P3 Assess Learners Report

Vishu Gupta

vgupta359@gatech.edu

guptavishu55@gmail.com

Abstract:

The purpose of this project is to witness how different decision trees based on different algorithms perform when trying to predict data. Here we made use of 4 different algorithmic decision trees: DTLearner, RTLearner, BagLearner, and InsaneLeaner. For each of these experiments we have training data that allows us to train our model and then we have test data to see how well our model behaves in predicting labels based on our given features. We have also made use of in sample test data which we predict to be nearly perfect every time we make use of our model and out of sample test data which we predict to be less accurate then the in-sample data, but still perform relatively well in predicting our labels.

DTLearner is a decision tree that makes use of some special feature to use as the split feature. Here we made use of the highest correlation as that special feature

RTLearner is a decision tree that makes use of a random feature to use as the split feature.

BagLearner is an aggregate decision forest made up of multiple decision trees.

InsaneLearner is an even more aggregate decision forest as it calls on bag learner a total of 20 times.

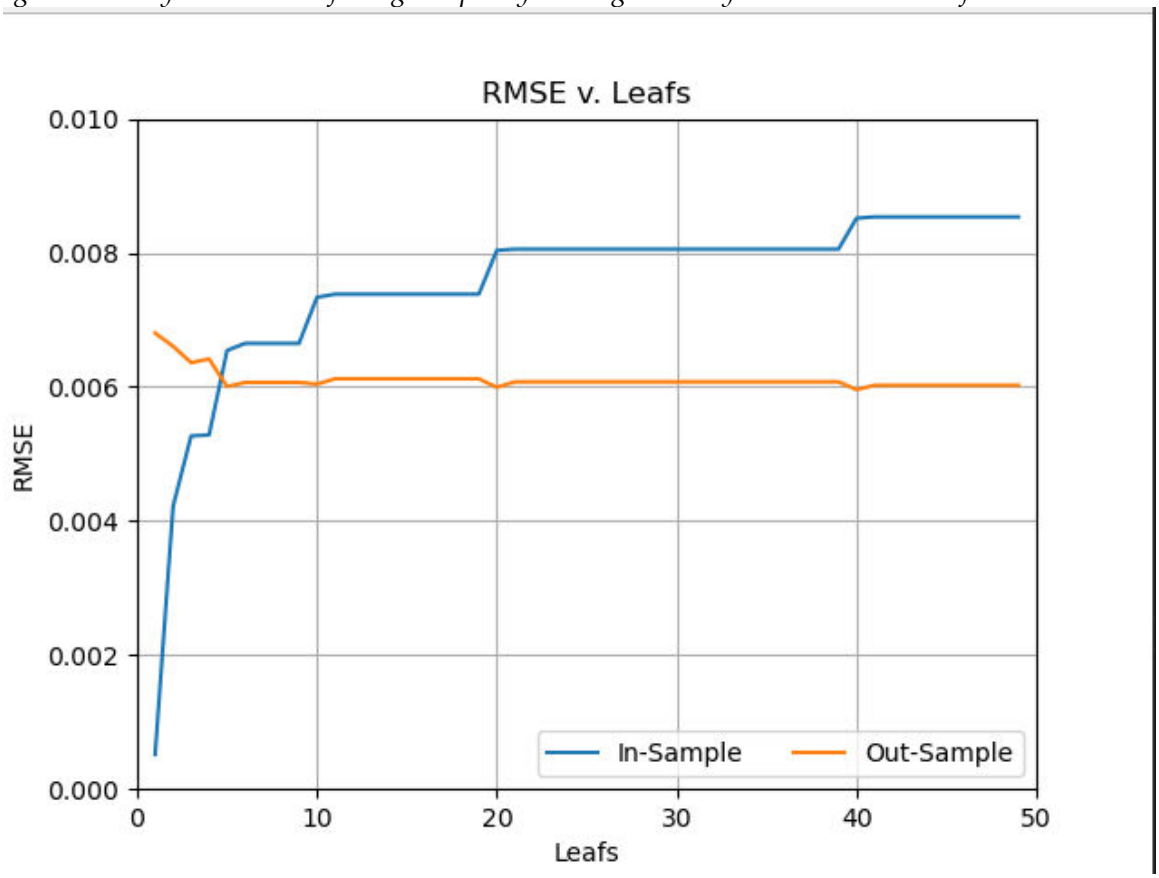
Introduction:

With the data we collected through running several experiments on multiple decisions trees we can conclude that decision trees perform fairly well in predicting data if we already have a good amount of training data to utilize. Although the cost of building decision trees is fairly high querying decision trees based on a test sample is not that bad. In this assignment it was our goal to calculate the RMSE and the CORR for 4 separate decision trees/forests: RTLearner, DTLearner, BagLearner, and InsaneLearner. In order to train these models, we have some data. We split the data into training data and test data. We then train our trees with the training data and see how well they perform with sample testing, which is testing with the training data, and out of sample testing, which is testing with unused data. The hypothesis here is that the in-sample testing is going to give us pretty accurate results while the out of sample testing may gave us less accurate results. Another hypothesis is that the trees will be most accurate in the following order from most accurate to lowest: InsaneLearner, BagLearner, DTLearner, RTLearner.

Experiment 1:

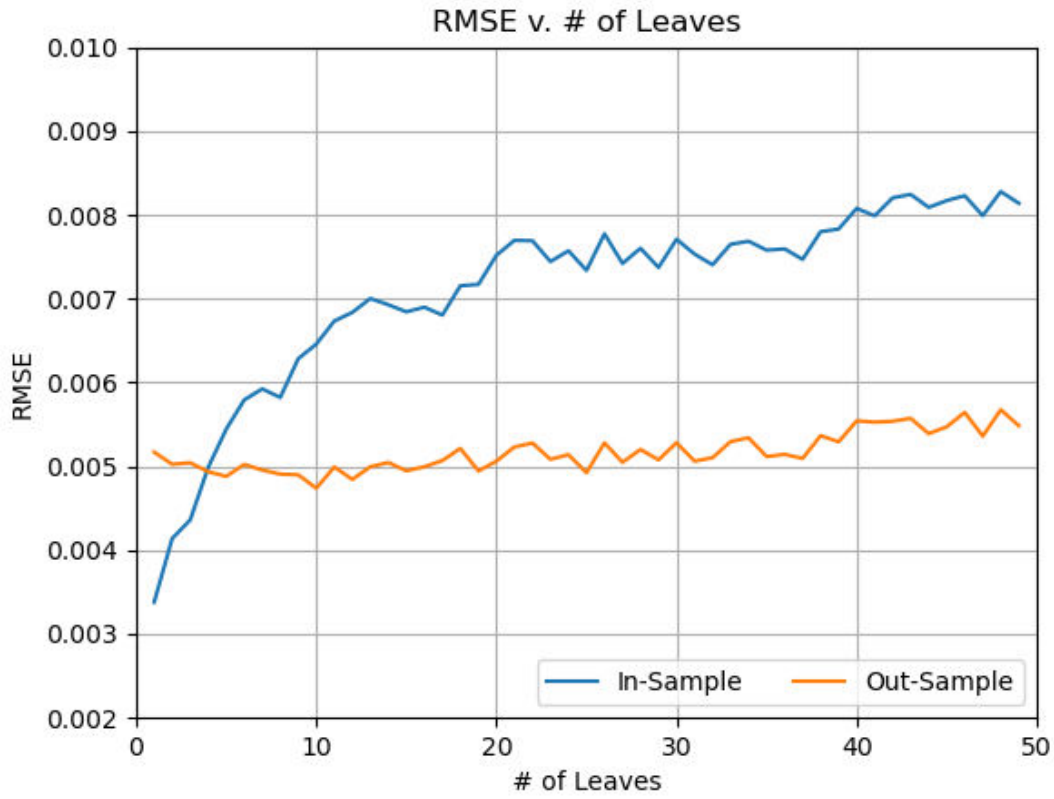
Overfitting does occur for smaller leaf sizes. As seen in the graph below the out of sample RMSE's are high for smaller leaves and become lower when more leaves are present. Clearly there seems to be

overfitting from leaves 1 to 5. Overfitting happens when the model is too close to the training data and doesn't work properly for data other than what was used for it to be trained. When the two lines intersect and diverge we can say that the overfitting has pretty much gone away. This can be seen after around 5 leaves.



Experiment 2:

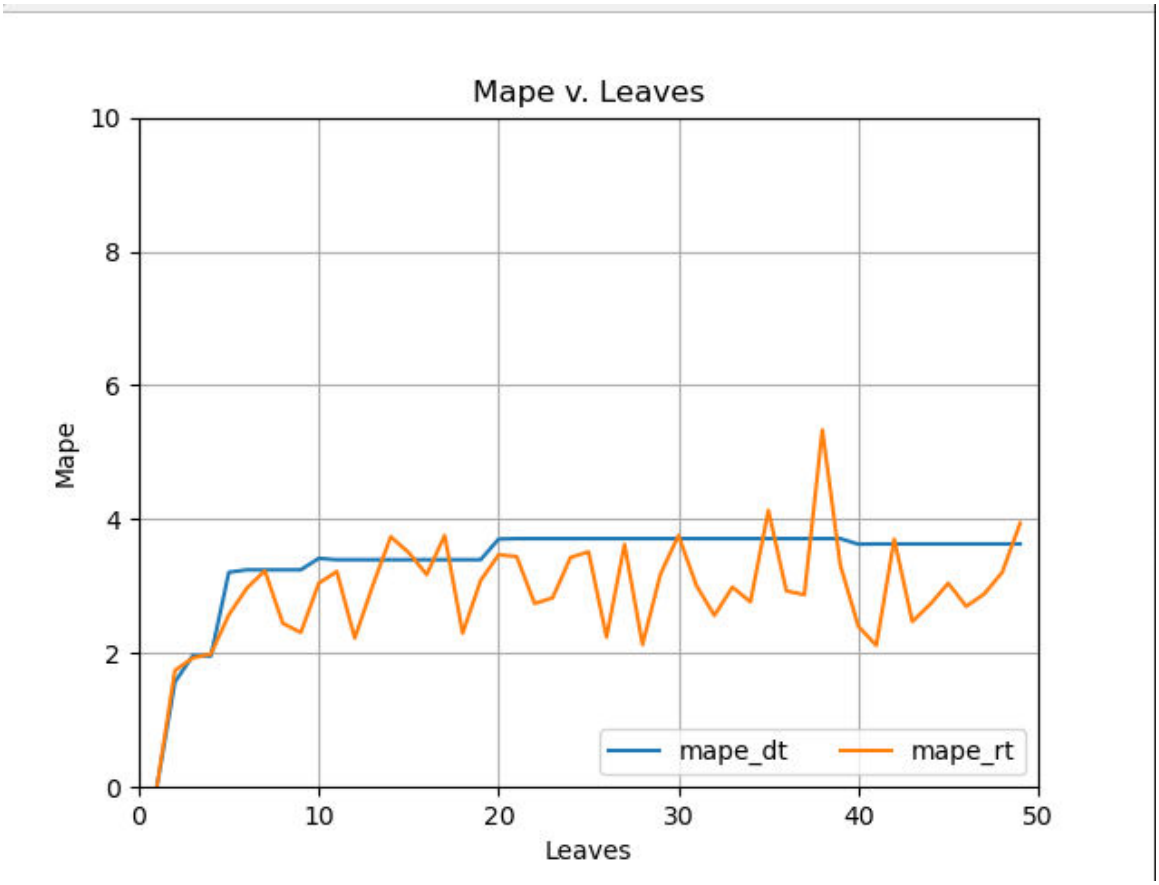
As seen from the plot below, using bagging decreases the potential problem of overfitting but does not eradicate it completely. When contrasted with the previous graph the difference between the in-sample and out-sample data has lowered but it still exists. As the number of leaves increases overfitting lessens. At around 3 leaves when the two lines intersect is where we can say that the overfitting



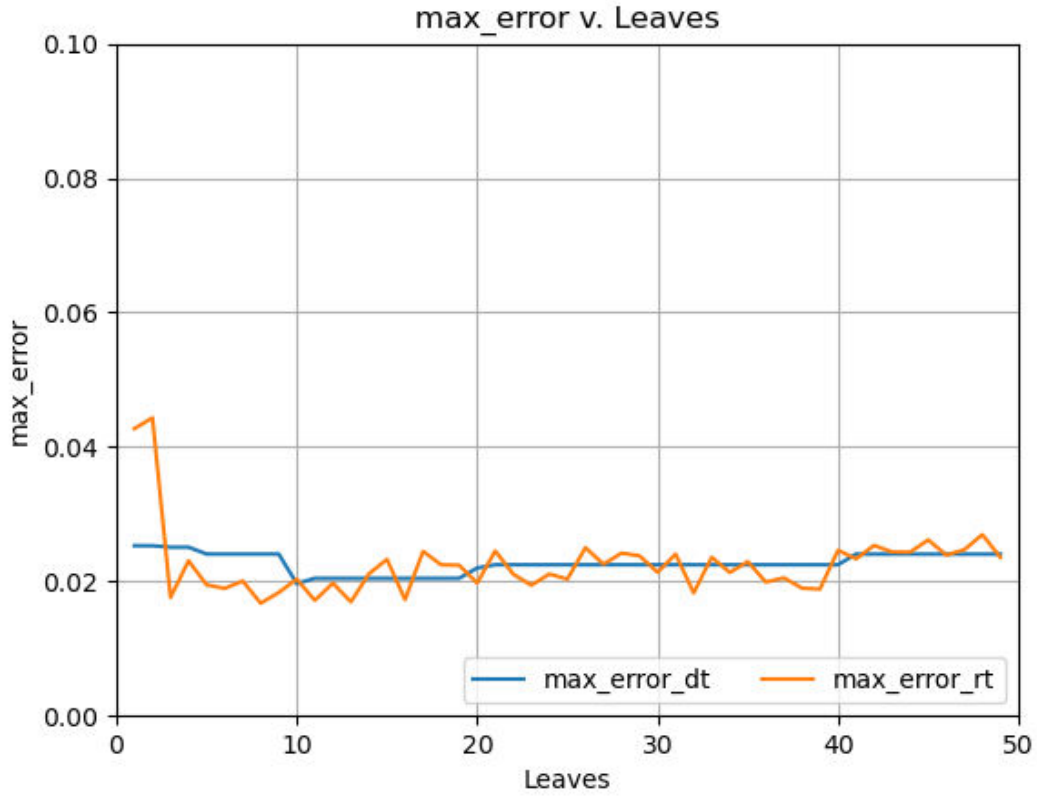
stops.

Experiment 3:

As shown in the figure below we can see that the DTlearner is more stable and has a more consistent MAPE than the RTLearner counterpart. While the RTLearner counterpart has a MAPE that seems to be fluctuating and goes as high as 5, the DTLearner has a MAPE that is more stable and stays below 4. MAPE stands for mean absolute percentage error and it gives us the error as a percentage. For both the RTLearner and the DTLearner, it seems like the error percentage is relatively low (below 7%).



As seen from the figure below the max error has similar characteristics to the MAPE. In this experiment as well the RTLearner has a more unsteady max error. It jumps high to low as the leaf count increases. The DTLearner, however, is relatively more stable and gives consistent results. One thing to note is that for both types of decision trees as the number of leaves increases the max error doesn't tend to change by much. It is also evident by the plot below that the max error was significantly higher for RTLearner when the number of leaves was less.



The MAPE was a better metric than the max_error. The MAPE captures the overall error percentage instead of just the maximum and it also depicts how the error percentage for the RTLearner was unstable while the error percentage for DTLearner was a lot more stable. The DTlearner had better performance according to MAPE, but for max error both learners behaved similarly. It is hard to conclude that the DTLearner will always be better. Although it seems that way from the plots above it is not always a guarantee.

Summary:

Ultimately, the purpose of this project was to get familiarity with decision trees and the performance of RTLearner and DTLearner. We also went one step ahead in this project where we used BagLearner and InsaneLearner to aggregate our models even more and make them more accurate as a forest rather than just a single tree. It was interesting to compare all the different models and see the difference in performance and accuracy.