**A close-up of a logo

Description automatically generated**

**Data Science Dissertation**

**Credit Card Fraud Detection Using Machine Learning Techniques IN FINANCIAL SECTOR**

**BY**

**OLUWASEYI ASHIPA PAUL**

**SUPERVISED BY - JOSEPH ANAN**

**AUGUST 2024**

# **CERTIFICATION**

This is to certify that this project was undertaken by ASHIPA, Oluwaseyi Paul, and presented to the School of Architecture Computing and Engineering, in partial fulfilment of the requirements for the award of the Master of Science Degree in Data Science, University of East London.

**………………………………………… …………………………**

**ASHIPA Oluwaseyi Paul (U2442187) Date**

(Researcher)

**……………………………… …………………………**

**Joseph Anan Date**

(Supervisor)

**………………………………. …………………………**

**Dr. Yang Li Date**

(Programme Leader)

# **DEDICATION**

This project is dedicated to Almighty GOD - the Most Gracious, the Ever Merciful. He who granted me good health and knowledge to complete the project work.

# **ACKNOWLEDGEMENT**

I would like to begin by expressing my deepest gratitude to Almighty God for His guidance, strength, protection, and blessings that have sustained me throughout the execution of this project.

I want to appreciate my parents for providing for me, supporting me, and giving me this opportunity to further my studies and have new experiences. May God continue to protect you and meet you at every point of your need.

I wish to express my profound gratitude to my supervisor, Mr. Joseph Anan, who, despite his busy schedule, took his time to show me and look through my work, guided me right, and dedicated his time to refining my work. His expertise and insights have been instrumental in shaping the quality and direction of this research. May God bless you and yours.

I would like to honour and appreciate Dr. Li Yang, the esteemed program leader, for all his teachings and guidance throughout the journey of my master’s degree program. His teachings have greatly contributed to my academic growth.

I am most grateful to my family members—my siblings, friends turned family—who supported and motivated me throughout the course of my master’s program. My unreserved appreciation goes to Eniola, Olayinka, Uncle Tola, Mom Omotola, and Uncle Tola Olebajo for their support, prayers, and encouragement during this period.

Finally, I would like to express my profound gratitude to all tutors of the department who taught and guided me throughout the course of my M.Sc. program.

# **ABSTRACT**

Credit card fraud is an ever-growing problem, with billions of dollars lost in the financial industry. This study employs machine learning techniques for detecting credit card fraud within the financial sector; it also aimed to identify the most effective machine learning model for fraud detection. Using a dataset containing one million transactions obtained from Kaggle, the dataset has an imbalance distribution, and the oversampling technique "SMOTE" was applied to balance the distribution. Several algorithms were implemented: random forest, decision tree, K nearest neighbour, logistic regression, and support vector machine. The random forest and decision tree model both demonstrated outstanding results, achieving the same accuracy of 100%, but random forest was shown to be the best by giving a lesser number of Type I errors of "1" and Type II errors of "2,"  making it the most reliable algorithm for this task. In contrast, with K nearest neighbour with an accuracy of 100%, logistic regression with an accuracy of 93%, and support vector machines with an accuracy of 95%, this model exhibits a higher misclassification error despite also showing high accuracy. The precision, recall, F1-score, and ROC-AUC were also used to evaluate all the models to ensure the models' effectiveness. This work also provides a comprehensive framework for detecting credit card fraud, highlighting the potential of machine learning to mitigate fraudulent activities in the financial sector. The project finishes by suggesting new research opportunities, such as investigating additional machine learning techniques and applying these models to diverse datasets to improve prediction accuracy.

Table of Contents

[CERTIFICATION 2](#_Toc175940214)

[DEDICATION 3](#_Toc175940215)

[ACKNOWLEDGEMENT 4](#_Toc175940216)

[ABSTRACT 5](#_Toc175940217)

[LIST OF FIGURES 7](#_Toc175940218)

[LIST OF TABLES 9](#_Toc175940219)

[1.1 PROJECT SUMMARY 10](#_Toc175940220)

[1.2 RESEARCH AREA (a brief literature review) 10](#_Toc175940221)

[1.3. AIM AND OBJECTIVE 12](#_Toc175940222)

[**1.3.1. Objectives** 12](#_Toc175940223)

[1.4. RESEARCH QUESTION 12](#_Toc175940224)

[1.5. EXPECTED PRACTICAL ELEMENT OUTPUT (FRAMEWORK) 12](#_Toc175940225)

[1.6. DATA SOURCE 13](#_Toc175940226)

[1.7. REQUIRED RESOURCES 13](#_Toc175940227)

[1.8. PREREQUISITE KNOWLEDGE/SKILLS REQUIREMENT 14](#_Toc175940228)

[Chapter 2: LITERATURE REVIEW 15](#_Toc175940229)

[2.1. Overview 15](#_Toc175940230)

[2.2. FRAUD 15](#_Toc175940231)

[2.3. CREDIT CARD 16](#_Toc175940232)

[2.4. CREDIT CARD FRAUD 18](#_Toc175940233)

[2.5. MACHINE LEARNING 19](#_Toc175940234)

[**2.5.1 Supervised Learning** 20](#_Toc175940235)

[**2.5.2 Unsupervised Learning** 21](#_Toc175940236)

[**2.5.3 Reinforcement Learning** 21](#_Toc175940237)

[2.6. Related Work Using Machine Learning Technique To Detect Credit Card Fraud 23](#_Toc175940238)

[**2.6.1. CONCLUSION** 26](#_Toc175940239)

[CHAPTER 3 – METHODOLOGY 27](#_Toc175940240)

[3.1. OVERVIEW 27](#_Toc175940241)

[3.2. DATASET 27](#_Toc175940242)

[3.3. DATA PRE-PROCESSING 28](#_Toc175940243)

[**3.3.1 Missing Data** 28](#_Toc175940244)

[**3.3.2 Correlation Testing** 28](#_Toc175940245)

[**3.3.3 Feature Scaling** 29](#_Toc175940246)

[**3.3.4 Handing Imbalanced Data** 30](#_Toc175940247)

[3.4. EXPLORATORY DATA ANALYSIS (EDA) 30](#_Toc175940248)

[3.5. IMPLEMENTATION OF MACHINE LEARNING MODEL 30](#_Toc175940249)

[**3.5.1 Model Selection** 31](#_Toc175940250)

[3.6. MODEL EVALUATION 32](#_Toc175940251)

[**3.6.1 Confusion Matrix** 33](#_Toc175940252)

[**3.6.2 Roc Curve and AUC** 34](#_Toc175940253)

[**Key Metrics** 35](#_Toc175940254)

[CHAPTER 4- IMPLEMENTATION 36](#_Toc175940255)

[4.1. OVERVIEW 36](#_Toc175940256)

[4.2. SETTING WORKING DIRECTORY 36](#_Toc175940257)

[4.3. IMPORT THE LIBRARY 36](#_Toc175940258)

[4.4. CHECKING THE STRUCTURE OF THE DATASET 37](#_Toc175940259)

[4.5. CHECKING FOR MISSING VALUES 37](#_Toc175940260)

[4.6. BOXPLOT OF ORIGINAL DATA 37](#_Toc175940261)

[4.7. CORRPLOT 38](#_Toc175940262)

[4.8. HISTOGRAM FOR DEPENDENT VARIABLE 39](#_Toc175940263)

[4.9. BALANCE THE DISTRIBUTION OF THE DEPENDENT VARIABLE 39](#_Toc175940264)

[4.10. NORMALIZATION OF DATASET 40](#_Toc175940265)

[4.11. SPLITTING THE DATA 40](#_Toc175940266)

[4.12. RANDOM FOREST - MODEL 1 40](#_Toc175940267)

[**4.12.1 Random Forest Model With Regularization Parameters** 41](#_Toc175940268)

[**4.12.2 Confusion Matrix** 41](#_Toc175940269)

[**4.12.3 AUC & ROC** 42](#_Toc175940270)

[**4.12.4 Feature Importance** 43](#_Toc175940271)

[4.13. DECISION TREE – MODEL 2 43](#_Toc175940272)

[**4.13.1 Pruning of Decision Tree Model** 43](#_Toc175940273)

[**4.13.2 Pruning of Decision Tree Model 2** 44](#_Toc175940274)

[**4.13.3 Confusion Matrix** 44](#_Toc175940275)

[**4.13.4 AUC & ROC** 45](#_Toc175940276)

[**4.13.5 Feature Importance** 45](#_Toc175940277)

[4.14. K NEAREST NEIGHBOUR – MODEL 3 45](#_Toc175940278)

[**4.14.1 Algorithm to Determine the Best Nurmber of K** 46](#_Toc175940279)

[**4.14.2 Confusion Matrix** 46](#_Toc175940280)

[**4.14.3 AUC-ROC** 47](#_Toc175940281)

[**4.14.4 Feature Importance** 47](#_Toc175940282)

[4.15. LOGISTIC REGRESSION MODEL 4 48](#_Toc175940283)

[**4.15.1 Confusion Matrix** 48](#_Toc175940284)

[**4.15.2 Cross Validation** 49](#_Toc175940285)

[**4.15.3 AUC-ROC** 49](#_Toc175940286)

[4.16 SUPPORT VECTOR MACHINE MODEL 5 50](#_Toc175940287)

[**4.16.1 SVM Model Changing The Kernel** 50](#_Toc175940288)

[**4.16.2 Confusion Matrix** 51](#_Toc175940289)

[**4.16.3 AUC-ROC** 52](#_Toc175940290)

[CHAPTER 5- RESULTS AND DISCUSSION 52](#_Toc175940291)

[5.1 OVERVIEW 52](#_Toc175940292)

[5.2. SETTING WORKING DIRECTORY 52](#_Toc175940293)

[5.3. IMPORT THE LIBRARY 53](#_Toc175940294)

[5.4. CHECKING THE STRUCTURE OF THE DATASET 53](#_Toc175940295)

[5.5. CHECKING FOR MISSING VALUES 54](#_Toc175940296)

[5.6. BOXPLOT OF ORIGINAL DATASET 55](#_Toc175940297)

[5.7. CORRPLOT 56](#_Toc175940298)

[5.8. DISTRIBUTION OF THE DEPENDENT VARIABLE 57](#_Toc175940299)

[5.9. DISTRIBUTION OF THE DEPENDENT VARIABLE AFTER OVERSAMPLIN 58](#_Toc175940300)

[5.10. Normalization 59](#_Toc175940301)

[5.11. Splitting Data training and testing 61](#_Toc175940302)

[5.12. MODEL 1 RANDOM FOREST 61](#_Toc175940303)

[**5.12.1 Evaluation of Model 1** 62](#_Toc175940304)

[5.13 MODEL 2 DECISION TREE 67](#_Toc175940305)

[**5.13.1 Evaluation of Model 2** 67](#_Toc175940306)

[5.14 MODEL 3 K NEAREST NEIGHBOUR 73](#_Toc175940307)

[**5.14.1 Evaluation of Model 3** 74](#_Toc175940308)

[5.15 MODEL 4 LOGISTIC REGRESSION 79](#_Toc175940309)

[**5.15.1 Evaluation of Model 4** 79](#_Toc175940310)

[5.16 MODEL 5 SUPPORT VECTOR MACHINE 82](#_Toc175940311)

[**5.16.1 Evaluation of Model 5** 82](#_Toc175940312)

[5.17 COMPARISONS OF ALL MODEL 85](#_Toc175940313)

[5.17 CRITICAL EVALUATION OF RESULTS 86](#_Toc175940314)

[CHAPTER 6- CRITICAL EVALUATION AND CONCLUSION 87](#_Toc175940315)

[6.1. OVERVIEW 87](#_Toc175940316)

[6.2. LIMITATIONS 87](#_Toc175940317)

[6.3. CONCLUSION 87](#_Toc175940318)

[6.4. RECOMMENDATION 88](#_Toc175940319)

[6.5. PERSONAL EVALUATION OF EXPEREIENCE 88](#_Toc175940320)

[REFERENCES 89](#_Toc175940321)

[APPENDIX 91](#_Toc175940322)

# **LIST OF FIGURES**

[Figure 2. 1 CreditCard Fraud (Fraud – Pixabay, 2020) 9](#_Toc175917555)

[Figure 2. 2 History of the first Credit Card (Jennifer Rosenberg, 202AD) 10](#_Toc175917556)

[Figure 2. 3 Modern Credit Card (Head For Points, 2017) 11](#_Toc175917557)

[Figure 2. 4 Machine Learning: Core Techniques (S Ishwarya, 2024) 12](#_Toc175917558)

[Figure 3. 1 Correlation…………………………………………………………………….....20](#_Toc175917746)

[Figure 4. 1 setting directory code snippet…………………………………………………. ..26](#_Toc175917884)

[Figure 4. 2Import library code snippet. 26](#_Toc175917885)

[Figure 4. 3 Data structure code snippet. 26](#_Toc175917886)

[Figure 4. 4 missing values code snippet. 27](#_Toc175917887)

[Figure 4. 5 Boxplot code snippet. 27](#_Toc175917888)

[Figure 4. 6 corrplot code snippet. 28](#_Toc175917889)

[Figure 4. 7 Histogram code snippet. 28](#_Toc175917890)

[Figure 4. 8 Balance data distribution code snippet. 29](#_Toc175917891)

[Figure 4. 9 Normalizing data code snippet. 29](#_Toc175917892)

[Figure 4. 10 splitting data code snippet. 29](#_Toc175917893)

[Figure 4. 11 Random forest modelcode snippet. 30](#_Toc175917894)

[Figure 4. 12 Random Forest Model 2 code snippet. 30](#_Toc175917895)

[Figure 4. 13 Confusion matrixcode snippet. 30](#_Toc175917896)

[Figure 4. 14 AUC & ROCcode snippet 31](#_Toc175917897)

[Figure 4. 15 Feature Importance code snippet 31](#_Toc175917898)

[Figure 4. 16 Decision Treecode snippet. 32](#_Toc175917899)

[Figure 4. 17 Pruning the Decision Tree 1 32](#_Toc175917900)

[Figure 4. 18 Pruning the Decision Tree 2 32](#_Toc175917901)

[Figure 4. 19 Confusion Matrixcode snippet. 33](#_Toc175917902)

[Figure 4. 20 AUC & ROCcode snippet. 33](#_Toc175917903)

[Figure 4. 21 Feature Importance code snippet. 33](#_Toc175917904)

[Figure 4. 22 K nearest neighbour code snippet 34](#_Toc175917905)

[Figure 4. 23 Determine the best K code snippet. 34](#_Toc175917906)

[Figure 4. 24 Confusion Matrix code snippet. 34](#_Toc175917907)

[Figure 4. 25 AUC & ROCcode snippet. 35](#_Toc175917908)

[Figure 4. 26 Feature importance 35](#_Toc175917909)

[Figure 4. 27 Logistic Regression code snippet 36](#_Toc175917910)

[Figure 4. 28 Confusion Matrix code snippet. 36](#_Toc175917911)

[Figure 4. 29 Cross Validation code snippet 37](#_Toc175917912)

[Figure 4. 30 AUC & ROCcode snippet. 37](#_Toc175917913)

[Figure 4. 31 Support Vector machine kernel – linear code Snippet 37](#_Toc175917914)

[Figure 4. 32 Support Vector Machine kernel – poly code snippet 38](#_Toc175917915)

[Figure 4. 33 Confusion Matrix code snippet. 38](#_Toc175917916)

[Figure 4. 34 AUC & ROCcode snippet. 38](#_Toc175917917)

[Figure 5. 1 Setting Work Directory…………………………………………………………..40](#_Toc175917918)

[Figure 5. 2 Import Library 40](#_Toc175917919)

[Figure 5. 3 Structure of Data 40](#_Toc175917920)

[Figure 5. 4 Missing Values 41](#_Toc175917921)

[Figure 5. 5 Boxplot of original dataset. 42](#_Toc175917922)

[Figure 5. 6 Correlation Plot 43](#_Toc175917923)

[Figure 5. 7 Distribution of dependent variable 44](#_Toc175917924)

[Figure 5. 8 Balance data distribution 45](#_Toc175917925)

[Figure 5. 9 Before Normalization 45](#_Toc175917926)

[Figure 5. 10 Split train and test 46](#_Toc175917927)

[Figure 5. 11 Random Forest Model 1 Regulating parameter Confusion Matrix 47](#_Toc175917928)

[Figure 5. 12 Random Forest Model 2 Confusion Matrix 48](#_Toc175917929)

[Figure 5. 13 Classification Report 49](#_Toc175917930)

[Figure 5. 14 . Random Forest AUC-ROC 49](#_Toc175917931)

[Figure 5. 15 Feature Importance 50](#_Toc175917932)

[Figure 5. 16 Classification Report 51](#_Toc175917933)

[*Figure 5. 17 Confusion Matrix Decision tree model 1* 52](#_Toc175917934)

[Figure 5. 18 Confusion Matrix Pruning Decision Tree model 2 53](#_Toc175917935)

[*Figure 5. 19* *Confusion Matrix Pruning Decision Tree Model 3* 54](#_Toc175917936)

[*Figure 5. 20* *Decision Tree AUC-ROC* 55](#_Toc175917937)

[Figure 5. 21 Feature Importance 56](#_Toc175917938)

[*Figure 5. 22 Determine best K* 57](#_Toc175917939)

[*Figure 5. 23 Classification Report* 57](#_Toc175917940)

[Figure 5. 24 . K Nearest Neighbour Confusion Matrix 58](#_Toc175917941)

[*Figure 5. 25 K Nearest Neighbour AUC-ROC* 59](#_Toc175917942)

[*Figure 5. 26 Feature Importance* 60](#_Toc175917943)

[Figure 5. 27 Logistic Regression Confusion Matrix 61](#_Toc175917944)

[*Figure 5. 28 Classification Report* 62](#_Toc175917945)

[*Figure 5. 29 Cross Validation* 62](#_Toc175917946)

[*Figure 5. 30 Logistic Regression AUC-ROC* 62](#_Toc175917947)

[Figure 5. 31 Svm confusion matrix kernel – linear 63](#_Toc175917948)

[*Figure 5. 32 Svm confusion Matrix kernel – poly* 64](#_Toc175917949)

[Figure 5. 33 Classification Report 65](#_Toc175917950)

[Figure 5. 34 SVM AUC-ROC curve 65](#_Toc175917951)

# **LIST OF TABLES**

[Table 3.1 Metadata. 23](#_Toc175922268)

[Table 5. 1 Comprising all Model…………………………………………………………… 70](#_Toc175922279)

[Table 5. 2 Comprising Error of All models 70](#_Toc175922280)

**CHAPTER 1- INTRODUCTION**

## **1.1 PROJECT SUMMARY**

The project is investigating ways and means by which credit card fraud can be mitigated in the financial sector with various machine learning techniques. Credit card fraud is a critical issue in the financial sector. E-commerce and many other forms of payment using credit cards increase the risk of fraud, which is characterized by unauthorized transactions that lead to significant financial losses for both consumers and financial institutions(Vaishnavi Nath Dornadula and Geetha, 2019).

Having experienced fraudulent transactions, but fortunately, the bank successfully identified and prevented the unauthorized transactions from being processed. I wonder what was the mechanism that was used to effectively detect that the transaction was fraudulent. Credit card fraud has been a going problem in the digital world today. Day by day, credit card fraud has been involving various types of tactics, which has been increasing and changing the dataset used to detect fraud. (Kubat Miroslav, 1996) Machine learning requires minimal programming to detect patterns, enabling real-time updates on various fraud types. This project breaks down how machine learning has been leveraged in the detection of fraudulent activities on a credit card.

Several steps were involved in the assembly of the project. Collecting the dataset from Kaggle and conducting research on previous related work gives a better understanding of the approach to be taken in building this model. Putting together a methodology for the development of the model, arriving at results, discussion, and conclusions .

## 

## **1.2 RESEARCH AREA (a brief literature review)**

A credit card is a small plastic card provided to individuals as a means of payment. It enables its cardholder to purchase goods and services based on the cardholder's promise to pay for them. The security of credit cards relies on both the physical protection of the plastic card and the confidentiality of the credit card number. The phenomenon of globalization and the growing prevalence of online purchasing have led to a substantial growth in credit card transactions worldwide. Consequently, the significant increase in credit card transactions has resulted in a considerable surge in fraudulent activities.(Benson Edwin Raj and Annie Portia, 2011) Credit card fraud refers to the act of stealing and deceiving by using a credit card as a fraudulent means of obtaining money during a transaction. Credit card fraudsters utilize a myriad of tactics to perpetrate fraudulent activities. In order to effectively prevent credit card fraud, it is crucial to have a comprehensive understanding of the methods used to detect fraudulent activities. In recent years, credit card fraud has significantly decreased as a result of the implementation of numerous credit card fraud detection and prevention techniques.

Credit card fraud occurs when an individual uses another person's credit card without the knowledge or consent of the card owner or issuer. Additionally, the person utilizing the card has no affiliation with the cardholder or issuer and has no intention of either contacting the card owner or making payment for the item (Bhatla, Prabhu and Dua, 2003).

Financial fraud has a huge impact on both the financial world and daily life. Fraud can hurt people's trust in a business, make savings less stable, and raise the cost of living. Different types of fraud security models are used by financial institutions to deal with this issue. But scammers are flexible, and over time, they come up with a number of ways to get around these defences. Financial crime keeps getting worse, even though banks, police, and the government are all working hard to stop it. The people who commit fraud today can be very creative, smart, and quick.(Sadgali, Sael and Benabbou, 2019)

There have been several methods used in detecting credit card fraud, but machine learning stood out as the best and most progressive way to detect credit card fraud. It follows the pattern between historical data on the fraudulent and non-fraudulent transactions.

The field of machine learning originated nearly four decades ago with the bold objective of developing computational methods that could effectively apply different types of learning, namely systems capable of extracting knowledge from examples or data(Kubat Miroslav, 1996). Machine learning according to (Michie, 1998) allow computers to learn from data without specific instruction. The algorithms uncover hidden patterns within the dataset, which allow them to make predictions on new, similar data. Machine learning is commonly understood to include automatic computing techniques based on logical or binary operations that learn a task from a set of instances.

Machine learning is a subfield of artificial intelligence that enables computers to learn from data and improve performance over time. There are three main approaches to machine learning: supervised, unsupervised, and reinforcement learning. Supervised learning involves learning from labelled data, which already has a target, and using that to predict its own target.  Unsupervised learning is the study of patterns and structures in data that do not have any labels, whereas reinforcement learning is concerned with learning the best possible behaviour by interacting with the environment and adjusting based on trial and error.

## **1.3. AIM AND OBJECTIVE**

This dissertation aims to use machine learning technique to detect credit card fraud in financial sector.

### **1.3.1. Objectives**

1. Literature review
2. Methodology
3. Implementation
4. Results, findings, interpretations, and discussions
5. Critical evaluation and conclusions

## **1.4. RESEARCH QUESTION**

* Which machine learning algorithms would be most effective in detecting credit card fraud ?
* What are the ethical implications of using machine learning for credit card fraud detection?
* What is the cost-benefit analysis of implementing machine learning-based fraud detection systems for financial institutions?

## **1.5. EXPECTED PRACTICAL ELEMENT OUTPUT (FRAMEWORK)**

The methodology employed for detecting credit card fraud using machine learning techniques involves several key steps. Firstly, the dataset will be sourced from Kaggle. Machine learning model would be generated, using python and all the models will be evaluated based on their accuracy, precision, recall, and F1-score. The confusion matrix will be evaluated to determine the number of Type I and Type II errors and plot the ROC-AUC to view the model's performance; all will be compared, and the algorithm that has the greatest accuracy, precision, recall, and F1-score is considered the best algorithm that is used to detect fraud.

## **1.6. DATA SOURCE**

The data is a structured numerical data measured from various appliances. However, the data was downloaded from <https://www.kaggle.com/datasets/dhanushnarayananr/credit-card-fraud/data>.The dataset has 1,000,000 instances and 8 attributes. The attribute of the dataset contains,distance\_from\_home,distance\_from\_last\_transaction,ratio\_to\_medain\_purchese\_price,repeat\_retailer,used\_chip, used\_pin\_number, online\_order, and the target variable Fraud.

## **1.7. REQUIRED RESOURCES**

**Academic journals and papers** that include topics such as credit card fraud, techniques used to detect it, and elements for successfully building the model. Journals such as History of Credit Cards, Meaning of Fraud, and Financial Fraud.

**Data sources**: dataset from a financial organization for every transaction made. but the dataset used in this project was obtained from Kaggle, containing 1 million transactions.

**Software and tools:** Python, a statistical software, used in conjunction with libraries such as Stats Models to construct prediction models. Data visualization tools, such as Matplotlib, are valuable for analysing and displaying data, while Pandas is beneficial for managing and manipulating data frames.

**Data Analysis Techniques:** Knowledge in diverse data analysis approaches, including machine learning algorithms such as decision trees and random forests.

**Writing guides and templates:** Academic writing advice and dissertation templates can enhance the organization and effectiveness of a dissertation, ensuring it adheres to academic standards..

## **1.8. PREREQUISITE KNOWLEDGE/SKILLS REQUIREMENT**

**Statistics and Data Analysis:** A researcher must possess a comprehensive knowledge of statistical procedures, encompassing classifier/regression analysis, correlation, handling imbalanced data, and maybe more sophisticated techniques such as machine learning algorithms.

**Data Science knowledge and Programming:** Proficiency in utilizing statistical software such as R, Python, or other pertinent tools for the purpose of data manipulation and analysis.

**Research Methods:** Understanding of methodologies such as experimental design, data collection techniques (such as surveys and interviews), and awareness of ethical considerations in doing research involving human subjects.

**Domain knowledge** in understanding of the involves the most in the detection of a fraudulent transaction, such as online\_order, ratio\_to\_median\_purchase\_price and used\_chip and so on.

**Predictive Modelling:** Acquiring proficiency in predictive modelling approaches, such as decision trees, random forests, K closest neighbour, and others, and learning how to effectively utilize them for predicting fraudulent transactions.

**Critical Thinking and Writing Skills**: The capacity to scrutinize existing literature, formulate research inquiries, analyse data, interpret outcomes, and proficiently communicate findings through scholarly writing.

**Ethical Considerations:** Gaining comprehension of the ethical implications linked to research involving human subjects, including obtaining informed consent, safeguarding privacy, and addressing potential biases in data collection and analysis.

# **Chapter 2: LITERATURE REVIEW**

## **2.1. Overview**

This chapter aimed to explain every single concept of the topic by breaking down what a credit card is, what fraud is, what causes it, the methods that can be used to prevent or detect fraud early, the machine learning method being one of the best methods for preventing fraud, the different types of technology that can be used, and also examining a literature study of earlier work in which the technique has been implemented and showing its performance

## **2.2. FRAUD**

"Fraud" refers to any activity that relies on deception to achieve a gain. Fraud is considered a crime when it involves the "knowing misrepresentation of the truth or concealment of a material fact to induce another to act to his or her detriment." In other words, fraud occurs when you lie in order to deprive someone or an organization of money or property (Akers and Gissel, 2024). Fraud has a significant impact on enterprise performance and accounting decision-making. As a result, fraud analysis is highly valued by executives in state-owned firms and budgetary agencies. The probability of catching a fraud is lower than that of finding errors since fraudulent operations are typically carried out utilizing intricate and carefully designed strategies (Giriūnas and Mackevičius, 2014). According to the findings of the empirical research, the conditions under which diverse individuals can commit fraud are of the utmost relevance in fostering fraud. A thorough examination of scientific literature, accounting, and audit requirements allowed the authors to prepare the classification of the principal conditions raising fraud risk.

A person in a black mask riding on a credit card

Description automatically generated

Figure 2. 1 CreditCard Fraud (Fraud – Pixabay, 2020)

Financial fraud has a huge impact on both the financial world and daily life. Fraud can hurt people's trust in a business, make savings less stable, and raise the cost of living. Different types of fraud security models are used by financial institutions to deal with this issue. But scammers are flexible, and over time, they come up with a number of ways to get around these defences. Financial crime keeps getting worse, even though banks, police, and the government are all working hard to stop it. The people who commit fraud today can be very creative, smart, and quick. (Sadgali, Sael, and Benabbou, 2019).

## **2.3. CREDIT CARD**

A “credit card” is a means of payment, most often issued by financial institutions, that enables users to make purchases or obtain cash advances on credit. Credit cards are the most widely used payment method across the world (Bhatla, Prabhu and Dua, 2003). (Edward Bellamy, 1887) utopian novel Looking Backward introduced the concept of using a card for transactions. Bellamy used the term credit card eleven times in this story, although it referred to a card for spending a fixed minimum income rather than borrowing, making it more like a debit card. In 1950, the Diners Club card achieved broad popularity as the first shop card, after the inspiration of its founder Frank McNamara, who forgot his wallet at home while dining. He and Ralph Schneider co-founded the inaugural Diners Club card, which is largely regarded as the inception of the contemporary charge card.(Lance Cothern CPA, 2019)

A yellow and black ticket

Description automatically generated

Figure 2. 2 History of the first Credit Card (Jennifer Rosenberg, 202AD)

A credit card is a small plastic card provided to individuals as a means of payment. It enables its cardholder to purchase goods and services based on the cardholder's promise to pay for them. The security of credit cards relies on both the physical protection of the plastic card and the confidentiality of the credit card number. The phenomenon of globalization and the growing prevalence of online purchasing have led to a substantial growth in credit card transactions worldwide. Consequently, the significant increase in credit card transactions has resulted in a considerable surge in fraudulent activities. (Benson, Edwin Raj, and Annie Portia, 2011) Credit card fraud refers to the act of stealing and deceiving by using a credit card as a fraudulent means of obtaining money during a transaction. Credit card fraudsters utilize a myriad of tactics to perpetrate fraudulent activities. In order to effectively prevent credit card fraud, it is crucial to have a comprehensive understanding of the methods used to detect fraudulent activities. In recent years, credit card fraud has significantly decreased as a result of the implementation of numerous credit card fraud detection and prevention techniques.

A close-up of credit cards

Description automatically generated

Figure 2. 3 Modern Credit Card (Head For Points, 2017)

## **2.4. CREDIT CARD FRAUD**

Credit card fraud occurs when an individual uses another person's credit card without the knowledge or consent of the card owner or issuer. Additionally, the person utilizing the card has no affiliation with the cardholder or issuer and has no intention of either contacting the card owner or making payment for the item (Bhatla, Prabhu, and Dua, 2003). Credit card frauds are easy targets. Without any risks, a substantial sum can be withdrawn without the owner's awareness. Fraudsters actively try to have each fraudulent transaction appear real, hence posing a formidable challenge for fraud detection.

 “Credit card fraud” is one of the biggest financial threats to a business and an individual. However, in order to effectively combat fraud, it is essential to first comprehend the mechanics involved in carrying out fraudulent activities. Credit card thieves utilize a wide range of methods to carry out fraudulent activities. Credit card fraud is the act of using someone else's credit card without the knowledge or consent of the card owner or issuer for personal purposes. Moreover, the person utilizing the card lacks any affiliation with the cardholder or issuer and has no intention of either reaching out to the card owner or making payments for the items made. (Bhatla, Prabhu, and Dua, 2024).

## **2.5. MACHINE LEARNING**

Machine learning is a form of artificial intelligence (AI) that enables computers to acquire knowledge and improve their performance without explicit programming (Abraham Iorkaa, Barma, & GAYA Muazu, 2021). Machine learning, computational statistics, and data science all have a common focus on prediction. Machine learning allows computers to acquire knowledge and improve their performance by analyzing and interpreting data, without the need for explicit programming or guidance. The algorithms reveal latent patterns within the sample, enabling them to make predictions on novel, analogous data. Machine learning refers to the use of logical or binary operations in autonomous computing systems to learn a task from a given set of cases (Michie, 1998). It entails the development of computer systems that possess the ability to learn and enhance their performance as time progresses. Traditional machine learning involves utilizing statistical techniques to analyze data and make predictions, resulting in practical and useful information. This technique is utilized in many fields such as image and audio recognition, natural language processing, recommendation systems, fraud detection, portfolio optimization, and task automation. The citation "Bi et al., 2019" refers to a publication by Bi and colleagues in the year 2019.

There are three different types of machine learning: supervised learning, unsupervised learning, and reinforcement. But supervised learning and its technique will be used for credit card fraud detection.

A diagram of machine learning

Description automatically generated

Figure 2. 4 Machine Learning: Core Techniques (S Ishwarya, 2024)

### **2.5.1 Supervised Learning**

Supervised learning is a type of machine learning where the model is trained using a dataset that has been annotated with labels. The purpose is to obtain a mapping from input data to appropriate output labels. The input comprises data that describes a collection of distinct objects, commonly referred to as instances or examples. The output is a culmination or consequence delivered by a supervisor (Nasteski, 2017).

Supervised learning can be categorized into two primary subgroups: regression algorithms, which generate continuous output, and classification algorithms, which generate discrete output. Regression algorithms aim to find the most optimal function that can accurately fit the data points in the training dataset. Over time, the learning algorithm enhances its ability to anticipate the output by improving its predictions, with the goal of minimizing the difference between its forecasts and the actual output.

The three primary categorizations of regression algorithms consist of linear regression, multiple linear regression, and polynomial regression. Classification algorithms may effectively determine the appropriate class for a given set of data by associating each input with its respective class. In this scenario, the prediction function yields a categorical outcome, which is assigned to one of the possible classes (Taye Mohammad Mustafa, 2023).

The algorithms commonly employed for supervised learning classification include Random Forest, Naïve Bayes, Support Vector Machine, Decision Tree, and K Nearest Neighbor. For the purposes of regression analysis, the methods employed include linear regression, logistic regression, and Bayesian regression.

### **2.5.2 Unsupervised Learning**

Unsupervised learning is a type of machine learning where the model is trained with data that does not include any labels. In essence, the algorithm endeavors to gain an understanding of the patterns and structure inherent in the input data, without any explicit guidance for the desired outputs. Unsupervised learning seeks to identify concealed patterns or intrinsic structures within the data (Tyagi et al., 2022). Unsupervised learning is an algorithmic approach that aims to establish a learning framework solely for the sake of learning. The rationale behind unsupervised learning is that while the data processed by unsupervised learning algorithms possesses a complex intrinsic structure, the true information and the measurement used for training are usually limited in quantity. Unsupervised learning is employed to derive characteristics from unlabelled data and assign categories or labels to them in cases where the input data lacks labels. The subject matter was classified into two distinct categories: clustering and dimension reduction. Clustering refers to the procedure of categorizing data items that exhibit similarities to one another. Dimension reduction is the act of decreasing the number of characteristics in a dataset while still preserving important information (Taye Mohammad Mustafa, 2023).   
Two commonly used techniques for clustering are k-means clustering and hierarchical clustering. Three techniques commonly used for dimension reduction include principal component analysis (PCA) and singular value decomposition (SVD). Unsupervised learning can be applied to several domains such as finance for client segmentation, retail for inventory management, manufacturing for quality control, and telecommunication for network optimization.

### **2.5.3 Reinforcement Learning**

Reinforcement learning is a method of learning in which an agent interacts with its environment by performing actions and learns from the resulting errors or rewards it receives. Reinforcement learning is characterized by the use of trial-and-error exploration and the consideration of delayed rewards as important factors. This approach allows machines and software agents to independently uncover the most effective behaviour for a certain situation in order to improve their performance. The agent's learning process is guided by a reinforcement signal, which provides basic reward feedback to choose the most advantageous actions. Out of the different branches of machine learning, describing reinforcement learning to someone who are not knowledgeable with the field is perhaps the easiest. It entails the acquisition of knowledge by repeated interaction and reaction, generally marked by learning through experimenting and adjusting to tasks through trial and error. In essence, it involves actively interacting with a surrounding, obtaining input, and adapting behaviors accordingly, typically in the pursuit of incentives. (Abraham Iorkaa and GAYA Muazu, 2021)

## **2.6. Related Work Using Machine Learning Technique To Detect Credit Card Fraud**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Author Name | Article Title | Data Source | Method Used | Result and Discussion | Significance and relevance |
| (Vaishnavi Nath Dornadula and Geetha, 2019) | Credit card fraud detection using machine learning algorithms | The dataset contains a total transaction of 284,807 in a duration of 2 day, with attribute of Transaction id, cardholder id, amount, time, and class. | Decision tree, Random forest, logistic regression, support vector machine, isolation forest and local outlier factor. | In the evaluation of the model, Random forest, decision tree, support vector and logistic regression had the highest accuracy of 99% all the same accuracy, while local outlier factor accuracy 89% and isolation 90%. | Random forest, decision tree, support vector machine and logistic regression all had the same accuracy of 99% for the dataset. |
| (Varmedja *et al.*, 2019) | Credit Card Fraud Detection – Machine Learning methods | The dataset was download from Kaggle containing transaction made in September 2013 by European cardholders, the data involves 31 numerical feature which was transformed using PCA into Time, Amount and class variable. | Random forest, Logistic regression, naïve Bayes, and Multilayer perceptron. | Result shows Random forest 99%, logistic regression 97% naïve bayes 99% and multilayer perception 99%. | All the models show high accuracy, but the random forest has very few misclassifications in fraudulent and non-fraudulent transactions, making it the best model. |
| (Khatri, Arora and Agrawal, 2020) | Supervised machine learning algorithms for credit card fraud detection : A comparison | The dataset consists of 284,807 transactions made over a span of two days, with 28 features and 492 were found out to be Fraud. | Decision tree, Random forest, KNN, Logistic regression, and Naïve bayes | The evaluation was based on the sensitivity and the time taken for the algorithms to run. The sensitivity of the KNN model is greater than the second great model decision tree, but as time taken by KNN for testing the data is very large, so decision tree was chosen | The K nearest neighbour and decision tree algorithms exhibited great performance, however the decision tree model had a shorter process time. |
| (Naik and Kanikar, 2019) | Credit card fraud detection based on machine learning algorithms | The dataset has 1000 records, and the attribute are. Credit card usage, purpose, Current balance in credit card, Average credit balance, Holder of a credit card, Holder status, CC age, Holder’s Property, Housing, Job, Employment, Location, Own telephone, Foreign worker etc. Credit Card and Holder’s Detail. | Naïve Bayes, Logistic regression , J48, AdaBoost | Result from the implemented algorithms shows Logistic regression and AdaBoost both have an accuracy of 100% | Both logistic regression and AdaBoost demonstrate exceptional performance, making them suitable for the dataset. |
| (Trivedi *et al.*, 2020) | An Efficient credit card fraud detection model based on machine learning methods | The dataset from ULB machine learning community and can be discovered on Kaggle. | Random forest, decision trees, support vector machine, logistic regression, KNN, gradient boosting classifier . | Evaluation results Random forest and KNN with an accuracy of 94% and 94% with GBM 94%. | All three models performed exceptionally well; however, both KNN and GBM exhibited a significant amount of misclassification. making random forests the best. |
| (Varma *et al.*, 2021) | Credit card fraud detection using random forest Algorithm | Dataset 284,315 with 492 fraud transaction from Kaggle | Random forest | Accuracy 95%, precision 96%, recall 96%, f1-score 96% and AROC score 95% | The random forest model shown and excellent performance with a accuracy of 95% and an AUC OF 95%. |
| (John O. Awoyemi, 2017) | credit card fraud detection using machine learning | Dataset from ULB Machine learning group containing 284,807 credit card transactions from the bank of European . | Logistic regression, K nearest neighbour, Naïve Bayes. | Result shows logistic regression 98%, k-nearest neighbour 96% and naïve bayes 97% . | Logistic regression is the highest performing model . |
| (Nadim *et al.*, 2019) | Analysis of machine learning technique for credit card fraud detection | The dataset consists of 284,807 transaction information with 0.172 positive fraud cases. | Logistic regression, Random forest, Decision tree and support vector machine. | Model evaluation Random forest the highest with an accuracy of 98% . | The random forest algorithm has strong performance, with an accuracy rate of 98%. |
| (Rathore *et al.*, 2021) | Credit card fraud detection using machine learning | The dataset obtained from Kaggle consists of 31 features and transformed using PCA. | Logistic regression, KNN, Decision tree and random forest | This model would be compare with their precision because their accuracy is almost the same 99%, random forest has the highest precision of 93%. | Random forest is the best perform model with a precision of 0.93%. |
| (Sumanth *et al.*, 2022) | Analysis of credit card fraud detection using machine learning techniques | Dataset from Kaggle attributes V1 to V27, showing 85,275 genuine transaction and 117 fraud transaction. | Support vector machine, Naïve Bayes, and Deep Neural Network | Deep neural network 99%, support vector 97% and Naïve Bayes 90%. | Deep neural network has the high accuracy of 99% . |

Table 2. 1 Related work Comparisons

### **2.6.1. CONCLUSION**

In this research of previous related work on using machine learning technique to detect credit card fraud, researchers had used different methods to make sure fraudulent transaction prediction is reliable and accurate. Based on the research algorithms like Random Forest with an accuracy of 99%, Decision Tree with an accuracy of 98%, K Nearest Neighbour with an accuracy of 96%, logistic regression with an accuracy of 97%, and support vector machine they are all shown to be the highest performing model for detecting credit cards, also producing a lesser miss classification error. This project would be using all the machine learning techniques mention: random forest, decision tree, K nearest neighbour, and logistic regression based on the above table.

# **CHAPTER 3 – METHODOLOGY**

## **3.1. OVERVIEW**

This chapter presents the methodology used to predict and detect credit card fraud using machine learning techniques. The methodology is organized to cover data collection, pre-processing, feature engineering, model selection, evaluation metrics, and experimental setup.

## **3.2. DATASET**

The dataset was retrieved from an open-source website, Kaggle.com, founded in 2010, which is a platform where people can find datasets, participate in competitions, and collaborate with others in the fields of machine learning and data science. The dataset consists of 8 attributes and 1,000,000 rows, which are all numeric. The attributes include "distance\_from\_home," which means the distance from home where the transaction happened, and "distance\_from\_last\_transaction," which also means the distance from where the last transaction happened. The last observation is the class "fraud," which contains binary variables where “1” is a case of fraudulent transaction and “0” is not a case of fraudulent transaction.

**A table of information with text

Description automatically generated with medium confidenceTable 3.1** Metadata.

## **3.3. DATA PRE-PROCESSING**

Data pre-processing is an essential step in this study, which involves handling missing data, features scaling, and dealing with imbalance data. These procedures are crucial for preparing the dataset for analysis, therefore guaranteeing accuracy and validity of detecting credit card fraud.

### **3.3.1 Missing Data**

Missing data refers to the absence of values in a dataset that should have been recorded. Missing data is a common issue in real-world data collection, and improper handling can significantly impact the performance of machine learning models. Missing data can occur due to various reasons, such as data collection errors, so handling missing data is necessary to ensure that the models built are robust, unbiased, and make accurate predictions. The method chosen to handle missing data should depend on the nature of the data and the context of the problem being solved. Properly addressing missing data can significantly improve the quality and reliability of the insights derived from the data.

This would be performance by using the (isnull) from pandas function in the python interface.

### **3.3.2 Correlation Testing**

Correlation analysis is a statistical technique employed to assess the degree of association between two quantitative variables. A high correlation signifies a substantial and robust connection between two or more variables, while a weak correlation implies merely a minimal association between the variables (Franzese and Iuliano, 2018). Put simply, it is the act of examining the intensity of the connection using existing statistical information. In other words, it is the process of studying the strength of that relationship with available statistical data. Because of the algorithm that is been developed, correlation testing would be used to understand the statistical relationship between the variables. Correlation testing would be done by importing spearman from SciPy. Stats into the python interface.

A diagram of a weak and negative

Description automatically generated

Figure 3. 1 Correlation

This would be done by using the (SciPy. Stats) function in the python interface.

### **3.3.3 Feature Scaling**

Feature scaling is the procedure of standardizing or aligning the characteristics of the dataset within a consistent range. In machine learning, the presence of different magnitudes, ranges, and units necessitates the application of a mode that can understand the phenomenon on a consistent scale. There are two methods of feature scaling: standardization and normalization. However, based on these datasets, normalization is the preferred approach.

Normalization is a technique employed to standardize the values of numerical columns in a dataset, ensuring they are on a consistent scale. This process preserves the relative differences between values without sacrificing any information.

MinMax Normalization

A math equation with black text

Description automatically generated with medium confidence

x is the normalized value, and it is in a range of 0 to 1.

This would be performance by using the (MinMaxScaler) function in the python interface.

### **3.3.4 Handing Imbalanced Data**

Unbalanced data refers to a dataset in which the target class has an uneven distribution of variance, i.e., they are more non-fraudulent transactions than fraudulent transactions within this dataset. To handle the imbalanced dataset, resampling technique has been used.

The Synthetic Minority Oversampling Technique (SMOTE) is an oversampling method that manages the imbalance data class by creating synthetic samples for the majority class. The algorithm also helps in preventing overfitting problems. It works by randomly picking a point from the minority class and computing the K-nearest neighbours for this point.

Handling imbalance data would be done by importing SMOTE function from imblearn.over\_sampling libraries.

## **3.4. EXPLORATORY DATA ANALYSIS (EDA)**

The exploratory data analysis (EDA) is the final test of the data quality. EDA provides insights that require a reassessment of domaining and framework design. Therefore, EDA, domaining, and wireframing are iterative procedures(Abzalov, 2016). This process entails examining the data, identifying patterns, and using visual and statistical techniques to understand its properties. This section analyses the data distribution, identifies outliers, and evaluates skewness or normality using a variety of graphical representations, including histograms, boxplots, pie charts, and bar charts. Also in the summary statistics are mean, median, mode, and range.

## **3.5. IMPLEMENTATION OF MACHINE LEARNING MODEL**

The Python programming language will be utilized to develop three distinct machine learning techniques for the identification of credit card fraud. The utilized supervised machine learning techniques included Random Forest, Decision Tree, and K-Nearest Neighbours (KNN).

In order to tackle the problem of class imbalance in the dataset, the oversampling approach is applied to the dependent variable "Fraud". This technique facilitated the equalization of the occurrence of fraud (1) and No-fraud (0) in the dataset, hence guaranteeing more precise forecasts.

We utilized a normal train-test split. The dataset was partitioned into a training set, which consisted of 70% of the data, and a test set, which included the remaining 30%. This partition provided a strong and thorough training of the model on a significant chunk of the data while allowing for an impartial evaluation of its ability to make predictions on new, unseen data.

### **3.5.1 Model Selection**

According to the systematic review carried out in chapter 2, it is mostly common to use the random forest, decision trees and K Nearest Neighbour models to predict similar phenomena.

* **Random Forest** is a potent and versatile machine learning technique that is extensively employed in many classification and regression applications (Bertsimas and Dunn, 2017).The system employs basic models that use binary splits on predictor variables to generate predictions for outcomes. Random forest can also be called the ensemble method, which is the combination of multiple machine learning models instead of one model. Random forest is a combination of a variety of decision trees. Each decision tree expert analyses a distinct subset of the material and provides their individual assessment. Subsequently, all the viewpoints are amalgamated to develop the ultimate forecast. This methodology facilitates the resolution of intricate issues and enhances the precision of forecasts. Instead of depending on a solitary authority, the Random Forest algorithm considers the viewpoints of numerous specialists. As the number of experts (trees) increases, the accuracy of the forecast also increases. This approach also mitigates the risk of errors caused by overfitting, a situation where the model becomes excessively tailored to the training data and fails to function adequately on new data.
* **Decision Tree** in machine learning is a graphical representation that facilitates decision-making or prediction by considering specific conditions. It starts with a question and splits the data into different paths based on the answer. These pathways can either lead to more questions or final conclusions. The goal is to create a tree-like structure that captures the patterns and relationships within the data. Every node represents a query, while the branches symbolize the potential answers or results. By following this hierarchical structure, we can provide forecasts regarding novel data by utilizing the patterns acquired from the data used for training(Bertsimas and Dunn, 2017). Decision trees are often preferred due to their inherent simplicity and clarity in comprehension and interpretation. his study implements five machine learning methods, namely random forest, logistic regression, support vector machine, K nearest neighbour, and decision trees, to forecast client turnover in the banking sector. We conducted a thorough evaluation of their performance. The study found crucial elements that have a major impact on customer attrition.
* **KNN (K-Nearest Neighbours)** , is a machine learning technique that clusters similar data points together in order to create predictions. The algorithm operates by collecting a collection of training data and then comparing new data points to identify the closest neighbours(Guo *et al.*, 2003). The algorithm as a result categorizes the new data by considering the majority of the closest neighbours. KNN is a straightforward and efficient approach employed for both classification and regression tasks. Nevertheless, the model's performance can be affected by the selection of the number of neighbours (K) and the metric used to determine similarity between data points. Additionally, the computation of findings for huge datasets may require more time.
* **Logistic regression** is a supervisory machine learning algorithm based on statistical methods used for binary logistic models, where the goal is to predict the probability that an instance belongs to a given class or not. The main objective of logistic regression is to predict the probability of an instance belonging to a particular class or not. Logistic regression is a statistical procedure used to examine the association between two data variables. This approach diverges from linear regression in that it forecasts the probability of a binary outcome rather than continuous values. Logistic regression is a statistical model that investigates the relationship between a set of input features and a binary dependent variable. It precisely forecasts the likelihood of an observation being classified into one of two categories. Logistic regression is a useful technique in spam detection that assesses the likelihood of an email being spam based on factors like word frequencies and sender information. It is widely employed in various domains, such as predicting customer churn in businesses or diagnosing diseases in healthcare. This model provides a probabilistic framework for making decisions between two options, making it a crucial tool in the set of machine learning algorithms used for handling classification tasks(Nick and Campbell, 2007).
* **Support Vector Machines** (SVM) is a powerful method that uses kernels to efficiently tackle regression and classification problems, with a strong computational capability. Support vector machines (SVM) provide superior generalization capabilities compared to other machine learning techniques. These algorithms are widely acknowledged to produce more precise outcomes than other algorithms in numerous applications, and they are supported by a strong theoretical foundation (Ukil Abhisek, 2007).

## **3.6. MODEL EVALUATION**

Model evaluation is a vital and critical step in machine learning that assesses the performance of a model on a specific dataset. An assessment of the model's predictions is conducted using various methods to evaluate its performance. The measures encompassed classification accuracy, precision, recall, specificity, and F1-Score. They offered valuable perspectives on various elements of the model's performance, including its general validity, the accuracy of its positive predictions, its capacity to identify positive occurrences, its ability to identify negative instances, and a comprehensive assessment of precision and recall. Confusion measures were employed to evaluate the occurrence of type 1 error, type 2 error, and ROC-AUC.

### **3.6.1 Confusion Matrix**

The confusion matrix is utilized to assess the efficacy of a classification model. The purpose of this matrix is to assess the performance of a model, detect incorrect classifications, and enhance the accuracy of predictions. It is presented in a tabular format and offers a concise overview of how well an algorithm performs, especially in tasks involving supervised learning. This work specifically addresses a binary classification problem, in which the predicted variable is divided into two distinct classes. Every column in the confusion matrix represents an occurrence of the real class, whereas each row represents an occurrence of the projected class. The matrix facilitates the identification of four distinct prediction categories: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

True Positive (TP): this indicates where the actual value was positive, and the model predicted a positive value.

True Negative (TN): this indicates where the actual value was negative, and the model predicted negative.

False Positive (FP): Type l Error: This indicates that the actual value was negative, but the model predicted a positive value.

False Negative (FN): Type ll Error: This indicates that the actual value was positive, but the model predicted negative values.

* **Accuracy -** The percentage of values that were correctly classified is determined using accuracy.

Accuracy =

* **Precision -** Precision tells us how many of the correctly predicted cases actually turned out to be positive.

Precision =

* **Specificity –** measures the proportion of actual negatives that are correctly identified

Specificity =

* **Recall -** Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

Recall =

* **F1-Score -** It is used when you need to take both precision and recall into account.

F1 score =

### **3.6.2 Roc Curve and AUC**

The Receiver Operating Characteristic (ROC) curve is a visual tool used to assess the effectiveness of a binary classification model in order to determine an appropriate decision threshold or operating point. The graph displays the TPR and FPR at different threshold values. The AUC, or area under the ROC curve, quantifies the model's ability to differentiate between the two groups.

**Key Metrics**

* **True Positive Rate (TPR) / Sensitivity / Recall**: The proportion of actual positives that are correctly identified by the model.

TPR =

* **False Positive Rate (FPR)**: The proportion of actual negatives that are incorrectly identified as positives by the model.

FPR =

# **CHAPTER 4- IMPLEMENTATION**

## **4.1. OVERVIEW**

This chapter examines the application of machine learning techniques to forecast whether a transaction is fraudulent or non-fraudulent. The study will offer valuable insights into the process of acquiring pertinent data, choosing the most efficient machine learning methods, training the models on crucial predictive features, and evaluating their effectiveness. By the end of this chapter, readers will acquire comprehension of how these models might aid in forecasting fraudulent transactions.

## **4.2. SETTING WORKING DIRECTORY**

A computer code with text

Description automatically generated with medium confidence

Figure 4. 1 setting directory code snippet.

This process involves importing the datasets (card\_transdate.csv) into the workspace and establishing a functional workspace. The imported dataset will be used for analysis.

## **4.3. IMPORT THE LIBRARY**

**A computer screen with text

Description automatically generated**

Figure 4. 2Import library code snippet.

This is the importing of the python library which contains functions, classes and variables used to analysis and build the model.

## **4.4. CHECKING THE STRUCTURE OF THE DATASET**

A black background with blue and yellow text

Description automatically generated

Figure 4. 3 Data structure code snippet.

This is observed using the functions info () and head (), which help examine the structure of the data and determine whether it consists of characters, factors, or numeric values. It allows for a deeper understanding, preparation, and utilization of the data for various purposes.

## 

## **4.5. CHECKING FOR MISSING VALUES**

A computer screen with colorful text

Description automatically generated

Figure 4. 4 missing values code snippet.

This process involves utilizing the (isnull) function to identify any missing values and generating a missingness map using the seaborn package. This is crucial in data pre-processing to ensure accurate analysis and reliable model building.

## 

## **4.6. BOXPLOT OF ORIGINAL DATA**

A computer code with colorful text

Description automatically generated

Figure 4. 5 Boxplot code snippet.

This is done using the boxplot() function. Box plots are important because they offer a clear and concise visual representation of data distribution. They provide information on central tendency, variability, and potential outliers, making it easier to spot patterns, trends, and anomalies in the data with efficiency.

## 

## **4.7. CORRPLOT**

A computer screen with text on it

Description automatically generated

Figure 4. 6 corrplot code snippet.

This analysis is performed following the installation of the required libraries. Instead of conducting separate correlational analyses, it presents them collectively. Data visualizations show the connections between variables in a given dataset. They aid in the identification of patterns, relationships, and potential multicollinearity issues, which are essential for activities such as feature selection, model construction, and result interpretation.

## **4.8. HISTOGRAM FOR DEPENDENT VARIABLE**

A computer screen with text and images

Description automatically generated

Figure 4. 7 Histogram code snippet.

This is to illustrate the dispersion of the data. Histograms are crucial since they offer a rapid visual overview of data distribution, enabling us to comprehend its shape, central tendency, variability, and identify any possible outliers or trends.

## 

## **4.9. BALANCE THE DISTRIBUTION OF THE DEPENDENT VARIABLE**

A screenshot of a computer code

Description automatically generated

Figure 4. 8 Balance data distribution code snippet.

This shows how the dependent variable was uniformed and put on the same proposition by using the function SMOTE to avoid overfitting in the training set.

## 

## **4.10. NORMALIZATION OF DATASET**

A computer screen with colorful text

Description automatically generated

Figure 4. 9 Normalizing data code snippet.

Normalization was performed on the dataset to ensure that the scales and observations were consistent. Using this technique allows the model to learn from features without being affected by variations in scale, resulting in more accurate and dependable predictions.

## 

## **4.11. SPLITTING THE DATA**

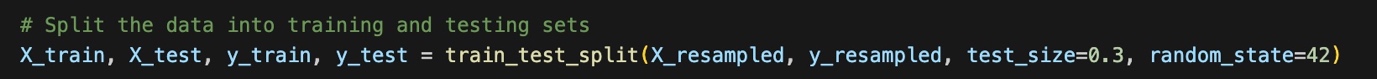


Figure 4. 10 splitting data code snippet.

This process consists of dividing the data into training and testing. The training data is used to build the algorithm, and the testing is used to validate the prediction of the model

## **4.12. RANDOM FOREST - MODEL 1**

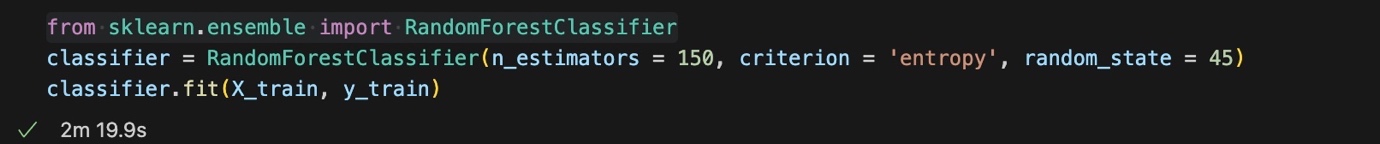
****

Figure 4. 11 Random forest modelcode snippet.

RandomForestClassifier imported from sklearn.ensemble use to train the model with an estimated tree of 150.

### **4.12.1 Random Forest Model With Regularization Parameters**

**A screenshot of a computer program

Description automatically generated**

Figure 4. 12 Random Forest Model 2 code snippet.

RandomForestClassifier built by setting regularization parameter

### 

### **4.12.2 Confusion Matrix**

**A computer screen with colorful text

Description automatically generated**

Figure 4. 13 Confusion matrixcode snippet.

This shows the number of accurate and inaccurate predictions performed by the model in a box form, as well as the accuracy, precision, and recall f1-score. The confusionMatrixDisplay module is imported from the sklearn library, while the visualization is done using the lib module from matplotlib.

### **4.12.3 AUC & ROC**

**A computer screen with colorful text

Description automatically generated**

Figure 4. 14 AUC & ROCcode snippet

Area under the curve is imported as roc\_curve and roc\_auc\_score from slearn libraries the code will display it in a graph form.

### **4.12.4 Feature Importance**

**A computer screen with colorful text

Description automatically generated**

Figure 4. 15 Feature Importance code snippet

Scikit\_learn provide “Feature importance” which is used to calculate the feature importance in a model.

## **4.13. DECISION TREE – MODEL 2**

**A screen shot of a computer code

Description automatically generated**

Figure 4. 16 Decision Treecode snippet.

The figure above illustrates the import of the decision tree classifier algorithm from sklearn libraire, a Python library used for machine learning algorithms, with the random state set to 50.

### **4.13.1 Pruning of Decision Tree Model**

**A black screen with green text

Description automatically generated**

Figure 4. 17 Pruning the Decision Tree 1

This figure above shows the importing of the Decision tree model from sklearn libraire and also pruning the tree.

### **4.13.2 Pruning of Decision Tree Model 2**

**A computer screen with text

Description automatically generated**

Figure 4. 18 Pruning the Decision Tree 2

This figure above shows the importing of the Decision tree model from sklearn libraire and also pruning the tree.

### **4.13.3 Confusion Matrix**

A computer screen with colorful text

Description automatically generated

Figure 4. 19 Confusion Matrixcode snippet.

This displays the number of accurate and inaccurate predictions performed by the model in a box form. confusionMatrixDisplay import from sklearn and lib imported from matplotlib use for visualization.

### **4.13.4 AUC & ROC**

A computer screen with text and images

Description automatically generated

Figure 4. 20 AUC & ROCcode snippet.

Area under the curve is imported as roc\_curve and roc\_auc\_score from slearn libraries the code will display it in a graph form.

### **4.13.5 Feature Importance**

A computer screen with colorful text

Description automatically generated

Figure 4. 21 Feature Importance code snippet.

Scikit\_learn provide “Feature importance” which is used to calculate the feature importance in a model.

## **4.14. K NEAREST NEIGHBOUR – MODEL 3**

****

Figure 4. 22 K nearest neighbour code snippet

The figure above shows how to import the K nearest neighbour classifier algorithm from sklearn libraire, a Python library for machine learning algorithms, with the nearest neighbour set as 5 and the default metrixs = Minkowski with standard Euclidean distance p = 2.

### **4.14.1 Algorithm to Determine the Best Nurmber of K**

**A computer screen with text and images

Description automatically generated**

Figure 4. 23 Determine the best K code snippet.

This code is used to determine the optimal number of k before building the model.

### 

### **4.14.2 Confusion Matrix**

A screen shot of a computer program

Description automatically generated

Figure 4. 24 Confusion Matrix code snippet.

This shows the number of accurate and inaccurate predictions performed by the model in a box form, as well as the accuracy, precision, and recall f1-score. The confusionMatrixDisplay module is imported from the sklearn library, while the visualization is done using the lib module from matplotlib.

### **4.14.3 AUC-ROC**

A computer screen shot of a program code

Description automatically generated

Figure 4. 25 AUC & ROCcode snippet.

Area under the curve is imported as roc\_curve and roc\_auc\_score from sklearn.metrics libraries the code will display it in a graph form.

### 

### **4.14.4 Feature Importance**

A screen shot of a computer code

Description automatically generated

Figure 4. 26 Feature importance.

Scikit\_learn provide “Feature importance” which is used to calculate the feature importance in a model.

## **4.15. LOGISTIC REGRESSION MODEL 4**

A computer screen with colorful text

Description automatically generated

Figure 4. 27 Logistic Regression code snippet.

The figure above shows the importing import of Logistic Regression algorithm from sklearn libraire.

### 

### **4.15.1 Confusion Matrix**

A computer screen shot of a program code

Description automatically generated

Figure 4. 28 Confusion Matrix code snippet.

This shows the number of accurate and inaccurate predictions performed by the model in a box form, as well as the accuracy, precision, and recall f1-score. The confusionMatrixDisplay module is imported from the sklearn library, while the visualization is done using the lib module from matplotlib.

### 

### **4.15.2 Cross Validation**

A computer screen shot of a program code

Description automatically generated

Figure 4. 29 Cross Validation code snippet.

Cross Validation score imported from sklearn library.

### 

### **4.15.3 AUC-ROC**

A computer screen with text

Description automatically generated

Figure 4. 30 AUC & ROCcode snippet.

Area under the curve is imported as roc\_curve and roc\_auc\_score from slearn libraries the code will display it in a graph form.

## **4.16 SUPPORT VECTOR MACHINE MODEL 5**

A computer screen with colorful text

Description automatically generated

Figure 4. 31 Support Vector machine kernel – linear code Snippet

The figure above shows the importing import of Support Vector Machine algorithm from sklearn libraire also set the Kernel to linear.

### **4.16.1 SVM Model Changing The Kernel**

A computer screen with text

Description automatically generated

Figure 4. 32 Support Vector Machine kernel – poly code snippet

The figure above shows the importing import of Support Vector Machine algorithm from sklearn libraire also set the Kernel to poly.

### **4.16.2 Confusion Matrix**

A computer screen with text and images

Description automatically generated

Figure 4. 33 Confusion Matrix code snippet.

This shows the number of accurate and inaccurate predictions performed by the model in a box form, as well as the accuracy, precision, and recall f1-score. The confusionMatrixDisplay module is imported from the sklearn library, while the visualization is done using the lib module from matplotlib.

### **4.16.3 AUC-ROC**

A computer screen with text

Description automatically generated

Figure 4. 34 AUC & ROCcode snippet.

Area under the curve is imported as roc\_curve and roc\_auc\_score from slearn libraries the code will display it in a graph form.

# **CHAPTER 5- RESULTS AND DISCUSSION**

## **5.1 OVERVIEW**

This chapter presents the study's results and analysis, In addition, any unexpected discoveries or constraints encountered during the study would be thoroughly examined. This examination offers insight into the application of machine learning to understand the complex dynamics of credit card fraud.

## **5.2. SETTING WORKING DIRECTORY**

**A screenshot of a computer

Description automatically generated**

Figure 5. 1 Setting Work Directory

Setting the python 3.10 with visual studio code as an integrated development environment (IDE).

## **5.3. IMPORT THE LIBRARY**

**A computer screen shot of text

Description automatically generated**

Figure 5. 2 Import Library

The figure 5.2 shown the importing of the necessary python library such as NumPy, pandas, seaborn and matplotlib which are used to analysis, split, and built the model.

## 

## **5.4. CHECKING THE STRUCTURE OF THE DATASET**

A computer screen with white text

Description automatically generated

Figure 5. 3 Structure of Data

Above is figure 5.3 shows the inspection of the data structure within the image, where the RangeIndex displays one million entries, indicating an observation. The data column contains a total of 8 variables, the Non-Null Count indicates that there are no non-null values, and the Dtype refers to the data type, float64, which is a numerical value with a decimal place number using 64 bits of memory. Working with numerical data allows for quantitative analysis, statistical modelling, visual chart creation, and computational analysis to inform more informed decision-making. Understanding the structure of data is crucial for successful modelling. It helps in choosing the right techniques and ensures that models accurately capture the patterns in the data.

## 

## **5.5. CHECKING FOR MISSING VALUES**

A close-up of a sign

Description automatically generated

Figure 5. 4 Missing Values

A blue rectangular object with white text

Description automatically generated

Based on the analysis in figure 5.4, missing data is represented by the colour black, and the percentage would be shown, while the dataset is represented by the colour blue, also with-it percentage. If there are any missing values in the data, they will be displayed as black. It appears that there is no missing data present in the dataset, as the percentage of missing data is 0%. Ensuring this is crucial, as it plays a vital role in the dependability and efficiency of the modelling process.

## **5.6. BOXPLOT OF ORIGINAL DATASET**

A graph with lines and a number

Description automatically generated with medium confidence

Figure 5. 5 Boxplot of original dataset.

As seen in figure 5.5 show the box plot provides a visual summary of the distribution of each feature. Additionally, the line within the box signifies the median (Q2, 50th percentile) of the data. Features such as distance\_from\_home, distance\_from\_last\_transaction, and ratio\_to\_median\_purchase\_price show significant variability and numerous outliers, indicating that these variables are key differentiators in the model; other features appear to have less variability.

## **5.7. CORRPLOT**

A graph with numbers and words

Description automatically generated with medium confidence

Figure 5. 6 Correlation Plot

The spearman correlation analysis in the above figure 5.6 shows the relationship between two or more variables. According to the analysis, distance\_from\_home and repeat\_retailer have a moderately positive correlation of 0.56, whereas ratio\_to\_median\_purchase\_price has a weak positive correlation of 0.34. This is done to understand the statistical relationship the variables.

## **5.8. DISTRIBUTION OF THE DEPENDENT VARIABLE**

A graph with a bar and a number of blue squares

Description automatically generated with medium confidence

Figure 5. 7 Distribution of dependent variable

The bar chat above illustrates the distribution of the dependent variable, which includes 912,597 non-fraudulent transactions and 87,403 fraudulent transactions. This significant difference in the number of classes could have an impact on the model's ineffective accuracy.

## **5.9. DISTRIBUTION OF THE DEPENDENT VARIABLE AFTER OVERSAMPLIN**

A graph showing a number of different colored squares

Description automatically generated

Figure 5. 8 Balance data distribution

As shown above, the dependent variable now has the same proposition. This was done with the oversampling technique "SMOTE," or synthetic minority oversampling, which is specific to managing imbalance data by creating synthetic samples for the majority class.

## 

## **5.10. Normalization**

A table of numbers and letters

Description automatically generated

Figure 5. 9 Before Normalization

The figures above, which show the data before normalization, are all numerical, with some in decimal places and the rest in whole numbers. This discrepancy in length could potentially affect the performance of the model.

A table of numbers and lines

Description automatically generated with medium confidence

*Figure 5.9.1 After Normalization*

Figure 5.5 displays the data after normalization, with the exception of the dependent variable, which should remain unnormalized. This is done to ensure that the values are aligned and fall within the appropriate range. This process normalizes the variables, ensuring that their mean is 0 and their standard deviation is 1. Scaling is crucial for achieving balanced feature contribution, leading to improved prediction accuracy.

## **5.11. Splitting Data training and testing**

A black background with white text

Description automatically generated

Figure 5. 10 Split train and test

The data was spitted into 70% training and 30% testing.

## 

## **5.12. MODEL 1 RANDOM FOREST**

The random forest model was developed two times: the first algorithm was built by setting a set of rules parameters that prevent the model from overfitting, after the confusion matrix was used to evaluate the performance of the model shown below figure 5.11. The second model was then generated without setting any parameters, also using the confusion matrix to evaluate the internal performance shown below figure 5.12. Both models show explicit performance, but model one has been selected because it was developed on regulating the parameter to help prevent overfitting: type I error = 1 and type II error = 6. In figure 5.13 the model is 100%, precision 100%, recall 100% and f1-score 100%.

**5.12.1 Evaluation of Model 1**

A graph showing a number of forest confusion matrix

Description automatically generated

Figure 5. 11 Random Forest Model 1 Regulating parameter Confusion Matrix

The confusion matrix shown above for the random forest model one with regularization parameters technique applied. The confusion matrix breaks down the number of correct and incorrect predictions by each class. The True Negative has a count of 273,946, indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 1 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 4, which was predicted as a fraudulent (1) transaction but was not a fraud.  The True Positive has a count of 273,608, showing that the model predicted fraudulent cases (1) and was correct.

|  |  |
| --- | --- |
| Type I Error | 1 |
| Type ii Error | 6 |

**A graph showing a number of forest confusion matrix

Description automatically generated**

Figure 5. 12 Random Forest Model 2 Confusion Matrix

The confusion matrix, displayed above for the random forest model 2 without regularization parameters technique, summarizes the number of correct and incorrect predictions for each class. The True Negative has a count of 273,946, indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 1 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 4, which was predicted as a fraudulent (1) transaction but was not a fraud. The True Positive has a count of 273,608, showing that the model predicted fraudulent cases (1) and was correct.

|  |  |
| --- | --- |
| Type I Error | 1 |
| Type ii Error | 4 |

A close-up of a number

Description automatically generated

Figure 5. 13 Classification Report

The performance of random forest model 1 accuracy 100%, precision 100%, recall 100% and f1-score 100%.

**A graph of a random roc curve

Description automatically generated**

Figure 5. 14 . Random Forest AUC-ROC

The figure 5.14 above represents a graphical demonstration of the overall performance of model 1, specifically the ability to differentiate between positive and negative classes. The Roc curve plots the true positive rate against the false positive rate at various threshold settings. A good model will have a curve that is closer to the top-left corner, ranging from 0.6 to 1.0.

The graph above illustrates the True Positive on the vertical axis, the False Positive on the horizontal axis, and the ROC, symbolized by the blue line at the top. The final result by the downright path of the graph, ROC curve = 100%, which means the model performance is very good.

**A graph with blue squares

Description automatically generated**

Figure 5. 15 Feature Importance

The analysis revealed that the ratio\_to\_median\_purchase\_price, distance\_from\_home, and online\_order are the key factors in this model used to differentiate fraudulent and non-fraudulent transactions. However, it identified the ratio\_to\_median\_purchase\_price as the most influential feature, suggesting that credit card fraud likely involves a large purchase to deduct all the money in a single transaction, potentially detecting credit card fraud.

Distance\_from\_last\_transaction, used\_chip, used\_pin\_number and repeat\_retailer show a less contribution in the prediction of the random forest model because there are less than one on the importance score.

## 

## **5.13 MODEL 2 DECISION TREE**

The decision tree model was developed three times: the first model was built by letting the tree grow to the fullest without setting any parameters; the second model was built by pruning the tree, limiting the maximum number of trees and nodes to help the model prevent overfitting; and the third model was built with pruning of the tree, just like the second model, but adding the model should consider the square root number when looking for the best split. The confusion matrix was used to evaluate the performance of each model; model one and model two had a very low misclassification error, as shown in figure 5.17 and figure 5.18, but model 3 in figure 5.19 has a very large misclassification error. After evaluating each model and comparing their performance, model 2 is selected as the optimal model with a type I error (2) and a type II error (6) and an accuracy of 100%. Also, the ROC-AUC 100% regularizing technique was applied to this model to prevent overfitting.

### **5.13.1 Evaluation of Model 2**

**A number of numbers on a white background

Description automatically generated**

Figure 5. 16 Classification Report

The performance of random forest model 1 accuracy 100%, precision 100%, recall 100% and f1-score 100%.

**A blue squares with numbers and a graph

Description automatically generated**

*Figure 5. 17 Confusion Matrix Decision tree model 1*

The diagram above displays the confusion matrix of the decision tree model 1 which states the summary of correct and incorrect predictions done by the model broken down by class. The true negative with a count of 273,946, indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 1 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 7, which was predicted as a fraudulent (1) transaction but was not a fraud. The True Positive has a count of 273,605, indicating that the model correctly predicted fraudulent cases (1).

**A blue squares with numbers and a graph

Description automatically generated**

Figure 5. 18 Confusion Matrix Pruning Decision Tree model 2

The confusion matrix shown above for decision tree model 2, which incorporates the concept of using regularization parameters to prune the tree. It provides a concise summary of the number of accurate and inaccurate predictions, categorized by each class. The True Negative has a count of 273,945, indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 2 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 6, which was predicted as a fraudulent (1) transaction but was not a fraud.  The True Positive has a count of 273,606, showing that the model predicted fraudulent cases (1) and was correct.

**A blue squares with numbers and a graph

Description automatically generated**

*Figure 5. 19* *Confusion Matrix Pruning Decision Tree Model 3*

The confusion matrix shown above for decision tree model 3, which applies regularization parameters to prune the tree development. This particular tree considers the square root number when looking for the best split. The confusion matrix provides a summary of the number of accurate and inaccurate predictions, categorized by each class. The True Negative count is 267,287, demonstrating that the model accurately predicted the non-fraudulent class (0). The instance with a count of 6660 was incorrectly labelled as fraudulent (1) but was actually not fraudulent. The count of false positives is 299, indicating instances where a transaction was predicted as fraudulent (1) but was actually not a fraud. The True Positive count is 273,313, indicating that the model accurately predicted occurrences of fraud (1).

**A graph of a tree roc curve

Description automatically generated**

*Figure 5. 20* *Decision Tree AUC-ROC*

The figure above represents a graphical demonstration of the Receiver Operating Characteristic (ROC) of model 2 which shows the overall performance, specifically the ability to differentiate between positive and negative classes. The Roc curve plots the true positive rate against the false positive rate at various threshold settings. A good model will have a curve that is closer to the top-left corner, ranging from 0.6 to 1.0.

The graph above illustrates the True Positive on the vertical axis, the False Positive on the horizontal axis, and the ROC, symbolized by the blue line. The final result by the downright path of the graph, ROC curve = 1.000, which means the model performance is very good.

**A graph with blue squares

Description automatically generated**

Figure 5. 21 Feature Importance

The analysis revealed that the ratio\_to\_median\_purchase\_price, distance\_from\_home, and distance\_from\_last\_transaction are the key factors in this model used to differentiate fraudulent and non-fraudulent transactions. However, it identified the ratio\_to\_median\_purchase\_price is the most influential feature, suggesting that credit card fraud likely involves a large purchase to deduct all the money in a single transaction, potentially detecting credit card fraud.

Online\_order, used\_chip, used\_pin\_number and repeat\_retailer show a less contribution in the prediction of the random forest model because there are less than one on the importance score.

## **5.14 MODEL 3 K NEAREST NEIGHBOUR**

K Nearest Neighbour was firstly built on an algorithm that tries each number of k on the dataset from the range of 1 of 40. After the output of the algorithm, which is K = 5 as the optimal number, the model is built on that, achieving high accuracy, precision, recall, and AUC in Figure 5.34, making it effective at identifying fraudulent transactions. However, the model also misclassified a significant number of non-fraudulent transactions as fraudulent (129, false positives), which could lead to unnecessary alarms and possibly inconvenience customers. Moreover, there are some instances (270, false negatives) where fraudulent transactions were not detected, which could lead to financial loss.

**A black and white text

Description automatically generated**

**A graph with red dots and blue dots

Description automatically generated**

*Figure 5. 22 Determine best K*

The algorithm above in figure 5.22 shows the Error rate and number of k from the range of 0 – 40, the higher the red don line the error in the number of k. given the output above as k = 0.5 – 5 is the optimal number for k for the dataset.

### 

### **5.14.1 Evaluation of Model 3**

**A close-up of a number

Description automatically generated**

*Figure 5. 23 Classification Report*

The performance of random forest model 1 accuracy 100%, precision 100%, recall 100% and f1-score 100%.

A graph of a number of blue squares

Description automatically generated with medium confidence

Figure 5. 24 . K Nearest Neighbour Confusion Matrix

The confusion matrix, displayed above for the K nearest neighbour summarizes the number of correct and incorrect predictions for each class. The True Negative has a count of 273,742 indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 129 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 270, which was predicted as a fraudulent (1) transaction but was not a fraud. The True Positive has a count of 258,59 showing that the model predicted fraudulent cases (1) and was correct.

|  |  |
| --- | --- |
| Type I Error | 129 |
| Type ii Error | 270 |

A graph of a positive rate

Description automatically generated with medium confidence

*Figure 5. 25 K Nearest Neighbour AUC-ROC*

The diagram shown above illustrates a visual representation of the overall efficacy of the model, particularly its capacity to distinguish between positive and negative class. The Roc curve plots the true positive rate against the false positive rate at various threshold settings. An optimal model will have a curve that is positioned in close proximity to the top-left corner, with values ranging from 0.6 to 1.0.

The graph above illustrates the True Positive on the vertical axis, the False Positive on the horizontal axis, and the ROC, symbolized by the blue line at the top. The final result by the downright path of the graph, ROC curve = 0.99, which means the model performance is very good.

A graph with blue squares

Description automatically generated

*Figure 5. 26 Feature Importance*

The analysis revealed that the ratio\_to\_median\_purchase\_price, online\_order and distance\_from\_home are the key factors in this model used to differentiate fraudulent and non-fraudulent transactions. However, it identified the ratio\_to\_median\_purchase\_price as the most influential feature, suggesting that credit card fraud likely involves a large purchase to deduct all the money in a single transaction, potentially detecting credit card fraud.

Distance\_from\_last\_transaction, used\_chip, used\_pin\_number and repeat\_retailer show a less contribution in the prediction of the random forest model because there are less than one on the importance score.

## 

## **5.15 MODEL 4 LOGISTIC REGRESSION**

The logistic regression model has robust performance in forecasting credit card fraud, achieving high levels of accuracy, precision, recall, and an exceptional AUC score (Figure 5.27). Although there are a few misclassifications (false positives and false negatives), the model is generally very dependable in distinguishing between fraudulent and non-fraudulent transactions, as shown in figure 5.28, and a cross-validation was done to ensure the model was performing well on the training and testing set; the average score of the validation is 92%. As a result, it can be used effectively in fraud detection systems, potentially leading to substantial decreases in financial losses caused by fraud.

### **5.15.1 Evaluation of Model 4**

**A graph showing the difference between a logistic regression and a confusion matrix

Description automatically generated**

Figure 5. 27 Logistic Regression Confusion Matrix

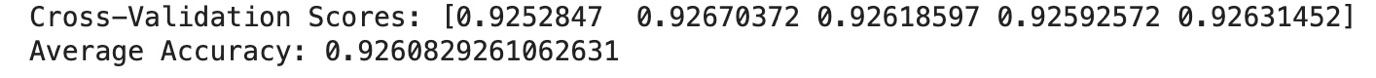
The logistic regression confusion matrix displayed above for the summarizes the number of correct and incorrect predictions for each class. The True Negative has a count of 253,137 indicating that the model predicted the non-fraudulent class (0) and was correct. The false negative with a count of 20810 was misclassified as a fraudulent (1) case but was not fraudulent. False positives give a count of 20204, which was predicted as a fraudulent (1) transaction but was not a fraud. The True Positive has a count of 253,403 showing that the model predicted fraudulent cases (1) and was correct.

**A number of numbers on a white background

Description automatically generated**

*Figure 5. 28 Classification Report*

The performance of logistic regression model 1 accuracy 93%, precision 100%, recall 100% and f1-score 100%.

*****Figure 5. 29 Cross Validation*

Cross validation average accuracy 92% ensuring that the model generalize well and not over fitting.

**A graph of a logistic regression curve

Description automatically generated**

*Figure 5. 30 Logistic Regression AUC-ROC*

The figure above shows the ROC curve for the Logistic regression model, and the diagonal line is the AUC score, which is 0.97%, indicating a very high accuracy of the model for being able to differentiate between the positive class (fraudulent transactions) and the negative class (non-fraudulent transactions).

## **5.16 MODEL 5 SUPPORT VECTOR MACHINE**

The Support vector machine algorithm was built in two different models, changing the kernel function to "linear" and "poly," but each of the models shows a long run time of 15 hours each. However, the model shows strong performance in Figure 5.33, with high accuracy, precision, recall, and AUC in Figure 5.34, making it effective at identifying fraudulent transactions. However, the model also misclassified a significant number of non-fraudulent transactions as fraudulent (19,291 false positives), which could lead to unnecessary alarms and possibly inconvenience customers. Moreover, there are some instances (7,611 false negatives) where fraudulent transactions were not detected, which could lead to financial losses.

### **5.16.1 Evaluation of Model 5**

**A blue and white chart

Description automatically generated**

Figure 5. 31 Svm confusion matrix kernel – linear

Displayed above in figure 5.31 is the confusion matrix for the support vector machine, which provides a summary of the number of correct and incorrect predictions for each class. With a count of 254,446, the True Negative signifies that the model accurately predicted the non-fraudulent class (0). There was a misclassification of a false negative with a count of 19628, which wrongly labelled it as a fraudulent (1) case when it was actually not fraudulent. There were 8941 false positives, meaning that 8941 transactions were incorrectly predicted as fraudulent when they were actually legitimate. With a count of 264544, the True Positive indicates that the model accurately identified and predicted fraudulent cases.

**A blue and white graph

Description automatically generated**

*Figure 5. 32 Svm confusion Matrix kernel – poly*

The confusion matrix, depicted above for the support vector machine, provides a concise summary of the accurate and inaccurate predictions made for each class. The True Negative count is 254,783, demonstrating that the model accurately predicted the non-fraudulent class (0). The instance with a false negative count of 19291 was erroneously labelled as a fraudulent case (1), despite not being fraudulent. The count of false positives is 7611, indicating transactions that were forecasted as fraudulent (1) but were really not fraudulent. The True Positive count is 265,874, indicating that the model accurately anticipated occurrences of fraud (1).

**A screenshot of a computer screen

Description automatically generated**

Figure 5. 33 Classification Report

Finally, the support vector shown figure 17, scored an accuracy of 95%, precision recall 93%, recall 97% and f1-score 95%.

**A graph with a line

Description automatically generated**

Figure 5. 34 SVM AUC-ROC curve

The figure above shows the ROC curve for the Support Vector Machine model, and the diagonal line is the AUC score, which is 0.99%, indicating a very high accuracy of the model for being able to differentiate between the positive class (fraudulent transactions) and the negative class (non-fraudulent transactions).

## **5.17 COMPARISONS OF ALL MODEL**

|  |  |  |
| --- | --- | --- |
| **Model** | **Accuracy** | **AUC-ROC** |
| Random Forest | 100% | 100% |
| Decision Tree | 100% | 100% |
| K Nearest Neighbour | 100% | 99% |
| Logistic Regression | 93% | 97% |
| Support Vector Machine | 95% | 99% |

Table 5. 1 Comprising all Model

Table 5.1 shows all of the accuracy and AUC of all the models that were built in the project; all models performed well in detecting fraudulent and non-fraudulent transactions and managed to score high accuracy. Out of all the algorithms, the ones that scored the best are random forest and decision tree, as their accuracy is 100% and the AUC is 100%, and then the K nearest model also achieves an accuracy score of 100% with an AUC of 99%, logistic regression accuracy is 93% and the AUC 97%, and support vector machine accuracy is 95% and the AUC 99%.

|  |  |  |
| --- | --- | --- |
| Model | Type I | Type II |
| Random Forest | 1 | 6 |
| Decision Tree | 2 | 6 |
| K Nearest Neighbour | 129 | 270 |
| Logistic Regression | 20810 | 20204 |
| Support Vector Machine | 19291 | 7611 |

Table 5. 2 Comprising Error of All models

Table 5.2 compares the performance of the model with the confusion matrix because they all demonstrate similar accuracy in Table 5.1 above. The confusion matrix model 1 random forest with a Type I error of 1 and Type II error 6, decision tree Type I error 2 and Type II error 6, and k nearest neighbour has the most misclassified Type I error of 129 and Type II error 270, followed by the logistic regression Type I error 20810 and Type II 20204, and support vector machine Type I error 19291 and Type II error 7611.

In conclusion, model 1 Random Forest is the best-performing model, with an accuracy of 100% and an AUC-ROC of 100%, while also maintaining less type I of 1 and type II error of 6.

## **5.17 CRITICAL EVALUATION OF RESULTS**

All the models had very good performance, but random forest stood out with an accuracy of 100%, an AUC-ROC of 100%, and Type I error "1" and Type II error "6". The decision tree also has an accuracy of 100%, AUC-ROC 100%, and Type I error "2" and Type II error "6."Close observation on the KNN, logistic regression and SVM models with their high accuracy and AUC-ROC scores but manage to produce large numbers of mis classification error above; that is why the confusion matrix was used for evaluation in table 5.2 . And also, identifying the problem of overfitting the model was built on regularization techniques to ensure the model's robustness and generalizability.

# **CHAPTER 6- CRITICAL EVALUATION AND CONCLUSION**

## **6.1. OVERVIEW**

In this final phase of the study, the effectiveness of machine learning algorithms in predicting credit card fraud is delved into. The assessment focuses on their capability to accurately forecast outcomes and the identification of areas where limitations may arise. Additionally, an examination of the factors influencing prediction, along with proposals for enhancing accuracy. By making the findings, an evaluation of the practicality of employing machine learning for this purpose is made.

## **6.2. LIMITATIONS**

While this study uses machine learning techniques to detect credit card fraud, several limitations should be noted: First, the dataset consists of high-imbalance data, which may cause a bias model toward non-fraudulent transactions. The study relied on a single dataset sourced from Kaggle, which limits the generalizability of the findings.

The SVM model was not fully explored because of the runtime taking up to 1344 minutes to run one kernel, and there are 5 kernels in the algorithm, which requires a large processing power. Applying the sampling technique should be carefully examined toward the building of the KNN model because it could cause some shift in the picking of the nearest neighbour, making a lot of misclassification errors. Further, the risk of overfitting should be considered because the models may overfit to training data, leading to poor performance on unseen data.

Despite these limitations, this study provides a foundational framework for understanding and detecting credit card fraud, highlighting avenues for further investigation and refinement in the field of fraud detection and machine learning.

## 

## **6.3. CONCLUSION**

In conclusion, this dissertation explores the efficacy of machine learning techniques in detecting credit card fraud in the financial sector. By implementing and evaluating several machine algorithms, such as random forest, decision tree, K-nearest neighbours, logistic regression and support vector machine, this study aimed to identify the most effective method for identifying fraudulent and non-fraudulent transactions.

This study reveals that the random forest algorithm is the best performing model, achieving an accuracy of 100% along with the lowest Type I of 1 and Type II errors of 6. This suggests that Random Forest, with its ensemble approach, is particularly well-suited for handling the complex patterns inherent in credit card fraud detection. The decision tree also came close to the same performance of Random Forest, given an accuracy of 100%. Although other models like K-Nearest Neighbours, Logistic Regression and support vector also performed well in terms of accuracy, they were less consistent in minimizing misclassification errors, highlighting the importance of selecting appropriate evaluation metrics beyond just accuracy. The support vector machine had a long run time of over 1388 minute on each kernel and still product a large mis classification error, which could be conclusion that it is not s suitable model for this data set.

The results of this study underscore the potential of machine learning to enhance fraud detection systems, providing financial institutions with tools that are both accurate and efficient. However, it became apparent that machine learning models are sensitive to the data's quality and the selected evaluation criteria. Problems like as data imbalance and overfitting were resolved using techniques like SMOTE and regularization. However, these difficulties highlight the importance of carefully model tuning and validation in practical scenarios.

In conclusion, this study examines the expanding knowledge on fraud detection by demonstrating the effectiveness of machine learning algorithms, specifically Random Forest, to mitigate credit card fraud. It offers a strong foundation for future study and practical application, with the ability to greatly decrease financial losses and improve the security of electronic transactions in the financial industry.

## **6.4. RECOMMENDATION**

In the future, it would be valuable to explore more machine learning methods to enhance prediction accuracy. Additionally, there are many ways to improve the model, such as using it on different datasets with various sizes and data types, changing the data splitting ratio, viewing it from a different algorithm perspective, and performing different types of sampling techniques.

## **6.5. PERSONAL EVALUATION OF EXPEREIENCE**

The first 3 weeks of the weekly meeting were a bit confusing because, wasn’t getting the fundamental understanding of how to organize and assemble the project; Was just fully focused on the implementation and result. But on the fourth week, was able to grasp these 3 key factors from my supervisor: “What are you doing?"  “Why are you doing it?” and “How will you do it?” These key words help in putting together my chapter one. Furthermore, the organization of a dissertation was outlined, consisting of six chapters: introduction, literature review, methodology, implementation, results analysis, and conclusion.

Chapter two of the project was done, but when presented to the supervisor, it got cancelled because it was not reflexing the topic of the project itself. After a full explanation, it understood that the topic of credit card fraud detection should be broken down into individuals and research should be conducted on them and then put into the work explaining what they are. The remaining chapter went well because a fully understanding of how to work the project was gotten after finishing section two.

Overall, the weekly section helps very much in understanding how to write and organize this project. Also fully gained the ability to multitask because most of the of the time, would be at work, but when it was time for my meeting,  would take a break, have the meeting, and go back to work. Despite these challenges, successful completion of the project was achieved, demonstrating effective time management, multitasking abilities, and focus.

# **REFERENCES**

Abzalov, M. (2016) ‘Exploratory Data Analysis’, in M. Abzalov (ed.) *Applied Mining Geology*. Cham: Springer International Publishing, pp. 207–219. Available at: https://doi.org/10.1007/978-3-319-39264-6\_15.

Benson Edwin Raj, S. and Annie Portia, A. (2011) ‘Analysis on credit card fraud detection methods’, in *2011 International Conference on Computer, Communication and Electrical Technology (ICCCET)*. *2011 International Conference on Computer, Communication and Electrical Technology (ICCCET)*, pp. 152–156. Available at: https://doi.org/10.1109/ICCCET.2011.5762457.

Bertsimas, D. and Dunn, J. (2017) ‘Optimal classification trees’, *Machine Learning*, 106(7), pp. 1039–1082. Available at: https://doi.org/10.1007/s10994-017-5633-9.

Bhatla, T.P., Prabhu, V. and Dua, A. (2003) ‘Understanding Credit Card Frauds’.

Edward Bellamy, E.B. (1887) *Edward Bellamy First Uses the Term ‘Credit Card’ in ‘Looking Backward’ : History of Information*. Available at: https://www.historyofinformation.com/detail.php?entryid=2044 (Accessed: 16 July 2024).

Guo, G. *et al.* (2003) ‘KNN Model-Based Approach in Classification’, in R. Meersman, Z. Tari, and D.C. Schmidt (eds) *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*. Berlin, Heidelberg: Springer, pp. 986–996. Available at: https://doi.org/10.1007/978-3-540-39964-3\_62.

Jennifer Rosenberg (202AD) *Incredible True Story of the Very First Credit Card*, *ThoughtCo*. Available at: https://www.thoughtco.com/the-first-credit-card-1779328 (Accessed: 31 July 2024).

John O. Awoyemi, A.O.A.S.A.O. (2017) *Credit card fraud detection using machine learning techniques: A comparative analysis*. Available at: https://ieeexplore.ieee.org/abstract/document/8123782 (Accessed: 29 August 2024).

Khatri, S., Arora, A. and Agrawal, A.P. (2020) ‘Supervised Machine Learning Algorithms for Credit Card Fraud Detection: A Comparison’, in *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 680–683. Available at: https://doi.org/10.1109/Confluence47617.2020.9057851.

Kubat Miroslav (1996) *An Introduction to Machine Learning*. Available at: https://doi.org/10.1007/978-3-030-81935-4.

Lance Cothern CPA, L.C. (2019) *When was the credit card invented?*, *Intuit Credit Karma*. Available at: https://www.creditkarma.com/credit-cards/i/credit-card-history (Accessed: 1 August 2024).

Michie, D. (1998) ‘Learning concepts from data’, *Expert Systems with Applications*, 15(3), pp. 193–204. Available at: https://doi.org/10.1016/S0957-4174(98)00044-X.

Nadim, A.H. *et al.* (2019) ‘Analysis of Machine Learning Techniques for Credit Card Fraud Detection’, in *2019 International Conference on Machine Learning and Data Engineering (iCMLDE)*. *2019 International Conference on Machine Learning and Data Engineering (iCMLDE)*, pp. 42–47. Available at: https://doi.org/10.1109/iCMLDE49015.2019.00019.

Naik, H. and Kanikar, P. (2019) ‘Credit card Fraud Detection based on Machine Learning Algorithms’, *International Journal of Computer Applications*, 182(44), pp. 8–12.

Nick, T.G. and Campbell, K.M. (2007) ‘Logistic Regression’, in W.T. Ambrosius (ed.) *Topics in Biostatistics*. Totowa, NJ: Humana Press, pp. 273–301. Available at: https://doi.org/10.1007/978-1-59745-530-5\_14.

Rathore, A.S. *et al.* (2021) ‘Credit Card Fraud Detection using Machine Learning’, in *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*. *2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART)*, pp. 167–171. Available at: https://doi.org/10.1109/SMART52563.2021.9676262.

S Ishwarya (2024) ‘Unveiling the Magic of Machine Learning: Core Techniques Explained’, *Medium*, 4 June. Available at: https://ishwaryasriraman.medium.com/unveiling-the-magic-of-machine-learning-core-techniques-explained-f3c77a49267d (Accessed: 31 July 2024).

Sadgali, I., Sael, N. and Benabbou, F. (2019) ‘Performance of machine learning techniques in the detection of financial frauds’, *Procedia Computer Science*, 148, pp. 45–54. Available at: https://doi.org/10.1016/j.procs.2019.01.007.

Sumanth, C.H. *et al.* (2022) ‘Analysis of Credit Card Fraud Detection using Machine Learning Techniques’, in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*. *2022 7th International Conference on Communication and Electronics Systems (ICCES)*, pp. 1140–1144. Available at: https://doi.org/10.1109/ICCES54183.2022.9835751.

Trivedi, N. *et al.* (2020) ‘An Efficient Credit Card Fraud Detection Model Based on Machine Learning Methods’, *MATTER: International Journal of Science and Technology* [Preprint].

Vaishnavi Nath Dornadula and Geetha, S. (2019) ‘Credit Card Fraud Detection using Machine Learning Algorithms’, *Procedia Computer Science*, 165, pp. 631–641. Available at: https://doi.org/10.1016/j.procs.2020.01.057.

Varma, Prof.T. *et al.* (2021) ‘CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM’, *International Journal Of Trendy Research In Engineering And Technology*, 05(03), pp. 24–27. Available at: https://doi.org/10.54473/IJTRET.2021.5305.

Varmedja, D. *et al.* (2019) ‘Credit Card Fraud Detection - Machine Learning methods’, in *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*. *2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pp. 1–5. Available at: https://doi.org/10.1109/INFOTEH.2019.8717766.

# 