International Journal for Modern Trends in Science and Technology, 8(09): 74-79, 2022

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0809012
Available online at: http://www.ijmtst.com/vol8issue09.html

An Efficient Prototype for Credit card Fraud Detection using Machine Learning

Divya C[1] | Harshitha K[2] | Harshitha D J[2]

[1]Assistant Professor, Department of Information Science and Engineering, Kalpataru Institute of Technology
[2]Department of Information Science and Engineering, Kalpataru Institute of Technology

To Cite this Article
Divya C, Harshitha K and Harshitha D J. An Efficient Prototype for Credit card Fraud Detection using Machine Learning. International Journal for Modern Trends in Science and Technology 2022, 8(09), pp. 74-79. https://doi.org/10.46501/IJMTST0809012

Article Info
Received: 20 August 2022; Accepted: 10 September 2022; Published: 17 September 2022.

ABSTRACT

As, the numbers of hackers have increased, credit card fraud cases have incredibly increased. Therefore it is crucial to find a solution to this issue or crisis. In this project, a customer's credit card transactions are masked using Principal component analysis (PCA) and the data classifier algorithms are applied to analyze whether a transaction carried out is fraudulent or not. Once the data values are processed, they are modeled using machine learning algorithms. Performance is compared between these modeling algorithms using test parameters or few metrics.

KEYWORDS: Credit card fraud detection, Machine Learning, PCA

1. INTRODUCTION

Credit cards provide cashless transactions which helps the customers to carry out hi/her purchases in a much easier manner. As per the survey carried out in the year 2021, nearly or around 62 million credit cards were used in India alone. Fraud detection is a set of tasks performed to prevent either money or property from being obtained through false illusion.

Advantages of credit cards:

1) They provide cashless transaction
2) Growing admiration for e-marketing
3) They provide unlimited reward points

advantages of machine learning includes it provides continuous improvement based on the data provided which can be used to improve credit score based on which enormous

Discounts for shopping become available.

The confrontations faced in credit card fraud detection includes the inaccessibility to the data in real time, the unstabilized set of data and the measurements with respect to the data set, parameters used for evaluation and finally the behavior of fraudster which is found to be dynamic over time.

Machine Learning is the recent trends in technology which authorizes computers to be self taught from the data which has been trained and improved over time without the need to be programmed. Various by the users, helps us choose appropriate algorithms, helps in collecting and analyzing data and so on.

STRUCTURE OF PAPER

The paper is organized as follows: In Section 1, the introduction of the paper is provided along with the structure, important concepts and the advantages of using credit cards. In Section 2 we discuss related work. In Section 3 we have the proposed architecture related to the credit card fraud detection system. Section 4 shares the test results and performance comparison for different data classification algorithms used. Section 5 tells us about the conclusion of this work with scope for future enhancements.

2. RELATED WORK

Sharayu Pradeep and Nitin [2], have implemented credit card fraud detection system wherein the goal of their implementation is to minimize false alarms for authentic transactions. ML algorithms are used for detecting local outlier factor and even hidden Markov model has been used for fraud detection.

Thirunavukkarasu M et.al [3], have proposed a credit card fraud detection system for real world scenarios. This system provides the required properties needed to determine the felonious and unauthorized transactions. Credit card data set is collected by the users and they are classified as trained and tested data set using random forest algorithm and decision tree. Based on the performance metrics used like accuracy, sensitivity, specificity and precision, random forest algorithm is found to have highest accuracy.

A credit card fraud detection system which is found to be upgraded using machine learning has been proposed by Ramya M et.al[4]. They have put forward an original real time system by depicting 4 different patterns of fraudulent transactions using suitable algorithms. In order to address the issues faced like detecting frauds in a credit card transaction, predictive analytics is used for this purpose. Finally, with the help of GUI, end user is notified about it.

Work carried out by Maniraj SP et.al [5] showed a credit card fraud detection system where their template or framework is used to acknowledge or admit whether a new transaction which has been carried out is found to be fraudulent or not. Their intention is to discover all transactions which are found to be fraudulent and then reduce the incorrect fraud classifications. Here several anomaly awareness algorithms have been deployed such a local outlier factor and isolation forest. This paper has explained how machine learning can be applied to get accurate results in fraud detection.

An ensemble classifier has been used for fraud detection in the work carried out by Masoumeh et.al [6]. The basic machine learning algorithms like Naïve Bayes (NB), support vector machines (SVM), K-nearest neighbor (KNN) have been used here. Among all, ensemble method has been recognized as popular and common method due to its predictive performance on real world problems. Performance evaluation has been carried out lastly to justify the advantages of bagging ensemble algorithm.

3. PROPOSED ARCHITECTURE

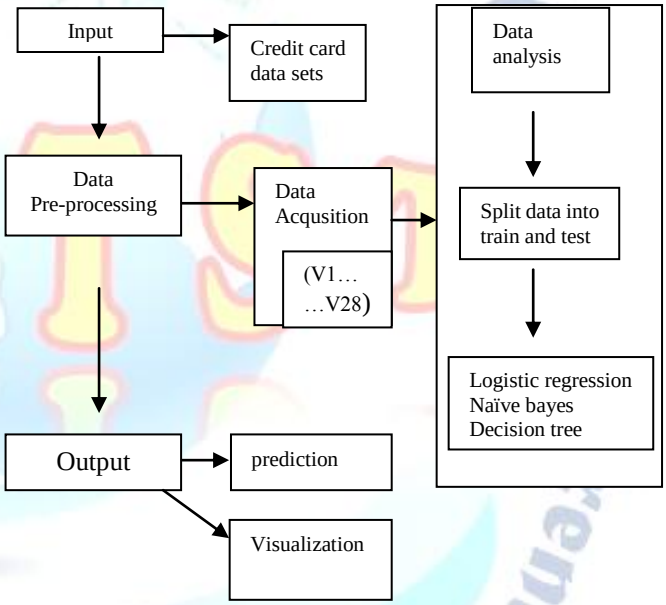


Fig 3.1 Proposed Framework for Credit card fraud detection

Here dataset has been collected from Kaggle, an exploration website for different datasets. There exists 31 columns in this dataset out of which 28 are named as V1-V28. They all are masked for security purpose by using PCA. There are also columns like time, amount and class. Here the amount is capped by the credit card company. If the amount is less than the capped value, then the transaction is considered to be fraudulent. If it is more than the capped value then, it is considered to be non fraudulent. The results of this are shown in the next section. In this project, we have implemented data classification algorithms like Logistics regression,

Decision –tree and Naïve bayes using scikit-learn. It is a simple and well-structured or well organized tool used to predict the data analysis section. It is built using Numpy, SciPy and matplotlib modules. Its features includes classification, clustering and regression algorithm and is designed to collaborate with many libraries. In our project, Jupiter Notebook platform is used to demonstrate or to illustrate this.

A. Logistic Regression

Using this algorithm, given a set of independent variables, it is always possible to predict dependent variables. The output is either a explicit value or disjunct value. These values can be represented as either 0 or 1, true or false

and so on. Rather than this representation, it is much better to represent them in a range which lies between 0 and 1. Since, this algorithm is used to classify samples, it can be categorized as classification algorithm.

The probability which lies either as 0 or 1 can be represented with the help of threshold value. Any value that is above the threshold is considered as 1 and below threshold is considered as 0.

Equation: In logistic regression, a straight line y can be between 0 and 1 only, so dividing it by (1-y)

y/(1-y); 0 for y=0, infinity for y=1

But when the range is needed between – [infinity] to +[infinity], equation becomes $\log[y/(1-y)]=b_0+b_1x_1+b_2x_2+b_3x_3+\ldots.+b_nx_n$

The pseudo code for this algorithm is written as

```
In [33]: xtrain_rus = rus_train.drop(columns=['V16', 'V18', 'Class'], axis=1)
         ytrain_rus = rus_train['Class']

In [34]: logReg_rus = LogisticRegression()
         logReg_rus.fit(xtrain_rus, ytrain_rus)

Out[34]: ▾ LogisticRegression
         LogisticRegression()

In [35]: # For the testing data
         xtest_rus = rus_test.drop(columns=['V16', 'V18', 'Class'], axis=1)
         ytest_rus = rus_test['Class']

         predictions_rus = logReg_rus.predict(xtest_rus)
         result_rus = evaluate_model(ytest_rus, predictions_rus)
```

B. Naïve Bayes

It helps to find the probability of the event which is to occur based on the probability of another event which has already occurred. The generalized equation is given as

$P(M|N)=P(N|M)P(M)/P(N)$

where M and N are events and P(N) !=0 with respect to the data set. Bayes theorem can be applied as

$P(z|A)=P(A|z)P(z)/P(A)$ where z is class variable and A is a feature dependent vector whose size is n where $A=(a_1,a_2,a_3,\ldots.a_n)$

The pseudo code for this algorithm is written as

```
from sklearn.naive_bayes import GaussianNB

NaiveBayes = GaussianNB()

NaiveBayes.fit(xtrain,ytrain)

predicted_values = NaiveBayes.predict(xtest)
x = metrics.accuracy_score(ytest, predicted_values)
acc.append(x)
model.append('Naive Bayes')
print("Naive Bayes's Accuracy is: ", x)

print(classification_report(ytest,predicted_values))
result_base = evaluate_model(ytest, predicted_values)
```

C. Decision Tree

Here, the traversal starts from the root node. It differentiates the values of the root with the attribute or value which has been recorded. After that it bifurcates as branches and moves onto the next or new node. For the remaining nodes, this process is repeated wherein the attribute values are compared with the subnodes and moves further until the auxiliary node has been reached.

Steps of this algorithm include:

1: Traverse the tree starting from root node say Y

2: The best attribute is to be found among the dataset which is available

3: Partition Y into subgroups which might include the values for the best attributes chosen

4: The best attribute might be included within the decision tree node which has to be generated.

5: The new decision trees can be created recursively using the subgroups of the dataset created in step 3. This process should be continued until the nodes cannot be further classified and ultimate node is called as auxiliary node.

The pseudo code for this algorithm is written as

```
from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics
from sklearn.metrics import classification_report
DecisionTree = DecisionTreeClassifier(criterion="entropy",random_state=2,max_depth=5)

DecisionTree.fit(xtrain,ytrain)

predicted_values = DecisionTree.predict(xtest)
x = metrics.accuracy_score(ytest, predicted_values)
acc.append(x)
model.append('Decision Tree')
print("DecisionTrees's Accuracy is: ", x*100)

print(classification_report(ytest,predicted_values))
result_base = evaluate_model(ytest, predicted_values)
```

4. TEST RESULTS AND PERFORMANCE COMPARISON

The code compares the number of false positives it detected with the actual values. Performance metrics like precision, recall, f1-score and support are used to determine the efficiency of the algorithms which have been implemented. The result with respect to whether a new transaction which has been carried out is fraudulent or not fraudulent is as shown below.

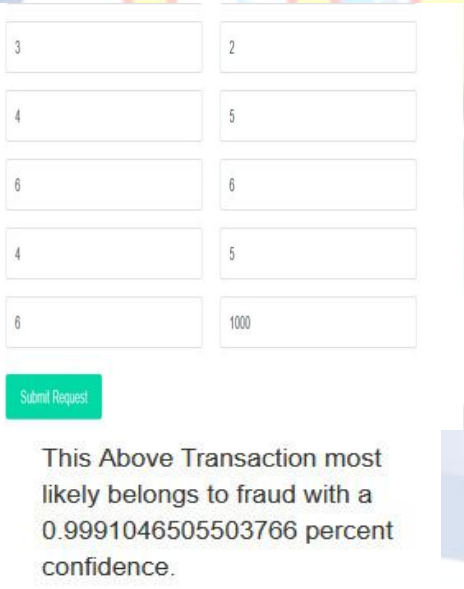


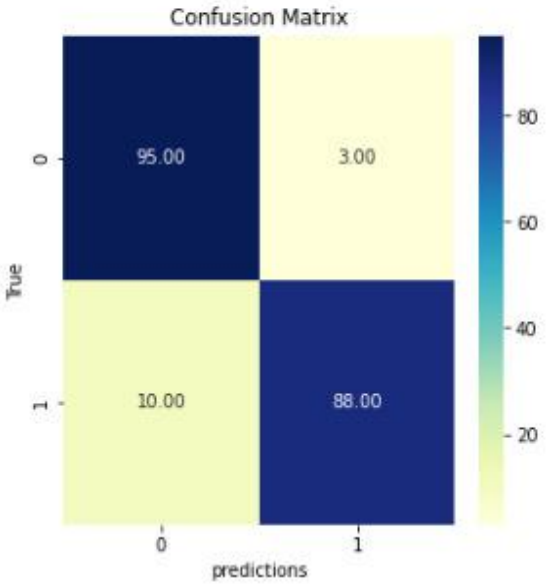
Fig 4.1: Fraudulent Transaction

Fig 4.2: Non- fraudulent Transaction

The results based on performance metrics used for Logistic Regression, Naïve Bayes and Decision tree algorithms together with their confusion matrices is as shown next.

Logistic Regression

```
Accuracy: 0.9336734693877551
AUC     : 0.9336734693877552
Precision: 0.967032967032967
Recall: 0.8979591836734694
F1_score: 0.9312169312169313
```

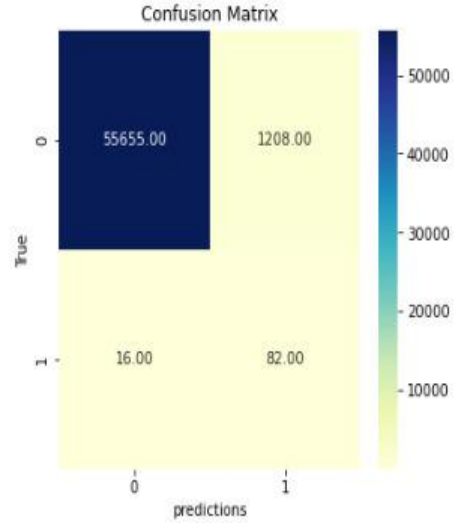


Naïve Bayes

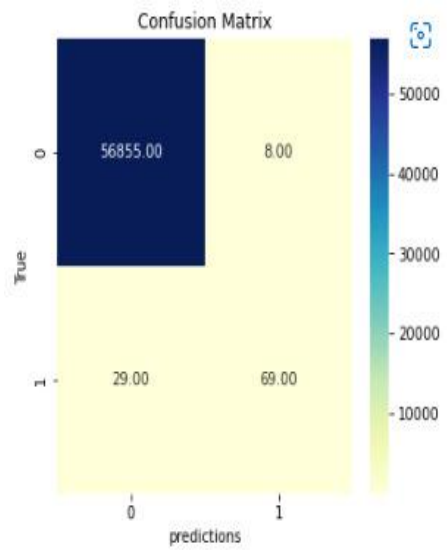
Naive Bayes's Accuracy is: 0.9785116132090378

	precision	recall	f1-score	support
0	1.00	0.98	0.99	56863
1	0.06	0.84	0.12	98
accuracy			0.98	56961
macro avg	0.53	0.91	0.55	56961
weighted avg	1.00	0.98	0.99	56961

Accuracy: 0.9785116132090378
AUC : 0.9077453255892161
Precision: 0.06356589147286822
Recall: 0.8367346938775511
F1_score: 0.11815561959654179



Decision Tree



DecisionTrees's Accuracy is: 99.9350432752234

	precision	recall	f1-score	support
0	1.00	1.00	1.00	56863
1	0.90	0.70	0.79	98
accuracy			1.00	56961
macro avg	0.95	0.85	0.89	56961
weighted avg	1.00	1.00	1.00	56961

Accuracy: 0.999350432752234
AUC : 0.8519704718142819
Precision: 0.8961038961038961
Recall: 0.7040816326530612
F1_score: 0.7885714285714286

5. CONCLUSION AND SCOPE FOR FUTURE ENHANCEMENT

Credit card fraud has indeed become the most crucial issue to be solved. There is a need for vigorous approach in order to expose such crimes. In this paper, we have considered or implemented few machine learning data classufiers like decision tree, Naïve Bayes and Logistic Regression.Few parameters like F1-score, precision, recall and accuracy are considered for performance comparison and based on the comparison, decision tree is found to have an highest accuracy of 99.93% among the Kaggle data sets which has been considered here. The proposed prorotype or model assures an efficient way of detecting fraud with respect to credit card system. In future, we would like to enhance the work by considering few more algorithms togther with different metrics considered.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] https://www.livemint.com/news/india/42-indians-experienced-financial-fraud- In-last-3-years-reveals-survey-11659536104393 .html

[2] Sharayu Pradeep Gulhane and Nitin N "Credit card Fraud Detection using Hidden Markov Model",nternational Journal for modern Trends in Science and Technology, 8(04): 106-109, 2022

[3] Mr. Thirunavukkarasu M, Achutha Nimisha, Adusumilli Jyothsna "Credit Card Fraud detection using Machine Learning", International journal for Computer Science and Mobile computing", vol 10, Issue 4, April – 2021, pg 71-79

[4] M Ramya, S Ajith Kumar, K.Anandh Raja, " Improved Credit Card Fraud Detection using Machine Learning", International Journal of Engineering Research and Technology, ISSN:2278-0181

[5] S P Maniraj, Aditya Saini, Swarna Deep Sarkar, Shadab Ahmed, " Credit Card Fraud Detection using Machine Learning and Data Science", International Journal of Engineering Research and Technology (IJERT),vol 8, Issue 9, Sep- 2019

[6] Masoumeh Zareapoor, Pourya ShamsolMoali, " Application of credit card Fraud detection based on Bagging Ensemble Classifier, International Conference on Intelligent Computing , Communication and Convergence (ICCC-2015)