Machine Learning for Trading

GtID: xhan306
Name: Xi Han

1. Overfitting Vs. Leaf_Size

When machine learning models fit limiting data points too much, overfitting occurs and the model is very complex. It is a very common problem in data mining. It may reduce error of model for training data but cause large error for training data.

For the first question, the correlation between overfitting and leaf_size in Decision Tree learner (DTLearner) is investigated. RMSE (root mean square error) is used as measurement metric to measure overfitting. Overfitting starts to occur when RMSE of Testing data (out-of-sample Data, yellow line) is higher than RMSE of training data (In-sample Data, blue line). It can be observed from Figure 1 that when leaf_size is reduced to less than 10, overfitting starts to occur.

Result:
Overfitting occurs with respect to leaf_size

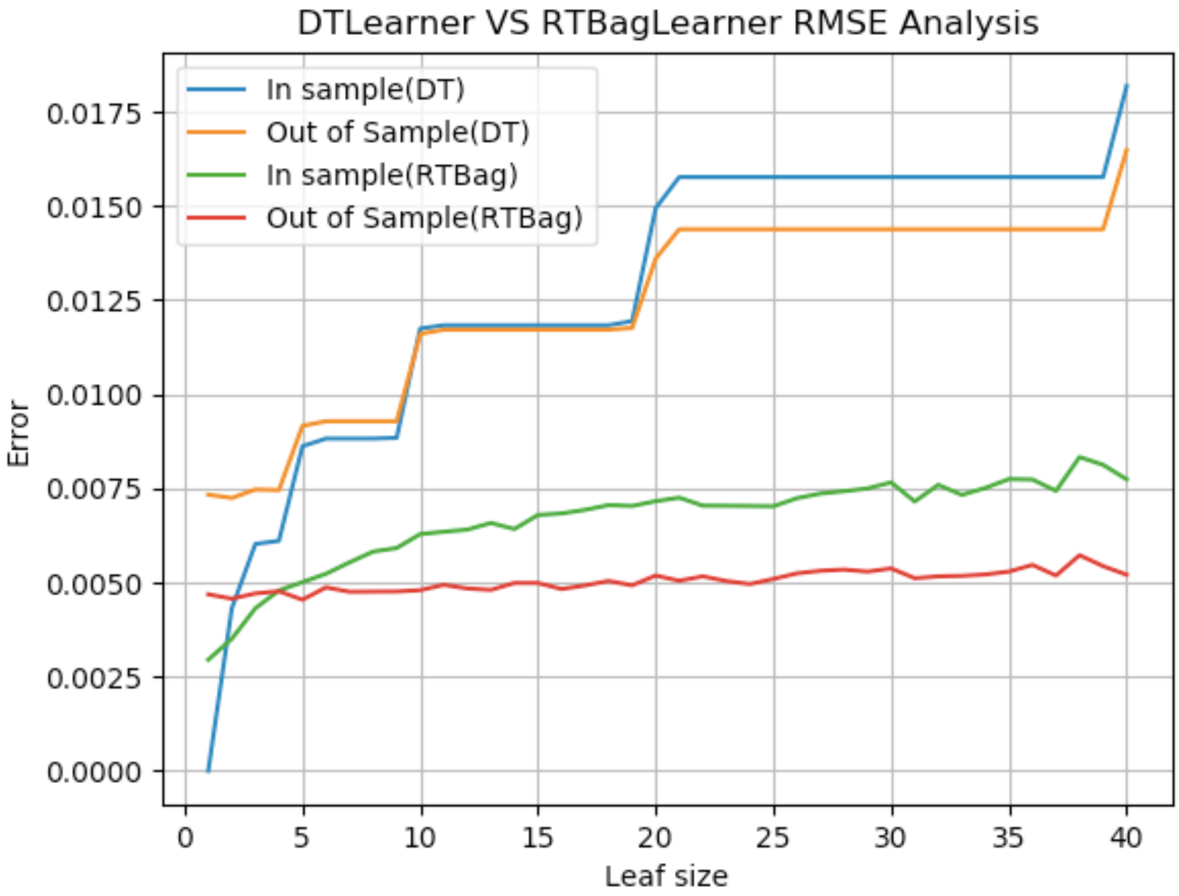

Figure 1. Assess overfitting for DTLearner by testing RMSE of In-sample and out-of-sample (Leaf_size range from 1-40).

2. Bagging to overfitting

One possible method to avoid overfitting of Decision tree is to apply bagging to the model. First, the plot of BagLearner is made to explore whether bagging can reduce or eliminate overfitting with respect to leaf_size in Decision Tree model.

Firstly, I used a fixed number of bags 20 and change the leaf_size to see whether bagging can reduce overfitting. For comparison, we plotted the correlation of RMSE of two learners (DTLearner and BagLearner) with leaf_size at the same figure. It can be shown in Figure 1 that when the leaf_size is lower than 4, overfitting starts to happen on BagLearner, while DTLearner has overfitting when leaf_size is lower than 10. Compared with larger leaf_size, the effect of bagging to reduce overfitting was more obvious.

Result:
Bagging can reduce or even eliminate overfitting with respect to leaf_size.

3. DTLeaner Vs. RTLearner

Decision tree choose the best feature to split on which has the highest correlation with target value, while RTLearner select the feature to split on randomly. It is easy to know the randomness of RTLearner will enable it higher calculation speed but low stability. RMSE was calculated for two learners in the same range of leaf_size(1-40).

Figure 2 clearly indicated that the overall performance of decision tree and random tree was very similar. RMSE of both were in the same range (0-0.02) for training data (In-sample data, green line for DT and blue line for RT), and around 0.008-0.02 for testing data (out-of-sample data, red line for DT and yellow line for RT). Moreover, they have the same trend. Overfitting will occur for both learners. Although a random tree learner is faster in speed, while fluctuated lines in graph reflected the randomness and instability of RTLearner.

Compared to DTLearner, RTLearner is more likely to overfit for smaller leaf sizes. For example, at leaf_size = 30, RTLearner overfits but dTLearner does not overfit.

Then I plotted a RTlearner with bagging to compare with a decision tree learner with bagging (number of bag=20), the RMSE of both training and testing data for DTLearner were smaller than in RTLearner (Figure 3).

Result:
As a single learner, a random tree is faster than a decision tree, but it is less stable than a decision tree. With bagging, decision tree learner can seems better than random tree learner when the leaf_size is low but similar when leaf_size increase.

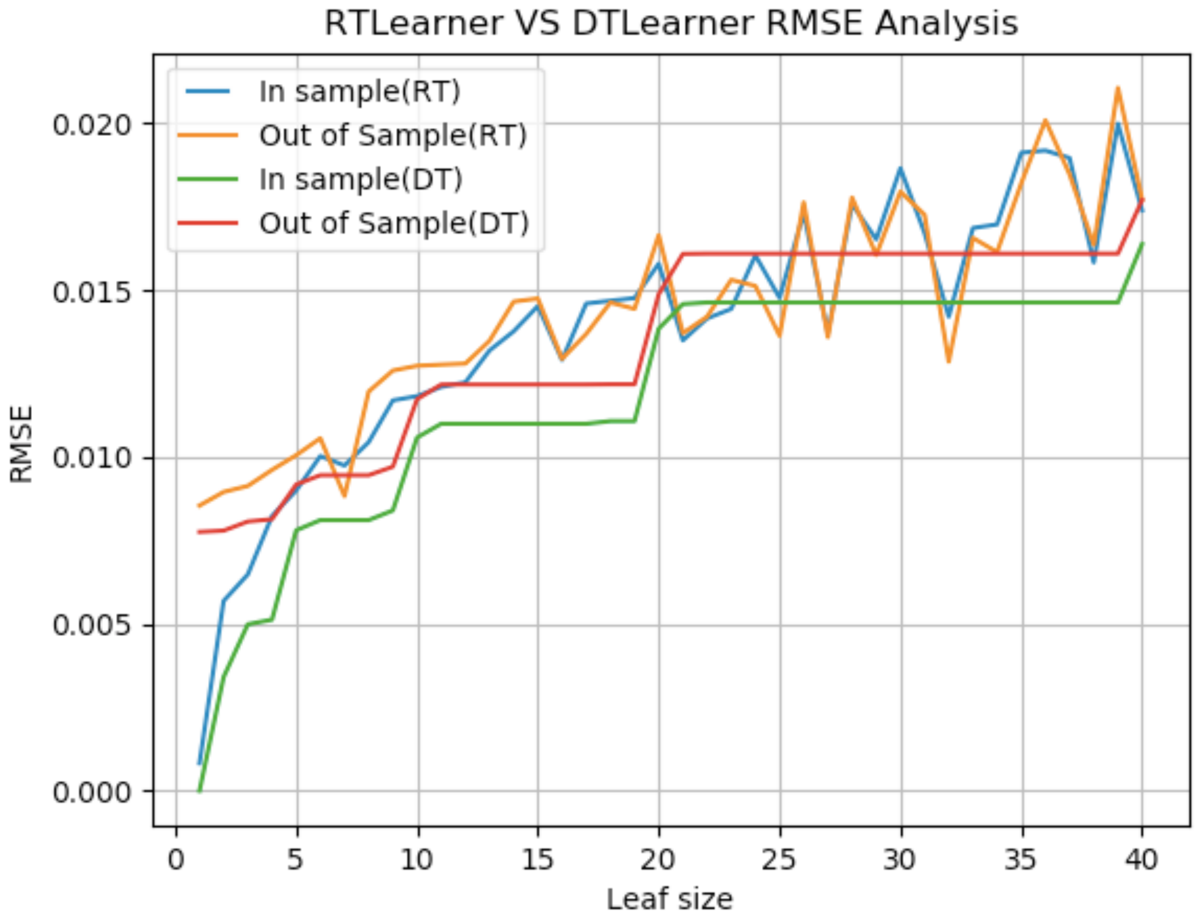

Figure 2. RMSE of In-sample and out-of-sample for DTLearner and RTLearner (Leaf_size range from 1-40).

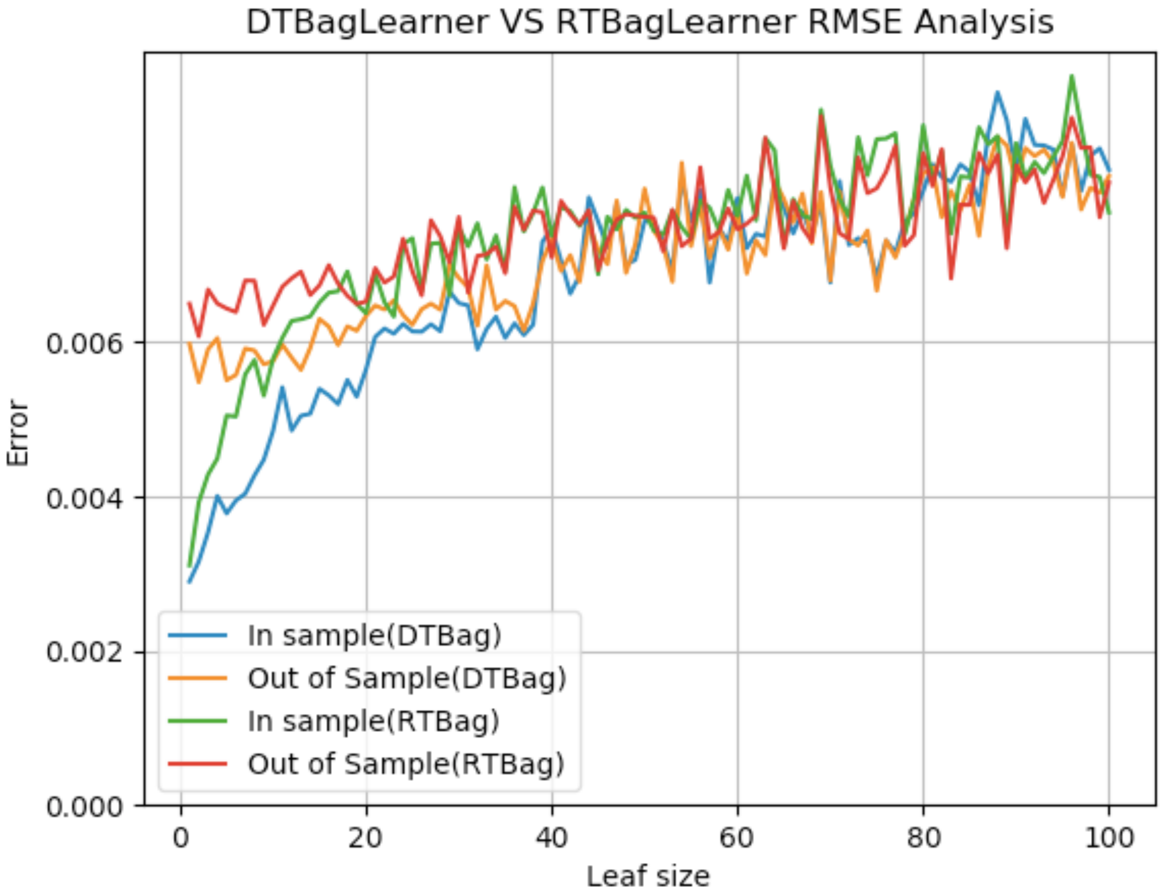

Figure 3. RMSE of In-sample and out-of-sample for DTBagLearner and RTBagLearner (Leaf_size range from 1-40).