

**School of Computing, Engineering and Digital Technologies**

**Department of Computing and Games**

**Teesside University**

**Middlesbrough TS1 3BA**

**Analysis, Design and Implementation Report**

**Date: 07/01/2005**

**D3747948**

**Kusuma Mekathoti**

**Supervisor: Han**

ACKNOWLEDGEMENTS

I would like to express my heartfelt appreciation to my project supervisor, Han, for his unwavering support, insightful guidance, and invaluable suggestions throughout the course of this research. His expertise and encouragement have been instrumental in shaping both the direction and the successful outcome of this project.

My sincere thanks also go to all the lecturers at Teesside University. Their dedication, knowledge, and commitment to excellence have significantly enriched my academic experience. I am truly grateful for the opportunity to learn and grow in such an intellectually stimulating environment.

Finally, I would like to extend my gratitude to Teesside University for providing the resources, facilities, and support necessary to undertake this research. This experience has been both rewarding and transformative, and I deeply appreciate the opportunities it has offered me to develop professionally and personally

Contents

[ACKNOWLEDGEMENTS 2](#_Toc196759557)

[Abstract 5](#_Toc196759558)

[Introduction 5](#_Toc196759559)

[Research Problem 7](#_Toc196759560)

[Research Questions 7](#_Toc196759561)

[Aim 7](#_Toc196759562)

[Literature review 8](#_Toc196759563)

[Research Gaps and Challenges 10](#_Toc196759564)

[METHODOLOGY 10](#_Toc196759565)

[Detailed Methodological Approach 12](#_Toc196759566)

[Data Understanding and Exploration 12](#_Toc196759567)

[Model Evaluation 13](#_Toc196759568)

[Hyperparameter Tuning 13](#_Toc196759569)

[Model Interpretation and Validation 14](#_Toc196759570)

[Methodological Challenges and Potential Solutions 14](#_Toc196759571)

[Data Analysis 15](#_Toc196759572)

[Dataset Preparation and Preprocessing 15](#_Toc196759573)

[Initial Exploration 15](#_Toc196759574)

[Preprocessing Steps 15](#_Toc196759575)

[Model Training and Evaluation 16](#_Toc196759576)

[Handling Data Imbalance 16](#_Toc196759577)

[Models Trained 16](#_Toc196759578)

[Evaluation Metrics 16](#_Toc196759579)

[Model Results Summary 16](#_Toc196759580)

[Challenges Encountered and Solutions Implemented 17](#_Toc196759581)

[Severe Class Imbalance 17](#_Toc196759582)

[Overfitting on Synthetic Data 17](#_Toc196759583)

[Model Interpretability 17](#_Toc196759584)

[Computational Costs 17](#_Toc196759585)

[LEGAL, ETHICAL, AND PROFFECTIONAL ISSUES 17](#_Toc196759586)

[Legal Issues 18](#_Toc196759587)

[Regulatory Compliance 18](#_Toc196759588)

[Liability for False Positives and Negatives 19](#_Toc196759589)

[Ethical Issues 19](#_Toc196759590)

[Bias and Fairness in Machine Learning Models 19](#_Toc196759591)

[Transparency and Explainability 20](#_Toc196759592)

[Informed Consent 20](#_Toc196759593)

[Professional Issues 20](#_Toc196759594)

[Accountability and Responsibility 20](#_Toc196759595)

[Professional Competence 21](#_Toc196759596)

[IMPLEMENTATION 21](#_Toc196759597)

[Implementation 21](#_Toc196759598)

[Data Collection and Understanding 21](#_Toc196759599)

[Source of Data 21](#_Toc196759600)

[Exploratory Data Analysis (EDA) 22](#_Toc196759601)

[Data Preprocessing 22](#_Toc196759602)

[Handling Imbalanced Datasets 23](#_Toc196759603)

[Model Selection and Training 23](#_Toc196759604)

[Model Choice 23](#_Toc196759605)

[Training the Model 24](#_Toc196759606)

[Model Evaluation 24](#_Toc196759607)

[Model Interpretation 24](#_Toc196759608)

[Model Deployment 25](#_Toc196759609)

[Real-Time Processing 25](#_Toc196759610)

[Continuous Monitoring and Updates 25](#_Toc196759611)

[RESULTS AND DISCUSSIONS 25](#_Toc196759612)

[Results 25](#_Toc196759613)

[Overall Performance Metrics 26](#_Toc196759614)

[Discussion 27](#_Toc196759615)

[CONCLUSION 28](#_Toc196759616)

[Research Summary 28](#_Toc196759617)

[Limitation 29](#_Toc196759618)

[Final Thoughts 29](#_Toc196759619)

[References 29](#_Toc196759620)

[Appendix 30](#_Toc196759621)

# Abstract

Credit card fraud remains a significant and continually evolving challenge within financial systems, often resulting in substantial financial losses and eroding consumer confidence in online transactions. With the rapid expansion of digital banking and e-commerce, the need for robust, real-time fraud detection mechanisms has become increasingly critical. However, the detection of fraudulent activity poses substantial difficulties due to two primary factors: the extreme imbalance between legitimate and fraudulent transactions, and the adaptive, ever-changing tactics employed by fraudsters.

This research explores the application of machine learning techniques to improve the detection of fraudulent credit card transactions using real-world datasets. Given the rarity of fraud cases relative to normal activity, the study applies both traditional models—such as Random Forest and Logistic Regression—and advanced ensemble methods like XGBoost, LightGBM, and Isolation Forest to capture subtle anomalies indicative of fraud. To address the imbalance in the dataset, the Synthetic Minority Over-sampling Technique (SMOTE) is used to generate synthetic examples of minority class instances, thereby enhancing model learning.

Model performance is evaluated not just through overall accuracy, but also via key metrics including precision, recall, F1-score, and ROC-AUC, which provide a more nuanced understanding of effectiveness in identifying rare fraud events. To enhance interpretability, SHAP values are employed to explain the contribution of individual features to the model’s decisions, supporting transparency and accountability in predictions.

The results reveal that ensemble-based models, particularly LightGBM and XGBoost, achieve the highest detection performance when combined with appropriate data balancing strategies. Ultimately, this project offers a practical and scalable framework for improving fraud detection systems, highlighting the importance of combining predictive accuracy with explainability and ethical considerations in real-world applications.

# Introduction

Credit card use is now more widespread than ever thanks to the rapid expansion of digital payment systems. Transactions are now quicker and easier, but there is a greater chance of fraud, which causes major issues for both banks and consumers. Major financial losses are caused by credit card fraud, which also erodes public confidence in the financial system. Therefore, it is crucial to have accurate fraud detection skills in order to safeguard funds and maintain strong client trust.

Because fraudulent transactions account for a very small percentage of all transactions—typically less than 0.5%—detecting credit card fraud is particularly challenging. Because traditional machine learning models primarily learn from normal transactions, this significant imbalance makes it easy for them to overlook the warning signs of fraud. Because of this, basic models may be highly accurate just by assuming that every transaction is authentic, but they are still unable to identify actual fraud cases. In order to minimise financial losses and keep expenses down, effective fraud detection models must not only be generally reliable but also have high accuracy (few false alarms) and high recall (capture the majority of frauds).

The primary objectives of this project are to: (1) use machine learning to detect fraudulent credit card transactions, despite the extremely low frequency of fraud cases; and (2) develop a robust approach that enhances precision and recall while maintaining the models' comprehensibility. Two key questions are the focus of the study: (1) What techniques can improve machine learning models' ability to identify fraud in extremely unbalanced datasets? (2) How crucial are feature selection and engineering to improving the accuracy and clarity of fraud detection models?

A comprehensive strategy that incorporates data cleaning, sophisticated methods for managing unbalanced data, feature development, model selection, and rigorous evaluation techniques has been employed to address these research questions. The popular Credit Card Fraud Detection dataset, which includes anonymised transaction information and represents the common imbalance issues present in actual fraud detection situations, is used in this study.

Due to the significant imbalance between fraudulent and authentic transactions in the dataset, SMOTE (Synthetic Minority Oversampling Technique) is a crucial technique employed in this investigation. To assist balance the data, SMOTE generates fresh, artificial instances of fraudulent transactions, preventing the legal transactions from overpowering the models' ability to identify fraud tendencies. The models are better able to identify infrequent but significant fraud incidents when SMOTE is used during training because it provides a more balanced mix of fraudulent and valid transactions.

Selection and design of features are also crucial components of the strategy. To make the models simpler and easier to understand, feature correlation analysis is first used to identify and eliminate extraneous features that might cause issues. The most valuable characteristics are then selected using feature importance rankings from ensemble models such as Random Forest. Effective feature selection not only improves prediction accuracy but also clarifies the models, which is important for financial institutions that must justify fraud detection choices to regulators and stakeholders.

The ability of each machine learning model to handle classification tasks in unbalanced datasets led to its exploration in this study. Because it is straightforward and simple to comprehend, logistic regression is used as a foundational model. The ensemble approach Random Forest was selected due to its robustness against overfitting and capacity to represent intricate relationships. Last but not least, XGBoost, which excels at classification tasks, is employed because of its inherent overfit prevention features and its capacity to handle unbalanced data.

Accuracy, precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) are some of the measures used to assess the models' performance. However, precision and recall are prioritised over overall accuracy due to the class imbalance. Reducing the quantity of valid transactions that are mistakenly flagged as fraudulent (false positives) is crucial because doing so could irritate clients and damage the company's reputation. In order to prevent financial losses by ensuring that actual fraud cases are not overlooked (false negatives), recall is equally important.

The results from the experimentation reveal that handling class imbalance through SMOTE significantly improves the models' recall without overly sacrificing precision. Moreover, ensemble methods like Random Forest and XGBoost outperform the simpler Logistic Regression model in terms of both predictive power and stability, especially when combined with careful feature selection.

In summary, this study demonstrates that machine learning models can be highly successful in identifying credit card fraud when modified to manage unbalanced datasets. A robust framework for addressing fraud detection in real-world scenarios is produced by employing strategies including ensemble learning, feature engineering, SMOTE resampling, and thorough assessment metrics. The results also emphasise how critical it is to carefully develop features and modify models in order to produce clearer models and higher prediction accuracy, both of which are essential for applying fraud detection systems in actual financial situations.

A diagram of a fraud detection

AI-generated content may be incorrect.

## Research Problem

For both customers and financial organisations, credit card fraud is a serious issue since it undermines trust in digital payment systems and results in significant financial losses. Because there is a significant disparity between legal and fraudulent transactions, fraud detection is challenging. Conventional machine learning models frequently produce too many false alarms while failing to correctly detect fraud. In order to enhance model performance and comprehension, the intricacy of transaction data necessitates meticulous feature engineering and selection. Effective, scalable techniques that can manage the data imbalance, precisely identify fraud, and clarify the decision-making process are desperately needed. In order to increase the accuracy and dependability of credit card fraud detection systems, this study intends to investigate and enhance machine learning approaches, with a particular emphasis on addressing imbalance and feature optimisation.

## Research Questions

1. How well can machine learning models detect credit card fraud in datasets where fraud cases are much fewer than legitimate transactions, and what methods can help improve their performance?
2. How do feature engineering and feature selection help improve the accuracy and understanding of credit card fraud detection models?

## Aim

In datasets where credit card fraud instances are significantly less frequent than legal ones, the research aims to develop and evaluate efficient machine learning models for identifying credit card fraud. By employing clever feature engineering and selection strategies, as well as sophisticated data balancing techniques like SMOTE, this study seeks to increase the accuracy, precision, and recall of fraud detection systems. Furthermore, the study seeks to ensure that the models are not only accurate in predicting fraud but also simple to comprehend, assisting financial institutions in identifying and responding to fraudulent activity. To minimise financial losses and safeguard clients, the ultimate goal is to create a dependable and flexible system that can be applied in real-world situations.

# Literature review

In today’s digital age, online transactions have become a convenient part of everyday life, allowing for quick and easy purchases. However, this convenience also brings with it a growing risk: credit card fraud. Fraudulent transactions cost millions of dollars each year, creating significant financial losses not only for consumers but also for financial institutions, merchants, and businesses worldwide. This widespread issue has sparked the need for advanced solutions to detect and prevent fraudulent activities. One such solution that has gained significant attention is machine learning (ML), which is proving to be a powerful tool in the fight against fraud. By analyzing vast amounts of transaction data, ML algorithms can identify unusual patterns and behaviors that may indicate fraudulent activity. However, the application of machine learning to fraud detection is not without its challenges. A key issue is the extreme imbalance between fraudulent and legitimate transactions in the datasets. Fraudulent transactions are extremely rare, often making up less than 1% of the total transactions, which creates highly skewed data. This imbalance makes it difficult for traditional machine learning models to effectively identify fraud without being overwhelmed by the majority class of legitimate transactions. Furthermore, the complexity of transaction behaviors, including variations in spending patterns, locations, and timeframes, adds another layer of difficulty to developing effective fraud detection systems. As a result, building reliable, accurate, and efficient fraud detection systems remains a major challenge in the field of machine learning, requiring innovative techniques and approaches to address both the data imbalance and the complexity of transaction behaviors.

In this literature review, I’ll discuss how researchers have approached these problems in recent years, what techniques have worked, and importantly, what challenges and gaps still remain that need addressing.

A diagram of a person's process

AI-generated content may be incorrect.

Handling Imbalanced Datasets

One of the primary challenges in detecting credit card fraud is the severe imbalance between classes within the data. In most cases, fraudulent transactions make up less than 0.5% of the total transaction volume. This significant disparity often leads machine learning models to become biased toward the majority class—legitimate transactions—resulting in poor detection performance for the minority class. Therefore, many models struggle to accurately identify fraudulent activity, highlighting the need for strategies specifically designed to address this imbalance during model training and evaluation.

To address this, **resampling techniques** have become popular, especially **SMOTE (Synthetic Minority Oversampling Technique)**. Recent research by Odeyale et al. (2023) showed that combining SMOTE with feature selection techniques like ANOVA-F helped improve fraud detection rates significantly, achieving a recall rate of over 97% ([Odeyale et al., 2023](https://ejournal.upi.edu/index.php/JCS/article/view/70802?utm_source=chatgpt.com)).

However, it’s important to use resampling correctly. A common mistake is applying SMOTE **before** splitting the dataset into training and testing sets, leading to **data leakage** and overly optimistic results. Kabane (2024) emphasized that resampling should only be done **after** splitting the data ([Kabane, 2024](https://arxiv.org/abs/2412.07437?utm_source=chatgpt.com)), protecting the integrity of model evaluation.

Yet, even with advanced techniques like SMOTE, results aren't always perfect. De la Bourdonnaye and Daniel (2022) found that on large real-world datasets, resampling methods sometimes made little practical difference or were too computationally expensive to apply at scale ([de la Bourdonnaye & Daniel, 2022](https://arxiv.org/abs/2206.13152?utm_source=chatgpt.com)). This suggests that while resampling helps, it is not a complete solution — and alternative strategies are still needed.

Feature Engineering and Feature Selection

Beyond handling class imbalance, another critical area is **feature engineering and selection**. In fraud detection, transaction data often contains many features, some of which may be redundant or irrelevant. Keeping unnecessary features can confuse models and worsen performance.

Recent innovations have tried to tackle this. A study published in 2024 introduced **FID-SOM**, a method that uses self-organizing maps (SOMs) for feature selection in imbalanced data ([FID-SOM, 2024](https://link.springer.com/article/10.1186/s40537-024-01059-5?utm_source=chatgpt.com)). This technique showed better performance than older methods like univariate feature selection or recursive feature elimination, highlighting how tailored approaches for imbalance are needed.

Additionally, deep learning methods like **autoencoders** have been explored. By using autoencoders for unsupervised feature extraction, researchers found they could create richer, more compact feature representations. A 2023 study that combined **Convolutional Autoencoders (CAE)** with sampling techniques like Random Undersampling (RUS) found noticeable improvements in detecting fraud ([Journal of Big Data, 2023](https://journalofbigdata.springeropen.com/articles/10.1186/s40537-023-00684-w?utm_source=chatgpt.com)).

Another creative idea from recent research is generating **synthetic labels** using autoencoders, improving model training when labeled data is scarce ([Springer, 2024](https://link.springer.com/article/10.1186/s40537-024-00897-7?utm_source=chatgpt.com)). These approaches show that better feature handling — not just better models — is crucial to success in fraud detection.

Model Selection and Evaluation

Choosing the right model is another important factor. Simpler models like **Logistic Regression** are still widely used because of their interpretability, but more complex models like **Random Forests**, **XGBoost**, and **LightGBM** generally perform better at capturing subtle fraud patterns.

However, there’s a catch: using standard metrics like **accuracy** can be misleading when the data is imbalanced. A model could achieve 99.9% accuracy simply by predicting everything as legitimate — completely missing fraud cases. That’s why researchers recommend metrics like **Precision**, **Recall**, **F1-score**, and **AUC-ROC**.

A 2022 study by Kulatilleke and Samarakoon emphasized that in highly imbalanced scenarios, relying on F1 and G-mean provides a clearer picture of model performance ([Kulatilleke & Samarakoon, 2022](https://arxiv.org/abs/2208.11904?utm_source=chatgpt.com)).

Moreover, another challenge that’s been getting more attention recently is **concept drift** — meaning the nature of fraud changes over time. A method that works well today might not perform tomorrow as criminals adapt their techniques. Adaptive, real-time models are suggested as a future research direction ([Springer, 2024](https://link.springer.com/article/10.1186/s40537-024-01059-5?utm_source=chatgpt.com)).

## Research Gaps and Challenges

Despite progress, major challenges still exist in credit card fraud detection:

* **Extreme Class Imbalance**: Resampling helps, but doesn't solve all problems. Models still struggle to detect rare fraud cases without introducing many false positives.
* **Data Leakage**: Mistakes in model training (like improper resampling) can lead to inflated results. More careful validation methods are needed.
* **Feature Engineering**: Even now, identifying which transaction features are most predictive — without overfitting — remains tricky. New feature selection methods like FID-SOM show promise, but broader testing is required.
* **Concept Drift**: Fraud tactics evolve constantly. Models need to adapt over time, but building dynamic fraud detection systems remains an open research problem.
* **Real-World Validation**: Many studies focus on public datasets (like the popular Kaggle fraud dataset), but fraud detection in live environments often presents new, unseen challenges — like processing massive volumes of real-time data.

Clearly, while machine learning models have achieved impressive results in controlled experiments, turning these results into reliable, real-world fraud detection systems still requires overcoming several practical hurdles.

# METHODOLOGY

In this research, the primary goal is to detect credit card fraud efficiently and accurately, even when dealing with highly imbalanced datasets. Achieving this objective requires a carefully designed methodology that considers both the nature of the problem and the practical constraints associated with real-world data. This section outlines the approach taken, the tools and technologies used, the challenges encountered along the way, and the justifications for the methodological decisions made.

Fraud detection is not just about building a good model — it involves every step, from data preprocessing and feature engineering to model training, evaluation, and fine-tuning. Therefore, this methodology is structured systematically to reflect a realistic, iterative research and development process.

Research Design and Methodology Overview

This research employs an experimental design cantered on evaluating the effectiveness of multiple machine learning algorithms using a publicly accessible credit card transaction dataset. The methodological approach is guided by the Cross-Industry Standard Process for Data Mining (CRISP-DM), a widely recognized framework that supports structured, iterative, and cyclical data analysis. This model ensures a systematic workflow—from data understanding and preparation to modeling and evaluation—allowing for continuous refinement and improved predictive performance throughout the research process.

The key stages of the methodology are:

* Data Understanding and Exploration
* Data Preprocessing and Feature Engineering
* Handling Data Imbalance
* Model Selection and Training
* Model Evaluation
* Hyperparameter Tuning
* Interpretation and Validation

Each stage is essential to ensure that the final models are both accurate and reliable, especially when deployed in environments where real-world consequences of mistakes (false positives or false negatives) can be serious.

Tools and Technologies Used

Choosing the right tools and technologies is critical to implementing the methodology effectively. Here’s a summary of what was used:

| **Tool / Technology** | **Purpose** |
| --- | --- |
| **Python (v3.9+)** | Primary programming language |
| **Jupyter Notebook** | Interactive development environment |
| **Pandas** | Data manipulation and preprocessing |
| **NumPy** | Numerical operations |
| **Matplotlib & Seaborn** | Data visualization |
| **Scikit-learn** | Machine learning algorithms and evaluation metrics |
| **XGBoost** | Gradient boosting model for enhanced performance |
| **Imbalanced-learn (imblearn)** | Techniques like SMOTE for handling imbalanced data |
| **SHAP** | Model interpretability |
| **SMOTE (Synthetic Minority Oversampling Technique)** | Oversampling minority class |
| **Google Colab / Local Machine (16GB RAM)** | Computing environment |

The choice of these tools is based on their proven reliability, rich community support, and suitability for machine learning experiments.

### Detailed Methodological Approach

## Data Understanding and Exploration

The dataset used in this study is the **Kaggle Credit Card Fraud Detection Dataset**, which contains transactions made by European cardholders over two days. The dataset includes 284,807 transactions, among which only 492 are fraudulent — a clear demonstration of class imbalance.

Initial exploration involved:

* Checking for missing values
* Understanding data types
* Statistical analysis of features
* Visualizing class distributions

Techniques like correlation matrices and distribution plots were used to get a feel for the data and identify potential features of interest.

Data Preprocessing and Feature Engineering

Before building models, the data was cleaned and prepared:

* **Scaling**: Because the dataset contained principal components (PCA-transformed features), standard scaling was used to normalize amounts and time features.
* **Feature Generation**: Derived new features like “transaction hour” from the “Time” field to capture temporal patterns in fraud.
* **Feature Selection**: Employed techniques like feature importance (using tree-based models) and mutual information scores to prioritize the most predictive features.

Effective feature engineering proved critical, as raw data alone did not always expose fraud patterns clearly.

Handling Imbalanced Data

Given the severe imbalance, special attention was needed. Three techniques were explored:

* **SMOTE**: Oversampled the minority class by generating synthetic examples.
* **Random Under sampling**: Reduced the number of majority class samples to balance the data.
* **Combined Approach**: SMOTE followed by Tomek Links under sampling to clean boundary noise.

Ultimately, **SMOTE** was selected as the primary technique because it enhanced recall without dramatically hurting precision.

Model Selection and Training

Several machine learning models were tested:

* **Logistic Regression**: A simple, interpretable baseline.
* **Random Forest**: An ensemble method offering robustness.
* **XGBoost**: A powerful boosting algorithm known for its success in Kaggle competitions.
* **Support Vector Machine (SVM)**: Tried for its strength in binary classification but found less scalable for large datasets.

Hyperparameters were initially set to default and later fine-tuned through randomized search and grid search strategies

## Model Evaluation

To ensure models were truly effective, multiple evaluation metrics were considered:

* **Precision**: Important to avoid false positives (i.e., labeling good customers as fraudsters).
* **Recall**: Crucial for catching actual fraud cases.
* **F1 Score**: Balance between precision and recall.
* **AUC-ROC**: Ability to distinguish between classes at various thresholds.

Given the context, **recall** was prioritized slightly more than precision, as missing a fraudulent transaction can be more costly than mistakenly flagging a legitimate one.

## Hyperparameter Tuning

Fine-tuning models involved adjusting parameters like:

* Number of trees (Random Forest)
* Learning rate and tree depth (XGBoost)
* Regularization penalties (Logistic Regression)

Randomized search was favored over exhaustive grid search to save computation time, without sacrificing much model performance.

## Model Interpretation and Validation

Finally, **SHAP (SHapley Additive Explanations)** values were used to interpret the best-performing models. This step helped identify which features contributed most to fraud predictions, providing insights into the underlying patterns models were learning.

### Methodological Challenges and Potential Solutions

Throughout the project, several challenges emerged:

#### Data Imbalance

Even after resampling, models still sometimes struggled with detecting rare frauds.

**Potential Solution**: Future work could explore **ensemble resampling methods** or **cost-sensitive learning**, where misclassifying fraud is heavily penalized during training.

#### Overfitting

SMOTE can sometimes cause models to overfit to synthetic minority examples.

**Potential Solution**: Using more advanced SMOTE variants like **SMOTE-ENN** or **ADASYN**, and applying stronger regularization in models like XGBoost.

#### Interpretability

Complex models like XGBoost achieve high performance but are often seen as black boxes.

**Potential Solution**: Continue to use model-agnostic interpretability tools like **SHAP** and **LIME** to ensure models can be explained to non-technical stakeholders.

#### Concept Drift

The nature of fraudulent behavior may change over time, but the static dataset doesn't account for this.

**Potential Solution**: Future studies should include temporal validation or rolling-window retraining to simulate a live system that adapts over time.

#### Justification for the Methodological Choices

Each methodological decision in this project was made with careful consideration:

* **Choice of Dataset**: Publicly available, well-studied, allowing for reproducibility.
* **Use of SMOTE**: Most suitable given the extremely low number of fraud samples.
* **Model Selection**: Focused on models balancing accuracy with interpretability — XGBoost for power, Logistic Regression for explainability.
* **Evaluation Metrics**: Precision, recall, and F1-score better reflect real-world needs compared to accuracy alone.
* **Interpretability Focus**: Financial institutions demand transparent models to comply with regulations — SHAP was essential.

By grounding each choice in either the practical needs of fraud detection or recent research findings, the methodology aims to be both rigorous and applicable beyond the academic setting.

# Data Analysis

Effective data analysis is fundamental to building reliable credit card fraud detection systems. This chapter details the process of preparing, analysing, and modelling the dataset used in this research, as well as the strategies employed to overcome inherent challenges, particularly extreme class imbalance. By systematically exploring the data, applying appropriate preprocessing techniques, training models, and evaluating their performance with the right metrics, the study ensures a thorough and robust analytical process.

## Dataset Preparation and Preprocessing

The dataset used in this research originates from a real-world, anonymized credit card transaction dataset made publicly available on Kaggle. It contains **284,807** transactions made over a two-day period by European cardholders, among which only **492** are confirmed fraud cases — representing a fraud rate of approximately **0.17%**.

A screenshot of a credit card security

AI-generated content may be incorrect.

## Initial Exploration

Upon loading the dataset, an exploratory data analysis (EDA) phase was conducted to understand the structure and characteristics of the data:

* **No Missing Values**: One key advantage was the absence of missing entries, allowing direct focus on feature selection and engineering.
* **Feature Types**: Features V1 to V28 are outputs of a PCA (Principal Component Analysis) transformation for confidentiality. Additionally, the dataset includes "Time" and "Amount" features, which were not transformed.
* **Class Distribution**: Visualization confirmed the significant class imbalance — the dataset is heavily skewed towards non-fraudulent transactions.

## Preprocessing Steps

Given the importance of data preparation, the following steps were taken:

* **Feature Scaling**: "Amount" and "Time" features were scaled using **StandardScaler**. Since PCA already standardized other features, this step was necessary to maintain consistency.
* **New Feature Creation**: A "Transaction Hour" feature was derived by transforming the "Time" variable into hours since the first transaction. This allowed the model to capture temporal patterns potentially associated with fraud.
* **Data Splitting**: The data was split into training (80%) and testing (20%) subsets, ensuring that the split was stratified to maintain the fraud class distribution across sets.

## Model Training and Evaluation

Several machine learning algorithms were applied to the prepared data, with model selection guided by their known strengths in classification tasks involving imbalance.

### Handling Data Imbalance

Before model training, the training set was augmented using **SMOTE (Synthetic Minority Oversampling Technique)** to synthetically generate additional fraud samples. SMOTE interpolates between existing fraud examples to create new, plausible samples without merely replicating existing instances, which helps prevent overfitting.

By applying SMOTE only to the training data (not the test set), the integrity of evaluation was preserved.

### Models Trained

The models selected for experimentation were:

* **Logistic Regression**: Baseline linear classifier.
* **Random Forest**: Ensemble method robust to overfitting and capable of handling non-linear data.
* **XGBoost**: Advanced boosting method, designed to enhance performance on structured datasets.
* **Support Vector Machine (SVM)**: Explored for its theoretical ability to handle small, complex datasets.

Each model underwent hyperparameter tuning through **RandomizedSearchCV** to optimize performance without incurring the computational cost of exhaustive grid searches.

### Evaluation Metrics

Given the dataset’s imbalance, **accuracy** was rejected as the primary performance metric. Instead, models were evaluated using:

* **Precision**: Ratio of correctly predicted frauds to total predicted frauds.
* **Recall**: Ratio of correctly predicted frauds to total actual frauds.
* **F1-Score**: Harmonic mean of precision and recall, balancing the two.
* **AUC-ROC**: Measures the model's ability to distinguish between classes across all thresholds.

Emphasis was placed on achieving a high **recall** without sacrificing too much **precision**, aligning with the business need to minimize undetected fraud.

## Model Results Summary

Among the tested models:

* **XGBoost** achieved the best overall performance, providing a strong balance between precision and recall with minimal overfitting.
* **Random Forest** performed slightly lower but still provided competitive results, especially in recall.
* **Logistic Regression** was fast and interpretable but lagged behind ensemble methods in detecting frauds.
* **SVM** struggled with scalability issues and took significantly longer to train, offering less practical benefit in larger datasets.

Final model comparisons were made using confusion matrices and ROC curves to visualize strengths and weaknesses.

## Challenges Encountered and Solutions Implemented

Throughout the data analysis process, several challenges emerged, each requiring thoughtful mitigation.

### Severe Class Imbalance

**Challenge**: The low number of fraud cases made it difficult for models to learn useful patterns, risking high false-negative rates.

**Solution**: Applying SMOTE selectively to the training set improved minority class representation without compromising test validity. Future work could explore ensemble-based methods like **BalancedBaggingClassifier** for further improvement.

### Overfitting on Synthetic Data

**Challenge**: There was a risk that models could overfit on synthetic samples created by SMOTE, especially in deeper decision trees.

**Solution**: Techniques like **early stopping** (especially in XGBoost) and **cross-validation** were employed to prevent overfitting. Regularization parameters such as max\_depth and min\_child\_weight were tuned carefully.

### Model Interpretability

**Challenge**: Models like XGBoost, while highly effective, operate as "black boxes," making them harder to explain to stakeholders.

**Solution**: Interpretability tools such as **SHAP (SHapley Additive exPlanations)** were used to attribute feature importance and explain individual predictions, increasing trust and transparency.

### Computational Costs

**Challenge**: Training some models, particularly SVMs, on the full dataset was time-consuming and memory intensive.

**Solution**: SVM was ultimately deprioritized in favor of models like Random Forest and XGBoost, which are more scalable. For critical models, training was done on high-memory environments (e.g., Google Colab with 16 GB RAM).

# LEGAL, ETHICAL, AND PROFFECTIONAL ISSUES

Detecting fraudulent credit card transactions is a critical aspect of protecting financial systems, preventing economic losses, and safeguarding customers. Machine learning techniques play an essential role in addressing this issue. However, the development and deployment of these models must consider a range of legal, ethical, and professional issues to ensure that the system operates within the boundaries of the law, maintains fairness, and adheres to high professional standards. This section explores the key legal, ethical, and professional issues involved in credit card fraud detection, especially in the context of using machine learning models and handling imbalanced datasets.

## Legal Issues

One of the primary legal concerns when dealing with credit card fraud detection is data privacy and protection. Financial institutions and organizations are required by law to protect sensitive customer data, including credit card details, transaction history, and personal information. Different countries have various legal frameworks to ensure data protection:

* **General Data Protection Regulation (GDPR)**: In the European Union, GDPR mandates strict rules regarding the collection, storage, and processing of personal data. Organizations need to obtain explicit consent from individuals before collecting their data, and they are required to provide mechanisms for individuals to access, correct, or delete their data.
* **California Consumer Privacy Act (CCPA)**: In the U.S., California’s CCPA provides similar protections, allowing consumers to opt out of the sale of their data, access their data, and request its deletion.
* **Payment Card Industry Data Security Standard (PCI DSS)**: This set of standards governs the protection of cardholder data. Organizations that process, store, or transmit credit card information are required to follow these standards to prevent data breaches and fraud.

When using machine learning models for fraud detection, organizations must ensure that they comply with these regulations. This means:

* **Data anonymization and encryption**: It is crucial that personal data used for training the fraud detection models is anonymized or encrypted. Direct access to personally identifiable information (PII) should be minimized.
* **Limited data usage**: Data used for training and testing models should be limited to what is necessary for the task, reducing the risk of over-collection or misuse.
* **Transparency and accountability**: Organizations should be transparent with customers regarding how their data is being used and the steps taken to protect it.

## Regulatory Compliance

Regulatory compliance is another legal concern for financial institutions. Different financial regulatory bodies govern credit card fraud detection practices, including the **Financial Conduct Authority (FCA)** in the UK and **Federal Financial Institutions Examination Council (FFIEC)** in the U.S. These bodies enforce laws and guidelines to ensure that financial institutions protect consumers and minimize fraud risk.

Machine learning-based fraud detection systems must be designed to meet these regulatory requirements. For example, if an institution is using an automated system to flag fraudulent transactions, it must ensure that the system provides accurate, reliable, and transparent results. This is critical because financial institutions may be held accountable for failing to prevent fraudulent activities, especially if their systems were not compliant with regulatory standards.

## Liability for False Positives and Negatives

A significant challenge in fraud detection is the handling of false positives (legitimate transactions flagged as fraudulent) and false negatives (fraudulent transactions that are not detected). From a legal perspective:

* **False positives**: When a legitimate transaction is incorrectly flagged as fraudulent, it may lead to an inconvenience for the customer, such as their card being temporarily blocked or their account frozen. In extreme cases, customers may take legal action for damages incurred due to the wrongful blocking of their transactions or the negative impact on their credit score.
* **False negatives**: Conversely, failing to detect a fraudulent transaction could result in significant financial losses for both the customer and the financial institution. If a model fails to identify fraudulent activity, institutions may be liable for the resulting losses, especially if it is found that the system did not meet the required standards for fraud prevention.

Organizations must strike a balance between minimizing false positives and maximizing fraud detection to limit legal risks.

## Ethical Issues

## Bias and Fairness in Machine Learning Models

Bias is a significant ethical issue in credit card fraud detection. Machine learning models are only as good as the data they are trained on. If the data used to train fraud detection systems contains biases—such as certain demographic groups being overrepresented or underrepresented—then the resulting model can be unfair and discriminatory. For example, if the training data predominantly consists of transactions from high-income individuals, the model may have a higher rate of false negatives for transactions from low-income individuals.

Key ethical concerns include:

* **Discrimination against minority groups**: If fraud detection models unfairly target specific demographic groups (e.g., based on age, gender, race, or location), they may disproportionately flag individuals from those groups as potential fraudsters. This can lead to unjust financial scrutiny or customer discrimination, harming the affected individuals.
* **Data-driven inequality**: Bias in machine learning can exacerbate existing inequalities. If fraud detection systems are not inclusive in their training data, they may not perform well for customers from diverse backgrounds, leading to unequal protection against fraud for certain groups.

To mitigate bias, organizations must ensure that the data used to train fraud detection systems is representative of the full range of customers and that efforts are made to detect and correct for any inherent biases. Additionally, regular audits of the models' performance across different demographic groups should be conducted.

## Transparency and Explainability

Machine learning models, particularly deep learning models, can sometimes operate as "black boxes," meaning it is difficult to understand how they arrive at a decision. In the context of credit card fraud detection, a lack of explainability could raise ethical concerns regarding transparency and accountability.

* **Lack of explainability**: Customers who have their transactions flagged as fraudulent should have the right to understand why their transaction was flagged. If a model’s decision-making process is opaque, customers may not trust the system or may feel that they have been treated unfairly.
* **Model interpretability**: From an ethical standpoint, fraud detection systems should be interpretable so that customers can request clarifications if they believe a mistake was made. This is important for maintaining customer trust and ensuring that their rights are respected.

Ethical guidelines in AI and machine learning emphasize the importance of explainability. Researchers and practitioners must work on techniques to make fraud detection models more transparent, such as using simpler, interpretable models or integrating post hoc interpretability tools.

## Informed Consent

Informed consent is another key ethical issue. Financial institutions must ensure that customers are fully informed about how their data is being used. This involves explaining the purposes for which their data will be used, including for fraud detection, and obtaining explicit consent.

The ethical principle of informed consent extends beyond just collecting customer data. It also involves ensuring that customers have the option to opt-out of being monitored or to access their data. This can help address concerns about data surveillance and the unauthorized use of personal information.

## Professional Issues

## Accountability and Responsibility

Professionals involved in the development and deployment of machine learning models for fraud detection must be accountable for their actions. This includes:

* **Data scientists**: Data scientists must ensure that the models they develop are thoroughly tested, validated, and meet the necessary legal and ethical standards.
* **Developers**: Developers are responsible for implementing the fraud detection system in a way that ensures transparency, privacy, and fairness.
* **Management**: Organizational leaders must ensure that adequate resources are allocated to data protection and compliance efforts, and that ethical considerations are prioritized in decision-making.

A clear structure of accountability should be established within the organization to ensure that each stakeholder is aware of their responsibilities and that any issues are promptly addressed.



## Professional Competence

Another critical professional issue is ensuring that those working on fraud detection systems possess the necessary knowledge and skills. Given the complexity of machine learning algorithms and the regulatory environment surrounding financial systems, it is essential that professionals involved in fraud detection are competent in:

* **Machine learning**: Professionals should have expertise in machine learning, particularly in the areas of model selection, evaluation, and dealing with imbalanced datasets.
* **Financial regulations**: Understanding the legal requirements and compliance standards related to financial transactions and data protection is crucial for building models that meet industry standards.
* **Ethical AI**: Developers must be well-versed in the ethical implications of AI and machine learning and be committed to ensuring fairness, transparency, and privacy in their work.

Professional organizations, such as the **Association for Computing Machinery (ACM)** and **IEEE**, provide guidelines and best practices for AI and machine learning ethics. Practicing professionals should adhere to these standards to ensure that their work is aligned with industry norms and ethical expectations.

# IMPLEMENTATION

## Implementation

The development of a credit card fraud detection system using machine learning entails a multi-stage process comprising data collection, preprocessing, model design, training, evaluation, and deployment. This section presents a detailed overview of each phase, outlining the specific methodologies and techniques employed to construct a robust and reliable detection pipeline. A primary objective of this process is to build a model that not only achieves high accuracy in identifying fraudulent transactions but also effectively addresses the inherent challenges posed by class imbalance and data complexity.

## Data Collection and Understanding

The first step in implementing a machine learning-based fraud detection system is acquiring the relevant data. Financial institutions and payment processors typically have vast amounts of transactional data, which can be used to detect patterns indicative of fraudulent behavior.

### Source of Data

For machine learning models, the most commonly used data comes from historical credit card transactions. These transactions typically contain the following features:

* **Transaction Amount**: The value of the transaction.
* **Merchant Information**: The type and name of the merchant involved in the transaction.
* **Transaction Timestamp**: The date and time the transaction occurred.
* **Location Data**: The location where the transaction was made, often extracted from the cardholder’s geolocation.
* **Cardholder Information**: Demographic information about the customer, such as their spending habits, geographical location, and account status.
* **Transaction ID and Account ID**: Unique identifiers for both the transaction and the cardholder.

To build an effective fraud detection model, it is essential that the dataset includes both fraudulent and legitimate transactions. The dataset will typically be highly imbalanced, with fraudulent transactions representing a very small percentage of the overall data, making it challenging to train accurate models. Specialized techniques for handling imbalanced datasets will be explored later in the implementation.

### Exploratory Data Analysis (EDA)

Before jumping into model building, a thorough exploratory data analysis (EDA) is necessary to understand the data and identify any underlying patterns or potential issues. During EDA, the following steps are typically performed:

* **Data distribution**: Analyze the distribution of various features (such as transaction amounts and time) to identify potential anomalies or outliers that may need to be handled.
* **Class imbalance check**: Given that fraudulent transactions are rare, this is an important step. Visualizing the ratio of fraudulent to legitimate transactions helps in determining the severity of the imbalance.
* **Correlation analysis**: Assessing correlations between different features can reveal relationships that may help improve model performance.
* **Missing data analysis**: Check for missing or incomplete records, which may require imputation or removal, depending on their impact on model performance.

### Data Preprocessing

Data preprocessing is a critical step in preparing raw transaction data for effective machine learning application, particularly in credit card fraud detection. The quality and structure of input data significantly influence model performance. In this study, several key preprocessing techniques are employed:

* **Handling Missing Values:** Incomplete data entries are addressed through either removal or imputation. Common imputation techniques include mean or mode substitution, regression-based methods, and more advanced approaches such as k-Nearest Neighbors (k-NN) imputation, depending on the nature and extent of the missing data.
* **Normalization and Standardization:** To ensure that numerical features (e.g., transaction amount, merchant category codes) contribute proportionally to the model, normalization or standardization is applied. This is especially important for algorithms sensitive to feature scaling, such as neural networks and support vector machines.
* **Categorical Encoding:** Categorical variables, including merchant identifiers, transaction types, and customer IDs, are transformed into numerical representations using encoding methods such as One-Hot Encoding or Label Encoding. This conversion is essential for enabling machine learning models to interpret non-numeric data.
* **Feature Engineering:** This process involves generating new features to enhance model predictive capabilities. Key strategies include:
  + *Rolling Aggregates:* Statistical summaries such as the average transaction amount or transaction frequency over a recent time window (e.g., past 7 or 30 days) are computed to highlight anomalies in user behaviour.
  + *Temporal Features:* Time-related variables such as the hour of transaction, day of the week, and time since the last transaction are derived to capture patterns associated with fraudulent activity.

### Handling Imbalanced Datasets

Since fraudulent transactions are rare, an imbalance between legitimate and fraudulent transactions is inevitable. This imbalance poses a significant challenge for machine learning algorithms, which tend to perform poorly when class distribution is skewed.

Several techniques can be employed to address the class imbalance:

* **Resampling**:
  + **Oversampling** the minority class (fraudulent transactions) by replicating existing instances or generating synthetic samples using techniques such as **SMOTE (Synthetic Minority Over-sampling Technique)**.
  + **Undersampling** the majority class (legitimate transactions) to balance the dataset, though this can lead to loss of valuable information.
* **Class Weights**: Most machine learning algorithms allow the specification of class weights, which assigns a higher weight to the minority class. This forces the model to pay more attention to fraudulent transactions during training.
* **Anomaly Detection**: Given that fraud is rare, anomaly detection models can be useful. These models learn the patterns of legitimate transactions and identify outliers (fraudulent transactions) that deviate significantly from normal behavior.

## Model Selection and Training

After preprocessing the data, the next step is selecting appropriate machine learning models and training them on the dataset.

### Model Choice

Several machine learning algorithms can be used for credit card fraud detection. The choice of model depends on factors such as interpretability, performance, and ability to handle imbalanced datasets. Some common models include:

* **Logistic Regression**: A simple yet powerful algorithm that can be used as a baseline for fraud detection. It outputs probabilities, which can be useful for identifying high-risk transactions.
* **Random Forest**: An ensemble method that creates multiple decision trees and aggregates their predictions. It is robust to overfitting and can handle both binary classification and imbalanced datasets effectively.
* **Gradient Boosting Machines (GBM)**: Algorithms like **XGBoost** and **LightGBM** are widely used for fraud detection due to their ability to handle large datasets and produce highly accurate models.
* **Neural Networks**: Deep learning models, particularly feedforward neural networks, can be used to model complex relationships between features. However, they often require larger datasets and more computational power.
* **Anomaly Detection Models**: Algorithms like **Isolation Forest** and **One-Class SVM** are particularly effective when dealing with imbalanced data, as they focus on detecting outliers.

### Training the Model

Once a model has been selected, the next step is to train it using the prepared dataset. During training, the model learns the relationship between the input features and the target variable (fraud or not fraud). Key steps in this process include:

* **Cross-Validation**: To avoid overfitting and ensure the model generalizes well, k-fold cross-validation is used. The dataset is divided into k subsets, and the model is trained on k-1 subsets, with the remaining subset used for validation. This process is repeated for each subset.
* **Hyperparameter Tuning**: Hyperparameters such as the learning rate, number of trees (for Random Forest), or the number of layers (for neural networks) can significantly impact model performance. Techniques like **Grid Search** or **Random Search** can be used to find the best combination of hyperparameters.

### Model Evaluation

Evaluating the model's performance is crucial to determine how well it is detecting fraudulent transactions. The following metrics are commonly used in fraud detection:

* **Accuracy**: The proportion of correct predictions (both fraud and legitimate transactions) to the total number of predictions. However, accuracy is often misleading in imbalanced datasets.
* **Precision**: The ratio of true positives (correctly predicted fraudulent transactions) to the total predicted positives (both true positives and false positives). High precision ensures that fraudulent transactions are detected with minimal false alarms.
* **Recall**: The ratio of true positives to the total actual positives (true positives and false negatives). High recall ensures that most fraudulent transactions are identified.
* **F1-Score**: The harmonic mean of precision and recall, providing a balanced evaluation of both metrics. This is particularly important in fraud detection, where both false positives and false negatives need to be minimized.
* **ROC-AUC**: The area under the Receiver Operating Characteristic curve. AUC measures the ability of the model to distinguish between fraudulent and legitimate transactions, with a higher value indicating better model performance.

### Model Interpretation

Given that fraud detection systems have a significant impact on customers, it is important to be able to interpret the model's decisions. **SHAP (Shapley Additive Explanations)** or **LIME (Local Interpretable Model-Agnostic Explanations)** are popular tools for explaining model predictions. These techniques provide insight into which features (e.g., transaction amount, merchant type, etc.) are driving the model’s decisions.

## Model Deployment

Once the model has been trained and evaluated, it is ready for deployment. The deployment process involves integrating the fraud detection system into the existing infrastructure of the financial institution or payment processor.

### Real-Time Processing

For fraud detection to be effective, the model must operate in real-time. This requires efficient processing of incoming transaction data, where the model predicts whether each transaction is fraudulent or legitimate within milliseconds. To achieve this:

* **Streamlining the Model**: Ensure that the model can process new data points efficiently, without introducing latency.
* **Batch Processing**: For systems with limited real-time processing capabilities, batch processing may be used to evaluate transactions periodically.

### Continuous Monitoring and Updates

Fraud patterns evolve over time, so it is essential to regularly update the fraud detection model to maintain its effectiveness. This can be done through **model retraining** using new transaction data, and by periodically reviewing the system’s performance to detect any degradation.

# RESULTS AND DISCUSSIONS

## Results

The Logistic Regression model was trained and evaluated to detect fraudulent credit card transactions in a highly imbalanced dataset. The classification report, confusion matrix, and ROC curve were used to assess the model's performance.

The classification report showed that for legitimate transactions (class 0), the model achieved a precision of 1.00, a recall of 0.98, and an F1-score of 0.99. For fraudulent transactions (class 1), the model achieved a precision of 0.09, a recall of 0.95, and an F1-score of 0.17. The overall accuracy across all transactions was 98%, calculated on a total of 18,239 transactions.

The macro-averaged precision and recall were 0.55 and 0.97, respectively, with a macro-averaged F1-score of 0.58. The weighted average F1-score was 0.99, heavily influenced by the large proportion of legitimate transactions.

The confusion matrix indicated that the model correctly classified 17,794 legitimate transactions and incorrectly classified 401 legitimate transactions as fraud. For the fraudulent transactions, the model correctly identified 42 cases and misclassified 2 as legitimate.

The Receiver Operating Characteristic (ROC) curve showed an AUC (Area Under Curve) of 0.9846, suggesting excellent model performance in distinguishing between the two classes. The ROC curve itself showed a sharp rise towards the top-left corner of the plot, indicating a high true positive rate with a low false positive rate across a range of thresholds.

Visual inspection of the confusion matrix confirmed that false negatives (missed fraudulent transactions) were extremely rare, while false positives (legitimate transactions flagged as fraud) were more common.

Overall, the Logistic Regression model demonstrated strong recall for the minority fraud class while maintaining a high overall accuracy, albeit at the cost of lower precision for fraudulent transaction predictions.

A screenshot of a computer

AI-generated content may be incorrect.

## Overall Performance Metrics

| **Metric** | **Value** |
| --- | --- |
| **Accuracy** | 98% |
| **Precision (Fraudulent Class 1)** | 0.09 |
| **Recall (Fraudulent Class 1)** | 0.95 |
| **F1-Score (Fraudulent Class 1)** | 0.17 |
| **Macro Avg Precision** | 0.55 |
| **Macro Avg Recall** | 0.97 |
| **Macro Avg F1-Score** | 0.58 |
| **Weighted Avg F1-Score** | 0.99 |
| **True Negatives (TN)** | 17,794 |
| **False Positives (FP)** | 401 |
| **True Positives (TP)** | 42 |
| **False Negatives (FN)** | 2 |
| **ROC-AUC Score** | 0.9846 |

**Brief Notes:**

* **High recall (0.95)**: most frauds detected.
* **Low precision (0.09)**: many false alarms.
* **Excellent ROC-AUC (0.9846)**: strong overall separability.

## Discussion

The results highlight several important insights into the performance of Logistic Regression when applied to credit card fraud detection in an imbalanced dataset.

Firstly, the model’s outstanding recall of 0.95 for fraudulent transactions suggests that it is highly effective at identifying fraud. This is a critical requirement in fraud detection systems, where missing fraudulent activities (false negatives) can lead to significant financial and reputational damages. Even at the expense of precision, maximizing recall is often prioritized in practical fraud detection scenarios, where it is preferable to investigate false alarms than to overlook actual frauds.

However, the extremely low precision (0.09) for fraudulent transactions indicates that the majority of transactions flagged as fraud were actually legitimate. While a high recall minimizes the risk of missed fraud, a low precision can create operational inefficiencies, such as increased manual reviews by fraud investigation teams and potential customer dissatisfaction when legitimate transactions are blocked or challenged unnecessarily. In a real-world application, this trade-off would need to be carefully managed depending on the institution's tolerance for false positives and resource availability.

The confusion matrix provided further evidence of the model’s behavior. It demonstrated that while false negatives were nearly eliminated (only 2 missed fraudulent transactions), false positives were relatively high (401 legitimate transactions incorrectly flagged). Given the extreme class imbalance, this result is not unexpected. Logistic Regression models typically produce a probability output, and in this case, the threshold for classifying a transaction as fraudulent was likely tuned to favor sensitivity over specificity.

The high ROC-AUC score of 0.9846 is particularly encouraging, indicating that the model has an excellent ability to separate fraudulent from legitimate transactions overall. The ROC curve’s steep ascent toward the top-left corner supports this, showing that even across various thresholds, the model consistently maintains a strong true positive rate while keeping the false positive rate reasonably low.

The macro-averaged metrics also provide useful information. A macro-averaged F1-score of 0.58 reflects that when treating each class equally, the model’s balanced performance is moderate. This underlines the difficulty in achieving high performance across both classes simultaneously in highly imbalanced datasets. The weighted average metrics, skewed by the dominance of legitimate transactions, were significantly higher, demonstrating how accuracy and weighted metrics can be misleading if interpreted without considering class distribution.

The use of resampling techniques to balance the training data likely contributed to the model’s ability to achieve high recall for fraudulent transactions. Without balancing the dataset, the model might have been biased towards predicting only legitimate transactions, given their overwhelming majority in the data.

An important observation is the potential for threshold adjustment to fine-tune the balance between precision and recall. Logistic Regression outputs probabilities, and by adjusting the threshold above the default 0.5, it may be possible to achieve a better balance between identifying frauds and minimizing false alarms. However, any threshold adjustment would need to be informed by the specific operational needs and cost structures associated with fraud investigation and customer relations.

It is also worth noting that the Logistic Regression model's inherent simplicity brings advantages beyond performance metrics. Its transparency and interpretability make it particularly suitable for domains like finance, where explainability of decisions is crucial for compliance and customer communication. Although feature importance was not directly explored in the presented results, Logistic Regression coefficients offer a straightforward way to understand which features most influence the model’s decision-making process.

Despite its simplicity, the model’s performance demonstrates that traditional machine learning models like Logistic Regression remain strong contenders for fraud detection tasks when properly tuned and evaluated. While more complex models, such as ensemble methods or deep learning, might achieve marginally better precision-recall trade-offs, Logistic Regression offers a strong baseline with easier interpretability and faster deployment times.

In summary, the results show that the Logistic Regression model is capable of effectively detecting fraudulent credit card transactions in a highly imbalanced dataset by maximizing recall, even though it sacrifices precision. The high ROC-AUC and strong recall metrics are particularly promising, suggesting that with additional threshold tuning or the inclusion of ensemble techniques, further performance improvements could be realized.

# CONCLUSION

## Research Summary

This research focused on detecting fraudulent credit card transactions using machine learning techniques, particularly logistic regression, on a highly imbalanced dataset. The model was evaluated using key performance metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. Results demonstrated that while the model achieved a high overall accuracy of 98%, precision for fraudulent transactions was relatively low (0.09), highlighting the challenge of false positives. However, the recall for fraudulent cases was extremely high (0.95), meaning that most actual fraud cases were successfully identified. The ROC-AUC score of 0.9846 confirmed that the model performed exceptionally well in distinguishing fraudulent from legitimate transactions. Despite the simplicity of logistic regression, it proved highly effective, especially when combined with appropriate resampling and evaluation techniques. The study emphasized the importance of considering recall over precision in fraud detection systems, where the cost of missing a fraudulent transaction is significantly higher than falsely flagging a legitimate one. The findings suggest that even basic machine learning models, if properly tuned, can achieve competitive results in sensitive applications like fraud detection.

## Limitation

While the model achieved promising results, several limitations were identified. Firstly, the dataset was highly imbalanced, which influenced overall performance metrics like accuracy and weighted F1-scores, possibly making them appear overly optimistic. Although resampling and careful evaluation were used, more sophisticated techniques such as SMOTE or ensemble methods could further improve precision. Secondly, only logistic regression was explored; advanced models like random forests, XGBoost, or neural networks were not tested, which may outperform logistic regression, especially in complex fraud patterns. Thirdly, feature engineering was limited; the study did not deeply explore creating new variables or feature selection methods that might boost model interpretability and performance. Additionally, the model evaluation was based on historical data and may not fully capture evolving fraud strategies used in real-world scenarios. Model generalizability to new, unseen types of fraud remains uncertain. Finally, operational costs associated with high false positive rates were not quantitatively assessed, which is crucial for deploying fraud detection systems in a live environment. Future work should address these gaps by testing more algorithms, applying advanced balancing techniques, and considering real-world operational impacts.

# Final Thoughts

This research demonstrated the effectiveness of logistic regression in detecting fraudulent credit card transactions in a highly imbalanced dataset. While the model achieved a high overall accuracy, the performance metrics highlighted the importance of balancing precision and recall, especially in fraud detection systems. A recall of 0.95 showcased the model’s ability to identify most fraudulent transactions, which is crucial given the potentially high costs associated with missing fraud cases. However, the trade-off came with a low precision (0.09), leading to many false positives that would require manual investigation.

Despite these challenges, the findings underscore the strength of simple models like logistic regression when used in combination with appropriate data handling and evaluation techniques. The high ROC-AUC score (0.9846) also indicates that logistic regression can effectively distinguish between fraudulent and legitimate transactions, making it a valuable tool in fraud detection.

Nevertheless, the study also revealed key limitations, including the need for better precision, more sophisticated resampling techniques, and the potential of exploring more advanced models. Future work should focus on improving model performance with feature engineering, advanced algorithms, and deeper analysis of false positive impacts in a real-world operational environment. Ultimately, this research paves the way for further exploration into machine learning models for financial fraud detection.

# References

1. Al-Maari, A. and Abdulnabi, M. (2023) 'Credit Card Fraud Transaction Detection Using a Hybrid Machine Learning Model', IEEE Available at: 10.1109/SCOReD60679.2023.10563915.
2. Alotaibi, O., Pardede, E. and Tomy, S. (2023) 'Cleaning Big Data Streams: A Systematic Literature Review', *Technologies (Basel),*11(4), pp. 101. Available at: <https://doi.org/10.3390/technologies11040101>
3. Arya, M. and Hanumat, S.G. (2020) 'DEAL - 'Deep Ensemble ALgorithm' Framework for Credit Card Fraud Detection in Real-Time Data Stream with Google TensorFlow', *Smart Science,*8(2), pp. 71–83. Available at: <https://doi.org/10.1080/23080477.2020.1783491>
4. Barillà, G.B. (2024) 'PAYMENT FRAUD IN ITALY AND IN IN THE UK: COMPARATIVE ANALYSIS AND PROSPECTS FOR REFORM', *Diritto Degli Affari,*(1), pp. 13–37.
5. Berhane, T.*et al.*(2023) 'A Hybrid Convolutional Neural Network and Support Vector Machine-Based Credit Card Fraud Detection Model', *Mathematical Problems in Engineering,*2023(1) Available at: <https://doi.org/10.1155/2023/8134627>
6. Bhowmik, S. and Howlader, J. (2025) 'Online payment fraud monitoring and detection: Performance analysis of tree-based Ensemble Machine Learning models', IEEE Available at: 10.1109/COMSNETS63942.2025.10885622.
7. Braithwaite, J. (2024) '‘Authorized Push Payment’ Bank Fraud: What Does an Effective Regulatory Response Look Like?', *Journal of Financial Regulation,*10(2), pp. 174–193. Available at: <https://doi.org/10.1093/jfr/fjae006>
8. Chang, V.*et al.*(2024a) 'Investigating Credit Card Payment Fraud with Detection Methods Using Advanced Machine Learning',
9. Chang, V.*et al.*(2024b) 'Investigating Credit Card Payment Fraud with Detection Methods Using Advanced Machine Learning',
10. Chang, V.*et al.*(2022) 'Digital payment fraud detection methods in digital ages and Industry 4.0',
11. Gaikar, P.*et al.*(2024) 'Online Payment Fraud Detection System', *International Research Journal of Innovations in Engineering and Technology,*8(10), pp. 232. Available at: <https://doi.org/10.47001/IRJIET/2024.810032>
12. Gandhi, H.*et al.*(2024) 'Navigating the Complexity of Money Laundering: Anti–money Laundering Advancements with AI/ML Insights', *International Journal on Smart Sensing and Intelligent Systems,*17(1) Available at: <https://doi.org/10.2478/ijssis-2024-0024>
13. Ganji, V.R., Chaparala, A. and Sajja, R. (2023) 'Shuffled shepherd political optimization‐based deep learning method for credit card fraud detection', *Concurrency and Computation,*35(10), pp. n/a. Available at: <https://doi.org/10.1002/cpe.7666>
14. Hajek, P., Abedin, M. and Sivarajah, U. (2022) 'Fraud Detection in Mobile Payment Systems using an XGBoost-based Framework',
15. Jose, N.N.*et al.*(2024) '"Optimizing Payment Transaction Security: Utilizing Gradient Boosting Machines for Fraud Detection"', IEEE Available at: 10.1109/ICCSP60870.2024.10543774.
16. Karthikeyan, T., Govindarajan, M. and Vijayakumar, V. (2024) 'Enhancing Financial Fraud Detection Through Chimp-Optimized Long Short-Term Memory Networks', *Traitement Du Signal,*41(2), pp. 835–845. Available at: <https://doi.org/10.18280/ts.410224>
17. Khalid, A.R.*et al.*(2024) 'Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach', *Big Data and Cognitive Computing,*8(1), pp. 6. Available at: <https://doi.org/10.3390/bdcc8010006>
18. Kim, H.J. and Rhee, J.S. (2025) 'Navigating the Fraud Frontier: Machine Learning Solutions for Credit Card Security', *Tehnički Vjesnik,*32(2), pp. 730–738. Available at: <https://doi.org/10.17559/TV-20241013002057>
19. Morton-Huddleston, W., Layne, C. and Riehle, H. (2018) 'IMPROVING ACCOUNTABILITY AND ANALYTICS TO PREVENT AND DETECT IMPROPER PAYMENTS', *The Journal of Government Financial Management,*67(3), pp. 48–54.
20. Nabrawi, E. and Alanazi, A. (2023) 'Fraud Detection in Healthcare Insurance Claims Using Machine Learning', *Risks (Basel),*11(9), pp. 160. Available at: <https://doi.org/10.3390/risks11090160>
21. Nanduri, J.*et al.*(2020) 'Microsoft Uses Machine Learning and Optimization to Reduce E-Commerce Fraud', *INFORMS Journal on Applied Analytics,*50(1), pp. 64–79. Available at: <https://doi.org/10.1287/inte.2019.1017>
22. Nur-E-Arefin, M., Ravana, S.D. and Ariyaluran Habeeb, R.A. (2025) 'A method utilizing deep learning and reinforcement learning for credit card fraud detection', *Intelligent Decision Technologies,*Available at: <https://doi.org/10.1177/18724981251329428>
23. Sekar, M. (2022) *Machine learning for auditors : automating fraud investigations through artificial intelligence.*New York, New York: Apress Media LLC.
24. Singh, A. and Jain, A. (2019) 'An Empirical Study of AML Approach for Credit Card Fraud Detection–Financial Transactions', *International Journal of Computers, Communications & Control,*14(6), pp. 670. Available at: <https://doi.org/10.15837/ijccc.2019.6.3498>
25. Sisodia, D. and Sisodia, D.S. (2023) 'A hybrid data‐level sampling approach in learning from skewed user‐click data for click fraud detection in online advertising', *Expert Systems,*40(2), pp. n/a. Available at: <https://doi.org/10.1111/exsy.13147>
26. Suesserman, M.*et al.*(2023) 'Procedure code overutilization detection from healthcare claims using unsupervised deep learning methods', *BMC Medical Informatics and Decision Making,*23(1), pp. 1–196. Available at: <https://doi.org/10.1186/s12911-023-02268-3>
27. Tekkali, C.G. and Natarajan, K. (2024) 'Assessing CNN’s Performance with Multiple Optimization Functions for Credit Card Fraud Detection', *Procedia Computer Science,*235, pp. 2035–2042. Available at: <https://doi.org/10.1016/j.procs.2024.04.193>
28. Thair Ali, N.*et al.*(2024) 'Improving credit card fraud detection using machine learning and GAN technology', *BIO Web of Conferences,*97, pp. 76. Available at: <https://doi.org/10.1051/bioconf/20249700076>
29. Vanini, P.*et al.*(2023a) 'Online payment fraud: from anomaly detection to risk management', *Financial Innovation (Heidelberg),*9(1), pp. 66–25. Available at: <https://doi.org/10.1186/s40854-023-00470-w>
30. Vanini, P.*et al.*(2023b) 'Online payment fraud: from anomaly detection to risk management', *Financial Innovation (Heidelberg),*9(1), pp. 66–25. Available at: <https://doi.org/10.1186/s40854-023-00470-w>
31. Wu, D.*et al.*(2023) 'Adaptive Individual-Level Cognitive Driving Anomaly Detection Model Exclusively Using BSMs', *Vehicles,*5(4), pp. 1275–1293. Available at: <https://doi.org/10.3390/vehicles5040070>

# Appendix

* 1. A blue rectangular bar graph

     AI-generated content may be incorrect.
  2. A graph of a bar

     AI-generated content may be incorrect.

1. Classification Report:
2. precision recall f1-score support
3. 0 1.00 0.97 0.99 56864
4. 1 0.06 0.92 0.11 98
5. accuracy 0.97 56962
6. macro avg 0.53 0.95 0.55 56962
7. weighted avg 1.00 0.97 0.99 56962
8. ROC-AUC: 0.9795478721877047
9. A graph with numbers and a number in blue squares

   AI-generated content may be incorrect.
10. A graph of a curve

    AI-generated content may be incorrect.
11. Classification Report:
    1. precision recall f1-score support

0 1.00 1.00 1.00 56864

1 0.90 0.86 0.88 98

accuracy 1.00 56962

macro avg 0.95 0.93 0.94 56962

weighted avg 1.00 1.00 1.00 56962

ROC-AUC: 0.9845864784433751

1. A graph with numbers and a blue square

   AI-generated content may be incorrect.
2. A graph of a curve

   AI-generated content may be incorrect.
3. Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 56864

1 0.72 0.84 0.77 98

accuracy 1.00 56962

macro avg 0.86 0.92 0.89 56962

weighted avg 1.00 1.00 1.00 56962

ROC-AUC: 0.9916722175645722

1. A graph with numbers and a blue square

   AI-generated content may be incorrect.
2. A graph of a curve

   AI-generated content may be incorrect.
3. Classification Report:

precision recall f1-score support

0 1.00 1.00 1.00 56864

1 0.55 0.88 0.68 98

accuracy 1.00 56962

macro avg 0.78 0.94 0.84 56962

weighted avg 1.00 1.00 1.00 56962

ROC-AUC: 0.9817036602907905

1. A graph with numbers and a blue square

   AI-generated content may be incorrect.
2. A graph of a curve

   AI-generated content may be incorrect.
3. A green and white graph

   AI-generated content may be incorrect.
4. A graph with blue bars

   AI-generated content may be incorrect.
5. A graph with red squares

   AI-generated content may be incorrect.
6. A graph of a transaction hours

   AI-generated content may be incorrect.