Credit Card Fraud Detection

Rahul Torlapati, Pratik Mankar, Amankumar Jain and Vinayak Patil
rahult1@umbc.edu, pt33819@umbc.edu, hw93730@umbc.edu, vpatil2@umbc.edu
University of Maryland, Baltimore County

***Abstract* - Fraud is one of the major ethical issues in the credit card industry. The main aims are, firstly, to identify the different types of credit card fraud, and, secondly, to review alternative techniques that have been used in fraud detection. The sub-aim is to present, compare and analyze recently published findings in credit card fraud detection. This article defines common terms in credit card fraud and highlights key statistics and figures in this field. Depending on the type of fraud faced by banks or credit card companies, various measures can be adopted and implemented. The proposals made in this paper are likely to have beneficial attributes in terms of cost savings and time efficiency. The significance of the application of the techniques reviewed here is in the minimization of credit card fraud. Yet there are still ethical issues when genuine credit card customers are misclassified as fraudulent.**

INTRODUCTION

For some time, there has been a strong interest in the ethics of banking (Molyneaux, 2007; George, 1992), as well as the moral complexity of fraudulent behavior (Clarke, 1994). Fraud means obtaining services/goods and/or money by unethical means and is a growing problem all over the world nowadays. Fraud deals with cases involving criminal purposes that, mostly, are difficult to identify. Credit cards are one of the most famous targets of fraud but not the only one; fraud can occur with any type of credit products, such as personal loans, home loans, and retail. Furthermore, the face of fraud has changed dramatically during the last few decades as technologies have changed and developed. A critical task to help businesses, and financial institutions including banks is to take steps to prevent fraud and to deal with it efficiently and effectively, when it does happen (Anderson, 2007).

With this extensive use of credit card, fraud appears as a major issue in the credit card business. In the European Union, the first signs could have been seen in the United Kingdom in the 90s. In fact, total losses through credit card fraud in the United Kingdom have been growing rapidly (1997, 122 million; 1998, 135 million; 1999, 188 million; 2000, 293 million [Association for Payment Clearing Services London (APACS), no date]. Yet, in 2006, APACS reported 423 million losses, a decrease of nearly £80 million over the previous two years. The main reason for this improvement is the success of chip & PIN that has led to a decrease of face-to-face fraud. However, if mail-non-receipt fraud and lost and stolen card fraud are decreasing, counterfeit card fraud and card-not-present (CNP) fraud are increasing although they are increasing at reducing rates (APACS, no date).

Hence, we aim to tackle this problem using Data Exploration and Statistical Learning. Our dataset was collected and curated by Machine Learning Group at Université Libre de Bruxelles (ULB). The dataset presents transactions that occurred in two days where we have 492 frauds out of 284807 number of transactions. The dataset is highly unbalanced. Because of the confidentiality issues, the original data is PCA transformed into 28 principal features. So before doing any data analysis on this data, we first need to resolve the issue data unbalance. Then carry out scaling and other data exploration techniques.

TOOLS USED

We used Python3 as the base coding language and Jupyter Notebook as the editor and environment. In Python3 we used libraries like Metrics, Pandas, Sklearn, NumPy, Seaborn, Matplotlib etc.

EXPLORATORY DATA ANALYSIS

The dataset is already anonymized with the help of Principal Component Analysis (PCA) expect for 2 features Transaction Amount and Transaction Time. The dataset consists of 284807 transaction and does not have any missing values. For the column representing amount the mean values of the feature are $88.55 and maximum is $25691.16. The distribution of the monetary value of all transactions is heavily right-skewed. The vast majority of transactions are relatively small and only a tiny fraction of transactions comes even close to the maximum.

From the dataset description we found out that all the transaction in our dataset have occurred within a two-day time period in the European Union. The distribution of transaction time is bimodal instead on being positively or negatively skewed. This indicates that approximately 28 hours after the first transaction there was a significant drop in the volume of transactions. While the time of the first transaction is not provided, it would be reasonable to assume that the drop in volume occurred during the night.

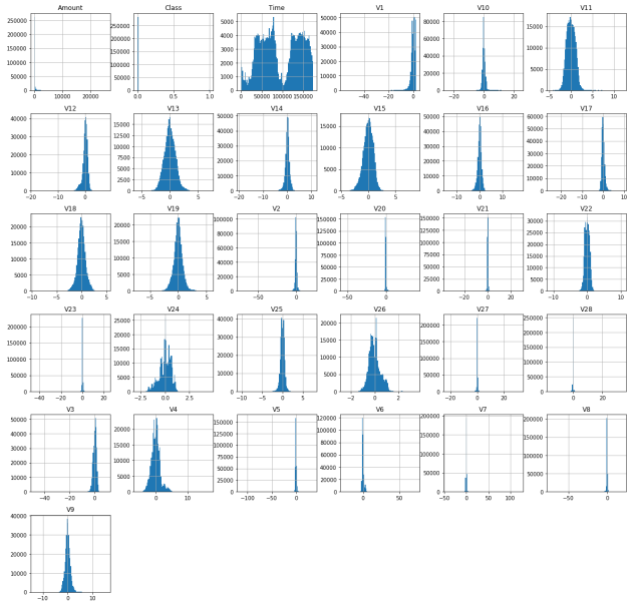

FIGURE I
DISTRIBUTION ALL THE ATTRIBUTES

I. Random Sampling

Since our dataset was highly unbalanced, we had a hard time preparing a training set that will allow our algorithms to pick up the specific characteristics that make a transaction more or less likely to be fraudulent. Since over 99% of our transactions are non-fraudulent, an algorithm that always predicts that the transaction is non-fraudulent would achieve an accuracy higher than 99%. Nevertheless, that is the opposite of what we want. We do not want a 99% accuracy that is achieved by never labeling a transaction as fraudulent, we want to detect fraudulent transactions and label them as such. To create a balanced training data set, we took all of the fraudulent transactions in our data set and counted them. Then, we randomly selected the 9 times more of non-fraudulent transactions and concatenated the two. After shuffling this newly created data set is in 90:10 ratio. Finally has 5412 data points.

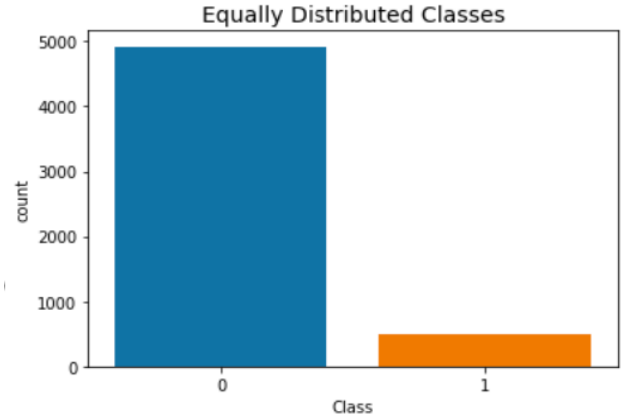

FIGURE II
DISTRIBUTION OF FRAUDLENT AND NON – FRAUDLENT TRANSACTIONS

II. Scaling

Since the only attribute with any background information had large disparity in them, we had to scale transaction amount and transaction time to a smaller distribution.

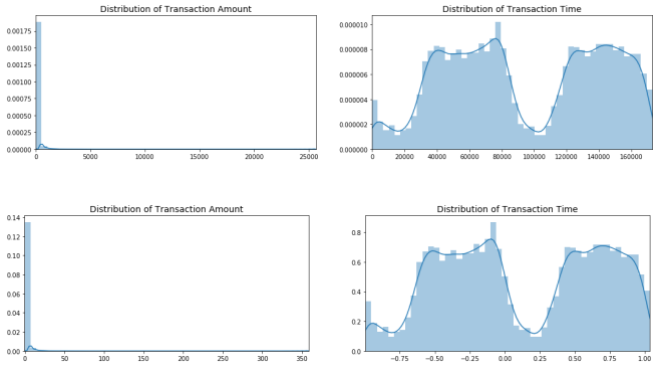

FIGURE III
SCALED TRANSACTION TIME AND TRANSACTION AMOUNT

III. Correlation Heatmap

We got some interesting insights from the correlation heat map. Attributes V2, V4, V11, and V19 are positively correlated and attributes V10, V12, V14 and V17 are negatively correlated.

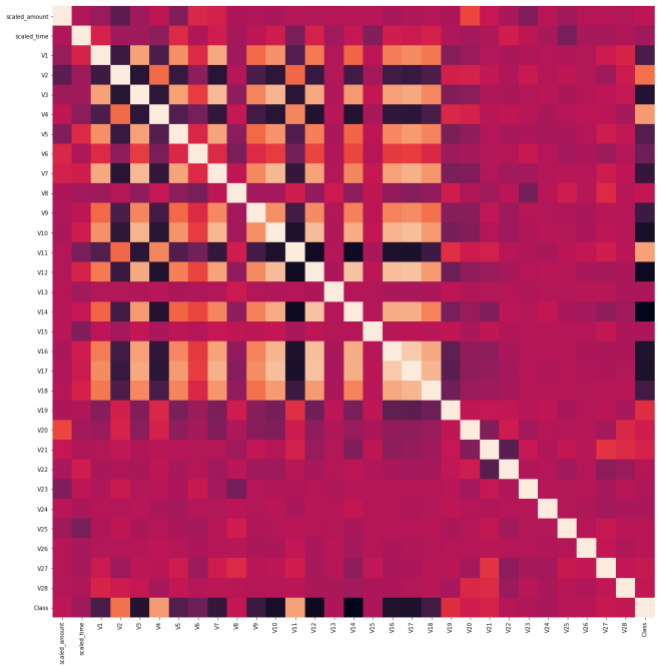

FIGURE IV
CORRLEATION HEAT MATRIX OF THE DATASET

With the insights from the correlation heatmap, we've decided to lessen impact from negatively correlated attributes.

In order to do that we are using quartile outlier removal technique.

IV. Inter - Quartile Outlier Removal

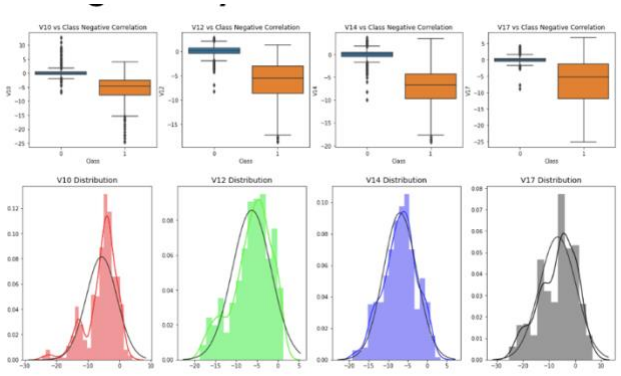

FIGURE V
BOXPLOT AND DISTRIBUTION GRAPHS OF NEGATIVELY CORRELATED ATTRIBUTES

From the heatmap we found out that Attributes V2, V4, V11, and V19 are positively correlated and attributes V10, V12, V14 and V17 are negatively correlated. In order to not the let the negatively correlated attributes to negatively impact our algorithms we need to lessen their impact. We can do this by removing the outliers in those attributes and making sure most of the data points lie under the distribution curve.

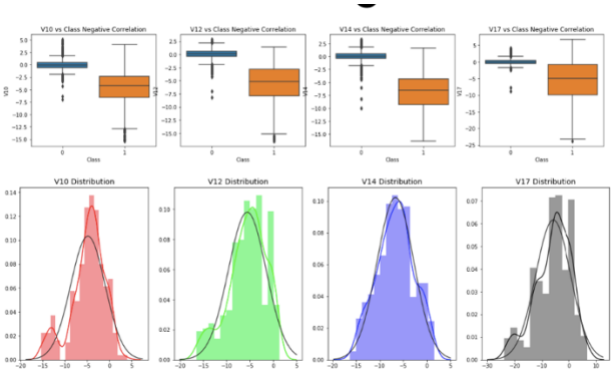

FIGURE VI
BOXPLOT AND DISTRIBUTION GRAPHS OF NEGATIVELY CORRELATED ATTRIBUTES AFTER REMOVING OUTLIERS

APPLICATION OF MACHINE LEARNING ALGORITHMS

I. DBSCAN()

Density based spatial clustering of applications with noise (DBSCAN) is a data clustering algorithm that is commonly used in machine learning. Based on the set of points, DBSCAN groups together points that are closer to each other based on minimum number of points and distance measurement. The DBSCAN algorithm uses two parameters eps and minPoints. Eps shows how close the points should be to each other to be a part of cluster. It is done by calculating mean between two points. MinPoints is the minimum number of points to form a dense region. The following algorithm is used to find associations and structures in our dataset, that are difficult to find manually, but that can be useful to find the patterns.

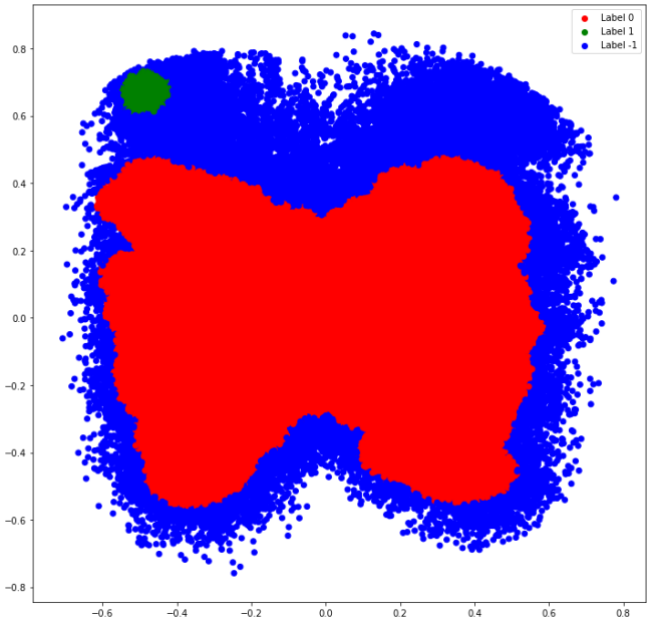

FIGURE VII
CLUSTERING WITH THE HELP OF DBSCAN()

In order to apply the DBSCAN() algorithm on the data we had to first scale the data with the help of standardscaler() and then normalize it. Then the normalized data is converted into 2 components with the help of Principal Component Analysis (PCA). When the DBSCAN() algorithm is applied on this data it detects most of non-fraudulent transactions either as positive label or as noise. Fraudulent transactions are clustered in green.

II. Logistic Regression

With change in parameter 'C': [0.01,0.1,1,10,100,1000]. We ran over 60 models with different C parameters, fitting 10 folds for each of the 6 candidates. The best Logistic Regression Model with {'C': 0.1} has an accuracy of 99.0%.

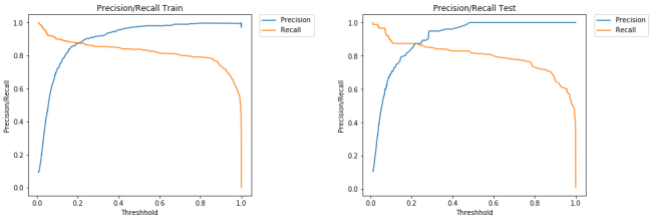

FIGURE VIII
PRECISION AND RECALL CURVES FOR TRAIN AND TEST DATASETS.

III. K-Nearest Neighbor Classifier

With change in parameters 'n_neighbors': [1,10,50,100,500], 'algorithm': ['auto', 'ball_tree', 'kd_tree', 'brute']. We ran over 200 models with different C parameters, fitting 10 folds for each of the 20 candidates. The best K-nearest neighbor with {'algorithm': 'auto', 'n_neighbors': 3} classifier has an accuracy 98%.

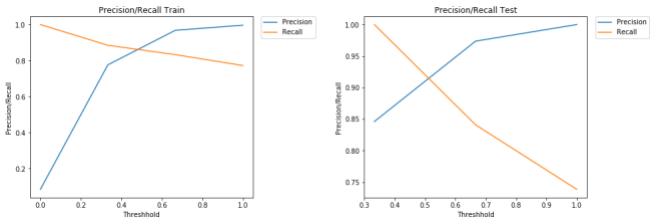

FIGURE IX
PRECISION AND RECALL CURVES FOR TRAIN AND TEST DATASETS.

IV. Support Vector Machine Classifier

With change in parameters 'C': [0.1,0.8,0.9,1,1.1,1.2,1.3,1.4], 'kernel': ['linear', 'rbf'], 'gamma':[0.1,0.8,0.9,1,1.1,1.2,1.3,1.4]
We ran over 1280 models with different C parameters, fitting 10 folds for each of the 128 candidates. The best Support Vector Machine classifier with {'C': 0.7, 'gamma': 'auto_deprecated, 'kernel': 'poly'} classifier has an accuracy 98%

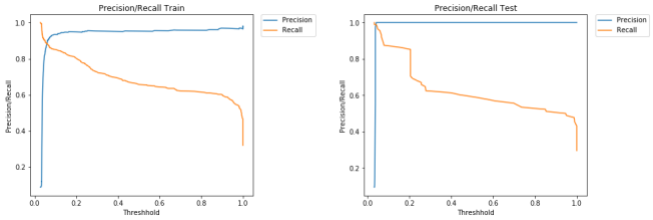

FIGURE X
PRECISION AND RECALL CURVES FOR TRAIN AND TEST DATASETS.

V. Decision Tree Classifier

With change in parameters 'criterion': ['gini', 'entropy'], 'min_samples_split': [2,3,4,5,6,7,8,9,10], 'min_samples_leaf': [1,2,3,4,5,6,7,8,9,10]. We ran over 1800 models with different C parameters, fitting 10 folds for each of the 180 candidates. The best Decision Tree Classifier with {'criterion': 'gini', 'min_samples_leaf': 5, 'min_samples_split': 2} has an accuracy of 98%.

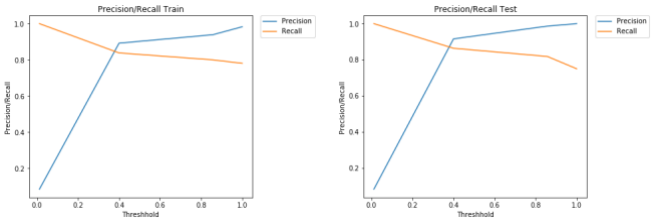

FIGURE XI
PRECISION AND RECALL CURVES FOR TRAIN AND TEST DATASETS

VI. Random Forest Classifier

With change in parameters ‘n_estimators': [1,10,50,100,500], 'criterion': ['gini','entropy'], 'min_samples_split’: [2,3,4,5,6,7,8,9,10], 'min_samples_leaf’: [1,2,3,4,5,6,7,8,9,10]. We ran over 9000 models with different C parameters, fitting 10 folds for each of the 900 candidates. The best Random Forest classifier with {'criterion': 'gini', 'min_samples_leaf': 8, 'min_samples_split': 6, 'n_estimators': 100} has an accuracy 99%.

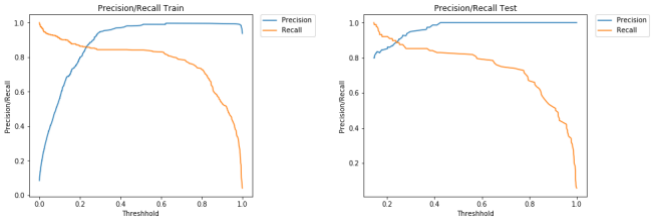

FIGURE XII
PRECISION AND RECALL CURVES FOR TRAIN AND TEST DATASETS.

PRECISION RECALL CURVE

There are many ways to evaluate the skill of a prediction model. An approach in the related field of information retrieval (finding documents based on queries) measures precision and recall. These measures are also useful in applied machine learning for evaluating binary classification models. Precision is a ratio of the number of true positives divided by the sum of the true positives and false positives. It describes how good a model is at predicting the positive class. Precision is referred to as the positive predictive value. The precision and recall can be calculated for thresholds using the precision_recall_curve() function that takes the true output values and the probabilities for the positive class as output and returns the precision, recall and threshold values. When comparing among models the model with the highest curve is considered to be a better performing model than that of a model with a lesser curve.

CONCLUSION

Looking at the Precision Recall Curve of the performing algorithms on test data, we can clearly say that Support Vector Machine Classifier and Random Forest Classifier are the best possible algorithms to predict fraudulent and non-fraudulent transactions. And when Support Vector Machine Classifier and Random Forest Classifier are considered, it is Random Forest Classifier that has the better performance among both. Although Logistic Regression Model did have a 99% accuracy, the algorithm is primitive and lacks the flexibility to handle complex multi- dimensional data.

REFERENCES

[1] Molyneaux, D. 2007. 'Two case study scenarios in banking: a commentary on *The Hutton Prize for Professional Ethics*, 2004 and 2005'. *Business Ethics: A European Review*, 16:4, 372-386.

[2] Clarke, M. 1994. 'Fraud and the Politics of Morality'. *Business Ethics: A European Review*, 3: 2, 117-122.

[3] George, E.1992.'Ethics in Banking' *Business Ethics: A European Review*, 1:3, 162-171.

[4] Anderson, R. 2007. The Credit Scoring Toolkit: theory and practice for retail credit risk management and decision automation. New York: Oxford University Press.

[5] Aleskerov, E., Freisleben, B. & B Rao. 1997. 'CARDWATCH: A Neural Network-Based Database Mining System for Credit Card Fraud Detection', Proc. of the IEEE/IAFE on *Computational Intelligence for Financial Engineering*, 220-226

[6] Bolton, R. & Hand, D. 2002. 'Statistical Fraud Detection: A Review'. *Statistical Science*, 17; 235-249.

[7] Bolton, R. & Hand, D. 2001. Unsupervised Profiling Methods for Fraud Detection, Credit Scoring and Credit.

[8] Chan, P., Fan, W. Prodromidis, A. & S Stolfo. 1999. 'Distributed Data Mining in Credit Card Fraud Detection'. *IEEE Intelligent Systems*, 14; 67-74.

[9] Encyclopedia Britannica, no date. Credit Card. Available at: http://www.britannica.com/eb/article/9026818/credit- card (Accessed: October 2008).

[10] Fawcett, T. & Provost, F. 1997. 'Adaptive Fraud Detection' *DataMiningandKnowledgeDiscovery*,1;3.

[11] Wheeler, R. & Aitken, S. (2000). 'Multiple Algorithms for Fraud Detection'. *Knowledge-BasedSystems*,13;93-99.

[12] Zaslavsky V. & Strizhak A. 2006. 'Credit card fraud detection using self-organizing maps' *Information and Security*, 18; 48-63.