# Hackstat 2k19

# HASHTAG

First round report

**Leader**

D.M.A.I.Divisekara

**Members**

K.H.M.K.D.Kehelella

A.P.P.S.Pathirana

T.G.V.S.H.Kumara

**University**

University of Moratuwa

**Registered Team Name**

HASHTAG

**Kaggle User Name**

asithaindrajith

**Display Name**

HASHTAG

## INTRODUCTION

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees. Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

## METHODOLOGY

We approached a trial and error methodology. Following are the supervised learning classification algorithms we analysed using AUC and trial and error method.

- K-Nearest Neighbors
- Support Vector Machines
- Decision Tree Classifiers/Random Forests
- Naive Bayes
- Linear Discriminant Analysis
- Logistic Regression

Logistic regression was our first approach. Accuracy of this classifier was 0.78. Support vector machines classifier was our second approach. Accuracy of this classifier was 0.881. Afterwards by further analyzing the accuracy of classifiers we selected random forest classifier. Initial accuracy of the random forest algorithm was 0.8945. Then we optimized the random forest algorithm using dimensional reduction and one hot encoding techniques.

## RESULTS

The confusion metrics of our last result which was from Random Forest.

```
Number of Training examples 7850
Number of Testing examples 2616
Predicted Revenue      0     1
Actual Revenue
0                    2131    92
1                     149   244
accuracy 0.9078746177370031
```
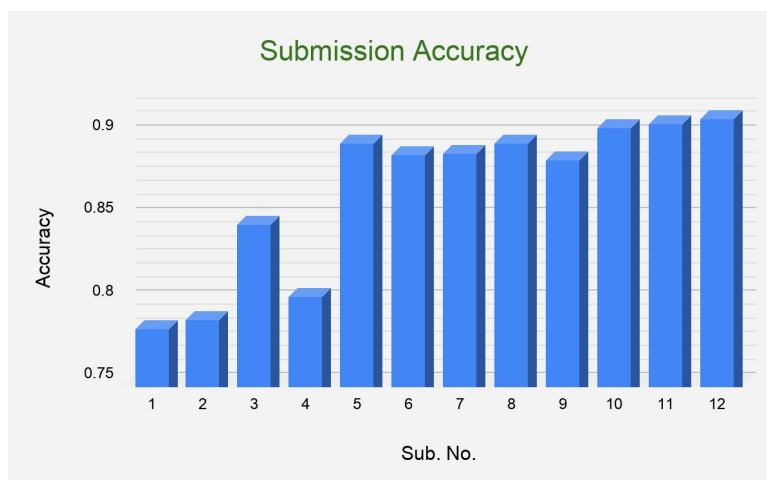


## CONCLUSION

As we have experienced through the first round, no classifier was able to reach upto accuracy of 0.90000 at the initial stage. Then we optimized the accuracy upto 0.90000 by further using hyperparameter tuning and dimensional reduction methods. Mainly the fundamental knowledge about the classifier is essential in order to successfully tune the parameters. Furthermore, we can conclude that the best classifier for this model is the random forest classifier.