

**Transactional Fraud Detection: A Technical Analysis of  
Machine Learning Approaches on Real Time Transaction Data  
using Dataset Balancing with SMOTE**



**By**

***Ms. Zainab Saeed Butt***

00000320641

Supervisor

**Dr. Abdul Wahid**

Department of Computing

A thesis submitted in partial fulfillment of the requirements for the  
degree of Master of Science in Computer Science (MS CS)

In

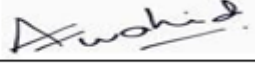
In School of Electrical Engineering & Computer Science (SEECS),  
National University of Sciences and Technology (NUST), Islamabad,

Pakistan

(October 2022)

## Thesis Acceptance Certificate

Certified that final copy of MS thesis entitled “**Transactional Fraud Detection: A Technical Analysis of Machine Learning Approaches on Real Time Transaction Data using Dataset Balancing with SMOTE**” written by **Zainab Saeed Butt, (2019-MSCS 320641 SEECS)**, of School of Electrical Engineering & Computer Science (SEECS) has been inspected by the undersigned, found completed in all respects as per NUST regulations. It is free of plagiarism, errors and anomalies and is accepted as partial fulfillment for award of MS degree. It is further certified that essential amendments as pointed out by GEC members of the scholar were incorporated in the said thesis.

Signature: \_\_\_\_\_ 

Name of Advisor: \_\_\_\_\_ Dr. Abdul Wahid

Date: \_\_\_\_\_ 19-Aug-2022

HoD/Associate Dean: \_\_\_\_\_

Date: \_\_\_\_\_

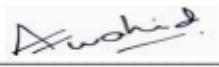
Signature (Dean/Principal): \_\_\_\_\_

Date: \_\_\_\_\_

## Approval

It is certified that the contents and form of the thesis entitled “**Transactional Fraud Detection: A Technical Analysis of Machine Learning Approaches on Real Time Transaction Data using Dataset Balancing with SMOTE**” submitted by **Zainab Saeed Butt** have been found satisfactory for the requirement of the degree.

Advisor : Dr. Abdul Wahid

Signature: 

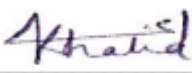
Date: 19-Aug-2022

Committee Member 1: Dr. Dr. Muhammad Zeeshan

Signature: 

19-Aug-2022

Committee Member 2: Mr. Shah Khalid

Signature: 

Date: 31-Aug-2022

## **Dedication**

This thesis is dedicated to my family, fiancé, friends, colleagues, and respectable teachers who supported throughout in research and career counseling.

## **Certificate of Originality**

I hereby declare that this submission is my own work and to the best of my knowledge it contains no research that has been previously published or written by other person, nor material which to a considerable extent has been accepted for MS Degree at NUST SEECS or at any other educational institution, except where due acknowledgement has been made in the thesis. Any contribution made to the research other than me, with whom I have worked at School of Electrical Engineering & Computer Science (SEECS) or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work. The assistance from others in the project's implementation and designs, presentation and linguistics has been acknowledged.

Author Name: **Zainab Saeed Butt**

Signature:



## **Acknowledgement**

Glory be to Allah (S.W.A), the Creator, the Sustainer of the Universe. Who only has the power to honor whom He please, and to abase whom He may pleases. Verily no one can do anything without His will. From the day, I came to NUST till the day of my departure, He was the only one Who blessed me and opened ways for me and showed me the path of success. There is nothing which can payback for His bounties throughout my research period to complete it successfully.

**Zainab Saeed Butt**

## List of Contents

Thesis Acceptance Certificate.....	i
Approval .....	ii
Dedication .....	iii
Certificate of Originality .....	iv
Acknowledgement .....	v
Abstract.....	ix
Chapter 1 .....	1
Introduction.....	1
1.1. Introduction:.....	2
1.2 Motivation:.....	3
1.3 Problem Statement and Objectives: .....	3
1.4 Thesis Contribution and Outline: .....	4
Chapter 2.....	6
Literature Review.....	6
2.1. Dataset Creation, Balancing, Training and Fraud Detection: .....	7
Chapter 3.....	19
3.1. Transactions Dataset Formation: .....	21
3.2. Dataset Transformation and Fraud Prediction Before SMOTE:.....	21
3.2.1. Isolation Forest.....	29
3.2.2. Local Outlier Factor Algorithm .....	30
3.3. Dataset balancing with SMOTE: .....	32
3.3.1 Solving the problem of imbalance dataset: .....	33
3.4 Implementing Classifiers on balanced dataset:.....	35
Chapter 4.....	37
Results and Discussion .....	37
4.1. Resultant Confusion Matrices:.....	39
4.2. Logistic Regression and Parameters Tuning.....	40
4.3. Pre balancing results on dataset: .....	40
4.3.1. Post balancing results on dataset (Logistic Regression): .....	41
4.5.How SMOTE Worked?:.....	41
4.6. Balanced Dataset Creation Using SMOTE:.....	41
5. Recommendations.....	42
6. Conclusion .....	43
6.1. Future Scope.....	43
References.....	44

## List of Figures

Figure 1 The hybrid framework by (Li 2021).....	8
Figure 2a. Architecture by (Longfei Li 2020).....	10
Figure 3 2b. Comparison of Algorithms .....	12
<i>Figure 4. Fraud Detection Steps.....</i>	<i>13</i>
Figure 5 Histogram of equally distributed classes after subsampling.....	15
Figure 6. Dataset Balancing and Fraud Prediction.....	20
Figure 7 Dataset Transformation and Correlation Matrix.....	23
Figure 8 Visualization of Dependent Columns.....	24
Figure 9 Heat map featuring correlation matrix.....	25
Figure 10 Principal Component Analysis – Visualization.....	26
Figure 11 Isolation Forest – Visualization.....	30
Figure 12 Local Outlier Factor – Visualization .....	31
Figure 13 Local Outlier Factor - Visualization.....	32
Figure 14 Results of SMOTE are discussed in results section.....	34
Figure 15 The balanced data curve .....	36



## List of Tables

Table 1. Key hyper-parameters of models for different datasets .....	8
Table 2 Global Performance of models by means of 24 times series .....	9
Table 3 Fraud Detection Dataset Description .....	10
Table 4 Mean AUCP R and AUCP R0.2 .....	11
Table 5 Accuracies on classifiers .....	14
Table 6 Confusion Matrix .....	15
Table 7 Dataset Transformation and Normalizing .....	22
Table 8 Isolation Forest Results .....	30
Table 9 Local Outlier Factor Results .....	31
Table 10 Logistic Regression and Parameter Tuning .....	36
Table 11 Applied Models and Hyperparameters .....	38

## **Abstract**

Machine Learning algorithms have been performing expedient predictions in the fields like IT-banking and data rich business-related problems. In transactional fraud detection, it is important to analyze customers' pattern of transactions as every user has few operation patterns that must be taken into consideration. One of the major challenges that transactional fraud detection has to face is confidentiality of actual user data and the sensitive variables of the dataset used for training purpose. It is difficult to obtain real world datasets specially when there are very low number of fraudulent transactions present in millions of genuine transactions. Another challenge is to evaluate one's work to judge its performance based on evaluation matrices and criteria. In Machine Learning, problems like anomaly detection cannot be simply evaluated based on accuracy. In this thesis, we have created a transaction dataset from scratch containing transactions from March 2021 to May 2021. Since our initial data is raw and highly imbalance, we experiment with data transformation and machine learning techniques in order to detect fraudulent transactions. After comparison of multiple methods, we share our results and conclude that balanced dataset is the key to achieve highest accuracy on applied classifiers. We achieve 78% accuracy after balancing out dataset via SMOTE analysis on dataset that happen to be 28% more than that of imbalance dataset i.e. 50% in first few trials.

# **Chapter 1**

## **Introduction**

## **1.1. Introduction:**

Transactional fraud has been a problem for financing industries and banks since the beginning of online transaction systems. Every other day we hear scams and frauds being placed via commercial transactions. With the rapid advancement of economy globalization in two decades, credit cards are much more popular in commercial transactions. This makes a corresponding problem of the credit card fraud emerge consequently. Machine learning approaches have been proposed to overcome such contests. Despite of the possible measures taken by financial institutions, millions and billions are lost due to fraud transactions and analyzing these frauds in IT-banking domain has become one of the hot research topics today. In recent decades, researchers have been experimenting with modern techniques in Artificial Intelligence related domains like Machine Learning, Data Mining and Genetic Programming. Most of the work done in fraud detection has been implemented on Credit Card Transactions taken from several institutions who share their data voluntarily or as a research participator to perform experimentation and fraud predictions.

Machine Learning algorithms have been performing useful predictions in the fields like IT-banking and data rich business related problems. It is important to analyze customers' pattern of transactions as every user has few operation patterns that must be taken into consideration. One of the major challenges that transactional fraud detection has to face is confidentiality of actual user data and the sensitive variables of the dataset used for training purpose. It is difficult to obtain real world datasets specially when there are very low number of fraudulent transactions present in millions of genuine transactions. Another challenge is to evaluate one's work to judge its performance based on evaluation matrices and criteria. In Machine Learning, problems like anomaly detection cannot be simply evaluated on the basis of accuracy. In this thesis, we have created a transaction dataset from scratch containing transactions from March 2021 to May 2021.

The use of online credit/debit card transactions looks to be increasing as the internet and e-commerce increase. Increased use of credit and debit cards has resulted in a rise in fraud. The scams can be identified using a variety of methods, but each has its own set of problems and inaccuracies. To increase the accuracy of the categorization, behavior-based machine

learning classifiers have been applied in this study. Frauds are predicted and taken for further processing if there are any changes in the transaction's conduct. [1]

As there is a lot of work been done on commercial purchases and credit cards, we have worked with cards and INET fund transfers for this research thesis. In the report below, we tend to propose a model based on combination of existing techniques that performs transactional fraud detection on dataset containing 1 million transactions. Dataset has been created from scratch while taking care of all confidential measures.

## **1.2 Motivation:**

Transactional Fraud has been a problem since the technical advancements became normal in day to day customer transactions. Credit card and internet transactions are becoming more and more popular in financial transactions, at the same time frauds are also increasing.

Conventional methods use rule-based expert systems to detect fraud behaviors, neglecting diverse situations e.g. extreme imbalance of positive and negative samples. Most students avoid using real time financial data due to its confidentiality that is why in IT-Banking sector, transactional fraud detection remains unresolved when compared to other hot research topics in Machine Learning. We aim to use real world dataset of card based transactions and fund transfers to detect fraudulent transactions.

## **1.3 Problem Statement and Objectives:**

- In IT-Banking sector, transactional fraud detection still remains unresolved due to confidentiality of the costumer transaction data. Frauds can happen through various transactions including credit cards, debit cards, online payments and other INET based transactions.
- We aim to use real-world datasets from a well reputed bank to predict and analyze faulty transactions by applying machine learning classifiers to achieve optimal accuracies and perform technical evaluation on our proposed architecture.
- One of the problems that must be taken into consideration is the problem of imbalance data. In electronic fraud transaction detection, class imbalance with overlap is a difficult problem to solve. To avoid being discovered, fraudsters have strained their brains to create a fake transaction that looks exactly like the real one. As a result, a large amount

of data from fraudulent transactions overlaps with data from legitimate transactions, making it difficult to distinguish between the two. However, the focus has been on class imbalance rather than overlapping concerns for Machine Learning based fraud transaction detection approaches.

- Imbalanced data classification is a type of classification predictive modelling issue in which the number of examples in the training dataset for each class label are unevenly distributed. That is, when the class distribution is partial and biased or skewed rather than being equal or close to equal. In our model we incorporate a hybrid version of techniques like SMOTE, clustering and under sampling to solve data unbalancing.

## **1.4 Thesis Contribution and Outline:**

- As most recent works in financial fraud domain have been done on data with rich history of transactions mostly on credit card transactions and online payment systems.
- In our proposed model illustrated below, we create a huge dataset of debit card, credit card and other miscellaneous transactions (including INET etc.) belonging to a commercial bank and solve problems related to imbalance data (e.g. customers who don't have much history and certain pattern of transactions) by experimenting with techniques like under sampling combined with clustering that we form of data classes.
- As there are very few frauds found in transactional data of banks, that is why we are focusing on balanced data techniques so that our ML model best fits to achieve optimal accuracy on prediction
- After balancing the data we perform data cleaning and analysis of nullified transactions and then implement supervised machine learning algorithms to predict fraudulent transactions. We then aim to submit best fit classifier and analyze precision metric and evaluate results. We then try to compare our pre and post balancing results to make a technical comparison on how data balancing affects performance of classifiers on predictions
- Gather dataset of real world customer transactions and fund transfers with old and new updated balances with in premises of the bank.

- Balance dataset using under sampling and clustering techniques and optimize existence of fraudulent patterns of transactions like transaction amount and balance being exceeded and immediate cash outs after fund transfers.
- Thesis report is divided into chapters listed below:

Chapter 2: **Literature Review**

Chapter 3: **Methodology**

Chapter 4: **Results and Discussion**

Chapter 5: **Conclusion**

Chapter 6: **Recommendation**

Chapter 7: **References**

# **Chapter 2**

## **Literature Review**

In IT-Banking sector, transactional fraud detection remains unresolved due to confidentiality of the customer transaction data. Frauds can happen through various transactions including credit cards, debit cards, online payments and other INET (Internet Banking) based transactions. Researchers have made attempts on real – world datasets from well reputed institutions to predict and analyze faulty transactions by applying machine learning classifiers to achieve optimal accuracies and performed technical evaluation on proposed architectures. However, most of the research has been done on credit card transactions due to ease of availability of third party datasets.

As our work starts from dataset creation, then balancing and training, we have listed below some popular research works in this domain by dividing work into two phases i.e. Dataset balancing then transactional fraud detection.



## 2.1. Dataset Creation, Balancing, Training and Fraud Detection:

Problem of imbalance dataset has been a point of discussion in Anomaly Detection Domain especially when data has to be confidential and contains sensitive information about customers involved. [2] In this article, dataset balancing with smote has been discussed. When the number of samples representing one class is substantially smaller than the others, this is known as an unbalanced class data distribution. The prediction accuracy on minority data suffers as a result of this conditioning. Synthetic Minority Oversampling Technique (SMOTE) is a pioneer oversampling method in the research field for imbalanced classification to address this difficulty. Due to the need of avoiding overfitting and assist the classifier in discovering decision boundaries between classes, the basic principle of SMOTE is oversampled by producing a synthetic instance in feature space generated by the instance and its K-nearest neighbors. They have discussed performance issues and evaluations and provided a survey on a new extension of SMOTE.

[3] Tackles the problem of imbalance datasets. This paper suggests a novel hybrid strategy based on the divide-and-conquer concept to address the problem of class imbalance with overlap. To begin, an anomaly detection model is trained on the minority samples in order to eliminate a few minority class outliers as well as a large number of majority samples from the original dataset. The remaining samples then combine to form an overlapping subset with a lower imbalance ratio and less learning interference from both the minority and majority classes than the original dataset. After that, a non-linear classifier is used to deal with this challenging overlapping subset in order to separate them well. They offer a new assessment criterion, Dynamic Weighted Entropy (DWE), to evaluate the quality of the overlapping subset in order to attain good attributes. They presented a trade-off between the number of minority class outliers which are normally excluded and the ratio of class imbalance in overlapping subsets. The amount of time spent searching for good hyper-parameters is greatly reduced when using DWE. Extensive tests on the Kaggle fraud detection dataset as well as a large real-world electronic transaction dataset show that their solution beats state-of-the-art methods. Figure below is inspired by the idea of Divide-and-Conquer, their hybrid framework consists of two steps, Divide step and Conquer step.

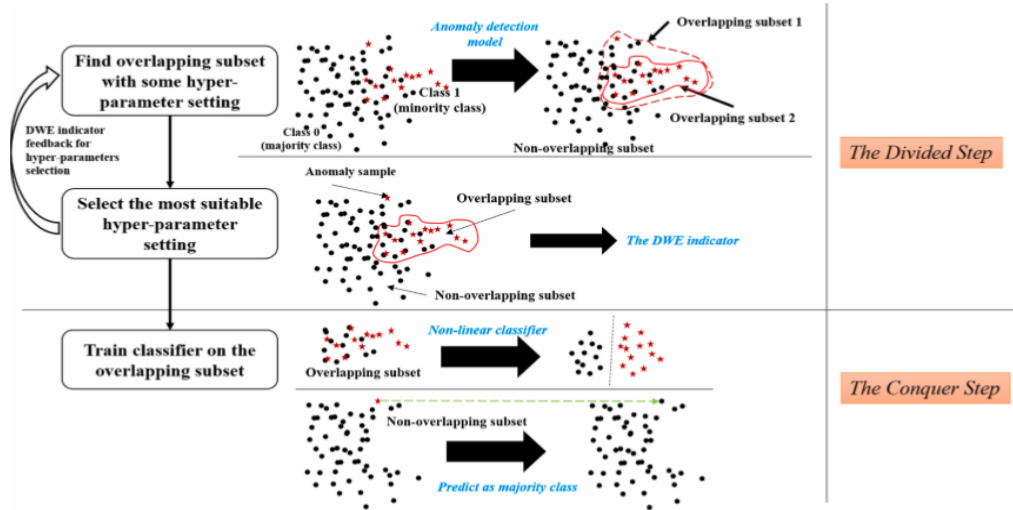


Figure 1 The hybrid framework by [3]

Models Datasets	Kaggle	Private
RF	tree number = 50, max deep =5	tree number = 100, max deep = 7
AE	It has 4 fully connected layers: encoder-1:16 + elu encoder- 2:8 + tanh decoder-1:8 + elu decoder-2:16 + tanh	It has 6 fully connected layers: encoder-1:32 + elu encoder- 2:16 + tanh encoder-3:8 + elu decoder- 1:8 + elu decoder- 2:16 + tanh decoder-3:32 + elu
ANN	It has 3 fully connected layers: FC-1:32 + elu FC-2:64 + elu FC-3:16 + softmax	It has 5 fully connected layers: FC-1:64 + elu FC-2:128 + elu FC-3:128 + elu FC-4:64 + elu FC-5:32 + softmax

Table 1. Key hyper-parameters of models for different datasets

[4] Authors have addressed an unsupervised methodology of credit card fraud detection in unbalanced datasets by introducing the ARIMA model. The ARIMA model is designed to fit with the frequent spending behavior of the customer and is used to detect fraud if some irregular patterns appear. Their model is applied to credit cards and pos transactions and is compared to multiple anomaly detection approaches: isolation forest, K-means, box plot, local outlier factor.

ARIMA has performed better than other benchmarks as mentioned in below table. In the training set, the ARIMA model is first standardized on the basic of daily genuine customer transactions in order to learn the regular spending behavior of the customer. In the second phase, the fitted model is used to predict fraud in the testing set by using the rolling windows. The measure of flagging fraud is based on the Z-score calculated on the prediction errors in the testing set.

METRICS	ARIMA	BOX-PLOT	LOF	IF	K-MEANS
<b>Precision</b>	34.29%	28.96%	6.41%	19.94%	22.51%
<b>Recall</b>	42.03%	60.54%	69.57%	64.09%	68.16%
<b>F-Measure</b>	36.19%	34.91%	11.17%	24.82%	26.81%

*Table 2 Global Performance of models by means of 24 times series*

Another recent work in transactional fraud and credit card datasets has been done by [5]. They have implemented deep learning models in order to handle sequence data of user transactions. They have used LSTM and RNN to keep check of the time frequency intervals between transactions. They have further deployed the proposed framework as a real system at Alipay (a third party payment scheme by Alibaba group), and the results have been validated on real-world scenarios. They have proposed their LSTM version by introducing time attention based recurrent layers.

They have compared model results with existing models. Their dataset comprises of real transaction data from Alipay where both fraudulent and genuine transactions are available for training and testing purpose. Their proceedings are mentioned below:

Dataset	User	Sequences	Non Fraud Transaction	Fraud Transaction
Train Set	1,221,706	3,837,624	3,832,560	5,064
Validation Set	656,521	1,248,912	1,247,315	1,597
Test Set	674,057	1,302,226	1,302,091	135

Table 3 Fraud Detection Dataset Description

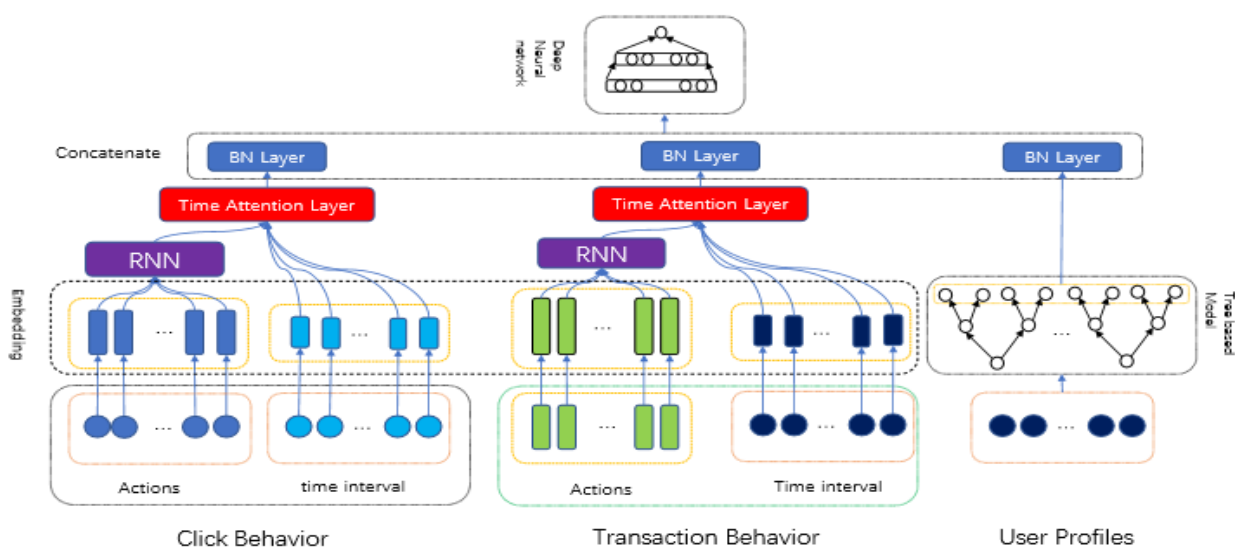


Figure 2a. Architecture by [5]

A study by [6] has referred fraud detection as sequence classification task which means they have used neural networks to predict transactions as fraudulent or usual by training labeled datasets. They have performed fraud detection on both online and offline detections using LSTM and Random Forest and concluded that different types of frauds are detected by these techniques; hence, a combination of both must be implemented. They have experimented with feature engineering techniques i.e. Time Delta and Feature Aggregations to explicitly abridge the purchasing activities of customers.

Their research is carried on a dataset of credit-card transactions, recorded from March to May 2015. Each transaction in the dataset has a Boolean label assigned that indicates whether the transaction was a fraudulent or not. They have trained LSTM and Random Forest for each combination of feature set, data set and sequence length and then tested its classification performance on the held-out test set as given below in table 2c.

ECOM					
		AUCPR ( $\mu$ )		AUCP R <sub>0.2</sub> ( $\mu$ )	
FEATURES		RF	LSTM	RF	LSTM
BASE		0.179	0.180	0.102	0.099
TDELTA		0.236	0.192	0.124	0.107
AGG		0.394	0.380	0.158	0.157

**Table 4 Mean AUCP R and AUCP R0.2**

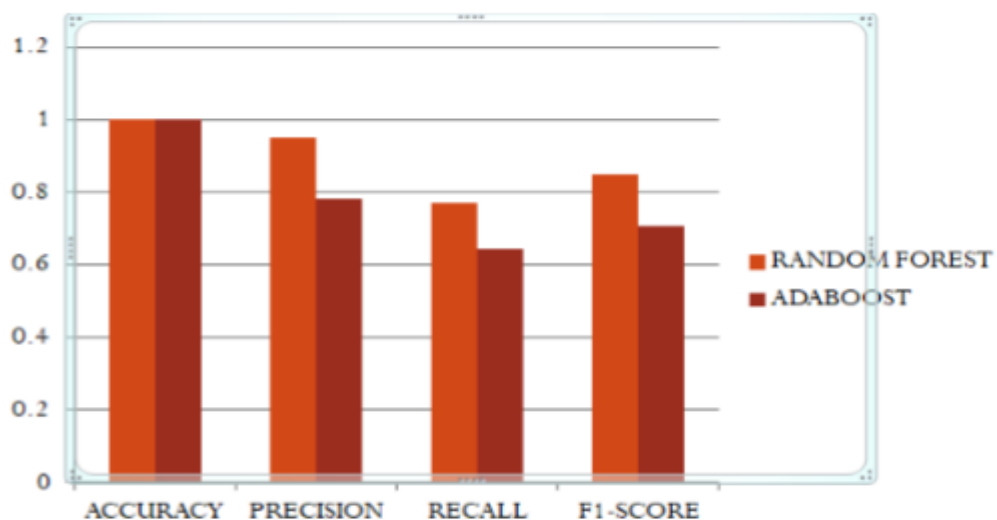
Another researcher named [7] has proposed fraud detection algorithm for online transactional frauds by developing a decision tree IFDT<sub>C4.5</sub> using intuition based fuzzy logic and C<sub>4.5</sub> decision tree to detect fraudulent transactions. They have achieved optimal accuracy better than recent works. Their dataset is of credit card transactions and taken from a Singaporean bank. Proposed decision tree model has performed better than existing techniques in terms of accuracy as it has merged irrelevant branches of decision tree and removed

redundant resultantly, reducing computational cost. Normal, indeterminable and fraudulent transactions have been trained based on false positive and negative values

A novel method introduced by [8] Performs credit card fraud detection by grouping customers based on their transaction patterns. They have extracted behavioral patterns of customers' transactions and then developed a profile. They have compared results produced by Machine Learning Classifiers applied on imbalanced and balanced datasets. The dataset contains total of 284,807 transactions in which 492 transactions are fraudulent. Finally, they achieve optimal accuracy on Adaboost Algorithm and Random Forest Classifiers. Proposed Adaboost algorithm works as:

- Algorithm Adaboost : Input Transactions Adjust weights,  $w_1(n)=1/n$
- A decision tree is created and the one that has the lowest Entropy is selected
- If Incorrectly classified , Calculate Total Error (TE)= sum of up incorrectly Classified sample weights
- Evaluate Performance, Loop: For each Incorrectly classified,
- Increase weights: Weights incorrect = old weight \* Correctly classified,
- Decrease the weights: Weight correct = old weight \* Normalized weight of each sample: Normalized weight = End for End if

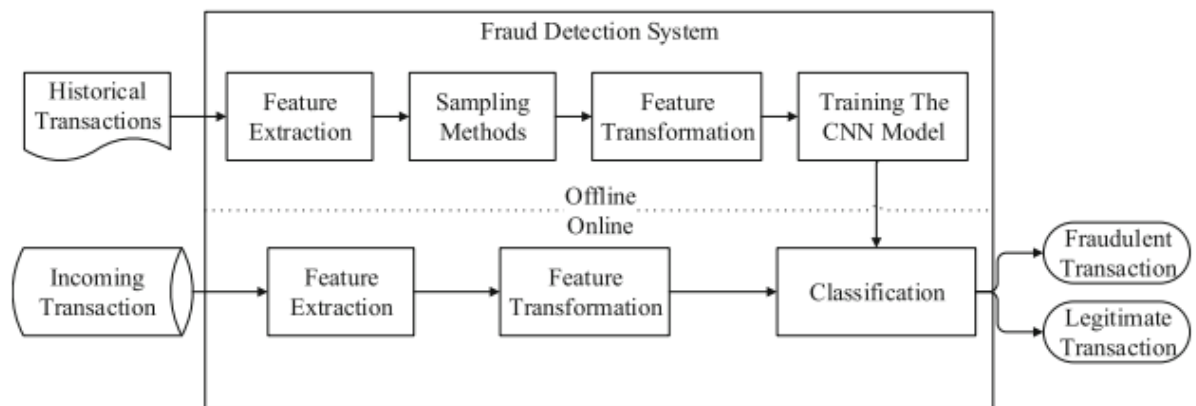
Their highest test result on precision and recall is around 98% on random forest. They submit best results against Random forest as shown in figure below:



*Figure 3 2b. Comparison of Algorithms*

[9] Proposes a CNN based fraud detection framework to capture intrinsic patterns of fraud behaviors learned from dataset. Have incorporated the art of feature engineering in their proposed model and performed transaction fraud detection on a commercial bank’s dataset. They have proposed trade entropy to mode trading behaviors and then tried to configure trading features into feature matrices to predict customers’ behaviors upon transactions. They have achieved better accuracy than state of the art models.

A feature matrix is formed using abundant transaction data and then a convolutional neural network is trained to recognize a set of underlying patterns. They have used real world datasets from a commercial bank.



**Figure 4. Fraud Detection Steps**

For testing models, they have used real transactions of credit card dataset. It contains around 260 million transactions of credit cards in a year out of which four thousand transactions are labeled as frauds and the rest are genuine transactions. The transaction data is divided into two sets. They have taken data of the first eleven months as the training set and the data of the twelfth month as the testing set. They have set F1 score as their evaluation measure. They have made a generic comparison between CNN and other existing models as such as Neural Networks, SVM and Random Forest and achieve highest F1 score on CNN however they did not share a noticeable result section.

[10] In this paper researchers have analyzed transaction the dataset which is taken from Kaggle. Their dataset contains Credit card transactions which were made by European customers during September, 2013. They have monitored the behavior of the transactions and have characterized them into two categories fraudulent and non-fraudulent. Anomalies are

created based upon these two classes. Then by using existing techniques: Local Outlier Factor and Isolation Forest the behavior of these anomalies has been investigated and compared with existing algorithms. [11] had introduced the Local Outlier Factor (LOF) algorithm to find the anomalous data points by measuring the local deviation of a specified data point with respect to its neighbors. [10] Experimented with transactional fraud proposing that LOF can be a good substitute to other famous algorithms used for fraud detection. Their results are given below in table 2c:

<b>Algorithm</b>	<b>Accuracy</b>
<b>Logistic Regression</b>	90.0%
<b>Decision Tree</b>	94.3%
<b>Random Forest</b>	95.5%
<b>Isolation Forest</b>	71%
<b>Local Outlier Factor</b>	97%

*Table 5 Accuracies on classifiers*

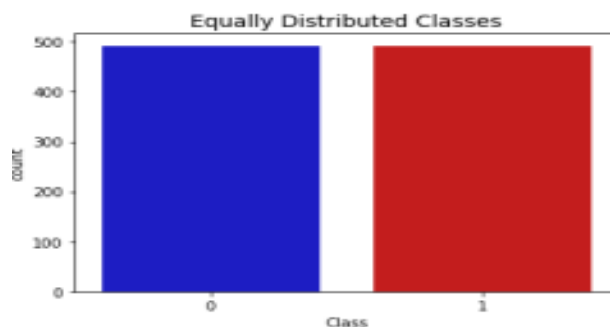
[9] Have incorporated the art of feature engineering in their machine learning algorithms (LSTM and CNN) and performed transaction fraud detection on a commercial banks dataset. They have proposed trade entropy to mode trading behaviors and then configured trading features into feature matrices to predict customers' behaviors upon transactions. They have achieved better accuracy than state of the art models. They evaluated resulted upon confusion matrix as below:



Actual Class	Predicted Class	
	Negative	Positive
Negative	True Negative	False Positive
Positive	False Negative	True Positive

**Table 6 Confusion Matrix**

This work [12] aims to research on a set of mathematical models and algorithms that examine the data of a single payment transaction to categorize it as scam or verified. They have treated it as a classification problem. Their main goal was to apply different machine learning techniques to find the most accurate one. They have evaluated accuracy in terms of cross-validation i.e. ideal classifier works best when the cross-validation score is utmost. Out of all tested models (K-Nearest neighbors, Support Vectors Method (SVM), Logistic Regression, Decision Tree Classifier and Artificial Neural Networks), Logistic regression showed the best accuracy with an estimate of 94%. They also tackle the problem of data balancing, however over sampling doesn't show best results as over-sampled data shows 99.9% accuracy but slips a significant amount of fraudulent operations due to overfitting.



**Figure 5 Histogram of equally distributed classes after subsampling.**

In this research [13] authors have tackled the problem of class unbalancing. The distribution of classes between the dominant and minority classes is not equal, which causes a class imbalance. Data on unequal distribution of wealth might range from minor to major. To address this problem oversampling techniques are used to balance the skewness between the data and one of those techniques is SMOTE which is used in this paper to tackle the over sampling problem. In the end the same dataset is used to predict the classes before and after SMOTE is applied on the dataset. Their experiments show that Random Forest with Borderline-SMOTE gives the best value with an accuracy value of 0.99, 0.94 precision, 0.85 recall and 0.90 F1-score respectively.

[14] The methods now in use are insufficient to conduct detection with high precision, even though several research have sought to examine them. Authors have provided a Deep-forest based method for detecting online transaction fraud that combines a deep-forest model with a differentiation feature generating technique. This research has included a transaction time-based differentiation feature generating approach into scheme since a single-time transaction's information, which lacks information like the user's behavior, is insufficient for identifying fraudulent transactions.

[16] To differentiate between legitimate and fraudulent transactions, two metrics based on transaction time are developed: Individual Credibility Degree (ICD) and Group Anomaly Degree (GAD). Additionally, they use the Deep-forest algorithm to identify fraudulent transactions to address the enormous imbalance of online transactions. While the raw deep-forest model may be able to disregard the outlier transaction samples, they improve the model by adding an outlier detection method and paying closer attention to outliers to increase the model's accuracy in detecting fraud. Finally, they run tests utilizing transaction data from a bank. Their proposed technique increases recall rate and accuracy rate by 15% and 20%, respectively, as compared to the random Forest-detection model.

[15] The idea of plastic money has been extensively adopted in our day and age, but every new technology also has its flaws. Numerous abnormalities of different kinds might occur in this case, harming the consumer financially. These abnormalities might be categorized as financial sector scams. The researchers have suggested a wide range of strategies and models to identify these kinds of scams. The suggested work in this paper aims to create an automated model for the identification of certain types of frauds, particularly those connected to credit card transactions. On a large dataset, the suggested models employed four machine learning

methods to predict the fraud: Naive Bayes, Random Forest, Logistic Regression, and SVM. Among all machine learning algorithms, the Naive Bayes method does exceptionally well at detecting credit card fraud, with an accuracy rate of 80.4% and an area under the curve of 96.3%.

[16] SMOTE and ADASYN, two over-sampling methods, are compared for performance. Three unbalanced data sets are compared using three distinct classification models and assessment criteria, while also altering the pre-processing of the data. According to the results, SMOTE and ADASYN generally enhance classifier performance. It is also discovered that when the degree of class imbalance grows, SVM in combination with SMOTE performs better than ADASYN. Additionally, when the degree of class imbalance increases, SMOTE and ADASYN both improve the relative performance of the Random Forest. Although the degree of class imbalance fluctuates, no pre-processing technique consistently surpasses the others in terms of its contribution to improved performance. [20] This study shows that workflows and neural network algorithms can detect with up to 95% accuracy even with a relatively tiny fraud sample of only 0.17% or 492 of 284,807 transactions, using data from European cardholders in 2013. Additionally, the Adam optimizer outperforms the Adamax optimizer in terms of accuracy. The consequence is that this advancement in supervisory technology can be used to reduce financial services industry transaction crimes.

[17] Research article talks about smote and its features. The procedure computes the median value of the standard deviations after computing the standard deviations for all continuous qualities in the minority class; The metric for determining the k-nearest neighbors is determined using the Euclidean distance, with an additional correction for any category qualities that do not match. The pre-computed median is added as a component to the Euclidean distance calculation for each category attribute that differs between the chosen minority member and a possible neighbor; Correction for synthetic samples: Using vanilla SMOTE, the continuous features for the synthetic samples are computed. The most frequent values in the categorical characteristics of the k-nearest neighbors are used to set the synthetic categorical attributes.

[26] First, they put forth a formalization of the fraud-detection problem with the aid of an industrial partner that accurately captures the operational circumstances of Functionally Designed Systems FDSs that regularly examine enormous streams of credit card transactions. We also provide examples of the most suitable performance metrics for fraud detection. Second, we create and evaluate a unique learning technique that successfully combats idea drift, class imbalance, and verification delay. Third, we show in our studies how class imbalance and idea drift affect a real-world data stream of more than 75 million transactions that were allowed over a three-year period.

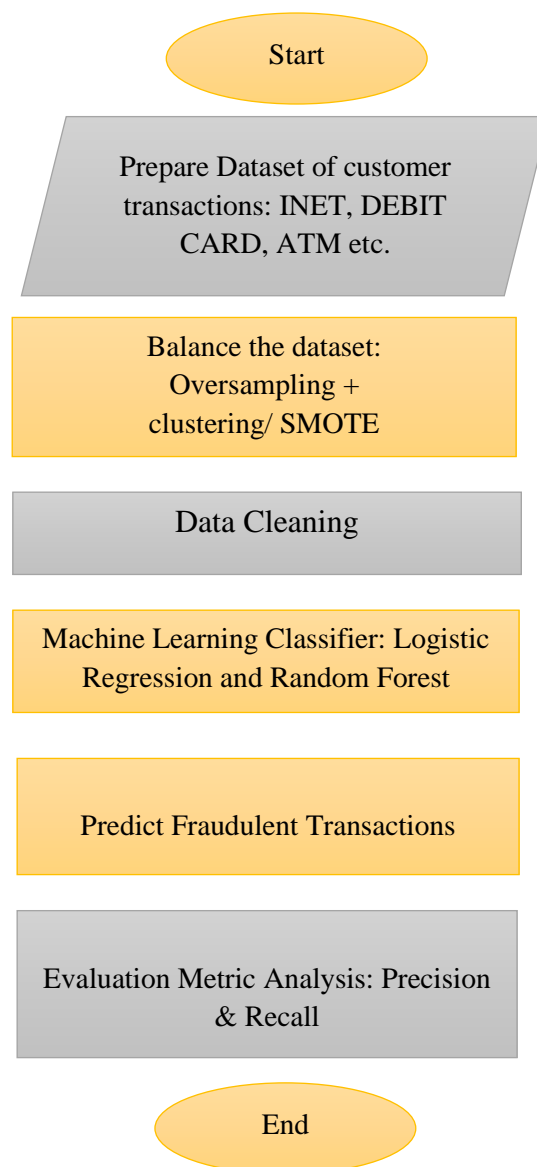
[27] Online class imbalance learning is a developing study area that frequently combines the difficulties of both class imbalance and concept drift. It works well with data streams where concept drift could happen that have highly skewed class distributions. Despite the recent rise in research interest, virtually little has been done to address the combined issue of class imbalance and concept drift. Mentioned research article is the first systematic investigation of managing idea drift in class-imbalanced data streams. After that, a thorough experimental investigation is conducted to determine the most effective way to combat idea drift in online learning when there are unequal numbers of instances in classes.

[31] Today, using a credit card for payment has become increasingly common. The simplest way to pay directly from your bank account is with a credit card. This research has analyzed comparative algorithms to classify fraud. Then they have used KNN for fraud classification. They come up with accuracy of 91.8% on KNN and 77% on C5.

# **Chapter 3**

## **Methodology**

In our proposed model, we use a real-time dataset of debit card and other miscellaneous transactions (including INET and Interbank Transfers via ATMs etc.) belonging to a commercial bank and solve problems related to imbalance data (e.g. customers who don't have much history and certain pattern of transactions) by experimenting with techniques like SMOTE combined with clustering that we form of data classes. After balancing the data we perform data cleaning and analysis of nullified transactions and then implement supervised machine learning algorithm to predict fraudulent transactions. We then analyze precision metric and evaluate results. We define our work in below illustrated steps.



**Figure 6. Dataset Balancing and Fraud Prediction**

### 3.1. Transactions Dataset Formation:

Dataset based on 2 million customer debit card fund transfer transactions over the period of January 2021 to March 2021 had been created. Some of the significant dataset fields are given below.

**Type:** Type of Transaction e.g. Fund Transfer etc.

**Amount:** Transaction Amount

**Name Originator:** Sender name

**Old balance Org:** Sender's old balance

**New balance:** Sender's new balance

**Name Destination:** Receiver's name

**Old balance Destination:** Receiver's old balance

**New balance Destination:** Receiver's new balance

**Is Fraud: Fraudulent flag (0/1)** – Based on immediate cash out transaction after fund transfer.

**Is Flagged Fraud: Fraudulent flag (0/1)** – based on balance exceeding

### 3.2. Dataset Transformation and Fraud Prediction Before SMOTE:

In another experiment, we start with exploring data and transforming it to form correlation matrix. For that, we convert null `is_fraud` field of our dataset into 0's and 1's in order to normalize the data range. Now that we have valid data values for all fields.

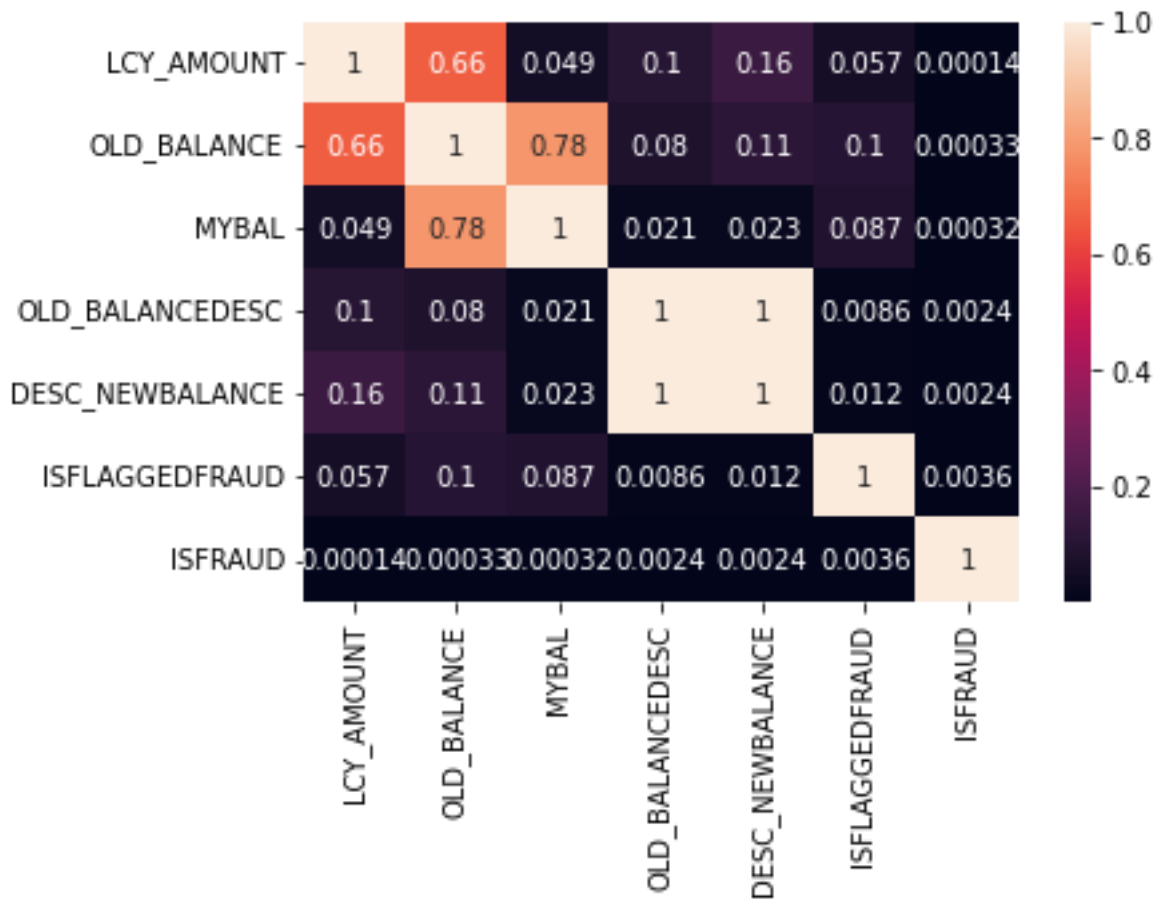
After converting data types we form meaningful data fields we can see it via `fraud_dataset.head()`.

SENDER_AC	REC_AC	TRN_DT	LCY_AMOUNT	OLD_BALANCE	MYBAL	OLD_BALANCEDESC	DESC_NEWBALANCE	ISFLAGGEDFRAUD	ISFRAUD
*3**1*** 74415S	**1*1***4 *3*9YY	2021- 03-30	25,000.000 00000	205,849.630 00000	180,849.630 00000	20,332,662.830 00000	20,357,662.830 00000	0.00000000	0.0 000 000 0
**1*1*** 4*3*9S	*3**1***7 4415YY	2021- 04-13	20,000.000 00000	20,377,662.8 3000000	20,357,662.8 3000000	160,849.63000 000	180,849.63000 000	0.00000000	0.0 000 000 0
**11***1 12792S	**1*1***7 *6*5YY	2021- 05-03	3,000.0000 0000	3,588.50000 000	588.5000000 0	2,550.1900000 0	5,550.1900000 0	0.00000000	0.0 000 000 0
**1*1*** 77*58S	*9**32*2* 862*YY	2021- 02-06	19,000.000 00000	271,123.550 00000	252,123.550 00000	2,679.3000000 0	21,679.300000 00	0.00000000	0.0 000 000 0
*9**32*2 *862*S	**1*1***7 7*58YY	2021- 02-08	145,000.00 000000	161,320.700 00000	16,320.7000 0000	107,123.55000 000	252,123.55000 000	0.00000000	1.0 000 000 0

**Table 7 Dataset Transformation and Normalizing**

To find relationships of fields with one another, we plot correlation matrix to see dependence of variables. In the plot mention below, we highlight figures closer to 1 that show most dependent variables. As we investigate into each highlighted column, we get to know that old balance is an important field since it has highest degree of dependence.



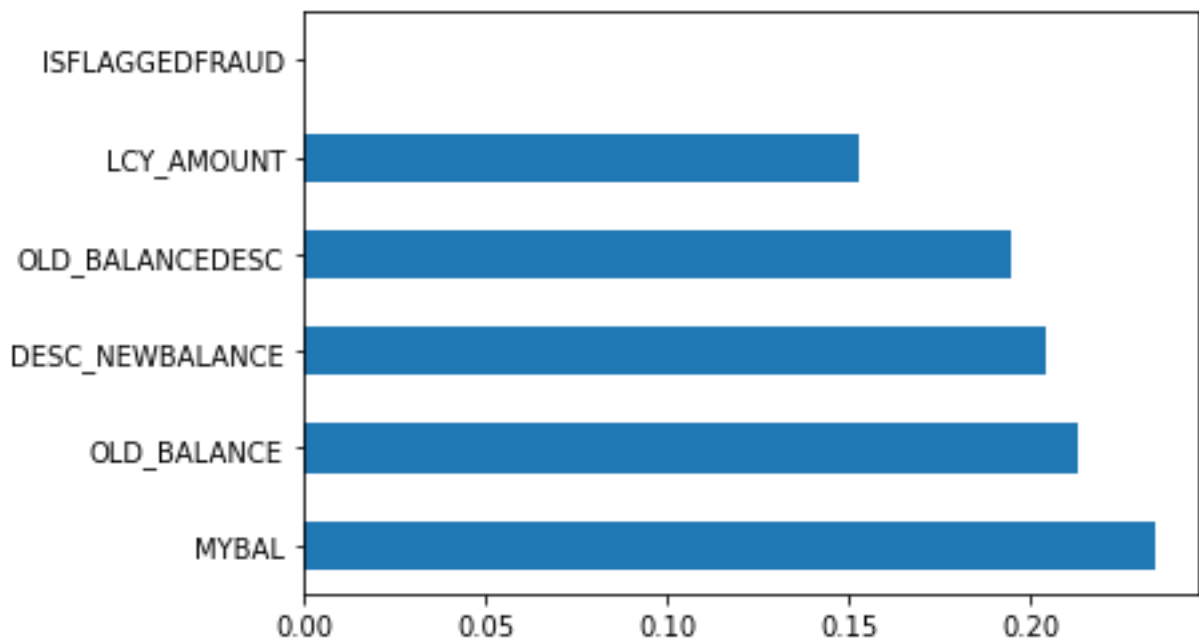


**Figure 7 Dataset Transformation and Correlation Matrix**

To ensure that all the variables fall within the same range, we normalize the data. To overcome the model learning challenge, we normalize the training data. To enable faster gradient descents, we ensure that the different features have comparable value ranges (feature scaling).

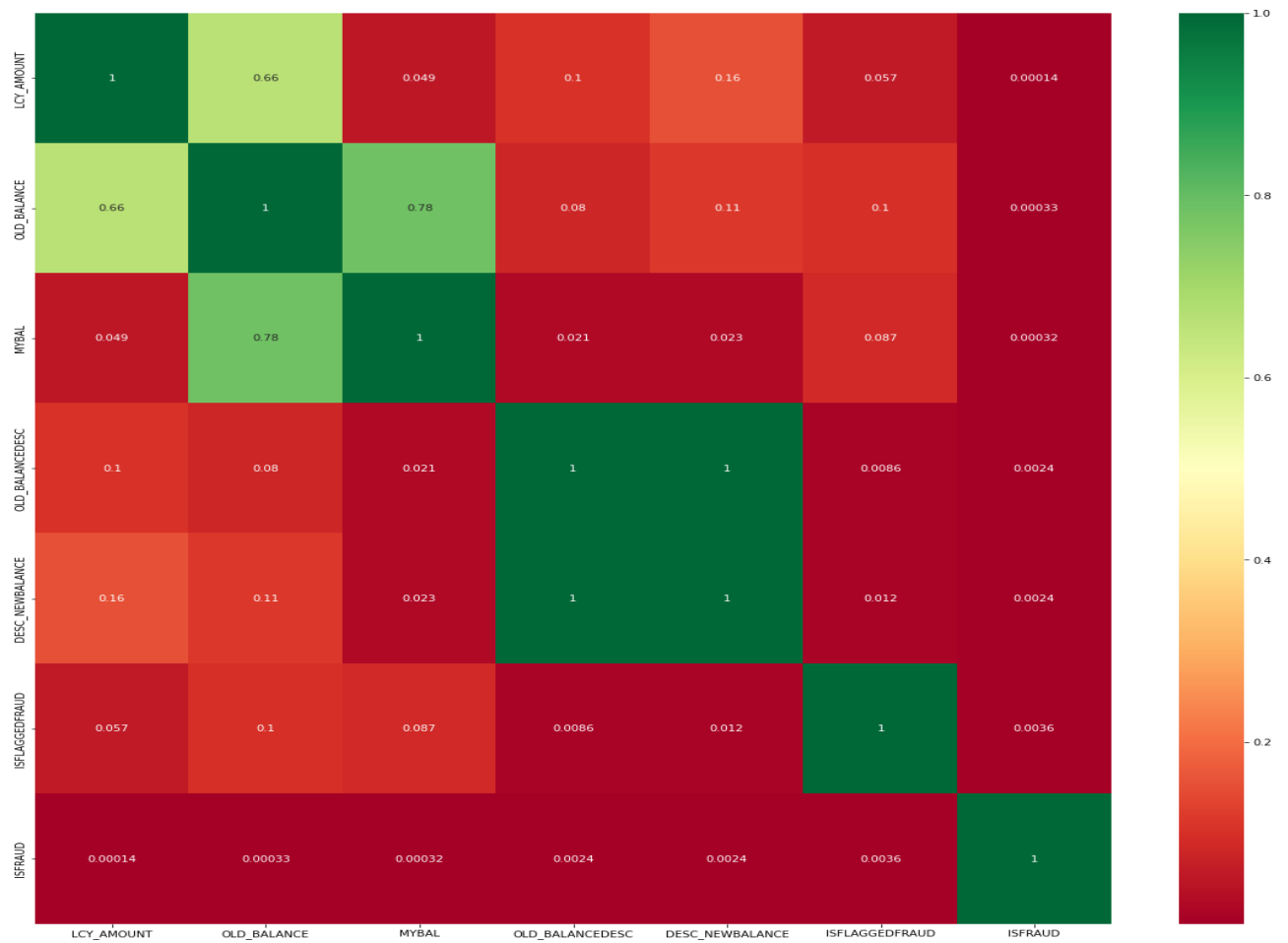
After normalization, specific features are chosen using the *SelectKBest* method based on the k highest score. We can apply the approach to both classification and regression data by modifying the scores option. When we prepare a huge dataset for training, choosing the right features is a crucial process. Resultant scores are mentioned below in grid.

	Specs	Score
0	LCY_AMOUNT	2,326,956.86315614
1	OLD_BALANCE	2,305,168.29438174
2	MYBAL	1,336,425.18399819
3	OLD_BALANCEDESC	2,454,692,934.11462641
4	DESC_NEWBALANCE	2,445,556,820.07628822
5	ISFLAGGEDFRAUD	13.02595115



*Figure 8 Visualization of Dependent Columns*

**x-axis = score of dependency | y-axis dataset fields**



**Figure 9 Heat map featuring correlation matrix**

After Exploration of fraud flag columns we can see that suspicious transactions are 14892 out of a sample of 100000 transactions but actual fraudulent are 16 only.

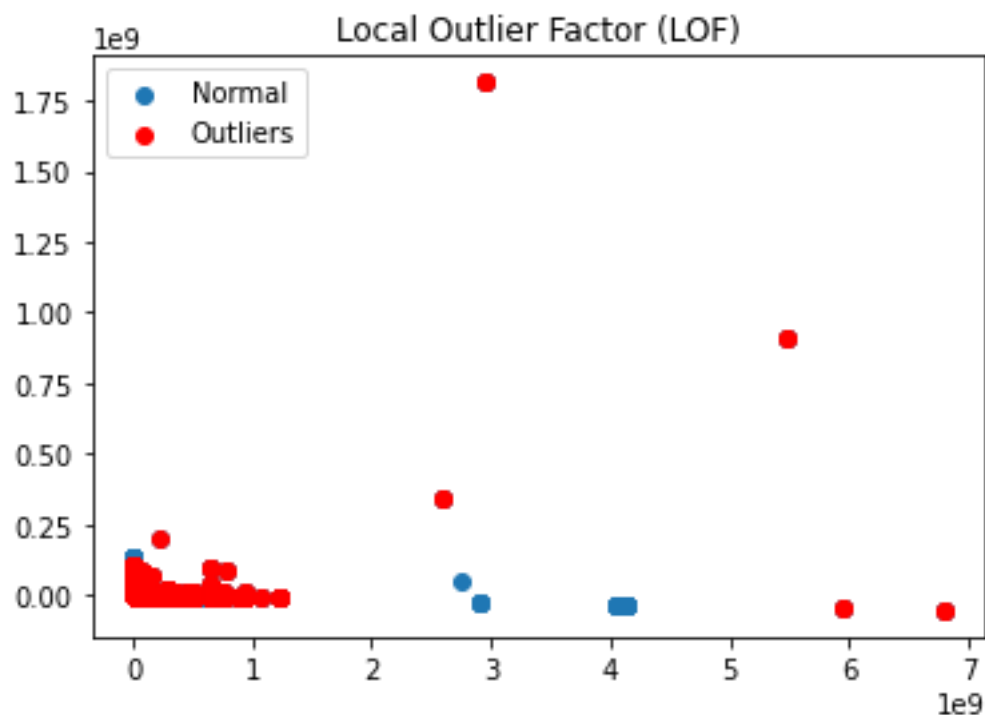
```
dfi['ISFLAGGEDFRAUD'].value_counts()
```

0.0	985104
1.0	14892

```
dfi['ISFRAUD'].value_counts()
```

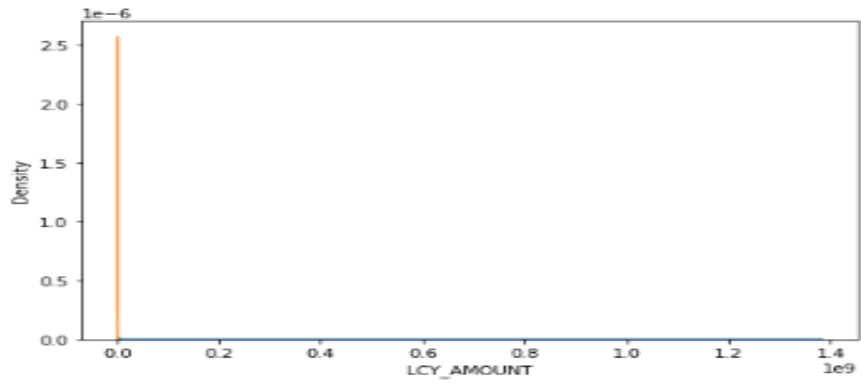
0.0	999980
1.0	16

We evaluate out independent variables by performing Principal Component Analysis and locating outliers on our below pasted graph. We detect outliers in our distribution by using *EllipticEnvelope* class of **Sklearn** Library. A method of depressing the dimensionality of such datasets where improving the understanding is concerned while minimizing information loss is called as Principal Component Analysis (PCA). This is achieved by producing new, uncorrelated sample variables that maximize variance one after the other. An observation that differs abnormally from other values in a population-based random sample is referred to as an **Outlier**. In a way, this definition defers to the analyst's (or a consensus process') judgement as to what constitutes aberrant behavior. It is vital to define typical observations before distinguishing abnormal ones.

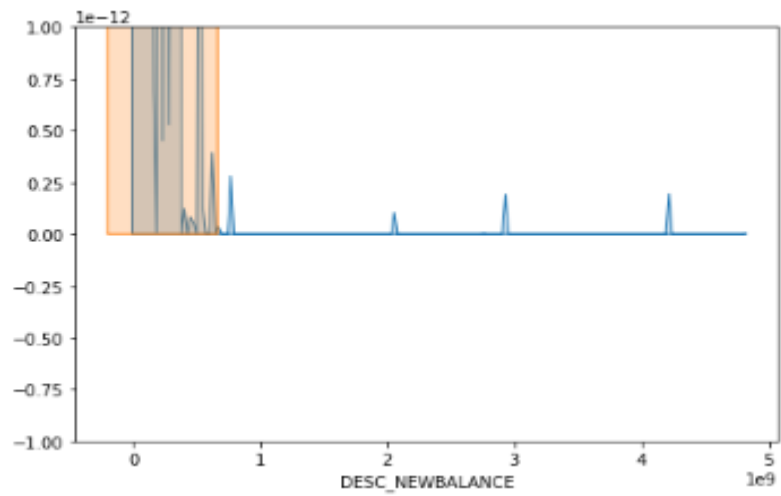


*Figure 10 Principal Component Analysis – Visualization*

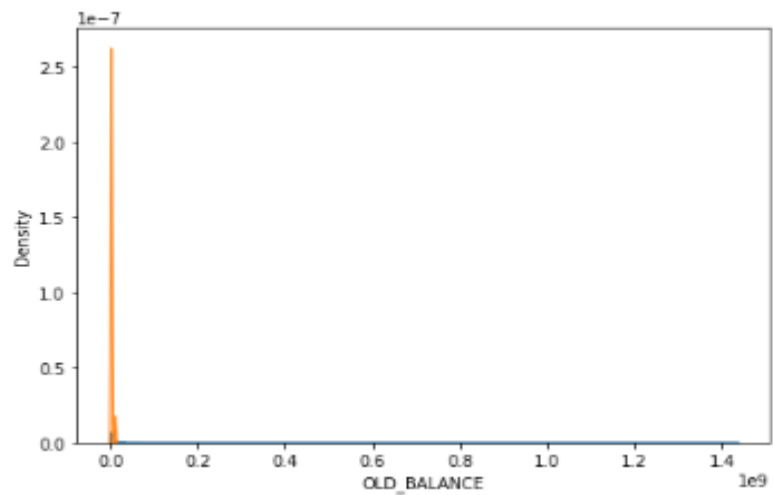
As our dataset was highly imbalance we can see that our data isn't well distributed but highly saturated having more outliers than usual. This highlights the need of data balancing. Before any further experimentation, we visualize below the pre normalization values of each dataset field.



*Graph 1. Exploration of customer amount field*

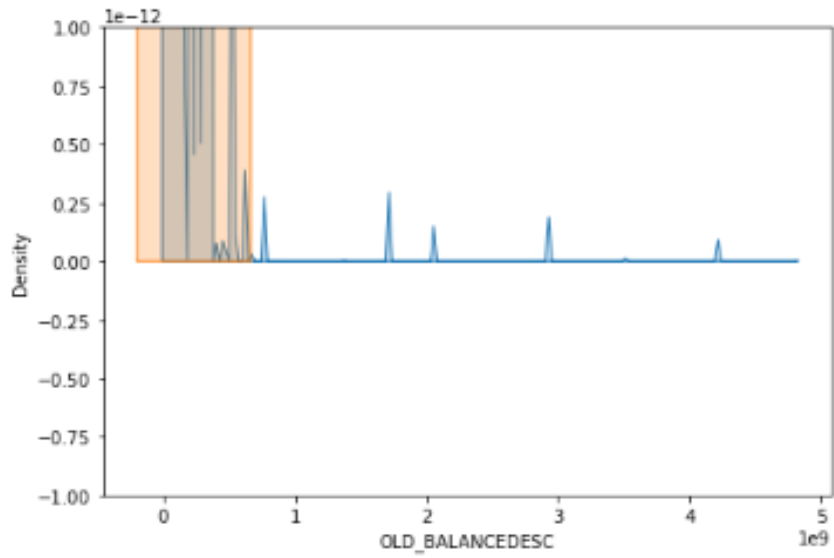


*Graph 2. Exploration of New Balance field*

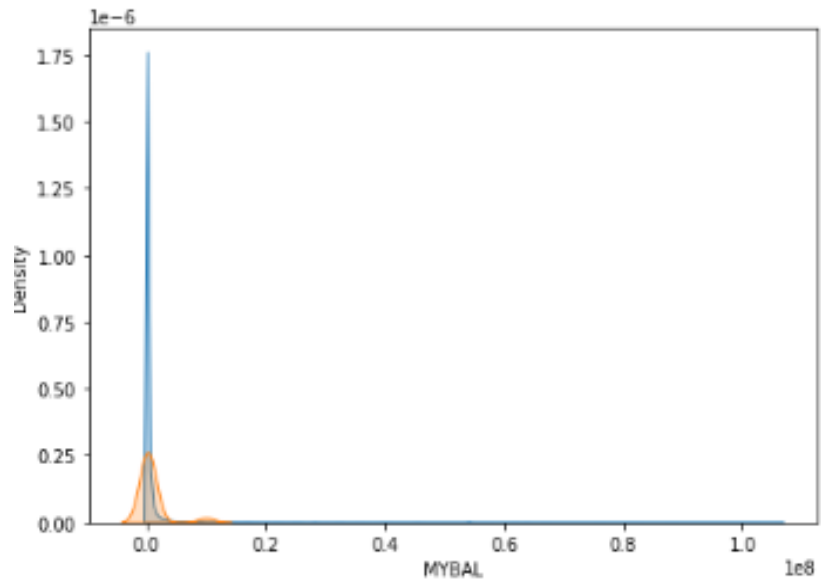


*Graph 3. Exploration of old balance amount field*

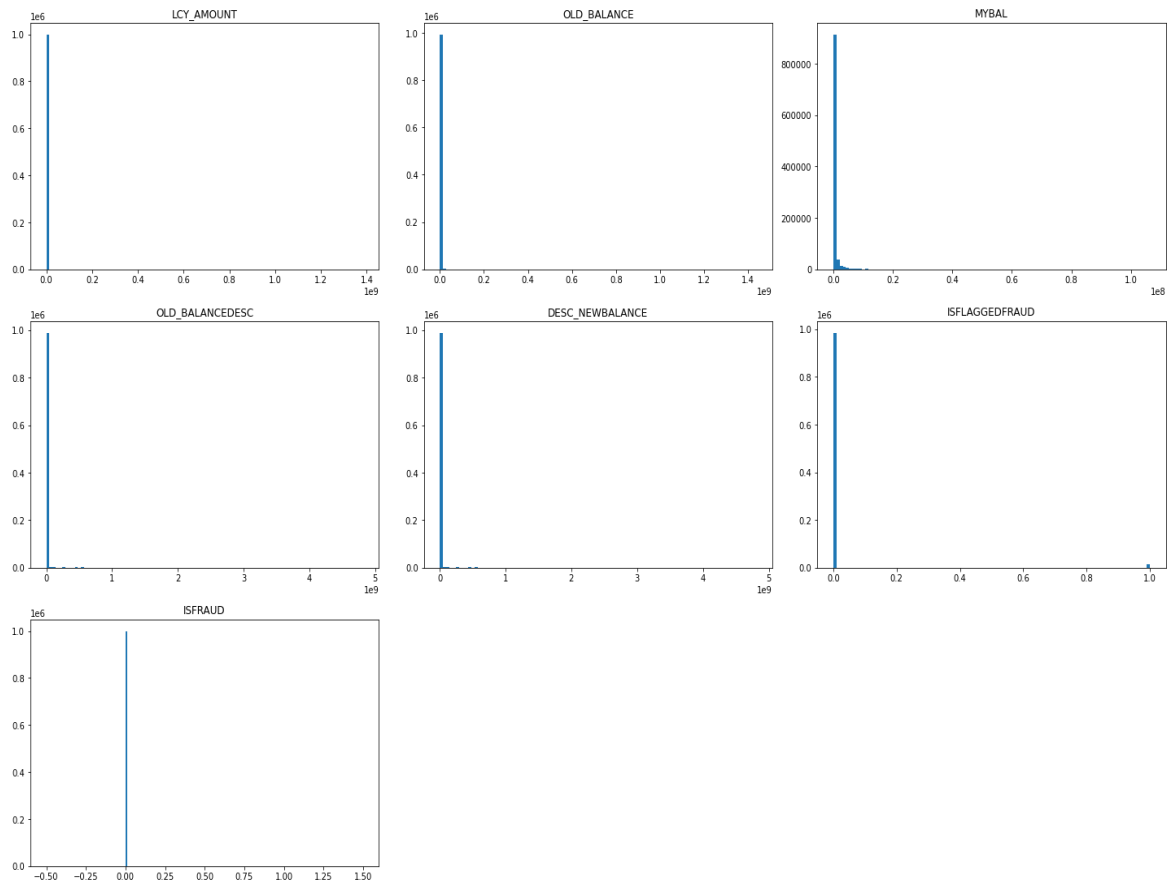
*Graph 3. Exploration of old balance amount field*



*Graph 4. Exploration of old balance description amount field*



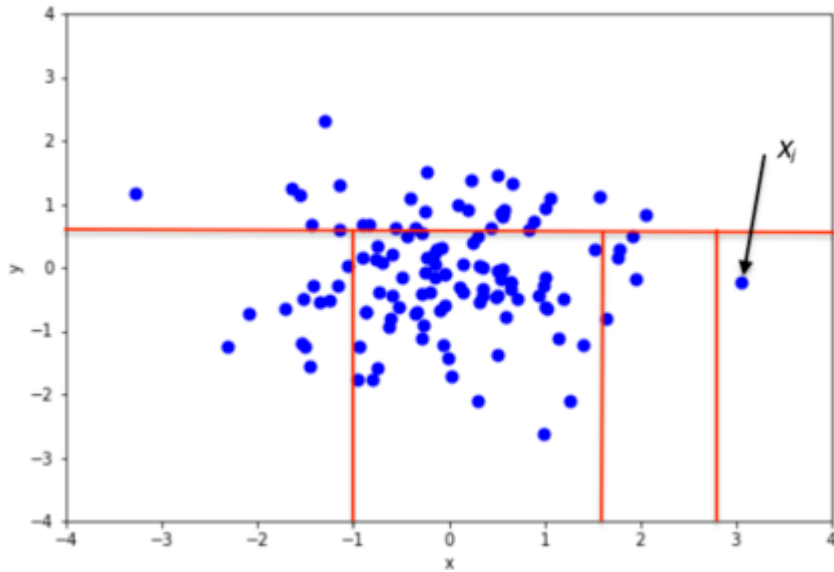
After normalization, all of our columns fall between 0s and 1s as shown in graphs below.



Now we train and test our classifiers after data exploration of imbalance sets to see accuracy and evaluate our confusion matrix.

### 3.2.1. Isolation Forest

An unsupervised approach for anomaly detection based on the idea of isolating anomalies is called the Isolation Forest. It directly isolates anomalous points in the collection instead of attempting to create a model of typical examples. It is an extremely quick algorithm with less memory usage. Attached below example clearly picks odd on out of the verified transactions.



**Figure 11 Isolation Forest – Visualization**

Figure 11 visualizes Isolation forest on the scattered plot having few outliers

### Isolation Forest Hyper parameters

n_estimators	155
max_samples	len(X)
contamination	0.0194
random_state	42
verbose	0

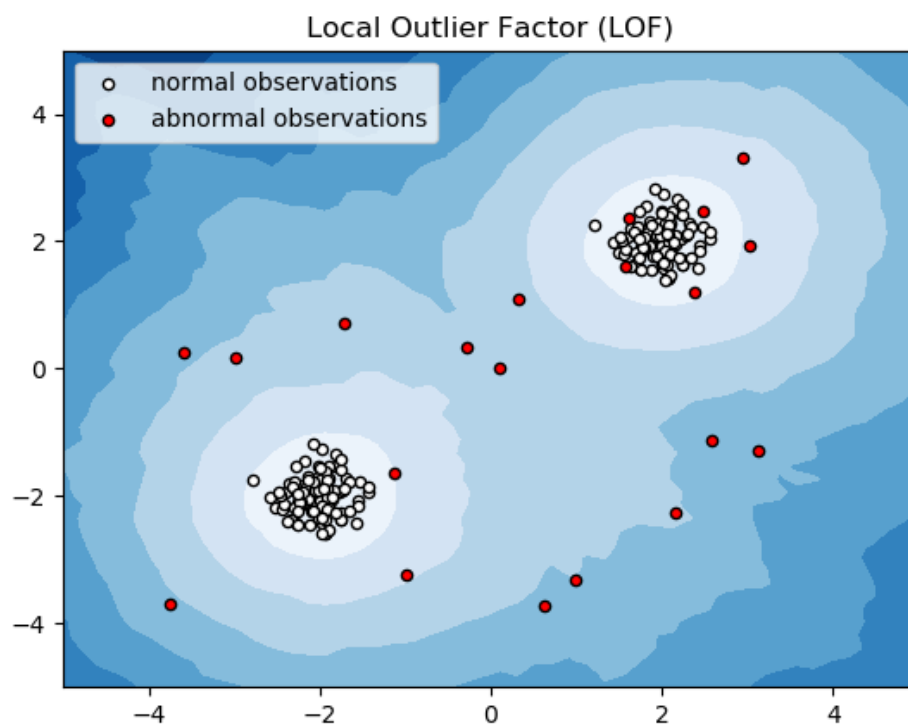
**Table 8 Isolation Forest Results**

Results are discussed in results section

### 3.2.2. Local Outlier Factor Algorithm

The Local Outlier Factor (LOF) as shown in Figure 12, algorithm calculates the local density deviation of a particular data point with respect to its neighbors. It is an unsupervised anomaly identification technique. [18] In an example attached below the samples that have a significantly lower density than their neighbors are regarded as outliers.





*Figure 12 Local Outlier Factor – Visualization*

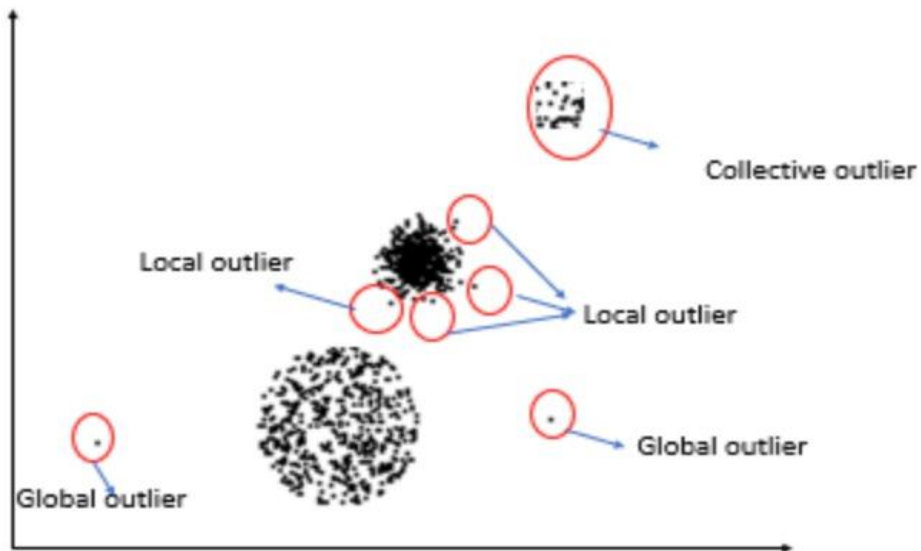
### Local Outlier Factor Hyper parameters

n_neighbors	<b>200</b>
algorithm	Auto
leaf_size	<b>230</b>
metric	'euclidean'
p	<b>1</b>
metric_params	<b>None</b>
contamination	0.0021

*Table 9 Local Outlier Factor Results*

Typically, the argument `n_neighbors` specifies the number of neighbors to take into account. Larger than the minimum number of items a cluster must include to allow for the possibility of additional objects becoming local outliers with respect to this cluster, less than

the maximum number of nearby objects that may also be local outliers. In reality, these details are typically unavailable, and using  $n_{\text{neighbors}}=20$  seems to work well in most cases.



*Figure 13 Local Outlier Factor - Visualization*

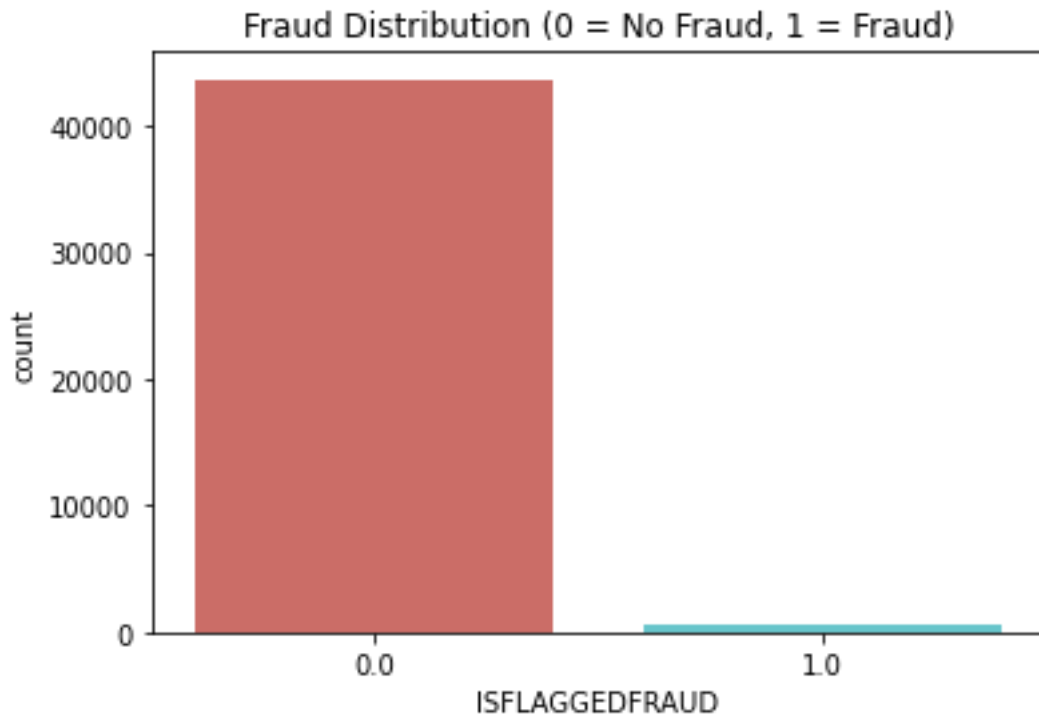
Outlier detection is important to identify unusual data patterns as shown in Figure 13, Local Outlier Factor does not assume the outliers to be a binary property but captures the degree to which the object is isolated from its surrounding neighbors. LOF can be applied to high-dimensional datasets to identify local outliers that may be outlying only on some dimensions of the dataset.

### **3.3. Dataset balancing with SMOTE:**

SMOTE selects examples from the feature space that are closer to [18] one another, draws a line between the examples, and then creates a new sample or cluster at a location along the line. To be more precise, a random representative from the minority class is initially picked. Next,  $k$  nearest neighbors for that example are located. A synthetic example is formed at a randomly chosen position in feature space between two instances and their randomly chosen neighbor. SMOTE has identified the  $k$  closest minority class neighbors of a minority class instance. It has selected by choosing  $k$  at random.

### 3.3.1 Solving the problem of imbalance dataset:

The most common cause of data unbalancing is an unbalanced distribution of classes within a dataset. In our debit card fraud dataset, a vast majority of card transactions are not fraudulent and only a few instances are fraud. As a result, the ratio between the fraud and non-fraud classes is roughly 50:1. We take out fifty thousand samples to perform Exploratory Data Analysis (EDA). Around 43689 transactions are verified ones as shown in figure below.



*Figure. 13a Distribution of Both Classes on Curve*

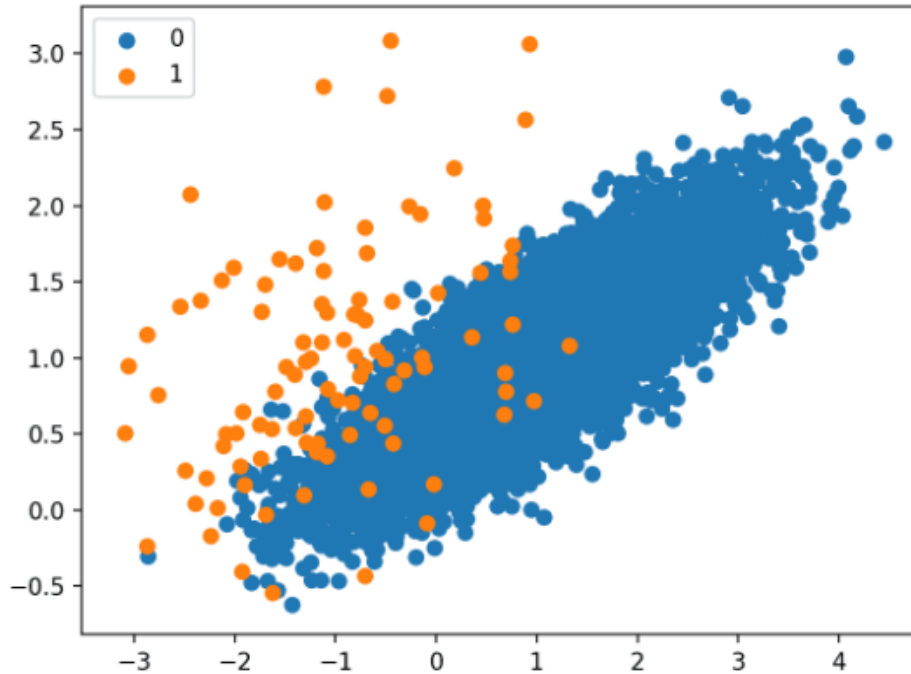
We group our classes in to fraud and verified by using grouping and mean() functions.

Resultantly, we divide:

	LCY_AMOUNT	OLD_BALANCE	MYBAL	OLD_BALANCEDESC	DESC_NEWBALANCE
ISFLAGGEDFRAUD					
0.0	2.534136e+04	5.393586e+05	5.140173e+05	4.079998e+06	4.105340e+06
1.0	1.062901e+06	3.492926e+06	2.429959e+06	7.005606e+06	8.069437e+06

After Data Exploration we have seen that due to lack of instances in minority class our classifier will result in overfitting, so we have implemented oversampling using Synthetic Minority Oversampling Technique called SMOTE to balance our dataset. It is a popular oversampling technique to solve imbalance data problem. It resamples the data by randomly increasing minority class examples. It tries to replicate existing examples of minority class and aims to balance class distribution. SMOTE aims to create new instances by randomly selecting on one or more K-nearest neighbors for each example in the minority class. When we reconstruct our data, it can serve as a balanced dataset and classification models can be applied on it for fraud detection.

To perform SMOTE, we start with importing SMOTE module from library: imblearn. We synthesize data from the minority class rather than merely replicating data from the minority class. This is a sort of data augmentation that can be particularly useful for tabular data. The Synthetic Minority Oversampling Technique, or SMOTE for short, is a method for synthesizing data for technical processing. In figures 14 and 14a, we have visualized how SMOTE has scattered the instances of classes on plot.



**Figure 14** Results of SMOTE are discussed in results section

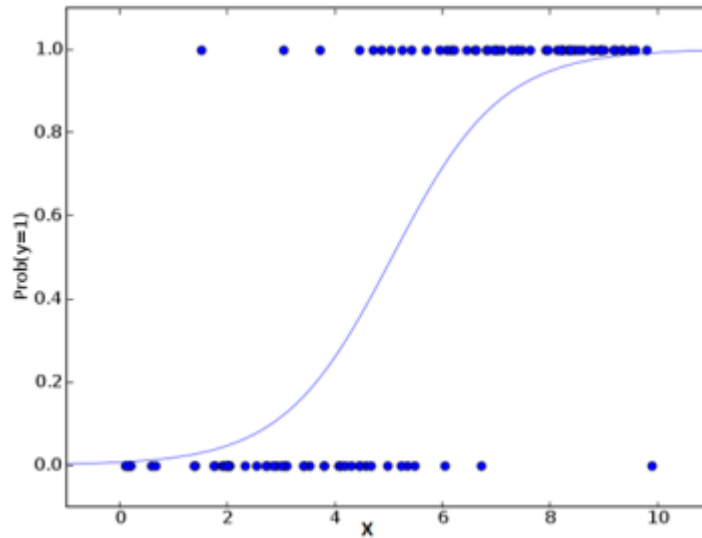


*Figure 14 a Results of SMOTE are discussed in results section*

### **3.4 Implementing Classifiers on balanced dataset:**

In our first experiment, we treated this thesis problem as Anomaly Detection Problem. Now after dataset balancing, we treat it as a classification problem. We tend to use Logic Regression to achieve optimal results. Logistic Regression is a classification-oriented procedure. For a set of independent variables, it is used to predict a binary outcome i.e. yes or no or either 0 or 1. Dummy variables are used to characterize outcomes. In simple terms, it fits data to a logit function to forecast the probability of an event that is occurring.

In our case after data balancing in figure.15, possibility of occurrence of fraud transactions is now equivalent to verified one as given below.



**Figure 15** The balanced data curve

The logistic regression classification algorithm is used to attribute observations to a different group of groups. Spam or unsolicited email, fraud or non-fraud in online transactions, and malignant or benign tumors are some examples of labeling problems. The classifier passes the weighted combination of the input characteristics to a sigmoid function. Any real number can be converted to a number between 0 and 1 using the sigmoid algorithm. In the text data, the logistic regression was found to be very accurate, and the underlying algorithm is also reasonably simple to understand. More specifically, in the field of natural language processing, logistic regression is widely considered a good first algorithm for classifying text. We apply Logistic Regression by following hyper parameters:

MODEL	HYPER PARAMETERS
Logistic Regression	(C=1.0, class_weight=None, dual=False, fit_intercept=True, intercept_scaling=1, l1_ratio=None, max_iter=100, multi_class='auto', n_jobs=None, penalty='l2', random_state=None, solver='lbfgs', tol=0.0001, verbose=0, warm_start=False)

**Table 10** Logistic Regression and Parameter Tuning

We discuss parameter tuning and results in the next section

# **Chapter 4**

## **Results and Discussion**

Before balancing we implement Isolation forest, local outlier factor and one class SVM as mentioned below in pseudo code:

<b>MODEL</b>	<b>HYPER PARAMETERS</b>
<b>Isolation Forest</b>	Isolation Forest (estimators=155, maxsamples=len(X), contamination=.00194,random_state=42, verbose=0)
<b>Local Outlier Factor</b>	Local Outlier Factor (neighbors=200, algorithm='auto', leafsize=230, metric='euclidean', p=1, metricparams=None, contamination=.0021)
<b>One Class SVM</b>	One Class SVM (kernel='rbf', degree=3, gamma = 'auto', nu=0.00215, max_iter=-1)

*Table 11 Applied Models and Hyperparameters*



#### 4.1. Resultant Confusion Matrices:

```
ISOLATION FOREST number of errors: 391
col_0      0    1
ISFRAUD
0.00000000 199608 388
1.00000000     3    0
silhouette coefficient: 0.9667450261377697 3
Adjusted Rand index  : -2.9712837499311397e-05 3
Classification Report :
              precision    recall  f1-score   support

   0.0         1.00         1.00         1.00    199996
   1.0         0.00         0.00         0.00         3

 accuracy          1.00    199999
 macro avg         0.50         0.50         0.50    199999
 weighted avg      1.00         1.00         1.00    199999
```

```
LOCAL OUTLIER FACTOR number of errors: 423
col_0      0    1
ISFRAUD
0.00000000 199576 420
1.00000000     3    0
silhouette coefficient: 0.9718498203249544 3
Adjusted Rand index  : -2.97254321525199e-05 3
Classification Report :
              precision    recall  f1-score   support

   0.0         1.00         1.00         1.00    199996
   1.0         0.00         0.00         0.00         3

 accuracy          1.00    199999
 macro avg         0.50         0.50         0.50    199999
 weighted avg      1.00         1.00         1.00    199999
```

Before balancing, we treat our thesis problem as anomaly detection problem and we can see that scores are not agreeable. Results for both isolation forest showed that 3 out of 2 million transactions were fraudulent where as actual figure in our data seemed to be 16 in our early data exploration. After SMOTE analysis we tend to balance our data and treat it as classification problem, using Logistic Regression we get a good raise in results and accuracy.

## 4.2. Logistic Regression and Parameters Tuning

MODEL	HYPER PARAMETERS
<b>Logistic Regression</b>	C=1.0, class_weight=None, dual=False, fit_intercept=True, intercept_scaling=1, l1_ratio=1, max_iter=100, multi_class='auto', n_jobs=None, penalty='l2', random_state=None, solver='lbfgs', tol=0.0001, warm_start=False

For problems like transactional fraud detection, recall and precision are best matrices to be evaluated for results [19].

### 4.3. Pre balancing results on dataset [32]:

Accuracy: 0.99

Precision: 0.35

Recall: 0.44

f1-score: 0.39

Before balancing the dataset using SMOTE our accuracy was 99% which is an impact of overfitting but our precision and recall was less. **The accuracy came out to be around 100% as we noticed that** the recall of the minority class in way fewer. It proves that our model has resulted to be biased towards majority class and there is need for data balancing. Here's why:

- Our algorithm tends to be biased towards the majority class and thus tend to predict output as the majority class.
- Minority of class observations seemed as noise to the model and have been ignored
- Imbalanced dataset resulted in misleading accuracy score

Hence, this is not the best way to detect fraud. Later, we applied **imbalanced data handling technique SMOTE** and witnessed that accuracy and recall results were improved.

#### **4.3.1. Post balancing results on dataset (Logistic Regression):**

Accuracy: 0.58

Precision: 0.77

Recall: 0.58

f1-score: 0.50

#### **4.4. After SMOTE Over sampling:**

Target class 'No fraud' = 788117 records

- Target class 'Fraud' = 788117 records
- Patterns are not lost which has enhanced our model performance.

Average precision score on Random Forest Classifier: 0.7761626595788199

Recall improved to 0.58

Accuracy on Logistic Regression: 78%

Following is how the SMOTE algorithm operates:

#### **4.5. How SMOTE worked?**

One member of the minority group is chosen at random. It is to figure out who the k nearest neighbors are for each observation in this sample. After that, it works out the vector between the current data point and one of those neighbors using that neighbor. A random value between 0 and 1 is applied to the vector as a multiplier. This is combined with the current data point to create the synthetic data point. Similarly, to shifting a data point slightly in the direction of a neighbor, this method moves the data point. It makes sure that your artificial data point is not an exact clone of an existing data point and that it is not too different from known observations in your minority class.

#### **4.6. Creation of balanced dataset via SMOTE**

- For each sample, determine the k-nearest neighbors.
- Choose random samples from a k-nearest neighbor.
- Calculate the new samples using the formula: new samples = original samples + difference \* gap (0,1).

- Expand the minority with fresh samples. A new dataset has been prepared for further experimentation.

## **5. Recommendations**

Results can be improved by taking transactions of longer period with increased probability of fraudulent transactions. Furthermore, one can always experiment with more balancing techniques like under sampling. For extensive research, innovative algorithms can be designed to perform predictions on imbalance datasets.

## 6. Conclusion

In recent decades, researchers have been experimenting with modern techniques in Artificial Intelligence related domains like Machine Learning, Data Mining and Genetic Programming. Most of the work done in fraud detection has been implemented on Credit Card Transactions taken from several institutions who share their data voluntarily or as a research participator to perform experimentation and fraud predictions. The use of online credit/debit card transactions looks to be increasing as the internet and e-commerce increase. Increased use of credit and debit cards has resulted in a rise in fraud. In this thesis, we created a transaction dataset from scratch containing transactions from March 2021 to May 2021. Since our initial data is raw and highly imbalance, we experiment with data transformation and machine learning techniques to detect fraudulent transactions. After comparison of multiple methods, we share our results and conclude that balanced dataset is the key to achieve highest accuracy on applied classifiers. We achieve 78% accuracy after SMOTE analysis on dataset that was 28% more than that of imbalance dataset i.e. 50% in first few trials. Before balancing the dataset using SMOTE our accuracy was 99% which is an impact of overfitting but our precision and recall was less. **The accuracy came out to be almost 100% as we noticed that** the recall of the minority class in way fewer. It proves that our model has resulted to be biased towards majority class and there is need for data balancing. Here is why:

- Our algorithm got biased towards the majority class and thus tend to predict output as the majority class.
- Minority class observations looked like noise to the model and have been ignored
- Imbalanced dataset resulted in misleading accuracy score.

In the second experiment, Patterns are not lost which has enhanced our model performance. Average precision score on Random Forest Classifier: 0.7761626595788199 Recall improved to 0.58 and Accuracy on Logistic Regression improved to 88% respectively.

### 6.1 Future Scope

In the future, more work should be done on dataset balancing apart from SMOTE analysis. Innovative methods must be incorporated to existing research to increase minority class instances that will eventually improve algorithm accuracy on detection and prediction of fraud.

## References

- [1] M. Mary, "Online Transaction Fraud Detection System," 2021.
- [2] G. A. Pradipta, "SMOTE for Handling Imbalanced Data Problem : A Review," in *International Conference on Informatics and Computing (ICIC)*, 2021.
- [3] Z. Li, "A hybrid method with dynamic weighted entropy for handling the problem of class imbalance with overlap in credit card fraud detection," *Elsevier: Expert Systems With Applications*, 2021.
- [4] R. H. Giulia Moschini, "Anomaly and Fraud Detection in Credit Card Transactions Using the ARIMA Mode," *MDPI*, 2021.
- [5] Z. L. Longfei Li, "A Time Attention based Fraud Transaction Detection Framework," *Ant Financial Services Group, Hangzhou, China* , 2020.
- [6] M. G. Johannes Jurgovsky, "Sequence classification for credit-card fraud detection," January 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0957417418300435#!>.
- [7] S. M. S. Askari, " Intuitionistic fuzzy logic based decision tree for E-transactional fraud detection," *Journal of Information Security and Applications*, 2020.
- [8] G. Vaishnavi Nath Dornadulaa, "Credit Card Fraud Detection using Machine Learning Algorithms," *Science Direct - Procedia Computer Science*.
- [9] D. C. Y. T. Kang Fu, "Credit Card Fraud Detection Using Convolutional Neural Networks," 2016.
- [10] S. N. Hyder John, "Credit Card Fraud Detection using Local Outlier Factor and Isolation Forest," 2019.
- [11] H.-p. K. R. T. N. a. J. S. M. Breunig, "LOF: Identifying Density-Based Local Outliers," Dalles, 2000.
- [12] V. Shpyrko, "Fraud detection models and payment transactions analysis," *SHS Web of Conferences 65, 02002 (2019)*, 2019.
- [13] P. Wibowo, "An in-depth performance analysis of the oversampling," *semantic scholar*, 2021.

- [14] sciencedirect, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0065245820300851>.
- [15] A. Gupta, "Financial fraud detection using naive bayes algorithm in highly imbalance dataset," *Journal of Discrete Mathematical Sciences and Cryptography*, 2021.
- [16] E. L. Jakob Brandt, "diva.org," 2020. [Online]. Available: <https://www.diva-portal.org/smash/get/diva2:1519153/FULLTEXT01.pdf>.
- [17] N. Manchev, 2021. [Online]. Available: <https://www.dominodatalab.com/blog/smote-oversampling-technique>.
- [18] scikit-learn, "Anomaly detection with Local Outlier Factor (LOF)," [Online]. Available: [https://scikit-learn.org/0.19/auto\\_examples/neighbors/plot\\_lof.html](https://scikit-learn.org/0.19/auto_examples/neighbors/plot_lof.html).
- [19] "guide-precision-recall-confusion-matrix.," 2020. [Online]. Available: <https://www.kdnuggets.com/2020/01/guide-precision-recall-confusion-matrix.html>.
- [20] K. F. C. T. Zhang, "Credit Card Fraud Detection Using Convolutional Neural Networks," [Online]. Available: [https://link.springer.com/chapter/10.1007/978-3-319-46675-0\\_53#citeas](https://link.springer.com/chapter/10.1007/978-3-319-46675-0_53#citeas).
- [21] Keras (2022). Adamax optimizer. <https://keras.io/api/optimizers/adamax/>.
- [23] Perceptions of Energy Resources Efficiency for Sustainable Development in the Developing Context of Nigeria: Implications for Enterprise Development in the Energy Sector. P. 184.
- [24] <https://www.sciencedirect.com/science/article/abs/pii/S0957417418300435#!>.
- [25] S. N. Hyder John, "Credit Card Fraud Detection using Local Outlier Factor and Isolation Forest," 2019.
- [26] <https://ieeexplore.ieee.org/abstract/document/8038008>
- [27] <https://www.semanticscholar.org/paper/Supervised-Machine-Learning-Algorithms-for-Credit-A-Dhankhad-Mohammed/43f76c30869059039a097e921dfab0a10df7abaf>
- [28] C. Alippi, G. Boracchi, and M. Roveri, "Hierarchical change-detection tests," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 246–258, Feb. 2016.

[29] S. Pan, J. Wu, X. Zhu, and C. Zhang, "Graph ensemble boosting for imbalanced noisy graph stream classification," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 954–968, May 2015. [88] J. Gao, B.

[30] Ding, W. Fan, J. Han, and P. S. Yu, "Classifying data streams with skewed class distributions and concept drifts," *IEEE Internet Comput.*, vol. 12, no. 6, pp. 37–49, Nov. 2008.

[31] Rohilla, Ankur. "Comparative Analysis of Various Classification Algorithms in the Case of Fraud Detection." *International Journal of Engineering Research & Technology* 6, no. 09, 2017.

[32] <https://towardsdatascience.com/precision-vs-recall-evaluating-model-performance-in-credit-card-fraud-detection-bb24958b2723>