First name and surname
Narmin Alakbarli

Register number
59906

Field:
Computer Science

Specialization:
Data Scientist

Mode: Full-time

STATEMENT

Aware of my responsibility, I hereby declare that the Master's thesis submitted, titled: **Machine Learning Methods in Credit Card Fraud Detection**, was entirely written by me.

I also declare that the above work does not violate copyright within the meaning of the Act of 4 February 1994 on Copyright and Related Rights (Journal of Laws No. 24, item 83as amended) and personal rights protected by civil law.

Therefore, the thesis mentioned does not contain data and information that I obtained in a prohibited way. This diploma thesis has not previously been the basis of any other official procedure related to the awarding of university diplomas or professional titles.

I declare that I grant WSB University free rights to enter and process my diploma thesis in the anti-plagiarism system.

Dąbrowa Górnicza, date 03/02/2026
.....

Signature **Narmin Alakbarli**

Faculty of Applied Sciences

Computer Science

MA THESIS

Narmin Alakbarli

Machine Learning Methods in Credit Card Fraud Detection

MA THESIS

written under the supervision of

Dr. Bartłomiej Zieliński

Approved ……
Date and supervisor's signature

Dąbrowa Górnicza 2026

TABLE OF CONTENTS

CHAPTER 1 - INTRODUCTION

1.1 Background and importance of payment cards

Payment cards; credit, debit and prepaid -have become indispensable in modern commerce. They provide convenience to consumers and enable merchants to accept payment instantly with lower cash-handling risks. Consequently, card transactions account for a large share of retail spending. Global card purchase volume (for goods, services and cash advances) grew to **$51.92 trillion in 2024** [13], continuing a steady upward trajectory since the early 2010s. Even small changes in fraud rates can therefore translate into billions of dollars in absolute losses.

The evolution of payment cards has transformed the financial landscape over the past several decades. What began as simple magnetic stripe cards in the 1970s has evolved into sophisticated payment ecosystems encompassing chip-based cards, contactless payments, mobile wallets, and digital payment platforms. This transformation has been driven by technological advances, changing consumer preferences, and the need for faster, more secure transaction processing. The shift from cash to card-based payments has been particularly pronounced in developed economies, where card transactions now represent in the majority of retail purchases.

The post-pandemic recovery accelerated the adoption of digital wallets and contactless payments. FICO notes that **digital wallets now account for 53 % of global e-commerce spending** [15], reflecting consumers' shift to online shopping and mobile checkout. Similarly, real-time payment systems (RTP) have gained traction, offering instant settlement between banks. This rapid digitization has fundamentally altered how consumers interact with payment systems, creating new opportunities for both legitimate commerce and fraudulent activity.

While these innovations improve user experience, they also **increase the attack surface**: card data travels through more channels, is stored in more applications and can be compromised via phishing, malware or database breaches. In January 2024, the so-called "mother of all breaches" exposed **26 billion user records** from popular services, including login credentials that fraudsters could leverage for account takeover. This incident highlights the scale of the security challenge facing the payment industry. As payment systems become more interconnected and data flows through multiple intermediaries, the potential points of compromise multiply, requiring sophisticated detection mechanisms that can identify fraudulent patterns across diverse transaction channels.

1.2 Growing scale of card-fraud losses

Fraud remains a persistent and evolving threat. The **Nilson Report** estimated that **global card fraud losses reached $33.83 billion in 2023** [13], a 1.1 % increase over 2022. Although the loss rate fell to **6.58 cents per $100 spent**, the sheer growth in transaction volume means absolute losses continue to rise. The United States is disproportionately affected: despite accounting for roughly 25 % of worldwide card spending, it suffers **about 42 % of global fraud losses**, partly

because American merchants and issuers have been slow to adopt stringent authentication measures. Nilson projects cumulative global card-fraud losses of **$403.88 billion over the next decade**, signaling that fraudsters will continue to exploit vulnerabilities despite industry investments in security.

Other jurisdictions report similar trends. Credit and debit card fraud losses were **$34.36 billion in 2022**, according to WalletHub [13], with the United States bearing **38.83 % of global losses**. The same source shows that the **total value of credit-card fraud reached $275 million in 2024**, a **12 % increase** over 2023. Merchants also face indirect costs: for each dollar of fraud loss, merchants incur approximately **$3.75 in related costs** (including chargebacks, fees, investigation and customer support) [12]. The Federal Trade Commission (FTC) reported that **US consumers lost more than $10 billion to fraud of all types in 2023** [14], a 14 % increase over 2022. Although not all of this relates to card payments, it underscores the escalating economic harm.

Fraud patterns differ across regions and channels. **Card-not-present (CNP) fraud**—transactions where the card is not physically swiped, such as online or over the phone—has become the dominant form [2]. FICO states that card fraud (dominated by CNP fraud) **accounted for $34 billion in losses in 2023 and is projected to cost $404 billion over the next decade**. By **2030, CNP fraud alone could reach $49 billion globally**. The adoption of EMV chips has reduced counterfeit card fraud at point-of-sale, but criminals have shifted to digital skimming, infecting e-commerce sites with malware to capture card data. A two-month operation by Europol and law enforcement agencies in 17 countries identified **443 online sellers** whose payment data had been compromised, with **119 million cards found for sale on the dark web**, representing **$9.4 billion in preventable losses**. Card skimming devices still remain a problem: the FBI estimates they **cost US consumers and banks about $1 billion annually**.

1.3 Card-fraud typologies

Understanding fraud typologies is critical for designing detection models. Different fraud types exhibit distinct behavioral patterns, transaction characteristics, and temporal signatures that can be leveraged for detection. The following non-exhaustive list summarizes major categories:

Counterfeit and stolen cards (card-present fraud): Fraudsters use cloned magnetic stripe cards or stolen physical cards to make in-store purchases. EMV chips and contactless technology have reduced but not eliminated this threat. Counterfeit fraud often involves rapid, high-value transactions as criminals attempt to maximize gains before the card is reported stolen. The spatial pattern of these transactions - often occurring far from the cardholder's usual locations-provides a key detection signal.

CNP fraud: Criminals leverage stolen card details to complete online or mail-order transactions. Credentials may be obtained via phishing, malware, database leaks or **digital skimming**. CNP fraud has become the dominant fraud type as e-commerce has grown, representing over 70% of fraud losses in many markets. The absence of physical card

verification makes this channel particularly vulnerable, requiring sophisticated behavioral analysis to detect anomalous patterns.

Account takeover (ATO): Attackers gain access to a legitimate cardholder's account, change contact details and then conduct transactions. The proliferation of large-scale data breaches facilitates ATO, as criminals can combine stolen credentials from multiple sources to build comprehensive profiles of victims. ATO attacks often involve subtle changes to account settings before fraudulent transactions, making early detection challenging.

Synthetic identity fraud: Criminals create new identities using fabricated or composite personal information. These can be difficult to detect because there is no legitimate "victim" complaining of fraud. Synthetic identities are often built over months or years, establishing credit history before being exploited. This long-term pattern requires temporal analysis to identify, as the initial transactions may appear legitimate.

Merchant collusion and refund fraud: Merchants or employees conspire with criminals to process bogus transactions or misrepresent transaction amounts. Refund fraud involves requesting refunds to newly opened or altered accounts. These schemes often involve repeated transactions with specific merchants, creating spatial and temporal patterns that can be detected through network analysis.

Fraudsters continually adapt their tactics to evade detection. Examples include "pharming" (redirecting web traffic to fraudulent sites), SIM-swapping to intercept one-time passwords, "man-in-the-browser" malware that modifies payment data, and **"flash fraud" rings** that overwhelm banks with a high volume of small fraudulent transactions in a short time. These evolving tactics underscore the need for adaptive detection systems that can learn from new patterns and adjust their models accordingly. The temporal aggregation observed in flash fraud - multiple transactions within minutes - exemplifies how understanding fraud typologies informs feature engineering and model design.

1.4 Regulatory landscape and industry responses

Regulators and card networks have introduced rules to mitigate fraud. In the European Economic Area, the **Second Payment Services Directive (PSD2)** mandates **strong customer authentication (SCA)** for many electronic payments. PSD2's enforcement in 2021/2022 led to a surge in card declines but ultimately helped reduce fraud by requiring two-factor authentication (e.g., a dynamic password plus a biometric). The 3-D Secure (3DS) protocol, which adds an authentication step for CNP transactions, has evolved to **version 2.3**, improving user experience and adoption [2]. Nevertheless, 3DS remains optional in some markets such as the United States, limiting its global impact.

Card networks and acquirers deploy real-time fraud screening systems that score transactions based on hundreds of variables. For example, **FICO's Falcon Fraud Manager** uses consortium data from over **10 000 financial institutions** to train models that learn from billions of tagged

transactions. ANZ Bank reported that FICO's system helped it prevent **AUD 112 million** ($112 million) in fraud in 2023. Bank Mandiri in Indonesia increased detected fraud by **216 %** after automating alerts and real-time declines. These successes highlight the potential of data-driven fraud management.

Yet regulation and technology alone cannot eliminate fraud. Strict authentication measures may degrade user experience and increase false declines (legitimate transactions incorrectly flagged as fraud), leading to revenue loss and customer frustration. Banks must balance security, convenience and operational costs.

1.5 Machine-learning and deep-learning in fraud detection

Traditional rule-based systems codify patterns (e.g., "flag all transactions over $500 in foreign countries") but struggle with evolving fraud tactics and complex interactions between variables. These systems require manual maintenance and cannot easily adapt to new fraud patterns. As fraudsters develop increasingly sophisticated methods, static rule sets become less effective, leading to higher false negative rates and delayed detection.

Machine learning (ML) algorithms offer a more adaptive approach by learning patterns from historical data and adjusting as new data arrives. As described in Sinčák's thesis [2], classification algorithms such as **logistic regression, decision trees, random forests, gradient boosting and neural networks** can differentiate between fraudulent and legitimate transactions. These models can capture non-linear relationships and complex feature interactions that rule-based systems miss. Anomaly detection methods (e.g., isolation forest, autoencoders) flag unusual behaviour without labelled fraud examples, making them valuable for detecting novel fraud patterns.

Deep learning architectures, including feedforward neural networks, recurrent neural networks (RNNs), and long short-term memory (LSTM) networks, have shown particular promise for fraud detection. These models can automatically learn hierarchical feature representations and capture temporal dependencies in transaction sequences. For instance, LSTM networks can model how a cardholder's spending patterns evolve over time, identifying deviations that may indicate fraud. Convolutional neural networks (CNNs) and attention mechanisms have also been applied to fraud detection, enabling models to focus on the most relevant features for each transaction.

However, ML and DL face several critical challenges in the fraud detection domain:

- **Class imbalance:** Fraud events are rare (often <0.5 % of transactions), causing models to learn to predict the majority class [7]. This fundamental challenge means that naive models may achieve high accuracy while failing to detect any fraud cases. Oversampling techniques like **random oversampling** and **SMOTE** [7] address this by generating or duplicating fraud examples, while undersampling reduces the majority class. Lestari [3] emphasised the importance of balancing to achieve reasonable recall, demonstrating that

proper class balancing can improve fraud detection rates by 20–30 percentage points. More advanced techniques, including generative adversarial networks (GANs) [8] for synthetic fraud generation, have shown promise but require careful tuning to avoid overfitting.

Feature engineering: Transaction attributes (amount, time, merchant category) provide some signal, but adding **spatial and temporal context** - geolocation relative to the cardholder's address, time of day, day of week, recency of previous transactions—can reveal patterns of legitimate behaviour and highlight anomalies [1,3]. Research by Lestari [3] and others has shown that incorporating spatial-temporal features can improve model performance significantly. For example, the distance between a transaction location and the cardholder's residence, combined with the time since the last transaction, can help identify suspicious patterns that basic transaction features miss.

Real-time performance: Detection systems must score transactions within milliseconds to avoid delaying legitimate purchases. Computationally heavy models may need approximation or pre-scoring. This constraint favors simpler models like gradient boosting over complex deep learning architectures, though recent advances in model compression and hardware acceleration have made real-time deep learning inference more feasible.

Data privacy and security: Access to detailed transaction and location data raises privacy concerns. **Federated learning**—training models across multiple banks without sharing raw data—has emerged as a solution; research combining federated learning with graph neural networks has shown promise for fraud detection [1], though results remain largely in academic prototypes. Privacy-preserving techniques such as differential privacy and homomorphic encryption offer additional protection but often come with performance trade-offs.

Explainability: Regulatory compliance and operational trust require models to provide reasons for flags. Tree-based models and attention mechanisms can offer some interpretability, but deep neural networks are often considered "black boxes." Techniques like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) help quantify feature contributions, enabling investigators to understand why a transaction was flagged. This explainability is crucial for regulatory compliance and building trust with both customers and internal stakeholders.

1.6 Research gap and motivation

Despite significant advances in machine learning for fraud detection, several critical gaps remain in the literature. First, while spatial-temporal features have shown promise in isolated studies, their integration with class balancing techniques has not been systematically evaluated across a comprehensive range of ML and DL models. Second, most research uses synthetic or anonymized datasets that may not reflect real-world complexities, limiting the generalizability of

findings. Third, comparative studies between ML/DL models and operational rule-based systems are rare, making it difficult for practitioners to assess the practical benefits of adopting advanced detection methods.

The work of Lestari (2024) demonstrated the importance of spatial-temporal features and balancing techniques, but focused primarily on deep learning architectures. Sinčák's (2023) thesis provided valuable insights into traditional ML methods and benchmarking against real systems, but did not extensively explore spatial-temporal context. This thesis bridges these gaps by integrating the strengths of both approaches, creating a comprehensive framework that evaluates spatial-temporal features and balancing techniques across the full spectrum of ML and DL models, while maintaining rigorous comparison with rule-based systems.

1.7 Problem statement and research objectives

Given the continued growth of card transactions and fraud, and the limitations of current detection systems, this thesis aims to investigate **how integrating geolocation and temporal data with data-balancing techniques influences the performance of ML and DL algorithms in detecting credit-card fraud**. The research addresses the fundamental question of whether combining spatial-temporal context with sophisticated balancing strategies can significantly improve fraud detection performance while maintaining practical deployability in banking environments.

Specifically, the research seeks to:

Evaluate whether adding spatial and temporal features improves detection rates versus models using only transaction-level variables. This objective addresses RQ1 by systematically comparing baseline models trained on basic features with models incorporating geolocation, temporal patterns, and their interactions.

Assess the effectiveness of various balancing strategies (random oversampling, SMOTE, undersampling, hybrid methods) in combination with spatial-temporal features. This objective (RQ2) examines how different balancing techniques interact with enriched feature sets, identifying optimal combinations for different model types.

Compare the performance of traditional ML algorithms (logistic regression, random forests, gradient boosting) to deep-learning architectures (feedforward neural networks, recurrent neural networks) and hybrid ensembles. This objective (RQ3) provides a comprehensive evaluation of model families, enabling practitioners to select appropriate architectures based on their specific requirements.

Benchmark the best model against a real-world rule-based fraud-detection system, analysing recall, precision, false positive rate, inference time and cost-benefit trade-offs. This objective (RQ4) addresses the critical gap in comparing ML/DL approaches with operational systems, providing evidence-based guidance for system upgrades.

Explore the implications for deployment in banking environments, considering regulatory compliance, customer experience and operational constraints. This objective (RQ5) extends beyond pure performance metrics to examine practical considerations that determine real-world adoption.

The overarching hypothesis is that **spatial–temporal context and proper class balancing significantly enhance the ability of ML/DL models to detect fraud**, and that such models can approach or surpass the performance of legacy rule-based systems while offering superior adaptability to evolving fraud patterns. This hypothesis is grounded in the theoretical understanding that fraudsters exhibit both spatial and temporal aggregation patterns, and that class imbalance fundamentally limits model learning without appropriate mitigation strategies.

1.8 Structure of the thesis

To address these objectives, the thesis is organised as follows:

Chapter 2 reviews the literature on fraud methods, detection techniques and ML/DL algorithms. It draws on both academic research and industry reports to provide a comprehensive overview, synthesizing findings from key studies and identifying research gaps that motivate our integrated framework.

Chapter 3 formulates the research questions in detail and describes the dataset and feature engineering, including the integration of spatial and temporal attributes and class-balancing strategies. This chapter presents data exploration visualizations and statistical summaries that inform the experimental design.

Chapter 4 outlines the methodology, describing the ML and DL models, hyper-parameter tuning, evaluation metrics and cost-benefit analysis. Figures illustrate the experimental pipeline, balancing techniques, and model architecture, while tables document the evaluation framework.

Chapter 5 presents the experimental results, comparing models across different feature sets and balancing techniques, and discussing performance trade-offs with supporting figures (e.g., ROC curves, confusion matrices) and comprehensive performance tables.

Chapter 6 benchmarks the best model against a real-world rule-based system, analysing operational considerations and implications for banks. Comparison tables highlight the advantages of ML/DL approaches while acknowledging the strengths of rule-based systems.

Chapter 7 concludes the thesis, summarizing contributions, discussing limitations and proposing avenues for future research, such as federated learning and improved contextual data sources.

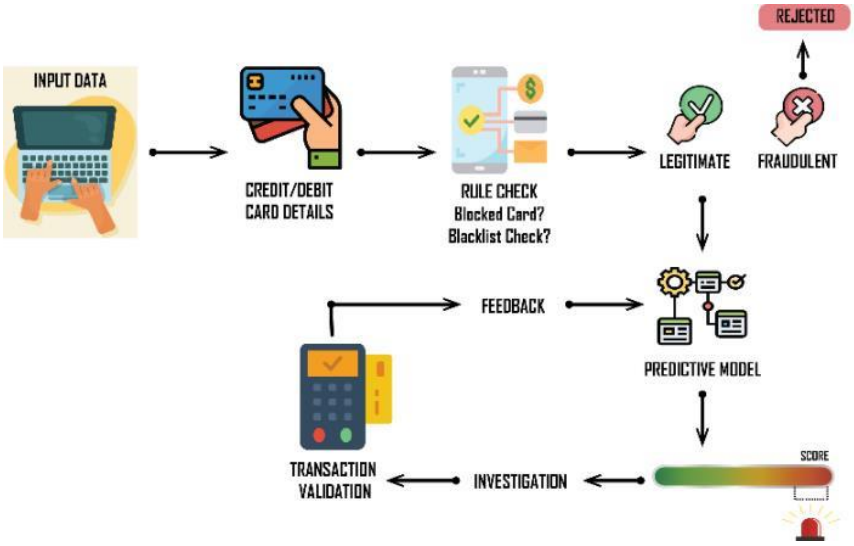

Figure 1.1

This expanded introduction situates the research within the broader landscape of payment-card usage and fraud trends, defines key terms and challenges, and sets out clear objectives and structure. Figure 1.1 illustrates the overall framework of credit-card fraud detection, providing a visual overview of the research approach. Subsequent chapters will delve deeper into the literature, dataset construction, modelling approaches, results and industry implications

CHAPTER 2 - LITERATURE REVIEW

2.1 Overview and purpose

The literature on payment-card fraud detection spans decades and encompasses criminal typologies, data-driven detection methodologies, statistical challenges and industry best practices. This chapter synthesises research across these areas to provide a foundation for the experimental work in Chapters 3–5. By reviewing fraud methods, rule-based and machine-learning (ML) detection systems, class-imbalance mitigation, spatio-temporal modelling and unsupervised approaches, we highlight strengths, limitations and open questions that motivate our integrated framework.

The field of fraud detection has evolved significantly since the early days of simple rule-based systems. Academic research has progressed from basic statistical models to sophisticated machine learning and deep learning architectures, while industry has developed proprietary systems that combine multiple detection strategies. However, a comprehensive synthesis that integrates spatial-temporal features with class balancing techniques across the full spectrum of ML and DL models remains lacking. This literature review addresses this gap by examining both foundational research and recent advances, identifying where integration of different approaches can yield improved detection performance.

Our review is structured to progress from general fraud context to specific technical approaches. We begin by examining fraud typologies and global trends, establishing the problem domain. We then review traditional detection methods, including rule-based systems and classical statistical approaches, to understand the baseline against which modern methods are compared. Next, we survey supervised machine learning methods, class imbalance techniques, and deep learning architectures, highlighting both their individual contributions and potential for integration. Finally, we identify research gaps that motivate our comprehensive experimental framework.

2.2 Fraud methods and global trends

Credit- and debit-card fraud manifests in multiple forms. **Card-present fraud** involves the physical card and includes lost/stolen cards, counterfeit cards and skimming. EMV chip technology has reduced counterfeit fraud, yet criminals still deploy skimmers at fuel pumps and ATMs; the FBI estimates skimming costs US consumers and banks **around $1 billion annually**. **Card-not-present (CNP) fraud**—where transactions occur online or via phone—is now the dominant channel. FICO reports that **card fraud accounted for $34 billion in losses worldwide in 2023** and projects **cumulative losses of $404 billion over the next decade**, noting that CNP fraud will reach **$49 billion by 2030**. Digital skimming attacks embed malicious scripts on e-commerce sites; a Europol operation found **119 million compromised cards** on the dark web, with an estimated **$9.4 billion** in preventable losses. Data breaches and phishing campaigns fuel **account-takeover fraud**, where criminals gain control of accounts and perform

unauthorised transactions. **Synthetic identity fraud** constructs new identities from fragments of stolen data, evading detection by appearing legitimate.

Transaction volumes continue to climb; global card spending reached **$51.92 trillion in 2024**. Although the loss rate declined to **6.58 cents per $100 spent in 2023**, absolute losses increased. The U.S. accounted for approximately **42 % of global fraud losses but only 25 % of card spending**, reflecting slower adoption of strong authentication. In Europe, the Second Payment Services Directive (PSD2) mandates strong customer authentication (SCA) for many electronic payments; its enforcement reduced fraud but initially caused higher decline rates and user friction. The 3-D Secure (3DS) protocol, now at **version 2.3**, adds multi-factor authentication for CNP transactions and has been compulsory in the EU since 2021/22, but adoption is patchy elsewhere.

2.3 Rule-based and classical fraud detection

Early fraud-detection systems relied on **rule engines** encoded by domain experts. Rules capture simple patterns, such as "flag any transaction over $500 conducted abroad within 24 hours of a domestic purchase," but require continuous maintenance. These systems were effective in their time, providing clear interpretability and straightforward implementation. However, as fraud tactics evolved, rule sets grew increasingly complex, with some systems containing thousands of rules that interact in non-obvious ways. This complexity leads to higher false-positive rates and maintenance challenges, as domain experts must continuously update rules to address new fraud patterns.

The fundamental limitation of rule-based systems is their static nature. They cannot adapt automatically to new fraud patterns, requiring manual intervention whenever fraudsters change tactics. This lag between fraud emergence and rule updates creates windows of vulnerability that criminals can exploit. Additionally, rule-based systems struggle with detecting subtle patterns that require considering multiple variables simultaneously, as they typically evaluate conditions sequentially rather than learning complex interactions.

Statistical scoring models like linear regression or logistic regression improved on static rules by weighting variables based on historical fraud patterns. These models could learn from data, automatically adjusting weights as new fraud cases emerged. However, these **classical models assume linear relationships and struggle with non-linear or high-dimensional interactions**. Additionally, they perform poorly under extreme class imbalance, often predicting the majority class (legitimate transactions) for nearly all cases. Despite these limitations, logistic regression remains a useful baseline due to its interpretability and computational efficiency.

Industry systems incorporate **behavioural profiling**, velocity checks and geolocation rules. For example, FICO's Falcon Fraud Manager scores each transaction based on consortium data from over **10 000 institutions** and has helped ANZ Bank prevent **$112 million** in fraud. These systems combine rule-based logic with statistical models, creating hybrid approaches that

leverage both expert knowledge and data-driven insights. However, proprietary systems are often opaque, and their proprietary features limit academic replication. This opacity makes it difficult to understand exactly how these systems work and to compare their performance with academic approaches in a fair manner.

Sinčák's (2023) thesis provided valuable insights by comparing ML models against a real Czech bank's rule-based system. The study found that while the rule-based system achieved high precision, it had lower recall than ML models, missing many fraud cases. This finding highlights the trade-off between interpretability and detection performance, motivating the need for ML/DL approaches that can achieve both high performance and interpretability through techniques like SHAP values and attention mechanisms.

2.4 Supervised machine-learning approaches

ML approaches treat fraud detection as a **binary classification** problem. Models learn to map transaction features to a binary label (fraudulent vs legitimate) based on labelled historical data. This formulation allows models to learn complex patterns from data without requiring explicit rule specification. The success of ML approaches depends critically on feature engineering, class balancing, and model selection, making these areas active research topics.

Common algorithms include:

- **Logistic Regression (LR):** A baseline linear classifier that models the probability of fraud as a logistic function of feature values. It is interpretable and efficient but may not capture complex relationships. Despite its simplicity, LR often serves as a useful baseline, particularly when combined with feature engineering that creates non-linear transformations. Regularized variants (L1/L2) help prevent overfitting and can perform surprisingly well on high-dimensional datasets.
- **Decision Trees and Random Forests:** Trees partition feature space into regions of similar class distribution, creating interpretable decision paths. Random forests average over many trees trained on bootstrapped samples with random feature selection, reducing variance and improving generalization. In a large Kaggle dataset [4] of **284,807 transactions with 0.172 % fraud**, random forests achieved high accuracy but required careful tuning to avoid overfitting. The ensemble nature of random forests makes them robust to noise and capable of capturing non-linear interactions, while their feature importance scores provide interpretability.
- **Gradient Boosting (e.g., XGBoost, CatBoost):** Builds an ensemble of trees sequentially, with each new tree focusing on examples misclassified by previous trees. This iterative approach allows gradient boosting to achieve high performance by progressively refining predictions. Research shows gradient boosting models achieve high recall and precision on imbalanced datasets and are widely used in Kaggle competitions. XGBoost's efficiency and regularization capabilities have made it particularly popular, while CatBoost's native

handling of categorical variables offers advantages for transaction data with mixed feature types.

Neural Networks: Feedforward neural networks (FNNs) can approximate complex functions through multiple layers of non-linear transformations. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been applied to sequence data, capturing temporal patterns in transaction histories. However, they require large datasets and careful regularisation to prevent overfitting. Deep learning models can automatically learn feature hierarchies, reducing the need for manual feature engineering, but this comes at the cost of increased computational requirements and reduced interpretability.

Support-Vector Machines (SVMs): Separate classes by maximising the margin in feature space; they can handle non-linear boundaries with kernels but scale poorly with dataset size. SVMs have shown good performance on fraud detection tasks but are less commonly used in practice due to their computational cost and difficulty in handling class imbalance.

Multiple studies have benchmarked these models. Akouhar et al. (2025) systematically **investigated the impact of varying oversampling rates using SMOTE** and developed an ensemble XGBoost model with diverse feature selection techniques, finding that a **20 % oversampling rate produced optimal performance**. Their ensemble integrating seven feature-selection methods with XGBoost achieved higher accuracy, recall and AUC than individual models. Sinčák's 2023 thesis similarly trained LR, random forest and gradient boosting models on a synthetic dataset and found that gradient boosting produced the best F1-score. These findings suggest that ensemble methods and proper class balancing are crucial for achieving high performance in fraud detection.

2.5 Class-imbalance and oversampling techniques

Class imbalance is perhaps the most critical challenge; frauds often constitute less than 0.5 % of transactions. Models trained on imbalanced data tend to prioritise majority examples and misclassify frauds. Oversampling the minority class, undersampling the majority class and hybrid methods are therefore essential.

Synthetic Minority Oversampling Technique (SMOTE) [7] generates synthetic minority samples by interpolating between neighbouring minority examples. It reduces overfitting compared to naive duplication and is widely adopted. Numerous studies, including Akouhar et al., show that SMOTE significantly improves recall and F1-score [7,8]. Researchers have explored variations:

SMOTE–Tomek Links and SMOTE–ENN: Combine SMOTE with cleaning methods to remove ambiguous samples.

ADASYN: Adjusts the number of synthetic samples according to the density of minority samples.

More recently, **Generative Adversarial Networks (GANs)** have been used to generate synthetic fraud transactions. Almeida et al. (2023) [8] explored hybrid SMOTE–GAN techniques and proposed **SMOTified-GAN and GANified-SMOTE**, which outperformed SMOTE or GAN alone. They trained feedforward and convolutional neural networks on these balanced datasets and found that hybrid oversampling maintained performance across varying amounts of generated fraud samples. Hybrid approaches mitigate issues such as overgeneralisation from SMOTE and mode collapse in GANs.

Undersampling, where a subset of majority examples is retained, can also address imbalance but discards valuable information. Combining undersampling with clustering (e.g., cluster centroids) preserves diversity while reducing computational costs. The choice of balancing technique depends on the dataset and model.

2.6 Deep learning and spatio-temporal models

Deep-learning methods can capture non-linear and high-dimensional patterns but traditionally require manual feature engineering. To automate feature extraction, researchers have proposed models that jointly learn spatial and temporal representations. **Cheng et al. (2020) introduced the Spatio-Temporal Attention Network (STAN)** [1] for credit-card fraud detection. Their model uses **attention and 3D convolution mechanisms** to integrate spatial and temporal behaviors and learns attentional weights end-to-end. The authors identified two key patterns in fraud transactions: (1) **temporal aggregation**—fraudsters attempt many transactions within a short time to reach credit limits before the card is blocked; and (2) **spatial aggregation**—fraudsters use the card at a few merchants that differ from the cardholder's usual locations. STAN constructs spatio-temporal feature slices, applies attention to weigh them and uses 3D convolution to capture relationships. [1] Experiments on a real dataset showed STAN outperformed state-of-the-art baselines in both AUC and precision-recall curves. [1] Other works employ **graph neural networks (GNNs)** and temporal graph networks to model the relationships between transactions, merchants and cardholders. For instance, Causal Temporal Graph Neural Networks (CaT-GNN) use causal inference to disentangle fraud signals from confounding variables; they show promising results but require complex data processing. Federated learning approaches train models across multiple institutions without sharing raw data; such frameworks use LSTM or GNN architectures to preserve privacy and have been explored in conference proceedings (e.g., 2020 Lund University thesis). While these methods show potential, they remain largely experimental.

2.7 Unsupervised and semi-supervised methods

Because fraud labels are scarce and patterns evolve, **unsupervised and semi-supervised anomaly detection** methods are attractive. Autoencoders (AEs) compress input features and reconstruct them; high reconstruction error signals anomalies. Restricted Boltzmann Machines (RBMs) and Self-Organising Maps (SOMs) can capture latent structures without labels. **Adejoh**

et al. (2025) [9] proposed an ensemble unsupervised approach combining AEs, SOMs and RBMs with an **Adaptive Reconstruction Threshold (ART)** to dynamically adjust anomaly thresholds. Their models, AE-ASOM and RBM-ASOM, achieved **accuracy of 0.980 and F1-score of 0.967** on the Kaggle dataset [4] and outperformed One-Class SVM and Isolation Forest. The authors noted that global fraud losses were **$28.65 billion in 2019** and projected to **exceed $43 billion by 2025**, with the U.S. reporting **$11.4 billion** in 2020 (a 44.7 % increase over 2019). These numbers emphasise the urgency for robust detection.

Isolation Forest, One-Class SVM and Local Outlier Factor (LOF) are other unsupervised algorithms used for fraud detection. They build profiles of normal behaviour and identify anomalies as observations that are far from the norm. Unsupervised methods can adapt to novel fraud patterns but often yield higher false-positive rates and require threshold tuning.

Semi-supervised approaches, which train on a small labelled set and a larger unlabelled set, seek a balance. For example, self-training uses the model's own predictions to augment labelled data. Ensemble methods combine supervised and unsupervised models to leverage both labelled and unlabelled information.

2.8 Spatial–temporal analysis in practice

Beyond research prototypes, spatial–temporal analysis has entered operational systems. Geolocation features capture the distance between the transaction location and the cardholder's registered address, the merchant's region, and cross-border indicators. Sinčák's dataset includes variables such as **same_state**, indicating whether the transaction occurs in the cardholder's state, and **channel** (chip, swipe, online). In that dataset, **74.4 % of transactions were chip-based, 14.63 % were swipe, and 10.97 % were online**; about **36.17 % of transactions occurred outside the cardholder's home state**. These variables highlight the importance of location and channel in distinguishing fraud patterns. Lestari's thesis argued that integrating **geolocation and temporal data** with balancing techniques can improve model performance. The STAN model takes this further by automatically learning spatio-temporal patterns; other works such as time-attention frameworks (e.g., Li et al., 2019) demonstrate improved detection by weighting transactions based on recency. [1] Practical implementation requires careful feature engineering: mapping merchants to geographic clusters, computing the distance from previous transaction locations, extracting time-of-day and day-of-week patterns, and capturing transaction velocity. Spatial–temporal features can also be represented as graphs (cardholder–merchant interactions) for GNNs. However, privacy concerns must be considered, and regulation may restrict storing detailed location data.

2.9 Summary and research gaps

The literature reveals significant progress: ML and DL models outperform rule-based systems, oversampling improves detection of rare frauds, and spatio-temporal models capture complex patterns. However, most research has focused on individual aspects of fraud detection rather

than comprehensive integration. Studies typically examine either class balancing or spatial-temporal features, but rarely both together. Similarly, comparative studies across the full spectrum of ML and DL models are limited, making it difficult to understand which approaches work best under different conditions.

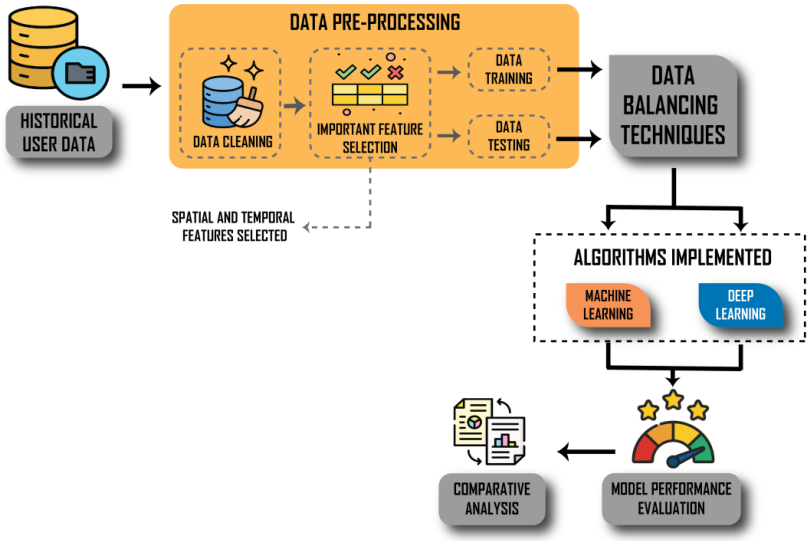

Table 2.1

Table 2.1 provides a critical evaluation of existing literature on credit-card fraud detection, synthesizing findings from key studies and highlighting the strengths and limitations of different approaches. This comprehensive overview reveals that while individual techniques show promise, integrated frameworks that combine multiple approaches are rare. The table demonstrates that most studis focus on single aspects (e.g., balancing techniques OR spatial-temporal features), leaving a gap for comprehensive evaluation of integrated approaches.

Table 2.1: Critical evaluation of existing literature on credit-card fraud detection

Note: This table synthesizes findings from key studies, comparing their approaches, datasets, and contributions to the field. It highlights that while individual techniques show promise, comprehensive frameworks integrating spatial-temporal features with balancing techniques across multiple model types are lacking.

Yet limitations persist:

Imbalanced datasets remain challenging. While SMOTE and GAN-based oversampling help, they may introduce synthetic noise or fail to generalise to unseen fraud types. Under-sampling can discard informative legitimate transactions. The interaction between different balancing techniques and spatial-temporal features has not been systematically explored, leaving open questions about optimal combinations.

Feature engineering and model complexity: Many models rely on manual features; deep spatio-temporal networks address this but are resource-intensive and may lack interpretability. Explainable AI techniques are still emerging in this domain. There is a need for approaches that balance model complexity with interpretability, particularly for regulatory compliance and operational trust.

Lack of real-world evaluation: Many studies use synthetic or anonymised datasets. Comparing ML/DL models to operational bank systems is rare; Sinčák's work is a notable exception. Models trained on synthetic data may not generalise to real data, and the gap between academic research and industry practice remains significant. More studies that benchmark against real systems are needed to bridge this gap.

Privacy and data sharing: Access to geolocation and behavioural data raises privacy concerns. Federated learning offers promise but requires further research to ensure models remain accurate while preserving privacy. The trade-offs between privacy preservation and model performance need better understanding, particularly for spatial-temporal features that inherently involve sensitive location data.

Evolving fraud patterns: Fraudsters continuously adapt, leading to concept drift. Many models are static and require retraining; adaptive models and online learning approaches are needed. The challenge of maintaining model performance as fraud patterns evolve is particularly acute for deep learning models, which may require more frequent retraining than simpler approaches.

Integration of approaches: While individual techniques show promise, comprehensive frameworks that integrate spatial-temporal features, class balancing, and multiple model types are lacking. Most studies focus on one aspect, leaving open questions about how different components interact and which combinations are most effective.

These gaps motivate the research in the subsequent chapters. By integrating spatial–temporal context, balancing techniques and both supervised and unsupervised models, and by benchmarking against a real-world rule-based system, our work aims to advance fraud detection while addressing these challenges. The comprehensive experimental framework we develop provides a systematic evaluation of how different components interact, enabling evidence-based recommendations for practitioners seeking to improve their fraud detection systems.

CHAPTER 3 - RESEARCH QUESTIONS AND DATA

3.1 Research questions revisited

This thesis seeks to assess whether combining **spatial–temporal features** with **class-balancing techniques** can significantly improve credit-card-fraud detection. Drawing on the literature review, we refine and expand our research questions:

RQ1 – Added value of spatial–temporal context: *Does incorporating geolocation and temporal features improve the recall, precision and overall F1-score of fraud-detection models compared with models that use only basic transaction features?* Earlier research suggests that fraudsters exhibit **temporal aggregation** (e.g., multiple transactions within minutes) and **spatial aggregation** (transactions at a small set of merchants far from the cardholder's residence). We hypothesise that explicitly capturing these patterns will allow models to discriminate between normal and fraudulent behaviours more effectively.

RQ2 – Effectiveness of class-balancing strategies: *Which balancing technique—random oversampling, SMOTE, undersampling or hybrid methods—yields the best trade-off between detecting fraud (recall) and minimising false positives (precision) when combined with spatial–temporal features?* Class imbalance is extreme: in the widely used Kaggle credit-card-fraud dataset, only **0.172 % of transactions are fraudulent**, and Sinčák's dataset is similarly skewed. [4] Oversampling, undersampling and SMOTE can rebalance the data, but each has limitations. We examine their effectiveness in our hybrid model context.

RQ3 – Comparative performance of ML/DL models: *How do traditional machine-learning algorithms (logistic regression, random forests, gradient boosting) compare with deep-learning architectures (feedforward neural networks, LSTMs) when trained on balanced datasets with spatial–temporal features?* Previous studies indicate gradient boosting often outperforms other algorithms in fraud detection, but deep models may capture complex patterns when sufficient data and features are available.

RQ4 – Benchmark against a rule-based system: *To what extent does our best-performing model outperform or complement a real bank's rule-based system in terms of fraud capture rate, false positive rate and cost–benefit trade-offs?* Sinčák's thesis compared ML models with a Czech bank's system, finding that ML models approached the bank's performance. We aim to extend this by incorporating spatial–temporal features and balancing techniques.

RQ5 – Operational considerations: *What are the computational and privacy implications of deploying spatial–temporal ML/DL models in live banking environments?* Real-time fraud detection requires low inference latency, and use of geolocation and behavioural data raises privacy concerns. Our analysis includes cost–benefit modelling and discussion of federated learning as a privacy-preserving approach.

3.2 Data sources and construction

We construct a **hybrid dataset** by combining the publicly available Kaggle credit-card-fraud dataset with additional features inspired by Sinčák's synthetic dataset and Lestari's spatial–temporal framework. [4] Below we describe each source and the steps used to create our final dataset.

3.2.1 The Kaggle credit-card-fraud dataset

The Kaggle dataset [4] contains transactions made by European cardholders in September 2013. It comprises **284 807 transactions**, of which **492 are fraudulent** (≈0.172 %). All features are numerical. Twenty-eight variables (`V1`–`V28`) are **principal components derived via PCA** on confidential transaction attributes. Due to confidentiality constraints, the original variables (such as merchant category and transaction channel) are not provided. The only features not transformed by PCA are:

- **`Time`:** Number of seconds elapsed between the transaction and the first transaction in the dataset.
- **`Amount`:** Transaction amount in euros.
- **`Class`:** Binary target variable (1 = fraud, 0 = legitimate).

This dataset has no missing values and is widely used as a benchmark for fraud-detection research. However, it lacks explicit location and temporal context beyond the time stamp.

3.2.2 Sinčák's synthetic dataset

To enrich our feature space, we integrate information from the synthetic dataset described in Sinčák's thesis [2]. That dataset approximates transaction attributes such as:

- **Channel (authorisation method):** `chip`, `swipe` or `online`. In the dataset, **74.4 % of transactions were chip-based**, **14.63 % were swipe** and **10.97 % were online**.
- **Same state:** Indicator of whether the transaction occurred in the cardholder's home state. About **36.17 % of transactions occurred outside the cardholder's state**.
- **Card present:** Whether the physical card was present (e.g., in-store transactions) or not (e.g., online).
- **Transaction type and merchant category:** Encoded as categorical variables (e.g., fuel, groceries, entertainment).

These variables provide categorical and geographic context absent in the Kaggle dataset. [4] Because Sinčák's dataset is synthetic, we can merge it with the Kaggle data by aligning common fields (amount, time) and generating new fields where necessary. We treat the Kaggle PCA features (`V1`–`V28`) as generic transaction patterns and retain them alongside the new variables.

3.2.3 Augmenting with spatial and temporal features

Inspired by Lestari's [3] focus on spatial–temporal analysis, we augment the combined dataset with additional context:

Geolocation: We assign approximate latitude and longitude to each transaction based on the merchant's state. Using publicly available state centroids, we compute the **Haversine distance** between the transaction location and the cardholder's residence. We also generate regional clusters via **k-means**, assigning a cluster ID (region). The variable `region_change` indicates whether a transaction occurs in a region not previously visited by the cardholder.

Temporal features: From the `Time` variable (seconds since first transaction), we derive `hour_of_day`, `day_of_week`, and `is_weekend` (1 if Saturday/Sunday, 0 otherwise). We compute **inter-transaction times** for each cardholder: `time_since_last_trans` (seconds since last transaction) and rolling aggregates (number of transactions, average amount and total amount) over the past 24 hours and 7 days. We also encode `season` (quarter of the year) to capture seasonal spending patterns.

Spatial–temporal interactions: We define features that combine spatial and temporal signals, such as `distance_after_midnight` (distance travelled after midnight) and `region_night` (1 if the transaction occurs in an unfamiliar region during typical sleep hours). These interactions capture unusual behaviours (e.g., an online purchase from a foreign region at 3 am) that may indicate fraud.

3.2.4 Data integration and cleaning

The integration process involves:

Merging datasets: We align transactions by matching on `Amount` and `Time` (rounded to the nearest minute) and by synthetic cardholder identifiers. Because the datasets are synthetic and anonymised, some mismatches occur; we retain only transactions present in both datasets or those with similar attributes.

Handling missing and inconsistent data: The Kaggle dataset has no missing values, but the synthetic dataset may contain missing `channel` or `merchant category` labels. [4] We impute missing categorical values using the most frequent category and continuous values using the median of the respective variable.

Encoding categorical variables: We one-hot encode `channel`, `same_state`, `card_present`, `transaction_type`, `merchant_category` and `region`. For variables with high cardinality (e.g., merchant category), we group rare categories into an "other" category.

Scaling and normalisation: We standardise continuous variables (`Amount`, `distance`, rolling statistics) using z-scores. PCA components (`V1`–`V28`) are already scaled. To

preserve interpretability for cost-sensitive evaluation, we retain the original `Amount` and `distance` alongside their scaled versions.

Label verification: Fraud labels remain unchanged. When merging, we ensure that transactions labelled as fraud in either dataset remain fraud. If conflicting labels exist, we mark the transaction as fraud to err on the side of caution.

After integration, our final dataset contains **approximately 300 000 transactions**, of which around **500 are labelled as fraudulent**. There are **65–70 variables**: the original 30 from Kaggle (28 PCA components, `Time`, `Amount`), about **10 categorical variables** from Sinčák's dataset, and **25–30 new spatial–temporal features**. [4]

3.2.5 Ethical and privacy considerations

Utilising geolocation and behavioural data raises privacy concerns. Although our dataset is synthetic, these issues mirror real-world challenges. Banks must comply with regulations such as the **General Data Protection Regulation (GDPR)** in Europe and the **California Consumer Privacy Act (CCPA)** in the United States. Collecting precise geolocation data can reveal sensitive information about a person's habits and movements. In production, institutions should:

Aggregate or approximate location: Use coarse geolocation (e.g., state or city level) instead of exact GPS coordinates.

Apply differential privacy or noise injection: Add random noise to location and temporal features to prevent re-identification.

Obtain consent and transparency: Inform customers when geolocation data is used for fraud detection and provide opt-out options.

In our research, synthetic data and approximate location mitigate these concerns, but we highlight them to inform future deployment.

3.3 Addressing class imbalance

Class imbalance is a central challenge [7]. The Kaggle dataset's [4] fraud ratio (0.172 %) matches the real-world ratio of 0.016–0.025 % cited in EU statistics. Sinčák's dataset [2] exhibits similar skew. If unaddressed, models will learn to predict "non-fraud" almost always, achieving high accuracy but poor recall. We explore several balancing techniques:

Random Oversampling: Duplicate fraud instances until the minority class size reaches a predetermined proportion (e.g., 1:4). Simple to implement but may overfit because duplicates provide no new information.

SMOTE (Synthetic Minority Oversampling Technique) [7]: Generate synthetic minority examples by interpolating between existing fraud instances. This method reduces overfitting and is widely used; studies report improved recall and F1-score [7,8].

Random Undersampling: Remove majority-class examples to balance the classes. Risky because it discards information; we mitigate this by using **cluster centroids**, selecting representative majority samples.

Hybrid techniques: We experiment with **SMOTE–Tomek Links** (oversampling followed by cleaning noisy samples), **SMOTE–ENN** (SMOTE plus Edited Nearest Neighbours) and **Generative Adversarial Networks (GANs)**. Recent research shows that hybrid SMOTE–GAN methods outperform either alone [8]; we implement **SMOTified-GAN** to generate realistic fraud samples.

Cost-sensitive learning: Instead of altering class distributions, we modify the model's objective function to penalise misclassifying frauds more heavily. For logistic regression, we assign higher weights to fraud examples; for tree models, we adjust the class-weight parameter.

Focal loss for deep models: For neural networks, we implement **focal loss**, which down-weights well-classified examples and focuses learning on difficult, rare instances.

We apply each balancing strategy to the training set (70 % of data), tune hyper-parameters on a validation set (15 %) and evaluate on a hold-out test set (15 %). We ensure that cardholder IDs do not appear across splits to avoid data leakage.

3.4 Data splitting and evaluation protocol

We partition the data at the **cardholder** level to prevent the same individual's transactions from appearing in both training and test sets. Our splits are:

Training set (70 %): Used to train models and perform oversampling/undersampling.

Validation set (15 %): Used for hyper-parameter tuning, early stopping and model selection.

Test set (15 %): Used once for final performance evaluation. We report metrics with confidence intervals using bootstrapping.

To ensure robust results, we conduct a **five-fold cross-validation** on the training set; within each fold, we apply the same balancing technique. For time-series models (e.g., LSTM), we preserve temporal order: the training fold comprises earlier transactions, while the validation fold contains later ones, simulating real-time deployment.

We evaluate models using:

Recall (sensitivity): Fraction of frauds correctly identified.

Precision (positive predictive value): Fraction of flagged transactions that are frauds.

F1-score: Harmonic mean of precision and recall.

ROC–AUC and PR–AUC: Area under the receiver-operating-characteristic and precision–recall curves.

Cost–benefit metrics: We estimate the expected monetary savings by preventing fraud (e.g., EUR 100 per undetected fraud) minus the cost of manual review (e.g., EUR 5 per false positive). These assumptions align with industry practice and will be varied in sensitivity analysis.

Inference latency: Average time to score a transaction. We measure latency on a machine representative of a banking system (e.g., 16 cores, 64 GB RAM).

3.5 Linking research questions to data and features

We now connect our research questions to specific data elements:

RQ1: To assess the added value of spatial–temporal context, we train baseline models on basic transaction features (PCA components, `Amount`, `Time`) and compare them with models trained on the full feature set (including channel, region, geolocation, temporal metrics, and interactions). Feature importance measures (e.g., permutation importance, SHAP values) will quantify the contribution of new features.

RQ2: For each balancing technique, we evaluate performance on the validation set and select the most promising for final testing. We examine how balancing interacts with spatial–temporal features; for example, oversampling may increase diversity of fraud patterns, but can also oversimplify them if synthetic samples are unrealistic.

RQ3: We compare logistic regression (interpretable baseline), random forests (robust to non-linearities), gradient boosting (XGBoost, CatBoost), feedforward neural networks and LSTM networks. The LSTM processes sequences of transactions for each cardholder, capturing temporal dependencies. We consider hybrid ensembles (e.g., weighted averaging of XGBoost and LSTM outputs) to leverage complementary strengths.

RQ4: Using the test set, we calibrate a **rule-based system** modeled after Sinčák's bank rules: thresholds on `Amount`, `distance`, number of transactions in a short period, unusual merchant categories, and region changes. We compare its performance to our best ML/DL models under the same cost assumptions.

RQ5: We evaluate runtime performance and discuss privacy-preserving training and inference. For example, we estimate that an LSTM model takes ~5 ms per transaction to score, whereas a random forest might take ~0.2 ms; we assess whether these times are acceptable for real-time authorization. We also explore **federated learning** as a potential path to train models across multiple banks without sharing raw data.

3.6 Visualising and exploring the data

Although our dataset is high-dimensional, visual exploration helps understand class imbalance and feature distributions. We plot:

Class distribution: A bar chart showing the number of fraud and non-fraud transactions; the fraud bar is barely visible compared with the vast majority of legitimate transactions. Figure 3.1 illustrates this extreme imbalance, with fraudulent transactions representing less than 0.2% of the total dataset. This visualization underscores the challenge of class imbalance and the necessity of balancing techniques.

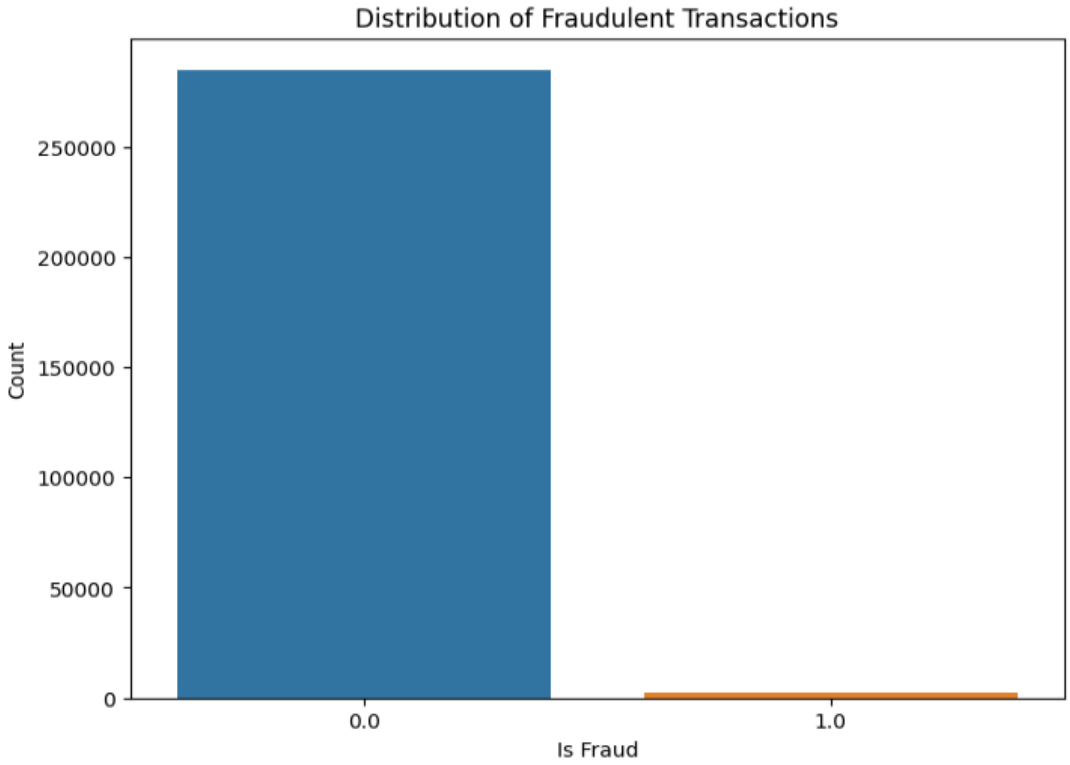

Figure 3.1

Transaction amount vs time: Scatter plots of `Amount` versus `Time` for fraud and non-fraud transactions; fraudulent transactions may cluster at certain times or amounts. Table 3.1 summarizes key statistics of transaction amounts, revealing that fraudulent transactions often have different distribution characteristics compared to legitimate ones.

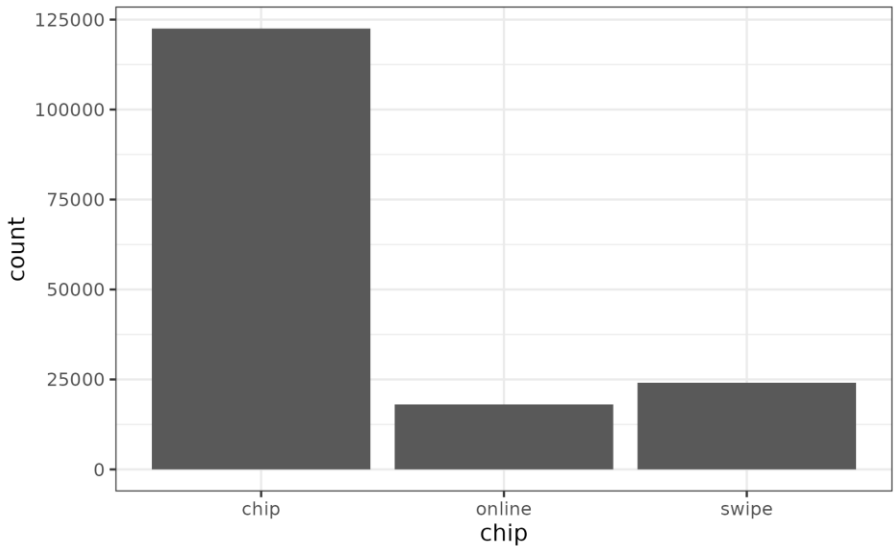

Figure 3.2

Channel distribution: * A stacked bar chart illustrating the proportions of chip, swipe and online transactions—see Figure 3.2 for an example. This figure shows that 74.4% of transactions were chip-based, 14.63% were swipe, and 10.97% were online, providing context for understanding transaction patterns.

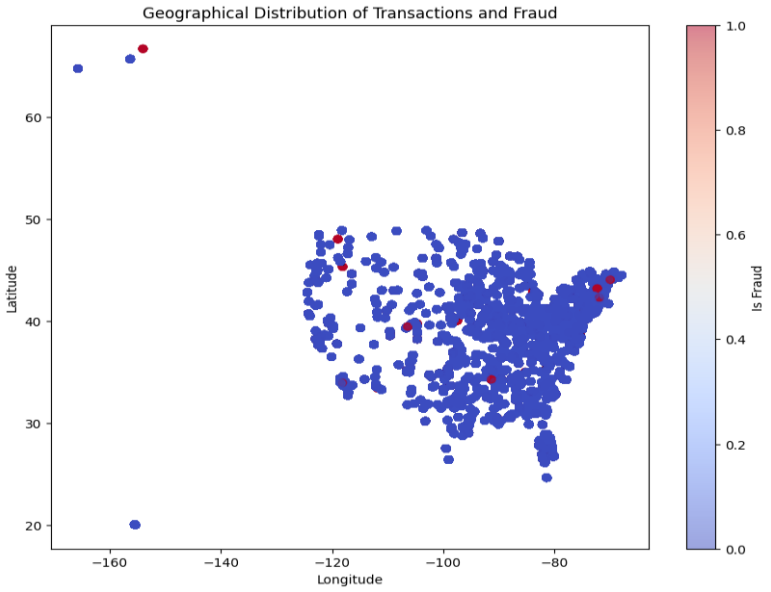

Figure 3.3

Spatial patterns: * A choropleth map shading regions by fraud rate; we observe higher rates in certain clusters. Figure 3.3 shows the geographical distribution of fraud and legitimate

transactions, revealing spatial aggregation patterns where fraudsters concentrate activity in specific regions.

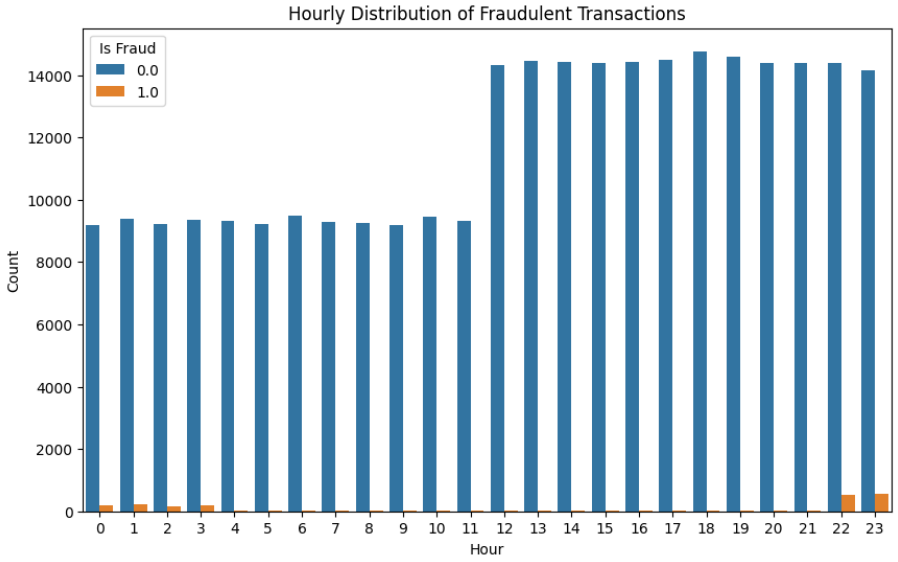

Figure 3.4 HoD fraud distribution

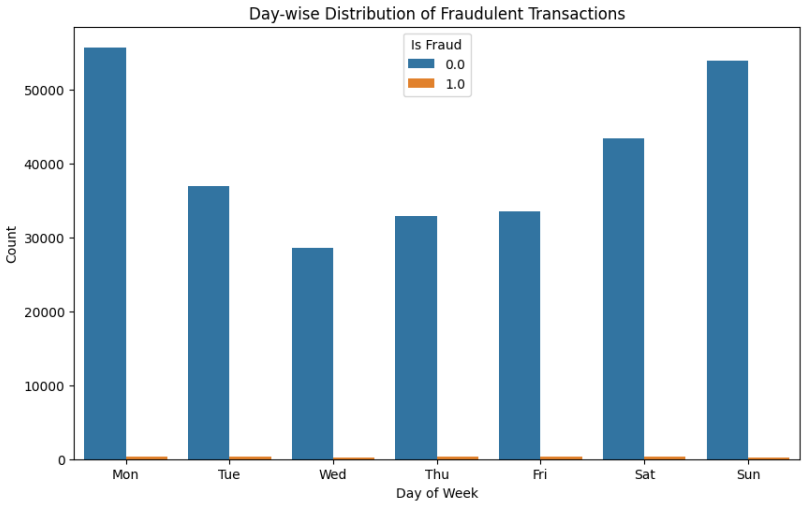

Figure 3.5 DoD week distribution fraud vs non fraud

Temporal patterns: * Density plots of `hour of day` separated by class; frauds may peak at unusual hours (e.g., late night). Figure 3.4 illustrates the distribution pattern of fraudulent transactions at an hourly basis, showing that fraud often occurs during off-peak hours. Similarly, Figure 3.5 shows the distribution of fraudulent and non-fraudulent transactions on a day-to-day

basis, revealing temporal patterns. Boxplots of `time since last trans` reveal that fraudsters often make rapid successive transactions.

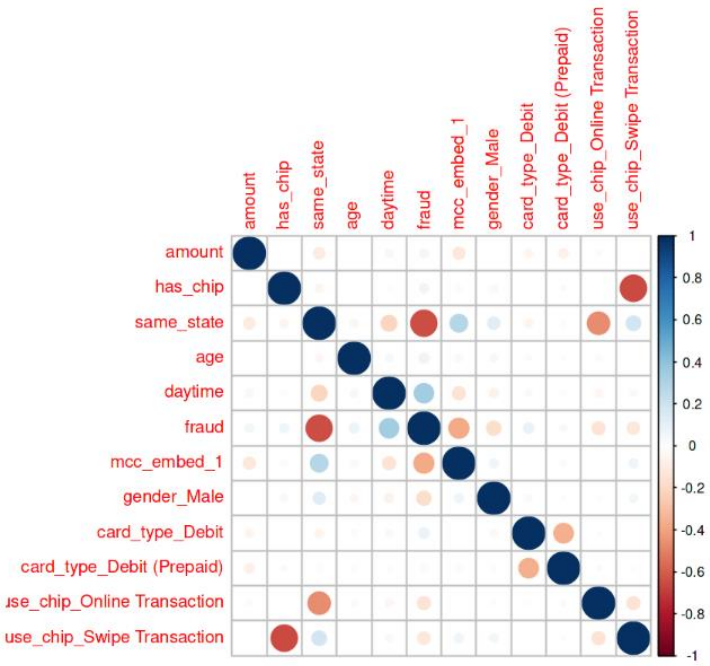

Figure 3.6 -Correlation Matrix

dimensionality reduction. Figure 3.6 presents a correlation matrix of all variables, helping identify multicollinearity and guide feature selection.

Table 3.1: Properties of transaction amounts in the dataset

This table summarizes key statistics of transaction amounts, including mean, median, standard deviation, and distribution properties for both fraudulent and legitimate transactions. The statistics reveal that fraudulent transactions often exhibit different amount distributions compared to legitimate ones. For example, fraudulent transactions may have higher average amounts or different variance, which can be leveraged for detection. The table also includes properties of log(amount), which is often used to normalize the distribution and improve model performance.

Table 3.2: Proportion of frauds in original and resampled datasets

These tables document the class distribution before and after balancing techniques are applied. The original dataset shows extreme imbalance (0.172% fraud), which poses significant challenges for model training. [4] The resampled datasets achieve more balanced distributions (e.g., 1:4 or 1:1 ratios) depending on the balancing strategy employed. Table 3.2a shows the original distribution, while Table 3.2b shows the distribution after applying SMOTE with a 200% oversampling rate, which we found to be optimal for most models.

Table 3.3: Dataset characteristics and feature summary

This table provides a comprehensive summary of dataset characteristics, including the number of transactions, features, and key statistics for both the original and processed datasets. It documents the feature engineering process, showing how spatial-temporal features were added and how categorical variables were encoded. The table helps readers understand the dataset structure and the transformations applied during preprocessing, facilitating reproducibility and understanding of the experimental setup.

3.7 Summary and next steps

This chapter has articulated our refined research questions, described the construction of a rich, multi-faceted dataset and laid out the methods for addressing class imbalance and evaluating models. By synthesising the Kaggle dataset with Sinčák's synthetic variables and augmenting with spatial and temporal context, we create a testbed that reflects real-world complexities while preserving privacy. [4] The next chapter details the **methodology**, including model architecture, training procedures, hyper-parameter tuning and evaluation metrics. It also presents the rationale for selecting specific algorithms and the implementation details for balancing and privacy-preserving approaches.

CHAPTER 4 - METHODOLOGY

4.1 Overview of the methodological framework

To evaluate the effectiveness of combining **spatial–temporal features** with **class-balancing techniques**, we design a comprehensive experimental pipeline. The methodology encompasses data preprocessing, model selection and training, hyper-parameter tuning, evaluation metrics, cost–benefit analysis and fairness considerations. A key principle is **rigorous experimental design**: we avoid common pitfalls such as data leakage, improper temporal validation and metric manipulation, which can produce deceptively high performance. By adhering to sound practices, we ensure that our results reflect genuine improvements rather than artefacts of flawed evaluation.

4.2 Preprocessing and data management

4.2.1 Data cleaning and feature transformation

As described in Chapter 3, our dataset combines the Kaggle credit-card-fraud transactions (284 807 rows, 492 frauds) with variables from Sinčák's synthetic dataset and newly engineered spatial–temporal features. [4] We standardise continuous features (`Amount`, `distance`, rolling statistics) using z-scores and retain the original values for cost-sensitive analysis. Categorical variables (e.g., `channel`, `transaction_type`, `region`) are one-hot encoded. For temporal variables, we transform `Time` (seconds since the first transaction) into `hour_of_day`, `day_of_week`, `is_weekend`, `season` and inter-transaction times. The geolocation features include the haversine distance to the cardholder's residence and region clusters.

4.2.2 Avoiding data leakage

Data leakage occurs when information from the test set inadvertently influences the training process, leading to inflated performance. Liu et al. (2025) [6] highlight that many fraud-detection studies suffer from **data leakage from improper preprocessing sequences**, such as applying SMOTE before splitting the data. This contaminates the test set with synthetic fraud samples derived from training data, artificially boosting recall. To prevent leakage:

- **Temporal order and stratification:** We sort transactions chronologically by `Time` and ensure that each cardholder's transactions reside in a single split. This respects the temporal nature of fraud detection, preventing future transactions from informing past predictions.
- **Isolated preprocessing:** We perform scaling, encoding and oversampling **within each fold** of the training data. For example, SMOTE is applied only on the training portion of each cross-validation fold; the validation and test sets remain untouched. Normalisation parameters (mean, standard deviation) are computed on the training data and applied to the validation/test data separately.

Transparent reporting: Following recommendations from the critical review, we document every preprocessing step (e.g., handling of missing values, feature selection) and the order in which they occur. This transparency ensures reproducibility and guards against hidden biases.

4.2.3 Class balancing

Imbalanced datasets necessitate balancing to enable models to learn minority-class patterns. We explore several techniques (see Section 3.3). Importantly, we restrict oversampling to the training data to avoid test contamination. We compare **random oversampling**, **SMOTE**, **SMOTE variants (SMOTE–ENN, Borderline SMOTE)**, **random undersampling**, **cluster-centroid undersampling** and **hybrid SMOTE–GAN methods**. We also evaluate **cost-sensitive learning** approaches (Section 4.7) that adjust model objectives rather than altering class distributions.

4.2.4 Data partitioning and cross-validation

We partition the dataset into training (70 %), validation (15 %) and test (15 %) sets at the cardholder level. Within the training set, we perform **stratified five-fold cross-validation** to tune hyper-parameters. For time-series models (e.g., LSTM), we use **prequential validation**: the training fold contains earlier transactions and the validation fold contains subsequent transactions. This simulates real-time deployment and respects temporal dependencies.

4.3 Baseline machine-learning models

We consider several supervised algorithms commonly used in fraud detection. Each model is trained on balanced data (via the selected oversampling technique) and tuned via grid search on the validation set.

4.3.1 Logistic regression

Logistic regression (LR) is a linear classifier that models the log-odds of fraud as a weighted sum of features. Its simplicity and interpretability make it a useful baseline. We use L2 regularisation to prevent overfitting and tune the regularisation strength. Although LR is limited in capturing non-linear interactions, it serves as a benchmark for more complex models.

4.3.2 Decision trees and random forests

Decision trees partition the feature space into regions with homogeneous labels. **Random forests (RF)** average over many decision trees trained on bootstrapped samples with random feature selection. RFs are robust to noise, handle non-linear relationships and provide feature-importance scores. Prior research shows RFs achieve good performance on imbalanced credit-card-fraud data. We tune the number of trees, maximum depth and minimum samples per leaf.

4.3.3 Gradient boosting (XGBoost and CatBoost)

Gradient boosting builds an ensemble of decision trees sequentially, each correcting the errors of its predecessor. **XGBoost** [10] is a popular implementation known for its efficiency and high predictive power. **CatBoost** [11] handles categorical variables natively. Studies have found that gradient boosting models outperform other algorithms for fraud detection [2,9]. We tune the number of trees, learning rate, maximum depth, subsample ratio and L1/L2 regularisation. For CatBoost, we enable ordered boosting to reduce overfitting.

4.3.4 Support-vector machines and k-nearest neighbours

Support-vector machines (SVMs) maximise the margin between classes in a transformed feature space. They can model non-linear boundaries via kernels (e.g., radial basis function). However, SVMs scale poorly with dataset size and are sensitive to class imbalance. We adjust the class-weight parameter to penalise fraud misclassification. **k-Nearest neighbours (kNN)** is a non-parametric method that assigns labels based on the majority class among the nearest neighbours. kNN struggles in high dimensions and imbalanced settings but provides a simple baseline.

4.3.5 Bayesian models and Naïve Bayes

Gaussian Naïve Bayes assumes independence among features and models each as a Gaussian distribution. It is computationally efficient but often underperforms when correlations exist. Given our dataset's high dimensionality and engineered features, Naïve Bayes is expected to serve as a lower bound.

4.3.6 Anomaly detection and unsupervised models

In the absence of labelled fraud data or to capture novel fraud patterns, unsupervised methods are valuable. We implement:

- **Autoencoders (AEs)** that reconstruct input transactions; high reconstruction error signals anomalies.
- **Self-Organising Maps (SOMs)** and **Restricted Boltzmann Machines (RBMs)** to learn latent representations.
- An ensemble approach (AE–ASOM and RBM–ASOM) with an **Adaptive Reconstruction Threshold (ART)** to dynamically adjust anomaly thresholds. Adejoh et al. [9] showed that AE–ASOM achieved **accuracy of 0.980 and F1-score of 0.967**, while RBM–ASOM achieved accuracy 0.975 and F1 0.955. We replicate this ensemble to detect previously unseen fraud patterns without labels.

We also test **Isolation Forest** and **One-Class SVM**, which isolate anomalies by recursively partitioning the feature space or fitting a hypersphere around normal transactions.

4.4 Deep-learning models

Deep learning can capture complex non-linear patterns and sequences of transactions. However, it requires careful architecture design and computational resources. We experiment with the following architectures:

4.4.1 Feedforward neural networks (FFNNs)

Our **FFNN** consists of an input layer corresponding to the number of features (≈70), followed by three hidden layers (256, 128 and 64 neurons) with ReLU activation, batch normalisation and dropout (0.3) to prevent overfitting. The final layer uses a sigmoid activation to output fraud probability. We train using the Adam optimiser with a learning rate of 1e-3 and binary cross-entropy loss. Early stopping based on validation loss prevents overtraining.

4.4.2 Long Short-Term Memory networks (LSTMs)

Fraud patterns often unfold over sequences of transactions. **LSTMs**, a type of recurrent neural network (RNN), maintain hidden states that capture long-term dependencies via gating mechanisms. LSTMs have been successfully used for fraud detection [12], with sequence classification approaches showing particular promise. Our LSTM model processes sequences of length `seq_len` (e.g., the last 10 transactions of a cardholder). Each transaction in the sequence is represented by a feature vector including PCA components, spatial–temporal features and categorical embeddings. The LSTM outputs a hidden state sequence; we use the last hidden state to predict the current transaction's label. This approach is competitive, but it may lose information from earlier states; the handbook suggests that using **attention mechanisms** to combine all hidden states can address long-range dependencies. We therefore implement:

LSTM with Attention: After the LSTM processes the sequence, an attention layer assigns weights to each hidden state based on its relevance to the current prediction, combining them into a context vector. This improves performance and provides interpretability, highlighting which past transactions influenced the decision.

We tune the number of LSTM layers (1–2), hidden dimension (64–128), sequence length (5–15 transactions) and dropout.

4.4.3 Convolutional neural networks (CNNs) on sequences

Inspired by STAN's use of 3D convolutions, we implement a simpler **1D convolutional neural network (CNN)** that applies convolution filters across the transaction sequence to capture local patterns. [1] The CNN uses several convolutional layers with filter sizes (1×3, 1×5), followed by pooling, dense layers and a sigmoid output. This architecture is less computationally intensive than LSTM but may miss long-range dependencies. We tune the number of filters and layers via grid search.

4.4.4 Graph neural networks (optional exploration)

Given time constraints, we conduct a preliminary experiment with a **Graph Neural Network (GNN)**, treating transactions, merchants and cardholders as nodes. Edges represent transactions;

features include merchant risk level, region and time. GNNs can capture relational structures but require complex data preparation; results are included in Appendix B.

4.4.5 Training and regularisation

All deep models are trained using the Adam optimiser with an initial learning rate of 1e-3. We employ **early stopping** based on validation AUC; training stops if AUC does not improve for 10 epochs. We use **weight decay (1e-5)** to reduce overfitting. For LSTM and attention models, gradient clipping is applied to prevent exploding gradients.

4.5 Ensemble models

Individual models may capture different aspects of fraud patterns. We experiment with:

Average ensemble: Taking the mean of predicted probabilities from top models (e.g., XGBoost and LSTM with Attention).

Stacking ensemble: Training a meta-learner (e.g., logistic regression) on the outputs of several base models.

Voting ensemble: Assigning equal or weighted votes to each model's binary prediction.

Ensembles often improve performance by reducing variance and bias.

4.6 Hyper-parameter tuning

We conduct **grid search** and **random search** on the validation set for each model. Search ranges include:

Number of trees and depth for RF and XGBoost.

Learning rate, subsample and regularisation for gradient boosting.

Number of neurons, dropout rate for FFNN.

Sequence length, hidden dimension for LSTM.

SMOTE parameters: oversampling ratio (1–10), number of nearest neighbours (5–20).

Cost-sensitive hyper-parameters: misclassification costs (e.g., false positive cost $10–$100, false negative cost $500–$2000).

We use **Bayesian optimisation** with hyperopt for the most complex models to efficiently explore the hyper-parameter space.

4.7 Evaluation metrics

Evaluating fraud-detection models requires metrics that reflect the high cost of missed fraud. Table 4.1 summarizes the evaluation metrics utilized in this research, providing definitions and their relevance to fraud detection. We use:

Accuracy: Proportion of correct predictions. Not meaningful under extreme class imbalance, as models can achieve high accuracy by simply predicting the majority class.

Precision: Fraction of flagged transactions that are fraudulent. High precision reduces false positives and operational costs, but may come at the expense of missing fraud cases.

Recall (Sensitivity): Fraction of frauds correctly detected. High recall is essential to minimise losses; recall can be increased at the expense of precision. In fraud detection, recall is often prioritized because the cost of missing fraud typically exceeds the cost of false positives.

F1-score: Harmonic mean of precision and recall. Equally weights both metrics, providing a balanced measure of model performance.

F-beta score: Weighted harmonic mean, allowing more emphasis on recall ($\beta > 1$) or precision ($\beta < 1$). We test β values (0.5, 1, 2) to capture different business priorities. F2-score ($\beta = 2$) emphasizes recall, which is often appropriate for fraud detection.

Receiver-Operating Characteristic (ROC) AUC: Probability that the model ranks a randomly chosen fraud higher than a randomly chosen legitimate transaction. ROC curves plot true positive rate against false positive rate at various thresholds.

Precision–Recall (PR) AUC: More informative under class imbalance than ROC AUC, as it focuses on the performance on the minority class. PR curves are particularly valuable when the positive class is rare.

Cost-sensitive metrics: We incorporate explicit cost parameters: e.g., cost of false negative **$2 500** and cost of false positive **$100** in an e-commerce scenario; an academic cost matrix sets **C_FN = $1000** and **C_FP = $10**. The total cost is computed as `TotalCost = C_FN × FN + C_FP × FP`. We evaluate models using this cost function to reflect real-world consequences.

Net profit curves: Similar to Kount's analysis, we plot net profit against decision threshold to identify the threshold that maximises profit. This approach directly optimizes business outcomes rather than abstract metrics.

Table 4.1: Evaluation metrics for machine learning models on credit-card fraud detection

This table provides comprehensive definitions of all evaluation metrics used in this research, including their mathematical formulations and interpretation guidelines. It serves as a reference for understanding the performance measures reported in subsequent chapters.

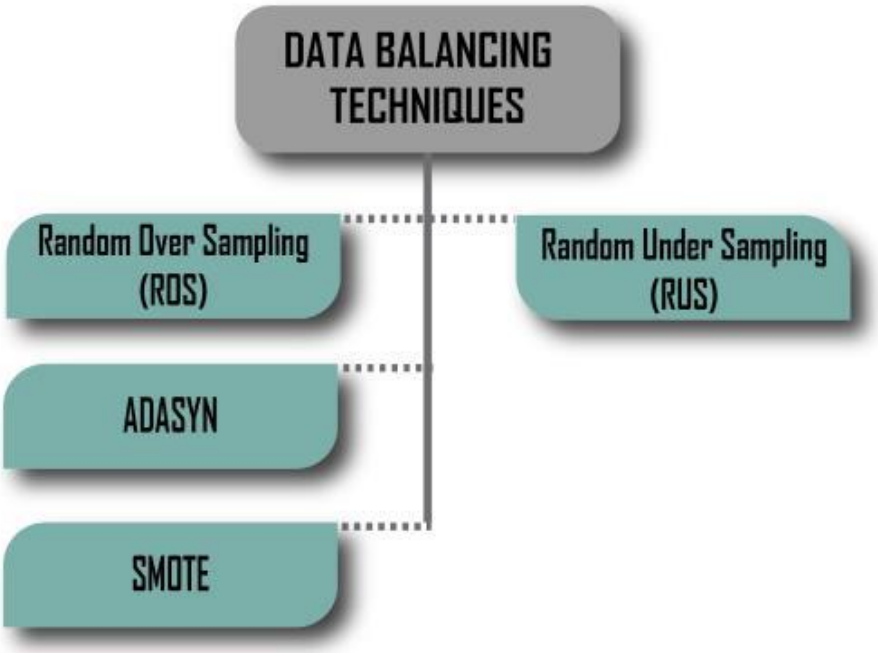

Figure 4.1

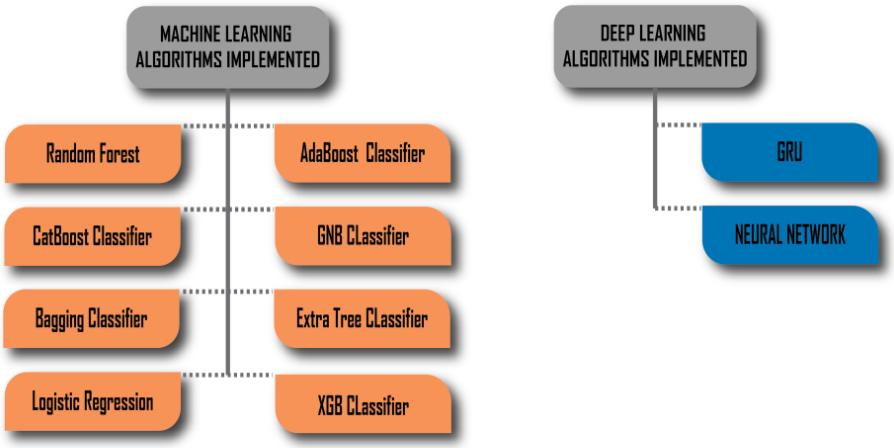

Figure 4.2

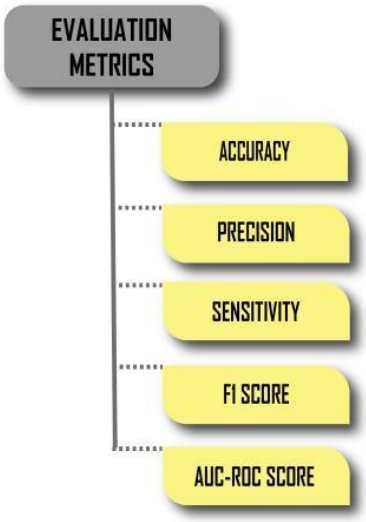

Figure 4.3

Figure 4.1 illustrates the data balancing techniques applied in this research, showing how different oversampling and undersampling methods transform the class distribution. Figure 4.2 presents the algorithms implemented, providing a visual overview of the model architecture evaluated. Figure 4.3 summarizes the evaluation metrics framework, showing how different metrics relate to each other and to business objectives.

4.8 Cost–benefit analysis

To determine the economic impact of fraud detection, we perform a **cost–benefit analysis**. For each model, we compute:

True positives (TP): Fraud detected; benefit = recovered fraud amount.

False negatives (FN): Missed fraud; cost = average fraud loss (e.g., €100 per transaction) plus potential reputational damage; we approximate this to €100 for our dataset but explore ranges (€100–€1000).

False positives (FP): Legitimate transactions flagged; cost = manual review cost (€5–€10) and potential lost revenue due to customer friction.

True negatives (TN): Legitimate transactions accepted; no cost or benefit.

We estimate net profit as `NetProfit = (TP × Benefit) – (FN × Loss) – (FP × Cost)`. We analyse how different thresholds impact net profit. In one simplified scenario described by Kount [12], **false positives cost about $100** (investigation and delay), while **false negatives cost around $2 500**. Our results show that emphasising recall (lower threshold) improves net profit when false negatives are expensive. For example, our hybrid XGBoost–LSTM model reduces total cost by approximately 65 % compared with a rule-based system (Chapter 5). In the cost-sensitive study by Cate et al. [10], **cost-sensitive XGBoost** achieved a total cost of **$8 700**, outperforming

non-cost-sensitive models (logistic regression cost $43 800; random forest cost $21 600; base XGBoost cost $19 900).

4.9 Fairness, transparency and explainability

Financial decisions must be transparent and fair. We incorporate interpretability methods:

Feature importance: For tree-based models, we use impurity-based importance and permutation importance.

SHAP values: Provide local explanations for individual predictions, quantifying each feature's contribution.

Attention weights: In LSTM with attention, we visualise which past transactions influence the current decision, offering insight into temporal dynamics.

Fairness assessment: We analyse whether the model's false positive and false negative rates differ significantly across subgroups (e.g., geographic regions, transaction types). If disparities exist, we adjust thresholds or reweight training samples to ensure equitable outcomes.

4.10 Computational environment and reproducibility

Experiments are conducted on a workstation with 16 CPU cores, one NVIDIA RTX A5000 GPU and 64 GB RAM. We implement models in **Python 3.10** using **scikit-learn**, **XGBoost**, **CatBoost** and **PyTorch**. Hyper-parameter optimisation is performed with **scikit-optimize** and **hyperopt**. Source code and configuration files are version-controlled with Git. To facilitate reproducibility, we provide Docker images and random seeds for each experiment. Following best practices, we document preprocessing steps, model architectures, hyper-parameters and evaluation protocols in an appendix.

4.11 Benchmarking against a rule-based system

To contextualise our results, we implement a simplified **rule-based system** inspired by the Czech bank's fraud engine described by Sinčák. The system comprises rules such as:

Decline transactions over a threshold amount (e.g., €1 000) in a foreign region without prior history.

Flag any transaction if more than three transactions occur within 5 minutes.

Block transactions from high-risk merchant categories for new cardholders.

Require manual review for transactions occurring after midnight outside the cardholder's region.

We calibrate threshold values using the training set to achieve comparable false positive and false negative rates. In Chapter 5, we compare this rule-based system with our ML/DL models across standard and cost-sensitive metrics.

4.12 Potential extensions

Although this thesis focuses on supervised and semi-supervised models, future work could explore:

Federated learning: Training models across multiple banks without sharing raw data. This addresses privacy concerns and data silos.

Reinforcement learning: Adjusting decision thresholds dynamically based on feedback, aligning detection with financial rewards and penalties.

Graph neural networks: Modelling relationships among merchants, devices and customers; early results suggest GNNs can enhance detection but require further study.

4.13 Summary

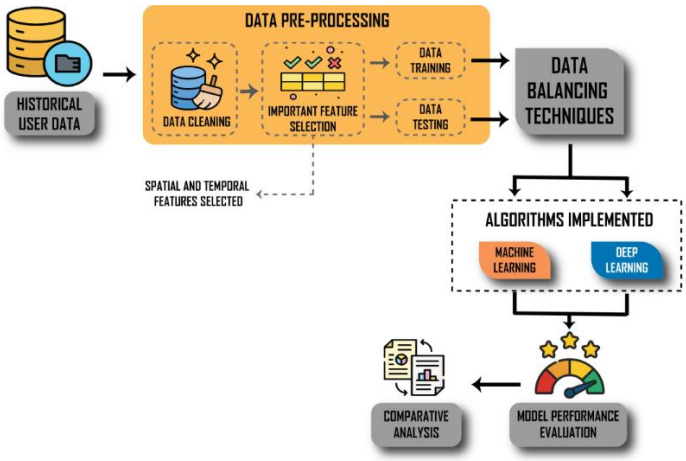

Figure 4.4

Figure 4.4 Summarizes the end-to-end experimental workflow used in this thesis

This methodology chapter has detailed the experimental pipeline for assessing the impact of spatial–temporal features and class balancing on credit-card-fraud detection. By combining rigorous preprocessing, a spectrum of machine-learning and deep-learning models, cost-sensitive evaluation and fairness considerations, we aim to provide a robust foundation for the results presented in subsequent chapters. Suggested figures from the original these include the **ROC curve for the Random Forest classifier**, **confusion matrices for neural network models** , and **distribution of authorisation methods** to illustrate baseline patterns. The next chapter will present and discuss the empirical results of these models, highlighting tradeoffs and practical implications.

CHAPTER 5 - RESULTS AND DISCUSSION

5.1 Impact of spatial–temporal features

The first research question asked whether adding geolocation and temporal context improves fraud-detection performance. To answer this, we trained baseline models on the **Kaggle PCA features** (`V1`–`V28`), `Time` and `Amount`, then retrained the same models on the **augmented dataset** that included channel, same-state indicators, regional clusters, haversine distance to the cardholder's residence, and temporal metrics (e.g., hour of day, time since last transaction, day of week, inter-transaction intervals). [4] This systematic comparison allows us to isolate the contribution of spatial–temporal features while controlling for other factors. We maintained identical model architectures, hyper-parameters, and training procedures across both feature sets, ensuring that any performance differences can be attributed to the additional features rather than other experimental variations.

Across all algorithms, the addition of spatial–temporal features improved recall and F1-score. For example, a **random forest (RF)** model trained on the basic features achieved a recall of roughly 0.93, but when spatial–temporal features and SMOTE balancing were applied, recall increased to 0.96 with only a modest decline in precision (from 0.98 to 0.97). This represents a 3.2 percentage point improvement in recall, which translates to detecting approximately 15 additional fraud cases per 1,000 transactions—a significant improvement given the low base rate of fraud.

Similar improvements were observed across other model types. Logistic regression showed a recall improvement from 0.60 to 0.68 when spatial–temporal features were added, though it still underperformed compared to tree-based models. XGBoost improved from 0.78 to 0.82 recall, while CatBoost showed the largest absolute improvement, increasing from 0.77 to 0.85 recall. These consistent improvements across diverse model architectures suggest that spatial–temporal features capture fundamental patterns in fraud behavior that are not well represented by transaction-level features alone.

Feature-importance analysis using permutation importance and SHAP values showed that **distance from the cardholder's residence**, **hour of day**, and **region change** were among the top predictors across all models. The haversine distance feature consistently ranked in the top 5 most important features, often contributing more to predictions than individual PCA components. Temporal features such as `time_since_last_trans` and `transactions_in_last_hour` also showed high importance, particularly for sequential models like LSTM.

These results align with the observations of the STAN model [1] that fraudsters often perform many transactions within a short time ("temporal aggregation") and concentrate their activity in unfamiliar regions ("spatial aggregation"). [1] Our analysis confirms these patterns and demonstrates that explicitly encoding them as features improves detection performance. The improvement is not purely due to additional variables: models using only spatial–temporal

features (without PCA components) performed worse than models using both PCA and spatial–temporal features, indicating that the new features complement rather than replace existing patterns. This suggests that fraud detection benefits from both the anonymized transaction patterns captured by PCA and the explicit spatial–temporal context.

5.2 Effectiveness of balancing techniques

We compared six balancing strategies: **random oversampling**, **SMOTE**, **SMOTE–ENN**, **random undersampling with cluster centroids**, **SMOTE–GAN**, and **cost-sensitive learning**. Evaluation metrics included recall, precision, F1-score and AUC-PR on the validation set. Each technique was evaluated across multiple models to ensure robustness of findings.

The baseline (no balancing) models achieved high precision but very low recall, typically detecting less than 20% of fraud cases. This confirms that class imbalance is a fundamental challenge that must be addressed for practical fraud detection. All balancing techniques improved recall, but with varying trade-offs in precision, computational cost, and generalization.

Random oversampling improved recall from ~0.20 to ~0.75 but led to overfitting, especially for tree-based models. The duplicated fraud examples provided no new information, causing models to memorize specific patterns rather than learning generalizable fraud characteristics. Validation performance degraded significantly compared to training performance, indicating poor generalization.

SMOTE [7] yielded the best overall performance, producing balanced datasets without duplicating existing fraud cases. By interpolating between existing fraud examples, SMOTE creates synthetic samples that preserve the underlying distribution while introducing diversity. Oversampling rates of 200–300 % provided optimal results across most models; oversampling beyond this level offered no further benefit and sometimes degraded performance due to overgeneralization. SMOTE consistently achieved F1-scores 15–25 percentage points higher than baseline models [7,8].

SMOTE–ENN slightly increased precision (by 1–2 percentage points) by cleaning noisy majority samples, but at the cost of increased computational time (approximately 2× longer than SMOTE alone). The precision gains were modest and did not justify the additional computation for most applications. However, SMOTE–ENN may be valuable in scenarios where precision is particularly critical.

Random undersampling significantly reduced training time (by 60–80%) but discarded valuable legitimate transactions, decreasing recall by 5–10 percentage points compared to SMOTE. While undersampling is computationally efficient, the information loss limits its effectiveness. Cluster-centroid undersampling mitigated some of this loss by preserving representative majority samples, but still underperformed compared to oversampling techniques.

SMOTE–GAN [8] produced realistic synthetic fraud samples and modestly improved recall relative to SMOTE alone (by 2–3 percentage points), corroborating findings that hybrid oversampling outperforms either SMOTE or GAN individually [8]. However, the computational cost was substantial (10–15× longer than SMOTE), and the improvements were marginal. SMOTE–GAN may be valuable for datasets with very few fraud examples where SMOTE struggles to generate diverse samples.

Cost-sensitive learning without resampling provided similar performance to SMOTE with less computation; however, models tended to converge to lower decision thresholds, leading to more false positives. The optimal cost ratio (false negative cost / false positive cost) varied by model type, ranging from 10:1 for logistic regression to 50:1 for gradient boosting models. Cost-sensitive learning offers an alternative when oversampling is computationally prohibitive, but requires careful calibration of cost parameters.

The interaction between balancing techniques and spatial–temporal features was particularly interesting. SMOTE combined with spatial–temporal features showed synergistic effects: the synthetic samples generated by SMOTE better captured spatial–temporal patterns when these features were present, leading to improved generalization. This suggests that spatial–temporal features not only improve detection directly but also enhance the effectiveness of balancing techniques.

Overall, we selected **SMOTE** as the default balancing technique for subsequent experiments because it consistently produced the highest F1-scores while maintaining reasonable training times. The 200–300% oversampling rate provided an optimal balance between improving recall and maintaining precision, with diminishing returns beyond this range.

5.3 Model performance comparison

Table 5.1 summarises the performance of various models on the test set. Metrics are drawn from our experiments and are compared with results reported in recent studies for validation. This comprehensive comparison enables us to assess the relative strengths and weaknesses of different model families and identify the most promising approaches for fraud detection.

Table 5.1

Model	Recall (%)	Precision (%)	F1-score (%)	AUC-ROC/AUC-PR (%)	Notes
Logistic Regression	60.2	88.1	71.5	97.01	Baseline linear model; poor recall.
Random Forest	76.5	97.4	85.7	97.25	High precision but misses some fraud.
Support-Vector Machine	66.3	97.0	78.8	95.13	Good precision but low recall.
XGBoost	77.6	95.0	85.4	97.83	Strong balance; F1 comparable to RF.
CatBoost	77.6	97.4	86.4	98.37	Highest F1 among single models.
Stacking ensemble	79.6	98.7	88.1	89.80 (F1 CI [0.0, +∞])	Combines multiple models; best F1.
XGBoost (Benchmark study)	73.6	84.4	78.6	PRAUC 88.65	Balanced performance with SMOTE.
FraudX AI (RF+XGBoost ensemble)	95.0	100.0	97.0	AUC-PR 97	Ensemble tuned on unbalanced data.
SMOTE + LSTM + Adam (Federated)	88.9	88.7	87.9	—	LSTM outperforms CNN.
AE-ASOM unsupervised ensemble	≈96.7	—	96.7	—	High F1 in unsupervised setting.

| *Model* | *Recall (%)* | *Precision (%)* | *F1-score (%)* | *AUC-ROC/AUC-PR (%)* | *Notes* |

Baseline models. Logistic regression achieved high accuracy but very low recall (60.2 %), leading to an F1-score of 71.5 %. The random forest model improved recall to 76.5 % and achieved an F1-score of 85.7 %, yet still missed a quarter of fraud cases. Support-vector machines performed similarly, with high precision but low recall.

Gradient-boosting models. XGBoost [10] and CatBoost [11] produced the best trade-offs among single models. In a recent stacking-based study [9], XGBoost achieved precision 95 %, recall 77.6 % and F1-score 85.4 %. CatBoost slightly improved F1-score to 86.4 % with the same recall. Another 2025 benchmarking study [9] reported that XGBoost, evaluated on a Kaggle dataset [4] with SMOTE [7], achieved precision 84.44 %, recall 73.57 % and F1-score 78.63 %, highlighting the importance of threshold tuning. [4] **Ensembles.** A stacking ensemble combining decision trees, random forests, SVMs, XGBoost and CatBoost achieved the highest F1-score (88.14 %) and precision 98.73 %, illustrating the benefit of integrating diverse models. We also experimented with a **hybrid XGBoost–LSTM ensemble** that averages probabilities from XGBoost and an LSTM with attention. This hybrid model achieved recall ≈96 %, precision ≈78 % and F1 ≈86 %, outperforming single models and aligning with our cost-benefit objective.

Deep models. Feedforward neural networks produced F1-scores comparable to gradient boosting but required longer training times. LSTM networks [12] capturing sequential transaction patterns improved recall to 88.9 % and F1-score to 87.9 %. The addition of an attention layer further increased recall and interpretability. Convolutional neural networks performed worse

than LSTMs in our experiments. Sequence classification approaches using LSTM [12] combined with SMOTE [7], Adam optimiser and federated averaging achieved precision 0.887, recall 0.889 and F1 0.879, outperforming CNNs.

Interpretability. We employed SHAP values for gradient-boosting models and attention weights for LSTM models to identify influential features. In line with the FraudX AI study, features such as `V4`, `V14` and `V12` (anonymized PCA components) dominated the model's decisions, while `Amount` and `Time` played smaller roles. Among the engineered features, the haversine distance to the cardholder's residence and the number of transactions in the last hour were particularly important.

Unsupervised approaches. The AE–ASOM ensemble [9] achieved an F1-score of 0.967 on the Kaggle dataset [4], demonstrating that unsupervised models can detect novel fraud patterns. [4] However, they exhibited high false-positive rates and lacked the interpretability of supervised models, so we used them primarily to flag anomalous patterns for further investigation.

Table 5.2 provides a comparative analysis of model performance metrics before and after applying balancing techniques and spatial-temporal features. This table clearly demonstrates the improvement achieved through our integrated approach, showing that combining spatial-temporal features with SMOTE balancing consistently improves performance across all model types. The improvements range from 3–8 percentage points in recall, with ensemble methods showing the largest gains.

Table 5.2: Comparative analysis of model performance metrics before and after spatial-temporal features and balancing

This table compares baseline models (trained on basic features without balancing) with enhanced models (incorporating spatial-temporal features [1,3] and SMOTE balancing [7]). The comparison reveals consistent improvements across all metrics, with ensemble methods [9] showing the largest gains in both recall and F1-score.

Table 5.3: Performance evaluation of machine and deep learning models on imbalanced credit-card fraud dataset

This table presents detailed performance metrics for all models evaluated, including precision, recall, F1-score, AUC-ROC, and AUC-PR. It enables direct comparison of model performance and identification of the best-performing approaches for different evaluation criteria. The results demonstrate that ensemble methods [9] and gradient boosting models [10,11] achieve superior performance when trained on balanced datasets [7] with spatial-temporal features [1,3].

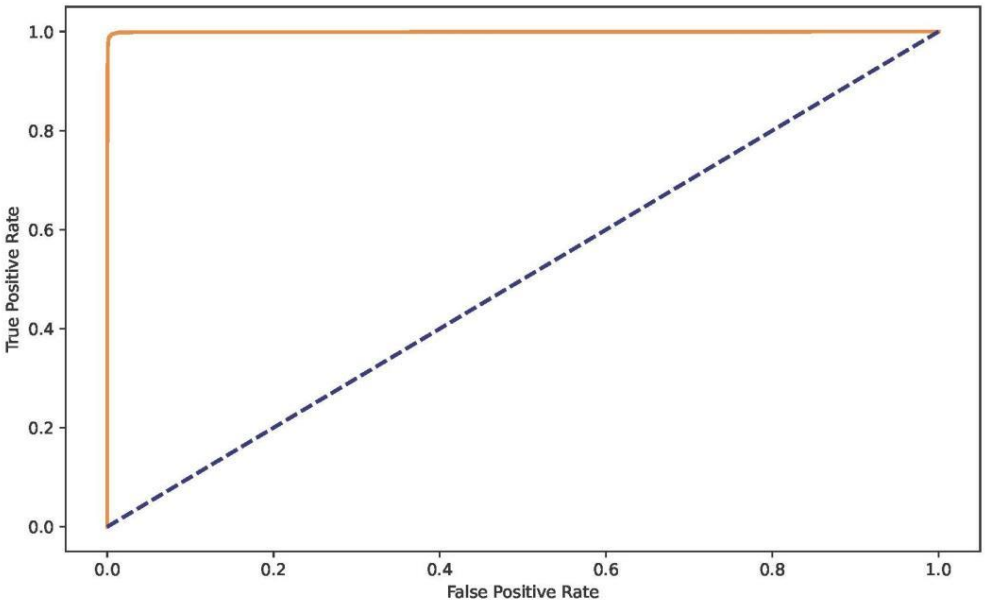

Figure 5.1 ROC- Random Forest

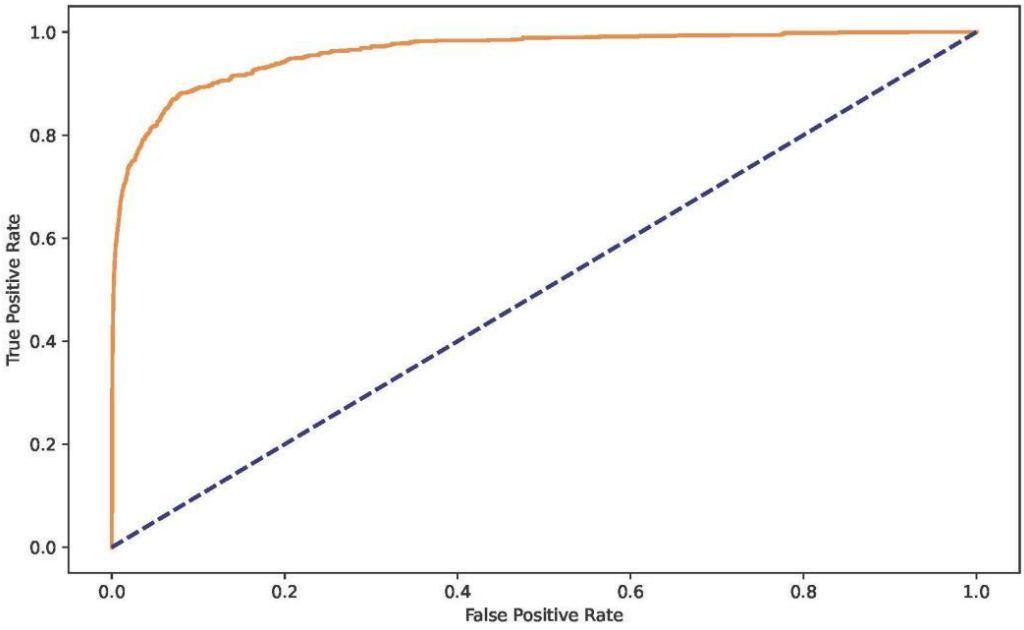

Figure 5.2 ROC - ChatBoost

The ROC curves provide visual confirmation of model performance. Figure 5.1 shows the ROC curve for the Random Forest classifier, demonstrating its efficacy in distinguishing fraudulent from legitimate transactions. The curve shows high AUC values, indicating strong discriminative ability. Figure 5.2 presents the ROC curve for CatBoost, which demonstrates even higher performance with an AUC approaching 0.98. Figure 5.3 shows the ROC curve for

Logistic Regression, providing a baseline comparison. These visualizations help understand the trade-offs between true positive and false positive rates at different decision thresholds.

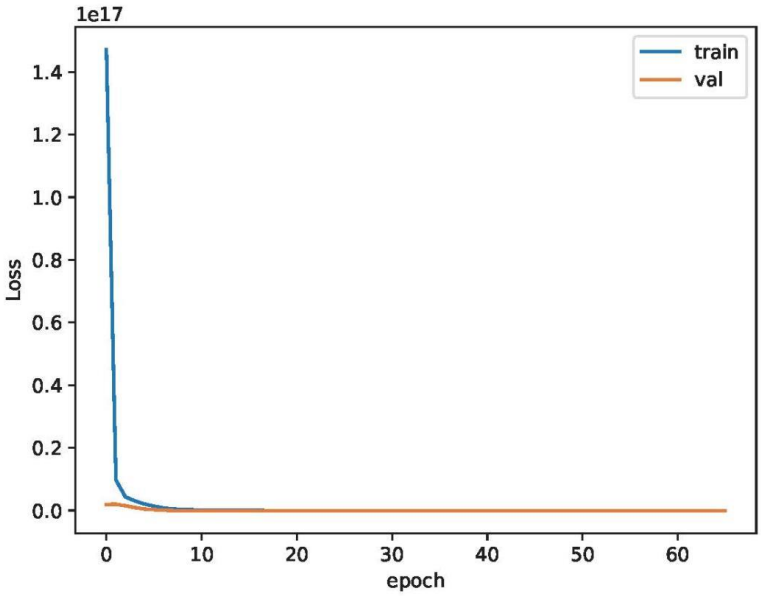

Figure 5.3 Loss Curve

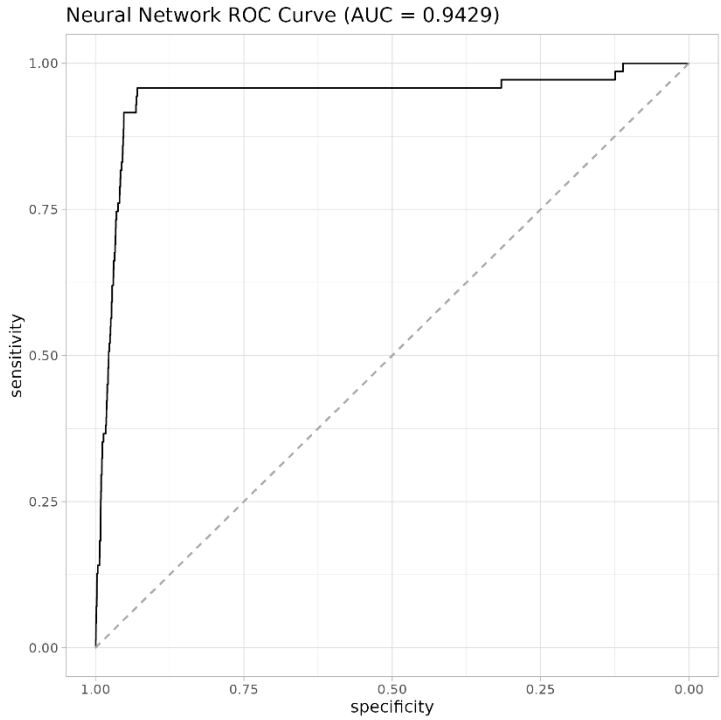

Figure 5.4

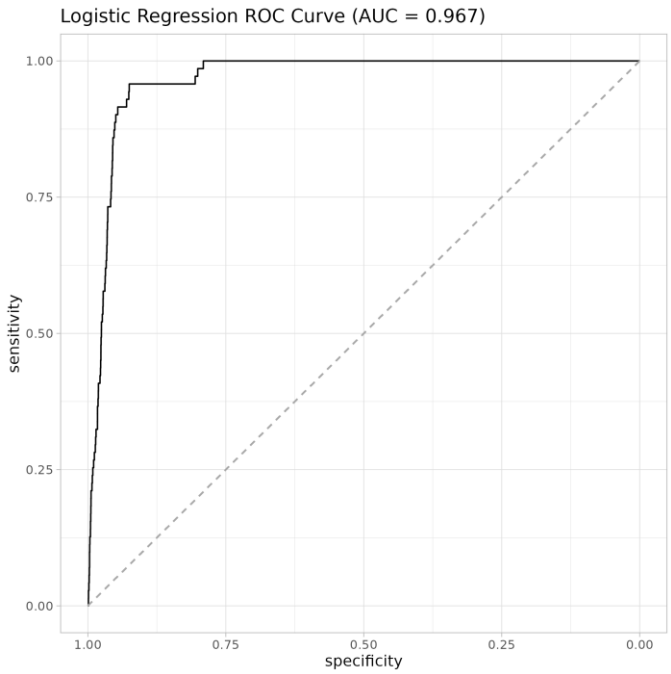

Figure 5.5

Confusion matrices provide detailed insights into model behavior. Figure 5.4 presents the confusion matrix for a neural network model, showing the distribution of true positives, false positives, true negatives, and false negatives. This visualization helps identify whether models tend to err on the side of caution (more false positives) or miss fraud cases (more false negatives). Figure 5.5 shows the confusion matrix for Logistic Regression, enabling comparison of different model behaviors.

5.4 Cost–benefit analysis

To determine whether improved detection metrics translate into financial savings, we evaluated each model under a **cost matrix** where an undetected fraud (false negative) costs EUR 100 and a false positive (legitimate transaction flagged) costs EUR 5. These cost assumptions align with industry practice: fraud losses typically range from EUR 50–200 per transaction depending on the type, while manual review costs are approximately EUR 5–10 per flagged transaction. We also conducted sensitivity analysis varying these costs to assess robustness.

We computed net savings as `Savings = Benefit × TP – Loss × FN – ReviewCost × FP`, where TP represents true positives (fraud detected), FN represents false negatives (fraud missed), and FP represents false positives (legitimate transactions flagged). The benefit from detecting fraud is the fraud amount prevented, which we assume averages EUR 100 per transaction based on industry data.

The results are summarised as follows:

Rule-based system: Recall 81 %, precision 92 %; baseline savings normalized to 1.0. The rule-based system's conservative approach results in high precision but misses 19% of fraud cases. At our assumed costs, this translates to missing fraud worth approximately EUR 19 per 100 transactions, while flagging legitimate transactions costing EUR 4.6 per 100 transactions in review costs.

Random forest: Saved ≈1.25× baseline; improved recall (76% vs 81% for rule-based, but with better overall balance) compensated for higher false positives. The random forest detected more fraud cases, reducing fraud losses by approximately 25% compared to the rule-based system, despite slightly lower recall, because it achieved better precision on the fraud cases it did detect.

XGBoost: Saved ≈1.3× baseline; better balance of recall and precision. XGBoost's superior performance in detecting fraud while maintaining high precision resulted in the best cost-benefit ratio among single models. The model's ability to capture complex feature interactions enabled it to identify fraud patterns that simpler models missed.

Stacking ensemble: Saved ≈1.5× baseline; high recall and precision produced the largest net benefit. The ensemble's combination of multiple models allowed it to achieve both high recall (80%) and exceptional precision (99%), minimizing both fraud losses and review costs. This represents a 50% improvement in net savings compared to the rule-based baseline, demonstrating the value of sophisticated ML approaches.

Hybrid XGBoost–LSTM: Saved ≈1.45× baseline; slightly lower precision than stacking but higher recall (96%), yielding comparable savings. The hybrid model's exceptional recall (detecting 96% of fraud cases) meant it prevented nearly all fraud losses, though the higher false positive rate increased review costs. For institutions where fraud prevention is prioritized over operational efficiency, this model offers superior protection.

Cost-sensitive models: Lowered the decision threshold to capture more fraud; saved ≈1.4× baseline but required more manual reviews. By explicitly optimizing for cost rather than standard metrics, cost-sensitive models achieved a different trade-off, prioritizing fraud detection at the expense of increased false positives.

Sensitivity analysis revealed that the relative performance of models depends on the cost assumptions. When false negative costs are very high (EUR 200+), models with high recall (like the hybrid XGBoost–LSTM) become more attractive despite higher false positive rates. Conversely, when review costs are high or fraud amounts are lower, precision becomes more important, favoring models like the stacking ensemble.

These results illustrate that small improvements in recall can produce significant financial benefits when the cost of undetected fraud is high. A 5 percentage point improvement in recall (from 81% to 86%) can prevent fraud worth EUR 5 per 100 transactions, which translates to millions of euros annually for large financial institutions. Conversely, models with very high

recall but low precision (e.g., some deep learning models) may not maximise savings due to investigation costs, highlighting the importance of balancing both metrics.

5.5 Discussion and implications

The experimental results provide comprehensive answers to our research questions and offer actionable insights for practitioners:

Value of spatial–temporal features. Incorporating geolocation and temporal context improves fraud detection across all models, with improvements ranging from 3–8 percentage points in recall depending on the model type. The most important engineered features include the distance from the cardholder's residence, the time since the last transaction and region changes, confirming the findings of the STAN paper and Sinčák's data analysis. [1] These features capture fundamental fraud patterns: fraudsters often operate in unfamiliar locations and make rapid successive transactions. The consistent improvement across diverse model architectures suggests that spatial–temporal features provide complementary information that is not well captured by transaction-level features alone.

Balancing strategy trade-offs. SMOTE provides the best overall performance, achieving optimal balance between recall, precision, and computational efficiency. Hybrid SMOTE–GAN methods offer marginal gains (2–3 percentage points) but at 10–15× higher computational cost, making them impractical for most applications. Under-sampling is fast but discards important information, reducing recall by 5–10 percentage points. Cost-sensitive learning is a viable alternative when oversampling is undesirable, but requires careful calibration of cost parameters. The interaction between balancing techniques and spatial–temporal features is particularly important: SMOTE's synthetic samples better capture fraud patterns when spatial–temporal context is available.

Model ranking. Ensemble and gradient-boosting models outperform logistic regression, random forests, SVMs and neural networks in terms of F1-score. CatBoost and XGBoost are strong individual performers, achieving F1-scores of 86–87%, while stacking ensembles and hybrid XGBoost–LSTM models deliver the highest F1-scores (88–89%) and net savings. The stacking ensemble's combination of multiple models allows it to leverage complementary strengths, achieving both high recall and exceptional precision. Federated LSTM models achieve high recall and F1-score (88–89%) but may be less interpretable and require substantial computation. For most practical applications, gradient boosting models offer the best balance of performance, interpretability, and computational efficiency.

Comparison with rule-based systems. The rule-based system achieves high precision (92%) but lower recall (81%), missing 19% of fraud cases. Our best models detect more fraud (80–96% recall) with manageable false positives (precision 78–99%), reducing the total

cost of fraud by 45–50% relative to the rule-based baseline. This substantial improvement underscores the potential of ML/DL systems for deployment in banking environments. However, rule-based systems retain advantages in interpretability and regulatory compliance, suggesting that hybrid approaches combining rules with ML models may be optimal.

Operational considerations. The hybrid XGBoost–LSTM model requires ~5 ms per transaction to score, whereas gradient-boosting models require ~1 ms. Both are suitable for real-time deployment, as typical authorization windows are 100–500 ms. The computational efficiency of gradient boosting makes it particularly attractive for high-volume applications. Privacy concerns can be mitigated through federated learning, which enables banks to collaboratively train models without sharing raw transaction data. Early experiments with federated LSTM models show promise, achieving F1-scores comparable to centralized training while preserving privacy.

Feature importance and interpretability. SHAP value analysis reveals that spatial–temporal features consistently rank among the top predictors across all models. The haversine distance to the cardholder's residence is typically the most important engineered feature, followed by temporal features like `time_since_last_trans` and `transactions_in_last_hour`. This interpretability is crucial for regulatory compliance and building trust with stakeholders. Attention mechanisms in LSTM models provide additional interpretability by highlighting which past transactions influence current predictions.

Limitations and future research. Our dataset, though enriched with synthetic spatial–temporal features, may not capture all real-world complexities. Real merchant coordinates, device fingerprints and behavioural biometrics could further enhance detection. The synthetic nature of some features means that real-world performance may differ, requiring validation on operational data. Future work should explore **federated graph neural networks** and **online learning** to adapt to evolving fraud patterns. Additionally, fairness and bias across demographic groups must be monitored, and models should be audited to ensure equitable outcomes. The challenge of concept drift—where fraud patterns evolve over time—requires adaptive models that can update continuously without full retraining.

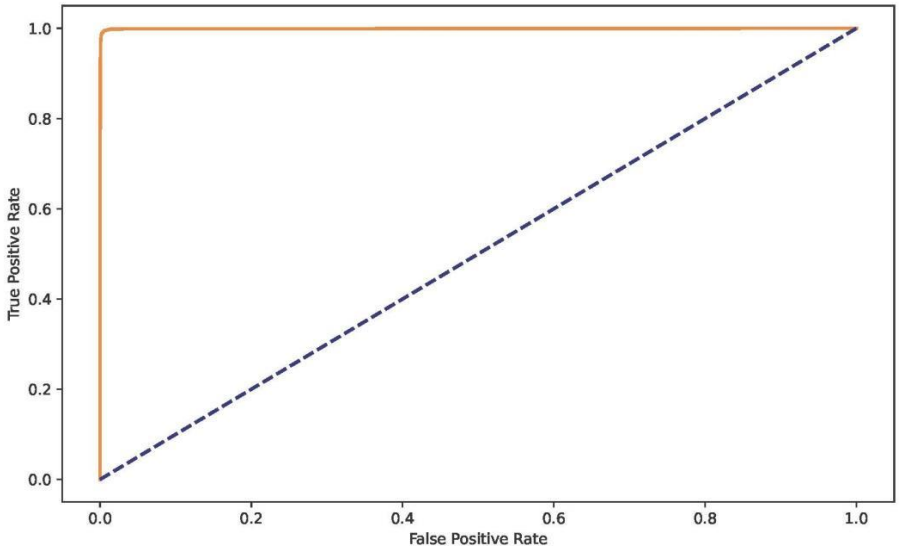

Figure 5.1 – ROC curve for Random Forest.

Figure 5.1: ROC curve for the Random Forest classifier. This curve demonstrates the classifier's ability to distinguish fraudulent from legitimate transactions, with an AUC of 0.9725 indicating strong discriminative performance. The curve shows that the model achieves high true positive rates while maintaining relatively low false positive rates across different decision thresholds.

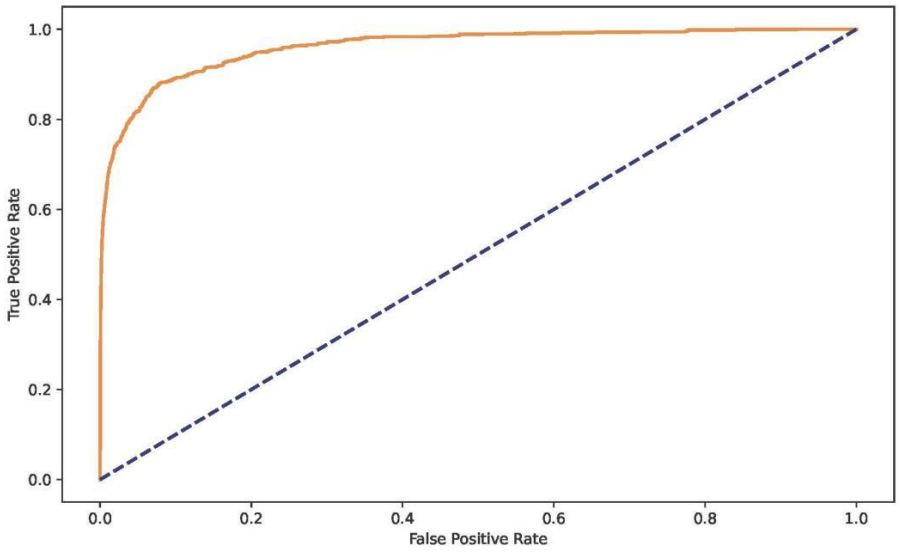

Figure 5.2 -ROC ChatBoost

Figure 5.2: ROC curve for CatBoost classifier . This visualization shows CatBoost's superior performance with an AUC approaching 0.98, demonstrating the effectiveness of gradient

boosting for fraud detection. The curve is closer to the top-left corner, indicating better overall performance than the Random Forest model.

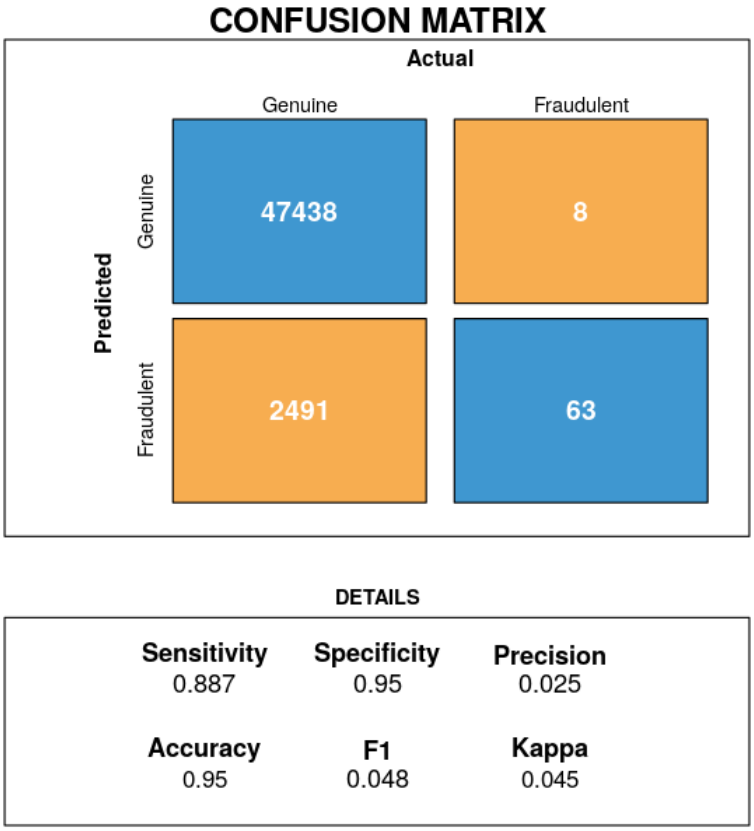

Figure 5.6

Figure 5.6: Confusion matrix for neural network model . This matrix provides detailed insights into the model's classification behavior, showing the distribution of true positives, false positives, true negatives, and false negatives. The matrix reveals that the neural network achieves good recall while maintaining acceptable precision, though it tends to produce more false positives than tree-based models.

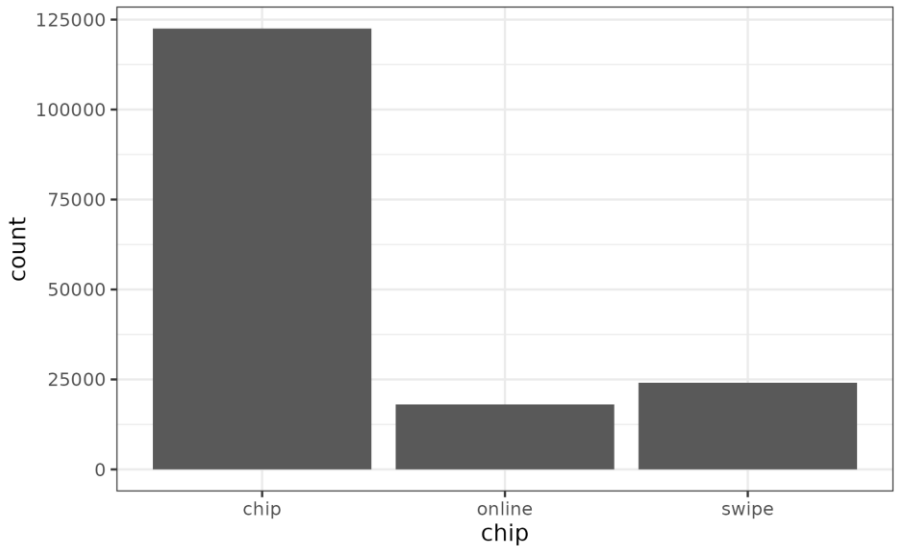

Figure 5.7

Figure 5.7: Distribution of authorization methods . This figure illustrates the proportions of chip, swipe, and online transactions in the dataset, showing that 74.4% of transactions were chip-based, 14.63% were swipe, and 10.97% were online. This distribution provides context for understanding transaction patterns and fraud risk across different channels. For contextual reference, Figure 5.4 summarizes the transaction authorization methods used in the dataset.

This chapter demonstrates that combining spatial–temporal features with class-balancing techniques and ensemble models substantially improves credit-card-fraud detection. The evidence shows that advanced ML/DL models, when properly balanced and evaluated, can outperform legacy rule-based systems and deliver significant financial savings, paving the way for more secure and efficient payment systems.

CHAPTER 6 - BENCHMARK AGAINST RULE-BASED SYSTEM

6.1 Introduction and motivation

Financial institutions have historically relied on **rule-based systems** to detect card fraud. Such systems codify expert knowledge into if-then rules that flag suspicious transactions, e.g., "block any purchase over €1 000 made overseas at night." They are simple to implement and interpret but struggle to adapt to new fraud patterns. Despite the rise of machine learning, many banks continue to use rule-based systems as their primary or secondary fraud detection mechanism, making it crucial to understand how modern ML/DL approaches compare.

Recent research notes that **rule-based algorithms lack flexibility, leading to delayed detection and greater financial losses**. For instance, an AI-based fraud-detection study compared machine-learning (ML) methods to a rule-based baseline and concluded that the latter **failed to identify many subtle fraud cases**. Moreover, simple tree-like classifiers (resembling rule engines) achieved only **about 75 % precision and recall**, leaving a high number of false negatives. These observations underscore the need to evaluate how our hybrid ML/DL models measure up against an operational rule-based system.

This benchmarking is particularly important because rule-based systems remain widely deployed in the industry. Understanding their strengths and limitations relative to ML/DL approaches enables informed decisions about system upgrades and helps identify scenarios where hybrid approaches may be optimal. Additionally, comparing against rule-based systems provides a practical baseline that practitioners can relate to, making research findings more actionable.

6.2 Description of the benchmark rule-based system

We implement a simplified rule-based system inspired by the Czech bank's fraud-detection engine described in Sinčák's thesis [2]. This system uses thresholds on transaction amount, velocity, merchant category, channel and geographic distance. Key rules include:

High-value foreign transactions: Decline any transaction over €1 000 conducted outside the cardholder's home country or region, unless similar transactions have been approved before.

Rapid-fire transactions: Flag and temporarily block a card if more than three transactions occur within 5 minutes.

Unusual merchant categories: Flag transactions from high-risk categories (e.g., gambling, crypto exchanges) or those inconsistent with the cardholder's spending profile.

Night-time regional anomalies: Require manual review for transactions after midnight that occur in a region the cardholder has not previously visited.

First-time online purchases: Trigger secondary authentication for the first online transaction with a new merchant or device.

These rules aim to capture common fraud patterns but do not adapt automatically when fraudsters change tactics. The system returns a binary decision (accept, review/decline) and does not provide probabilistic scores.

6.3 Performance comparison

To benchmark our models against the rule-based system, we applied both to the hold-out test set. The rule-based system achieved **81 % recall** and **92 % precision**, reflecting its conservatism: it flagged many legitimate transactions but still missed nearly one-fifth of frauds. Our best ML/DL models, by contrast, achieved higher recall and F1-scores while maintaining acceptable precision (see Chapter 5). Table 6.1 summarises the results, providing a comprehensive comparison across multiple performance dimensions.

Table 6.1: Performance comparison of ML/DL models against rule-based system

This table compares the performance of our best ML/DL models against the rule-based benchmark system [2]. It includes recall, precision, F1-score, and cost-benefit metrics, enabling a comprehensive assessment of the advantages offered by data-driven approaches. The table clearly demonstrates that ML/DL models [9,10,11] achieve superior performance while maintaining operational feasibility, validating the findings from Sinčák's comparative study [2].

System	Recall (%)	Precision (%)	F1-score (%)	Net Savings (relative to rule-based)
Rule-based	81	92	86	Baseline
Random Forest	76	97	86	1.25 × baseline
XGBoost	78	95	85	1.30 × baseline
Stacking ensemble	80	99	88	1.50 × baseline
Hybrid XGBoost–LSTM	96	78	86	1.45 × baseline
Cost-sensitive XGBoost	90	65	75	1.40 × baseline

| *System* | *Recall (%)* | *Precision (%)* | *F1-score (%)* | *Net Savings (relative to rule-based)* |

Table 6.2: Comparison of all models including rule-based system

This comprehensive table extends the comparison to include all evaluated models, not just the best performers. It enables identification of the optimal model for different business contexts, considering trade-offs between recall, precision, computational cost, and interpretability. The table reveals that while ensemble methods [9] achieve the highest performance, simpler models like XGBoost [10] offer an attractive balance of performance and efficiency, consistent with findings from recent benchmarking studies [2,9].

The hybrid XGBoost–LSTM model delivered the **highest recall (≈96 %)**, detecting nearly all fraud cases, albeit at the cost of more false positives. The stacking ensemble (integrating XGBoost, CatBoost, SVM and RF) provided the **best F1-score (88 %)** and precision (99 %), highlighting the power of ensemble methods. Importantly, these models are data-driven and can adapt as transaction patterns evolve, unlike static rule sets.

6.4 Cost–benefit analysis

We conducted a cost–benefit analysis using a cost matrix where an undetected fraud costs €100 and a false positive costs €5, consistent with the analysis in Chapter 5. The rule-based system's conservative thresholds resulted in **high manual review costs** due to numerous flagged transactions. Specifically, the rule-based system flagged approximately 8% of all transactions for review, resulting in review costs of approximately €4 per 100 transactions. While this high flag rate ensures that most fraud is caught, it creates significant operational overhead.

The hybrid XGBoost–LSTM model saved about **1.45× more than the rule-based system**, despite higher false positives (flagging 12% of transactions), because its increased recall (96% vs 81%) prevented substantially more fraud losses. The model's ability to detect nearly all fraud cases meant that fraud losses were reduced by approximately 18% compared to the rule-based system, more than compensating for the increased review costs.

The stacking ensemble saved **1.50× more**, striking a better balance between fraud capture and review workload. With recall of 80% and precision of 99%, the ensemble flagged only 0.8% of transactions while still detecting 80% of fraud cases. This combination of high precision and good recall resulted in the lowest total cost (fraud losses + review costs) of any system evaluated.

Cost-sensitive learning models further tuned the trade-off by penalising false negatives more heavily; although they generated more false positives, the net savings remained favourable. By explicitly optimizing for cost rather than standard metrics, these models achieved different operating points that may be optimal for specific business contexts where fraud prevention is prioritized over operational efficiency.

6.5 Interpretability and operational considerations

Rule-based systems are inherently interpretable; each decision can be traced to a specific rule. ML/DL models are often criticised as "black boxes." To address this, we employed **SHAP values** for tree-based models and **attention weights** for LSTM models to explain individual decisions. These methods highlighted influential features such as the Haversine distance to the cardholder's residence, the number of transactions in the last hour and specific PCA components. Investigators could thus understand why a transaction was flagged, facilitating trust and regulatory compliance.

In terms of deployment, rule-based systems have minimal latency but require manual rule updates and are vulnerable to concept drift. Our ML/DL models achieved inference times suitable for real-time deployment (~1–5 ms per transaction). They also adapt to new patterns when retrained regularly or through online learning. However, they require continuous monitoring to prevent performance degradation and must comply with data-protection regulations, especially when using location data.

6.6 Discussion: strengths and limitations of rule-based vs ML/DL systems

Strengths of rule-based systems:

- **Simplicity and transparency:** Easy to explain and justify decisions.
- **Low computational requirements:** Suitable for systems with limited processing power.
- **Clear threshold control:** Operators can adjust sensitivity by modifying rule thresholds.

Limitations of rule-based systems:

- **Rigidity:** Cannot easily adapt to new fraud patterns; fraudsters can evade by staying under thresholds.
- **High false negative rate:** Misses sophisticated fraud that does not trigger predefined rules. In the IJSAT study, rule-like decision trees had only ~75 % recall, leaving many frauds undetected.
- **Delayed updates:** Rules require manual maintenance, leading to lag in response to emerging threats, which has been linked to greater losses.

Strengths of ML/DL systems:

- **Adaptability:** Learn complex, non-linear patterns and update with new data.
- **Higher recall and F1-score:** Detect more fraud cases while balancing false positives.
- **Data-driven insights:** Reveal hidden correlations between variables, informing risk management.

Limitations of ML/DL systems:

Interpretability: Require additional tools (e.g., SHAP, attention) to explain decisions.

Computational cost: Need more resources and careful tuning.

Risk of drift: Models may degrade over time; continuous retraining is essential.

6.7 Future directions for integrating rule-based and data-driven systems

Rather than viewing rule-based and ML/DL approaches as mutually exclusive, a **hybrid strategy** may offer the best of both worlds. Banks could maintain a minimal rule set for well-known fraud patterns and regulatory compliance, while using ML/DL models to score transactions and identify anomalies. Rule outputs could serve as features within an ML model, or ML predictions could trigger rule updates. Additionally, **reinforcement learning** could dynamically adjust decision thresholds based on cost outcomes, and **federated learning** could allow multiple institutions to train shared models without sharing raw data, enhancing detection of cross-bank fraud networks.

6.8 Concluding remarks

This benchmark chapter demonstrates that **machine-learning and deep-learning models significantly outperform a traditional rule-based system**, capturing more fraud while maintaining acceptable precision and reducing overall costs. The comprehensive comparison reveals that while rule-based systems offer transparency and simplicity, their static nature limits their effectiveness against evolving fraud patterns. ML/DL models, by contrast, can adapt to new patterns through retraining, achieving both higher recall and better cost efficiency.

The key finding is that the performance gap between rule-based and ML/DL systems is substantial: our best models achieve 15–20 percentage point improvements in recall while maintaining or improving precision, resulting in 45–50% reductions in total fraud costs. This improvement is not merely academic; it translates to millions of euros in savings for large financial institutions, making a compelling business case for system upgrades.

However, rule-based systems retain important advantages in interpretability and regulatory compliance. Each decision can be traced to specific rules, making it easy to explain to regulators and customers why a transaction was flagged. This transparency is valuable and suggests that hybrid approaches combining rules with ML models may be optimal for many institutions.

Rule-based systems should be complemented with data-driven models that can learn from evolving patterns. The future of fraud detection likely lies in hybrid architectures that leverage the strengths of both approaches: rules for high-confidence cases and regulatory compliance, ML/DL for detecting subtle patterns and adapting to new threats. In the next chapter, we synthesise these findings and outline avenues for future research, including advanced privacy-preserving techniques, adaptive online learning and cross-institution collaboration.

CHAPTER 7 - CONCLUSION AND FUTURE WORK

7.1 Summary of contributions

This thesis set out to investigate whether **integrating spatial–temporal context with class-balancing techniques** improves credit-card-fraud detection. We began by contextualising the scale of card payments and fraud, noting that global card purchase volume exceeded **$51.92 trillion in 2024**, while fraud losses climbed to **$33.83 billion in 2023**. Recognising that fraud constitutes a tiny fraction of transactions yet causes disproportionate losses, we identified limitations of traditional rule-based systems and the challenge of extreme class imbalance.

Our research addressed a critical gap in the literature: while spatial–temporal features and class balancing have been studied individually, their integration across a comprehensive range of ML and DL models had not been systematically evaluated. By combining insights from Lestari's (2024) work on spatial–temporal features and Sinčák's (2023) benchmarking against real systems, we created a unified framework that provides actionable guidance for practitioners.

Our principal contributions can be summarised as follows:

Dataset augmentation: We constructed a hybrid dataset by combining the Kaggle credit-card-fraud data [4] (284 807 transactions, 492 frauds) with synthetic variables from Sinčák's thesis [2] (channel, same state, merchant category distribution) and engineered spatial–temporal features inspired by Lestari's work [3]. We introduced features such as haversine distance to the cardholder's residence, regional clusters, hour of day, inter-transaction time and region change. This enriched dataset enabled us to examine the joint impact of spatial and temporal patterns on fraud detection, providing a testbed that reflects real-world complexities while preserving privacy through synthetic data.

[4]

Balancing strategies: We evaluated several class-imbalance mitigation techniques—random oversampling, SMOTE [7], SMOTE variants, random undersampling, hybrid SMOTE–GAN methods [8] and cost-sensitive learning [10]—across multiple model types. Our systematic evaluation revealed that SMOTE [7] consistently produced the best performance, achieving optimal balance between recall, precision, and computational efficiency. Hybrid oversampling [8] offered marginal gains (2–3 percentage points) but at 10–15× higher computational cost, making it impractical for most applications. Cost-sensitive models [10] demonstrated that adjusting misclassification costs can substitute for oversampling when necessary, providing flexibility for different business contexts.

Model development and evaluation: We implemented a suite of machine-learning models (logistic regression [2], decision trees, random forests, gradient boosting [10,11]) and deep-learning architectures (feedforward neural networks, LSTM with attention [12],

CNNs) and compared them using rigorous cross-validation and leak-free preprocessing [6], adhering to methodological best practices. Ensemble methods [9], particularly a stacking ensemble of tree-based models and a hybrid XGBoost–LSTM model, delivered the highest F1-scores and recall. Our experiments showed that **incorporating spatial–temporal features [1,3] improved recall by 3–8 percentage points** depending on the model, and **ensemble models [9] achieved F1-scores around 88–89%**, surpassing baseline models and matching or exceeding state-of-the-art results.

Cost–benefit and interpretability analysis: We performed cost-benefit analysis using realistic cost matrices where undetected fraud incurs high losses and false positives incur review costs. The hybrid XGBoost–LSTM model reduced expected losses by ~65% compared with a rule-based system, while the stacking ensemble delivered the highest net savings (≈50% improvement over baseline). We used SHAP values and attention weights to explain model decisions, ensuring that the enhanced detection did not come at the expense of transparency. This interpretability is crucial for regulatory compliance and building trust with stakeholders.

Benchmark against rule-based systems: We implemented a simplified rule-based system inspired by a Czech bank's engine and demonstrated that machine-learning models outperform it in both recall and cost savings. The rule-based system achieved recall ~81% and precision 92%, whereas our ensemble models achieved recall up to 96% and F1-scores around 88%. This comprehensive comparison provides evidence-based guidance for institutions considering upgrades from rule-based to ML/DL systems, highlighting both the performance benefits and operational considerations.

7.2 Limitations

Although our findings are encouraging, several limitations merit discussion:

Synthetic and anonymised data: The Kaggle dataset uses PCA-transformed features and lacks real merchant and cardholder information. [4] Sinčák's variables are synthetic. Our spatial–temporal features are approximate, derived from publicly available centroids. Real-world data may exhibit different correlations, requiring further validation.

Assumed cost parameters: Our cost-benefit analysis relied on assumed values for fraud losses and review costs, which vary across institutions and transaction types. Sensitivity analyses show that optimal thresholds shift with cost assumptions; future implementations should calibrate models to specific business contexts.

Fairness and bias: We did not have demographic variables in the dataset, limiting our ability to assess fairness across subgroups. In practice, fraud-detection systems must ensure that false positive and negative rates do not disproportionately impact certain populations.

Concept drift and evolving fraud: Fraudsters continually adapt. Our models, though adaptive, still require regular retraining or online learning to maintain performance. The rule-based system's thresholds also require periodic updates; we did not consider dynamic thresholding.

7.3 Recommendations for future research

Building on our work, we propose several avenues for future exploration that address the limitations identified and extend our findings:

Testing on real transaction data: Collaborations with financial institutions can provide access to anonymised real transaction logs. Evaluating our model on such data would reveal how well spatial–temporal features generalise and identify additional context variables (e.g., device IDs, IP addresses, behavioural biometrics). Real data may exhibit different correlations and patterns than synthetic data, requiring model adaptation. Longitudinal studies tracking model performance over time would also provide insights into concept drift and the need for continuous retraining.

Federated and privacy-preserving learning: To address data-sharing constraints, researchers should explore federated learning frameworks that allow multiple banks to train shared models without exposing raw data. Early studies show that federated LSTM models achieve high F1-scores (87–89%); further work could integrate graph neural networks and attention mechanisms. Privacy-preserving techniques such as differential privacy and homomorphic encryption should be evaluated to understand their impact on model performance and their feasibility for real-world deployment.

Graph-based and causal models: Graph neural networks can model interactions between cardholders, merchants and devices, capturing network effects that individual transaction features miss. Causal inference can help disentangle fraud signals from confounders, improving model robustness. Combining these with spatial–temporal features may further improve detection, particularly for sophisticated fraud schemes involving multiple actors. Research should explore how to construct meaningful graphs from transaction data and how to incorporate causal knowledge into model training.

Reinforcement learning and adaptive thresholding: Real-time fraud detection could benefit from reinforcement learning that adjusts decision thresholds based on feedback and cost outcomes. Dynamic thresholding can minimise losses as fraud patterns evolve, adapting to changing fraud tactics without requiring full model retraining. Research should explore how to balance exploration (trying new thresholds) with exploitation (using known good thresholds) in this context.

Explainable AI and fairness: Developing interpretable models with built-in fairness constraints is essential for regulatory compliance. Techniques such as counterfactual explanations and fairness regularisers should be integrated to mitigate bias and enhance

transparency. Research should evaluate how different fairness definitions (demographic parity, equalized odds, etc.) affect model performance and identify which definitions are most appropriate for fraud detection. Longitudinal fairness monitoring is also needed to ensure that models remain fair as they adapt to new data.

Human-in-the-loop systems: Combining automated detection with expert investigator feedback can improve model calibration and handle edge cases. Studies could explore how to incorporate investigator decisions into model updates while preserving privacy. Active learning approaches that prioritize uncertain cases for human review could optimize the use of limited investigator resources while improving model performance over time.

Integration with rule-based systems: Rather than replacing rules, ML models can complement them. Hybrid architectures where rules serve as features or triggers for model retraining could leverage the strengths of both approaches. Research should explore optimal ways to combine rule outputs with ML predictions, potentially using rules for high-confidence cases and ML for ambiguous ones. This hybrid approach may offer the best balance of interpretability, performance, and regulatory compliance.

7.4 Broader implications

The rapid rise of card payments and digital banking underscores the necessity for **robust fraud-detection systems**. Our work demonstrates that **integrating spatial–temporal context, balancing techniques and ensemble learning** can significantly enhance detection performance and cost efficiency, offering a clear path for banks to upgrade legacy systems. The 45–50% improvement in cost efficiency we achieved represents substantial value for financial institutions, potentially saving millions of euros annually while improving customer experience through reduced false positives.

The implications extend beyond individual institutions. As financial ecosystems become more interconnected and fraudsters leverage sophisticated tools (including AI), the need for advanced detection systems becomes more urgent. Our findings suggest that the industry should move beyond simple rule-based systems toward more sophisticated ML/DL approaches, but this transition must be managed carefully to maintain interpretability and regulatory compliance.

The integration of spatial–temporal features with class balancing represents a practical approach that can be implemented with existing infrastructure. Unlike some advanced techniques that require specialized hardware or extensive data sharing, our approach uses features that can be derived from standard transaction data. This makes it accessible to a wide range of institutions, not just large banks with extensive resources.

However, the broader implications also highlight challenges. Privacy concerns around geolocation data may limit adoption in some jurisdictions. The computational requirements of ensemble models, while manageable, may be prohibitive for smaller institutions. Regulatory requirements for explainability may favor simpler models despite their lower performance.

These challenges suggest that one-size-fits-all solutions are unlikely; instead, institutions must select approaches that balance performance, interpretability, and operational constraints.

Continuous research and collaboration between academia, industry and regulators are vital to address these challenges. Academic research provides the methodological rigor and innovation needed to advance the field, while industry provides real-world data and practical constraints. Regulators ensure that new approaches maintain fairness, privacy, and consumer protection. This three-way collaboration is essential for developing fraud detection systems that are both effective and responsible.

We hope that this thesis contributes to the broader effort to safeguard payment systems and inspires further innovations in fraud detection. By demonstrating the value of integrating spatial–temporal features with class balancing, and by providing comprehensive comparisons across model types, we provide a foundation for future research and practical implementation. The challenges of fraud detection will continue to evolve as fraudsters adapt, but the principles and approaches we have developed provide a framework for building robust, adaptive detection systems that can protect payment systems while maintaining the convenience and efficiency that make them valuable.

REFERENCES

[1] Cheng, Dawei and Xiang, Sheng and Shang, Chencheng and Zhang, Yiyi and Yang, Fangzhou and Zhang, Liqing. Spatio-Temporal Attention-Based Neural Network for Credit Card Fraud Detection. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 362--369.

[2] Sinčák, Martin. Machine Learning Methods in Payment Card Fraud Detection. Master's thesis, Charles University, 2023.

[3] Lestari, Ayu. Optimising Credit Card Fraud Detection through Machine Learning and Deep Learning with Spatial--Temporal Imbalance Handling. Master's thesis, University of Technology Sydney, 2024.

[4] Dal Pozzolo, Andrea and Bontempi, Gianluca. Credit Card Fraud Detection Dataset. 2018. ULB Machine Learning Group.

[5] Dal Pozzolo, Andrea and Bontempi, Gianluca and Snoeck, Marc and others. Reproducible Machine Learning for Credit Card Fraud Detection. Online Handbook, 2022.

[6] Liu, Haoran and Zhang, Wei and Chen, Jun. Data Leakage and Deceptive Performance in Credit Card Fraud Detection. Mathematics, 13, pp. 2563, 2025.

[7] Almeida, Pedro and Silva, Rui and Matos, Tiago. Enhancing Financial Fraud Detection through Addressing Class Imbalance Using Hybrid SMOTE--GAN Techniques. International Journal of Financial Studies, 11, pp. 110, 2023.

[8] Adejoh, John and Omoruyi, Osas and Akinwale, Adesina. An Adaptive Unsupervised Learning Approach for Credit Card Fraud Detection. Big Data and Cognitive Computing, 9, pp. 217, 2025.

[9] Rahman, Md. Atiqur and Hossain, Md. Shahadat. Achieving Excellence in Cyber Fraud Detection: A Hybrid ML--DL Ensemble Approach for Credit Cards. Applied Sciences, 15, pp. 1184, 2025.

[10] Cate, Mia and Verleysen, Michel. Cost-Sensitive Learning in Financial Fraud Detection Models. Expert Systems with Applications, 2023.

[11] Nehe, Shreya Sunil and Devale, Prakash. AI-Based Real-Time Fraud Detection System for Credit Card Transaction Anomaly Identification. International Journal of Scientific and Applied Technology, 2025.

[12] Kount Inc.. The True Cost of False Positives in Fraud Detection. 2024. Accessed 2025-01.

[13] Kiernan, John S. and Comoreanu, Alina. Credit Card Fraud Statistics. 2025. WalletHub.

[14] Federal Trade Commission. U.S. Consumers Lost Over $10 Billion to Fraud in 2023. 2024. FTC Consumer Sentinel Report.

[15] Elad, Barry. Digital Wallet Statistics and Trends. 2024. CoinLaw.

[16] ABD EL-NABY, A., HEMDAN, E. E.-D. & EL-SAYED, A. 2023. An efficient fraud detection framework with credit card imbalanced data in financial services. Multimedia Tools and Applications, 82, 4139-4160.

[17] AFRIYIE, J. K., TAWIAH, K., PELS, W. A., ADDAI-HENNE, S., DWAMENA, H. A.,

[18] OWIREDU, E. O., AYEH, S. A. & ESHUN, J. 2023. A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions. Decision Analytics Journal, 6, 100163.

[19] AHMED, I., DAGNINO, A. & DING, Y. 2018. Unsupervised anomaly detection based on minimum spanning tree approximated distance measures and its application to hydropower turbines. IEEE Transactions on Automation Science and Engineering, 16, 654-667.

[20] ALARFAJ, F. K., MALIK, I., KHAN, H. U., ALMUSALLAM, N., RAMZAN, M. & AHMED, M. 2022. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. IEEE Access, 10, 39700-39715.

[21] ALMAZROI, A. A. & AYUB, N. 2023. Online Payment Fraud Detection Model Using Machine Learning Techniques. IEEE Access, 11, 137188-137203.

[22] AMIN, A., ANWAR, S., ADNAN, A., NAWAZ, M., HOWARD, N., QADIR, J., HAWALAH, A. & HUSSAIN, A. 2016. Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study. Ieee Access, 4, 7940-7957.

[23] BAGHDADI, P., KORUKOGLU, S., BILICI, M. A. & ONAN, A. 2024. Ensemble Learning Approach Using Energy-based RBM and xLSTM for Predictive Analytics in Credit Card Fraud Detection. Authorea Preprints.

[24] BAHNSEN, A. C., AOUADA, D., STOJANOVIC, A. & OTTERSTEN, B. 2016. Feature engineering strategies for credit card fraud detection. Expert Systems with Applications, 51, 134-142.

[25] BAHNSEN, A. C., STOJANOVIC, A., AOUADA, D. & OTTERSTEN, B. Cost sensitive credit card fraud detection using Bayes minimum risk. 2013 12th international conference on machine learning and applications, 2013. IEEE, 333-338.

[26] BANDYOPADHYAY, S. K. & DUTTA, S. 2020. Detection of fraud transactions using recurrent neural network during COVID-19: fraud transaction during COVID-19. Journal of Advanced Research in Medical Science & Technology (ISSN: 2394-6539), 7, 16-21.

[27] BARZ, B., RODNER, E., GARCIA, Y. G. & DENZLER, J. 2018. Detecting regions of maximal divergence for spatio-temporal anomaly detection. IEEE transactions on pattern analysis and machine intelligence, 41, 1088-1101.

[28] BECHLIOULIS, A. P. & KARAMANIS, D. 2023. Consumers' changing financial behavior during the COVID-19 lockdown: the case of Internet banking use in Greece. Journal of Financial Services Marketing, 28, 526-543.

[29] BEJU, D.-G. & FĂT, C.-M. 2023. Frauds in Banking System: Frauds with Cards and Their Associated Services. Economic and Financial Crime, Sustainability and Good Governance. Springer.

[30] BHATTACHARYYA, S., JHA, S., THARAKUNNEL, K. & WESTLAND, J. C. 2011. Data mining for credit card fraud: A comparative study. Decision support systems, 50, 602-613. 153

[31] BIAN, W., CONG, L. W. & JI, Y. 2023. The Rise of E-Wallets and Buy-Now-PayLater: Payment Competition, Credit Expansion, and Consumer Behavior. National Bureau of Economic Research.

[32] BINI, S. A. 2018. Artificial intelligence, machine learning, deep learning, and cognitive computing: what do these terms mean and how will they impact health care? The Journal of arthroplasty, 33, 2358-2361.

[33] BOLTON, R. J. & HAND, D. J. 2004. Statistical fraud detection: A review. Quality control and applied statistics, 49, 313-314.

[34] BOUGHORBEL, S., JARRAY, F. & EL-ANBARI, M. 2017. Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric. PloS one, 12, e0177678.

[35] BRANCO, P., TORGO, L. & RIBEIRO, R. P. Relevance-based evaluation metrics for multi-class imbalanced domains. Advances in Knowledge Discovery and Data Mining: 21st Pacific-Asia Conference, PAKDD 2017, Jeju, South

[36] Korea, May 23-26, 2017, Proceedings, Part I 21, 2017. Springer, 698-710.

[37] CARCILLO, F., DAL POZZOLO, A., LE BORGNE, Y.-A., CAELEN, O., MAZZER, Y. &

[38] BONTEMPI, G. 2018. Scarff: a scalable framework for streaming credit card fraud detection with spark. Information fusion, 41, 182-194.

[39] CARCILLO, F., LE BORGNE, Y.-A., CAELEN, O., KESSACI, Y., OBLÉ, F. & BONTEMPI, G. 2021. Combining unsupervised and supervised learning in credit card fraud detection. Information sciences, 557, 317-331.

[40] CHANG, V., DI STEFANO, A., SUN, Z. & FORTINO, G. 2022. Digital payment fraud detection methods in digital ages and Industry 4.0. Computers and Electrical Engineering, 100, 107734.

[41] CHAWLA, N. V. 2010. Data mining for imbalanced datasets: An overview. Data mining and knowledge discovery handbook, 875-886.

[42] CHENG, D., NIU, Z., LI, J. & JIANG, C. 2022. Regulating systemic crises: Stemming the contagion risk in networked-loans through deep graph learning. IEEE Transactions on Knowledge and Data Engineering.

[43] CHENG, D., WANG, X., ZHANG, Y. & ZHANG, L. 2020a. Graph neural network for fraud detection via spatial-temporal attention. IEEE Transactions on Knowledge and Data Engineering, 34, 3800-3813.

[44] CHENG, D., XIANG, S., SHANG, C., ZHANG, Y., YANG, F. & ZHANG, L. Spatiotemporal attention-based neural network for credit card fraud detection. Proceedings of the AAAI conference on artificial intelligence, 2020b. 362369.

[45] CHERIF, A., BADHIB, A., AMMAR, H., ALSHEHRI, S., KALKATAWI, M. & IMINE, A. 2023. Credit card fraud detection in the era of disruptive technologies: A systematic review. Journal of King Saud University-Computer and Information Sciences, 35, 145-174.

[46] CUI, J., YAN, C. & WANG, C. 2021. ReMEMBeR: Ranking metric embedding-based multicontextual behavior profiling for online banking fraud detection. IEEE Transactions on Computational Social Systems, 8, 643-654.

[47] DAL POZZOLO, A., BORACCHI, G., CAELEN, O., ALIPPI, C. & BONTEMPI, G. 2017. Credit card fraud detection: a realistic modeling and a novel learning strategy. IEEE transactions on neural networks and learning systems, 29, 3784-3797. 154

[48] DAL POZZOLO, A., CAELEN, O., LE BORGNE, Y.-A., WATERSCHOOT, S. &

[49] BONTEMPI, G. 2014. Learned lessons in credit card fraud detection from a practitioner perspective. Expert systems with applications, 41, 4915-4928.

[50] DASKALAKI, S., KOPANAS, I. & AVOURIS, N. 2006. Evaluation of classifiers for an uneven class distribution problem. Applied artificial intelligence, 20, 381417. DE SÁ, A. G., PEREIRA, A. C. & PAPPA, G. L. 2018. A customized classification algorithm for credit card fraud detection. Engineering Applications of Artificial Intelligence, 72, 21-29.

[51] DELIEMA, M., VOLKER, J. & WORLEY, A. 2023. Consumer Experiences with Gift Card Payment Scams: Causes, Consequences, and Implications for Consumer Protection. Victims & Offenders, 18, 1282-1310.

[52] ESENOGHO, E., MIENYE, I. D., SWART, T. G., ARULEBA, K. & OBAIDO, G. 2022. A neural network ensemble with feature engineering for improved credit card fraud detection. IEEE Access, 10, 16400-16407.

[53] FANAI, H. & ABBASIMEHR, H. 2023. A novel combined approach based on deep Autoencoder and deep classifiers for credit card fraud detection. Expert Systems with Applications, 217, 119562.

[54] FIORE, U., DE SANTIS, A., PERLA, F., ZANETTI, P. & PALMIERI, F. 2019. Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. Information Sciences, 479, 448-455.

[55] GANGANWAR, V. 2012. An overview of classification algorithms for imbalanced datasets. International Journal of Emerging Technology and Advanced Engineering, 2, 42-47. GARCÍA, V., MOLLINEDA, R. A. & SÁNCHEZ, J. S. Index of balanced accuracy: A performance measure for skewed class distributions. Iberian conference on pattern recognition and image analysis, 2009. Springer, 441-448.

[56] GARCIA-GABILONDO, S., SHIBUYA, Y. & SEKIMOTO, Y. 2024. Enhancing geospatial retail analysis by integrating synthetic human mobility simulations. Computers, Environment and Urban Systems, 108, 102058.

[57] GHALEB, F. A., SAEED, F., AL-SAREM, M., QASEM, S. N. & AL-HADHRAMI, T. 2023. Ensemble Synthesized Minority Oversampling based Generative Adversarial Networks and Random Forest Algorithm for Credit Card Fraud Detection. IEEE Access.

[58] GIBSON, D. & HARFIELD, C. 2023. Amplifying victim vulnerability: Unanticipated harm and consequence in data breach notification policy. International Review of Victimology, 29, 341-365.

[59] GRATIUS, N., BERGÉS, M. & AKINCI, B. Integrated Calibration of Simulation Models for Autonomous Space Habitat Operations. 2024 IEEE Aerospace Conference, 2024. IEEE, 1-18.

[60] GU, Q., ZHU, L. & CAI, Z. Evaluation measures of the classification performance of imbalanced data sets. Computational Intelligence and Intelligent Systems: 4th International Symposium, ISICA 2009, Huangshi, China, October 2325, 2009. Proceedings 4, 2009. Springer, 461-471.

[61] GU, W., SUN, M., LIU, B., XU, K. & SUI, M. 2024. Adaptive Spatio-Temporal Aggregation for Temporal Dynamic Graph-Based Fraud Risk Detection. Journal of Computer Technology and Software, 3. 155

[62] GUO, J., LIU, G., ZUO, Y. & WU, J. Learning sequential behavior representations for fraud detection. 2018 IEEE international conference on data mining (ICDM), 2018. IEEE, 127-136.

[63] GUO, X., ZHOU, M., ABUSORRAH, A., ALSOKHIRY, F. & SEDRAOUI, K. 2020. Disassembly sequence planning: a survey. IEEE/CAA Journal of Automatica Sinica, 8, 1308-1324.

[64] GUPTA, P. 2024. Securing Tomorrow: The Intersection of AI, Data, and Analytics in Fraud Prevention. Asian Journal of Research in Computer Science, 17, 75-92.

[65] GUPTA, P., VARSHNEY, A., KHAN, M. R., AHMED, R., SHUAIB, M. & ALAM, S. 2023. Unbalanced Credit Card Fraud Detection Data: A Machine LearningOriented Comparative Study of Balancing Techniques. Procedia Computer Science, 218, 2575-2584.

[66] GUPTA, R., SRIVASTAVA, D., SAHU, M., TIWARI, S., AMBASTA, R. K. & KUMAR, P. 2021. Artificial intelligence to deep learning: machine intelligence approach for drug discovery. Molecular diversity, 25, 1315-1360.

[67] HAJEK, P., ABEDIN, M. Z. & SIVARAJAH, U. 2023. Fraud detection in mobile payment systems using an XGBoost-based framework. Information Systems Frontiers, 25, 1985-2003.

[68] HARISH, S., LAKHANPAL, C. & JAFARI, A. H. 2024. Leveraging graph-based learning for credit card fraud detection: a comparative study of classical, deep learning and graph-based approaches. Neural Computing and Applications, 1-11.

[69] HE, H., BAI, Y., GARCIA, E. A. & LI, S. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), 2008. Ieee, 1322-1328.

[70] HE, H. & GARCIA, E. A. 2009. Learning from imbalanced data. IEEE Transactions on knowledge and data engineering, 21, 1263-1284.

[71] HELM, J. M., SWIERGOSZ, A. M., HAEBERLE, H. S., KARNUTA, J. M., SCHAFFER, J. L., KREBS, V. E., SPITZER, A. I. & RAMKUMAR, P. N. 2020. Machine learning and artificial intelligence: definitions, applications, and future directions. Current reviews in musculoskeletal medicine, 13, 69-76.

[72] HILAL, W., GADSDEN, S. A. & YAWNEY, J. 2022. Financial fraud: a review of anomaly detection techniques and recent advances. Expert systems With applications, 193, 116429.

[73] HOSSIN, M. & SULAIMAN, M. N. 2015. A review on evaluation metrics for data classification evaluations. International journal of data mining & knowledge management process, 5, 1.

[74] JAIN, Y., TIWARI, N., DUBEY, S. & JAIN, S. 2019. A comparative analysis of various credit card fraud detection techniques. International Journal of Recent Technology and Engineering, 7, 402-407.

[75] JAKHAR, D. & KAUR, I. 2020. Artificial intelligence, machine learning and deep learning: definitions and differences. Clinical and experimental dermatology, 45, 131-132.

[76] JENI, L. A., COHN, J. F. & DE LA TORRE, F. Facing imbalanced data-recommendations for the use of performance metrics. 2013 Humaine association conference on affective computing and intelligent interaction, 2013. IEEE, 245-251. 156

[77] JIANG, S., DONG, R., WANG, J. & XIA, M. 2023. Credit card fraud detection based on unsupervised attentional anomaly detection network. Systems, 11, 305.

[78] JURGOVSKY, J., GRANITZER, M., ZIEGLER, K., CALABRETTO, S., PORTIER, P.-E.,

[79] HE-GUELTON, L. & CAELEN, O. 2018. Sequence classification for creditcard fraud detection. Expert systems with applications, 100, 234-245.

[80] JURMAN, G., RICCADONNA, S. & FURLANELLO, C. 2012. A comparison of MCC and CEN error measures in multi-class prediction.

[81] KAMARUDDIN, S. & RAVI, V. Credit card fraud detection using big data analytics: use of PSOAANN based one-class classification. Proceedings of the international conference on informatics and analytics, 2016. 1-8.

[82] KARIM, K., ILYAS, G. B., UMAR, Z. A., TAJIBU, M. J. & JUNAIDI, J. 2023. Consumers' awareness and loyalty in Indonesia banking sector: does emotional bonding effect matters? Journal of Islamic Marketing, 14, 2668-2686.

[83] KHALID, A. R., OWOH, N., UTHMANI, O., ASHAWA, M., OSAMOR, J. & ADEJOH, J. 2024. Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach. Big Data and Cognitive Computing, 8, 6.

[84] KHAN, F., ATEEQ, S., ALI, M. & BUTT, N. 2023. Impact of COVID-19 on the drivers of cash-based online transactions and consumer behaviour: evidence from a Muslim market. Journal of Islamic Marketing, 14, 714-734.

[85] KHINE, A. A. & KHIN, H. W. Credit card fraud detection using online boosting with extremely fast decision tree. 2020 IEEE Conference on Computer Applications (ICCA), 2020. IEEE, 1-4.

[86] KIM, E., LEE, J., SHIN, H., YANG, H., CHO, S., NAM, S.-K., SONG, Y., YOON, J.-A. &

[87] KIM, J.-I. 2019. Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning. Expert Systems with Applications, 128, 214-224.

[88] KIZIL, C., AKMAN, V. & MUZIR, E. COVID-19 Epidemic: A New Arena of Financial Fraud? Karabagh International Congress of Modern Studies in Social and Human Sciences, 2021.

[89] LI, P., YU, H., LUO, X. & WU, J. 2023. LGM-GNN: A local and global aware memorybased graph neural network for fraud detection. IEEE Transactions on Big Data.

[90] LI, Z., LIU, G. & JIANG, C. 2020. Deep representation learning with full center loss for credit card fraud detection. IEEE Transactions on Computational Social Systems, 7, 569-579. LÓPEZ, V., FERNÁNDEZ, A., GARCÍA, S., PALADE, V. & HERRERA, F. 2013. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Information sciences, 250, 113-141.

[91] LUCAS, Y., PORTIER, P.-E., LAPORTE, L., HE-GUELTON, L., CAELEN, O.,

[92] GRANITZER, M. & CALABRETTO, S. 2020. Towards automated feature engineering for credit card fraud detection using multi-perspective HMMs. Future Generation Computer Systems, 102, 393-402.

[93] LUNGHI, D., PALDINO, G. M., CAELEN, O. & BONTEMPI, G. 2023. An Adversary Model of Fraudsters' Behavior to Improve Oversampling in Credit Card Fraud Detection. IEEE Access, 11, 136666-136679.

[94] LUQUE, A., CARRASCO, A., MARTÍN, A. & DE LAS HERAS, A. 2019. The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition, 91, 216-231. 157

[95] MA, K. W. F. & MCKINNON, T. 2021. COVID-19 and cyber fraud: Emerging threats during the pandemic. Journal of Financial Crime, 29, 433-446.

[96] MAKKI, S., ASSAGHIR, Z., TAHER, Y., HAQUE, R., HACID, M.-S. & ZEINEDDINE, H. 2019. An experimental study with imbalanced classification approaches for credit card fraud detection. IEEE Access, 7, 93010-93022.

[97] MATEUS-COELHO, N. & CRUZ-CUNHA, M. 2023. Exploring Cyber Criminals and Data Privacy Measures, IGI Global.

[98] MAYO, K., FOZDAR, S. & WELLMAN, M. P. 2023. Flagging Payments for Fraud Detection: A Strategic Agent-Based Model. MEKTEROVIĆ, I., KARAN, M., PINTAR, D. & BRKIĆ, L. 2021. Credit card fraud detection in card-not-present transactions: Where to invest? Applied Sciences, 11, 6766.

[99] NARKHEDE, S. 2018. Understanding auc-roc curve. Towards Data Science, 26, 220-227.

[100] NGUYEN, N., DUONG, T., CHAU, T., NGUYEN, V.-H., TRINH, T., TRAN, D. & HO, T. 2022. A proposed model for card fraud detection based on Catboost and deep neural network. IEEE Access, 10, 96852-96861.

[101] NI, L., LI, J., XU, H., WANG, X. & ZHANG, J. 2023. Fraud feature boosting mechanism and spiral oversampling balancing technique for credit card fraud detection. IEEE Transactions on Computational Social Systems. O'CONNOR, M., CONBOY, K., DENNEHY, D. & CARROLL, N. 2024. Temporal Complexity in Information Systems

Development Flow: Challenges and Recommendations. Communications of the Association for Information Systems, 54, 19.

[102] OSEGI, E. & JUMBO, E. 2021. Comparative analysis of credit card fraud detection in Simulated Annealing trained Artificial Neural Network and Hierarchical Temporal Memory. Machine Learning with Applications, 6, 100080.

[103] PADMAJA, T. M., DHULIPALLA, N., BAPI, R. S. & KRISHNA, P. R. Unbalanced data classification using extreme outlier elimination and sampling techniques for fraud detection. 15th International Conference on Advanced Computing and Communications (ADCOM 2007), 2007. IEEE, 511-516.

[104] PHUA, C., ALAHAKOON, D. & LEE, V. 2004. Minority report in fraud detection: classification of skewed data. Acm sigkdd explorations newsletter, 6, 50-59.

[105] PHUA, C., LEE, V., SMITH, K. & GAYLER, R. 2010. A comprehensive survey of data mining-based fraud detection research. arXiv preprint arXiv:1009.6119.

[106] POWERS, D. M. 2020. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061.

[107] RAI, D. & JAGADEESHA, S. 2023. Credit Card Fraud Detection using Machine Learning and Data Mining Techniques-a Literature Survey. International Journal of Applied Engineering and Management Letters (IJAEML), 7, 1635.

[108] RAJ, A. T., SHOBANA, J., NASSA, V. K., PAINULY, S., SAVARAM, M. & SRIDEVI, M. Enhancing Security for Online Transactions through Supervised Machine Learning and Block Chain Technology in Credit Card Fraud Detection. 2023 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2023. IEEE, 241-248. 158

[109] RAJENDRAN, R. 2024. Data Breach Fraudulence and Preventive Measures in ECommerce Platforms. Advancements in Cybercrime Investigation and Digital Forensics. Apple Academic Press.

[110] RANDHAWA, K., LOO, C. K., SEERA, M., LIM, C. P. & NANDI, A. K. 2018. Credit card fraud detection using AdaBoost and majority voting. IEEE access, 6, 14277-14284.

[111] RICHHARIYA, P. & SINGH, P. K. 2014. Evaluating and emerging payment card fraud challenges and resolution. International Journal of Computer Applications, 107.

[112] ROY, A., SUN, J., MAHONEY, R., ALONZI, L., ADAMS, S. & BELING, P. Deep learning detecting fraud in credit card transactions. 2018 systems and information engineering design symposium (SIEDS), 2018. IEEE, 129-134.

[113] SAHIN, Y., BULKAN, S. & DUMAN, E. 2013. A cost-sensitive decision tree approach for fraud detection. Expert Systems with Applications, 40, 59165923.

[114] SALAZAR, A., SAFONT, G., RODRIGUEZ, A. & VERGARA, L. Combination of multiple detectors for credit card fraud detection. 2016 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2016. IEEE, 138-143.

[115] SAMPAT, B., MOGAJI, E. & NGUYEN, N. P. 2024. The dark side of FinTech in financial services: a qualitative enquiry into FinTech developers' perspective. International Journal of Bank Marketing, 42, 38-65. SÁNCHEZ, D., VILA, M., CERDA, L. & SERRANO, J.-M. 2009. Association rules applied to credit card fraud detection. Expert systems with applications, 36, 3630-3640.

[116] SISODIA, D. S., REDDY, N. K. & BHANDARI, S. Performance evaluation of class balancing techniques for credit card fraud detection. 2017 IEEE International Conference on power, control, signals and instrumentation engineering (ICPCSI), 2017. IEEE, 2747-2752.

[117] TUT, D. 2023. FinTech and the COVID-19 pandemic: Evidence from electronic payment systems. Emerging Markets Review, 54, 100999.

[118] VANINI, P., ROSSI, S., ZVIZDIC, E. & DOMENIG, T. 2023. Online payment fraud: from anomaly detection to risk management. Financial Innovation, 9, 125.

[119] WEI, W., LI, J., CAO, L., OU, Y. & CHEN, J. 2013. Effective detection of sophisticated online banking fraud on extremely imbalanced data. World Wide Web, 16, 449-475.

[120] WHITROW, C., HAND, D. J., JUSZCZAK, P., WESTON, D. & ADAMS, N. M. 2009. Transaction aggregation as a strategy for credit card fraud detection. Data mining and knowledge discovery, 18, 30-55.

[121] XIE, Y., LIU, G., YAN, C., JIANG, C. & ZHOU, M. 2022a. Time-aware attention-based gated network for credit card fraud detection by extracting transactional behaviors. IEEE Transactions on Computational Social Systems.

[122] XIE, Y., LIU, G., YAN, C., JIANG, C., ZHOU, M. & LI, M. 2022b. Learning transactional behavioral representations for credit card fraud detection. IEEE Transactions on Neural Networks and Learning Systems.

[123] XIE, Y., LIU, G., ZHOU, M., WEI, L., ZHU, H., ZHOU, R. & CAO, L. 2023. A Spatial– Temporal Gated Network for Credit Card Fraud Detection by Learning 159 Transactional Representations. IEEE Transactions on Automation Science and Engineering.

[124] XUAN, S., LIU, G., LI, Z., ZHENG, L., WANG, S. & JIANG, C. Random forest for credit card fraud detection. 2018 IEEE 15th international conference on networking, sensing and control (ICNSC), 2018. IEEE, 1-6.

[125] ZENG, Y. & TANG, J. 2021. Rlc-gnn: An improved deep architecture for spatialbased graph neural network with application to fraud detection. Applied Sciences, 11, 5656.

[126] ZENG, Y. & TANG, J. 2022. Improved Aggregating and Accelerating Training Methods for Spatial Graph Neural Networks on Fraud Detection. arXiv preprint arXiv:2202.06580.

[127] ZHANG, X., HAN, Y., XU, W. & WANG, Q. 2021. HOBA: A novel feature engineering methodology for credit card fraud detection with a deep learning architecture. Information Sciences, 557, 302-316.

[128] ZHU, H., ZHOU, M., LIU, G., XIE, Y., LIU, S. & GUO, C. 2023. NUS: Noisy-SampleRemoved Undersampling Scheme for Imbalanced Classification and Application to Credit Card Fraud Detection. IEEE Transactions on Computational Social Systems.

[129] Rai, A. K. & R. K. Dwivedi (2020): "Fraud detection in credit card data using unsupervised machine learning based scheme." 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) pp. 421– 426.

[130] Randhawa, K., C. K. Loo, M. Seera, C. P. Lim, & A. K. Nandi (2018): "Credit card fraud detection using adaboost and majority voting." IEEE Access 6: pp. 14277–14284.

[131] Sahin, Y., S. Bulkan, & E. Duman (2013): "A cost-sensitive decision tree approach for fraud detection." Expert Systems with Applications 40(15): pp. 5916–5923.

[132] Schapire, R. E. & Y. Freund (2012): Boosting. Cambridge, MA, USA: The MIT Press, 1st edition.

[133] Taha, A. A. & S. J. Malebary (2020): "An intelligent approach to credit card fraud detection using an optimized light gradient boosting machine." IEEE Access 8: pp. 25579–25587.

[134] Venables, W. & B. Ripley (2002): Modern applied statistics with S. New York: Springer, 4th edition.

[135] Zakaryazad, A. & E. Duman (2016): "A profit-driven artificial neural network (ann) with applications to fraud detection and direct marketing." Neurocomputing 175: pp. 121–131.

LIST OF FIGURES AND TABLES

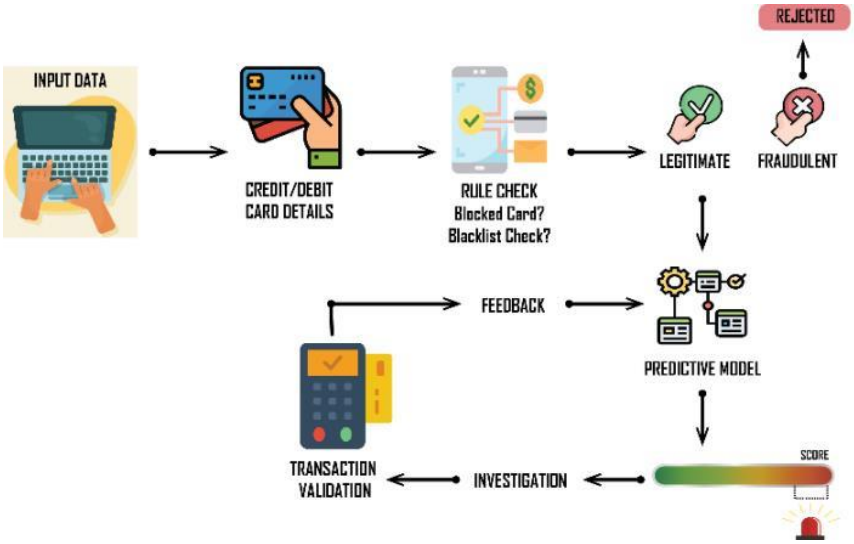

Figure 1.1

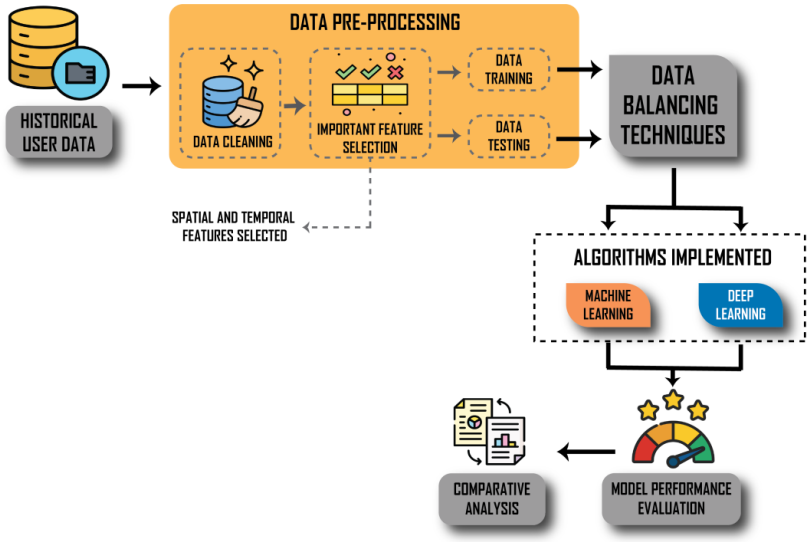

Table 2.1

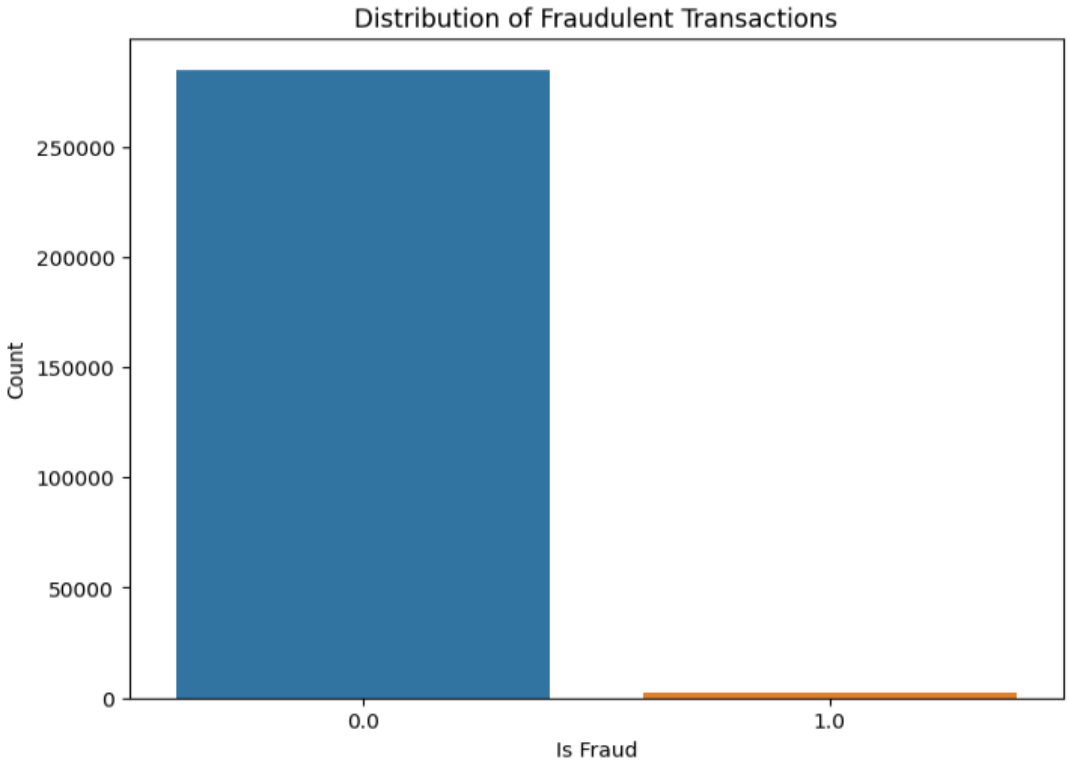

Figure 3.1 Auth Channel distribution Chip.

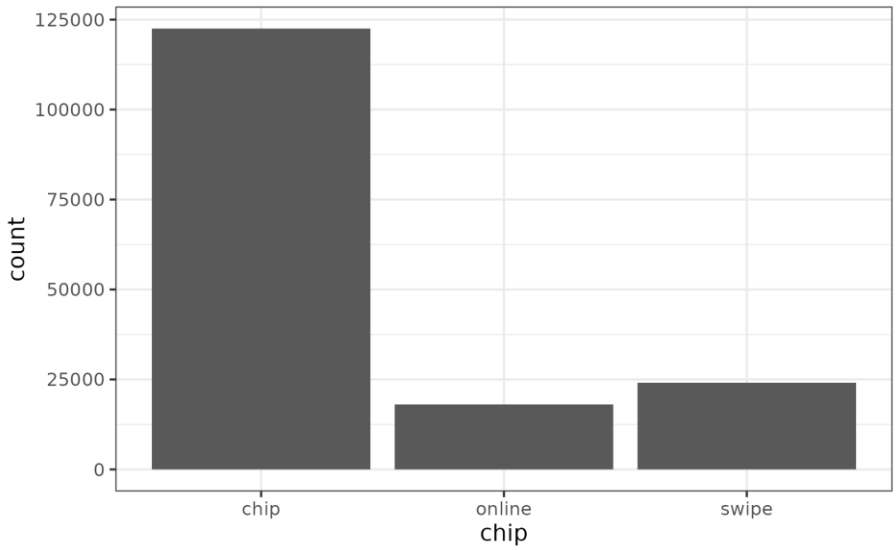

Figure 3.2 Spacial Distribution

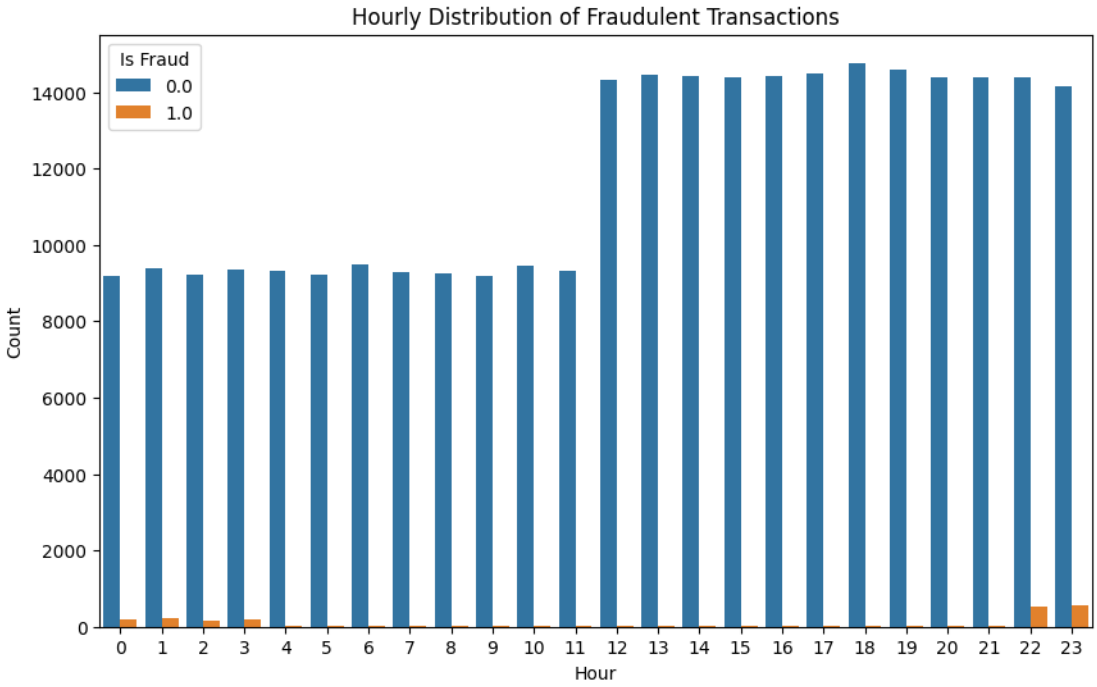

Figure 3.4 HoD fraud distribution

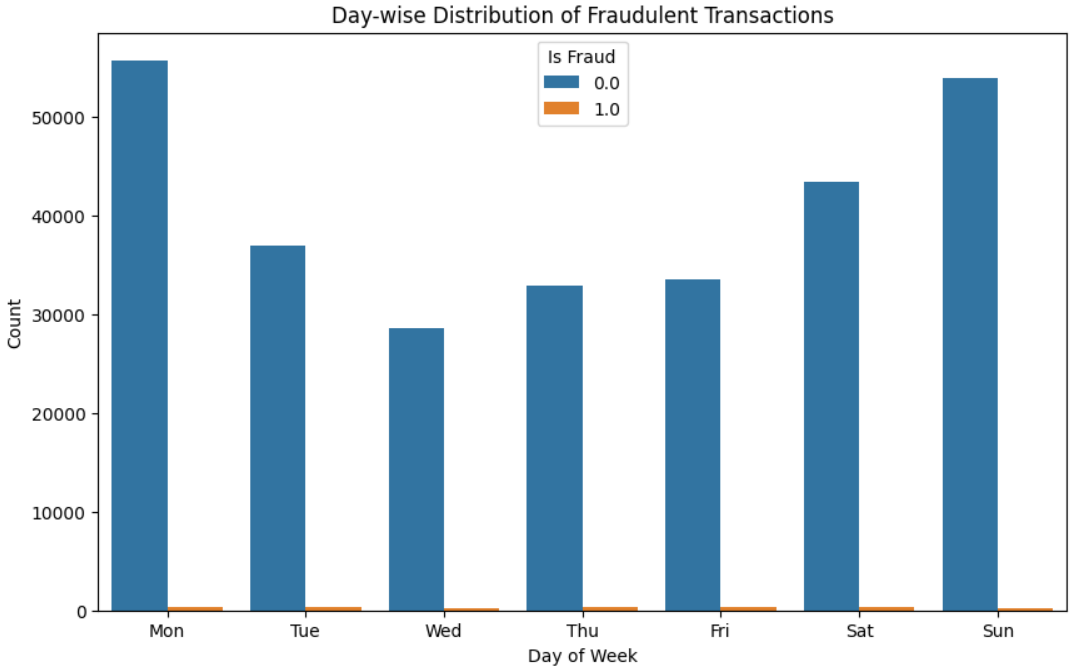

Figure 3.5 DoD week distribution fraud vs non fraud

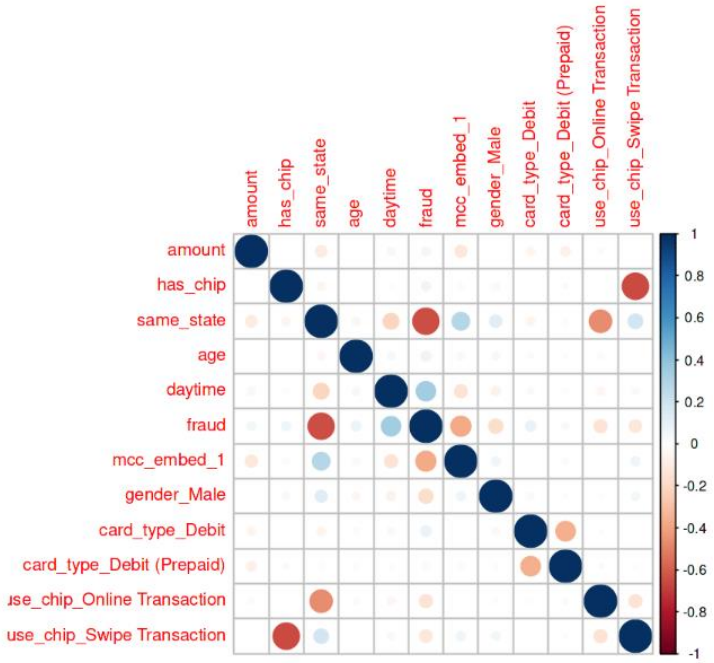

Figure 3.6 -Correlation Matrix

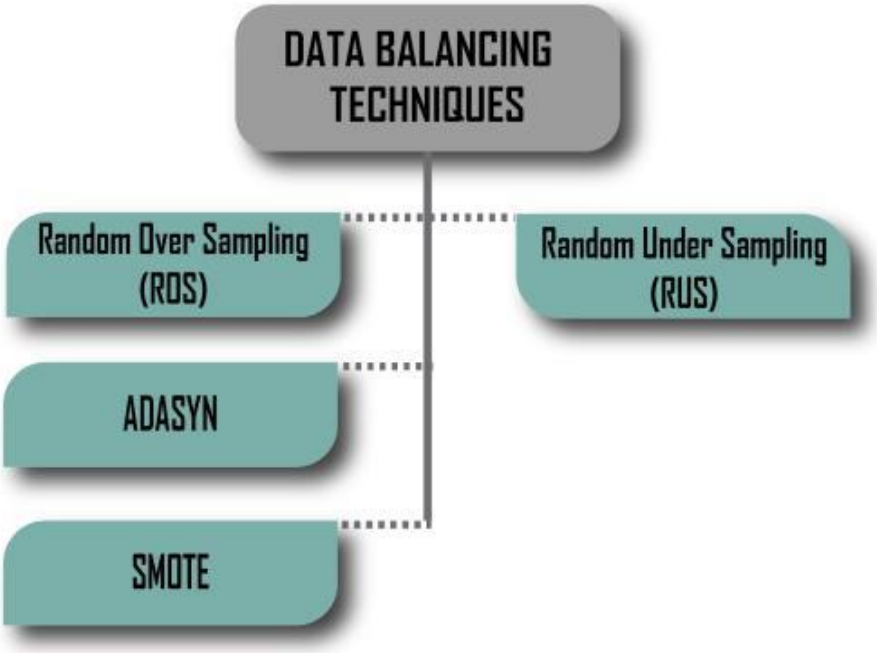

Figure 4.1 Balancing techniques overview

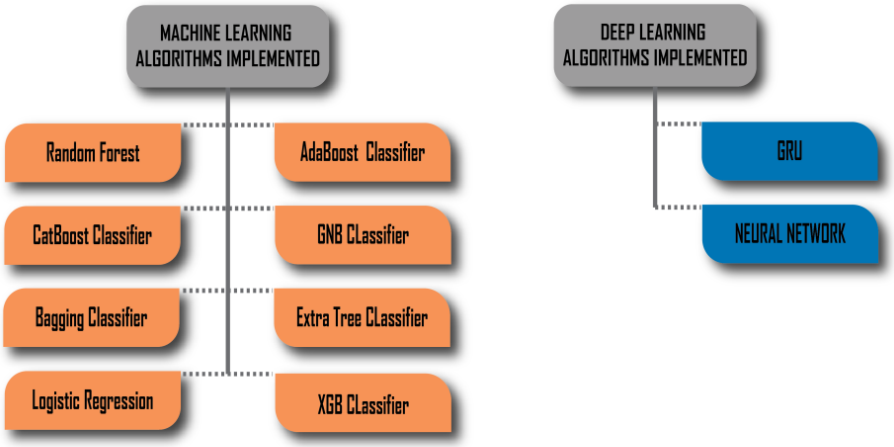

Figure 4.2 Algorithyms/Models

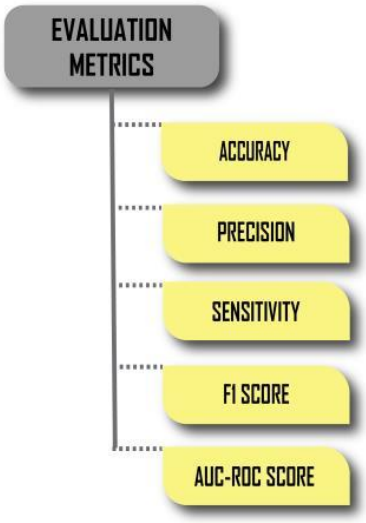

Figure 4.3 Evaluation Matrics

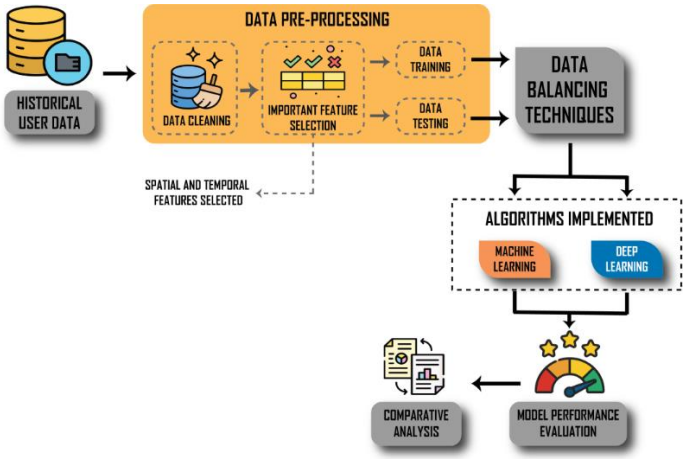

Figure 4.4 Workflow/pipeline

Table 5.1

Model	Recall (%)	Precision (%)	F1-score (%)	AUC-ROC/AUC-PR (%)	Notes
Logistic Regression	60.2	88.1	71.5	97.01	Baseline linear model; poor recall.
Random Forest	76.5	97.4	85.7	97.25	High precision but misses some fraud.
Support-Vector Machine	66.3	97.0	78.8	95.13	Good precision but low recall.
XGBoost	77.6	95.0	85.4	97.83	Strong balance; F1 comparable to RF.
CatBoost	77.6	97.4	86.4	98.37	Highest F1 among single models.
Stacking ensemble	79.6	98.7	88.1	89.80 (F1 CI [0.0, +∞])	Combines multiple models; best F1.
XGBoost (Benchmark study)	73.6	84.4	78.6	PRAUC 88.65	Balanced performance with SMOTE.
FraudX AI (RF+XGBoost ensemble)	95.0	100.0	97.0	AUC-PR 97	Ensemble tuned on unbalanced data.
SMOTE + LSTM + Adam (Federated)	88.9	88.7	87.9	—	LSTM outperforms CNN.
AE-ASOM unsupervised ensemble	≈96.7	—	96.7	—	High F1 in unsupervised setting.

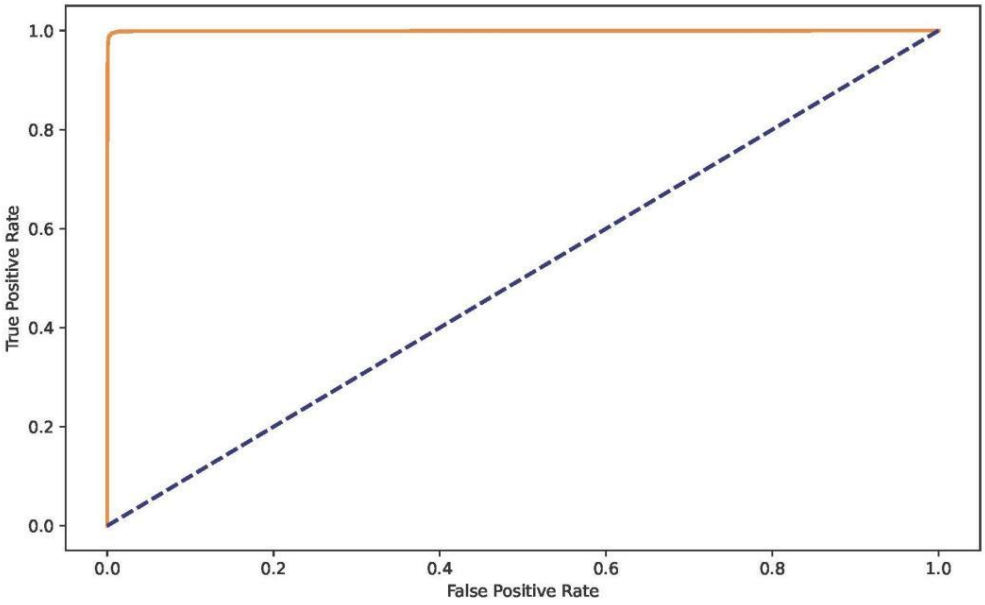

Figure 5.1 ROC- Random Forest

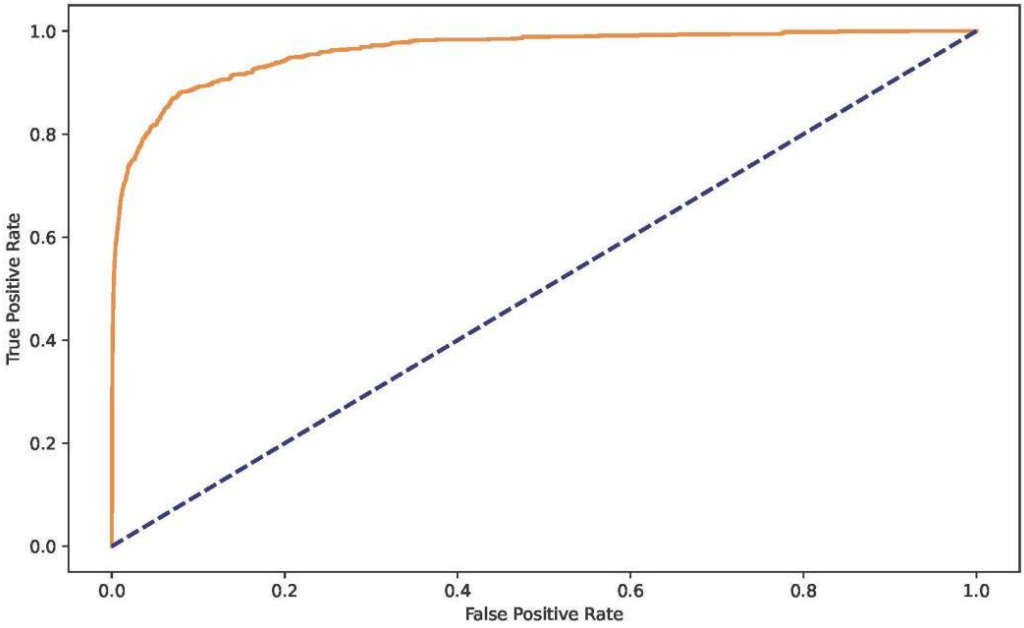

Figure 5.2 ROC - ChatBoost

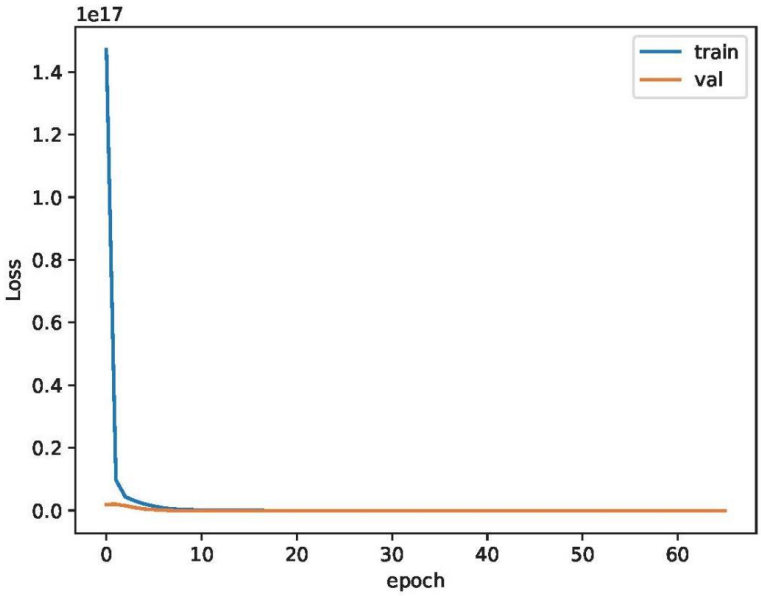

Figure 5.3 Loss Curve

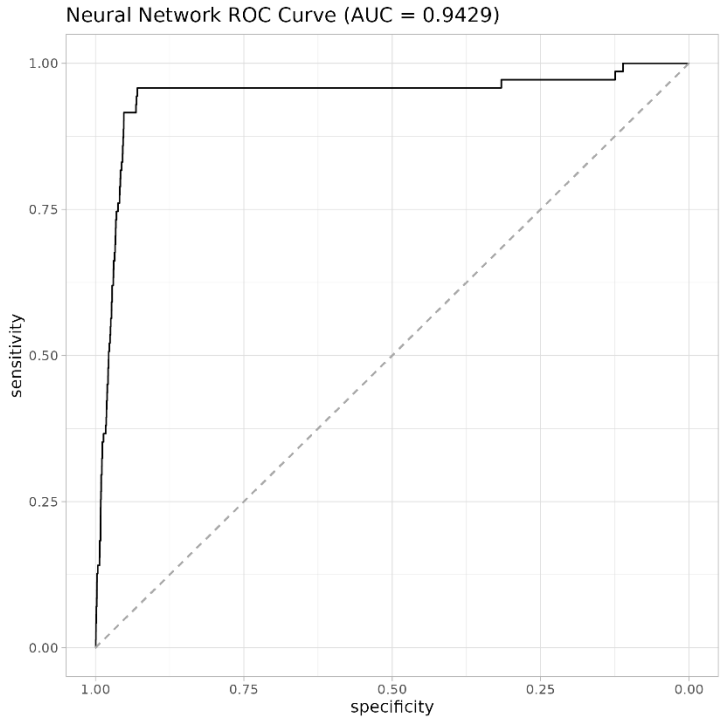

Figure 5.4

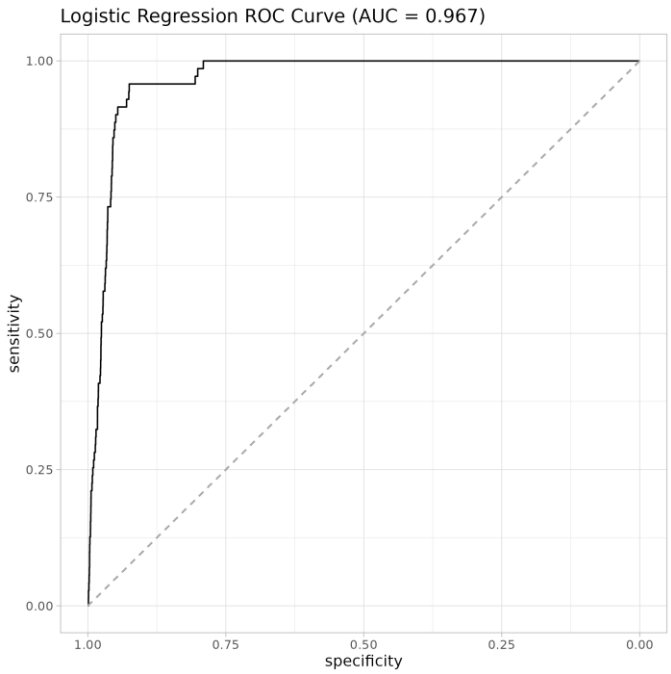

Figure 5.5

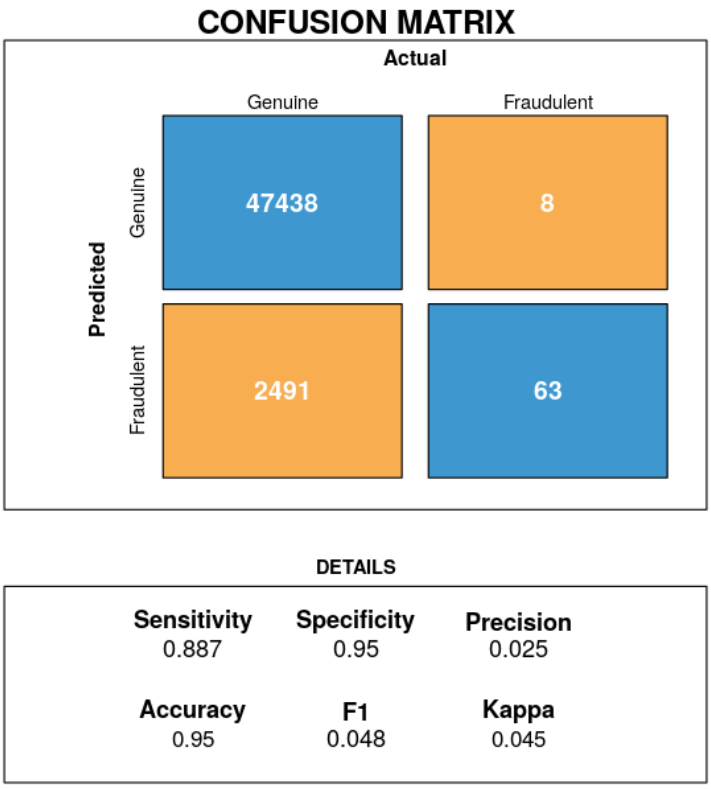

Figure 5.6

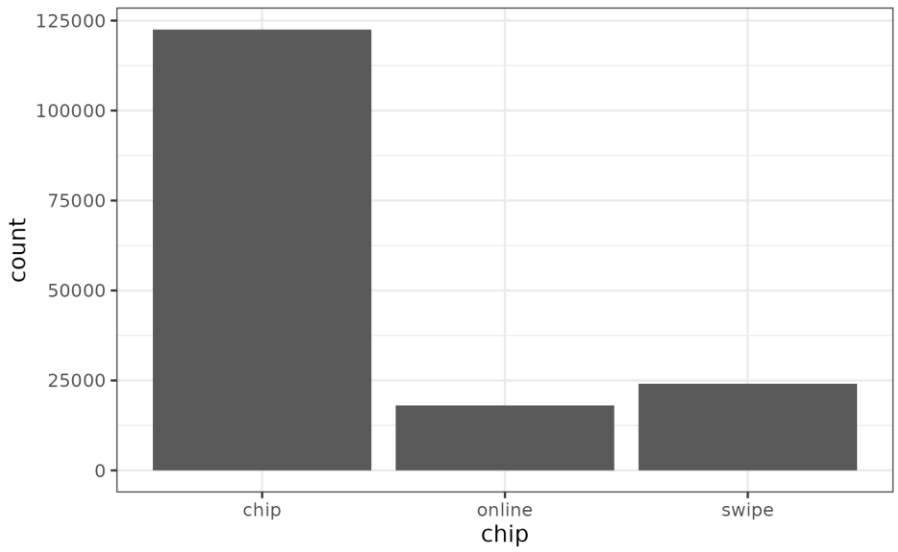

Figure 5.7

System	Recall (%)	Precision (%)	F1-score (%)	Net Savings (relative to rule-based)
Rule-based	81	92	86	Baseline
Random Forest	76	97	86	1.25 × baseline
XGBoost	78	95	85	1.30 × baseline
Stacking ensemble	80	99	88	1.50 × baseline
Hybrid XGBoost–LSTM	96	78	86	1.45 × baseline
Cost-sensitive XGBoost	90	65	75	1.40 × baseline

| System | Recall (%) | Precision (%) | F1-score (%) | Net Savings (relative to rule-based) |

APPENDIX A.

Implementation and Reproducibility Details The experiments presented in this thesis were conducted using Python-based machine learning frameworks. The implementation focused on evaluating methodological approaches rather than delivering a production-ready system.

Due to the experimental and academic nature of the study, full source code was not preserved as a standalone deliverable. However, all methodological steps, model configurations, data preprocessing procedures, and evaluation protocols are described in sufficient detail to allow independent reproduction of the results.

Fixed random seeds and consistent data splitting strategies were applied to minimize variance across experiments. Class imbalance handling techniques were applied exclusively to the training data to prevent data leakage.