Journal of Ubiquitous Systems & Pervasive Networks
Volume 12, No. 1 (2020) pp. 25-30

Various Methods for Fraud Transaction Detection in Credit Cards

Hardik Manek[a], Nikhil Kataria[b], Sujai Jain[c], Chitra Bhole[d]

[a]*Acadia University, Wolfville, Nova Scotia, Canada, NS B4P 2R6*
[b,c,d]*K.J.S.I.E.I.T, Mumbai, India, 400022*

Abstract

Cashless payments have become effortless with the advent of new technology and the internet. But, for online transactions, you don't have to be in a certain place where the transaction occurs, making it vulnerable to fraudulent attacks. A cyber-attacker can pretend to be the owner of a credit card and make a fraudulent transaction. There are several techniques to determine the nature of the transaction, for instance, by comparing the current transaction with previous transactions. If the monetary difference between current transaction and previous transaction is too large, then there is a greater probability of current transaction being a fraudulent transaction. This method is not reliable for anomaly detection. In some countries like India and China, banks deploy a two-step verification process which strengthens the security of the transaction. While in other countries, employees in the bank manually segregate the transactions to be fraud or not. These methods are highly dependent on human intervention. Machine Learning can be utilized to automate the process of anomaly detection. Supervised algorithms such as Logistic Regression can be used to build a model that will predict the output in the form of binary classes i.e. 0 for a valid transaction and 1 for a fraudulent transaction. Autoencoder Neural Network is one of the unsupervised algorithms using which better accuracy can be obtained for anomaly detection. In this paper, we explain different machine learning algorithms viz; Hidden Markov Model, Artificial Neural Network, and Convolutional Neural Network. Moreover, Logistic Regression is implemented, and the results obtained are highlighted.

Keywords: *Credit Card Fraud, Neural Network, Cashless Transaction*

1. Introduction

Recently, the payment card industry has grown exponentially. Independent of location, customers can do shopping using smart devices. Hence e-commerce has led to advancement in terms of efficiency, accessibility, and competition, but it also has some disadvantages. Evolution is accompanied by greater vulnerability to threats. The problem of doing business through the Internet lies in the fact that neither the card nor the cardholder must be present at the point of sale. So, it is impossible for the trader to check if the customer is the actual holder of the card or not. Financial institutions have attention focused on recent computational methods for managing the problem of credit card fraud. Legitimate and fraudulent will be the two categories used to classify the transactions. This process of sorting will be done based on the card holder's spending behavior. Different techniques are compared in this paper. Multiple techniques [1] have been applied for credit card fraud detection such as artificial neural network, genetic algorithm, support vector machine, frequent itemset mining, decision tree, migrating Birds optimization algorithm. Bayesian performance and the neural network. Decision tree, neural networks, and logistic regression have demonstrated its applicability in fraud detection.

2. Related Work

Ghosh and Reilly [3] used three-layer neural networks to detect fraud in 1994. The neural network was trained on examples of fraud containing stolen cards, application fraud, counterfeit fraud, non-received fraud problems (NRIs) and orders for post fraud.

Abhinav and Amlan [4] proposed a hidden Markov model to detect credit card fraud. The proposed model does not require fraudulent signatures and can still detect frauds considering the cardholder's spending habits.

Y. Sahin and E. Duman [5] proposed an approach to identify credit card fraud using the decision tree and the Support Vector Machine. The different methods of tree decision- making

* Corresponding author. Tel.: +1234567890
Fax: +9876543210; E-mail: hardikmanek@acadiau.ca

DOI: 10.5383/JUSPN.12.01.004

performance intensifier classification models (C5.0, C and RT and CHAID) and several other SVM methods (SVM with polynomial, sigmoidal, linear and RBF kernels) are compared in this study.
Fuzzy clustering and neural network are the methods proposed [6] for fraud detection in banking transactions. In this approach, fraud detection is performed in three phases. The first step is initial user authentication and verification of card details. After completion of this operation, a blurry half clustering algorithm is performed to discover the behavior of normal user usage based on past transactions. If it turns out that a new transaction is uncertain at this stage, based on a neural network to determine whether it was, in fact, a fraudulent transaction or the mechanism applies.
Convolutional based neural network approach (CNN) [7] is proposed by Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang. A convolutional neural network is a part of deep learning and is a type of advanced neural network composed of more then one hidden layer. In this paper, to find more complex fraud models and improve the accuracy of classification, a new feature of commercial entropy is proposed. In this paper for the first time, CNN is used to detect fraud.

Different outlier techniques [8] can also use to differentiate fraudulent transaction as outlier data

The description of the various techniques are as follows: -
1. Credit Card Fraud Detection with a Neural Network by Sushmito Ghosh (IEEE 1994) [3] with the techniques, feed forward artificial Neural Network and data sets from Mallon bank 450000 transactions to train model.
2. In one of the approaches [11], Bayesian Neural Networks is implemented, and the dataset is obtained from Euro pay International. Credit Card Fraud Detection using Bayesian and Neural Networks by Sam Maes (International Naiso Congress on Neuro-Fuzzy Technology, 2002)
3. Hidden Markov Model and data sets from completely simulated and simplified data. Credit Card Fraud Detection using Hidden Markov model [4] by Abhinav Srivastava (IEEE Dep and Sec Comp, 2002).
4. Detecting Credit Card by Decision Trees and Support Vector Machines [5] by Y. Sahin (Proc Int. MultiConf of Eng and Comp Sci 2011) with the techniques Decision Tree (C5.0, C and RT and CHAID) SVM (polynomial sigmoid, linear and RBF kernel functions) and data sets from National bank Credit card data warehouse 978 fraud, 22 million normal transactions.
5. Credit Card Fraud Detection using Convolutional Neural Network [6] by Tanmay kumar Behera (IEEE Computer Society, 2015) with the technique, Fuzzy Clustering and Neural Network using Synthetic data.
6. Credit Card Fraud Detection using Convolutional Neural Network by Kang Fu [7] (Springer, 2016 Convolutional Neural Network, Cost-Based Sampling for imbalance data) with the technique Commercial Bank and data sets from 260 million transactions and 4000 fraud.

3. Credit Card Fraud Detection Problems

The main issue with creating a Mastercard fraud detection system is getting information for coaching. It is difficult to induce real information as a result of this type of information is sensitive and personal. In several techniques [7],[3],[5],[11], the researches have trained with real-world information by arrival with banks. But otherwise, synthetic information is often generated and is accessible for coaching.
Second issue is to contend with the distinction between varieties the amount the quantity of legitimate and therefore the number of deceitful transactions. Synthetic minoring over-sampling ways square measure accustomed increase range of low incidence information in information set that generate artificial deceitful transactions connected with original dataset. In [7], value primarily based sampling is employed to get artificial deceitful transactions to balance information set.
Overlapping of information is a new drawback as several dealings seem like deceitful transaction, once truly they're legitimate transactions it's conjointly doable that deceitful transactions seem to be traditional transactions.

4. Fraud Detection Techniques

4.1 Hidden Markov Model

A hidden Markov model of science model (HMM) is also a math's Markov model inside that the system being sculptural is assumed to be a Markov chain with hidden states academic degree. HMM is also a double embedded likelihood distribution technique with hierarchy levels. Fraud detection Approach victimization HMM is projected. They need thought-about three price ranges low, medium and high asset of potential observation, as an example, let l= (0,200USD], m= (USD250, USD600], h= (USD 700, credit card limit). If a user makes a bunch of action of USD 400, then resultant observation image is medium. Every human action amount generally depends on the equivalent kind of purchase. The set of all potential sorts of purchase and also the set of all potential lines of business of merchants forms the set of hidden states of the HMM. The projected approach in [7], Hidden Andre Markoff Model (HMM) - based master card FDS does not require fraud signatures and still it can detect frauds by considering a user's spending pattern. Different entities such as hidden states, observable states, and transition probabilities are shown in Fig. 1.

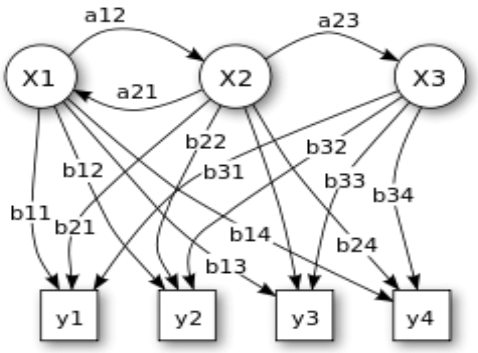

Fig. 1. Hidden Markov Model

4.2 Artificial Neural Network

Artificial Neural Network is one of the most popular and powerful unsupervised anomaly classifiers. ANN performs similar to the human brain. ANN has various layers where the first layer is the input layer and the last layer is the output layer. It ought to have type of hidden layer or no hidden layer. If Neural network embrace quite one hidden layer, then it's deep learning each layer has completely totally different neurons, and every somatic cell is connected with weighted edges. Output of each somatic cell could also be a performance of its unit. This performance is called activation perform. Example of varied activation functions used square measure

sigmoid performs, step performs, function, linear performs etc. Fig. 2. Depicts the different layers included in the architecture of Artificial Neural Network.

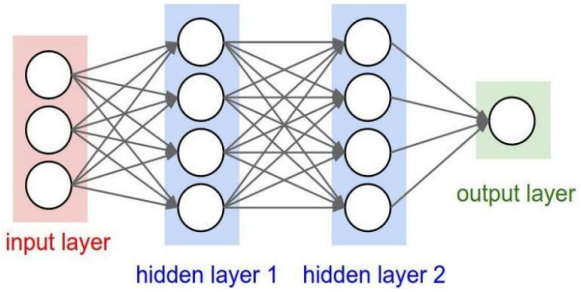

Fig. 2. Architecture of Artificial Neural Network

Mostly used to perform is Sigmoid perform among all output layer has the same type of neurons as classification label, each vegetative cell of output layer offers likelihood of being that category. In the figure, four neurons square measure in input layer that creates five picks regarding to importance of input options. Neurons of second layer connected to output layers neurons. Neural network makes an alternative of weights on edges from data given there to for employment and regulate weights.

4.3 Convolutional Neural Network

Convolutional Neural Network (CNN) is also a section of deep learning. Mapping of input into hidden layer represents one feature map, each feature map represents one characteristic method of press neurons into feature map is termed convolution as shown in the figure below. Sub-sampling reduces parameters of feature map fully connected layer is same as neural network [11].

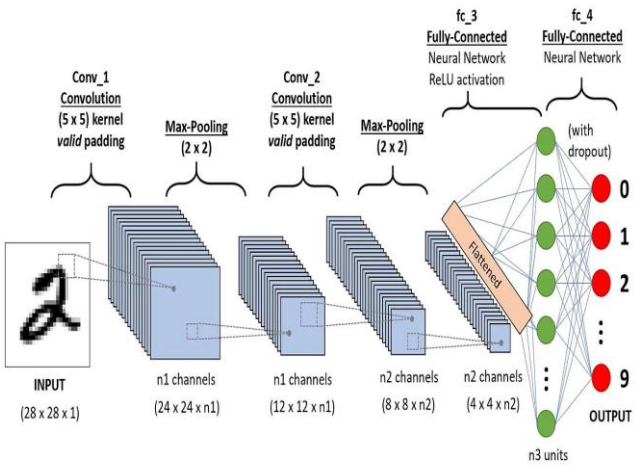

Fig. 3. Convolutional Neural Network

CNN is with success applied in face recognition, character recognition, image classification, etc. Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang [7] projected a convolutional neural network primarily based approach to find fallacious transactions in MasterCard. Input options square measure reworked into feature matrices so reborn into pictures. For finding additional complicated fraud patterns and to enhance classification accuracy, replacement feature commerce entropy is projected to alleviate the matter of the unbalanced dataset; they used price primarily based sampling technique to get completely different range of artificial frauds to coach the model. They applied CNN model as a result of it's appropriateness for coaching giant size of information and CNN has mechanism to avoid over fitting. Fig.3. shows how a model is created using Convolutional Neural Network. Input image is 2 from EMNIST dataset which is passed through various layers to create a model which can classify digits.

5. System Design and Dataset

The data has been extracted from Kaggle.com. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.[11]

There is a total of 30 columns out of which 2 features have been extracted using Principal Component Analysis (PCA). These features are considered for training the model. The system design follows two approach viz. Logistic Regression and Autoencoder neural network.

Fig. 4. Depicts the flow of system. In our system we had used two approaches viz; Logistic Regression and Autoencoder Neural Network. Finally, accuracy obtained from both the methods is compared.

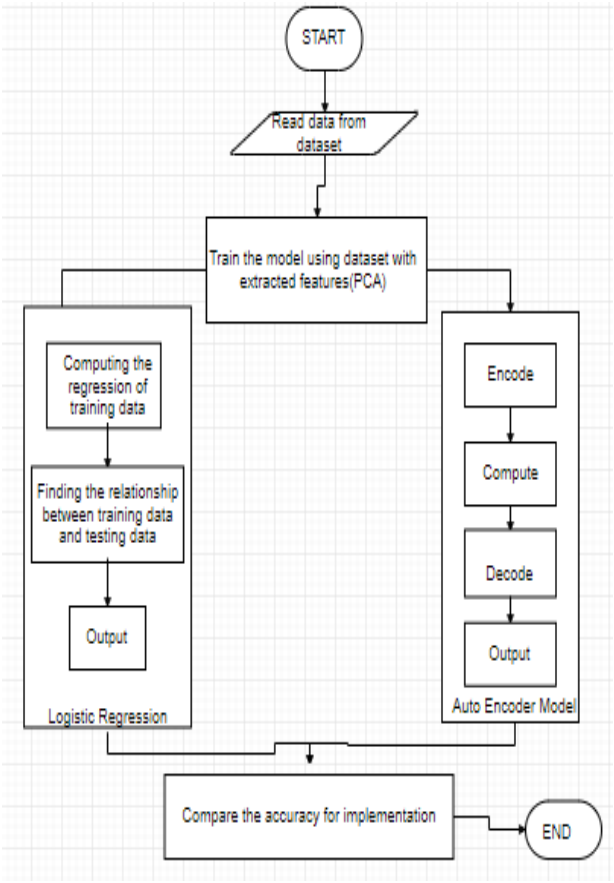

Fig. 4. System Design

6. Machine Learning Techniques

Table 1 describes and compares different machine learning techniques. Content of this table allows us to narrow down the selection process of algorithm for our problem statement.

TABLE 1
Machine Learning Algorithms

Method	Advantage	Limitations
Linear Regression	Linear models can be updated with new data easily using stochastic gradient descent	Linear Regression performs poorly when there are non-linear relationships.
K-means	K-means is undoubtedly most popular clustering algorithm because of its flexible nature. That means we can fine tune the parameters easily if we pre-process data.	Details about number of clusters, needed to be specified beforehand.
Naïve Bayes	Even though Suffers from conditional dependencies it's easy to implement and scale with dataset.	Naïve Bayes is very simple algorithm and hence model created using Naïve Bayes are beaten by models of other complex algorithms
Support Vector Machines	SVM's can model non-linear decision boundaries, and there are many kernels to choose from.	SVM's are memory intensive, trickier to tune due to the importance of picking the right kernel, and don't scale well to larger datasets.
Logistic Regression	Outputs have nice probabilistic interpretation, and the algorithm can be regularized to avoid overfitting	Tends to underperform when there are multiple or non-linear decision boundaries
Hidden Markov Model	Can handle complex data and has ability to learn	Very slow to train and requires a lot of power
Artificial Neural Networks	Can handle large data	Expensive
Convolutional Neural Network	Less training time	Avoid Model Over fitting

7. Proposed System

7.1 Autoencoder Neural Network

Autoencoders are a type of Neural Network which is used to learn data coding in an unsupervised manner. The main functionality of the encoder mode is to find an appropriate method to encode the data such that decoded data will be close to the input data. Autoencoder model consists of 3 main layers which are the input layer, an output layer and one or more hidden layer connecting them, although the number of nodes in the output layer are same as that of input layer. Fig.5. shows the architecture of Autoencoder Neural Network where the encoding is performed from input layer till hidden layer and from hidden layer to output layer decoding is performed.

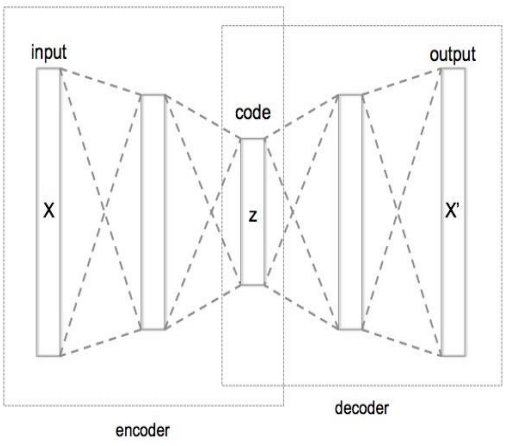

Fig. 5. Autoencoder Neural Network

Figure Structure of an Autoencoder Model with 3 connected hidden layers which encodes input x and fives output x(Z)f W, b(x) = x The Reconstruction error is minimized by using traditional squared error given by: $L(x, x) = ||xx||2$.

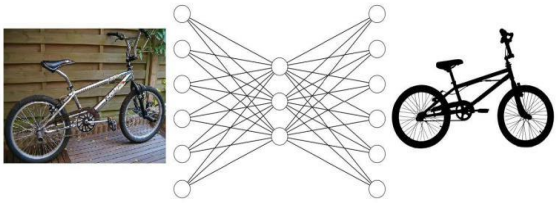

Fig. 6. Demonstration of Autoencoder Model

A Simple demonstration of Autoencoder model is given in above Fig. 5. in which first part that is input layer and hidden layer encodes a bicycle and subsequently, the 2nd part i.e. outer layer and hidden layer decodes to achieve similar bicycle as the output. Yimin Yang, Q.M.Jonathan Wu, Yaonan Wang [10] have proposed Autoencoder model for dimension reduction and image reconstruction.

7.2 Logistic Regression

Logistic Regression is the most renowned machine learning statistical model after Linear Regression. From multiple points of view, Linear Regression and Logistic Regression are comparative. The greatest contrast lies in what they are utilized for, Linear Regression calculations are utilized to anticipate continuous variables however Logistic Regression is utilized for binary classification.

There are numerous arrangement assignments done routinely by individuals. For instance, arranging whether an email is a spam or not, characterizing whether a tumor is harmful or

kindhearted, ordering whether a site is fake or not, and so on. These are run of the mill precedents where machine learning calculations can make our lives significantly simpler. An extremely basic, basic and valuable calculation for characterization is the Logistic Regression calculation.

Sigmoid Function (Logistic Function) Calculated relapse calculation likewise utilizes a direct condition with free indicators to anticipate an esteem. The anticipated esteem can be anyplace between negative interminability to positive vastness. We require the yield of the calculation to be class variable, i.e. 0-no, 1-yes. Along these lines, we are squashing the yield of the straight condition into a scope of [0,1]. To squash the anticipated an incentive somewhere in therangeof0and1, we utilize the sigmoid capacity.

$$z = \theta_0 + \theta_1.x_1 + \theta_2.x_2 + ... \quad (1)$$

$$h = g(z) = 1/(1 + e^{-z}) \quad (2)$$

We take the output (z) of the straight condition and provide for the capacity g(x) which restores a squashed esteem h, the esteem h will lie in the scope of 0 to 1. To see how sigmoid capacity squashes the qualities inside the range, how about we imagine the chart of the sigmoid capacity. Fig. 7. Represents the S-shaped curve known as logistic curve. The curve ranges from 0 to 1 and this value is calculated using equation 2.

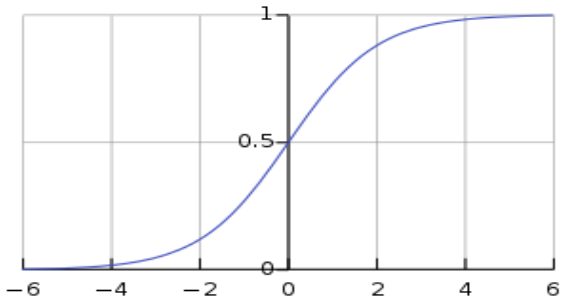

Fig. 7. Logistic Regression Graph

7.2.1 Implementation using Logistic Regression

The results of training are shown in Fig. 9. Four parameters precision, recall, f1-score, and support are as follows:
Precision = True Positive/ (True Positive + False Positive).
Recall = True Positive/ (True Positive + False Negative).
F1-Score = Weighted average of precision and recall.
The values of precision, recall, and f1-score are obtained from confusion matrix. The structure of confusion matrix is shown in Fig. 8.

	Predicted class		
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Fig. 8. Confusion Matrix

```
PS C:\Users\jansu\Desktop\fraud> python script.py
C:\Python\Python37-32\lib\site-packages\sklearn\externals\joblib\externals\cloudpickle\clo
udpickle.py:47: DeprecationWarning: the imp module is deprecated in favour of importlib; s
ee the module's documentation for alternative uses
  import imp
C:\Python\Python37-32\lib\site-packages\sklearn\linear_model\logistic.py:432: FutureWarnin
g: Default solver will be changed to 'lbfgs' in 0.22. Specify a solver to silence this war
ning.
  FutureWarning)
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     56859
           1       0.58      0.18      0.28       103

   micro avg       1.00      1.00      1.00     56962
   macro avg       0.79      0.59      0.64     56962
weighted avg       1.00      1.00      1.00     56962
```

Fig. 9. Precision, Recall, f1-score, and support achieved using Logistic Regression model.

Fig. 10. The outlier distinguishes the transactions into legitimate and non-legitimate. The clusters formed at bottom left corner depicts the non-fraudulent transactions and data points at the far end highlights the potential fraudulent transactions.

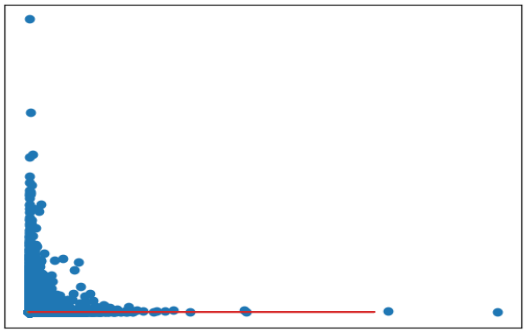

Fig. 10. Graphical plot of result

8. Conclusion and Future Work

In this paper, we have compared various Machine Learning models for fraud detection in the banking transactions. According to our research Autoencoder model gives suitable results, even though other methods can also be employed for fraud detection.

In recent times Machine Learning models are created in the computation nodes situated at data centers of the companies. This process can be considered as Centralized system. By the invention of Federated Learning, machine learning models can be trained in the local device i.e. in a decentralized manner. In Federated Learning first proxy dataset is used to create a model after that the created model is sent to the participating local devices such as cell phones. The local data is applied to this pre-trained model to create a new and improved model. Eventually, after the new model is created by all the participating local devices these models are aggregated. This process is repeated in the form of epochs and finally, an ideal model is obtained.

8.1 Implementation Idea

Google has been developing a Tensorflow Federated library to introduce federated learning in various problems. Gboard is the Android keyboard which is implemented using Federated Learning. In our problem statement of fraudulent transaction detection, federated learning can help to build the model using real-time data available in local devices without compromising privacy.

8.1.1 TensorFlow Federated Operations

1. tff.federated_broadcast(value): This method is used to send the information from the engineer's (server) side to the local devices (client).

Args:

value: A value of a TFF federated type placed at the tff.SERVER, all members of which are equal (the tff.FederatedType.all_equal property of value is True).

Returns: A value is broadcasted at the client-side and further comparison can be done.

2. tff.federated_map(mapping_fn, value): Used to perform computation in local devices, such as comparing the values sent by the engineers with local values.

Args:

mapping_fn: A mapping function to apply pointwise to member constituents of value on each of the participants in tff.CLIENTS. The parameter of this function must be of the same type as the member constituents of value.

value: At the tff.CLIENTS a value of a TFF federated type is placed, or a value that can be implicitly converted into a TFF federated type, e.g., by zipping.

Returns: Values at the client-side and broadcasted value are compared and the mapped result is returned.

3. tff.federated_mean(value, weight=none): Used to calculate the average value from all the local devices.

Args:

value: The value of which the mean is to be computed. Must be of a TFF federated type placed at tff.CLIENTS. The value may be structured, e.g., its member constituents can be named tuples. The tensor types that the value is composed of must be floating-point or complex.

weight: An optional weight, a TFF federated integer or floating-point tensor value, also placed at tff.CLIENTS.

Returns: Mean is calculated, and the value is returned to the server's side using the zero-sum cryptography method.

References

[1] Credit card fraud detection using Machine Learning Techniques, John O. Awoyemi, Adebayo O. Adetunmbi, Samuel A.Oluwadare, Akure, Nigeria.

[2] Alejandro Correa Bahnsen, Djamila Aouada, Aleksandar Stojanovic, Bjurn Ottersten, Feature engineering strategies for credit card fraud detection, 0957-4174/ 2016 Elsevier. https://doi.org/10.1016/j.eswa.2015.12.030

[3] Ghosh, S., Reilly, D.L.: Credit card fraud detection with neural network. In Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences, 1994, vol. 3, IEEE (1994). https://doi.org/10.1109/HICSS.1994.323314

[4] Abhinav Srivastava, Amlan Kundu, Shamik Sural, Senior Member, IEEE, and Arun K. Majumdar, Senior Member, IEE, Credit Card Fraud Detection Using Hidden Markov Model, IEEE transactions on dependable and secure computing, vol. 5, no. 1, January-March 2008. https://doi.org/10.1109/TDSC.2007.70228

[5] Y. Sahin and E. Duman, Detecting Credit Card Fraud by Decision Trees and Support Vector Machines, IMECS vol 1, 2011. https://doi.org/10.1109/INISTA.2011.5946108

[6] Tanmay Kumar Behera, Suvasini Panigrahi, Credit Card Fraud Detection: A Hybrid Approach Using Fuzzy Clustering and Neural Network, IEEE Computer Society, 2015. https://doi.org/10.1109/ICACCE.2015.33

[7] Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang, Credit Card Fraud Detection Using Convolutional Neural Networks, Springer International Publishing AG 2016.

[8] Krishna Modi, Bhavesh Oza, Outlier Analysis Approaches in Data Mining, IJIRT vol 3 issue 7.

[9] Sam Maes, Karl Tuyls, Bram Vanschoenwinkel, Bernard Manderick, Credit card fraud detection using Bayesian and neural network, International Naiso Congress on Neuro Fuzzy Technology, 2002.

[10] "Autoencoder With Invertible Functions for Dimension Reduction and Image Reconstruction", Yimin Yang, Q. M. Jonathan Wu, Yaonan Wang, IEEE 2018.

[11] Michael Nielsen (2017, March 15), Deep learning available, https://neuralnetworksanddeeplearning.com/chap6.html.

[12] Abhinav Shrivastava,Amlan Kundu,Shamik Sural,Senior member IEEE and Arun k Majumdaar, senior member IEEE"Credit card Fraud detection using Hidden Markov Model",IEEE transactions on depend- able and secure computing,vol 5, no 1, January-March-2008. https://doi.org/10.1109/TDSC.2007.70228

[13] "Improved Fuzzy Multicategory Support Vector Machines Classifier", Xi-zhao Wang, Shu-xia Lu, 2006 International Conference on Machine Learning and Cybernetics.

[14] Hardik Manek, Sujai Jain, Nikhil Kataria, Chitra Bhole. “Review on Various Methods for Fraud Transaction Detection in Credit Cards,” International Journal for Innovative Engineering and Management Research, Vol 7, No. 12, 2018.