

Sergiy Palguyev

GTID: spalguyev3

Machine Learning for Trading Assignment 3: Assess Learners

Question 1: Does overfitting occur with respect to leaf_size? Consider the dataset istanbul.csv with DTLearner. For which values of leaf_size does overfitting occur? Use RMSE as your metric for assessing overfitting. Support your assertion with graphs/charts. (Don't use bagging).

Answer: Over-fitting would occur at a point where in-sample error is decreasing while out-of-sample error is increasing. In Figure 1 below, leaf size was varied from 1 to 50. We can see that in the region of leaf-size 1-10, the exact behavior of over-fitting is occurring. This makes sense as for low leaf-sizes, the In-Sample RMSE would trend toward zero as the learner will use every training point to learn. This of course would not help predict Out-Of-Sample as seen by the RMSE increase for small leaf-size.

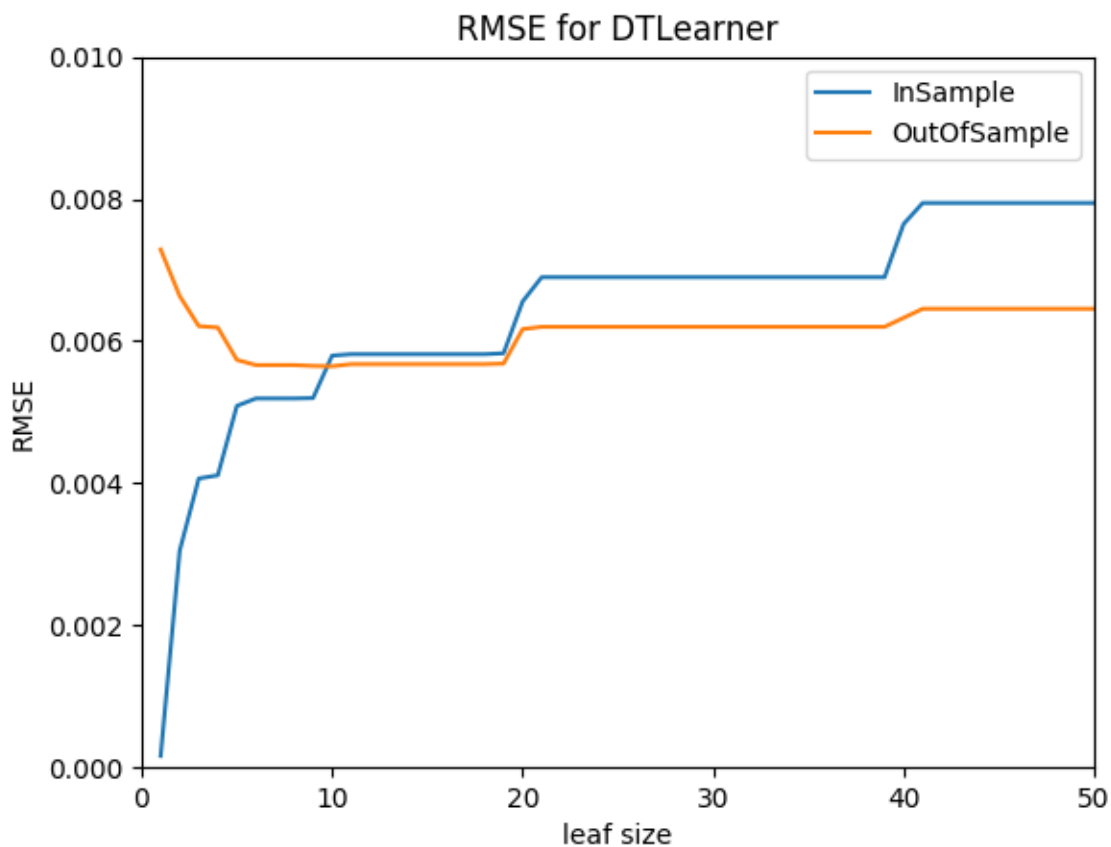


Figure 1: Overfitting with DTLearner

Question 2: Can bagging reduce or eliminate overfitting with respect to leaf_size? Again consider the dataset istanbul.csv with DTLearner. To investigate this, choose a fixed number of bags to use and vary leaf_size to evaluate. Provide charts to validate your conclusions. Use RMSE as your metric.

Answer: As seen by the charts below, bagging can reduce overfitting, but it cannot completely eliminate it. As we use more and more bags, the RMSE for both in-sample and out-of-sample is lowered consistently for all leaf sizes. The extremes of overfitting for leaf-size = 1 is also significantly reduced compared to simply using DTLearner by itself. However the general behavior of overfitting as seen in leaf-size < 10 is not eliminated and persists even as bagging is quadrupled from 4 bags up to 256 bags as seen in the Figures 2-5 below. It is important to notice that the RMSE variation in the Figures is smoothed as the number of bags is increased. The more bags used, the less noise is introduced to the random choosing of values during bagging.

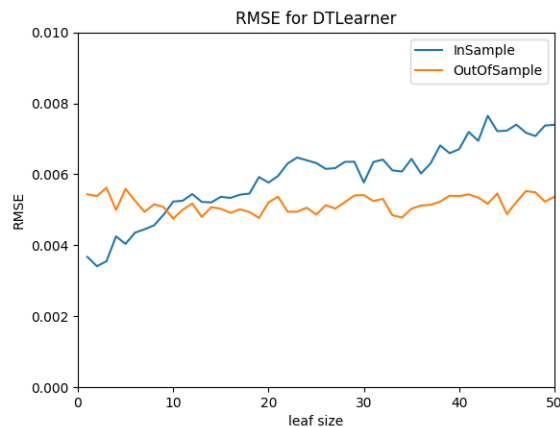


Figure 2: Bagging (bags = 4) with DTLearner

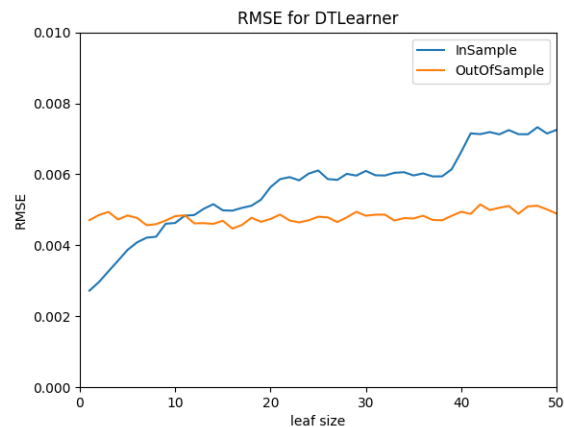


Figure 3: Bagging (bags = 16) with DTLearner

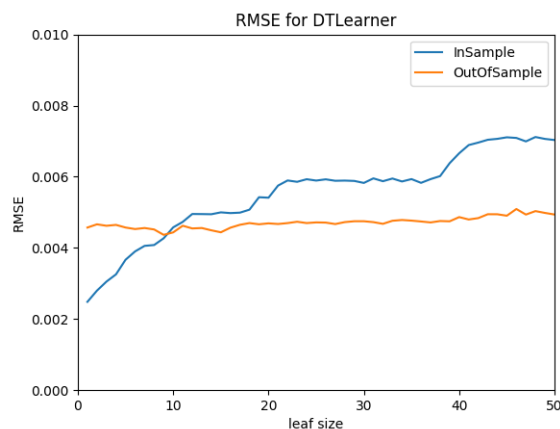


Figure 4: Bagging (bags = 64) with DTLearner

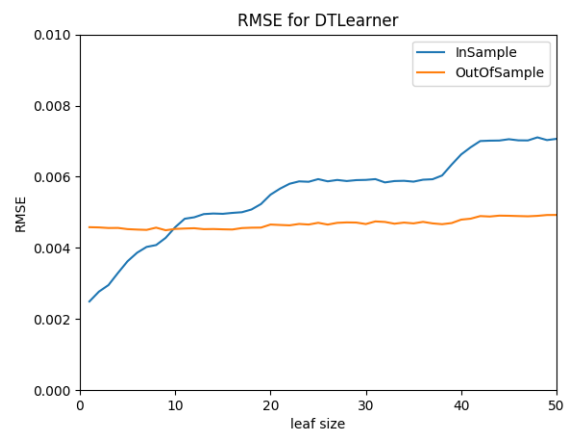


Figure 5: Bagging (bags = 256) with DTLearner

Question 3: Quantitatively compare "classic" decision trees (DTLearner) versus random trees (RTLearner). In which ways is one method better than the other? Provide at least two quantitative measures. Note that for this part of the report you must conduct new experiments, don't use the results of the experiments above for this.

Answer: From quantitative analysis, it seems that DTLearner is better from the perspective of variability in RMSE. DTLearner is more stable and consistent as the leaf_size increases. However, RTLearner overfitting behavior stops on smaller leaf_sizes as seen in Figure 6. Additionally, RTLearner is much quicker to train as demonstrated by Figure 7 where a Bag Learner of RTLearners with 20 bags is implemented and compared to a BagLearner of 20 bags with DTLearner by timing the learning time. We can see that in complex scenarios such as this, RTLearning is quicker. Overall, RTlearning allows for quicker learning and smaller leaf_sizes while introducing some noise in the RMSE values.

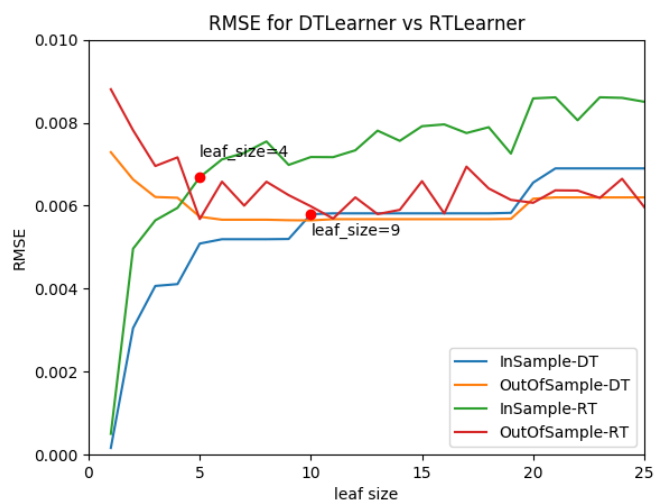


Figure 6: Overfitting DTLearner vs RTLearner

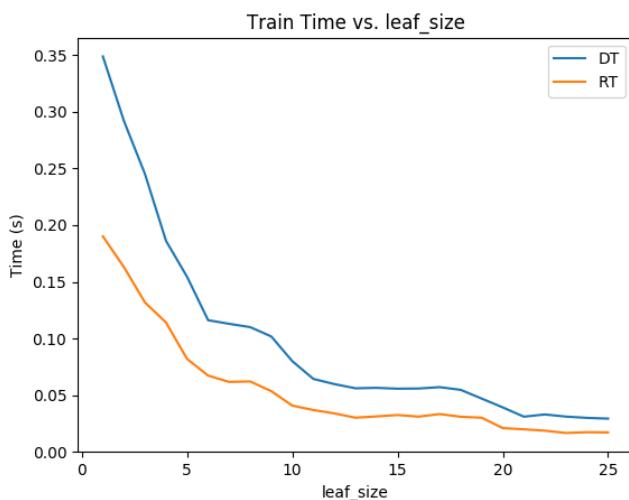


Figure 7: BagLearner training time DTLearner vs RTLearner