Credit Card Fraud Detection by Implementing Machine Learning techniques

Debachudamani Prusti[1], S S Harshini Padmanabhuni[1], Santanu Kumar Rath[1]

[1]National Institute of Technology Rourkela, Odisha, India
debaprusti@gmail.com, harshinipadmanabhuni@gmail.com, rath.santanu@gmail.com

Abstract. Application of machine learning techniques for fraud detection in the credit card operations has been an important component of research in the domain of digital transactions. The evolution of various machine learning techniques like classification and clustering have shown the requirement for application of related algorithms in detecting frauds of credit card transactions. In this study, we have proposed the application of various classification techniques by using machine learning algorithms for detecting the accuracy of the fraud detection. We have implemented some commonly considered classification methods used for a large volume of data. The different algorithms we have evaluated are Naïve Bayes classifier, Extreme learning machine (ELM), K-Nearest Neighbor (K-NN), Multilayer Perceptron (MLP) and Support Vector Machine (SVM). We have proposed a model by hybridizing SVM, K-NN and MLP models, in which the prediction accuracy has improved significantly.

Keywords: Credit card fraud, Classification Techniques, Fraud detection, Prediction accuracy

1 Introduction

Credit card fraud is a very pertinent problem and the way of fraudulent source of funds in an online transaction throughout the credit card industry [1]. Of late, various researchers, as well as applicationists, have shown interest in the analysis of fraud issues in the credit card by applying machine learning algorithms. In real situations, it is necessary to respond in a very short time to stop fraudulent transactions. Additionally, due to the varying behavior of fraudulent methods, a frequent re-training is essential for any credit card fraud detection model.

Still the major troublesome problem to handle the counterfeit is that, in detecting the fraud for credit cards is a cost sensitive issue, in which the expense delivered by a false alarm is not quite the same as the cost of a false negative class [2]. When the model predicts an online transaction as fraudulent, but actually it is not (false positive), the organization has both an administrative cost and a large decline in consumer satisfaction. In addition, when the model is able to identify a fraudulent transaction (false negative), unconditionally the loss occurs for that transaction. Most importantly, it is not sufficient to have a fixed cost variance in between false positives as well as false negatives, as the total cost

of the transactions vary in a significant way; Hence, its financial impact is not fixed.

In this study, a model has been proposed to classify a transaction to be either fraudulent or not. Many statistical and computational methods, including Bayes classifier, discriminant analysis, nearest neighbor and logistic regression have been proposed in literature to develop models for the prediction of accuracy [3]. Other artificial neural networks and classification based trees, artificial intelligence and machine learning techniques were also applied for the classification task [4] [7]. In view of risk control, estimation of default will be more informative, as compared to classifying the customers into fraudulent and non-fraudulent. Hence, the major issue lies in estimating the probability of defaults, produced from different data mining techniques that represents the actual probability of defaults.

2 Credit card Fraud Issue

Credit card fraud is a technique of stealing the identity where the unauthorized person uses some other persons credit card credentials to pay for purchasing or to transfer amounts from the cardholders account [5]. The credit card fraud also affects the fraudulent transaction of debit card and can exploit by stealing the original card. It illegally uses the cardholders account information, which includes the credit card credentials like card number, pin number, name and address.

Despite the fact that credit card takes numerous structures, there are several methods of important classifications [6]. Counterfeit, because of lost cards and stolen cards for the most part represents a specific base dimension of misrepresentation action. The measure of this base dimension can be influenced by general monetary conditions. Fraud because of fake cards has turned into a current developing issue in the course of recent years, in spite of the more modern card producing innovations (3D images on the cards) and the data encryption on the attractive stripe. Clearly, fraud will in general be an increasingly sorted out and deliberate issue in specific areas, instead of the more artful and subsequently determined nature of most counterfeit because of lost or stolen cards.

2.1 Current Methods of Fraud Detection

The decent variety of fraud action as confirmed by the numerous forms of counterfeit that makes the detection of deceitful behavior is a complex task [7]. In most of the financial institutions, some piece of the scrutiny procedure to apply for new credit cards includes routine data checks to identify probable deceitful applications. Sometimes, the scrutiny process of the application forms for obvious strategies for handwriting has driven investigators to spot fake applications put together by sorted out criminal components.

3 Application of Machine Learning Models for Fraud Detection

In machine learning algorithms, classification technique is regarded as an occurrence of supervised learning, i.e., training where a learning set of accurately viewed perceptions is fully accessible. The corresponding unsupervised learning is called as clustering, and includes gathering information into classifications dependent on some proportion of intrinsic similarity or separation. Our study considers the following classifiers.

3.1 Naïve Bayes Classifier

The Naïve Bayes classifier works on Bayes theory of Conditional Probability [8]. It assumes that the attributes are independent to each other (Naïve) which is not true in reality because there always exists dependency between the attributes. It gives hypothetical support to different classifiers that implicitly utilizes Bayes hypothesis. The Naïve assumption decreases the computational complexity because of the class conditional independence.

3.2 Extreme learning machines (ELM)

Extreme learning machines (ELM) are feedforward neural network systems, mostly applied for classification, clustering, regression, compression, feature learning and sparse approximation [9]. It can be implemented using only one layer or numerous layers of hidden nodes. The parameters of hidden nodes are not required to be tuned. These models can deliver a generalized performance and learn a number of times quicker than systems trained utilizing backpropagation.

Generally, all parameters in the feedforward neural network are often tuned in a manner where the dependency exists between various layers of parameters (i.e. loads and biases). For the past few years, gradient descent based techniques are being utilized in different learning algorithms like feedforward neural systems. In any case, obviously gradient descent based learning techniques are commonly eased back because of ill-advised steps of learning which can effectively combine to reach the local minimum.

3.3 K-Nearest Neighbor (K-NN)

The classifier, K-Nearest Neighbor (K-NN) is used for classification problems which are dependent on learning algorithms by analogy [10]. For a given unknown sample, the algorithm looks for the K-Nearest Neighbors in the sample space. Distance is calculated in the form of separation. The test sample is allotted to the nearest recognized class from its K-Nearest Neighbors. The advantage of this methodology is that we do not need to build a model before the classification. The disadvantages include a non-delivery of straightforward classification probability formula and the proportion of separation and the cardinality k of the area exceptionally influence the predictive accuracy.

3.4 Multilayer Perceptron (MLP)

The Multilayer Perceptron (MLP) is a feedforward neural network used for classification and regression problems [11]. A MLP has minimum of three layers: an input layer, one or more hidden layer and an output layer. Apart from the input layer nodes, other two nodes are the neurons that utilizes a nonlinear activation function (NAF). MLP uses a supervised learning system known as backpropagation. It is used for training purpose. Its various layers, nonlinear activation functions differentiates Multilayer Perceptron from the linear perceptron. In MLP all the neurons have a linear activation function, i.e., a linear function which points out the weighted contributions to the output layer. In MLP a few neurons utilize NAF, developed to display the recurrence of activity possibilities, or terminating, of organic neurons.

3.5 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a statistical learning strategy and has effective application in a scope of issues [12]. They are firmly identified with neural network systems and with kernel functions; they can be viewed as an elective method to get neural system classifiers. SVM model is a supervised machine learning technique that is connected to abnormality recognition (Anomaly detection) in the one-class setting. Such methods utilize one class learning systems for SVM and take in an area that contains the training samples. The essential thought of SVM classification algorithm is to develop a hyperplane as the decision plane which making the separation between the positive and negative mode maximized. The effectiveness of SVMs originates from two crucial properties they have kernel representation and margin optimization.

This model finds a very uncommon sort of linear model, the most extreme edge hyper plane, and it classifies all training data instances effectively by isolating them into right classes through a hyperplane. The most extreme edge hyper plane is one that gives most prominent partition between the classes. There is dependably at least one support vector for each class and often more. In credit card counterfeit recognition, for each test data instance, it decides whether the test example falls inside the learned area. At that point, if a test instance occurs inside the training area, then it is confirmed as expected otherwise anomalous. This model shows that it has a higher accuracy of detection when compared to other algorithms. It likewise has a superior time effectiveness and generalization capacity.

4 Dataset used

The optimized use of dataset is an important requirement for performing the classification technique. The dimension of the dataset can affect both training and testing of a model. The data we have used in this study is the Credit card fraud classification data (https://archive.ics.uci.edu/ml/machine-learning-databases/00350/). It has a total 690000 data with a dimension of 23 columns

and 30000 rows. For our study purpose, we have considered 80% of the data for training purpose and 20% of the data for testing purpose to get the optimized value of accuracy. The accuracy percentage is optimized when we split the dataset as 80% of it for training and 20% of it for testing. The use of Credit card fraud classification dataset can improve the efficiency of our research by saving the data collection time and data access time.

5 Result and Discussion

5.1 Experimental setup

We have implemented five classification algorithms in Matlab platform version R2018a. The system configuration is of i7 processor with 3.4 GHz clock speed. The secondary memory space and main memory space in each system is 1TB and 10GB respectively.

5.2 Evaluation Parameters

Confusion Matrix
A confusion Matrix is a representation technique for the execution of classification models. The confusion matrix demonstrates to us the quantity of accurately and inaccurately classified samples, contrasted with the real results (target value) in the test information. A confusion matrix framework of two class classification is a 2 X 2 table designed by adding the quantity of the four results of a binary classifier and we more often denote them as TP, FP, TN, and FN.

A binary classifier predicts a test dataset as either positive class or negative class for all data instances. This prediction (or classification) produces four results, for example, true positive (TP), false negative (TN), false positive (FP) and false negative (FN).

- True positive (TP): correct prediction of a positive class
- False positive (FP): incorrect prediction of a positive class
- True negative (TN): correct prediction of a negative class
- False negative (FN): incorrect prediction of a negative class

Various performance parameters have been evaluated from the confusion matrix for our proposed model.

Accuracy: Accuracy parameter is evaluated to know upto what extent the classifier is correct. We calculate it by dividing the number of all correct predictions with the total number of the dataset. The best accuracy is assumed 1.0, whereas 0.0 is the worst. We can also calculate it by using the formula

$$1-\text{Error Rate} \tag{1}$$

We have calculated the accuracy for all the five classifiers by using the confusion matrix.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \tag{2}$$

Precision: Precision is evaluated to find the total relevant positively classified instances from the retrieved instances. We calculate precision by dividing the total number of correct positive predictions with the total number of positive predictions. It is also known as positive predictive value (PPV). The best precision is 1.0, whereas 0.0 is the worst.

$$Precision = TP/(TP + FP) \tag{3}$$

Sensitivity: Sensitivity is calculated to correctly identify the true positive rate and to estimate the error. It specifies that how good is the test to detect the positive classes. We calculate sensitivity by dividing the total number of correct positive predictions with the total number of positives. It is also known as true positive rate (TPR) or recall. The best sensitivity is 1.0, whereas 0.0 is the worst.

$$Sensitivity = TP/(TP + FN) \tag{4}$$

Specificity: It is evaluated to identify how accurately it identifies the false alarms. We calculate specificity by dividing the total number of correct negative predictions with the total number of negatives. It is also known as true negative rate (TNR). The best specificity is 1.0, whereas 0.0 is the worst.

$$Specificity = TN/(TN + FP) \tag{5}$$

F1-Score: In two class classification, the F1 score (also F-measure) is a measure to find the testing accuracy. It considers both the precision as well as recall of the test to calculate the score.

$$F1 - score = 2TP/(2TP + FP + FN) \tag{6}$$

5.3 Proposed model

We have considered five models for classification such as Naïve Bayes, ELM, K-NN, MLP and SVM to find prediction accuracy. Among all the individual models, SVM classifier has the maximum prediction accuracy i.e., 81.40%.

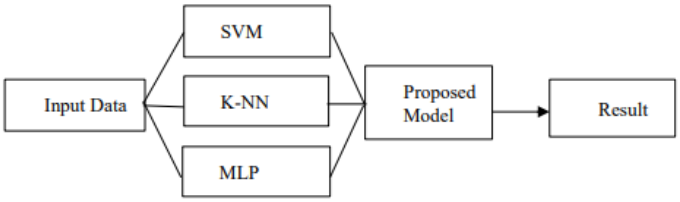

Fig. 1. Block Diagram for the proposed model

The combined model for prediction result is more robust and better accuracy can be achieved. Ensemble of machine learning classifiers improvise the predictive performance as compared to a single model [13]. More prominently, it decreases the variance and increases the prediction accuracy. We have combined SVM, K-NN and MLP models from heterogeneous family to design a combined model for finding a better predictive result. The three classifiers i.e., SVM, K-NN and MLP are chosen as their individual accuracy are good as compared to Naïve Bayes and ELM classifiers. The block diagram of the proposed model is shown in Figure 1. The decision boundary is optimized by stacking of classifiers.

Table 1. Confusion Matrix for the proposed model

	Actual Class	
Predicted Class	True Positive 4547	False Positive 159
	False Negative 896	True Negative 398

The parameters of the confusion matrix for our proposed model to classify the samples as true positive class, true negative class, false positive class and false negative class are shown in Table 1. The objective is to maximize the correct prediction and to minimize the false alarms.

The testing data as input data in the proposed model is given to yield a better result. In this proposed model, the prediction accuracy obtained is 82.42% and this accuracy is better than the values obtained using the individual models. The margin of separation between the optimal boundary values reduces the prediction error for the binary classification techniques. Based on the result shown in the Table 2, we observed the highest accuracy for the proposed model. Using the proposed approach, we achieved better accuracy as compared to individual models.

Table 2. Accuracy results for various models

Classifiers / Parameters	Naïve Bayes	ELM	K-NN	MLP	SVM	Proposed Model
Accuracy	0.4990	0.7988	0.8040	0.8110	0.8140	0.8242
Precision	0.8875	0.8045	0.8218	0.9515	0.6937	0.9662
Sensitivity	0.4085	0.9799	0.9555	0.8444	0.2634	0.8354
Specificity	0.8176	0.1613	0.2705	0.6911	0.9670	0.7145
F1-score	0.5595	0.8836	0.8836	0.8948	0.3818	0.8960

In Table 2, the parameters such as accuracy, precision, sensitivity, specificity and F1-score are calculated for different models. The proposed model, as shown

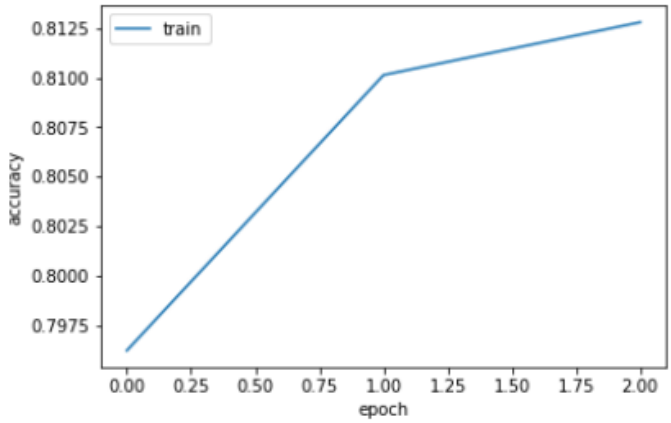

Fig. 2. Accuracy of proposed model for training data

in Table 2 helps to find better prediction accuracy. We observed that the highest accuracy turns out to be 82.42% for the proposed model.

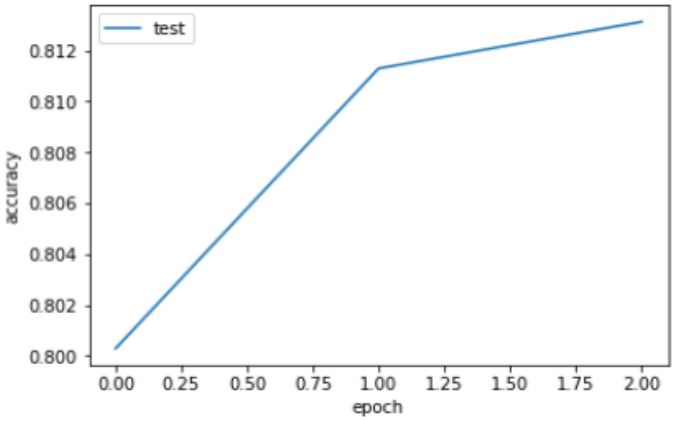

Fig. 3. Accuracy value of proposed model for testing data

We have plotted Figure 2 and Figure 3 for training and testing data respectively to show the increase in accuracy percentage with respect to periodical time. In Figure 2, we have calculated the accuracy value for training purpose by considering 80% of the data from the dataset. In Figure 3, we have calculated the accuracy value for testing purpose by considering 20% of the data from the dataset.

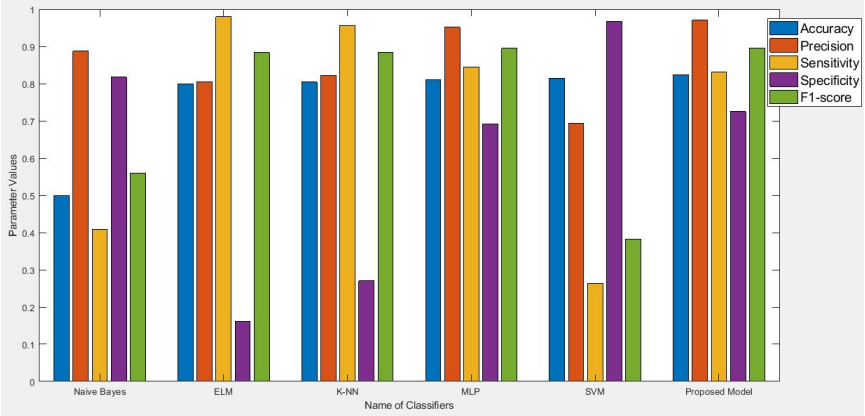

Fig. 4. Comparison between classifiers for various parameters

In Figure 4, we have shown the comparison between the various parameters like accuracy, precision, sensitivity, specificity and F1-score with respect to number of classifiers we discussed as well as for our proposed model. We observed that the accuracy is more for the proposed model as compared to other classifiers.

6 Conclusion

In our study, we have examined the five important classification models of machine learning techniques and compared the performance analysis of classification and predictive accuracy between them. While considering the prediction accuracy among the five classification models, the results show that there are marginal differences in error rates between the five individual models and it is observed that the accuracy percentage of SVM model is 81.40% among the individual models. In the proposed model, the correctly predictive class is more with minimized false alarms. The accuracy percentage for the proposed model is observed to be 82.42%. As compared to the individual model's prediction accuracy value,the accuracy percentage for the proposed model is observed to be highest.

References

1. Dal Pozzolo, Andrea, Olivier Caelen, Yann-Ael Le Borgne, Serge Waterschoot, and Gianluca Bontempi. "Learned lessons in credit card fraud detection from a practitioner perspective." *Expert systems with applications* 41, no. 10, 4915-4928, 2014
2. Sahin, Yusuf, and Ekrem Duman. "Detecting credit card fraud by ANN and logistic regression." In Innovations in *Intelligent Systems and Applications (INISTA), 2011 International Symposium on*, pp. 315-319. IEEE, 2011.

3. Kou, Yufeng, Chang-Tien Lu, Sirirat Sirwongwattana, and Yo-Ping Huang. "Survey of fraud detection techniques." In *Networking, sensing and control*, 2004 IEEE international conference on, vol. 2, pp. 749-754. IEEE, 2004.
4. Brause, R., T. Langsdorf, and Michael Hepp. "Neural data mining for credit card fraud detection." In *Tools with Artificial Intelligence, 1999. Proceedings. 11th IEEE International Conference on*, pp. 103-106. IEEE, 1999.
5. LNCS Dorronsoro, Jose R., Francisco Ginel, Carmen R. Snchez, and Carlos Santa Cruz. "Neural fraud detection in credit card operations." *IEEE transactions on neural networks*, vol. 8, pp. 827-834, 1997.
6. Bahnsen, Alejandro Correa, Djamila Aouada, Aleksandar Stojanovic, and Bjrn Ottersten. "Feature engineering strategies for credit card fraud detection." *Expert Systems with Applications* 51, 134-142, 2016
7. Jagielska, Ilona, and Janusz Jaworski. "Neural network for predicting the performance of credit card accounts." *Computational Economics* 9, no. 1, 77-82, 1996.
8. Ng, Andrew Y., and Michael I. Jordan. "On discriminative vs. generative classifiers: A comparison of logistic regression and Naïve bayes." In *Advances in neural information processing systems*, pp. 841-848. 2002.
9. Huang, Guang-Bin, Qin-Yu Zhu, and Chee-Kheong Siew. "Extreme learning machine: a new learning scheme of feedforward neural networks." In *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 2, pp. 985-990. IEEE, 2004
10. Fukunaga, Keinosuke, and Patrenahalli M. Narendra. "A branch and bound algorithm for computing k-nearest neighbors." *IEEE transactions on computers* 100, no. 7, 750-753, 1975.
11. Collobert, Ronan, and Samy Bengio. "Links between perceptrons, MLPs and SVMs." In *Proceedings of the twenty-first international conference on Machine learning*, p. 23. ACM, 2004
12. Hearst, Marti A., Susan T. Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. "Support vector machines." *IEEE Intelligent Systems and their applications* 13, no. 4, 18- 28, 1998.
13. Dietterich, Thomas G. "Ensemble methods in machine learning." In *International workshop on multiple classifier systems*, pp. 1-15. Springer, Berlin, Heidelberg, 2000.