Project 3 : Assess Learners

ML4T FALL 2024

Huida David Shi
hshi320@gatech.edu

Abstract

This project investigates and compares the performances of decision tree and random tree learners. We also deep dive into how leaf sizes and bagging (Decision Tree Learner as base) affects overfitting and model performance.

Introduction

The objective of the project is to use investment return data of 8 stocks to predict the return of another stock (MSCI Emerging Markets).

Clarification of files:

- DTLearner.py = Decision Tree Learner
- RTLearner.py = Random Tree Learner
- BagLearner.py = Decision Tree Learner with bagging (bootstrap samples)
- InsaneLearner.py = Ensemble of 20 Linear Regression Learners with Bagging
- testlearner.py = file to ingest and clean data then model data using the different learners

Method

For the purpose of this project we remove the date column and convert the time series data into random sample data. Training data is 60% of the data randomly selected, test data would be the remaining 40%. We train different algorithmic methods on the same training data set and compare model evaluation using RMSE (root mean squared error), MAE (mean absolute error) and R-squared.

1.1 Leaf Size and Overfitting [experiment 1]

Overfitting is when a model performs well on training data but poorly on testing data. Leaf size refers to the minimum number of data points that a leaf can hold. When the leaf size is equal to the number of data points, the whole dataset will fall under one leaf. This means as the number of leaf sizes increases, the complexity of the model decreases. As we increase the leaf size from 1

we reduce overfitting until around leaf size 5. We can see the RMSE on the training data is extremely low when leaf size is 1, however the leaf size 1 RMSE on test data is very high. This is an example of overfitting. The gap between training RMSE and test RMSE decreases as we increase leaf size 1 to 5. By fitting the model more loosely to the training set, the test RMSE decreases as the training RMSE increases. If the leaf size is increased too much the RMSE gap between training and test stays the same but overall RMSE for both increases. This is an example of underfitting.

1.1.1 *FIGURE 1*

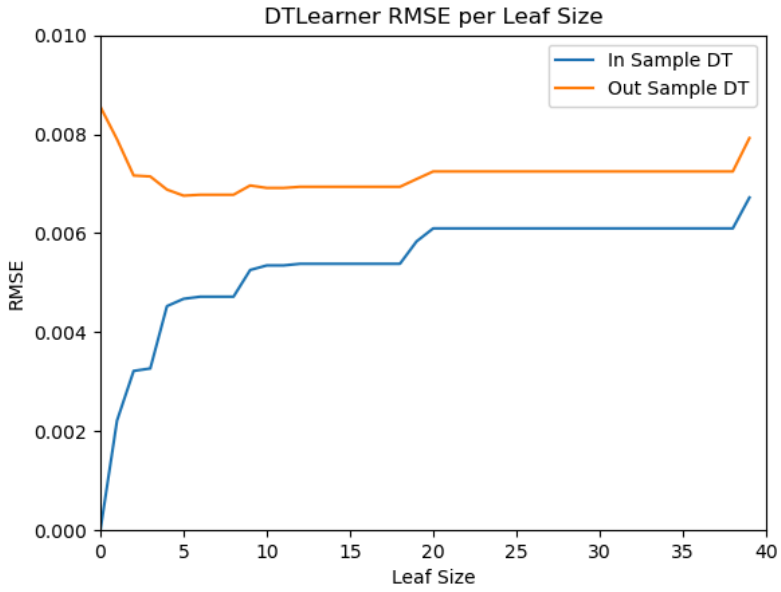

Figure1 shows Decision Tree RMSE for in and out of sample.

1.1.2 *FIGURE 2*

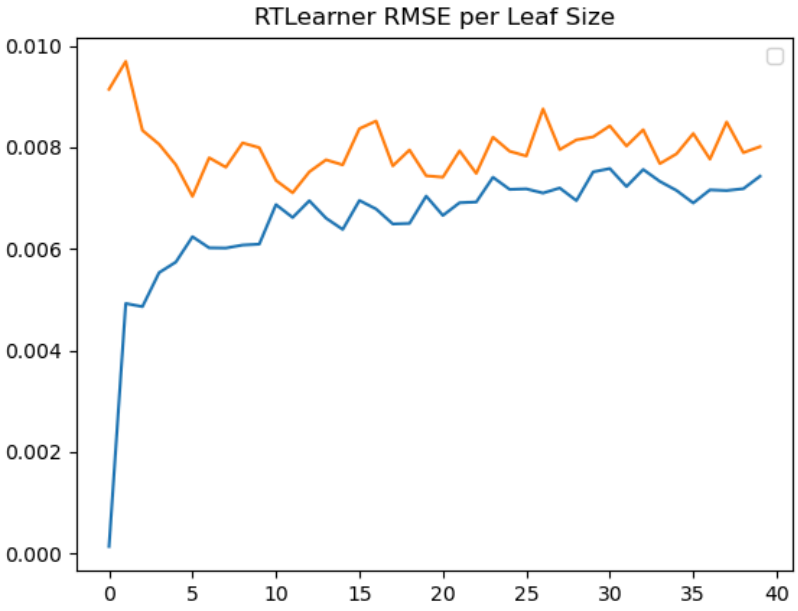

Figure2 shows Random Tree RMSE for in and out of sample.

1.2 Bagging and Overfitting [experiment 2]

Bagging is training several versions of a model on random subsets of the data. Subsets are created by random sampling with replacement, these samples are called bootstrap samples. The predictions of each model are averaged to form the final prediction. This creates a generalization of the data and reduces prediction variance which ultimately reduces overfitting. As we decrease the leaf size, the training RMSE decreases showing closer fit to the data set. However compared to figure 1 decision tree without bagging, the test RMSE with bagging is extremely stable for all leaf sizes. Even when we try to overfit by reducing the leaf size to 1 (figure 3) the test RMSE is fairly normal compared to the other leaf sizes.

Bagging reduces overfitting by averaging out predictions, less weight on outliers and stabilizing model performance across different bootstrap samples. As leaf size increases, the gap between training and test RMSE narrows indicating reduction in overfitting, but we can still see a noticeable gap between the two RMSE. This shows that bagging can reduce overfitting but not eliminate it.

1.2.1 *FIGURE 3*

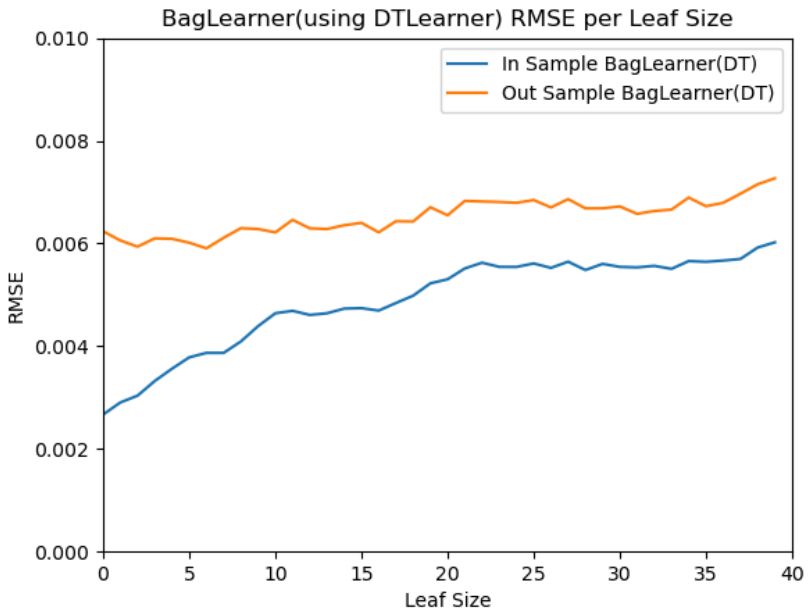

Figure shows Decision Tree with Bagging RMSE for in and out of sample. (Bags = 20)

1.3 Comparing Decision Tree and Random Tree Learners [experiment 3]

1.3.1 *COMPARISON BASED ON MAE (FIGURE 4, FIGURE 5):*

MAE is a metric to measure average errors in a set of predictions. Absolute value is used so when we average the errors negatives and positives do not cancel each other out

Both algorithms show reduction in overfitting when leaf size is increased, especially in the ranges 1-5. DTLearner shows better performance specifically when leaf size is low. The MAE for DTLearner is much lower in and out of sample compared to RTLearner. In addition, RTLearner has more fluctuation in both in and out of sample error indicating the model is less stable than DTLearner.

1.3.2 *COMPARISON BASED ON R-SQUARED (FIGURE 6, FIGURE 7):*

R-squared is a metric to measure how well the model fits the data. It tells us how much variance is explained by the independent variables. When R-squared is 1, the predictions match actual

values exactly. When R-squared is 0, the predictions are worse than simply using the mean of the training data.

Both algorithms have high (close to 1) and stable in sample r-squared indicating that the models fit the training data extremely well. When comparing the out of sample r-squared, DTLearner is more stable and has a higher r-squared than RTLearner. This shows although the training fit is quite similar for both Learners, DTLearner is performing more accurately out of sample.

1.3.3 *FIGURE 4*

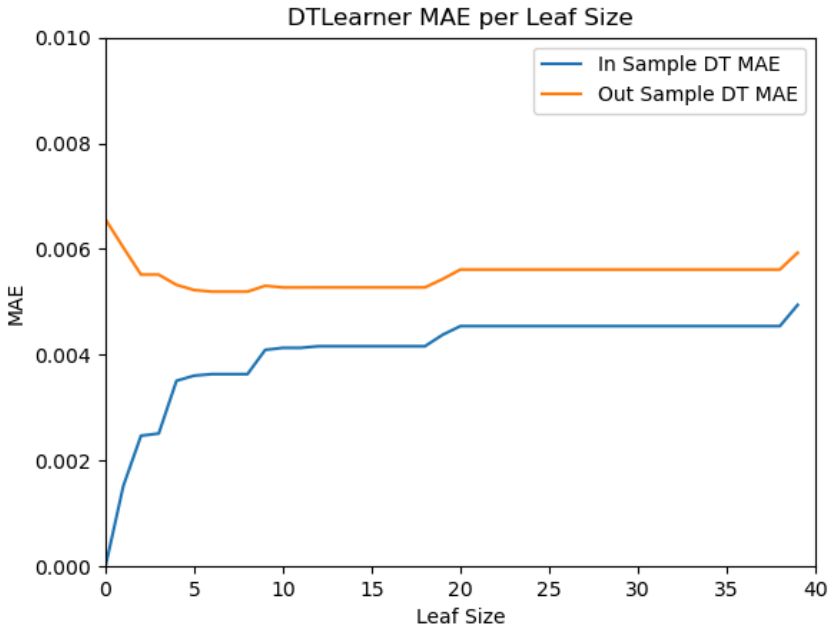

Figure shows Decision Tree MAE (mean absolute error) for in and out of sample.

1.3.4 *FIGURE 5*

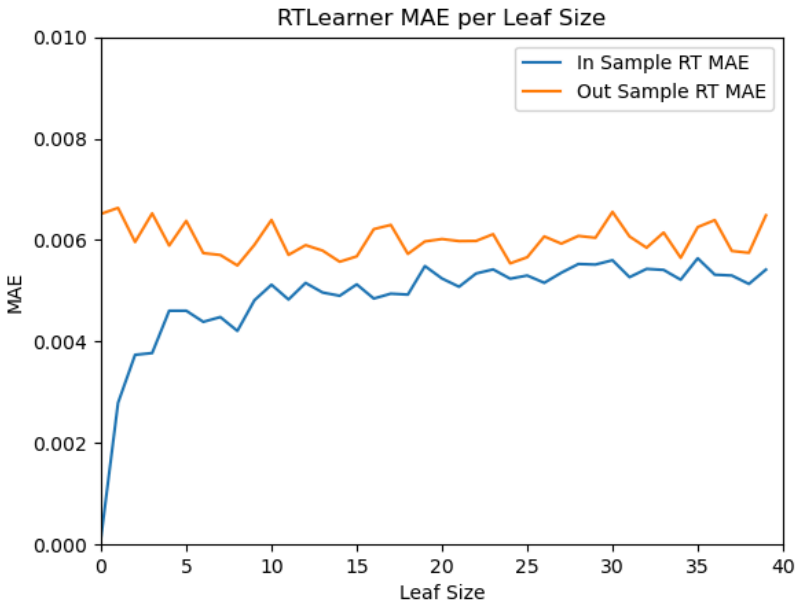

Figure shows Random Tree MAE (mean absolute error) for in and out of sample.

1.3.5 *FIGURE 6*

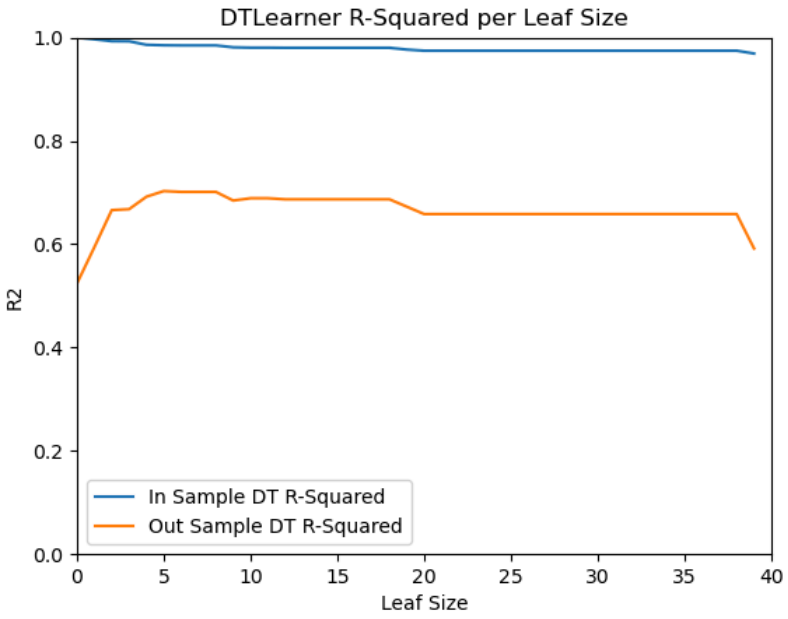

Figure shows Decision Tree R-squared for in and out of sample.

1.3.6 *FIGURE 7*

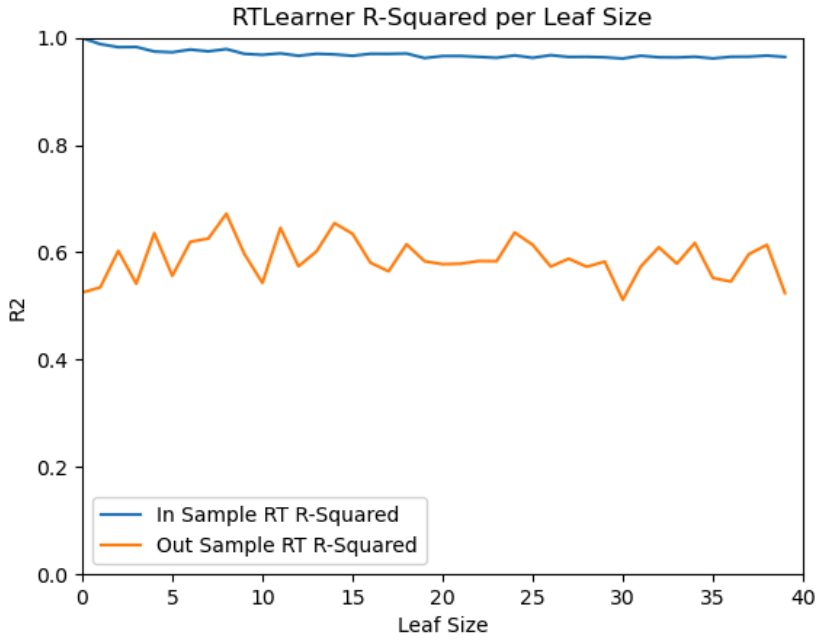

Figure shows Random Tree R squared for in and out of sample.

Summary

We compare the performance of Decision Tree Learner and Random Tree Learner, focusing on their susceptibility to overfitting and the impact of bagging on model performance. The findings indicate that DTLearner outperforms RTLearner in terms of both Mean Absolute Error (MAE) and R-squared, especially when leaf size is small. Additionally overfitting is reduced when leaf sizes are increased. Lastly, bagging reduces overfitting but does not entirely eliminate it, as shown by the gap between training and test RMSE for both learners .