Project 3: Assess Learners

Aditya Kommi
akommi3@gatech.edu

***Abstract*—This project tests the four CART regression algorithms: a Decision Tree learner based on JR Quinlan's algorithm, a Random Tree learner based on A Cutler's algorithm, a Bootstrap Aggregating Learner, and an Insane Learner.**

1 INTRODUCTION

This project is about implementing four supervised machine learning algorithms called Classification and Regression Trees (CARTs) and evaluating their behavior and performance as one of their hyperparameters (leaf size) varies. The main dataset used to test the CARTs is the Istanbul dataset, which included the historical returns of international indexes across several days, and the trees are used to predict the MSCI Emerging Markets (EM) index. The main goal is to determine how leaf size affects overfitting through the performance metrics.

2 METHODS

The CART Learners implemented include a Decision Tree Learner based on JR Quinlan's algorithm, a Random Tree learner based on A Cutler's algorithm, a Bootstrap Aggregating Learner (bag learner) that can use other learners, and an Insane Learner, which uses a collection of bag learners. The data is split into x and y values as well as a training set and testing set with the training set accounting for 60% of the total data. In the experiment, the data set was randomized with a set random key for reproducibility.

The main hyperparameter changed is the leaf size of the underlying Tree. Leaf size refers to the maximum number of samples used to form a leaf node. The trees are built recursively with Decision Trees using correlation to find the best feature to split the data and Random Trees selecting a random feature to split the data. The median of the feature data is used to determine the split between the left and right nodes. At leaf nodes, the mean of the remaining y values is used to predict during queries to the tree.

To determine overfitting, the trees formed with various leaf sizes (from 1 to 50) are accessed based on in-sample error and out-sample error, which is based on predictions made on the training set and testing set respectively. In Experiment 1, Root Mean Squared Error (RMSE) is used to test Decision Trees. In Experiment 2, RMSE is used to test bagged Decision Trees with a fixed bag size of 20. In Experiment 3, Decision Trees and Random Trees without bagging are tested again on other error metrics as Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Maximum Error (ME), as well as Coefficients of Determination (R-Squared) and Training Time.

3 DISCUSSION

3.1 Experiment 1

This experiment looks at overfitting on basic Decision Trees as leaf size varies. Overfitting is when out-sample or out-of-sample error increases while in-sample error decreases where sample refers to the data set used to train the model. Overfitting occurs when models attempt to gain accuracy on predicting the training set, while sacrificing robustness and general application outside of the training sample. This is important to minimize and mitigate as models are trained for use outside of training samples. As shown in Figure 1, while the errors diverge under leaf size 20, overfitting dramatically increases as leaf size decreases past leaf size 10 and is greatest at leaf size 1. At leaf size 1, out-sample RMSE is near non-existent as every training sample is categorized in a leaf node, however the model is not able to generalize for data outside of the training sample.

Figure 2 depicts a similar situation with Random Trees with overfitting dramatically increasing under leaf size 6. As leaf size approaches 1, the model is not able to generalize past the training set. Random Trees however can work with smaller leaf sizes compared to Decision Trees as the randomness in the splits adds to the robustness of the model. The RMSE of in-sample error does not continuously increase in Random Trees.

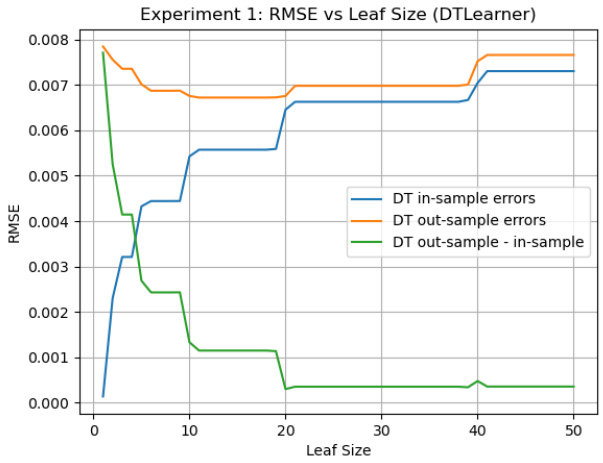

Figure 1— The Decision Tree's RMSE for in-sample vs out-sample is plotted with respect to the leaf size of the decision tree used to query.

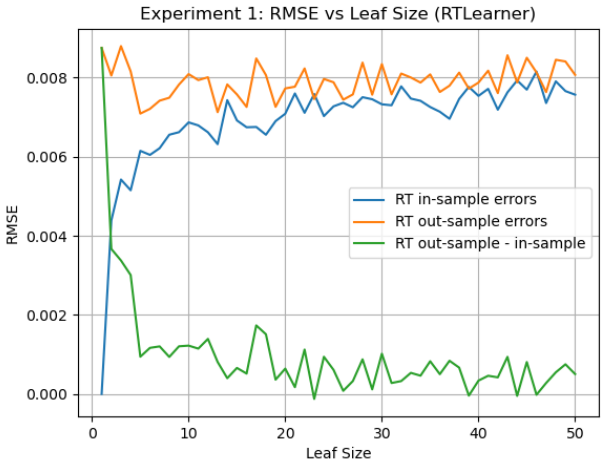

Figure 2— The Random Tree's RMSE for in-sample vs out-sample is plotted with respect to the leaf size of the decision tree used to query.

3.2 Experiment 2

Experiment 2 looks at the effect of bagging on overfitting. Figure 3 when compared to figure 1 shows for DT Learners bagging reduces out-sample RMSE and for in-sample RMSE for higher leaf-sizes at the cost of higher RMSE for in-sample with small leaf sizes. Similarly for RT Learners, bagging reduces overall RMSE, at the cost of low leaf size in-sample RMSE. Bagging does not eliminate

overfitting as there is still overfitting under leaf size 14 using Decision Trees and less than leaf size 12 for Random Trees, however it is reduced. Bagging is less effective for Random Trees compared to Decision Trees in terms of improvement, overall RMSE and the effect of bagging is reduced at higher leaf sizes as the models already generalize well.

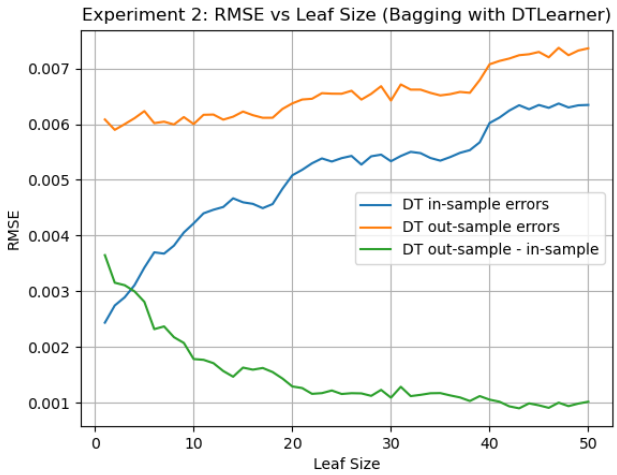

Figure 3 — The Bagged Decision Tree's RMSE (bag size 20) for in-sample vs out-sample is plotted with respect to the leaf size of the decision tree used to query.

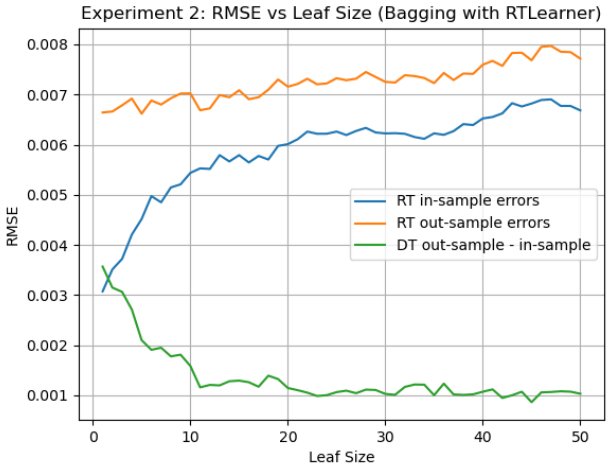

Figure 4 — The Bagged Random Tree's RMSE (bag size 20) for in-sample vs out-sample is plotted with respect to the leaf size of the decision tree used to query.

3.3 Experiment 3

Decision Trees show lower MAE, higher R-Squared values, lower MAPE, and smaller ME values. Almost across the board for the Istanbul dataset, Decision Trees outperformed Random Trees even at lower leaf sizes in those metrics while still being overfit to training samples. These metrics were calculated for both in-sample and out-sample query predictions for each leaf size.

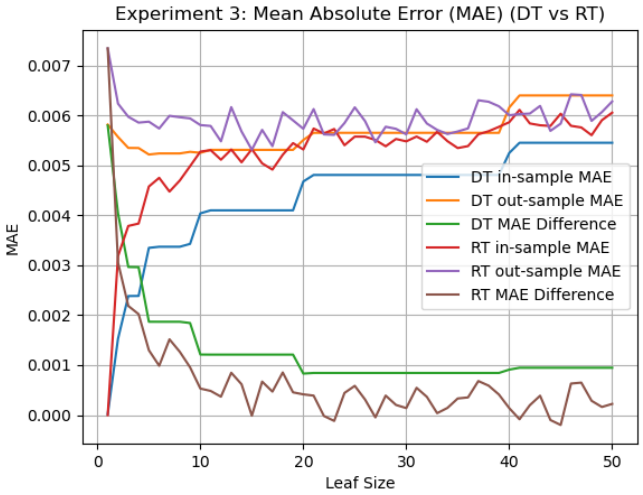

Figure 5—MAE for DT and RT learners and the difference.

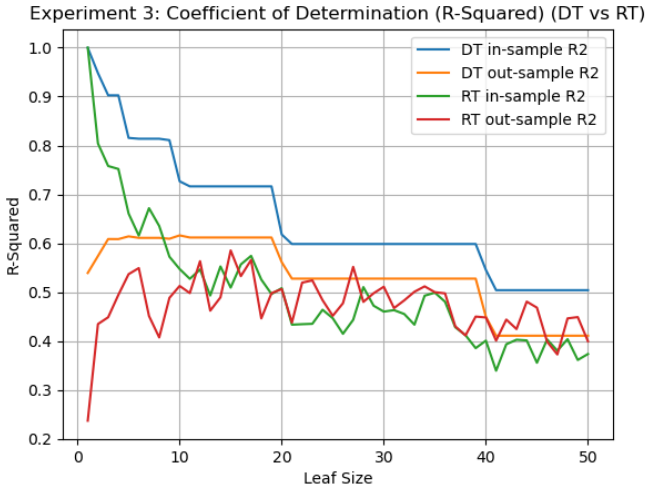

Figure 6— R-Squared for DT and RT learners.

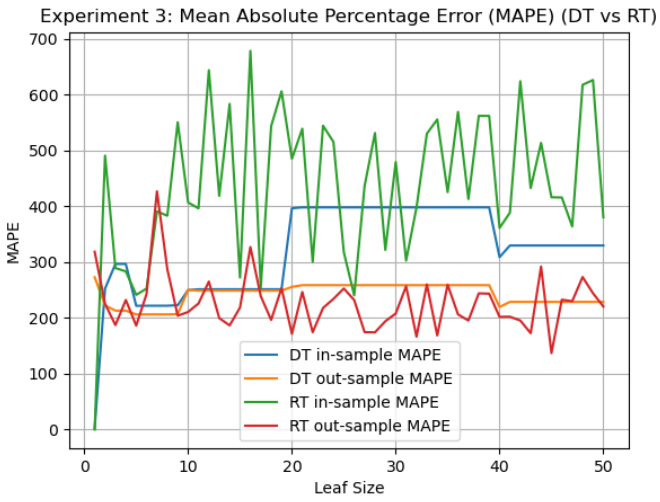

Figure 7— MAPE for DT and RT learners.

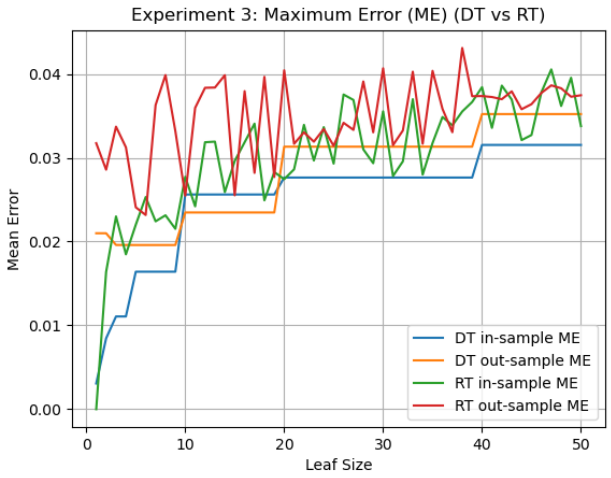

Figure 8— Maximum Errors for DT and RT learners.

Figure 9 shows the time to train each Tree based off leaf size. Time was taken before the training data was added and after the tree was built. In this metric, Random Tree Learners outperform Decision Tree Learners across the board. While Decision Trees have better explainability and greater accuracy for smaller datasets, Random Trees have a relatively negligible time to train and offer relatively decent performance. They would have the potential to shine with much larger datasets that would take decision tree too long to train. Random trees are less prone to overfitting than Decision trees, however with a reasonable

leaf size Decision trees are more accurate. Neither is always superior, rather it depends on the use case and need for accuracy vs speed.

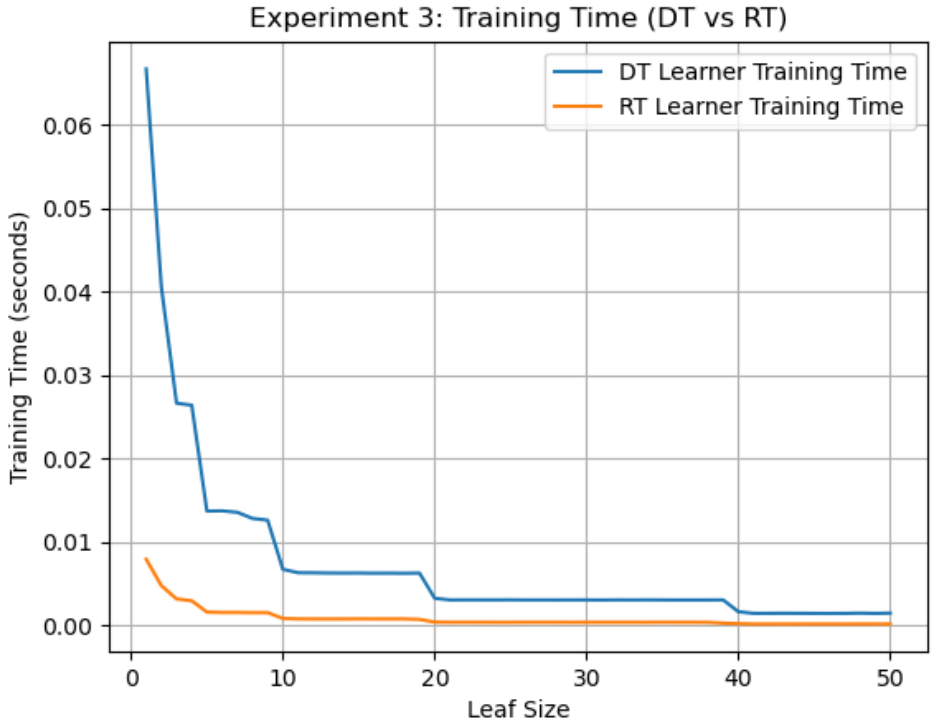

Figure 9— Time to Train for DT and RT learners.

4 SUMMARY

For the Istanbul dataset without bagging, Decision Trees with a leaf size between 10-19 perform the best with a compromise between error and time to train while maintaining the highest R-squared value for out-of-sample data. Decision trees are more accurate than Random trees but suffer from greater risk of overfitting the data without a sufficient leaf size to improve generalization. Bagging or ensemble methods can also help improve the accuracy and robustness of decision tree learners. Random Trees can generalize well at the cost of accuracy without bagging and are much faster to construct. This would be magnified with a larger, or continuously updating training dataset that would require rebalancing the learner or when implementing ensemble methods such as bagging or an Insane Learner that uses multiple bags of Random Trees.