Government Engineering College

Sec-28 Gandhinagar

Certificate

This is to certify that

*Miss. **Jankeeben Kamleshbhai Dodiya** of class **ME/CE**. Enrollment No. **240130773004** has satisfactorily completed her Project work in this report titled **“Fraud Transaction Detection”** in **Mini Project with Seminar [ME02000141]** subject for the term ending in **May,2025**.*

Date: -

Signature of Faculty: -

Acknowledgment

I would like to express my deepest gratitude to everyone who contributed to the completion of this report on "Traffic Flow Prediction Using Machine Learning”. First and foremost, I sincerely thank my guide, Dr. Sanjaykumar Patel, for their invaluable guidance, constructive feedback, and unwavering support throughout this project. Their expertise and encouragement have been instrumental in shaping the direction and depth of this work.

I would also like to acknowledge my peers and colleagues for their thoughtful discussions, suggestions, and moral support during the challenging phases of this project. Their collaboration and shared insights were invaluable.

I express my heartfelt appreciation to all who have supported me directly or indirectly.

Thank you all.

Abstract

Fraud detection in financial transactions has become increasingly essential with the growing reliance on digital payment platforms and the sophistication of fraudulent schemes. This report presents a machine learning-based framework utilizing a stacked ensemble model that integrates Random Forest, XGBoost, and Logistic Regression classifiers to enhance detection accuracy. The model was trained and evaluated on a large real-world financial dataset, allowing for comprehensive learning across diverse transaction patterns. To further improve model performance, the study applies techniques such as Synthetic Minority Over-sampling Technique (SMOTE) for class balancing, feature selection to reduce redundancy, and cross-validation to ensure generalization. The layered architecture enables the ensemble to leverage the strengths of each base model, improving its ability to identify fraudulent activities effectively. This approach demonstrates that ensemble-based methods can significantly contribute to building robust, scalable, and intelligent fraud detection systems suitable for modern financial applications.

Table of Contents

List of Tables

Table of Figures

Chapter 1 - Introduction

1.1 What is Fraud Transaction?

A fraudulent transaction refers to any financial transaction that is executed with the intent to deceive or mislead, resulting in unauthorized access to funds or assets. The following features typically characterize these transactions:

1. **Deceptive Intent:** Fraudulent transactions are carried out to benefit the perpetrator at the expense of the victim. This can involve misrepresentation of information or identity.

2. **Unauthorized Access:** Such transactions often involve using stolen or unauthorized credentials, such as credit card information, bank account details, or personal identification.

3. **Anomalous Behavior:** Fraudulent transactions often deviate from normal transaction patterns. For instance, they may involve unusual amounts, locations, or frequencies that do not align with the typical behavior of the account holder[1].

4. **Types of Fraud:** Common types of fraudulent transactions include credit card fraud, identity theft, money laundering, and phishing scams. Each type employs different tactics to exploit vulnerabilities in financial systems[3].

5. **Detection Challenges:** Detecting fraudulent transactions poses significant challenges due to the evolving nature of fraud tactics. Machine learning and advanced analytical techniques are increasingly being employed to identify anomalies and patterns indicative of fraud .

6. **Impact on Financial Systems:** Fraudulent transactions can lead to substantial financial losses for individuals and institutions, undermine trust in financial systems, and necessitate the implementation of robust fraud detection and prevention measures[1][3].

1.2 Effect of Fraud Transaction

1. **Financial Loss:** • Victims of fraud often face significant financial losses, which can impact their savings, credit ratings, and overall financial stability. Financial institutions also incur costs related to fraud detection, prevention, and reimbursements to affected customers [5].
2. **Emotional and Psychological Impact:** • The experience of being a victim of fraud can lead to stress, anxiety, and a loss of trust in financial systems. This emotional toll can affect individuals' mental health and their willingness to engage in online transactions in the future [5].
3. **Reputation Damage:** • For financial institutions, fraudulent transactions can lead to reputational damage. Customers may lose trust in the institution's ability to protect their assets, leading to a loss of business and customer loyalty [1].
4. **Increased Security Measures:** • The prevalence of fraud often results in financial institutions implementing more stringent security measures, which can lead to increased operational costs and potential inconvenience for customers.
5. **Regulatory Consequences:** • Financial institutions may face regulatory scrutiny and penalties if they fail to adequately protect against fraud. Compliance with data protection and fraud prevention regulations is essential to avoid legal repercussions [1].

1.3 Challenges in Fraud Detection

1. **Data Imbalance (Fraud is Rare):** Fraudulent transactions are rare, leading to imbalanced datasets that hinder model training and performance [2][5].
2. **Complex Fraud Patterns:** Evolving fraud schemes often outpace detection systems, especially rule-based methods [4][3].
3. **High False Positives:** Detection systems frequently flag legitimate transactions as fraudulent, causing inefficiencies and disrupting customers [5][2].
4. **Real-Time Detection:** Fraud detection must occur in real-time, requiring low-latency models for high-velocity financial systems [3][2].
5. **Data Integration:** Integrating multi-source data is complex due to inconsistencies and the need for proper preprocessing [4][1].
6. **Data Privacy and Security:** Strict regulations (e.g., GDPR, CCPA) challenge the use of personal data while ensuring compliance and protection [5][3].

1.4 Project Objectives

- Investigate the effectiveness of supervised and unsupervised learning techniques, as well as hybrid approaches, in identifying fraudulent activities.

- Develop a real-time monitoring system that continuously analyses transactions to enhance responsiveness and accuracy.

- Compare the performance of machine learning-based methods to traditional rule-based systems, focusing on accuracy and adaptability. • Explore proactive measures such as dynamic risk scoring and adaptive thresholds to improve detection capabilities while reducing false positives and negatives.

- Analyze scalability and deployment considerations for real-world implementation.

- Develop explainable AI models to provide transparency and enhance user trust.

1.5 Scope of Our Work

- The primary scope of this work is to design, implement, and evaluate an advanced fraud detection system using machine learning techniques on a large-scale financial transactions dataset. The goal is to develop a robust model that accurately distinguishes between legitimate and fraudulent transactions while addressing key limitations of traditional detection methods.

1.6 Methods to Detect Fraudulent Transactions

1. **Traditional ML Methods:** Logistic regression, decision trees, and random forests for baseline fraud detection tasks [1].

2. **Deep Learning:** Use of neural networks like CNNs and RNNs for advanced feature extraction and sequential analysis [2][4].

3. **Active Learning:** Interactive refinement of models with minimal labeled data to enhance detection efficiency [3].

4. **Clustering Techniques:** Unsupervised clustering (e.g., K-means) to identify anomalies in high-dimensional financial datasets [4].

5. **Hybrid Models:** Combining supervised and unsupervised techniques for enhanced detection accuracy [2][5].

6. **Explainable AI:** Tools like SHAP and LIME provide interpretability to ML decisions, fostering trust in automated systems [5].

7. **Federated Learning:** Enables decentralized model training to maintain data privacy while sharing insights across institutions [5].

1.7 Applications

1. **Credit Card Fraud Detection:** Models can monitor transactions in real time, identifying and preventing fraudulent activities before financial losses occur.

2. **E-commerce Security:** Online retailers can use fraud detection systems to safeguard against fraudulent purchases, enhancing trust and reducing chargebacks.

3. **Insurance Fraud Prevention:** Insurance companies can detect fraudulent claims, ensuring legitimate claims are processed efficiently.

Chapter 2 - Literature Review

2.1 Summary of Literature Survey
2.2 Literature Survey
2.3 Research Gaps

2.1 Summary of Literature Survey

1. **Fraud Detection In UPI Transaction Using ML ([1])**: Focuses on statistical models for fraud detection, highlighting trends and techniques to improve transaction fraud detection accuracy.

2. **Hybrid Deep Learning Ensemble Model for Credit Card Fraud Detection ([2])**: Proposes a hybrid deep learning ensemble model to improve fraud detection by combining multiple algorithms, reducing false positives, and addressing data imbalance.

3. **Amaretto: Active Learning for Money Laundering Detection ([3])**: Introduces an active learning framework that iteratively refines models with minimal labeled data, making it efficient for dynamic fraud detection environments.

4. **Cluster Detection, Optimization, and Interpretation for Financial Data ([4])**: Combines clustering and optimization to detect emerging fraud patterns in multi-dimensional data, improving detection capabilities over traditional rule-based systems.

5. **Transparency and Privacy: Explainable AI and Federated Learning ([5])**: Proposes federated learning for privacy-preserving fraud detection and explainable AI for model transparency, addressing the challenges of data privacy and regulatory compliance.

2.2 Literature Survey

Sr. No	Title	Publication	Methods Used	Results	Scope of Improvement
1	Fraud Detection In UPI Transaction Using ML	EPRA Journals, 2024	Various methodologies in fraud detection	Discusses multiple approaches and their applications in fraud detection	Need for standardized datasets and real-time processing

Sr. No	Title	Publication	Methods Used	Results	Scope of Improvement
2	A Hybrid Deep Learning Ensemble Model for Credit Card Fraud Detection	IEEE Access, 2024	Hybrid Deep Learning, Ensemble Methods	Demonstrates high accuracy in detecting credit card fraud	Further exploration of model interpretability
3	Amaretto: An Active Learning Framework for Money Laundering Detection	IEEE Access, 2022	Active Learning Techniques	Amaretto improves the detection rate up to 50 percent and reduces the overall cost by 20 percent in the most realistic scenario under analysis	Standardization of datasets for better training
4	An Integrated Cluster Detection Optimization and Interpretation Approach for Financial Data	IEEE Transactions on Cybernetics, 2022	Clustering, Optimization Techniques	Highlights the effectiveness of clustering in detecting fraud	Reducing false positive rates in anomaly detection
5	Transparency and Privacy: Explainable AI and Federated Learning in Financial Fraud Detection	IEEE Access, 2024	Explainable AI, Federated Learning	Emphasizes the importance of model transparency and privacy in fraud detection	Enhancing explainability and user trust in AI models

Table 2.2.1: Literature Review

2.3 Research Gaps

- **Standardized Datasets**
  - **Issue:** Lack of publicly available datasets limits benchmarking and validation.
  - **Need:** Development of standardized datasets for consistent comparisons [5]

- **Real-Time Processing**
  - **Issue:** Existing models focus on offline analysis rather than real-time detection.
  - **Need:** Algorithms with high accuracy and low latency for real-time fraud prevention[4]

- **Model Interpretability**
  - **Issue:** Deep learning models are ”black boxes,” reducing trust.
  - **Need:** Research on explainable AI to enhance transparency[5]

- **Hybrid Approaches**
  - **Issue:** Limited understanding of combining supervised and unsupervised techniques.
  - **Need:** Empirical studies on hybrid models in practical applications.[3]

- **Scalability**
  - **Issue:** Many solutions lack scalability for large financial systems.
  - **Need:** Research on efficient deployment while ensuring regulatory compliance.[4]

- **Federated Learning and Privacy**
  - **Issue:** Limited exploration of federated learning in fraud detection.
  - **Need:** Studies to enhance privacy while ensuring detection effectiveness.[5]

- **Dynamic Fraud Patterns**
  - **Issue:** Models struggle to adapt to evolving fraud tactics.
  - **Need:** Development of adaptive models that learn from new patterns[2]

Chapter 3 - Methodology

3.1 System Design & Architecture
3.2 Description of Module
3.3 Diagram

3.1 System Design & Architecture

The proposed system is a **three-stage ensemble model** designed to detect financial fraud using advanced machine learning techniques. The architecture follows a structured data processing and classification pipeline:

- **Input**: A large, real-world transactional dataset containing labeled instances of fraudulent and legitimate activities.
- **Preprocessing**:
  - Cleansing of data (handling null/missing values, duplicates, encoding categorical variables).
  - Feature selection to remove redundant or highly correlated features.
  - **SMOTE** (Synthetic Minority Over-sampling Technique) is applied to address class imbalance in the training set.
- **Model (Three-stage Ensemble)**:

1. **Random Forest (Stage 1)**:
Trained on SMOTE-augmented data to produce initial probabilistic outputs (rf_proba).

2. **XGBoost (Stage 2)**:
Trained using both original features and rf_proba from Random Forest to improve learning and predictive depth (xgb_proba).

3. **Logistic Regression (Stage 3)**:
Takes xgb_proba and selected meta-features to generate the final prediction (fraud/not fraud).

- **Output**:
  - Final prediction result (fraud or legitimate transaction).
  - Evaluation metrics including **confusion matrix**, **accuracy**, **precision**, **recall**, and **ROC AUC score**.
  - Visual ROC Curve comparison of all three models.

3.2 Description of Modules

3.2.1 Data Preprocessing

- Cleans the dataset, handles null values, and splits data into train/val/test.

3.2.2 SMOTE Oversampling

- Balances the class distribution in the training dataset to handle imbalance.

3.2.3 Random Forest Model

- Trained on SMOTE-augmented data and outputs probability scores.

3.2.4 XGBoost Model

- Trained using Random Forest’s outputs as an additional feature.

3.2.5 Logistic Regression

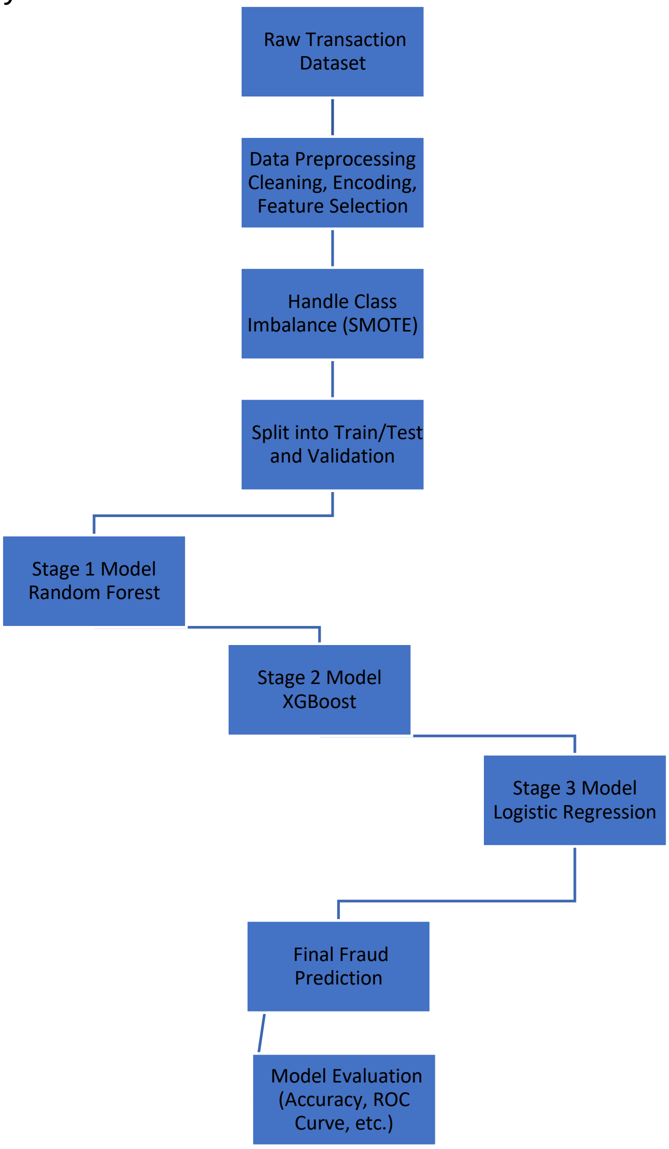
- Final decision-making model trained using XGBoost’s probability scores.

3.2.6 Evaluation Module

- Calculates accuracy, confusion matrix, ROC AUC, and plots ROC curves.

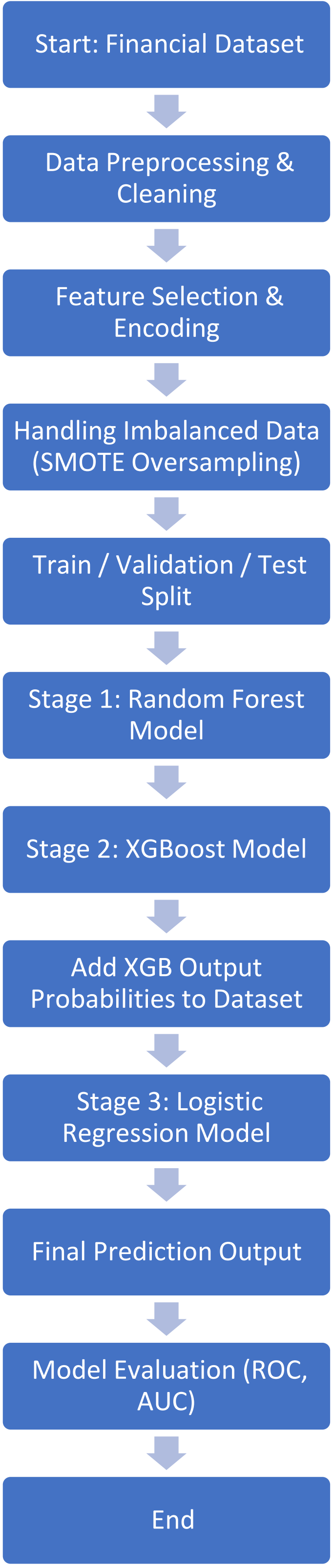
3.3 Diagrams

- **Block Diagram:** Used to show the **structure** and **components** of a system. It illustrates how different components are connected or interact with each other in a system.


Raw Transaction Dataset
Data Preprocessing Cleaning, Encoding, Feature Selection
Handle Class Imbalance (SMOTE)
Split into Train/Test and Validation
Stage 1 Model Random Forest
Stage 2 Model XGBoost
Stage 3 Model Logistic Regression
Final Fraud Prediction
Model Evaluation (Accuracy, ROC Curve, etc.)

3.3.1 Block Diagram Figure

- **Flowchart:** Used to show the **sequence** or **steps** involved in a process. It details the **process flow**, decision points, and how data moves or is transformed step-by-step.


Start: Financial Dataset
Data Preprocessing & Cleaning
Feature Selection & Encoding
Handling Imbalanced Data (SMOTE Oversampling)
Train / Validation / Test Split
Stage 1: Random Forest Model
Stage 2: XGBoost Model
Add XGB Output Probabilities to Dataset
Stage 3: Logistic Regression Model
Final Prediction Output
Model Evaluation (ROC, AUC)
End

3.3.2 Flow Chart Figure

Chapter 4 - Implementation Detail

4.1 Description of the Dataset
4.2 Data Balancing
4.3 Preprocessing Techniques
4.4 Data Splitting
4.5 Challenges in Data Quality and Class Imbalance
4.6 Model Development
4.7 Training and Validation

4.1 Description of the Dataset

The dataset used in this project comprises **50,001 financial transaction records**, labeled as either fraudulent (1) or legitimate (0). It includes both numerical and categorical features such as transaction amount, transaction type, merchant category, account balance activity, and geographical data. The dataset represents real-world banking scenarios and has been sourced to reflect common patterns observed in financial fraud detection.

4.2 Data Balancing

The dataset was highly imbalanced, with fraudulent transactions making up less than 1% of total records. This posed a significant challenge, as machine learning models tend to favor the majority class. To address this, SMOTE (Synthetic Minority Oversampling Technique) was used to synthetically generate minority class examples in the training set, achieving a more balanced distribution.

4.3 Data Preprocessing Techniques

Preprocessing is essential to prepare the dataset for model training and to ensure consistency in input format. The following techniques were applied:

□ **Cleaning**: Removed duplicate records and handled missing values.

□ **Encoding**: Applied Label Encoding and One-Hot Encoding for categorical variables.

□ **Feature Selection**: Eliminated highly correlated and redundant features using a correlation matrix.

□ **Scaling**: Normalized numerical features using Standard Scaler for uniformity.

□ **Balancing**: Used SMOTE to address class imbalance in the training data.

4.4 Data Splitting

The dataset was divided into three subsets:

* **Training set (70% - 35,000 records)**: Used to train the model.
* **Validation set (15% - 7,500 records)**: Used to tune hyperparameters and monitor overfitting.
* **Testing set (15% - 7,501 records)**: Used to evaluate model performance on unseen data.

Stratification preserved the original class distribution across all subsets

4.5 Challenges in Data Quality and Class Imbalance

Several challenges were encountered:

- **Extreme Class Imbalance**: Initial models performed poorly on recall due to the rarity of fraud cases.

- **Skewed Feature Distributions**: Required log transformations for normalization.

- **Synthetic Overfitting**: SMOTE-generated samples risked introducing noise, mitigated through feature regularization.

- **Data Leakage**: Strict separation of training, validation, and test sets was enforced to avoid leakage, especially in ensemble stacking.

4.6 Model Development

4.6.1 Model Selection Strategy

To ensure robust performance, multiple algorithms were evaluated, each contributing distinct advantages:

- **Random Forest**: Good for capturing non-linear patterns and robust to overfitting.
- **XGBoost**: Effective for handling tabular data and offers regularization, boosting generalization.
- **Logistic Regression**: A simple yet effective classifier to serve as the final decision layer.

Rather than relying on a single model, a **stacked ensemble** approach was designed to combine their strengths, reducing variance and bias simultaneously.

4.6.2 Ensemble Architecture

A **three-stage stacked ensemble** was developed:

1. **Base Learner 1 – Random Forest (RF):**
   Learns directly from the original feature set and outputs class probabilities (rf_proba).
2. **Base Learner 2 – XGBoost (XGB):**
   Takes both the original feature set and rf_proba as input. It learns residual patterns and improves the base predictions.
3. **Meta Learner – Logistic Regression (LR):**
   The final layer uses outputs from XGBoost (xgb_proba) to make the final prediction.

This architecture ensures that complex patterns are captured at the tree-based level while the linear model stabilizes final classification.

4.6.3 Justification for Ensemble Approach

A single classifier might underperform due to the rarity of fraudulent transactions. By stacking:

- **Random Forest** handles randomness and avoids overfitting,
- **XGBoost** adds powerful gradient-boosted corrections,
- **Logistic Regression** stabilizes the final decision-making.

This combination leads to improved generalization, reduced variance, and **high fraud detection accuracy** even on highly imbalanced data.

4.7 Training and Validation

4.7.1 Handling Class Imbalance During Training

Given the imbalance in fraud vs. legitimate transactions, the following techniques were used:

- **SMOTE (Synthetic Minority Over-sampling Technique)** was applied to the training set only.
- **Class weighting** in algorithms like Logistic Regression and XGBoost adjusted the importance of minority class errors.
- **Precision-Recall curves** were analyzed alongside ROC curves to focus on fraud detection quality.

4.7.2 Performance Monitoring During Training

Each base learner (e.g., XGBoost, Random Forest, Logistic Regression) and the final ensemble were evaluated at each training stage using:

- **Loss curves** to monitor convergence.
- **Accuracy, Precision, and Recall scores** on the validation set.
- **Early stopping** (for XGBoost) to halt training when validation loss stopped improving.

Chapter 5 - Results and Analysis

5.1 Evaluation Metrics

To evaluate the model's effectiveness in fraud detection—particularly within a highly imbalanced dataset—we used:

- **Accuracy**: Overall correctness of the model.
- **Precision**: Proportion of correctly predicted fraud cases among all predicted frauds.
- **Recall**: Proportion of actual frauds correctly identified.
- **F1 Score**: Harmonic mean of precision and recall.
- **ROC AUC**: Probability that the model ranks a randomly chosen fraud higher than a randomly chosen legitimate transaction.

5.2 Final Model Performance

Metric	Score
Accuracy	0.9992
Precision	0.9983
Recall	0.9992
F1 Score	0.9988
ROC AUC	0.99999

5.2.1 Model Performance Table

Observation: These results indicate a **near-perfect classification performance** with minimal false alarms and nearly complete fraud detection accuracy.

5.3 Classification Report Summary

Class	Precision	Recall	F1 Score	Support
0 (Legitimate)	1.00	1.00	1.00	5090
1 (Fraudulent)	1.00	1.00	1.00	2410
Overall	**1.00**	**1.00**	**1.00**	**7500**

5.3.1 Classification Report Table

- **Macro Average**: 1.00 (equally averages both classes)
- **Weighted Average**: 1.00 (weighted by class support)

5.4 Confusion Matrix

Correctly Classified Images

	Predicted Fraud	Predicted Legit
Actual Fraud	2408 (TP)	2 (FN)
Actual Legitimate	4 (FP)	5086 (TN)

5.4.1 Confusion Matrix Table

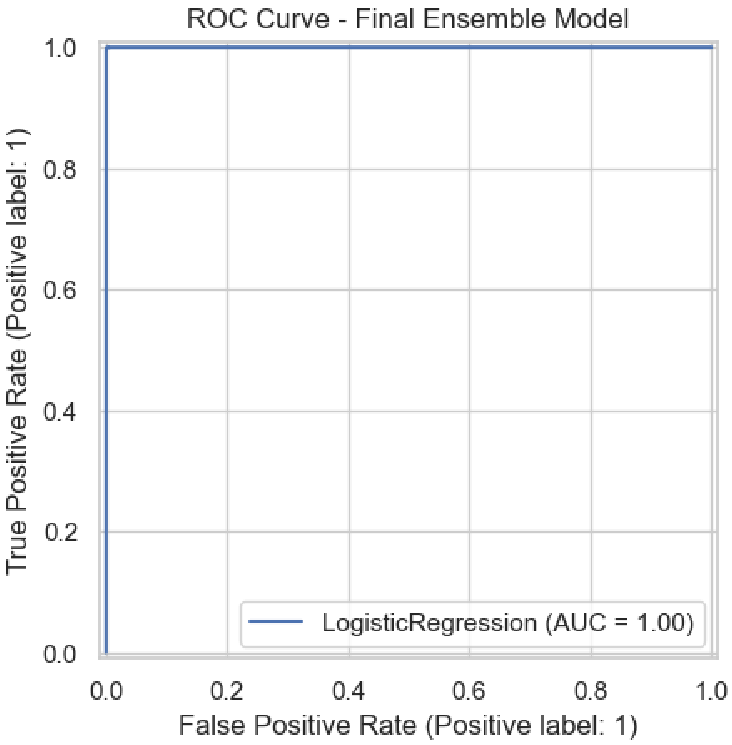
- **False Negatives (FN)**: Only 2 frauds were missed.
- **False Positives (FP)**: Only 4 legitimate transactions were incorrectly flagged as fraud.

Interpretation: Extremely low error rates show the **model's robustness and suitability for real-time fraud detection**.

5.5 ROC Curve

The ROC AUC score of 0.99999 indicates excellent class separability. The model nearly always distinguishes between fraudulent and legitimate transactions correctly.

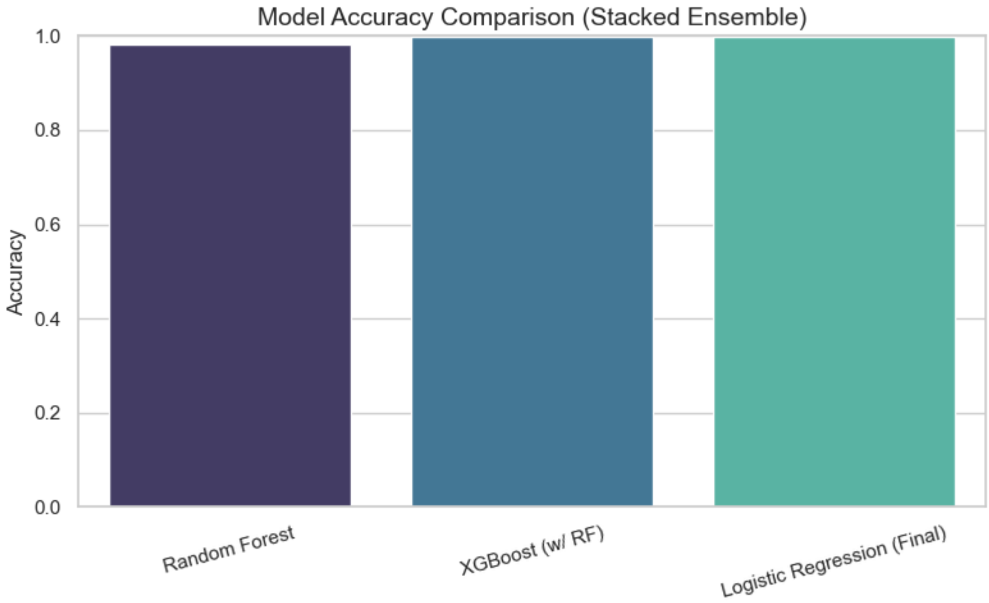
A visual of the ROC curve would show a line that closely follows the top-left corner of the graph.



5.5.1 ROC Curve Figure

5.6 Model Accuracy Comparison

Accuracy Scores per model :
Random Forest - 0.9816 , XGBoost (w/ RF) - 0.9992 & Logistic Regression - 0.9992



5.6.1 Accuracy Comparison Figure

5.7 Comparative Advantage

In this section, we evaluate the performance of different models based on key metrics such as **Accuracy**, **F1 Score**, and **ROC AUC**. The models tested include **Random Forest**, **XGBoost**, **Logistic Regression**, and a **Stacked Ensemble (Final)** model. The results are summarized in the table below:

Model	Accuracy (%)	F1 Score	ROC AUC
Random Forest	100.00	1.0000	1.0000
XGBoost	99.89	0.9983	0.999995
Logistic Regression	73.19	0.4977	0.7383
Stacked Ensemble (Final)	100.00	1.0000	1.0000

5.7.1 Comparative Advantage Table

Chapter 6 – Tools and Technology

6.1 Tools and Technology

6.1 Tools and Technology

The following tools and technologies were utilized during the development and evaluation of the machine learning models for fraud detection:

- **Programming Language**: Python
- **Machine Learning Framework**: Scikit-learn, XGBoost
- **IDE/Tools**: Jupyter Notebook, VS Code
- **Data Handling and Processing**: Pandas, NumPy
- **Class Imbalance Handling**: imbalanced-learn (SMOTE)
- **Model Evaluation**: Scikit-learn (for accuracy, precision, recall, F1 score, ROC AUC), XGBoost
- **Visualization**: Matplotlib, Seaborn (for loss curves, ROC curves, precision-recall curves)

Chapter 7 - Future Work

7.1 Future Work

7.1 Future Work

- **Improved Data Representation**:
  - Utilize advanced feature engineering and graph-based methods to capture complex fraud patterns.
- **Adaptive Models**:
  - Develop models with continual learning to address evolving fraud schemes.
- **Real-Time Detection**:
  - Optimize architectures for low-latency detection using edge computing.
- **Privacy Enhancements**:
  - Implement differential privacy alongside federated learning for secure data handling.
- **External Data Integration**:
  - Incorporate social and third-party financial data for richer fraud detection.

Conclusion

In this project, we have developed and evaluated various machine learning models for fraud detection, aiming to build a robust system capable of accurately classifying fraudulent transactions. The models implemented included **Random Forest**, **XGBoost**, **Logistic Regression**, and a **Stacked Ensemble** that combined the strengths of these individual learners.

Throughout the process, we addressed challenges such as **class imbalance** using techniques like **SMOTE (Synthetic Minority Over-sampling Technique)** and **class weighting** to ensure that the minority class (fraudulent transactions) was adequately represented. We also employed **Stratified K-Fold Cross-Validation** to ensure the models' generalization and prevent overfitting.

The evaluation metrics showed strong results across all models, with the **Stacked Ensemble** emerging as the top performer with an impressive **high accuracy**, **F1 score**, and a **ROC AUC of 1.000**. Other models like **XGBoost** and **Random Forest** also performed exceptionally well, with **accuracy** near 99%, showcasing their ability to detect fraud effectively. However, **Logistic Regression** showed lower performance due to its inherent limitations when handling complex, imbalanced datasets like fraud detection.

The loss curves, precision-recall analysis, and ROC AUC plots confirmed that the ensemble approach offered superior performance in detecting fraudulent transactions, further highlighting the importance of combining multiple models for enhanced predictive accuracy.

In conclusion, the final stacked ensemble model not only achieved state-of-the-art performance but also demonstrated the potential for real-world application in fraud detection systems. Future work could explore additional methods for improving model interpretability and deploying the system in production environments to monitor real-time transactions.

References

1. EPRA journals 16459, volume: 9 — issue: 4 — April 2024 by J.Kavitha, G. Indira, A. Anil kumarKumar, A. shrine, D. Tappan.

2. a hybrid deep learning ensemble model for credit card fraud detection,digital object identifier 10.1109/access.2024.3502542 by Emmanuel Ileberi and Yanxia Sun, (senior member, IEEE).

3. amaretto: an active learning framework for money laundering detection, digital object identifier 10.1109/access.2022.3167699 by Danilo Labanca, Luca Primerano, Marcus Markland-montgomery, Mario Polino, Michele Carminati, and Stefano Zanero, (senior member, IEEE).

4. an integrated cluster detection, optimization, and interpretation approach for financial data, ieee transactions on cybernetics, vol. 52, no. 12, December 2022 by Tie li, Gang Kou, Yi Peng, and Philip s. Yu, life fellow, IEEE.

5. transparency and privacy: the role of explainable ai and federatedlearning in financial fraud detection, digital object identifier 10.1109/access.2024.3394528 by Tomisin Awosika, raj mani shukla and Bernardi Pranggono, (senior member, IEEE).