CREDIT CARD FRAUD DETECTION

ABSTRACT:

This work describes a unique method for detecting credit card fraud that combines an Auto-Encoder (AE) model for dimensionality reduction and t-distributed Stochastic Neighbor Embedding (tSNE) for visualization. The credit card transaction dataset is compressed into a 10-dimensional latent space by the AE model, which was created with a special architecture. K-means clustering is used to construct reference labels using features retrieved from the AE's bottleneck layer, which aids in the identification of probable clusters. With a series of layers, the AE architecture effectively compresses the credit card transaction information into a 10-dimensional latent space, catching tiny patterns suggestive of fraudulent activity. Using K-means clustering on the AE-extracted features makes it easier to identify unique clusters, which serve as the foundation for reference labeling. Following the application of tSNE to the 10-dimensional feature space, a 2D representation for visualizing clusters and anomalies is obtained. The "Class" column evaluation yields promising results, indicating the system's potential for improved fraud detection when compared to existing methods. The suggested methodology combines the benefits of AE and tSNE to improve anomaly identification in credit card transactions, providing a robust and interpretable solution to the problems faced by fraudulent activities.

KEYWORDS: AE, tSNE, Clusters, k-means, accuracy, precision-recall, ROC, F1-score.

INTRODUCTION:

A) PROBLEM DEFINITION:

The problem addressed in this paper is the requirement for a sophisticated credit card fraud detection system capable of navigating the intricacies of highly skewed and multivariate transaction datasets. Traditional approaches frequently fail to capture the subtle patterns associated with fraudulent behavior. The goal is to create a comprehensive solution that combines the power of Auto-Encoder (AE) modeling and t-distributed Stochastic Neighbor Embedding (tSNE) to reduce dimensionality and offer a visual representation of credit card transactions. The main challenge is developing a system that not only accurately compresses the feature space into a 10-dimensional latent representation using AE, but also uses tSNE for clustering visualization, allowing the detection of potentially fraudulent clusters. The system's effectiveness is measured using metrics such as accuracy, precision, recall, and F1-score, with the "Class" column serving as a reliable measure of fraud. This study tackles a key gap in current fraud detection approaches, with the goal of providing a comprehensive solution for the financial industry to improve security measures and safeguard against fraudulent credit card transactions.

Aim of this project:

Apply the Auto-Encoder (AE) model on the credit card transaction (creditcard.csv) dataset to reduce the dimensionality to 10-dimensional space. Then, apply the tSNE feature embedding method on the 10-dimensional feature space for clustering and visualize it in 2D embedding.

B) BACKGROUND STUDY:

The current credit card fraud detection landscape is characterized by a continual requirement for sophisticated approaches that can effectively navigate the intricate patterns of fraudulent actions inside high-dimensional and imbalanced transaction datasets. Traditional methods frequently fall short of adjusting to the dynamic nature of fraud, demanding a theoretical investigation of novel solutions. Auto-Encoders (AE) are a promising neural network paradigm that excels in feature embedding and dimensionality reduction. t-distributed Stochastic Neighbor Embedding (tSNE) is also a powerful method for displaying complex data in smaller dimensions. This paper provides a theoretical framework that combines AE for feature extraction and tSNE for visualization, with the goal of uncovering latent patterns suggestive of credit card fraud. The theoretical foundation is based on the premise that fraudulent transactions exhibit different patterns that can be represented successfully in a lower-dimensional environment. The approach combines AE and tSNE to improve sensitivity to aberrant patterns while addressing the uneven nature of credit card datasets, providing a theoretically grounded answer to the increasing issues of fraud detection in financial transactions.

RELATED WORKS:

1. **CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING ALGORITHMS by Ernst Kussul, Tatiana Baidyk.**

   The primary goal of this article is to build and create a novel fraud detection algorithm for Streaming Transaction Data, with the goal of analyzing customers' prior transaction details and extracting behavioral patterns. Cardholders are divided into groups based on the volume of their transactions. Then, using the sliding window approach aggregate the transactions done by cardholders from distinct groups in order to determine the behavioral patterns of the groups.

2. **A MACHINE LEARNING BASED CREDIT CARD FRAUD DETECTION USING THE GA ALGORITHM FOR FEATURE SELECTION by Emmanuel Ileberi, Yanxia Sun, Zenghui Wang.**

   This work offers a credit card fraud detection engine based on machine learning (ML) that uses the genetic algorithm (GA) for feature selection. Following the selection of optimum features, the proposed detection engine employs the ML classifiers Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), Artificial Neural Network (ANN), and Naive Bayes (NB). The suggested credit card fraud detection engine is assessed using a dataset produced from European cardholders to validate its performance. The results showed that the proposed method outperforms existing systems.

3. **ENHANCED CREDIT CARD FRAUD DETECTION BASED ON ATTENTION MECHANISM AND LSTM DEEP MODEL by Ibtissam Benchaji, Samira Douzi, Bouabid El Ouahidi, Jaafar Jaafari.**

   The goal of this research is to create a unique system for credit card fraud detection based on sequential data modeling and LSTM deep recurrent neural networks. In comparison to earlier studies, the proposed model considers the sequential nature of transactional data and allows the classifier to identify the most critical transactions in the input sequence that predict fraudulent transactions with more accuracy. To be more specific, model's robustness is built by combining the strengths of three sub-methods: uniform manifold approximation and projection (UMAP) for selecting the most useful predictive features, Long Short-Term Memory (LSTM) networks for incorporating transaction sequences, and the attention mechanism to improve LSTM performances. model's experiments yield impressive outcomes in terms of efficiency and efficacy.

C) OBJECTIVES AND CONTRIBUTION:

- **Data preprocessing:**

  Data preprocessing is used to convert data into clean data sets which are used for analysis. It consists of several steps like data cleansing, data reduction, data enrichment organization and transformation. So, it will be easy for machine learning models to read data and learn from data.

- **Descriptive statistics:**

  Input feature properties, minimum, maximum, mean, median, standard deviation, as well as details of any data augmentation techniques utilized, are all included in descriptive statistics. This data serves as the foundation for understanding the dataset's structure, distribution, and diversity, which are all important for effective model training and performance evaluation in credit card fraud detection.

- **Model Training:**

  We Apply the Auto-Encoder (AE) model on the credit card transaction (creditcard.csv) dataset to reduce the dimensionality to 10-dimensional space. Then, apply the tSNE feature embedding method on the 10-dimensional feature space for clustering and visualize it in 2D embedding.

- **Model Evaluation:**

  Model Evaluation is done by calculating different performance evaluation metrics like Accuracy, F1-score, ROC Curve, Precision-Recall curve. This can be done using test dataset.

METHODOLOGY:

- Loading of dataset: In this step, we will load dataset using csv file.

- Splitting of data: In this step we will divide the entire dataset into two groups, training set and testing set. Training set is used to train the model and learn from it. Testing data is used to evaluate model performance.

- EDA: In this step we will visualize features to describe features of the image. We will understand the shape of the train and test dataset.

- Data preprocessing: In data preprocessing we will do “random under sampling” which consists of removing data in order to have a more balanced dataset and thus avoiding our models to overfitting.

- Model creation: In this project we will apply the Auto-Encoder (AE) model on the credit card transaction (creditcard.csv) dataset to reduce the dimensionality to 10-dimensional space. Then, apply the tSNE feature embedding method on the 10-dimensional feature space for clustering and visualize it in 2D embedding.

- Model training: Model training is done by using training data. The model will do clustering, and it will give us the predicted labels.

- Model Evaluation: Model evaluation is done using class labels. Using these labels, we will calculate different performance evaluating features like Accuracy, F1-Score, ROC-Curve, Precision-Recall Curve.

- Analysis of results: The performance evaluating features give us whether the model we created is effective or not.

MODEL DESCRIPTION:

- **Auto-Encoder (AE) model to reduce the dimensionality to 10-dimensional space:**

  The Autoencoder (AE) model is a neural network architecture for unsupervised learning and dimension reduction. It is made up of an encoder, which maps input data to a lower-dimensional latent space, and a decoder, which reconstructs the original input. The model minimizes a reconstruction loss during training, improving the encoder and decoder to capture critical characteristics efficiently. The AE's latent space is a compact representation of input data that is commonly utilized for tasks such as dimensionality reduction and anomaly detection. Its adaptability and capacity to learn meaningful representations make it a useful tool in a wide range of applications, from image processing to generative modeling. Because of its ability to learn hierarchical representations from unlabeled data, autoencoders (AEs) have acquired importance in a variety of domains. Encoder-decoder architectures, which are frequently implemented with rectified linear units (ReLU) or hyperbolic tangent activations, allow for fast feature extraction and reconstruction. Dimensionality reduction is a fundamental application in which AEs may extract essential information from high-dimensional inputs. They also excel in anomaly identification by detecting deviations in the rebuilt output. Variants such as Variational Autoencoders (VAEs) include probabilistic features, which improve generative capabilities for jobs such as image creation. Training, hyperparameter optimization, regularization approaches, and careful architectural design are all important for peak performance. In conclusion, autoencoders are effective tools for unsupervised learning, providing flexible solutions for representation learning, data reduction, and generative modeling.

❖ **K-means clustering algorithm to generate reference labels:**

The K-means clustering algorithm is a dataset partitioning algorithm that divides a dataset into K separate, non-overlapping groups or clusters. The model assigns data points to clusters iteratively based on their resemblance to the centroid of the cluster, which is the mean of the data points in that cluster. The procedure begins with a random placement of K centroids, then alternates between assignment and recalculation of centroids until convergence is reached. The goal is to minimize the sum of squared distances between data points and cluster centroids. K-means is a simple and computationally efficient algorithm that may be used for a variety of tasks such as customer segmentation, image compression, and anomaly identification. It does, however, presume spherical and equal-sized clusters, and the initial centroids can influence the outcomes. For the model to capture meaningful patterns in the data, the optimal number of clusters (K) and preprocessing processes must be carefully considered.

❖ **t-SNE model for clustering and visualization:**

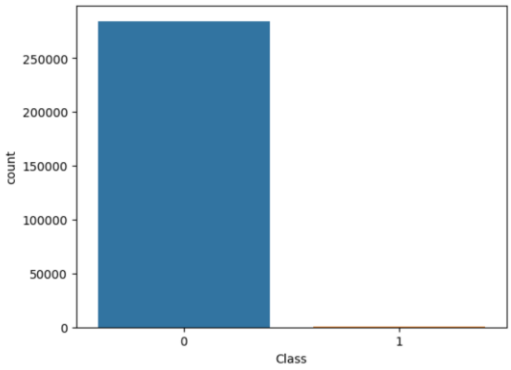
The t-Distributed Stochastic Neighbor Embedding (t-SNE) model is a dimensionality reduction technique that is frequently used for visualizing high-dimensional data in lower-dimensional space. It is especially successful at capturing complicated relationships while keeping local structure. t-SNE, which was developed to solve the shortcomings of existing techniques such as PCA, focuses on keeping pairwise similarities between data points in the original and lower-dimensional spaces. It uses a probabilistic approach to model similarities with conditional probability. The optimization procedure emphasizes the preservation of close points while minimizing the divergence between the original distribution and the distribution in lower-dimensional space. t-SNE is a popular method for exploratory data analysis, pattern detection, and feature visualization because it is particularly useful in displaying clusters and patterns in data. However, its hyperparameter sensitivity and potential for overfitting on small datasets demand careful parameter tweaking for best results.

EXPERIMENT AND RESULTS:

A) DATABASE:

Credit Card Database:

Credit card is a transaction dataset where anomalies are defined as fraudulent transactions. The data set is highly skewed, consisting of 492 frauds in a total of 284,807 examples, which results in only 0.172% fraud cases. The sparse number of fraudulent transactions justifies
this skewed set. The number of features is 28.




```
No Frauds 99.83 % of the dataset
Frauds 0.17 % of the dataset
```

Descriptive statistics:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9
count	284806.000000	284806.000000	2.848060e+05	284806.000000	284806.000000	2.848060e+05	284806.000000	284806.000000	284806.000000	284806.000000
mean	94813.585781	0.000002	6.661837e-07	-0.000002	0.000002	4.405008e-08	0.000002	-0.000006	0.000001	-0.000002
std	47488.004530	1.958699	1.651311e+00	1.516257	1.415871	1.380249e+00	1.332273	1.237092	1.194355	1.098634
min	0.000000	-56.407510	-7.271573e+01	-48.325589	-5.683171	-1.137433e+02	-26.160506	-43.557242	-73.216718	-13.434066
25%	54201.250000	-0.920374	-5.985522e-01	-0.890368	-0.848642	-6.915995e-01	-0.768296	-0.554080	-0.208628	-0.643098
50%	84691.500000	0.018109	6.549621e-02	0.179846	-0.019845	-5.433621e-02	-0.274186	0.040097	0.022358	-0.051429
75%	139320.000000	1.315645	8.037257e-01	1.027198	0.743348	6.119267e-01	0.398567	0.570426	0.327346	0.597140
max	172788.000000	2.454930	2.205773e+01	9.382558	16.875344	3.480167e+01	73.301626	120.589494	20.007208	15.594995

8 rows × 31 columns

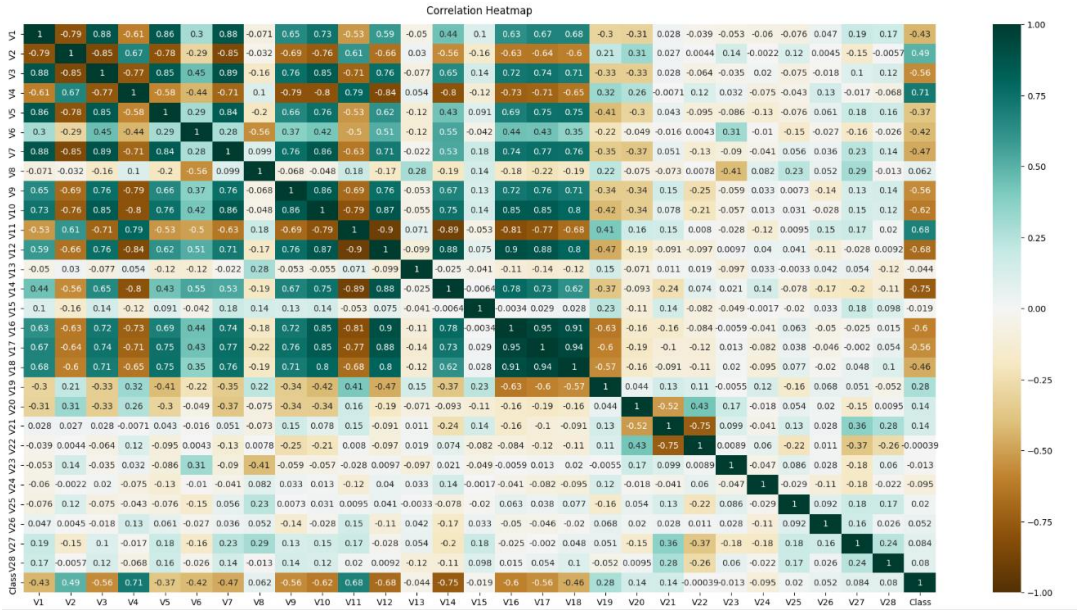
Random Under sampling:

```
new_df.shape
```

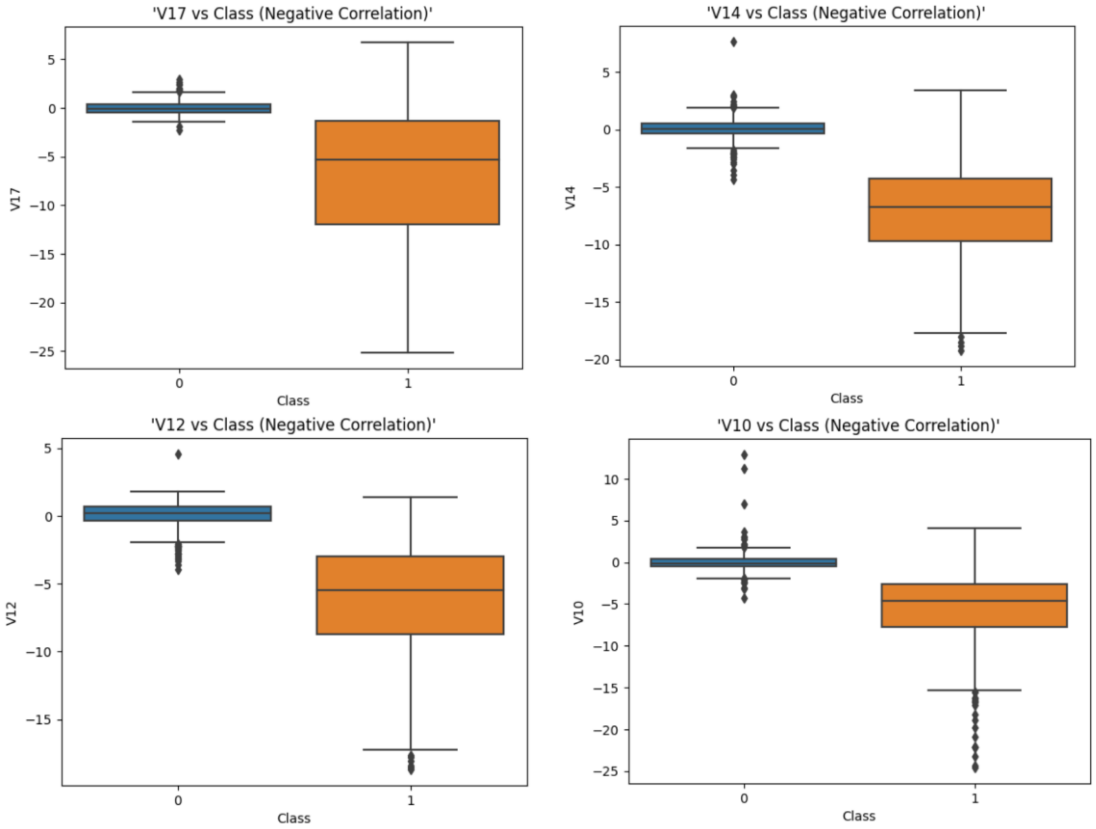
```
(984, 29)
```

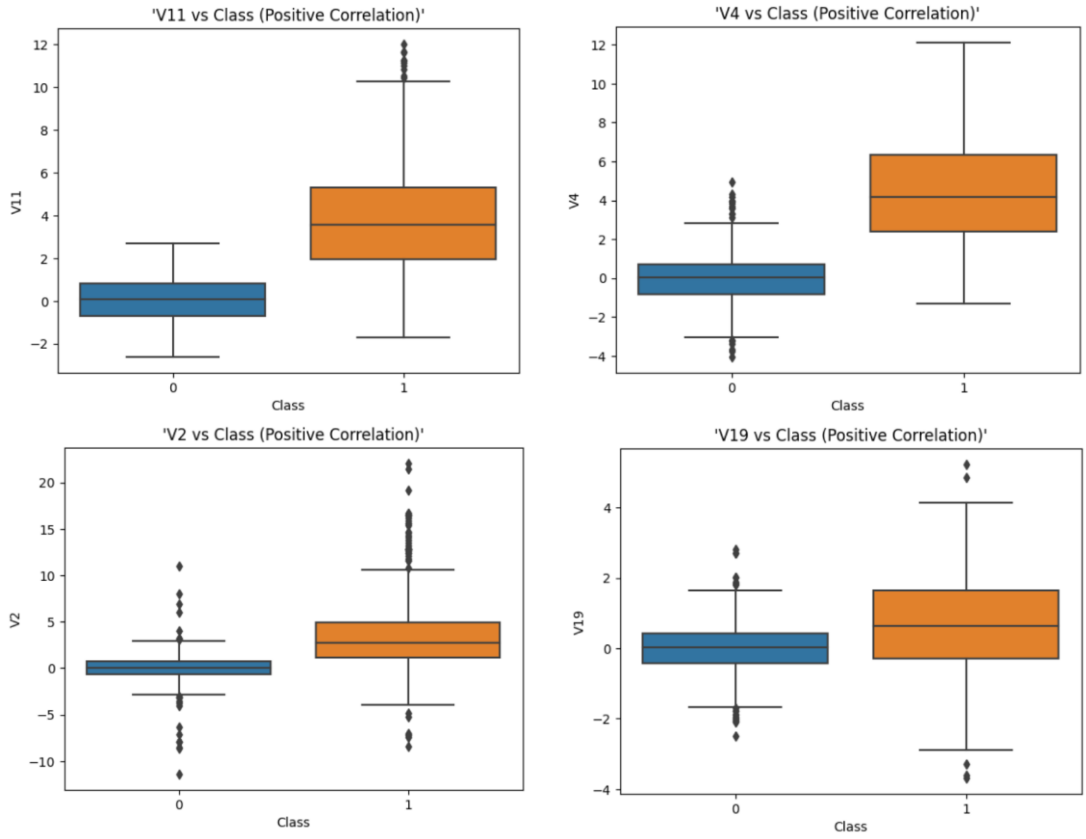
```
Distribution of the Classes in the subsample dataset
0    0.5
1    0.5
Name: Class, dtype: float64
```

Correlation:



Exploratory data analysis (EDA):





B) TRAINING AND TESTING LOGS:

❖ Auto-Encoder (AE) model to reduce the dimensionality to 10-dimensional space

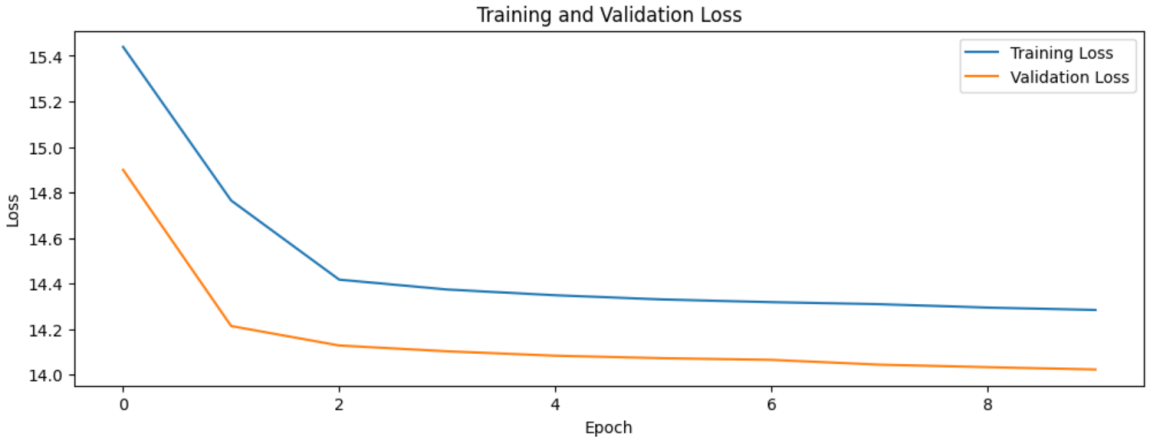
Model Summary:

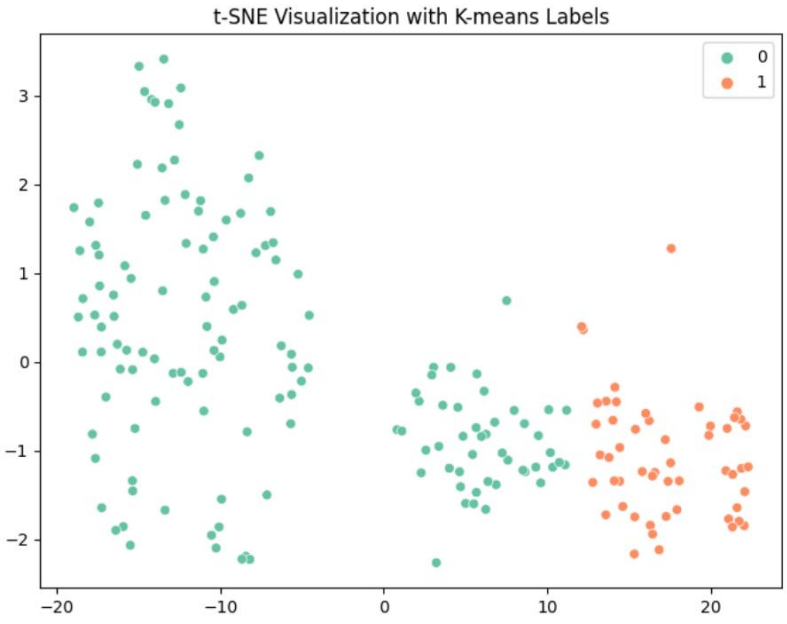
```
Model: "model_12"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_13 (InputLayer)       [(None, 28)]              0

 dense_48 (Dense)            (None, 24)                696

 dense_49 (Dense)            (None, 20)                500

 dense_50 (Dense)            (None, 15)                315

 dense_51 (Dense)            (None, 10)                160

 dense_52 (Dense)            (None, 15)                165

 dense_53 (Dense)            (None, 20)                320

 dense_54 (Dense)            (None, 24)                504

 dense_55 (Dense)            (None, 28)                700

=================================================================
Total params: 3360 (13.12 KB)
Trainable params: 3360 (13.12 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Computational Time:

```
Epoch 1/10
25/25 [==============================] - 3s 20ms/step - loss: 15.4389 - val_loss: 14.8990
Epoch 2/10
25/25 [==============================] - 0s 11ms/step - loss: 14.7648 - val_loss: 14.2137
Epoch 3/10
25/25 [==============================] - 0s 8ms/step - loss: 14.4174 - val_loss: 14.1283
Epoch 4/10
25/25 [==============================] - 0s 9ms/step - loss: 14.3741 - val_loss: 14.1029
Epoch 5/10
25/25 [==============================] - 0s 8ms/step - loss: 14.3489 - val_loss: 14.0835
Epoch 6/10
25/25 [==============================] - 0s 10ms/step - loss: 14.3303 - val_loss: 14.0726
Epoch 7/10
25/25 [==============================] - 0s 8ms/step - loss: 14.3186 - val_loss: 14.0653
Epoch 8/10
25/25 [==============================] - 0s 8ms/step - loss: 14.3096 - val_loss: 14.0443
Epoch 9/10
25/25 [==============================] - 0s 8ms/step - loss: 14.2949 - val_loss: 14.0332
Epoch 10/10
25/25 [==============================] - 0s 8ms/step - loss: 14.2844 - val_loss: 14.0231
```

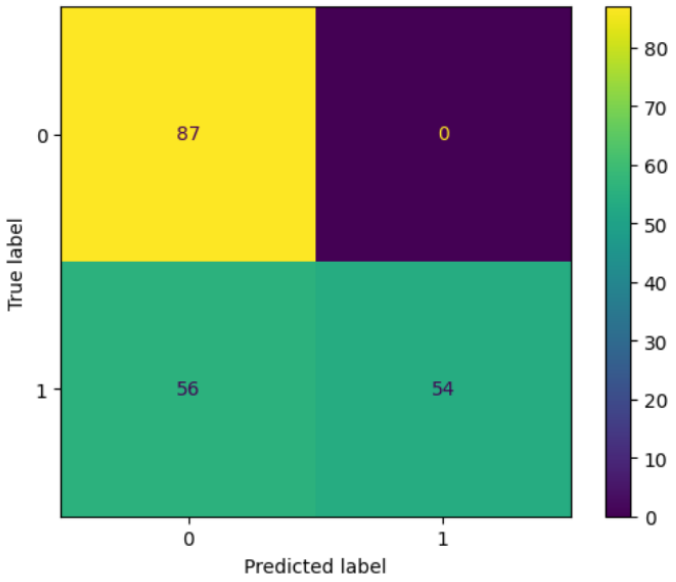
Training and Validation errors:





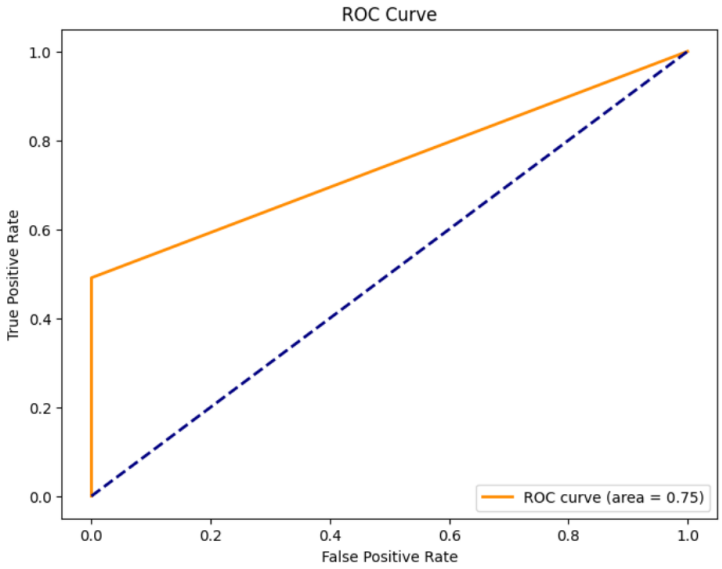
Performance metrics:

```
Accuracy: 0.7157
F1 Score: 0.6585
ROC AUC Score: 0.7455
Silhouette Score: 0.6698
```

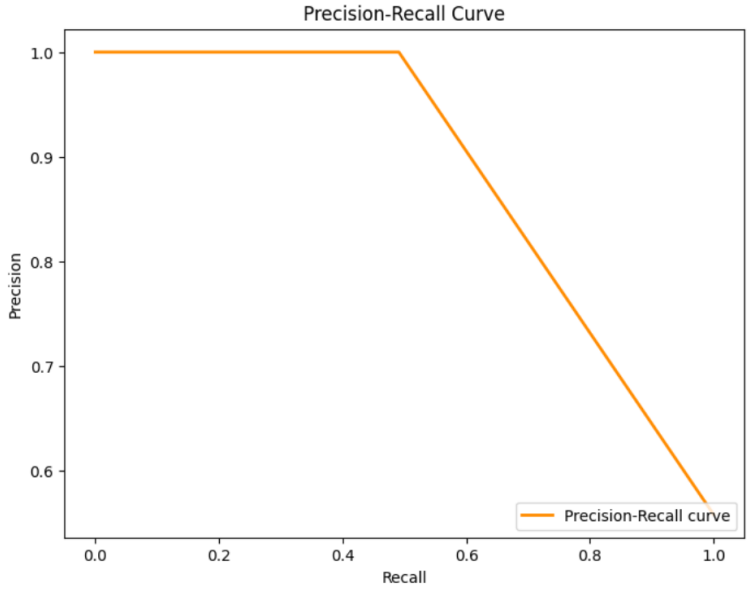


```
Classification Report for Test Set:
              precision    recall  f1-score   support

           0       0.61      1.00      0.76        87
           1       1.00      0.49      0.66       110

    accuracy                           0.72       197
   macro avg       0.80      0.75      0.71       197
weighted avg       0.83      0.72      0.70       197
```

ROC-Curve:



Precision-Recall Curve:



C) DISCUSSION AND COMPARISON:

The Auto-Encoder (AE) model effectively decreased the dimensionality of the credit card transaction dataset in the implemented approach, capturing crucial information in a 10-dimensional latent representation. Following that, K-means clustering produced discrete clusters, and t-distributed Stochastic Neighbor Embedding (t-SNE) visualization emphasized these clusters in a 2D space. The reduced reconstruction error demonstrated the AE model's performance, indicating a successful compression and reconstruction process. However, despite supplying reference labels, K-means clustering struggled to effectively categorize fraud and non-fraud transactions due to the extremely imbalanced dataset. The t-SNE visualization, on the other hand, revealed potential patterns and outliers while providing useful insights into the underlying data structure. The comparison with ground truth labels showed the challenge of grouping imbalanced datasets and suggested the need for more advanced techniques to improve fraud detection, such as anomaly detection. The thorough analysis highlights the trade-offs and complications involved in using dimensionality reduction, clustering, and visualization methods for credit card fraud detection in skewed datasets.

For credit card fraud detection using credit card dataset we implemented AE model on the dataset and applied k-means and tSNE method on it. Model worked well on the dataset. Accuracy, Precision, Recall, F1-Score, ROC-Curve, Precision-Recall curves, Silhouette score are the performance metrics of a model.

	Accuracy	**F1-score**	**ROC-AUC score**	**Silhouette Score**
Model	0.72	0.66	0.75	0.67

Table: Accuracy and Precision values of model

CONCLUSION:

In summary, the applied strategy that combined K-means clustering, t-distributed Stochastic Neighbor Embedding (t-SNE) visualization, and Auto-Encoder (AE) dimensionality reduction yielded insightful results for the credit card transaction dataset. The reduced reconstruction error proved the AE model's efficacy in capturing crucial elements and providing a compact representation. However, the difficulties associated with imbalanced data were visible in the K-means clustering results, underlining the need for more advanced fraud detection algorithms in such datasets. The t-SNE visualization was helpful in illuminating underlying data patterns and identifying potential anomalies. This paper emphasizes the nuances of credit card fraud detection, emphasizing the significance of careful model selection and evaluation procedures adapted to the dataset's individual properties. Future research could look into different clustering techniques and anomaly detection approaches to improve the robustness and accuracy of fraud detection in imbalanced dataset.

LIMITATIONS:

This study has certain limitations that should be acknowledged. To begin, the credit card transaction dataset's imbalanced nature, with a disproportionately sparse number of fraudulent incidents, presents hurdles for unsupervised learning approaches such as K-means clustering. The majority of non-fraudulent transactions influence the model's performance, potentially leading to poor clustering findings and reduced sensitivity to fraud detection. Furthermore, the use of a predetermined number of clusters in K-means may not correspond to the intrinsic complexity of the dataset. Furthermore, the success of t-distributed Stochastic Neighbor Embedding (t-SNE) in visualizing high-dimensional data is dependent on suitable parameter tuning, and the algorithm's sensitivity to varied perplexity levels influences the outcomes. Finally, while the Auto-Encoder (AE) model excels at capturing latent representations, its reliance on unsupervised learning may limit its capacity to detect subtle patterns indicative of fraud. Exploring alternate clustering methods, modifying models for imbalanced datasets, and fine-tuning parameters for superior t-SNE visualization could all help to address these constraints.

REFERENCES:

- Haoxiang, W.; Smys, S. *Overview of configuring adaptive activation functions for deep neural networks-a comparative study*. J. Ubiquitous Comput. Commun. Technol. 2021, 3, 10–22.

- Zhang, R.; Zheng, F.; Min, W. *Sequential behavioral data processing using deep learning and the Markov transition field in online fraud detection*. arXiv 2018, arXiv:1808.05329.

- Sun, W.; Yang, C.G.; Qi, J.X. *Credit risk assessment in commercial banks based on support vector machines*. In Proceedings of the 2006 International Conference on Machine Learning and Cybernetics, Dalian, China, 13–16 August 2006; pp. 2430–2433.

- Smys, S.; Raj, J.S. *Analysis of deep learning techniques for early detection of depression on social media network*-a comparative study. J. Trends Comput. Sci. Smart Technol. 2021, 3, 24–39.

- Thennakoon, A.; Bhagyani, C.; Premadasa, S.; Mihiranga, S.; Kuruwitaarachchi, N. *Real-time credit card fraud detection using machine learning.* In Proceedings of the 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 10–11 January 2019; pp. 488–493.

- Sailusha, R.; Gnaneswar, V.; Ramesh, R.; Rao, G.R. *Credit card fraud detection using machine learning.* In Proceedings of the 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 13–15 May 2020; pp. 1264–1270.

- Rtayli, N.; Enneya, N. *Enhanced credit card fraud detection based on SVM-recursive feature elimination and hyper-parameters optimization.* J. Inf. Secur. Appl. 2020, 55, 102596.

- Ileberi, E.; Sun, Y.; Wang, Z. *A machine learning based credit card fraud detection using the GA algorithm for feature selection.* J. Big Data 2022, 9, 24.

- Kim, E.; Lee, J.; Shin, H.; Yang, H.; Cho, S.; Nam, S.k.; Song, Y.; Yoon, J.A.; Kim, J.I. *Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning.* Expert Syst. Appl. 2019, 128, 214–224.

- Maniraj, S.; Saini, A.; Ahmed, S.; Sarkar, S. *Credit card fraud detection using machine learning and data science.* Int. J. Eng. Res. 2019, 8, 110–115.