STM JOURNALS

International Journal of
Algorithms Design and Analysis Review

Volume 1, Issue 1, 2023
January–June
DOI (Journal): 10.37591/IJADAR

https://journals.stmjournals.com/ijadar

Research IJADAR

Card Fraud Detection Using Artificial Neural Network and Multilayer Perception Algorithm

Baku Agyo Raphael[1,*], Bisen Gambo Adashu[2], Andrew Ishaku Wreford[1]

Abstract

Fraud has posed a significant challenge for merchants, especially in the online business sector, over the course of many years. This is primarily due to the advancements in technology that have made credit card transactions a common method of payment. Credit card fraud refers to the unauthorized use of a credit card by an individual for personal purposes, without the owner's consent and with no intention of paying for the incurred expenses or engaging in deceptive activities to gain financial advantage. Given the efforts made by fraudsters to disguise their transactions as legitimate, this study introduces an artificial neural network model powered by a machine learning algorithm to identify and detect fraudulent activities in credit card transactions. The researchers effectively filtered and cleansed the dataset sourced from Kaggle machine learning repository selection techniques. The experiment was set up on a 64-bit Windows OS on an Intel (R) Core (TM) i5-3530 QM CPU @ 2.40 GHZ. Python 3.10 via Anaconda environment using Jupyter notebook was used as the integrated development environment. Dataset exploration, reading, scaling and performance evaluation were done successfully. The study result found prediction accuracy of 0.9184, which is equivalent to 92% at step 716 with 4.6 ms conducted per step and also loss metric based on binary entropy of 2.0%. The study recommends future research and advancement in artificial neural network by hybridizing deep neutral network and Relu neural network for multi-perception optimized performance.

Keywords: Entropy, classifier, credit card fraud, artificial neural network (ANN)

INTRODUCTION

Merchants, particularly those in the online business sector, have faced significant challenges due to fraud over the course of many years. The prevalence of credit card transactions as a popular payment method, enabled by technological advancements, has contributed to this problem. The consequences of fraud have resulted in substantial financial losses. Credit card fraud involves the unauthorized use of a credit card by an individual without the owner's consent, with no intention of reimbursing the incurred expenses, or engaging in deceptive practices for financial benefit [1]. As technology evolves, the fraudsters innovate diverse techniques to bye-pass the security of the system. Merchants and individuals get scammed for simple transactions. Internet banking is more vulnerable to such cases as fraudsters intercept transactions through cross-site forgery. In the real-world fraud detection system, the bulk stream of payment requests is quickly scanned by an automated machine learning model, which authorizes a transaction. Supervised methods are by far the most applied methods in fraud detection, where dataset

***Author for Correspondence**
Baku Agyo Raphael
E-mail: bakuralph@fuwukari.edu.ng

[1]Lecturer, Department of Computer Science, Federal University, Wukari, Taraba, Nigeria.
[2]Lecturer, Department of Computer Science, Kwararafa University, Wukari, Taraba, Nigeria.

Received Date: June 21, 2023
Accepted Date: July 03, 2023
Published Date: July 25, 2023

Citation: Baku Agyo Raphael, Bisen Gambo Adashu, Andrew Ishaku Wreford. Card Fraud Detection Using Artificial Neural Network and Multilayer Perception Algorithm. International Journal of Algorithms Design and Analysis Review. 2023; 1(1): 21–30p.

labels are exploited for training a classifier. Hence, there have been many attempts by researchers to come up with an effective credit card fraud detection system capable of detecting attacks efficiently and effectively, most of which are based on machine learning and deep learning approaches. Several research studies [2–5] have applied the viability of deep learning approaches such as the long short-term memory (LSTM), auto-encoder, etc., to automatically recognize normal and abnormal events happening in the systems and networks due to fraudulent activities. Furthermore, some researchers have also applied machine learning approaches to detect credit card fraud [6–8]. However, with the continuous evolution of fraudulent acts, the challenges of credit card fraud continue. It on this premise, this study proposed to explore the viability of an artificial neural network machine learning model on a credit card dataset obtained from Kaggle to solve the issue of credit card fraud.

The aim of this study is to implement the artificial neural network as a model for detecting credit card fraudulent activities through these specific objectives:

1. To effectively filter and cleanse the dataset using the appropriate data preprocessing model.
2. To apply the correlation matrix as a feature selection technique.
3. To tune the performance measure of the model, to meet an acceptable performance rate.
4. To use the accuracy models to validate the model's performance.

RELATED WORK

To detect credit card fraud, a mixed machine learning method was proposed. The hybrid methods proposed by the study incorporate AdaBoost, majority voting methods, and others (such as Bayesian, random forest, decision tree, neural network, linear regression, logistic regression, and support vector machine) were applied [7]. The model's performance was evaluated using a publicly available dataset of credit card transactions. Additionally, an analysis was conducted on a batch of real credit card data obtained from a financial institution. To assess the algorithms' resilience, noise was introduced to the data samples. The experimental results indicate that the majority voting method exhibits a high level of accuracy in identifying instances of credit card fraud.

The performance of naïve Bayes, k-nearest neighbors, and logistic regression was examined using a highly imbalanced dataset of credit card fraud [8]. The dataset, obtained from European cardholders, consisted of 284,807 transactions. To address the data skewness, a hybrid approach combining under-sampling and oversampling techniques was applied. The effectiveness of the techniques was assessed based on various metrics, including accuracy, sensitivity, specificity, precision, Matthews correlation coefficient, and balanced classification rate. The results revealed that the accuracy rates for naïve Bayes, k-nearest neighbors, and logistic regression classifiers were 97.92%, 97.69%, and 54.86%, respectively. The comparative results show that the k-nearest neighbor performs better than naïve Bayes and logistic regression techniques [9] proposed and designed a credit card fraud detection system using machine learning approaches such as decision tree (DT), k-nearest neighbor (KNN) algorithm, Extreme learning machine (ELM), multilayer perceptron (MLP), and support vector machine (SVM) to detect the accuracy in an attempt to identify fraud. For effective data interchange across numerous diverse systems, they employed two web-based protocols called representational state transfer (REST) and simple object access protocol (SOAP). They compared five machine learning algorithm results based on accuracy metrics. SVM performed better than other algorithms with an accuracy score of 81.63% but the hybridized system in general achieved an accuracy of 82.58%.

Deepika and Senthil [4] presented a credit card fraud detection system using moth-flame earthworm optimization algorithm-based deep neural network [4].

The proposed system utilizes a database containing credit card transaction information, which is then passed through the pre-processing stage. During the pre-processing step, the database undergoes a log

transformation to regulate the data. After, the appropriate features were selected by the information gain criterion, and the selected features were utilized to train the classifier using the adopted moth-flame earthworm optimization-based deep belief network (MF-EWA-based DBN). The weights for the classifier were selected by the newly developed moth-flame earthworm optimization algorithm (MF-EWA). The study claimed that the proposed MF-EWA-based DBN classifier improved detection with astounding performance and outclassed other existing models with 85.89% accuracy [5]. A novel method was suggested to efficiently identify instances of credit card fraud by employing a neural network ensemble classifier combined with a hybrid data resampling technique. The ensemble classifier was constructed using an LSTM neural network as the underlying learner within the adaptive boosting (AdaBoost) framework. To enhance the performance, a hybrid resampling method was employed, involving a synthetic minority oversampling technique and the edited nearest neighbor (SMOTE-ENN) method. The efficacy of this approach was demonstrated by utilizing publicly available real-world credit card transaction datasets, and its performance was compared against several established algorithms including SVM, MLP, decision tree, traditional AdaBoost, and LSTM. The experimental findings revealed that the classifiers yielded superior results when trained with the resampled data, with the proposed LSTM ensemble displaying the highest performance by achieving a sensitivity and specificity of 0.996 and 0.998, respectively [10].

Faraji [10] in his study "A Review of Machine Learning Applications for Credit Card Fraud Detection with a Case Study" highlighted the widely used supervised techniques applied for fraud detection. The author further targeted the application of some techniques to evaluate the performance of real-world data and develop an ensemble model as a potential solution for this problem. The method applied by the study for fraud detection purposes were logistic regression, decision tree, random forest, KNN, and XGBoost. To validate the performance of the models the study optimized the precision and recall metrics to evaluate the performance, calculated based on the confusion matrix. The study predicted that XGBoost would be the fastest and perform the best; nevertheless, it only outperforms the random forest in terms of accuracy, precision, recall, and F1-score. In general, the author noted that the KNN and logistic regression have better performance, which means they better detect fraudulent transactions.

Bommala et al. [11] reported on an innovative model-based approach for credit card fraud detection optimized the viability of machine learning techniques. The study optimized the viability of the KNN as an intelligent approach that can improve the detection of fraud in credit card transactions. The experimental result of the study found an accuracy of 94%. It was claimed that the large percentage obtained was due to inconsistency in the number of legal and illegal transactions.

METHODOLOGY

This study structured a three-phase methodological approach that encompasses data filtering and feature selection, deep learning model application, and model evaluation as shown in Figure 1.

Dataset Description

The dataset adopted by this study is the credit card fraud dataset sourced from the Kaggle machine learning repository. The objective of the dataset is to predict customers that will be qualified to be issued loans by the financial institution or not based on certain previous financial records measurements included in the dataset. Certain restrictions were imposed on the choice of these cases from a larger database, specifically requiring that all customers be aged 21 years or older. The datasets encompass a variety of credit risk predictor variables, which are independent factors, as well as a single target variable known as the outcome, which is the dependent variable. Independent variables include the person's age, person income, person home ownership, person employment length, loan intent, loan grade, loan amount, loan interest rate, loan status, loan percent income, citizens band person default on file, and person credit history length.

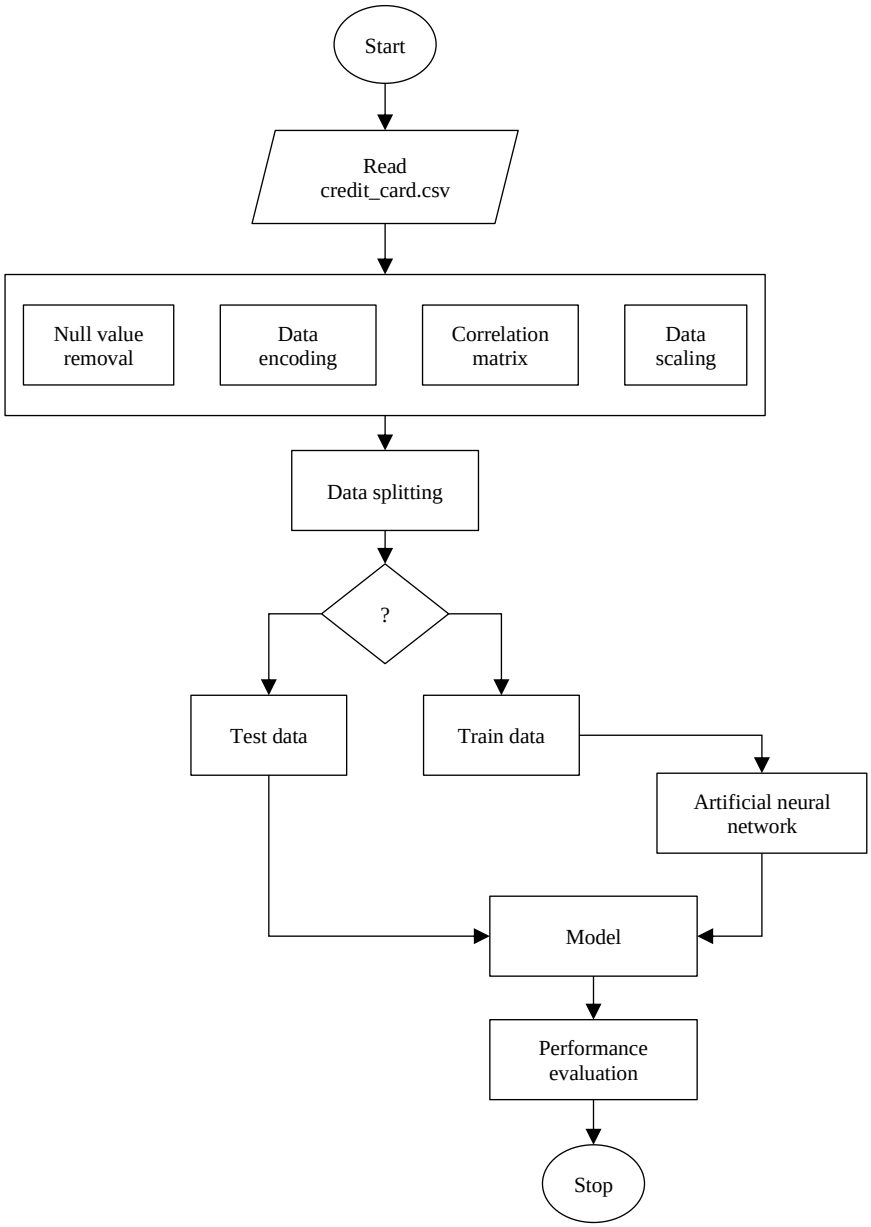

Figure 1. Methodological approach.

Data Preprocessing

The data preprocessing steps adopted by this study include:

1. Importing/reading the sourced dataset using the Pandas framework and the identification and handling of Null and Na values.
2. Object encoding is meant to transform categorical data values into a numerical field.
3. Identifying feature relevance for feature selection based on the correlation matrix.
4. Scaling the selected dataset to a range between 0 and 1.

Training and Testing Dataset

Mathematically, suppose that there is a total of *n* training data points in the training dataset *S*. And for each data point denoted as d^n, there exists a vector of predictive variables denoted as $x^n = x_1^n, x_2^n, \ldots x_m^n$ and a class label denoted as $y^n = 1$ if the subject met the definition of credit card fraud; otherwise, $y^n = 0$).

Hence, for the training and testing datasets, 75% of the sample dataset size was used as training dataset *S*, from the independent and dependent variable domain x^n and y^n whereas, the remaining 25% of the dataset from the sample size *S* was used as the testing data for also both the independent and dependent variable domain x^n and y^n.

Artificial Neural Network

The study after an extensive survey identified the viability of the artificial neural network (ANN) in the classification domain and hence proposed its adoption with a multilayer perceptron to detect credit card fraudulent transactions as shown in Figure 2.

The MLP is a conceptual representation of a feed-forward ANN consisting of interconnected neurons with associated linking weights. Its purpose is to transform a set of inputs into the desired outputs. Figure 3 illustrates the structure of an MLP, which comprises three fundamental components: an input layer, a hidden layer, and an output layer. The input layer receives the data and passes it on to the first hidden layer, which in turn transmits it until it reaches the output layer. It is worth noting that each layer consists of a specific number of neurons, and the connections between neurons are established using weights and biases. The output (o_n) of each artificial neuron n in the hidden layer can be computed using the following equation:

$$o_n = f\sum_{i=1}^{n} w_i x_i + b \quad (1)$$

Algorithm 3.1: Artificial Neural Network

Step 1: Passed the input with some weight to the hidden layers $(x^1 x^2, \ldots \ldots x^6)$

Step 2: Connect all the inputs to each neuron

Step 3: perform computation at the hidden layers

Step 3.1: Get the summation of all input with their weight

Step 3.2: Get bias (check Figure 2).

Step 3.3: Get the threshold unit (check Figure 2).

Step 4: Repeat step 3 for each of the hidden layers

Step 5: Pass the result to an output layer

Step 6: Get predictions from the output layers and hence calculate the performance metrics.

Step 7: Calculate error, i.e., the difference between the actual and predicted output.

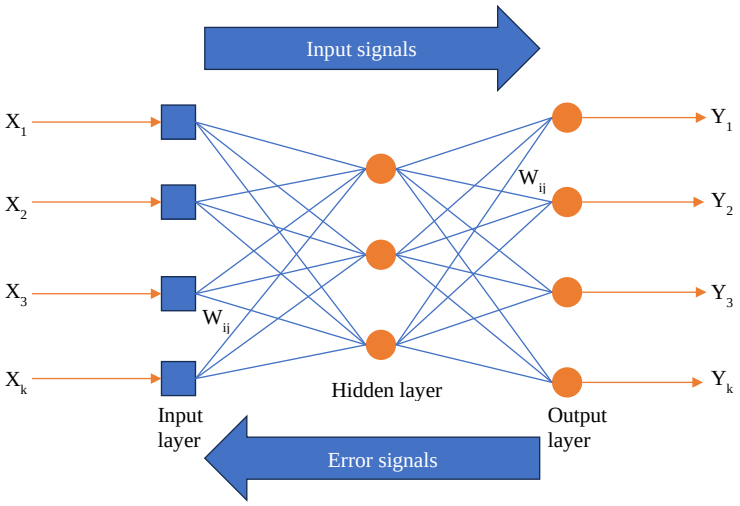

Figure 2. Artificial neural network.

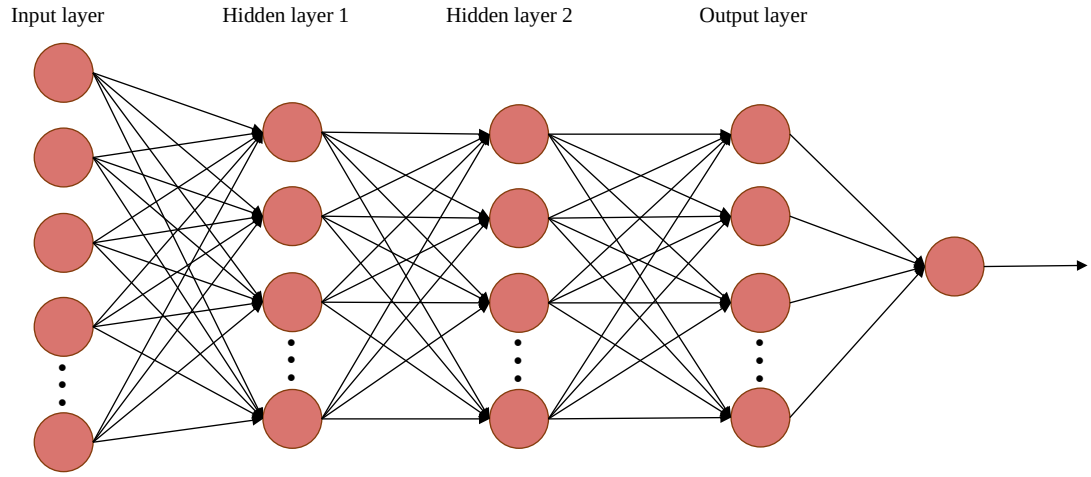

Figure 3. Multilayer perceptron.

Software Tool and Design Materials

The proposed implementation utilized the viability of the below packages for its effective implementation:

1. Python software development kit (SDK).
2. Numpy, Pandas, Sklearn, MatplotLib, and Keras from TensorFlow a high-level API (application programming interface).
3. Jupyter Notebook as the programming environment.

Performance Evaluation Metric

The evaluation metrics employed to assess the performance of the proposed ANN model extended with a multilayer perceptron are as follows:

1. *Accuracy:* The number of correct predictions made by the model over all types of predictions is defined as the percentage of correctly classified instances out of all instances. Accuracy is a good metric when the target variable classes in the data are almost evenly distributed.

$$Accuracy = \frac{(TP+TN)}{(TP+FP+FN+TN)} \quad (2)$$

   Where, The label True Positives (TP) defines an instance in which the data point's actual class is True and the model predicted it to be True, True Negatives (TN) defines a scenario where the data point's actual class is False and the model anticipated it to be False, False Positives (FP) defines an instance circumstance where the data point's real class is False but is projected to be True and finally the False Negatives (FN) label that defines an instance scenario where the data point's real class is True but is anticipated to be False.
2. *Time taken to extract:* It is the amount of time it takes the algorithm to achieve a feasible accurate score. It is timed in seconds.

RESULTS AND DISCUSSION

In this section, we assess the performance of our proposed ANN model, which was extended with a multilayer perceptron, using the following evaluation metrics.

Experimental Setup

The experiments were set up on a 64-bit Windows OS on an Intel(R) Corel (TM) i5-3630QM CPU @2.40 GHZ with 8.00 GB of RAM. The dataset used for the experiments was the credit card approval dataset from the Kaggle machine learning repository webpage. The SDK used is Python 3.10 via the Anaconda environment using Jupyter Notebook as the integrated development environment.

Dataset Reading and Exploration

One of the first processes in the analysis process is data exploration, which is used to start looking for patterns and trends in the dataset.

Dataset Reading

This study utilized the Python Pandas framework, as the library provides functionalities for reading comma-separated value file format in a tabular structure making it easy to visualize patterns from the dataset in a readable and understandable format. Figure 4 shows the reading of the adapted credit card fraud dataset using the read_csv function of Pandas library while Figure 5 depicts the first five records after a successful data read using the read_csv function of Pandas library.

Dataset Exploration

The dataset exploration defines traversing through the dataset statistics. Hence, considering the dataset statistics after the sourced credit card dataset was read via the panda's data frames. Figure 6 depicts the count, mean, standard deviation, min, max, and scale data interval of 25%, 50%, and 75%, respectively. The count column corresponds to the number of tuples within the dataset, which equate to a total of 28368 records from a total of 12 column set. The min defines the minimum floating value for the data column, std defines the dataset standard deviation for a specific labeled column, and max defines the maximum values in a particular column.

```
In [2]: df = pd.read_csv("credit_risk_dataset.csv")

In [3]: df.head()
```

Figure 4. Pandas' read CSV.

```
In [3]: df.head()
```

Out[3]:

	person_age	person_income	person_home_ownership	person_emp_length	loan_intent	loan_grade	loan_amnt	loan_int_rate
0	22	59000	RENT	123.0	PERSONAL	D	35000	16.02
1	21	9600	OWN	5.0	EDUCATION	B	1000	11.14
2	25	9600	MORTGAGE	1.0	MEDICAL	C	5500	12.87
3	23	65500	RENT	4.0	MEDICAL	C	35000	15.23
4	24	54400	RENT	8.0	MEDICAL	C	35000	14.27

Figure 5. Credit card fraud dataset sample.

```
In [10]: df.describe()
```

Out[10]:

	person_age	person_income	person_emp_length	loan_amnt	loan_int_rate	loan_status	loan_percent_income	cb_
count	28638.000000	2.863800e+04	28638.000000	28638.000000	28638.000000	28638.000000	28638.000000	
mean	27.727216	6.664937e+04	4.788672	9656.493121	11.039867	0.216600	0.169488	
std	6.310441	6.235645e+04	4.154627	6329.683361	3.229372	0.411935	0.106393	
min	20.000000	4.000000e+03	0.000000	500.000000	5.420000	0.000000	0.000000	
25%	23.000000	3.948000e+04	2.000000	5000.000000	7.900000	0.000000	0.090000	
50%	26.000000	5.595600e+04	4.000000	8000.000000	10.990000	0.000000	0.150000	
75%	30.000000	8.000000e+04	7.000000	12500.000000	13.480000	0.000000	0.230000	
max	144.000000	6.000000e+06	123.000000	35000.000000	23.220000	1.000000	0.830000	

Figure 6. Descriptive statistics.

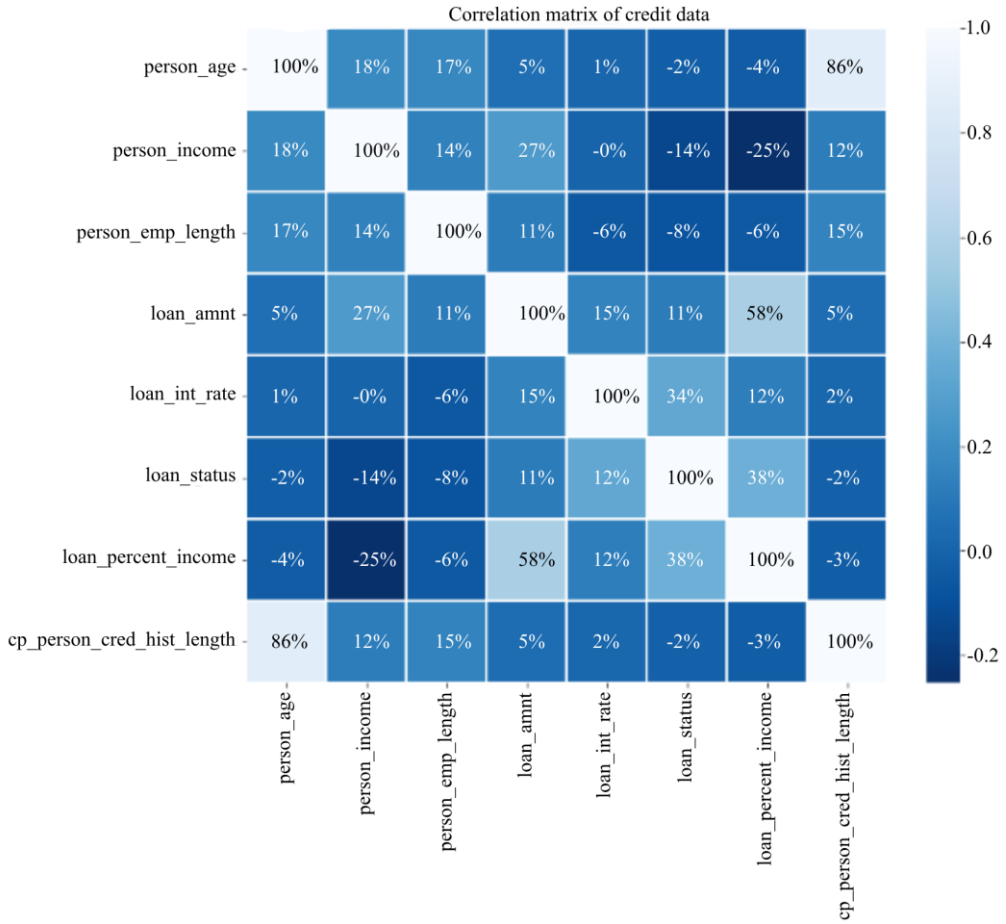

Figure 7. Data correlation.

Furthermore, the correlation matrix from Figure 7 defines the cohesiveness of the independent variable in an account of the prediction of credit card presence. More extensively, the correlation matrix determines the feature importance of each data attribute to another attribute in fraud identification. It explicitly enhances the performance of the predictive model, if the relevance attribute is selected. Hence, the correlation of the dataset features is depicted in Figure 7. The diagonal of the correlation diagram identifies that each column is at 100% correlation to itself while other cells define a particular data cell correlation concerning an intercepted data cell. An instance is that the person income to person income attribute has a 100% value correlation whereas person income to person age has an 18% correlation.

Data Scaling

This study conducted data scaling to bound each feature attributes value to a feature range between 0 and 1 in an attempt to normalize the dataset and thus enable the ANN model to control the magnitude of the dataset set while ensuring faster convergence to a global optimum state where the result of the model is the feasible best result. To scale the dataset, the standard scaler module from the Sklearn library was utilized. Evidence of the standard scaler preprocessor adherence is shown in Figure 8.

Model Build-Up

This study towards the detection of loan defaulters for early treatment optimized the viability of the ANN with a Flatten to remember prediction state. The network was designed with three separate layers with 256, 128, and 1 neurons for each respective layer as shown in Figure 9. The activation function optimized was the relu (rectified linear function) and sigmoid mostly for binary classification of either been of fraud or not (0 or 1).


```
In [14]: #features scaling
         x_scaler = MinMaxScaler()
         x_scaler.fit(x)
         x[x.columns] = x_scaler.transform(x)
```

Figure 8. Data scaling.

```
In [16]: # building the model
         model = Sequential()
         model.add(Flatten())
         model.add(Dense(256, input_dim=len(x.columns), kernel_initializer=k.initializers.random_normal(seed=13
         model.add(Dropout(0.3))

         model.add(Dense(128, activation="relu"))
         model.add(Dropout(0.3))

         model.add(Dense(1, activation="sigmoid"))
```

Figure 9. Artificial neural network.

Presentation of Results

After passing the scaled independent and dependent values to the model, the model predicted and compiled an overall prediction accuracy of 0.9184 value, which is equivalent to 92% at step 716 with 4.6 ms conducted per step and also loss metrics based on binary cross entropy of 0.2%. Taking into consideration the time evaluation metric proposed to evaluate the performance of the model, the response time (4.6 ms) of the model in achieving an accuracy of 92% is quite astonishing. The accuracy of the model was also revalidated using the accuracy score metric from the Sklearn modules and the result of the accuracy matches very closely with a score of 91%. This is shown in Figure 10 as captured from the Jupyter Notebook, which was the coding environment.

```
Epoch 50/50
716/716 [==============================] - 4s 6ms/step - loss: 0.2421 - accuracy: 0.9167
```

Figure 10: Artificial neural network accuracy.

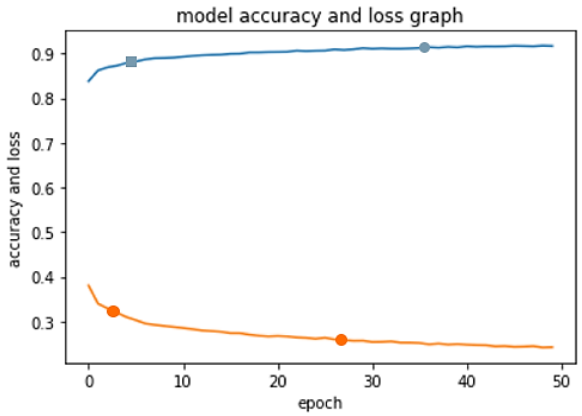

Figure 11. Training epochs.

Figure 11 depicts a graph that shows the accuracy and loss growth as the loss diminishes over 50 training epochs scheduled, with an increase in the model prediction upon the same 50 epochs.

CONCLUSION

This study investigated the performance of the ANN on a credit card fraud dataset obtained from the Kaggle machine repository. The credit card fraud dataset was sampled in achieving two sets of data distributions as either being a fraud or not after the conductance of data filtration and feature selection based on the value from the correlation matrix table as a feature selection technique. The implementation of the model was conducted in python programming accompanied by some third-party libraries such as

Numpy, Tensorflow, Keras, Matplotlib, Pandas, and Sklearn, to facilitate the effective performance of the predictive ANN model. The steps incorporated to achieve the predictive model include data reading, cleansing, feature selection via the correlation matrix algorithm, scaling and normalization, feeding the filtered dataset after being split into train and test data to the artificial neural network, and performance evaluations. The performance of the model was examined using the accuracy score metric with a binary cross-entropy as the loss function and Adam as the optimizer while sigmoid was applied to the last dense layer because the problem under study is of binary classification type. The model's performance demonstrated significant success when the data was split into a 75:25 ratio for training and testing, yielding an impressive accuracy rate of 91%. Hence, this study revealed the viability of the ANN with MLP in the detection of credit card fraud. Its major application is considered to be financial institutions. Hence, the bank and other financial enterprises can adopt the designed model to detect customers who will fail to remit credit if they borrow.

REFERENCES

1. Bhatla TP, Prabhu V, Dua A. Understanding credit card frauds. Cards Business Rev. 2003; 1 (6): 1–15.
2. Tsai CF. Combining cluster analysis with classifier ensembles to predict financial distress. Inform Fusion. 2014; 16: 46–58.
3. Kang F, Cheng D, Tu Y, Zhang L. Credit card fraud detection using convolutional neural networks. In: Hirose A, Ozawa S, Doya K, Ikeda K, Lee M, Liu D, editors. Neural Information Processing. ICONIP 2016. Lecture Notes in Computer Science, Volume 9949. Cham, Switzerland: Springer; 2016. pp. 483–490. doi: 10.1007/978-3-319-46675-053.
4. Deepika S, Senthil S. Credit card fraud detection using moth-flame earthworm optimization algorithm-based deep belief neural network Int J Electron Security Digital Forensics. 2021; 14 (1): 53–75.
5. Esenogho E, Mienye ID, Swart TG, Aruleba K, Obaido G. A neural network ensemble with feature engineering for improved credit card fraud detection. IEEE Access. 2022; 10: 16400–16407. doi: 10.1109/ACCESS.2022.3148298.
6. Bhattacharyya S, Jha S, Tharakunnel K, Westland JC. Data mining for credit card fraud: a comparative study. Decis Support Syst. 2011; 50 (3): 602–613.
7. Randhawa K, Loo CK, Seera M, Lim CP, Nandi AK. Credit card fraud detection using AdaBoost. IEEE Access. 2018; 6: 14277–14284. doi: 10.1109/ACCESS.2018.2806420.
8. John OA, Adebayo OA, Samuel AO. Credit card fraud detection using machine learning techniques: a comparative analysis. Int J Soft Comput Eng. 2017; 1: 32–38.
9. Prusti D, Rath SK. Web service-based credit card fraud detection by applying machine learning techniques. In: Proceedings of the TENCON 2019: -IEEE Region 10 Conference (TENCON), October 17–20, 2019, Kochi, India, pp. 492–497. doi: 10.1109/TEN-CON.2019.8929372.
10. Faraji Z. A review of machine learning applications for credit card fraud detection with a case study. SEISENSE J Manage. 2022; 5 (1): 49–59. doi: /10.33215/sjom.v5i1.770.
11. Bommala H, Basha RM, Rajarao B, Sangeetha K. An innovative model-based approach for credit card fraud detection using K-nearest. In: Reddy AB, Kiranmayee B, Mukkamala RR, Srujan Raju K, editors. International Conference on Advances in Computer Engineering and Communication Systems. Algorithms for Intelligent Systems. Singapore: Springer; 2022. pp. 199–206. doi: 10.1007/978-981-16-7389-4_19.

