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:

1. Fit a random forest to the model, R_0, and calculate base variable importance for each variable V
2. For every predictor $X_j \in X_1, ..., X_n$:

- 2a. Conditionally permute X_j given the splits found in R_0
- 2b. Fit a new random forest R_j with the permuted data
- 2c. Calculate a new variable importance $\hat{V}_j$

3. 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 $X_1, ..., X_n$.

Chapter 2

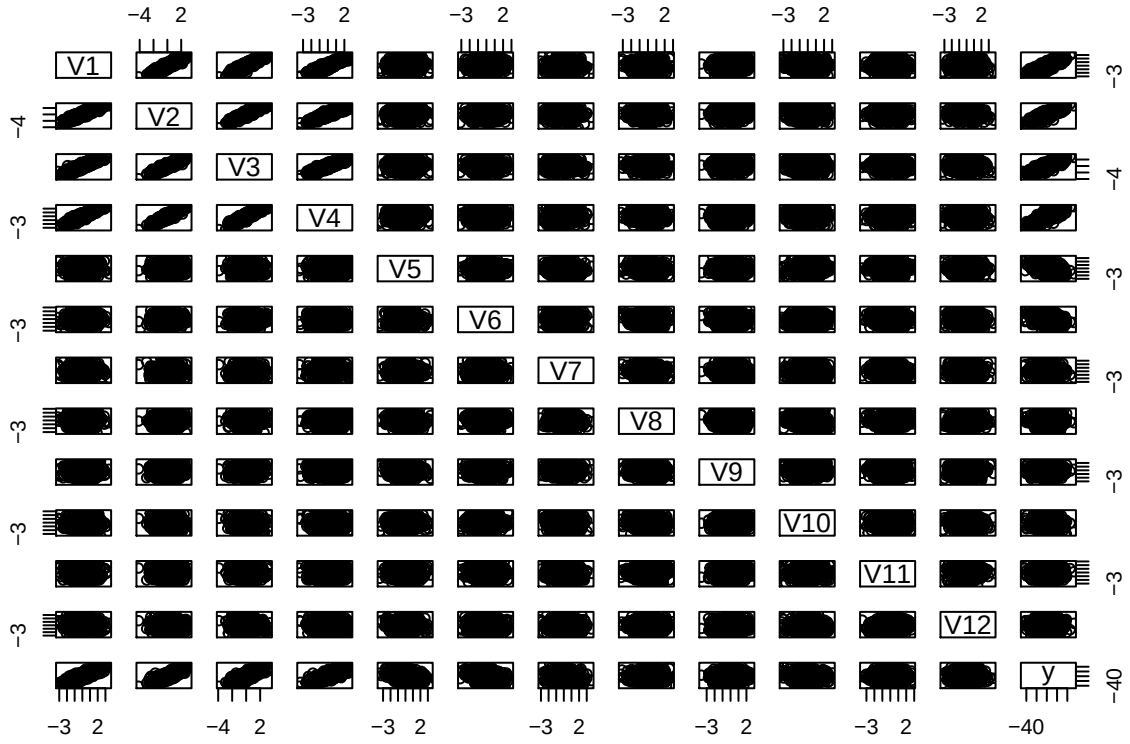
Simulations and Comparisons

2.1 Simulated Data

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 coefficents for $V_7, ..., V_{12}$ are all zero.

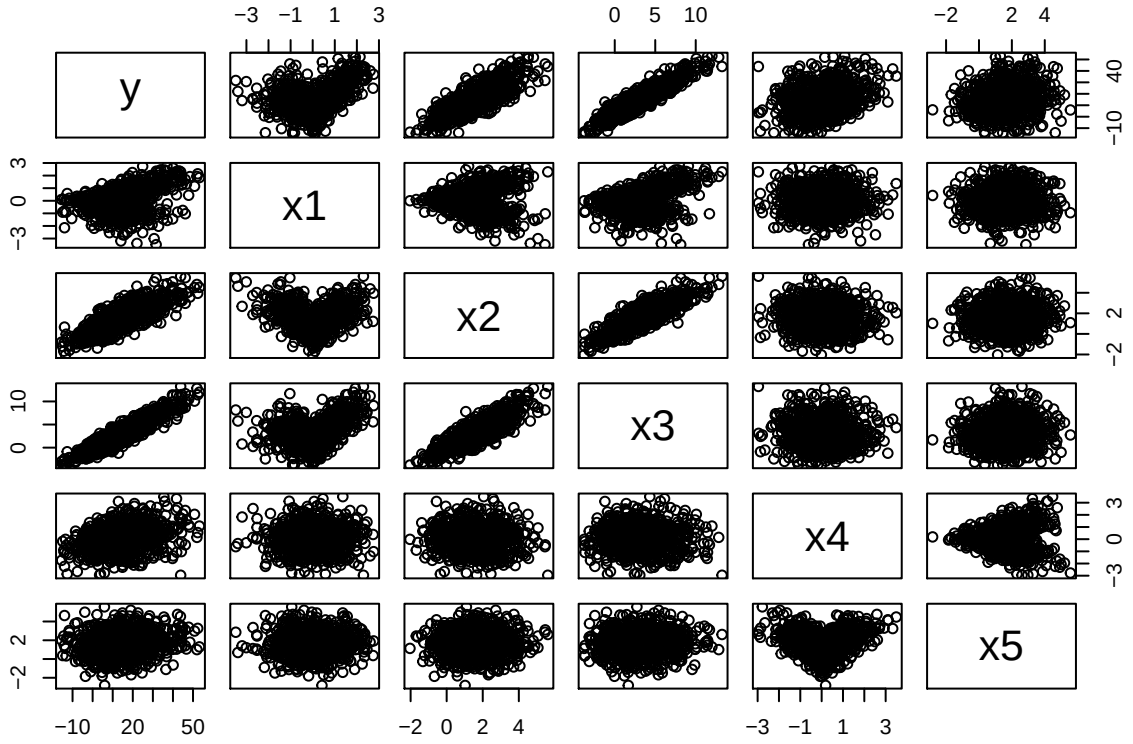


```
              V1            V2           V3           V4           V5
V1   1.000000000  0.9146541132  0.907895351  0.907066765 -0.034234292
V2   0.914654113  1.0000000000  0.913780977  0.913731929 -0.020066937
V3   0.907895351  0.9137809775  1.000000000  0.902766836 -0.016703104
V4   0.907066765  0.9137319288  0.902766836  1.000000000 -0.002169185
V5  -0.034234292 -0.0200669374 -0.016703104 -0.002169185  1.000000000
V6   0.006059950 -0.0007016153 -0.006844401 -0.014814462  0.043610533
V7   0.011864999 -0.0013307105  0.006951044  0.022724703  0.005306628
V8   0.012429245  0.0180018750  0.021232744  0.020871464 -0.031895007
V9   0.035378928  0.0383359616  0.036338439  0.037906433 -0.024882964
V10 -0.027164872 -0.0205322792 -0.021277153 -0.021746391  0.005239249
V11 -0.062839803 -0.0447873818 -0.029615955 -0.053826143 -0.018439527
V12  0.001895378 -0.0040846930 -0.005222882 -0.017304444 -0.043550921
y    0.828867870  0.8301277767  0.808338105  0.788791607 -0.388422174
              V6           V7          V8           V9          V10
V1   0.0060599505  0.011864999  0.01242924  0.035378928 -0.027164872
V2  -0.0007016153 -0.001330711  0.01800187  0.038335962 -0.020532279
V3  -0.0068444008  0.006951044  0.02123274  0.036338439 -0.021277153
V4  -0.0148144624  0.022724703  0.02087146  0.037906433 -0.021746391
V5   0.0436105331  0.005306628 -0.03189501 -0.024882964  0.005239249
V6   1.0000000000 -0.004856916 -0.02539841 -0.013632377  0.036331634
V7  -0.0048569164  1.000000000 -0.02367510  0.006567436  0.018156112
V8  -0.0253984113 -0.023675099  1.00000000 -0.038840769 -0.089374372
V9  -0.0136323768  0.006567436 -0.03884077  1.000000000  0.001732069
V10  0.0363316342  0.018156112 -0.08937437  0.001732069  1.000000000
V11 -0.0217988171 -0.033514884  0.02826398 -0.075004886  0.010676447
V12 -0.0422332641  0.032644574  0.02539601  0.009560373  0.015536556
y   -0.3637688830 -0.141169908  0.03665375  0.045997408 -0.037738613
             V11          V12          y
V1  -0.06283980  0.001895378  0.82886787
V2  -0.04478738 -0.004084693  0.83012778
V3  -0.02961595 -0.005222882  0.80833811
V4  -0.05382614 -0.017304444  0.78879161
V5  -0.01843953 -0.043550921 -0.38842217
V6  -0.02179882 -0.042233264 -0.36376888
V7  -0.03351488  0.032644574 -0.14116991
V8   0.02826398  0.025396015  0.03665375
V9  -0.07500489  0.009560373  0.04599741
V10  0.01067645  0.015536556 -0.03773861
V11  1.00000000  0.022521656 -0.02183787
V12  0.02252166  1.000000000  0.02427605
y   -0.02183787  0.024276053  1.00000000
```

Let's move on to a more difficult situation. The dataset $D2$ contains five predictors, $X_1, ...X_5$, that have an interesting structure- several of the predictors are

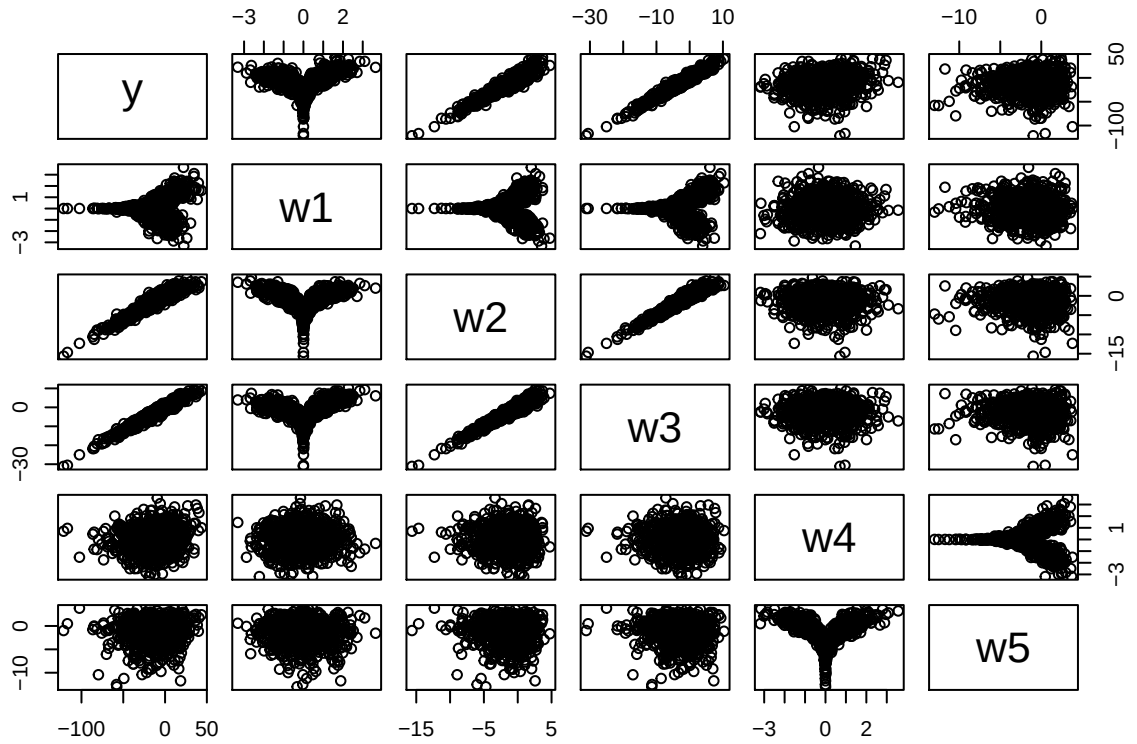
correlated, but are not one -to -one. This violates an important assumtion of the linear model and means that these variables have low correlation. Note that this only makes sense in higher dimensions where we are estimating the value of X_j given $X_1, ..., X_n$.

	y	x1	x2	x3	x4	x5
y	1.0000000	0.37370066	0.81852208	0.92926643	0.30615522	0.15357237
x1	0.3737007	1.00000000	0.01253512	0.37828457	0.02553116	0.01196494
x2	0.8185221	0.01253512	1.00000000	0.85944303	-0.04770287	0.04253338
x3	0.9292664	0.37828457	0.85944303	1.00000000	-0.03116991	0.04579883
x4	0.3061552	0.02553116	-0.04770287	-0.03116991	1.00000000	0.01984835
x5	0.1535724	0.01196494	0.04253338	0.04579883	0.01984835	1.00000000



The tricker relationship between the variables in $D2$ was because they were generated using the square root of the absolute value of the other variable. Here, in $D3$ the process is repeated but with the log of the absolute value.

	y	w1	w2	w3	w4	w5
y	1.00000000	0.15728407	0.94764704	0.97416878	0.17279746	0.05578763
w1	0.15728407	1.00000000	-0.02714818	0.15752508	0.03050113	-0.05109218
w2	0.94764704	-0.02714818	1.00000000	0.96531545	-0.01975103	-0.04414540
w3	0.97416878	0.15752508	0.96531545	1.00000000	-0.01042494	-0.05419898
w4	0.17279746	0.03050113	-0.01975103	-0.01042494	1.00000000	-0.03325170
w5	0.05578763	-0.05109218	-0.04414540	-0.05419898	-0.03325170	1.00000000



2.2 Models and Comparisons

2.3 Trees

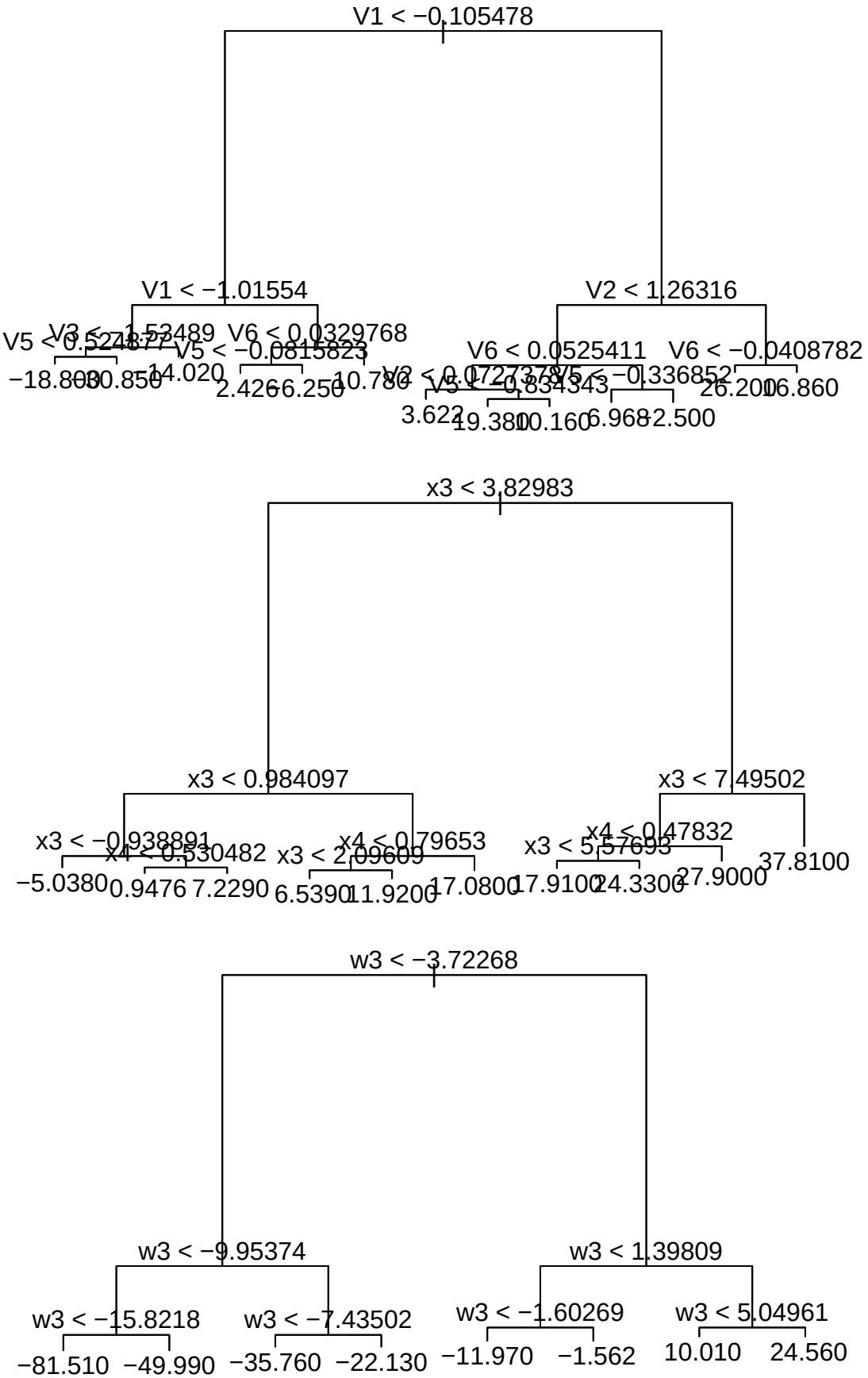
2.3.1 CART: Regression Trees

As outlined in the 1984 textbook, *Classification and Regression Trees*, _____ described their method for creating, pruning, and testing regression trees. What follows is their basic algorithm for fitting regression trees:

Algorithm 2: CART, Regression Trees

1.
2.
3.
4.

Let's return to our simulated datasets. Here is what trees grown for each dataset using the model $Y \sim X_1, ...X_n$.


V1 < −0.105478
V1 < −1.01554
V2 < 1.26316
V3 < 1.53489
V6 < 0.0329768
V6 < 0.0525411
V6 < −0.0408782
V5 < −0.0815823
V2 < 0.0727375
V5 < −0.336852
V5 < −0.834343
−18.800
−30.850
−14.020
2.426
−6.250
−10.780
3.622
19.380
10.160
6.968
−2.500
26.200
16.860
x3 < 3.82983
x3 < 0.984097
x3 < 7.49502
x3 < −0.938891
x4 < 0.530482
x3 < 2.09609
x4 < 0.79653
x4 < 0.47832
x3 < 5.57693
−5.0380
0.9476
7.2290
6.5390
11.9200
17.0800
17.9100
24.3300
27.9000
37.8100
w3 < −3.72268
w3 < −9.95374
w3 < 1.39809
w3 < −15.8218
w3 < −7.43502
w3 < −1.60269
w3 < 5.04961
−81.510
−49.990
−35.760
−22.130
−11.970
−1.562
10.010
24.560

2.3.2 Conditional Inference Trees

Another popular method for creating regression tree models is Conditional Inference Trees.

CI trees

```
1. For case weights  $w$  test the global null hypothesis of independence between any of

2. Choose a set  $A \subset X_{j}$  in order to split  $X_{j}$  into two disjoint sets

3. Recursively repeat steps 1 and 2 with modified case weights  $w_left$  and  $w_right$
```

from `https://eeecon.uibk.ac.at/~zeileis/papers/Hothorn+Hornik+Zeileis-2006.pdf`

After step 1 is completed, any goodness of fit method can be used to generate the split and choose the set A. Note that in this method the splitting is done separately from the variable selection.

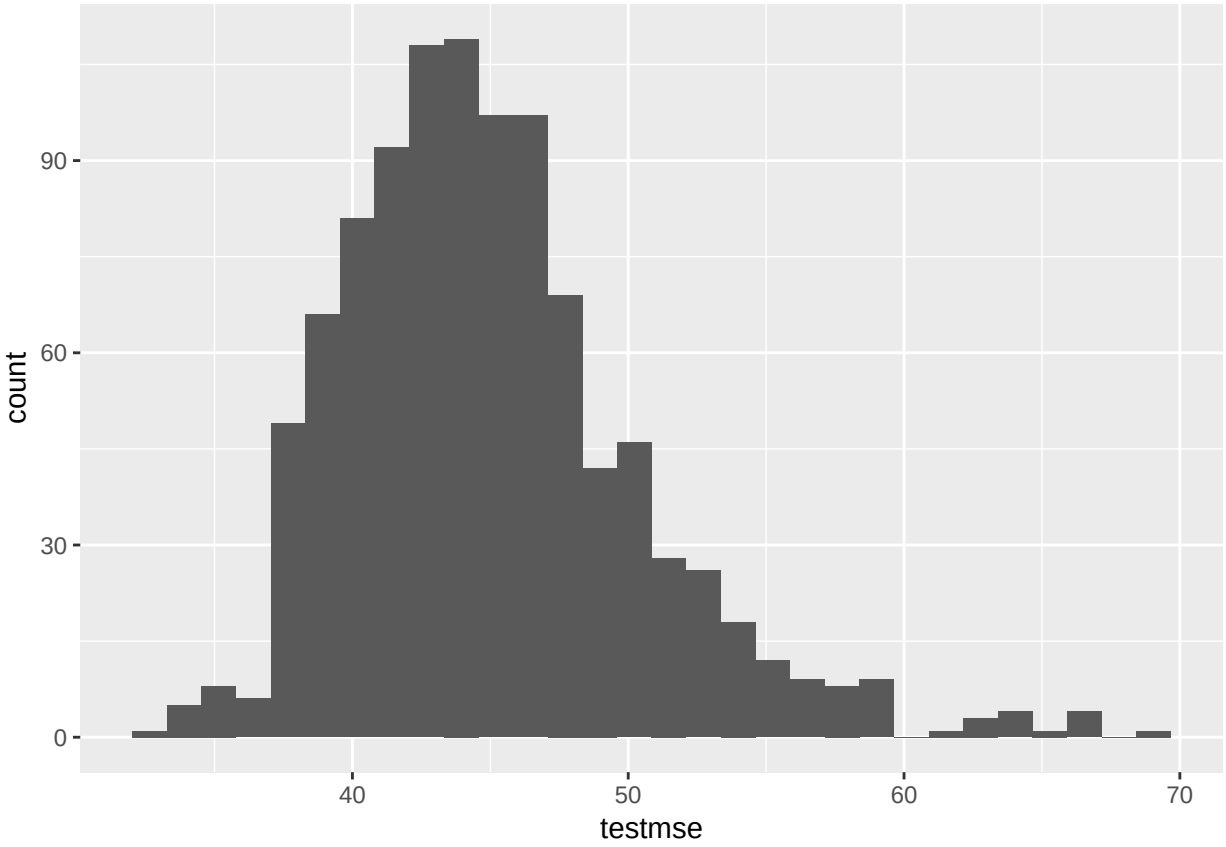
2.4 Bagged Forests

As mentioned in the introduction, single trees can have some variablility that in practice, limits their use. This simulation demonstrates this principle:

```
for 1000 trials:

1. separete D3 into a training set and a test set where the training set contains 2/3 of

2. fit a CART tree

3. Predict the response using the test set

4. Calculate the mean squared error
```

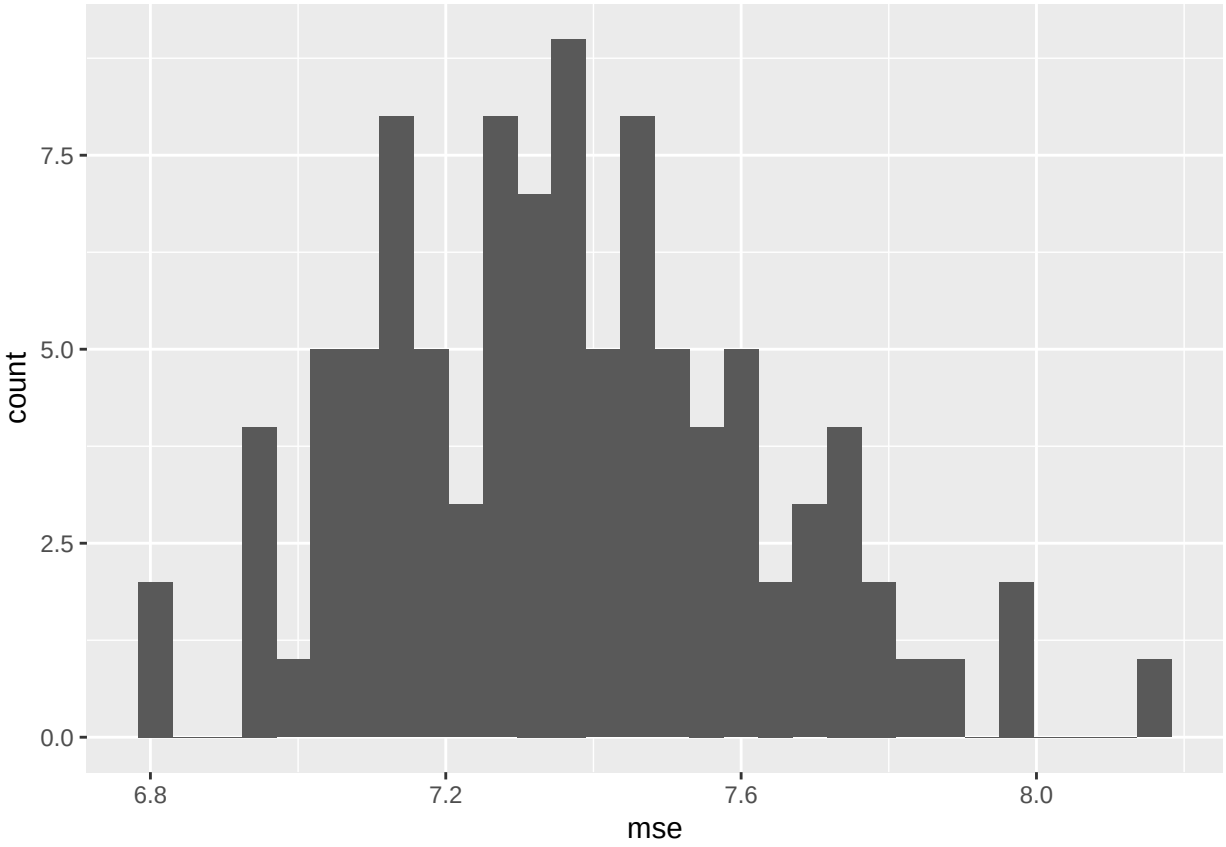
Here is a histogram of the MSE found using this method:

```
         testmse
testmse 28.56955
```



As one can see, 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 bootsrtapped samples are taken from the itintial 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.




```
            mse
mse 0.07051985
```

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.

2.5 Random Forests

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.

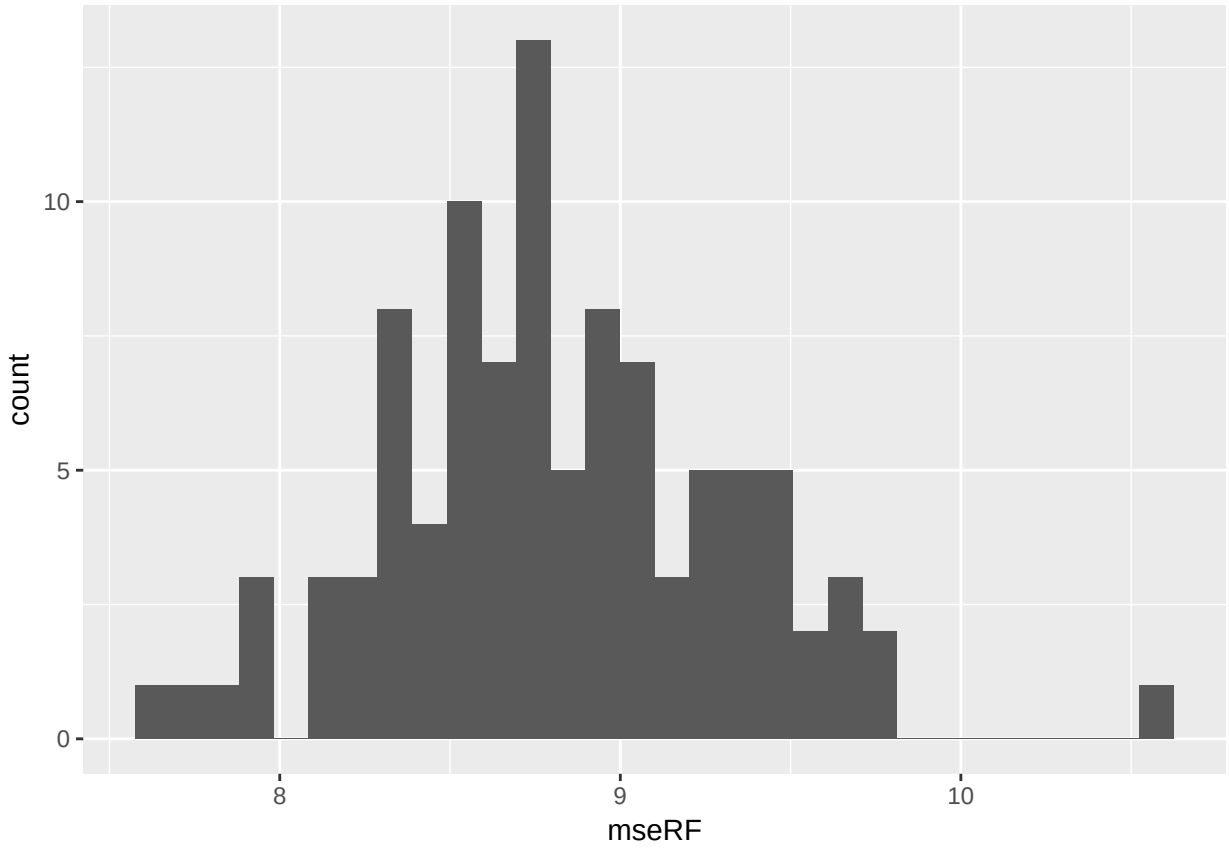
For an illustration, let's build a random forest on $D3$ and compare the MSE.

(include note about oob error vs test mse)

```
mseRF <- rep(0,100)
train = sample(1000,666)
for (i in 1:100){
```

```
  rf <- randomForest(y~., data = d3, subset = train, ntree = 100)
  mseRF[i] <- mean((d3$y[-train] - predict(rf, d3[-train,]))^2)
}

mseRF <- as.data.frame(mseRF)

ggplot(aes(x = mseRF), data = mseRF) + geom_histogram()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
var(mseRF)
```

```
          mseRF
mseRF 0.2528736
```

Chapter 3

Permuatations Tests Theory and Application to Conditional Variable Importance

Chapter 4

Implementation of Our Method

Conclusion

If we don't want Conclusion to have a chapter number next to it, we can add the `{.unnumbered}` attribute. This has an unintended consequence of the sections being labeled as 3.6 for example though instead of 4.1. The LaTeX commands immediately following the Conclusion declaration get things back on track.

More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

Appendix A

The First Appendix

This first appendix includes all of the R chunks of code that were hidden throughout the document (using the `include = FALSE` chunk tag) to help with readibility and/or setup.

In the main Rmd file:

```
# This chunk ensures that the reedtemplates package is
# installed and loaded. This reedtemplates package includes
# the template files for the thesis and also two functions
# used for labeling and referencing
if(!require(devtools))
  install.packages("devtools", repos = "http://cran.rstudio.com")
if(!require(reedtemplates)){
  library(devtools)
  devtools::install_github("ismayc/reedtemplates")
}
library(reedtemplates)
```

In :

```
# This chunk ensures that the reedtemplates package is
# installed and loaded. This reedtemplates package includes
# the template files for the thesis and also two functions
# used for labeling and referencing
if(!require(devtools))
  install.packages("devtools", repos = "http://cran.rstudio.com")
if(!require(dplyr))
    install.packages("dplyr", repos = "http://cran.rstudio.com")
if(!require(ggplot2))
    install.packages("ggplot2", repos = "http://cran.rstudio.com")
```

```
if(!require(reedtemplates)){
  library(devtools)
  devtools::install_github("ismayc/reedtemplates")
  }
library(reedtemplates)
#flights <- read.csv("data/flights.csv")
```

Appendix B

The Second Appendix, for Fun

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.