**ONLINE PAYMENTS FRAUD DETECTION USING MACHINE LEARNING**

# SRI VENKATESWARA COLLEGE OF ENGINEERING AND TECHNOLOGY

Marri yahmitha 3rd btech 2nd sem

Manthina rajarishika 3rd btech 2nd sem

Kuttagula safa 3rd btech 2nd sem

# Abstract

Fraudulent online transactions have caused signi cant damage and loss to individuals and companies over a period of time. There has been an increase in online fraud with the progression of state-of-the-art technologies and worldwide communication. The design of e cient fraud detection algorithms is critical for reducing these losses. Machine learning and statistical techniques play a vital role in the detection of fraudulent transactions. Fraud detection model implementation is particularly challenging due to the lack of data, sensitive nature of data, and the unbalanced class distributions. It is tough to draw inferences and build better models due to the con dentiality of the records. In this paper we aim to focus on the problems: i) the role of sampling in the presence of class imbalance (i.e., nonfraudulent transactions are more in the percentage of the total transactions), ii) building and analyzing various machine learning models, iii) to assess and validate the performances of different fraud detection techniques. This paper has research directions toward applying machine learning for data analysis. We have designed and assessed a prototype of a fraudulent transactions detection system that will be able to meet real-world demands and increase the security of transactions for the customers.

To address the above-mentioned challenges, this research has employed various techniques to handle the problems in online fraud detection. The following are the major contributions.

The main objective of this research is to build statistical machine learning models to detect fraudulent transactions in real-time with great accuracy.

To come up with various statistical techniques to deal with the imbalanced nature of the data. Like synthetic scaling of the data, which can decrease the skew without affecting the data integrity. This can help to improve the accuracy of the models.

To build an application that can detect the legitimacy of the transaction in real-time and increase the security to prevent fraud.

**1.INTRODUCTION**

Online transactions have become inevitable due to the ease of online purchases. To balance their busy schedules, people stick to online shopping. It enables us to trade any things available in different geographic regions. Due to advancements in e-commerce, people bene t from offers because they are attracted to online shopping. Though online shopping has enabled easy transactions, fraudulent transactions are also possible [1]. Fraudulent online transactions have caused signi cant damage and loss to individuals and companies over a period. There has been an increase in online fraud with the progression of state-of-the-art technologies and worldwide communication. The development of new technologies paves new ways for criminals to commit fraud like credit card frauds, e-commerce frauds, online transaction frauds, etc. Figure 1 presents the estimated fraudulent transaction loss by 2024[1]. According to the Associate of Certi ed Fraud Examiners estimates, the fraud costs the organization nearly $3.7trillion a year, i.e., a company typically loses ve percent of its revenue due to fraud. As shown in Fig. 1 it is estimated that online payment fraud is growing annually and may amount to $50 billion by 2024 which was $26 billion in 2019 [1]. It has become the need of the hour to address the issue of online transaction fraud.

Online fraud is a term used for referring to theft and fraud committed using any kind of payment method like cards, online transactions, etc. Different types of online fraud are as follows (shown in Fig. 2):

1. Clean: It is a type of fraud where the fraudulent can pass through the merchant's checks as he pretends to be a legitimate user.
2. Account Takeover: A fraudulent connects his credit card to an account of a legitimate user.
3. Friendly: A Legitimate user denies the transaction made and the merchant has chargeback. It is also called Chargeback fraud.
4. Identity: Acquisition of personal sensitive details like passport or account number by impersonation is called Identity fraud.
5. A liate: The use of company details for individual purposes online is called A liate Fraud.

. Re-shipping: Use of recruited person or mule for re-shipping products purchased using stolen credit cards.

7. Botnets: A machine or Robot impersonates the geographical location of a legitimate credit card and performs transactions online. Due to the same geographical location, the transaction seems to be legitimate.

. Phishing: The fraudulent gathers sensitive information from authenticated users by sending seemingly o cial emails.

1. Whaling: Similar to phishing except that the target is a xed set of consumers belonging to a pro table online rm.
2. Pharming: When performing online shopping customers are redirected to unauthorized websites where the user’s sensitive information may be gathered.
3. Triangulation: Credit card information is gathered from authenticated users through third-party online auctions or ticketing sites.

Credit card fraud is a type of Identity fraud. The mechanism used in identity fraud or Identity theft is impersonation. An authenticated user tries to impersonate an existing or non-existing user in Identity theft. Credit card frauds are considered to contribute a major part in frauds as they are easy due to transactions online and physical absence online. [1]. Fraudsters have their mechanisms for obtaining credit card details and the transactions are as though they are legitimate transactions. Transactions are classi ed into Card present Transactions (CP) and Card Not Present (CNP). A transaction is of the former type if the transaction is done in person at the time of sales. A transaction is of a later type when neither the card nor the cardholder is physically available during the transaction. The most common scenario of CNP transactions are:

1. Online Transactions: Cardholder and Card is in the same location and the merchant is elsewhere.

Based on authentication veri cation, the purchase is sanctioned.

1. Purchase using Telephonic conversations: Similar to the previous method, merchants and cards are at different locations. The telephone number from which the call is made enables security checks.
2. Mail Orders: As with earlier types of the purchase order is accepted through the mail. A digital signature enables hassle-free transactions.
3. Subscription Payments: A customer pays for Mobile or TV etc. subscriptions. In the earlier method, similar orders were placed. It differs from the earlier case where the customer interaction does not occur at all during purchase.
4. Cloud Wallet Payment: This is an option that is widely used in the present era. In this type of transaction, mobile phone applications may act as a proxy for cards. The transaction is between the user and the cloud server. Security is the main concern in this type of transaction.

The proposed system works to identify credit card fraud. Researchers have been working in detecting credit card fraud, which accounts for the majority of online frauds. Detection of fraudulent transactions using the traditional condition-based system is inecient and viable for hackers. Hence it is essential to deduce a process that is capable of analyzing tremendous amounts of data which should be adaptive to the tactics used by the fraudsters. When addressing the problem of fraud detection, the rst challenge anyone faces is the imbalance in the dataset. Only 0.01% of transactions that take place around the world are fraudulent, and the rest are legitimate. [1] The dominance of the non-fraudulent class causes a major challenge. It causes an imbalance in detecting the characteristics of fraudulent transactions. In this project, we observe and analyze the various sampling techniques like random under-sampling, random oversampling, and Synthetic Minority Over-sampling Technique (SMOTe). This helps to tackle the imbalanced data problem.

The second part of this paper mainly focuses on building and training various machine learning models, which will help in the classi cation. Some of the models built in this project are logistic regression, decision tree, naïve Bayes, random forest, XGboost, KNN, and arti cial neural networks (ANN). All these models are built, trained, tested, and validated based on various statistical parameters like precision, recall, and F1 score. Once the model is built and analyzed, the best-performing model is selected to be incorporated into the web application, which can predict the nature of the transaction in real-time. The idea is that the web application acquires the data from the ongoing and runs it through the model to determine the nature of the transaction.

Credit card fraud detection is one of the hottest topics in the eld of banking and computer science. Lots of time and money are being spent on the research to come up with a better prediction system. Researchers have done lots of work in the eld of machine learning and have come up with various techniques that incorporate machine learning to detect online frauds [2]. The thesis [1] by Dal Pozzolo Andrea,” Adaptive Machine learning for credit card fraud detection”, gives an overall ideology of how the credit card fraud detection system works. The rst part of this section explains the prediction and classi cation using machine learning. The second part justi es the use of various sampling techniques to improve the performance. The third part focuses on the analysis and performance evaluation of the various ML models. Finally, it shows the implementation of the Fraud Detection System (FDS) in the realtime scenario using web API

# **2 1 CREDIT CARD FROUD DETECTION**

[3–9] discusses the working of various ML techniques. [5] [6] [8] shows the use of simple and primitive machine learning techniques for the binary classi er. [9] Is a comparative study between various machine learning algorithms and neural networks. The main challenge faced in all the mentioned papers is the availability of the dataset. Since the banking information is sensitive, they are hidden and protected which makes it di cult to derive inference from the data. Lots of processing capacity and time are spent on building the models. In [4] the problem with this approach is that random forest always leads to an over tting problem. [14–25] are about the implementation of the best-performing models in the real-time scenario. According to these papers, most the real-time implementation is in the beginning stage. A web application or an API can be used to check the legitimacy of the transaction when it is taking place. It requires additional computational requirements. Also, the dataset is dynamic, so the model must adapt on its own. The main challenge is that the system must be capable of handling millions of transactions taking place every minute. It requires a powerful computing system to handle this process. [26–27] provide online information about credit card fraud and the issues that online transactions face if the problem becomes undetected.

**2 2 IMMBALENCED CLASSIFIER**

The imbalanced nature of the dataset and how it can be handled is the next part of the research. [12] [13] discusses in detail various sampling techniques and how the skewed data set affects the results of the model. The large distribution of non-fraudulent transactions, when compared to fraudulent transactions, is known as skewed distribution. The oversampling method generates or recreates the existing samples to overcome the problem of skewed distribution in data samples. Synthetic cost-based data set overcomes the problem of skewed distribution [12]. [17] Mainly covers the different analysis techniques that can be used to evaluate the models. In [6], various cross-validation techniques are used to validate the models. K-fold CV and leave one out CV are some of the commonly used techniques. Also, this paper covers statistical evaluation tools like the ROC curve, Area under Precision-Recall Curve (AUPRC), and F1 score. These evaluate the classi cation ability of the models. Paper [17] explains the feature extraction used to nd the importance of each variable. Given the imbalanced nature of the data, the results of the model are always skewed towards the major class. So, the AUC score cannot be used to evaluate the models. Other statistical parameters like precision and recall are used.

## **3. Research Contribution**

To address the above-mentioned challenges, this research has employed various techniques to handle the problems in online fraud detection. The following are the major contributions.

The main objective of this research is to build statistical machine learning models to detect fraudulent transactions in real-time with great accuracy.

To come up with various statistical techniques to deal with the imbalanced nature of the data. Like synthetic scaling of the data, which can decrease the skew without affecting the data integrity. This can help to improve the accuracy of the models.

To build an application that can detect the legitimacy of the transaction in real-time and increase the security to prevent fraud.

# 3.1 Challenge and motivation

**Enormous Data**: A large amount of data is processed on a daily basis and the models built must be suitable and fast to detect fraudulent transactions.

**Imbalanced Data** i.e., the fraudulent transaction in the dataset is very low this makes it di cult to identify the frauds.

**Data availability**: The banking data is con dential and hence difficult to obtain.

**Misclassified Data**: hard to catch and report every fraudulent transaction. This causes misclassi cation of the models.

**Adaptive techniques** are found and used against the model by the hackers.

## **4. Dataset Used**

[Dataset][1] https://www.kaggle.com/aherparesh/credit-card-fraud-detection/data

The above dataset has nearly 3500 records of transactions. It has around 400 records of fraudulent transactions. The dataset has 10 predictors and one target. The dataset is cleaned and pre-processed before implementing any machine learning algorithms. Data attributes in the dataset are Average Amount per transaction, Transaction amount, Is declined, Total Number of declines/day, is Foreign Transaction, is

High-Risk Country, Daily\_charge back\_avg\_amt, 6\_month\_avg\_chbk\_amt,6-month\_chbk\_freq and Is Fraud

## 5**. Pre-processing Of Data**

The dataset which was imported has to be cleaned before it is t for building models. The abovementioned credit card data has empty columns and unnecessary elds and missing values which have to be cleaned.

Figure 3 depicts a data set before preprocessing. In Fig. 3 the transaction date is empty, and the fraudulent column has Y and N which cannot be observed by the model. It has to be formatted as 0 and 1.

In Fig. 4 the data is preprocessed. The unnecessary columns which are not helpful in the model building were dropped, the empty columns were removed, and the rows missing values are dropped.

# 5.1 Standardaization

The dataset has to be brought to s normal form for the models to be built properly. This helps in improving the data exploration and data modeling part. Figure 5 presents the Normalized dataset.

**6.Proposed work**

This research proposes a novel technique that focuses on the task of detecting online transaction frauds in real-time. The objective of this work is to design a computationally inexpensive and accurate machine learning model for fraud detection. Various machine learning models are deployed. Three different techniques are deployed to handle the imbalanced data. They are random oversampling, random undersampling, and SMOTe technique. Figure 6 shows the Steps in Credit Card Processing

**6.1.Model Bulding**

Various models have been used to detect fraudulent transactions online. These models are brie y discussed in this section. Since fraud detection is a classi cation problem, our focus is mainly on classi cation models. Also, as we are planning to deal with the imbalanced nature of the data, we intend to build models that handle the minority class.

**6.1.1 Logistic Regression**

Logistic regression is one of the powerful classi cation tools borrowed by machine learning from the eld of statistics. It is a simple yet powerful way to model binomial outcomes with one or more predicting variables. It measures the relationship between the predictor and the target variables by calculating probabilities using a logistic function, which is the cumulative logistic distribution. The vector β = (β1, β2, β3,…, βn) represents the coe cients and X = (X1, X2, X3,…, Xn) represents the multiple predictors in the dataset, and ϵ is the model’s error.

Y = β1 X1 + β2 X2 + β3 X3 + …… + βnXn+ ϵ (Equ. 1)

Equ. 1 represents the primary logistic function. The probability of the classes can be determined by the logarithmic function given below in Equ. 2

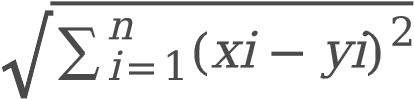
p

l = logb (1−p) = β0 + β1 X1 + β2 X2 (Equ.2)

By simple algebraic manipulation to Equ. 2, we get p, which is the probability of the minority class.

**6.1.2 K Nearest Neighbors(KNN)**

Based on nearest neighbors for a given query point, a non-parametric method KNN is devised. For a given data set, the Euclidean distance is computed between the data set and the available data set. Among the data set, the data points with minimal Euclidian

 (Equ. 3)

Equ. 3 shows the Euclidean distance function.

**6.1.3 Navie Bayes**

## +3.30

## .

P(X1,X2, . . . ,Xn/Y)P(Y)

P(Y/X1, X2,...,Xn) = P(X1,X2, . . . ,Xn) (Equ. 4)

Equ. 5 shows the Bayesian conditional probabilities of each class with respect to the predictors X1, X2, ….,

Xn.

**6.1.4 Decision tree**

A decision tree is a tree data structure that helps to arrive at inference derivation based on conditions to predict the outputs. A class attribute is represented by a leaf node and each intermediate node is split into sub-nodes based on the attribute. The path obtained from Root to leaf is used for forming the classi cation rule. Data are classi ed by using the attributes of each node. It is used for further classi cation and regression models.

**6.1.5 Random Forest**

Just like the Decision tree, the random forest can also be used for both classi cation and regression problems. It is mainly used for classi cation problems. Just like a real forest is made up of many trees, a random forest is nothing but a combination of multiple decision-making trees. Nodes of the random forest are composed of decision trees, each one deriving the prediction from the dataset. It is an ensemble technique consisting of multiple trees. The only problem is the over tting of the model.

Figure 7 presents the working of the Random Forest model.

**6.1.6 SVM**

Support Vector Machine (SVM) is a supervised machine learning algorithm. It can be used to handle both classi cation and regression problems. Just like random forest SVM is mainly used as a classi er. In

SVM, the data points are plotted in a dimensional space, where the coordinate is the value of the variables. Now the classi cation can be performed by nding the hyper-plane. Fig explains the splitting of the data points in a n-dimensional space.

**6.1.7 XGBoost**

XGBoost is an example of the ensemble learning technique. Using just a single model to handle realworld problems is not usually enough. Ensemble learning techniques are collections of models and it is suitable in many scenarios. The resultant output is achieved by the aggregation of several models.

The models above which other models are built are called the base learners. Different algorithms can be integrated with the base model. The commonly used base learners are bagging and boosting. This technique can be used with any model, but it is most commonly used with decision trees.

**6.2 Imbalenced classifier**

Two different approaches are taken to tackle the imbalanced class problem. The rst approach is a data pre-processing technique to handle the classes, like oversampling and under-sampling. The second approach is based on generating new data

* **Under-sampling** is sampling class is points from the major class so that both the classes will be balanced.
* **Oversampling** is adding more data points to the minor class in order to increase their numbers.
* **SMOTE** synthetically generates new data points in the minor class within a bounded region to increase its number.

**6.2.1 Random under sampling**

The Random Under sampling technique balances class distribution by randomly eliminating the majority of class data points. This is done until the major and the minor class data points are balanced. Random under sampling involves randomly selecting data points from the majority class to delete from the training dataset. This has the effect of reducing the number of data points in the major class in the sampled version of the training dataset.

**6.2.2 Random Oversampling**

In this method, the minority class is replicated or new observations are created. Using the new minority class the imbalance may be reduced. This method of replication and re-creation enables the increase in minority data. But the drawback of the method is that it may encounter an over- tting problem as the data are replicated.

**6.2.3 SMOTe**

SMOTe is an oversampling technique that arti cially generates data points in the minority class to increase the number and balance the dataset. This technique works by choosing a random point from the minority class and making a bounded region inside the minority class now the data points are generated within this region. In this way, we can be sure that the integrity of the dataset is not compromised by these additional instances.

In this research, we primarily relied on SMOTe technique to handle the imbalanced nature of the data. All the models are subjected to SMOTe to see the improvement in their performance.

**7.Results**

The results and accuracies of models are compiled together to perform the evaluation of different models under different conditions. We also analyze the other statistical performance of the models to see how well the models perform the classi cation. Because of the skewed nature of the data, it is important to check the Area under the Precision-Recall Curve (AUPRC) because the accuracy parameter is not reliable

**7.1 Accuracy,Precision,Recall,and F1 score**

To measure Accuracy, Precision, Recall, and F1 score, the following parameters are needed.

TP = True Positive. Fraudulent transactions. The model is predicted as fraudulent.

TN = True Negative. Legitimate transactions. The model is predicted as Legitimate.

FP = False Positive. Legitimate transactions. The model is predicted as fraudulent.

FN = False Negative. Fraudulent transactions. The model is predicted as Legitimate.

**Accuracy**

TP+TN

Accuracy = TP+TN+FP+FN (Equ. 5)

Equ 5 gives the accuracy of the model which depicts how many data points are correctly classi ed. Precision

TP

Recall =TP+FP (Equ. 6)

Equ 6 gives the precision of the model. It answers the question “out of all transactions that are predicted to be fraudulent, what percent are actually fraudulent”.

**Recall**

TP

Recall =TP+FN (Equ. 7)

Equ 7 gives the recall of the model. It basically answers the question “out of all the fraudulent transactions how many are correctly classi ed by the model”.

**F1 Score**

Recall∗Precision

F1Score =2 \*Recall+Precision (Equ. 8)

Equ. 8 gives the F1 score of the model. F1 score is the harmonic mean of the recall and precision. F1 score takes both false positives and false negatives into account. In an imbalanced data F1 score is more effective than the accuracy

# 7.2. Model Perform~~a~~nce

The Confusion Matrix of the applied methods is given in Fig. 8. The F1 score, precision, recall, and AUC of each model are calculated to analyze the performance of the models. All these data are tabulated below. The best-performing model is used to build the fraud predicting Application. The below table gives the nal performance of the model after the SMOTe technique is deployed.

Table 1

Model Performance for a small sample data set

|  |
| --- |
| Models Accuracy Precision Recall F1-score |
| Logistic Regression 0.985 0.970 0.923 0.946 |
| KNN 0.977 0.948 0.885 0.916 |
| Naive Bayes 0.951 0.840 0.800 0.819 |
| Decision tree 0.924 0.960 0.466 0.628 |
| SVM 0.992 0.990 0.952 0.970 |
| Random forest 0.985 0.960 0.933 0.947 |
| XGBoost 0.984 0.951 0.933 0.942 |

According to Table 1, the highest F1 score is provided by the SVM model and closely followed by the random forest. This signi es that the best classi cation is provided by the SVM with a F1 score of 97%.

So, the SVM model is used in the application to detect frauds.

Figure 9 shows the ROC curve is plotted for all the models built to analyze the performance of the models. From the curves, we can infer that the SVM classi er has the maximum AUC (Area under the Curve). The AUC score of the SVM is 0.975.

In order to run some more comparative analyses on the model, the model has been trained using a larger dataset with nearly 22500 records. This will give an insight into how the model performs once the transaction is increased and the dataset skewed. The dataset is extended to 22500 transactions. The model's new metrics are observed so as to compare them with the previous results of the model. A sample screenshot for the Random Forest model is available in Fig. 10. It was observed that when data sets are increased, Logistic regression fails to achieve F1 score when compared to the other models. It is because the Random Forest Model exhibits a high F1 score of 0.978. The ROC curve is plotted for the models and is available in Fig. 12.

Although Logistic regression is a relatively simple model and will underperform against more complex models, Logistic regression works under the assumption that there is no multicollinearity. Multicollinearity occurs when there are two or more independent variables in a multiple regression model, which have a high correlation among themselves. One way to nd multicollinearity in a dataset is to check the Variance In ation Factor (VIF) of the features. The VIF values during the algorithm performance is given in Fig. 11.

We can see that VIF values of ‘Is declined’, ‘isHighRiskCountry’, ‘Daily\_chargeback\_avg\_amt’ and ‘6\_month\_avg\_chbk\_amt’ are high, which indicates that there is a multicollinearity in the dataset 2. This issue can be resolved by dropping the columns. But because of the lack of availability of data we cannot rely on the performance of logistic regression.

Table 2

Model Performance for large Data set

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Models** |  |  |  |  | **Acuracy** |  |  |  |
|  | 4657 | 136 | 143 | 641 | 0.859 | 0.175 | 0.487 | 0.257 |
|  | 4787 | 736 | 13 | 41 | 0.990 | 0.947 | 0.982 | 0.964 |
|  | 4769 | 752 | 25 | 31 | 0.989 | 0.967 | 0.960 | 0.964 |
|  | 4799 | 712 | 1 | 65 | 0.988 | 0.916 | 0.998 | 0.955 |
|  | 4762 | 743 | 8 | 64 | 0.987 | 0.920 | 0.989 | 0.953 |
|  | 4791 | 753 | 9 | 24 | 0.994 | 0.969 | 0.988 | 0.978 |
|  | 4798 | 713 | 2 | 64 | 0.988 | 0.917 | 0.997 | 0.955 |

Finding fraudulent online transactions is one of the major problems faced in banking and online societies today. In this paper, we have built various machine learning models and analyzed their performances against different datasets based on the capacity and the volume of the data. Testing the models against a different dataset with considerable volume helps us to understand the model performance in real-time. By applying the SMOTE, we handled the imbalanced nature of the dataset. The other way of handling an imbalanced dataset is to use one-class classi ers like one-class SVM, random under sampling and random oversampling. After analyzing the models with large data sets, Random Forest gives the maximum F1 score of 97% and it is used as a primary model to build the web application. We nally observed that random forest and XGBoost are the algorithms that gave better results. The work can be extended to develop adaptive machine learning fraudulent detection methods for complex real-world data which is characterized by imbalance data set.

The manuscript is original and has not been published, accepted for publication or under editorial review for publication elsewhere.

Not applicable

Not applicable

No funding was received to assist with the preparation of this manuscript.

Free available data used

To editor of this journal and all reviewer, Thanks so much.

## **Online Payments Fraud Detection using Python**

I will start this task by importing the necessary Python libraries and the **data set** we need for this task:

import pandas as pd

import numpy as np

data = pd.read\_csv("credit card.csv")

print(data.head())

**step type amount nameOrig oldbalanceOrg newbalanceOrig \**

**0 1 PAYMENT 9839.64 C1231006815 170136.0 160296.36**

**1 1 PAYMENT 1864.28 C1666544295 21249.0 19384.72**

**2 1 TRANSFER 181.00 C1305486145 181.0 0.00**

**3 1 CASH\_OUT 181.00 C840083671 181.0 0.00**

**4 1 PAYMENT 11668.14 C2048537720 41554.0 29885.86**

**nameDest oldbalanceDest newbalanceDest isFraud isFlaggedFraud**

**0 M1979787155 0.0 0.0 0 0**

**1 M2044282225 0.0 0.0 0 0**

**2 C553264065 0.0 0.0 1 0**

**3 C38997010 21182.0 0.0 1 0**

**4 M1230701703 0.0 0.0 0 0**

Now, let’s have a look at whether this dataset has any null values or not:

1

**print(data.isnull().sum())**

**step 0**

**type 0**

**amount 0**

**nameOrig 0**

**oldbalanceOrg 0**

**newbalanceOrig 0**

**nameDest 0**

**oldbalanceDest 0**

**newbalanceDest 0**

**isFraud 0**

**isFlaggedFraud 0**

**dtype: int64**

So this dataset does not have any null values. Before moving forward, now, let’s have a look at the type of transaction mentioned in the dataset:

1

# Exploring transaction type

print(data.type.value\_counts())

**CASH\_OUT 2237500**

**PAYMENT 2151495**

**CASH\_IN 1399284**

**TRANSFER 532909**

**DEBIT 41432**

**Name: type, dtype: int64**

type = data["type"].value\_counts()

transactions = type.index

quantity = type.values

​

import plotly.express as px

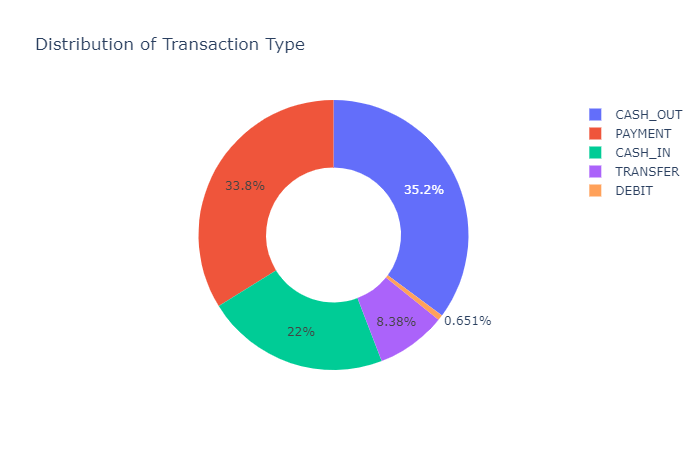
figure = px.pie(data,

values=quantity,

names=transactions,hole = 0.5,

title="Distribution of Transaction Type")

figure.show()



Now let’s have a look at the correlation between the features of the data with the **isFraud** column:

# Checking correlation

correlation = data.corr()

print(correlation["isFraud"].sort\_values(ascending=False))

**isFraud 1.000000**

**amount 0.076688**

**isFlaggedFraud 0.044109**

**step 0.031578**

**oldbalanceOrg 0.010154**

**newbalanceDest 0.000535**

**oldbalanceDest -0.005885**

**newbalanceOrig -0.008148**

**Name: isFraud, dtype: float64**

Now let’s transform the categorical features into numerical. Here I will also transform the values of the **isFraud** column into No Fraud and Fraud labels to have a better understanding of the output:

1

data["type"] = data["type"].map({"CASH\_OUT": 1, "PAYMENT": 2,

2

"CASH\_IN": 3, "TRANSFER": 4,

3

"DEBIT": 5})

4

data["isFraud"] = data["isFraud"].map({0: "No Fraud", 1: "Fraud"})

5

print(data.head())

1. Donald Di Nardo, Identity theft , Handbook of Loss Prevention and Crime Prevention (Fifth Edition), Chapter 34, pp. 420-422, 2012. https://doi.org/10.1016/B978-0-12-385246-5.00034-1.
2. Kamta Nath Mishra, Subash Chandra Pandey, “Fraud Prediction in Smart Societies using Logistic Regression and K- fold Machine Learning techniques”, Wireless personal Communications, pp. 1-22,

2021. https://link.springer.com/article/10.1007/s11277-021-08283-9

1. K. Modi and R. Dayma, "Review on fraud detection methods in credit card transactions," 2017 International Conference on Intelligent Computing and Control (I2C2), Coimbatore, 2017, pp. 1-5, doi:

10.1109/I2C2.2017.8321781.

1. M. S. Kumar, V. Soundarya, S. Kavitha, E. S. Keerthika and E. Aswini, "Credit Card Fraud Detection Using Random Forest Algorithm," 2019 3rd International Conference on Computing and Communications Technologies (ICCCT), Chennai, India, 2019, pp. 149-153, doi:

10.1109/ICCCT2.2019.8824930

1. S. Dhankhad, E. Mohammed and B. Far, "Supervised Machine Learning Algorithms for Credit Card

Fraudulent Transaction Detection: A Comparative Study," 2018 IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, 2018, pp. 122-125, doi:

10.1109/IRI.2018.00025.

. S. Mittal and S. Tyagi, "Performance Evaluation of Machine Learning Algorithms for Credit Card

Fraud Detection," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Con uence), Noida, India, 2019, pp. 320-324. doi: 10.1109/CONFLUENCE.2019.8776925.

7. J. O. Awoyemi, A. O. Adetunmbi and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," 2017 International Conference on Computing Networking and Informatics (ICCNI), Lagos, 2017, pp. 1-9, doi: 10.1109/ICCNI.2017.8123782.

. R. Popat and J. Chaudhary, "A Survey on Credit Card Fraud Detection Using Machine Learning," 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, 2018, pp.

1120-1125, doi: 10.1109/ICOEI.2018.8553963.

1. D. Dighe, S. Patil and S. Kokate, "Detection of Credit Card Fraud Transactions Using Machine

Learning Algorithms and Neural Networks: A Comparative Study," 2018 Fourth International

Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp.

1-6, doi: 10.1109/ICCUBEA.2018.8697799.

1. Altyeb Altaher Taha and Sharaf Jameel Maleberry, An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine, IEEE Access, vol. 8, pp. 25579 – 25587, 2020.
2. P. Kumar and F. Iqbal, "Credit Card Fraud Identi cation Using Machine Learning Approaches," 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), CHENNAI, India, 2019, pp. 1-4, doi: 10.1109/ICIICT1.2019.8741490.
3. A.Mishra and C. Ghorpade, "Credit Card Fraud Detection on the Skewed Data Using Various

Classi cation and Ensemble Techniques," 2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, 2018, pp. 1-5,doi:

10.1109/SCEECS.2018.8546939.

1. S Makki, Z Assaghir, Y Taher, R Haque, MS Hacid, H Zeineddine,An experimental study with imbalanced classi cation approaches for credit card fraud detection, IEEE Access 7, 93010-93022.
2. Carcillo Fabrizio, Le Borgne Yann-Aël, Caelen Olivier and Bontempi Gianluca“Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization”, International Journal of Data Science and Analytics, pp. 1-18, Springer 2018.
3. A.Thennakoon, C. Bhagyani, S. Premadasa, S. Mihiranga and N. Kuruwitaarachchi, "Real-time Credit Card Fraud Detection Using Machine Learning," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Con uence), Noida, India, 2019, pp. 488-493, doi:

10.1109/CONFLUENCE.2019.8776942.

1 . D. Prusti and S. K. Rath, "Web service based credit card fraud detection by applying machine learning techniques," TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), Kochi, India, 2019, pp.492497.

17. Y. Xie, G. Liu, R. Cao, Z. Li, C. Yan and C. Jiang, "A Feature Extraction Method for Credit Card Fraud Detection," 2019 2nd International Conference on Intelligent Autonomous Systems (ICoIAS), Singapore, Singapore, 2019.

1 . C. Jiang, J. Song, G. Liu, L. Zheng and W. Luan, "Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism," in IEEE Internet of Things Journal, vol. 5, no.

5, pp. 3637-3647, Oct. 2018, doi: 10.1109/JIOT.2018.2816007.

1. Andrea Dal Pozzolo, Giacomo Boracchi, Olivier Caelen, Cesare Alippi, and Gianluca Bontempi, Credit Card Fraud Detection: a Realistic Modeling and a Novel Learning Strategy, IEEE Transactions on Neural Networks and Learning Systems, pp 1-14, September 2017, DOI:

10.1109/TNNLS.2017.2736643

1. Z. Li, G. Liu and C. Jiang, "Deep Representation Learning With Full Center Loss for Credit Card Fraud Detection," in IEEE Transactions on Computational Social Systems, vol. 7, no. 2, pp. 569-579, April 2020, doi: 10.1109/TCSS.2020.2970805.
2. Online Payment Fraud White Paper, 2016-2020, pp. 1 – 29, Juniper Research.
3. C. Phua, K. Smith-Miles, V. Lee and R. Gayler, "Resilient Identity Crime Detection," in

IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 3, pp. 533-546, March 2012, doi:

10.1109/TKDE.2010.262.

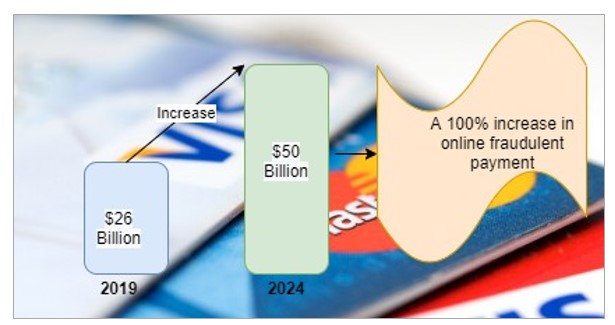
1. S. Yu, X. Li, X. Zhang and H. Wang, "The OCS-SVM: An Objective-Cost-Sensitive SVM With SampleBased Misclassi cation Cost Invariance," in IEEE Access, vol. 7, pp. 118931-118942, 2019, doi:

10.1109/ACCESS.2019.2933437.

1. L. Zheng, G. Liu, C. Yan and C. Jiang, "Transaction Fraud Detection Based on Total Order Relation and Behavior Diversity," in IEEE Transactions on Computational Social Systems, vol. 5, no. 3, pp. 796806, Sept. 2018, doi: 10.1109/TCSS.2018.2856910.
2. Z. Zhang, L. Chen, Q. Liu and P. Wang, "A Fraud Detection Method for Low-Frequency Transaction," in IEEE Access, vol. 8, pp. 25210-25220, 2020. doi: 10.1109/ACCESS.2020.2970614.

2 . https://www.ftc.gov/news-events/press-releases/2019/02/imposter-scams-top-complaints-made-ftc2018

27. https://www.simpletaxindia.net/2016/12/card-not-present-transactions.html

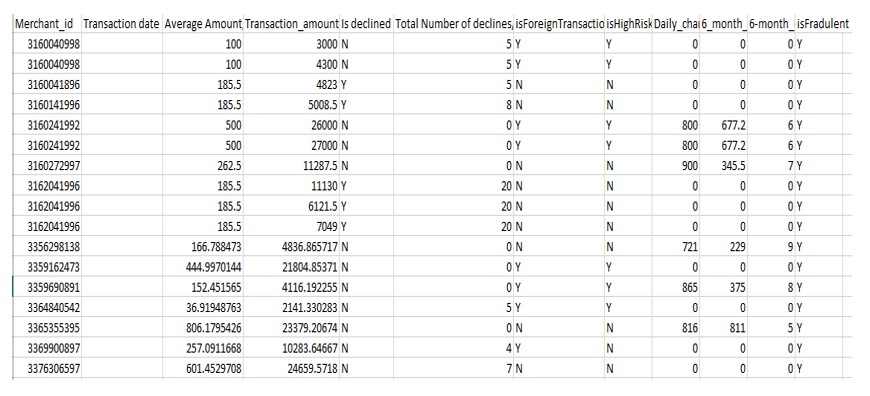


1

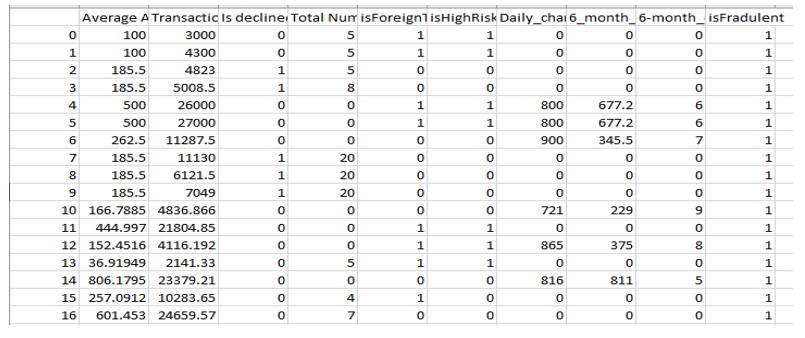
Increase in Online fraudulent payment.



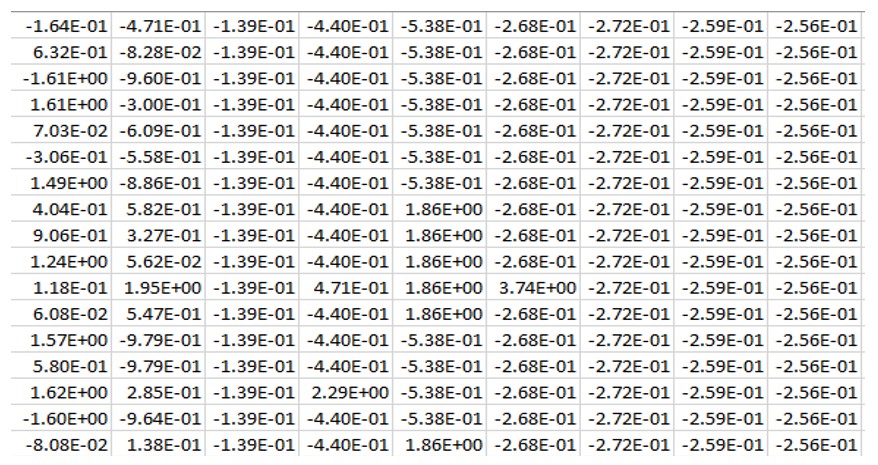
Types of Online Frauds



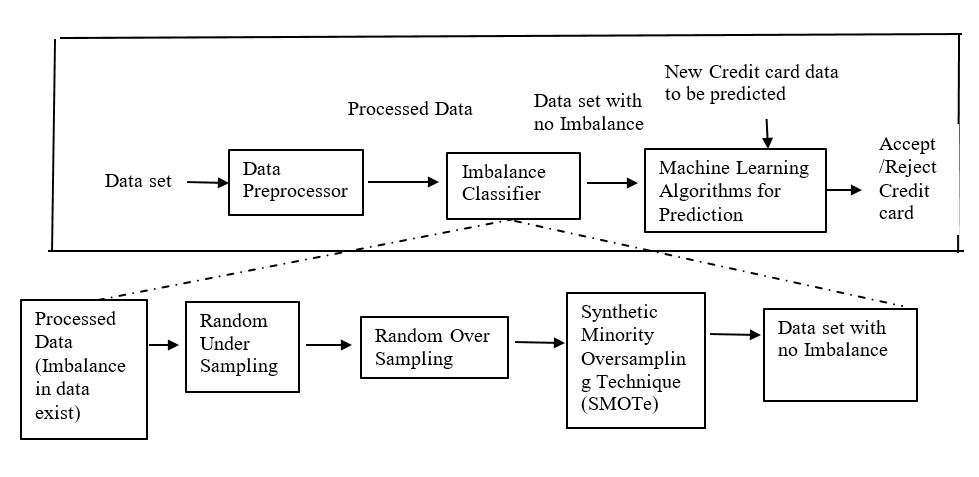
Dataset before preprocessing



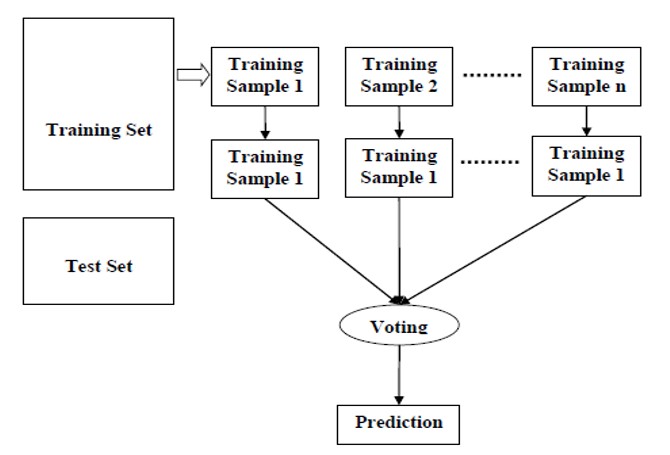
Dataset after preprocessing and formatting



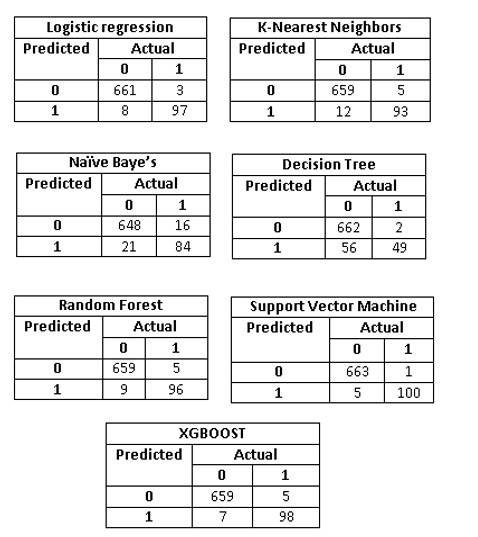
Normalized dataset



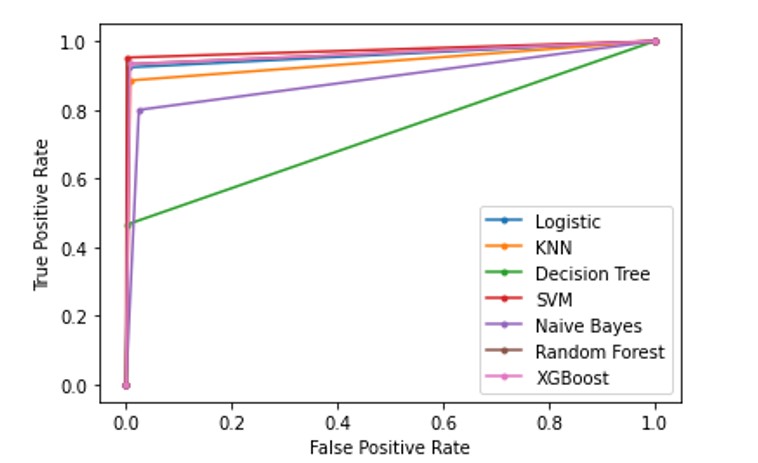
Steps in Credit Card Processing



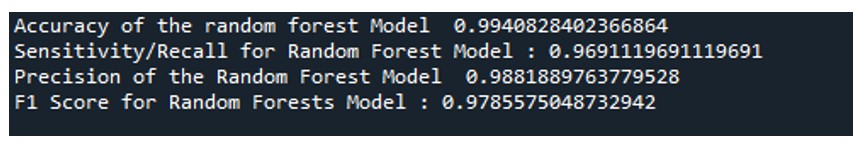
Working of Random Forest Model



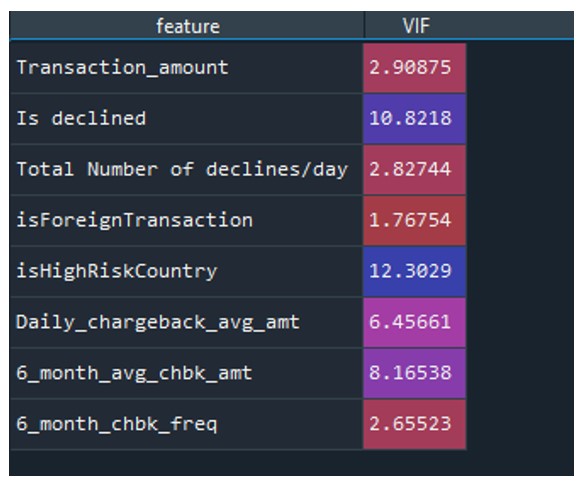
Confusion matrix for the algorithms



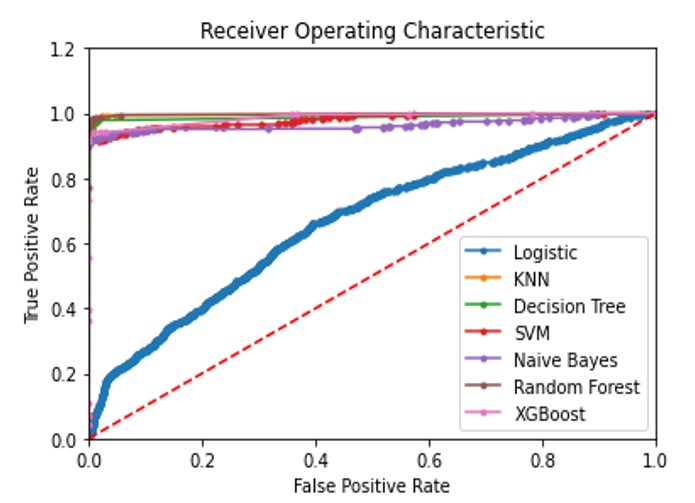
ROC curve 1



Output for Random Forest Model



VIF of Logistic Regression



ROC Curve 2