+FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION
OF HIGHER EDUCATION
ITMO UNIVERSITY

Report
on the practical task No. 1
“Supervised learning algorithms”

Performed by
Chernobrovkin T.V. (412642)
Academic group J4133c
Accepted by
Gladilin P.E.

St. Petersburg
2023

Goal

To study supervised learning algorithms using decisive trees and random forest as examples.

Problem

1. Train 4 different classifiers using sklearn library to predict "Activity" (biological response of the molecule) field from the "bioresponse.csv" dataset:

- small decision tree;
- deep decision tree;
- random forest on small trees;
- random forest on deep trees;

Refer to ‘Random_Forrest.ipynb’ and ‘Decision_Trees.ipynb’ notebooks for examples. Split the data to train and test as 75%/25%.

2. Calculate the following metrics to check the quality of your models:

- precision;
- recall;
- accuracy;
- F1-score;
- log-loss;

3. Plot precision-recall and ROC curves for your models.

4. Train a classifier who avoids Type II (False Negative) errors and calculate metrics from p.2 for it. Recall for it should be not less than 0.95.

Theory

Supervised learning) - one of the methods of machine learning, during which a test system is forced to learn by means of stimulus-response examples. In terms of cybernetics, it is a type of cybernetic experiment. There may be some dependence between inputs and reference outputs (stimulus-response), but it is unknown. Only a finite set of precedents - stimulus-response pairs, called a training sample - is known. On the basis of this data, we need to restore the dependence (build a model of stimulus-response relations suitable for prediction), i.e., build an algorithm capable of producing a sufficiently accurate response for any object. To measure the accuracy of responses, as well as in example-based learning, a quality functional can be introduced.

A classification task is a task in which there is a set of objects (situations) divided into classes in some way. There is a finite set of objects for which it is known to which classes they belong. This set is called a sample. The class belonging of the remaining objects is unknown. We need to construct an algorithm capable of classifying (see below) an arbitrary object from the initial set.

Cross-validation is a procedure of empirical evaluation of generalization ability of algorithms. Cross-validation is used to emulate the presence of a test sample that does not participate in training, but for which the correct answers are known.

A decision tree is a decision support tool used in machine learning, data analysis and statistics. The structure of the tree is represented by "leaves" and "branches". The edges ("branches") of the decision tree contain the attributes on which the target function depends, the "leaves" contain the values of the target function, and the remaining nodes contain the attributes that distinguish cases. To classify a new case, one has to go down the tree to a leaf and output the corresponding value.

Such decision trees are widely used in data mining. The goal is to create a model that predicts the value of a target variable based on several variables in the input.

Each leaf represents the value of the target variable changed as it travels from the root along the edges of the tree to the leaf. Each internal node is mapped to one of the input variables.

Random forest method is a machine learning algorithm that uses an ensemble of decision trees. The algorithm is applied to classification, regression and clustering tasks. The main idea is to use a large ensemble of decision trees, each of which by itself gives a very low classification quality, but due to their large number the result is good.

Materials and methods

In this task, all calculations were performed on the student's personal laptop. The work was performed in the Python programming language.

Results

1. 4 different classifiers were trained using the sklearn library to predict the "Activity" field (biological response of the molecule) from the "bioresponse.csv" dataset. The training and testing data were split as 75%/25%. The code is provided in the appendix.

2. The parameters precision, recall, accuracy, F1-score and log-loss were calculated for small decision tree, deep decision tree, random forest on small trees and random forest on deep trees. The result is presented in Table 1.

Table 1 - Results of model training

	small decision tree	deep decision tree	random forest on small trees	random forest on deep trees
precision	0.79087	0.75719	0.71597	0.82718
recall	0.77757	0.78692	0.79626	0.79626
accuracy	0.75586	0.73454	0.70362	0.78891
F1-score	0.78417	0.77177	0.75398	0.81143
log-loss	8.79957	9.56809	10.68245	7.60836

3. We plotted precision-recall and ROC curves for each model (Figure 1-8).

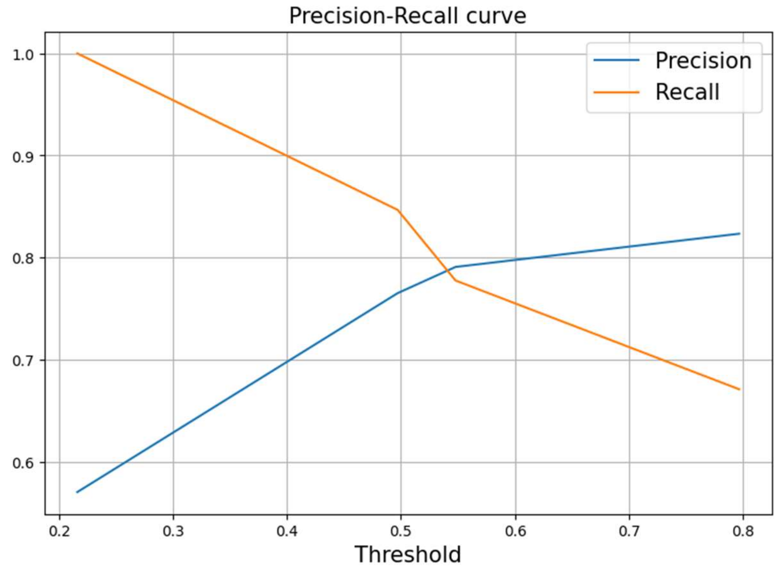

Figure 1 - Precision-recall curve for small decision tree

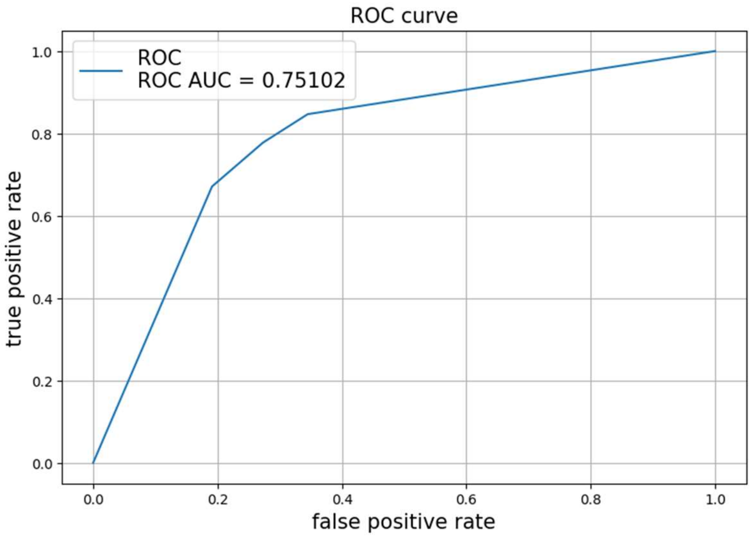

Figure 2 - ROC curve for small decision tree

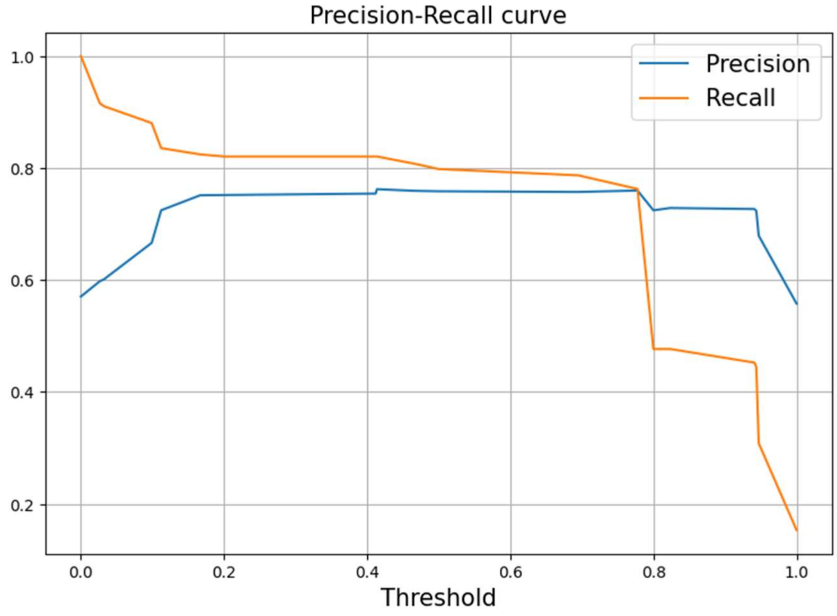

Figure 3 - Precision-recall curve for deep decision tree

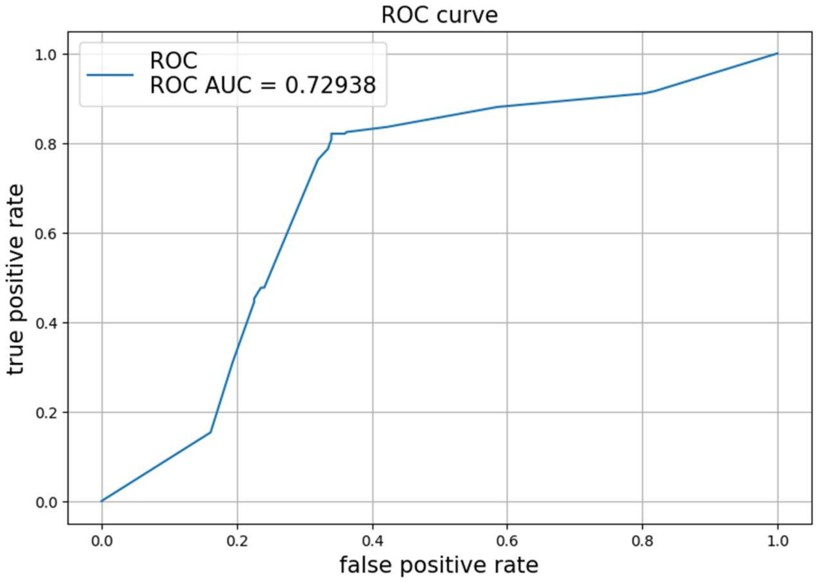

Figure 4 - ROC curve for deep decision tree

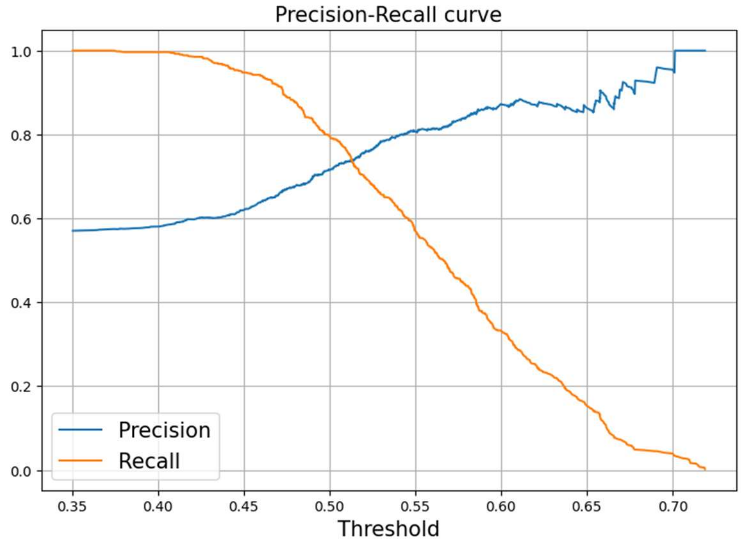

Figure 5 - Precision-recall curve for random forest on small trees

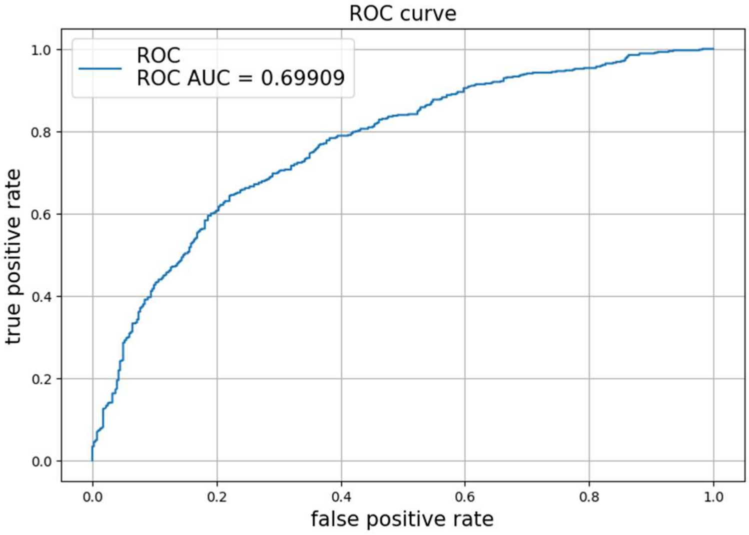

Figure 6 - ROC curve for random forest on small trees

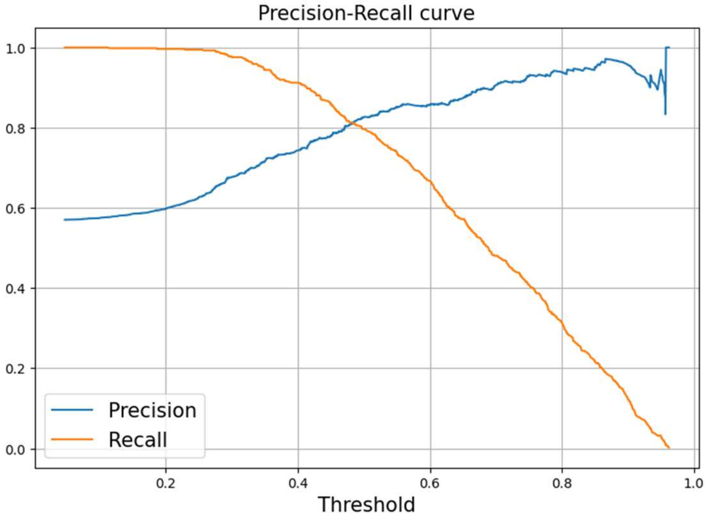

Figure 7 - Precision-recall curve for random forest on deep trees

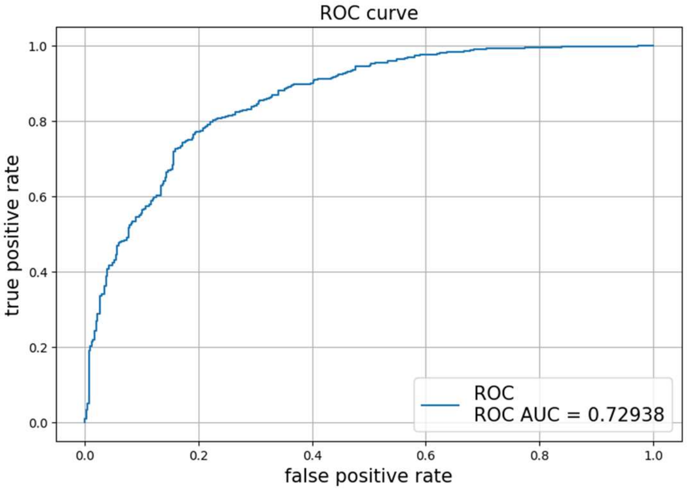

Figure 8 - ROC curve for random forest on small trees

4. Trained a classifier that avoids type II (false negative) errors and compute the metrics from step 2 for it. Recall should be at least 0.95.

By selecting the hyperparameters n_estimators = 91, depth = 1, min_samples_leaf = 1, min_samples_split = 2, bootstrap = False, max_features = 'log2' we achieved recall = 0.98505, accuracy = 0.62473, precision = 0.60505, F1-score = 0.74964, log_loss = 13.52598.

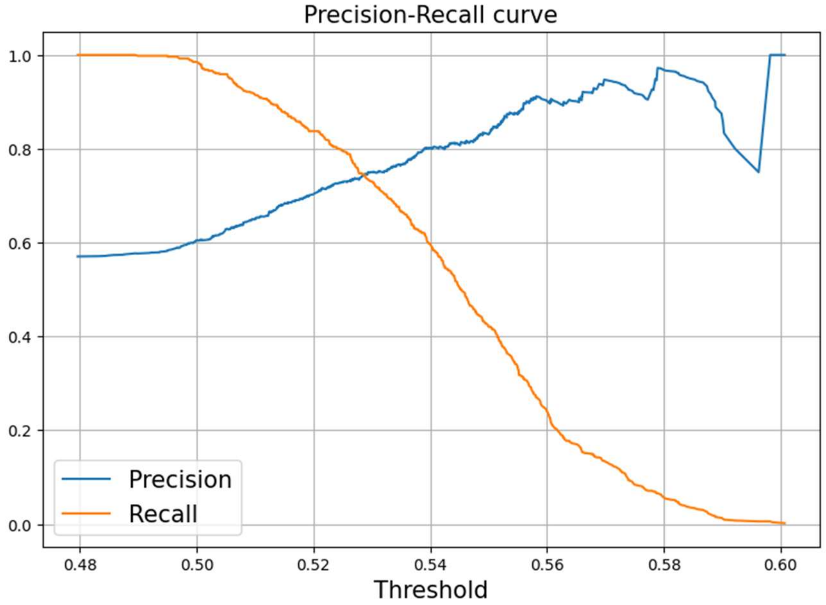

Figure 9 - Precision-recall curve for a classifier avoiding a genus 2 error

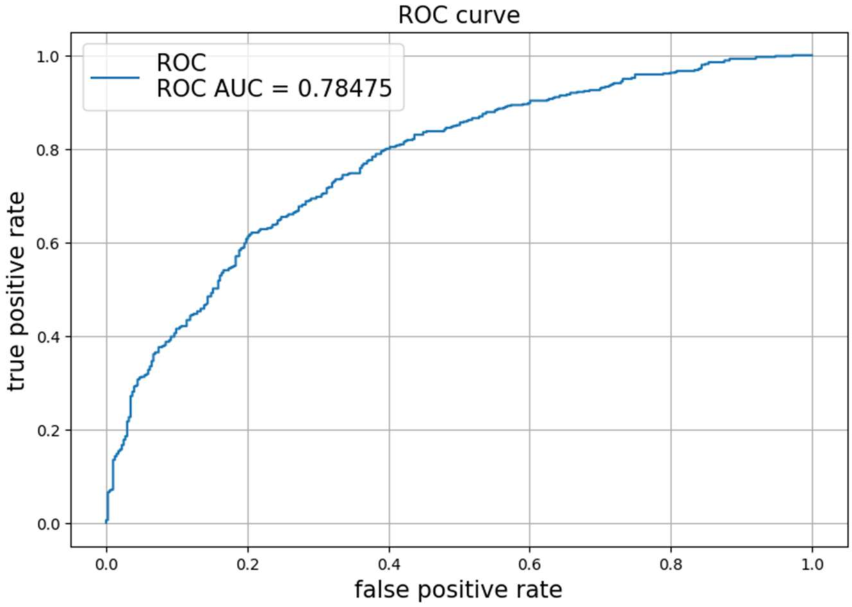

Figure 10 - ROC curve for a classifier avoiding a genus 2 error

Conclusion

In this assignment, aspects of supervised learning algorithms were studied on solving trees and random forest as examples. The architectures of small and deep solver trees and random forests on small and deep solver trees were trained. The parameters accuracy, precision, recall, F1-score and log_loss were derived for all architectures. When these parameters are compared, the random forest on deep trees has the best results. The precision-recall and ROC curve plots were derived for all models.

A classifier model of a classifier avoiding error of the 2nd kind was also trained. For such a classifier the recall value should be greater than 0.95. By selecting hyperparameters we managed to reach the recall value of 0.98505. This means that we managed to train a classifier avoiding Type II errors.

Appendix

GitHub link:

https://github.com/LesostepnoyGnom/homework_ML/blob/main/Task1_Chernobrovkin_Timofei_4126412_J4133c.ipynb