**Approach/LLD**

**Name:** Nabarup Maity

**Gmail:** [**nabarupmaity@gmail.com**](mailto:nabarupmaity@gmail.com)

**LinkedIn:** [**https://www.linkedin.com/in/nabarup-maity-9ba60916b/**](https://www.linkedin.com/in/nabarup-maity-9ba60916b/)

**Contact:** +91 8777054897

**Challenge link**: <https://www.hackerearth.com/challenges/competitive/hack-the-hackers-hsbc-ml-hackathon-2023/problems/>

**Background:**

At 48 billion, India accounts for largest number of real-time transactions in the world in 2021 according to ACI a real-time payments infrastructure firm, ~3x that of nearest challenger China and 6.5x times greater than the US, Canada, UK, France & Germany combined. Covid-induced movement to real-time money movement ecosystem has improved customer experience but at the same time has opened doors for fraudsters to siphon-off money from bank accounts through innovative methods.

Balancing enhanced CX with safety requires financial institutions to invest in Machine Learning capabilities to detect frauds on a real-time basis. In addition to the vast amount of on-us data about customers, the banks are leveraging consortium solutions to augment their capabilities and producing algorithms to be a step ahead of the fraudsters.

**Problem Description:**

You are working as a data scientist with the Payments team of the bank. The team is continually responding to the emerging threats by building up cutting-edge machine learning driven models and strategies, working with the best-in-class service providers specialized in counter-fraud solutions. In recent years, there has been an increased scrutiny of the digital payments to check for its genuineness.

To aid the team to deal with this problem, you are provided with the payments data to predict whether the customer themselves have made the transfer or not. The payments data contains the attributes which gets captures when a payment is initiated by a banking customer.

**Task:**

You are required to build a machine learning model that can predict whether the customer themselves have made the transfer or not.

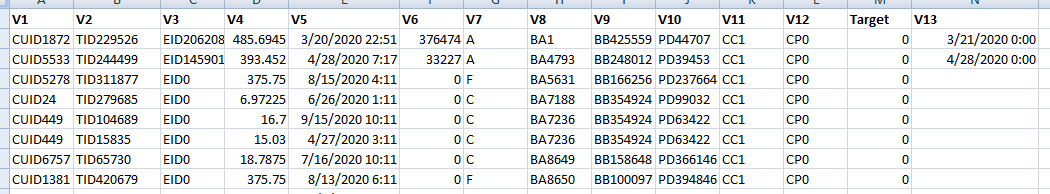
**Dataset description**

The dataset folder contains the following files:

* train.csv: 233633 x 14
* train\_helper.csv: 1231200 x 10
* test.csv: 215852 x 13
* test\_helper.csv: 1160950 x 10
* sample submission.csv: 215852 x 3

**We will work only on train.csv file. Because we do not have enough data in train\_helper.csv file. Out of** 3684 unique customer id(V1) only 92 unique customer information is present in helper file.

**Detailed analysis in details is present on jupyter notebook.**



1. **Data Exploration:**
2. Categorize numerical and Categorical variable. Check if there are variables that are numerical but represent categorical values or vice versa. We observe that V6 is categorical.
3. Analysis of each numerical and categorical variable. For numerical variable with histogram and for categorical variable with their unique variable.
4. Handling Missing Value: Only V13 has missing value of 1871(0.8%). For imputation we have used the median value of delay between V5 and V13 column. Detailed has been there in feature engineering section.
5. Target distribution/count check related to other columns

Few notable observations are:

1. If we group any transaction based on customer id(column V1) and sort it based on transaction time(column V5) in historical order then if there is little gap between those transaction time(very little time gap between previous transaction and current transaction) then it is most likely a fraud.
2. If we group any transaction based on customer id(column V1) and sort it based on transaction time(column V5) in historical order then if the value of column V8 and V9 are exactly same to their previous value then it is most likely a fraud.
3. **Feature Engineering:**

First we have tried to include additional features which we may or may not going to use in the model.

1. unique\_transactions: Number of unique transactions for a given customer ID
2. is\_large\_amount: For a given customer ID(V1) if the current transaction amount(V4) is higher than average \* 1.5.
3. is\_large\_amount\_quantile: For a given customer ID(V1) if the current transaction amount(V4) is higher than 95% quantile.
4. similar\_transaction: For a given customer id(V1) if we order it by transaction time(V5) older to newest and if the current amount(V4) is same as previous transaction.
5. Potential fraud: Based on excel observation if we group any transaction based on customer id(column V1) and sort it based on transaction time(column V5) in historical order then if the value of column V8 and V9 are exactly same to their previous value then it is most likely a fraud.
6. For date time column V5 and V13 we have extracted related features as day\_of\_year, week\_of\_year, month, is\_weekend. Then we have included a future V13\_V5\_diff that will hold the difference between V5 and V13
7. multiple\_transactions: column indicating whether there have been multiple transactions within a short period of time
8. time\_diff: indicates the time difference between each transaction and the previous transaction for each customer in V1 column.
9. Dummy encoding of V7 variable
10. **Feature Selection:**
11. Searching for Duplicated or Quasi-constant features: 'hour13', 'V7\_C', 'V7\_D', 'V7\_E', 'V7\_F'are constant. Hence removed
12. Looking for Correlated Features: Correlated Feature Sets [{'day\_of\_year', 'month', 'week\_of\_year'}, {'week\_of\_year13', 'day\_of\_year13', 'month13'}]. Features to drop {'week\_of\_year13', 'month', 'week\_of\_year', 'month13'}
13. Select top k features using chi square method
14. **Scaling:**

We have used Minmax, Standard and Robust scaler

1. **Building Model and Hyperparameter Tuning:**

Used below models with hyperparameter tuning. Used oversampling in some cases(personally not recommended for this type of data)

1. LDA
2. Logistic Regression
3. KNN
4. SVM
5. XgBoost
6. Basic Neural Networks
7. **Evaluate Model:**

Internal Evaluation has been done with macro average precision, recall and f1 score. Also Evaluated with AUC-ROC precision-recall score.

Although in (75-25)training-validation data all models are performing well but they all failed in final test set. Also apart from xgboost precision is low for all models. Means the models are making a lot of false positive predictions, but is correctly identifying most of the positive cases in the data. All Score has been there in notebook. Please refer.

1. **Submission:**

Around 40 submission has been made

1. **Future Task**
2. Use domain knowledge to extract/use features
3. Try one class classification technique
4. More feature engineering
5. Check non linearity correlation between features
6. Look for polynomial features and use them precisely as a feature
7. Oversampling and undersampling smart way as this is a highly imbalanced problem.