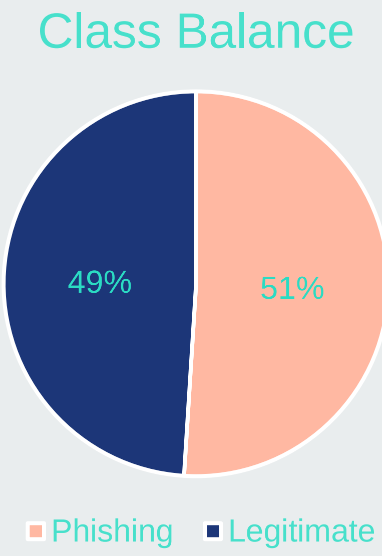
Machine Learning Hackathon CG 2022

Team Name - Tensor HOD
Team Leader Name - Soumen Sardar
Team Leader Email Address - soumensardarintmain@gmail.com

Brief description of the problem at hand:

- Attacker use various fraudulent techniques to perform phishing attract while user is visiting certain websites.
- Here we have a dataset containing **30 features** indicating 3 flags:
  - `-1 : Phishing`
  - `0 : Suspicious`
  - `1 : Legitimate`
- We have a **Result** column in the dataset which indicates:
  - `-1 : Phishing`
  - `1 : Legitimate`
  - **`Good balance between classes`**
- Our task is to identify whether a certain website is **Legitimate(1) or Phishing(-1)** based on given 30 features.

Solution proposed and description:



- The given dataset does **not** contain **any null value** which is good for us.
- After **dropping duplicate** records, we have 5000 records
- Our target variable is - **Result.**
- All features are **categorical** in nature.
- On performing *Exploratory Data Analysis(EDA)*, we found **few features are highly correlated** with the target variable.
- We wish to build machine learning classification model to perform our classification task

Technology/Tool Stack Used:

Language: Python

Data Processing Libraries: Pandas, Numpy

Data Visualization Libraries: Matplotlib, Seaborn, Sweetviz

Machine Learning Library: scikit-learn

Platform: Google Collab

Model Serializer: Joblib 1.1.0

Algorithms: Logistic Regression, Decision Tree, Random Forest, XGBoost

Approach:

Performance Metric:

- Our idea is to keep the **False Negative Rate(FNR)** as **low** as possible.
- We took **ROC_AUC** as our evaluation metric.

Model:

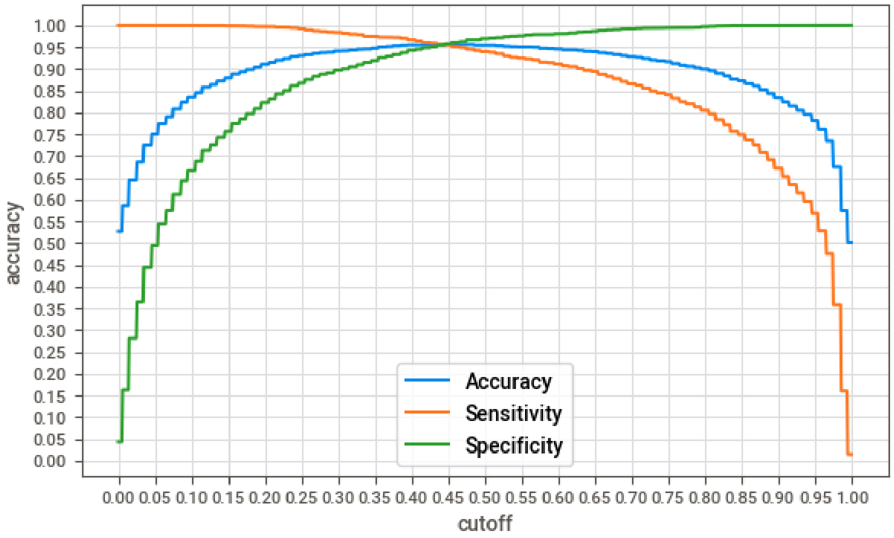
- As the number of features is **not** too high, we started with **simple logistic regression** model
- We used 5 folds cross-validation. As we have almost 5000 data, each fold is having 1000 samples for validation
- Later we have experimented with **Decision Tree, Random Forest and XGBoost** models for better and reliable result.
- We performed **Hyperparameter** tuning and **Fine Tuning** to all the variants.
- We have chosen **Random Forest Classifier** as our final model

Continue…

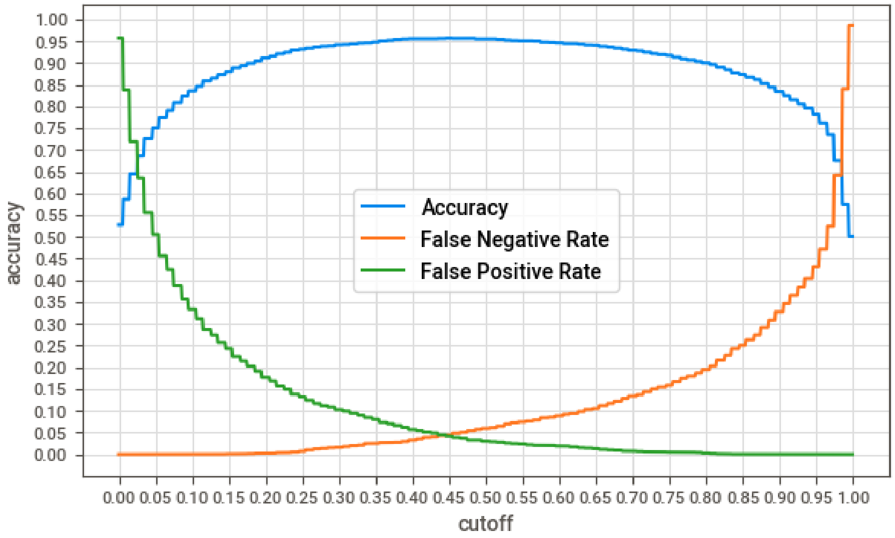
Approach:

- Using machine learning approach, on submission we achieved **92.0906** score.
- Later to improve the score we took an unconventional approach:
- We have compared all **identical** records from both(train and test) dataset and **copy** the **Result** variable form **train to test**.
- Then found **653 unseen records** in **test** data. We have predicted the **Result** variable with our **final model** for those unseen records.
- And we have achieved **95.9677** score

Execution Demo(Video/Screenshots) of the solution:

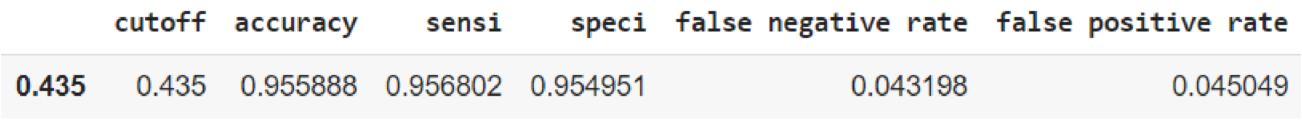


Accuracy vs Sensitivity vs Specificity



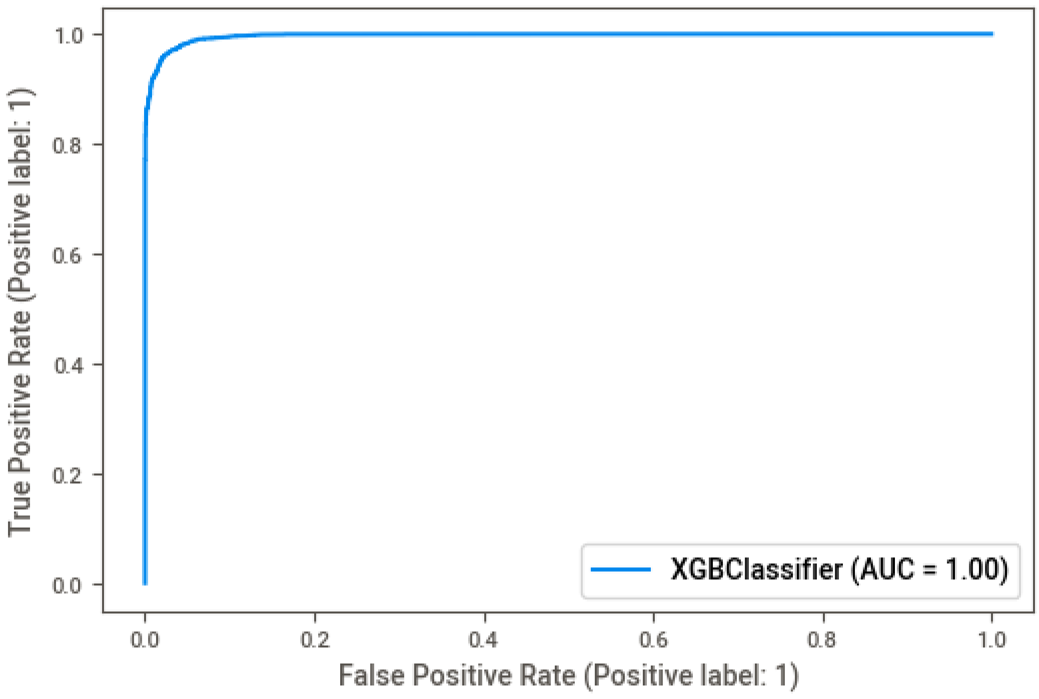
Accuracy vs FN Rate vs FP Rate

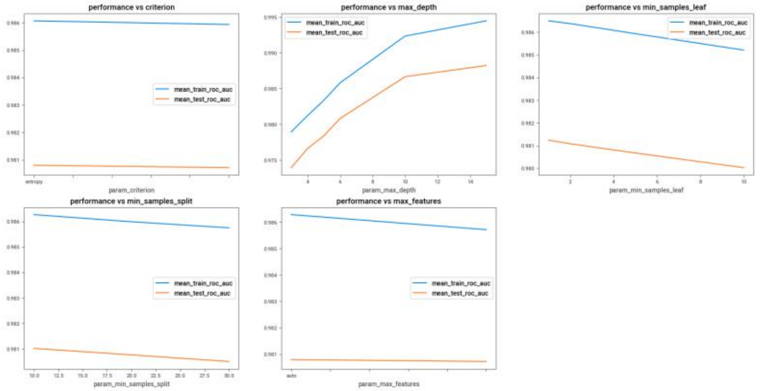
Optimal Probability cutoff: 0.435



	cutoff	accuracy	sensi	speci	false negative rate	false positive rate
0.435	0.435	0.955888	0.956802	0.954951	0.043198	0.045049

Execution Demo(Video/Screenshots) of the solution:





AUC: 0.9967586280475468

Source code as ZIP or Github URL:

GitHub URL:

https://github.com/Redcof/HPE_Machine_Learning_Hackathon_CG_2022

THANK YOU