Classification trees, random forests

Matej Miočić, 63180206, mm9520@student.uni-lj.si

I. Introduction

In this homework we were tasked with implementation of classification trees and random forests. All work is implemented in file *hw_tree.py*. Developed methods were tested on FTIR spectral dataset. We provided misclassification rates and standard errors on training and testing data. Based on this dataset we also computed variable importance.

II. Experiments

A. Misclassification rates of classification trees

Our first task was to show misclassification rates and standard errors we obtain when using classification trees on training and test data (see Table I).

Table I
Misclassification rates and their standard errors for train and test dataset with **CLASSIFICATION TREE**.

	Misclassification rate	Standard error
Train	0.00%	0.00%
Test	20.69%	40.51%

We show that classification trees performs with no error on training set, but is misclassifying 20.69% of the test set. This could be an indication of overfitting. Standard error is computed as standard deviation of predicted values from classification rate.

B. Misclassification rates of random forests

We also show misclassification rates and standard errors on the same dataset with random forest (see Table II).

Table II
Misclassification rates and their standard errors for train and test dataset with **RANDOM FOREST**.

	Misclassification rate	Standard error
Train	0.00%	0.00%
Test	1.73%	13.02%

When comparing results with classification trees, we can observe a decrease in mislassification rate on the test set when using random forests. The model is not overfitting anymore and is performing with almost no error. Standard error is computed as standard deviation of predicted values from classification rate.

C. Misclassification rates versus the number of trees

In this section we show how number of trees impacts misclassification rate when using random forests on provided dataset (see Figure 1).

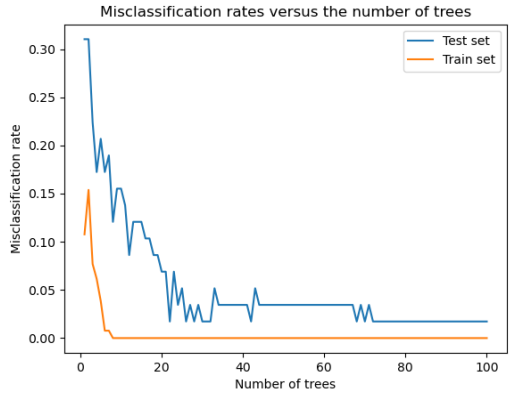

Figure 1. We see a fast decline in misclassification rate when increasing the number of trees.

D. Variable importance

In this section we show variable importance for the given dataset. It is computed as the average of differences in misclassification rates between unshufled out-of-bag data and each randomly permuted variable in the same out-of-bag dataset for every tree in random forest. We also show variables from the roots of 100 non-random trees (see Figure 2).

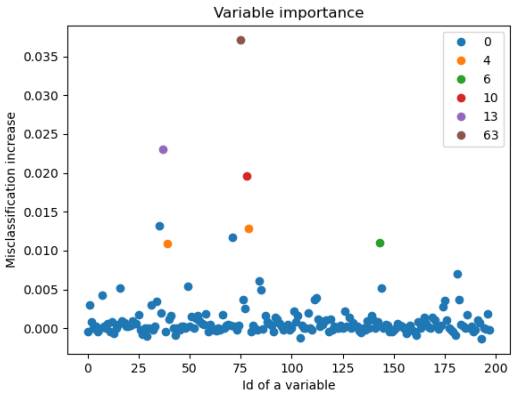

Figure 2. In this figure we show how variable importance in random forest correlates with the decision of a root variable in 100 non-random, bootstrapped trees. Colour indicates the number of times a variable was chosen for the root node out of 100 trees.

III. Conclusion

In this homework we learned how to implement trees and random forests. We showed that random trees decrease misclassification rate since they reduce overfitting compared to trees at the cost of process speed.