Simulations and Comparisons

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

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

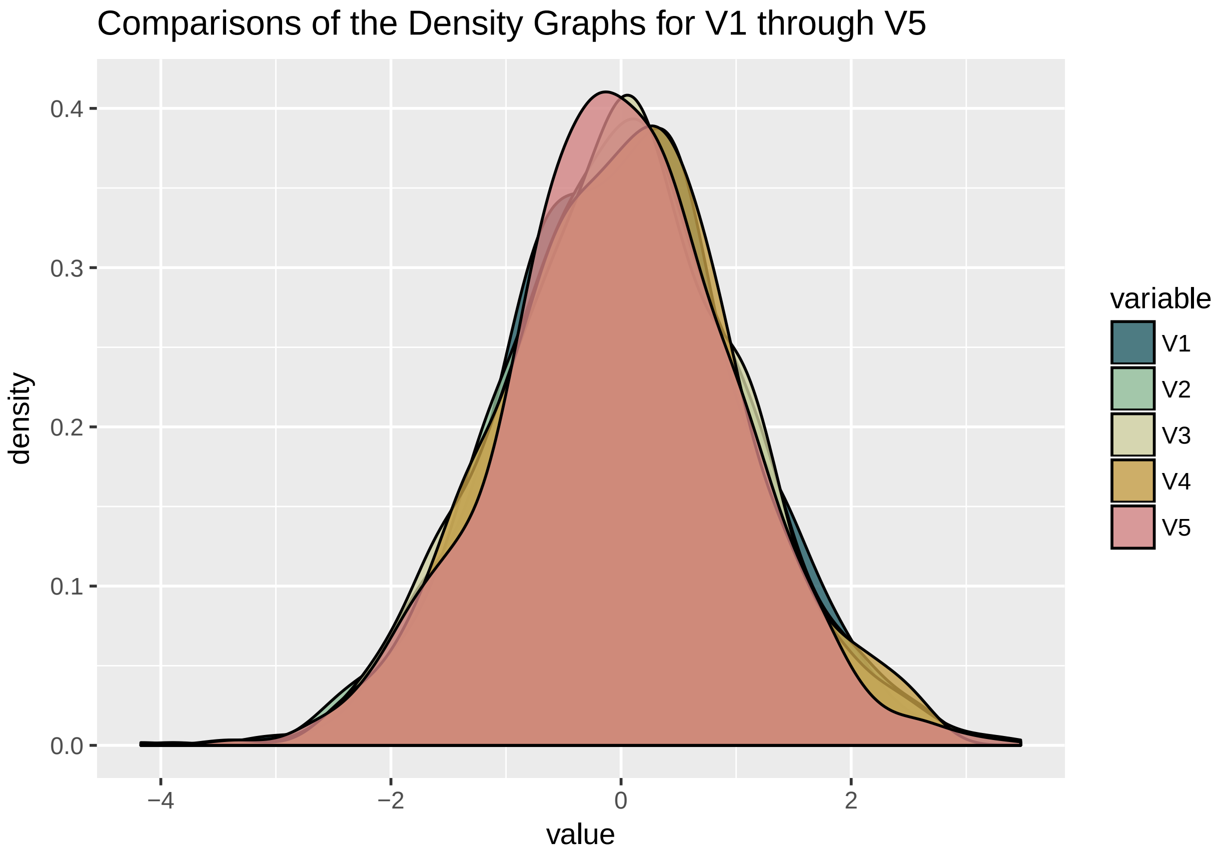
	V1	V2	V3	V4	V5	V6	V7	y	beta
V1	1.000	0.915	0.908	0.907	-0.034	0.006	0.012	0.829	5
V2	0.915	1.000	0.914	0.914	-0.020	-0.001	-0.001	0.830	5
V3	0.908	0.914	1.000	0.903	-0.017	-0.007	0.007	0.808	2
V4	0.907	0.914	0.903	1.000	-0.002	-0.015	0.023	0.789	0
V5	-0.034	-0.020	-0.017	-0.002	1.000	0.044	0.005	-0.388	-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 1:

[1] A brief note on uncertainty is needed here. It's true that in this setting we can say that V_4 is actually unimportant to understanding Y, but in situations with real data this is profoundly more difficult to parse. Often like in the social science situations that Morgan and Sonquist encountered, the real relationship between correlated predictors is complicated and often there is some theoretical backing or other insight that is gained to include variables that may not be important to the model.

[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 unsuccesful 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 unconsequential. I apologize.



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:

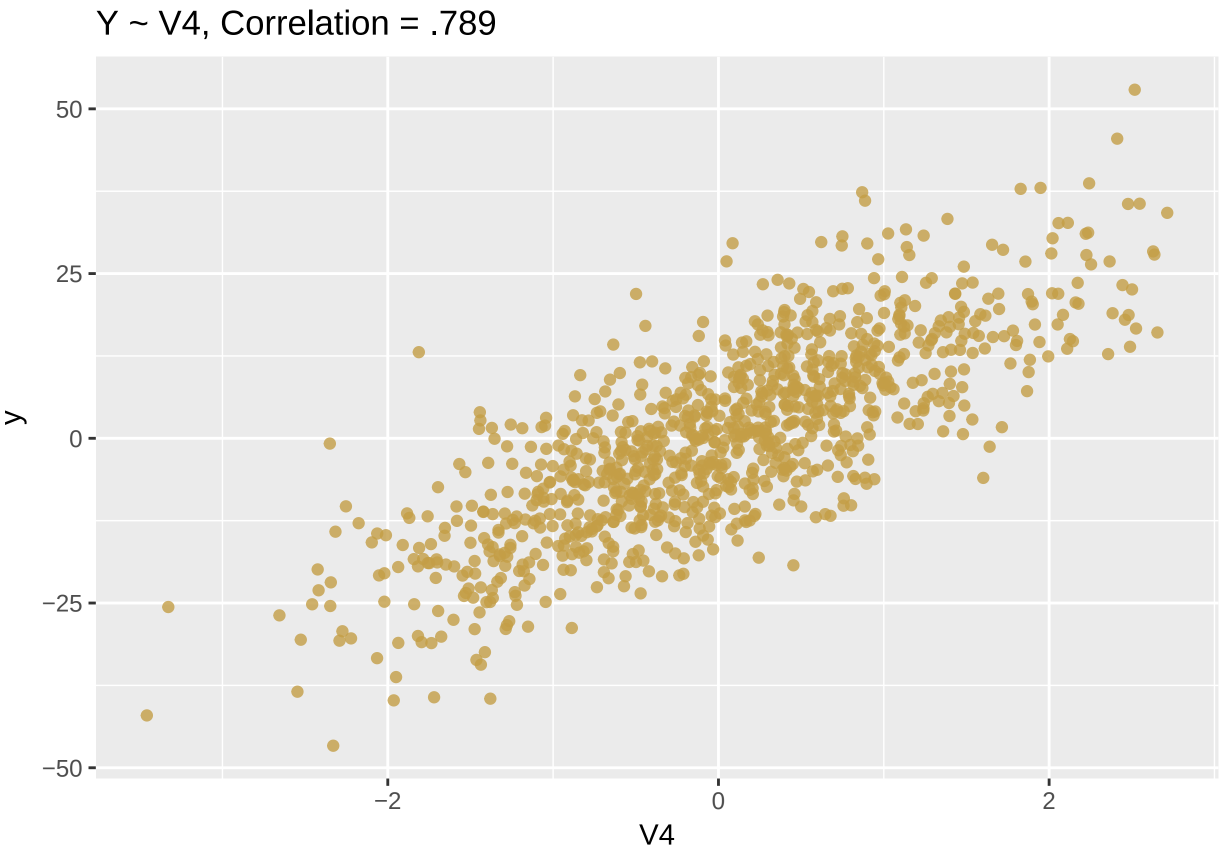


Figure 2 is an illustration of the relationship between $Y \sim V_4$ with linear correlation of .789.

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:

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

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 $\rho = .7$.

Figure 3:

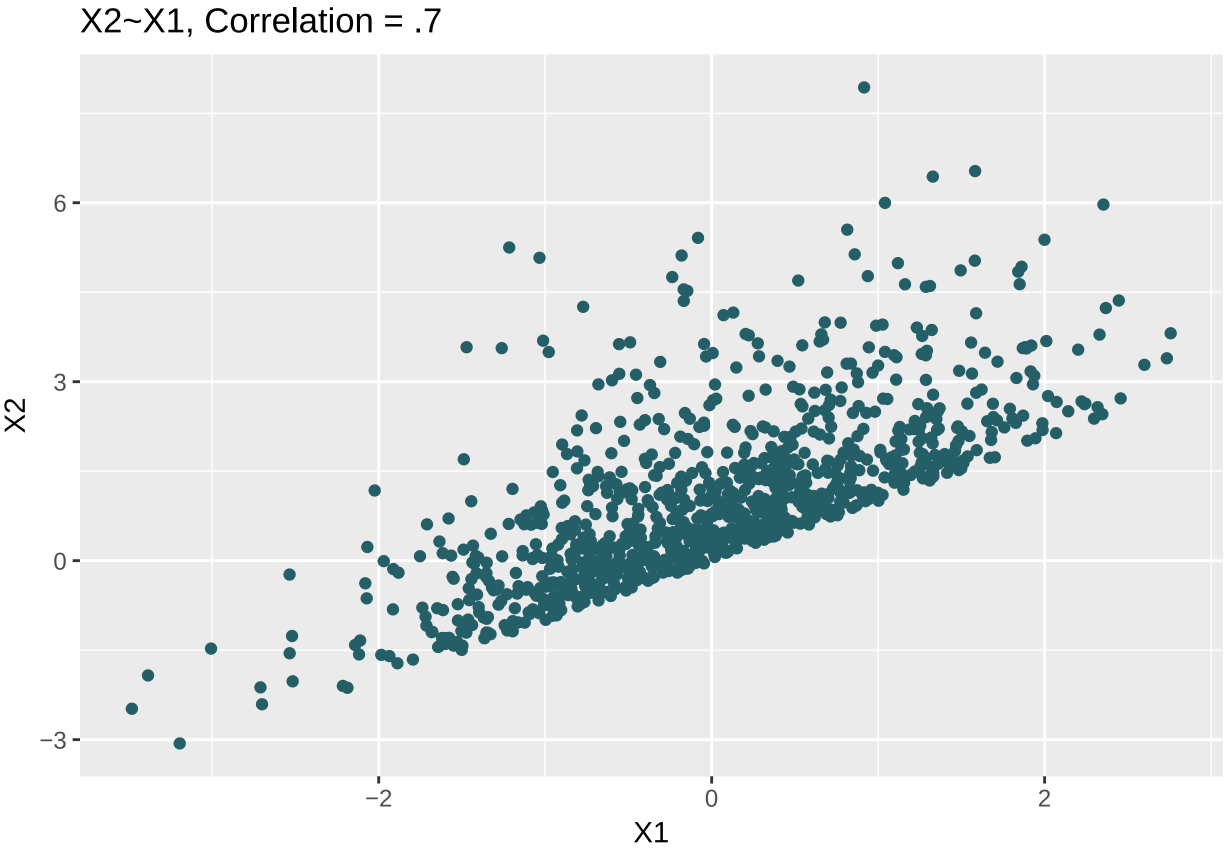
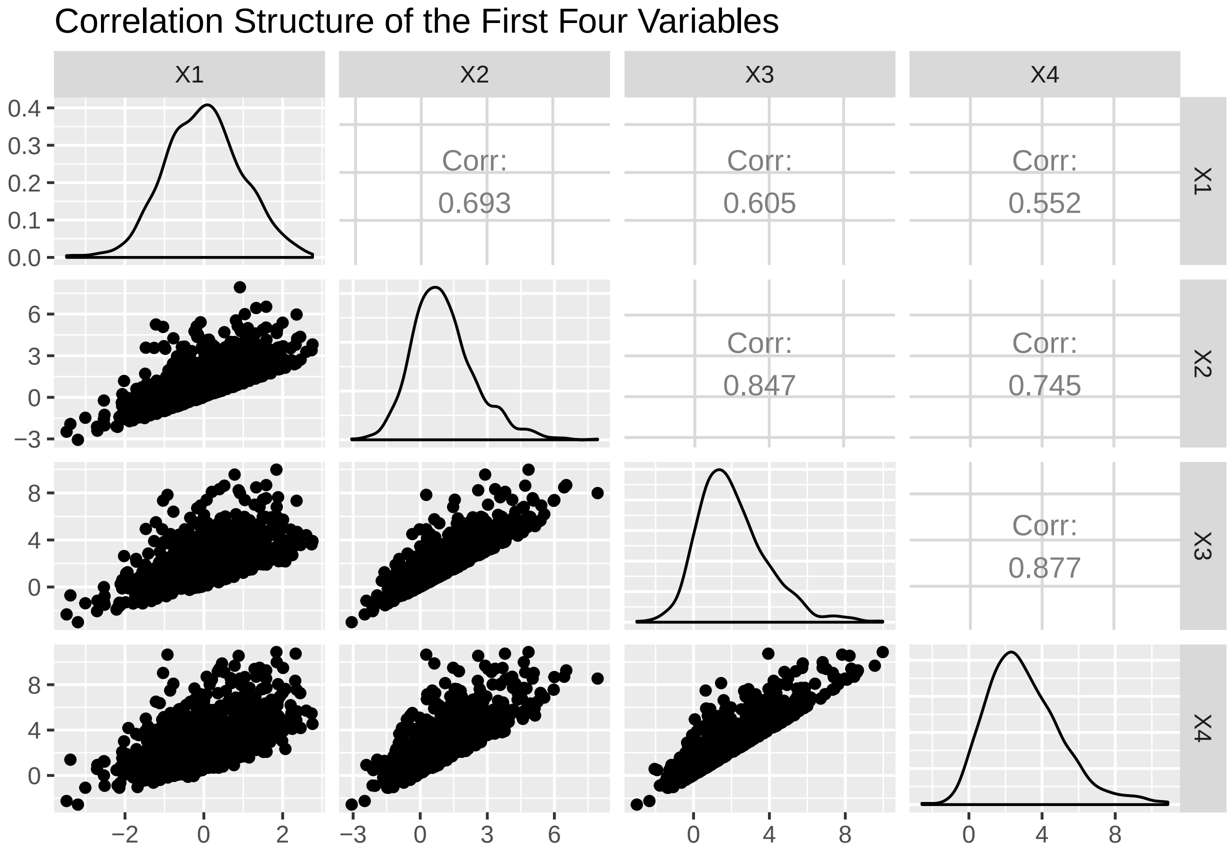


Figure 4:



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:

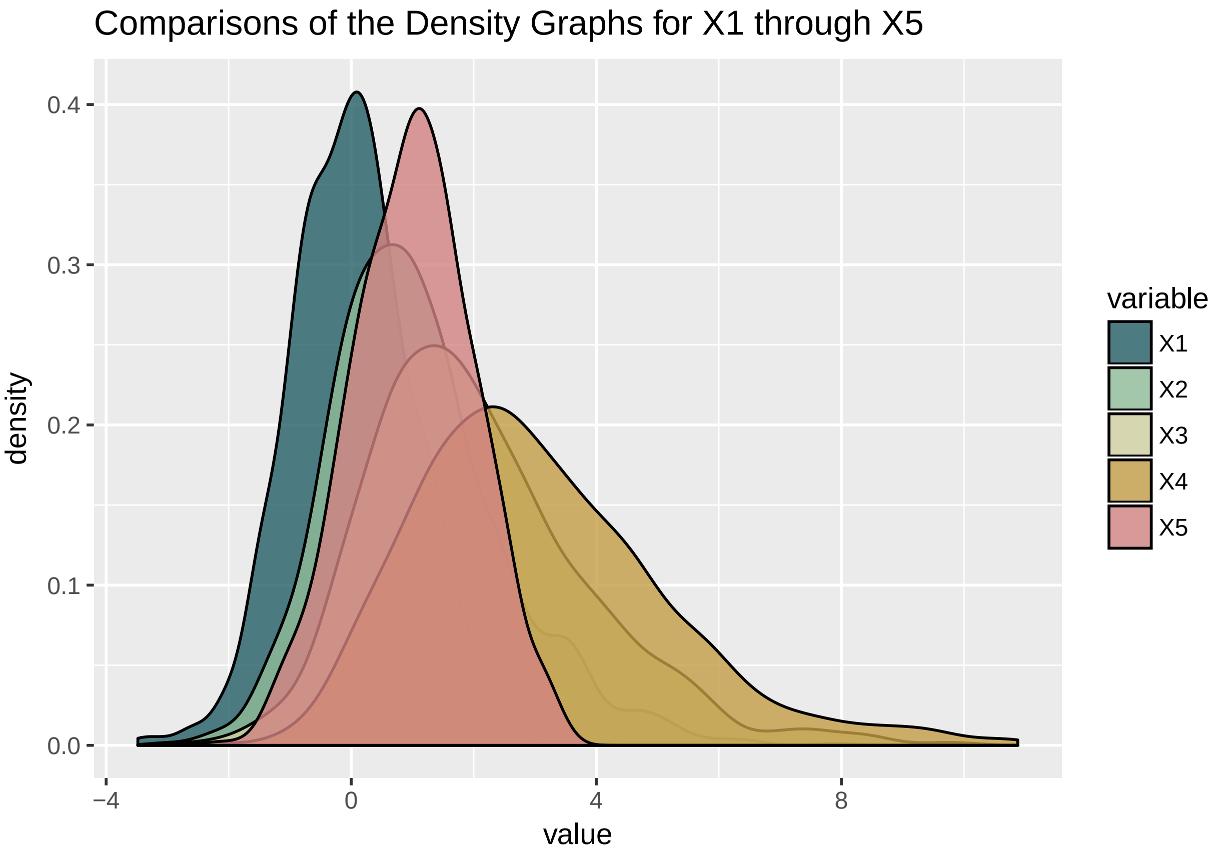


Figure 5 demonstrate how the correlation between a few of the predictors and Y may be effected by slope. Scale is much more a factor in this dataset, with some variables like X_3 having a larger range than the variables $X_1 \sim N(0,1)$ or $X_5, ..., X_{12} \sim MVN()$.

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:

$$\omega_1 \sim N(1,0)$$
$$\omega_2 = log(\omega_1) + E, E \sim N(1,0)$$
$$\omega_3 = log(\omega_2) + E, E \sim N(1,0)$$
$$\omega_4 = log(\omega_4) + E, E \sim N(1,0)$$

Table 3:Correlation of $\omega_1, ..., \omega_7$ and Y

	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

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 6:

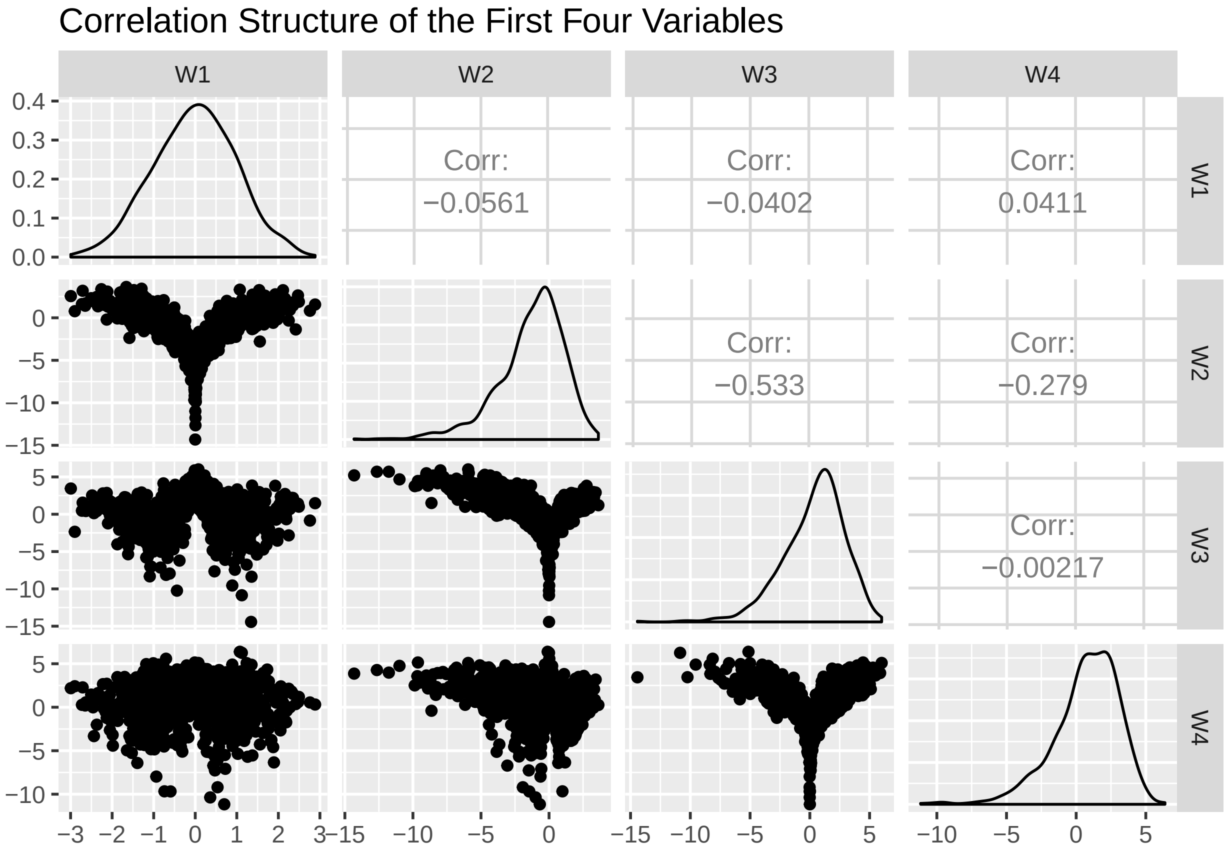
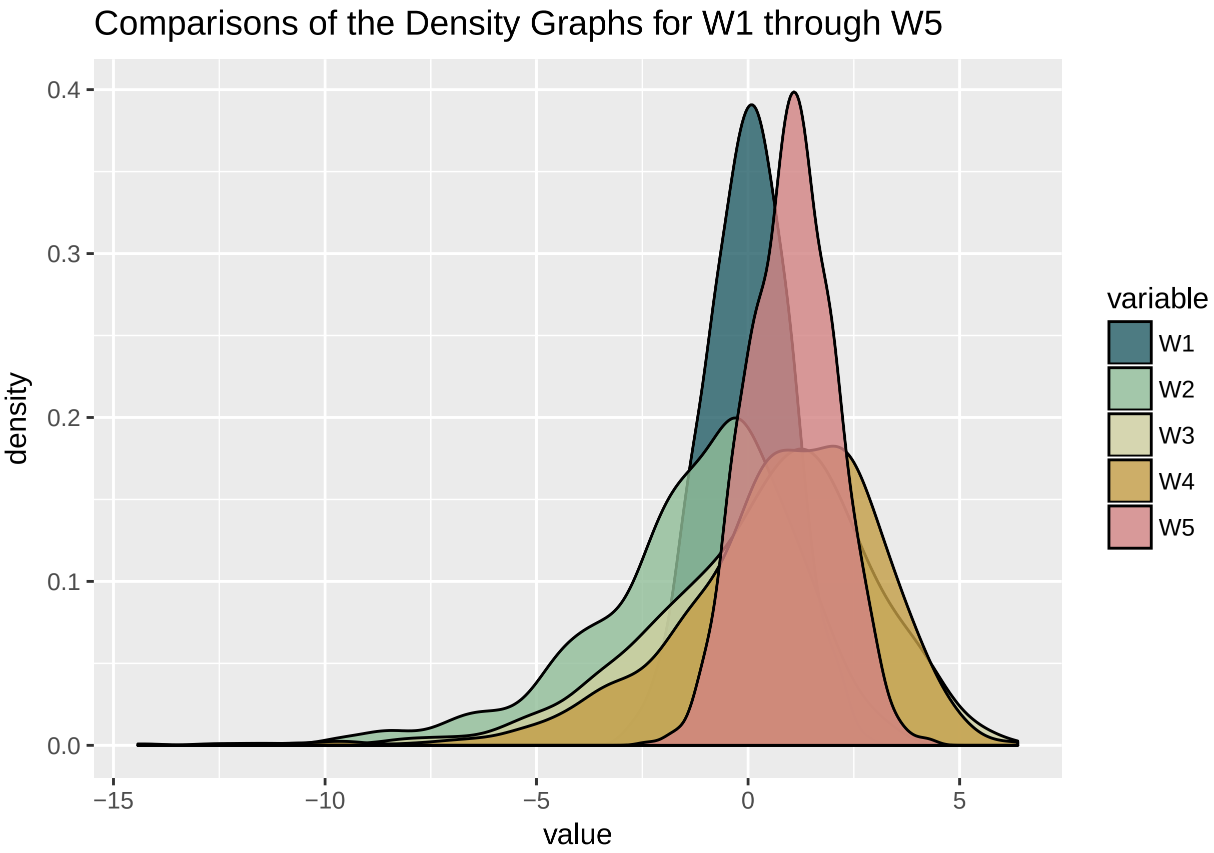


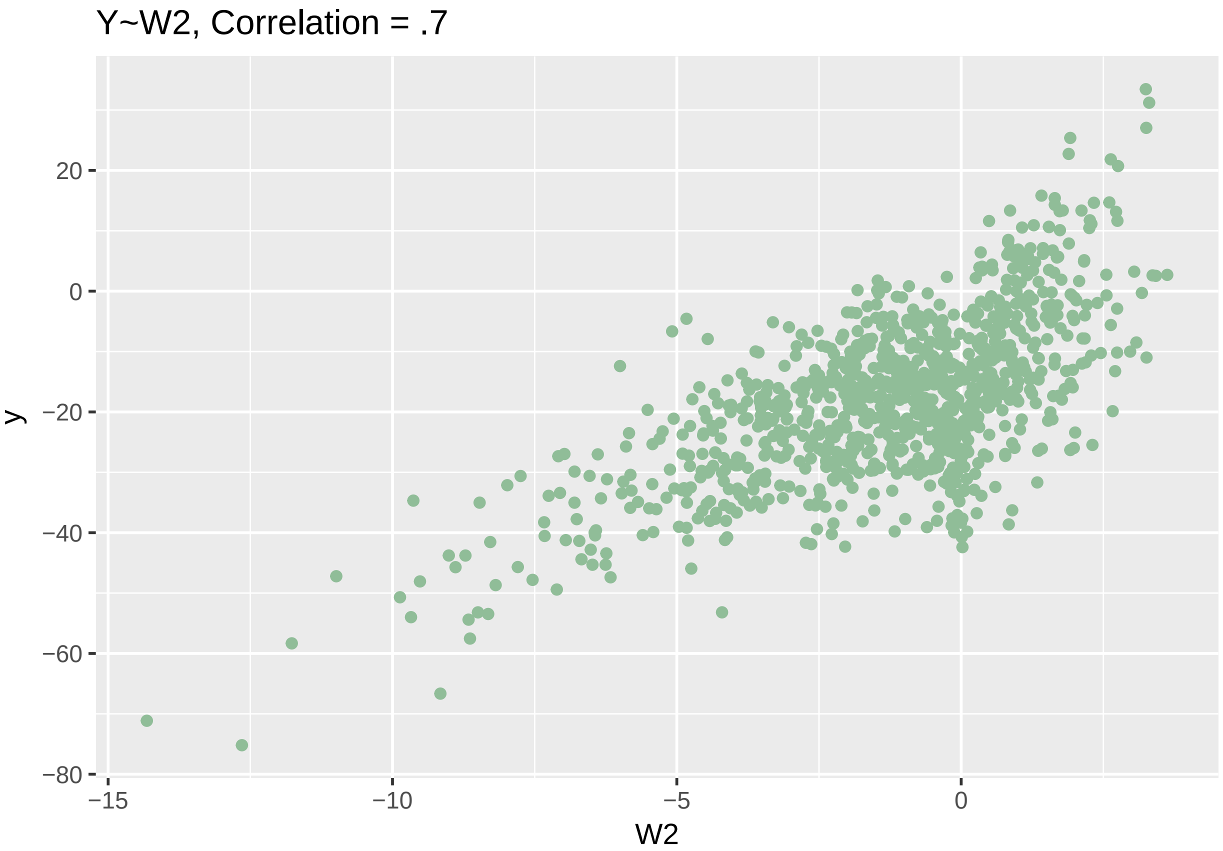
Figure 6 provides another way of visualizing some of the information given in Table 3. Here we can see the densities as well as the paired correlations of the first four variables in D_3.

Figure 7:



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.

Figure 8:



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

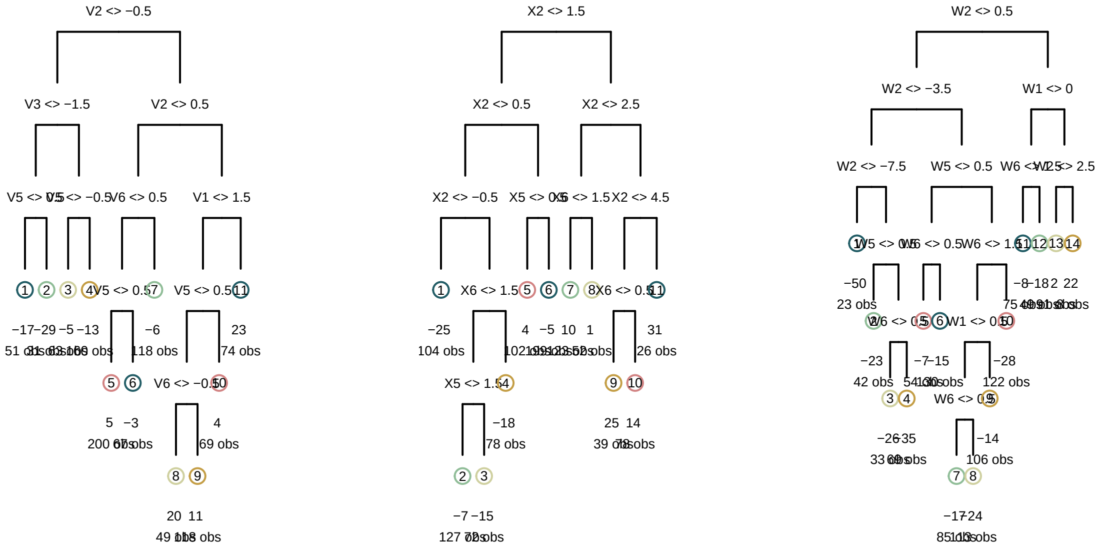
Models and Comparisons

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 over, two, partition that variable space in two distinct partitions, and three, set our intial 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 overfitting.

Figure 9:

CART Representing Y~, from datasets D1, D2, and D3



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 1 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
```

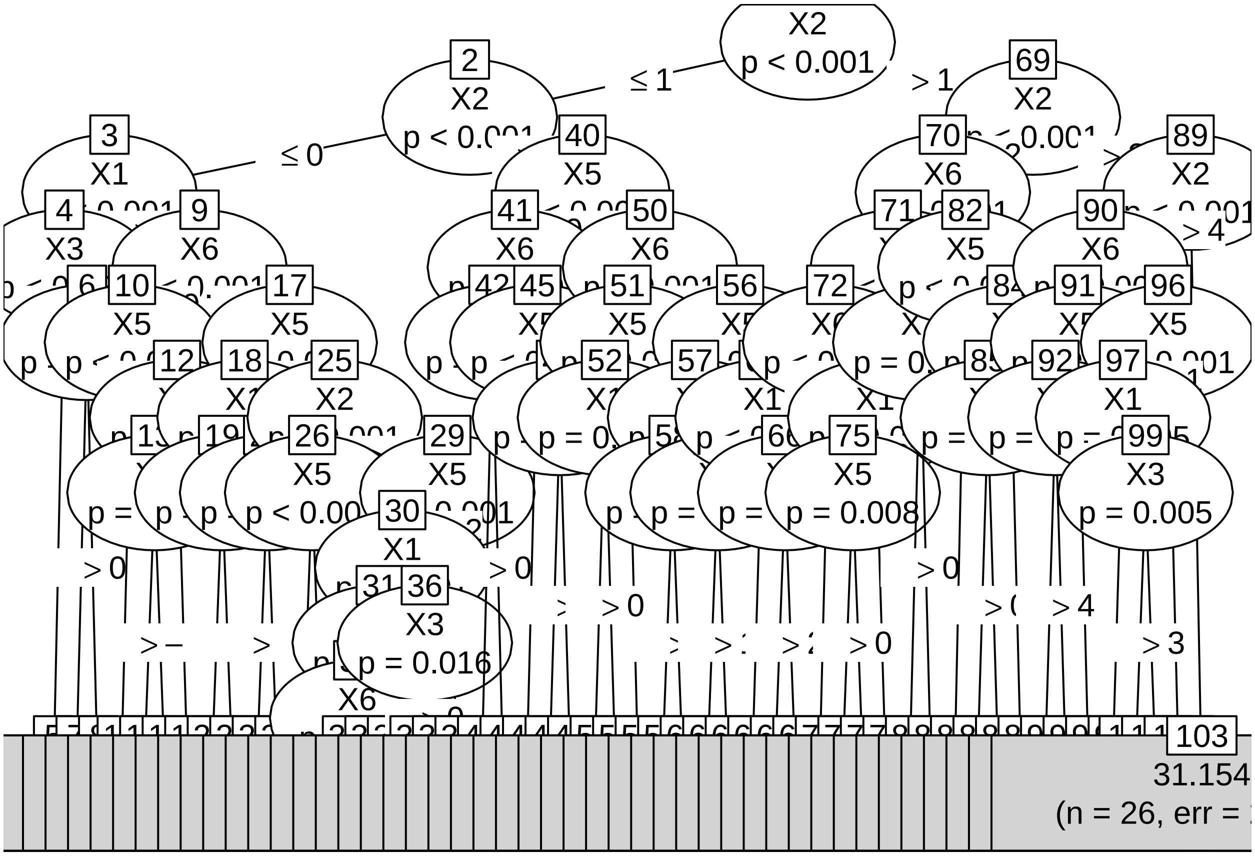
The case weights, $w_i \in w_1, ..., w_n$, correspond to nodes are defined as:

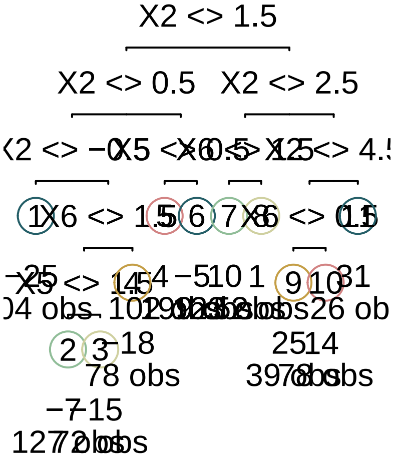
$$w_i = I(x_i \in N_t)$$

Where x_i is a vector of observations and N_t is a node in the tree. CTree's inferential power comes from their use of permuation tests, which will be expanded upon in Chapter 3. For now, lets compare the CART tree fitted on D_2 above with a conditional inference tree as implemented in the R package `partykit`.

Figure 10:

CART and CTree fitted to D_2





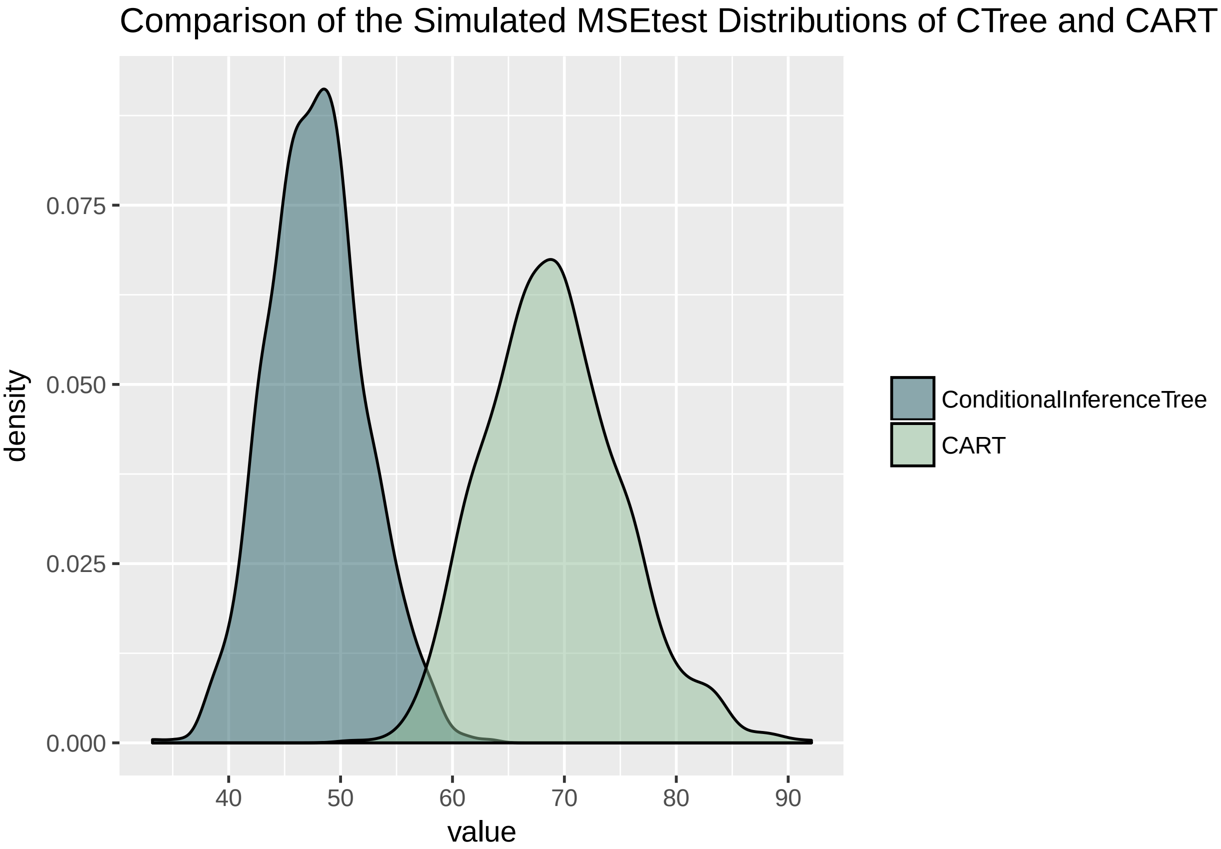
Trees can be quite variable, so to get a better idea of the differences between the methods let's run a simulation.

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.

Figure 11:



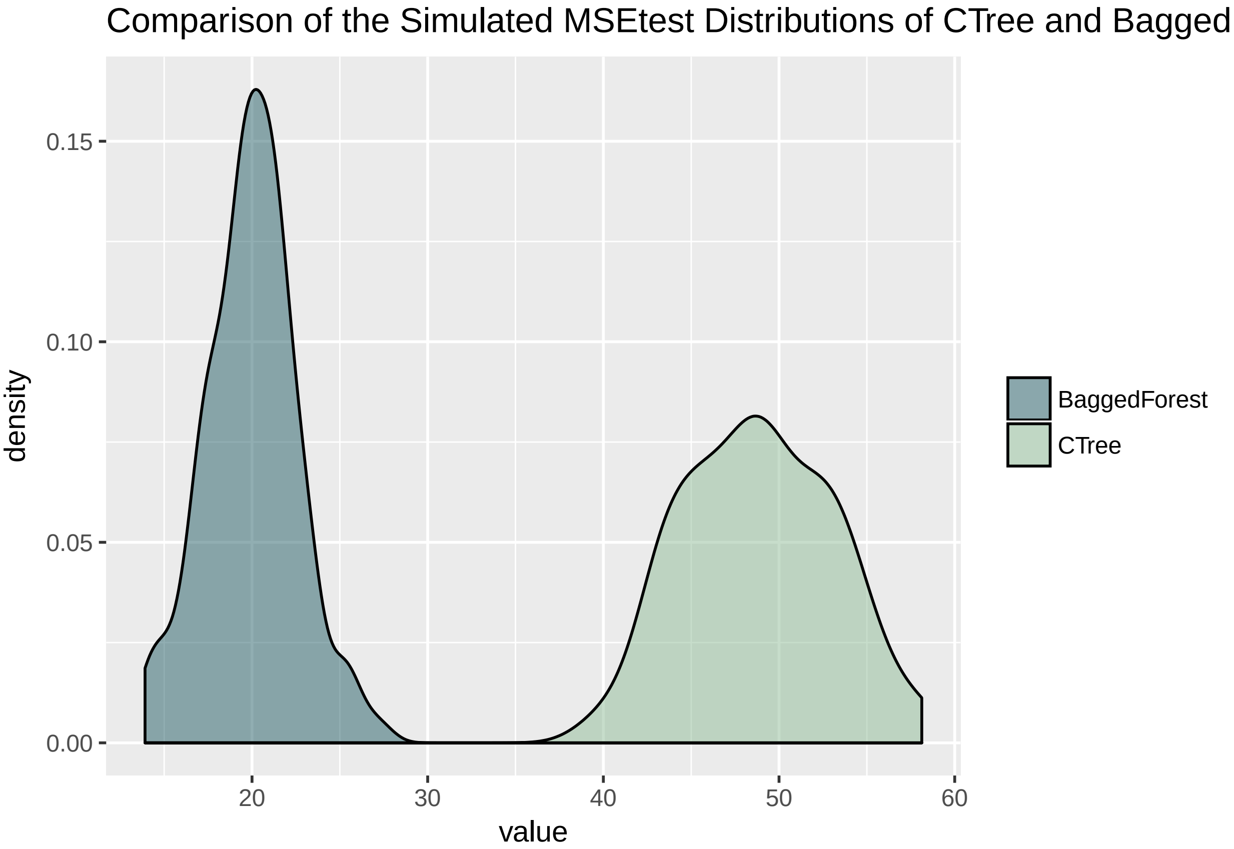
The CTree tree outperformed the CART one on MSE_{test}.

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 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.

Figure 12:



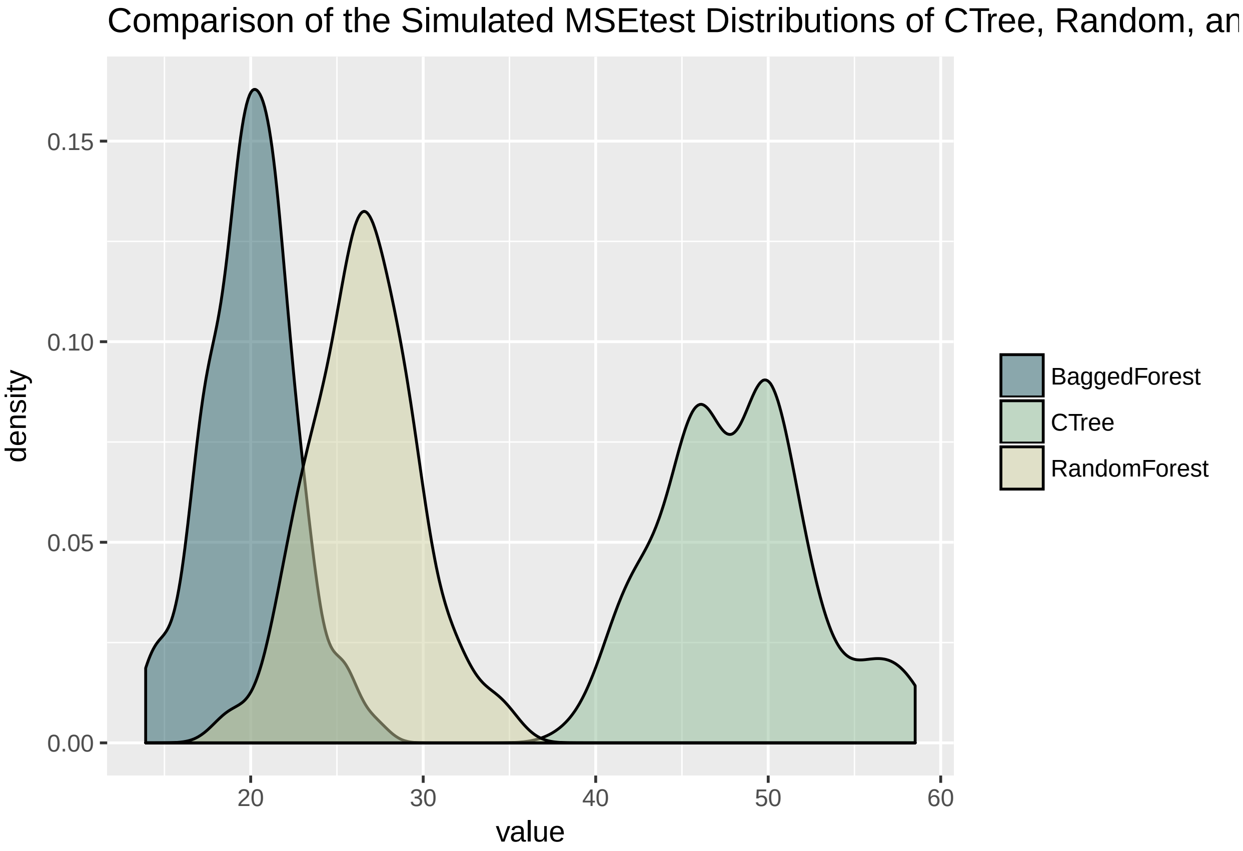
As one can see, the values of MSE_{test} for the bagged forest were entirely below the MSE_test for the trees and the variance was much smaller.

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.

Figure 13:



In this situation, the bagged forest outplayed the random forest. **Why?**