TAYLOR'S UNIVERSITY

Wisdom • Integrity • Excellence

MASTER OF APPLIED COMPUTING

CAPSTONE 2

Credit Card Fraud Detection with Hybrid Machine Learning Models

By

Ooi Jin Jie

School of Computer Science
and Engineering

Taylor's University

October 2023

Table of Contents

Definition of Terms

LR – Logistic Regression

RF – Random Forest

SVM – Support Vector Machine

XGB – XGBoost

ADA - AdaBoost

EDA - Exploratory data analysis

PCA - Principal component analysis

GPU – Graphics Processing Unit

TPU – Tensor Processing Unit

Abstract

Credit card fraud is an ongoing issue worldwide, affecting both developed and developing countries. Increased fraudulent activities, along with evolving tactics used by fraudsters, contribute to the problem. Credit card fraud leads to financial losses for individuals and businesses, and it undermines trust in financial systems. Financial institutions and authorities face significant challenges in detecting and preventing fraud in real-time.

To address the problem of credit card fraud, extensive research has been conducted in the field of machine learning and data analytics. Many studies have explored the application of machine learning techniques for fraud detection, each building upon the insights of previous research. Various machine learning models have been employed, including Logistic Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), XGBoost and AdaBoost, each with its strengths and weaknesses.

In this particular project, the focus is on developing hybrid machine learning models for credit card fraud detection. The project aims to answer critical questions related to fraud patterns, anomalies, and early detection methods. The insights gained from data analysis, coupled with the best-performing machine learning model.

By applying these advanced machine learning models, the project seeks to enhance credit card fraud detection, reduce financial losses, and bolster security measures in the financial industry. This approach benefits both financial institutions and credit cardholders by providing a robust defence against fraudulent activities.

Chapter 1: Introduction

1.1 Introduction

Credit card fraud is a prevalent and costly problem for financial institutions, merchants, and consumers worldwide. Fraudulent activities involving credit cards can lead to substantial financial losses, damage to a business's reputation, and inconvenience for cardholders. To mitigate these risks, credit card fraud detection systems have become a critical component of the payment processing ecosystem.

Starting from 2013, there has been a significant increase in worldwide losses attributed to payment fraud. These losses surged from USD 13.70 billion in 2013 to USD 32.96 billion in 2023, and it is projected to become an even more severe global issue, with anticipated costs reaching USD 38.50 billion by 2027. People in their 30s are the most vulnerable to credit card fraud while 40s are 2nd most vulnerable.

1.2 Project Background

Financial fraud has become an increasingly concerning issue with extensive outcome affecting various sectors such as government, corporate organizations, and the finance industry. In today's digital era, the reliance on internet technology has led to a surge in credit card transactions. However, this surge has also accelerated the occurrence of credit card fraud, both in online and offline transactions.

As credit card transactions have gained widespread acceptance as a mode of payment, there has been a growing emphasis on employing modern computational methodologies to address the issue of credit card fraud. Numerous fraud detection solutions and software applications have been developed to combat fraud in various sectors, including credit card companies, retail, e-commerce, insurance, and other industries.

1.3 Problem Statement

The increase usage of e-commerce platforms has led to a growing reliance on online services by individuals and financial institutions, resulting in a substantial surge in credit card fraud cases. These fraudulent credit card transactions cause significant financial losses, the development of an effective fraud detection system to mitigate these losses for both customers and financial companies.

Extensive research has been conducted to devise models and methods for preventing and detecting credit card frauds. Some credit card fraud datasets exhibit data imbalance issues. An effective fraud detection system should possess the capability to accurately identify fraudulent transactions and enable real-time detection.

Credit card fraud can be divided into three main groups.: traditional card-related frauds (including application fraud, stolen cards, account takeover, and counterfeit cards), merchant-related frauds (involving merchant collusion and triangulation schemes), and internet-related frauds (such as site cloning, credit card generators, and fraudulent merchant websites).

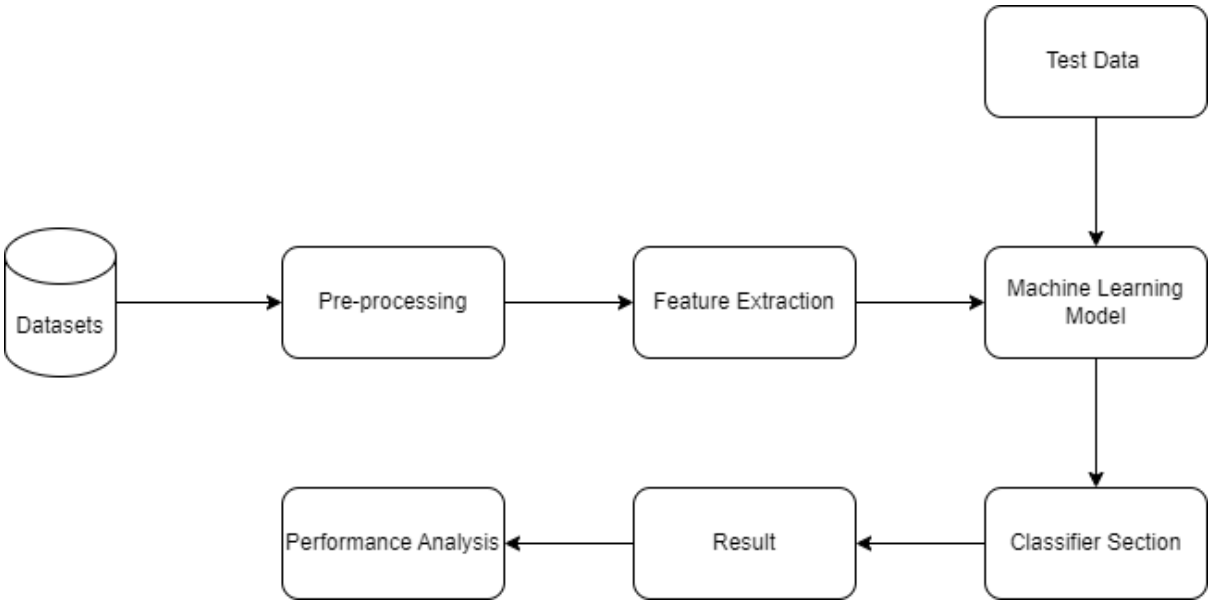
1.4 Justification of Research

The importance of this project lies in its capacity to offer insights into the existing status of credit card fraud and forecast its future trends. This data can empower financial institutions and security organizations to make more data-driven decisions and assess the outcomes of their actions regularly, whether it's implementing enhanced security measures or reporting suspicious activities to law enforcement. Our data will reflect the impact of these initiatives as improvements in fraud detection or potential fraud incidents surface, necessitating swift action. Additionally, this project has the potential to serve as a model for other financial fraud monitoring and prediction systems, inspiring their development.

1.5 Methodology

During the credit card fraud detection project, we will employ the following methods:

1. Data Collection: We will collect historical credit card transaction data, including transaction timestamps, amounts, merchant details, and cardholder information, from financial institutions.
2. Data Preprocessing: Data will undergo preprocessing to handle missing values, outliers, and duplicates. Techniques such as data imputation and feature scaling will be applied.
3. Exploratory Data Analysis (EDA): EDA will involve descriptive statistics and data visualization to gain insights into transaction patterns and trends.
4. Feature Engineering: Relevant features will be selected or engineered to enhance model performance. Feature importance analysis will guide feature selection.
5. Machine Learning Models: We will develop machine learning models, including decision trees, random forests, and neural networks, to learn patterns of fraudulent transactions from historical data.
6. Model Evaluation: Performance metrics such as accuracy, precision, recall, and F1-score will be used to evaluate model performance through cross-validation techniques.



1.6 Research Questions

Research questions for a credit card fraud detection project should address specific aspects of fraud detection, prevention, or understanding fraud patterns.

1. What are the effective strategies for handling imbalanced datasets in credit card fraud detection?
2. How effective are hybrid models to identify fraudulent credit card transactions compared to traditional models?

1.7 Research Objectives

Research objectives for a credit card fraud detection project are specific, measurable goals that outline what the research aims to achieve. These objectives guide the project and help researchers focus their efforts effectively.

1. To identify strategies for handling imbalanced datasets in credit card fraud detection.
2. To analyse and predict credit card fraud by using hybrids and traditional machine learning models and compare them.

1.8 Project Scope

Credit card fraud can take place in various forms and locations, including online transactions, point-of-sale terminals, and ATM withdrawals. This project targets credit card fraud and aims to create different machine learning models to detect credit card fraud. The dataset is collected from Kaggle and analysed during a research collaboration of Worldline and the Machine Learning Group of ULB on big data mining and fraud detection. It's essential to consider the potential influence on fraud activities, involving the collection of several years of transaction data to discern any recurring trends or patterns. This project will also explore methods to address highly imbalance dataset, such as, oversampling, undersampling.

1.9 Project Contributions

The primary contributions in this credit card fraud detection project lies in the creation of a machine learning model designed to detect fraud in credit card transactions. Develop and explore the utilization of various hybrid models on same dataset and identify the most effective hybrid model by evaluating their performance. Conducting experiments on different datasets, resulting to the conclusion that hybrid machine learning models will always give better outcome than traditional machine learning models. This report will encompass the secondary research conducted and the methodologies applied to generate these outcomes.

1.10 Reports Outline

As depicted in Figure 1, this section provides an overview of the three chapters within this report, offering a visual representation of the report's structure, along with concise descriptions of the primary subsections.

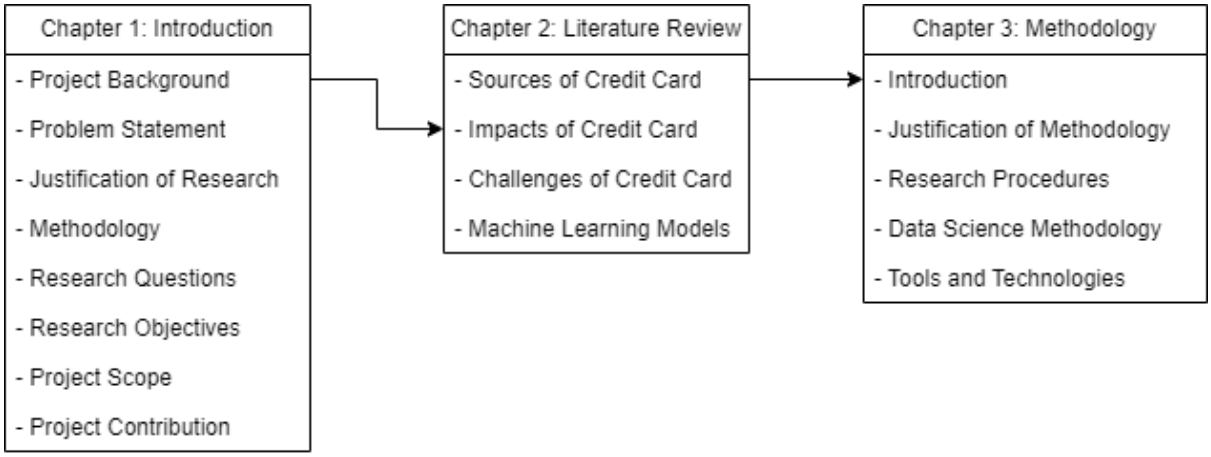


Figure 1

Chapter 1 covers project background that introduces the increasing concerning issues of credit card fraud. The problem statement talked about the outcomes of the credit card fraud if the ongoing issues still haven't been resolved yet. Then it will go over to justification of research, overview of the methodology, defining research questions and research objectives. Lastly, the scope and contribution of the project.

Chapter 2 covers literature review by looking into Domain Research and Technical Research. Domain research covers how credit card fraud happen, where is it come from and why. It will look into existing data collection methods and evaluate and analyse how to detect credit card fraud. Technical research will cover on technical side, such as usage of machine learning algorithms and dataset collection from the sources. Lastly, it will also cover similar system like dashboard to detect credit card fraud.

Chapter 3 will cover the usage of methodology for this project. Here will describe the justification of methodology and research procedures. It will show the process of data science methodology on how to process the credit card fraud data. Lastly, it will show the tools and technologies that will be used for this project.

Chapter 2: Literature Review

Due to the significant challenges posed by credit card fraud in the financial sector, numerous financial industries have invested substantial resources and assembled teams of experts to create fraud detection systems. Many researchers have been diligently addressing issues encountered during the development of such systems, including problems like class imbalance, class overlap, and the dynamic nature of fraudulent behaviours.

2.1 Sources of Credit Card Fraud

One primary source of credit card fraud is the theft of physical cards. Criminals may steal credit cards from unsuspecting persons' wallets or purses, gaining access to the card's details and using it for unauthorized transactions.

Lost or misplaced cards also present a risk. If cardholders misplace their credit cards, there's a window of opportunity for someone else to find and misuse them before the loss is reported.

Data breaches are also a major concern, where hackers infiltrate businesses, retailers, or financial institutions to steal cardholder data. This stolen data is often sold on the dark web or used for fraudulent transactions.

Card Not Present (CNP) fraud occurs in online and phone transactions. Fraudsters use stolen card details to make purchases without physically possessing the card.

2.2 Impacts of Credit Card Fraud

For businesses, credit card fraud can damage their reputation and erode customer trust. Customers may hesitate to engage in transactions with a company that has experienced a security breach, resulting in lost revenue and customer churn.

One of the most immediate and tangible consequences of credit card fraud is the financial burden it places on victims. When unauthorized transactions occur, individuals often find themselves held responsible for the fraudulent charges until the issue is resolved. This can result in significant financial distress, as victims must cover these expenses while awaiting reimbursement.

Moreover, credit card fraud can extend to identity theft, leading to additional financial and personal harm. Stolen information may be used to commit various fraudulent activities, further complicating the victim's life.

2.3 Challenges of Credit Card Fraud

Using machine learning to enhance the identification of fraudulent transactions is a common practice among businesses and financial institutions. Nevertheless, the difficulty of fraud detection can create obstacles for machine learning for various reasons.

1. The data distribution is skewed significantly due to the limited number of fraudulent transactions.
2. The data is constantly changing and evolving as time progresses.
3. A shortage of real-world datasets exists because of privacy-related issues.

2.4 Machine Learning Models

Machine learning models have transformed credit card fraud detection by providing efficient and effective tools to combat fraudulent activities in real-time. These models analyse vast amounts of transaction data to identify suspicious patterns, enabling financial institutions and businesses to protect their customers and mitigate financial losses. Below machine learning models will be used to detect credit card fraud.

2.4.1 Logistic Regression

Logistic regression is a machine learning-based classification method employed for predicting categorical outcomes based on a set of independent variables. This technique uses the logistic function's curve to represent the probability of events such as whether cells are carcinogenic, whether a rodent is obese based on its body weight, and so forth. In logistic regression, a threshold value is utilized to quantify the probability, resulting in outcomes of either 0 or 1.

Typically, values greater than or equal to the threshold are classified as 1, while data rounded down to values equivalent to the threshold are typically classified as 0. This method is valuable for making binary predictions and is widely used in various fields for tasks like disease classification, fraud detection, and more, where the goal is to determine the likelihood of an event occurring based on given factors.

2.4.2 Random Forest

Random Forest is a machine learning algorithm built upon the foundation of decision trees. It is widely utilized for addressing both regression and classification problems, making it a versatile tool in the field of data analysis. One of its key strengths is its ability to provide highly accurate predictions, even when dealing with extensive datasets.

Random Forest employs an ensemble learning approach, combining multiple individual classifiers to tackle complex problems effectively. It predicts outcomes by aggregating the mean results from the various decision trees in the forest. Increasing the number of trees typically enhances the precision of the predictions. Moreover, Random Forest overcomes

several limitations associated with standalone decision tree algorithms, It effectively reduces overfitting issues and subsequently increases prediction accuracy.

However, Random Forest does have certain limitations, particularly in the context of regression problems. Training Random Forests on a wide range of datasets can be challenging, and it may not perform optimally in all regression scenarios. However, Random Forest continues to be a valuable asset in the field of machine learning due to its ability to provide precise outcomes in diverse applications.

2.4.3 Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a versatile machine learning technique employed for both classification and regression analysis across various problem domains. In the context discussed, researchers have applied SVM to analyse customer credit card usage patterns. They collected data on customer payment behaviours from datasets. SVM is used to classify these consumer patterns into two categories: fraudulent or non-fraudulent transactions.

SVM demonstrates its effectiveness by providing accurate results, particularly when working with a smaller subset of features from the dataset. However, challenges arise when dealing with larger datasets, typically those exceeding 100,000 records. In such cases, SVM's performance tends to decline, making it less effective, especially when dealing with real-time processing demands. The limitations become apparent as the dataset size grows, making SVM less suitable for handling large volumes of data in real-time credit card fraud detection scenarios.

2.4.4 AdaBoost

The AdaBoost classifier is a meta-estimator employed in machine learning. It functions by training an initial classifier on the original dataset and then sequentially training additional copies of the classifier on the same dataset. What sets AdaBoost apart is that it adjusts the weights of instances that were previously classified incorrectly during each iteration. This adjustment process gives more importance to those instances that posed challenges in the past, allowing subsequent classifiers to focus on these tricky cases. AdaBoost is a widely used ensemble learning method that combines the predictions of various weak classifiers to create a strong classifier, and it is particularly effective for tasks like classification and object detection.

2.4.5 XGBoost

Extreme Gradient Boosting (XGBoost) is a tree-based algorithm that falls under the supervised learning of Machine Learning. It is a versatile algorithm capable of handling both classification and regression problems. XGBoost is known for its high performance and has become a popular choice among data scientists and machine learning practitioners due to its effectiveness in various applications.

2.4.6 Hybrid Models

In a study conducted by the authors, a hybrid model was introduced to enhance the accuracy of fraud detection by combining various models. The researchers established multiple criteria for computing outlier scores at different levels of granularity, ranging from highly specific card-specific outlier scores to broader global outlier scores. Subsequently, they assessed the effectiveness of these outlier scores when integrated as features within a supervised learning approach.

Unfortunately, the results of the study were not particularly persuasive when it came to both local and global outlier detection methods. However, it's worth noting that the model demonstrated more significant promise in terms of the Area Under the Precision-Recall Curve (AUC-PRC), indicating its potential to improve precision in fraud detection.

In this project, I will combine different types of machine learning models and compare their performance result to see which one is the best. Below is the example of hybrid machine learning models:

1. AdaBoost + XGBoost
2. Random Forest + Support Vector Machine
3. XGBoost + Random Forest
4. AdaBoost + Random Forest

Chapter 3: Methodology

In this chapter, we will look into data science methodology. Data science and AI methodology will outline our data cleaning process and the development of machine learning algorithms. Furthermore, this chapter will provide a comprehensive overview of the proposed hybrid machine learning models.

3.0 Data Science Methodology

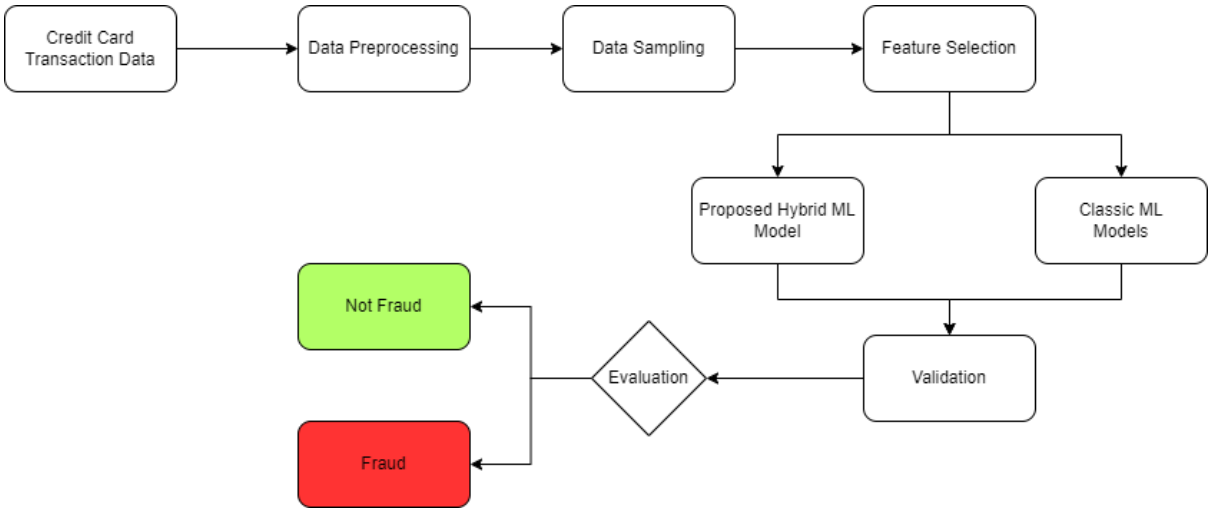


Figure 2

Figure 2 illustrates the data science methodology employed in this project. This methodology encompasses five principal stages, each contributing to the development and evaluation of the hybrid machine learning model. These stages are as follows: data collection, data preparation and exploration, model development, model testing, and visualization.

Within this structured framework, iterative processes play a pivotal role in enhancing the final machine learning model and its outcomes. These iterations encompass activities such as re-cleaning data for improved data analysis, fine-tuning data pre-processing techniques, and optimizing results by revisiting the model training process. This iterative approach ensures the continual refinement and enhancement of the hybrid machine learning model throughout the project lifecycle.

3.0.1 Data Collection

The dataset is collected from the Kaggle website, it is a data science platform for data scientist under Google. Based on the Kaggle dataset description, the dataset is collected from the collaboration of Worldline and the Machine Learning Group of ULB (University of Bruxelles) on data mining and fraud detection.

This dataset comprises transactions that occurred over a span of two days, where 492 of the 284,807 transactions are classified as frauds. The dataset is severely imbalanced, with fraudulent transactions accounting for just 0.172% of the total. It includes only numerical input variables, which result from a PCA transformation. Unfortunately, due to confidentiality constraints, we can't show the original column names or give more background information about the data. Features V1 to V28 represent the principal components obtained through PCA, while “Time” and “Amount” are the only features not subjected to PCA transformation. “Time” indicates the elapsed seconds between each transaction and the first one in the dataset, while “Amount” represents the transaction amount and can be used for instance-dependent cost-sensitive learning. The “Class” column will be used as the dependant variable, taking 1 as fraud and 0 as non-fraud.

3.0.2 Data Preparation

The data preparation phase is the most important in the success of any predictive analytics study, especially when dealing with real-world datasets, known for their complexity due to numerous outliers, missing values, imbalance, and other challenges. If not addressed appropriately, these issues can jeopardize the research. In this proposal, our data preparation involves addressing missing values, converting categorical features, feature scaling, feature selection, and resampling.

3.0.3 Data Cleaning

Data cleaning is the essential procedure of fixing or eliminating inaccurate, corrupted, improperly formatted, duplicate, or missing data within a dataset. When combining data from various sources, there is a heightened risk of data duplication or misidentification. Inaccurate data can lead to unreliable results and flawed algorithms, even if they appear valid. Although there's no universally applicable, one-size-fits-all method for specifying the precise steps in the data cleaning process, it's imperative to establish a consistent framework for data cleaning to ensure its executed correctly on every occasion. Based on this dataset description, there is no missing values in the data but there are highly imbalance data present in this dataset.

3.0.4 Categorical Encoding

The majority of machine learning algorithms require that input and output features are represented in a numerical format. This means that categorical data must be transformed into numerical values before creating a predictive model. Since the dataset we using have no categorical data so we do not need to perform categorical encoding.

3.0.5 Feature Scaling

Standardizing features involves transforming them to adhere to the characteristics of a standard normal distribution with a mean of 0 and a standard deviation of 1. This is a common prerequisite in many machine learning models where feature standardization is essential prior to employing machine learning methods. Failing to standardize the features can potentially impact the model's performance.

3.0.6 Data Resampling

The dataset discussed in this paper exhibits a significant imbalance, where one class (majority class) greatly outweighs the other class (minority class). This imbalance presents a challenge for many machine learning algorithms, as they often assume an equal distribution between classes. When dealing with imbalanced datasets, these algorithms can produce inaccurate and suboptimal predictive models.

Moreover, the issue of imbalance is further complicated by factors such as class overlapping, where instances from different classes share similar characteristics or features. This can lead to difficulties in distinguishing between the classes and result in reduced model performance.

3.0.7 Machine Leaning Models

Various machine learning classification techniques have been explored for the detection of fraudulent transactions, as discussed earlier. However, it's important to note that there is no universally optimal algorithm that can be applied to every specific problem. As a result, in this study, the researchers selected five different machine learning algorithms, encompassing both linear and nonlinear approaches. These algorithms were chosen based on their promising performance in the context of fraud detection. The selected algorithms include Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), XGBoost and Adaboost.

The development of hybrid models in this study was conducted in two distinct phases. In the initial phase, we built a single machine learning model. This model incorporated the five machine learning algorithms, namely LR, RF, SVM, XGBoost and Adaboost. The primary objective of this phase was to evaluate the performance of these algorithms individually and comprehensively.

The results obtained from the initial phase of model development offered valuable insights into the strengths and limitations of each algorithm, serving as a foundation for the creation of more advanced hybrid models in the second phase of the study. These hybrid models were designed to capitalize on the strengths of multiple algorithms, ultimately enhancing the accuracy and reliability of credit card fraud detection.

3.0.8 Metrics Evaluation

In machine learning, we follow a two-step process: first, we train a model using a training dataset, and then we assess the model's ability to generalize its predictions when tested on new, unseen data. To gauge how well the model performs, we rely on evaluation metrics, and the choice of these metrics depends on the nature of the problem at hand, whether it's a regression or classification task. In this section, we will focus exclusively on the evaluation metrics relevant to classification problems.

Classification problems involve assigning instances to specific categories or classes, such as classifying emails as spam or not spam. For such tasks, we use various evaluation metrics to assess the model's performance. These metrics offer information about various aspects of the model's classification capabilities.

- **Accuracy**: This metric computes the proportion of accurate predictions out of all predictions made, offering a comprehensive assessment of the model's classification performance.
- **Precision**: Precision calculates the ratio of accurate positive predictions relative to all the positive predictions. It evaluates the model's capacity to accurately recognize positive cases while minimizing false positives.
- **Recall**: Recall calculates the ratio of true positive predictions to the total actual positive instances. It assesses the model's capacity to correctly identify all positive instances while minimizing false negatives.
- **F1 Score**: The F1 score represents a compromise between precision and recall, calculated as their harmonic mean. It proves valuable, particularly in situations with imbalanced data.
- **Confusion Matrix**: The confusion matrix breaks down the model's predictions into categories such as true positives, true negatives, false positives, and false negatives, offering a more detailed view of its performance.

3.1 Tools and Technologies

This section will provide an overview of the technical tools and technologies that will be utilized for various tasks, including data cleaning, analysis, and machine learning algorithm development. These tools can be categorized into several areas, such as the development environment, programming languages, machine learning libraries and data visualization.

3.1.1 Development Environment

The development environment for this project is a crucial choice as it impacts the programming languages and tools available. We primarily consider the development environments which is Visual Studio Code. This environment provides the necessary tools for coding, data analysis, and machine learning.

3.1.2 Programming Languages

Python is a favoured choice for data science due to its versatility, robust community support, ease of learning, open-source nature, rich visualization libraries, seamless integration capabilities, scalability, and cross-platform compatibility. Its wide range of data science libraries and tools empowers data scientists to efficiently perform various tasks, making it the primary language for data science endeavours.

- Versatility: Python is an incredibly flexible programming language that provides an extensive set of libraries and frameworks tailored for data science, including but not limited to NumPy, pandas, scikit-learn, TensorFlow, and PyTorch. These tools cover everything from data manipulation and analysis to machine learning and deep learning.
- Ease of Learning and Readability: Python is known for its simplicity and readability. Its syntax is clear and easy to understand, making it an excellent choice for both beginners and experienced programmers. This readability also facilitates collaboration among team members.

- Open Source: Python and most of its data science libraries are open source. This means you can use them freely and contribute to the community. Open-source software tends to evolve quickly and is well-maintained.
- Cross-Platform Compatibility: Python is available on all major operating systems (Windows, macOS, Linux), makes it an excellent option for developing on multiple platforms.
-

3.1.3 Machine Learning Libraries

Below are the list of machine learning libraries and frameworks for Logistic Regression, AdaBoost, XGBoost, Random Forest, and Support Vector Machines (SVM) in Python:

Logistic Regression:

- Scikit-Learn (sklearn): Scikit-Learn is one of the most popular Python libraries for machine learning, including logistic regression. It offers a wide range of tools for classification and regression tasks.

AdaBoost:

- Scikit-Learn (sklearn): Scikit-Learn includes the AdaBoost algorithm as part of its ensemble module. It's easy to use and well-documented.

XGBoost (Extreme Gradient Boosting):

- XGBoost: XGBoost is a powerful and efficient gradient boosting library that excels in predictive accuracy. It is widely used in machine learning competitions and data science projects. You can install it using pip: pip install xgboost.

Random Forest:

- Scikit-Learn (sklearn): Scikit-Learn provides an excellent implementation of the random forest algorithm, making it easy to work with ensemble methods.

Support Vector Machines (SVM):

- Scikit-Learn (sklearn): Scikit-Learn offers a comprehensive set of tools for SVM, supporting both classification and regression tasks. It provides different kernel options for SVM.

3.2 Expected Results

it is anticipated that hybrid machine learning models will surpass the general effectiveness, of traditional machine learning models. These hybrid models are also expected to exhibit greater robustness and resilience when dealing with real-world data inconsistencies. If the outcome of the hybrid models is outperforming the traditional models, then we can use hybrid models to detect the credit card fraud.

3.3 Summary

Figure 3

Figure 3 illustrates the final choice of key tools and technologies for credit card fraud detection. Visual Studio Code and Python serve as the development environment and programming language, influencing the decision to leverage Python Libraries. While it confines us to Python, this language is well-suited for this project due to its readability and the availability of libraries that meet our requirements.

For data manipulation and analysis, we've opted for Pandas, which able to integrates with Matplotlib and Seaborn for data visualization. Additionally, we've incorporated machine learning libraries such as Scikit-Learn and XGBoost, providing the essential models and components necessary for creating hybrid and traditional machine learning models. Scikit-Learn will also play a role in assessing regression metrics.

Chapter 4: Results and Analysis

In this chapter, the methods outlined in Chapter 3 will be applied to preprocess data related to Internet user churn. The data will be modelled and the resulting model will be assessed.

4.0 Data Analysis and Preprocessing

The information gathered requires an analysis to identify its structure and undergo processing to determine data types and address missing data values. These aspects are thoroughly examined in this section.

4.0.1 Dataset Overview

Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0

Figure 4.0

Due to confidentiality, actual columns name was replaced by V1 to V28.

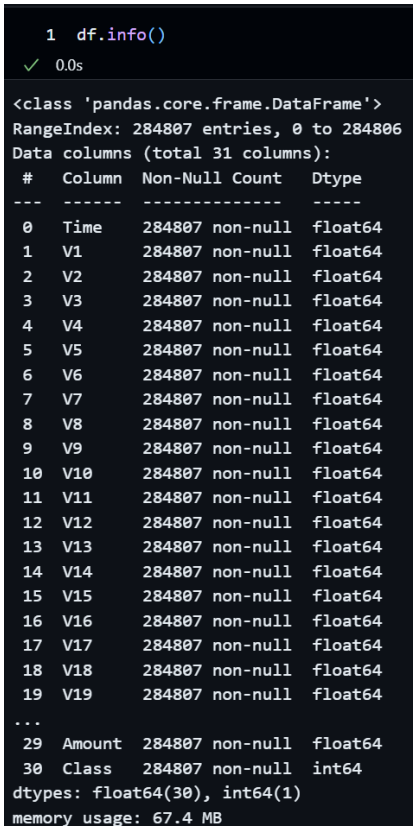


Figure 4.1

As you can see in Figure 4.1, we examine the data types of each column. It is showed that all data types are numeric, aligning with our criteria for building a machine learning model.

4.0.2 Missing Values

```
1  # Check Null Values
2  df.isna().sum()
3
4  # There is no null values in this dataset
```

```
Time      0
V1        0
V2        0
V3        0
V4        0
V5        0
V6        0
V7        0
V8        0
V9        0
V10       0
V11       0
V12       0
V13       0
V14       0
V15       0
V16       0
V17       0
V18       0
V19       0
V20       0
V21       0
V22       0
V23       0
V24       0
...
V27       0
V28       0
Amount    0
Class     0
```

Figure 4.2

In figure 4.2 showed that there are no missing values in our dataset.

4.0.3 Data Standardization

```
# Since most of our data has already been scaled we should scale the columns that are left to scale (Amount and Time)
from sklearn.preprocessing import StandardScaler, RobustScaler

# RobustScaler is less prone to outliers.

std_scaler = StandardScaler()
rob_scaler = RobustScaler()

df['scaled_amount'] = rob_scaler.fit_transform(df['Amount'].values.reshape(-1,1))
df['scaled_time'] = rob_scaler.fit_transform(df['Time'].values.reshape(-1,1))

df.drop(['Time','Amount'], axis=1, inplace=True)
```

Figure 4.3

I used Robust Scaler and Standard Scaler to scale the Time and Amount column. Robust Scaler is better than Standard Scaler when the features have outliers because it uses median and IQR instead of mean and STD which is more resistant to outliers.

4.0.4 Outliers

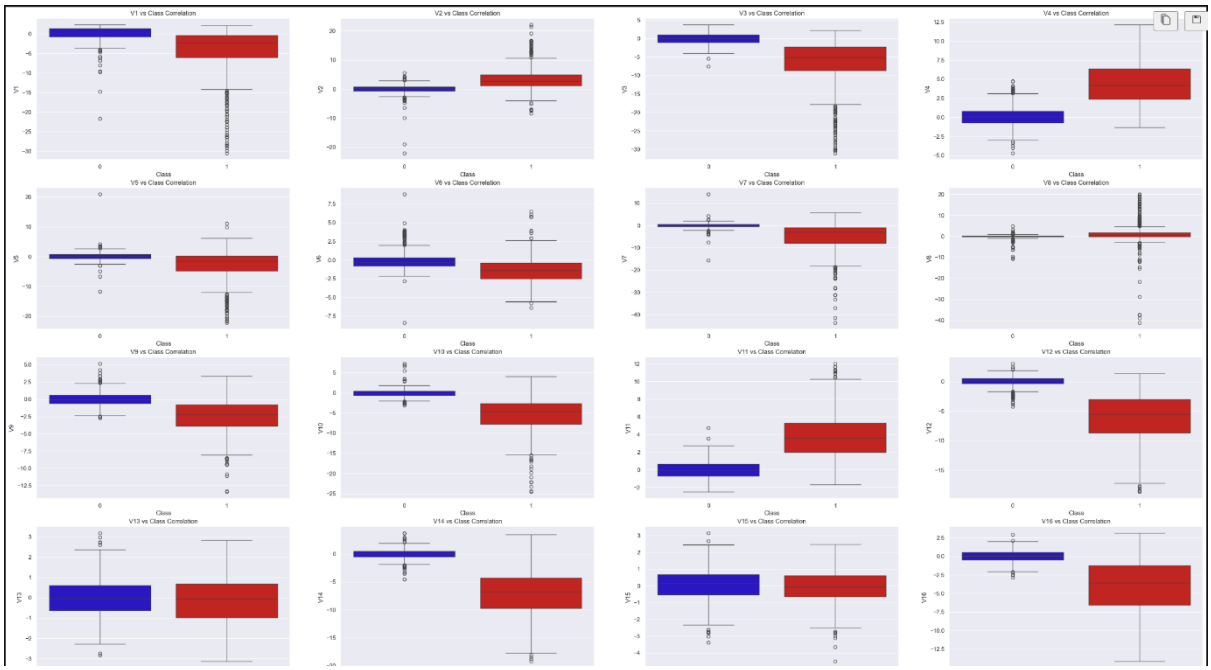


Figure 4.4

As you can see in Figure 4.4, there are quite a lot of outliers in this dataset. I used Interquartile Range method to remove the outliers

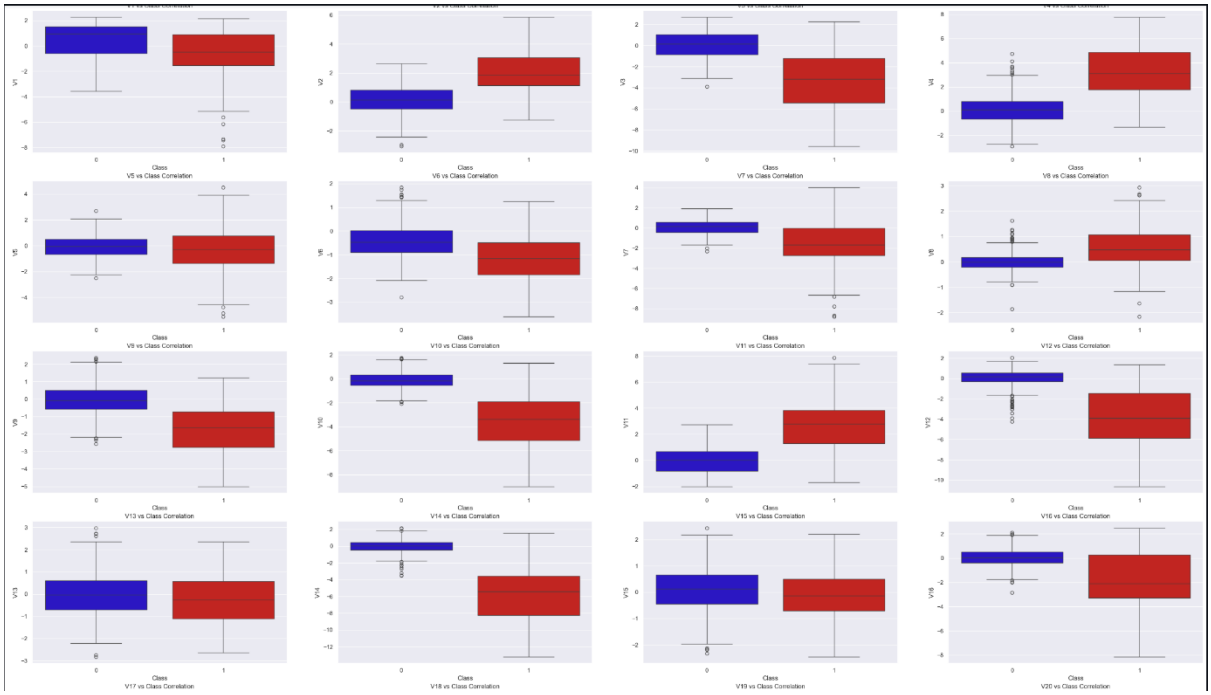


Figure 4.5

Image above are the results of after removing the outliers using IQR.

	Accuracy with no outliers
LogisiticRegression	92.31
Support Vector Classifier	93.60
RandomForestClassifier	93.59
XGBoost	92.31
AdaBoost	91.89

	Accuracy
LogisiticRegression	94.54
Support Vector Classifier	93.77
RandomForestClassifier	94.54
XGBoost	94.41
AdaBoost	93.40

We also tested the model performance without removing outliers. Based on the image above, dataset without outliers performed worse than dataset with outliers. So we decided to keep the outliers in the dataset since it's giving better results.

4.0.5 Exploratory Data Analysis

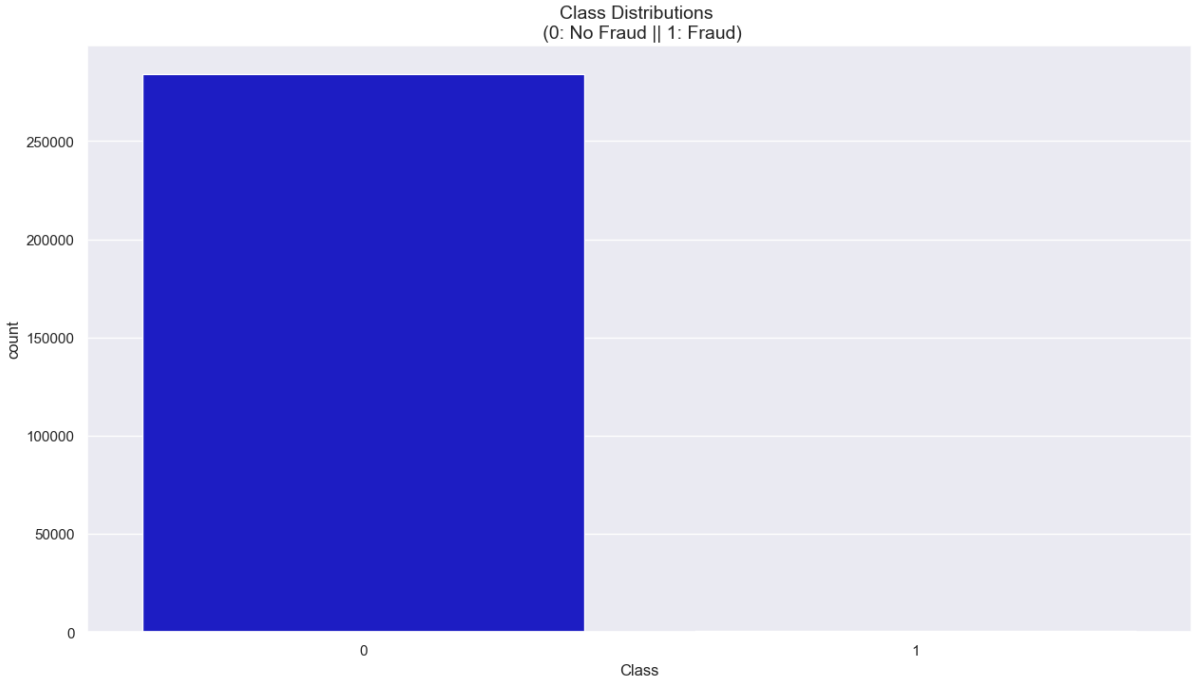


Figure 4.6

```
No Frauds 99.83 % of the dataset
Frauds 0.17 % of the dataset
```

Figure 4.7

As you can see in figure 4.6 and figure 4.7, the dataset has highly imbalanced class which only 0.17% of the dataset is fraud while 99.83% is no fraud.

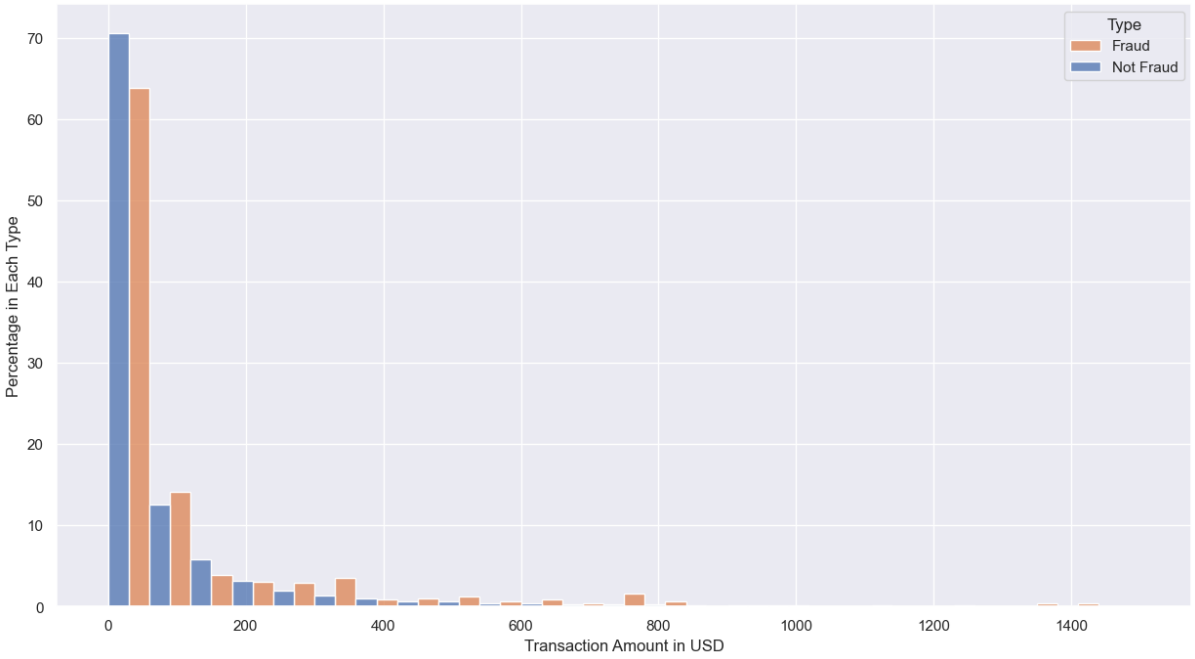


Figure 4.8

In the figure 4.8, the higher the transaction amount spent is, the higher the fraud compared to no fraud after 300 USD spent.

4.0.6 Data Sampling

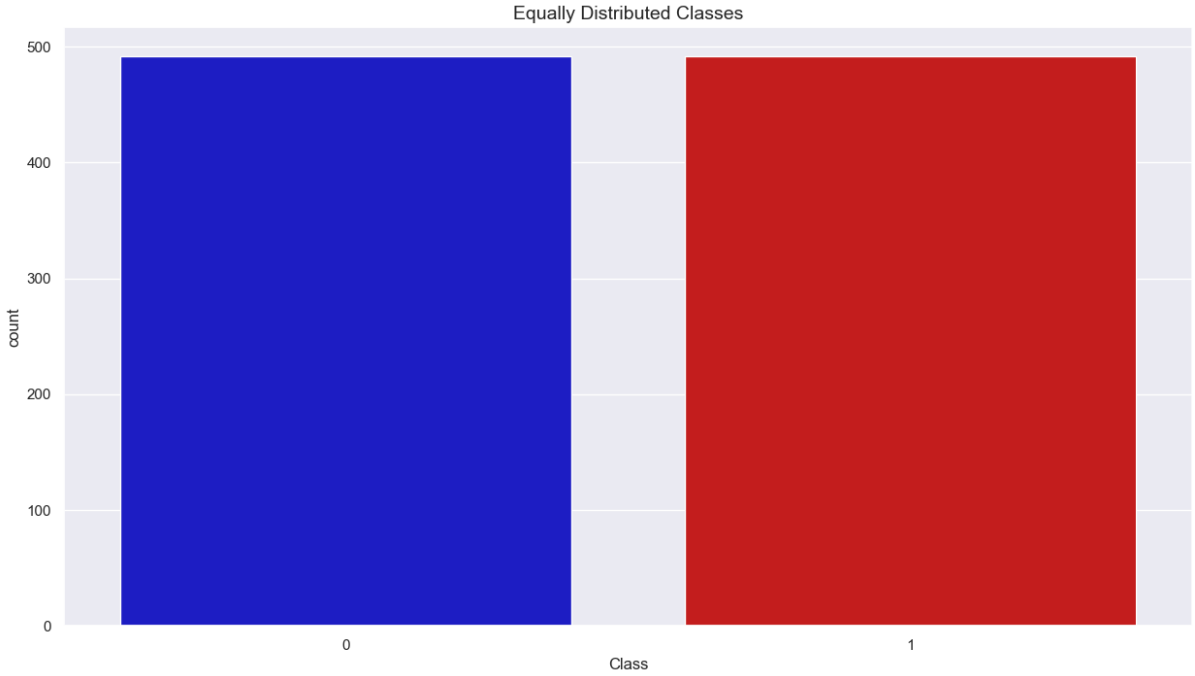


Figure 4.9

I used subsampling method to sample the data. Subsampling is a technique that reduces the size of data by choosing a subset from the original dataset. As you can see in figure 4.9, the class distribution is 50% each after using the subsample method which is much better than 99.83% to 0.17%.

4.0.7 Data Correlation

We can examine the correlation between the feature columns and the label columns to identify columns that show a stronger relation to the label outcomes. This analysis helps determine which columns positively influence the results and which ones have a negative impact. The correlation can be visualized through graphs.

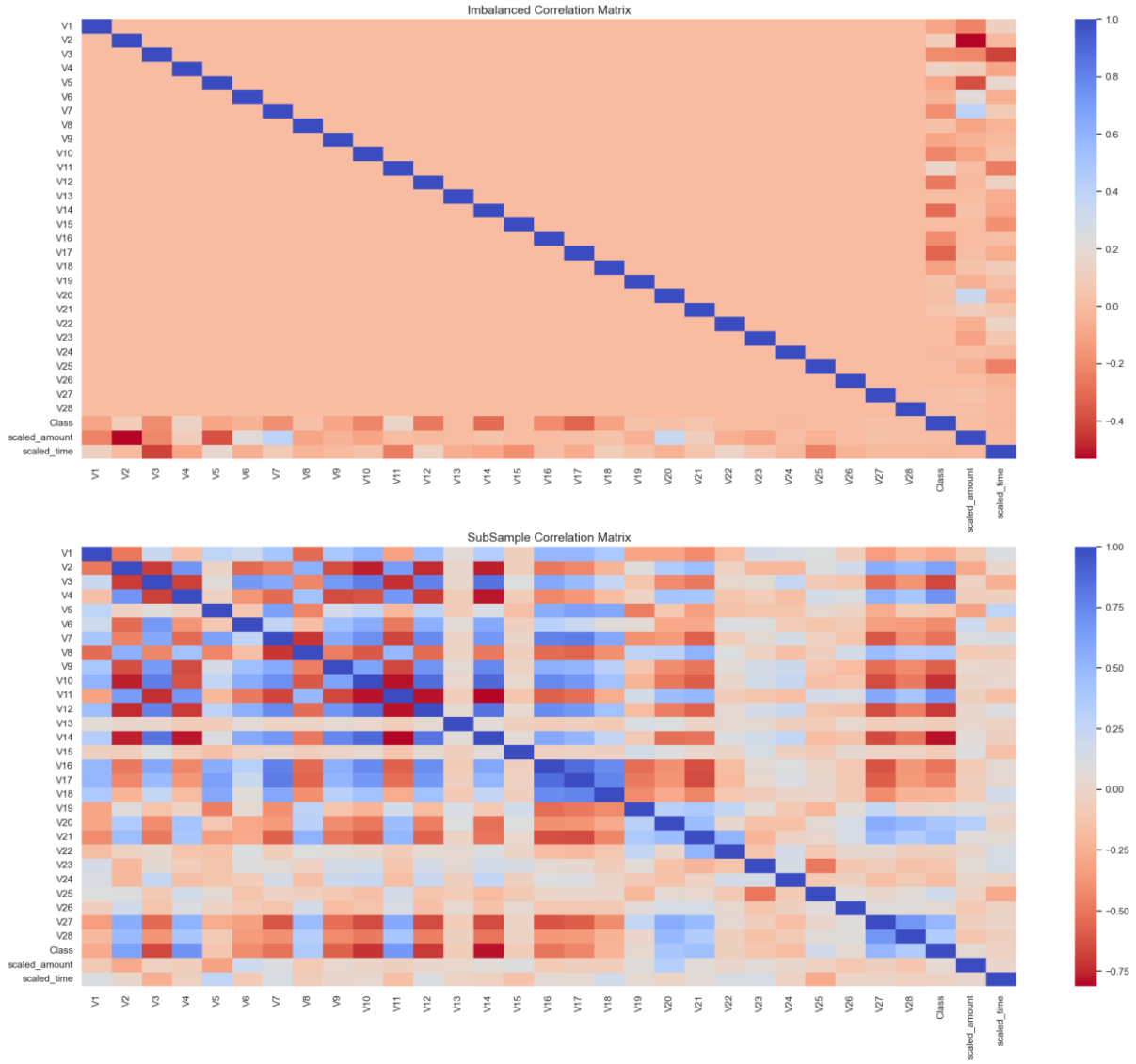


Figure 4.10

In figure 4.10, the subsample correlation matrix showed much better result than original dataset.

4.1 Model Results

In this research, five machine learning models, logistic regression, random forest, SVM, XGBoost and AdaBoost are employed to analyse the credit card fraud detection. The process involves constructing models, refining their performance by adjusting parameters, and assessing the outcomes using various evaluation metrics. The five machine learning models will compare to my hybrid machine learning models.

4.1.1 Model Building

We adhere to the Pareto Principle, splitting the dataset into an 80% training set and a 20% test set. Utilizing the train_test_split function, we create four subsets: Xtrain, Xtest, Ytrain, and Ytest. During the model training phase, the Xtrain and Xtest datasets are employed, while the Ytrain and Ytest datasets are used for model testing to evaluate its performance.

The implementation leverages the scikit-learn library (sklearn) for logistic regression and random forest, SVM and AdaBoost model creation. For XGBoost, its native dependency package is utilized, as it offers superior performance and speed compared to the interface provided in scikit-learn.

The fit() function is employed in logistic regression, random forest, SVM and AdaBoost to input the training dataset into the respective models. In the case of XGBoost, its specific dataset format is used, requiring conversion of the pandas DataFrame into Dmatrix format. The training is initiated using its own train() function.

To optimize model parameters, learning curves are employed to identify and address overfitting. This process aims to enhance the model's consistency across both training and test sets.

4.1.2 Model Evaluation

A various of model evaluation metrics is used such as accuracy score, AUC and ROC curve score as our assessment metrics.

	Accuracy
LogisiticRegression	94.54
Support Vector Classifier	93.77
RandomForestClassifier	94.54
XGBoost	94.41
AdaBoost	93.40
ADA + LR	94.41
ADA + SVC	94.79
ADA + RF	94.79
ADA + XGB	95.05

Figure 4.11

As you can see in figure 4.11, all the machine learning models perform quite good in this dataset. AdaBoost and XGBoost has the highest accuracy score being 95.12%.

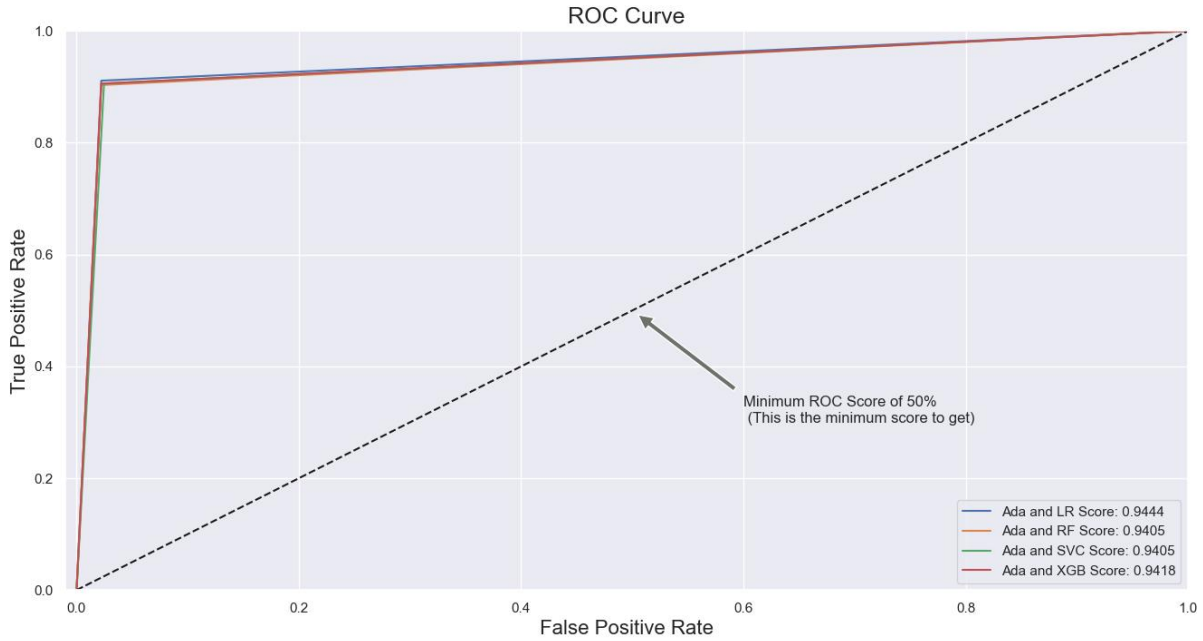


Figure 4.12

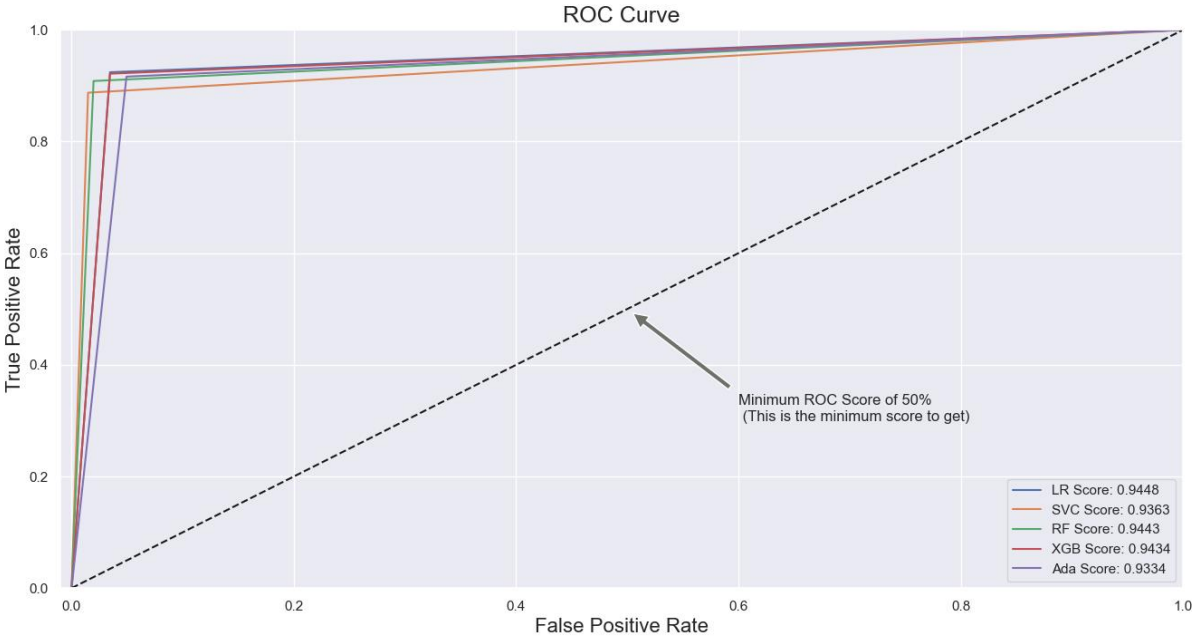


Figure 4.13

In the both figures above, hybrid machine learning models and single machine learning models performed around the same performances.

4.1.3 Removing Outliers Performance

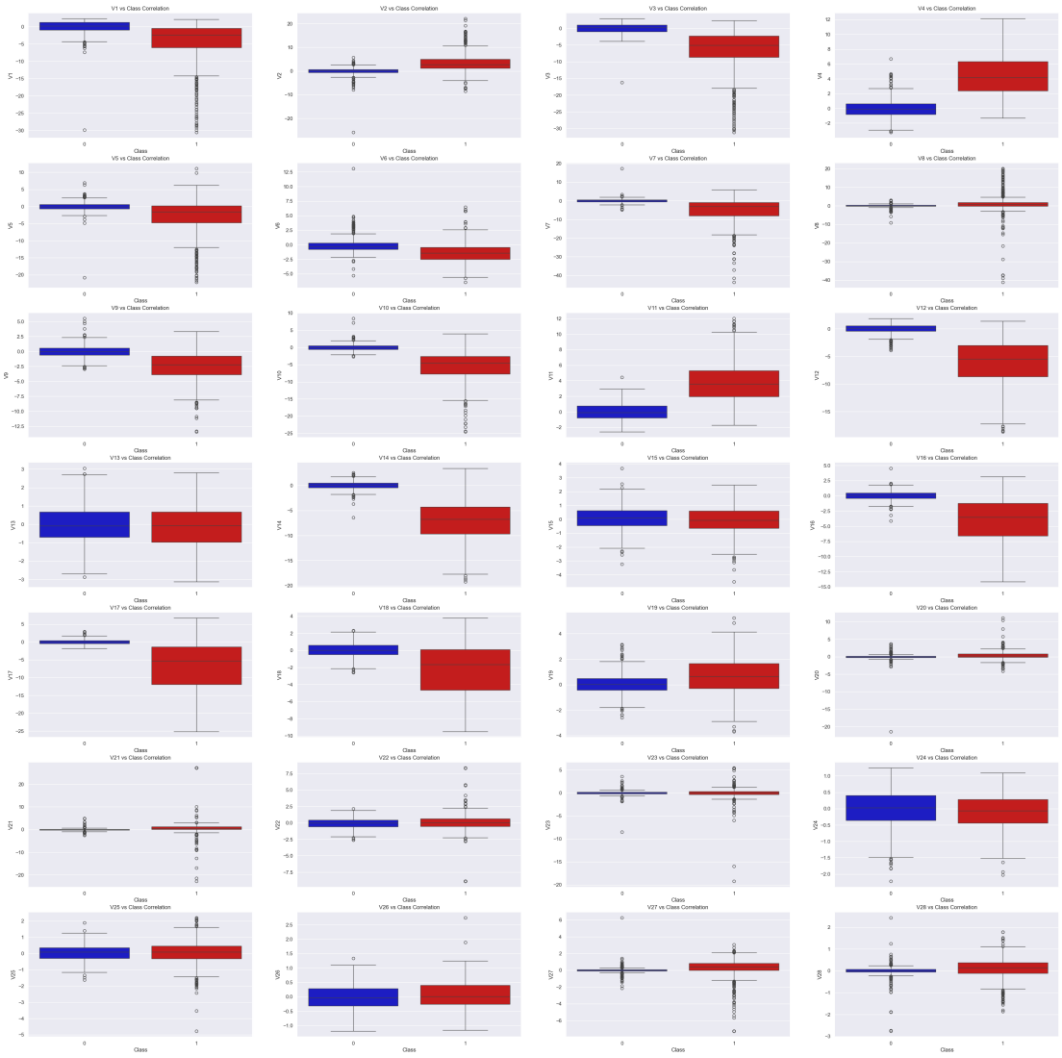


Figure 4.14

There were quite a lot of outliers showed in figure 4.14.

```python
# Removing outliers using IQR method
for i in range(1, 29):
    outlires_fraud = new_df['V' + str(i)].loc[new_df['Class'] == 1].values
    q25, q75 = np.percentile(outlires_fraud, 25), np.percentile(outlires_fraud, 75)
    print('Quartile 25: {} | Quartile 75: {}'.format(q25, q75))
    outliers_iqr = q75 - q25
    print('iqr: {}'.format(outliers_iqr))

    outliers_cut_off = outliers_iqr * 1.5
    outliers_lower, outliers_upper = q25 - outliers_cut_off, q75 + outliers_cut_off
    print('Cut Off: {}'.format(outliers_cut_off))
    print(f'V{str(i)} Lower: {outliers_lower}')
    print(f'V{str(i)} Upper: {outliers_upper}')

    outliers = [x for x in outlires_fraud if x < outliers_lower or x > outliers_upper]
    print('Feature V14 Outliers for Fraud Cases: {}'.format(len(outliers)))
    print(f'V{str(i)} outliers:{outliers}')

    new_df = new_df.drop(new_df[(new_df['V' + str(i)] > outliers_upper) | (new_df['V' + str(i)] < outliers_lower)].index)
    print('----' * 44)
```

Figure 4.15

We used IQR method to detect outliers and remove them.

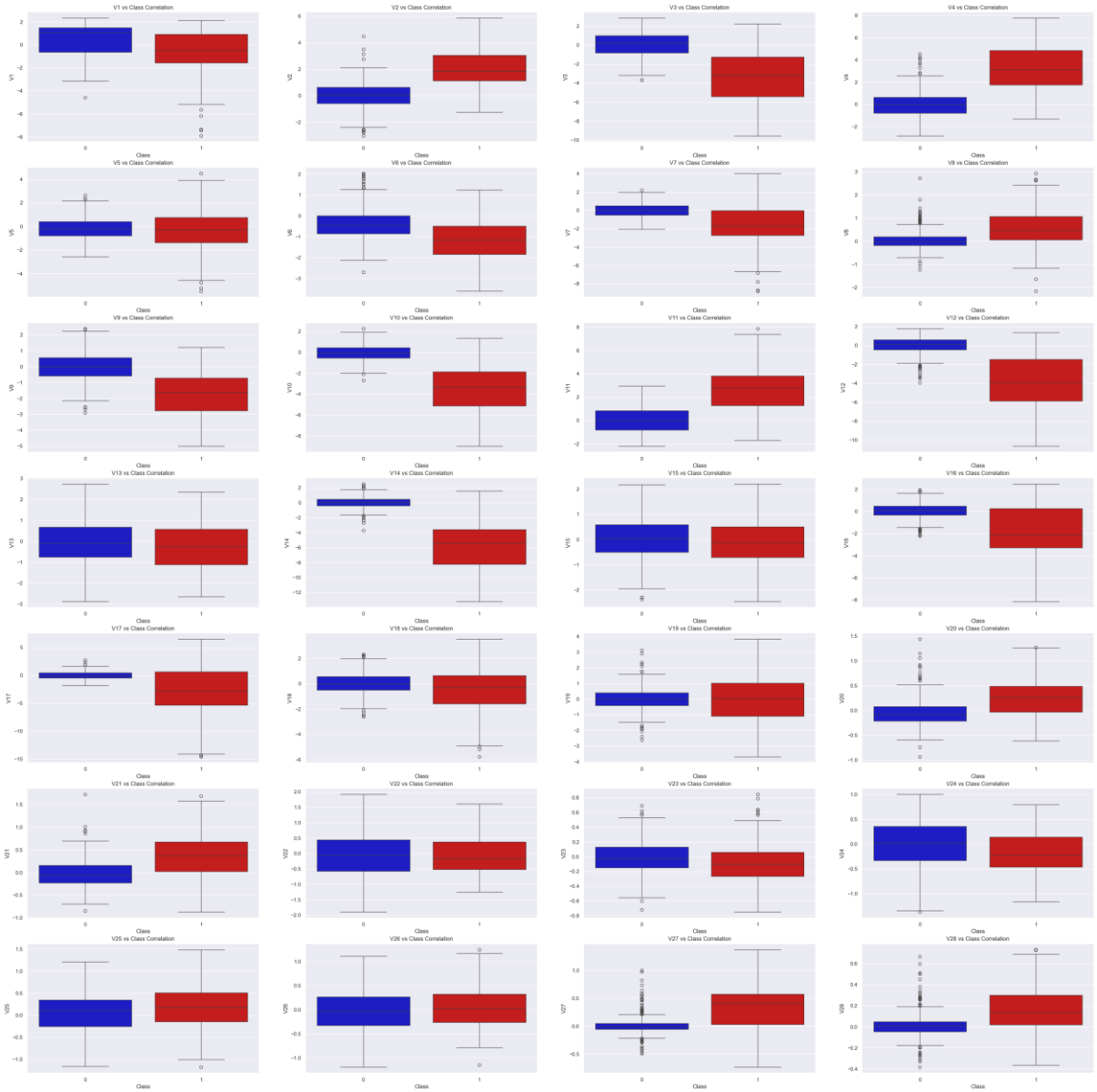


Figure 4.16

In figure 4.16 boxplot showed that majority of outliers had been removed.

	Accuracy with no outliers
LogisiticRegression	92.31
Support Vector Classifier	93.60
RandomForestClassifier	93.38
XGBoost	92.31
AdaBoost	91.89

Figure 4.17

	Accuracy
LogisiticRegression	94.54
Support Vector Classifier	93.77
RandomForestClassifier	94.41
XGBoost	94.41
AdaBoost	93.40

Figure 4.18

As you can see above figure 4.17 and 4.18, dataset with outliers performed better accuracy result compared to dataset without outliers. So, I had decided to keep the accuracy with outliers as my final performance metrics.

Chapter 5: Discussions and Conclusions

In the upcoming chapters, we will provide summaries and discussions regarding the study's contents. We will also share our perspectives, explain the implications and contributions of the research, and explore possibilities for future works.

5.0 Introduction

In this study, we conduct a comprehensive examination of credit card fraud detection. We outline our challenges and objectives, summarizing various factors impacting the predictive performance of machine learning models. To enhance current existing models, we employed using hybrid models to improve the performance and yield a slightly better result.

5.1 Conclusions

Credit card fraud has emerged as a significant global concern, particularly for financial institutions. While various approaches have been employed in the past to identify fraudulent activities, there is an ongoing need to explore reliable methods for detecting fraudulent credit card transactions.

This research focused on developing and investigating several hybrid machine learning models by combining supervised machine learning techniques for credit card fraud detection. The hybridization of different models was found to offer a substantial advantage over existing models. However, not all hybrid models demonstrated effective performance with the provided dataset, emphasizing the necessity for conducting numerous experiments to determine the most effective models.

Through a comparative analysis of the hybrid model against state-of-the-art models, it was determined that Adaboost + XGBoost emerged as the superior model for this dataset. The results also indicated a reduction in the error rate with the use of hybrid methods. For future works, the hybrid models employed in this study will be extended to other datasets within the credit card fraud detection domain.

5.2 Research Implications and Contributions

According to the findings of this research, banking industry have the opportunity to tailor their internal scenarios and assess various factors influencing prediction performance within the organization. This approach enables the mitigation of significant amounts of noisy data during the data collection phase, thereby enhancing the model's prediction performance at its source.

The insights gained from this study offer valuable guidance for researchers in related fields. They can leverage this study as a foundation for secondary research, making their own enhancements to achieve desired outcomes and contributing to the advancement of knowledge in their respective domains.

In this research, Multiple hybrid machine learning models were created and examined by combining supervised machine learning techniques in a credit card fraud detection study. The hybridization of various models was observed to provide advantage over existing state-of-the-art models.

5.3 Limitations

In this research, the inability to gather data directly from actual banking industry and identify factors influencing prediction performance posed a constraint. Consequently, there was a limitation in thoroughly processing the data from the initial collection stage, impacting the ultimate performance of the machine learning model to a certain extent.

1. The data distribution is skewed significantly due to the limited number of fraudulent transactions.
2. The data is constantly changing and evolving as time progresses.
3. A shortage of real-world datasets exists because of privacy-related issues.

5.4 Future Work

In the future work, there is potential to added more datasets, allowing for the flexibility to switch between different datasets while utilizing the same models for training and prediction. This approach enables the assessment of machine learning model performance across various datasets.

Additionally, future works could involve the incorporation of more models into the system. By training and optimizing these new hybrid models, subsequent comparisons can be made by evaluating their parameters.

References

Lawton, G., Burns, E., & Rosencrance, L. (2022, January 20). *What is logistic regression? - definition from Searchbusinessanalytics*. Business Analytics. https://www.techtarget.com/searchbusinessanalytics/definition/logistic-regression

Mohammed, N. H., & Maram, S. C. R. (2022, June 30). *Fraud detection of credit card using logistic regression*. SSRN. https://deliverypdf.ssrn.com/delivery.php?ID=616064002106077064127019117030124121037016025093044007014074023026003097074069112120028062030124045033010028075092087099120119046083078061083103068028072094122022094010018046068066069122097113064117101086127101064125117120120123099064080088104007125098&EXT=pdf&INDEX=TRUE

Chaturvedi, P., Agrawal, S., & Mishra, S. (2022, October 9). *IJRASET Journal for Research in Applied Science and Engineering Technology*. Credit Card Fraud Detection Using Hybrid Machine Learning Algorithms. https://www.ijraset.com/research-paper/credit-card-fraud-detection-using-hybrid-ml-algorithms#:~:text=More%20recent%20research%20was%20conducted,detection%20rates%20of%2099.99%20percent.

Shakya, R. (2018, December). *Application of machine learning techniques in credit ... - UNLV libraries*. digitalscholarship. https://digitalscholarship.unlv.edu/cgi/viewcontent.cgi?article=4457&context=thesesdissertations

Burke, J. (2023, September 25). *Why and how to use google colab: TechTarget*. Enterprise AI. https://www.techtarget.com/searchenterpriseai/tutorial/Why-and-how-to-use-Google-Colab

Emeritus. (2023, May 22). *Here's why use Python for data science - emeritus*. Here's Why Use Python for Data Science. https://emeritus.org/in/learn/heres-why-use-python-for-data-science/

Malik, E. F., Khaw, K. W., Belaton, B., Wong, W. P., & Chew, X. (2022, April 28). *Credit card fraud detection using a new hybrid machine learning architecture*. MDPI. https://www.mdpi.com/2227-7390/10/9/1480