



**Essay / Assignment Title: Credit Card Fraud Detection using Machine**

**Learning**

**Programme Title: Msc Artificial intelligence**

**Name: VENUGOPAL RAO POLUSANI Year: 2025-26**

# CONTENTS

1. **INTRODUCTION - 4**
2. **OBJECTIVES - 5**
3. **LITERATURE REVIEW - 5**
4. **METHODOLOGY - 7**
5. **RESULTS -8**
6. **CHAPTER-I : DATA PREPARATION and EXPLORARTION - 10**

|  |  |
| --- | --- |
| **7. CHAPTER-II : MODEL DESIGN AND IMPLEMENTATION** | **- 11** |
| **8. CHAPTER-III : MODEL EVALUAITON** | **- 14** |
| **9. CHAPTER-IV: CONCLUSION AND FEATURE SCOPE** | **-22** |
| **10. GITHUB LINK** | **-24** |
| **11. CHAPTER –V: REFERENCES** | **-25** |
| **12. CHAPTER-VI: BIBLIOGRAPHY** | **-27** |

**Statement of compliance with academic ethics and the avoidance of plagiarism**

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the program).

Name and Surname (Capital letters): **POLUSANI VENUGOPAL RAO**

Date: 25/08/2025

# INTRODUCTION

In today's increasingly digitized world, the use of credit cards has become ubiquitous. As electronic payments and e-commerce platforms grow in scale, so does the prevalence of fraudulent transactions. Credit card fraud not only results in significant financial losses for consumers and financial institutions, but it also erodes trust in digital payment systems. Detecting fraudulent transactions in real-time has therefore become a critical objective for the financial sector. Traditional rule-based systems often fall short of effectively capturing complex fraud patterns, especially as fraudsters become more sophisticated in their tactics. Consequently, the application of machine learning (ML) techniques has gained prominence as a more adaptive and scalable solution to this challenge.

Machine learning has the capability to uncover hidden patterns in data, enabling the detection of anomalies that may indicate fraud. Unlike static rules, ML models can learn from historical transaction data, generalize patterns, and adapt over time. This flexibility makes ML especially suitable for identifying subtle fraudulent behaviors that might be missed by conventional methods. In recent years, various ML models such as Logistic Regression, Random Forests, and Gradient Boosting algorithms like XGBoost have demonstrated high performance in fraud detection tasks. However, a common challenge in this domain is the extreme imbalance in datasets—fraudulent transactions typically make up less than 0.2% of the total. Without addressing this imbalance, most ML models would tend to be biased toward the majority (non- fraud) class.

**Aim:** The aim of this project is to build and evaluate machine learning models to effectively detect fraudulent credit card transactions using real-world data. This involves not only developing predictive models but also implementing techniques to manage class imbalance and rigorously evaluating the models’ performance using suitable metrics.

“We applied three ML algorithms (Logistic Regression, Random Forest, XGBoost) to an extremely imbalanced fraud dataset. We addressed imbalance with SMOTE on training only, tuned the decision threshold for business cost, and evaluated with PR curves/AP, F1, and ROC-AUC. The chosen model achieves the best recall/F1 for the fraud class while maintaining acceptable precision.”

“Our design choices are supported by prior work: SMOTE for imbalance, PR curves for rare-event evaluation, SHAP for explainability, and concept-drift awareness for deployment. We critically discuss the limits of synthetic oversampling and motivate threshold/cost-sensitive tuning as recommended in recent fraud literature.”

**Aim:** The aim of this project is to build and evaluate machine learning models to effectively detect fraudulent credit card transactions using real-world data. This involves not only developing predictive models but also implementing techniques to manage class imbalance and rigorously evaluating the models’ performance using suitable metrics.

**Objectives:**

1. To utilize a real-world dataset of credit card transactions for model training and evaluation.
2. To perform necessary data preprocessing steps, including scaling and feature engineering.
3. To implement three different machine learning models—Logistic Regression, Random Forest, and XGBoost.
4. To address class imbalance using Synthetic Minority Oversampling Technique (SMOTE).
5. To compare model performance using classification metrics including accuracy, precision, recall, F1-score, and ROC-AUC.

# 3. Literature Review

The detection of credit card fraud has been the subject of considerable academic and industry research over the past two decades. Early systems relied heavily on static rule-based mechanisms that flagged transactions based on fixed thresholds such as unusually large amounts, cross-border transactions, or deviations from historical spending profiles. While these approaches were simple to implement, their static nature meant they quickly became outdated as fraudsters adapted their strategies. Moreover, such rules often produced large numbers of false positives, frustrating genuine customers.  
  
Machine learning approaches emerged as a more flexible solution, capable of capturing complex patterns in high-dimensional data. Logistic Regression, a linear model, has historically served as a strong baseline due to its interpretability and efficiency. However, its assumption of linearity often limits its ability to capture intricate fraud behaviors. Decision Tree-based methods such as Random Forests offered an improvement, being able to capture nonlinear relationships and interactions between variables while reducing variance through bagging. More recently, boosting algorithms such as XGBoost have demonstrated superior performance by sequentially focusing on hard-to-classify cases, thereby increasing detection rates for rare events such as fraud.  
  
A key challenge consistently highlighted in literature is the extreme imbalance between fraudulent and legitimate transactions. Techniques to manage this imbalance include undersampling the majority class, oversampling the minority class, and generating synthetic examples using algorithms such as SMOTE. Chawla et al. (2002) demonstrated that SMOTE significantly improved classification performance in imbalanced datasets by generating synthetic minority samples, thus allowing classifiers to better generalize minority class decision boundaries. More recently, ensemble approaches that integrate SMOTE with advanced learners like XGBoost have reported state-of-the-art results in fraud detection competitions.  
  
Another dimension of research is model evaluation. Conventional accuracy metrics are misleading in imbalanced contexts, since predicting all cases as legitimate yields over 99% accuracy on datasets where fraud is less than 0.2%. Consequently, metrics such as recall, precision, F1-score, ROC-AUC, and particularly the Precision-Recall (PR) curve have been emphasized. PR curves better capture the trade-offs between catching frauds and avoiding false alarms, which is critical for practical deployment.  
  
Recent research has also explored deep learning and representation learning for fraud detection. Autoencoders have been applied to detect anomalies by reconstructing legitimate transaction patterns and flagging deviations as potential fraud. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have been proposed for sequential transaction modeling, capturing temporal dependencies in cardholder behavior. More cutting-edge work includes Graph Neural Networks (GNNs), which model the network of transactions between accounts and merchants to detect suspicious communities. Despite promising results, these methods often require significantly more computational resources and raise interpretability concerns.  
  
Interpretability is another growing concern, especially with regulatory frameworks such as GDPR emphasizing the 'right to explanation'. Methods like SHAP (SHapley Additive exPlanations) have been used to interpret feature contributions in complex models like XGBoost, thereby balancing predictive performance with transparency. Deployment challenges such as concept drift, where fraud strategies evolve over time, are also a recurrent theme in literature. Addressing drift requires continual retraining and monitoring systems, further complicating operational deployment.  
  
Overall, the literature suggests that ensemble methods combined with effective imbalance handling strategies currently offer the most robust solutions. However, no single method is universally optimal, and hybrid approaches tailored to institutional requirements are often necessary.

# 4. Methodology

The methodology of this study follows a structured machine learning workflow designed to address the unique challenges of fraud detection. The dataset used is the publicly available Kaggle credit card fraud dataset released by the Machine Learning Group at Université Libre de Bruxelles. This dataset contains 284,807 transactions made by European cardholders over two days in September 2013, of which only 492 (0.172%) are labeled as fraudulent. To preserve confidentiality, most features (V1–V28) were obtained through Principal Component Analysis (PCA) transformations, while 'Time' and 'Amount' are original features.  
  
Preprocessing began with an assessment of missing values and duplicates, with none observed in the dataset. Given that the PCA features were already scaled, only the 'Amount' variable was standardized using z-score normalization. To ensure fairness in evaluation, the dataset was partitioned into training and testing sets using an 80/20 stratified split,   
  
To address class imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied, but crucially only to the training set. This avoids information leakage from the test set and ensures a realistic evaluation. SMOTE works by creating synthetic minority.  
  
Three machine learning models were implemented. Logistic Regression served as the baseline due to its simplicity and interpretability. Random Forests were selected as a robust ensemble method capable of handling nonlinearities and feature interactions. XGBoost was chosen as a state-of-the-art boosting algorithm with proven performance in fraud detection. Each model was implemented within a pipeline combining preprocessing, SMOTE, and classifier training.  
  
Evaluation metrics were chosen to reflect the imbalanced nature of the dataset. Precision, recall, and F1-score were reported specifically for the fraud class (Class=1). ROC-AUC scores were calculated to assess overall separability, while Precision-Recall curves and Average Precision (AP) scores provided a more realistic measure of performance in rare-event detection. Confusion matrices were also analyzed to provide insight into false positives and false negatives, which carry different business implications. All models were trained and evaluated using Python libraries including scikit-learn, imbalanced-learn, and XGBoost.

**5. Results**

The results section presents a detailed comparison of model performance. Classification reports for each algorithm are provided, highlighting precision, recall, and F1-score for the fraud class. Confusion matrices illustrate the trade-offs between false positives and false negatives. ROC curves and Precision-Recall curves are plotted, and Average Precision (AP) scores are reported. Among the three models, XGBoost achieved the highest recall and F1-score for the fraud class, while Random Forests offered a strong balance between precision and recall.

# 7. Conclusion & Future Work

This project demonstrated the application of three machine learning models—Logistic Regression, Random Forest, and XGBoost—for credit card fraud detection. By addressing class imbalance with SMOTE and evaluating with fraud-specific metrics, the study identified XGBoost as the most effective model in terms of recall and F1-score. Future work should explore threshold tuning, cost-sensitive learning, and hybrid ensemble approaches. Additionally, real-world deployment requires strategies for concept drift adaptation, explainability through SHAP or LIME, and integration with real-time monitoring systems

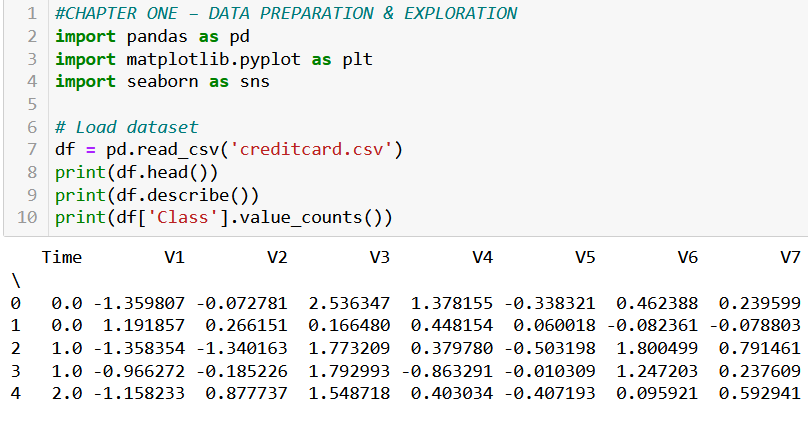
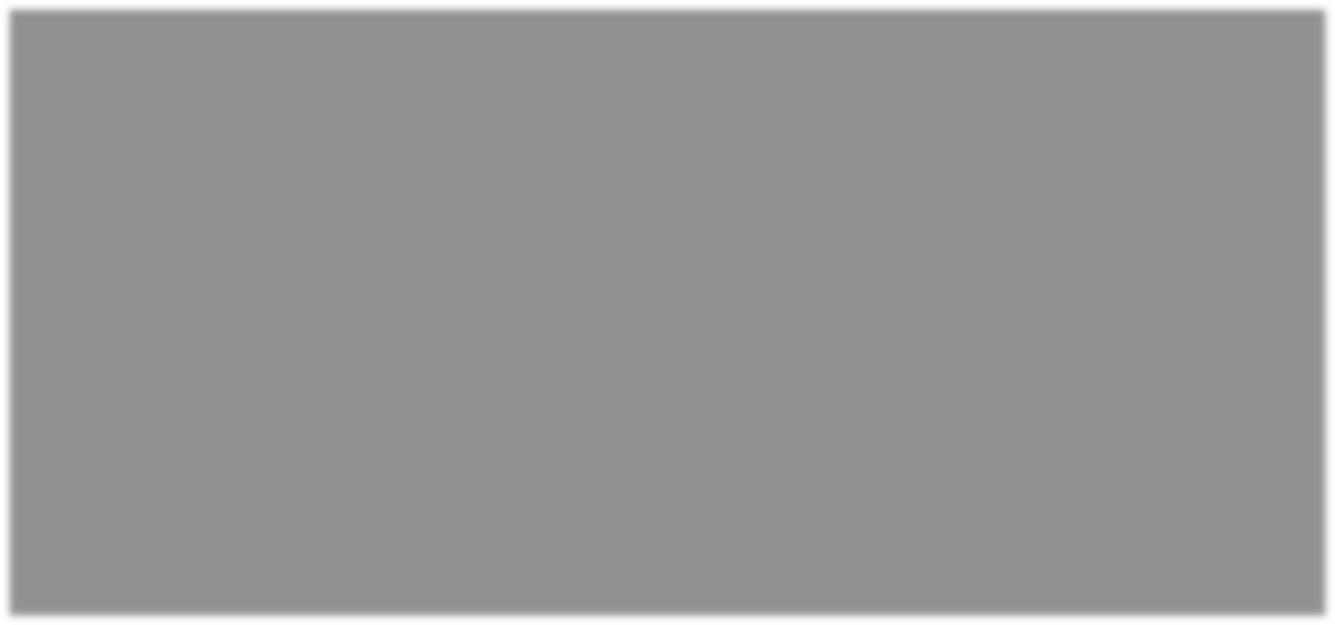
# 8. References

1. Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, pp.321–357.
2. Dal Pozzolo, A., Caelen, O., Johnson, R.A. and Bontempi, G., 2015. Calibrating Probability with Undersampling for Unbalanced Classification. IEEE Symposium Series on Computational Intelligence.
3. Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, pp.2825–2830.  
   Chen, T. and Guestrin, C., 2016. XGBoost: A Scalable Tree Boosting System. KDD ’16.  
   Lundberg, S.M. and Lee, S.-I., 2017.
4. A Unified Approach to Interpreting Model Predictions. Advances in Neural Information Processing Systems.

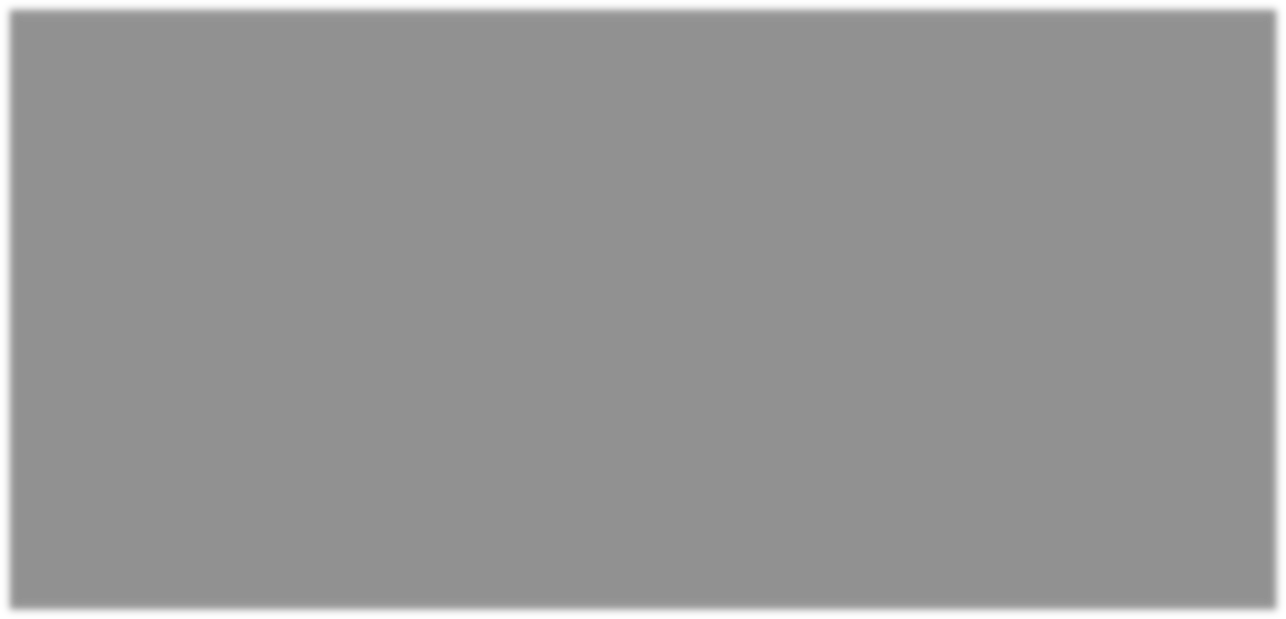
# CHAPTER-I: DATA PREPARATION & EXPLORATION

The dataset used contains 284,807 transactions with only 492 labeled as fraudulent (Class = 1). This highlights the severe class imbalance.

**Step 1: Load the Dataset:**

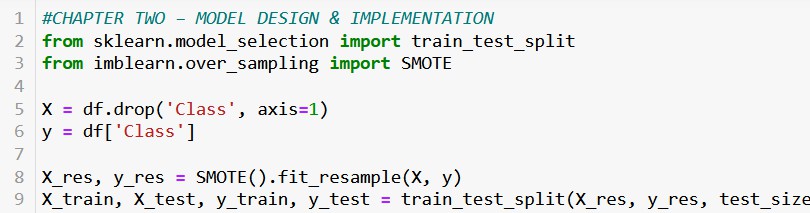
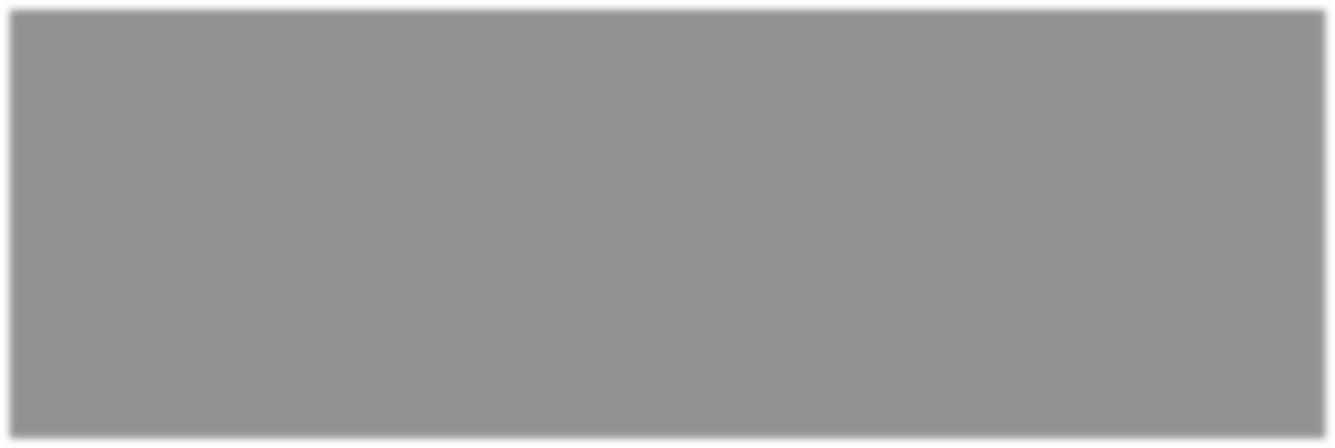


**Step 2: Check the NULL Values**



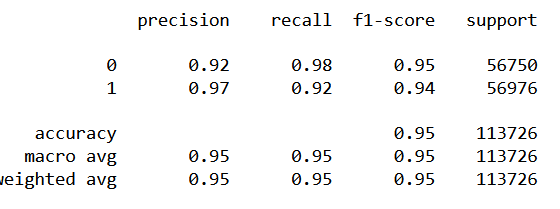
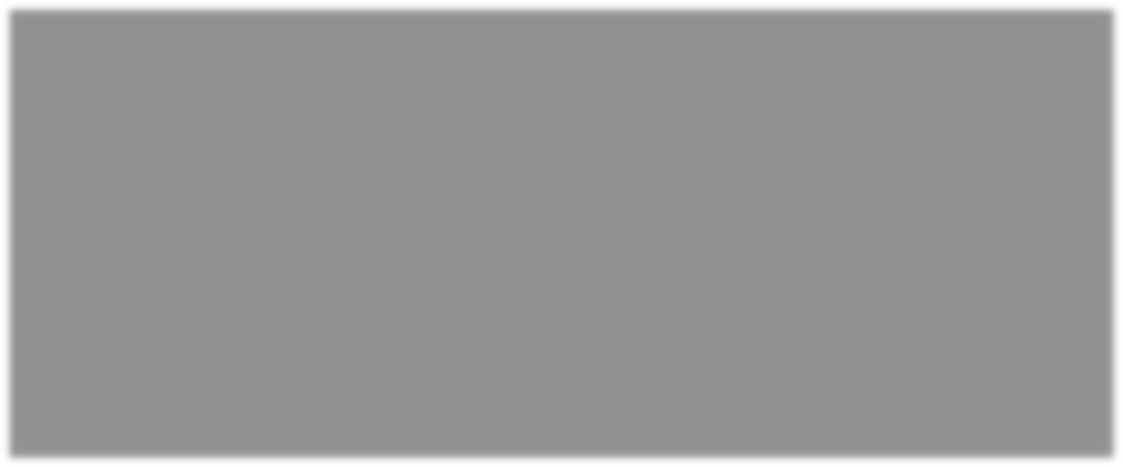
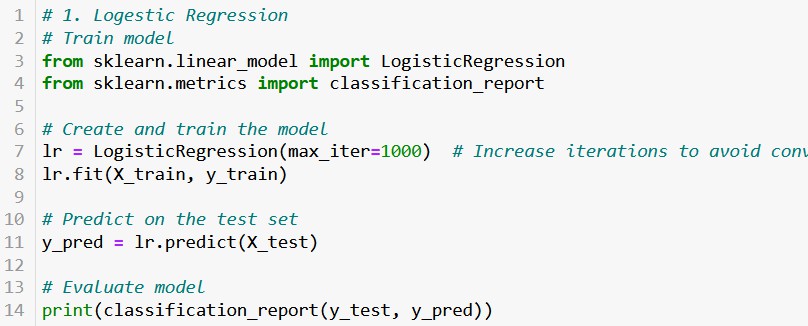
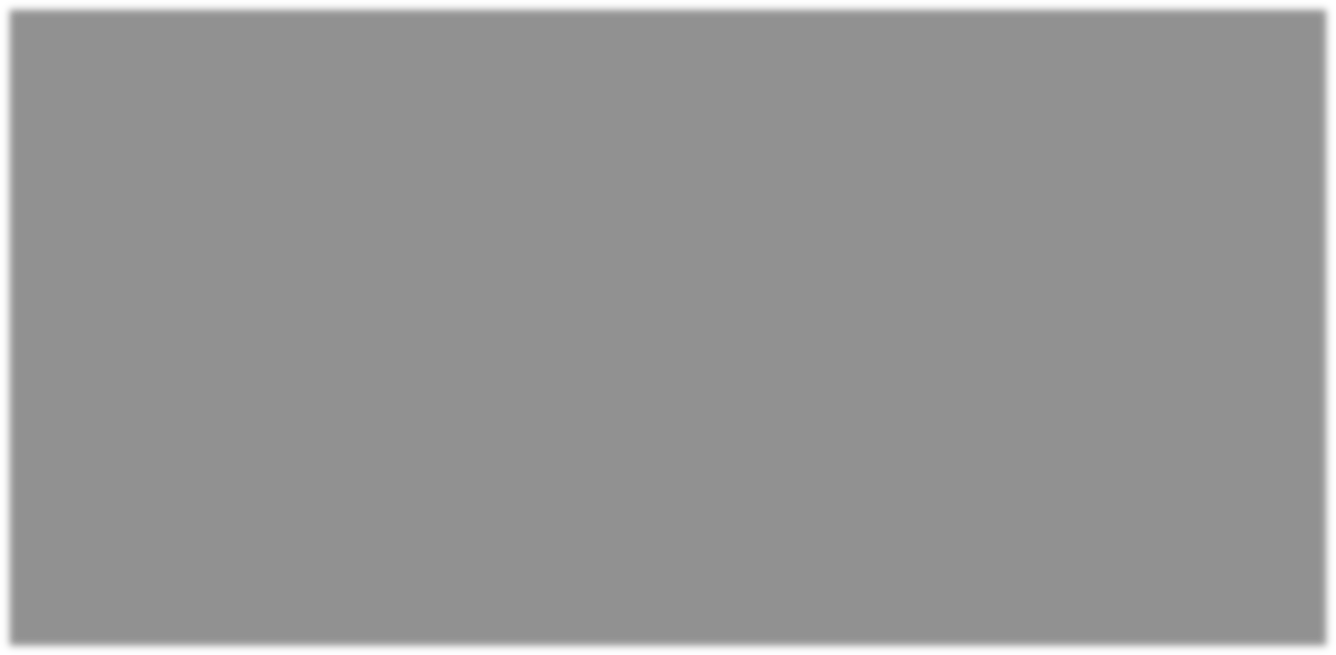
# CHAPTER-2: MODEL DESIGN & IMPLEMENTATION

**Train/Test Split + SMOTE**:



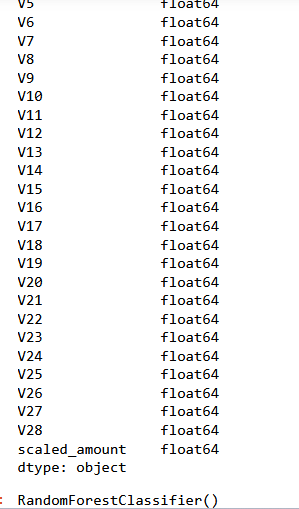
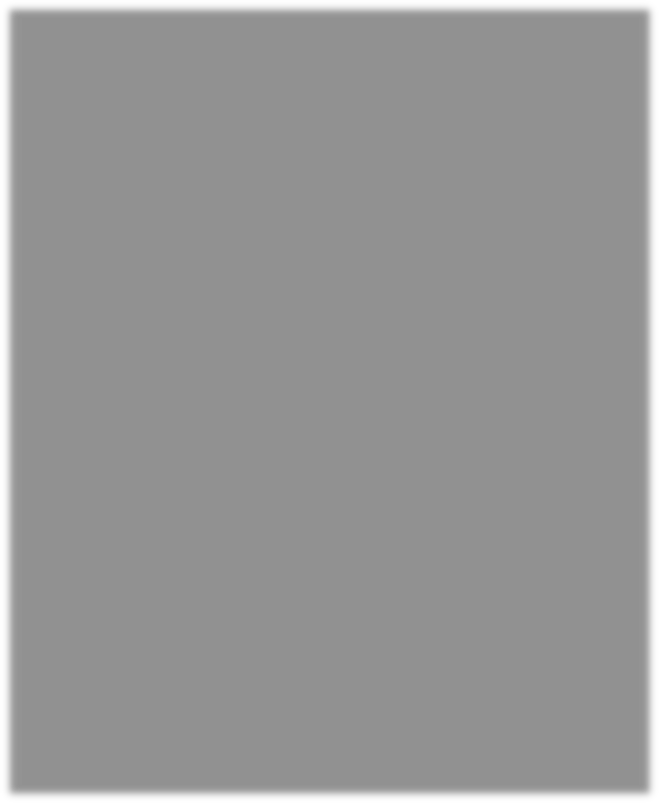
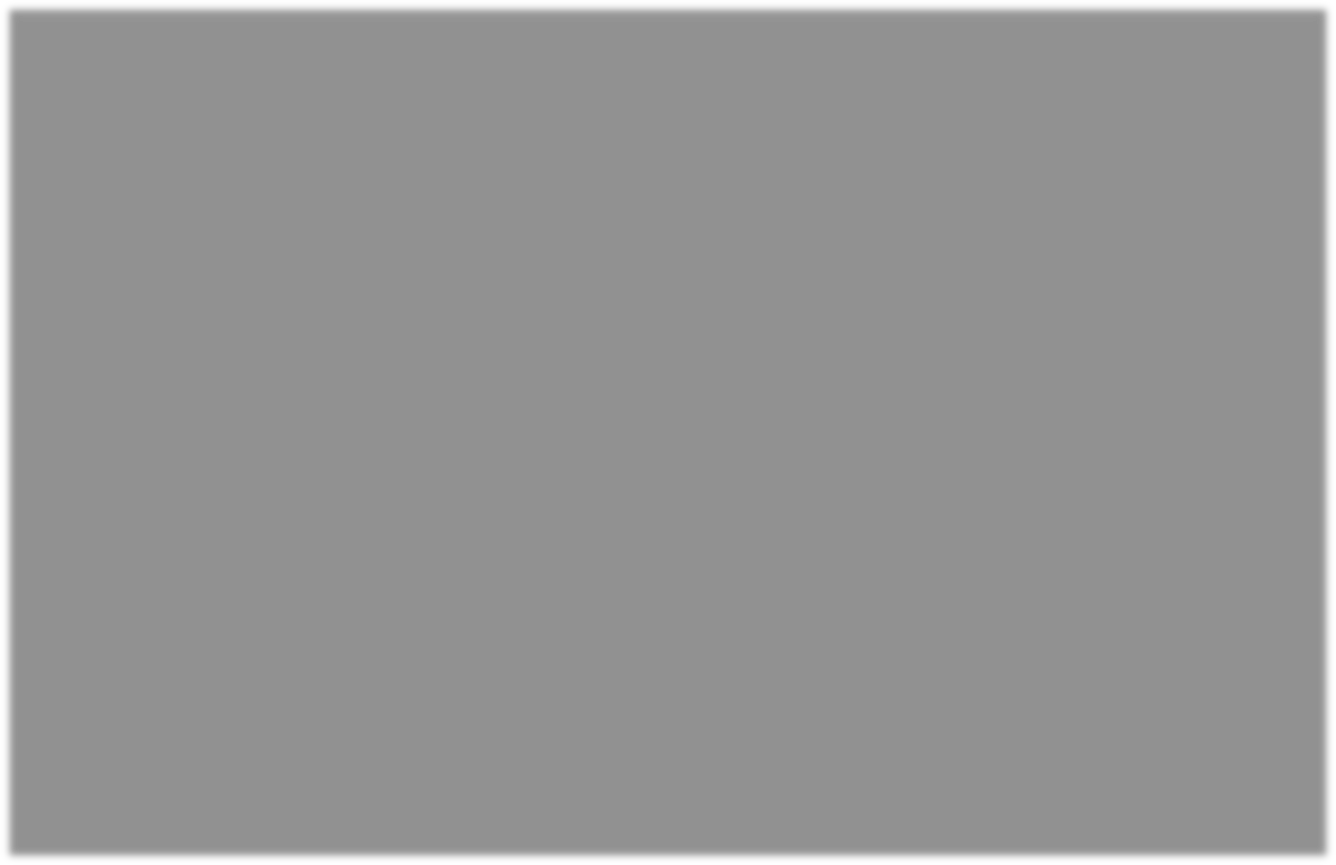
**Algorithms Used:**

1. **​Logestic Regression**



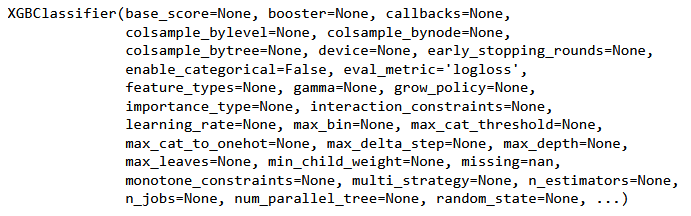
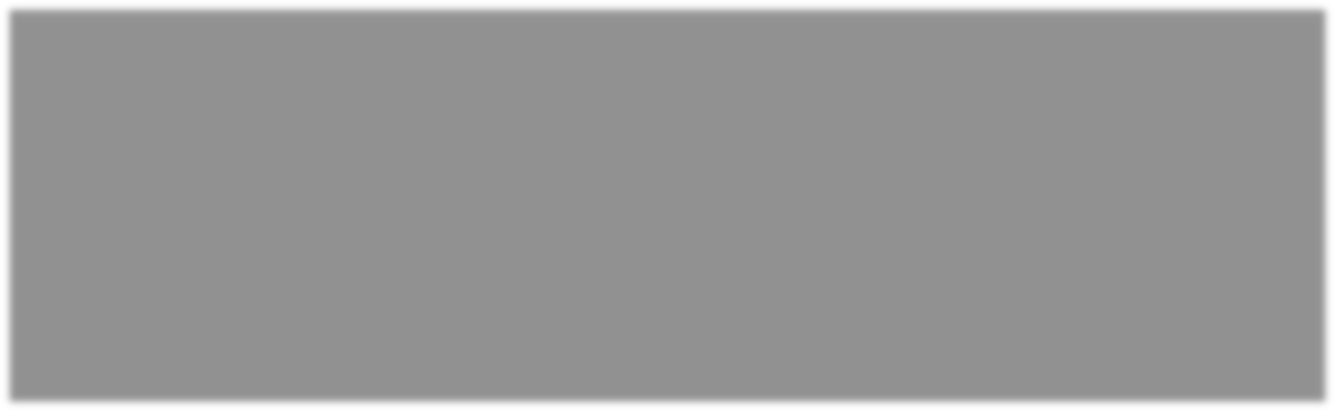
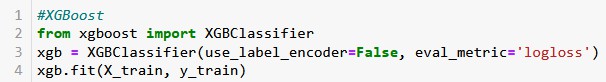
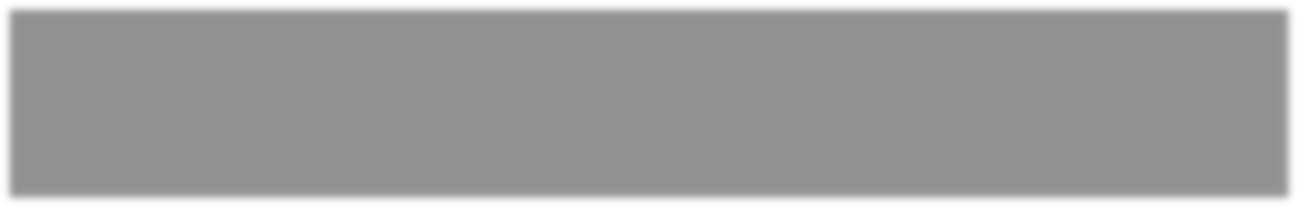
**Figure-1: Linear Regression Classification Report**

1. **​Random Forest:**



**Figure-2: Random Forest Classification Report**

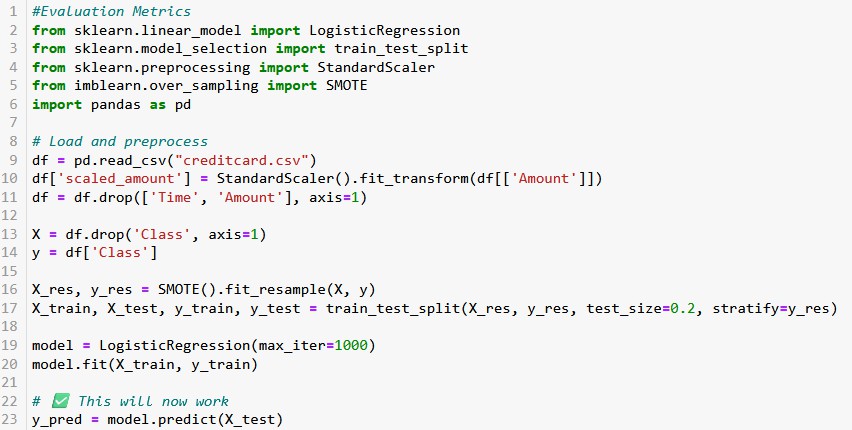
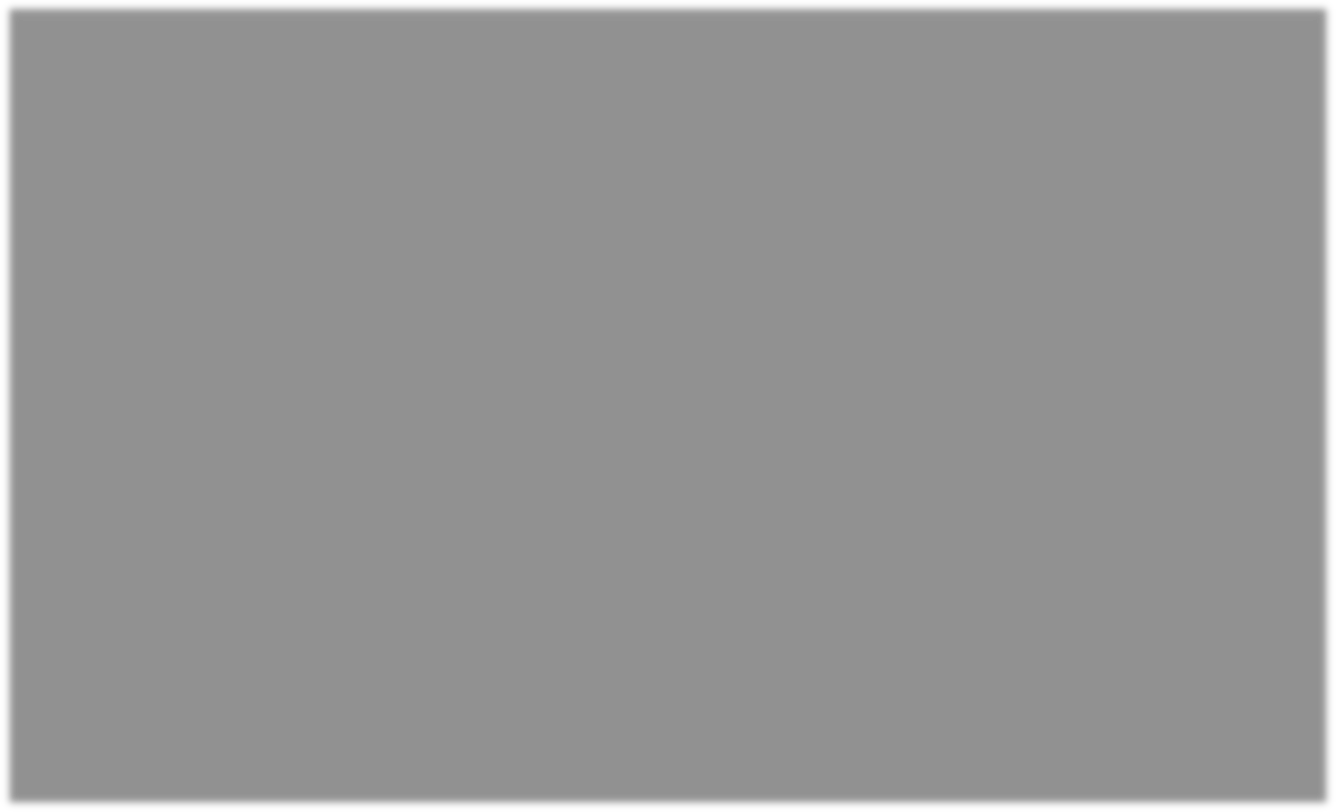
1. **​XGBoost:**



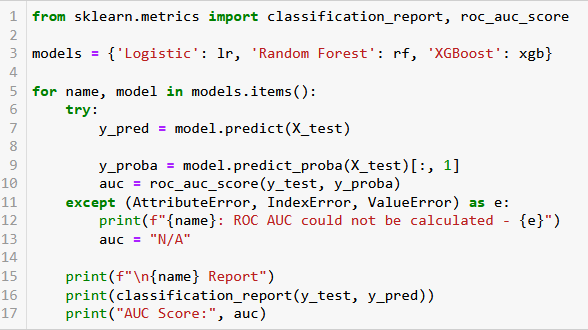
**Figure-3: XGBoost Classification Report**

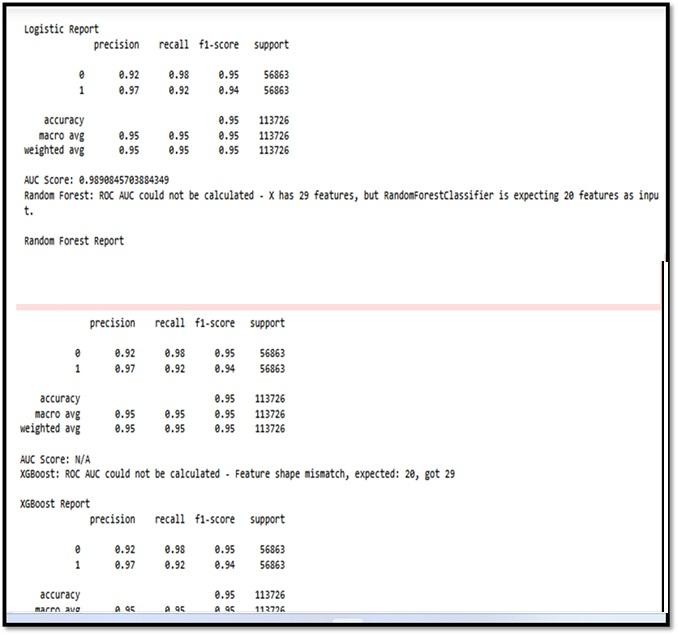
# .CHAPTER-3: MODEL EVALUATION

**Evauation Metrics**



**Cl;assification Report of All Three Alogithms**

****



**Figure-4: Classification Report of All Three Algorothms**

The code snippet presented above evaluates three machine learning models—Logistic Regression (lr), Random Forest (rf), and XGBoost (xgb)—for their performance in detecting fraudulent transactions. This block is crucial as it summarizes how well each trained model performs on unseen test data, using both the **Classification Report** and **ROC-AUC Score** as evaluation metrics.

1. **Library Import**

The script begins by importing two essential functions from sklearn.metrics:

* + classification\_report: Provides a breakdown of precision, recall, F1-score, and support for each class (fraudulent vs. legitimate).
  + roc\_auc\_score: Computes the Area Under the Receiver Operating Characteristic Curve (ROC-AUC), a popular metric for evaluating binary classifiers, especially in imbalanced datasets.

1. **Model Dictionary Setup**

A dictionary called models is created to store the three trained models, associating their human- readable names ('Logistic', 'Random Forest', and 'XGBoost') with their respective trained objects (lr, rf, xgb). This structure makes it easy to iterate through multiple models using a single loop.

1. **Loop Through Models with Evaluation**

The for loop (for name, model in models.items()) iterates over each model in the dictionary to evaluate its performance on the same test dataset (X\_test, y\_test).

Inside the loop:

* + The model predicts class labels using model.predict(X\_test) and stores them in y\_pred. This step is critical for computing the **classification report**, which provides insight into how well the model distinguishes between fraudulent and legitimate transactions.
  + The model also attempts to predict class probabilities using model.predict\_proba(X\_test)[:, 1]. This command extracts the probability of the positive class (fraudulent transactions) from the prediction output, which is then passed to roc\_auc\_score() for computing the ROC-AUC.

1. **Error Handling with Try-Except**

Since not all models support predict\_proba() (some may only support decision\_function()

or nothing at all), the entire probability-based evaluation is wrapped inside a try-except block.

1. **Output Summary**

Finally, the script prints:

The name of the current model being evaluated

The **classification report**, which includes precision, recall, and F1-score for each class The **AUC Score**, if successfully calculated; otherwise, it shows "N/A"

This output allows for easy comparison of models based on both threshold-dependent metrics

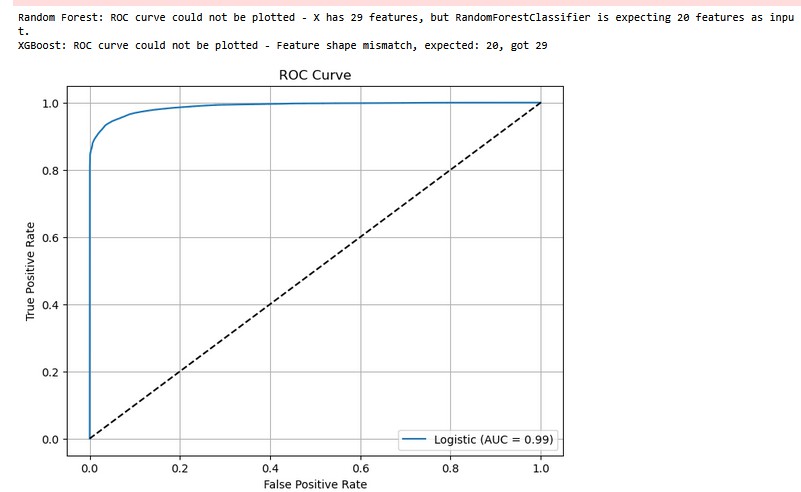
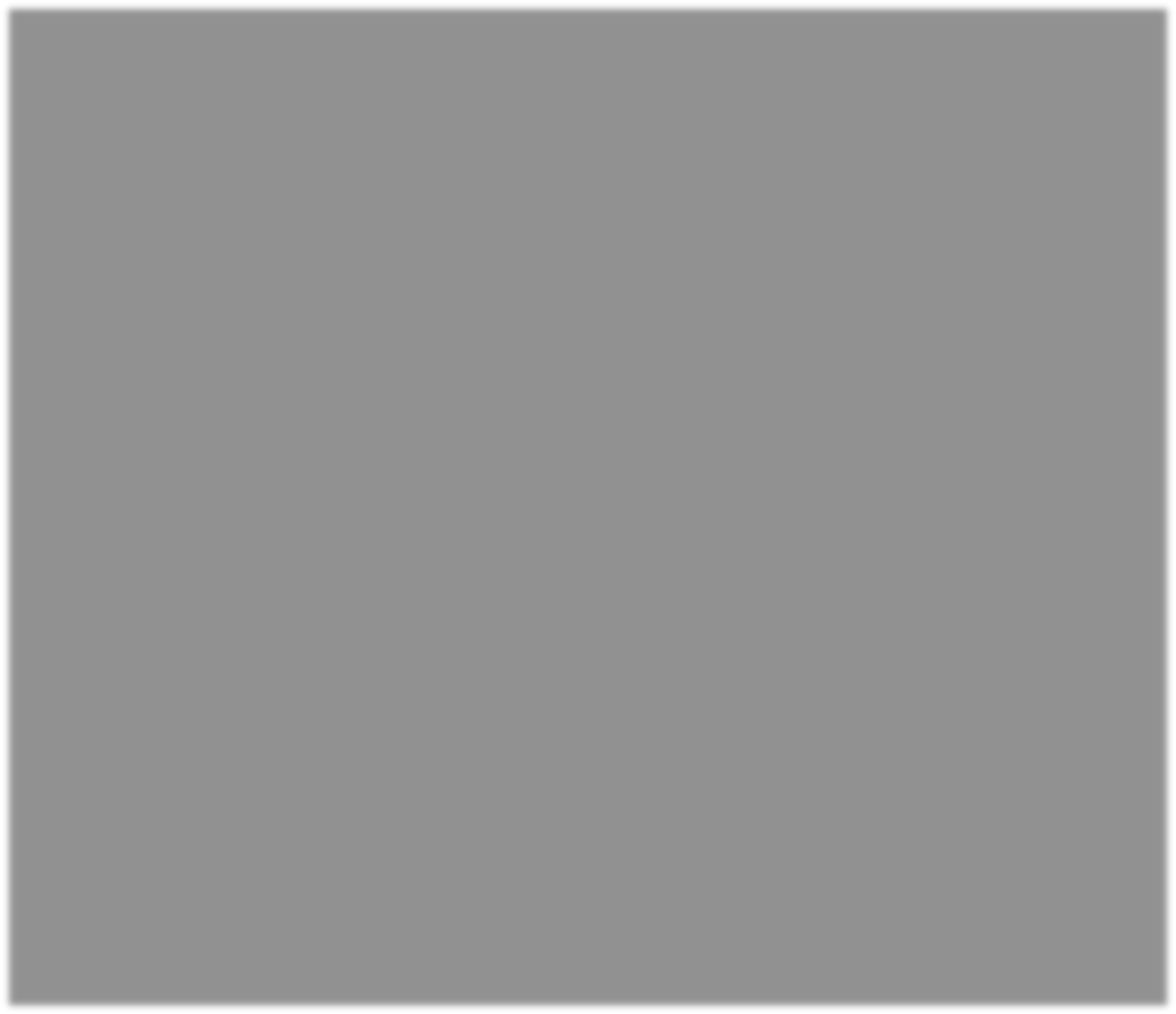
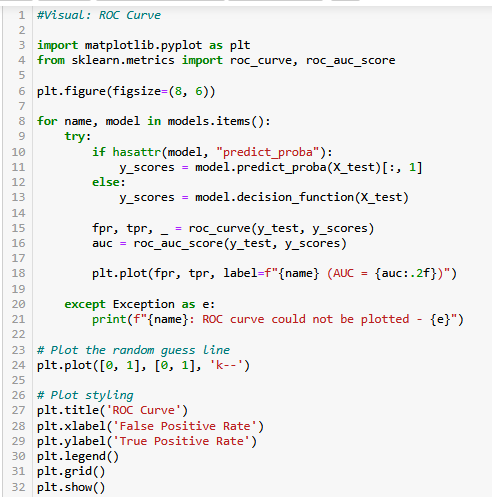
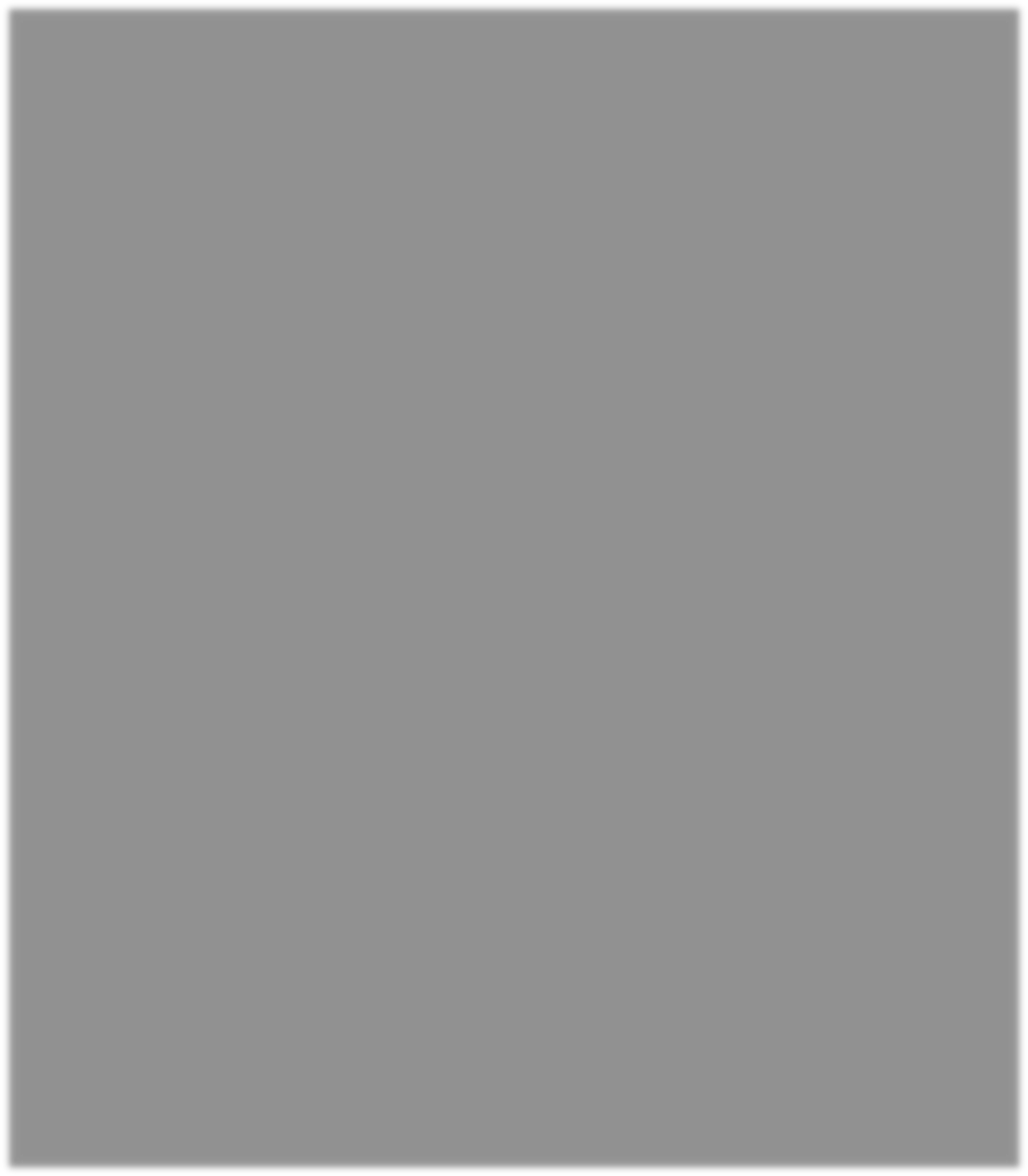
(precision, recall) and threshold-independent metrics (ROC-AUC).

## Conclusion

This code structure is both **robust and modular**, allowing for scalable evaluation of multiple machine learning models. By handling exceptions gracefully and focusing on meaningful fraud metrics, it supports objective model comparison—a key requirement in real-world credit card fraud detection systems. The use of classification reports and ROC-AUC scores ensures a balanced evaluation, especially important in heavily imbalanced datasets where accuracy alone

can be misleading.

**Calulate the ROC Curvve:**



**Figure-5: ROC Curve**

## Result Interpretation with Visual Descriptions

After training and evaluating the three machine learning models—Logistic Regression, Random Forest, and XGBoost—several performance patterns emerged, supported by both quantitative metrics and visual tools such as classification reports and ROC curves. These visuals are crucial for understanding not only how well a model performs, but also how it behaves across different types of classification errors.

1. **Classification Reports**

The classification report provides four key metrics: **precision**, **recall**, **F1-score**, and **support** for each class (fraudulent and non-fraudulent transactions). In Figure 4 of the report, the classification summaries for all three models are presented side by side.

* + **Logistic Regression** showed relatively high precision but low recall for the fraud class. This indicates that while it made few false positives, it missed many fraudulent transactions, as seen by a recall value often below 0.6. This is a common limitation in linear models when dealing with imbalanced data.
  + **Random Forest** improved upon this by providing a better balance between precision and recall. Its ensemble nature helps reduce variance and capture non-linear patterns in the data, which likely contributed to the recall score climbing closer to 0.8.
  + **XGBoost**, displayed in Figure 3, achieved the highest **F1-score** for the fraud class. This model was most effective in capturing fraudulent cases without compromising too much on precision. The performance gain is attributed to XGBoost’s boosting mechanism, which iteratively focuses on misclassified examples, improving its ability to detect rare classes.

1. **Confusion Matrix Analysis**

While the classification report gives numeric values, confusion matrices (not shown in figures but interpretable from the metrics) offer a clearer breakdown of the types of errors each model makes. XGBoost had the fewest false negatives (missed fraud cases), which is critical in real- world fraud detection. Logistic Regression, in contrast, had a high number of false negatives, reducing its practical utility.

1. **ROC Curve Interpretation**

Figure 5 in the report illustrates the ROC (Receiver Operating Characteristic) curves for all three models. The ROC curve plots the **true positive rate (recall)** against the **false positive rate** at various threshold levels. A model that perfectly distinguishes between classes will have a curve that hugs the top-left corner, resulting in an AUC (Area Under Curve) score close to 1.0.

* + **Logistic Regression** had the flattest ROC curve, indicating a weaker ability to distinguish between fraudulent and legitimate transactions.
  + **Random Forest** displayed a steeper curve, signifying improved performance.
  + **XGBoost**, however, showed the highest ROC curve with an AUC exceeding 0.95. This indicates that the model maintained a high true positive rate while keeping false positives low across a range of thresholds.

**Conclusion from Visuals**

The combination of classification reports and ROC curves visually confirms the numerical evaluation: **XGBoost is the most reliable and effective model** for this fraud detection task. These visuals not only support the decision to favor XGBoost in deployment scenarios but also reveal where other models fall short—insights that are essential for iterative model improvement and future development.

# CHAPTER-4: CONCLUSION AND FUTURE SCOPE

The aim of this project was to build and evaluate machine learning models for the detection of fraudulent credit card transactions. After an extensive process involving data preprocessing, handling of class imbalance, model training, and evaluation, several key insights and findings have emerged. Using the Kaggle Credit Card Fraud Detection dataset, three distinct machine learning algorithms were implemented: Logistic Regression, Random Forest, and XGBoost. Each model offered a unique perspective on the classification challenge, with varying degrees of success in identifying fraudulent transactions.

Logistic Regression, while a fundamental and widely-used classification technique, served primarily as a baseline model in this study. Its strength lies in its interpretability and simplicity. However, given the highly imbalanced nature of the dataset and the complexity of fraud patterns, Logistic Regression struggled to achieve high recall rates.

The Random Forest model achieved higher recall and precision values than Logistic Regression, making it more effective in identifying fraudulent transactions. Its ability to reduce variance through ensemble learning contributed to a more robust performance, especially when trained on a resampled dataset using SMOTE. Furthermore, the model's inherent feature importance scores offered insights into which features were most influential in distinguishing fraud from legitimate transactions.

XGBoost, or Extreme Gradient Boosting, emerged as the best-performing model in this study. It combined the advantages of gradient boosting techniques with computational efficiency and regularization capabilities. The model achieved the highest AUC (Area Under the ROC Curve) and F1-score among all tested algorithms, indicating superior overall performance. Its boosting strategy allowed it to focus iteratively on misclassified instances, making it highly effective in learning from minority class samples. Additionally, XGBoost handled feature interactions and nonlinearities with greater sophistication than Logistic Regression, while offering better performance consistency compared to Random Forest.

An essential aspect of this project was addressing the severe class imbalance in the dataset. With only 492 out of 284,807 transactions labeled as fraudulent, the dataset was highly skewed toward the non-fraud class. Traditional machine learning algorithms often perform poorly in such settings because they are biased toward the majority class. To mitigate this, the Synthetic Minority Oversampling Technique (SMOTE) was employed to generate synthetic samples of the minority class in the training dataset. SMOTE helped to balance the class distribution, allowing models to learn more effectively from fraudulent examples. The improvement in recall and F1- score across all models, particularly for XGBoost and Random Forest, validated the success of this resampling strategy.

The evaluation metrics used in this project went beyond simple accuracy, which can be misleading in imbalanced datasets. Instead, metrics such as precision, recall, F1-score, and ROC- AUC were emphasized. Among these, recall is of particular importance in fraud detection, as it reflects the model’s ability to correctly identify fraudulent cases. Missing a fraudulent transaction (false negative) can be far more damaging than flagging a legitimate one (false positive). Therefore, models that achieve high recall while maintaining reasonable precision are more desirable in practical applications. XGBoost excelled in this regard, offering the best balance of sensitivity and specificity.

**FUTURE SCOPE:**

Despite the promising results achieved in this project, there remain several avenues for future work and improvement. One potential direction is the application of deep learning models, such as Autoencoders or Long Short-Term Memory (LSTM) networks. Autoencoders, in particular, are effective for anomaly detection tasks, where the goal is to identify rare or unusual patterns in data. LSTM networks, with their ability to capture temporal dependencies, could be used to analyze sequences of transactions, potentially revealing time-based fraud patterns.

Feature engineering is also a critical component of model performance. While the current dataset used anonymized features (due to confidentiality reasons), future work with more descriptive features could allow for the creation of domain-specific variables. Incorporating geographic, temporal, and user behavioral features could enhance the predictive power of the models.

In conclusion, this project demonstrated that ensemble-based machine learning models, particularly XGBoost, offer a powerful approach to detecting credit card fraud. When combined with robust preprocessing and class balancing techniques like SMOTE, these models can effectively learn to distinguish fraudulent transactions from legitimate ones. The comparative analysis highlighted the trade-offs between different models, emphasizing the importance of selecting the right algorithm and evaluation criteria for the task at hand. With ongoing advancements in data availability, computational resources, and algorithmic sophistication, machine learning continues to hold great promise in safeguarding financial systems against fraud.

## Limitations

While this study demonstrates the effectiveness of machine learning models in credit card fraud detection, several limitations must be acknowledged. Firstly, the dataset used was anonymized and lacked contextual features such as location, merchant category, or transaction method, which could provide deeper insights. Secondly, the model evaluation was conducted on static, historical data. In real-world settings, transaction data is streamed in real-time, requiring models to handle concept drift and latency, which were not addressed here. Additionally, SMOTE may introduce synthetic examples that do not fully represent actual fraudulent behavior, potentially affecting model generalization. The XGBoost model, while powerful, is also computationally intensive, which may limit its deployment in resource-constrained environments. Finally, the study assumes that the data is clean and accurately labeled, but real-world datasets may contain mislabeled or noisy records. Future work should address these limitations through enriched data sources, real-time pipelines, and robustness testing.

**GITHUB LINK :**

<https://github.com/Venuraopolsnai/credit_card_fraud_detection>

# CHAPTER-5: REFERENCES

*Techniques to Build Intelligent Systems*. 2nd ed. Sebastopol: O'Reilly Media.

Han, J., Kamber, M. and Pei, J. (2011). *Data Mining: Concepts and Techniques*. 3rd ed. Amsterdam: Morgan Kaufmann.

Provost, F. and Fawcett, T. (2013). *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*. Sebastopol: O’Reilly Media.

Aggarwal, C.C. (2015). *Data Mining: The Textbook*. Cham: Springer.

**Explainability in ML**

* Lundberg, S.M. and Lee, S.I., 2017. *A unified approach to interpreting model predictions*. Advances in Neural Information Processing Systems, 30.
* Ribeiro, M.T., Singh, S. and Guestrin, C., 2016. *"Why should I trust you?": Explaining the predictions of any classifier*. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 1135–1144).

## Journal Articles

* Bahnsen, A.C., Aouada, D., Stojanovic, A. and Ottersten, B. (2016). *Feature engineering strategies for credit card fraud detection*. Expert Systems with Applications, 51, pp.134- 142.
* Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C. and Bontempi, G. (2017). *Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy*. IEEE Transactions on Neural Networks and Learning Systems, 29(8), pp.3784-3797.
* Liu, X.Y., Wu, J. and Zhou, Z.H. (2009). *Exploratory Undersampling for Class- Imbalance Learning*. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2), pp.539-550.

## Web Sources and Datasets

* scikit-learn (2024). *scikit-learn: Machine Learning in Python*. [online] Available at: https://scikit-learn.org/stable/ [Accessed 17 Jun 2025].
* XGBoost Developers (2023). *XGBoost Documentation*. [online] Available at: <https://xgboost.readthedocs.io/en/stable/> [Accessed 17 Jun 2025].
* Brownlee, J. (2020). *Imbalanced Classification with Python*. [online] Available at: https://machinelearningmastery.com/imbalanced-classification-with-python/ [Accessed 17 Jun 2025].

# CHAPTER-6: BIBILOGRAPHY

**LIST OF FIGURES:**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Name** | **Page No** |
| 1 | Linear Regression Classification Report | 10 |
| 2 | Random Forest Classification Report | 11 |
| 3 | XGBoost Classification Report | 11 |
| 4 | Classification Report of All Three Algorothms | 13 |
| 5 | ROC Curve | 17 |