Detecting Anomalies For Credit Card Transactions

Sree Gowri Addepalli
Computer Science
New York University, Courant
Instiitute of Mathematical Sciences
New York, USA
sga297@nyu.edu

Sree Lakshmi Addepalli
Computer Science
New York University, Courant
Institute of Mathematical Sciences
New York, USA
sla410@nyu.edu

Anasse Bari
Computer Science
New York University, Courant
Institute of Mathematical Sciences
New York, USA
abari@nyu.edu

***Abstract*—The rapid growth of technology has significantly both led to an increase in revenue for multinational companies and yet at the same time there are billion-dollar business of financial fraud and losses around the world every year. The financial institutions and companies lose lots of money due to various frauds and most of the fraudsters continuously try to find new ways to commit such fraudulent actions for monetary benefits. Thus, having a fraud detection system is a requirement for every bank issuing a credit card to lessen such losses and build a brand value among its customers and a trust factor. The growing research and community members for predictive analytics directs us to build an intelligent system that help us find these fraudulent transactions. The paper presents a Cross Industry Process for Data Mining (CRISP-DM) that helps us build predictive models for evaluating financial transactions. With the help from areas like Data Science, Statistics, Mathematics and Computer Science, we used various classification machine learning algorithms such as Logistic Regression, Support Vector Machines (SVM), T- SNE, Stochastic Gradient Descent, and K-Nearest Neighbor for fraud detection. We used data balancing techniques like Random Under sampling and Random Oversampling for improving model performance. Aggregating various machine learning models, we successfully try to create an ensemble model to see whether the accuracy improves or not. This paper presents the CRISP-DM process of the project as well as the results obtained by using the above techniques independently as well as in collaboration of various techniques and evaluates each model based on accuracy and speed.**

Keywords — Credit Card, Fraud Detection Techniques, CRISP-DM, Machine Learning Algorithms, Logistic Regression, Support Vector Machines, Stochastic Gradient Descent, K-Nearest Neighbors, T-SNE, Data imbalance, Random Oversampling, Random Undersampling, Ensemble Model, Bagging, Boosting, Random Forest, XGBoost, AdaBoost, Deep Learning.

I. INTRODUCTION

The credit card is a card which is issued to the user as a system of payment by the bank. Card owner uses his/her card to make payments to a merchant. Credit card fraud is a serious issue in banking services that makes use of the card or card's information without the knowledge of the owner of the card. Fraud is a criminal activity and violates the public law in which the fraudster causes financial loses. There are many ways to carry out fraudulent transactions. One way is to steal the credit card. Other ways are to take the important information about the card such as its card number, secure code, expiration date and others fields required to make the payment. In most of the cases, the card owner doesn't know that someone else has stolen or seen his card information. Most of such fraudulent transactions are masked as genuine transactions which make it difficult to identify such anomalies. These anomalies can be detected with data mining techniques which internally uses mathematical computations to distinguish between such transactions based on similarity score. The data points that deviate from the regular dataset and are not consistent with them are called Outliers or anomalies and should be under scrutiny. Detecting fraud is a complex task and still, they are no systems that predicts any fraudulent transactions accurately but gives measures in probabilities.

II. LITERATURE SURVEY

A. Application of credit card fraud detection: based on bagging ensemble classifier, International conference on Intelligent Computing, Communication and Convergence, Elsevier.

This paper explains the benefits of using a bagging ensemble classifier over the traditional machine learning algorithms like Naïve Bayes, Support Vector Machine, k Nearest Neighbor due to its exceptional performance on practical problems. The paper also discusses the challenges in credit card fraud detection as:

- Non Availability of real dataset as banks are not ready to reveal sensitive data.
- Highly imbalanced data in credit card fraud detection due to less amount of fraudulent transaction in huge amounts of transactions.
- Size of dataset being large to be processed as the number of transactions that occur on a daily basis is very huge.
- Determining the accurate evaluation metric as false positive rate and false negative rate are inversely proportional and there is high cost in classifying false negative rate rather than false positive rate. Accuracy is never a good measure for measuring performance in imbalanced dataset.

True Positive (TP) = Number of fraud transactions predicted as fraud.
True Negative (TN) = Number of legal transactions predicted as legal.
False Positive (FP) = Number of legal transactions predicted as fraud.
False Negative (FN) = Number of fraud transactions predicted as legal.

Dynamic behaviour of fraudster keeps this topic much in research as there is no definitive structure to capture such patterns.

B. Analysis of Credit Card Fraud Detection Techniques: Based on Certain Design Criteria, International Journal of Computer Applications (0975 – 8887)

According, to this paper, properties of good fraud detection system are:

1) The frauds should be identified accurately.

2) The frauds should be detected quickly.

3) A genuine transaction should not be classified as fraud.

This paper does a comparative study of nine fraud detection methods based on credit card transactions research from various references. The comparative study analyses Decision Tree, Neural Network, Bayesian Network, Genetic algorithm, Support Vector Machine, K Nearest Neighbor and Artificial Immune System, Hidden Markov Model, Fuzzy Neural Network and Fuzzy Darwinian System. Thus, this survey enables us to build a hybrid approach for developing some effective algorithms which can perform well for the classification problem with variable misclassification costs and with higher accuracy.

Methods	Speed of detection	accuracy	cost
HMM	Fast	Low	High expensive
FDS	Very low	Very high	High expensive
AIS	Very fast	Good	Inexpensive
FNN	Very fast	Good	Expensive
NN	Fast	Medium	Expensive
DT	Fast	Medium	Expensive
BN	Very Fast	High	Expensive
KNN	Good	Medium	Expensive
SVM	Low	Medium	Expensive
SOM	Fast	Medium	Expensive
BP	low	Low	Expensive
GA	Good	Medium	Inexpensive

Table 1: comparison of different methods

Figure 1: Comparison of Different Methods

III. DATA SOURCES

The data source has been taken from Kaggle website, creditcard.csv that contains Time, Amount, V1, V2, V3…V28, Class (31 columns) and 284, 407 rows. The datasets has transactions made by credit cards in September 2013 by European cardholders. This dataset is highly imbalanced as it contains 492 fraudulent transactions out of 284,407 transactions which account for 0.172% of all transactions. Columns V1…V28 are a result of PCA transformation due to confidentiality issues. The class feature has a value of 1 if it is a fraudulent transaction and 0 if it is a non- fraudulent one.

IV. PROJECT WORKFLOW

The project workflow is designed on the CRISP- DM Model. The first phase is the discovery of the problem statement and developing hypotheses which requires business understanding. The next phase is designing the analytics framework where the data procurement and storage is done. The next stage is the most crucial stage where the data is cleaned, processed and prepared as per the requirements. Extract, Transform and Load (ETL) and Exploratory Data Analysis (EDA) are two important process of this stage. The next stage requires feature selection and iterating over models to select machine learning algorithms that are a good fit to give a predictive power. Model evaluation helps to select which model suits the business requirement the best using various mathematical measures like Accuracy, AUC etc. Model deployment is where the data driven intelligent model must be deployed on the server for the client/customer.

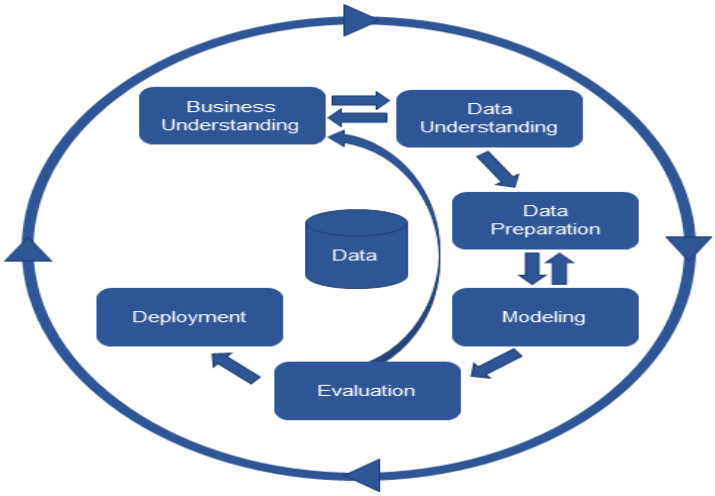

Figure 2: CRISP-DM MODEL

A. BUSINESS UNDERSTANDING

Credit cards are used by many customers of various financial institutions to perform various online, ATM transactions. These transactions can sometimes be fraudulent done by people who aim to gain monetary benefit without authorization. This leads to financial losses for the banks and creates a sense of mistrust between the bank and customer and could be a major source for banks losing their customers and trust. Hence, it becomes necessary for financial institutions to identify and hold such transactions accountable for security purposes. With advance in technology and availability of massive data the business can attempt to be build predictive models to aid them in identifying such fraudulent transactions.

B. DATA UNDERSTANDING

a) DATA IMBALANCE:

We can see that the dataset contains 31 columns and 284, 407 rows out of which 492 transactions are only fraudulent which accounts for 0.172% and hence data imbalance.

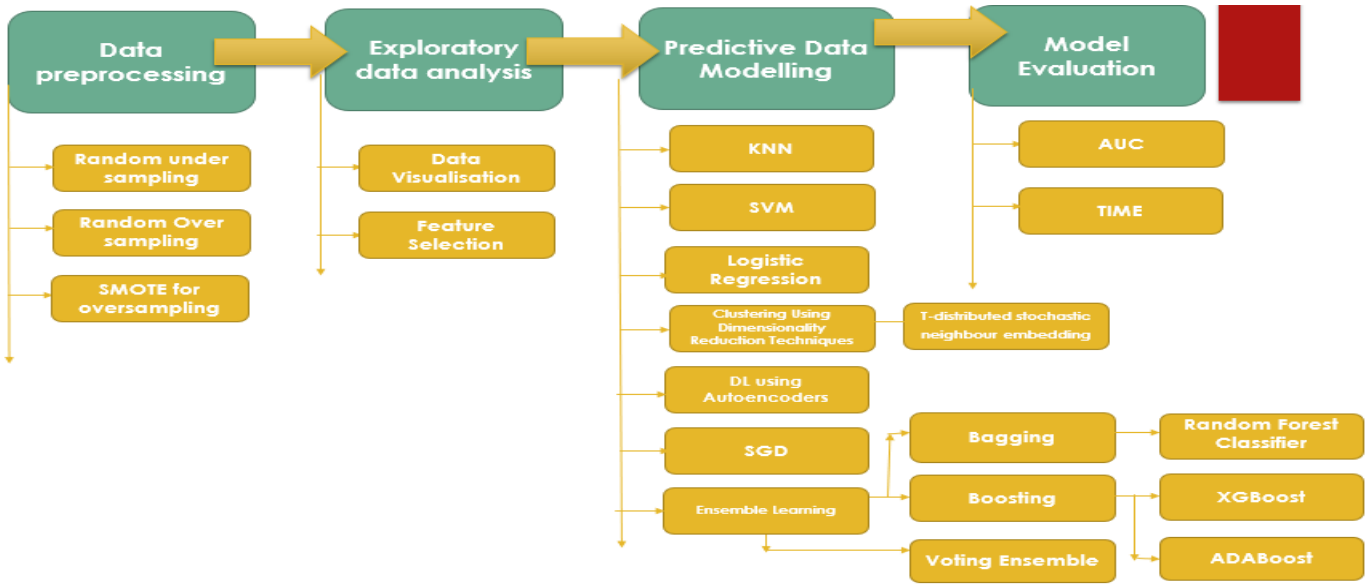

Figure 3: CRISP-DM MODEL

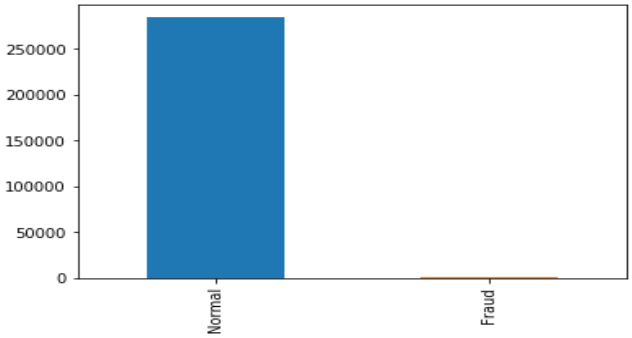

Figure 4: Normal vs Fraud transactions count

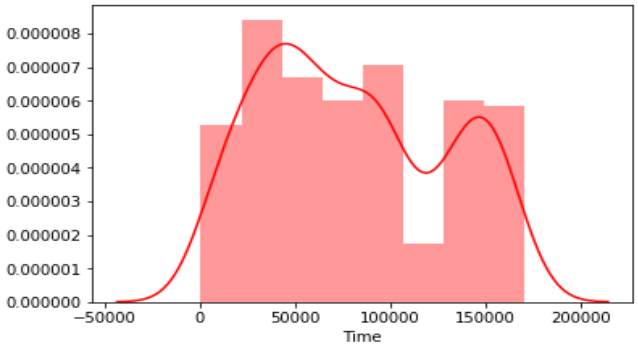

Figure 5: Fraudulent Transactions Vs Time

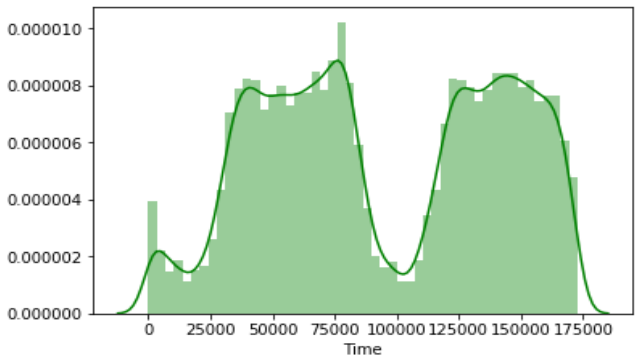

Figure 6: Non-Fraudulent Transactions Vs Time

B) FEATURE SCALING

As the other columns are transformed into a standard normal form using PCA we need to transform the columns Time and Amount too into standard normal form. Given the distribution of the data, each value in the dataset will have the sample mean value subtracted from it, and then divide it by the standard deviation of the whole dataset so that all features are on the same scale.

c) EXPLORATORY DATA ANALYSIS

We can explore the data to get a more holistic view of the data and get the following observations:

- There are no null values in the dataset. So, there is no need to handle such missing values.
- There are 284315 normal transactions and 492 fraudulent transactions.
- The visual representations show the number of transactions of both fraudulent (Red) and non-fraudulent (Green) transactions with time and hence, show that time is not a significant factor in distinguishing one from other.
- The visual representations show the amount of transactions of both fraudulent (Blue) and non-fraudulent (Pink) transactions with time and hence, show that amount is not a significant factor in distinguishing one from other.

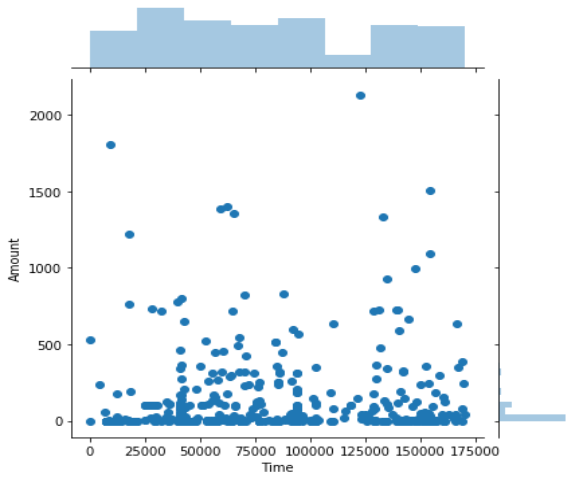

Figure 7: Fraudulent Transactions Amount Vs Time

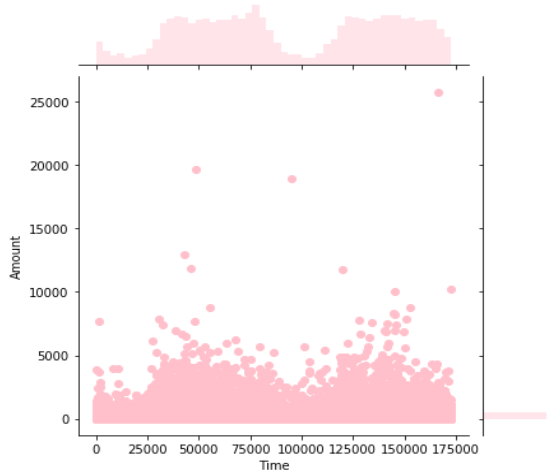

Figure 8: Nonfraudulent Transactions Amount Vs Time

- Below is a heatmap that shows the correlation between the features in the dataset. The features that are less correlated are more towards purple range as shown on a gradient scale below and highly correlated are on the light pink color.

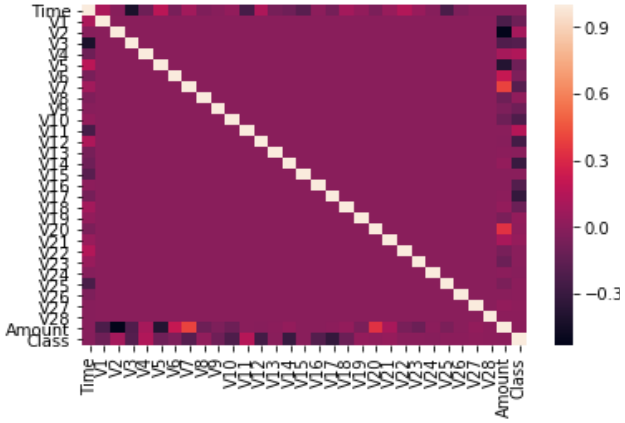

Figure 9: Correlation between features in the dataset

- From the below feature wise correlation, we can see that features 'V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V20', 'V15', 'V13', 'V8' are redundant and do not vary with class 'Fradulent' and 'Normal'.

- So, the major contributing features that distinguish between the fraudulent transactions and non fraudulent ones are V1-V7, V9, V10, V11, V12, V14, V16- V19, V21.

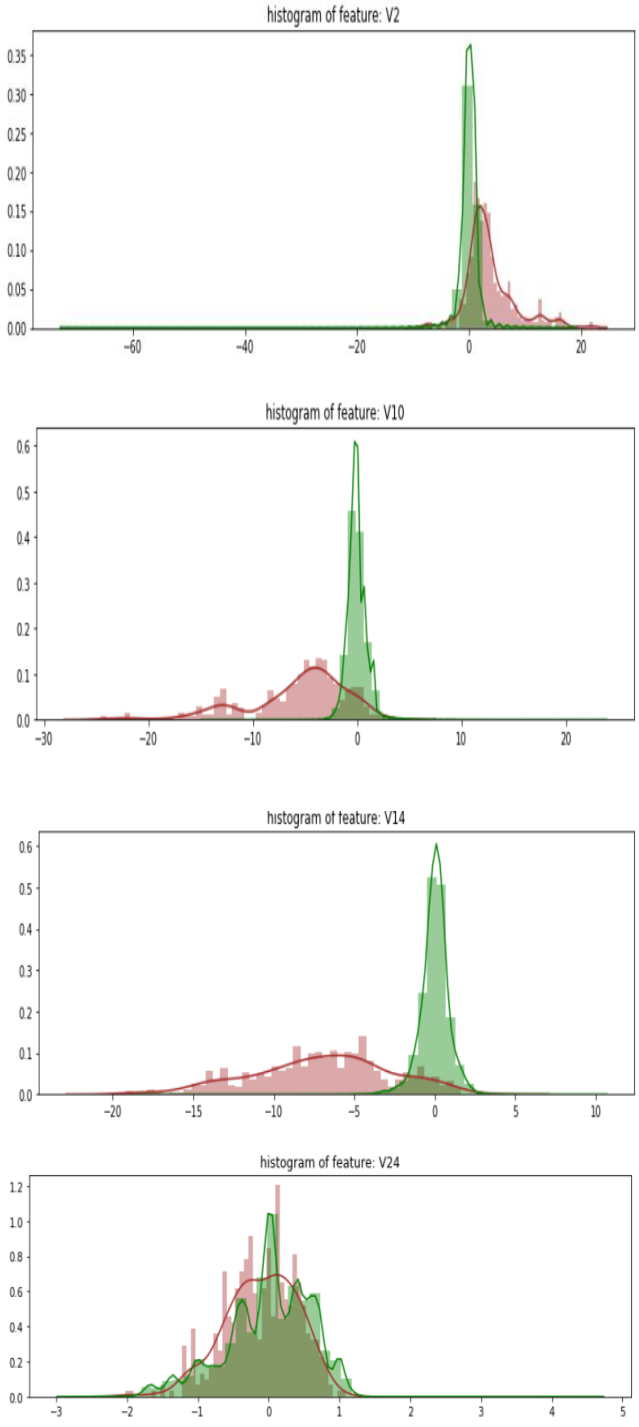

Figure 10: Feature wise correlation between fraudulent and non-fraudulent classes

C. DATA PREPARATION

a.) Data balancing strategies:

As the imbalance in the data can be seen below, we need to balance data in the dataset:

1. It can cause overfitting of the data and assume the major class as the output for the testing set.

2. We can fail to understand the correlations between the features due to these anomalies as they are in insignificant amount compared to the major class.

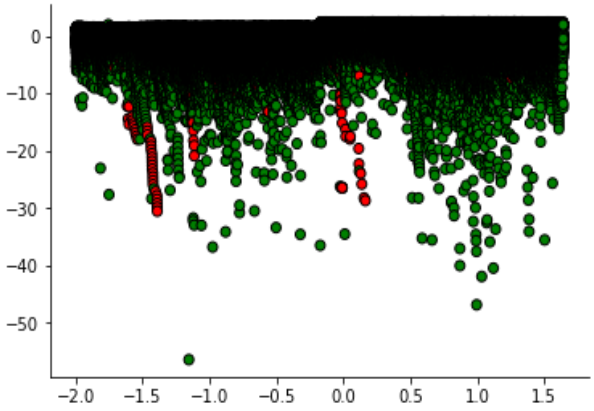

Figure 11: Scatterplot of Imbalanced Dataset

Sampling doesn't introduce new information in the dataset, it merely shifts it around to increase the "numerical stability" of the resulting models. The following are the techniques used to balance the dataset:

1. Random Undersampling: It is a technique to remove samples from the majority class to equal the minority class.

Advantages:
It reduces the storage space and reduces runtime and works well in case of large training set.

Disadvantages:

- There is loss of data which could be important in building rules important for classifiers.
- Randomly removing the data samples may lead to biased sample which may not serve the purpose and would not give a good model.
- The below picture shows how Random Undersampling works on our dataset.

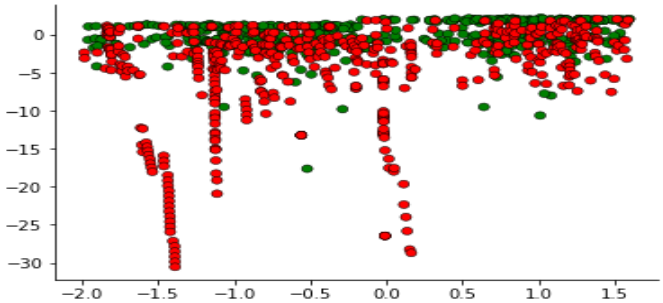

Figure 12: Scatterplot of Random Undersampling on the dataset

- **Random Oversampling:** It is a technique to add samples to the minority class to equal the majority class.

Advantages:

- It doesn't lead to information loss and hence better than random Under sampling.
- It performs better than random Under sampling.

Disadvantages:

- Random Oversample can lead to overfitting as only minor class samples are replicated.
- The below picture shows how Random Oversampling works on our dataset.

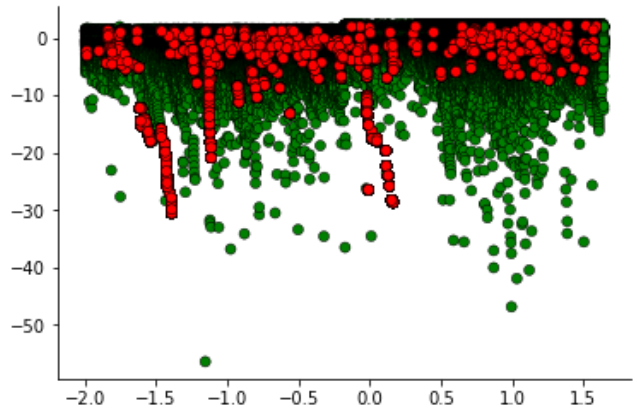

Figure 13: Scatterplot of Random Oversampling on the dataset

- **Random Oversampling using SMOTE (Synthetic Minority Over-sampling Technique):**

It is an oversampling technique that reduces overfitting. It takes a subset of minority class and generates synthetic samples like them and added to the original dataset.

Advantages:

- Removes the issue of overfitting.
- Loss of information is not present.

Disadvantages:

- SMOTE doesn't generate samples of other class and hence could be adding noise to the dataset.
- It is not effective for high dimension data.

The below picture shows how Random Oversampling using SMOTE works on our dataset.

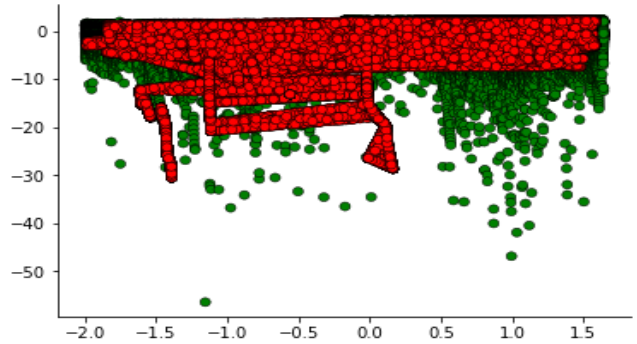

Figure 14: Scatterplot of SMOTE on the dataset

b) Feature Selection:

Using Random Forest Classifier, we can score the features according to certain weight. Below is the bar chart according to the weight. We can see that V10, V14, V4 have highest weightage.

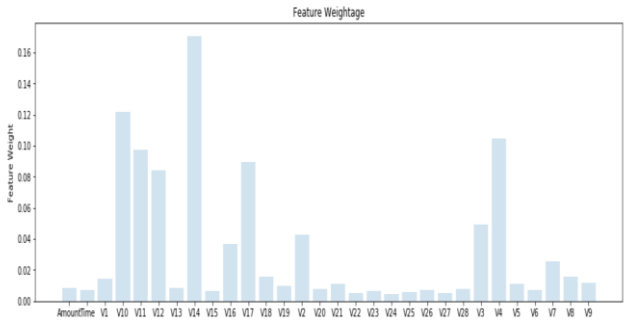

Figure 15: Bar Graph distribution of features according to predictive power.

D. PREDICTIVE MODELLING FOR CLASSIFICATION

a. Logistic regression

Logistic function is a linear algorithm and used in regression which maps any real number between 0 and 1 to make it a classification model. Logistic regression models the probability of a class where these values must be transformed to 0 or 1 to make it a binary classifier. It can overfit the data if it contains multiple correlated values.

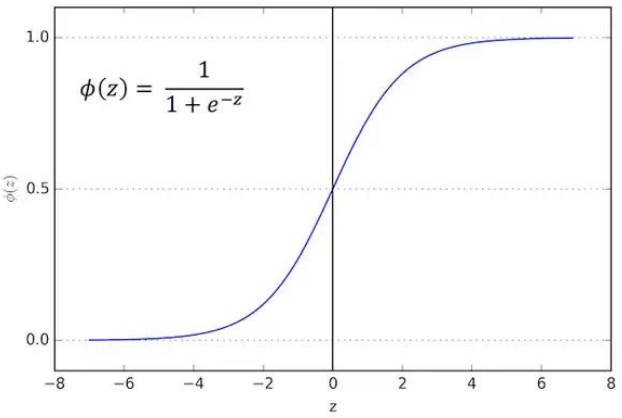

Figure 16: Logistic Regression Curve

b. K Nearest Neighbour

K Nearest Neighbor is a non-parametric classification algorithm that uses the majority of the classes of K nearest neighbours as the output of the variable whose class must be predicted.

c. Support Vector Machine

Support vector machine is a non-probabilistic supervised machine learning model used for classification. It ensures that examples of different classes can be separated by a hyperplane and are at the maximum distance and hence the predicted variables must be mapped accordingly.

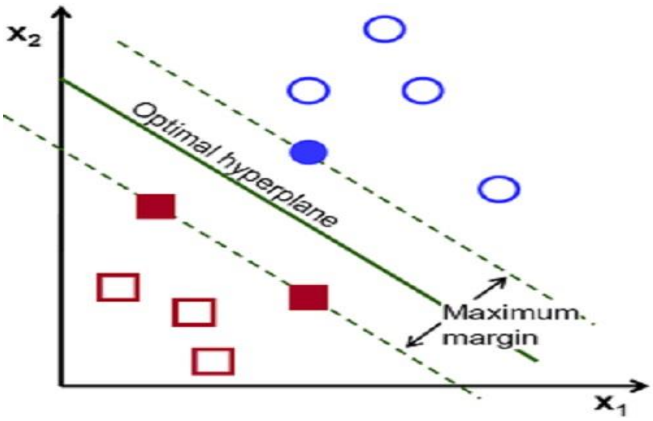

Figure 17: Support Vector Machine Model

d. Stochastic Gradient Descent

It is an iterative optimizing technique used to minimize the objective function. So, it is called stochastic as the samples are randomly selected and the gradient of the loss is estimated each time with a decreasing learning rate. So, when a classifier is implemented with Stochastic gradient descent training it is called as SGD classifier.

e. Clustering Using Dimensionality Reduction Technique: t-distributed Stochastic Neighbour Embedding

t-SNE is a dimensionality reduction algorithm that can cluster or differentiate between fraud and non-fraud samples very clearly after performing dimensionality reduction. It gives different output every run due to shuffle it performs. It is a good indicator in the start to whether other predictive models will be able to perform classification better or not as it clearly gives clear distinction between the clusters.

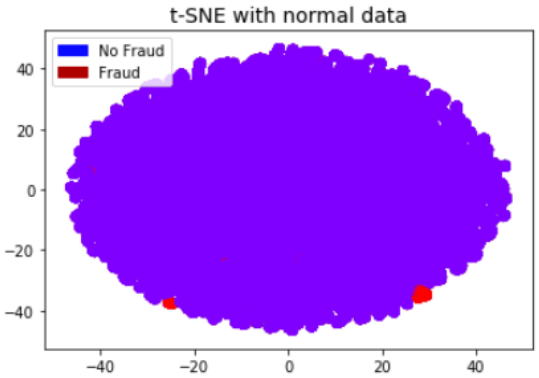

Figure 18: t-SNE on normal data

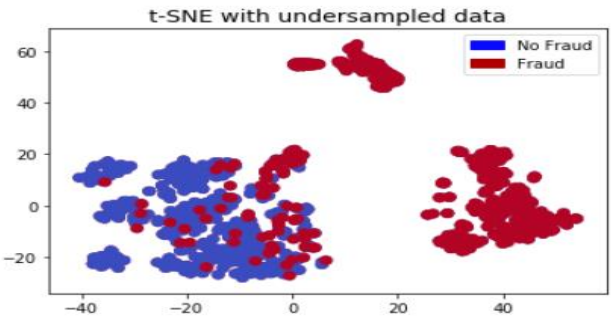

Figure 19: t-SNE on undersampled data

f. Ensemble Modelling

Ensemble Modelling is a technique where many predictive models are used together to make a better decision rather than a single model. The project uses the following ensembling strategies out of the many:

1. Bagging using Random Forest Classifier.

Bagging is known as Bootstrap Aggregating. It means that instead of applying different models to the same set of data which has a high probability to give the same output, it vies for creating multiple subsets from the original data and applying the predictive model on all of them in parallel to combine them in the end to give a strong combined prediction. In our project we used Random Forest for Bagging which internally uses decision trees.

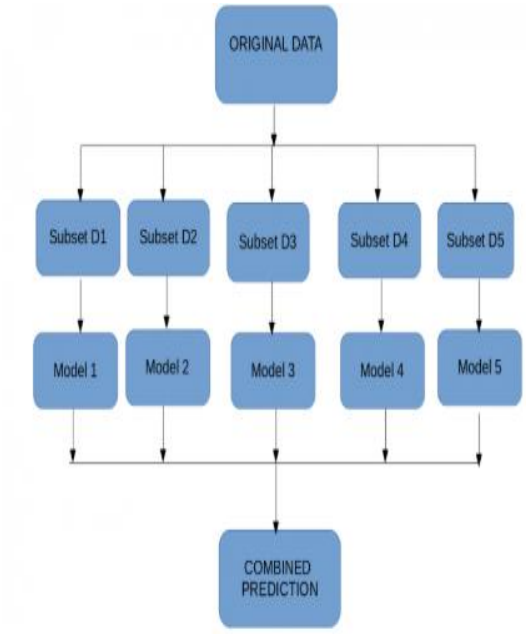

Figure 20: Concept of Bagging

Random Forest:

Random Forest internally uses decision trees and with the help of few features it decides the best split at every node. Random forest selects random data points and features to build multiple trees (Forest).

2. Boosting

Boosting is a better technique to reduce errors and provide better predictions. It is a sequential process where the model is applied to subset and the predictions are made on the entire dataset. The error value is calculated on the predictions and the wrong values are given more weightage. Then the process of creating subsets and predicting on the whole dataset is repeated and multiple models are created that try to minimize the error of the previous model and combines them all in a weighted mean manner to give the final output.

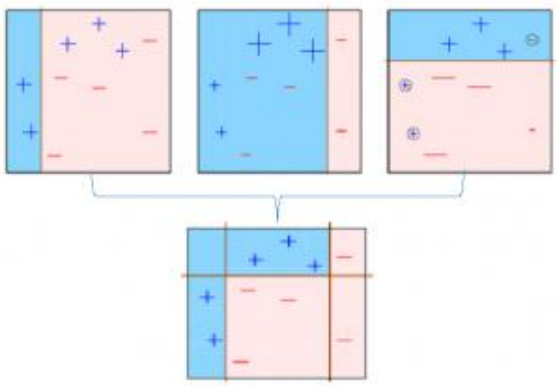

Figure 21: Concept of Boosting

Our project uses the following boosting algorithms:

XGBoost (Xtreme Gradient Boosting)

It is one of the most powerful machine learning algorithms that runs 10 times faster and has high predictive power. It uses a variety of regularizations which reduces overfitting and hence is known as regularized boosting. It implements parallel processing and allows users to define custom optimization and evaluation criteria.

AdaBoost

Adaptive boosting also known as AdaBoost is the simplest boosting algorithm which assigns more weight to the incorrect predicted values to reduce the error.

3. Voting Ensemble

In this multiple independent machine learning models are run on the training data and as per the weighted (optional) output is taken into consideration as the number of votes or average of all the classifiers. In our project we tried to create a voting ensemble of Logistic regression, K Nearest Neighbour, Stochastic Gradient Descent classifier and SVM classifier.

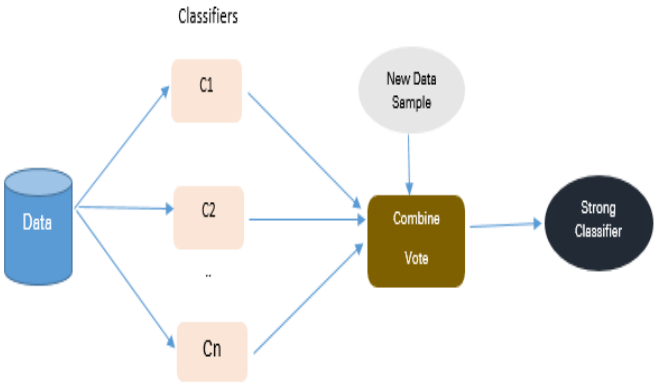

Figure 22: Concept of Voting Ensemble

g. AutoEncoders for Deep Learning

With the improvement in the computation speed and the price of hardware devices drastically dropping Deep Learning is now being incorporated building powerful intelligent products due to its inherent capacity to self-identify important or predictive features through the data fed into the model. Autoencoders are one of the deep learning models that are a type of neural network that take an input, deconstructs it into its lower layer and to reconstructs these layers to form the input again. If the output and input match, then the data is said to be normal else it has been tampered. This property of autoencoder is used detecting anomalies in the transactions as the fraudulent transactions will contain high reconstruction error.

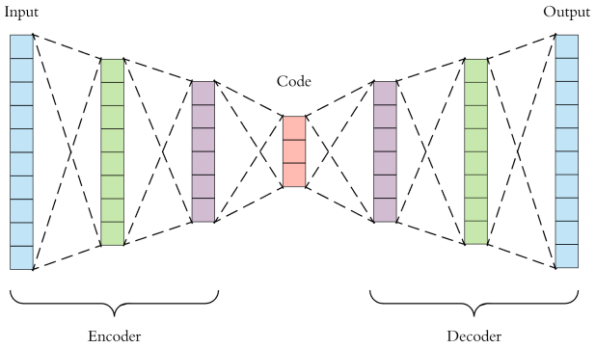

Figure 23: Concept of Autoencoders

E. MODEL EVALUATION

Model evaluation done with oversampled data with SMOTE. Choosing performance metric as accuracy isn't the correct measure and gives the great feeling of your model being good as it majorly predicts the major class as the predicted value. The following could be considered as metrics for imbalanced data:

- Kappa (or Cohen's kappa): Classification accuracy normalized by the imbalance of the classes in the data.
- ROC (Receiver operating characteristic curves) Curves: Like precision and recall, accuracy is divided into specificity and sensitivity as models can be chosed based on balance thresholds of these corresponding values. It is always a graph showing the performance of classification model and it is curve between True Positive Rate (TPR) and False Positive Rate (FPR).

TPR is also known as Recall.

$$TPR = \frac{TP}{TP + FN}$$

FPR

$$FPR = \frac{FP}{FP + TN}$$

We choose area under the ROC curve (AUC) as the performance metric which can test the correct number of classifications for a model. A model that has 100% correct classification has an AUC of 1.0 while 0.0 if all are wrong.

The following table below shows the accuracy of the classifier models and the speed it has taken to give the classification result.

Sr_no	Classification Model	Accuracy (AUC)	Speed (Time)	Average Rank (Accuracy & Speed)
1	Logistic Regression	0.9365	7.6 s	4
2	K Nearest Neighbour	0.9998	1.5e+02 s	1
3	Support Vector Machine	0.9991	2.7e+03 s	5
4	Stochastic Gradient Descent	0.9426	1.2e+02 s	4
5	Random Forest	1.0000	7.6e+02 s	2
6	XGBoost	0.9834	2.4e+02 s	3
7	ADABoost	0.9689	5e+02 s	5
8	Voting Ensemble	0.9536	3e+03 s	6

Table 1: Results of various classification model

We have developed the predictive models on data generated through random oversampling using SMOTE. This prevents the data from loss. The voting ensemble is made of four algorithms, Logistic Regression, K Nearest Neighbour, Support Vector Machine and Stochastic Gradient Descent.

As per the accuracy, the order of algorithms with the highest accuracy to lowest is Random Forest, K Nearest Neighbour, Support Vector Machine, XGBoost, AdaBoost, Voting Ensemble, Stochastic Gradient Descent and Logistic Regression.

As per the speed, the order of algorithms with the highest speed to lowest is Logistic Regression, Stochastic Gradient Descent, K Nearest Neighbour, XGBoost, AdaBoost, Random Forest, Support Vector Machine and Voting Ensemble.

We need to take into both accuracy and speed of detection into consideration as trade-off can be done between speed and accuracy as it is very important for banks to have both accuracy and speed. As per both speed and accuracy, the order of algorithms from highest to lowest is K Nearest Neighbour, Random Forest, XGBoost, Logistic Regression, Stochastic gradient descent, Support Vector Machine, AdaBoost and Voting Ensemble.

It can be seen ensemble models are not always the best predictive models mostly as they take time to build though they give good accuracy. They are best to be used in certain cases where they give large improvement in accuracy than single classifier.

From the above table we can see that K nearest Neighbour has the best accuracy and speed of detection. The following graphs below show the comparative analysis of the above algorithms with respect to the ROC curve and time.

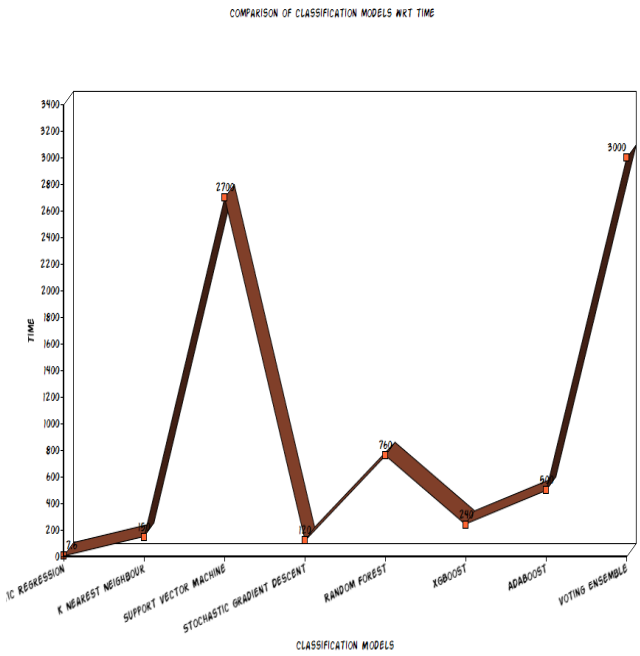

Figure 24: Comparison of Classification Models with respect to time.

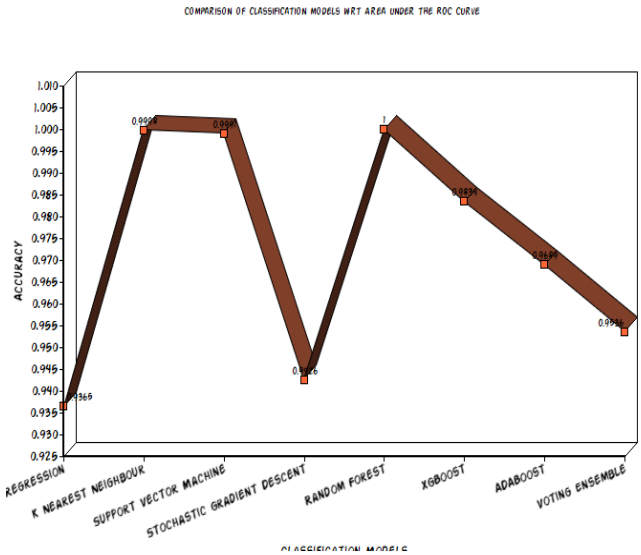

Figure 25: Comparison of Classification Models with respect to AUC

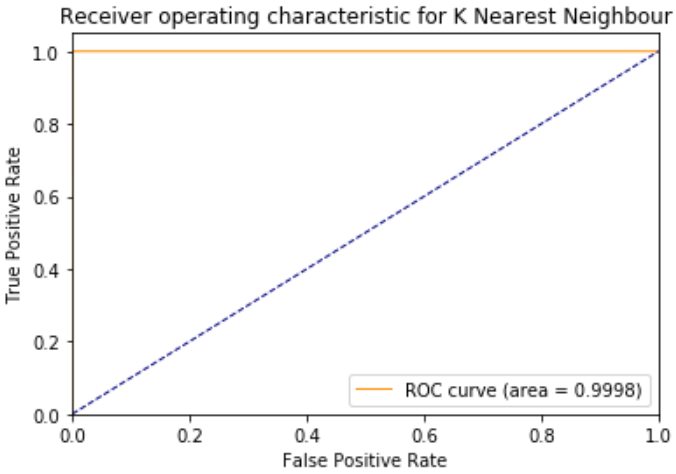

Figure 26: ROC for KNN

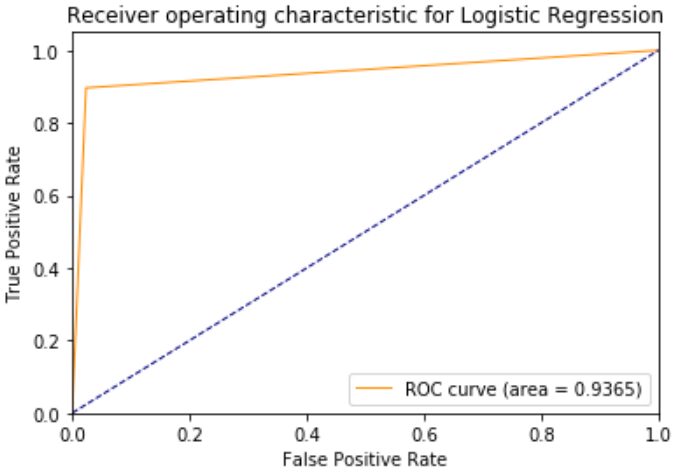

Figure 27: ROC for Logistic Regression

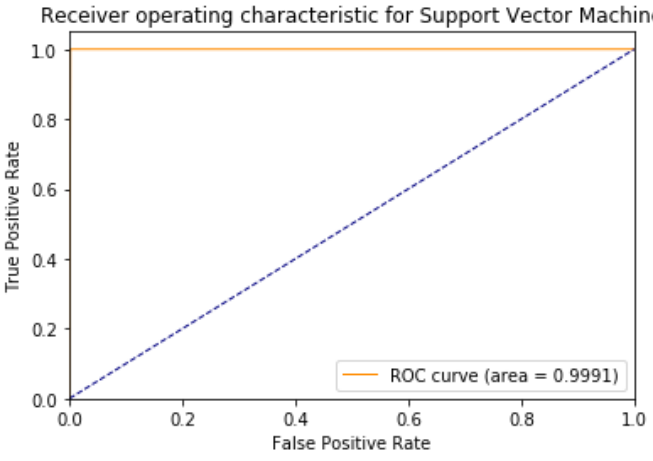

Figure 28: ROC for SVM

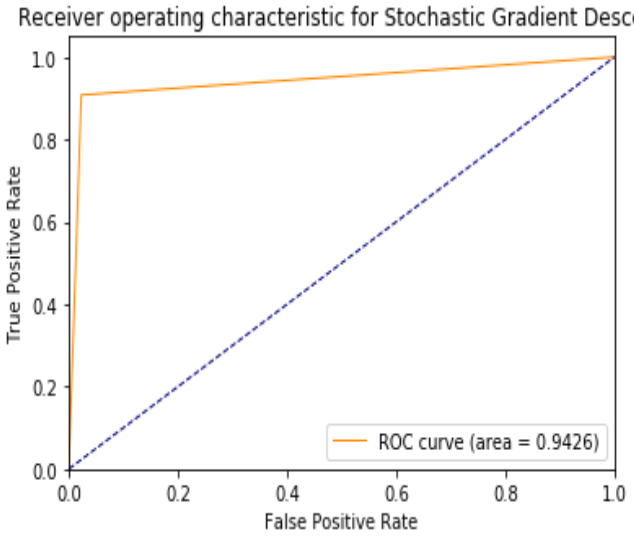

Figure 29: ROC for Stochastic Gradient Descent

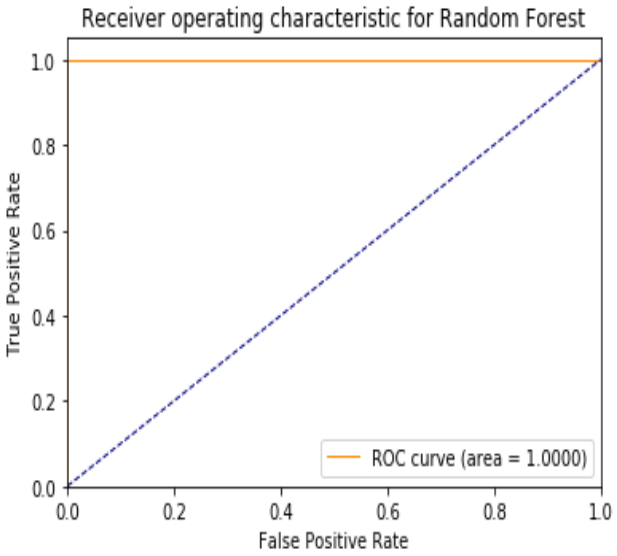

Figure30: ROC for Random Forest

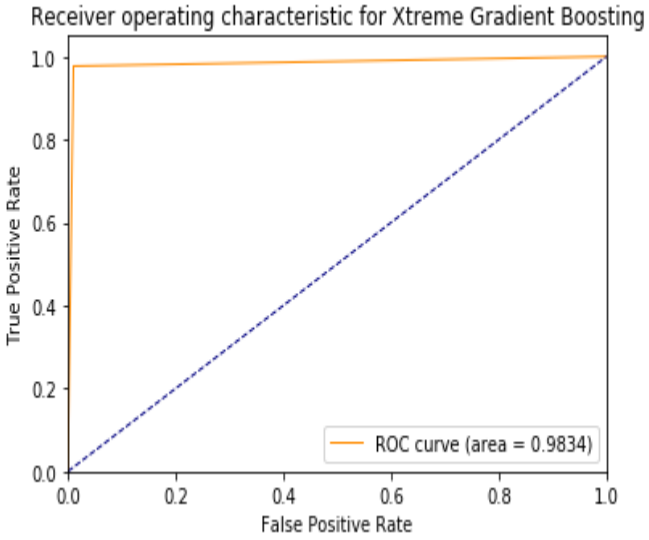

Figure 31: ROC for Xtreme Gradient Boosting

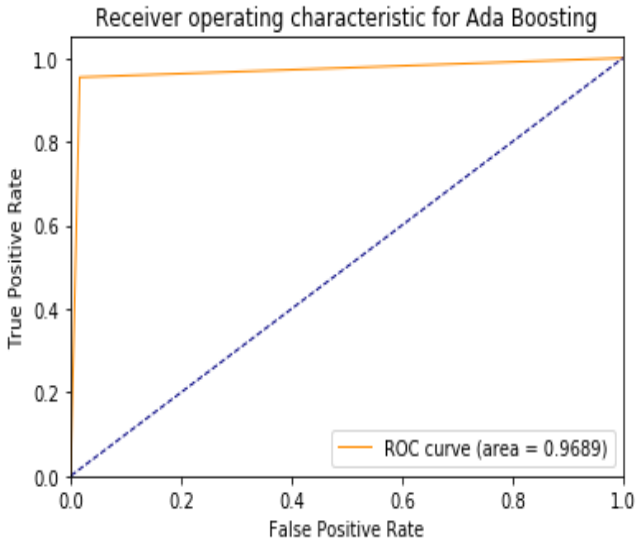

Figure 32: ROC for Ada Boosting

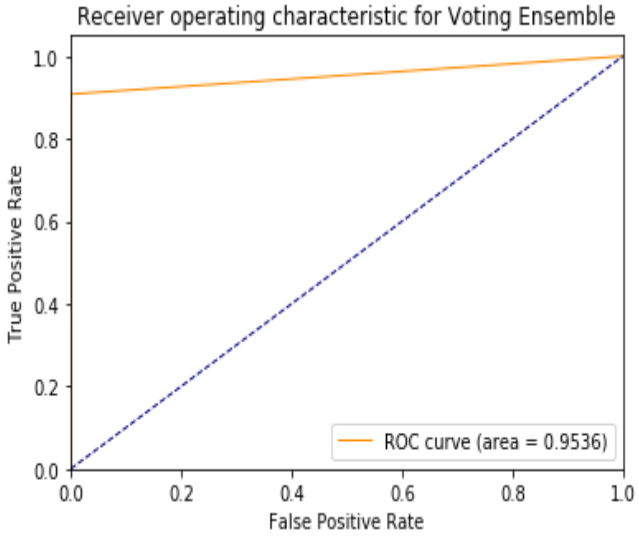

Figure 33: ROC for Voting Ensemble

MODEL DEPLOYMENT.

The project has been developed using python 3.6 and deployed on Jupyter Notebook. It further uses libraries like Pandas for data manipulation, imblearn for data balancing, Scikit- Learn for predictive modelling for machine learning algorithms and evaluation, NumPy for data manipulation, Matplotlib and Seaborn for Data Visualization and Keras for developing deep learning model.

CONCLUSION AND FUTURE WORK

So, when a dataset is imbalanced, we need to convert it into a balanced dataset and use area under the curve as the evaluation metric.

Detecting fraudulent credit card transactions is a tedious process which is subjected to change with varying patterns of the fraudsters and building a 100% accurate fraud detection system is in research with various state of art and more powerful approaches coming into picture like deep learning which has the ability to detect features or changes in the data using deep neural networks in the run rather than it being

provided features unlike machine learning models. So, the future work can be how to improvise the model with deep learning and build a model that adapts the changes in the data on the go which then can be useful for building a more powerful credit card fraud detector system.

REFERENCES

[1] Analysis of Credit Card Fraud Detection Techniques: Based on Certain Design Criteria, International Journal of Computer Applications (0975 – 8887).

[2] Application of credit card fraud detection: based on bagging ensemble classifier, International conference on Intelligent Computing, Communication and Convergence, Elsevier.

[3] https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/

[4] https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-for-ensemble-models/

[5] https://machinelearningmastery.com/ensemble-machine-learning-algorithms-python-scikit-learn/

[6] https://www.3pillarglobal.com/insights/credit-card-fraud-detection

[7] https://towardsdatascience.com/detecting-financial-fraud-using-machine-learning-three-ways-of-winning-the-war-against-imbalanced-a03f8815cce9

[8] https://weiminwang.blog/2017/06/23/credit-card-fraud-detection-using-auto-encoder-in-tensorflow-2/

[9] https://www.datascience.com/blog/fraud-detection-with-tensorflow

[10] https://www.linkedin.com/pulse/analyzing-transaction-data-like-scientist-taha-mokfi/

[11] https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

[12] https://medium.com/@curiousily/credit-card-fraud-detection-using-autoencoders-in-keras-tensorflow-for-hackers-part-vii-20e0c85301bd

[13] https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798

[14] https://machinelearningmastery.com/logistic-regression-for-machine-learning/

[15] https://www.quora.com/Why-is-logistic-regression-considered-a-linear-model

[16] https://aitrends.com/ai-insider/support-vector-machines-svm-ai-self-driving-cars/

[17] https://stackoverflow.com/questions/45455209/is-stochastic-gradient-descent-a-classifier-or-an-optimizer

[18] https://machinelearningmastery.com/assessing-comparing-classifier-performance-roc-curves-2/