,

International Institute of Information Technology Bangalore

Machine Learning

AI511

# Project Report: It's A Fraud

Submitted by:
*Darshak Jivrajani*
(IMT2020119)

*Arhant Arora*
(IMT2020503)

December 14, 2022

Pioneering Excellence in Education, Research & Innovation

## Contents

## Overview

GitHub Repository Link: https://github.com/A9Aru/MLProject_ItsAFraud
Given data about transactions, train a model which tells if a given transaction is fraudulent or not.

The Train Dataset given to us has 434 columns and rows. To train a model accurately, we clearly need to perform heavy preprocessing and EDA on the dataset.
On observation, we also find that the dataset is highly biased with only 3.5% of the entries as fraud.

## 1. Preprocessing and EDA:

### 1.1 Removing Null Values:

Since the dataset has only 3.5% of fraud entries, any row with more than 96.5% of NULL Values was removed.

### 1.2 Dealing with Vxx, Cx and Dx columns:

We grouped the V columns on the basis of NULL value %, got 9 different groups and looked at the correlation matrix of each group. This helped us remove a large number of columns.
For the V Columns left, we again looked at the correlation matrix to remove any other similarities among columns. We then performed the same with the C and D columns. This helped us decrease number of columns from 434 to around 102.

### 1.3 Dealing with Skewness:

We looked at the skewness of all the columns. We calculated the square root for the columns with skewness greater than 5.1 and square for the columns with skewness less than -4. After this we calculated the skewness again and this time we removed the columns who's skewness was outside of the range -4 to 5.1.

### 1.4 Filling NULL Values:

For categorical columns, we filled the with the median of the data values we had. For non-categorical columns, same process as mentioned above except we filled the values with mean if we still had any empty columns, we'll fill it as per our data.

### 1.5 Outlier Removal:

We checked the outlier for all columns ad for values between 2% and 98%, we kept it and removed the rest. We are finally ready to train our models.

## 2. Models and Final Scores:

Here is a summary of the models used:

| **Model** | **Hyperparameter** | **Value** | **Model Score** |
|---|---|---|---|
| KNN | metric<br>algorithm<br>leaf_size<br>n_neighbours<br>weights | manhattan<br>ball_tree<br>10<br>11<br>distance | 0.82645 |
| Logistic Regression | C<br>max_iter<br>penalty<br>solver | 0.1<br>100000<br>l2<br>lbfgs | 0.756 |
| Naive Bayes | var_smoothing | 1 | 0.686 |
| Bagging (With Decision Tree) | base_estimator__max_depth<br>max_samples | 5<br>0.5 | 0.85897 |
| ADA Boost (With Decision Tree) | base_estimator__max_depth<br>base_estimator_min_samples _leaf | 10<br>10 | 0.8475 |
| XG Boost | colsample_bytree<br>gamma<br>learning_rate<br>max_depth<br>reg_alpha<br>objective<br>n_estimators<br>njobs | 0.75<br>0.65<br>0.1<br>20<br>0.4<br>binary:logistic<br>8000<br>-1 | 0.928 |
| Neural Networks | Layers<br>loss_function<br>epochs<br>batch_size<br>optimizer | 2(relu, sigmoid)<br>binary_crossentropy<br>20<br>100<br>adam | 0.69716 |

Table 1: Table showing Hyperparameter values for different models and their final scores.

## 3. Observation And Conclusion:

We can observe that the best model for our data is XGBoost. This can be because our data is a high bias one and XGBoost aims to reduce bias. Also, XGBoost works well on a heterogeneous data (our data becomes highly uncorrelated after the initial preprocessing.) Thus we can conclude XGBoost is the best model for a highly-biased heterogeneous binary classification data like the one we had.