Classification trees, bagging and random forest

Maruša Oražem[1]

[1] *mo3757@student.uni-lj.si, 63200439*

Introduction

For this homework I have implemented 3 classes, those are **Tree**, **Bagging** and **RandomForest**, that all have method **build**, which returns the model as an object. Classes also have **predict** method that predict target class of given input samples. Quick overview of input attributes for following classes:

1. **Tree**
   - ***rand*** *- random generator of type random.Random*
   - ***get_candidate_columns*** *- a function that returns a list of column indices considered for a split*
   - ***min_samples*** *- the minimum number of samples, where a node is still split*
2. **Bagging**
   - ***rand*** *- random generator of type random.Random*
   - ***tree_builder*** *- an instance of Tree used internally*
   - ***n*** *- number of bootstrap samples*
3. **RandomForest**
   - ***rand*** *- random generator of type random.Random*
   - ***n*** *- number of bootstrap samples*
   - ***min_samples*** *- the minimum number of samples, where a node is still split*

In above classes, I have used the Gini index for selecting the best split.

Results

I have applied the developed methods to housing3.csv data set. I have used the first 80% of data for the training set and remaining 20% for the testing set.

Misclassification rates from hw_tree_full

In function *hw_tree_full* I have build a tree with $min\_samples = 2$ and the misclassification rates I got are:

- train set: 0.0025
- test set: 0.1600

Misclassification rates from hw_cv_min_samples

In function *hw_cv_min_samples* I have found the best value for *min_samples* with 5-fold cross-validation on training data. The result is $min\_samples = 4$. Misclassification rates for $min\_samples = 4$ are:

- train set: 0.0175
- test set: 0.1600

On figure 1 we can see how misclassification rates are changing, when changing the values of *min_samples*. Results also change for different seed.

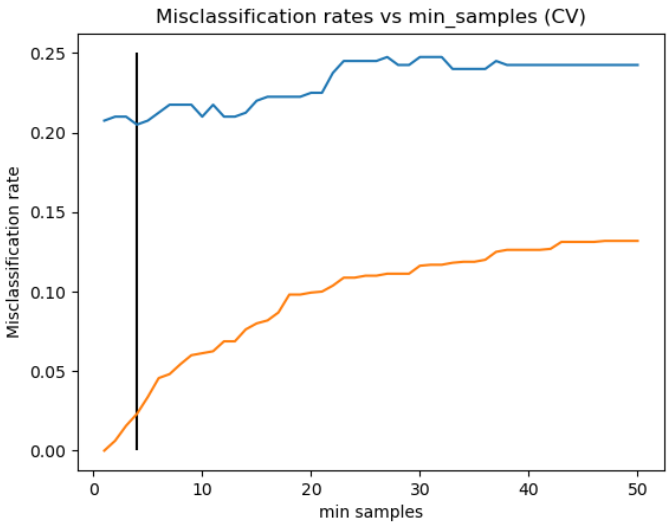

Figure 1. On this figure we can see how misclassification from internal cross-validation is changing, when we change the values of *min_samples*. Minimum misclassification rate on test set is denoted with vertical black line.

Misclassification rates from hw_bagging

In function *hw_bagging* I have set attributes to *n=50* and *min_samples=2*. Results change for different seed. Misclassification rates are:

- train set: 0.0
- test set: 0.109

On figure 2 we can how misclassification rates are changing, when changing number of trees n and also different seeds.

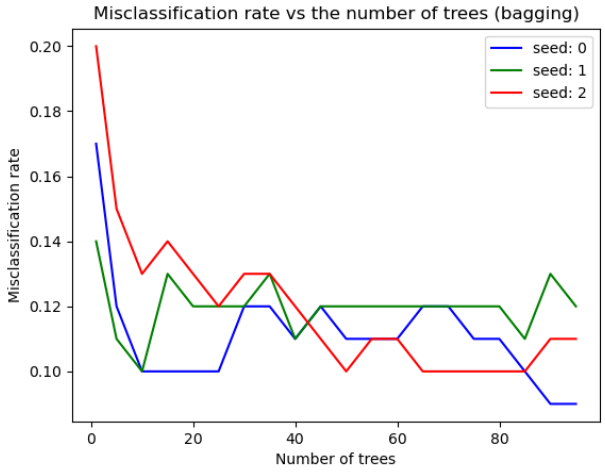

Figure 2. On this figure we can see how misclassification rates change on test set, when we change the number of trees n. Each color shows different seed.

Misclassification rates from hw_randomforest

In function *hw_randomforest* I have set attributes to $n = 50$ and $min\_samples = 2$. Misclassification rates are:

- train set: 0.007
- test set: 0.12

On figure 3 we can how misclassification rates are changing, when changing number of trees n and also different seeds.

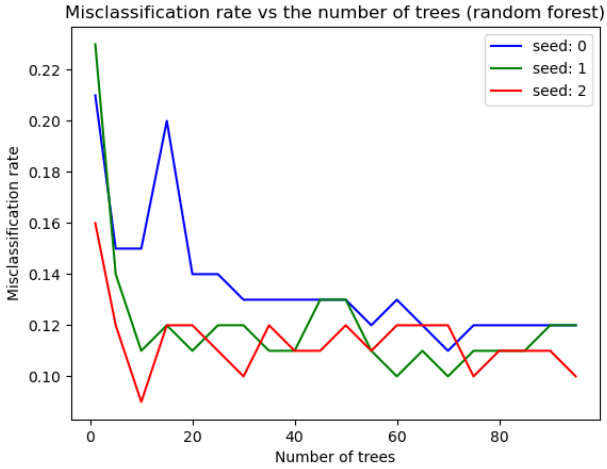

Figure 3. On this figure we can see how misclassification rates change on test set, when we change the number of trees n. Each color shows different seed.