Credit Card Fraud Detection

CMPE -255 Group: 3

Mahesh Reddy Konatham
Reg. No: 013823095
Computer Engineering Department
San Jose State University
maheshreddy.konatham@sjsu.edu

Shivangi Nagpal
Reg. No: 013852696
Computer Engineering Department
San Jose State University
shivangi.nagpal@sjsu.edu

Rakesh Kumar Amireddy
Reg. No: 013858871
Computer Engineering Department
San Jose State University
rakeshkumar.amireddy@sjsu.edu

Mohdi Habibi
Reg. No: 010302851
Computer Engineering Department
San Jose State University
mohdi.habibi@sjsu.edu

Saumil Shah
Reg. No: 013761293
Computer Engineering Department
San Jose State University
saumil.j.shah@sjsu.edu

***Abstract*—Credit card fraud is perhaps considered to be one of the most intimidating and infamously increasing threats to the financial industry. Credit card fraud happens mainly through online transactions as it is hard to identify if the person doing the transaction is authorized or a fraudster who might have stolen the credit card information. Data mining plays an important role in the detection of fraud in online financial transactions. To identify the fraudulent transaction, the key is to check the usage patterns over the historical transactions and then compare it with the current transaction. Spotting the fraudulent transactions in a timely manner not only prevents revenue loss for merchants but also increases customer satisfaction. The dataset we are using for this project is provided by vesta corporation, which offers solutions for guaranteed ecommerce payments. Firstly techniques such as Dimensionality reduction will be used for feature selection as the part of data preprocessing. Finally Naive Bayes, Logistic Regression, Support Vector machines, K-means algorithm and Random Forest classification algorithms will be leveraged to classify fraudulent or non fraudulent transactions.**

Keywords—Data Mining, Credit Card, Machine Learning, Naive Bayes, Logistic regression, Random Forest

I. Introduction

Financial fraud is an ever growing problem with huge impact on finance industry, corporate organizations, and government. Fraud can be defined as criminal deception with intent of acquiring financial gain. With the advancement in technology and increased usage of the internet, As the credit card transactions become the most popular mode of payment for both online and offline transactions, the credit card fraud rate also increased to a great extent. Traditional ways of detecting fraud transactions using manual methods is time consuming and inefficient, thus the financial institutions have focused attention to use data mining techniques to analyze historical data and look for patterns to identify fraudulent transaction in a timely manner. Credit card fraud detection is the process of identifying those transactions that are classified into two classes of legitimate (genuine) and fraudulent transactions.

As part of data mining techniques, applied supervised learning classification algorithms Naive Bayes, Logistic Regression, Support Vector machines and Random Forest and also applied unsupervised learning classification K-means in an attempt to evaluate and compare different metrics accuracy, sensitivity, specificity to identify best approach to detect credit card fraud. However solving this problem involves several challenges mainly 1) the behaviour of legitimate and fraudulent transactions keeps changing 2) fraudsters change the strategies from time to time and 3) the data for credit card fraud detection is highly imbalanced 4) appropriate feature selection for the models using dimensionality reduction 5) most suitable metric to evaluate the performance of the data mining techniques.

Data Preprocessing techniques used before applying learning models are Memory Reduction to reduce memory usage due to the size of the data set, removing duplicates and replacing null values with zero as part

of data cleaning, transform all the categorical features into numerical using one hot encoding, standardize some of the features in order to have the right scale for ML models and applied PCA(Principal Component Analysis) as part of dimensionality reduction to reduce the number of features. The dataset used for credit card fraud detection is highly imbalanced with only 5% of transactions are being classified as fraudulent, Downsampling Technique is used to match number of fraudulent transactions with the genuine transactions to deal with class imbalance.

II. Literature Review

According to Maes et. al [1], credit card fraud detections can be classified into two categories; one being class of genuine transactions and other being class of Fraudulent transactions; which will also be the aim of this research project. Credit card fraud detection can be analysed by observing patterns in cardholder's spending behavior[2]. The work in [3] list a number of statistical and machine learning approaches that can be applied to fraud detection. A lot of research has been performed in this area. The cost effective credit card Fraud evaluation methods have been discussed in [4]. In [1], artificial neural networks(ANN) and Bayesian Belief Networks(BNN) have been applied. The Naive Bayes, K-nearest neighbor and Logistic regression techniques have been explored for imbalanced credit card fraud data in [2].

III. Methodology

The dataset for our experiment is provided by Kaggle. It is a real world data set and most of the features have masked meaning. Our approach to this problem is as follows: (1) Analyze the data: this includes exploratory data analysis (EDA) of several features provided in the dataset. (2) Merging of the components of the dataset: as described in the following section the dataset provided is split into sections so to proceed to the next step, the different sections are merged. (3) Data Preprocessing: this is the most important part as the success of the following parts depends on how carefully the preprocessing is done, this part includes several subcomponents such as memory reduction, data cleaning and dimension reduction. (4) Application of the classification algorithms, each type of algorithms has its own advantage and disadvantage, we have supervised and unsupervised learning techniques to classify the genuine and fraudulent transaction. The following supervised techniques have been used in the experiment:

1. Naive Bayes: Naive bayes is a probabilistic approach for classification. It is based on Bayes' theorem and makes a naive assumption that features are independent of each other. Since this method is highly scalable and is suitable when there is high dimensionality so this method was chosen to do the analysis.
2. Logistic Regression: This method gives the output in the form of probability value which can be mapped to zero or one for the classification problem. This uses the maximum likelihood estimation function to converge the problem.
3. Support Vector Machines(SVM): This is a non-probabilistic method. This method uses kernel trick to make non linear classifications. The algorithm uses a maximum margin hyperplane to separate the two classes.
4. Random Forest: This is an ensemble method for classification. It fits a number of decision tree classifiers and averages their output to predict a value. The internal trees are trained on different set of inputs and also use different features to predict an outcome. Thus different trees cover up for the errors of other trees.

The following unsupervised learning techniques is also used for comparison purposes:

1. K-means: This is an iterative algorithm, for this experiment a k value of two is chosen as only two clusters are desired. Kmeans follows the Expectation Maximization approach to solve the problem. The disadvantage of this method is that it can converge to a local optimum points instead of global optimum.

IV. Data collection and Preprocessing

A. Exploratory Data Analysis

Our dataset contains real life data and to keep it anonymous, all the values and features are masked. Due to this, it was very difficult to understand the data. Our biggest challenge was to understand the data, clean it and choose an appropriate algorithm for the classification. Data Analysis was a major step for the project, to make a clear understanding of the data. So, to visually interpret the data, we carried out exploratory data analysis, to simplify the large, complex dataset.

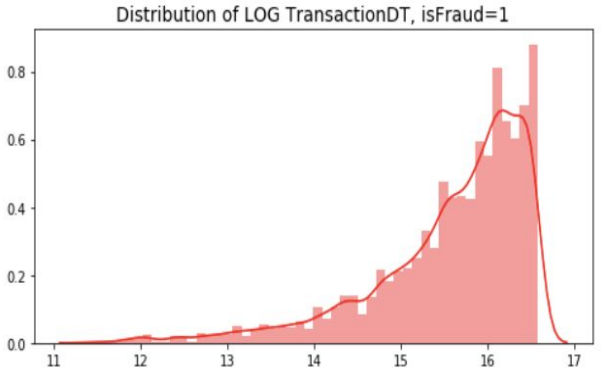

Fig 1. Distribution of LOG TransactionDT for fraud transactions

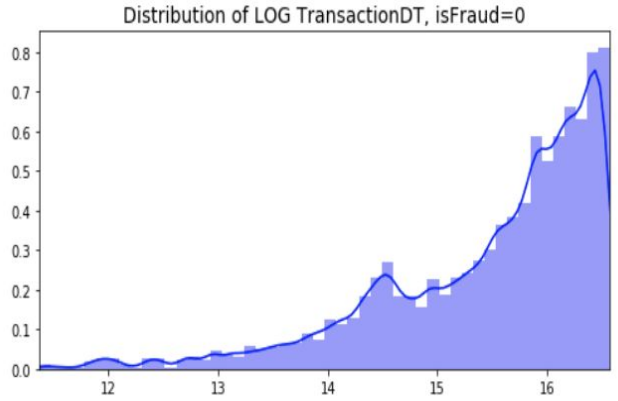

Fig 2. Distribution of LOG TransactionDT for genuine transactions

The above images show the LOG distribution of TransactionDT classified for fraud and genuine transactions. TransactionDT is not a timestamp, but can be used to measure time.

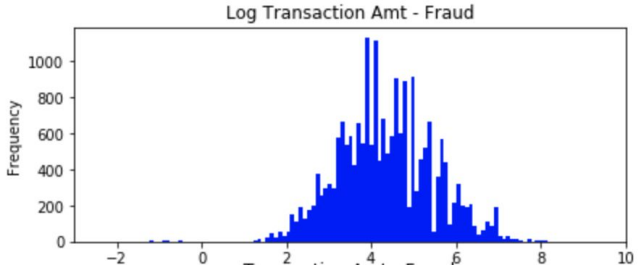

Fig 3. Distribution of LOG TransactionAmt for fraud transactions

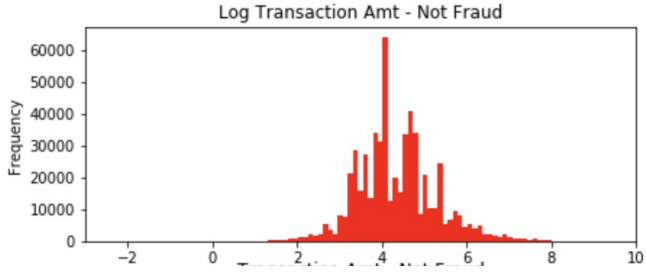

Fig 4. Distribution of LOG TransactionAmt for genuine transactions

The above visualizations represent the LOG distribution of TransactionAmt classified for fraud and genuine transactions. The transaction amount is a key factor as it will allow us to differentiate fraudulent and genuine transactions, since fraudulent transactions tend to have greater transaction amount. The reason to apply log transformation is to better depict the original, skewed data. This will help to make the patterns in the data more interpretable and to meet the assumptions of inferential statistics.

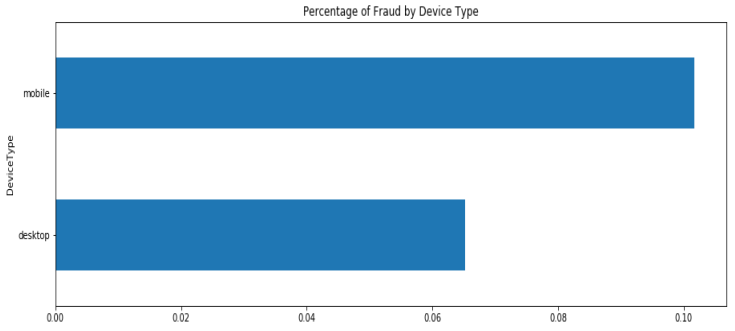

Fig 5. Percentage of fraud by device type

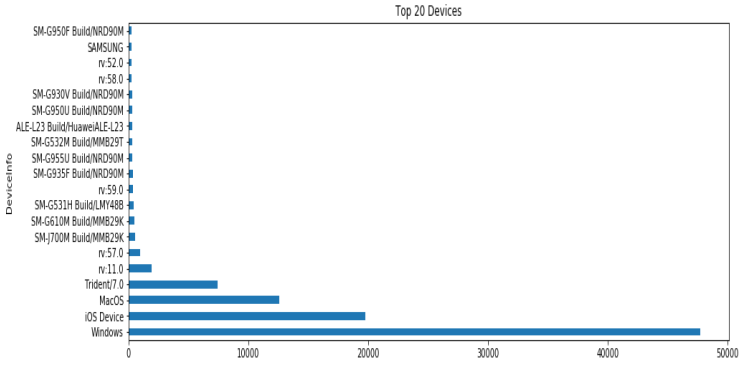

Fig 6. Information of the top 20 devices

The above visualizations classify the fraud depending on the type of device, i.e, whether it is a desktop or a mobile and gives the information about the Top-20 devices used for fraudulent transactions. This helps us to understand the devices majorly used for fraud activities. These are some of the exploratory data visualizations that helped us gain an idea of the data in our dataset, which further helped us for data preprocessing and feature selection.

B. Data Preprocessing

Data for this project has been taken from kaggle. The dataset is given in two parts, identity and transactions. The identity contains 41 columns and the transaction contains 393 columns. The tables identity and transaction are joined by key TransactionId. The target variable isFraud is a binary value and is present in the transaction file. Many of the features are masked for keeping the data anonymized.

The first step in our preprocessing was to reduce the amount of memory our dataset taking to make it easier for the systems to handle such data. Many of the operations running on the original dataset made google colab crashed and limit the use of our systems to work with such big data. With reducing the memory that is used by the dataset, it eased the pressure on the system and let us to run the entire process on our own systems. As the data set is huge we have to use some memory reduction functions. Using the memory reduction function we could reduce the memory usage from 1959.88 Mb to 650.48 Mb (66.8% reduction).

In the next step of preprocessing, our data needed to be massaged so that it can be used for training different models. First duplicate rows were removed from the dataset. Also the dataset contained many rows with NaN which was replaced with 0. For many of the models replacing NaN with 0 is not causing any issue but among others Naïve Bayes cannot handle 0 as input value. This is one of the reasons why we have poor accuracy for our Naïve Bayes model. When we tried to replace NaN with mean value of column, both google colab and Jupyter Notebook was not able to handle the computation power. In the initial dataset, there are 414 cells with value NaN.

The next step in preprocessing was to transform all the categorical features to numerical so that it makes it easier for machine learning models to be trained. There are few techniques to do this but label encoding and one hot encoding is one of the most popular ones. In order to transform a category to number, we used sklearn label encoder which simply iterate through all the categories and map them to a number. The only problem is if we feed this new data to some of the algorithms, there is a possibility that the algorithm favor one over the other. For example if we map dog:1 and cat:2, some ML algorithms favor dogs over cats or vice versa due to the order of them. But in reality the order of those categories have no effect on the outcome. This is why we applied one-hot encoding so that we overcome the aforementioned issue.

With our dataset, we first filtered all the features that are categorical and separated them from the rest of the data. In our dataset, 31 features were categorical or had categorical values in one of their instances. Using this newly created dataframe, we used sklearn label encoder to transform the data to numerical. We will need to run one hot encoding on our categorical data as well to prevent from the problem discussed earlier. Once data transformed, then we combined the features back to our original dataset.

Most of the data is already standardized except the amount and date. In order to have the right scale for our ML algorithms, we need to reshape the data to get them ready for next steps. To do this, we used StandardScaler in our project to reshape both transaction amount and transaction date.

Our dataset contained 433 features with most of them being masked. In order to reduce the complexity of our data and help reduce training time, we needed to apply PCA to reduce dimensions. This much complexity and dimension can slow down any models at the time of training and also introduces the problem of overfitting the model. This can result in poor accuracy and very slow training process. This is when we need to use dimensionality reduction to overcome the curse of dimensionality. PCA is the most widely used tool that can be helpful in this step to prioritize the features that have the most weight on the outcome.

In our intermediate report, we elaborated on further work needed for the final result. After applying PCA, we extracted the eigenvalues to understand the variance of each component and pick the right number of components. After further observation, we realized that the top 2 components represent 99.99% of our data. This analysis helped us to understand a key important reason behind getting low accuracy. We ran our experiment with several different number of PCAs to understand the impact on our results. The highest accuracy achieved with our dataset was when we removed PCA from preprocessing step. This was part of learning through this process that PCA does not help with every case. Eliminating some of our features affected the accuracy of our models in this case.

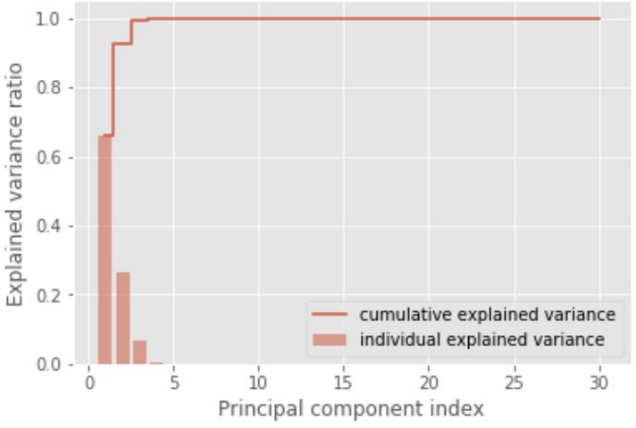

Fig 7. Principal component Index vs variance Ratio

Class imbalance result in a poor accuracy if it is not dealt with in the preprocessing step. The credit card transactions were consist of 3% fraudulent transactions. To better understand this problem, let's look at the distribution below:

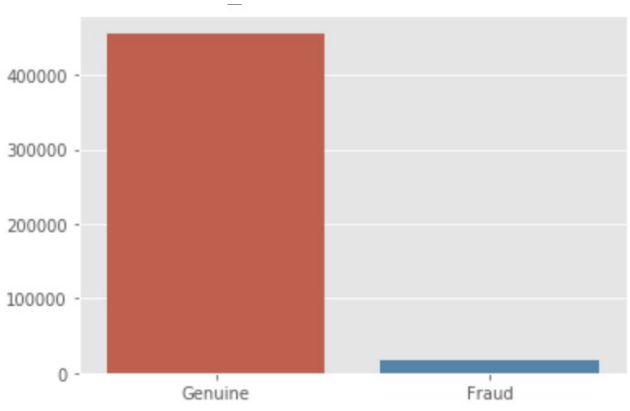

Fig 8. Original Dataset

Fig 9. Dataset after applying upsample

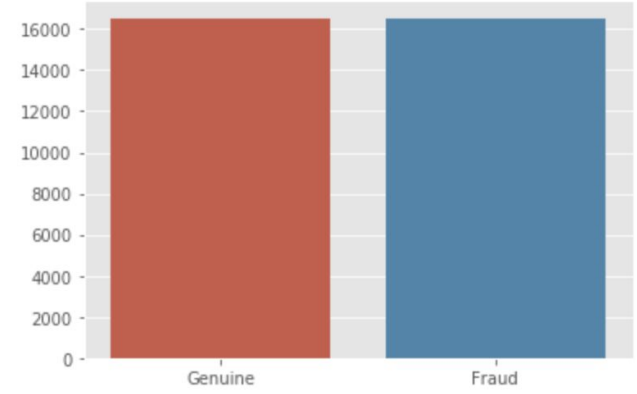

Fig 10. Dataset after applying downsample

In order to overcome this problem, we used downsampling and upsampling to create balanced data.

V. Evaluation and Results

In this project after experimenting the downsampled and preprocessed data with different algorithms like Naive Bayes, K-means, Random Forest, Logistic Regression and SVM we calculated different metrics of evaluation using the below formulas:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

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

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

$$\Pr ecision = \frac{TP}{TP + FP}$$

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

The Table 1 below shows that results for different metrics like Accuracy, Sensitivity, Precision and Mathews correlation coefficient(MCC). When the ratio of confusion matrix's four categories is imbalanced we get improper accuracy and f1-scores leading us to wrong evaluation of model. But MCC helps us in overcoming imbalanced ratios of confusion matrix categories and asymmetric nature of f1 score.

The formula to calculate the MCC is very easy as shown above, it calculates the correlation coefficient between true class and predicted class considering them as two different variables. The higher the MCC value the better is the classifier. MCC ranges from -1 to 1.

Although SVM has highest Accuracy score of 0.95 its MCC score is least when compared with other models. Random Forest performs better in all the evaluation metrics with Accuracy value of 0.85, Area under ROC curve as 0.83 and MCC value of 0.33.

TABLE 1. Evaluation Metrics for different algorithms

Metrics	Classifiers				
	NB	**LR**	**KM**	**RF**	**SVM**
Accuracy	0.0953	0.7932	0.5148	0.8548	0.9584
Sensitivity	0.9654	0.6442	0.5172	0.8120	0.0280
Precision	0.0360	0.1039	0.0372	0.1702	0.1157
Mathews correlation coefficient	0.0221	0.1974	0.0117	0.3294	0.0406

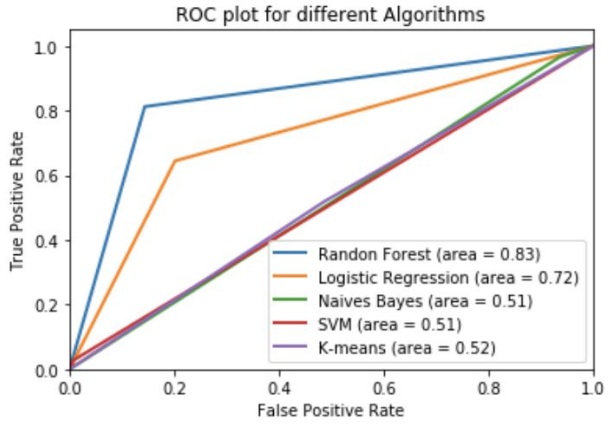

Figure 11. Roc plot for different algorithms

VI. Conclusion

The main aim of this research is to detect fraud accurately and before the fraud is committed. Credit Card Fraud Detection can be used for applications using credit card payments making it easier to avoid losses from fraud. This paper evaluates and compare the performance of Naive Bayes, Logistic Regression, Support Vector machines, Random Forest and K-means models in binary classification of imbalanced credit card fraud data.

These classifiers are examined using accuracy, sensitivity, specificity and precision metrics. Plotted ROC curve (False Positive Rate vs True Positive Rate) to compare the performance of classifiers. Results from the experiment shows that the Random Forest shows significant performance for all metrics evaluated whereas Naives Bayes shows least performance.

VII. References

1. Maes, S., Tuyls, K., Vanschoenwinkel, B. and Manderick, B., (2002). Credit card fraud detection using Bayesian and neural networks. Proceeding International NAISO Congress on Neuro Fuzzy Technologies.
2. John, O. A., Adebayo, O. A., Samuel, A. O. Credit card fraud detection using machine learning techniques: A comparative analysis, ICCNI 2017: International Conference on Computing, Networking and Informatics, Lagos, Nigeria.
3. Bolton, R. J., and Hand, D. J. (2002). Statistical fraud detection: a review. Statistical Science, 17(3), 235-249.
4. Bahnsen, A. C., Stojanovic, A., Aouada, D., & Ottersten, B.(2013). Cost sensitive credit card fraud detection using Bayes minimum risk. In Machine Learning and Applications (ICMLA),2013 12th International Conference on (Vol. 1, pp. 333-338).IEEE.
5. https://www.kaggle.com/c/ieee-fraud-detection/data
6. https://towardsdatascience.com/understanding-random-forest-58381e0602d2
7. https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a
8. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

VIII. Appendix

[1]https://github.com/MaheshReddy108/CMPE-255-Credit-card-Fraud-detection

[2] https://github.com/MaheshReddy108/CMPE-255-Credit-card-Fraud-detection/blob/master/CMPE_255_Group_project.ipynb

IX. Key Learnings

- Work with masked data: understanding and analysing data for which the real meaning is not known.
- Downsampling and Upsampling of data: Data

set is highly imbalanced with only 5% of transactions are being classified as fraudulent. As the data set is huge, learnt that downsampling performs better than upsampling.

- Importance of laplacian correction in naive bayes model if the data set has null values replaced by zero which will make the conditional probability zero as a result predicted probability will be zero.
- PCA analysis - when to use and when not.
- PCA automatically takes care of Overfitting problem. So depends on the data set we deal with, we can choose either PCA or Regularization to deal with Overfitting
- Importance of data preprocessing: One hot encoding improved the accuracy of SVM algorithm.
- Random Forest Algorithm: We learned an ensemble algorithm while working on the project. The algorithm was intuitive and provided good accuracy results for our experiment.
- We learned a new evaluation metric named Matthews correlation coefficient based on confusion matrix which outperforms Accuracy and F1-score when the ratio of confusion matrix categories are imbalanced or asymmetric.