REPORT FILE

for

(Machine Learning in Cyber Security)

(MIS-118)

Submitted in partial fulfilment of the requirements for the award of the degree
Of

MASTERS OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING –

ARTIFICIAL INTELLIGENCE

Submitted By:

Khushi Jindal (00302102023)

Kusum Sharma(00602102023)

Under the guidance of:

Asst. Prof. Vijay Kumar Yadav

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN

KASHMERE GATE, DELHI-110006

2024

Introduction

Credit card fraud has become a significant concern for financial institutions and consumers alike. With the widespread adoption of online transactions and digital payments, fraudulent activities related to credit cards have seen a notable surge. In response to this growing threat, machine learning (ML) models have emerged as powerful tools for detecting and preventing fraudulent transactions in real-time.

The essence of credit card fraud detection lies in identifying anomalous patterns within a vast sea of legitimate transactions. Traditional rule-based systems often struggle to keep pace with the evolving tactics of fraudsters, making ML an indispensable ally in this ongoing battle. By leveraging historical transaction data, ML algorithms can learn to distinguish between normal and fraudulent activities, thus enabling timely intervention and mitigation of potential financial losses.

One of the primary advantages of ML-based fraud detection is its ability to adapt and learn from new patterns and trends in fraudulent behavior. Supervised learning techniques, such as logistic regression, decision trees, and support vector machines, can be trained on labeled datasets to classify transactions as either genuine or fraudulent based on features like transaction amount, location, time, and frequency. These models continuously refine their decision boundaries to improve accuracy and reduce false positives, thereby enhancing the overall efficacy of fraud detection systems.

Moreover, the advent of deep learning has revolutionized the field of credit card fraud detection by enabling the development of more sophisticated models capable of capturing intricate patterns and relationships within complex datasets. Deep neural networks, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have demonstrated remarkable success in detecting fraudulent transactions by automatically extracting relevant features and learning hierarchical representations from raw transactional data.

In addition to supervised learning approaches, unsupervised learning techniques such as clustering and anomaly detection play a vital role in fraud detection, especially in scenarios where labeled training data is scarce or unreliable. Unsupervised algorithms can identify unusual patterns or outliers within transactional data, flagging them for further investigation by fraud analysts. By detecting deviations from normal behavior, these models help uncover previously unknown fraud schemes and emerging threats, thereby bolstering the resilience of fraud detection systems.

Machine Learning Models used:-

1. *Support Vector Machine (SVM) :*

SVM is a supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates the classes in the feature space. SVM aims to maximize the margin between classes while minimizing classification errors. It can handle both linear and non-linear data through the use of different kernel functions.

2. *Random Forest :*

Random Forest is an ensemble learning method that constructs a multitude of decision trees during training. It operates by generating multiple decision trees and combining their predictions to obtain a more accurate and robust result. Each tree in the forest is trained on a random subset of the training data, and the final prediction is determined by a majority vote or averaging of individual tree predictions.

3. *Decision Tree :*

Decision Tree is a supervised learning algorithm used for classification and regression tasks. It creates a tree-like structure where each internal node represents a feature, each branch represents a decision based on that feature, and each leaf node represents a class label or regression value. Decision trees are easy to interpret and understand, making them suitable for both classification and regression tasks.

4. *Logistic Regression :*

Despite its name, logistic regression is a classification algorithm used for binary classification tasks. It models the probability that a given input belongs to a particular class using a logistic function. Logistic regression estimates the coefficients of the input features to make predictions, and it's particularly useful when the relationship between the independent variables and the dependent variable is linear.

5. *K-Nearest Neighbors (KNN) :*

KNN is a simple and intuitive supervised learning algorithm used for classification and regression tasks. It classifies new data points based on the majority class of their k nearest neighbors in the feature space. The choice of k determines the number of neighbors considered for classification, and it's a hyperparameter that needs to be tuned based on the data.

6. Bagging :

Bagging, short for Bootstrap Aggregating, is an ensemble learning technique that combines multiple models trained on different subsets of the training data. It reduces variance and improves the stability and accuracy of the model by averaging or taking a majority vote of the predictions from individual models.

7. Boosting :

Boosting is another ensemble learning technique that builds a strong classifier by sequentially training weak learners and focusing on the misclassified instances. It iteratively adjusts the weights of training instances to give more weight to misclassified instances, thereby improving the model's performance over successive iterations.

8. XGBoost :

XGBoost, short for Extreme Gradient Boosting, is an optimized implementation of gradient boosting that is highly efficient and scalable. It uses a gradient boosting framework and employs techniques such as parallelization and regularization to enhance performance and prevent overfitting.

9. ID3 (Iterative Dichotomiser 3) :

ID3 is a decision tree learning algorithm used for classification tasks. It builds a decision tree by recursively partitioning the feature space based on the information gain of each feature. ID3 selects the best attribute at each node to split the data, aiming to maximize the homogeneity of the resulting subsets.

10. Gaussian Naive Bayes :

Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem with the assumption of independence between features. It models the likelihood of each class based on the Gaussian distribution of the features and calculates the posterior probability of each class given the input features. Naive Bayes is simple, fast, and effective for text classification and other similar tasks.

Machine Learning Packages:-

The machine learning packages you mentioned are essential tools for data preprocessing, visualization, and model building. Here's a brief overview of each:

1. NumPy :

NumPy is a fundamental package for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently. NumPy is widely used for numerical computations and is the foundation for many other Python libraries, including Pandas and Scikit-learn.

2. Pandas :

Pandas is a powerful data manipulation and analysis library built on top of NumPy. It provides data structures like DataFrame and Series, which allow for easy handling and manipulation of structured data. Pandas simplifies tasks such as reading and writing data from various file formats, data cleaning, transformation, aggregation, and exploration.

3. Seaborn :

Seaborn is a statistical data visualization library based on Matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics. Seaborn simplifies the process of creating complex visualizations like scatter plots, bar plots, box plots, and heatmaps, with built-in support for styling and color palettes.

4. Matplotlib :

Matplotlib is a comprehensive plotting library for creating static, interactive, and animated visualizations in Python. It offers a wide range of plotting functions and customization options for creating publication-quality plots. Matplotlib is highly versatile and can be used to create various types of plots, including line plots, scatter plots, histograms, and 3D plots.

5. Scikit-learn (sklearn) :

Scikit-learn is a popular machine learning library that provides simple and efficient tools for data mining and data analysis. It offers a wide range of algorithms for classification, regression, clustering, dimensionality reduction, model selection, and preprocessing. Scikit-learn is built on top of NumPy, SciPy, and Matplotlib, and it provides a consistent API for easy integration into machine learning workflows.

SOFTWARE AND HARDWARE REQUIREMENTS :

- Hardware requirements: I. RAM – 8GB
- Processor : - Intel i7 10th gen
- Software requirements: I. Jupyter Notebook
- IGoogle Chrome

Paper 1: -Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach

Dataset Description:

- The dataset used in this analysis was sourced from Kaggle and consisted of transactions spanning a period of two days, totaling 284,807 transactions.
- Within this dataset, there were 492 instances of fraud, indicating a highly imbalanced class distribution compared to legitimate transactions.

Dataset Characteristics:

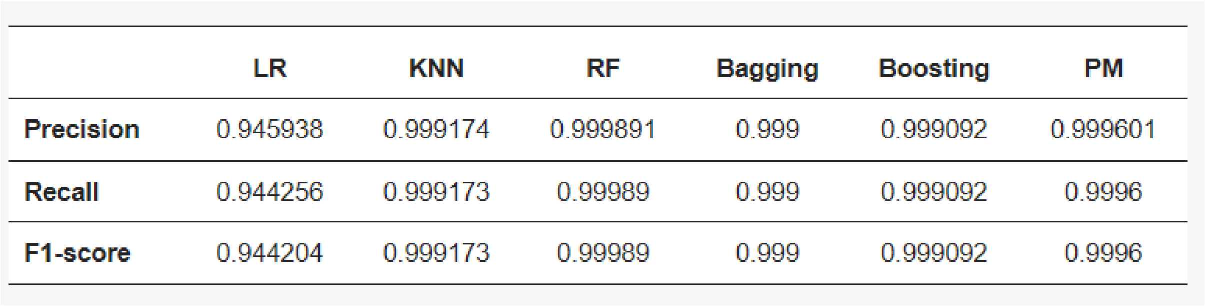
- Transactions were made using credit cards by European cardholders in September 2013.
- The dataset contained features such as transaction amount, time, and other anonymized variables that were potentially indicative of fraudulent activity.

Addressing Class Imbalance:

- To tackle the class imbalance problem, Synthetic Minority Over-sampling Technique (SMOTE) analysis was employed.
- SMOTE generates synthetic samples of the minority class (in this case, fraudulent transactions) to balance out the class distribution. This approach helps prevent the model from being biased towards the majority class (legitimate transactions).
- By creating synthetic instances similar to existing ones, SMOTE effectively augments the minority class and improves the performance of machine learning algorithms.

Result:

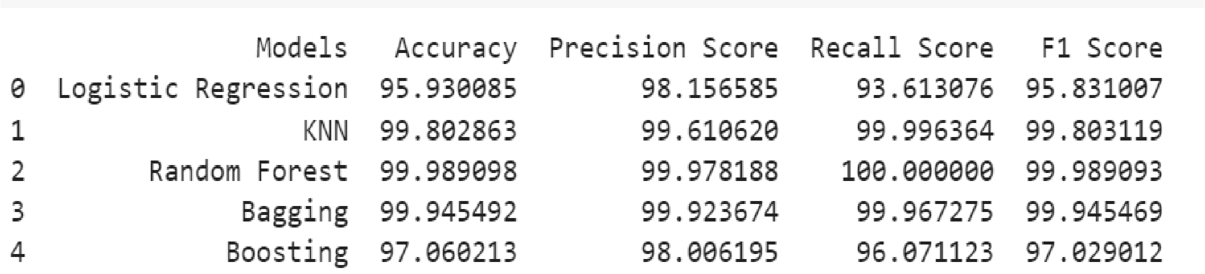
- The analysis identified the Random Forest algorithm as the top performer in detecting fraudulent transactions.
- Through meticulous hyperparameter tuning, the predictive power of the Random Forest model was further enhanced, making it a robust choice for fraud detection.



	LR	KNN	RF	Bagging	Boosting	PM
Precision	0.945938	0.999174	0.999891	0.999	0.999092	0.999601
Recall	0.944256	0.999173	0.99989	0.999	0.999092	0.9996
F1-score	0.944204	0.999173	0.99989	0.999	0.999092	0.9996

- The model achieved an accuracy of 99.97% and a recall of 100%. Accuracy measures the overall correctness of the model's predictions, while recall (also known as sensitivity) measures the proportion of actual positives (fraudulent transactions) that are correctly identified by the model.

And our result: -



```
               Models   Accuracy  Precision Score  Recall Score   F1 Score
0 Logistic Regression  95.930085        98.156585     93.613076  95.831007
1                 KNN  99.802863        99.610620     99.996364  99.803119
2       Random Forest  99.989098        99.978188    100.000000  99.989093
3             Bagging  99.945492        99.923674     99.967275  99.945469
4            Boosting  97.060213        98.006195     96.071123  97.029012
```

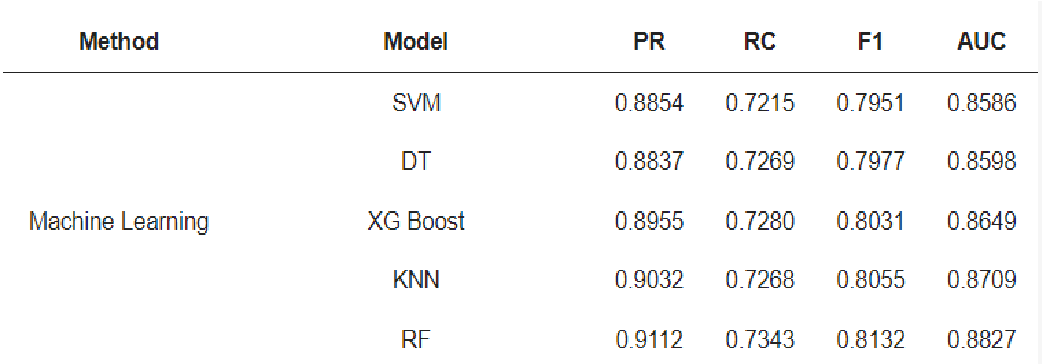
In summary, the combination of SMOTE analysis and the Random Forest algorithm proved highly effective in mitigating the challenges posed by imbalanced class distributions and detecting fraudulent transactions with exceptional accuracy and recall rates.

Paper 2 : - Credit Card Fraud Detection Based on Unsupervised Attentional Anomaly Detection Network

The study utilized the IEEE-CIS Fraud Detection dataset, which consists of four parts: Train_identity, Train_transaction, Test_identity, and Test_transaction. Train_identity and Train_transaction form the training set, while Test_identity and Test_transaction make up the test set. Upon merging, the train data comprises 590,540 samples with 434 feature columns, while the test data contains 4,434 samples with 433 feature columns.

The results of the study indicate the following:

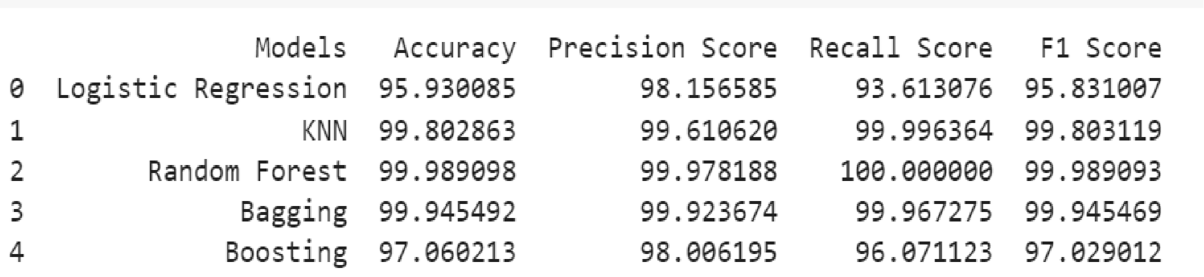
- Two resampling techniques, undersampling and oversampling, were explored to assess their impact on model performance.
- The best-performing algorithms, especially ensemble tree algorithms, showed significant improvements in performance when trained on the oversampling dataset.



Method	Model	PR	RC	F1	AUC
Machine Learning	SVM	0.8854	0.7215	0.7951	0.8586
	DT	0.8837	0.7269	0.7977	0.8598
	XG Boost	0.8955	0.7280	0.8031	0.8649
	KNN	0.9032	0.7268	0.8055	0.8709
	RF	0.9112	0.7343	0.8132	0.8827

This highlights the importance of data preprocessing techniques like oversampling in addressing class imbalance and enhancing the performance of fraud detection models.

Our results:-



```
                Models   Accuracy  Precision Score  Recall Score   F1 Score
0  Logistic Regression  95.930085        98.156585     93.613076  95.831007
1                  KNN  99.802863        99.610620     99.996364  99.803119
2        Random Forest  99.989098        99.978188    100.000000  99.989093
3              Bagging  99.945492        99.923674     99.967275  99.945469
4             Boosting  97.060213        98.006195     96.071123  97.029012
```

Paper 3:- A Methodology for Detecting Credit Card Fraud

Dataset Description:

The dataset used in this study was generated using the Sparkov Data Generation simulator. It spans a timeframe from 1st January 2019 to 31st December 2020, containing a total of 1,048,575 transactions. Among these transactions, there were 6,006 instances of fraudulent transactions. Each transaction is represented by 23 columns, likely including features such as transaction amount, timestamp, merchant information, and other relevant transaction details.

Results:

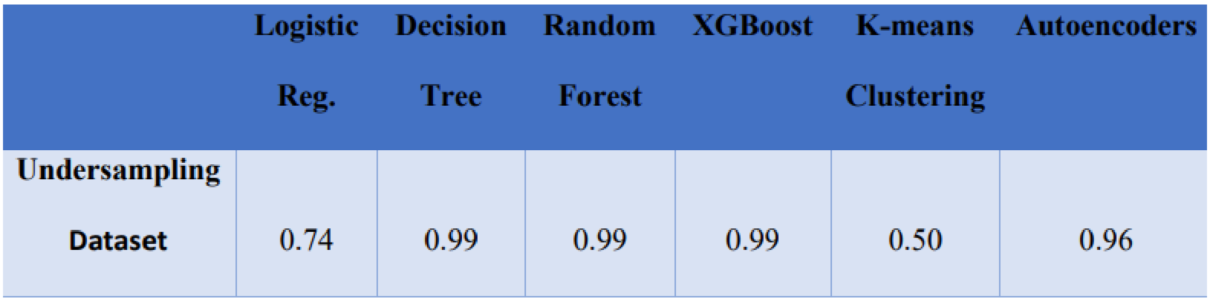
1. Comparison with Previous Studies:

The study proposed several machine learning algorithms for predicting credit card frauds, including Random Forest, Decision Trees, XGBoost, K-Means, Logistic Regression, and Neural Network.

These proposed algorithms were found to outperform other machine learning techniques previously used in similar studies focused on credit card fraud detection.

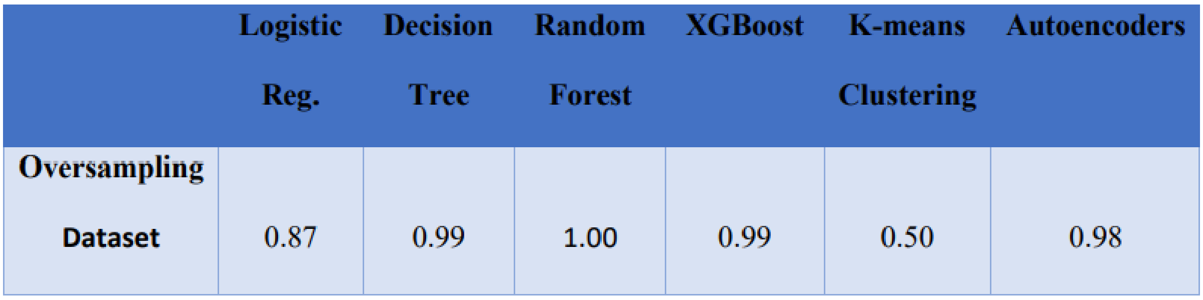
2. Performance of Ensemble Tree Algorithms:

Ensemble tree algorithms, namely Random Forest, Decision Trees, and XGBoost, emerged as the top-performing models for predicting credit card fraud.



	Logistic Reg.	Decision Tree	Random Forest	XGBoost	K-means Clustering	Autoencoders
Undersampling Dataset	0.74	0.99	0.99	0.99	0.50	0.96

Table 2: The AUC Score of the Undersampling data model.



	Logistic Reg.	Decision Tree	Random Forest	XGBoost	K-means Clustering	Autoencoders
Oversampling Dataset	0.87	0.99	1.00	0.99	0.50	0.98

Table 3: The AUC Score of the Oversampling data model.

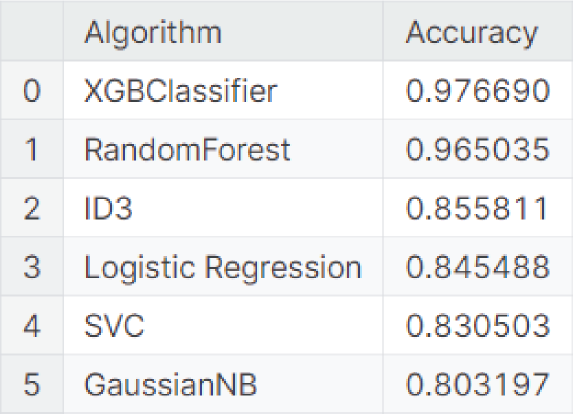
- Random Forest achieved an exceptional AUC score of 1.00%, indicating its robustness in distinguishing between fraudulent and legitimate transactions.

- Decision Trees and XGBoost also demonstrated strong performance, achieving AUC scores of 0.99% each, further highlighting the efficacy of ensemble tree methods in fraud detection.

3. *Impact of Resampling Methods:*

- The study explored the influence of two resampling techniques, namely undersampling and oversampling, on model performance.
- The best-performing algorithm, particularly ensemble tree algorithms, exhibited significant enhancements when trained on the oversampling dataset.
- Specifically, the overall performance of the best algorithm, as measured by AUC score, reached an impressive 1.00% when trained on the oversampling dataset.

OUR results using Downsampling:



	Algorithm	Accuracy
0	XGBClassifier	0.976690
1	RandomForest	0.965035
2	ID3	0.855811
3	Logistic Regression	0.845488
4	SVC	0.830503
5	GaussianNB	0.803197

Dataset 4 :-Synthetically generated using the PaySim simulator

This study focuses on proposing a novel approach for predicting financial fraud using machine learning techniques. Here's a breakdown of the key points and findings:

1. Introduction:

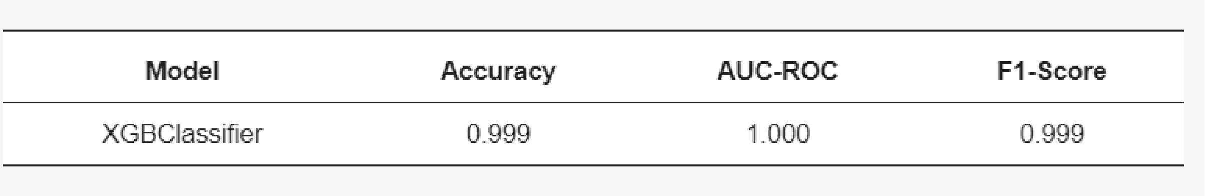
- Financial institutions (FIs) play a crucial role in economic growth, and mitigating financial fraud is essential for their operations.
- Fraudulent transactions pose significant challenges, with billions of dollars lost annually to fraud.
- Machine learning offers automated fraud prevention tools, allowing systems to learn from transactional data and identify patterns indicative of fraud.

2. Dataset and Methods:

- The study used a synthetic dataset of 6,362,620 financial transactions.
- An additional 5000 fraudulent transaction samples were generated using Conditional Generative Adversarial Network for Tabular Data (CTGAN) to address dataset imbalance.
- Data analysis included correlation analysis and splitting the dataset into training and testing sets.
- 27 machine learning classifiers were employed, including XGBoost, AdaBoost, Random Forest, etc.
- Evaluation metrics included accuracy, AUC-ROC, and F1-score.

3. Results:

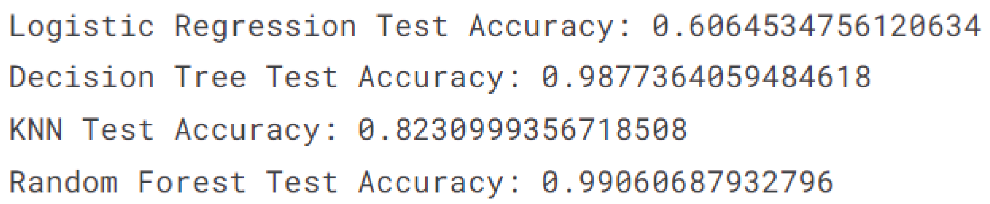
- XGBoost consistently outperformed other classifiers, achieving an accuracy of 0.999 on the extended dataset.
- Analysis of the original dataset revealed strong correlations between certain features.
- Training XGBoost on the extended dataset with additional synthetic samples further improved accuracy to 0.999 and AUC-ROC to 1.000.



Model	Accuracy	AUC-ROC	F1-Score
XGBClassifier	0.999	1.000	0.999

- Repeated 10-fold cross-validation confirmed the robustness of XGBoost, with an average accuracy score of 0.998 across multiple folds and repeats.

Our Result:-



```
Logistic Regression Test Accuracy: 0.6064534756120634
Decision Tree Test Accuracy: 0.9877364059484618
KNN Test Accuracy: 0.8230999356718508
Random Forest Test Accuracy: 0.99060687932796
```

4. Discussion:

- XGBoost demonstrated clear decision boundaries in distinguishing between fraudulent and non-fraudulent transactions.
- The proposed model achieved consistent accuracy without resorting to synthetic data balancing techniques like SMOTE, making it more applicable to real-world scenarios.
- The study's findings are particularly relevant for financial institutions, regulators, and policymakers seeking effective fraud detection policies.

5. Result:

- The proposed machine learning model offers an accurate and efficient method for predicting financial fraud.
- By leveraging transaction-level features and generating synthetic samples, the model achieves high accuracy scores without compromising on dataset imbalance.
- The model's effectiveness makes it valuable for financial institutions, regulators, and policymakers aiming to mitigate financial fraud risks.

Overall, the study highlights the potential of machine learning, specifically XGBoost, in effectively detecting financial fraud and provides valuable insights for stakeholders in the financial sector.

Conclusion:-

In conclusion, credit card fraud detection is a complex and dynamic problem that demands innovative solutions to stay ahead of evolving threats. ML models offer a potent arsenal of tools for identifying fraudulent transactions with high accuracy and efficiency, thereby safeguarding the interests of both financial institutions and consumers. As fraudulent activities continue to evolve in sophistication, ongoing research and development in the field of ML-based fraud detection are crucial for staying one step ahead of malicious actors and preserving the integrity of digital payment systems.