# Project 3 : Assess Learners (CS7646)

Stella Lai-Hoong Soh

lsoh3@gatech.edu

***Abstract*—**This project aims to bring readers to an understanding of four CART regression algorithms : a "classic" Decision Tree Learner, a Random Tree Learner, a Bag Learner (also known as Bootstrap Aggregating Learner) and an Insane Learner (combining Linear Regression with bagging).

## 1 INTRODUCTION

The performance of machine learning models on an in-sample training set versus an out-of-sample testing set is impacted by hyperparameter tuning. Often, overfitting and underfitting occur. In this project, experiments will be conducted to quantitatively measure metrics such as RMSE, variance and Mean Absolute Error (MAE) to assess the effectiveness of Decision Tree learner and Random Tree learner.

## 2 METHODS

Assuming you have the framework for local environment and ML4T Software setup, the experiments could easily be repeated by running the four files - DTLearner.py, RTLearner.py, BagLearner.py and InsaneLearner.py with the driver, testlearner.py. Conduct the experiments by editing the testlearner.py to display in-sample and out-of-sample RMSE values versus leaf size for Experiment 1 and Experiment 2. Tabulate the results as shown in **3. Discussion** below, and plot the graphs. For Experiment 3, I have decided on Variance as one metric and Mean Absolute Error (MAE) as another metric to quantitatively compare the use of decision trees versus random trees.

## 3 DISCUSSION

### Experiment 1

Using the Istanbul.csv with DTLearner, I found the following relationship between leaf size and in-sample and out-of-sample RMSE values:

| Leaf_size | In-sample RMSE | Out-of-sample RMSE |
|---|---|---|
| 5 | 0.0042253 | 0.0056957 |
| 10 | 0.0042253 | 0.0056957 |
| 15 | 0.0053468 | 0.0053756 |
| 30 | 0.0052860 | 0.0040336 |
| 50 | 0.0074501 | 0.0062093 |
| 60 | 0.0075624 | 0.0062300 |
| 80 | 0.0087720 | 0.006274 |
| 100 | 0.0090121 | 0.0062247 |

***Table 1*** —Leaf size versus In-sample RMSE and Out-of-sample RMSE

In plotting out the in-sample and out-of-sample RMSE values versus the leaf size, I have a chart as follows:

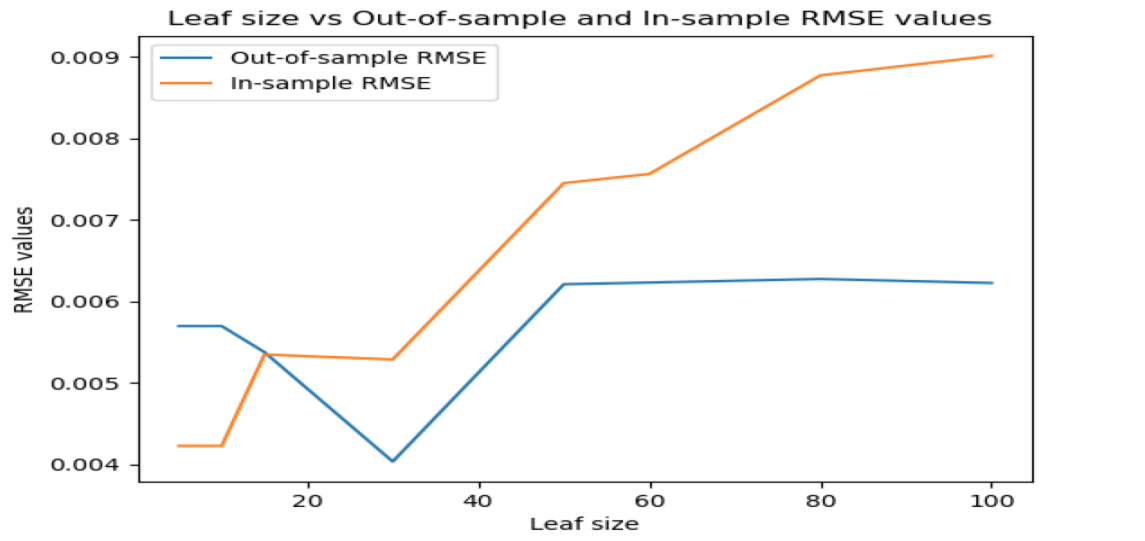


***Figure 1***—Leaf size vs. Out-of-sample and In-sample RMSE values with DTLearner

As shown in Figure 1 above, overfitting - whereby out-of-sample RMSE is greater than in-sample RMSE - does occur with respect to leaf size. In particular, we can see that leaf size is inversely proportional to overfitting. The lower the leaf size, the higher will be the overfitting. In other words, at lower leaf size values, there is a tendency for testing data error to be greater than training data error.

As illustrated in Figure 1, for leaf size less than 15, the decision tree learner tends to overfit with Istanbul.cvs data. Once leaf size increases beyond 15, the in-sample data error is larger than out-of-sample data error. This means beyond leaf size of 15, training data error is larger than testing data error, and does not capture the underlying trend of the data. In this case, beyond leaf size of 15, the model does not fit the data well.

**Experiment 2**

For experiment 2, I trained 20 DT learners with 20 bags of the Istanbul.csv dataset. The table below:

| Leaf_size | In-sample RMSE | Out-of-sample RMSE |
|---|---|---|
| 5 | 0.0034627 | 0.0044303 |
| 10 | 0.0044694 | 0.0045442 |
| 15 | 0.00503925 | 0.0045675 |
| 30 | 0.0058578 | 0.0046818 |
| 50 | 0.0065003 | 0.0048613 |
| 60 | 0.0065906 | 0.0062300 |
| 80 | 0.0079512 | 0.006274 |
| 100 | 0.00795118 | 0.0062247 |

***Table 2*** —Leaf size versus In-sample RMSE and Out-of-sample RMSE

shows how the in-sample and out-of-sample RMSE values change with respect to leaf size.

In plotting out the in-sample and out-of-sample RMSE values versus the leaf size, I have a chart as follows:

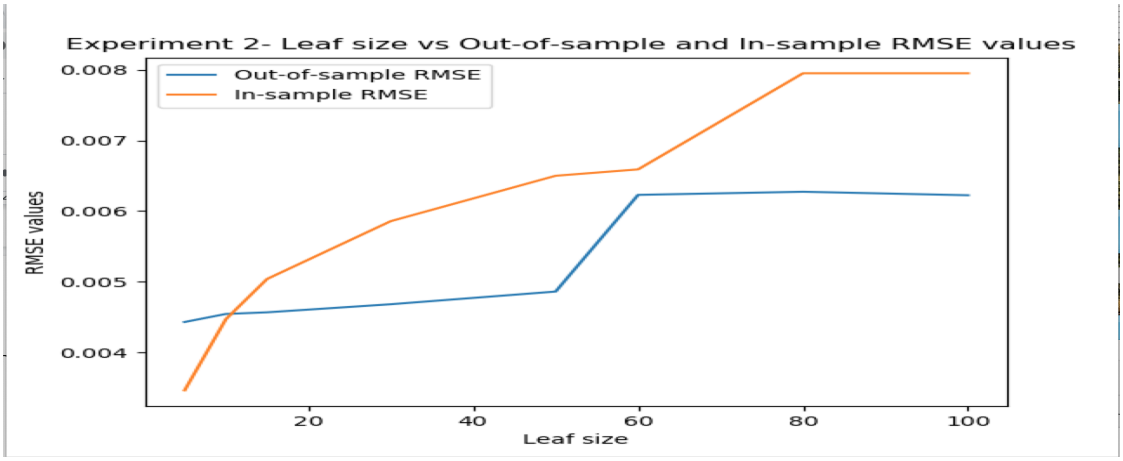


***Figure 2***—Leaf size versus Out-of-sample and In-sample RMSE values with BagLearner

Overfitting is determined by measuring the out-of-sample RMSE versus the in-sample RMSE with respect to the leaf size. The above Figure 2 shows that as leaf size increases, the in-sample RMSE is greater than the out-of-sample RMSE. This means that with bagging, the issue of overfitting is reduced as leaf size increases. This is logical as bagging calls for different samples of data to be used for different trees, and this helps the learner to generalize well over the test data, and hence the out-of-sample RMSE is smaller than the in-sample RMSE with increases in leaf size. However, bagging does not entirely eliminate overfitting. As we see in Figure 2 above, there is a point somewhere around leaf size = 10, whereby out-of-sample RMSE is greater than in-sample RMSE. From this observation, overfitting seems to occur at a smaller leaf size with BagLearner than using a pure Decision Tree learner as in the case of Experiment 1.

**Experiment 3-A**

I conducted an experiment to measure in-sample and out-of-sample DTLearner and RTLearner versus the leaf size. The variance is obtained by taking the np.nanvar() of the predicted y values, as I wanted to skip the NaN values.

The two tables below are my data:

| **Leaf_size** | **DT_ In-sample_variance** | RT_**Out-of-sample RMSE** |
|---|---|---|
| 5 | 0.000121058 | 9.70629e-05 |
| 10 | 0.000110289 | 8.87709e-05 |
| 20 | 0.00010029 | 7.23203e-05 |
| 40 | 8.82499e-05 | 6.30011e-05 |

| | | |
|---|---|---|
| 50 | 8.32888e-05 | 6.543121e-05 |
| 60 | 8.15989e-05 | 6.54312e-05 |
| 80 | 6.17778e-05 | 5.02808e-05 |
| 100 | 5.74966e-05 | 4.17260e-05 |

***Table 3*** —Leaf size versus DT-in-sample_variance and RT-in-sample_variance

| **Leaf_size** | **DT_Out-of-sample_variance** | RT_**Out-of-sample_variance** |
|---|---|---|
| 5 | 8.10550e-05 | 6.56987e-05 |
| 10 | 6.33288e-05 | 7.10736e-05 |
| 20 | 6.94411e-05 | 6.00291e-05 |
| 40 | 6.57966e-05 | 4.47358e-05 |
| 50 | 6.23759e-05 | 5.29354e-05 |
| 60 | 6.07161e-05 | 5.29354e-05 |
| 80 | 5.17165e-05 | 4.18489e-05 |
| 100 | 4 .66295e-05 | 4.29914e-05 |

***Table 4*** —Leaf size versus DT_Out-of-sample_variance and RT_Out-of-sample_variance

I plotted the graph below from Table 3's data:

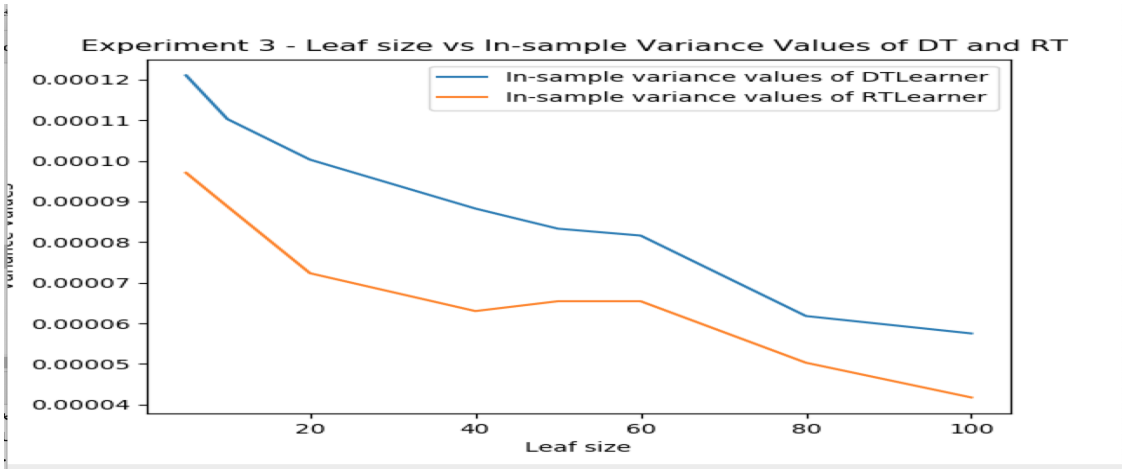


***Figure 3*** —Leaf size versus In-sample variance values of DTLearner and RTLearner

I plotted the graph below from Table 4's data:

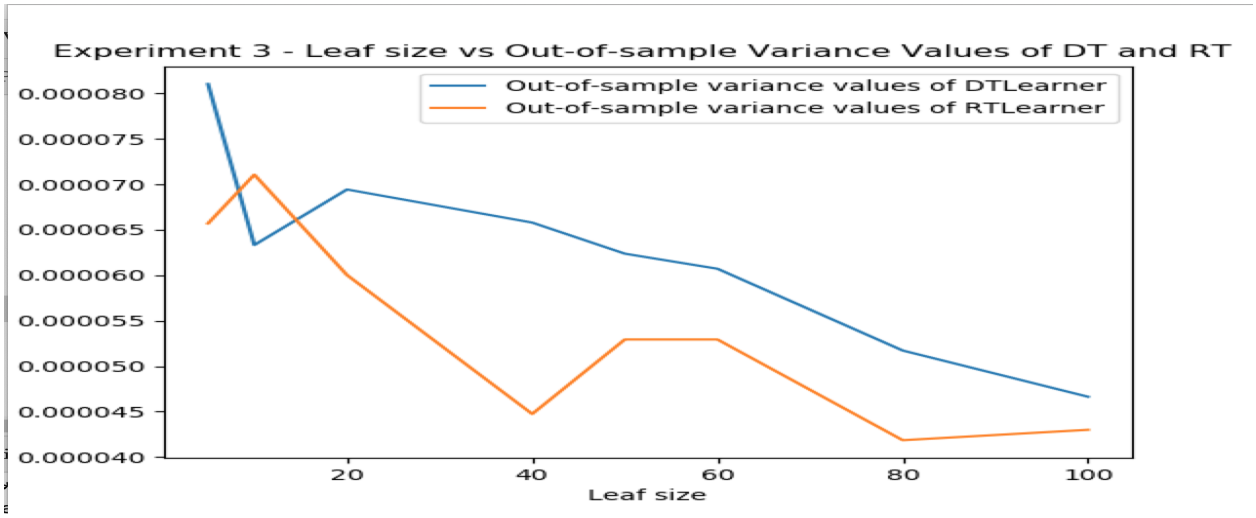


***Figure 4*** —Leaf size vs Out-of-sample Variance Values of DTLearner and RTLearner

From Figure 3, I observed that random tree's (RT) average in-sample variance (~6.30011e-05) is lower than the decision tree's (DT)average in-sample variance (~8.82499e-05). The same can be said of the random tree's (RT) average out-of-sample variance (~4.47358e-05) compared to the decision tree's (DT) average out-of-sample variance(~6.57966e-05), as seen in Figure 4. This means that decision tree (DT), having a larger in-sample and out-of-sample variance, is more sensitive to the specific data used during training.

**Experiment 3-B**

Another metric that I have used in conducting Experiment 3 is Mean Absolute Error (MAE) which measures the accuracy of a given model. The graphs I obtained are shown below:

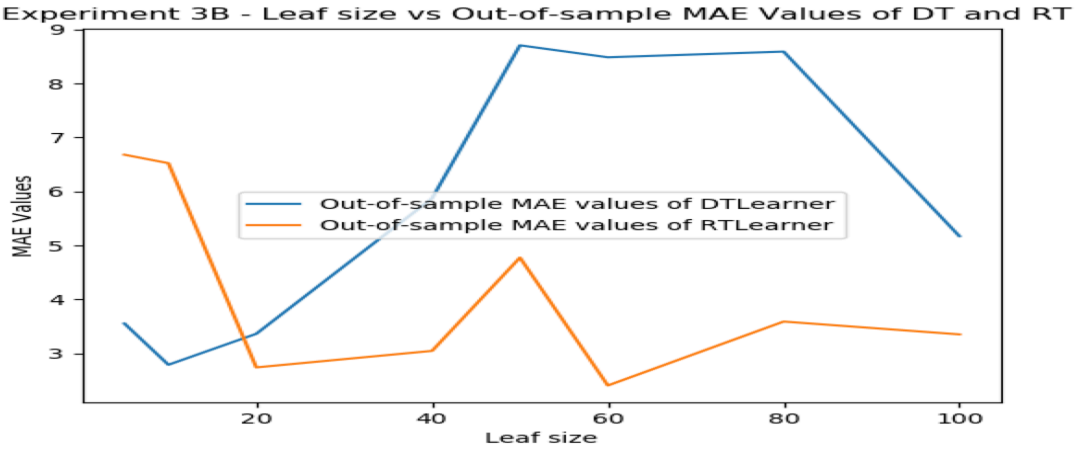


***Figure 5*** —Leaf size vs Out-of-sample MAE Values of DTLearner and RTLearner

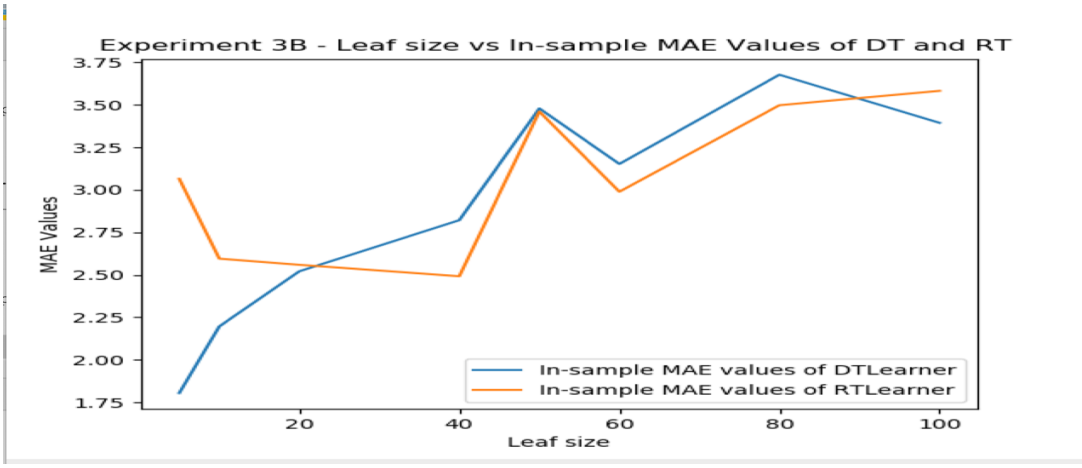


***Figure 6*** **--**Leaf size vs In-sample MAE Values of DTLearner and RTLearner

The lower the MAE is for a given model, the more closely the model is able to predict the actual values. From Figure 5, the out-of-sample MAE values of RTLearner are on average lower than the out-of-sample MAE values of DTLearner. From Figure 6, between leaf size of 20 and 90, the in-sample MAE values of RTLearner are lower than those of DTLearner. This means that between leaf size of 20 and 90, RTLearner is more accurate in predicting the actual values compared to DTLearner.

**Conclusion from Experiment 3**

While the classic decision tree (DT) learner is more sensitive to the data used during training, the random tree (RT) learner is more accurate in predicting the actual values, as shown by Figure 6 above. This is a little surprising as I would expect the DTLearner, with more sophistication in its algorithm, to be more accurate in predicting the actual values.

Based on these two selected metrics, it is hard to conclude which learner has better performance as other factors such as computing power and the computing architecture may also impact the time to train, and thus the sensitivity of the model and its accuracy in predicting.

As we can see above, no one learner is always superior to another. One has to evaluate the effectiveness of a training model based on more than the variance and the MAE.

**SUMMARY**

In Experiment 1, we observe that as the leaf size values decrease, the DTLearner tends to overfit the training data. In Experiment 2, with the BagLearner, the effect of overfitting is reduced as the leaf size increases. In Experiment 3, from a variance perspective, the DTLearner is more sensitive to the data during training, but from an accuracy perspective, the RTLearner wins.