CS-7646 Machine Learning for Trading

Report- Assess_Learners

Name: Shanshan Wang
GtID: swang637
Date: 02-12-2018

1. Overfitting Vs. Leaf_Size

Overfitting is a very common issue in machine learning and data science. Sometimes our model doesn't generalize well from our training data to testing or validating data.

In this section, I studied if overfitting occurs with respect to leaf_size in Decision Tree learner (DTLearner).

I used RMSE (root mean square error) as my metric for assessing overfitting. Overfitting occurs when RMSE of Testing data (out-of-sample Data, green line) starts to increase while RMSE of training data (In-sample Data, blue line) keep decreasing.

From Figure 1, we can see when leaf_size reduced to less than 10, overfitting occurred. We zooned in this region (leaf_size 1-10, in Figure 2.) and pinned down the extract leaf_size when overfitting happened. With leaf_size less than 5, overfitting occurred within this model.

Conclusion: Overfitting occurs with respect to leaf_size.

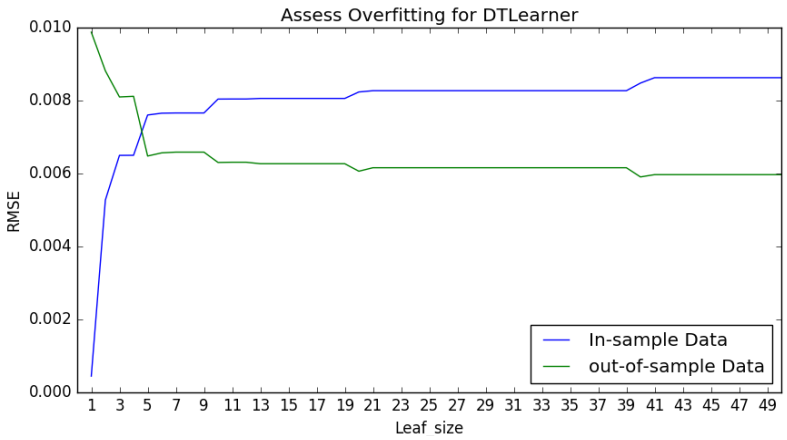

Figure 1. Assess overfitting for DTLearner by testing RMSE of In-sample and out-of-sample. Leaf_size range from 1-50.

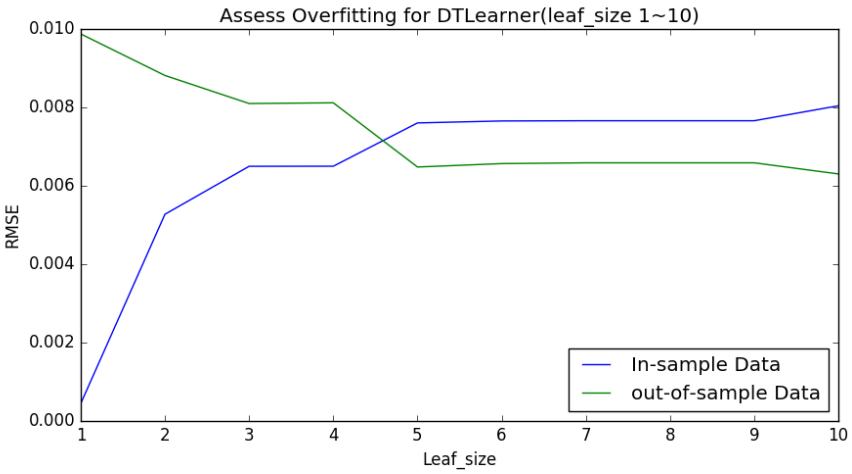

Figure 2. Assess overfitting for DTLearner by testing RMSE of In-sample and out-of-sample. Leaf_size range from 1-10.

2. The effect of bagging to overfitting

To avoid overfitting of Decision tree, one way can go is applying bagging to the model. Here I plotted BagLearner to see if bagging can reduce or eliminate overfitting with respect to leaf_size in Decision Tree model.

Firstly I used a fixed leaf_size of 5, since from the above result I knew that the overfitting occurred when leaf_size was 5. The result shown in Figure 3 was very different from previous result. There was no obvious overfitting pattern. So I narrowed down the bag_size to 1-20, still there was no overfitting pattern and the RMSE for training data and testing data were very close.

Secondly I tried the model with different leaf_size to see if the effect of bagging to overfitting was also related to leaf_size. According to Figure 4, with larger leaf_size, the effect of bagging to reduce overfitting was more obvious. Although when leaf_size was 10, the RMSE of training data was slightly smaller than in leaf_size 20.

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

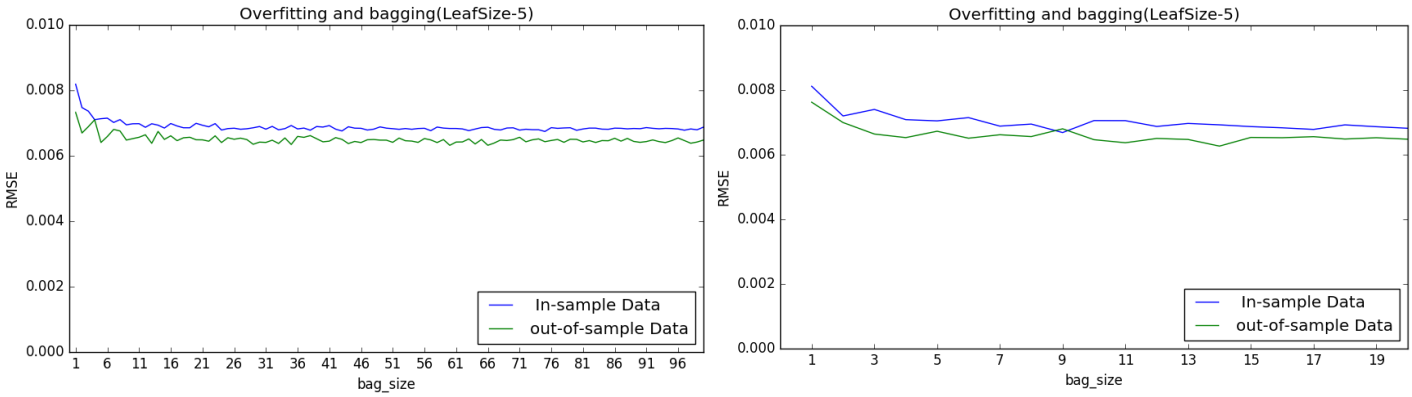

Figure 3. Assess overfitting for DTLearner with Bagging. Left: bag_size 1-100, right:bag_size 1-20. Leaf_size is fixed at 5.

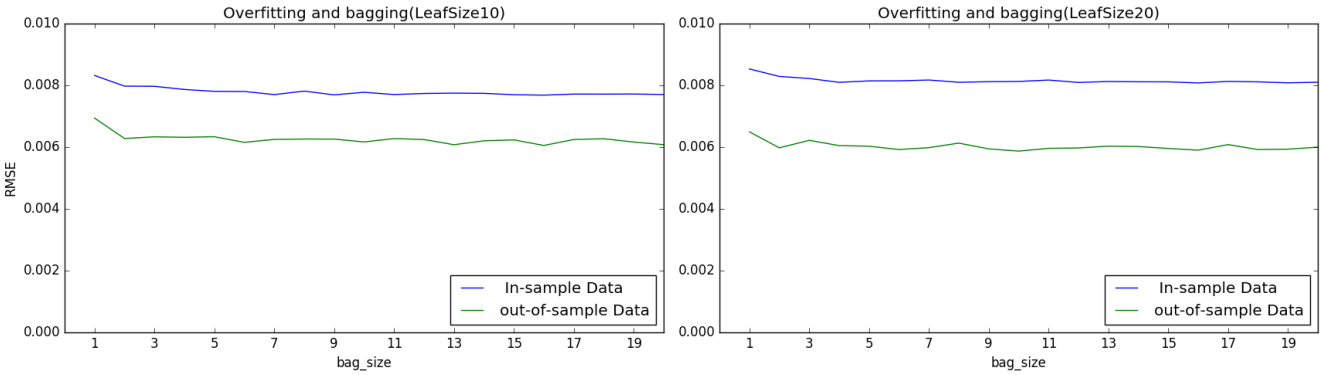

Figure 4. Assess overfitting for DTLearner with Bagging. Bag_size 1-20. Leaf_size is fixed at 10 in left and 20 in right.

3. DTLeaner Vs. RTLearner

In order to quantitatively compare "classic" decision trees (DTLearner) versus random trees (RTLearner), I firstly plotted these two learners in the same scale of leaf_size(1-50). Figure 5 clearly indicated that comparing a single decision tree to a random tree, the overall performance was the same. RMSE of both were in the same range, around 0.008 for training data (In-sample data, red line for DT and blue line for RT), and around 0.006 for testing data (out-of-sample data, cyan line for DT and green line for RT). In spite of the speed, a random tree learner was much more unstable than a decision tree learner, indicating by the fluctuated lines in graph, which reflected the randomness.

Then I used a RTlearner with bagging to compare with a decision tree learner, although with a cost of running time, bagged RTlearner didn't show an obvious overfitting, and also the RMSE of both training and testing data were smaller than in DTLearner(Figure 5).

Conclusion: As a single learner, a random tree is much faster than a decision tree, but is also less stable than a decision tree. With bagging, random tree learner can beat decision tree learner both in accuracy and speed.

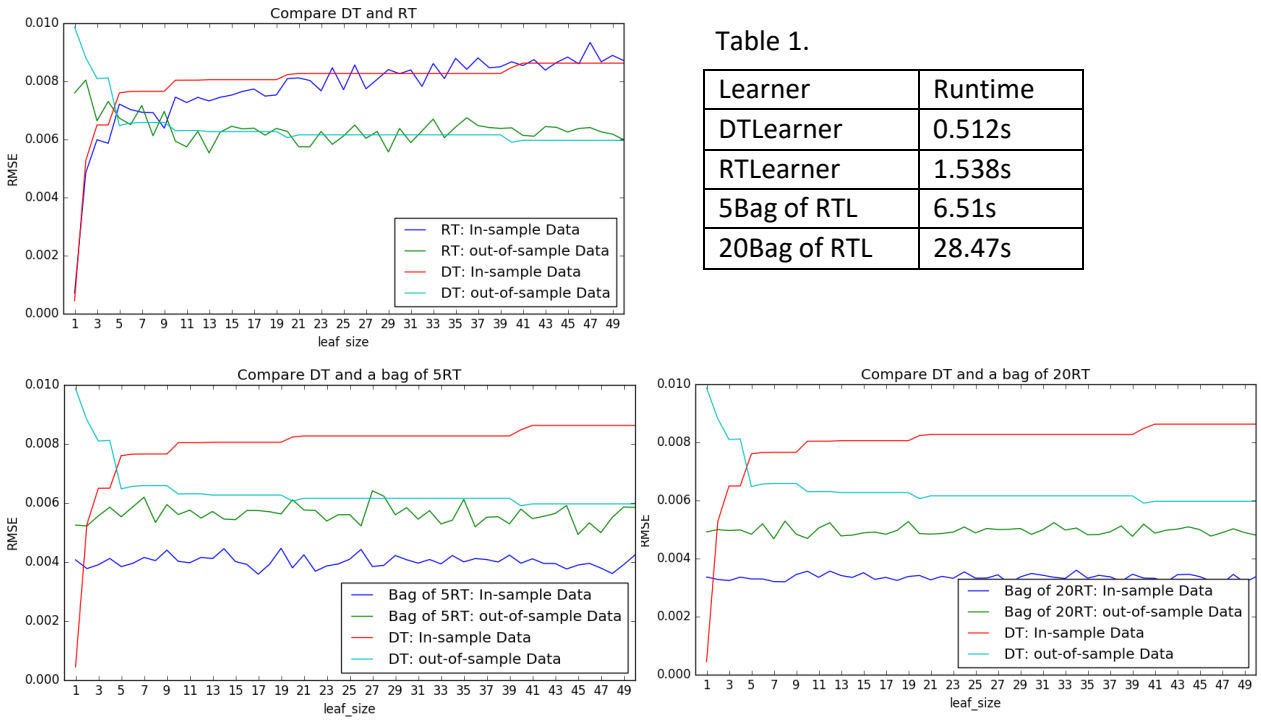

Table 1.

Learner	Runtime
DTLearner	0.512s
RTLearner	1.538s
5Bag of RTL	6.51s
20Bag of RTL	28.47s

Figure 5. Comparison of DTLearner and RTLearner