iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** V **Month of publication:** May 2019

DOI: https://doi.org/10.22214/ijraset.2019.5193

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

Credit Card Duplicity Reduction System Using Classification Metrics

Vaka Sravani[1], Dr. V. Murali Krishna[2]
[1]*M.Tech Student,* [2]*Assist. Prof, CSE, Vaagdevi college of Engineering, Warangal, TS*

Abstract: ***Recognizing cheats in Credit card exchanges is maybe a standout amongst the best test beds for computational knowledge calculations. Actually, this issue includes a few significant difficulties, to be specific: idea float (clients' propensities develop, and fraudsters change their methodologies after some time), class lopsidedness (real exchanges far dwarf fakes), and confirmation inactivity (just a little arrangement of exchanges are opportune checked by investigators).most of learning calculations that have been proposed for misrepresentation recognition depend on suspicions that barely hold in a genuine "Fraud Detection System" (FDS). Advanced classification methods provide the ability to detect fraudulent transactions without disturbance of legitimate transactions and unnecessarily spent resources on fraud forensics for financial institutions. However, it is significant that Credit Card organizations can perceive fake exchanges so clients are not charged for things that they didn't buy the point of this undertaking is to distinguish whether a Credit Card exchange is false or not utilizing Logistic Regression, Neural Network and Decision tree models and to look at the exhibitions of these models. Our dataset is PCA transformed to protect client privacy and is also highly imbalanced. We aim to identify potential for further feature reduction and feature engineering and overcome the data imbalance to improve model accuracy.***

Keywords: Credit card frauds, neural networks, Decision Tree, Logistic Regression

I. INTRODUCTION

In 2015, credit, debit and prepaid cards generated over $31 trillion in total volume worldwide with fraud losses reaching over $21 billion [1]. In that same year there were over 225 billion purchase transactions, a figure that is projected to surpass 600 billion by 2025 [2]. Fraud associated with credit, debit, and prepaid cards is a significant and growing issue for consumers, businesses, and the financial industry. Historically, software solutions used to combat credit card fraud by issuers closely followed progress in classification, clustering and pattern recognition [3, 4]. Today, most Fraud Detection Networks (FDS) continue to use increasingly in a genuine world FDS, the gigantic stream of installment demands is immediately checked via programmed instruments that figure out which exchanges to approve. These exchanges stay unlabeled until clients find and report fakes, or until an adequate measure of time has slipped by with the end goal that no contested exchanges are considered true. Sophisticated machine learning algorithms to learn and detect fraudulent patterns in real-time, as well as offline, with minimal disturbance of True transactions [5]. Generally, FDS need to address several inherent challenges related to the task: extreme un balanced of the dataset as frauds represent only a fraction of total transactions, distributions that evolve due to changing consumer behavior, and assessment challenges that come with real time data processing [6]. For example, difficulties arise when learning from an unbalanced dataset as many machine intelligence methods are not designed to handle extremely significant differences between class sizes [5]. Also, dynamic trends within the data require robust algorithms with high tolerance for concept drift in legitimate consumer behaviors [7]. Although specialized techniques exist that may handle large class imbalance such as outlier detection, fuzzy inference networks and knowledge-based networks [8, 9], current state-of the- art research suggests that conventional algorithms in fact may be used with success if the data is sampled to produce equivalent class sizes [10, 11, 12]. It is important that credit card companies can recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase.

A. Dataset

The credit card fraud dataset used in this paper is obtainable from Kaggle.com [13] and contains a subset of online European credit card transactions made in September 2013 over a period of two days, consisting of highly imbalanced 492 frauds out of 284 807 total transactions [5, 6, 7, 14, 15]. For confidentiality the dataset is simply provided as 28 unlabeled columns resulting from a PCA transformation. Additionally, there are three labeled columns: Class, Time, and Amount.
Note that the dataset is highly imbalanced towards legitimate transactions which have the label of "0", visible in fig 1

```
data.Class.value_counts()
0    284315
1       492
Name: Class, dtype: int64
```

Fig 1: Class imbalance in the Kaggle credit-card fraud dataset

II. METHODS

1) *Imbalance Learning:* Standard decision trees such as ID3 and C4.5 use information gain as the splitting criterion for learning which results in rules biased towards the majority [3]. Research also shows that imbalanced datasets pose a problem for kNN, neural networks (NN), and support vector machines (SVM) [3, 16, 17]. This problem is most pronounced when the two classes overlap as in the case of the Kaggle dataset; the majority of machine intelligence algorithms are not suited to handle both class unbalanced and overlapped class distributions [3, 10, 11]. Fortunately, algorithms exist that can take class imbalance into account. In addition, there are techniques at the data level and algorithm level which can reduce the negative effects of these biases.
2) *Concept Drift:* Credit card fraud is prone to concept drift as consumer trends change due to changing preferences, seasonality and new products, as well as evolving fraud attack strategies [7]. The net effect of this is that the statistical properties of the underlying data change over time. Recent research has shown that it is possible to overcome this while still maintaining conventional machine intelligence techniques. For example, a sliding window approach where a classifier is trained every day on the most recent samples, or an ensemble approach where the oldest component is replaced with a new classifier [7]. However, for the purposes of this paper the challenges associated with concept drift, as well as their resolutions, are not explored for two reasons. The first is that the dataset used is collected over a period of two days, which may not be sufficient for concept drift to take place. The second reason is that as mentioned, research shows that concept drift in FDS may be appropriately tackled by using conventional methods that are employed to maintain only a local temporal memory of learned attributes [7]. In other words, once a method is found that performs well for short periods of time, i.e. sufficiently small periods of time where concept drift does not take place, its implementation can then be further improved to account for concept drift. Therefore the fraud detection methods explored in this paper would need further refinement before being applied to a data stream of longer than a handful of days. These refinements are discussed in more detail at the end of the paper.
3) *Sampling:* Sampling methods are used to compensate for the unbalancedness of the dataset by reducing the classes to near equivalence in size. Under sampling and oversampling are two roughly equivalent and opposite techniques which use a bias to achieve this purpose. More complex algorithms such as synthetic minority oversampling technique (SMOTE) and the state-of-the-art adaptive synthetic sampling approach (ADASYN) actually create new data points based on known samples and their features instead of simply replicating the minority class [3, 18]. However these algorithms rely on assumptions of the minority class and are generally computationally expensive. Particularly, the created data generally is an interpolation of prior data which may not actually provide a realistic approximation of if the classes were in fact, balanced. Despite this, sampling methods can provide a more robust approach to imbalance learning than other methods, for example cost-based techniques which penalize errors differently depending on class such that the minority class is favoured [12, 19]. For example, what cost to use? In either case, sampling using ADASYN on the training data is compared with classic under sampling in prior research, and no sampling at all. The package used is “Imbalance-Learn” in Python2.7 [20]. The testing and validation data are not sampled such that reported final accuracies are not distorted. This is representative of real-life where the fraudulent cases would be in the extreme minority class. Finally, these results will be compared to cost-based balancing methods, if applicable. 4) Classification: Most FDS use supervised classification techniques to discriminate between fraudulent and legitimate transactions. Research on similar credit-card fraud datasets shows Random Forest (RF) having superior performance than kNN and SVM when using under sampling to correct for class imbalance [6]. This finding is verified by exploring three general classes of techniques against the Kaggle dataset: linear methods, ensemble methods, and neural networks. Specifically, linear SVM, RF and MLP (multi-layer perceptrons) with training data subjected to ADASYN are explored in this paper.
4) *Decision Tree:* A choice tree is a graphical portrayal of explicit choice circumstances that are utilized when complex stretching happens in an organized choice procedure. A choice tree is a prescient model dependent on a fanning arrangement of Boolean tests that utilization explicit realities to make increasingly summed up ends. The fundamental segments of a choice tree include choice focuses spoken to by hubs, activities and explicit Decisions from a choice point. Each standard inside a choice tree is spoken to by following a progression of ways from root to hub to the following hub, etc until an activity is come to Decision

Trees are a well known and useful asset utilized for grouping and forecast purposes. Decision Trees give an advantageous option in contrast to survey and overseeing vast arrangements of business rules, permitting them be deciphered in a manner that enables people to get them and apply the standards limitations in a database with the goal that records falling into a particular class are certain to be recovered. Decision Trees for the most part comprise of the accompanying four stages: Structuring the issue as a tree by making end hubs of the branches, which are related with a particular way or situation along the tree Assigning subject probabilities to each spoken to occasion on the tree Assigning settlements for results. This could be a particular dollar sum or utility esteem that is related with a situation. Recognizing and choosing the suitable course(s) of activity dependent on investigations

5) *Neural Networks:* Neural Networks are a lot of calculations, displayed freely after the human cerebrum, that are intended to perceive designs. They decipher tactile information through a sort of machine discernment, marking or bunching crude info. The examples they perceive are numerical, contained in vectors, into which all true information, be it pictures, sound, content or time arrangement, must be interpreted. Neural Networks help us group and order. You can consider them a grouping and order layer over the information you store and oversee. They help to bunch unlabeled information as per similitudes among the model sources of info, and they arrange information when they have a named dataset to prepare on. (Neural Networks can likewise extricate highlights that are encouraged to different calculations for bunching and arrangement; so you can consider profound Neural Networks as segments of bigger AI applications including calculations for support learning, grouping and relapse.)

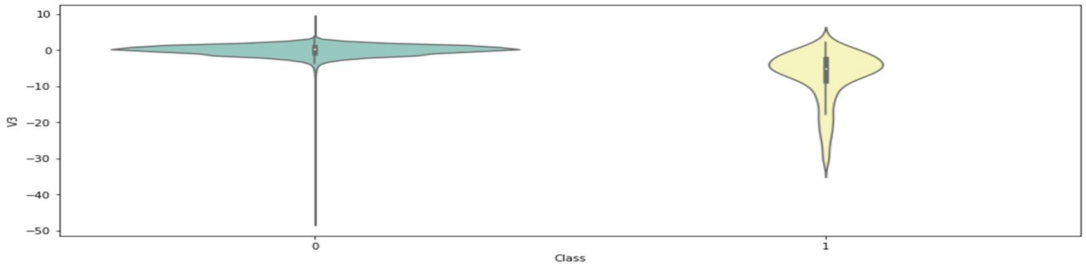
6) *Logistic Regression:* If you review Linear Regression, it is utilized to decide the estimation of a nonstop needy variable. Strategic Regression is commonly utilized for arrangement purposes. In contrast to Linear Regression, the needy variable can take a predetermined number of qualities just i.e, the needy variable is clear cut. At the point when the quantity of potential results is just two it is called Binary Logistic Regression. How about we take a gander at how strategic relapse can be utilized for characterization undertakings. In Linear Regression, the yield is the weighted whole of data sources. Strategic Regression is a summed up Linear Regression as in we don't yield the weighted entirety of sources of info straightforwardly, however we go it through a capacity that can delineate genuine incentive somewhere in the range of 0 and 1. On the off chance that we take the weighted aggregate of contributions as the yield as we do in Linear Regression, the esteem can be multiple yet we need an incentive somewhere in the range of 0 and 1. That is the reason Linear Regression can't be utilized for characterization undertakings. We can see from the underneath assume that the yield of the direct relapse is gone through an actuation work that can outline genuine incentive somewhere in the range of 0 and 1. We can see that the estimation of the sigmoid capacity dependably lies somewhere in the range of 0 and 1. The esteem is actually 0.5 at X=0. We can utilize 0.5 as the likelihood edge to decide the classes. On the off chance that the likelihood is more prominent than 0.5, we characterize it as Class-1(Y=1) or else as Class-0(Y=0). Before we manufacture our model we should take a gander at the suppositions made by Logistic Regression. The needy variable must be categorical. The free variables (features) must be autonomous (to dodge multicollinearity)

III. IMPLEMENTATION

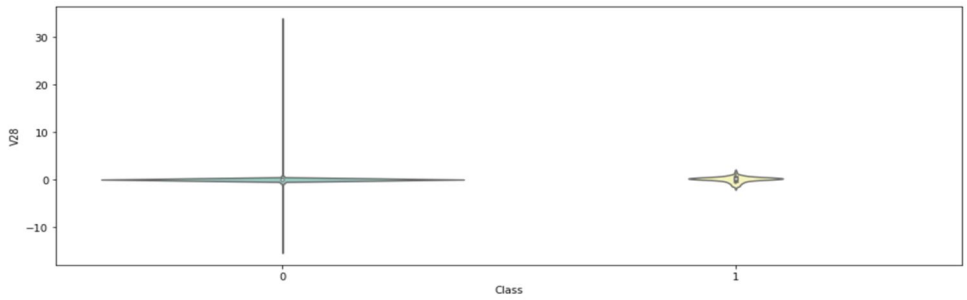
A. *Feature Engineering*

1) Feature engineering means that building further options out of existing information that is usually spread across multiple connected tables. Feature engineering needs extracting the relevant data from the information and obtaining it into one table which can then be utilized train a machine learning model. A transformation acts on one table (thinking in terms of Python, a table is simply a Pandas Data Frame) by making new advantages out of 1 or additional of previous columns. On another hand, aggregations are performed across tables, and use a one-to-many relationship to cluster observations then calculate statistics. for instance, if we have another table with data on the loans of clients, where every customer may have multiple loans, we will calculate statistics like the average, maximum, and minimum of loans for every client.

2) Visualizing the discrete variables using Violin Plots



The distribution for Class 0 and Class 1 are different for the v3 feature



3) The distribution for Class 0 and Class 1 are similar for the v28 feature.

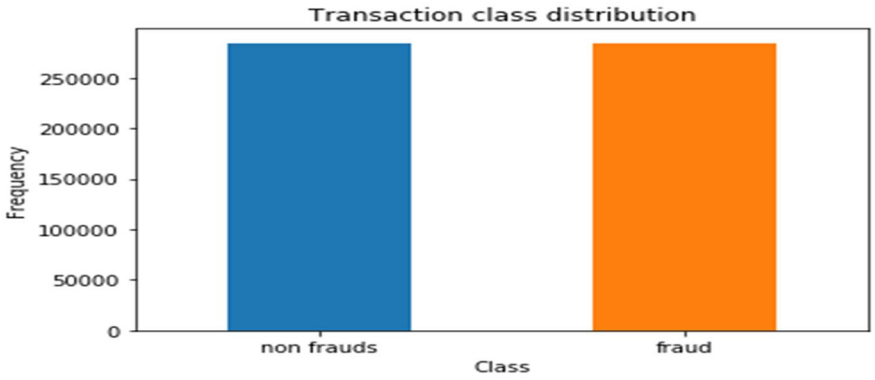
4) Drop all of the features that have very similar distributions between the two types of transactions, as they are not useful in further analysis.

This procedure involves grouping the loans table by the customer, evaluating the aggregations, and then merging the resulting information into the client information. Here's however we would do that in Python utilizing the language of Pandas

B. Handling Imbalance Data

1) Balanced data- Using up sampling method

Data now is ready for analysis

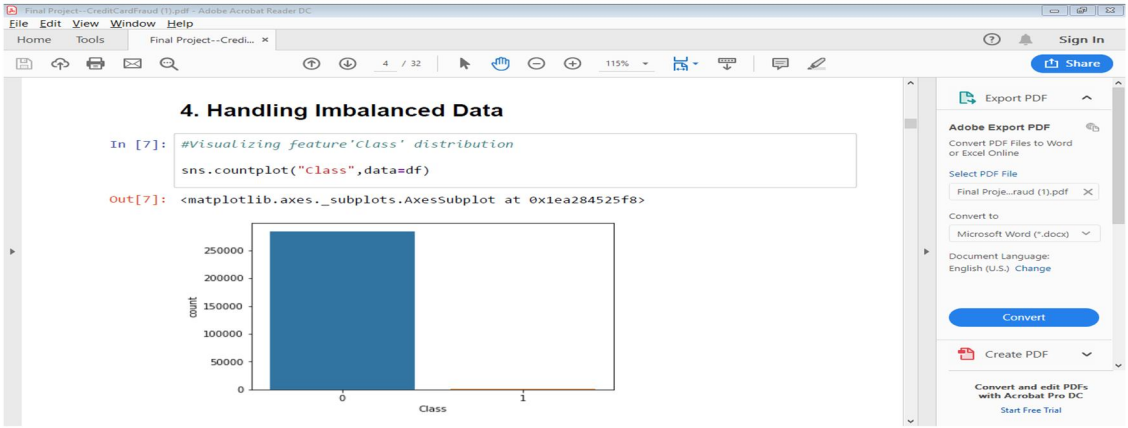


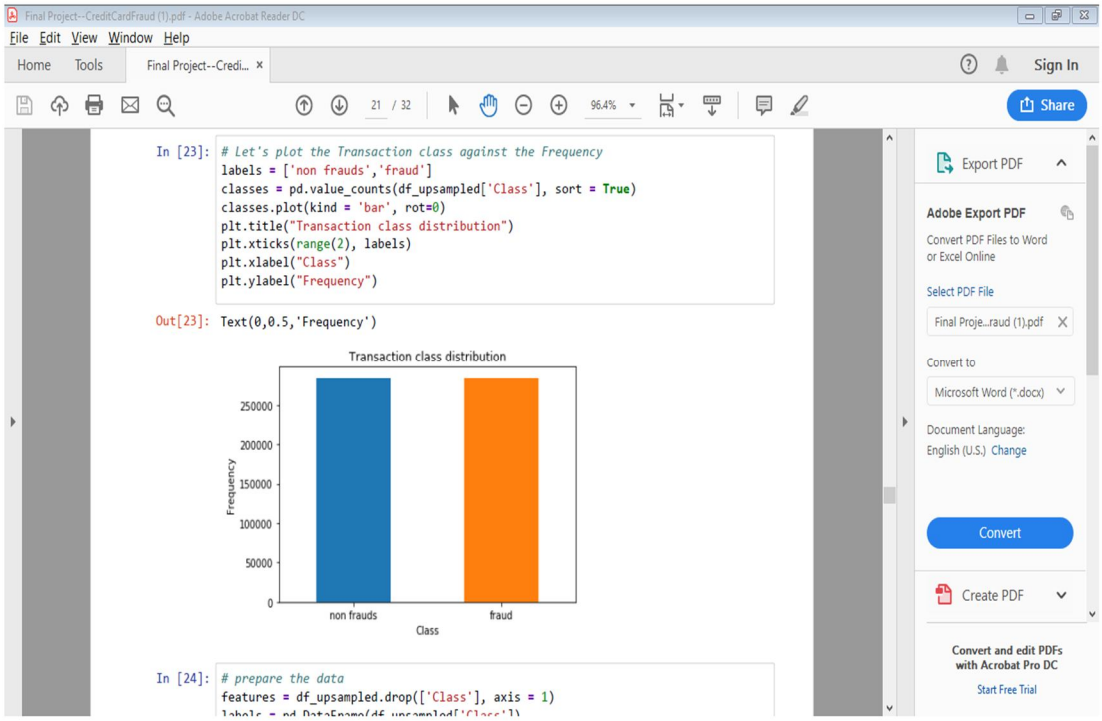
IV. OUTPUT SCREEN

A. Handling Imbalance Data

1) Under sampling:- it means taking the less number of majority class (In our case taking less number of Normal transactions so that our new data will be balanced

2) Oversampling: it means using replicating the data of minority class (fraud class) so that we can have a balanced data

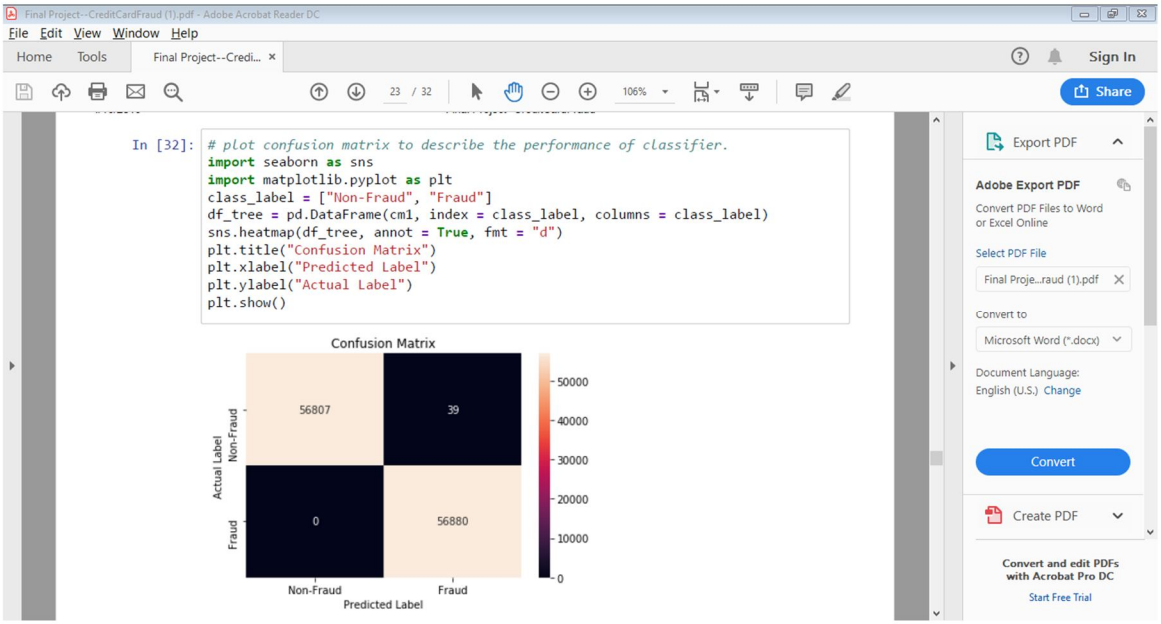




B. Decision Tree

Accuracy: It is calculated by correctly classified points divided by total no of points.Theese Model gives accuracy of 99.9% after using 10 fold cross validation with cross validation score 99%

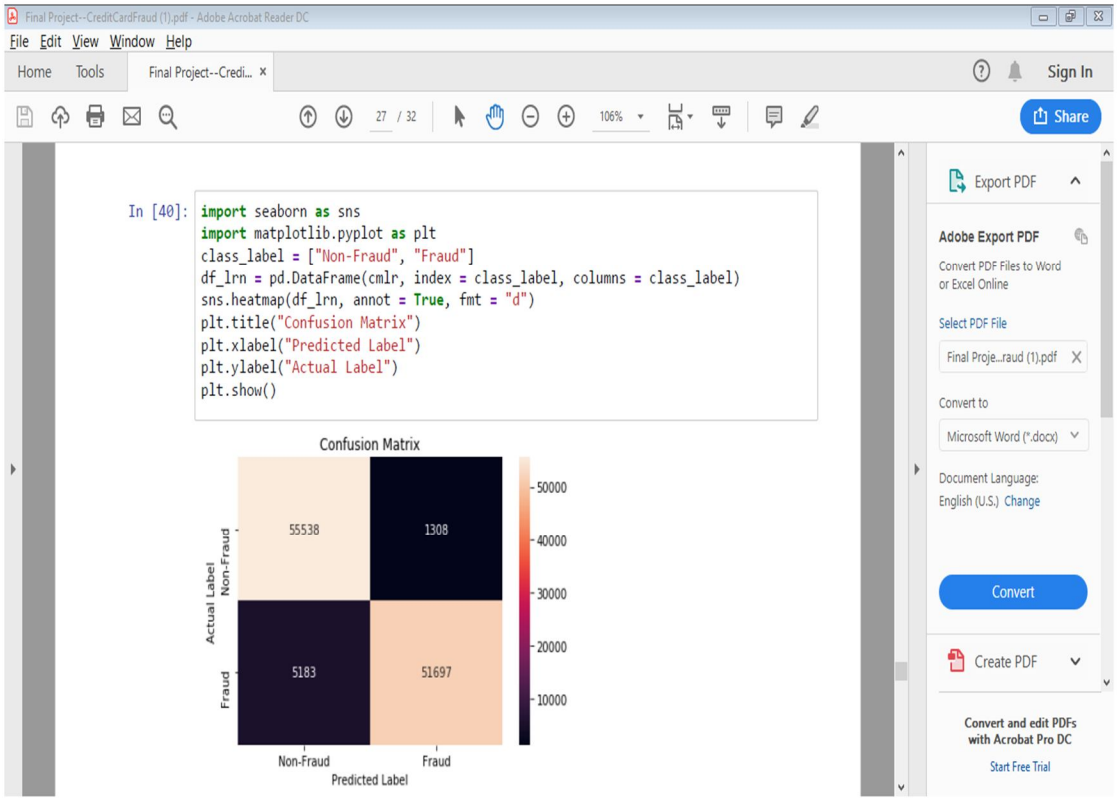
Sensitivity or Recall: It gives the recall of 100% by predicting 56880 people as "fraud" out of all the people. While 99.9% by predicting 56807 people as non fraud out of 56846 people.



C. Logistic Regression

Accuracy: It is calculated by correctly classified points divided by total no of points.Theese Model gives accuracy of 94.2% after using 10 fold cross validation with cross validation score 94.3%.

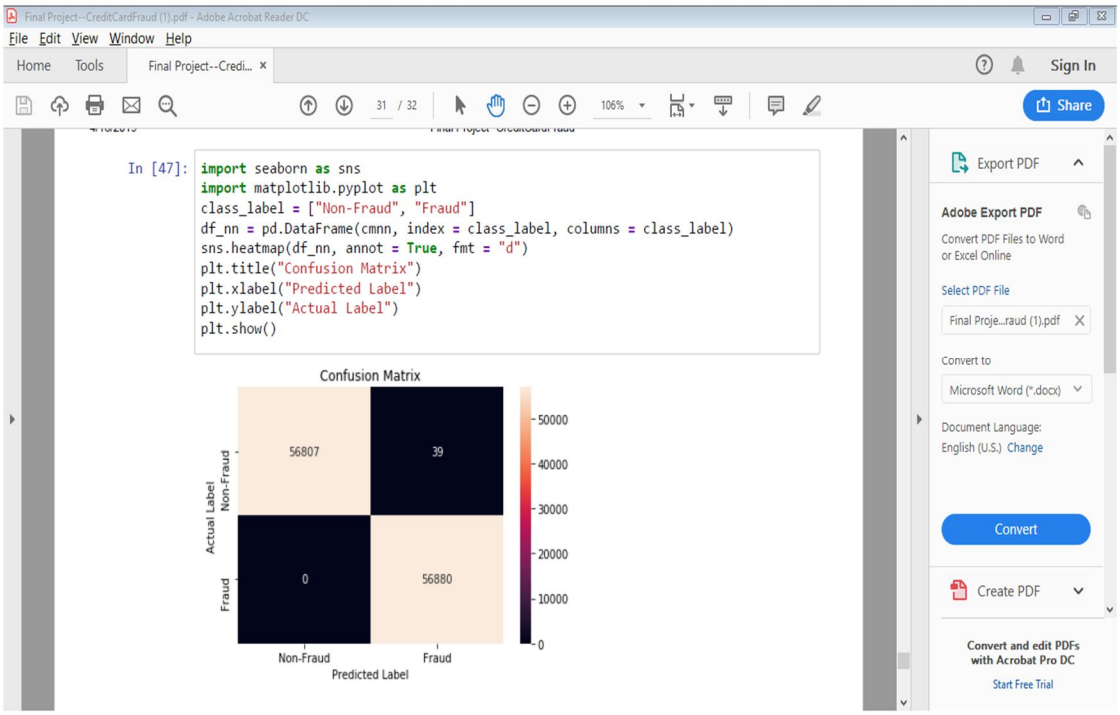
Sensitivity or Recall: It gives the recall of 91% by predicting 51697 people as "fraud" out of all the 56880 people. While 98% by predicting 55538 people as non fraud out of 56846 people.



D. Neural Networks

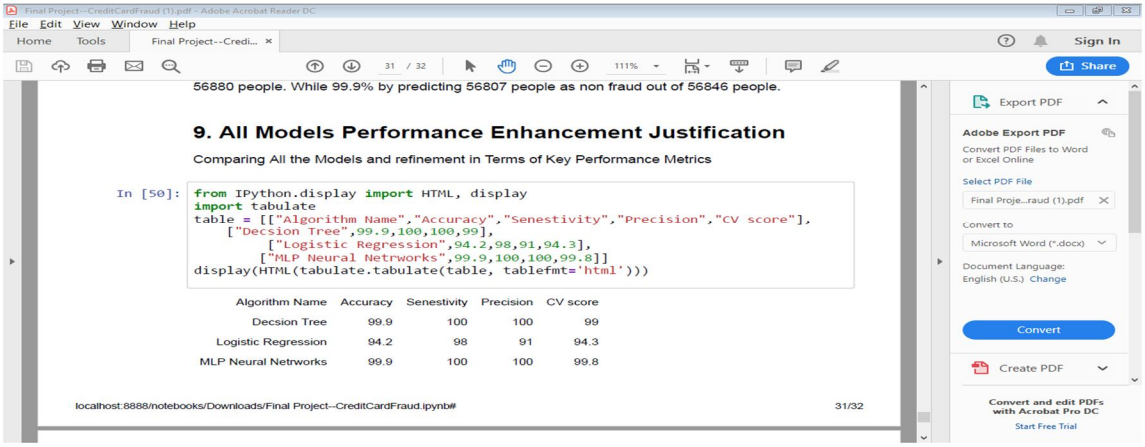
Accuracy: It is calculated by correctly classified points divided by total no of points.Theese Model gives accuracy of 99.9% after using 10 fold cross validation with cross validation score 99.8%

Sensitivity or Recall: It gives the recall of 100% by predicting 56880 people as "fraud" out of all the 56880 people. While 99.9% by predicting 56807 people as non fraud out of 56846 people



E. All Models Performance Enhancement Justification

Comparing All the Models and refinement in Terms of Key Performance Metrics

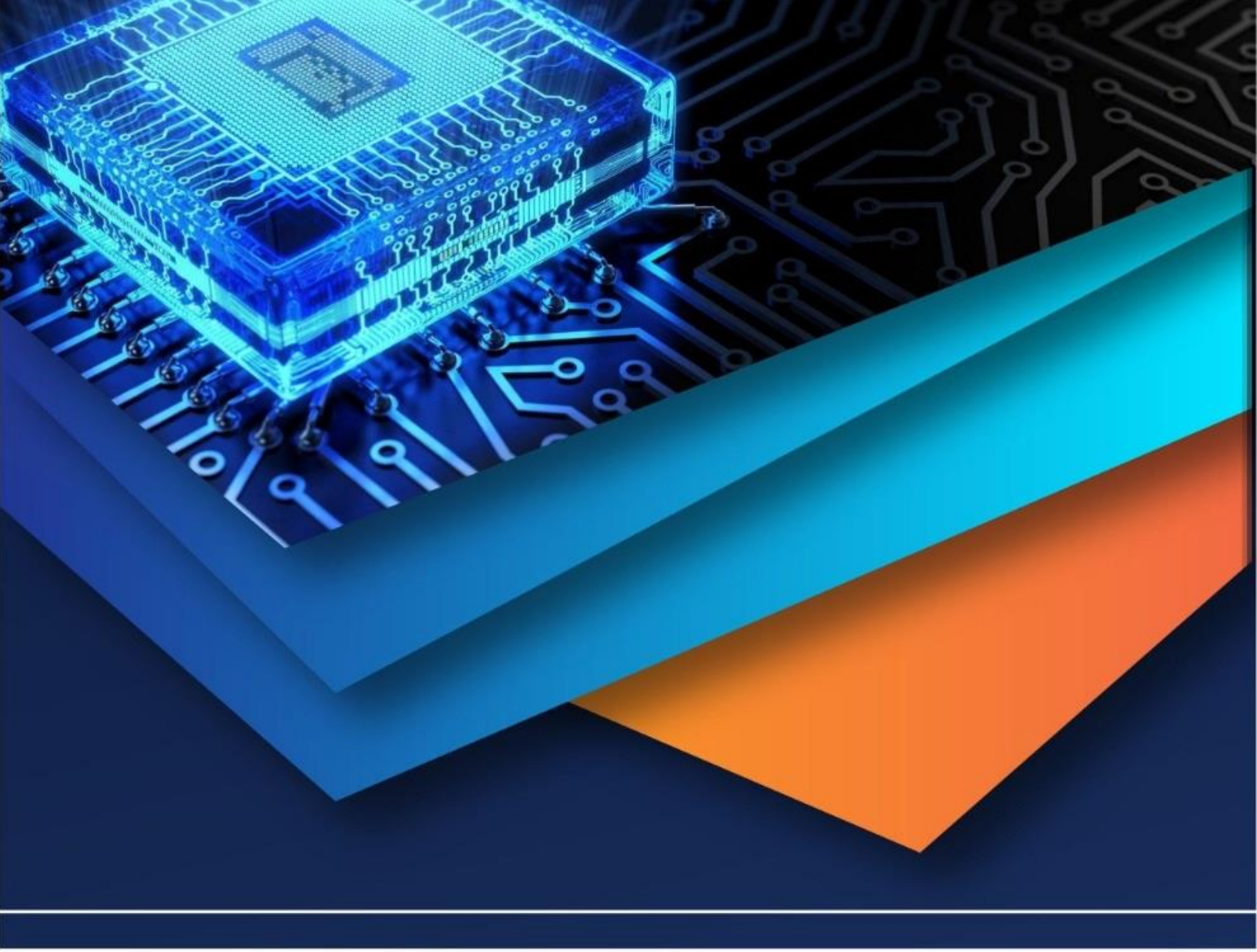


V. CONCLUSION

Therefore, we have performed up sampling for the data and fed this balanced dataset to the algorithms for the predictions. By observing the performance of the three models, we concluded that both the model's Decision Tree and Neural Network gives the highest accuracy and recall. Therefore these two models can be used to predict fraudulent transactions**.** One of the exciting parts of the project was dealing with the imbalance data. We tried two sampling methods Downsampling and Upsampling to balance the data. We found out that with the downsampling method, there was a loss of information, which resulted in inaccurate results.

REFERENCES

[1] KhyatiChaudhary, JyotiYadav, BhawnaMallick, " A review of Fraud Detection Techniques: Credit Card", International Journal of Computer Applications Volume 45– No.1 2012.
[2] Michael Edward Edge, Pedro R, Falcone Sampaio, "A survey of signature based methods for financial fraud detection", journal of computers and security, Vol. 28, pp 3 8 1 – 3 9 4, 2009.
[3] Linda Delamaire, Hussein Abdou, John Pointon, "Credit card fraud and detection techniques: a review", Banks and Bank Systems, Volume 4, Issue 2, 2009.
[4] Salvatore J. Stolfo, David W. Fan, Wenke Lee and Andreas L. Prodromidis; "Credit Card Fraud Detection Using Meta-Learning: Issues and Initial Results"; Department of Computer Science- Columbia University; 1997.
[5] Maes S. Tuyls K. Vanschoenwinkel B. and Manderick B.; "Credit Card Fraud Detection Using Bayesian and Neural Networks"; Vrije University Brussel – Belgium; 2002.
[6] Andreas L. Prodromidis and Salvatore J. Stolfo; "Agent-Based Distributed Learning Applied to Fraud Detection"; Department of Computer Science- Columbia University; 2000.
[7] Salvatore J. Stolfo, Wei Fan, Wenke Lee and Andreas L. Prodromidis; "Cost-based Modeling for Fraud and Intrusion Detection: Results from the JAM Project"; 0-7695-0490-6/99, 1999 IEEE.
[8] Soltani, N., Akbari, M.K., SargolzaeiJavan, M., "A new user-based model for credit card fraud detection based on artificial immune system," Artificial Intelligence and Signal Processing (AISP),2012 16th CSI International Symposium on., IEEE, pp. 029-033, 2012.
[9] S. Ghosh and D. L. Reilly, "Credit card fraud detection with a neural-network", Proceedings. https://www.kaggle.com/mlg-ulb/creditcardfraud/
[10] The dataset has been collected and analysed during a research collaboration of Worldline and the Machine Learning Group (http://mlg.ulb.ac.be) of ULB (Université Libre de Bruxelles) on big data mining and fraud detection. More details on current and past projects on related topics are available http://mlg.ulb.ac.be/BruFence and http://mlg.ulb.ac.be/ARTML.
[11] d. l. g. s. chandrahas mishra, "credit card fraud detection using neural networks," international journal of comoputer science, vol. 4, no. 7, July 2017.
[12] h. s.,. j. g. d.,. b. snehal patil, "credit card fraud detection using decision tree induction algorithm," international journal of computer science and mobile computing, vol. 4, no. 4, pp. 92-95.
[13] a. pansy khurana, "credit card fraud detection using fuzzy logic and neural network," SpringSim, 2016.
[14] a. nancy demla, "credit card fraud detection using svm and reduction of false alarms," inyternation journal of innovations in engineering and technology, vol. 7, no. 2, 2016.
[15] D. S. G. S.Saranya, "fraud detection in credit card transaction using bayesian network," international research journal of engineering and technology, vol. 4, no. 4, April 2017.
[16] T. R. C.Sudha, "credit card fraud detection in internet using k nearest neighbour algorithm," IPASJ international journal of computer science, vol. 5, no. 11, 2017.
[17] A. O. A. S. A. O. John o. Awoyemi, "credit cars fraud detection using machine learning techniques: A comparative analysis," in International conference on computing networking and infomatics.
[18] E. Aji M. Mubarek, "Multilayer perceptron neural network technique for fraud detection," in International Conference on Computer Science and Engineering(UBMK), 2017.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY