

I. Introduction

The most prominent form of payment is a credit card. Credit card is a small thin plastic or fiber card that contains information about the person such as picture or service to his linked account charges for which will be debited regularly. Now a day's card information is read by ATM's, swiping machines, store readers, bank and online transaction. Each card has a unique card number which is very important, its security is mainly relies on physical security of the card and also privacy of the credit card number.

Credit card fraud is regarded as an illegal activity in which anyone tries to use the physical card information without the consent and knowledge of the cardholder. Credit card can be dealt in two ways: one is online fraud that can be detected through mobile phone, internet, web, shopping and other is offline fraud that detects the card which is stolen by using their personal details [3]. Credit card fraud is one of the malicious activities that occur in an online transaction. Generally, credit card fraud refers to the unauthorized access of credit or debit card for making payments. These are simply the fraudulent source of funds used in different transactions.

According to Federal Trade Commission (2020, 2021). Just in USA alone credit card fraud went up 44.7% from 2019 to 2020. It makes the credit card second most common type of identity theft reported. Fraud is at an alarming rate with the emergence of technologies causing huge financial loss.

This is a very relevant problem that demands the attention of machine learning and data science communities where the solution to this problem can be automated. This problem is particularly challenging from the perspective of learning, as it is characterized by various factors such as class imbalance.

The banking fraud can't be completely but at least we can prevent from happening and its occurrence to certain level using machine learning techniques and Deep learning techniques. In this paper, we compare performance of different algorithms to Deep learning mode like Autoencoders.

II. Literature Review

Zarrabi et.al[1] . Deep Auto encoder is proposed by the author that serves as a best extraction of the details of the features from the fraud transaction occurred in the credit card. For the class mark problems, softmax tools used here by the author. For classifying a form of fraud an AutoEncoder is used here which maps the data into a high-dimensional space. Deep learning can be said as one of the most effective methods for detecting the credit card fraud. To understand the dynamic distribution of the data in the networks types becomes difficult to understand. For extraction the best features of data with a high amount of precision and low variance the networks via Deep Auto encoder was used.

K.Ratna Sree Vall et .al [2]. Supervised learning, provided the unexpected input example of the associate degree, and is designed to perform predictions. The supervised methods used in this paper are, Random Forest, Logistic Regression, Naive Bayes and a boosting technique (AdaBoost) to enhance the classification algorithm. AdaBoost or Adaptive Boosting is a boosting method, which provides us with a single "strong classifier" with the combination of multiple "weak classifiers". It was therefore concluded here that compared to logistic regression and even the Naïve Bayes techniques with a boosting technique, the random forest classifier is stronger.

Francis Cheon Dong Hee Lee Han Seon Joo Ook Lee et.al[3] Deep learning hybrid based approach is suggested by the author using LSTM and Bi-LST network.

John and Naaz et.al[4] used local outlier factor and Isolation Forest to get the higher accuracy and they obtained the accuracy of 97% by outlier factor and 76% by Isolation Forest.

Rinky D. Patel and Dheeraj Kumar Singh et.al[5] discuss about class imbalance and how to handle it and also discuss how to work on large dataset. The implemented work was overcome these challenges.

III. Dataset

Our dataset is from Kaggle, and the dataset is available from [https://www.kaggle.com/mlg-ulb/creditcardfraud\[7\]](https://www.kaggle.com/mlg-ulb/creditcardfraud[7]). The dataset is from transactions of a European bank in September 2013 that are CSV format. It contains 492 (0.17%) fraud transactions and 284,807 (99.83%) genuine transactions. Due to privacy and confidentiality many background information is not provided only PCA transformed data is given. Only time and amount are not transformed to PCA all other given values v1, v2, v3 v28 are PCA transformed numeric values .The dataset is composed of 31 numerical features and V1, V2... V28 are from the Principal Component Analysis (PCA). Among the features, "Time" represents the time between the first transaction and other transactions. "Amount" indicates the total amount of transactions from the credit card. The target column is the "Class" that contains only 1 representing fraud transactions, and 0 for genuine transactions. Therefore, the binary classification should be conducted for treating the dataset



Fig-1:Showing Imbalance of data in percentage

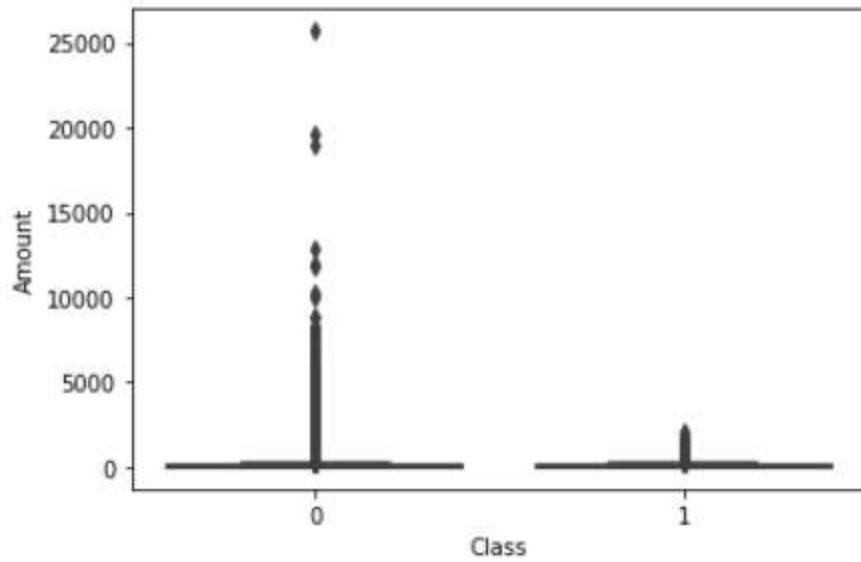


Fig-2: Showing Imbalance of class with respect with Amount

About this file

BigQuery Table `bigquery-public-data.fraud_detection.comments`

# Time	# V1	# V2	# V3	# V4	#
Number of seconds elapsed between this transaction and the first transaction in the dataset	may be result of a PCA Dimensionality reduction to protect user identities and sensitive features(v1-v28)				
0 1724	-56.4 2.45	-72.7 22.1	-48.3 8.38	-5.68 16.9	-1
0	-1.3598871336738	-0.8727811733898497	2.53634673796914	1.37815522427443	-E
0	1.19185711121486	0.26615871285963	0.16648811235321	0.448154878460911	0.
1	-1.35835486159823	-1.54816387473689	1.77328934263119	0.379779593034328	-E
1	-0.966271711572887	-0.18522688882898	1.79299333957872	-0.863291275836453	-E
2	-1.15823389349523	0.877736754848451	1.548717846511	0.483833933955121	-E
2	-0.425965884412454	0.968523844882985	1.14118934232219	-0.168252879768382	0.
4	1.22965763458793	0.141883587849326	0.8453787735899449	1.28261273673594	0.
7	-0.644269442348146	1.41796354547385	1.8743883763556	-0.492199818495815	0.
7	-0.89428688228282	0.286157196276544	-0.113192212729871	-0.271526138888604	2.

Fig-3 : Showing dataset page from Kaggle

IV. Algorithms used

Logistic Regression->Logistics Regression is less inclined to the overfitting, but it can overfit in high-dimensional datasets. We can consider regularization techniques to avoid overfitting. Any big outliers will be transform into the range of 0 and 1. Its help mainly to solve classifications problem and supply us the knowledge whether the event is happening or not.

XGBoost -> XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.

RandomForestClassifier ->A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

Adaboost-> An AdaBoost [1] classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases.

LGBMClassifier ->LightGBM is a fast, distributed, high performance gradient boosting framework based on decision tree algorithms, used for ranking, classification and many other machine learning tasks

KNN ->. It is a supervised machine learning algorithm. The algorithm can be used to solve both classification and regression problem statements. The number of nearest neighbours to a new unknown variable that has to be predicted or classified is denoted by the symbol 'K

GaussianNB ->In this classification method, the probability of an object associated, with a particular category or class with certain feature is learned. Naive Bayes algorithm can fit model fast and provide high accuracy when applied to big data and need less training.

ANN -> A collection of nodes are interlinked which is designed to depict the functioning similar to the human brain which is known as Artificial Neural Network. All the other nodes that are present in the adjacent layers of the node are being assigned the weighted connection.

Autoencoders ->Autoencoder has a structure of feed-forward neural network that consists of the same input and out dimension. Autoencoder simply copies the input to the output .

Setting a limitation to the neural network can make this simple neural network more complicated. Representative constraint is to limit the number of neurons between input and output. This constraint has two advantages. Firstly, it would prevent autoencoders from simply copying inputs to the output. Secondly, it would learn how to represent data more effectively. This simple autoencoder is defined as an under complete autoencoder, which is a symbolic model of the autoencoder. The autoencoder incorporates two separate parts, which are the encoder and decoder. The encoder converts inputs into inner representations. The decoder transforms inner representations into outputs

Isolation Forest ->Isolation Forest is an unsupervised anomaly detection learning algorithm that operates on the concept of isolating anomalies. The Isolation Forest algorithm aims to make it easier to separate anomalous instances in a dataset from the rest of the sample (isolate), relative to usual points. The algorithm recursively creates partitions in the sample in order to isolate a data point by randomly selecting an attribute and then randomly selecting a split value for an attribute between the minimum and maximum values permitted for that attribute.

LocalOutlineFactor ->Local Outlier Factor is an algorithm used for the identification of unattended outliers. It creates an anomaly score that reflects data points in the dataset that are outliers. It does this by calculating a given data points local density variance with respect to the data points around it.

v. Result and Experiment analysis

A comparison table was prepared to compare different machine learning and deep learning model. We calculated precision, recall, ROC curve and accuracy with the help of inbuilt models present in Sklearn like accuracy and AUC_Score.

Accuracy: It represents the fraction of the total number of transactions that have been detected correctly (fraudulent and nonfraudulent).

True Positive: It is the number of transactions to represent fraudulent transactions were correctly classified as fraudulent.

False Positive Rate: The false positive rate is represented as the ratio between the number of negative events wrongly categorized as positive (false positives) and the total number of actual negative events.

Precision: Precision represents precise/accuracy of model. Precision is a good measure to determine when the cost of false positive is high.

Recall: Recall calculates how many of the Actual Positives our model capture through labeling it as Positive (True Positive). It should be the model metric that we use our best model when there is high cost associated with False Negative.

ROC curve: It stands for Receiver Operating Characteristic (ROC) curve. In this, for different cut-off points, the true positive rate (Sensitivity) is plotted in function of the false positive rate (100-Specificity). If the ROC curve is closer to the upper left corner, then the overall accuracy of the test will be high. The area under the ROC curve (AUC) is a measure that parameter can distinguish between two groups (fraudulent/nonfraudulent)

Due to the nature of the dataset , first dataset was applied without any oversampling then it was applied with oversampling

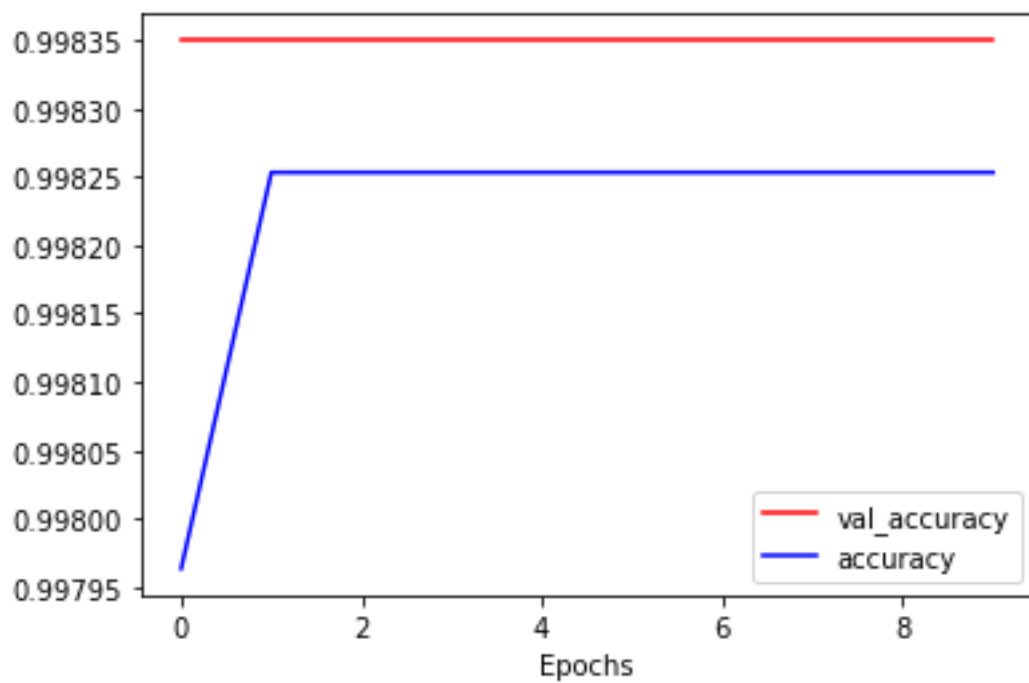
Table 1

Without Sampling

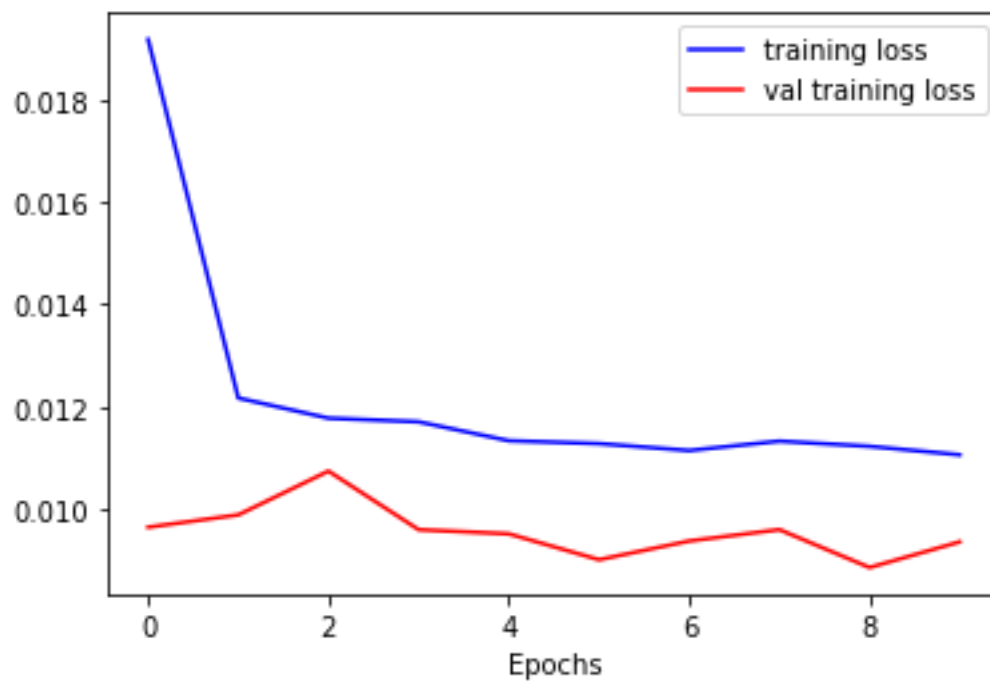
	Accuracy	Precision	Recall	ROC_score
Logistic Regression	99.92	0.994	0.998	0.8350
XGBclassifier	99.96	0.9997	0.9986	0.9095
RandomForestClassifier	99.97	0.9997	0.9986	0.9148
AdaboostClassifier	99.94	0.9996	0.9986	0.8988
LGBMClassifier	99.92	0.9994	0.9988	0.8350
Kneighbors	99.95	0.9996	0.9987	0.8829
GaussianNB	97.70	0.9997	0.9985	0.9035
ANN	99.95	0.9996	0.9987	0.8829
DenseAutoEncoder	0.9983			0.9585
LocalOutlierFactor	0.95(class 0) 0.31(class 1)	1(class 0) 0.18(class 1)	1(class 0) 0.18(class 1)	-
IsolationForest	0.95(class 0) 0.83(class 1)	0.95(class 0) 0.83(class 1)	1 (class 0) 0.04 (class 1)	-

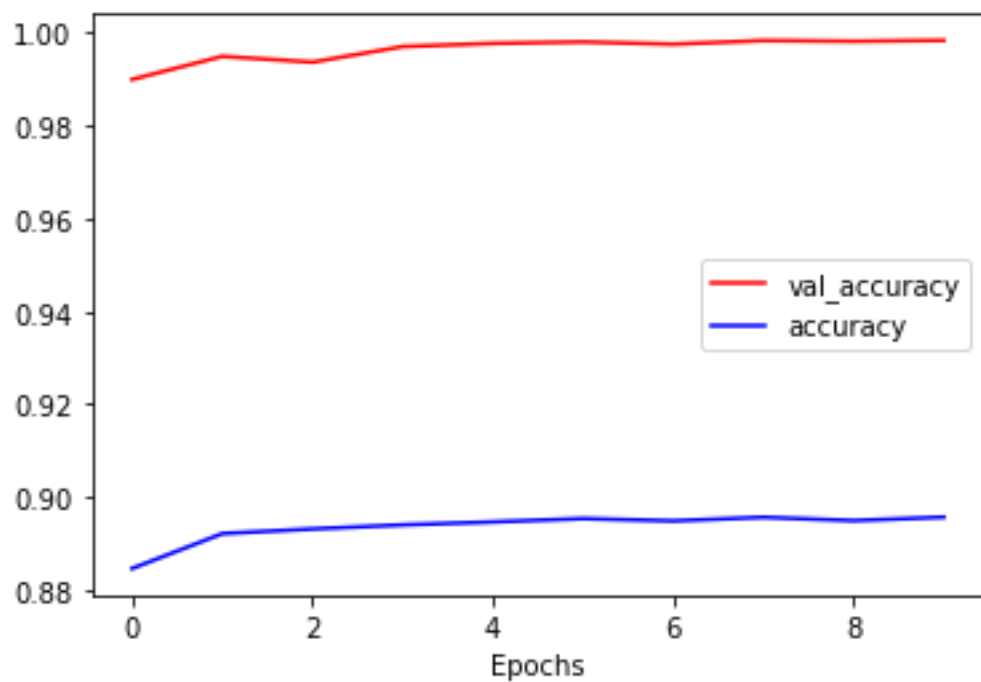
Table 2 :After sampling

	Accuracy	Precision	Recall	ROC_score
Logistic Regression	97.41	0.9998	0.9984	0.9445
XGBclassifier	99.07	0.9997	0.9985	0.9095
RandomForestClassifier	99.95	0.9997	0.9986	0.9094
AdaboostClassifier	97.71	0.9998	0.9985	0.9301
LGBMClassifier	98.18	0.9998	0.9984	0.9431
Kneighbors	99.76	0.9997	0.9985	0.9297
GaussianNB	97.39	0.9997	0.9985	0.9073
ANN	99.90	0.9996	0.9986	0.8829
DenseAutoEncoder	99.82			0.9598
LocalOutlierFactor	0.95(class 0) 0.31(class 1)	1(class 0) 0.18(class 1)	1(class 0) 0.18(class 1)	-
IsolationForest	0.96(class 0) 0.78(class 1)	0.95(class 0) 0.83(class 1)	1 (class 0) 0.04 (class 1)	-

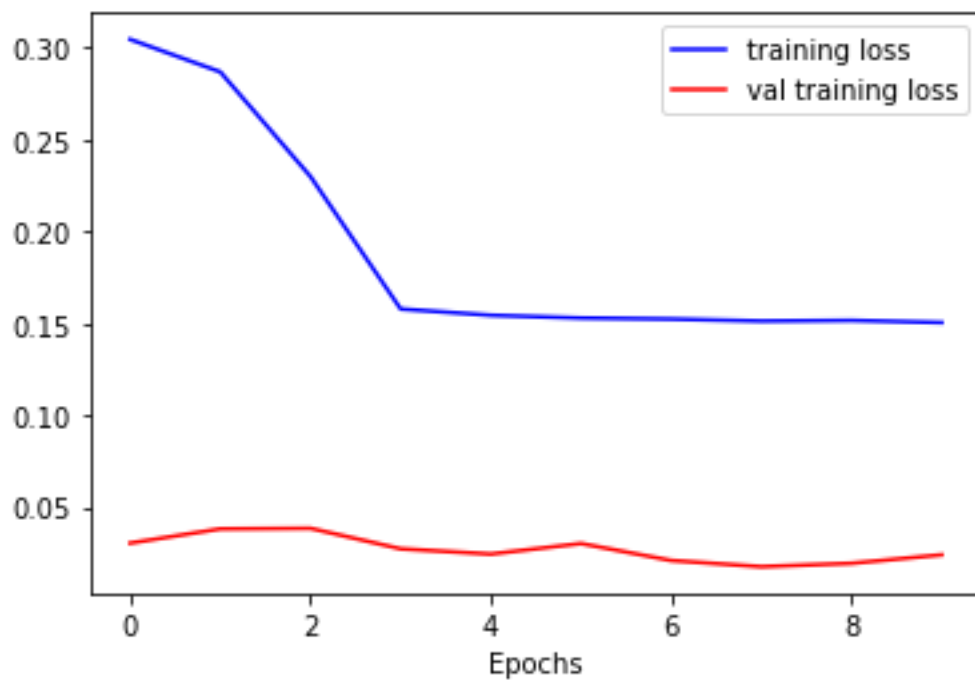


For unbalanced dataset





For balanced dataset



VI. Conclusion and Future Scope

The credit card fraud detection methods have gained popularity in the past decade with the evolution of statistical models, machine learning algorithms, data mining techniques. The fraud transaction prediction has 2 phases which are feature extraction and classification. Within the first phase, the feature extraction technique is applied and within the second phase, classification is applied for fraud transaction detection, Fraud transaction detection is that the major issue of prediction because of a frequent and enormous number of transactions. During this comparative research study, we tried to analyze the dataset through various graphs and also tried to detect fraud using some classification algorithms and make comparative analysis.

As we can see from the data above simple algorithm like Logistic Regression performs same as compared to much dense Deep learning model like AutoEncoder. And even after balancing the imbalance between classes performance of various algorithms only improve by a little margin.

As far as ROC score is considered for evaluation before balancing the dataset AutoEncoder had the highest number(0.9585) while RandomForest classifier came second with 0.9148 . But After balancing the train dataset first place was maintained by AutoEncoder and surprisingly LogisticRegression took the second place with value(0.9445).

More surprising thing is that Logstic Regression take very less time to train compared to Deep AutoEncoder but both have almost similar result.

In future Deep learning models like LSTM, BiLSTM , And unsupervised learning methods like clustering or even a combination of both could be use but more importantly a balanced dataset could be used to properly measure their performance.

VII. References

1. Zarrabi, H. Kazemi, "Using deep networks for fraud detection in the credit card transactions, "IEEE 4th International Conference In Knowledge-Based Engineering and Innovation (KBEI), pp. 0630-0633, 2017
2. Khyati Chaudhary, Jyoti Yadav, Bhawna Mallick, "A review of Fraud Detection Techniques: Credit Card", International Journal of Computer Application, Volume 45- No.1 2012.
3. MIN JONG CHEON, 2DONG HEE LEE, 3HAN SEON JOO, OOK LEE, "DEEP LEARNING BASED HYBRID APPROACH OF DETECTING FRAUDULENT TRANSACTIONS" Journal of Theoretical and Applied Information Technology 31st August 2021. Vol.99. No 16
4. Credit Card Fraud Detection Based on Transaction Behaviour -by John Richard D. Kho, Larry A. Veal" published by Proc. of the 2017 IEEE Region 10 Conference (TENCON), Malaysia, November 5-8, 2017 [2] CLIFTO
5. Rinky D. Patel and Dheeraj Kumar Singh, "Credit Card Fraud Detection & Prevention of Fraud Using Genetic Algorithm", published by International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-6, January 2013.
6. Aditya Saini, Swarna Deep Sarkar, Shadab Ahmed, SP Maniraj " Credit Card Fraud Detection using Machine Learning and Data Science", International Journal of Engineering Research & Technology, ISSN: 2278-0181, Vol. 8 Issue 09, September-2019