Financial Fraud Detection using Machine Learning

**Adelene Ng, Gareth Jonathan Halim**

# Introduction

With the advent of e-commerce and online purchases, both businesses and consumers alike are increasingly reliant on digital payments for their financial transactions. With the increasing use of online payments, there is a corresponding increase in financial fraud. The sheer volume of transactions together with increasingly sophisticated methods employed by cyber criminals, manual detection or even static rule-based fraud detection is no longer effective. In order to combat increasingly sophisticated methods adopted by cyber-criminals, we will use machine learning and artificial intelligence techniques and tools to fight the war against fraud.

# Problem Framing

We want the machine learning model to be able to predict if a financial transaction is fraudulent or not. Our problem can be best framed as a binary classification problem which predicts if a transaction is genuine or not. The ideal outcome would be to correctly predict a transaction which is fraudulent as actually fraudulent.

Success of the model is measured by the reduction in fraud volume by up to 60% & keeping customers secure with real-time fraud detection.

Our machine learning model is deemed a failure if it fails to detect any fraudulent transactions.

The output from our machine learning model will return true if a given transaction is deemed fraudulent, false otherwise. Our machine learning model will take as input (e.g. transaction amount, transaction type) and the output will be a value of 1 if a transaction is fraudulent, 0 otherwise. The output will be used to decide if the transaction should be approved or denied.

If no machine learning is used, then we either have to resort to manual or rule-based detection, using heuristics such as sudden unusual transaction amounts, unexpected spikes in transactional activities, increased frequency of transactions that are potential red flags.

# Datasets

Since financial datasets cannot be easily collected by individuals, we will use pre-prepared datasets. The lack of public datasets in the financial sector is due to the intrinsically private nature of financial transactions. We will use the dataset from Kaggle, “[Synthetic Financial Datasets For Fraud Detection](https://www.kaggle.com/ealaxi/paysim1)”, which has been generated by the PaySim mobile generator. It simulates mobile money transactions based on a sample of real transactions extracted from one month of financial logs from a mobile money service implemented in an African country. The original logs were provided by a multinational company, who is the provider of the mobile financial service which is currently running in more than 14 countries all around the world.

Table 1 shows the features and their descriptions in the dataset used for this project.

| Features | Description |
| --- | --- |
| step | Maps a unit of time in the real world. In this case 1 step is 1 hour of time. |
| type | CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER |
| amount | Amount of the transaction in local currency |
| nameOrig | Customer who started the transaction |
| oldbalanceOrg | Initial balance before the transaction |
| newbalanceOrig | New balance after the transaction |
| nameDest | Customer who is the recipient of the transaction |
| oldbalanceDest | Initial balance of recipient before the transaction |
| newbalanceDest | New balance of recipient after the transaction |
| isFraud | This is the transactions made by the fraudulent agents inside the simulation. In this specific dataset the fraudulent behaviour of the agents aims to profit by taking control or customers’ accounts and try to empty the funds by transferring to another account and then cashing out of the system. |
| isFlaggedFraud | The business model aims to control massive transfers from one account to another and flags illegal attempts. An illegal attempt in this dataset is an attempt to transfer more than 200.000 in a single transaction. |

Table – Description of Dataset Features

# Machine Learning Approaches to Fraud Detection

We will apply supervised learning approaches to the problem of detecting fraudulent financial transactions. We plan to use the following algorithms for building a fraud detection model:

1. Logistic Regression
2. Random Forest
3. CatBoost
4. Decision Trees

The above list is subject to changes pending the outcome of our experiments.

# Experiments and Evaluation Metrics

## Imbalanced Datasets

The datasets are highly imbalanced. As a result, we need to apply some strategies to alleviate this imbalance in the dataset. Over-sampling (e.g. SMOTE) and/or Under-sampling (e.g. random under-sampling) are some of the techniques we can consider to address this problem.

## Evaluation Metrics

Accuracy alone is not a good metric for highly skewed datasets. We will have to look at other metrics such as:

* Confusion Matrix
* Precision
* Recall
* F1 Score
* Precision Recall curve

to get more insights into the performance of the model.

Using the above metrics, we find the best values for the different algorithms. The winning algorithm will be the final model that will be deployed in production.

# Deployment

The final model will be deployed as a flask-based web application. A Restful API will be provided for third party users to query the model. Given a set of new features as input to the model, the results returned will be a 0 (transaction is genuine) or 1 (transaction is fraudulent).