CS7646 Project 3: Assess Learners

Yuxi Chen

ychen3281@gatech.edu

Abstract—In this report, I will discuss the implementation of DTLearner, RTLearner and BagLearner on the Istanbul.csv dataset. The parameters that I will alter are the leaf size, bag size and metric type. Using Root Mean Squared Error (RMSE), I will determine how overfitting changes with the parameters. Coefficient of Determination (R-Squared) and Mean Absolute Percentage Error (MAPE) will be used as metrics to compare goodness of fit and model accuracy between "classic" decision trees (DTLearner) versus random trees (RTLearner).

1 INTRODUCTION

Decision Tree Learner (DTLearner), Random Tree Learner (RTLearner) and Bootstrap Aggregating learner (BagLearner) are three Classification and Regression Tree (CART) algorithms. The aim of this project is to assess the three types of learners to understand how to properly implement these learners given different datasets and problems. Figure 1 depicts, when a classification tree is used, the aim is to split the dataset at hand into two parts using the homogeneity of data as criterion [4]. In this study, the dataset I will use is the Istanbul.csv dataset which contains the return of the Istanbul stock exchange and seven other stock indexes dated from January 5, 2009 to February 22, 2011. Based on the methodologies used in my approach my initial hypothesis is that increase in leaf size and bag size will reduce overfitting. Bagging has been demonstrated to give impressive improvements in accuracy by combining together hundreds or even thousands of trees into a single procedure [3]. Additionally, I predict that DTLearner will perform better than RTlearner, due to the randomness in RTlearner's feature selection.

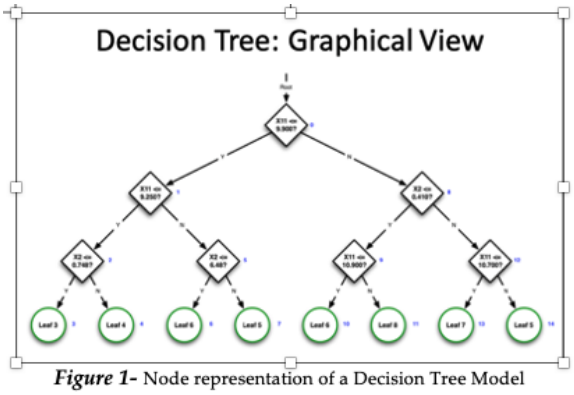


Figure 1- Node representation of a Decision Tree Model

2 METHODOLOGY

To begin experimentation, Istanbul.csv dataset is cleaned, randomized and split using a 60/40 ratio for training and testing respectively. This results in four datasets train_x, train_y, test_x, and test_y. DTLearner takes in the parameter, leaf_size to determine the splits in the decision tree. Feature selection is determined by taking the highest absolute correction of the features in train_x with train_y. RTLearner performs similarly, but the feature selection is determined randomly. In Baglearner, collect a random subset of data from the train_y and fit it into the learner used in Baglearner. This process is repeated by the number of bag sizes and resulting values are averaged.

In experiment 1, RSME was used as a metric to compare the predicted value to the actual values produced by the DTLearner. In this experiment, the leaf size was modified and charted to display when the values become overfitted. Leaf sizes from the range of 1 to 50 were used.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

In experiment 2, RSME was used as a metric to compare the predicted value to the actual values produced by bagging DTLearner. In this experiment, both the leaf size and the bag size were modified and charted to display when the values become overfitted. Leaf sizes from the range of 1 to 50 and bag sizes of [10, 25, 50] were used.

In experiment 3, MAPE and R-Squared are used as the metric to compare the values produced by DTLearner and RTLearner to determine any differences and analyze which model is better. MAPE, compares the predicted and actual values by average percentage difference. Whereas, R-squared is used to determine the variance produced by the model.

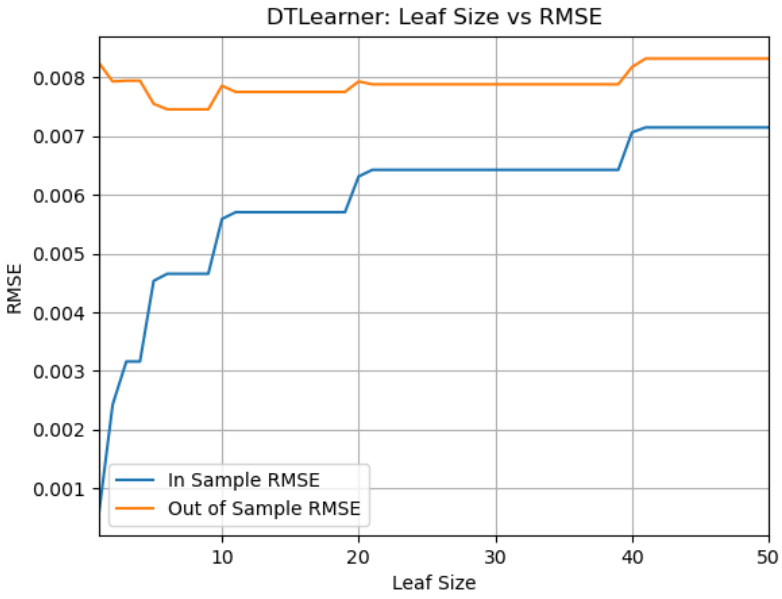
$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i} \cdot 100\%$$

$$R^2 = \frac{SSR}{SST} = \frac{\sum (\hat{y}_i - \bar{y})^2}{\sum (y_i - \bar{y})^2}$$

3 EXPERIMENTS & DISCUSSION

3.1 Experiment 1

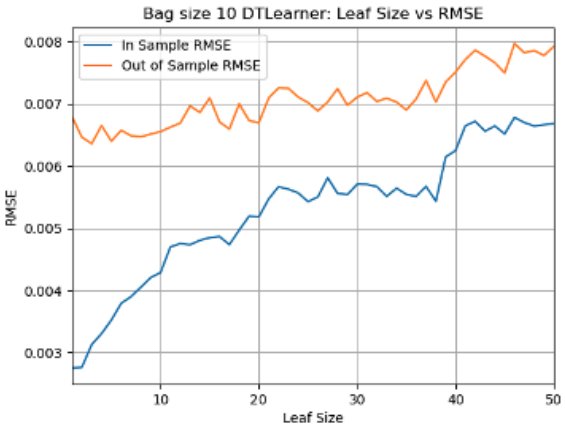
RMSE values produced by the training data are labeled as In Sample RSME and testing data are labeled as Out of sample RMSE. Overfitting does occur with respect to leaf size. Based on the chart Experiment 1, it is shown that overfitting begins from leaf size 5 to 10, with leaf size 5 showing the largest difference in RMSE. Overfitting is determined when the values of the In Sample RMSE sharply decreases, while the Out of Sample RMSE sharply increases. Since the chart is read from right to left, it can be said to start at leaf size ten and continue to overfit as the leaf size decreases. Smaller leaf sizes are known to be prone to overfitting, one way to reduce this is by pruning to reduce noise and variance.



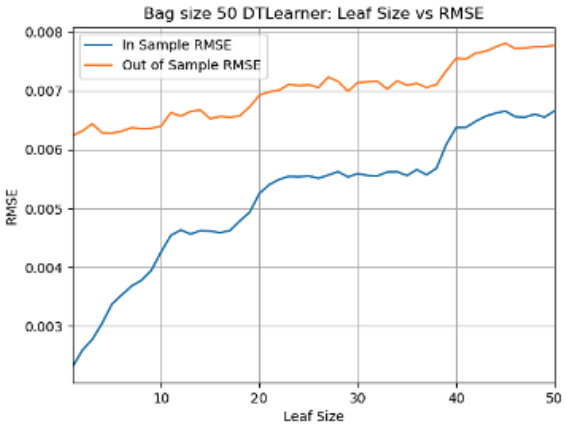
Experiment 1- RSME from DTLearning by changing leaf sizes

3.2 Experiment 2

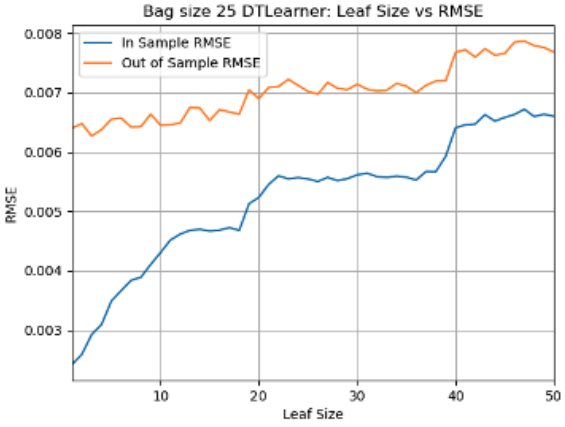
Bagging is known to reduce overfitting and improve the model's generalization by reducing the noise and variance. This can be seen in the three charts depicted by Experiment 2. Although all three charts show similar In Sample RMSE and Out of Sample RMSE values, it is noted that the chart with bag size 10 contains the most noise. Whereas, the chart with bag size 50 displays a smoother line. The reason for this is because, by increasing the number of sample subsets collected to use in DTLearner, you reduce variance and noise. In experiment 2, leaf size is used along with bag size; however, it can be seen that the overfitting was reduced. In Sample RMSE decreases, the Out of Sample RMSE also decreases. Although the model's overfitting improved, based on these results I can not conclude the bagging completely eliminates overfitting. Additionally, increasing bag size adds complexity to the model and increases run time.



Experiment 2– RMSE using 10 samples of DTLearner



Experiment 2– RMSE using 50 samples of DTLearner



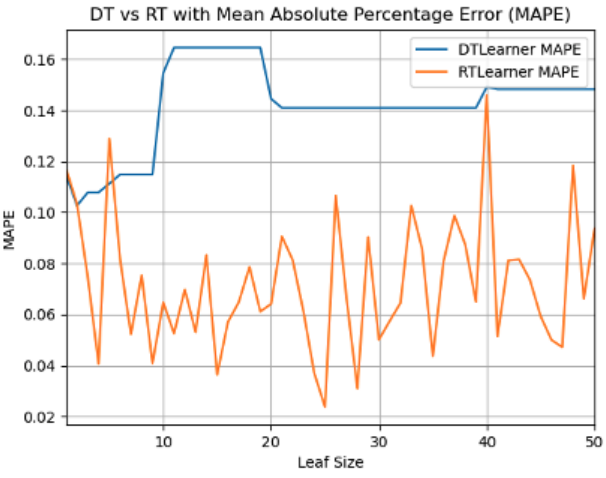
Experiment 2– RMSE using 25 samples of DTLearner

3.3 Experiment 3

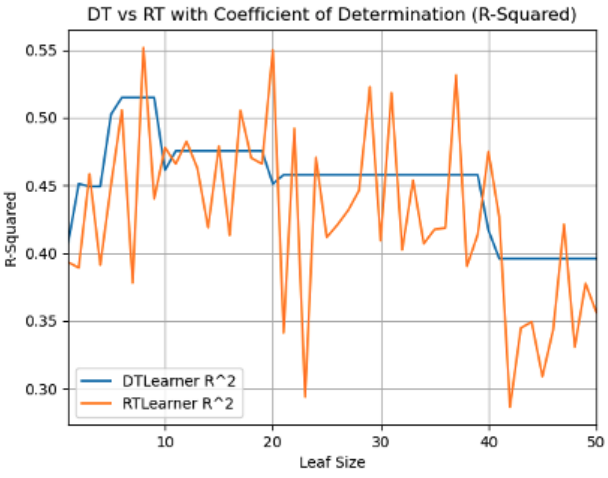
In MAPE, lower values indicate predictive performance as there is less percentage error. When comparing DTLearner to RTLearner, it can be seen that DTLearner starts with a lower MAPE value of 0.10 but increases sharply from leaf size 9 to 20 to 0.16 before plateauing at 0.14 from 20 to 50. Whereas, RTlearner's highest MAPE value is 0.14, with a lowest MAPE value of 0.02. Although the MAPE value shifts, with leaf size due, this is mainly due the randomness of the feature selection.

In R-squared, higher values indicate better goodness of fit. A higher value indicates that the predicted values are closer to the actual values. When comparing DTLearner to RTLearner, it can be seen that DTLearner the R-squared value increases from 1 to 9 leaf size, but starting from leaf size 10 it gradually decreases. Whereas, RTLearner's values change sharply from a high of 0.55 to 0.30 with no trend pattern.

Based on these results, strengths and weaknesses depicted in both models. DTLearner's feature selection is based on correlation and produces more reliable results with less deviation. On the other hand, both MAPE and R-squared depict sharp drops in value around leaf size 10, indicating overfitting in DTLearner. Whereas, RTLearner is not overfitted but the values contain high variance and may be difficult to interpret as a single random tree. It is difficult to say which learner will always be superior, as it may depend on how it is implemented. Even in this experiment, MAPE values for RTLearner, were mostly lower than DTLearner's values.



Experiment 3_1- Comparing DTLearner and RTLearner using Mean Absolute Percentage Error (MAPE)



Experiment 3_2- Comparing DTLearner and RTLearner using Coefficient of Determination (R-Squared)

4 CONCLUSION

In conclusion, leaf size, bag size greatly impact a learner's predictive performance. With smaller leaf size, the learner is more prone to overfitting, but this can be reduced by introducing bagging. Increasing the number of bags used will further improve model performance by reducing noise and variance, but increases the complexity and time it takes to run the model. I hypothesized that DTLearner will perform better than RTLearner, but that is not always the case as shown in experiment 3. To further evaluate the learners, I will have to use another method of feature selection such as entropy or gini index or apply boosting in addition to bagging.

5 REFERENCES

1. Learning, M. (1997). Tom mitchell. *Publisher: McGraw Hill*.
2. Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*, 81-106.
3. Witten, D., & James, G. (2013). *An introduction to statistical learning with applications in R*. springer publication.
4. Zacharis, N. Z. (2018). Classification and regression trees (CART) for predictive modeling in blended learning. *IJ Intelligent Systems and Applications*, *3*, 1-9.