JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 VOL 07, ISSUE 07, 2020

Supervised Learning Algorithm for Credit Card Fraud Detection

Nusrath Mohammad, Patlannagari Hasitha Reddy

Department of Computer Science and Engineering

Sree Dattha Group of Institutions, Hyderabad, Telangana, India.

Abstract

In this project we mainly focus on credit card fraud detection in real world. Here the credit card fraud detection is based on fraudulent transactions. Generally, credit card fraud activities can happen in both online and offline. But in today's world online fraud transaction activities are increasing day by day. So, to find the online fraud transactions various methods have been used in existing system. In proposed system we use random forest algorithm (RFA) for finding the fraudulent transactions and the accuracy of those transactions. This algorithm is based on supervised learning algorithm where it uses decision trees for classification of the dataset. After classification of dataset a confusion matrix is obtained. The performance of RFA is evaluated based on the confusion matrix.

Keywords: Credit card fraud, supervised learning, random forest algorithm (RFA).

1. Introduction

In the twenty-first century, most financial institutions have increasingly made business facilities available for the public through internet banking. E-payment methods play an imperative role in today's competitive financial society. They have made purchasing goods and services very convenient. Financial institutions often provide customers with cards that make their lives convenient as they go shopping without carrying cash. Other than debit cards the credit cards are also beneficial to consumers because it protects them against purchased goods that might be damaged, lost or even stolen. Customers are required to verify the transaction with the merchant before carrying out any transaction using their credit card.

According to statistics, Visa and Mastercard issued 2287 million total credit cards during 2020 (4th quarter) worldwide (Figs. 1 and 2). Visa issued 1131 million, whereas master card issued 1156 million cards worldwide. These statistics show how the usage of card-based transactions became easy and famous to the end-users. Fraudsters pave their way to manipulate this group of people due to the massive portion of global transactions falling in this category. And perhaps sometimes it is easy to social engineer humans easily.

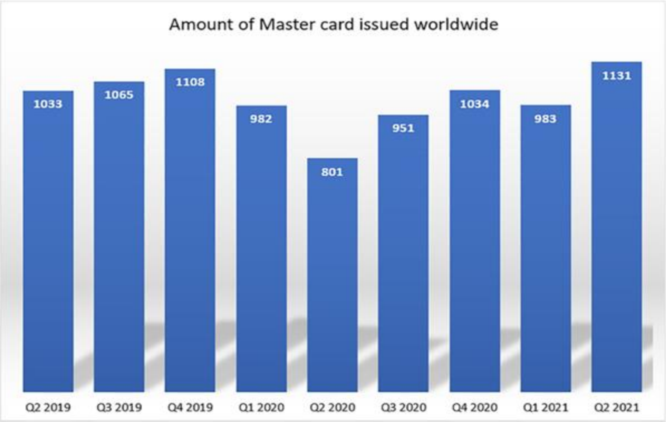


Fig. 1: Amount of Master credit card issued worldwide.

Despite the several benefits that credit cards provide to consumers, they are also associated with problems such as security and fraud. Credit card fraud is considered a challenge which banks and financial institutions are facing. It occurs when unapproved individuals use credit cards for gaining money or property using fraudulent means. Credit card information is sensitive to be stolen via online platforms and web pages that are unsecured. They can also be obtained from identity theft schemes. Fraudsters can access the credit and debit card numbers of users illegitimately without their consent and knowledge.

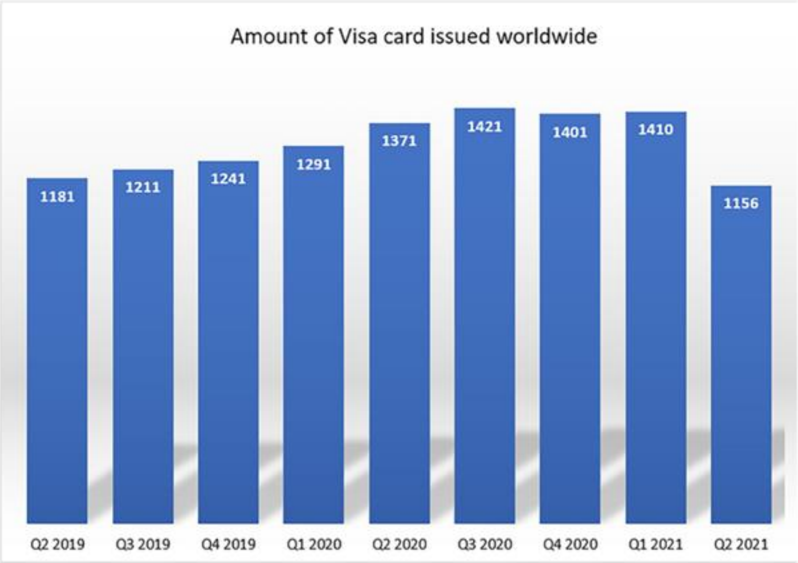


Fig. 2: Amount of Visa credit card issued worldwide.

According to “U.K. finance”, fraudulent activities associated with credit and debit cards have proven to be one of the major causes of financial losses in the finance industry. Due to the advancement of technology, it is big threat that leads to massive loss of finances globally. Therefore, it is imperative to carry out credit card fraud detection to reduce financial losses.

Machine learning is effective in determining which transactions are fraudulent and those that are legitimate. One of the main challenges associated with detection techniques is the barrier to exchanging ideas related to fraud detection. According to a study by “U.K. finance”, the number of credit and debit fraud cases reported in the U.K. worth £574.2 million in 2020.

Problem Statement

CCFD involves quite complex procedures and techniques for developing an effective detection system. Following is some of the problems in CCFD that have been analyzed from the literature review, and it has motivated us to propose an effective solution to the problems. Credit card transactions are substantial in number and are heterogeneous. The users use the credit cards for various purposes based on geographical locations and currencies, which shows that the fraudulent transactions are widely diverse. This problem has motivated me to devise a solution that can potentially help to detect the fraudulent transaction irrespective of geographical location. Fraud detection is also a multi-objective task. Banks and financial institutions need to give their users a good experience and service at all times. Therefore, it is challenging to use the customer datasets for experimental purposes while ensuring service availability and privacy. To compensate this challenge, my motivation leads to introduce the framework of federated learning for data privacy assurance. Fraudulent transaction diversity and imbalanced datasets is also a big challenge in CCFD. Getting real-time datasets of credit card transactions is quite challenging. Banks and financial sectors do not expose their customer's data due to GDPR. Therefore, it creates a challenge for the researchers to gather the datasets for credit card fraud detection. My motivation leads to helping research communities and data scientist who work in the financial sector to devise a system to fulfil the challenges of getting big data for an effective machine learning model.

2. Literature survey

Dornadula et al. designed and developed a novel fraud detection method for Streaming Transaction Data, with an objective, to analyse the past transaction details of the customers and extract the behavioural patterns. Where cardholders are clustered into different groups based on their transaction amount. Then using sliding window strategy, to aggregate the transaction made by the cardholders from different groups so that the behavioural pattern of the groups can be extracted respectively. Later different classifiers are trained over the groups separately. And then the classifier with better rating score can be chosen to be one of the best methods to predict frauds. Thus, followed by a feedback mechanism to solve the problem of concept drift. In this paper, this framework worked with European credit card fraud dataset.

Awoyemi et al. investigated the performance of naïve bayes, k-nearest neighbor and logistic regression on highly skewed credit card fraud data. Dataset of credit card transactions is sourced from European cardholders containing 284,807 transactions. A hybrid technique of under-sampling and oversampling is carried out on the skewed data. The three techniques are applied on the raw and pre-processed data. The work is implemented in Python. The performance of the techniques is evaluated based on accuracy, sensitivity, specificity, precision, Matthew’s correlation coefficient and balanced classification rate.

Sulaiman et al. conducted a comparative analysis of the literature review considering the ML techniques for credit card fraud detection (CCFD) and data confidentiality. In the end, this framework have proposed a hybrid solution, using the neural network (ANN) in a federated learning framework. It has been observed as an effective solution for achieving higher accuracy in CCFD while ensuring privacy.

Maniraj et al. illustrated the modelling of a data set using machine learning with Credit Card Fraud Detection. The Credit Card Fraud Detection Problem included modelling past credit card transactions with the data of the ones that turned out to be fraud. This model is then used to recognize whether a new transaction is fraudulent or not. this objective here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications. Credit Card Fraud Detection is a typical sample of classification. In this process, this framework has focused on analysing and pre-processing data sets as well as the deployment of multiple anomaly detection algorithms such as Local Outlier Factor and Isolation Forest algorithm on the PCA transformed Credit Card Transaction data.

Suryanarayana et al. utilized the logistic regression, based machine learning to detect credit card fraud. The results showed logistic regression-based approaches outperforms with the highest accuracy and it can be effectively used for fraud investigators.

Lakshmi et al. investigated the performance of logistic regression, decision tree and random forest for credit card fraud detection. Dataset of credit card transactions is collected from kaggle and it contains a total of 2,84,808 credit card transactions of a European bank data set. It considered fraud transactions as the “positive class” and genuine ones as the “negative class”. The data set is highly imbalanced, it has about 0.172% of fraud transactions and the rest are genuine transactions. The author has been done oversampling to balance the data set, which resulted in 60% of fraud transactions and 40% genuine ones. The three techniques are applied for the dataset and work is implemented in R language. The performance of the techniques is evaluated for different variables based on sensitivity, specificity, accuracy, and error rate.

Marabad et al. examined and overviewed the performance of K-nearest neighbors, Decision Tree, Logistic regression, and Random Forest, XGBoost for credit card fraud detection. The assignment is implemented in Python and uses five distinct machine learning classification techniques. The performance of the algorithm is evaluated by accuracy score, confusion matrix, f1-score, precision and recall score and auc-roc curve as well.

Zioviris et al. proposed the use of two autoencoders to perform feature selection and learn the latent data space representation based on a nonlinear optimization model. On the delivered significant features, this framework subsequently applies a deep convolutional neural network to detect frauds, thus combining two different processing blocks. The adopted combination has the goal of detecting frauds over the exposed latent data representation and not over the initial data.

Makki et al. revealed that the existing approaches result in a large number of false alarms, which are costly to financial institutions. This may lead to inaccurate detection as well as increasing the occurrence of fraud cases.

Carcillo et al. presented a hybrid technique that combined supervised and unsupervised techniques to improve the fraud detection accuracy. Unsupervised outlier scores, computed at different levels of granularity, are compared, and tested on a real, annotated, credit card fraud detection dataset. Experimental results showed that the combination is efficient and does indeed improve the accuracy of the detection.

3. Proposed system

In this project we mainly focus on credit card fraud detection in real world. Here the credit card fraud detection is based on fraudulent transactions. Generally, credit card fraud activities can happen in both online and offline. But in today's world online fraud transaction activities are increasing day by day. So, to find the online fraud transactions various methods have been used in existing system. In proposed system we use random forest algorithm (RFA) for finding the fraudulent transactions and the accuracy of those transactions. This algorithm is based on supervised learning algorithm where it uses decision trees for classification of the dataset.

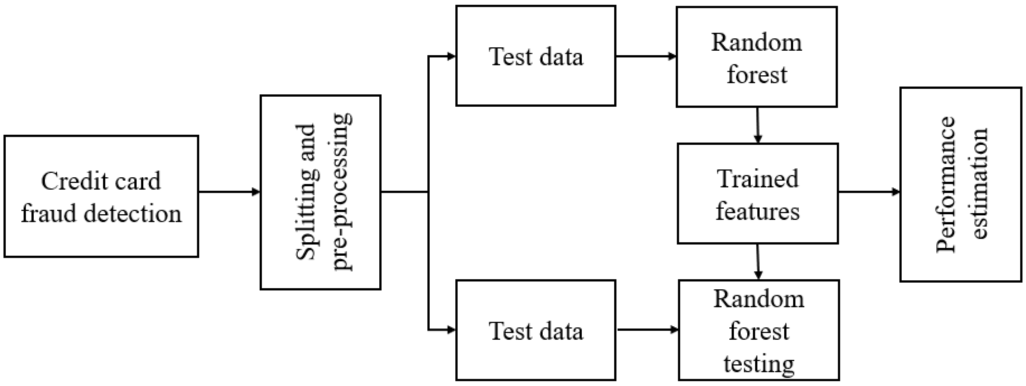


Fig. 3: Block diagram of proposed system.

Credit card fraud detection dataset

31-Columns: Time, V1, V2, V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20, V21, V22, V23, V24, V25, V26, V28, Amount, Class.

284808- Rows

Pre-processing

Data Pre-processing in Machine learning

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data pre-processing task.

Why do we need Data Pre-processing?

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

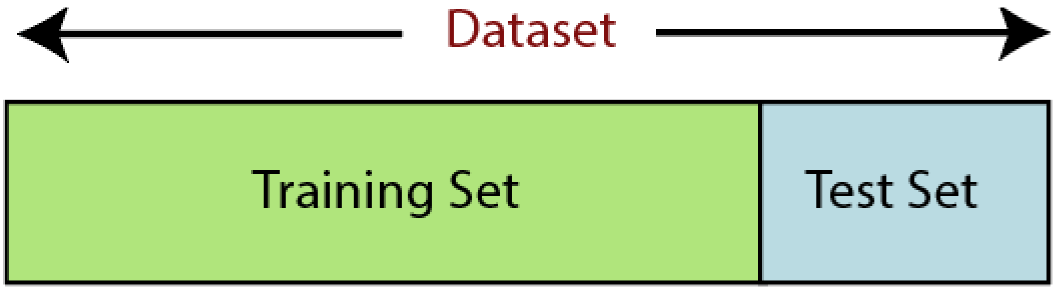
- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Encoding Categorical Data
- Splitting dataset into training and test set
- Feature scaling

Splitting the Dataset into the Training set and Test set

In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model.

Supposeif we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:



Training Set: A subset of dataset to train the machine learning model, and we already know the output.

Test set: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

Random Forest Algorithm

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

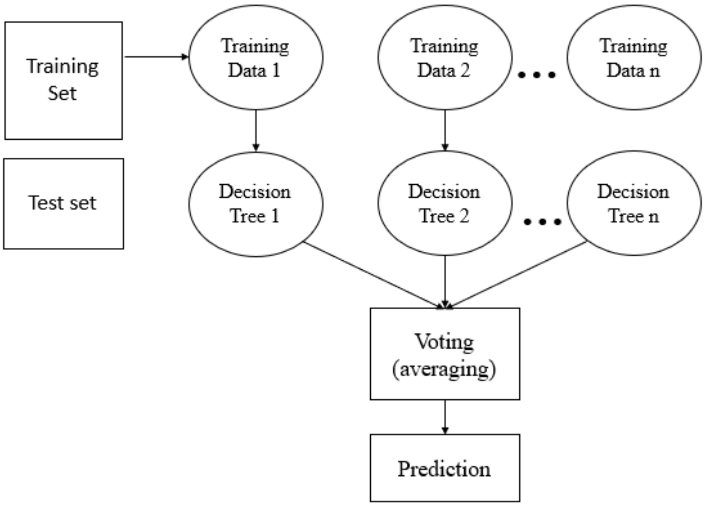


Fig. 4: Random Forest algorithm.

Step 1: In Random Forest n number of random records are taken from the data set having k number of records.

Step 2: Individual decision trees are constructed for each sample.

Step 3: Each decision tree will generate an output.

Step 4: Final output is considered based on Majority Voting or Averaging for Classification and regression respectively.

Important Features of Random Forest

- **Diversity-** Not all attributes/variables/features are considered while making an individual tree, each tree is different.
- **Immune to the curse of dimensionality-** Since each tree does not consider all the features, the feature space is reduced.
- **Parallelization-**Each tree is created independently out of different data and attributes. This means that we can make full use of the CPU to build random forests.
- **Train-Test split-** In a random forest we don’t have to segregate the data for train and test as there will always be 30% of the data which is not seen by the decision tree.

- **Stability-** Stability arises because the result is based on majority voting/ averaging.

Assumptions for Random Forest

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random Forest classifier:

- There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.
- The predictions from each tree must have very low correlations.

Below are some points that explain why we should use the Random Forest algorithm

- It takes less training time as compared to other algorithms.
- It predicts output with high accuracy, even for the large dataset it runs efficiently.
- It can also maintain accuracy when a large proportion of data is missing.

Types of Ensembles

Before understanding the working of the random forest, we must investigate the ensemble technique. Ensemble simply means combining multiple models. Thus, a collection of models is used to make predictions rather than an individual model. Ensemble uses two types of methods:

Bagging– It creates a different training subset from sample training data with replacement & the final output is based on majority voting. For example, Random Forest. Bagging, also known as Bootstrap Aggregation, is the ensemble technique used by random forest. Bagging chooses a random sample from the data set. Hence each model is generated from the samples (Bootstrap Samples) provided by the Original Data with replacement known as row sampling. This step of row sampling with replacement is called bootstrap. Now each model is trained independently which generates results. The final output is based on majority voting after combining the results of all models. This step which involves combining all the results and generating output based on majority voting is known as aggregation.

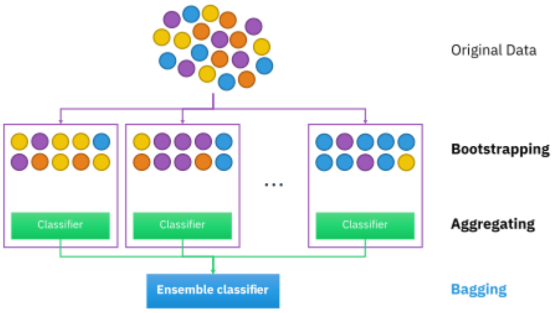


Fig. 5: RF Classifier analysis.

Boosting– It combines weak learners into strong learners by creating sequential models such that the final model has the highest accuracy. For example, ADA BOOST, XG BOOST.

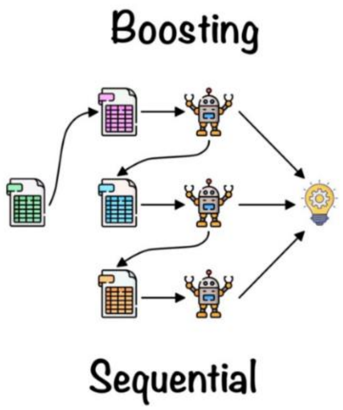


Fig. 6: Boosting RF Classifier.

Advantages of proposed system

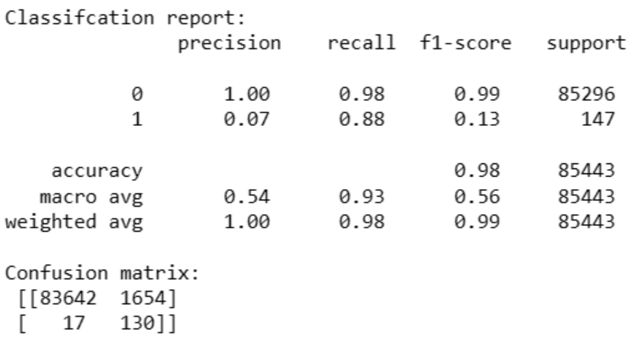
- It can be used in classification and regression problems.
- It solves the problem of overfitting as output is based on majority voting or averaging.
- It performs well even if the data contains null/missing values.
- Each decision tree created is independent of the other thus it shows the property of parallelization.
- It is highly stable as the average answers given by many trees are taken.
- It maintains diversity as all the attributes are not considered while making each decision tree though it is not true in all cases.
- It is immune to the curse of dimensionality. Since each tree does not consider all the attributes, feature space is reduced.

4. Results

Module implementation

- Credit card fraud detection
- Splitting and pre-processing
- Test data
- Random forest
- Trained features
- Random forest testing
- Performance estimation

Under sampling technique



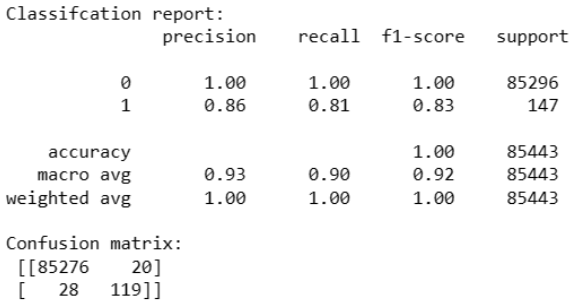
```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99     85296
           1       0.07      0.88      0.13       147

    accuracy                           0.98     85443
   macro avg       0.54      0.93      0.56     85443
weighted avg       1.00      0.98      0.99     85443

Confusion matrix:
 [[83642  1654]
 [   17   130]]
```

Over sampling technique



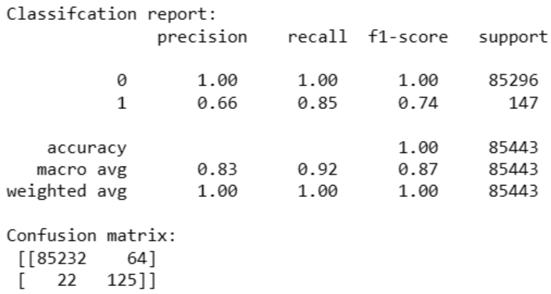
```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85296
           1       0.86      0.81      0.83       147

    accuracy                           1.00     85443
   macro avg       0.93      0.90      0.92     85443
weighted avg       1.00      1.00      1.00     85443

Confusion matrix:
 [[85276    20]
 [   28   119]]
```

SMOTE



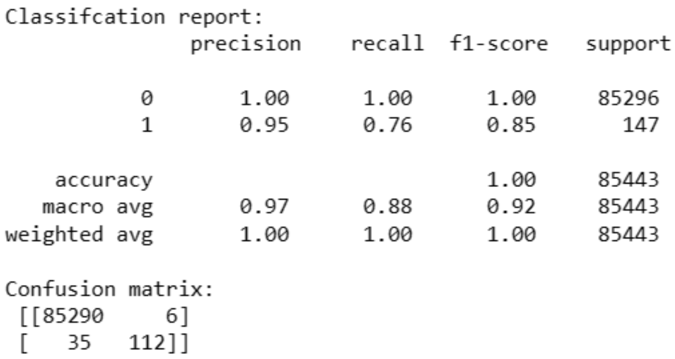
```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85296
           1       0.66      0.85      0.74       147

    accuracy                           1.00     85443
   macro avg       0.83      0.92      0.87     85443
weighted avg       1.00      1.00      1.00     85443

Confusion matrix:
 [[85232    64]
 [   22   125]]
```

Random forest

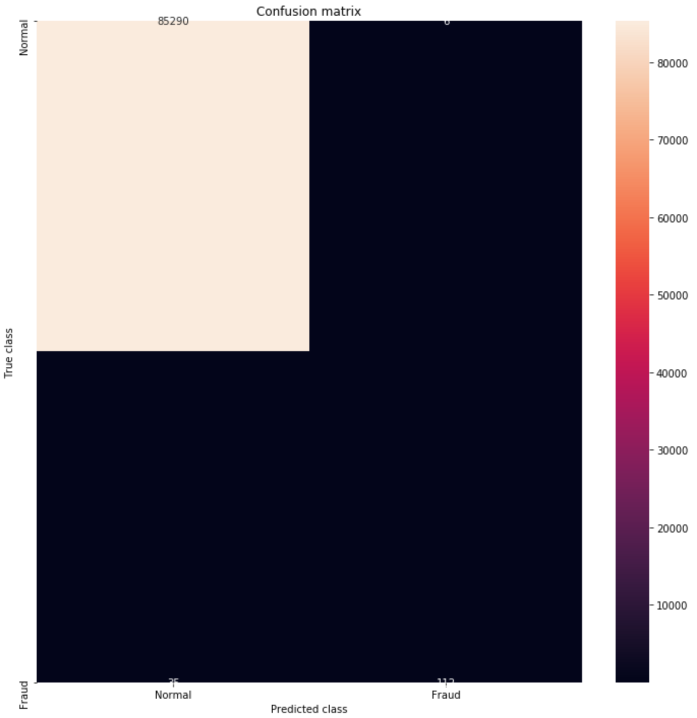


```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85296
           1       0.95      0.76      0.85       147

    accuracy                           1.00     85443
   macro avg       0.97      0.88      0.92     85443
weighted avg       1.00      1.00      1.00     85443

Confusion matrix:
 [[85290     6]
 [   35   112]]
```



5. Conclusion

This work has examined the performance of two kinds of random forest models. A real-life B2C dataset on credit card transactions is used in our experiment. Although random forest obtains good results on small set data, there are still some problems such as imbalanced data. Our future work will focus on solving these problems. The algorithm of random forest itself should be improved. For example, the voting mechanism assumes that each of base classifiers has equal weight, but some of them may be more important than others. Therefore, we also try to make some improvement for this algorithm.

5.1 Future scope

CNN is a deep learning method heavily associated with spatial data such as image processing data. Like ANN, CNN has the same hidden layer structure in addition to special convolution layers with a different number of channels in each layer. The word convolution is linked with the idea of moving filters that capture the key information from the data. CNN is widely used in image processing as it automatically performs the feature reduction which makes it less prone to overfitting and thus training CNN does not require heavy data pre-processing. The role of using CNN for image processing is to minimize the processing by reducing the image without losing key features to make predictions. The key terms in CNN are feature maps, channels, pooling, stride, and padding. In comparison to the popular Random Forest, CNN are not fully connected in layer-to-layer connection and unlike MLP that has different weights associated with each node, CNN has constant weight parameter for each filter and these two features reduce the number of parameters in a CNN model. Also, the pooling method improves the feature detection process making it more robust to size and position changes of an element in an image. CNN models are conventionally used for image and video processing that has two-dimensional data as input and therefore named as 2DCNN. The feature mapping process is used to learn the internal representation from the input data and the same procedure can be used for one-dimensional data as well where the location of features is not relevant. A very popular example of 1DCNN application is in Natural Language Processing which is a sequence classification problem. In

1DCNN, the kernel filter moves top to bottom in a sequence of a data sample instead of moving left to right and top to bottom in 2DCNN.

References

[1] V. N. Dornadula, S Geetha, Credit Card Fraud Detection using Machine Learning Algorithms, Procedia Computer Science, Volume 165, 2019, Pages 631-641, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.01.057.

[2] J. O. Awoyemi, A. O. Adetunmbi and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," 2017 International Conference on Computing Networking and Informatics (ICCNI), 2017, pp. 1-9, doi: 10.1109/ICCNI.2017.8123782.

[3] B. Sulaiman, R., Schetinin, V. & Sant, P. Review of Machine Learning Approach on Credit Card Fraud Detection. Hum-Cent Intell Syst 2, 55–68 (2022). https://doi.org/10.1007/s44230-022-00004-0.

[4] S P, Maniraj & Saini, Aditya & Ahmed, Shadab & Sarkar, Swarna. (2019). Credit Card Fraud Detection using Machine Learning and Data Science. International Journal of Engineering Research and. 08. 10.17577/IJERTV8IS090031.

[5] Suryanarayana, S & Gn, Balaji & Venkateswara Rao, Gurrala. (2018). Machine Learning Approaches for Credit Card Fraud Detection. International Journal of Engineering and Technology (UAE). 7. 10.14419/ijet. v7i2.9356.

[6] Lakshmi S V S S, Selvani Deepthi Kavila, Machine Learning for Credit Card Fraud Detection System, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 24 (2018) pp. 16819-16824 © Research India Publications. http://www.ripublication.com

[7] S. Marabad, Credit Card Fraud Detection Using Machine Learning, Vol 7 No 2 (2021): Volume 7 Issue Ii, DOI https://doi.org/10.33130/AJCT.2021v07i02.023.

[8] G. Zioviris, K. Kolomvatsos, and G. Stamoulis. 2022. Credit card fraud detection using a deep learning multistage model. J. Supercomput. 78, 12 (Aug 2022), 14571–14596. https://doi.org/10.1007/s11227-022-04465-9

[9] S. Makki, Z. Assaghir, Y. Taher, R. Haque, M. -S. Hacid and H. Zeineddine, "An Experimental Study with Imbalanced Classification Approaches for Credit Card Fraud Detection," in IEEE Access, vol. 7, pp. 93010-93022, 2019, doi: 10.1109/ACCESS.2019.2927266.

[10] F. Carcillo, Y. Borgne, O. Caelen, Y. Kessaci, F. Oblé, G. Bontempi, Combining unsupervised and supervised learning in credit card fraud detection, Information Sciences, Volume 557, 2021, Pages 317-331, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2019.05.042.