Credit Card Fraud Detection with NCA Dimensionality Reduction

Beyazıt Bestami Yüksel
Faculty of Computer and Informatics,
Istanbul Technical University
İstanbul, Turkey
yukselbe18@itu.edu.tr

Şerif Bahtiyar
Faculty of Computer and Informatics,
Istanbul Technical University
İstanbul, Turkey
bahtiyars@itu.edu.tr

Ayşe Yılmazer
Faculty of Computer and Informatics,
Istanbul Technical University
İstanbul, Turkey
yilmazerayse@itu.edu.tr

ABSTRACT

Credit card transactions for online payments have increased dramatically and fraud attempts on these payments have become prevalent with more advanced attacks. Thus, conventional fraud detection mechanisms are inadequate to provide acceptable accuracy for fraud detections. Machine learning algorithms may provide a proactive mechanism to prevent credit card fraud with acceptable accuracy. In paper, we propose a new approach with machine learning for credit card fraud detections by increasing the performance of classification algorithms. We use the Neighborhood Component Analysis (NCA) dimensionality reduction to improve success rate for credit card fraud detections that use K-Nearest Neighbors (KNN) classification algorithm. We implemented the proposed approach and we tested it on a dataset. Particularly, we evaluated the results with the Area under the Receiver Operating Characteristic (AUROC) metric. The analyses results show that our approach provides better accuracy for credit card fraud detections.

CCS CONCEPTS

• **Security and Privacy** → Cyber Fraud; • **Computing Methodologies** → Machine learning; Machine learning approaches.

KEYWORDS

KNN, NCA, Credit Card Fraud

ACM Reference Format:
Beyazıt Bestami Yüksel, Şerif Bahtiyar, and Ayşe Yılmazer. 2020. Credit Card Fraud Detection with NCA Dimensionality Reduction. In *13th International Conference on Security of Information and Networks (SIN 2020), November 04–07, 2020, Merkez, Turkey.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3433174.3433178

1 INTRODUCTION

With the rise of the digital economy, the behavior of consumers has fundamentally changed. This change results in a huge amount of data that provide a very suitable ground for frauds. Consequences of fraud have significant impact on society and economy. There have been various researches and studies that are carried out to prevent credit card frauds.

Machine learning algorithms use big data to automatically detect patterns that may be used to predict frauds with better accuracy which can be used to prevent economic losses. Generally, a machine-learning model divides a dataset to a training set and a test set that determines the performance of the model. On the other hand, credit card data differ from other data therefore we need to tune the machine learning models to have better performance results.

An effective credit card fraud detection system should provide high accuracy to deal with current fraud challenges. General challenges regarding credit card fraud detections are as follows:

- Available datasets for credit card fraud detection systems are imbalanced. In other words, a very small percentage of transactions in the dataset are classified as fraudulent.
- It is hard to obtain real credit card fraud data.
- High percentage of academic works use the same datasets, which do not reflect real world [21].

In this paper, we have proposed a novel model with Neighborhood Component Analysis (NCA) dimensionality reduction. We have used outlier detection and KNN classification algorithms with optimum parameters chosen by grid search method. The proposed model has better performance results in terms of credit card fraud detections. Specifically, performance results show that machine learning algorithms that use NCA has % 7 improvement and Principal Component Analysis (PCA) has 4 % improvement than basic KNN method.

The rest of this paper organized as follows: Section 2 provides information about credit card fraud and an overview of credit card detection. We present our model in Section 3. In Section 4 is devoted for performance analysis. We conclude our paper with Section 5.

2 FRAUD DETECTION AND MACHINE LEARNING

Credit card fraud activities continue to be the subject of many studies as it causes huge financial losses and remains current. According to the literature, it is possible to group the studies on this subject in terms of the methods used in general as follows:

- Different Machine learning methods used in [1]. The results obtained from Logistic Regression, Decision Tree and Random Forest methods were examined comparatively. While effective results can be obtained on small data clusters with Random Forest Method, performance decreases in unbalanced data.

SIN 2020, November 04–07, 2020, Merkez, Turkey

ACM ISBN 978-1-4503-8751-4/20/11…$15.00
https://doi.org/10.1145/3433174.3433178

- SVM methods used in [3], [4], [5] and imbalanced data problem solved by oversampling method. These model performs much better at recognizing fraud transactions but it also misclassify the genuine as fraud ones. ML algorithms implemented directly in Spark [6] and took more than 91% accuracy. In addition to the studies that derive new features from the existing features with Feature Engineering [7], there are also studies that detect fraud if they create behavior patterns. [8].
- Adaboost and majority voting approaches are used [12] and novel fraud monitoring systems consisting of online risk fraud scoring system and offline fraud prediction systems [13] can be described as new approaches in this field. In these studies, the issues of how to deal with the imbalanced dataset were not addressed.

In this paper, a method that compensates the studies of [14], [15] using the KNN method and SMOTE technique presented as a solution to the imbalanced data problem [16], [17] which can achieve higher accuracy. KNN algorithms are suitable for fraud detection, but have memory size restrictions. In our recommended method, we managed to minimize the memory restriction problem by using the NCA and PCA method together.

Most data mining algorithms ignore outliers to avoid the complexity of the model. However, these values cannot be ignored and should be investigated in credit card frauds. Instead of completely removing the outliers, we increased the overall success of the system by excluding those above a certain threshold level from the classification process. In our study, we used the Local Outlier Factor (LOF) method while handle the outlier issue, and in this way, we determined the points that do not appear as outlier when consider the entire dataset, but can be accepted as outlier in the regions with local densities.

Most of the machine learning algorithms applied in such studies yield over 90% accuracy results. Signature alarms (True Positive, True Negative, False Positive, and False Negative) have widely been used for comparison purposes. In fraud detection, we believe that recall is more important than precision because of the importance of FN. Based on this idea, we used the Area under Receive Operating Characteristic Curve metric (with the abbreviation AUC) to evaluate how well our model is distinguishing different classes. When the AUC is large, we can determine how accurate our model can distinguish two classes from each other with the correct threshold value.

3 A MODEL FOR DIMENSIONALITY REDUCTION

This section contains information about the proposed model and corresponding methods.

3.1 System Overview

The general steps applied in this study to create a model to detect credit card fraud are shown in Figure 1

The steps of credit card fraud detection are described as follows:

- Obtain the dataset and add the required libraries related to the project. Class distributions, feature properties, dataset properties according to the distribution of data are some of the properties determined at this stage.
- Exploratory data analysis carried out and thus, information about dataset obtained. With this information, an idea obtained about data types, content of the data, whether there are missing values, outliers, and the correlation between the features.
- After obtaining knowledge about the data, outlier detection performed.
- Dataset splitted into two different parts, one for the independent features x, and one for the dependent variable y (which is the last column).
- Before model training with machine learning algorithm, standardization process implemented on dataset.
- Grid search method used to find the optimum parameters of KNN algorithm. This method is also used to find the optimum parameters for the database reduced with PCA and NCA algorithms.
- The results obtained by the KNN method with the application of the optimum parameters were compared with the results obtained by the application of the basic method.
- With PCA, which is the dimension reduction process, the database reduced into two dimensions. KNN algorithm was implemented again on the reduced data obtained by PCA.
- After implemented PCA, Neighborhood Component Analysis implemented in parallel to make comparisons with PCA.
- At the last step, the fraud detection results obtained at each step compared and the suitability of the implemented methods determined.

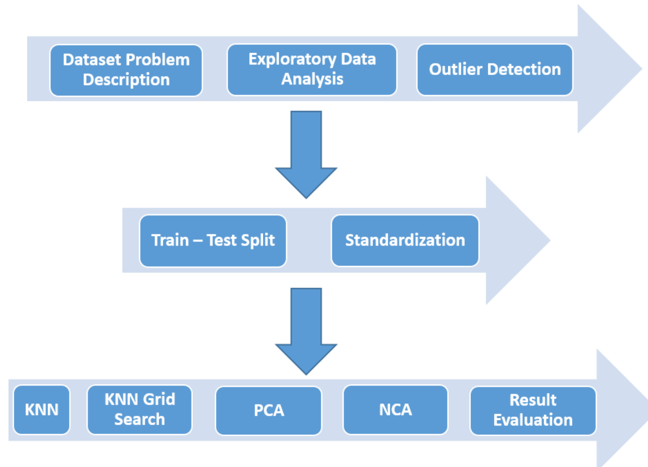

Figure 1: Fraud Detection with NCA

The AUC value and classification performance obtained at each step are shown visually to compare effectiveness of our proposed method. A general method has been developed about which optimization steps can be used to improve the model performance which obtained using a standard classification algorithm.

3.2 Local Outlier Factor

Since our data is skewed data, outliers detected and removed from the dataset. Local Outlier Factor (LOF) method used for this process. LOF is an unsupervised Outlier Detection method. In order to calculate whether point x is outlier or inlier, the LOF value of that point is checked. If $\mathrm{LOF_X}$ is greater than compare value (generally selected value 1) that point accepted as outlier. Equation 1 used to calculate the $\mathrm{LOF_X}$ value.

$$LOF_x = \frac{LRD_y + LRD_z}{LRD_x} * \frac{1}{k} \tag{1}$$

The $\mathrm{LRD_Y}$ and $\mathrm{LRD_Z}$ values in the equation are the Local Reachability Density values of y and z, which are the two points closest to the x point. The value of k is the number of neighbors chosen. To calculate the $\mathrm{LRD_x}$ value use Equation 2;

$$LRD_x = \frac{1}{ARD_x} \tag{2}$$

In order to calculate Average Reachability Density $\mathrm{ARD_X}$ we used the Equation 3;

$$ARD_x = \frac{RD_x}{2} \tag{3}$$

$\mathrm{RD_X}$ value is calculated as Equation 4:

$$\begin{aligned} RD_x &= \sum \max(k.\ distance\ of\ y\ ,\ dist\,(x,y)) \\ &+\ \max(k.\ distance\ of\ z\ ,\ dist\,(x,z)) \end{aligned} \tag{4}$$

Euclidean distance metric is used to calculate distance values. Numerical values in dataset are in a different scales. So we need standardization process. Standardization is the process of rescaling data. After standardization, the data will become in a structure with a mean of zero standard deviation of one.

3.3 K-Nearest Neighbor Method

The KNN (K-Nearest Neighbor) algorithm is one of the simplest and most widely used classification algorithms. KNN is a non-parametric, lazy learning algorithm. Unlike eager learning, lazy learning does not have a training phase. It does not learn the training data; instead, it memorizes the training dataset. When we want to make a prediction, it looks for the nearest neighbors in the whole dataset.

KNN Steps:

1. Select the K value
2. Find the nearest data points in K
3. Calculate how many of the class from the nearest neighbor K
4. Find out which class the tested data belongs to.

The main reasons for choosing the KNN algorithm that we can list the training process is fast easy to implement, easy to tune because it has only k and distance parameters. Beside these, it is sensitive against outliers, not very suitable for big data, and if there are too many features in the dataset it can be troublesome. To classify a test point, the entire dataset is stored and searched. The time elapsed for the test time for n dimensional data without any optimization is O (n). This situation is considered negative in terms of time and space complexity. The choice of the distance metric can have a significant effect on its performance.

The 30 features in the dataset are not many for KNN. In this study, dimension reduction is performed using PCA and NCA algorithms and it is observed that the performance of KNN increased. It is affected by the different scaling of features. In addition, it is very affected by imbalance data. To prevent this, SMOTE technique used. The most appropriate KNN parameters found by using the grid search method.

3.4 Principle Component Analyze (PCA)

PCA method reduces the size of the data by keeping as much information as possible. If there is time and power limitation, features can be reduced by PCA method. Another reason for using PCA is that if we have a correlation matrix and some of the features correlated, PCA plays a role in eliminating these features. The size reduction of the PCA is based on converting the existing, correlated variables in the dataset into the same number but not correlated (orthogonal) variables with some linear transformations. These new variables are a linear combination of existing ones and are referred to as Principal Components. This technique works well when there is excessive correlation between the variables in the datasets (as in econometric models) and the data may contain high errors.

3.5 Neighborhood Component Analyze (NCA)

Neighborhood Components Analysis aims to "learn" a distance metric by finding a linear transformation of input data so that the average leave-one-out (LOO) classification performance in the transformed area is maximized. LOO classification algorithm, on the other hand, is a method in which the closest neighbor of the k tries to predict a single point together using a certain distance measure. Unlike PCA, NCA is not an unsupervised learning algorithm. Therefore, it needs class information when performing fit operations. NCA is proposed to improve the classification performance of KNN. With the size reduction method applied with NCA, it is possible to visualize the data and fast classification. [21]

3.6 Area under Receiving Operating Characteristic Curve

Receiving Operating Characteristic (ROC) curve is a commonly used way to visualize the performance of a binary classifier meaning a classifier with two possible output classes. ROC curve is a plot of the True Positive Rate (on the y-axis) versus the False Positive Rate (on the x-axis) for every possible classification threshold shown in figure 2. Dashed red line essentially represents a classifier that does no better than random guessing. AUC stands for "Area under the ROC Curve". The scope of this area under the ROC curve is AUC. The larger the area covered, the better the machine learning models are in distinguishing the given classes. The ideal value for AUC is 1.0 that indicates all data is correctly classified. AUC metric cares about how well your classifier separated the two classes, and thus it is said to only be sensitive to rank ordering. It is acceptable that AUC as representing the probability that a classifier will rank a randomly chosen positive observation higher than a randomly chosen negative observation, and thus it is a useful metric even for datasets with highly unbalanced classes.

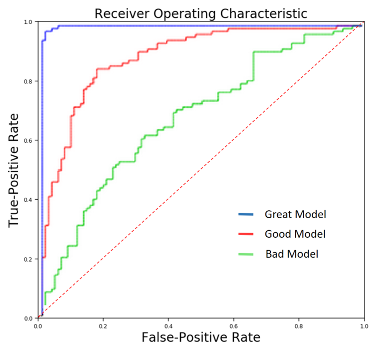

Figure 2: Area under the ROC Curve (AUROC)

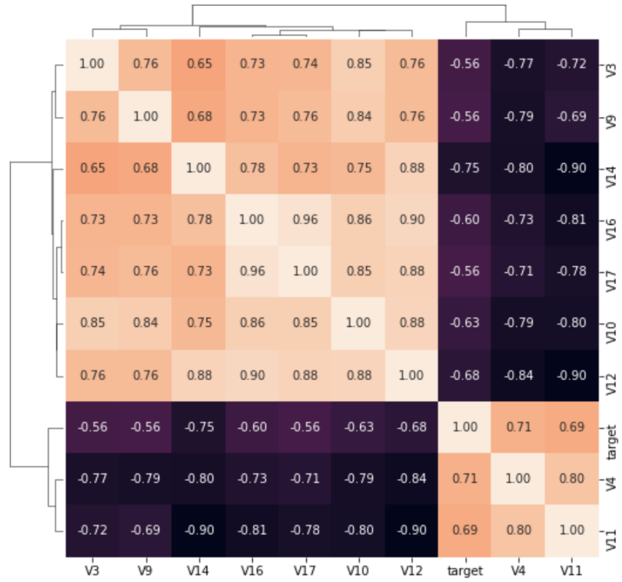

Figure 3: Correlation Matrix

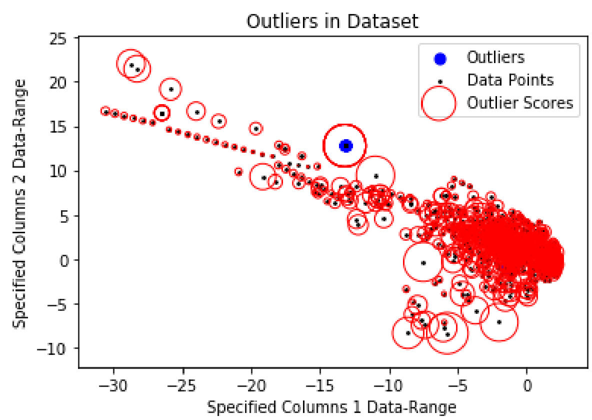

Figure 4: Outliers in Dataset with Threshold

4 EXPERIMENTAL ANALYSIS

In this section we share and discuss the results of our proposed method.

4.1 Dataset Description

Our dataset [18] used for training the model consists of 284,807 records in total, of which only 492 of them are fraudulent cases and thus 0.172% of fraud cases resulting in extremely imbalance dataset. Features V1 to V28 are numerical values obtained from Principal Component Analysis (PCA). The attributes that are not changed by PCA are Time and Amount. Attribute 'time' contains the seconds gone by between each trade and the essential trade in the dataset. The component 'amount' is the total exchange Amount. Feature 'Class' is the response variable and it takes a value of one if there is an occurrence of fraud and zero generally.

4.2 Method Steps

First, it is necessary to recognize the problem and the data set. Since the dataset has an unbalanced distribution, choosing the right sampling method and applying an appropriate classification technique plays an important role in detecting credit card fraud detection. [20] We implemented random sampling, oversampling and undersampling method between the various sampling techniques. After the sampling process, the exploratory data analysis phase was initiated. The features with the dependent variable are associated (with a certain threshold level) can be seen in the figure 3. In the correlation matrix, features with correlations above 0.5 are shown.

After exploratory data analysis is performed, outliers are detected and these outliers removed from the dataset. Local Outlier Factor method used for this process. In figure 4, for the same data, three different color and size dots are drawn on top of each other. These are the data itself, the outlier value of the data and whether it is considered as outlier according to this value. In this way, seven outlier points in total eliminated. This situation shown in figure 4. In this figure, columns selected arbitrarily to visualize in two-dimensional plane. Outliers negatively affect the performance of the KNN algorithm.

Our dataset is standardized after dividing it into two as train and test sets. These stages are data preprocessing stages before applying KNN algorithm.

The AUC value is obtained by applying the KNN algorithm with default parameters as number of neighbors is two and weight function used in prediction is uniform, and the power parameter for the Minkowski metric p is 2 indicating euclidean_distance is shown in figure 5

The optimum parameters are found in the KNN algorithm to achieve the higher accuracy (Figure 6). For this operation, grid search method used. While finding the optimum parameters, the parameters that we tune are; n value to perform the classification process by looking at how many neighbors, weight value that can take uniform and distance values and p value is the distance calculation method to be used when measuring distance between neighbors.

Different methods are applied to increase the accuracy value when using the KNN algorithm. These methods are finding the

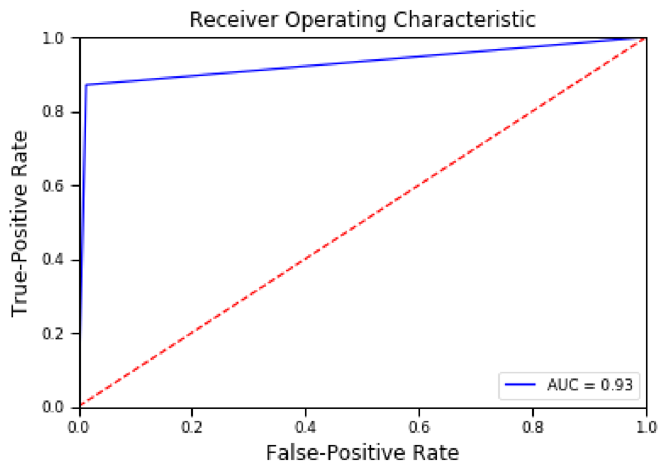

Figure 5: AUC for KNN with default parameters

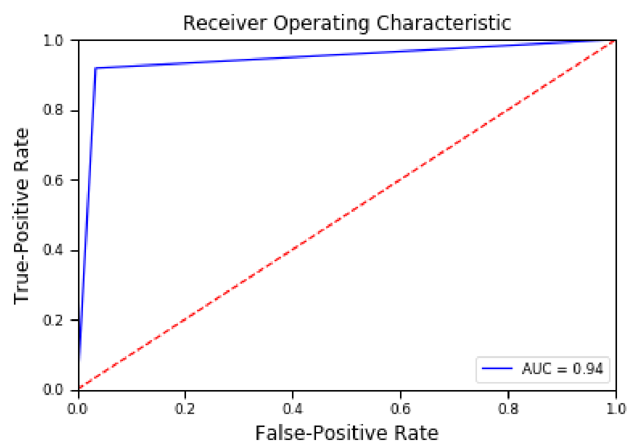

Figure 6: AUC for KNN with Optimum Parameters

optimum parameters, PCA and NCA respectively. It is observed that the success rate and AUC value on the test set obtained in the previous section increased by obtaining the KNN optimum parameters and applying them to the algorithm. When we fit the model using the optimum parameters, the test score was 94%, while the train score was 100%. The model encountered an overfitting problem. While the model achieved the perfect score for the train set, the model for the test set could not be generalized. Model complexity increased with finding the optimum parameters. Therefore, the model is in high variance situation. Since the dataset is multi-dimensional, it is only possible to visualize it with the PCA method. With PCA, we visualized the implemented KNN algorithm in figure 7.

We implemented the KNN algorithm again to the reduced data which is obtained after the dimension reduction process with PCA. The p1 and p2 values in the x and y axes shown in the figure 8, indicate the newly formed principal components. As we expected,

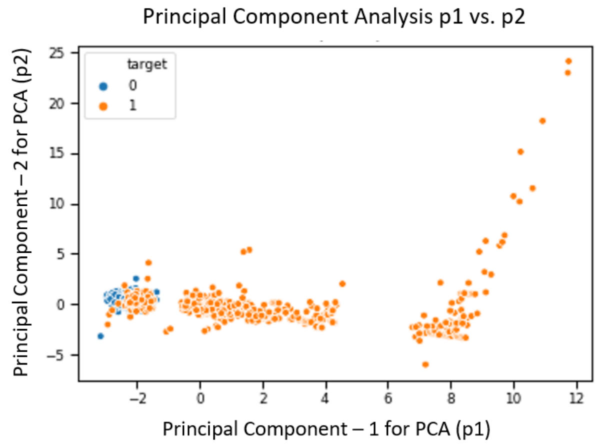

Figure 7: Dimensionality Reduction with PCA

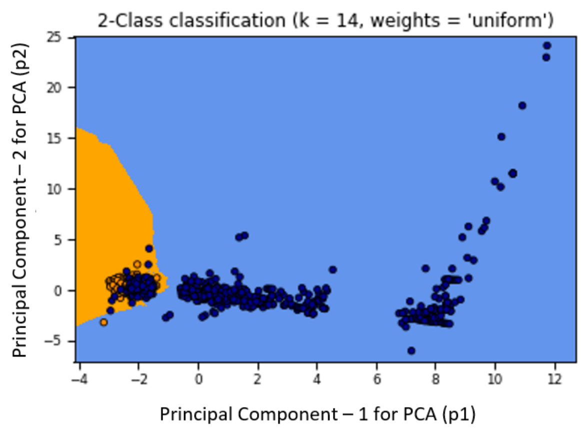

Figure 8: PCA for KNN with Optimum Parameters

our accuracy rate and AUC value decreased partially shown in figure 9

When we analyze the figures comparatively, it is seen that the performance obtained without any optimization, with the standard KNN algorithm performance increased by implemented again after the optimum parameters are found with grid search method. Nonetheless, when the dimension reduction is performed with the PCA method and implement KNN consecutively, the performance decreases again. Ultimately, the best performance among this approaches achieved with the KNN method applied by using the optimum parameters on the reduced data with the NCA method.

In supervised learning, NCA is more successful than standard PCA methods, both in terms of classification performance in predicted notation and visualizing class separation. [22]

NCA needs class label when performing fit operations. The p1 and p2 values in the x and y axes shown in the figure 10 and figure 11 indicate the newly formed NCA principal components. The results on a datasets show that after using the Neighborhood Component Analysis Dimensionality Reduction technique, the performance of the K-Nearest Neighbor provides consistently higher performance

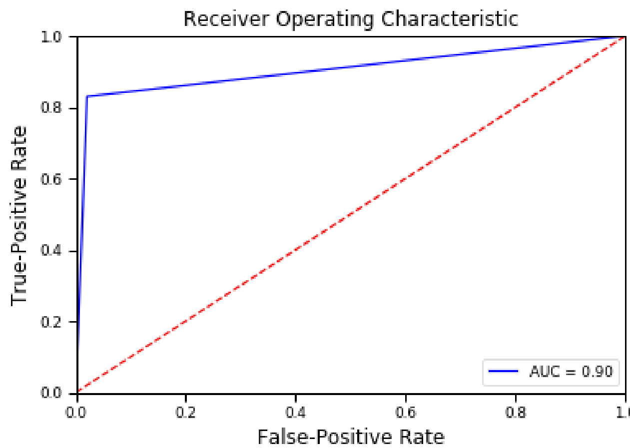

Figure 9: AUC for PCA

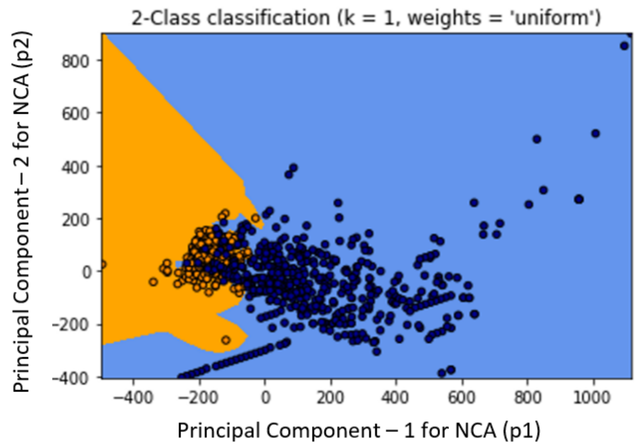

Figure 11: NCA for KNN with Best Parameters

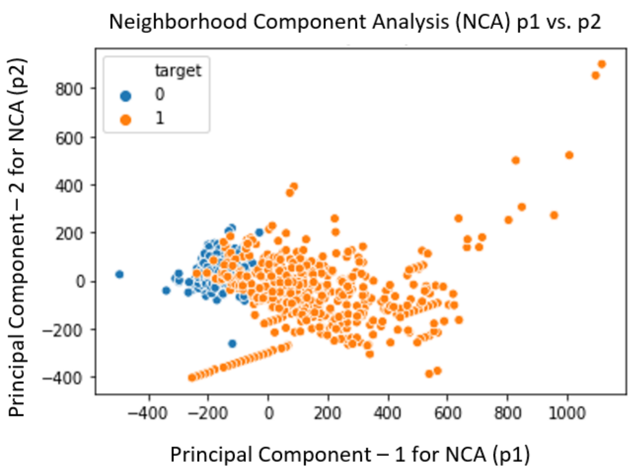

Figure 10: Dimensionality Reduction with NCA

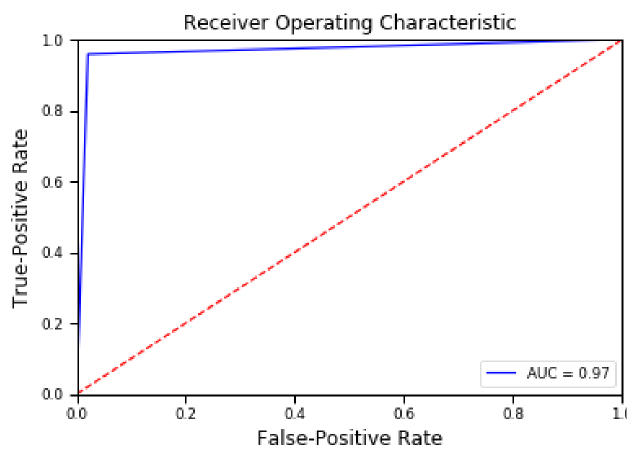

Figure 12: AUC for NCA

than other implemented dimensionality reduction methods PCA. We got better result with NCA showed in figure 11 and higher AUC value shown with figure 12

We implemented NCA, which finds a distance metric that maximizes the leave one out (LOO) error on the training set for a stochastic variant of KNN. The main idea of stochastic neighbor selection is select a single neighbor stochastically and look at the expected votes for each class instead of picking a fixed number K of nearest neighbors and voting their classes. We obtained 0.97 AUC value for fraud detection which can be improved. Because we had memory restrictions to handle complete transformation. Unlike other methods, this classification model is a non-parametric method that does not have any assumptions about the shape of the class distributions or the boundaries between them.

For datasets with unbalanced class distribution, the accuracy value loses its importance. The accuracy metric does not reliably measure performance. This situation makes model training much

Table 1: European Dataset Result

Results Comparison	Method			
	KNN	KNN with Grid Search	PCA	NCA
Accuracy	0.92	0.94	0.90	0.96
AUC	0.93	0.94	0.90	0.97

more difficult. Table 1 summarizes the results obtained on the European Dataset.

5 CONCLUSION AND FUTURE WORKS

In this work, we proposed a new model to identify transactions that are described as fraudulent for credit cards by using historical data and KNN algorithm with better accuracy. We obtained 0.97 AUC score which is higher than KNN models known so far for fraud detections. The implemented model was tested on different datasets to prove the performance. Performance results show that dimension reductions improve the performance of credit card fraud detections.

As a future work, we will be working on different dataset and thus substantially reducing storage and search costs at test time to optimize processes that will completely eliminate the error rate.

ACKNOWLEDGMENTS

The dataset has been collected and analyzed 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.

REFERENCES

[1] G. Goy, C. Gezer and V. C. Gungor, Credit Card Fraud Detection with Machine Learning Methods, *2019 4th International Conference on Computer Science and Engineering (UBMK), Samsun, Turkey, 2019, pp. 350-354*, DOI: 10.1109/UBMK.2019.8906995.

[2] N. K. Gyamfi and J. Abdulai, Bank Fraud Detection Using Support Vector Machine, 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, 2018, pp. 37-41, DOI: 10.1109/IEMCON.2018.8614994.

[3] H. Ali Shukur and S. Kurnaz, Credit Card Fraud Detection using Machine Learning Methodology, *2019 International Journal of Computer Science and Mobile Computing - IJCSMC (Vol.8, No. 3)*

[4] I. Sadgali, N. Sael and F. Benabbou, Fraud detection in credit card transaction using machine learning techniques, *2019 1st International Conference on Smart Systems and Data Science (ICSSD), Rabat, Morocco, 2019, pp. 1-4*, DOI: 10.1109/ICSSD47982.2019.9002674.

[5] A. ARMEL and D. ZAIDOUNI, Fraud Detection Using Apache Spark, *2019 5th International Conference on Optimization and Applications (ICOA), Kenitra, Morocco, 2019, pp. 1-6*, DOI: 10.1109/ICOA.2019.8727610.

[6] L. Zheng *et al.*, A new credit card fraud detecting method based on behavior certificate, *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), Zhuhai, 2018, pp. 1-6*, DOI:10.1109/ICNSC.2018.8361328.

[7] V. N. Dornadula, S. Geetha, Credit Card Fraud Detection using Machine Learning Algorithms, *Procedia ComputerScience, Volume 165, 2019, Pages 631-641, ISSN 1877-0509*, https://doi.org/10.1016/j.procs.2020.01.057. (http://www.sciencedirect.com/science/article/pii/S187705092030065X)

[8] Z. Kazemi and H. Zarrabi, Using deep networks for fraud detection in the credit card transactions, *2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, 2017, pp. 0630-0633*, DOI: 10.1109/KBEI.2017.8324876.

[9] A. Pumsirirat and L. Yan, Credit Card Fraud Detection using Deep Learning based on Auto-Encoder and Restricted Boltzmann Machine, *International Journal of Advanced Computer Science and Applications (ijacsa), 9(1),218,* http://dx.doi.org/10.14569/IJACSA.2018.090103

[10] E. Kim, J. Lee, H. Shin, H. Yang, S. Cho, S. Nam, Y. Song, J. Yoon, J. Kim, Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning, Expert Systems with Applications, Volume 128, 2019, Pages 214-224, ISSN 0957-4174https://doi.org/10.1016/j.eswa.2019.03.042

[11] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim and A. K. Nandi, *Credit Card Fraud Detection Using AdaBoost and Majority Voting, in IEEE Access, vol. 6, pp. 14277-14284, 2018,*DOI: 10.1109/ACCESS.2018.2806420.

[12] C. Guo, H. Wang, H. Dai, S. Cheng and T. Wang, "Fraud Risk Monitoring System for E-Banking Transactions, *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), Athens, 2018, pp. 100-105.*

[13] K. Masoud, F. Mehdi, Fraud Detection In Banking Using Knn (K-Nearest Neighbor) Algorithm, *2016 International Conference On Research In Science And Technology, Volume 5*

[14] N. Malini and M. Pushpa, Analysis on credit card fraud identification techniques based on KNN and outlier detection, *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics AEEICB), Chennai, 2017, pp. 255-258,* DOI: 0.1109/AEEICB.2017.7972424.

[15] A. Fernández, S. García, F. Herrera, and N. V. Chawla,"SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary", J. Artif. Int. Res. 61, 1 (January 2018), 863–905.

[16] M. F. Zeager, A. Sridhar, N. Fogal, S. Adams, D. E. Brown and P. A. Beling, "Adversarial learning in credit card fraud detection,"*2017 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, 2017, pp. 112-116*, DOI: 10.1109/SIEDS.2017.7937699.

[17] A. Mishra and C. Ghorpade, Credit Card Fraud Detection on the Skewed Data Using Various Classification and Ensemble Techniques, *2018 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, 2018, pp. 1-5,*DOI:10.1109/SCEECS.2018.8546939.

[18] https://www.kaggle.com/mlg-ulb/creditcardfraud

[19] Siddique, Md. Abu & Sakib, Shadman & Rahman, Md., Performance Analysis of Deep Autoencoder and NCA Dimensionality Reduction Techniques with KNN, ENN and SVM Classifiers,2019

[20] H. Akira, O. Seiichi, Kenji Doya, I. Kazushi, L. Minho, L. Derong, Neural Information Processing, *23rd International Conference, ICONIP 2016, Kyoto, Japan, October 16–21, 2016, Proceedings, Part IV, 2016, Springer International Publishing, ISBN:978-3-319-46681-1*, DOI: 10.1007/978-3-319-46681-1

[21] J. Goldberger, T. Roweis, G E Hinton, R. R Salakhutdinov, Neighborhood components analysis, *NIPS'04: Proceedings of the 17th International Conference on Neural Information Processing Systems, December 2004 Pages 513–520*

[22] Sam Roweis, Neighbourhood Components Analysis, University of Toronto Department of Computer Science, *Machine Learning Summer School, Taiwan July 24, 2006*