HW1: Classification trees, random forests

Žan Pušenjak
MLDS 2024/25 , FRI, UL
zp4613@student.uni-lj.si

***Abstract*—We implemented decision tree and random forest classification models and tested them on TKI resistance FTIR spectral data set to observe the performance of both models.**

I. Decision Tree

We decided to build full trees, so a node was split further until we got a pure node. The splitting feature was selected using the Gini impurity.

II. Random forest

We further implemented the random forests using our tree model described in the first section. Each forest consisted of n number of trees. To further increase the randomness we only considered $\sqrt{k}$ features picked at random for each split where k was the total number of features.

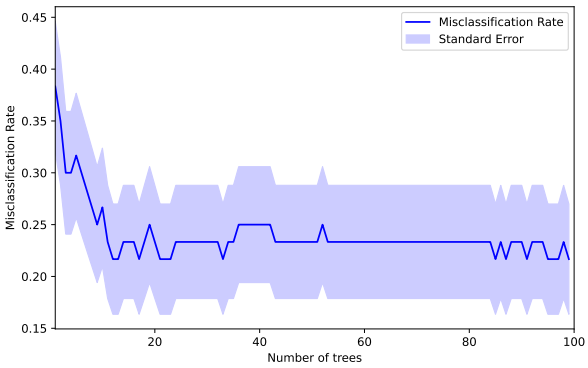

Fig. 1. Example of an binarized image after zero crossing detection.

III. Results

A. Training times

The training times for TKI resistance FTIR spectral data set can be observed in table I

TABLE I
Training times for both models using the TKI resistance FTIR spectral data set.

Model	Training time in seconds
Decision tree	3.14
Random forest	10.18

B. Model evaluation

To test the model performance we looked at the misclassification rate (denoted MCR) on the predictions of the test data of our dataset. We quantified the uncertainty using the formula

$$\frac{MCR(1-MCR)}{n}$$

where n is the total number of predictions. Final model evaluations are seen in Table II

TABLE II
Misclassification rates and their quantified uncertainties for final model predictions on test and train data.

Model	Test data	Train data
Decision tree	0.316 ±0.06	0.0 ±0.0
Random forest (100 trees)	0.23 ±0.05	0.0 ±0.0

We explored how changing the number of trees in a random forest affects misclassification rates (Figure 1).

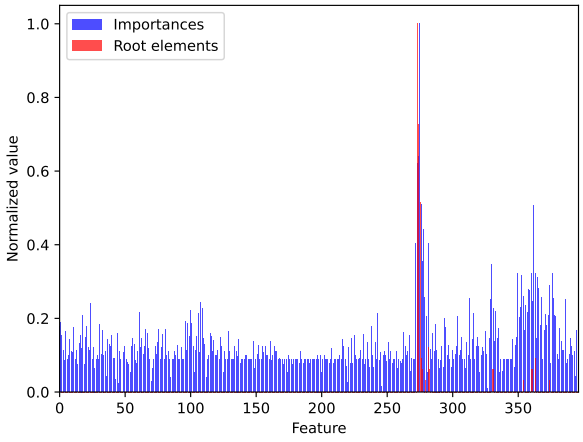

Fig. 2. Normalized feature importances compared to the occurrences of features in the root nodes of 100 trees built with bootstrapped data.

C. Feature importance

We calculated the feature importance in random trees. We used out-of-bag data to calculate the importances for each of the trees in the forest then summed them for all the trees. We built hundred full trees with bootstrapped data

and compared the number of occurrences of each feature in the root of the tree to the importance of the feature. (Figure 2).

Additionally, we tried computing the importance of each 3-tuple of features using 1000 tree random forests, but the computation took too long, indicating inefficiencies in the implementation that were sadly not improved further.