**ALY6020 PREDICTIVE ANALYTICS**

**INSTRUCTOR NAME- DR. MARCO MONTE DE OCA**

REETIKA CHATURVEDI

Northeastern University



Note:

Reetika Chaturvedi, College of Professional Studies, Northeastern University, Boston, MA 02115.

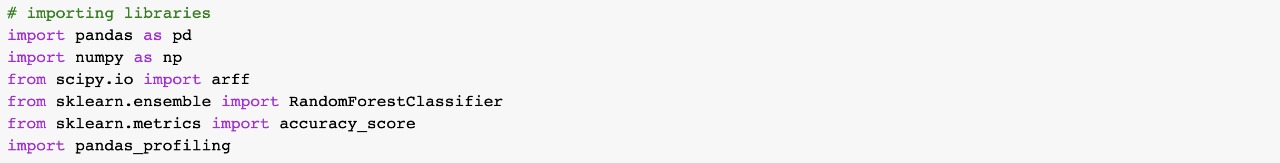
Report based on random forests algorithm

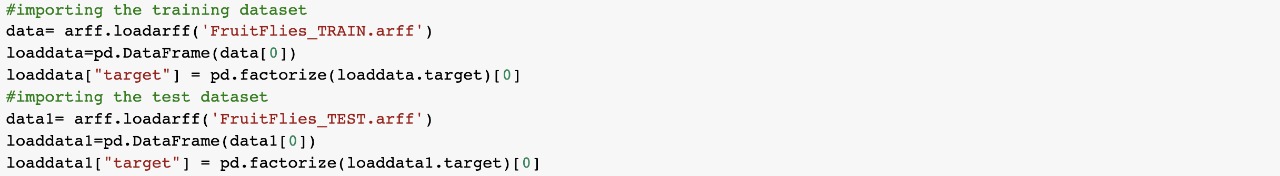
Contact: chaturvedi.re@northeastern.edu

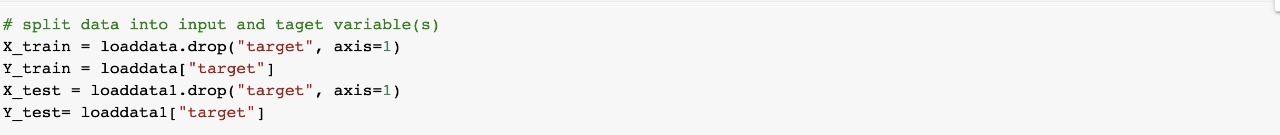
**INTRODUCTION**

In this assignment we used Random Forests which is an ensemble learning method for classification. It is flexible, easy and does not require hyper parameter tuning. Random forests are a collection of individual trees which operate as an ensemble. We constructed decision trees with samples of 5,50 and 500, thereby forecasting the best sample for prediction. The best thing about using random forest is that it does not overfit. The dataset was large but the output was efficient as the algorithm is capable of handling thousands of input variables.

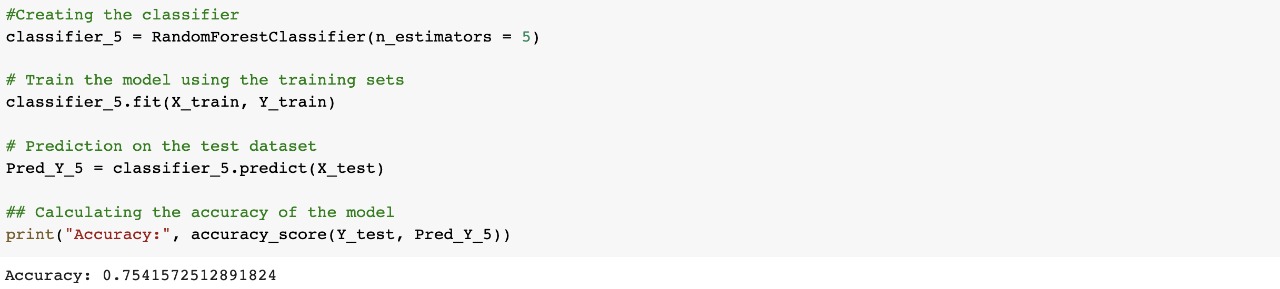
**APPROACH**

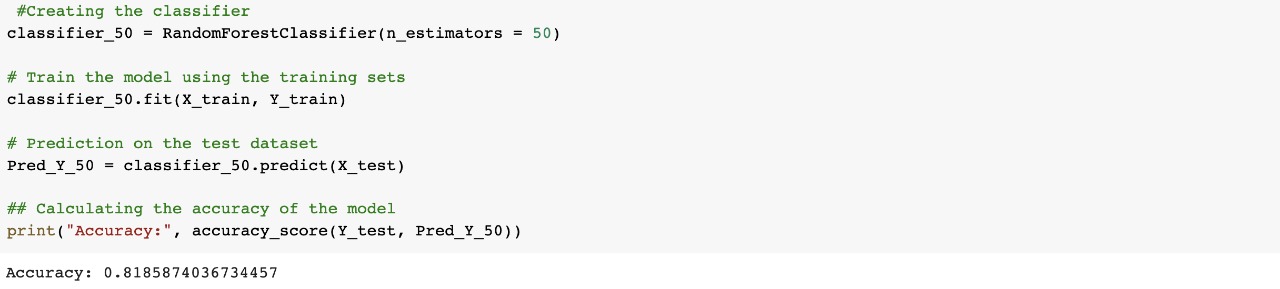
The dataset provided to us was in the arff file format and hence it was imported with the help of scipy.io library. Three species of insects were given and we were asked to classify them based on the reconstructed audio of its wingbeat. To load the data and for modelling, requisite libraries were imported.

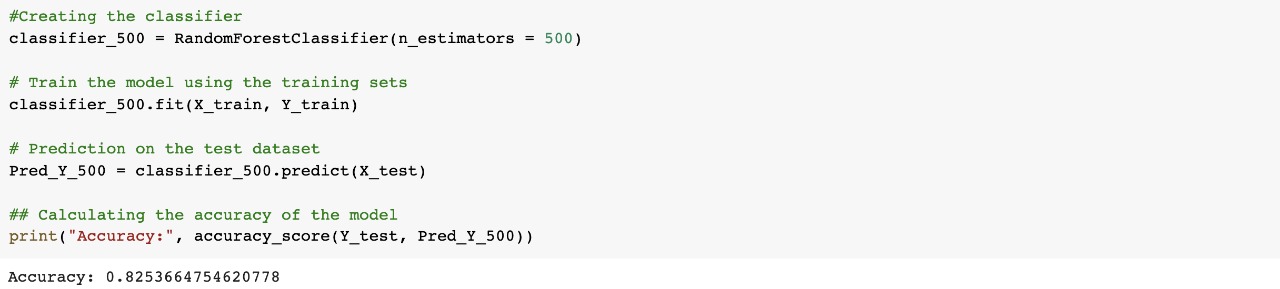
 Since the data was already given as test and train data, the next steps were to import the data.

Once the data was imported and we saw the data heads, the data was divided into input and target variable. X\_train and Y\_train is the division of training dataset as input and target variable. Similarly, Y\_train and Y\_test are also divided.

When the input and target variables were set, next step was to create the classifier. The estimators were taken as 5, 50 and 500. After training the model using training sets, prediction was made on the test data. After this, the model accuracy was computed. The accuracy differs with different n\_estimators.







**ANALYSIS & CNCLUSION**

Considering our dataset, it can be concluded that as we increased the hyperparameter (n\_estimators data), the prediction accuracy increases. By bootstrapping we decreased the variance and maintained the bias by not overfitting. When we used n\_estimator as 5 the accuracy was 75%. The accuracy changed to 81% when the estimators were increased to 50. It further improved by 1% when the estimators were 500.

Random forest classification was the best algorithm for this dataset since we had a lot of predictors for the target variable. With a huge dataset, it is a good idea to build multiple trees and then each tree can be trained on a data subset. It clearly exhibits the power of using more values, as by using more values bias can be reduced. Also, we had categorical value with numerical features which made random forest as one of the best choices.

The best accuracy was when we took 500 n\_estimators.

**REFERENCES**

1. Meinert, R. (2019, June 07). Optimizing hyperparameters in random forest classification. Retrieved February 13, 2021, from <https://towardsdatascience.com/optimizing-hyperparameters-in-random-forest-classification-ec7741f9d3f6>
2. Srivastava, T. (2020, June 26). Random forest parameter tuning: Tuning random forest. Retrieved February 13, 2021, from <https://www.analyticsvidhya.com/blog/2015/06/tuning-random-forest-model/>