Homework 1 report

Our test set consists of IID observations, and each of our observations produces a misclassification error. We are interested in the mean of misclassification distribution, and clearly the best estimator would be the sample average. We can use various techniques to quantify the uncertainty of the sample average, here we will construct a 95% confidence interval (using the fact that the sample average is asymptotically normal by the **Central Limit Theorem**).

Model	**MC rate**	**SE**	**MC Quantification**
Decision Tree	0.2069	0.0532	0.2069 ± 0.1043
Random Forest	0.0172	0.0171	0.0172 ± 0.0335

Table I

Results obtained on the given dataset. For both models, MC rate on the training set was 0.

Now let us discuss how the number of trees affects the RF model. Regarding that, we have the following figure:

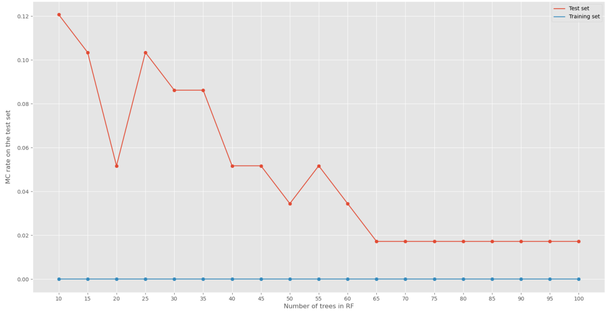

Fig. 1. There are 2 interesting things to observe on this figure. The first one is that, regardless of the number of trees, the model always fully adapted to the training set. The second one is the region $[20,65]$, where we have oscillations in the misclassification rate, which is counter-intuitive, since we would expect that the predictive power of the model increases as the number of trees increases.

Finally, we implemented the feature importance algorithm, as described in the paper by Breiman. There are 198 features present in the dataset, and after running the algorithm we get 0 importance for 168 ($\approx$ 85%) features, which we excluded from the figure that follows. Within remaining features, we identified 4 different groups of features that had the same importance. For transparency, we united these features to 4 groups to get the following figure:

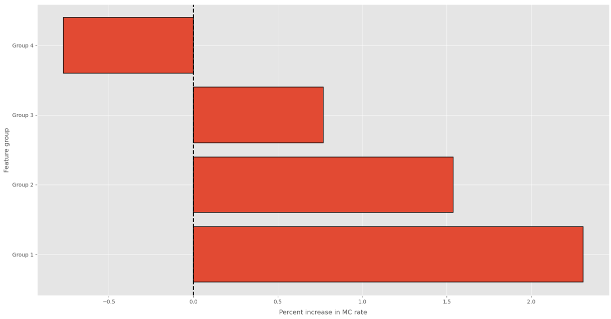

Fig. 2. 4 groups of same feature importance consist of following features:

- Group 1: [1148, 1284, 1340]
- Group 2: [1156, 1196, 1312, 1724]
- Group 3:[1036, 1068, 1104, 1124, 1188, 1224, 1280, 1336, 1376, 1476, 1496, 1572, 1608, 1696, 1704]
- Group 4: [1140, 1216, 1296, 1316, 1348, 1444, 1700, 1728]

For comparison, we generated 100 bootstrap datasets from the original dataset, and on each of those datasets we constructed a full DT. For each tree we recorded the feature that was in the root to get the resulting distribution:

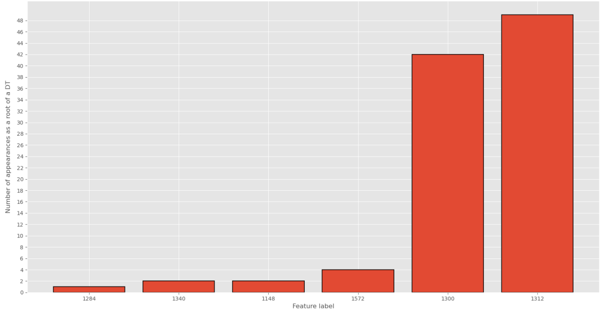

Fig. 3. We can see that the most frequent root feature is 1312, which is also deemed important by Breiman's algorithm. A total of 5 times we got that the feature in the root of the tree had negative Breiman score (refering to features 1284, 1340, 1148).

Results are unexpected. First, we would not expect that so many features have the same importance. We effectivley have 5 groups of equal importance. Estimating feature importance by Breiman's algorithm requires us to permute values of a particular feature in the out of bag dataset, and we would at the end expect at least a slight change in the MC rate. However, MC rate remains unchanged majority of times. We also have a group with negative importance, which raises a question about the nature of features in that group, and answering this question is impossible without the domain knowledge about this dataset, which we do not have.