A Deep Learning Based Approach with Regularization for Credit Card Fraud Detection on Imbalanced Datasets

Raj Kumar Ahirwar[1], Ms. Pooja Hardiya[2]
SDBCT, Indore, India[1,2]

Abstract— The rapid growth of digital transactions and online banking has significantly increased the occurrence of credit card fraud. Conventional rule-based fraud detection systems are often unable to cope with the evolving and sophisticated nature of fraudulent activities. Consequently, Machine Learning (ML) and Deep Learning (DL) techniques have emerged as effective solutions for identifying fraudulent transactions. Millions of credit card transactions occur every day, making manual inspection impractical. Machine learning and deep learning algorithms can automatically process vast amounts of transaction data and identify suspicious activities in real time. Fraudsters continuously develop new techniques to bypass traditional security mechanisms. Unlike static rule-based systems, ML and DL models can learn from historical transaction patterns and adapt to emerging fraud strategies, thereby improving detection capability. This work proposes a supervised machine learning algorithm to be trained to detect credit card frauds based on the Bayes Net with penalty based regularization. It is shown that the proposed approach attains higher classification accuracy compared to existing work.

Keywords— Credit Card Fraud Detection, Machine Learning, Feature Selection, Imbalanced Datasets, Probabilistic Classifier, Classification Accuracy

I. INTRODUCTION

Credit card fraud has become one of the most significant challenges faced by financial institutions due to the rapid growth of electronic commerce, online banking, and digital payment systems [1]. The increasing number of transactions performed every day has created opportunities for fraudsters to exploit vulnerabilities in payment networks. Traditional fraud detection systems based on manually defined rules are often inadequate because they cannot effectively adapt to new and sophisticated fraudulent strategies. Consequently, machine learning and deep learning approaches have gained considerable attention for their ability to provide intelligent, automated, and accurate fraud detection mechanisms. One of the primary reasons for employing machine learning and deep learning techniques is their capability to process and analyze enormous volumes of transaction data. Modern payment systems generate millions of transactions daily, making manual examination practically impossible. Machine learning algorithms can efficiently handle large datasets and identify suspicious patterns in real time, enabling financial institutions to detect fraudulent activities before significant losses occur [2].

Another important factor is the constantly evolving nature of credit card fraud. Fraudsters continuously modify their tactics to evade conventional security measures. Rule-based systems rely on predefined conditions and therefore become ineffective when new fraud patterns emerge. Machine learning models can learn from historical transaction records and continuously improve their prediction capabilities, while deep learning models can automatically adapt to complex and changing fraud scenarios, making them more robust against emerging threats [3]. Machine learning and deep learning approaches are also essential because fraudulent transactions often involve complex and nonlinear relationships among multiple attributes such as transaction amount, time, location, merchant category, and customer spending behavior. Conventional statistical techniques may fail to capture these intricate relationships. Deep neural networks, however, possess powerful feature extraction capabilities that allow them to discover hidden patterns and dependencies within transaction data, thereby enhancing detection accuracy [4].

Another major requirement is the need for real-time fraud detection. Unauthorized transactions must be identified immediately to prevent financial damage and protect customers. Machine learning models can rapidly classify transactions as genuine or fraudulent and trigger appropriate actions such as transaction blocking or customer verification. This ability to provide instant responses significantly improves the effectiveness of fraud prevention systems. Machine learning and deep learning techniques are particularly useful in dealing with the class imbalance problem inherent in credit card datasets. Fraudulent transactions constitute only a very small percentage of the total transaction volume, which makes their identification challenging. Advanced algorithms combined with data balancing techniques and cost-sensitive learning approaches can effectively recognize minority fraudulent cases without compromising the classification of legitimate transactions [5].

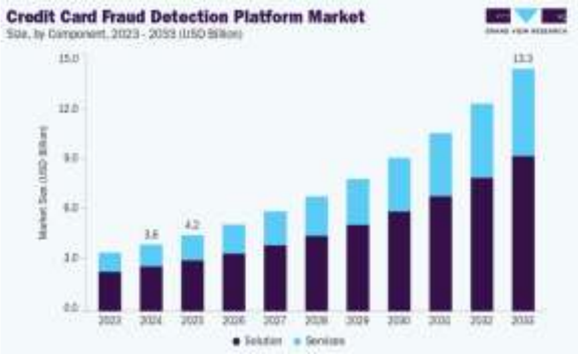

Fig.1 Market Cap for Credit Card Fraud Detection Models

Machine learning (ML) algorithms have the ability to analyze vast amounts of transaction data in real time, identifying patterns and anomalies that may indicate fraudulent activity. Unlike rule-based systems, which rely on predefined criteria, ML models can learn from historical data and adapt to new types of fraud as they emerge. This dynamic learning capability enhances the accuracy and effectiveness of fraud detection systems, reducing false positives and enabling quicker responses to potential threats [6].

II. MACHINE LEARNING MODELS FOR IDENTIFYING CREDIT CARD FRAUDS

Various machine learning algorithms are employed to detect credit card fraud, each with its unique strengths. Supervised learning algorithms, such as decision trees and support vector machines, are trained on labeled datasets where examples of both fraudulent and non-fraudulent transactions are provided [7]. This training enables the models to classify new transactions with a high degree of accuracy. On the other hand, unsupervised learning algorithms, like clustering and anomaly detection methods, do not require labeled data and can identify outliers in transaction data that may represent fraud. These algorithms are particularly useful for detecting novel fraud patterns that have not been previously encountered [8].

Decision trees: Decision trees are commonly used for fraud detection since they are straightforward and easy to understand. They operate by partitioning the dataset into subsets according to the input feature values, resulting in a hierarchical structure of decision nodes. Every node in the representation reflects a specific feature, each branch represents a decision rule, and each leaf represents an outcome. Decision trees possess the capability to process both numerical and categorical input, rendering them adaptable and comprehensible [9].

Random Forests: Random forests use the idea of decision trees by employing a collection of numerous trees to enhance accuracy and resilience. The construction of each tree in a random forest involves using a random subset of the data, which helps to reduce overfitting and improve prediction performance. Random forests excel at detecting intricate fraud patterns in extensive datasets, providing exceptional accuracy and robustness against interference [10].

Logistic regression: Logistic regression is a statistical model that is specifically designed for binary classification tasks, allowing it to effectively differentiate between fraudulent and non-fraudulent transactions. The logistic function is used to evaluate the probability of a given input belonging to a specific class. Logistic regression is renowned for its simplicity, efficiency, and interpretability. It is particularly useful in cases where the relationships between features may be approximated as linear [11].

Support Vector Machines (SVM): Support vector machines (SVM) are robust classifiers that identify the most effective hyperplane for distinguishing between various classes in a space with many dimensions. Support Vector Machines (SVMs) are highly efficient in dealing with data that has a large number of dimensions. They are particularly valuable when the classes cannot be separated by a straight line, as they can employ kernel functions to transform inputs into spaces with even more dimensions. The capacity of SVMs to detect fraud makes them a highly advantageous option [12].

Artificial neural networks: Neural networks, particularly deep learning models, have become popular due to their capacity to acquire intricate patterns from extensive datasets. Neural networks are capable of representing complex connections between characteristics in fraud detection, enabling them to detect tiny deviations that are indicative of fraudulent activity. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are utilized depending on the characteristics of the input and the specific demands of the task [13].

K-Nearest Neighbors (KNN): The K-nearest neighbors (KNN) technique is an instance-based learning method utilized for categorization. It identifies the 'k' most comparable transactions (neighbors) to a particular transaction. The class that has the highest number of occurrences among the neighbors is allocated to the new transaction. KNN is characterized by its simplicity and intuitiveness, yet it may incur high computing costs when dealing with extensive datasets. However, it is still efficient for smaller datasets and can yield rapid, easily understandable outcomes [14].

Despite the benefits, implementing machine learning for fraud detection comes with challenges. One major issue is the imbalance in datasets, where fraudulent transactions are significantly outnumbered by legitimate ones. This imbalance can skew the model's performance, making it less effective at identifying fraud. Techniques such as oversampling, undersampling, and synthetic data generation are often employed to address this issue. Additionally, ensuring the privacy and security of transaction data is paramount, as any breaches could have severe consequences for both consumers and financial institutions [15].

III. EXISTING CHALLENGS OF CLASS IMBALANCE

Class imbalance in the context of credit card fraud detection using deep learning refers to the unequal distribution of fraudulent and non-fraudulent transactions in the dataset. In most real-world scenarios, fraudulent transactions constitute only a tiny fraction of the overall transaction volume, while the majority of transactions are legitimate. This imbalance can pose challenges for machine learning models because they tend to be biased towards the majority class, leading to poor performance in identifying the minority class (fraudulent transactions). The imbalanced nature of the dataset can cause the model to prioritize accuracy at the expense of effectively detecting fraudulent transactions. As a result, the model may tend to classify most transactions as non-fraudulent, achieving high accuracy due to the dominance of the majority class but failing to detect fraudulent activities adequately [16].

Imbalanced Datasets

Imbalanced datasets pose significant challenges in credit card fraud detection, where the number of legitimate transactions far outweighs the instances of fraud. This imbalance can lead to biased models and hinder the effectiveness of fraud detection systems. Here are several challenges associated with imbalanced datasets in credit card fraud detection [17]:

Limited Representation of Fraudulent Cases*:* Imbalanced datasets often result in a scarcity of fraudulent transactions for model training. This limited representation makes it challenging for the algorithm to learn the patterns and characteristics of fraudulent activities, leading to a less accurate and robust model.

Biased Model Performance*:*

Traditional machine learning algorithms are biased towards the majority class, in this case, non-fraudulent transactions. As a result, the model may prioritize accuracy on the majority class while neglecting the minority class (fraudulent transactions). This bias can lead to poor fraud detection performance.

High False Negative Rates*:*

Imbalanced datasets can contribute to a higher rate of false negatives, where fraudulent transactions are incorrectly classified as non-fraudulent [18].

Dynamic Nature of Fraud Patterns:

Fraudulent activities evolve over time, and imbalanced datasets may not capture the latest patterns. As fraudsters adapt their tactics, models trained on imbalanced historical data may struggle to generalize to emerging fraud patterns.

Class Imbalance Mitigation

Addressing class imbalance is crucial in credit card fraud detection to ensure that the model can effectively identify fraudulent transactions while minimizing false positives. Various techniques can be employed to mitigate class imbalance, including [19]:

1. Resampling methods: This involves either oversampling the minority class (fraudulent transactions) to balance the class distribution or under sampling the majority class (non-fraudulent transactions) to reduce its dominance [20].
2. Algorithmic approaches: Some algorithms, such as ensemble methods like Random Forest or boosting algorithms like XGBoost, inherently handle class imbalance by adjusting the training process to give more weight to the minority class.
3. Cost-sensitive learning: Assigning different misclassification costs to different classes during model training to penalize misclassifying fraudulent transactions more severely can help mitigate class imbalance.
4. Synthetic data generation: Generating synthetic samples for the minority class using techniques like SMOTE (Synthetic Minority Over-sampling Technique) can help balance the class distribution and improve model performance.

By addressing class imbalance effectively, deep learning models for credit card fraud detection can achieve better sensitivity and specificity, thereby enhancing their ability to accurately detect fraudulent transactions while minimizing false alarms [21].

IV. PROPOSED ALGORITHM

This work proposes the BayesNet with penalty based regularization, to update weights more effectively compared to the conventional Naïve Bayes. The gradient is considered as the objective function to be reduced in each iteration. A probabilistic classification using the Bayes theorem of conditional probability is given by [22]:

$$\boldsymbol{P}\left(\frac{\boldsymbol{H}}{\boldsymbol{X}}\right) = \frac{\boldsymbol{P}\left(\frac{\boldsymbol{X}}{\boldsymbol{H}}\right)\boldsymbol{P}(\boldsymbol{H})}{\boldsymbol{P}(\boldsymbol{X})} \qquad \textbf{(1)}$$

Here,

Posterior Probability [P (H/X)] is the probability of occurrence of event H when X has already occurred

Prior Probability [P (H)] is the individual probability of event H

X is termed as the tuple and H is is termed as the hypothesis.

Here, [P (H/X)] denotes the probability of occurrence of event X when H has already occurred.

Each node is associated with a conditional probability distribution that quantifies the effect of its parents in the graph. Bayes Nets provide a structured way to model joint probability distributions, allowing for efficient inference and learning. They are particularly useful in domains where relationships among variables are complex and uncertain, such as medical diagnosis, risk assessment, and machine learning.

The probability function can be computed using equation 24.

$$P\left(\frac{X}{X_i, k_1, k_2, M}\right) = \frac{P\left(\frac{X_i}{X, k_2, M}\right) P\left(\frac{X_i}{k_1, M}\right)}{P\left(\frac{X}{k_1, k_2, M}\right)} \quad (2)$$

Here,
P denotes probability
X_i denotes the set of weight and bias
X denotes the training data set
M denotes the network architecture in terms of the hidden layers and neurons
k_1 and k_2 are the regularization parameters for the network

Incorporating prior distributions over the parameters or network structures, guiding the learning process towards more plausible models. Priors can reflect domain knowledge or be designed to favor simpler models, thereby enhancing generalization [23]

Generally, the term $\rho = \frac{k_1}{k_2}$ is called the regularization ratio. The regularization parameter is adopted in this case to limit the variations in the weights by introducing a penalty factor to the learning algorithm's cost function or objective function $\boldsymbol{J}$. The regularization is different from early stopping or convergence in the sense that the earlier truncates the iterations prior to convergence to a minimum value of $\boldsymbol{J}$ whereas the latter tries to restrict the values of weights and number of parameters by modifying the cost function. Thus, regularization allows a much steeper decrease in the cost function and eventually lesser values as compared to early stopping. This significantly helps to reduce the time complexity of the algorithm.

Algorithm:
The training algorithm adopted in this work is given by:

Step.1: Initialize weights (w) randomly.

Step.2: Fix the maximum number of iterations (n) and compute $\rho = \frac{k_1}{k_2}$

Step.3: Update weights using gradient descent with an aim to minimize the objective function J given by:

$$J = \frac{1}{m}\sum_{i=1}^{m}(v_i - v'_i)^2 \quad (3)$$

Step.4: Compute the Jacobian Matrix $\boldsymbol{J}$given by:

$$J = \begin{bmatrix} \frac{\partial^2 e_1}{\partial w_1^2} & \cdots & \frac{\partial^2 e_1}{\partial w_m^2} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 e_n}{\partial w_1^2} & \cdots & \frac{\partial^2 e_n}{\partial w_m^2} \end{bmatrix} \quad (4)$$

Here,
The error for iteration 'i' designated by e_i is computed as:

$$e_i = (y_i - y'_i) \quad (5)$$

Here
y_i is the actual value
y'_i is the predicted value

Step.5: Iterate steps (1-4) till the cost function J stabilizes or the maximum number of iterations set in step 2 are reached, whichever occurs earlier.

Regularization enhances the robustness and generalizability of Bayesian Networks by preventing overfitting. By constraining the model complexity, regularization techniques ensure that the learned network captures the essential dependencies among variables without being influenced by noise. This leads to improved predictive performance on new data and more reliable inferences. Additionally, regularization facilitates the interpretation of the network by avoiding unnecessarily complex structures, making it easier to understand and communicate the relationships among variables.

Bayes Nets offer several advantages in the context of credit card fraud detection. Firstly, they provide a structured way to combine various sources of information, such as transaction history, user behavior, and contextual data. This holistic approach enables the detection of subtle fraud patterns that might be missed by simpler models. Secondly, Bayes Nets can handle missing data effectively, which is common in real-world scenarios. They can also update their predictions in real-time as new data becomes available, making them highly adaptive to evolving fraud tactics.

Overlapping features values with fuzzy boundaries can not be classified accurately based on hard boundary conditions. Hence the Bayes Net is applied. The final classification accuracy is computed as:

$$Ac = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

Here.
TP represents true positive
TN represents true negative
FP represents false positive
FN represents false negative

V. RESULTS

This section presents the experimental results. The dataset is extracted from Kaggle (httpswww.kaggle.comdatasetsnelgiriyewithanacredit-card-fraud-detection-dataset-2023)
The dataset obtained for the prototype ML model using the Deep BayesNet has the following attributes:
This dataset contains credit card transactions made by European cardholders in the year 2023.
It comprises over 550,000 records, and the data has been anonymized to protect the cardholders' identities. The primary objective of this dataset is to facilitate the development of fraud detection algorithms and models to identify potentially fraudulent transactions.
id: Unique identifier for each transaction
V1-V28: Anonymized features representing various transaction attributes (e.g., time, location, etc.)
Amount: The transaction amount

Class: Binary label indicating whether the transaction is fraudulent (1) or not (0) **(target).**

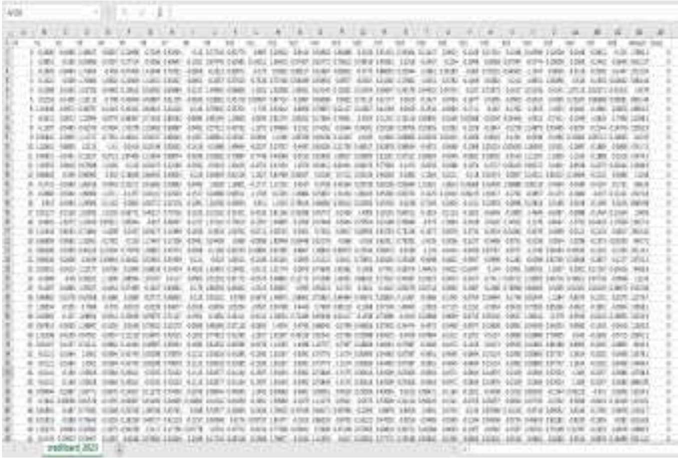

Fig.2 Raw Credit Card Fraud Dataset

Figure 2 depicts the raw data to be analyzed.

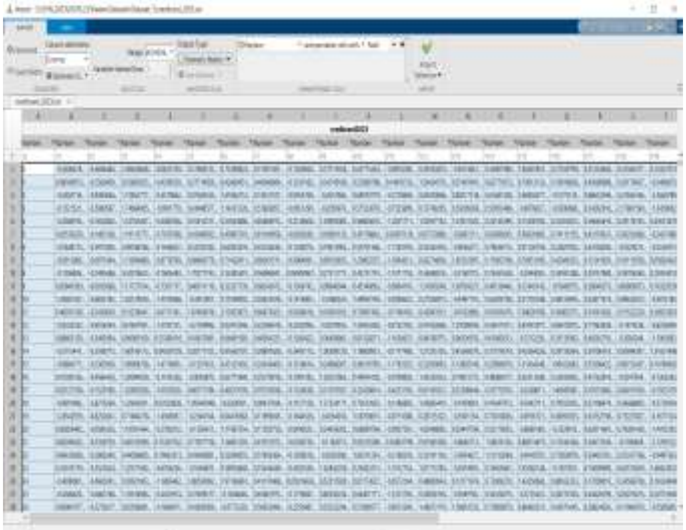

Fig.3 Importing raw data to MATLAB workspace

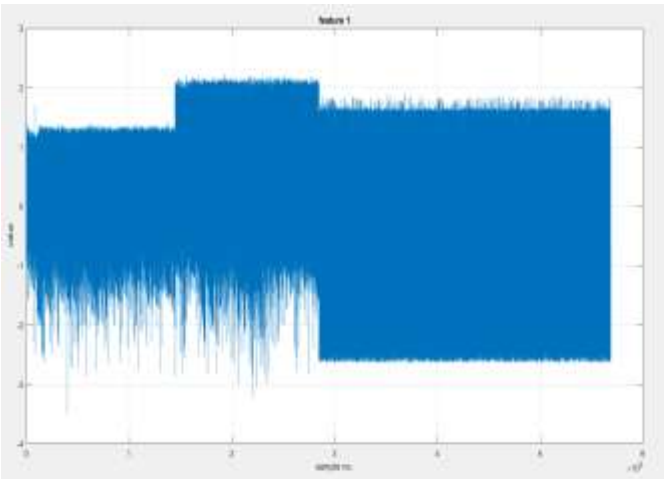

Fig.4 Variation in Feature 1 of raw dataset

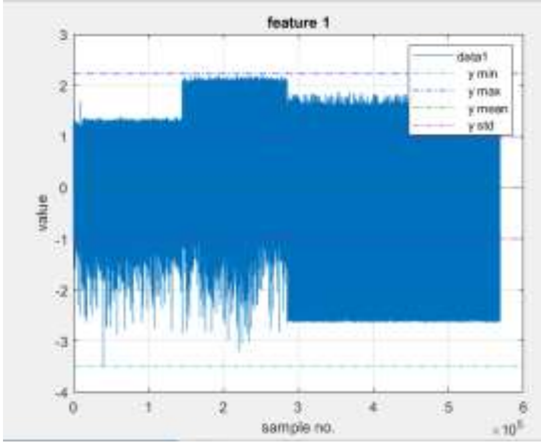

Fig.5 Statistical features of Feature 1

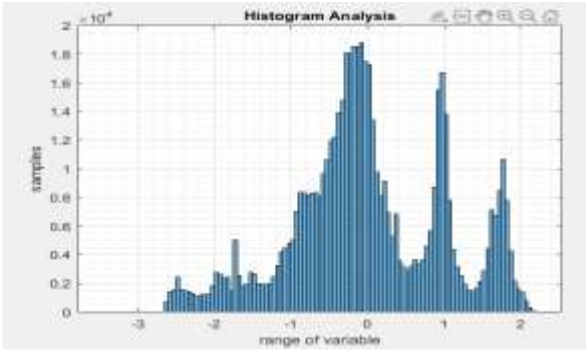

Fig.6 Histogram Analysis for Feature 1

Table 1. Statistical Features of Data

S.No.	Parameter	Value
1	Minimum	-3.496
2	Maximum	2.229
3	Mean	1.885×10^{-15}
4	Standard Deviation	1

Table 1 depicts the statistical values of feature 1.

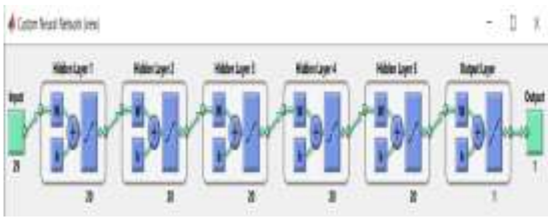

Fig.7 Network Visualization

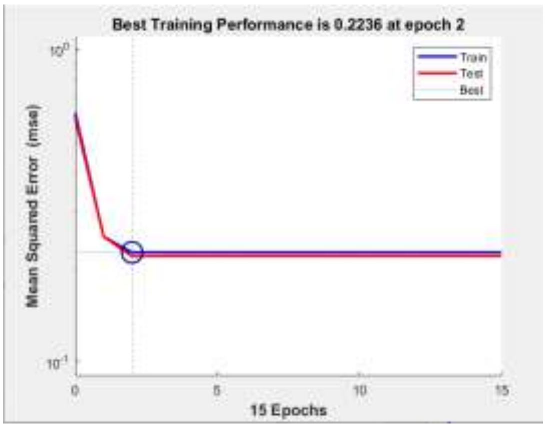

Fig.8 MSE to Convergence

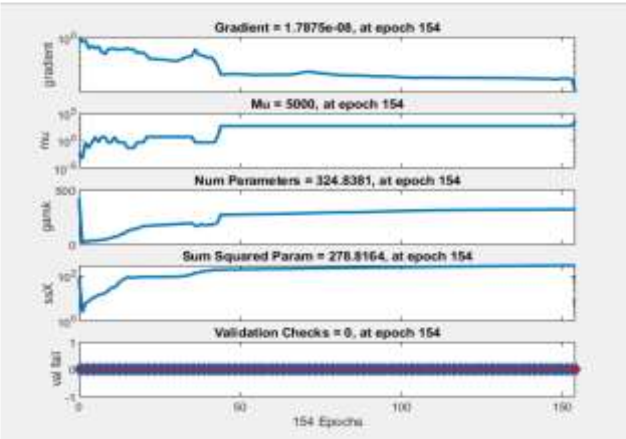

Fig.9 Training States

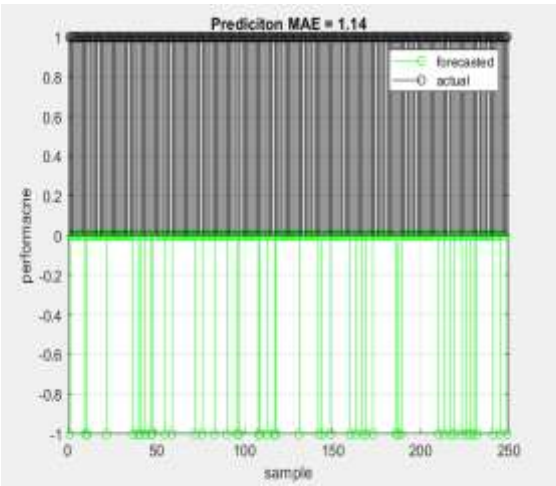

Fig.10 MAE obtained.

It can be observed that the proposed work attains an MSE of 0.22 and MAE of 1.14 at convergence which depicts the accurate classification capability of the proposed work.

Table 2 Results

S.No	Parameter	Value
1	Dataset	httpswww.kaggle.comdatasetsnelgiriyewithanacredit-card-fraud-detection-dataset-2023
2	Model	Deep Neural Network
3	Algorithm	Bayesian Regularization
4	MSE at convergence	0.224
5	MAE at convergence	1.14
6	Hidden Layers	5
7	Neurons in each layer	20
8	Variables(features)	29

The approach attains higher classification accuracy compared to baseline approaches [1].

CONCLUSION: Previous discussion conclude that the need for machine learning and deep learning approaches in credit card fraud detection arises from the increasing complexity, volume, and dynamic nature of fraudulent activities. Their capabilities in handling massive datasets, learning complex patterns, adapting to evolving threats, performing real-time detection, and improving prediction accuracy make them indispensable tools for modern financial security systems. As digital transactions continue to expand worldwide, intelligent fraud detection systems based on machine learning and deep learning are expected to play an increasingly crucial role in safeguarding electronic payment ecosystems. The proposed deep learning model with regularization attains higher accuracy of classification compared to existing work in the domain.

References

[1] Y. Wang, "A Data Balancing and Ensemble Learning Approach for Credit Card Fraud Detection," IEEE Transactions on Dependable And Secure Computing, vol. 3, no. 4, 2025, pp. 386-390

[2] J. Wang, J. Liu, W. Zheng and Y. Ge, "Temporal Heterogeneous Graph Contrastive Learning for Fraud Detection in Credit Card Transactions," in IEEE Access, 2024, vol. 13, pp. 145754-145771

[3] E. Esenogho, I. D. Mienye, T. G. Swart, K. Aruleba and G. Obaido, "A Neural Network Ensemble With Feature Engineering for Improved Credit Card Fraud Detection," in IEEE Access, vol. 10, pp. 16400-16407, 2023

[4] KS Adewole, NB Anuar, A Kamsin, “SMSAD: a framework for spam message and spam account detection”, Journal of Multimedia Tools and Applications, Springer 2022, vol. 78, pp. 78, 3925–3960.

[5] Aliaksandr Barushka, Petr Hajek, “Spam filtering using integrated distribution-based balancing approach and regularized deep neural networks”, Springer 2020

[6] Surendra Sedhai, Aixin Sun, “Semi-Supervised Spam Detection in Twitter Stream”, IEEE 2019

[7] Chao Chen, Yu Wang, Jun Zhang, Yang Xiang, Wanlei Zhou, Geyong Min, “Statistical Features-Based Real-Time Detection of Drifted Twitter Spam”, IEEE 2018

[8] Nida Mirza, Balkrishna Patil ,Tabinda Mirza ,Rajesh Auti, “Evaluating efficiency of classifier for email spam detector using hybrid feature selection approaches”,IEEE 2017

[9] Hammad Afzal ,Kashif Mehmood, “Spam filtering of bi-lingual tweets using machine learning”,IEEE 2016

[10] Hailu Xu ,Weiqing Sun ,Ahmad Javaid,” Efficient spam detection across Online Social Networks”, IEEE 2016

[11] Nadir Omer Fadl Elssied,Othman Ibrahim ,Ahmed Hamza Osman,” Enhancement of spam detection mechanism based on hybrid kkkk-mean clustering and support vector machine”,SPRINGER 2015

[12] Tarjani Vyas , Payal Prajapati , Somil Gadhwal,” A survey and evaluation of supervised machine learning techniques for spam e-mail filtering”,IEEE 2015

[13] Nishtha Jatana ,Kapil Sharma,” Bayesian spam classification: Time efficient radix encoded fragmented database approach”, IEEE 2014

[14] Kamalanathan Kandasamy ,Preethi Koroth,” An integrated approach to spam classification on Twitter using URL analysis, natural language processing and machine learning techniques”, IEEE 2014

[15] Navneel Prasad ,Rajeshni Singh ,Sunil Pranit Lal,” Comparison of Back Propagation and Resilient Propagation Algorithm for Spam Classification”,IEEE 2013

[16] Wojciech IndykEmail author, Tomasz Kajdanowicz, Przemyslaw Kazienko,Slawomir Plamowski,” Web Spam Detection Using MapReduce Approach to Collective Classification”, SPRINGER 2013

[17] Ashwin Rajadesingan, Anand Mahendran,” Comment Spam Classification in Blogs through Comment Analysis and Comment-Blog Post Relationships”, Springer 2016.

[18] Lauret, P., Fock, E., Randrianarivonyh, R.N., Manicom-Ramsamy, J.-F. (2008), Bayesian neural network approach to short time load forecasting. Energy Convers. Managagement, vol.49, no.5. pp.1156–1166.

[19] E. Esenogho, I. D. Mienye, T. G. Swart, K. Aruleba and G. Obaido, "A Neural Network Ensemble With Feature Engineering for Improved Credit Card Fraud Detection," in IEEE Access, vol. 10, pp. 16400-16407, 2022.

[20] R. Sailusha, V. Gnaneswar, R. Ramesh, and G. R. Rao, “Credit Card Fraud Detection Using Machine Learning,” in 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, May 2020, pp. 1264–1270.

[21] A. Shivanna, S. Ray, K. Alshouiliy, and D. P. Agrawal, “Detection of Fraudulence in Credit Card Transactions Using Machine Learning on Azure ML,” in 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, Oct. 2020, pp. 268–273.

[22] X. Jia, J. Yang, R. Liu, X. Wang, S. D. Cotofana, and W. Zhao, “Efficient Computation Reduction in Bayesian Neural Networks Through Feature Decomposition and Memorization,” IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 4, pp. 1703–1712, Apr. 2021

[23] J. Zhang, Y. Hua, Z. Xue, T. Song, C. Zheng, R. Ma, and H. Guan, “Spike-and-Slab Shrinkage Priors for Structurally Sparse Bayesian Neural Networks,” IEEE Transactions on Neural Networks and Learning Systems, vol. 36, no. 6, pp. 11176–11188, 2025