NOVA IMS Information Management School

MGI

Master Degree Program in

Information Management

A Comparative Analysis of Imbalanced Learning Techniques for Optimizing Credit Card Fraud Detection

Rita Ávila da Silva

Master Thesis

presented as partial requirement for obtaining a Master's Degree in Information Management

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

A Comparative Analysis of Imbalanced Learning Techniques for Optimizing Credit Card Fraud Detection

by

Rita Ávila da Silva

Master Thesis presented as partial requirement for obtaining the Master's degree in Information Management, with a specialization in Business Intelligence

Supervised by

Roberto Henriques, PhD, NOVA IMS

April, 2025

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Lisbon, April 2025

ABSTRACT

Credit card fraud is a growing concern for financial institutions and consumers, leading to significant financial losses and increased security risks. One of the main challenges in fraud detection is the extreme class imbalance, where fraudulent transactions make up only a tiny fraction of all transactions. This imbalance makes it difficult for machine learning models to correctly identify fraud, as they tend to be biased toward the majority class. This paper explores and compares the implementation of various imbalanced learning techniques, including SMOTE, ROS, Borderline-SMOTE, ADASYN, K-means SMOTE, SMOTE-ENN, SMOTE-Tomek, CT-GAN, and CT-GAN Synthesizer. The goal is to assist in the selection of high-performance imbalanced learning techniques for fraud detection, ensuring its applicability and robustness across imbalanced fraud datasets. Empirical results of extensive experiments with 5 datasets show that traditional oversampling methods like ROS and SMOTE variants, consistently improved model performance when combined with strong classifiers like Random Forest and XGBoost. These methods not only increased recall, ensuring a higher detection rate of fraudulent transactions, but also maintained a favorable balance with precision, reducing the risk of flagging legitimate transactions as fraudulent. In contrast, more advanced techniques, including GAN's and K-means SMOTE, did not demonstrate the expected improvements. Instead, these methods occasionally introduced variability that did not translate into overall performance gains when compared to the traditional oversampling strategies.

KEYWORDS

Classification; Imbalanced Learning; Credit Card Fraud Detection; Sampling Techniques; Evaluation Metrics

Sustainable Development Goals (SDG):

TABLE OF CONTENTS

Statement of Integrity i
Abstract ii
List of Figures iv
List of Tables v
List of Abbreviations and Acronyms vi
1. Introduction 1
2. Literature review 6
2.1. Imbalanced Learning Approaches 6
2.1.1. Data-Level Approach 6
2.1.1.1. Oversampling 6
2.1.1.2. Undersampling 13
2.1.1.3. Hybrid Sampling 15
2.1.2. Algorithm-Level Approach 16
2.1.2.1. Cost Sensitive Approach 16
2.1.2.2. Ensemble Learning 16
2.1.3. Hybrid Approach 18
2.2. Imbalanced Learning Approaches for Credit Card Fraud Data 19
3. Methodology 30
3.1. Data Collection and Understanding 31
3.2. Data Preparation 32
3.2.1. Inconsistencies Fix 32
3.2.2. Missing Values 32
3.2.3. Data Encoding 32
3.2.4. Feature Selection 33
3.2.5. Splitting the Datasets 33
3.2.6. Data Normalization 33
3.3. Imbalanced Learning Techniques 34
3.4. Modelling 35
3.4.1. Classifiers 36
3.5. Evaluation 37
4. Results and discussion 39
5. Conclusions and future works 47
Bibliographical References 49
Appendix A 61

LIST OF FIGURES

Figure 1 – SMOTE linearly interpolates a randomly selected minority sample and one of its $k = 4$ nearest neighbors... 7
Figure 2 – Behavior of SMOTE in the presence of noise and within-class imbalance 8
Figure 3 – K-means SMOTE oversamples safe areas and combats within-class imbalance 9
Figure 4 – General block diagram of Generative Adversarial Networks..... 10
Figure 5 – Conditional Generative Adversarial Network.....11
Figure 6 – The CT-GAN Model.....12
Figure 7 – The Architecture of the Duo-GAN.....13
Figure 8 – Proposed Methodology.....30
Figure 9 – Confusion Matrix.....37
Figure 10 – Recall scores for Dataset 1.....39
Figure 11 – Recall scores for Dataset 2.....39
Figure 12 – Recall scores for Dataset 3.....40
Figure 13 – Recall scores for Dataset 4.....40
Figure 14 – Recall scores for Dataset 5.....40

LIST OF TABLES

Table 1 – Comparison of Imbalanced Learning Techniques for Credit Card Fraud Detection....26
Table 2 – Summary of the Datasets.....31
Table 3 – Hyperparameter Optimization for Imbalanced Learning Techniques.....34
Table 4 – Hyperparameter Optimization for Classifiers.....36
Table 5 – Comparison of the Best Imbalanced Techniques for Credit Card Fraud datasets.....45
Table 6 – Experimental Results.....61

LIST OF ABBREVIATIONS AND ACRONYMS

AdaBoost	Adaptive Boosting
ADASYN	Adaptive Synthetic Sampling
ANN	Artificial Neural Networks
Bagging	Bootstrap Aggregating
cGAN	Conditional Generative Adversarial Network
CNN	Convolution Neural Network
CRISP-DM	Cross-Industry Standard Process for Data Mining
CT-GAN	Conditional Tabular GAN
DA-SMOTE	Deep Adversarial SMOTE
DB-SMOTE	Density-based SMOTE
DT	Decision Tree
ENN	Edited Nearest Neighbor
GANs	Generative Adversarial Networks
GBDT	Gradient Boosted Decision Tree
GBM	Gradient Boosting Machines
G-Mean	Geometric Mean
IF	Isolation Forest
KNN	K-nearest Neighbor
LOF	Local Outlier Factor
LR	Logistic Regression
MCC	Matthews Correlation Coefficient
MLP	Multi-layer Perceptron
OSCNN	Oversampling Convolution Neural Network
OSE	Optimized Stacking Ensemble
PCA	Principal Components Analysis

PR-AUC	Area Under the Precision-Recall Curve
RF	Random Forest
ROC-AUC	Area Under the Receiver Operating Characteristic Curve
ROS	Random Oversampling
RUS	Random Undersampling
SMOTE	Synthetic Minority Oversampling Technique
SVM	Support Vector Machines
TGAN	Tabular Generative Adversarial Network
XGBoost	Extreme Gradient Boosting

1. INTRODUCTION

Credit card fraud is a significant and evolving threat to financial institutions and consumers worldwide, as the increasing volume of online transactions has led to more sophisticated fraudulent activities (Vanini, Rossi, Zvizdic, & Domenig, 2023). It refers to the unauthorized use of a credit card or its information to make purchases or withdraw money without the cardholder's consent. It typically involves fraudsters obtaining sensitive information like credit card numbers, CVV codes, or personal identification numbers (PINs) through various methods, such as phishing, data breaches, or card skimming devices. These details are then used to make fraudulent transactions, either by making direct purchases or selling the stolen data on the dark web, resulting in substantial financial losses and reputational damage to financial institutions (Alamri & Ykhlef, 2022).

In 2021 alone, global losses from credit card fraud amounted to $32.34 billion, with projections indicating that these losses could exceed $38.5 billion by 2027 (Nilson Report, 2022). According to a report published by Security.org (2024), 60% of U.S. credit cardholders have experienced fraud, and 45% have experienced fraud multiple times in 2023. The increasing volume of online transactions, driven by the expansion of e-commerce, has amplified the complexity and scale of fraudulent activities, posing substantial challenges for fraud detection systems (Vanini et al., 2023). As a result, the demand for effective fraud detection mechanisms has never been more urgent.

In response to these challenges, throughout the years, machine learning has emerged as a powerful tool for fraud detection (Ngai, Hu, Wong, Chen, & Sun, 2011). By analyzing large datasets and identifying complex patterns that may indicate fraudulent activity, machine learning algorithms can significantly improve the accuracy and efficiency of detecting fraudulent transactions (Ngai et al., 2011). Anomaly detection techniques have been employed in credit card fraud detection to discover nonconforming patterns in credit card transactions referred to as 'oddities' that can translate to fraudulent activities (Alamri & Ykhlef, 2022). Anomaly detection systems train a model on normal transactions using a variety of techniques to catch novel frauds. Classification algorithms such as Neural Networks, Decision Trees, Bayesian approaches, Random Forests, K-nearest neighbors, and Support Vector Machines have been used to detect fraudulent transactions in consumer behavior (Roy, Sun, Mahoney, Alonzi, Adams, & Beling, 2018).

However, researchers have been discussing several challenges in the detection of financial fraud (Hilal, Gadsden, Yawney, 2022). One of the most significant challenges in credit card fraud detection is the highly imbalanced nature of most fraud datasets, where fraudulent transactions represent a tiny fraction of the total (Dal Pozzolo, Boracchi, Caelen, Alippi, & Bontempi, 2017). The class imbalance problem in machine learning refers to classification tasks in which classes of data are not equally represented. According to different articles

(Krivko, 2010; Dal Pozzolo, Johnson, Caelen, Waterschoot, Chawla, & Bontempi, 2014), frauds are typically less than 1% of the overall transactions. Machine learning models trained on such imbalanced data often fail to correctly identify fraudulent activities, leading to high false-positive rates (Alamri & Ykhlef, 2022). The nature of fraudulent transactions often implies a heavy skew in the class distribution of this binary classification problem, i.e., in identifying fraudulent or non-fraudulent transactions. As many algorithms aim to maximize classification accuracy, the essential assumption of classification algorithms is that the data is balanced, which induces bias in modeling towards the majority class (Muaz, Jayabalan, & Thiruchelvam, 2020). A classifier can achieve high accuracy scores even when it does not correctly predict a single minority class instance. For instance, a trivial classifier that scores all credit card transactions as legit will score an accuracy of 99.9%, assuming that 0.1% of transactions are fraudulent; however, in this case, all fraudulent cases remain undetected.

Imbalanced learning techniques designed to enhance classification in the presence of class imbalance can be grouped into three main categories: data-level methods, algorithm-level methods, and hybrid approaches (Soh & Yusuf, 2019).

Data-level methods aim to change the class distribution towards a more balanced one to avoid the majority class's influence on the findings (Alamri & Ykhlef, 2022). This approach is divided into oversampling, undersampling, and hybrid sampling. Oversampling consists of adding instances to the minority class using duplicating existent samples or generating new ones while undersampling resamples the data by removing cases from the majority class. Because undersampling removes data, such methods face the risk of losing important concepts (Alamri & Ykhlef, 2022), while oversampling may lead to overfitting when observations are merely duplicated (Mînăstireanu & Meșniță, 2020). Hybrid sampling is a group of techniques that balances data by mixing oversampling and undersampling strategies. The goal of hybrid sampling is to overcome the limitations of oversampling and undersampling when applied alone (Khushi, Shaukat, Alam, Hameed, Uddin, Luo, Yang, & Reyes, 2021).

Algorithm-level methods, on the other hand, focus on adapting the learning algorithm itself to account for imbalances, often by altering the model's decision boundaries or assigning higher costs to misclassifying minority class instances (Alamri & Ykhlef, 2022). Within algorithm-level methods, there are two main approaches: cost-sensitive learning and ensemble learning (Shi, Li, Zhu, Yang, & Xu, 2023). Cost-sensitive methods aim to provide classification algorithms with different misclassification costs for each class. Alternatively, ensemble learning combines multiple models to improve overall performance. When combined with sampling techniques, these methods have proven to produce better results than other algorithms for datasets with class imbalance problems (Khan, Chaudhari, & Chandra, 2024).

Lastly, Hybrid approaches combine the benefits of both data and algorithm-level methods, optimizing performance through a more balanced dataset and an adjusted learning algorithm (Alamri & Ykhlef, 2022). In this approach, data-level methods modify the data distribution to

reduce the degree of the imbalanced data, whereas algorithm-level methods adjust the learning process to enhance classifier performance (Shi et al., 2023).

Another challenge in detecting fraudulent activities is understanding what “normal” behavior looks like in credit card transactions. Establishing a clear boundary between normal and anomalous behavior is challenging and lacks precision, which often leads to the misclassification of fraudulent transactions (Hilal et al., 2022). The notion of fraud also varies in the financial domain since a deviation of a certain degree might be considered normal (Hilal et al., 2022). For instance, if a person typically makes small, local purchases and suddenly spends a large sum at a foreign location, it might be seen as suspicious. However, if this same individual regularly travels and occasionally makes larger purchases abroad, that same transaction could be considered normal. This variability makes it difficult to standardize what constitutes a fraud, as certain behaviors could represent both legitimate and fraudulent activity, depending on the cardholder's profile and circumstances (Sulaiman, Nadher, & Hameed, 2024).

Furthermore, the idea of the current normal behavior of credit card transactions may not be representative enough of the future, as the increasing volume of online transactions has led to more sophisticated fraudulent activities, ultimately increasing the difficulty of the detection of fraud (Vanini et al., 2023).

Another concerning issue is the nature of the datasets available for research and model development. Due to privacy concerns, real-world transaction data is often anonymized, making it difficult to work with basic features such as customer identities, transaction locations, and other specific card details. This anonymization is done through techniques such as Principal Component Analysis (PCA) (Kulatilleke, 2017). For instance, datasets available for research, like the well-known European credit card dataset from the Machine Learning Group, contain anonymized variables (V1-V28) to remove any direct identifiers, leaving only abstracted data that still represents the core transactional details, such as transaction time and amount. This ensures that all personally identifiable information is either anonymized or removed from the dataset to avoid the misuse of personal data (Kennedy, Villanustre, Khoshgoftaar, & Salekshahrezaee, 2024). However, this anonymization limits the interpretability of the features and the ability to directly relate the dataset variables to real-world concepts since it is not possible to obtain semantic information from certain variables, thus complicating the development of effective fraud detection models by removing potentially informative features that could improve fraud prediction performance (Kulatilleke, 2017).

Besides privacy concerns, PCA is also used to reduce the dimensionality of credit card fraud since these datasets usually contain a large number of features (Hilal et al., 2022). When working in high-dimensional environments, anomalies often become hidden and unnoticed, which is also known as the curse of dimensionality (Zimek, Schubert, & Kriegel, 2012). In that sense, one can use PCA to transform the original features into new and uncorrelated variables

called principal components, reducing the number of original features (Lever, Krzywinski, & Altman, 2017). However, capturing intricate interactions among features in a high-dimensional space remains a significant challenge, especially for identifying fraudulent transactions that can manifest in subtle ways across multiple dimensions (Hilal et al., 2022). Ensuring that essential information is retained in this reduced feature space is vital for effective fraud detection. The heterogeneous nature of fraud further complicates this process, as various fraudulent activities can present diverse patterns within the data, making it difficult to maintain all relevant information during dimensionality reduction (Hilal et al., 2022).

Besides the limited availability of balanced datasets, credit card fraud datasets are rarely available in general (Alamri & Ykhlef, 2022). This limitation arises from the sensitive nature of financial data, which often leads researchers to rely on the same datasets in their studies (Kulatilleke, 2022). A notable example is the European credit card dataset from the Machine Learning Group, as previously mentioned, which has been widely used in fraud detection research over the years (Muaz et al., 2020; Itoo, Meenakshi, & Singh, 2021; Chole, Mukherjee, Gaikwad, Gawai, Bagde, Mahule, & Pawar, 2022; Bhakta, Ghosh, & Sadhukhan, 2023). This specific dataset has 284.807 credit card transactions related to a two-day period in September 2013 by European cardholders. Out of 284.807 transactions, there are 492 or 0.172% fraudulent transactions, indicating highly imbalanced data. Features are encoded in 28 principal components. Due to PCA encoding, it is not possible to obtain any semantic information from the features, except for the transaction date and amount. It is important to note that most contributions and results obtained from this dataset are specific to its characteristics and may not be easily generalized to other credit card fraud detection contexts. This dataset's unique features, such as only containing numerical features resulting from PCA, limit its applicability to broader fraud detection systems, highlighting the need for more diverse datasets in future research (Sinap, 2024).

Given the challenges of credit card fraud detection, particularly the highly imbalanced nature of fraudulent transactions and the limitations of currently available datasets, choosing the right sampling technique becomes a critical step for improving the performance of classifiers. Therefore, this thesis focuses on the following research questions: What is the impact of different imbalanced learning techniques on the performance of machine learning models for credit card fraud detection, and how can these techniques be optimized for better fraud identification?

There are 5 main objectives that will guide this research:

- Compare the effectiveness of different imbalanced learning techniques in addressing data imbalance in credit card fraud.
- Assess the performance of different classifiers when trained on datasets processed with different imbalanced learning techniques.
- Identify the optimal combination of imbalanced learning methods and classifier hyperparameters.

- Determine whether imbalanced learning techniques consistently improve fraud detection across multiple datasets or if their effectiveness varies.
- Evaluate the trade-offs between improving fraud detection rates and increasing false positives due to imbalanced learning techniques.

The remainder of this work is organized as follows. Section 2 presents a review and a summary of currently available imbalanced learning approaches. Special attention is paid to the related work on the use of imbalanced learning techniques in the context of credit card fraud detection. Section 3 presents the proposed framework to evaluate and compare the imbalanced learning techniques established. The experimental results and discussion are shown in section 4, followed by section 5, which presents the conclusions and future works.

2. LITERATURE REVIEW

2.1. IMBALANCED LEARNING APPROACHES

The most prevalent challenge that researchers face dealing with credit card fraud detection is that fraud datasets are often imbalanced (Alamri & Ykhlef, 2022; Ahmad, Kasasbeh, Aldabaybah, & Rawashdeh, 2022; Mienye & Sun, 2023). Most datasets are highly imbalanced since genuine credit card transactions significantly outnumber fraudulent transactions (Btoush, Zhou, Gururaian, Chan, & Tao, 2021).

Imbalanced learning techniques have been used in this context to address imbalanced data (Alamri & Ykhlef, 2022; Kaur, Pannu, & Malhi, 2019). These techniques can be categorized into three main types: the data-level, algorithm-level, and hybrid approaches (Soh & Yusuf, 2019). These approaches impact different phases of the learning process, where the data-level can be seen as a pre-processing step, while the algorithm-level implies a more customized and complex intervention in the algorithms, and the hybrid approach combines both techniques. In this section, we focus on previous work related to sampling techniques and ensemble learning methods while briefly explaining cost-sensitive solutions.

2.1.1. Data-Level Approach

The use of data-level methods in imbalanced learning applications involves the modification of an imbalanced dataset by removing the majority class instances and/ or generating artificial minority instances to provide a balanced distribution (Alamri & Ykhlef, 2022). The data-level approach has three subgroups of techniques: undersampling, oversampling, and hybrid methods (Tarekegn, Giacobini, & Michalak, 2021). We will be focusing on oversampling techniques as they are the most common approach to imbalanced learning in machine learning in general and credit card fraud in particular (Alamri & Ykhlef, 2022; Dal Pozzolo et al., 2017) while providing a brief explanation of undersampling and hybrid solutions.

2.1.1.1. Oversampling

Oversampling rebalances datasets by duplicating existing observations or generating new artificial instances belonging to the minority classes (Chawla et al., 2002). Unlike with the undersampling strategy, no valuable information is lost. However, when applied alone, researchers found that oversampling can lead to overfitting that may cause higher accuracy of the detection model due to the duplication of instances and overlapping, especially if the dataset is vast and highly unbalanced (Mînăstireanu & Meșniță, 2020; Khushi et al., 2021).

Several oversampling techniques have been proposed and developed in the past. Random oversampling (ROS) is a simple approach that randomly duplicates instances of the minority class until the desired balance between classes is met, which makes it easier to implement (Chawla et al., 2002). However, since the generated samples are merely replicates of the

original data, classifiers being trained on randomly oversampled data are more prone to overfit (Batista, Prati, & Monard, 2004; Chawla, Japkowicz, & Kołcz, 2004).

In 2002, Chawla et al. proposed the Synthetic Minority Oversampling Technique (SMOTE) algorithm, which prevents the risk of oversampling faced by the previous method since it generates artificial data instead of duplicating it. SMOTE generates synthetic instances through linear interpolation between a randomly chosen minority class sample and one of its nearest minority neighbors (Chawla et al., 2002). As shown in Figure 1, the process involves three steps: selecting a random minority instance $\vec{a}$, identifying one of its *k* nearest neighbors $\vec{b}$, and then creating a synthetic point $\vec{x}$ using the formula $\vec{x} = \vec{a} + w \times (\vec{b} - \vec{a})$, where w is a random weight between 0 and 1.

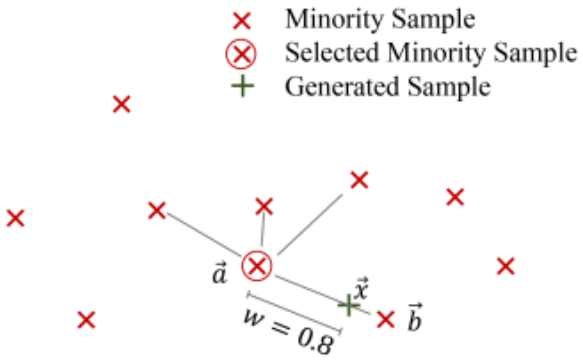

Figure 1 – SMOTE linearly interpolates a randomly selected minority sample and one of its $k = 4$ nearest neighbors, adopted from (Douzas, Bacao, & Last, 2018)

This method was developed to improve the simple random oversampling technique. However, it also has its own drawbacks in managing noise and class imbalance, as presented in Figure 2. While SMOTE effectively addresses *between-class imbalance* by uniformly selecting minority instances for synthetic generation (Douzas, Bacao, & Last, 2018), it does not resolve *within-class imbalance* or the issue of small disjuncts. As a result, regions densely populated with minority samples tend to become even more concentrated, while sparsely populated areas may continue to lack sufficient representation, potentially leading to underrepresented concepts in the dataset (Prati, Batista, & Monard, 2004). Additionally, a significant drawback of SMOTE is its vulnerability to generating noise, as it does not differentiate between regions where classes overlap and the "safe" areas where minority samples are more isolated (Bunkhumpornpat, Sinapiromsaran, & Lursinsap, 2009).

The algorithm treats all minority instances equally, without prioritizing those near the decision boundary. As a result, samples located far from the boundary are oversampled with the same likelihood as those situated close to it. Research suggests that classifiers might perform better if samples were generated closer to the boundary (Han, Wang, & Mao, 2005).

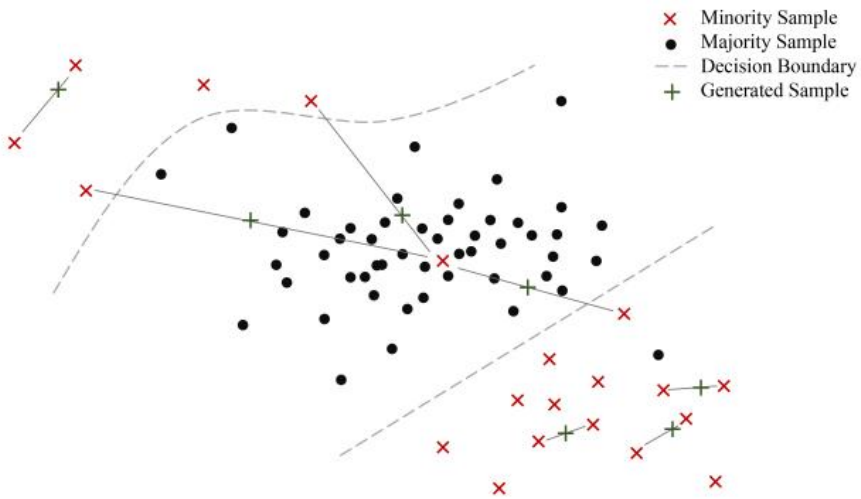

Figure 2 – Behavior of SMOTE in the presence of noise and within-class imbalance, adopted from (Douzas, Bacao, & Last, 2018)

Although SMOTE has certain limitations, it remains a widely accepted benchmark technique for handling imbalanced datasets. Over time, numerous adaptations and enhancements have been introduced to address its shortcomings and boost its effectiveness across various contexts.

Focusing its attention on the decision boundary, in 2005, Han et al. proposed the Borderline-SMOTE technique as an improved version of SMOTE that involves oversampling only the borderline of the minority class. Borderline-SMOTE oversamples the borderline data of the minority class, whereas SMOTE and random oversampling augment the minority class by using all or a random subset of the minority class (Han et al., 2005). There are two types of Borderline-SMOTE: Borderline-SMOTE1 and Borderline-SMOTE2. Borderline-SMOTE1 modifies the original SMOTE algorithm by replacing the random selection of samples with a more focused approach that prioritizes instances near the decision boundary between classes (Han et al., 2005). The method evaluates the labels of a sample's *k* nearest neighbors to determine whether it should be discarded as noise, retained due to its proximity to the boundary, or ignored for being too distant from it (Han et al., 2005). In contrast, Borderline-SMOTE2 builds upon this approach by enabling interpolation between a minority instance and one of its majority class neighbors. The interpolation weight is set to less than 0.5, ensuring that the synthetic sample remains closer to the minority class instance (Han et al., 2005).

In 2008, He et al. proposed the Adaptive Synthetic Sampling (ADASYN) method, which dynamically generates synthetic minority class samples based on their distribution characteristics. This technique emphasizes generating more synthetic data for minority instances that are difficult to classify, thereby placing greater focus on complex regions of the feature space. This strategy aims to reduce the bias caused by class imbalance and adjusts the decision boundary to better capture challenging cases (Zou, 2021). Compared to Borderline-

SMOTE, ADASYN concentrates more intensely on areas where class overlap is most significant (He, Bai, Garcia, & Li, 2008).

In 2018, Douzas et al. introduced the K-means SMOTE method, which combines the K-means clustering technique with the SMOTE to improve the oversampling process for imbalanced datasets. Empirical results show this method outperforms other oversampling techniques in the majority of cases since it avoids noise generation by ensuring that oversampling occurs primarily in safe regions (Douzas et al., 2018). It also focuses on both between-class and within-class imbalance, and mitigates the small disjuncts problem by increasing the representation of minority instances in sparsely populated areas (Douzas et al., 2018). An overview of the algorithm is presented in Figure 3.

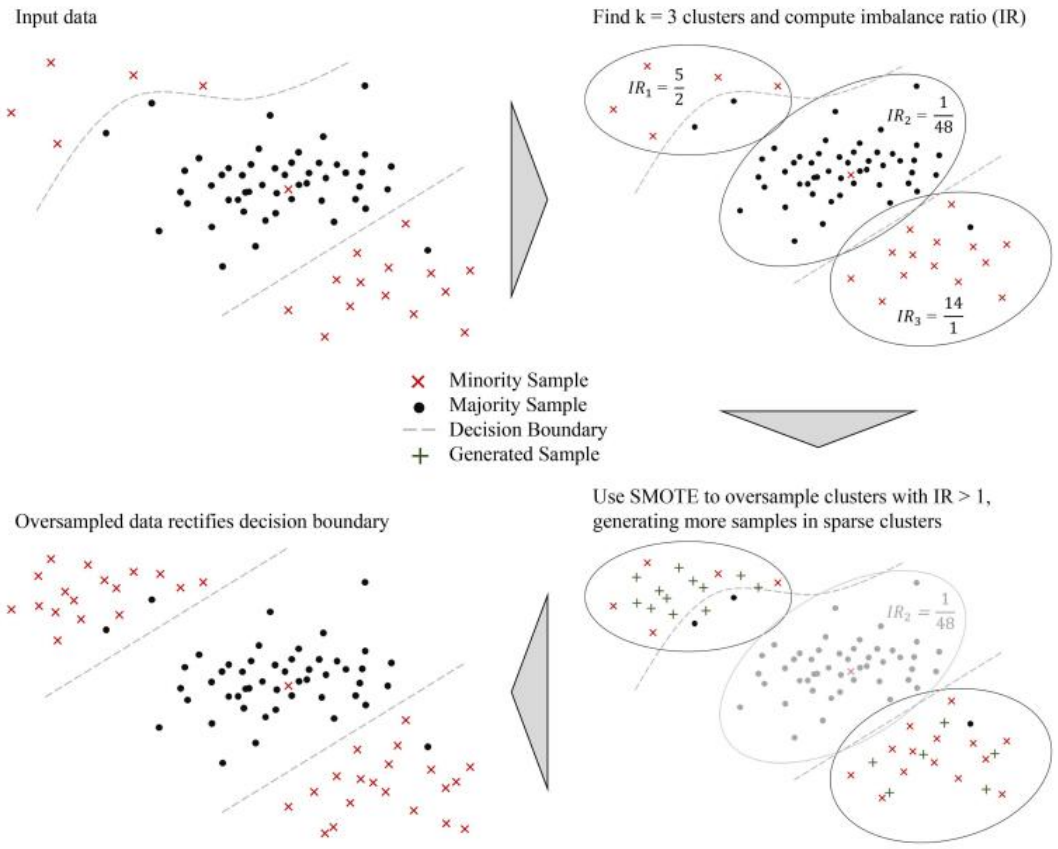

Figure 3 – K-means SMOTE oversamples safe areas and combats within-class imbalance, adopted from (Douzas, Bacao, & Last, 2018)

However, the application of this oversampler still remains underexplored in fraud detection. Researchers found that its performance is inconsistent across different credit card fraud datasets, and there is limited discussion on why clustering fails in some cases (Mahboob Alam, Shaukat, Hameed, Luo, Sarwar, Shabbir, Li, & Khushi, 2020). Furthermore, finding the optimal number of clusters and other hyperparameters is yet to be guided by rules of thumb (Douzas et al., 2018), which could be established through further analysis of the relationship between optimal hyperparameters and the specific characteristics of each dataset.

Researchers have also argued that synthetic data generative methods are one of the most effective approaches to dealing with imbalanced data (Assefa, Dervovic, Mahfouz, Tillman, Reddy, & Veloso, 2020; Lee, & Park, 2021). Among generative methods, one of the most popular is Generative Adversarial Networks (GANs) developed by Goodfellow et al. (2014). GANs have first revolutionized the area of generating synthetic images, as they can generate realistic images and videos (Goodfellow, Pouget-Abadie, Mirza, Xu, Warde-Farley, Ozair, Courville, & Bengio, 2014; Lu, Wu, Tai, & Tang, 2018). Due to their generative capability, GANs resemble an oversampling data-level approach (Sauber-Cole & Khoshgoftaar, 2022). GANs are based on the idea of competition, which consists of two neural networks trying to outsmart each other: the generator G and the discriminator D. The generator aims to generate realistic data from a batch of samples, while the discriminator distinguishes the synthetic data from the real data (Goodfellow et al., 2014).

To model the generator's distribution p_g over data x, the generator learns a mapping function from a prior noise distribution $p_z(z)$ to the data space through the function $G(z; \theta_g)$. The discriminator $D(x; \theta_d)$, on the other hand, produces a scalar output indicating the likelihood that a given sample x originates from the real training data rather than from the generator's distribution p_g. During training, the parameters of both neural networks are updated in an adversarial manner until the discriminator is no longer able to reliably distinguish between real and generated samples (Goodfellow et al., 2014). Figure 4 represents the block diagram of a GAN.

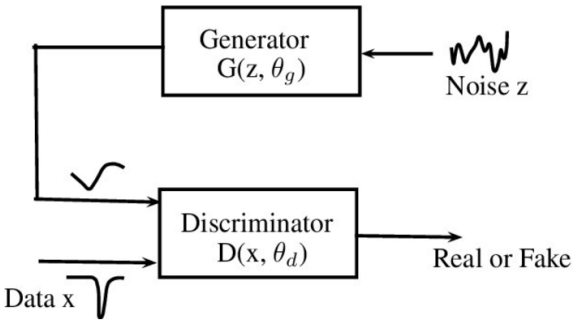

Figure 4 – General block diagram of Generative Adversarial Networks, adopted from (Bollepalli, Juvela, & Alku, 2017)

Throughout the years, multiple GAN variants have been proposed to expand GAN's applicability, including to generate artificial samples in tabular settings (Khan et al., 2024; Sauber-Cole & Khoshgoftaar, 2022; Cai, Xiong, Xu, Wang, Li, & Pan, 2021). Because GANs generate instances that do not represent random replications of existing instances without removing important information, researchers have argued that using GANs is a more effective technique for handling the imbalanced class problem compared to other machine learning techniques, including in the context of fraud detection (Ba, 2019; Sethia, Patel, & Raut, 2018; Liu & Lang, 2019). Several studies report that GANs performance on imbalanced datasets is superior to other imbalanced learning methods (Strelcenia & Prakoonwit, 2023). For instance,

a relative study was conducted on the performances of GANs with other sampling methods, such as SMOTE, and concluded that GANs can be more effective in mitigating the class imbalanced problem (Ngwenduna & Mbuvha, 2021). Furthermore, this method has been proven to be highly robust towards overlapping and overfitting due to its ability to understand hidden structures of data and their flexibility (Charitou, Dragicevic, & Garcez, 2021).

In 2014, Mirza et al. proposed the Conditional Generative Adversarial Network (cGAN), which extends the GAN model by conditioning the training procedure on external information y coming from the training data. The conditioning can be performed by feeding y into both the discriminator and generator as an additional input layer. Figure 5 illustrates the structure of a simple conditional adversarial network.

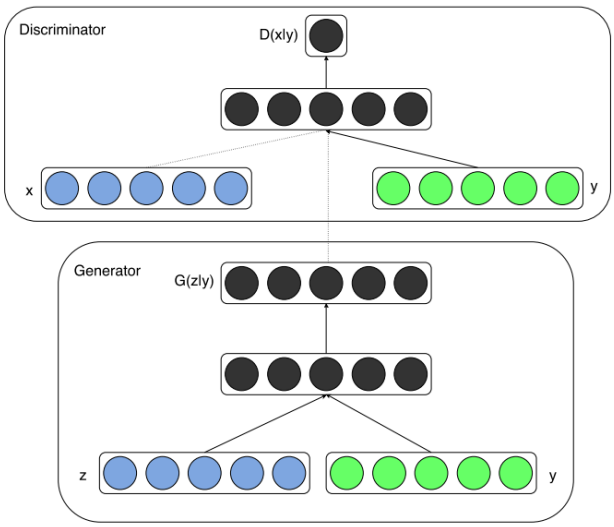

Figure 5 – Conditional Generative Adversarial Network, adopted from (Mirza & Osindero, 2014).

In the generator, y and the prior input noise $p_z(z)$ are combined into a joint hidden representation of a multi-layer perceptron (MLP), with the adversarial training framework allowing significant flexibility in determining the composition of this hidden representation (Mirza & Osindero, 2014). In the discriminator, x and y are provided as inputs to a discriminative function, embodied again by a MLP in this case (Mirza & Osindero, 2014).

Several studies have tested cGANs ability to generate synthetic data with high fidelity for imbalanced data. Douzas and Bacao (2018) used cGAN as an oversampling approach for binary class imbalance data on 71 datasets with different imbalance ratios and number of features. The proposed method was compared to ROS, SMOTE, Borderline SMOTE, ADASYN, and Cluster-SMOTE. For the evaluation of the oversampling methods, LR, SVM, KNN, DT, and GBM were used. Results showed cGAN performs better than other methods for most classifiers, evaluation metrics, and datasets. In fact, cGAN outperformed all other methods for any evaluation metrics when DT is used as a classifier. The authors concluded that this improvement in performance is due to the ability of cGAN to recover the distribution of the

training data, if given the proper time and enough capacity. Compared to standard oversampling methods, once the training is finished, the cGAN can generate instances of the minority class in a simple and effective way while accepting the noise and the label of the minority class as input and outputs the generated data (Douzas & Bacao, 2018). However, the authors highlight the need for more efficient ways to train the two neural networks in the context of the imbalanced learning problem (Douzas & Bacao, 2018).

On the other hand, GANs pose unique challenges to model realistic tabular data due to the need to simultaneously model discrete and continuous variables, the multi-modal non-Gaussian values within each continuous column, and the severe imbalance of categorical classes (Xu, Skoularidou, Cuesta-Infante, & Veeramachaneni, 2019).

To address the issues mentioned above, Xu et al. (2019) proposed a conditional tabular GAN (CT-GAN), which is an extension of cGANs, to generate synthetic tabular data. Its goal is to resample efficiently in a way that all categories from discrete attributes are sampled evenly during the training process and to recover the real data distribution during the test phase (Xu et al., 2019). Similarly to cGANs, CT-GAN uses a conditional generator that can generate synthetic rows conditioned on one of the discrete columns. In CT-GAN, the mode-specific normalization technique is leveraged to deal with columns that contain non-Gaussian and multi-modal distributions, while a conditional generator and training-by-sampling methods are used to combat class imbalance problems. The process starts by randomly selecting a discrete column D_i out of all the N_d total columns, with equal probability. For example, in Figure 6, the column selected was D_2, which means the selected column's index will be 2. With training-by-sampling, the vector *cond* and training data are sampled according to the log frequency of each category. Thus, CT-GAN can evenly explore all possible discrete values (Xu et al., 2019).

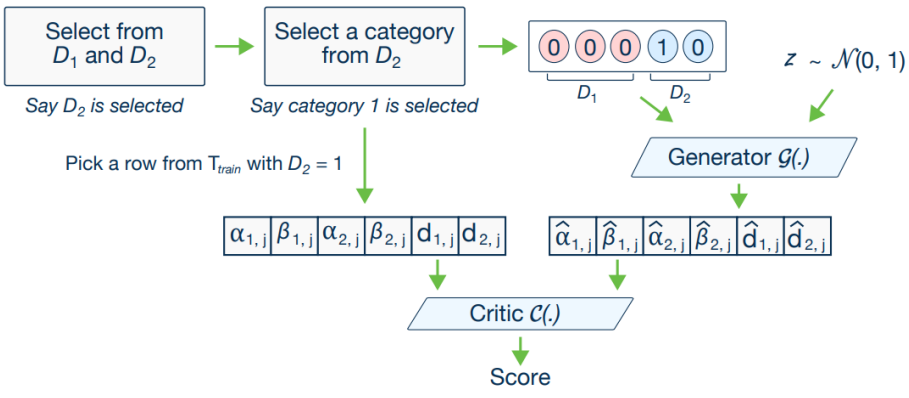

Figure 6 – The CT-GAN Model, adopted from Xu et al. (2019)

In 2021, Ferreira et al. introduced a novel generative framework called Duo-GAN, as demonstrated in Figure 7. The proposed method uses two GAN generators to handle the imbalance problem, one generator for fraudulent instances and another for legitimate

instances. This approach allows each GAN to learn the underlying distributions of data for each class, addressing the challenge of overexposure to non-fraudulent transactions commonly seen in single-GAN models (Ferreira, Lourenço, Cabral, & Fernandes, 2021). The Duo-GAN architecture was developed in the context of imbalanced learning in credit card fraud detection, proving to outperform single GAN generator models (Ferreira et al., 2021)., as further explained in the next section.

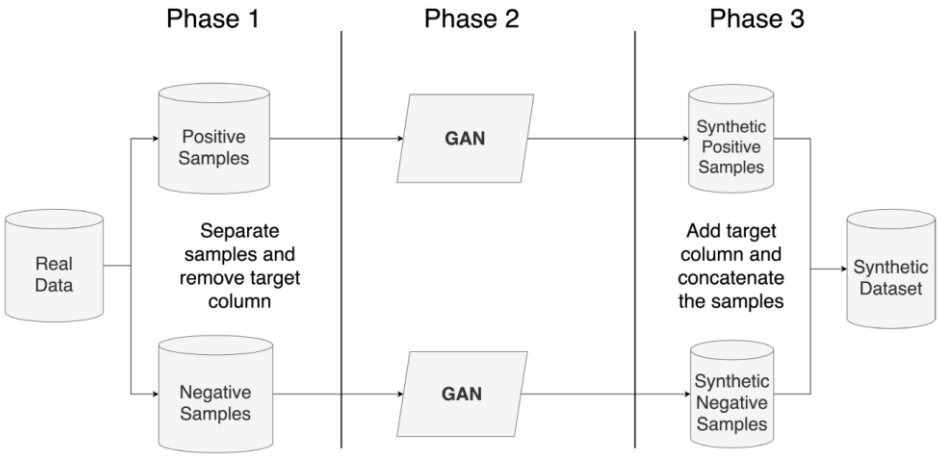

Figure 7 – The architecture of the Duo-GAN, adopted from (Ferreira et al., 2021)

In general, GANs have demonstrated considerable effectiveness in addressing class imbalance problems. Their deep neural network architectures allow them to capture complex data patterns, making them robust against issues like overfitting and class overlap. (Strelcenia & Prakoonwit, 2023). In addition, GANs have also achieved significant progress in credit card fraud detection by generating synthetic fraudulent samples, which improve model training efficacy (Strelcenia & Prakoonwit, 2023). However, the inherent complexity of GANs introduces challenges during training, such as instability and mode collapse, stemming from the adversarial, game-theoretic training of two neural networks simultaneously (Zhou, Zhang, Lv, Shi, & Chang, 2019). Another challenge regarding GANs for data augmentation in the context of credit card fraud detection is that the generated samples may not be representative enough of real-world fraudulent transactions (Niu, Wang, & Yang, 2019). Therefore, it is crucial to rigorously assess the quality and representativeness of synthetic data to ensure it accurately reflects real-world scenarios and is beneficial for training predictive models (Niu et al., 2019; Mullick, Datta, & Das, 2020).

2.1.1.2. Undersampling

Undersampling involves reducing the size of the original dataset by removing instances from the majority class to balance the data according to the minority class (Kubát & Matwin, 1997). This method addresses the class imbalance problem and increases the computational efficiency of the classification models, as it reduces the number of samples the model needs

to process (Xie et al., 2021). As a result, undersampling can be advantageous in scenarios where computational resources or time are limited, allowing for faster model training.

On the other hand, despite its computational benefits, undersampling has significant drawbacks. According to previous studies (Alamri & Ykhlef, 2022; Mahboob Alam et al., 2020), it may also result into information loss by eliminating instances from the majority class, which can be detrimental to the quality of the results, raising the false-positive rate and increasing the cost of investigations.

Random undersampling (RUS) is one of the simplest approaches, where instances from the majority class are randomly removed until the dataset achieves a desired class distribution (Batista et al., 2004). Despite its simplicity, this method has a significant flaw due to the loss of potentially important information that is pertinent to the classifiers since one cannot control what information about the majority class is eliminated (Mînăstireanu & Meșniță, 2020).

Tomek Links, Edited Nearest Neighbor, Near Miss, and cluster centroid undersampling are also frequently used undersampling techniques (Alamri & Ykhlef, 2022). Tomek Links is an undersampling method that identifies pairs of samples from different classes that are each other's nearest neighbors and removes the majority class sample from the pair (Tomek, 1976). This process refines the decision boundary by eliminating overlapping majority samples, thereby improving model performance in highly imbalanced scenarios (Batista et al., 2004). Edited Nearest Neighbors (ENN) extends this concept by removing the majority of samples misclassified by their k-nearest neighbors, further cleaning the dataset and reducing noise (Wilson, 1972).

Near Miss is an undersampling approach designed to reduce information loss when downsampling the majority class (Mani & Zhang, 2003). It operates by calculating average distances between majority class samples and points from the minority class to preserve those instances most informative for defining the class boundary. The method includes three variants, each targeting a clearer separation between classes (Mani & Zhang, 2003). NearMiss-1 chooses majority samples with the smallest average distance to their three nearest minority neighbors. NearMiss-2 selects majority samples closest to the three farthest minority points, emphasizing regions distant from minority clusters. Lastly, NearMiss-3 picks a specified number of majority points closest to each minority instance (Mani & Zhang, 2003).

To address the limitations of random undersampling, more sophisticated methods have been proposed. Cluster-based undersampling uses clustering algorithms to partition the majority class into clusters (Alamri & Ykhlef, 2022). Cluster centroid undersampling is a popular and successful unsupervised learning method that decreases the number of samples in the dataset (Yen & Lee, 2009). The proposed method divides the majority class into clusters of similar instances using the K-means clustering technique. Instead of keeping all the data points from the majority class, each cluster is represented by its centroid, which is the average position of

all the points in that cluster. The centroids effectively represent the overall structure of the majority class, ensuring the retained samples capture the diversity of the majority class while maintaining the data balance (Yen & Lee, 2009). This method helps mitigate the information loss problem but is computationally more expensive than RUS (Yen & Lee, 2009).

2.1.1.3. Hybrid Sampling

Hybrid methods combine both oversampling and undersampling techniques, resulting in the removal of instances of the majority classes and the generation of artificial instances of the minority classes (Alamri & Ykhlef, 2022). As previously explained, when applied alone, undersampling may discard useful information, while oversampling may lead to overfitting. Different techniques have been proposed to overcome these limitations, combining both methods (Khushi et al., 2021).

In 2004, Batista et al. developed a method that combines SMOTE with the undersampling method ENN. As previously discussed, ENN functions as a data cleaning process by removing majority class samples that disagree with the class labels of their nearest neighbors (Batista et al., 2004). The SMOTE-ENN approach leverages SMOTE's capability to generate synthetic minority class samples while using ENN to eliminate ambiguous or noisy instances from both classes, specifically those whose class label differs from that of their *k*-nearest neighbors (Vairetti, Assadi, & Maldonado, 2024).

In that same research, Batista et al. (2004) also proposed SMOTE-Tomek, combining SMOTE for oversampling and Tomek Links for undersampling. Tomek Links allow the removal of data that is identified as Tomek Links from the majority class, including samples of data from the minority class that are closest to the majority class (Alamri & Ykhlef, 2022). As a result, this method is able to eliminate examples from both classes.

Results show that both these methods provide very good results in practice, in particular for datasets with few positive (minority) examples (Batista et al., 2004). Thus, as a general recommendation, these techniques should be applied to datasets with fewer positive cases. The main issues with hybrid methods are that combining strategies can be extremely time-consuming (Vairetti et al., 2024), and the removal of instances when using undersampling may remove important data, which can negatively affect the overall classification performance (Alamri & Ykhlef, 2022). For instance, in 2024, Vairetti et al. implemented the SMOTE-ENN method over 35 different datasets, concluding that SMOTE-ENN was able to perform hybrid sampling in minutes and provide better results than well-known sampling techniques. However, a simple method such as RUS could perform very well in large-scale settings. The authors tested that RUS could resample all datasets in less than four seconds, suggesting that creating synthetic instances of the minority class is not strictly better nor necessary when millions of samples are available (Vairetti et al., 2024).

2.1.2. Algorithm-Level Approach

Algorithm-level methods are free of changes to the distribution of the data and instead enhance the classification performance by improving the classifiers (Alamri & Ykhlef, 2022). They attempt to eliminate algorithm sensitivity to the majority class so that the classifier's decision boundary is less biased toward the majority class (Kaur & Gosain, 2018). Within algorithm-level methods, there are two main approaches: cost-sensitive learning and ensemble learning (Shi et al., 2023).

2.1.2.1. Cost Sensitive Approach

Cost-sensitive methods aim to minimize the impact of misclassification by incorporating a cost matrix during the learning process, assigning different penalties to errors depending on the class (Alamri & Ykhlef, 2022). These algorithms tend to be more adaptable and better at identifying minority class instances because misclassifications of the minority class carry higher costs compared to the majority class (Fonseca, Douzas, & Bacao, 2021). By taking into account different losses for classification errors, cost-sensitive approaches avoid generating or augmenting data, which helps prevent the introduction of noise into the model (Singh, Ranjan, & Tiwari, 2021). The drawback of this method is the difficulty in precisely defining the cost matrix, which typically requires expert input from domain specialists (Xie et al., 2021). In the context of fraud detection, the cost of an undetected fraud is often assumed to be proportional to the transaction amount (Sahin, Bulkan, & Duman, 2013; Correa Bahnsen, Aouada, & Ottersten, 2015; Mahmoudi & Duman, 2015), leading to a higher misclassification cost to fraudulent cases. This cost structure encourages classifiers to generate more alerts to avoid missing fraud cases, often resulting in a high rate of false positives, while investigators require precise alerts (Dal Pozzolo et al., 2017).

2.1.2.2. Ensemble Learning

Ensemble learning is a meta-learning machine learning method that combines the predictions from multiple models to improve the predictive performance (Khan et al., 2024). The predictions of the different and diverse models are often combined by simple averaging or voting (possibly weighted). When combined with imbalanced learning techniques, ensemble methods have proven to produce better results than other algorithms for datasets with class imbalance problems (Khan et al., 2024). There are three main classes of ensemble learning: bagging, boosting, and stacking (Mienye & Sun, 2022; Ganaie, Hu, Malik, Tanveer, & Suganthan, 2022).

Bootstrap Aggregating (Bagging) (Breiman, 1996) combines bootstrapping and aggregation to form an ensemble model. It uses bootstrapping as a sampling method to create different samples to train different models and then aggregates the models' predictions to select the best prediction based on the average or majority of the results to compute a more accurate estimate. Bagging usually uses a Decision Tree (DT) as a base model (weak learner) to be trained on each bootstrapped dataset and combines the individual model predictions to

produce a final estimate. This results in a bias-variance trade-off since it reduces variance by averaging predictions from multiple models, making it less sensitive to noise and overfitting. However, it does not inherently reduce bias since all individual models are trained on similar data distributions, which results in the final model over-generalizing the data and missing relevant features between samples while training (Khan et al., 2024).

Random Forest (RF) was later proposed as an extension of bagging (Breiman, 2001). In comparison with conventional bagging, besides using the same ensemble learning framework, RF can also split the data feature-wise, i.e., random subsets of features can be selected for training each tree, enhancing the diversity between models. Hence, random forests can provide greater diversity in the decision tree and lower variance, as they are better at generalizing on test data when compared to decision trees (Biau & Scornet, 2015). In random forests, the base models are restricted to decision trees, whereas bagging methods allow any type of base model in the ensemble. However, random forests are the most prominent implementation of the bagging method (Khan et al., 2024).

In 1997, Freund and Schapire proposed an ensemble technique – Boosting – designed to sequentially improve the accuracy of weak learners by combining them into a strong learner. Each new weak learner tries to correct the errors made by the previous model, aggregating their predictions to make a final decision. Popular boosting methods include Adaptive Boosting (AdaBoost) (Freund & Schapire, 1996), Gradient Boosting (Friedman, 2001), and Extreme Gradient Boosting (XGBoost) (Chen & Guestrin, 2016).

AdaBoost is a boosting technique that was initially developed to improve the performance of binary classification (Freund & Schapire, 1996). It combines several weak classifiers based on decision trees, known as stumps, in order to form a single strong classifier. In contrast to RF, the stumps created are dependent on each other, and not all stumps have equal weightage since it works by iteratively adjusting the weights of the training samples (Freund & Schapire, 1996). Updated variations of AdaBoost have been developed to counter noisy data (Rätsch, Onoda, & Müller, 1999; Domingo & Watanabe, 2000; Oza, 2004; Gao & Gao, 2010), since AdaBoost is sensitive to noisy data and outliers by assigning equal weight to all training samples method (Khan et al., 2024).

Gradient Boosting is a versatile technique applicable to both classification and regression tasks. Gradient Boosting Regressor optimizes the mean-squared error loss, and Gradient Boosting Classifier minimizes the log-likelihood loss (Friedman, 2001), commonly referred to as Gradient Boosting Machines (GBM) (Natekin & Knoll, 2013). Its goal is to minimize the loss function by sequentially correcting the errors made by previous models by fitting subsequent models to the residual errors (Friedman, 2001). Gradient Boosting is known for its high predictive accuracy and capacity to handle large datasets effectively. However, as highlighted by Bentéjac et al. (2021), its performance heavily depends on careful tuning of hyperparameters such as the number of base learners and the learning rate.

XGBoost is an optimized and scalable implementation of Gradient Boosting, making it well-suited for high-dimensional problems and large-scale datasets for both classification and regression problems (Chen & Guestrin, 2016). It uses advanced regularization to improve model generalization capabilities, such as Lasso (L1) and ridge (L2) regularizations, which can help prevent overfitting. It improves computational performance since its training can be parallelized across clusters (Chen & Guestrin, 2016), and it is also considered to be a strong model in competitive machine learning, particularly in Kaggle competitions (Bojer & Meldgaard, 2021) due to the high accuracy and superior performance. Similar to Gradient Boosting, XGBoost requires fine-tuning for optimal performance, and it can be sensitive to the choice of parameters (Chen & Guestrin, 2016).

Wolpert (1992) proposed Stacking, an ensemble learning technique that combines multiple models (base learners) on the same data and uses a different model (meta-learner) to learn how to combine the predictions best. The predictions from the base learners are used as input for meta-learners. While stacking uses a single model that integrates the predictions from different models, boosting techniques use a sequence of models to enhance the predictions made by earlier models. Compared to bagging, stacking utilizes the entire dataset to train diverse base models, while bagging trains multiple base models on random subsets of the training data (Khan et al., 2024).

2.1.3. Hybrid Approach

The last category is hybrid methods, which combines data-level and algorithm-level methods (Alamri & Ykhlef, 2022). In hybrid methods, data-level methods can alter the data distribution to reduce the degree of the imbalanced data, while algorithm-level methods can change the learning process to improve the performance of the classifiers (Shi et al., 2023).

To address the instability in synthetic data generation and classification outcomes caused by SMOTE's randomness, Mansourifar and Shi (2020) introduced a method called Deep SMOTE, which is a deep neural network regression model designed to learn the mapping used in traditional SMOTE. In this architecture, a neural network is trained to accept two randomly selected minority class instances and generate a new sample along the line connecting them, preserving the original data dimensions (Mansourifar & Shi, 2020). Once trained, the model is applied to generate synthetic samples, thereby balancing the class distribution (Mansourifar & Shi, 2020). The authors concluded that Deep SMOTE performed better than SMOTE across all evaluation metrics (Mansourifar & Shi, 2020). In that same study, it was also proposed Deep Adversarial SMOTE (DA-SMOTE), which integrates concepts from SMOTE, generative adversarial networks, and Deep SMOTE (Mansourifar & Shi, 2020). Unlike Deep SMOTE, DA-SMOTE operates in an unsupervised manner, eliminating the need for explicit interpolation targets during training (Mansourifar & Shi, 2020). The authors concluded that both these techniques can enhance classification results, considering that DA-SMOTE has advantages over Deep SMOTE since it is trained in an unsupervised mode (Mansourifar & Shi, 2020).

In 2021, El Naby et al. proposed a new model, Oversampling with Convolution Neural Network (OSCNN), which is based on oversampling preprocessing SMOTE and convolution neural network (CNN). The model begins by oversampling the minority class by 0.25 to achieve a majority class ratio of 75:25, overcoming the overfitting caused by the SMOTE (El Naby, El-Din Hemdan, & El-Sayed, 2021). This model was proposed and developed to detect fraudsters in credit card transactions. Results show that OSCNN improves the predictive performance and surpasses other models (El Naby et al., 2021), which is further explained in the next section.

2.2. Imbalanced Learning Approaches for Credit Card Fraud Data

This section focuses on the application of the imbalanced learning techniques previously discussed, specifically applied in the context of credit card fraud detection. In recent years, the unique challenges posed by fraudulent transactions, combined with the highly imbalanced nature of the data, have led to the development of specialized methods tailored to improve fraud detection. This section reviews existing studies and approaches that have applied imbalanced learning strategies to credit card fraud detection, highlighting key findings, methodologies, and areas for future research.

In the context of credit card fraud, several studies have concluded that SMOTE-based sampling techniques tend to outperform undersampling methods (Alamri & Ykhlef, 2022). When applied alone, undersampling leads to a general decrease in the performance of the classification algorithms since it includes the removal of important information (Mînăstireanu & Meșniță, 2020).

Muaz et al. (2020) trained four classification models using artificial neural networks (ANN), gradient boosting, a stacked ensemble, and RF on different sampling methods. The imbalanced learning methods included RUS, SMOTE, density-based-SMOTE (DB-SMOTE), and the hybrid method SMOTE-ENN, which were used for all models. The dataset collected for this study consisted of transaction data of European credit cardholders, previously mentioned as one of the most common datasets used in fraud detection. Since the data had no missing values and outliers, the authors did no further pre-processing techniques. The dataset was split into 70% for training and 30% for testing using stratified random sampling. The findings of this study showed promising results with SMOTE-based techniques, concluding that the SMOTE method is the best sampling strategy to adopt. The authors considered the recall score to be the best metric to evaluate the performance of the machine learning models, followed by the precision and F1-score metrics. Even though the highest recall score of 83% was obtained with RUS with the ANN classifier, the model scored a very low precision of 27%. Thus, the authors concluded that while most fraudulent transactions are detected with this model, several genuine transactions are also misclassified as fraudulent. In that sense, the best combination of these metrics was obtained with the SMOTE sampling strategy by the Random Forest classifier, with a recall score of 81%, an F1-score of 84%, and a precision of 86% (Muaz et al., 2020).

Wibowo and Fatichah (2021) focused on analyzing oversampling approaches to address the problem of high-class imbalance. ROS, ADASYN, SMOTE, and Borderline-SMOTE approaches were compared in terms of performance. The classifiers RF, LR, and KNN were integrated with all oversampling approaches using the previous European credit card dataset. Six basic metrics were used for the evaluation of the study, namely accuracy, precision, recall, F1-score, Area Under the Precision-Recall Curve (PR-AUC), and the Area Under the Receiver Operating Characteristic Curve (ROC-AUC) to find out how well the model is able to differentiate between positive and negative classes. The authors concluded that Borderline-SMOTE achieved the highest results because it duplicates the margins of the minority class, which strengthens the difference between the minority and majority class data. The results revealed that combining RF with Borderline-SMOTE provided the best values, with an accuracy of 99.97%, precision of 94.74%, recall of 85.71%, F1-score of 90%, ROC-AUC of 93.88%, and PR-AUC of 85.81% (Wibowo & Fatichah, 2021).

Itoo et al. (2021) focused on random undersampling technique to balance a credit card fraud dataset with the classifiers Logistic Regression (LR), Naïve Bayes, and K-nearest neighbor (KNN). The dataset used was the same one previously described. In this research, the authors also split the dataset in two with the same percentages (70:30). However, they opted to implement the RUS by three different ratios: 50:50, 34:66, and 25:75 (fraud: nonfraud). To evaluate the performance of the three different classification techniques, the measures accuracy, recall, specificity, precision F1-measure and ROC-AUC were used. Overall, RUS improved the performance of the classifiers. The best evaluation scores were obtained with the LR model for all data proportions. In particular, LR obtained a recall of 87%, 77%, and 83% for the ratios 50:50, 34:66, and 25:75, respectively. However, authors have argued that one of the main disadvantages of RUS is that important information may be lost, and new sampling methods should be implemented to understand better how different techniques might affect the performance of the algorithms (Itoo et al., 2021).

Mahboob Alam et al. (2020) has shown that the combination of clustering methods with imbalanced learning techniques can improve the predictive accuracy of fraud detection. The authors applied different techniques, such as K-means SMOTE, ROS, SMOTE, ADASYN, SMOTE-Tomek, Borderline-SMOTE, RUS, Near Miss, and Cluster Centroid. Three highly imbalanced datasets related to credit card fraud were used and normalized using the Min-Max normalization method. The datasets were split into training and test data with a ratio of 70:30. Gradient Boosted Decision Tree (GBDT), AdaBoost, Bagging, Random Forest, KNN, Logistic Regression, and stacking models were evaluated using performance measures such as accuracy, precision, recall, F1-measure, ROC-AUC, and geometric mean (G-Mean). Overall, the results show that the performance of oversampling techniques is better than that of undersampling techniques. In fact, most classifiers improved their performance when K-means SMOTE oversampling was applied. Moreover, the GBDT model also presented a higher prediction accuracy rate than the traditional machine learning models. The best result obtained for one of the datasets was with the GBDT method while utilizing the K-means

SMOTE oversampling, with an accuracy of 88.7%, a recall of 92.2%, an F1-measure of 89%, and a G-Mean of 89%. However, for the other two datasets, the performance of K-means SMOTE is inconsistent, and there is limited discussion on why clustering fails in some cases (Mahboob Alam et al., 2020).

A novel fraud detection method has also been developed, where customers were grouped based on their transactions and behavioral patterns to create a profile for each cardholder (Dornadula & Geetha, 2019). Once again, the European dataset was used. The authors started the clustering method by dividing the cardholders into different groups based on their transaction amount, i.e., high, medium, and low, using range partitioning. After preprocessing and balancing the data using SMOTE, each group was trained on different classifiers - Local Outlier Factor (LOF), Isolation Forest (IF), Logistic Regression, Decision Tree, and Random Forest. Accuracy, precision, and the Matthews Correlation Coefficient (MCC) metrics were used to evaluate the performance of the classifiers. The MCC measures the quality of binary (two-class) classifications (Matthews, 1975). The authors considered MCC to be the best parameter for dealing with imbalanced datasets. Furthermore, LR, DT, and RF were the algorithms that provided the best results. In fact, RF scored the best, with an accuracy of 99%, a precision of 99%, and a MCC of 99% (Dornadula & Geetha, 2019).

When it comes to ensemble methods, several studies have reported random forests to achieve the best performance (Alamri & Ykhlef, 2022; Dal Pozzolo et al., 2017). In 2023, Jabeen et al. implemented numerous machine learning algorithms, including DT and LR, and applied the SMOTE technique to the same cardholder transactions from the European dataset, concluding that RF produced the best rests across all traditional performance metrics tested, achieving an f1-score of 98%, an accuracy of 99%, a precision of 98%, and a recall of 99% (Jabeen, Singh, & Vats, 2023). Furthermore, other researchers (Mahesh, Afrouz, & Areeckal, 2022) have also explored the use of RF with undersampling and hybrid techniques to detect fraudulent transactions, namely RUS and SMOTE-Tomek. The classification models used in this study were RF, LR, KNN, and support vector machines (SVM). Once again, the dataset used was the European credit card transactions dataset. Results showed that RF performed best across all data imbalance techniques. In particular, undersampled data shows the best results for classification predictions, with an F1-score of 0.94. However, the authors concluded that this cannot be generalized with a high confidence level since the number of undersampled data is much smaller. Among SMOTE and SMOTE-Tomek, the best results were obtained with SMOTE-Tomek with an F1-score of 0.93 (Mahesh et al., 2022). The effectiveness of RF with SMOTE oversampling has also been tested with different credit card fraud datasets. A study used four training datasets with different fraud rates from the University of California San Diego Fair Isaac Corporation (UCSD FICO) data mining competition 2009 dataset, which is a non-specific dataset for real-world credit card transactions (Ahirwar, Sharma, & Bano, 2020). The SMOTE technique was applied, and the data was trained with Naïve Bayes, KNN, SVM, Bagging Ensemble, and RF models. The authors concluded that balanced data with Random

Forest provides the best results, being able to achieve the best scores across all datasets, offering a good ROC area, as well as a good accuracy and a low error rate (Ahirwar et al., 2020).

Several studies have been using boosting techniques to deal with credit card fraud prediction (Jose, Devassy, & Antony, 2023; Khan et al., 2024; Mahboob Alam et al., 2020). In 2022, Chole et al. implemented several machine learning algorithms, such as logistic regression, random forest, decision trees, and XGBoost, for fraud detection using the well-known European credit card transactions dataset. The data was balanced with SMOTE, ROS, and RUS methods. The SMOTE oversampling produced the greatest area under the ROC curve for all algorithms used, resulting in XGBoost being the best classifier for identifying credit card fraud, with 99.96 % accuracy and precision.

A comparative study of ensemble learning algorithms (Bhakta et al., 2023) implemented XGBoost, random forest, voting, gradient boosting, AdaBoost, and stacking for the purpose of detecting fraudulent credit card transactions, including the traditional ones such as LR, DT, SVM, and Naïve Bayes. Once again, the European credit card transactions dataset was used. Contrary to the other studies that used the same dataset, the authors of this research did not implement any imbalanced learning technique. Instead, they focused on the feature engineering process of the dataset by performing feature selection techniques, such as correlation analysis and recursive feature elimination, and by creating new features using domain knowledge and data exploration (Bhakta et al., 2023). The final results suggest that the random forest and XGBoost algorithms demonstrated exceptional performance, achieving significant levels of accuracy and F1-score. The RF demonstrated a high level of accuracy of 99.96%, achieving a precision rate of 98.72%, a recall of 78.57%, and an F1-Score of 87.50%. The XGBoost algorithm exhibited robust performance, achieving a 99.94% accuracy rate, 96.67% precision, a recall of 82.86%, and an F1-Score of 89.23%. The authors concluded that ensemble learning methodologies outperformed traditional machine learning algorithms in detecting fraudulent credit card transactions (Bhakta et al., 2023).

Nallabothula (2022) also focused on the analysis of machine learning algorithms for fraud detection, including ensemble learning methods, for a more recent dataset. The dataset used was generated by Vesta Corporation and the IEEE Computational Intelligence Society (IEEE-CIS). The data was composed of two different files (identity and transaction), which are joined by the ID of each transaction for each train and test dataset. After merging the two datasets, the data had 590540 credit card transactions and 434 different variables. Once again, the data was highly imbalanced, with a total of 20663 fraudulent transactions, representing only 3.5% of the total transactions. The data contained missing values, outliers, and categorical and anonymized features, which required a more complex and detailed data pre-processing step, including data cleansing, label encoding, and data normalization. Due to its imbalance, data was resampled with SMOTE and RUS. The data was further trained with several models, including LR, RF, XGBoost, Decision Tree, SVM, and Gaussian Naïve Bayes. To evaluate the classifiers, accuracy, precision, recall, F1-score, and ROC-AUC metrics were used. The author

concluded that RF was the best classifier for predicting fraud transactions for both sampling methods. Overall, the implementation of the RF with the SMOTE technique achieved the best results with 97.4% accuracy, 81.8% precision, 65.6% ROC-AUC, 45.5% F1-score, and 31.5% recall (Nallabothula, 2022).

Hybrid methods have also been implemented to detect credit card fraud. El Naby et al. (2021) used SMOTE in a hybrid form with a convolution neural network approach to improve credit card fraud detection. Once again, the well-known European dataset was used, splitting it into 80% for training and 20% for validation. The oversampling convolution neural network model (OSCNN) model starts by oversampling the minority class. A ratio of 0.25 oversampling was used to solve the imbalance in the dataset so that the minority class-to-majority class ratio becomes 75:25. This ratio was selected to compensate for SMOTE's overfitting (El Naby et al., 2021). After oversampling the data, the OSCNN model employs the CNN with the respective hyperparameters and layers (El Naby et al., 2021). A multi-layer perceptron, with and without SMOTE, was also applied to the dataset to compare the effectiveness of the OSCNN model. Comparing the MLP-OSCNN results, they concluded that the OSCNN model achieved better results with 98% accuracy, 97% precision, and 91% recall. In fact, the OSCNN method was able to improve accuracy from 88% to 98% in comparison to the MLP model, with OSCNN surpassing all other models (El Naby et al., 2021).

Several studies have also tested the use of GANs as methods to cope with the class imbalance problem in the detection of credit card fraud. Ali, Hasan, Ghandour, and Al-Hchimy (2024) proposed a hybrid model that combined the predictions of DT, Naïve Bayes, LR, and SVMs models. Once again, the European dataset was used, and SMOTE, ROS, RUS, and a single GAN were applied to solve the imbalance problem. The authors focused on frequently used metrics to assess the model's performance, including F1-score, accuracy, precision, and recall. Experimental results showed that the GAN implementation achieved the best results, scoring an F1-score of 99.9%, in addition to other exceptional metrics. The SMOTE algorithm scored second-best, achieving an accuracy of 98.1% and an F1-score of 98.3%, followed by the random sampling methods that showed close performance between them. When examining the ROC curve to predict financial fraud, authors also concluded that the GAN algorithm provided the best results, concluding its superior ability to balance the data (Ali et al., 2024).

Ferreira et al. (2021) compared the performance of creating synthetic data with a double GAN architecture to a single tabular GAN (TGAN) architecture. Different machine learning models, such as XGBoost, AdaBoost, and Decision trees, were trained to evaluate the proposed method, using both single GAN and Duo-GAN approaches. The dataset used was the European dataset, which included some prior changes made to it to reduce the lack of balance and runtime of the experiments. In that sense, the authors sampled all 492 fraudulent transactions and then randomly sampled only 49.508 instances out of the remaining 284.315 legitimate transactions, still leaving the datasets highly imbalance with just 1% of the total transactions recorded as fraudulent (Ferreira et al., 2021). The experiments were conducted for 10,20, 50,

100, and 200 training epochs for both GANs. However, for some training epochs, the single GAN did not generate new fraudulent transactions during the epochs allowed to train. Therefore, the authors only considered the experiments after 50 epochs, where the single generator created fraudulent transactions. The results showed that Duo-GAN outperforms single GAN generator models, being able to generate high-quality synthetic datasets that allow the implementation of machine learning models that attain a performance similar to those same models trained on real fraud datasets. Furthermore, Duo-GAN can better capture the existing correlations between variables than the single GAN approach. Another interesting aspect the authors concluded is that the results obtained for the models trained for 100 or 200 epochs obtain worse performance than the models trained for 50 epochs. In particular, the best model was XGBoost, trained for 50 epochs with synthetic data generated by Duo-GAN, which obtained a classification performance with a 5% disparity in F1-scores between classifiers trained and tested on real data and those trained on synthetic data but tested on real data (Ferreira et al., 2021). Furthermore, with the results of the XGBoost model, the recall for the fraudulent class and a single GAN generator model is 3.62%, whilst for Duo-GAN as the generator model, the recall for the same class is 81.9%. Even though the authors concluded that Duo-GAN outperforms single GAN generator models, this framework did not include how different configurations of Duo-GAN could impact the quality and utility of synthetic data, including the computational resources and time required to train the models (Ferreira et al., 2021).

The effectiveness of conditional tabular GANs was also tested in fraud detection. Duggal (2022) has addressed the European dataset's skewness problem by using SMOTE and CT-GAN. Three classifiers were used to perform the experiments: IF, MLP, and RF. The performance metric considered for evaluating all models was the PR-AUC. Results showed that CT-GAN was successful for two of the three models, achieving a higher AUC score than SMOTE for IF and MLP. In particular, the best score was obtained from Isolation Forest for CT-GAN, with a PR-AUC of 86%. On the other hand, the SMOTE technique only performed better for the RF algorithm, achieving an AUC of 63%, while CT-GAN only achieved 53% (Duggal, 2022). Future work includes adjusting models' parameters, such as learning rate and node count, to improve the overall performance (Duggal, 2022). Moreover, Patil (2021) also combined the CT-GAN with LR, RF, and XGBoost, to balance the same European credit card dataset. After removing duplicates and scaling the original data, the author used the univariate feature selection technique SelectKBest to select the best and most relevant features to maintain in the dataset. After treating the dataset, the CT-GAN was applied to balance the data. Recall, F1-score, ROC-AUC, and G-Mean metrics were selected to evaluate the performance of the models. Results showed that classifiers trained on augmented data using CT-GAN significantly improved the overall prediction performance. For instance, RF and LR recorded an increase of 25% in recall score, which means that they made 25% fewer errors while classifying fraudulent transactions on balanced data. In terms of recall and F1-score values, RF combined with CT-GAN outperformed all other classifiers, achieving a recall and an F1-score of 100%. In second place

is the XGBoost model, with a recall of 91%, followed by the LR model, with a recall of 90% (Patil, 2021).

Lastly, Veigas, Regulagadda, and Kokatnoor (2021) implemented an optimized stacking ensemble (OSE) method using SMOTE-Tomek and tabular GANs to generate synthetic data. Once again, the European dataset was used. After normalizing the data, oversampling methods SMOTE-Tomek and GAN are used to balance the dataset. Individual models MLP, KNN, and SVM are trained on the balanced data. SMOTE-Tomek is used to train the SVM classifier, and GAN is used to train the KNN classifier and the MLP model. These base-learners are incorporated into the stacking ensemble model to fit the meta-learner, which creates an effective predictive model for credit card fraud detection. The test data is input to the classifiers' predictions, and these are sent to the OSE, i.e., the base learners SVM, KNN, and MLP models are used as input for the meta-learner. The OSE uses LR as a meta-learner since it forms no biases regarding the distributions of classes in a specific feature space, and the stacked prediction from the OSE is considered the final output (Veigas et al., 2021). The ensemble model starts by computing the accuracies and F1-scores of the classifiers and the neural network models, comparing them separately, followed by the evaluation of the final OSE model. The authors concluded that the performance of OSE enhanced both fraudulent and genuine classification accuracies, proving to be more robust than the stand-alone classifiers. The OSE achieved an F1-score of 90.5% and the highest accuracy of 99.8%. KNN showed the highest f1-score of 96% and an accuracy of 95%. Authors discussed that despite the slightly lower f1-score than KNN, the proposed ensemble OSE is still preferable due to its ability to harness the abilities of unsupervised MLP, which works better when finding hidden patterns of fraudulent transactions in real-life scenarios (Veigas et al., 2021). This research also highlighted the time taken for the OSE to train the given data as one major limitation. Since the dataset used is non-stationary, i.e., the credit card transactions do not have a stable or predictable behavior, it is harder to run the model with mostly pre-fitting and pre-trained parameters. To improve the model further, it is recommended to implement and test the concept of weighted voting to the predictions from the first layer of classifiers in the OSE. Furthermore, the use of boosting algorithms which can be trained on the synthetic data generated using the same oversampling techniques could also improve the model (Veigas et al., 2021).

The following table presents a comparison of the mentioned studies and their results.

Table 1 – Comparison of Imbalanced Learning Techniques for Credit Card Fraud Detection

Reference	Dataset	Imbalanced Techniques	Classifiers	Best Result	Performance Metrics
(Ahirwar et al., 2020)	UCSD FICO/2009	SMOTE	Naïve Bayes, KNN, SVM, Bagging Ensemble, and RF	SMOTE + RF	Accuracy = 99% F1-score = 99% Precision = 98% Recall = 99%
(Ali et al., 2024)	European dataset	SMOTE, ROS, RUS, GAN	DT + Naïve Bayes + LR + SVM	DT + Naïve Bayes + LR + SVM + GAN	Accuracy = 99.9% F1-score = 99.9% Precision = 99.9% Recall = 99.9%
(Bhakta et al., 2023)	European dataset	-	LR, DT, SVM, Naïve Bayes, XGBoost, RF, Voting, GB, AdaBoost, Stacking	RF/XGBoost	Accuracy = 99.96%/99.94% F1-score = 87.50%/ 89.23% Precision = 98.72%/ 96.67% Recall = 78.57%/ 82.86%
(Chole et al., 2022)	European dataset	SMOTE, ROS, RUS	LR, RF, DT, XGBoost	SMOTE + XGBoost	Accuracy = 99.96% Precision = 99.96% ROC-AUC = 91%
(Dornadula et al., 2019)	European dataset	SMOTE	LR, DT, RF, LOF, IF	SMOTE + RF	Accuracy = 99.98% MCC = 99.96% Precision = 99.96%
(Duggal, 2022)	European dataset	SMOTE, CT-GAN	RF, IF, MLP	CT-GAN + IF	PR-AUC = 86%
(El Naby et al., 2021)	European dataset	SMOTE, OSCNN	MLP	OSCNN	Accuracy = 98% Precision = 97% Recall = 91%
(Ferreira et al., 2021)	European dataset	TGAN, Duo-GAN	XGBoost, AdaBoost, Decision trees	Duo-GAN + XGBoost	F1-score = 85.26% Precision (fraud class) = 91.13% Precision (normal class) = 99.83% Recall (fraud class) = 81.88% Recall (normal class) = 99.97%
(Itoo et al., 2021)	European dataset	RUS	LR, Naïve Bayes, KNN	RUS + LR	Accuracy = 96% F1-score = 91%

					Precision = 99% Recall = 84% ROC-AUC = 92% Specificity = 99%
(Jabeen et al., 2023)	European dataset	SMOTE	LR, DT, RF	SMOTE + RF	Accuracy = 99% F1-score = 98% Precision = 98%
(Mahboob Alam et al., 2020)	3 UCI datasets	SMOTE, ROS, K-means SMOTE, ADASYN, SMOTE-Tomek, Borderline-SMOTE, RUS, Near Miss, Cluster Centroid	GBDT, AdaBoost, Bagging, RF, KNN, LR, Stacking	K-means SMOTE + GBDT	Accuracy = 88.7% F1-measure = 89% G-Mean = 89% Recall = 92.2%
(Mahesh et al., 2022)	European dataset	SMOTE, RUS, SMOTE-Tomek	RF, LR, KNN, SVM	RUS + RF	F1-score = 94% Precision = 94% Recall = 94%
(Muaz et al.,2020)	European dataset	SMOTE, RUS, DB-SMOTE, SMOTE-ENN	ANN, GBM, Stacking, RF	SMOTE + RF	F1-measure = 84% Precision = 86% Recall = 81%
(Nallaboth ula, 2022)	IEEE-CIS dataset	SMOTE, RUS	LR, RF, XGBoost, DT, SVM, and Naïve Bayes	SMOTE + RF	Accuracy = 97.4% F1-score = 45.5% Precision = 81.8% Recall = 31.5% ROC-AUC = 65.6%
(Patil, 2021)	European dataset	CT-GAN	LR, RF, XGBoost	CT-GAN + RF	F1-score = 100% G-Mean = 100% Recall = 100% ROC-AUC = 100%
(Veigas et al., 2021)	European dataset	SMOTE-Tomek + TGAN	SVM, KNN, MLP, Stacking	SMOTE-Tomek + TGAN + Stacking	Accuracy = 99.8% F1-score = 90.5%
(Wibowo et al., 2021)	European dataset	ROS, ADASYN, SMOTE, Borderline-SMOTE	RF, LR, KNN	Borderline-SMOTE + RF	Accuracy = 99.97% F1-score = 90% Precision = 94.74 PR-AUC = 85.81% Recall = 85.71% ROC-AUC = 93.88%

The review of relevant research dealing with the imbalance problem in credit card fraud data has shown that SMOTE-based oversampling techniques outperform undersampling methods, with Random Forest often being noted as the best-performing classifier. In fact, the SMOTE technique gives the best performance for classification across different performance metrics (Ahirwar et al., 2020; Nallabothula, 2022; Muaz et al., 2020; Dornadula & Geetha, 2019). As previously discussed, this technique has some limitations, including overfitting due to the duplication of instances and overlapping, which may cause higher accuracy than other approaches (Ahammad, Hossain, & Alam, 2020). Some oversampling extensions can solve the overfitting and overlapping issues and produce excellent results, such as Borderline-SMOTE (Wibowo & Fatichah, 2021), K-means (Mahboob Alam et al., 2020), DB-SMOTE (Muaz et al., 2020), and SMOTE-Tomek (Mahesh et al., 2022). Furthermore, GAN-based approaches, such as Duo-GAN (Ferreira et al., 2021) and CT-GAN (Patil, 2021), have shown that they can outperform traditional oversampling techniques in data augmentation for credit card fraud detection. Moreover, combining sampling methods with ensemble techniques has also proven to be consistently effective in fraud detection. Overall, independently of the sampling technique being implemented, ensemble methods, such as Random Forest, XGBoost, and Stacking, provided the best results (Ahirwar et al., 2020; Chole et al., 2022; Veigas et al., 2021).

In terms of performance measures, the best fraud detection performance is measured by low error and false negative rates (Alamri & Ykhlef, 2022). Accuracy, although commonly reported, is not a reliable metric for evaluating models on imbalanced datasets as it can be misleading due to the dominance of the majority class. For instance, Nallabothula (2022) observed that when applying traditional SMOTE, accuracy could reach very high levels, but F1-score and ROC-AUC were significantly lower. This occurred because the model would classify most of the majority class (non-fraudulent transactions) correctly, but it failed to detect enough fraudulent transactions, resulting in poor performance for the minority class. This illustrates that high accuracy does not necessarily translate to a well-performing model, especially when dealing with imbalanced data. Measurement metrics that consider the unbalanced nature of the data should be focused instead. For example, Wibowo et al. (2021) used the ROC-AUC and PR-AUC to demonstrate the effectiveness of Borderline-SMOTE in improving the separation between fraudulent and non-fraudulent transactions. Similarly, Muaz et al. (2020) emphasized the importance of the recall score, considering it the best metric to evaluate the performance of machine learning models on fraudulent data, followed by the precision and F1-score metrics.

Furthermore, most studies reviewed used the same European credit card transactions dataset, which contains anonymized features and a low fraud rate of 0.172%. This limits the ability to generalize the results to other datasets in the context of fraud detection. Relying on the results and conclusions obtained from a single dataset can lead to models not performing well when applied to new or different datasets since the results are specific to the dataset's unique characteristics, such as its features and the imbalance ratio. This makes the findings to be less applicable and potentially biased. Furthermore, the same data was preprocessed using

different techniques in different studies, complicating the ability to compare and generalize results. Some authors did not modify the dataset, applying the imbalanced learning approaches directly to the original dataset. Others implemented different preprocessing steps, such as feature selection, feature engineering, and data normalization. These variations in preprocessing can significantly impact the model performance, as certain techniques may enhance or worsen the data quality. Therefore, it is essential to incorporate a wider variety of datasets that reflect diverse credit card fraud patterns and behaviors, to contribute to more robust and generalizable findings in credit card fraud detection. Additionally, applying standardized and pre-defined preprocessing steps across the datasets would provide more consistency and allow for more precise comparisons between different imbalance approaches. To deal with the highly imbalanced nature of credit card fraud datasets, different sampling techniques should be implemented to understand their effectiveness in fraud prediction. Each sampling technique should tested using different classifiers to ensure that the results are not biased or overly dependent on a single model. To accurately assess model performance in the presence of class imbalance, appropriate evaluation metrics should be used instead of accuracy, which can be misleading in imbalanced scenarios. This comprehensive approach allows for a more reliable evaluation of which sampling strategies consistently improve fraud detection performance across different datasets, classifiers, and class imbalance ratios.

3. METHODOLOGY

This section describes the methodological approach used to implement and evaluate imbalanced learning techniques and machine learning models for credit card fraud detection. Figure 8 summarizes the steps taken in the proposed methodology. The first step focuses on the data collection process of credit card fraud datasets and the discovery of insights about the structure and quality of the data. This is followed by data preparation, covering all the steps needed to clean, transform, and integrate the data to ensure its suitability for the models. This includes treating inconsistencies, handling missing values, feature selection, data encoding, and normalization. Having the data prepared, the imbalanced learning techniques are implemented to address the imbalance problem in the chosen datasets. Once the data is balanced, different classifiers are implemented to build predictive models to reach optimal performance. Once the models are implemented, their performance is evaluated using appropriate performance metrics to assess their effectiveness in detecting fraudulent transactions. This experiment was carried out using Python 3.12.4 in a Jupyter Notebook environment. The notebook was executed on a system running Windows 10. The code implemented for this study is made available at https://github.com/rita-avila/fraud_detection.

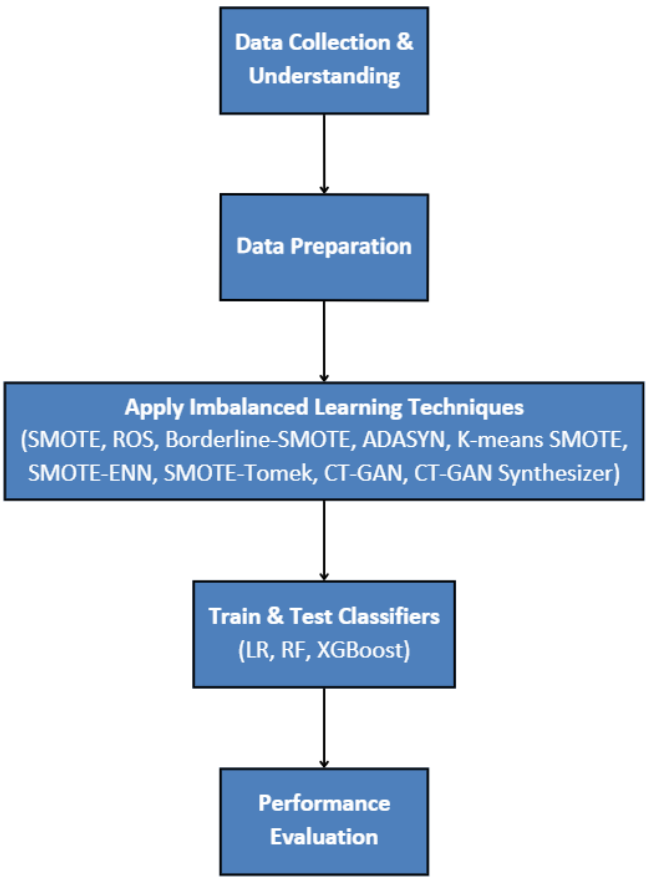

Figure 8 – Proposed Methodology

The following subsections provide a detailed explanation of each step, ensuring a systematic approach to addressing the challenges associated with fraud detection in highly imbalanced datasets.

3.1. DATA COLLECTION AND UNDERSTANDING

This section provides an overview of the datasets used, including their source, structure and key characteristics. To evaluate the performance of the different imbalanced learning techniques, five credit card fraud imbalanced datasets from Kaggle were used. Each dataset was analyzed to understand their characteristics, such as the presence of missing values and inconsistencies. Table 2 presents a summary of the dataset's characteristics.

Table 2 – Summary of the Datasets

Dataset	**Dataset 1**	**Dataset 2**	**Dataset 3**	**Dataset 4**	**Dataset 5**
Source	European Dataset	Simulated Dataset	IEEE-CIS Dataset	Synthetic Dataset	Unnamed Dataset
#Features	31	23	434	11	8
#Instances	284807	1296675	590540	6362620	1000000
#Minority Class	492	7506	20663	8213	87403
#Majority Class	284315	1289169	569877	6354407	912597
#Imbalance Ratio	577.88	171.75	27.58	773.70	10.44
Feature Types	Numerical	Numerical and categorical	Numerical and categorical	Numerical and categorical	Numerical
#Duplicate Instances	1081	0	0	0	0
Missing Values?	No	No	Yes	No	No
Inconsistencies?	No	No	No	No	No
Variables in the same scale?	No	No	No	No	No
Anonymized features?	Yes	No	Yes	No	No

All datasets used in this study share a common target variable, where 1 indicates fraud and 0 represents non-fraudulent transactions. However, they differ significantly in the number and type of features, dataset size, and imbalance ratio.

3.2. DATA PREPARATION

Data preparation is a crucial step involving the transformation of raw data into a format more suitable for analysis and modeling (Ndung'u, 2022). It ensures data quality, consistency, and relevance, directly impacting machine learning model performance. This section covers all the steps taken to clean, preprocess, and standardize the data.

3.2.1. Inconsistencies Fix

The analysis made to all datasets revealed no major inconsistencies, such as negative monetary units, invalid categories and incorrect timestamp. However, there were some cases spotted that required correction.

As shown in Table 2, Dataset 1 contained 1081 duplicate values. In that sense, the duplicate values were removed keeping the first occurrence. Furthermore, in Dataset 2, a meaningless variable called 'Unnamed: 0' that works as an ID for the table was dropped. Additionally, the variables that refer to *datetime* type, such as a user's date of birth and the transaction timestamp, were initially stored as *object* type. These were converted to the appropriate *datetime* format.

3.2.2. Missing Values

Only one dataset, Dataset 3, contains missing values, with 45% of its data missing. Out of 434 columns, 414 have missing values. Since a significant number of features were composed by mostly missing values, those with more than 5% of missing values were removed, eliminating 322 features and leaving 112. Among these, 92 features have minimal missing values, such as 0.2%. Given the low level of missingness, these features were retained, as they can be effectively imputed without introducing significant bias. In that sense, the numerical missing values were replaced with the median to mitigate the effect of outliers and ensure robust central tendency representation, and the categorical missing values were replaced with the mode to maintain the most frequently occurring category, minimizing data distortion (Lee & Yun, 2024).

3.2.3. Data Encoding

For the datasets that contain categorical features, namely datasets 2, 3, and 4, these features were converted into numerical format to ensure compatibility with machine learning models by assigning integer values to each category. Different techniques were applied, namely Label Encoding and One-Hot Encoding, depending on a feature's type. Whenever a categorical feature is not ordinal, i.e., it has no inherent order between categories, One-Hot Encoding is applied converting categorical variables into binary without introducing ordinal relationships, and whenever a categorical feature is ordinal, Label Encoding is applied assigning a unique number to each category (Poslavskaya & Korolev, 2023).

3.2.4. Feature Selection

As illustrated in table 2, the number of features significantly varies from dataset to dataset. Even though the credit card fraud datasets selected for this study are not equal in terms of the type of variables they have, the goal is that they are somehow comparable. Therefore, having a similar number of features can help to guarantee a degree of comparison between datasets. In that sense, different feature selection techniques were implemented, and their results were combined to identify the most relevant features to keep that will increase the performance of the models. For datasets with a large number of variables, such as datasets 1, 2, and 3, the techniques used were ANOVA f-test for continuous dependent variables (Siegel & Wagner, 2022), Chi-Square for categorical dependent variables (Sikri, Singh, & Dalal, 2023), Recursive Feature Elimination (Priyatno & Widiyaningtyas, 2024), and Decision Tree-based feature importance, which ranks features based on how effectively they reduce impurity when making splits (Scikit-learn, n.d.). Spearman's Correlation was also used for all datasets, including those with a smaller number of features, such as datasets 4 and 5, to remove redundant and non-discriminative variables (Bocianowski, Dorota, Krysztofiak-Kaniewska, Matusiak, & Wiatrowska, 2024). For each dataset, an assessment is made by considering the majority vote across all feature selection techniques to identify the variables that are most discriminative and relevant to the study.

3.2.5. Splitting the Datasets

The data is divided into two datasets, the train and test sets with a ratio of 70:30 using stratified train-test split. The train set will be used to find hyperparameters for both classifiers and imbalanced learning techniques, to balance the data and train the classifiers with the balanced data, while the test set will be used to test the performance of the models on unbalanced data. During training, cross-validation is applied with 5-folds to the train dataset. Once the optimal hyperparameters are identified, the final models are trained using the entire train set and evaluated on the test set. This process is detailed in the Modelling section.

3.2.6. Data Normalization

When working with multiple features, data normalization is essential to ensure consistency and comparability across different scales and distributions (Amorim, Cavalcanti, & Cruz, 2022). Features in each dataset have different scales, which can negatively impact the performance of machine learning models. Therefore, data was normalized using the Min-Max scaler (Amorim et al., 2022), which is a widely used normalization technique that rescales features to a fixed range, preserving the relationships between data points while eliminating scale disparities. In this case, data is rescaled between 0 and 1.

3.3. IMBALANCED LEARNING TECHNIQUES

The following table enumerates the most relevant imbalanced learning techniques mentioned in the Literature Review that are used as a benchmark to treat the imbalance present in the described datasets, along with the set of hyperparameters used for each. All techniques are used as implemented in the python library scikit-learn (Pedregosa et al., 2011) with default parameters unless stated otherwise, except for the GAN models which is further explained. The choice of the hyperparameters to optimize was based on the most relevant hyperparameters that significantly impact both model performance and computational efficiency, in order to avoid certain drawbacks, such as overfitting, noise, overlapping, discarding useful information, lack of flexibility and over-generalization (Alamri & Ykhlef, 2022).

Table 3 – Hyperparameter Optimization for Imbalanced Learning Techniques

Imbalanced Learning Techniques	**Parameters**
Random oversampling	Default
SMOTE	*k_neighbors* = 3, 5, 20
Borderline-SMOTE	*k_neighbors* = 3, 5, 20 *kind* = borderline-1, borderline-2
ADASYN	*k_neighbors* = 3, 5, 20
K-means SMOTE	*k_neighbors* = 3, 5, 20 *n_clusters*: 2, 8, 20
SMOTE-ENN	Default
SMOTE-Tomek	Default
CT-GAN	Default
CT-GAN Synthesizer	Default

Regarding the conditional tabular modelling technique CT-GAN, the implementation proposed by Xu et al. (2019) is selected, which is an open-source python package developed at MIT available at the Synthetic Data Vault. Similarly, CT-GAN Synthesizer was also implemented through an open-source python package developed at MIT available at the Synthetic Data Vault. This model is a more recent and improved approach from the Synthetic Data Vault library, which is built upon the CT-GAN model but offers additional improvements. It automates metadata handling, simplifying the training process by automatically identifying

and encoding categorical variables, which eliminates much of the manual setup required by the original CT-GAN (SDV, 2024). To keep any biases and contamination introduced by any specific parameterization from affecting the analysis of the results and increasing complexity, the default parameters for both GAN models were selected taking into account the recommendations of studies focused on improving fraud detection through the use of CT-GANs (Alqarni & Aljamaan, 2023; Duggal, 2022; Xu et al., 2019).

The optimal imbalance ratio is not obvious and has been discussed by other researchers (Chen et al., 2024; Aymaz, 2025). The goal of this work is to create comparability among these techniques. Consequently, it is most important that the same imbalance ratio is achieved when generating new samples for the minority class. Therefore, all the listed methods were parametrized so that minority and majority classes count the same number of samples for all datasets. Furthermore, since the ultimate goal of all these techniques is to improve classification results, their implementation is considered successful when compared to classifiers trained on the original, unbalanced data.

3.4. MODELLING

As discussed in the literature, the most common approach for handling imbalanced data is a combination of imbalanced learning techniques and classification algorithms (Alamri & Ykhlef, 2022). Both involve some hyperparameters that are often set to default values but could be tuned for better performance.

To achieve optimal results for all classifiers and imbalanced learning techniques, both components are going to be tuned. A grid search is first used to optimize the hyperparameters of each classifier using the original train set with cross-validation. This means that the unbalanced data will be used to determine the best hyperparameters (Kong, Kowalczyk, Nguyen, Bäck, & Menzel, 2019). Results are obtained by using 5-fold cross-validation, where each training dataset is split into five equal folds. For each possible combination of hyperparameters, a model is trained on four folds and tested on the remaining fold. This process is repeated five times, ensuring that each fold serves as a test set once.

After selecting the best hyperparameters for each classifier, imbalanced learning techniques are applied. The same classifier hyperparameters found using the unbalanced training data are retained for all imbalanced learning techniques. Once again, a grid search will also be used to find the best parameters for these techniques. The F1-score is used as the scoring parameter in these grid searches, not only because it is one of the key metrics highlighted in the Evaluation section but also because it helps select the best parameters by balancing a model's ability to detect fraud cases while minimizing false positives (Mahboob Alam et al., 2020). Having all parameters defined, the data is balanced, and then the classifiers are trained on the balanced data before being tested on the unbalanced test set. All performance metrics described in Evaluation section are recorded for each classification result for further comparison.

3.4.1. Classifiers

For the evaluation of the various imbalanced learning techniques, different classifiers are chosen to ensure that the results and conclusions obtained are not constrained to the usage of a specific classifier and can be generalized. The choice of classifiers is further motivated by the conclusions drawn from the literature review, which identify the most commonly used classifiers in credit card fraud detection and those that yield the best performance. Therefore, Random Forest, Logistic Regression, and XGBoost are used. These classifiers have demonstrated effectiveness in handling imbalanced datasets, making them well-suited for this study. All classifiers are used as implemented in the python library scikit-learn (Pedregosa et al., 2011) with default parameters unless stated otherwise. The choice of the hyperparameters to optimize was based on the most relevant hyperparameters that significantly impact both model performance and computational efficiency (Khan et al., 2024; Chereddy & Bolla, 2023; Bentéjac et al., 2021). Among the set of parameters that can be tuned, table 4 enumerates the classifiers used in this study along with a set of values for their respective hyperparameters:

Table 4 – Hyperparameter Optimization for Classifiers

Logistic Regression		
Parameter	**Default Value**	**Grid Search Values**
class_weight	None	None, balanced
C	1	3, 5, 7
Random Forest		
Parameter	**Default Value**	**Grid Search Values**
max_depth	None	None, 5, 10
min_samples_split	2	2, 10, 20
n_estimators	100	50, 75, 100
XGBoost		
Parameter	**Default Value**	**Grid Search Values**
learning_rate	0.1	0.025, 0.1, 0.3
gamma	0	0, 0.1, 0.2
max_depth	3	2, 3, 5

3.5. EVALUATION

Evaluation metrics are an important aspect to access and understand the performance of the machine learning models. The choice of a metric depends to a greater extent on the goal their user seeks to achieve.

The most widely employed evaluation metrics are based on four types of classifications: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). To illustrate the alignment of the predictions with the true distribution, a confusion matrix (Figure 9) can be constructed.

		Predicted Class	
		Predicted Positives	Predicted Negatives
Actual Class	Actual Positives (P)	TP	FN
	Actual Negatives (N)	FP	TN

Figure 9 – Confusion Matrix

The most common metrics for classification problems are accuracy and its inverse, the error rate.

Accuracy measures the number of correct overall predictions made by the model. However, this metric assumes a balanced and constant class distribution (Provost, Fawcett, & Kohavi, 1998), which is not the case in imbalanced datasets like credit card fraud detection, as previously discussed.

$$Accuracy = \frac{TP + TN}{P + N} \quad (1)$$

The same scenario happens for accuracy and its inverse, the error rate since these metrics show bias towards the majority class in imbalanced datasets. For example, a naive classifier which predicts all transactions as normal would achieve 99% accuracy in a dataset where only 1% of the transactions are fraudulent. While such high accuracy suggests an effective classifier, these metrics hide the fact that not a single minority instance was predicted correctly (He & Garcia, 2009).

$$Error\ Rate = 1 - Accuracy \quad (2)$$

To solve the impact of class imbalance on classification performance metrics, other metrics have been recommended (Luque, Carrasco, Martín, & de las Heras, 2019). The typical credit card fraud detection measure is the Area Under the Receiver Operating Characteristic Curve (ROC-AUC) (Dal Pozzolo et al., 2017). ROC is a probability curve, and AUC represents the

degree of separability. ROC-AUC evaluates a model's ability to distinguish between classes, and it can be interpreted as the probability that a classifier ranks frauds higher than real transactions (Muaz et al., 2020). Thus, the higher the AUC, the better the model is at predicting. Many researchers have adopted this metric which gives a good indication of the overall predictive performance of a model, being well suited for the class imbalanced problem (Jabeen et al., 2023; Ahmad et al., 2022; Alamri & Ykhlef, 2022; Muaz et al., 2020).

Another widely used metric is the Area Under the Precision-Recall Curve (PR-AUC) (Hilal et al., 2022). This metric shows the trade-off between recall and precision measures. Recall, also known as sensitivity, computes the proportion of positive cases that are correctly identified by the classifier, while the precision compares the instances the classifier said were positive with the actual number of real positive instances (Mahboob Alam et al., 2020).

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

PR-AUC focuses on the performance of a model concerning the minority class, i.e., fraudulent transactions, providing a more nuanced evaluation of a model's ability to detected rare events. In the context of credit card fraud, Leevy et al. (2022) concluded that PR-AUC can be a more reliable metric for highly imbalanced datasets than the ROC-AUC metric, since it doesn't consider true-negatives and it focuses directly on the positive class.

The F1-score is metric that combines the precision and recall as the harmonic mean. In the context of fraud, a high F1-score indicates a good trade-off between detecting fraud and avoiding false alarms (Mahboob Alam et al., 2020).

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5}$$

To take into account the unbalance nature of credit card fraud datasets and the metrics most commonly used by other researchers previously described, the following metrics are chosen to evaluate the performance of the classifiers:

- F1-score
- Recall
- Precision
- ROC-AUC
- PR-AUC

4. RESULTS AND DISCUSSION

This section discusses the experimental results from all techniques implemented. In Appendix A, Table 6 shows the performance of classification algorithms based on F1-score, Recall, Precision, ROC-AUC, and PR-AUC for all techniques across all datasets.

To facilitate the analysis, the focus will be placed on recall, as it ensures that fraudulent transactions are correctly identified despite the class imbalance. Given the high cost of missing fraud cases, prioritizing recall helps minimize undetected fraud, even if it means accepting some false positives as a trade-off. However, it is also essential to evaluate the trade-off between recall and precision, as an excessive number of false positives can lead to unnecessary operational costs and inefficiencies in fraud detection processes (Muaz et al., 2020). The figures below show the recall scores obtained for each dataset with each model.

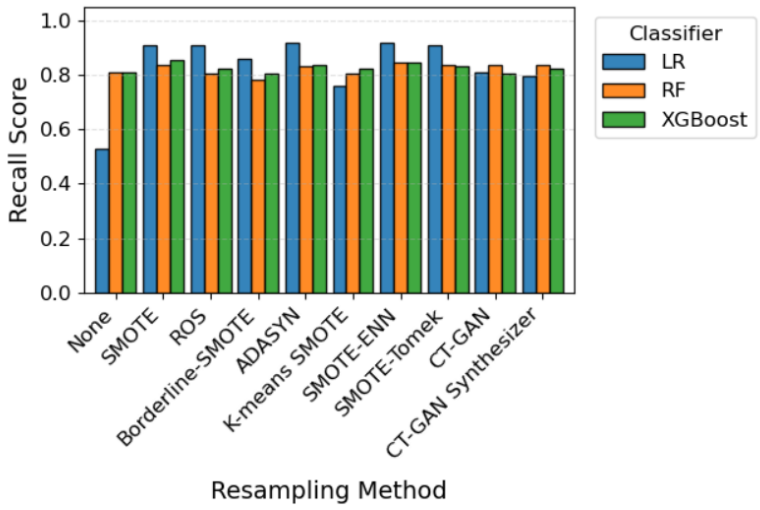

Figure 10 – Recall scores for Dataset 1

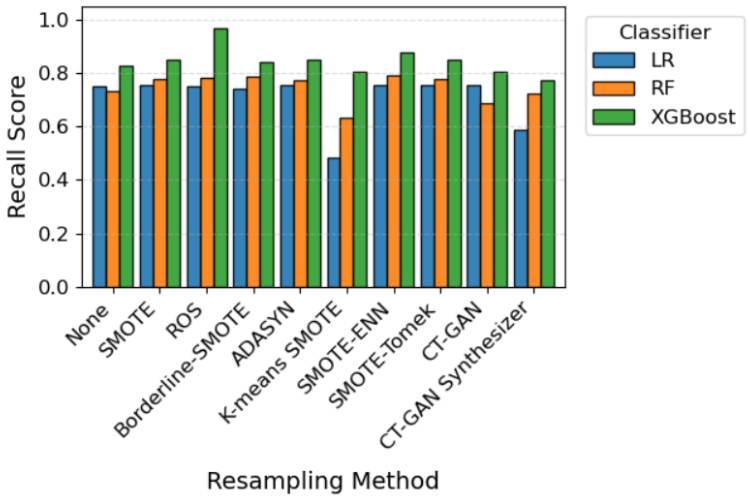

Figure 11 – Recall scores for Dataset 2

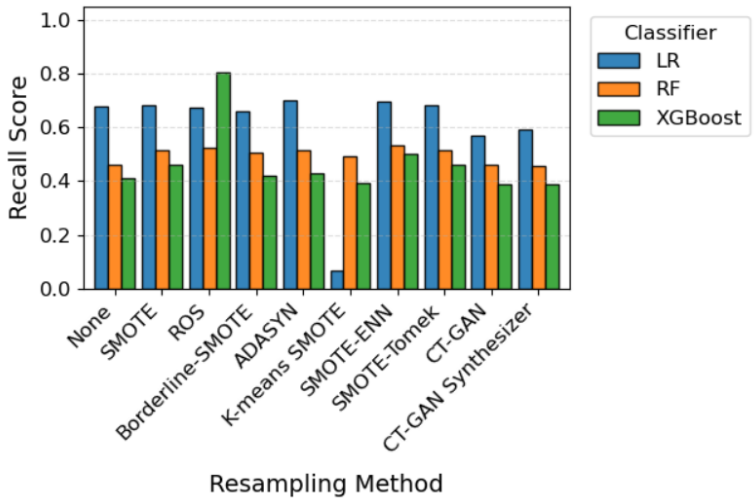

Figure 12 – Recall scores for Dataset 3

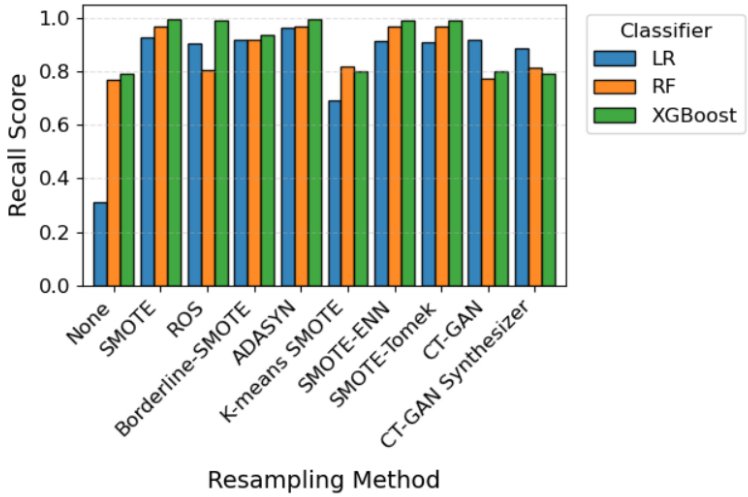

Figure 13 – Recall scores for Dataset 4

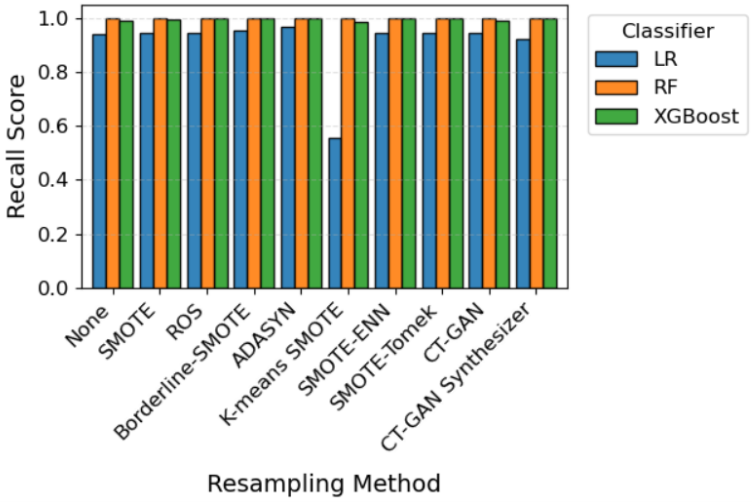

Figure 14 – Recall scores for Dataset 5

Starting with the analysis of the first dataset, the results indicate that the highest recall was achieved using ADASYN and SMOTE-ENN in combination with Logistic Regression, where the recall reached 92%. Based on the findings from the literature review, it is surprising that LR achieved the highest recall for this dataset, especially given the general consensus that more complex models like Random Forest and XGBoost are typically better suited for imbalanced datasets in fraud detection. The literature often emphasizes the superior performance of ensemble and gradient boosting methods, such as Random Forest and XGBoost, which are known to handle class imbalance more effectively through their ability to capture complex patterns and interactions within the data. Therefore, it was unexpected to observe that a simpler model like LR, typically associated with lower complexity, outperformed the others in terms of recall. Even though these techniques combined with LR achieved the highest recall score, they also drastically reduced precision, with values of 2% for ADASYN and 5% for SMOTE-ENN. This means that while these models were effective in identifying fraud, they flagged a large number of non-fraudulent transactions as fraudulent, leading to a higher number of false positives. In real-life situations, this could lead to unnecessary fraud alerts which is also not ideal. When examining stronger models based on their trade-off between recall and precision, XGBoost with SMOTE and Random Forest with SMOTE-ENN produced interesting results. Both techniques achieved a recall of 85%, while maintaining a better precision compared to the previously mentioned techniques, with XGBoost showing a precision of 60% and RF reaching 81%. These combinations provide a better balance between correctly identifying fraudulent transactions and reducing the number of false positives, offering an optimal trade-off between fraud detection and minimizing unnecessary alerts. Furthermore, SMOTE-Tomek and SMOTE combined with RF reached a slightly lower recall of 84% but improved precision achieving 93%. The highest ROC-AUC scores, 0.99, were obtained using XGBoost in combination with SMOTE-ENN, suggesting that this model exhibited a strong ability to separate fraud cases from non-fraud cases. Similarly, the PR-AUC values were highest for XGBoost with K-Means SMOTE, reaching 0.86. This indicates that this model exhibited strong fraud detection capabilities, particularly in scenarios where the data is highly imbalanced, as is often the case with credit card fraud datasets. Interestingly, RF and XGBoost without any imbalanced learning technique applied also performed well, with recall scores of 81% for both classifiers. This suggests that, in certain cases, imbalanced techniques may not always be necessary, as these classifiers were able to effectively handle the class imbalance on their own. These models alone also achieved high scores in terms of ROC-AUC, with scores ranging from 0.94 to 0.98. This reinforces the idea that imbalanced learning techniques are not always necessary for high-performing classifiers like Random Forest and XGBoost, which were able to effectively handle class imbalance without the need for additional sampling techniques. Although K-means SMOTE and GAN-based techniques yielded good performance in specific cases, they did not significantly outperform traditional techniques such as Borderline-SMOTE or SMOTE-Tomek, especially when considering F1-score and recall together. In fact, these techniques could not improve much the recall score compared to traditional methods and provided slightly lower F1-scores. This contradicts the expectations

set in the literature, where these advanced methods are often highlighted for their capacity to generate more realistic synthetic samples and improve classifier performance (Mahboob Alam et al., 2020; Patil, 2021).

Regarding the Simulated dataset, the recall scores obtained were higher than the previous dataset. Interestingly, in this dataset, the highest score of 97% was obtained with Random Oversampling and XGBoost. The fact that this method provides a better recall than other methods such as SMOTE is unexpected given the findings from the literature review. Prior research suggests that SMOTE generally outperforms ROS by generating synthetic minority instances that help classifiers learn a more generalized decision boundary (Chawla et al., 2002). However, the results indicate that for this dataset, SMOTE may have introduced noise or synthetic samples that did not effectively enhance fraud detection, leading to a lower recall compared to ROS. This could be due to the nature of the dataset's feature space, where fraudulent transactions may not be well-represented through interpolation-based synthetic sampling. On the other hand, ROS simply duplicates existing fraud samples, preserving the original distribution and potentially reinforcing patterns that the classifier can learn more effectively. These findings highlight that the effectiveness of imbalanced learning techniques is dataset-dependent and suggests that in certain cases, simpler oversampling strategies may yield better recall for fraud detection. However, the high recall was achieved at the cost of a significantly low precision of 42%, which results in an excessive number of false positives, which can be impractical in real-world applications. In contrast, SMOTE-based methods, such as SMOTE-ENN and SMOTE-Tomek, provided a better balance between recall and precision. Both methods with XGBoost achieved a recall of 87%, with higher precision scores of 68-69%. Another interesting result is the fact that XGBoost achieved alone the highest F1-score of 89%, along with a recall of 83% and a precision of 95%. The ability of XGBoost to perform well without any imbalanced learning technique suggests that the dataset may not suffer from extreme class imbalance or that XGBoost's intrinsic ability to handle skewed distributions is sufficient for effective fraud detection in this case. When comparing classifiers, XGBoost consistently outperformed both RF and LR across all methods. Random Forest performed reasonably well, achieving an F1-score of 84% without imbalance techniques, but showed a slightly lower recall than XGBoost. Logistic Regression, on the other hand, struggled significantly, with extremely low precision values across all sampling methods. This suggests that the dataset contains complex, non-linear relationships that are better captured by tree-based models rather than linear classifiers. Similarly to the previous dataset, more complex techniques including GANs did not significantly improve the results as expected. Even though they achieved higher precision scores, the recall dropped significantly.

Following the analysis of the third dataset, the IEEE-CIS Dataset achieved the lowest scores of all datasets tested. This might be due to the fact that this dataset was more complex and required more data preprocessing than the others. Similarly to the Simulated dataset, the highest recall value of 80% among all tested configurations was achieved by XGBoost with Random Oversampling. Once again, this came at a significant cost to a precision of 21%, resulting in a low F1-score of 34%. This is consistent with the expected behavior of ROS, which

replicates minority class instances without introducing new synthetic samples (Chawla et al., 2002). While this improves the model's ability to capture fraudulent transactions, it also increases false positives, leading to reduced precision. This combination also achieves the highest ROC-AUC of 0.93, showing strong overall classification capability. The following higher recall scores were achieved with Logistic Regression, facing the same problem of extremely low precision of only 10%, reinforcing the limitations of using linear classifiers for this task (Jabeen et al., 2023; Mahesh et al., 2022). On the other hand, RF with ROS provides a medium recall of 52% while maintaining a high precision of 85%, resulting in an F1-score of 65%. Furthermore, its ROC-AUC of 0.92 indicates a robust overall classification performance, demonstrating that it effectively discriminates between fraudulent and non-fraudulent transactions. SMOTE-ENN combined with RF was able to achieve a slightly higher recall of 53%, even though it decreased the precision to 72% and F1-score to 61%. When comparing the performance of SMOTE-ENN and ROS using RF, the results indicate that ROS delivers a slightly better overall performance. Although SMOTE-ENN achieves a marginally higher recall score, ROS outperforms all other evaluation metrics, including F1-score, precision, ROC-AUC, and PR-AUC. This suggests that ROS achieves a more favorable balance between correctly identifying fraudulent transactions and minimizing false positives. Given the small difference in recall but the significant gain in precision and overall F1-score, ROS can be considered the more effective technique in this case, ensuring a better trade-off between recall and precision.

When it comes to dataset 4, which is a synthetic dataset, methods such as ADASYN, SMOTE, SMOTE-ENN, SMOTE-Tomek, and ROS with XGBoost produced recall values of 99-100%, ensuring that nearly all fraudulent transactions are detected. However, these models suffer from extremely low precision of less than 20%, meaning they generate a large number of false positives. Borderline-SMOTE with XGBoost emerges as a strong candidate for the best trade-off. It achieves an F1-score of 79%, with 92% recall and a precision of 69%. Additionally, it achieves an AUC-ROC of 1.00, confirming its strong discriminatory ability. CT-GAN with XGBoost also shows a good trade-off, with an F1-score of 88%, 80% recall, and an exceptionally high 97% precision. This suggests that the model is detecting a significant proportion of fraudulent transactions while keeping false positives low. Additionally, it also achieves an AUC-ROC of 1.00. ROS with RF and XGBoost alone also perform well, both achieving an F1-score of 87% with relatively balanced recall, 81% and 79%, respectively, and high precision of 94% and 97%, respectively.

Lastly, the fifth dataset that represents real-world data reveals the highest results in the performance of different classifiers and imbalanced learning method combinations. Notably, both RF and XGBoost classifiers, when combined with almost any technique, achieved perfect or near-perfect performance metrics, with F1-scores, recall, precision, ROC-AUC, and PR-AUC all reaching values close to or equal to 1.00. For instance, methods such as ADASYN, SMOTE, Borderline-SMOTE, SMOTE-ENN, SMOTE-Tomek, ROS, and even the original unbalanced dataset, when used with Random Forest, resulted in an F1-score, recall, precision, ROC-AUC, and PR-AUC of 1.00. A similar trend is observed for XGBoost, where the performance slightly varies but remains consistently high. Additionally, the application of synthetic data generation techniques such as CT-GAN and CT-GAN Synthesizer also yielded strong results with both classifiers, although a slight drop is observed in precision and F1-score when compared to traditional oversampling methods. In contrast to these classifiers, LR exhibited significantly

lower performance across all metrics. Regardless of the technique used, the F1-scores remained between 69% and 72%, and recall ranged from 56% to 97%. The ROC-AUC values with Logistic Regression reached a maximum of 0.98 with some techniques but were mostly lower than those achieved with tree-based models. LR with K-means SMOTE was revealed to be the worst combination as it achieved the lowest recall of 56%.

Table 5 summarizes the performance of the best imbalanced learning techniques in terms of recall and the best trade-off between recall and precision for all datasets. Overall, across all datasets and imbalanced learning techniques, Random Forest and XGBoost consistently deliver the best performance, confirming findings from the literature review (Ahirwar et al., 2020; Chole et al., 2022; Veigas et al., 2021). Both classifiers maintain high F1-scores, recall, and precision, demonstrating their ability to effectively separate fraudulent and legitimate transactions, while Logistic Regression tends to struggle, with significantly lower F1-scores and precision, confirming its limitations when handling highly imbalanced datasets. Notably, XGBoost achieves near-perfect ROC-AUC values (~1.00) across multiple datasets, highlighting its strong generalization capabilities. Random Forest also exhibits stable and strong performance, especially in hybrid approaches, making it a highly reliable classifier. These results align with prior research, which has shown that ensemble methods tend to outperform single classifiers as they are the most reliable classifiers for credit card fraud detection, particularly when paired with sophisticated imbalanced learning techniques, due to their ability to mitigate overfitting and capture complex fraud patterns (Alamri & Ykhlef, 2022; Jabeen et al., 2023).

Applying oversampling techniques, including SMOTE, ROS, Borderline-SMOTE, and ADASYN, resulted in varying levels of performance improvement across the classifiers. While oversampling generally improved recall, it often led to a drop in precision, particularly for Logistic Regression. Among these techniques, SMOTE tended to introduce synthetic noise, which negatively impacted model performance. These findings align with prior research, which identified SMOTE as a widely used technique that improves classification performance but has inherent limitations (Ahammad, Hossain, & Alam, 2020). However, ROS performed slightly better than SMOTE, particularly for Random Forest and XGBoost. Borderline-SMOTE and ADASYN showed improved performance over standard SMOTE, as they focused on generating synthetic samples in more informative regions of the feature space, which helped XGBoost and Random Forest achieve higher F1-scores and more balanced precision-recall trade-offs.

Table 5 – Comparison of the Best Imbalanced Techniques for Credit Card Fraud Datasets

Dataset	Best Recall (Method + Classifier)	Performance Metrics for Best Recall	Best Trade-Off (Method + Classifier)	Performance Metrics for Best Trade-Off
European Dataset	SMOTE-ENN + LR	F1-score = 10% **Recall = 92%** Precision = 5% ROC-AUC = 98% PR-AUC = 72%	SMOTE-ENN + RF	F1-score = 82% **Recall = 85%** Precision = 81% ROC-AUC = 98% PR-AUC = 84%
Simulated Dataset	ROS + XGBoost	F1-score = 59% **Recall = 97%** Precision = 42% ROC-AUC = 100% PR-AUC = 94%	SMOTE-Tomek + XGBoost	F1-score = 77% **Recall = 87%** Precision = 69% ROC-AUC = 100% PR-AUC = 89%
IEEE-CIS Dataset	ROS + XGBoost	F1-score = 34% **Recall = 80%** Precision = 21% ROC-AUC = 93% PR-AUC = 59%	ROS + RF	F1-score = 65% **Recall = 52%** Precision = 85% ROC-AUC = 92% PR-AUC = 69%
Synthetic Dataset	ADASYN + XGBoost	F1-score = 27% **Recall = 100%** Precision = 16% ROC-AUC = 100% PR-AUC = 93%	Borderline-SMOTE + RF	F1-score = 79% **Recall = 92%** Precision = 69% ROC-AUC = 100% PR-AUC = 93%
Unnamed Dataset	ADASYN + RF	F1-score = 100% **Recall = 100%** Precision = 100% ROC-AUC = 100% PR-AUC = 100%	ADASYN + RF	F1-score = 100% **Recall = 100%** Precision = 100% ROC-AUC = 100% PR-AUC = 100%

The application of hybrid approaches, such as SMOTE-ENN and SMOTE-Tomek, revealed additional insights into classifier performance. These methods combined oversampling with data-cleaning techniques, aiming to enhance class separability by removing noisy or redundant samples from the majority class. As previously discussed in the literature review, SMOTE-Tomek has been noted for its ability to improve recall while addressing overfitting issues associated with traditional SMOTE (Mahesh et al., 2022). The results of this study confirmed these findings, as SMOTE-ENN and SMOTE-Tomek were able to improve the trade-off between recall and precision, making them highly suitable for fraud detection.

Although recent studies have emphasized the potential of advanced techniques such as K-means SMOTE and CT-GANs to improve fraud detection performance in imbalanced datasets

(Mahboob Alam et al., 2020; Patil, 2021), the empirical results obtained in this research suggest otherwise. Across all five datasets, these newer approaches did not consistently outperform traditional sampling methods like Borderline-SMOTE or ROS. In some datasets, K-means SMOTE and CT-GAN-based approaches matched the performance of traditional oversampling methods, while in others, they underperformed or resulted in higher variability, particularly in terms of recall and F1-score. This suggests that the increased complexity of these methods does not necessarily translate into better fraud detection performance, especially when combined with strong classifiers such as Random Forest and XGBoost. This behavior might be due to the fact the datasets used in this study may not have contained complex patterns or high-dimensional nonlinearities that would justify the use of sophisticated generative models like CT-GANs (Xu et al., 2019). Traditional imbalanced learning techniques were sufficient to expose fraudulent patterns. According to the literature, K-means SMOTE is identified as an effective alternative to traditional oversampling methods, reducing the risk of synthetic noise while improving class separability (Mahboob Alam et al., 2020). The unexpected lower results might be due to the fact this technique requires selecting an optimal number of clusters and can struggle when fraudulent instances are extremely sparse or do not form distinct clusters. This may have limited its effectiveness in the datasets evaluated. When comparing CT-GAN and CT-GAN Synthesizer, an interesting distinction emerged. While both techniques slightly enhanced classification performance, CT-GAN Synthesizer exhibited a slight advantage over standard CT-GAN, particularly in terms of precision and F1-score. This improvement suggests that Synthesizer's approach to generating synthetic fraud samples was more refined, leading to fewer misclassifications and a better balance between detecting fraudulent transactions and reducing false positives. While CT-GAN achieved slightly higher recall, its increased tendency to generate more diverse fraud examples also led to a small drop in precision. The CT-GAN Synthesizer, on the other hand, maintained strong recall while simultaneously improving precision, making it the more balanced option of the two.

5. CONCLUSIONS AND FUTURE WORKS

The growth of e-commerce and the increasing use of credit cards have resulted in a significant rise in credit card fraud, impacting both financial institutions and consumers. Detecting credit card fraud is a critical yet challenging task due to the complex and evolving nature of fraudulent activities.

This study contributes to the existing literature by evaluating various imbalanced learning techniques across five credit card fraud datasets to improve fraud detection performance while maintaining computational efficiency. By addressing the research question – "What is the impact of different imbalanced learning techniques on the performance of machine learning models for credit card fraud detection, and how can these techniques be optimized for better fraud identification?" – the study provides several key insights. Empirical results highlight that traditional oversampling methods like ROS and SMOTE variants, consistently improved model performance when combined with strong classifiers like Random Forest and XGBoost. These methods not only increased recall, ensuring a higher detection rate of fraudulent transactions, but also maintained a favorable balance with precision, thereby minimizing the incidence of false positives. Hybrid methods such as SMOTE-ENN and SMOTE-Tomek also improved recall while guaranteeing a balanced trade-off with precision. In contrast, more advanced techniques, including CT-GAN and K-means SMOTE, did not demonstrate the expected improvements. Instead, these methods occasionally introduced variability that did not translate into overall performance gains when compared to the simpler oversampling strategies.

In terms of optimization, the study shows that the effectiveness of imbalanced learning techniques improves when they are paired with high-performing classifiers, confirming that XGBoost and Random Forest are the most suited classifiers for credit card fraud detection, regardless of the sampling method applied. Additionally, tuning hyperparameters through grid search for all classifiers and imbalanced learning techniques allowed to identify the optimal combination of imbalanced learning methods and classifier, which improved the results. Furthermore, evaluating performance with metrics suited for imbalanced data was crucial for guiding model optimization and ensuring reliable performance comparisons.

Another key contribution of this research is the use of five different datasets to validate the robustness of imbalanced learning techniques. Most of the studies reviewed in the literature relied on the same benchmark dataset, limiting their generalizability. Given that fraud detection datasets vary in size, feature distributions, and fraud-to-non-fraud ratios, relying on a single dataset may lead to biased conclusions. Some techniques, such as SMOTE, performed well on datasets with moderate imbalance but struggled with extreme imbalance cases. GAN-based methods showed potential in generating more realistic fraud instances, but their performance varied across datasets. These findings emphasize the importance of dataset

diversity when developing fraud detection models, ensuring that the proposed solutions generalize well across different real-world scenarios.

Future work could explore additional ensemble methods beyond those tested in this study to assess their potential for further improving fraud detection performance. Investigating other ensemble approaches, such as stacked models that combine multiple classifiers and incorporate anomaly detection methods, could provide valuable insights into optimizing fraud detection while balancing interpretability and computational efficiency. Additionally, future studies could also include a broader range of datasets, particularly those with more complex and high-dimensional characteristics, to determine whether sophisticated generative models can better capture data patterns in fraud detection. Since the datasets used were very different from each other, data preprocessing could also be even more tailored to each dataset according to its unique characteristics and components.

BIBLIOGRAPHICAL REFERENCES

Ahammad, J., Hossain, N., & Alam, M. S. (2020). Credit Card Fraud Detection using Data Pre-processing on Imbalanced Data—Both Oversampling and Undersampling. *Proceedings of the International Conference on Computing Advancements*, 1–4. https://doi.org/10.1145/3377049.3377113

Ahirwar, Dr. A., Sharma, N., & Bano, A. (2020). Enhanced SMOTE & Fast Random Forest Techniques for Credit Card Fraud Detection. *Solid State Technology*, *63*, 4721–4733.

Ahmad, H., Kasasbeh, B., Aldabaybah, B., & Rawashdeh, E. (2022). Class balancing framework for credit card fraud detection based on clustering and similarity-based selection (SBS). *International Journal of Information Technology*, *15*(1), 325–333. https://doi.org/10.1007/s41870-022-00987-w

Alamri, M., & Ykhlef, M. (2022). Survey of Credit Card Anomaly and Fraud Detection Using Sampling Techniques. *Electronics*, *11*, 4003. https://doi.org/10.3390/electronics11234003

Ali, N. T., Hasan, S. J., Ghandour, A., & Al-Hchimy, Z. S. (2024). Improving credit card fraud detection using machine learning and GAN technology. *BIO Web of Conferences*, *97*, 00076. https://doi.org/10.1051/bioconf/20249700076

Alqarni, A., & Aljamaan, H. (2023). Leveraging Ensemble Learning with Generative Adversarial Networks for Imbalanced Software Defects Prediction. *Applied Sciences*, *13*, 13319. https://doi.org/10.3390/app132413319

Amorim, L. B. V. de, Cavalcanti, G. D. C., & Cruz, R. M. O. (2022). The choice of scaling technique matters for classification performance. *Applied Soft Computing*, *133*, 109924. https://doi.org/10.1016/j.asoc.2022.109924

Assefa, S. A., Dervovic, D., Mahfouz, M., Tillman, R. E., Reddy, P., & Veloso, M. (2020). Generating synthetic data in finance: Opportunities, challenges and pitfalls. *Proceedings of the First ACM International Conference on AI in Finance*, 1–8. https://doi.org/10.1145/3383455.3422554

Aymaz, S. (2025). Unlocking the power of optimized data balancing ratios: A new frontier in tackling imbalanced datasets. *The Journal of Supercomputing*, *81*(2), 443. https://doi.org/10.1007/s11227-025-06919-2

Ba, H. (2019). Improving Detection of Credit Card Fraudulent Transactions using Generative Adversarial Networks. *ArXiv*, *1907.03355*. https://doi.org/10.48550/arXiv.1907.03355

Batista, G., Prati, R., & Monard, M.-C. (2004). A Study of the Behavior of Several Methods for Balancing machine Learning Training Data. *SIGKDD Explorations*, *6*, 20–29. https://doi.org/10.1145/1007730.1007735

Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, *54*(3), 1937–1967. https://doi.org/10.1007/s10462-020-09896-5

Bhakta, S. S., Ghosh, S., & Sadhukhan, B. (2023). Credit Card Fraud Detection Using Machine Learning: A Comparative Study of Ensemble Learning Algorithms. *2023 9th International Conference on Smart Computing and Communications (ICSCC)*, 296–301. https://doi.org/10.1109/ICSCC59169.2023.10335075

Biau, G., & Scornet, E. (2015). A Random Forest Guided Tour. *ArXiv, 1511.05741*. https://doi.org/10.48550/arXiv.1511.05741

Bocianowski, J., Dorota, Krysztofiak-Kaniewska, A., Matusiak, K., & Wiatrowska, B. (2024). *Comparison of Pearson's and Spearman's correlation coefficients values for selected traits of Pinus sylvestris L.* https://doi.org/10.21203/rs.3.rs-4380975/v1

Bojer, C. S., & Meldgaard, J. P. (2021). Kaggle forecasting competitions: An overlooked learning opportunity. *International Journal of Forecasting*, *37*(2), 587–603. https://doi.org/10.1016/j.ijforecast.2020.07.007

Bollepalli, B., Juvela, L., & Alku, P. (2017). Generative Adversarial Network-Based Glottal Waveform Model for Statistical Parametric Speech Synthesis. *Interspeech 2017*, 3394–3398. https://doi.org/10.21437/Interspeech.2017-1288

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Btoush, E., Zhou, X., Gururaian, R., Chan, K., & Tao, X. (2021). A Survey on Credit Card Fraud Detection Techniques in Banking Industry for Cyber Security. *2021 8th International Conference on Behavioral and Social Computing (BESC)*, 1–7. https://doi.org/10.1109/BESC53957.2021.9635559

Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2009). Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling Technique for Handling the Class Imbalanced Problem. In *Advances in Knowledge Discovery and Data Mining, Volume 5476* (Vol. 5476, p. 482). https://doi.org/10.1007/978-3-642-01307-2_43

Cai, Z., Xiong, Z., Xu, H., Wang, P., Li, W., & Pan, Y. (2021). Generative Adversarial Networks: A Survey Toward Private and Secure Applications. *ACM Comput. Surv.*, *54*(6), 132:1-132:38. https://doi.org/10.1145/3459992

Card Fraud Losses Worldwide — 2021. (2002, December). Nilson Report. https://nilsonreport.com/articles/card-fraud-losses-worldwide/

Charitou, C., Dragicevic, S., & Garcez, A. d'Avila. (2021). Synthetic Data Generation for Fraud Detection using GANs. *ArXiv, 2109.12546*. https://doi.org/10.48550/arXiv.2109.12546

Chawla, N., Japkowicz, N., & Kołcz, A. (2004). Editorial: Special Issue on Learning from Imbalanced Data Sets. *SIGKDD Explorations*, *6*, 1–6. https://doi.org/10.1145/1007730.1007733

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Chen, W., Yang, K., Yu, Z., Shi, Y., & Chen, C. L. P. (2024). A survey on imbalanced learning: Latest research, applications and future directions. Artificial Intelligence Review, 57(6), 137. https://doi.org/10.1007/s10462-024-10759-6

Chereddy, N. V., & Bolla, B. K. (2023). Evaluating the Utility of GAN Generated Synthetic Tabular Data for Class Balancing and Low Resource Settings. *Multi-disciplinary Trends in Artificial Intelligence* (pp. 48–59). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-36402-0_4

Chole, V., Mukherjee, A., Gaikwad, K., Pradhnya, Bagde, G., Mahule, R., Pawar, P., Gawai, P., Bagde, P., & Ijmtst, E. (2022). Revelation of Credit Card Fraud using Machine Learning Algorithm. *International Journal for Modern Trends in Science and Technology*, *8*, 89–94. https://doi.org/10.46501/IJMTST0806012

Correa Bahnsen, A., Aouada, D., & Ottersten, B. (2015). Example-dependent cost-sensitive decision trees. *Expert Systems with Applications*, *42*(19), 6609–6619. https://doi.org/10.1016/j.eswa.2015.04.042

Credit Card Fraud. (n.d.). Kaggle. Retrieved 9 February 2025, from https://www.kaggle.com/datasets/dhanushnarayananr/credit-card-fraud

Credit Card Fraud Detection. (n.d.). Kaggle. Retrieved 6 January 2025, from https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

Credit Card Transactions Fraud Detection Dataset. (n.d.). Kaggle. Retrieved 9 February 2025, from https://www.kaggle.com/datasets/kartik2112/fraud-detection

Cruz, B. (2024). 52 Million Americans Experienced Credit Card Fraud Last Year. *Security.Org*. https://www.security.org/digital-safety/credit-card-fraud-report/

CTGAN Model—SDV 0.18.0 documentation. (n.d.). Retrieved 9 February 2025, from https://sdv.dev/SDV/user_guides/single_table/ctgan.html

CTGANSynthesizer | Synthetic Data Vault. (2024, October 2). https://docs.sdv.dev/sdv/single-table-data/modeling/synthesizers/ctgansynthesizer

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*(8), 3784–3797. IEEE Transactions on Neural Networks and Learning Systems. https://doi.org/10.1109/TNNLS.2017.2736643

Dal Pozzolo, A., Johnson, R., Caelen, O., Waterschoot, S., Chawla, N. V., & Bontempi, G. (2014). Using HDDT to avoid instances propagation in unbalanced and evolving data streams. *2014 International Joint Conference on Neural Networks (IJCNN)*, 588–594. https://doi.org/10.1109/IJCNN.2014.6889638

DecisionTreeClassifier. (n.d.). Scikit-Learn. Retrieved 4 April 2025, from https://scikit-learn/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

Domingo, C., & Watanabe, O. (2000). MadaBoost: A Modification of AdaBoost. *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*, 180–189.

Dornadula, V. N., & Geetha, S. (2019). Credit Card Fraud Detection using Machine Learning Algorithms. *Procedia Computer Science*, *165*, 631–641. https://doi.org/10.1016/j.procs.2020.01.057

Douzas, G., & Bacao, F. (2018). Effective data generation for imbalanced learning using conditional generative adversarial networks. *Expert Systems with Applications*, *91*, 464–471. https://doi.org/10.1016/j.eswa.2017.09.030

Douzas, G., Bacao, F., & Last, F. (2018). Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Information Sciences*, *465*, 1–20. https://doi.org/10.1016/j.ins.2018.06.056

Duggal, P. (2022). *Predicting Credit Card Fraud Using Conditional Generative Adversarial Network* [Masters, Dublin, National College of Ireland]. https://norma.ncirl.ie/6114/

El Naby, A. A., El-Din Hemdan, E., & El-Sayed, A. (2021). Deep Learning Approach for Credit Card Fraud Detection. *2021 International Conference on Electronic Engineering (ICEEM)*, 1–5. https://doi.org/10.1109/ICEEM52022.2021.9480639

Ferreira, F., Lourenço, N., Cabral, B., & Fernandes, J. P. (2021). When Two are Better Than One: Synthesizing Heavily Unbalanced Data. *IEEE Access*, *9*, 150459–150469. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3126656

Fonseca, J., Douzas, G., & Bacao, F. (2021). Improving Imbalanced Land Cover Classification with K-Means SMOTE: Detecting and Oversampling Distinctive Minority Spectral Signatures. *Information*, *12*(7), Article 7. https://doi.org/10.3390/info12070266

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, 148–156.

Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, *55*(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, *29*(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., & Herrera, F. (2012). A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *42*(4), 463–484. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). https://doi.org/10.1109/TSMCC.2011.2161285

Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., & Suganthan, P. N. (2022). Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, *115*, 105151. https://doi.org/10.1016/j.engappai.2022.105151

Gao, Y., & Gao, F. (2010). Edited AdaBoost by weighted kNN. *Neurocomputing*, *73*(16), 3079–3088. https://doi.org/10.1016/j.neucom.2010.06.024

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks. *Advances in Neural Information Processing Systems*, *3*. https://doi.org/10.1145/3422622

Han, H., Wang, W.-Y., & Mao, B.-H. (2005). Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In *Adv Intell Comput* (Vol. 3644, p. 887). https://doi.org/10.1007/11538059_91

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284. IEEE Transactions on Knowledge and Data Engineering. https://doi.org/10.1109/TKDE.2008.239

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. https://doi.org/10.1016/j.eswa.2021.116429

IEEE-CIS Fraud Detection. (n.d.). Kaggle. Retrieved 7 January 2025, from https://www.kaggle.com/competitions/ieee-fraud-detection/data

Itoo, F., Meenakshi, & Singh, S. (2021). Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. *International Journal of Information Technology*, *13*(4), 1503–1511. https://doi.org/10.1007/s41870-020-00430-y

Jabeen, U., Singh, K., & Vats, S. (2023). Credit Card Fraud Detection Scheme Using Machine Learning and Synthetic Minority Oversampling Technique (SMOTE). *2023 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, 122–127. https://doi.org/10.1109/ICIRCA57980.2023.10220646

Jose, S., Devassy, D., & Antony, A. M. (2023). Detection of Credit Card Fraud Using Resampling and Boosting Technique. *2023 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA)*, 1–8. https://doi.org/10.1109/ACCTHPA57160.2023.10083376

Kaur, H., Pannu, H. S., & Malhi, A. K. (2019). A Systematic Review on Imbalanced Data Challenges in Machine Learning: Applications and Solutions. *ACM Comput. Surv.*, *52*(4), 79:1-79:36. https://doi.org/10.1145/3343440

Kaur, P., & Gosain, A. (2018). Comparing the Behavior of Oversampling and Undersampling Approach of Class Imbalance Learning by Combining Class Imbalance Problem with Noise. *ICT Based Innovations*, 23–30. https://doi.org/10.1007/978-981-10-6602-3_3

Kennedy, R. K. L., Villanustre, F., Khoshgoftaar, T. M., & Salekshahrezaee, Z. (2024). Synthesizing class labels for highly imbalanced credit card fraud detection data. *Journal of Big Data*, *11*(1), 38. https://doi.org/10.1186/s40537-024-00897-7

Khan, A. A., Chaudhari, O., & Chandra, R. (2024). A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation. *Expert Systems with Applications*, *244*, 122778. https://doi.org/10.1016/j.eswa.2023.122778

Khushi, M., Shaukat, K., Alam, T. M., Hameed, I. A., Uddin, S., Luo, S., Yang, X., & Reyes, M. C. (2021). A Comparative Performance Analysis of Data Resampling Methods on Imbalance

Medical Data. *IEEE Access*, *9*, 109960–109975. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3102399

Kong, J., Kowalczyk, W., Nguyen, D. A., Bäck, T., & Menzel, S. (2019). Hyperparameter Optimisation for Improving Classification under Class Imbalance. *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 3072–3078. https://doi.org/10.1109/SSCI44817.2019.9002679

Krivko, M. (2010). A hybrid model for plastic card fraud detection systems. *Expert Systems with Applications*, *37*(8), 6070–6076. https://doi.org/10.1016/j.eswa.2010.02.119

Kubát, M., & Matwin, S. (1997). *Addressing the Curse of Imbalanced Training Sets: One-Sided Selection*. International Conference on Machine Learning. https://www.semanticscholar.org/paper/Addressing-the-Curse-of-Imbalanced-Training-Sets%3A-Kub%C3%A1t-Matwin/ebc3914181d76c817f0e35f788b7c4c0f80abb07

Kulatilleke, G. K. (2017). Credit card fraud detection—Classifier selection strategy. *ArXiv,* 2208.11900. https://doi.org/10.48550/arXiv.2208.11900

Kulatilleke, G. K. (2022). Challenges and Complexities in Machine Learning based Credit Card Fraud Detection. *ArXiv, 2208.10943*. https://doi.org/10.48550/arXiv.2208.10943

Lee, H., & Yun, S. (2024). Strategies for Imputing Missing Values and Removing Outliers in the Dataset for Machine Learning-Based Construction Cost Prediction. *Buildings*, *14*(4), Article 4. https://doi.org/10.3390/buildings14040933

Lee, J., & Park, K. (2021). GAN-based imbalanced data intrusion detection system. *Personal and Ubiquitous Computing*, *25*(1), 121–128. https://doi.org/10.1007/s00779-019-01332-y

Leevy, J. L., Khoshgoftaar, T. M., & Hancock, J. (2022). Evaluating Performance Metrics for Credit Card Fraud Classification. *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*, 1336–1341. https://doi.org/10.1109/ICTAI56018.2022.00202

Lever, J., Krzywinski, M., & Altman, N. (2017). Points of Significance: Principal component analysis. *Nature Methods*, *14*, 641–642. https://doi.org/10.1038/nmeth.4346

Liu, H., & Lang, B. (2019). Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey. *Applied Sciences*, *9*(20), Article 20. https://doi.org/10.3390/app9204396

Lu, Y., Wu, S., Tai, Y.-W., & Tang, C.-K. (2018). Image Generation from Sketch Constraint Using Contextual GAN. *ArXiv, 1711.08972*. https://doi.org/10.48550/arXiv.1711.08972

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. https://doi.org/10.1016/j.patcog.2019.02.023

Mahboob Alam, T., Shaukat, K., Hameed, I., Luo, S., Sarwar, M., Shabbir, S., Li, J., & Khushi, M. (2020). An Investigation of Credit Card Default Prediction in the Imbalanced Datasets. *IEEE Access*, *8*. https://doi.org/10.1109/ACCESS.2020.3033784

Mahesh, K. P., Afrouz, S. A., & Areeckal, A. S. (2022). Detection of fraudulent credit card transactions: A comparative analysis of data sampling and classification techniques. *Journal of Physics: Conference Series*, *2161*(1), 012072. https://doi.org/10.1088/1742-6596/2161/1/012072

Mahmoudi, N., & Duman, E. (2015). Detecting credit card fraud by Modified Fisher Discriminant Analysis. *Expert Systems with Applications*, *42*(5), 2510–2516. https://doi.org/10.1016/j.eswa.2014.10.037

Mani, I., & Zhang, J. (2003). knn approach to unbalanced data distributions: A case study involving information extraction. *Proceedings of Workshop on Learning from Imbalanced Datasets*. https://www.academia.edu/12815658/knn_approach_to_unbalanced_data_distributions_A_case_study_involving_information_extraction

Mansourifar, H., & Shi, W. (2020). Deep Synthetic Minority Over-Sampling Technique*. ArXiv,* 2003.09788. https://doi.org/10.48550/arXiv.2003.09788

Matthews, B. W. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA) - Protein Structure*, *405*(2), 442–451. https://doi.org/10.1016/0005-2795(75)90109-9

Mienye, I. D., & Sun, Y. (2022). A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. *IEEE Access*, *10*, 99129–99149. IEEE Access. https://doi.org/10.1109/ACCESS.2022.3207287

Mienye, I. D., & Sun, Y. (2023). A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection. *IEEE Access*, *11*, 30628–30638. IEEE Access. https://doi.org/10.1109/ACCESS.2023.3262020

Mînăstireanu, E.-A., & Meșniță, G. (2020). Methods of Handling Unbalanced Datasets in Credit Card Fraud Detection. *BRAIN. Broad Research in Artificial Intelligence and Neuroscience*, *11*(1), Article 1.

Mirza, M., & Osindero, S. (2014). Conditional Generative Adversarial Nets. *ArXiv, 1411.1784.* https://doi.org/10.48550/arXiv.1411.1784

Muaz, A., Jayabalan, M., & Thiruchelvam, V. (2020). A Comparison of Data Sampling Techniques for Credit Card Fraud Detection. *International Journal of Advanced Computer Science and Applications*, *11*(6). https://doi.org/10.14569/IJACSA.2020.0110660

Mullick, S. S., Datta, S., & Das, S. (2020). Generative Adversarial Minority Oversampling. *ArXiv, 1903.09730*. https://doi.org/10.48550/arXiv.1903.09730

Nallabothula, H. (2022). *An analysis of Machine learning Algorithms for Detection of Credit Card Fraud*.

Natekin, A., & Knoll, A. (2013). Gradient Boosting Machines, A Tutorial. *Frontiers in Neurorobotics*, *7*, 21. https://doi.org/10.3389/fnbot.2013.00021

Ndung'u, R. (2022). Data Preparation For Machine Learning Modelling. *International Journal of Computer Applications Technology and Research*, *11*, 231–235. https://doi.org/10.7753/IJCATR1106.1008

Ngai, E. W. T., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, *50*(3), 559–569. https://doi.org/10.1016/j.dss.2010.08.006

Ngwenduna, K. S., & Mbuvha, R. (2021). Alleviating Class Imbalance in Actuarial Applications Using Generative Adversarial Networks. *Risks*, *9*(3), Article 3. https://doi.org/10.3390/risks9030049

Niu, X., Wang, L., & Yang, X. (2019). A Comparison Study of Credit Card Fraud Detection: Supervised versus Unsupervised. *ArXiv, 1904.10604*. https://doi.org/10.48550/arXiv.1904.10604

Oza, N. C. (2004). AveBoost2: Boosting for Noisy Data. *Multiple Classifier Systems*, 31–40. https://doi.org/10.1007/978-3-540-25966-4_3

Patil, T. (2021). *Credit Card Fraud Detection Using Conditional Tabular Generative Adversarial Networks (CT-GAN) and Supervised Machine Learning Techniques* [Masters, Dublin, National College of Ireland]. https://norma.ncirl.ie/5209/

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. Journal of Machine learning research, 12:2825–2830.

Poslavskaya, E., & Korolev, A. (2023). Encoding categorical data: Is there yet anything 'hotter' than one-hot encoding? *ArXiv,* 2312.16930). https://doi.org/10.48550/arXiv.2312.16930

Prati, R., Batista, G., & Monard, M.-C. (2004). *Learning with Class Skews and Small Disjuncts* (p. 306). https://doi.org/10.1007/978-3-540-28645-5_30

Priyatno, A., & Widiyaningtyas, T. (2024). A SYSTEMATIC LITERATURE REVIEW: RECURSIVE FEATURE ELIMINATION ALGORITHMS. *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, *9*, 196–207. https://doi.org/10.33480/jitk.v9i2.5015

Provost, F., Fawcett, T., & Kohavi, R. (1998, July 24). The Case against Accuracy Estimation for Comparing Induction Algorithms. *International Conference on Machine Learning*. https://www.semanticscholar.org/paper/The-Case-against-Accuracy-Estimation-for-Comparing-Provost-Fawcett/77703a2783f64dfceb638aa9eebd9c9c501bb835

Rahman, Md. S., & Yogarajah, P. (2022). Evaluating the Performance of Common Machine Learning Classifiers using various Validation Methods. *Artificial Intelligence, Soft Computing and Applications*, 45–51. https://doi.org/10.5121/csit.2022.122304

Rätsch, G., Onoda, T., & Müller, K. R. (1999). Regularizing AdaBoost. *Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems II*, 564–570.

Roy, A., Sun, J., Mahoney, R., Alonzi, L., Adams, S., & Beling, P. (2018). Deep learning detecting fraud in credit card transactions. *2018 Systems and Information Engineering Design Symposium (SIEDS)*, 129–134. https://doi.org/10.1109/SIEDS.2018.8374722

Sulaiman, S., Nadher, I., & Hameed, S. M. (2024). Credit Card Fraud Detection Challenges and Solutions: A Review. *Iraqi Journal of Science*, 2287–2303. https://doi.org/10.24996/ijs.2024.65.4.42

Sahin, Y., Bulkan, S., & Duman, E. (2013). A cost-sensitive decision tree approach for fraud detection. *Expert Systems with Applications*, *40*(15), 5916–5923. https://doi.org/10.1016/j.eswa.2013.05.021

Sauber-Cole, R., & Khoshgoftaar, T. M. (2022). The use of generative adversarial networks to alleviate class imbalance in tabular data: A survey. *Journal of Big Data*, *9*(1), 98. https://doi.org/10.1186/s40537-022-00648-6

Sethia, A., Patel, R., & Raut, P. (2018). Data Augmentation using Generative models for Credit Card Fraud Detection. *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, 1–6. https://doi.org/10.1109/CCAA.2018.8777628

Shi, S., Li, J., Zhu, D., Yang, F., & Xu, Y. (2023). A hybrid imbalanced classification model based on data density. *Information Sciences*, *624*, 50–67. https://doi.org/10.1016/j.ins.2022.12.046

Siegel, A. F., & Wagner, M. R. (2022). Chapter 15 - ANOVA: Testing for Differences Among Many Samples and Much More. In A. F. Siegel & M. R. Wagner (Eds.), *Practical Business*

Statistics (Eighth Edition) (pp. 485–510). Academic Press. https://doi.org/10.1016/B978-0-12-820025-4.00015-4

Sikri, A., Singh, N. P., & Dalal, S. (2023). Chi-Square Method of Feature Selection: Impact of Pre-Processing of Data. International Journal of Intelligent Systems and Applications in Engineering, 11(3s), Article 3s.

Sinap, V. (2024). Comparative analysis of machine learning techniques for credit card fraud detection: Dealing with imbalanced datasets. *Turkish Journal of Engineering*, *8*(2), 196–208. https://doi.org/10.31127/tuje.1386127

Singh, A., Ranjan, R., & Tiwari, A. (2021). Credit Card Fraud Detection under Extreme Imbalanced Data: A Comparative Study of Data-level Algorithms. *Journal of Experimental & Theoretical Artificial Intelligence*, *34*, 1–28. https://doi.org/10.1080/0952813X.2021.1907795

Soh, W. W., & Yusuf, R. M. (2019). Predicting Credit Card Fraud on a Imbalanced Data. *International Journal of Data Science and Advanced Analytics*, *1*(1), Article 1. https://doi.org/10.69511/ijdsaa.v1i1.3

Strelcenia, E., & Prakoonwit, S. (2023). A Survey on GAN Techniques for Data Augmentation to Address the Imbalanced Data Issues in Credit Card Fraud Detection. *Machine Learning and Knowledge Extraction*, *5*(1), Article 1. https://doi.org/10.3390/make5010019

Synthetic Financial Datasets For Fraud Detection. (n.d.). Kaggle. Retrieved 9 February 2025, from https://www.kaggle.com/datasets/ealaxi/paysim1

Tarekegn, A. N., Giacobini, M., & Michalak, K. (2021). A review of methods for imbalanced multi-label classification. *Pattern Recognition*, *118*, 107965. https://doi.org/10.1016/j.patcog.2021.107965

Tomek, I. (1976). Two Modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*(11), 769–772. IEEE Transactions on Systems, Man, and Cybernetics. https://doi.org/10.1109/TSMC.1976.4309452

Vairetti, C., Assadi, J. L., & Maldonado, S. (2024). Efficient hybrid oversampling and intelligent undersampling for imbalanced big data classification. *Expert Systems with Applications*, *246*, 123149. https://doi.org/10.1016/j.eswa.2024.123149

Vanini, P., Rossi, S., Zvizdic, E., & Domenig, T. (2023). Online payment fraud: From anomaly detection to risk management. *Financial Innovation*, *9*(1), 66. https://doi.org/10.1186/s40854-023-00470-w

Veigas, K. C., Regulagadda, D. S., & Kokatnoor, S. A. (2021). Optimized Stacking Ensemble (OSE) for Credit Card Fraud Detection using Synthetic Minority Oversampling Model. *Indian*

Journal of Science and Technology, *14*(32), 2607–2615. https://doi.org/10.17485/IJST/v14i32.807

Wibowo, P., & Fatichah, C. (2021). An in-depth performance analysis of the oversampling techniques for high-class imbalanced dataset. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, *7*(1), 63–71. https://doi.org/10.26594/register.v7i1.2206

Wilson, D. L. (1972). Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-2*(3), 408–421. https://doi.org/10.1109/TSMC.1972.4309137

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*(2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

Xie, Y., Li, A., Gao, L., & Liu, Z. (2021). A Heterogeneous Ensemble Learning Model Based on Data Distribution for Credit Card Fraud Detection. *Wireless Communications and Mobile Computing*, *2021*, 1–13. https://doi.org/10.1155/2021/2531210

Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling Tabular data using Conditional GAN. *ArXiv,* 1907.00503. https://doi.org/10.48550/arXiv.1907.00503

Xu, L., & Veeramachaneni, K. (2018). Synthesizing Tabular Data using Generative Adversarial Networks. *ArXiv*, 1811.11264. https://doi.org/10.48550/arXiv.1811.11264

Yen, S.-J., & Lee, Y.-S. (2009). Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, *36*(3, Part 1), 5718–5727. https://doi.org/10.1016/j.eswa.2008.06.108

Zhou, Z., Zhang, B., Lv, Y., Shi, T., & Chang, F. (2019). Data Augment in Imbalanced Learning Based on Generative Adversarial Networks. *Neural Information Processing*, 21–30. https://doi.org/10.1007/978-3-030-36808-1_3

Zimek, A., Schubert, E., & Kriegel, H.-P. (2012). A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *5*(5), 363–387. https://doi.org/10.1002/sam.11161

Zou, H. (2021). *Analysis of Best Sampling Strategy in Credit Card Fraud Detection Using Machine Learning* (p. 44). https://doi.org/10.1145/3460179.3460186

APPENDIX A

The following table shows the output of the experiments done for all datasets, concerning all imbalanced learning techniques tested.

Table 6 – Experimental Results

European Dataset						
Method	**Classifier**	**F1**	**Recall**	**Precision**	**ROC-AUC**	**PR-AUC**
None	LR	0.66	0.53	0.88	0.97	0.75
	RF	0.89	0.81	0.98	0.94	0.85
	XGBoost	0.88	0.81	0.97	0.98	0.85
SMOTE	LR	0.10	0.91	0.05	0.98	0.73
	RF	0.88	0.84	0.93	0.97	0.85
	XGBoost	0.70	0.85	0.60	0.99	0.86
ROS	LR	0.11	0.91	0.06	0.98	0.72
	RF	0.88	0.80	0.97	0.95	0.86
	XGBoost	0.88	0.82	0.95	0.98	0.86
Borderline-SMOTE	LR	0.34	0.86	0.21	0.98	0.70
	RF	0.87	0.78	0.97	0.95	0.84
	XGBoost	0.85	0.80	0.91	0.98	0.85
ADASYN	LR	0.04	0.92	0.02	0.98	0.66
	RF	0.88	0.83	0.93	0.97	0.85
	XGBoost	0.66	0.84	0.55	0.99	0.84
SMOTE-ENN	LR	0.10	0.92	0.05	0.98	0.72
	RF	0.82	0.85	0.81	0.98	0.84
	XGBoost	0.61	0.85	0.47	0.99	0.85
SMOTE-Tomek	LR	0.11	0.91	0.06	0.98	0.73
	RF	0.88	0.84	0.93	0.98	0.86
	XGBoost	0.63	0.83	0.50	0.99	0.84
K-means SMOTE	LR	0.82	0.76	0.89	0.96	0.73
	RF	0.87	0.80	0.95	0.95	0.86
	XGBoost	0.87	0.82	0.92	0.98	0.86
CT-GAN	LR	0.76	0.81	0.71	0.97	0.75
	RF	0.84	0.84	0.84	0.95	0.84
	XGBoost	0.86	0.80	0.93	0.99	0.85
CT-GAN Synthesizer	LR	0.83	0.80	0.86	0.95	0.74
	RF	0.86	0.84	0.88	0.95	0.84
	XGBoost	0.86	0.82	0.90	0.99	0.86

Simulated Dataset						
Method	**Classifier**	**F1**	**Recall**	**Precision**	**ROC-AUC**	**PR-AUC**
None	LR	0.15	0.75	0.08	0.86	0.20
	RF	0.84	0.73	0.98	0.99	0.90
	XGBoost	0.89	0.83	0.95	1.00	0.94
SMOTE	LR	0.14	0.75	0.08	0.86	0.20
	RF	0.83	0.78	0.88	0.99	0.87
	XGBoost	0.80	0.85	0.75	1.00	0.90
ROS	LR	0.15	0.75	0.08	0.86	0.20
	RF	0.86	0.78	0.96	0.99	0.92
	XGBoost	0.59	0.97	0.42	1.00	0.94
Borderline-SMOTE	LR	0.18	0.74	0.10	0.86	0.20
	RF	0.82	0.78	0.86	0.99	0.86
	XGBoost	0.81	0.84	0.79	0.99	0.89
ADASYN	LR	0.14	0.75	0.08	0.86	0.20
	RF	0.82	0.77	0.88	0.99	0.87
	XGBoost	0.79	0.85	0.73	1.00	0.89
SMOTE-ENN	LR	0.14	0.75	0.08	0.86	0.20
	RF	0.82	0.79	0.84	0.99	0.86
	XGBoost	0.76	0.87	0.68	1.00	0.90
SMOTE-Tomek	LR	0.14	0.75	0.08	0.86	0.20
	RF	0.82	0.78	0.85	0.99	0.87
	XGBoost	0.77	0.87	0.69	1.00	0.89
K-means SMOTE	LR	0.07	0.48	0.03	0.76	0.10
	RF	0.76	0.63	0.94	0.98	0.85
	XGBoost	0.85	0.80	0.90	1.00	0.90
CT-GAN	LR	0.13	0.75	0.07	0.84	0.19
	RF	0.80	0.69	0.96	0.98	0.88
	XGBoost	0.88	0.81	0.96	1.00	0.93
CT-GAN Synthesizer	LR	0.17	0.59	0.10	0.83	0.17
	RF	0.72	0.72	0.71	0.98	0.78
	XGBoost	0.85	0.77	0.94	1.00	0.90

IEEE-CIS Dataset						
Method	**Classifier**	**F1**	**Recall**	**Precision**	**ROC-AUC**	**PR-AUC**
None	LR	0.16	0.68	0.09	0.78	0.17
	RF	0.61	0.46	0.91	0.92	0.69
	XGBoost	0.56	0.41	0.88	0.92	0.63
SMOTE	LR	0.16	0.68	0.09	0.78	0.17
	RF	0.63	0.52	0.81	0.91	0.66
	XGBoost	0.54	0.46	0.64	0.89	0.54
ROS	LR	0.16	0.68	0.09	0.78	0.17
	RF	0.65	0.52	0.85	0.92	0.69
	XGBoost	0.34	0.80	0.21	0.93	0.59
Borderline-SMOTE	LR	0.16	0.66	0.09	0.78	0.17
	RF	0.62	0.51	0.81	0.91	0.65
	XGBoost	0.53	0.42	0.70	0.90	0.54
ADASYN	LR	0.15	0.70	0.09	0.78	0.17
	RF	0.62	0.51	0.80	0.91	0.65
	XGBoost	0.53	0.43	0.69	0.89	0.54
SMOTE-ENN	LR	0.15	0.69	0.09	0.78	0.17
	RF	0.61	0.53	0.72	0.91	0.63
	XGBoost	0.53	0.50	0.58	0.89	0.54
SMOTE-Tomek	LR	0.16	0.68	0.09	0.78	0.16
	RF	0.62	0.50	0.82	0.91	0.65
	XGBoost	0.54	0.46	0.64	0.89	0.55
K-means SMOTE	LR	0.07	0.08	0.07	0.63	0.06
	RF	0.62	0.49	0.84	0.91	0.66
	XGBoost	0.53	0.39	0.85	0.91	0.59
CT-GAN	LR	0.17	0.57	0.10	0.75	0.16
	RF	0.61	0.46	0.90	0.92	0.69
	XGBoost	0.54	0.39	0.88	0.91	0.60
CT-GAN Synthesizer	LR	0.17	0.59	0.10	0.76	0.16
	RF	0.60	0.45	0.89	0.92	0.68
	XGBoost	0.54	0.39	0.87	0.91	0.60

Synthetic Dataset						
Method	**Classifier**	**F1**	**Recall**	**Precision**	**ROC-AUC**	**PR-AUC**
None	LR	0.45	0.31	0.78	0.96	0.44
	RF	0.86	0.77	0.98	0.99	0.91
	XGBoost	0.87	0.79	0.97	1.00	0.93
SMOTE	LR	0.05	0.93	0.03	0.99	0.56
	RF	0.71	0.97	0.56	1.00	0.96
	XGBoost	0.32	0.99	0.19	1.00	0.94
ROS	LR	0.04	0.91	0.02	0.98	0.50
	RF	0.87	0.81	0.94	0.99	0.92
	XGBoost	0.37	0.99	0.23	1.00	0.94
Borderline-SMOTE	LR	0.04	0.92	0.02	0.98	0.50
	RF	0.79	0.92	0.69	1.00	0.93
	XGBoost	0.60	0.93	0.45	1.00	0.92
ADASYN	LR	0.03	0.96	0.02	0.99	0.55
	RF	0.60	0.97	0.43	1.00	0.95
	XGBoost	0.27	1.00	0.16	1.00	0.93
SMOTE-ENN	LR	0.04	0.91	0.02	0.98	0.52
	RF	0.67	0.97	0.51	1.00	0.95
	XGBoost	0.29	0.99	0.17	1.00	0.93
SMOTE-Tomek	LR	0.04	0.91	0.02	0.98	0.50
	RF	0.71	0.97	0.56	1.00	0.95
	XGBoost	0.30	0.99	0.18	1.00	0.94
K-means SMOTE	LR	0.10	0.69	0.05	0.96	0.48
	RF	0.85	0.82	0.87	0.99	0.90
	XGBoost	0.83	0.80	0.85	1.00	0.90
CT-GAN	LR	0.02	0.92	0.01	0.97	0.51
	RF	0.86	0.77	0.98	0.99	0.91
	XGBoost	0.88	0.80	0.97	1.00	0.93
CT-GAN Synthesizer	LR	0.04	0.89	0.02	0.98	0.48
	RF	0.43	0.81	0.30	0.99	0.74
	XGBoost	0.31	0.79	0.19	0.99	0.71

Unnamed Dataset						
Method	**Classifier**	**F1**	**Recall**	**Precision**	**ROC-AUC**	**PR-AUC**
None	LR	0.70	0.94	0.56	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	0.99	0.99	1.00	1.00
SMOTE	LR	0.70	0.95	0.56	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	1.00	0.98	1.00	1.00
ROS	LR	0.70	0.94	0.56	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	1.00	0.98	1.00	1.00
Borderline-SMOTE	LR	0.72	0.95	0.58	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	1.00	1.00	0.99	1.00	1.00
ADASYN	LR	0.70	0.97	0.54	0.98	0.71
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	1.00	0.98	1.00	1.00
SMOTE-ENN	LR	0.70	0.95	0.56	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.98	1.00	0.97	1.00	1.00
SMOTE-Tomek	LR	0.70	0.95	0.56	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	1.00	0.97	1.00	1.00
K-means SMOTE	LR	0.69	0.56	0.90	0.96	0.78
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	0.99	1.00	1.00	1.00
CT-GAN	LR	0.69	0.95	0.55	0.98	0.76
	RF	1.00	1.00	1.00	1.00	1.00
	XGBoost	0.99	0.99	1.00	1.00	1.00
CT-GAN Synthesizer	LR	0.70	0.92	0.57	0.98	0.72
	RF	0.97	1.00	0.94	1.00	1.00
	XGBoost	0.96	1.00	0.93	1.00	1.00

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa