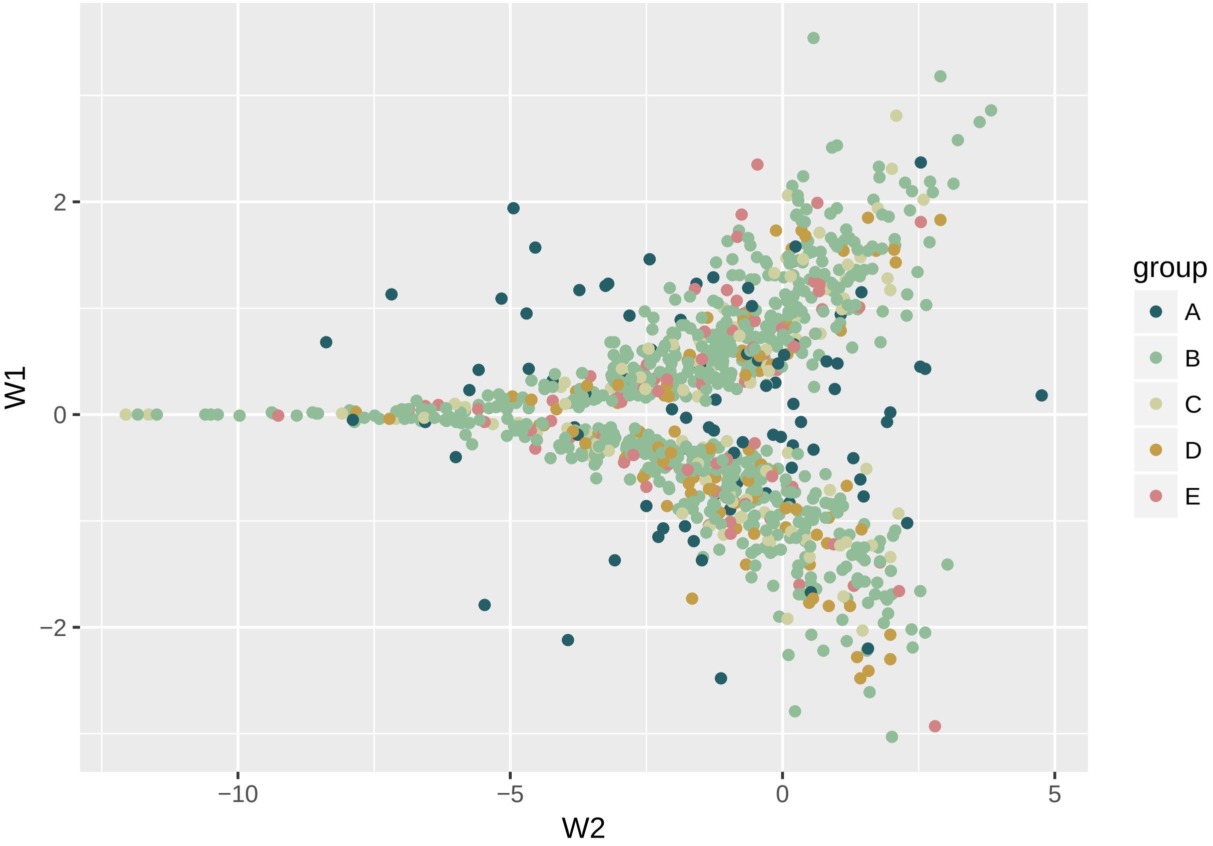


Figure 4.2: Partitions on the Predictor Space W2 from Y~W1,..,W4

As you can see in Figure **??** above, . . .

Under the INFTrees method, before permuting, fit another tree to the model $\omega_2 \sim \omega_1$

The partitions on ω_2 implied by this model are:

4.1.1 INFTrees

For a CART, T, representing the model $Y\ X_1, ..., X_p$ where $Y, X_1, ..., X_p$ are vectors of length n, the INFTrees algorithm proceeds as follows:

This procedure allows the null hypothesis that Y is independent of X_i given the values of $X_1, ...X_{i-1}, X_{i+1}, ...X_p$ to be tested. Therefor, values of VI_{inf} could be compared in a similar manner to the coefficients of linear regression.

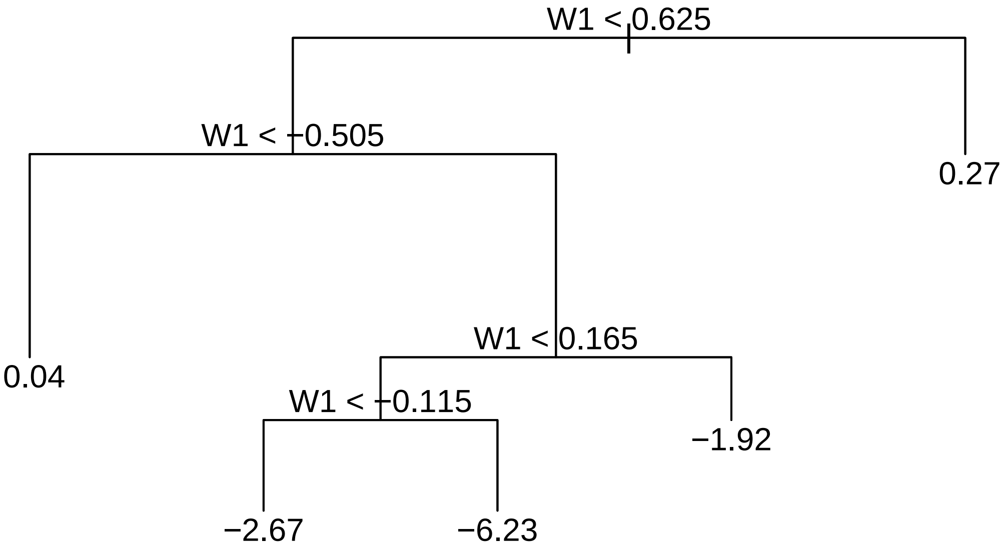


Figure 4.3: A Tree of the Model W2~W1

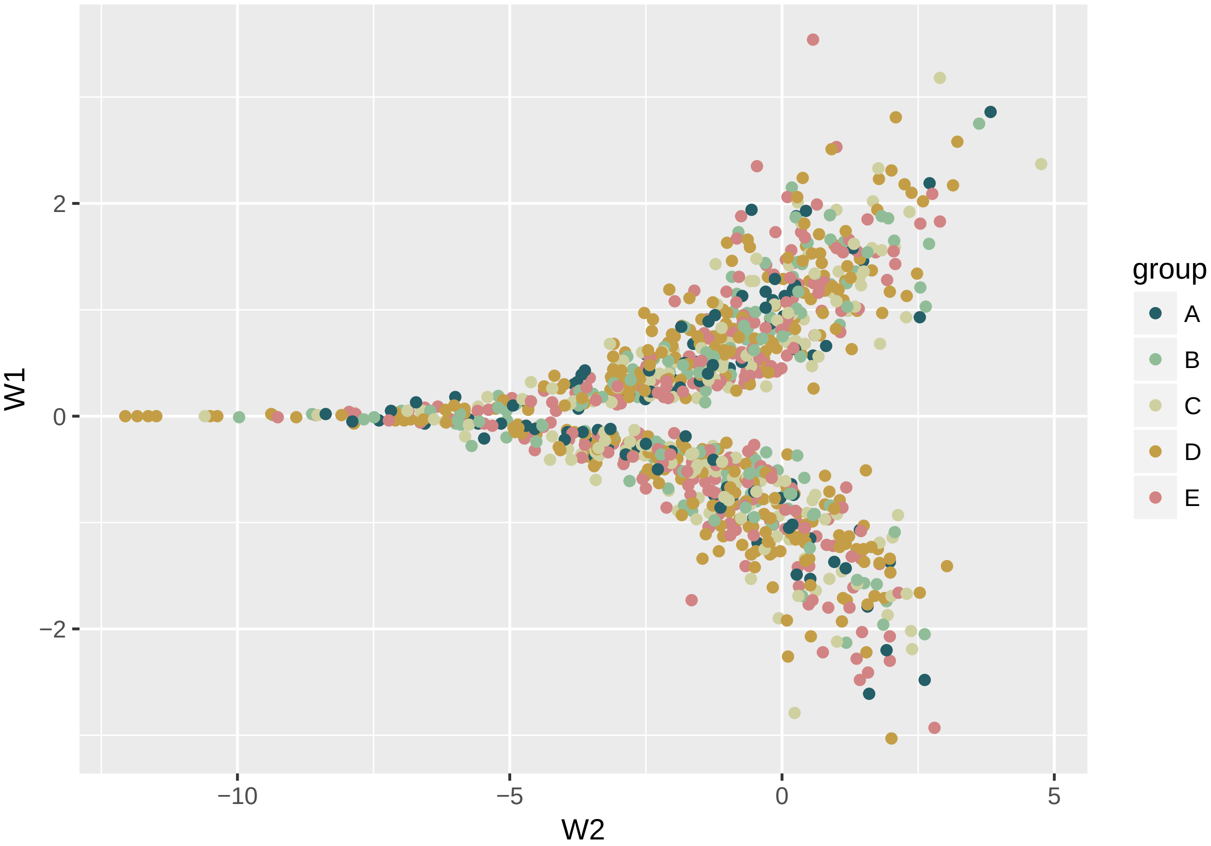


Figure 4.4: Partitions on the Predictor Space W2 from W2~W1

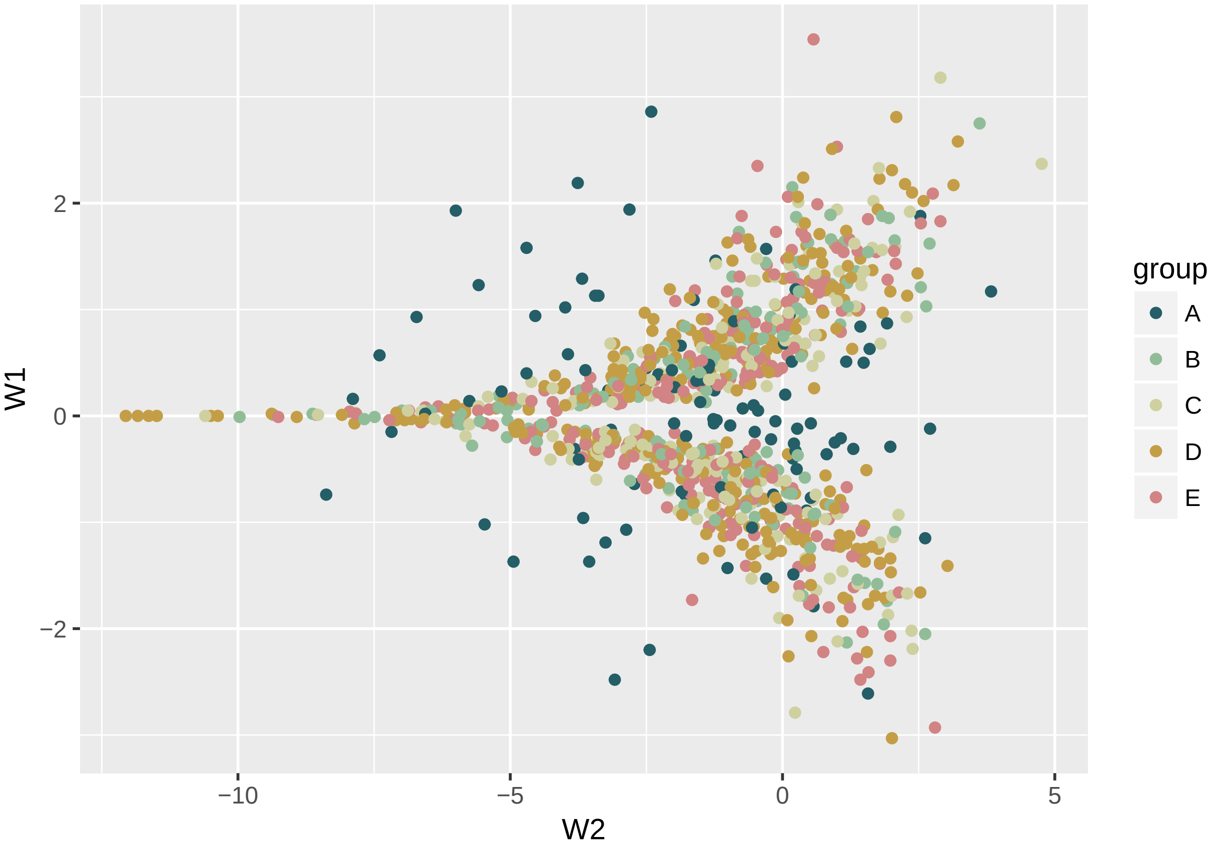


Figure 4.5: The Result of Permuting W2 WRT The Partitions

Algorithm 5 INFTree, $VI_{inf}(T)$

```
for each $X_i \in X_1, ..., X_p$ do
    Calculate: $\Phi_o = RSS(T, (Y, X_1, ..X_p))$
    Fit the tree $T_{X_i}$, where $T_{X_i} : X_i \sim X_1, ..., X_{i-1}, X_{i+1}, ...X_p$
    Extract the set $P_{X_i}$ of partitions on $X_i$ from $T_{X_i}$
    Permute $X_i$ with respect to $P_{X_i}$
    Find $\Phi^* = RSS(T, (Y, X_1, ..., \bar{X}_i, ...X_p))$
    Repeat the above procedure to find the distribution of $\Phi^*$
    Test the null hypothesis that $\Phi_o$ is the likely value of $RSS(T, (Y, X_1, ..X_p))$
end for
```

4.1.2 INFForests

The algorithm for determining $VI_{inf}(R)$ follows similarly.

Algorithm 6 INFForests, $VI_{inf}(R)$

```
Fit a random forest, $R$ on the dataset $D$ fitting the model $Y \sim X_1, ..., X_p$.
for each $X_i \in X_1, ..., X_p$ do
    for each $t \in R$ do
        Calculate: $\Xi_o = \frac{1}{\nu_t} RSS(t, \bar{B}^t)$
        Calculate a tree $T_i$ that predicts $X_i \sim X_1, ..., X_{i-1}, X_{i+1}, ...X_p$ using the subset of the observations used to fit $t$
        Permute the subset of $X_i$ contained in $\bar{B}_t$ with respect to the set of partions $P_{xi}$ from $T_i$.
        Now find $\Xi^* = \frac{1}{\nu_t} RSS(t, \bar{B}_t^*)$
        The difference between these values, $\Xi^* - \Xi_o$, is the variable importance for $X_i$ on $t$
    end for
    Test the null hypothesis that $\Xi_o$ is the likely value of $\frac{1}{\nu_t} RSS(t, \bar{B}_t^*)$ using the distribution of values of $\Xi^*$ gathered from each tree in $R$
end for
```

4.2 Implementation In INFTREES and Results

4.2.1 Notes on the Implemetation

Implementing the INFFOREST and therefor the INFTREES algorithms, required creating a suite of functions to create trees and random forests. The trees are fit following the standard two-part CART-like algorithm.[1] The function chooses a variable to split on with linear correlation with respect to Y, but instead of looking for correlations above a certain threshold which is common, it chooses the variable with the highest correlation when compared to its peers. This alleivates the situation where a variable with a non-linear relationship would be passed over again and again. The splitting is done via minimization of the following function with respect to i:

$$RSS_{node}(i, X, Y) = RSS_{leaf}(Y|X < i) + RSS_{leaf}(Y|X \geq i)$$

$$RSS_{leaf} = \sum (y - \hat{y})^2$$

$$\hat{Y} : \hat{y} \in \hat{Y} : \hat{y} = E(Y),\ where\ |\hat{Y}| = |Y|$$

This function considers the regression case only, and only numeric predictors. Leafs are created when the resultant split would be unsatisfactory, i.e. at least one daughter

[1] A great deal of effort was undertaken by the author to find the defenative, authentic CART algorithm. This implementation follows the rough strokes set out in the 1984 text *Classification and Regression Trees* to the best of the author's ability and may not be exactly the algorithm found in R packages like 'tree()'

node would have five members or less. This generates very large trees - a quality that is not an issue in random forests but may be problematic in a stand-alone setting. At this time, there is also no function to prune the trees.

The INFTREE function follows the algorithm above *reference*. The partitions on X_j are generated by fitting a tree, T, to the model $X_j \sim X_1, ..., X_{j-1}, X_{j+1}, ..X_p$ and calculating the predictions $T(X_1, ..., X_{j-1}, X_{j+1}, ..X_p)$. Then permuting X_j with respect to the partitions on X_j given by those predictions. For example, if $x_j \in X_j$ and the value of $T(x_1, ..., x_{j-1}, x_{j+1}, ..x_p)$ corresponding to x_j is α, x_j is permuted along with the other values of X_j that also have $T(x_1, ..., x_{j-1}, x_{j+1}, ..x_p)$ corresponding to α.

4.2.2 Results

NOTE INFFOREST, like any random forest method involving tree- level calculations is a computationally intensive function. The forests are large, unpruned at any level, and INFFOREST takes time to compute. Because of this reason the datasets discussed in CH2 have been altered so that instead of 1000 x 13 dimensional datasets they are 400 x 13. This decreases computation time immensely. (**see figure _____ in appendix**)

FIGURE OF INFFOREST DISTRIBUTION FOR EACH DATASET

There a little suprises in the distru

In the situation where there is little correlation between the predictors, the distribution of the INFFOREST output is a sharp peak ending at one of the end points, zero or one. When there are, however, strong correlations between the predictor variables, and `mtry` is suitably large but smaller than `p`, the trees in the forest must decide between them. In these situations, the INFFOREST distribution is multimodal, with one peak at one end of the interval, $INFFOREST(X_i) = 1$ and another when $INFFOREST(X_i) = 0$.

To demonstrate this situation, take the dataset $D2$, as described above. In the random forest corresponding to this model, the variables $X2$ and $X3$ are considered substitutes for each other. In the trees where $X2$ has $INFFOREST = 1$, $X3$ has $INFFOREST <<$ and visa versa.

FIGURE OF BOTH INFFOREST DISTRIBUTIONs OF X2 AND X3 TOGETHER FOLLOWED BY THEIR DISTRIBUTIONS CORRESPONDING TO THE SAME TREES

(i.e. the INFFOREST distributions of X2 and X3 in the trees where X3< .5)

Of course, one may be inclined to infer a p-value for the null hypothesis that $INFFOREST = 0$ for each of these variables. This could be done straight-forwardly enough in situations where there is not strong multicolinearity within the predictors as the distributions are relaibly half of the familiar bell shaped curve centered around either zero or one. It would be quite difficult, however, for INFFOREST alone to test the significance of the INFFOREST distribution corresponding to correlated, paired predictors and it may not makes sense to do so at all. *talk with Andrew about fixing this?*

Conclusion

4.1 INFFOREST Comparisons With Other Methods

INFFOREST holds its own amongst the other methods described in Chapter 4. The conditional permuted variable importance, when ran on the same random forest, had more difficulty parsing out the situation with paired variables than INFFOREST.

PLOT OF INFFOREST OUTPUT FOR LIL D2 NEXT TO COND INF FOR LIL D2

Why does this make sense though?

The permuted variable importance that operates without partitioned permuatations ignored the forth variable completely while setting the permuted variable importance for $V2$ to the max value every time. Perhaps this

PLOT OF INFFOREST OUTPUT FOR LIL D2 NEXT TO PERM INF FOR LIL D2

In the simulations considered here, it is difficult to judge which method perfomed the best. In each simulation, the predictors are related to Y by a linear function where the first three, and the fifth through seventh variable had non-zero coefficents. Then, the first four are correlated (see @**??**).

4.2 Data Modeling as a Journey You Take With Some Data You Love

This whole story began with a single node. By itself, a node is nothing but some of your data. It's an interval that could be the entire dataset or a small sample, but may not be clear what the next move should be. Trees are roadmaps through a dataset. Each node is a fork in the road, and each split points out the correct direction.

Chapter 5

The Second Appendix: CTree

5.0.1 Conditional Inference Trees

As mentioned in the introduction, CART has the tendency to bias towards variables with the most possible splits and overfitting. There is little head paid to statistical significance or general statistical theory. *Conditional Inference Trees* are a method proposed by Horthon et al, 2006, that utilizes permutation theory to create and algorithm that is sensitive to these issues. A crutial difference between CTree and CART is that while CART is a top down algorithm, CTree initially assumes each row of the dataset is a node and then gradually prunes them.

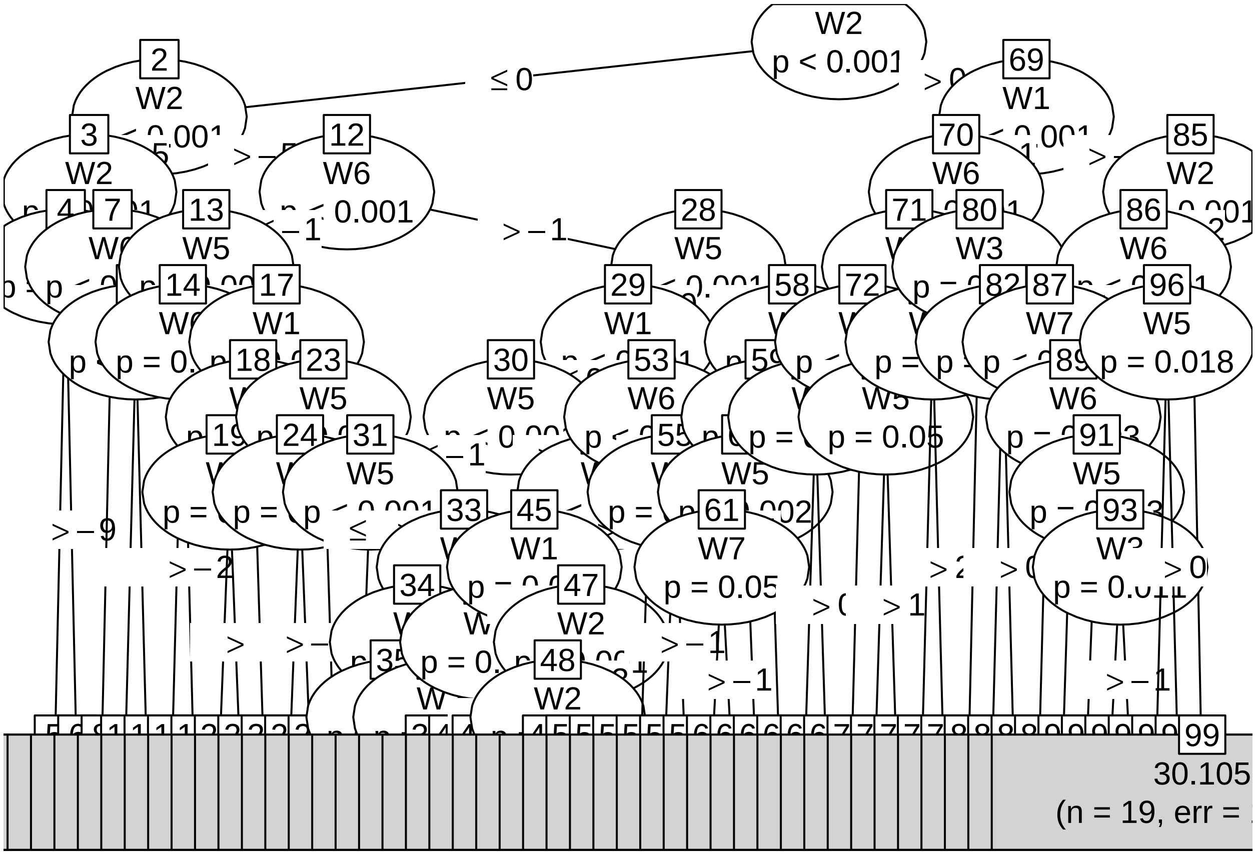
Algorithm 7 Conditional Inference Trees

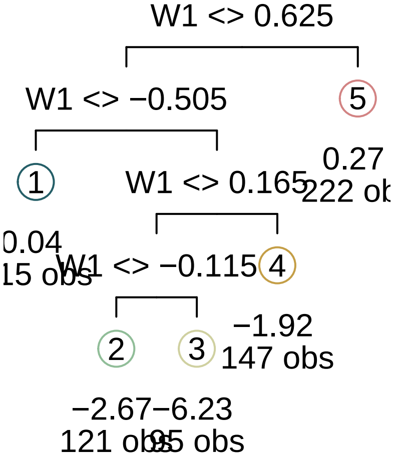
```
for $w_i, i \in \{w_1, ..., w_n\}$ do
    Test the global null hypothesis of independence between any of the $m$ covariates and the response.
    if $H_O$ cannot be rejected then
        Stop
    else
        Select predictor $X_j$ with the strongest linear association to $Y$
    end if
    Choose a set $A \in X_j$ such that $A \cup X_j\ A = A$
    The case weights, $w_{left}$ and $w_{right}$ are then defined as $w_{left,i} = w_i I(x_j \in X_j, \in A)$ and $w_{right,i} = w_i I(x_j \in X_j, x_j \notin A)$
end for
```

The case weights, $w_i \in w_1, ..., w_n$, correspond to nodes are defined as:

$$w_i = I(x_i \in N_t)$$

Where x_i is a vector of observations and N_t is a node in the tree.
CTree fitted to D_3


W2
p < 0.001
≤ 0
> 0
2
W2
p < 0.001
69
W1
3
W2
12
W6
p < 0.001
> −1
70
W6
85
W2
4
7
13
W5
28
W5
71
80
W3
86
W6
14
17
W1
29
W1
58
72
82
87
W7
96
W5
p = 0.018
18
23
W5
30
W5
53
W6
59
89
W6
p = 0.05
19
24
31
W5
55
W5
91
W5
> −9
33
45
W1
61
W7
p = 0.05
93
W3
> −2
34
47
W2
> −1
35
48
W2
99
30.105
(n = 19, err =


W1 <> 0.625
W1 <> −0.505
5
0.27
222 ob
1
0.04
15 obs
W1 <> 0.165
W1 <> −0.115
4
−1.92
147 obs
2
3
−2.67
121 obs
−6.23
95 obs

References

Angel, E. (2000). *Interactive computer graphics : A top-down approach with opengl.* Boston, MA: Addison Wesley Longman.

Angel, E. (2001a). *Batch-file computer graphics : A bottom-up approach with quicktime.* Boston, MA: Wesley Addison Longman.

Angel, E. (2001b). *Test second book by angel.* Boston, MA: Wesley Addison Longman.