CS 7446 Project 3: Assess Learners

Nick DiNapoli
ndinapoli6@gatech.edu

Abstract—In this work I implement and compare four supervised learning algorithms, three of which involve the use of regression trees at their core and one that employs linear regression. Additionally, I explore the usage of bootstrap aggregation as well as varying hyperparameters and inspecting the performance of the regression learners.

1 INTRODUCTION

I aim to show the benefits of implementing bag learners and ensemble learners compared to simple “normal" decision trees. By benefit I mean the reduction of overfitting and the ability to allow the model to generalize well to unseen data. I will compare ensemble methods to single decision tress and well and ensemble methods to one another. My initial hypothesis is that randomizing the decision trees in both feature selection and data selection (essentially hand-constructing a random forest) will lead to an enhanced model compared to single decision tree and a bag model of normal trees.

2 METHODS

In this work, I use the Istanbul dataset which includes the returns of many indexes worldwide from 1/5/09 to 2/22/11 in hopes to predict the daily returns of the MSCI Emerging markets index. This acts as the label for training my models. I build decision trees and forest according to two algorithms, JR Quinlan’s and A Cutler’s. The former utilizes decision tree partitioning on the feature of the data that has the highest absolute correlation to the data coming into an arbitrary node. The value that is used for splitting into sub-trees is the median of the data at the parent node such that the tree aims to remain balanced. In contrast, the latter algorithm is the backbone of a random forest in the sense that the feature to be split on is completely random and the split value at that node is the mean of two random data points at said parent node.

3 DISCUSSION

3.1 Experiment 1

I first use the Istanbul dataset to train and construct a decision tree. I use the first 60% of the data to build and train the single tree such that I do not peek into the future as the data is chronological. I utilize the remaining 40% of the dataset for examination of how well the decision tree can generalize to unseen data. By varying hyperparameters like the leaf size (maximum number of samples that can reside at a leaf node), visualizing the in-sample and out-of-sample error can be very telling when determining if overfitting has occurred. Figure 1 shows the RMSE as a function of leaf size for both the in and out-of-sample data points. As a rule of thumb in machine learning, the number of samples aggregated at a leaf should not exceed 5% of the total number of samples so I choose to vary the leaf size hyperparameter from 1 to 27 as there are 536 points in the dataset.

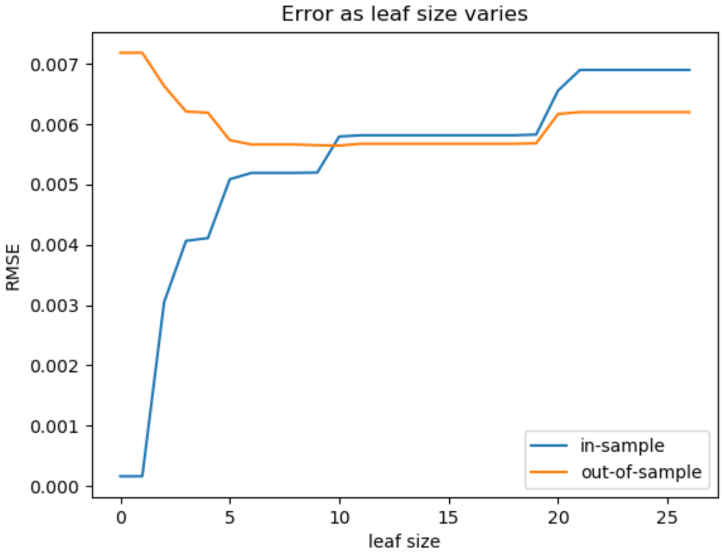

Figure 1—In-sample and out-of-sample error for a single decision tree as leaf size is varied.

Intuitively, as the leaf size approaches 1, the tree will be trained fitting the training data exactly and this will lead to overfitting and the lack of ability to generalize. In a similar light, as the leaf size parameter approaches the number of samples in the set, the model will have little predictive power so I look for a point where the RMSE is minimized. Qualitatively, it can be seen from Figure 1 that the out-of-sample error decreases and starts to increase once again. Quantitatively, the leaf size that minimizes the out-of-sample RMSE is 10 which is also the leaf size in which the in-sample error starts to overtake the out-of-sample RMSE. Therefore it is clear from the logic and Figure 1 that this model is overfit when the leaf size is < 10.

4 EXPERIMENT 2

I next transition to comparing my single decision regression tree learner to an ensemble of such learners. For this experiment I used the same dataset and train/test splits for an accurate comparison to the previous experiment. In the ensemble approach, I employ bootstrap aggregation (bagging) and the final prediction is taken as the mean of the individual predictions of the weak learners. In this ensemble however, the learners are normal decision trees and not randomized. I seek to discover whether the use of bagging can mitigate or even eliminate overfitting altogether. Once again I use RMSE as my error metric for determining this question. I arbitrarily chose to use and fix 50 bags for my ensemble bag learner and I vary the leaf size. Figure 2 shows the RMSE as a function of leaf size for the single decision tree and the ensemble of normal decision trees for both the training and test sets.

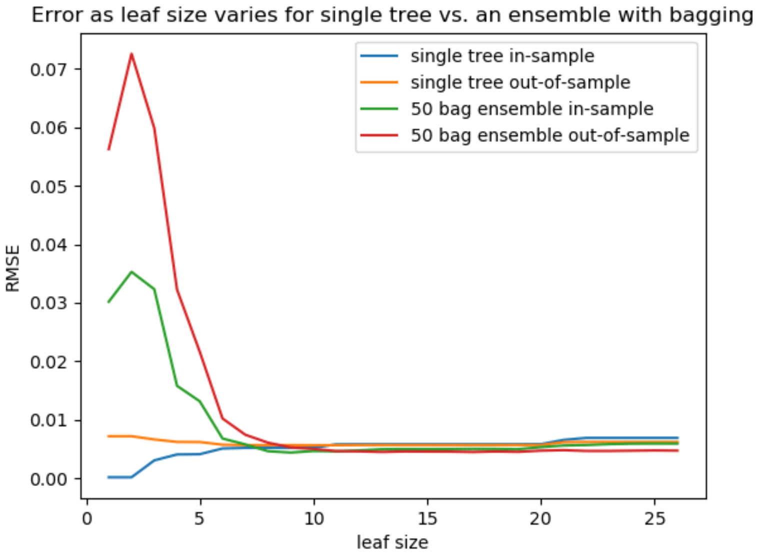

Figure 2—In-sample and out-of-sample error for a single decision tree and an ensemble of 50 with bagging as leaf size is varied.

Additionally, I aim to see whether an ensemble of random truly weak learners can mitigate overfitting even more than normal trees with bagging. I conducted the same experiment as seen in Figure 2 but with random trees and obtain the results shows in Figure 3 below.

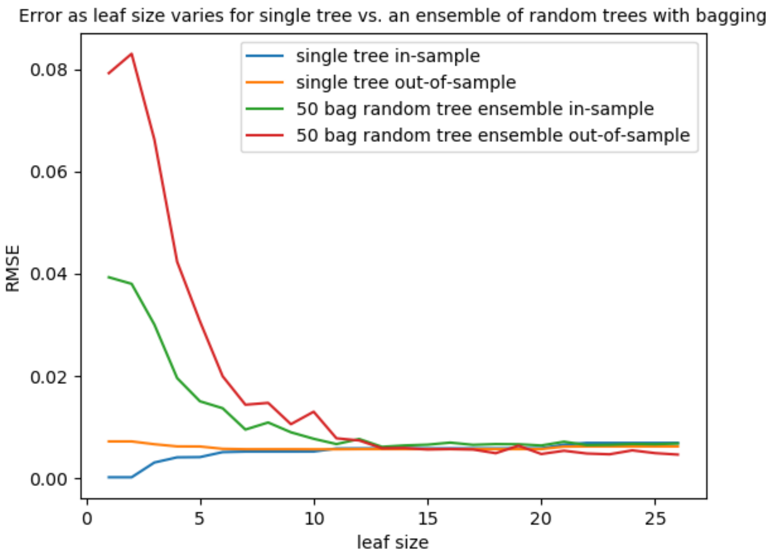

Figure 3—In-sample and out-of-sample error for a decision tree ensemble and an ensemble of 50 random trees with bagging as leaf size is varied.

One would expect the ensemble approaches to reduce the effect of overfitting because of their ability to generalize better to out-of-sample data. It can be seen from Figures 2 & 3 that ensemble approaches shift where the error is minimized and when the out-of-sample error falls below the in-sample error. As the model become increasingly randomized (both with trees and data selection), the shift in optimal leaf size tends towards the under-fitting end of the spectrum (higher leaf sizes). Additionally, it can also be seen that the ensemble approaches do eventually achieve lower RMSE error. Quantitatively, the optimal leaf sizes for the bag approach of normal trees and random trees is 16 and 25 respectively which shows their tendencies to migrate towards under-fitting.

5 EXPERIMENT 3

Finally, I use two additional metrics for comparing performance of the classic decision tree approach and the random decision tress approach. I once again use the same dataset and train/splits for an accurate comparison to the other figures. I use 50 bags for each method and inspect the mean absolute error (MAE) as well as the time to train the models as the leaf size is varied. MAE can have a slight advantage over RMSE if the dataset contains outliers or points whose errors are not greatly exploited and enhance through squaring. Also, in practical applications of more complicated versions of these models with larger datasets, the time to train the model can be greatly taken into consideration as time complexity and computational cost may be beneficial over achieve marginally higher performance (like RMSE). Figure 4 shows the MAE for the two different ensemble models.

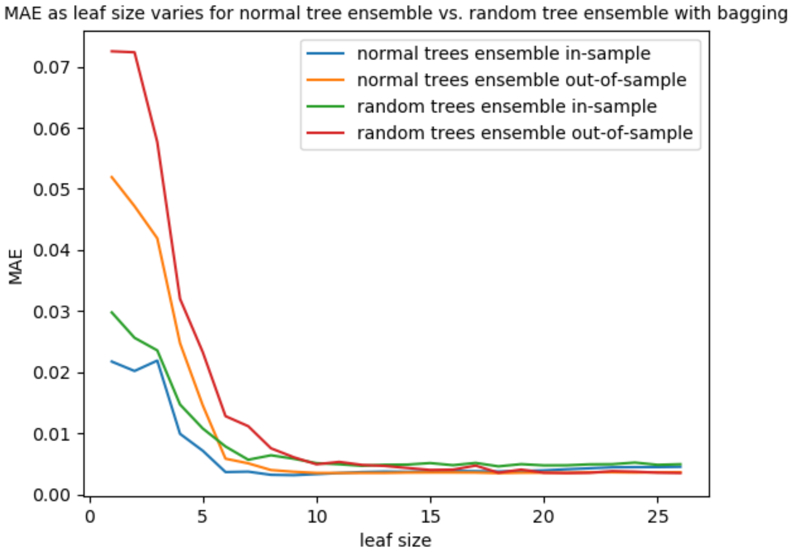

Figure 4—In-sample and out-of-sample MAE for a normal decision tree ensemble and an ensemble of 50 random trees with bagging as leaf size is varied.

Generally speaking, the normal trees ensemble appears to have lower MAE compared to the random trees ensemble when leaf size is low but because I previously determined that this is where overfitting is occurring, it is hard to make the distinction that it is the better choice. Also, the MAE for these models is extremely similar as we move to the less over-fit realm. Because of this I can now explore the time to train each of these models to see if there are notable differences. Figure 5 represents exactly this.

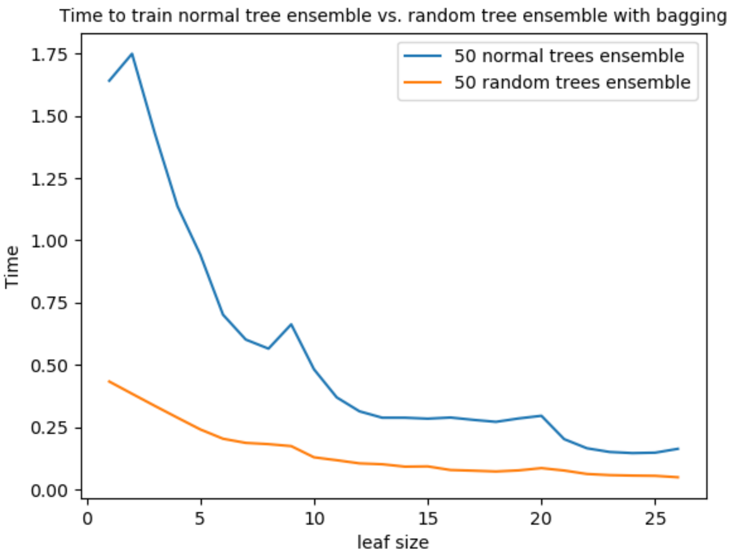

Figure 5—The time required to train a normal decision tree ensemble and a random decision tree ensemble both with bagging as leaf size is varied.

This shows that the random tree ensemble has a pretty significant training time benefit compared to the normal tree ensemble even when a leaf size of 10 were to be used. If these models were scaled, this could likely play a massive factor in model selection. These two metrics can work in tandem because as it can be seen in Figure 4, the MAE improvement may be marginal at large leaf sizes but the time savings could be huge. Because of these facts, I would argue that the random trees ensembles has better performance. This makes sense however as one thinks about the backbone of the A Cutler algorithm vs. the JR Quinlan algorithm. In the decision trees, when a random feature is selected this saves an immense amount of time compared to computing the correlation matrix for each feature with the labels. It can be expected that the random tree learner will be superior in this way nearly all the time due to computing the correlation matrix but that is not necessarily true for comparing MAE. However, in expectation, normal trees will likely show better MAE performance in the overfitting realm as the tree is split based on the “best" feature.