

Rishabh Kashyap

PGD - Data Science

16 March 2020

## **Credit Card Fraud Detection**

The problem statement of this project is to predict fraudulent credit card transactions while taking care that we do not misclassify non-fraudulent credit card transactions such that we diminish the benefit of this machine learning model.

Banks always have a loss in fraud cases - if customer follows the protocol, then bank is liable for the loss else bad experience will lead the customer to switch to a different bank. Hence it's quite imperative to minimise the misclassification of fraudulent cases.

We'll start by loading the data and understanding the data. We'll check for null values and treat them accordingly. There is no point in outlier analysis of PCA transformed features, though we'll check for outliers in amount and skewness in PCA transformed data to mitigate these. We'll skip exploratory data analysis and scaling due to PCA transformed data.

In data preparation, we'll use adasyn to create minority class density samples to decrease the class imbalance. Post handling class imbalance, we'll use K-Fold stratified cross-validation method to ensure reliable folds of data set and a better class balance to divide the data for classification algorithms.

For model building, we'll can try different models like logistic regression, random forest and XGBoost with and without regularisation, with an attempt to increase the auc\_roc\_score while validating it with optimal selection of threshold point to reduce misclassification of fraud and non-fraudulent transactions. We won't be trying K-NN and decision tree as there are too many data points and former algorithm is compute intensive also later algorithm has tendency to overfit. We'll use auc\_roc\_score along with precision and recall to evaluate this model and select the best performing model.

Once the best model is selected, we'll perform cost benefit analysis for the train and test data predictions by below mentioned steps -

- Calculate total fraud predictions (TP+FP) made by the model
- Assume INR 10/call for verifying predictions
- Calculate savings from the correct predictions (TP x cost of each correctly predicted transaction)
- Calculate loss from incorrect predictions (FN x cost of each incorrectly predicted transaction)
- Calculate total savings (TP x cost of each correctly predicted transaction - [ (TP x FN) x 10 + FN x cost of each incorrectly predicted transaction ] )