

A dissertation submitted to the **University of Greenwich**
in partial fulfilment of the requirements for the Degree of

Master of Science

in

Data Science

**Fraud Detection System for Online Transactions using
Machine Learning**

Name: Richard Raja

Student ID: 001370307

Supervisor: Dr. Samiya Khan

Submission Date: 20 Dec 2025

Word count: 16255

Fraud Detection System for Online Transactions using Machine Learning.

Computing & Mathematical Sciences, University of Greenwich, 30 Park Row, Greenwich, UK.

(Submitted 20 Dec 2025)

Abstract

This project focuses on this area of study whereby an attempt shall be made to use Logistic Regression as well as Decision Trees to estimate the possibility of fraudulent online transaction. As the fraudsters become smarter and the number of online transactions increases, the use of traditional measures of fraud detection was not sufficient, hence using the machine learning technology. This paper compares Logistic Regression and Decision Trees, pointing out their effectiveness and drawbacks in accordance with the precision and the recall and especially in terms of Interpretability. The research suggests using both the techniques simultaneously, and identifies the strategies including ensemble approaches, handling of imbalance problem and model explainability as the future research directions. Accordingly, the major implications of the study are essentially about the necessity of the real-time processing and the ethical factors in establishing sustainable fraud detection platforms. They provide useful suggestions for professionals and present the basis for additional investigations in fraud identification and artificial intelligence.

Keywords: Fraud Detection, Logistic Regression, Decision Tree, Seaborn, Matplotlib, Real time analysis, accuracy, precision, recall, F1-score, AUC-ROC, correlation matrix, Predictive Models, Credit Card Fraud, Logistic Regression, Decision Tree, anomaly detection.

Acknowledgements

I would especially like to thank Dr Samiya Khan for agreeing to be my supervisor and for her consistent advice, feedback, guidance and support throughout the lifecycle of this MSc Fraud Detection System for Online Transactions using Machine Learning project.

I want to thank both **Dr Samiya Khan** and **Professor Jixin Ma** for agreeing to have the project demonstration on the schedule day.

Contents

Abstract.....	ii
List of Figures	vi
List of Tables.....	vi
Chapter 1: Introduction	1
1.1. Background of the Study	1
1.2. Problem Statement	6
1.3. Aim	7
1.4. Objectives	7
1.5. Research questions.....	7
1.6. Significance of the Study	8
1.7. Scope of the Study.....	8
1.8. Organization of the Study.....	9
Chapter 2 Literature Review	10
2.1. Early Detection of Fraud: Using Logistic Regression Models	13
2.2. Credit Card Fraud Detection with the use of Neural Networks	13
2.3. Decision Trees for Fraudulent Transactions Discovery	14
2.4. Antecedents of Fraudulent Behaviour: Applying Random Forests	15
2.5. Use of Support Vector Machines for Predicting Cases of Fraud.	16
2.6. K Nearest Neighbors in Fraudulence Detection	17
2.7. Naïve bayes for fraud detection	17
Chapter 3 Methodology	19
3.1 Introduction	19
3.2 Data Collection and Preprocessing.....	19

3.3 Logistic Regression Model	20
3.3.1 Model Selection and Justification	20
3.3.2 Model Training.....	21
3.3.3 Model Evaluation	22
3.3.4 Implementation Considerations	22
3.4 Decision Tree Model	23
3.4.1 Model Selection and Justification	24
3.4.2 Model Training.....	24
3.4.3 Model Evaluation	25
3.4.4 Implementation Considerations	26
3.5 Comparison of Models.....	27
3.6 Conclusion.....	27
Chapter 4 Result and Discussion.....	29
4.1 Introduction	29
4.2 Performance of the Logistic Regression Model	34
4.2.1 Accuracy and Precision.....	34
4.2.2 Recall and F1-Score	35
4.2.3 AUC-ROC Analysis	35
4.3 Performance of the Decision Tree Model.....	36
4.3.1 Accuracy and Recall	36
4.3.2 Precision and F1-Score	37
4.3.3 AUC-ROC Analysis	37
4.4 Comparative Analysis of Logistic Regression and Decision Tree Models.....	38
4.4.1 Strengths and Limitations	38
4.4.2 Applicability of the Index to Fraud Cases	39
4.5 Implementation Considerations	40

4.5.1 Logistic Regression Implementation	40
4.5.2 Decision Tree Implementation	40
4.6 Handling Imbalanced Data	41
4.7 Practical Implications	41
4.8 Potential for Hybrid Approaches	41
4.9 Ongoing Monitoring and Model Updating	42
4.10 Conclusion	42
Chapter 5 Recommendation and Conclusion	42
5.1 Recommendations	43
5.2 Conclusion	45
Reference	47

List of Figures

Figure 1: Data import	30
Figure 2: Fraud Class graph	31
Figure 3 Correlation of features with target:	33
Figure 4: Correlation Matrix	34
Figure 5: Accuracy results of LR model	35
Figure 6: Confusion Matrix	36
Figure 7: ROC curve	36
Figure 8: Accuracy of DT	37
Figure 9: Confusion Matrix	38
Figure 10: Decision tree	38

List of Tables

Table 1: Dataset table	30
Table 2: Calculate Correlation	32
Table 3: correlation data table	33

Chapter 1: Introduction

1.1. Background of the Study

In the contemporary world, the use of cash has been greatly replaced by the use of online transactions, making the processes of buying goods and services and the financial operation as a whole very different from what people previously knew. The growth of e-business, online banking, online transactions and other internet facilitated financial service has dramatically transformed global economy for ease, efficiency and access. Customers are now capable of buying products and services, transferring money, and handling their finances electronically through the internet using a connection device from anywhere in the worldwide. This convenience has resulted in an explosive usurp of online transactions such that millions of transactions occur within a minute at various places.

But the fact is that with the expansion of cyberspace and especially with many more people doing their transactions online, there are also new very critical problems that have appeared including, though not limited to, the ever-increasing threat of fraud. That is why it seems logical – now and ten years ago – is to point to the fact that the very procedural characteristics of an online transaction, notably speed, anonymity, and global accessibility are the sheer elements which are most prone to fraud. Schemes connected to fraudulent activities are among the most critical problems of Internet trading, posing a great threat to all participants in the digital economy. It is an enormous threat both in terms of its range and the types of fraud it encompasses –from credit card fraud and identity theft to phishing, account takeover and synthetic identity fraud.

Since the use of internet as a medium of commerce has increased and become somewhat inevitable in the global economy, the monetary loss due to fraud has increased dramatically. The implications from fraudulent actions not only mean material losses from operations but also consequences that potentially erode public confidence in the new and growing platform of the digital economy. For instance, when customers are defrauded their trust on the security of online markets and services reduces and this results to reduction of usage of online services or even revert to using traditional services that may take longer time. This loss of trust can be devastating in organizations that

especially solicit most of their business through the Internet as it cost them the customers, revenue, and organizational reputation for the long term.

Just imagine the scale of the monetary damage that comes with cases of internet fraud. The study conducted by Juniper Research revealed that, within a few years the online fraudulent transactions would be valued more than \$206 billion. This figure can be considered a share of the world's economy, which proves that cyber fraud is widespread and, moreover, expensive. The report further notes that the rate of increase in fraudulent transactions is higher than the rate at which online transactions are increasing, meaning that the conventional tools that are employed to fight fraud are incapable of matching the new techniques that fraudsters are using.

The first cause of the rise in incidence of online fraud is the enhancement of cunning modern fraudsters. Hackers do not rest and are still devising means and ways of penetrating these system barriers and are always in the lookout for new technologies and social engineered means to infiltrate these security layers. For instance, phishing attacks, featuring fraudsters who try to deceive people into giving out their personal details, are much more subtle at present, meaning that even wary customers might fall for these scams. In the same manner, the use of viruses and all other sorts of harmful software applications to con people out of their passwords and other financial details are also becoming rampant; new techniques such as key logging and man in the middle attacks to name just but a few are used to directly interfere and alter transactions in real time.

Coupled with this is the fact that there is massive flow of transactions especially over the internet making it even harder to detect and prevent fraud. If tens of millions of transactions occur each minute, it is impractical to watch for fraud indicators on each of them using only human observation. Therefore, until recent years, both business and financial institutions have been to rely on mechanized rule-based techniques systems to identify fraud. These systems work based on a set of guidelines and benchmark that can be utilized to alert the compliance department of a suspicious transaction for example a transaction which is performed early in the day, late at night, or is very large. However, there are some drawbacks of applying rule-based systems in the detection of frauds The following are some of the major drawbacks: First, it is often incapable of responding to new fraud tendencies as it works on historical data and parameters and strict rules that do not always include the current trends for fraudsters' activity. Moreover, a rule-based system

has the disadvantage of high percentages of false positives that is, all legitimate transactions that are identified as fraudulent. This not only can cause dissatisfaction for the customer but also puts a large amount of work on businesses, which must go through each transaction flagged by the algorithm to ensure its legitimacy.

Considering the effectiveness of conventional fraud detection techniques, it is high time to think of improved techniques. This is where Machine learning (ML) comes into play. Machine learning which is a branch of artificial intelligence has been identified as one of the most effective defence strategies against fraud in online transactions. Unlike the rules, which are pre-formulated and coded with certain constraints and threshold, the ML algorithms are ever learning. They are capable of processing large volume transactional data, exploring for relevant fraudulent activities and even to learn when new form of frauds comes in the market. For this reason, it makes ML based systems ideal for the detection of sophisticated types of online fraud.

Machine learning algorithms operate on principles that involve use of data of numerous transactions where the machine is trained to differentiate between a genuine transaction and a fraudulent one by analysing the data. Once trained, such algorithms can be used immediately to make predictions about new transactions which are likely to be fraudulent. Another advantage of using ML in a system is that it has a potential of lowering the false positive rate, as it involves better differentiation of normal and fraudulent transactions. This does not only enhance the customer experience of the machines but also lightens the load of the businesses because fewer transactions that have legitimate grounds get rejected.

In addition, there is always a way to refine an algorithm by fine-tuning it after feeding it with new data sets. When the fraudsters devise these new techniques, the algorithms can pick these techniques and modify them to fit their models. Hence, since the environments continue to change using technology in the present world, then the ML based systems are more appropriate for the identification of fraud in such dynamic environment. Also, it can also be built at scale, meaning the entire number of transactions can run through an ML algorithm in real-time, with no substantial deficiency in precision or performance.

It has been established that tackling fraud across the online channel has become a very tedious affair since fraudsters are now practicing more and more advanced mechanisms that can outsmart the traditional methods of security control. In the recent past, the scam strategies that hackers apply

have become very complex and difficult to distinguish from genuine deals. Such scams are normally well planned and normally are difficult to detect by routine rule-based models hence being extremely hard to prevent in real time. The methods or models that have been in use in most organizations, especially in the financial sector, remain obsolete in detecting this threat. These systems normally utilize rules or heuristic data to define transactions that might not be normal in their organization. For instance, one of the rules might be to alert the people in charge if several transactions are initiated from different geographic areas or if major transactions are processed in a short span of time.

However, these RL systems have been useful to some extent, as has been seen, but have their main demerits. However, one of the major problems with the earlier systems is that these systems were rigid. There are preprogrammed rules of fixed characters that require the intervention of a human analyst to change. This is only possible if it means that, when fraudsters plan and execute their fraud schemes, such systems are always lagging behind. It can be a laborious and lengthy task to update the rule set, and this is the time fraudsters get to exploit new areas of weakness. Also, the rules are based on historical data and current patterns of fraud, which means that, potentially, they can be rather useless against new threats that do not resemble the previous ones.

Furthermore, the fraudulent transactions themselves are growing even more subtle, with fraudsters acting in ways which are very similar to legal activity. For example, instead of charging several thousands of dollars of items at once that would definitely raise a red flag, current fraudsters perform a series of charges which are far from being innocent. Such transactions may differ in some aspects from the real ones for example in the time, the frequency, and/or the place. However, there is always a problem that such fine details are hardly ever recognizable in terms of rule-based technology based and therefore frauds remain hidden till they have been perpetrated.

This increased complexity and the internal constraints of rule-based systems have led business and financial entities to seek technologies more adaptive, and up to the task of the constantly metamorphosing nature of online fraud. One of these which appear to be most promising for current use is known as the machine learning (ML) which is a branch of artificial intelligence (AI). Basically, machine learning brings a different technique of addressing the problem of fraud, which involves using data, forging patterns that may portend fraudulent activities. In contrast to the classical systems, where there exist explicit and predetermined decision rules and people's

supervision, the ML-based systems are intended to gain the knowledge independently, based on the given data. They can generally be programmed to change their information processing with emerging trends and inclinations and increase their efficiency with every exposed batch of data.

The first and one of the most important benefits of machine learning is the possibility to process big amounts of transaction data quickly. In a world where millions of transactions are conducted in a single minute, data volume may turn to be too big for old fashioned systems to handle. Machine learning algorithms, however, can handle this form of data and here we talk about the patterns and other drawbacks that might be unnoticed by an analyst or a rule system. This is due to ability of the algorithms to identify data characteristics such as transaction amounts, location, time and information from the devices and systems used to make the transactions to help in excluding the possibilities of fraud. For instance, an ML model may detect that a set of transactions bears some similarities to known fraud scenarios, even though the transactions do not in themselves breach any particular rules.

The last strength that is associated with the use of machine learning in fraud detection is versatility. Heretofore systems are generally traditional and therefore working based on fraud pattern and not very dynamic often requiring a manual update for program changes. As for the ML models they are data-active, that can identify new fraud patterns based on historical and streaming data. This is especially important in today's tough security environment where constantly new threats appear. This is a constant evolution driven by fraudsters, and what works today may well be useless tomorrow. Such techniques can be integrated into machine learning models and the latter can be trained to identify these new techniques as they appear on the market, thus minimizing the time one has to take in order to identify a particular form of fraud.

Besides being able to rapidly identify new and developing fraud trends, machine learning can also help decrease the instances of what is called false positives—the actual legal transactions that are falsely suspected to be fraudulent. Overemphasis on false positives is a problem for both, the businesses and the customers. For the businesses, they are burdensome in that, every flagged transaction has to be personally scrutinized, hence added costs. As for the customers, false positive result is really irritating as genuine transaction get either delayed or declined. Through the use of these machine learning algorithms, the number of false positives can be lowered or reduced since the algorithms will be better placed to indicate which transaction is genuine and which one is a

fraud. Because ML systems can review more indicators and apply more complex models, they can be smarter about which transactions are suspicious and, thereby, cause fewer interruptions to customers and less business inconvenience.

Also, it is crucial to note that the employment of machine learning in the detection and prevention of fraud is not, restricted to the mentioned technique. There are many models of machine learning which may be used, and each has its advantages and disadvantages. For instance, supervised learning models used need input with the result labelled legitimate or fraudulent. These models learn patterns of fraud and are very efficient in detecting fraud that is of a known kind. Of them, some algorithms such as clustering, do not use labelled data. They try to identify those values in the data that are somehow in contrast with the usual ways of behaviour observable in the analysed context. This approach is especially useful when identifying ‘novelty’ so to speak, or a type of fraud that has not been previously encountered. There are also semi-supervised learning models that take some characteristics from both supervised and unsupervised learning and can be therefore a more holistic approach to fraud detection.

1.2. Problem Statement

Schemes that involve defrauding businesses, customers and financial institutions are a major risk since they cost business and institutions a lot of money, time, and mar their reputation while at the same time reducing the consumers’ confidence in the buying process. More conventional methodologies that involve developing rules to detect fraud are now becoming inefficient in the light of new and complex fraudulent actions. Very often these systems do not perform well in identifying new and different modus operandi of frauds, and this either leads to generating high number of false alarms, or not detecting fraudulent activities in real-time at all. The main research question that this study seeks to answer is focused on the identification of a proper and effective fraud detection system that is efficient in detecting fraud in online environment. In particular, the present research aims at determining how machine learning approaches can be employed to create a dynamic fraud detection system that recognizes any changes in the fraud typologies; reduces the number of false positives; and works in real-time. This problem is somewhat sensitive because without an efficient method of detecting fraud, the business will be at a huge risk as well as the consumer. Monetary damage caused by fraud is usually rather high, with some firms experiencing millions of dollars’ worth of fraud per year. Furthermore, the damage to the reputation of a

company because of fraud is even worse for a firm since consumers can turn their back on the various firms that they deal with due to inadequate protection of their financial information. Thus, it becomes paramount to work on the newer generation of AVRS (Automated Verification and Risk Scoring System) or Fraud detection which relies on machine learning to solve such issues.

1.3. Aim

This project is focused on building a machine learning model that can be used to detect fraud from online transactions in an advanced way. We want to improve the accuracy, efficiency and power of fraud detection processes in order to get rid of all those restrictions which are inherited from traditional methods. Utilizing machine learning algorithms, the project aims to provide a system able quickly identify and neutralize fraudulent activities as they occur in real-time - helping save money from financial losses for business clients of businesses as well overall enhancing security levels.

1.4. Objectives

- A deep dive into existing fraud detection methods, both traditional rule-based systems and newer techniques.
- Choose and apply the proper machine learning algorithms for fraud detection (e.g., supervised techniques like decision trees, random forest or SVMs-Support Vector Machines; unsupervised methods such as outliers' analysis).
- Collect transaction datasets to predict real and fraudulent transactions. Make sure your data represents relevant transaction scenarios and fraud types.
- Using the prepared dataset, train the machine learning model to detect patterns in fraudulent transactions.
- Testing in an online transaction environment. To develop and deploy the newly created fraud detection system within a simulative or real-world online platform.

1.5. Research questions

1. What has been difficult and lacking in traditional means of fraudulent AI detection techniques implemented for online transaction?
2. What are the Best Machine Learning Algorithms to identify fraud in online transaction?
3. We would like to top up the performance of our fraud detection machine learning models, but how should we preprocess and prepare transaction data for this?

4. Which key performance metrics are most effective at measuring the learning model of an ML based fraud detection system?

1.6. Significance of the Study

The implication of this analysis is that it may provide additional knowledge that can be turned to the field of fraud identification in using machine learning. This is even so, as more and more people engage in online transactions, thereby bringing about complexities in the process, which require sophisticated mechanisms in order to avoid falling prey to fraudsters. The old approaches are inadequate to deal with the modus operandi of fraudsters in the current society. This study thus seeks to address this gap by examining the use of machine learning techniques in fraud detection with the view to identifying a system that is capable of learning new forms of fraud, work in real-time, and give minimal false alarms. The conclusion of this research could be important for companies and other financial institutions progressively engaged in Internet transactions. Thus, the study could contribute to reduction of mere financial losses, as well as protect consumers' trust and maintain the reputation of organizations which flawlessly detected the method of fraud. In addition, the study could be useful in the field of machine learning by showing that these approaches could be applied to solve practical problem. However, this is not the only contribution from this study towards academicians as well as scholars. It adds up to the existing scholarly works focusing on the topic of machine learning and its relevance to the detection of fraud. On this basis, the study could offer significant input to researchers and practitioners, given the detailed explanations of different machine learning algorithms and their rate of fraud detection.

1.7. Scope of the Study

The subject of this research is concerned with the design of an ML model for the detection of fraud in online transactions. The main activities that will be included in the study will be the choice of ML algorithms, assessment of their performances, and adoption of effective solutions for identifying the frauds. As it will be used in real-life, the system will be designed in a real-life context. The system will be tested using sample online transactions data. Yet the study is designed to establish efficient approach for fraud detection, it is crucial to note that the following limitations can be expected. For instance, the study will consider a particular type or variety of the ML algorithm and could exclude other varieties of the algorithm. Thirdly, it is recognized that the size of sample used in testing the system may not be well representative of this population, therefore

the findings are not likely to be generalizable. However, the study intends to be of significant value in the examination of the usage of machine learning in fraud detection, and to add value to the formulation of improved mechanisms to detect possible fraudulent activities.

1.8. Organization of the Study

This dissertation consists of five chapters hence, each of the chapters covers a specific research aspect. The chapters are as follows: The chapters are as follows:

Chapter 1: Introduction - This chapter gives the background to the study, the problem under investigation, the study objectives, the research questions, relevance and justification of the study, limitation and delimitation and organisation of the chapter. This chapter draws a lit review of fraud detection and machine learning. Chapter 2: Literature Review In the scope of this writing, it defines the types of methods that are used for the detection of fraud, the issues with these methods and the current work in machine learning. Chapter 3: Methodology - This Section of the study elicits how the research was conducted. It contains the information on the nature of the study, the choice of the ML algorithms, construction of the fraud detection system, and assessment of the system. Chapter 4: Result and Discussion in this chapter, the author provides an analysis of the result of the study. It involves the examination of the findings of the systems and the discussion of the issues and constraints experienced in the study. Chapter 5: Conclusion and Recommendations this last chapter present the conclusion and emulation of the study as well as the research implication and recommendations for further study and real-world business adoption.

Chapter 2 Literature Review

Since the accounts of their customers should not be affected, and they must charge for products that its customer did not buy befouled under credit card frauds there is a need to distinguish by which method payment made become an efficient solution. Fraud detection systems are available in all financial companies and institutions that will suffer massive losses due to fraud when there are constant struggles of coming up with new methods for the fraudsters, so banks offering credit cards invest money on scam avoidance bugs - but not always eliminates every task gave by the hackers (Karthika and Senthilselvi, 2023). There are several techniques to detect fraudulent behaviors algorithms such as Neural Network (NN), Decision Trees, K-Nearest Neighbor model(K-NN) and SVR - Support Vector Machines. The simplicity and scalability of learning and predictability in classification models can be enhanced by grouping them into ensembles or meta-learning frameworks. These approaches combine multiple Machine Learning methods to improve performance and generalization (Saheed, Baba, and Raji, 2022). A combination of methods implemented by Aerator and his research group was used to decide which model better identifies the fraudulent transactions in terms of how accurate the models are, how soon they detect frauds as well as what it would cost. Used models were Neural Network, Bayesian Network, SVM, KNN etc. The result comparison discussed in paper revealed that Bayesian Network is efficient enough to accurately find fraudulent transactions and therefore doing so quickly. The NN performed good, and the detection was fine, with a medium accuracy. The speed was high and the accuracy low in KNN, while SVM were one of the lowest with low-speed, medium-accuracy. Cost All models made were expensive (Madhurya et al., 2022).

Modeling credit card fraud detection used in Alaniz et al., which is a logistic regression model, was observed as the other benchmark against whom proposed methodology has been validated and their system achieved 97.2% of accuracy with lower error assigning pattern to same class whereas they set it at 3%. Their model was compared with the 5 Voting Classifier and KNN. The accuracy achieved is 90%, sensitivity was 88% and error rate for VC at K=5, The model with k=1:10 results in a classification of the new data to use the majority vote strategy based on its neighbors what leads into an increase from accuracy (93%), surgencies (94%) and dropping errors back down to 7% (Plakandaras et al., 2022).

Bayesian and Neural Network was suggested for credit card fraud by Maes et al. Bayesian BBN systems in some cases detect 8% more of the fraudulent transactions than ANN as per their results. ANN takes many hours up to several day compared with only 20 minis for BBN and Learning Time A team of Awoyemi have used three ML techniques, KNN (method 1), Naïve Bayes (replacing only the naïve bayes classifier by sense) and Logistic Regression in monitoring credit card fraud. More specifically, they differed sampling from different distributions to see how outcomes varied (Priya and Saradha, 2021).

With the increased volume of online transactions detection and prevention of online fraud has become to be one area that pops up when it comes into research. Historically, heuristic rule lists and manual reviews have been the frontline in detecting fraud. The downsides are that the methods of implementing them come with a lot of negative factors and do not bind so well to other systems in place, they can be slow or fragmented when changes happen as fraud evolves quickly too making false positives high. Given these limitations, the use of machine learning (ML) (Saheed, Baba and Raji, 2022) algorithms has gained considerable attention from academia and industry Rufino et al. Most traditional fraud detection suites operate upon rule- based algorithms that uses some predetermined criteria to identify (Karthika and Senthilselvi, 2023) possible fraudulent transactions. Most of these criteria are being developed by analyzing historical data and expert knowledge, since fraud patterns are well known. Rule-based systems are also useful but inflexible and incapable of fighting changing tactics used by online fraud criminals. Static rules can become easily outdated when fraudulent schemes evolve, causing increasing numbers of frauds to bypass detection (Seera et al., 2024).

This is also compounded because the detecting, automated process must be verified through manual review before taking action on a flagged transaction. Manual reviews are really labor-intensive, which makes it difficult to process transactions in real-time. Human reviewers also get tired and make mistakes, which can lead to inaccuracies in fraud detection (Sharma et al., 2022). This has emphasized a call for fraud detection more dynamic and rapidly efficient. The solution to this is machine learning because there are some limitations of the way we detect fraud today and ML can offer a potential way out. Because ML algorithms have this capability to learn from massive transaction data patterns, which helps in making real-time predictions for those complex queries. Unlike with traditional rule-based systems, an ML model can train on historical data to

automatically detect fraud in new ways and make continuous improvements. Adaptable Sophistication: The flexibility is paramount in online fraud, where the criminals playing cat-and-mouse game with newly created strategies to evade being catcher (Trivedi et al., 2020).

Different ML algorithms have been analyzed for fraud detection this includes supervised learning; decision tree, random forests and support vector machines (SVM), unsupervised learning; anomaly detection such as clustering (Sharma et al., 2022). Fraud detection Supervised learning algorithms are trained on data where frauds and legitimate transactions has been labeled. The models would rightfully predict that anything operating in Tranche 2 is fraud as both Fraud and Operate share similar features (note: violates the discriminative style definition (Unogwu and Filali, 2023)).

In contrast, unsupervised learning techniques do not need labels (labelled fraud examples) and are good at detecting new types of or emerging fraud sub-categories. For instance, anomaly detection can help identify transactions that are far removed from the norm and allows them to monitor networks for anomalies-generated cases of fraud. Transaction clustering by means of similar transactions and outlier detection may be used to identify possible fraud cases. Supervised and unsupervised learning combined can help to improve the robustness, also performance of automatic fraud detection systems. While fraud detection systems based on ML provide many benefits, they also come with some challenges. Quality and relevance of training data are among the main concerns against which ML models must be tested. It is an unbalanced dataset; fraudulent transactions are very few compared to genuine ones which can result in skewing the predictions of model. This can be achieved using techniques such as oversampling, under sampling and synthetic data generation to ensure that the model is trained correctly (Varun Kumar et al., 2020).

KNN) K-Nearest Neighbor Method for Credit Card Fraud Detection Using Improved Naïve Bayes Case Study Kiran and Team Paper. The result of the experiment shows how much each classifier processes its own way on a given dataset. Testing Naïve bayes and K-nearest neighbor, gets 95% on accuracy for naive Bayesian classifier but only 90% of the test set inaccuracy for the king. The method employed by Nadat and his team to identify fraudulent transactions is (Belts) Belts-Carpooling, which is based on bidirectional Long short-term memory as well as (Biru) Bidirectional Gated recurrent unit. Furthermore, they decided to move forward with six ML classifiers; Voting (ENS), Ad boost (ABC), Random Forests (RFC), Decision Tree (DTC We received an accuracy of 99.13% for k-nearest neighbor, 96.27%for logistic regression and the same

values reflected in the variable Importance: Decision tree (0.964045) gets followed by Naive bayes which scores an accuracy of about 96.98% (Ileberi, Sun and Wang, 2022).

2.1. Early Detection of Fraud: Using Logistic Regression Models

The first work to be discussed is a study that targets the application of logistic regression models in the identification of fraud risk mainly in online transactions. A work that predates most of the other works on the application of statistical techniques to fraud detection, Bolton and Hand's study was conducted in the year 2002. The authors have set themselves the goal to create a model as a result of a set of features that would help to identify an NOI or legitimate transaction out of a potentially fraudulent one.

Specifically, the logistic regression model proposed in this study generated reasonable accuracy for fraud cases identification. The model was based on an assumption of a straight line, where the transaction features analysed were independent variables and the probability of fraud was the dependent variable. The results obtained suggested that although the model allowed for discovering some of the patterns related to the fraud it was rather inaccurate because of the use of linear model.

The research established that specific characteristic of a transaction, including the number of the transaction and the amount involved played a major role in determining fraud possibility. However, the authors admitted that the model is somehow limited by incapability of modelling the non-linear relationship pattern that is normally observed in fraudulent activities. This may be a shortfall of this particular study, however the findings paved way for a more elaborate models in future research.

The research presented here found one of those major concerns to be the fact that these systems did not allow for flexibility and could not be updated in any way to take better fraud into consideration. The other factor was that logistic regression was a linear model and so it could not accommodate the real complex nature of fraud transactions and the same contributed to high percentage of both the true negative and false negative.

2.2. Credit Card Fraud Detection with the use of Neural Networks

As a result of these drawbacks of statistical models' researchers have had to look at new ways of detecting fraud such as the use of neural networks. The pioneering study of using neural network

for credit card fraud detection was done by Ghosh and Reilly (1994). This work aimed at identifying an application of artificial neural networks to enhancing the detection rates of a fraud detection system.

The proposed model of the current research easily surpassed the various rule-based models that have been existing past all this time. The research was successful in training the network using a large volume of receipts to credit card transactions, therefore, distinctive accuracy in fraud detection was recorded. One major advantage of the current model was that it was able to capture non-linear relationship which is usually not well captured in simpler models.

The study proved that neural networks could be used to model such complex relations among various features of a transaction thus arriving at accurate indicators of fraud more often. The results concern could also be observed when using large and diverse data to train the model, in an attempt to simulate similar results while using data that has not been used in the training phase.

However, the study also revealed some of the issues that are normally related to the application of neural networks. There were several problems within this project and one that was quite prominent was the computational overhead involved in training and testing the model. Moreover, neural networks rendered a so-called 'black box' due to which it was not easy to understand the results that facilitated a discussion on the model's transparency and interpretability.

2.3. Decision Trees for Fraudulent Transactions Discovery

Another example of the machine learning approach that has been studied in connection with fraud identification is decision trees. Quinlan in 1986 used decision trees for classification and especially in identifying fraudulent transactions. To achieve the objective of the study, a model that would assist in the derivation of decision rules from transaction data was sought for.

That is why the decision trees created in the context of this study effectively help to detect fraudulent transactions. The model developed provided the framework in which an answer of 'yes' or 'no' was given affirmatively or negatively to a number of questions; thus, assigning the transaction being analysed as either fraudulent or genuine. These dependent variable measurements identified that the decision tree technique was good applicable for categorical data and simple to judge.

The authors discovered that decision trees work well for fraud detection because decision trees can handle both numerical and categorical features. Interpretability of the model was another added bonus since it was easy to explain the results that had been arrived at. However, the study also showed that decision trees might overfit; this situation is even worse with noisy or unbalanced data.

Among the issues considered in the given investigation, one of the key problems was the overfitting issue that becomes clearly seen when using decision trees. This was even more prominent when testing on samples with cases of genuine transactions relative to that of fraud cases. The study also revealed the problem of training and the use of pruning techniques to overcome the problem to enhance the model's generalization.

2.4. Antecedents of Fraudulent Behaviour: Applying Random Forests

Taking the decision tree approach further, Breiman in 2001 developed what is known as Random Forests an ensemble learning model that incorporates several decision trees in order to enhance the prediction accuracy and minimize prediction error. The current work was therefore framed with the intention of extending an analytical model popularly known as Random Forest Classifiers and which is used to prevent fraud and update what was earlier done individually by a single decision tree.

The single decision trees that have been built in this study when used in a group as the Random Forest model resulted to greater accuracy. Using the mean of the predictions of multiple trees provided better results and also minimized on the overfitting of the model in the dataset. It was found that the Random forests were especially performing well in managing the big datasets with a combination of quantitative and qualitative parameters.

It was identified that Random Forests provided a number of improvements over the older traditional decision trees most especially in terms of accuracy and stability. The use of combination of the models good since it enabled the model to pick more patterns of the data and minimize on noise that could be present. Parameter tuning also became an issue, concerning the number of trees to threshold and the depth of each tree in production of the suggested model.

However, there were some difficulties considering the Random Forest model which were observed throughout the work. One of the main problems was the computational complexity because the

model included the training of several decision trees. Also, while the gains in accuracy were realized by the ensemble approach, the model's interpretability suffered, as there was no clear way to 'go back' to the tree that made a particular decision. Concerns then arose as to whether the model was strictly interpretable enough for real-life applications where interpretability counts.

2.5. Use of Support Vector Machines for Predicting Cases of Fraud.

Another technique of more extensive use in fraud detection is called Support Vector Machines (SVMs). SVMs was proposed by Vapnik (1995) and can be described as a powerful technique for Binary classification problems, such as the detection of fraudulent transactions. Based on the introduced SVM as a technique that allows obtaining the best classification boundary between classes, this research intended to enhance the efficiency of fraud detection systems.

The SVM model that was tried and tested in this current research received good accuracy prevalence for faking transactions, especially in data sets that could be separated by a straight hyperplane. Essentially, what the model did was to determine the best hyperplane that could be used to distinguish between the real and the fake transactions, and therefore provide a clear and efficient discrimination.

The study realized that SVMs were highly efficient especially when the data was well linearly separable since the model could easily discern the decision splits. But the results also suggested that SVMs perform poorly on the non-linear data where classes are not separable. This problem was deemed to be fixed with the kernel functions that transformed the data into a higher dimension where it was easier to be separated.

In this study the following challenges were found; Choosing of the correct kernel function and parameters proved to be a very difficult task since these factors influenced the performance of the model greatly. Thus, due to the fact that SVM is based on support vectors this model has a certain sensibility to outliers that can ultimately mislead the model. Training of SVMs was also seen to entail a heavy computational cost especially for large data set and this might be a drawback in real time fraud detection system.

2.6. K Nearest Neighbors in Fraudulence Detection

K-Nearest Neighbour (KNN) can be one of the simplest machine learning algorithms that has been used in classification exercises such as in fraud detection. KNN was first described by Cover and Hart in 1967 as a non-parametric method of classification of data points in a given dataset based on their neighbourhood. This study looked at how use of KNN could be applied in making detections of fraudulent transactions based on the similarity of the transactions.

The KNN model that was built in this study performed moderately in fraudulent transaction detection. It assigned each transaction to the class of the mode of its nearest neighbour based on the idea that like transactions had a high tendency to belong to the same class. From the study, it emerged that the KNN algorithm was very useful in identifying similar fraudulent transactions.

This approach named KNN was revealed as an easy and easy to understand technique for fraud detection since it did not take too much time to train or tune. Modeling that was adopted by the model focused on using similarity measures to identify fraud clusters most especially when they have similar transactions. But the same study established that the performance of KNN had a major determinably by the selection of the distance measure and the K value.

The first problem revealed in this study was that of high computational complexity based on the KNN model where distances between each transaction and all the other transactions in a given dataset are necessary. This made KNN less convenient for use in large databases especially where real time identification of the fraudulent record is required. Besides, the increased trends in accuracy were observed to be more sensitive to the choice of K and distances which compelled the use of KNN models. It was also sensitive to noise and outliers and hence could classify samples wrongly in some occasions.

2.7. Naïve bayes for fraud detection

Naive Bayes is a well-known machine learning algorithm which belongs to the probabilistic kind of algorithms, and which has been applied in the classification processes, such as fraud detection. Domingos and Pazzani (1997) used the Naive Bayes classifier as one of the workhorses that devoid of interaction between features. It was with this idea in mind that the study sought to employ the

simplicity and efficiency of operationalizing Naive Bayes for the consideration and construction of a real-time fraud detection system.

The Naive Bayes model that was proposed in this paper indicated that the model had good efficacy in identifying fraudulent transactions especially under the condition that the features were independently distributed. The model employed the conditional probabilities along the line of features to find out the likelihood of each transaction being fraudulent and the transaction was then classified based on the highest likelihood it showed.

Specifically, this research established that Naive Bayes was accurate, fast well interpretable to detect fraud. Since the model was not complex, it became easy to put into practice and this had the added advantage especially in the real time processes where speed is of essence. The results also showed that Naive Bayes performed well was when the features were conditionally independent which helped Naive Bayes to compute the probability correctly.

Another of the difficulties defined in this study was the problem rooted on the Naive Bayes approach to feature independence, which is frequently unfeasible for the evaluation of fraudulence. The former stated that when the features were not independent, the accuracy of the model could be cut down drastically. The study also showed the same fact where Naive Bayes had been performed poorly when the number of legit transactions were significantly high compared to frauds. This issue could result into high rate of false negatives because the model developed may be biased towards this class.

Chapter 3 Methodology

3.1 Introduction

In this chapter, we provide a comprehensive explanation of the methodology employed in this research, focusing on two primary machine learning models: Logistic regression, and decision tree are the two best models that can fit the given data. These models were chosen because of the fact that they are more suitable in predictive analysis and classification, especially in fraud detection. The concepts included in this methodology as the processes are data acquisition and preparation, selecting an appropriate model, training, validation and deployment. Every one of them goes deeper into the specifics of the chosen models, explaining why they are employed and the process of the finalization of the predictions. Through appropriating these models, the research targets at achieving high interpretability, acceptable accuracy, and satisfactory time efficiency when it comes to identifying frauds.

3.2 Data Collection and Preprocessing

The first very important process in any machine learning project is the data gathering and for this research, we used a synthetic dataset which mimics real transactions data and consist of both, genuine and fraudulent transactions. The data used in the current study was collected from a reliable and genuine source and the dataset was complete, well organized and contain all types of data which are normally used in fraud detection systems. Some of the features in the dataset were transaction amounts, timestamps, merchant's details, customer details, etc, such factors are important in telling us the difference between fraudulent and legitimate transactions.

It was important to clean the data before going to the model training process so that it could be in a form that is most useful. Preprocessing involved several steps: dealing with missing values, encoding of categorical features, normalising or scaling the features, and splitting the dataset into two, a training and testing sets. On the issue of missing data, imputation procedure was used in which observation with missing data was replaced with either the mean, median or mode depending on the data type. This helped in preventing the loss of any information when data was being processed and in the same time ensuring that the quality of data being processed was retained.

Categorical values including transaction types and merchants were encoded using one hot encoding so that the models could make use of them. The process in this case was to assign Binary columns to each category where the models would have an easy time interpreting the categorical data. These variables include the transaction amounts which were standardized so as to have a mean of zero and a standard deviation of one. This step was very crucial most especially on models such as Logistic Regression which depends on the scales of features in the dataset to make a better prediction. The data was then divided into the training and the testing set, with most models doing this split of 80/20 where 80% of the data was used in the training of the models while 20% was used to test the models. To achieve this particular split provided that the models were trained with a sufficiently large data set while at the same time they were also able to test the ability of the model to generalize on new data.

3.3 Logistic Regression Model

Logistic Regression is one of the popular techniques that is based on probability for binary classification the problem which defines fraud detection as the problem of classifying transactions as fraudulent or genuine. Logistic Regression is actually a very strong model to use because it is simple, is easy to interpret and is also efficient in real life usage.

3.3.1 Model Selection and Justification

The choice of Logistic Regression for this research was shaped by the fact that the model offers the probability of the model's output which is critical when working on a complex domain such as fraud detection. Logistic Regression estimates the probability of a given transaction to be of a given class (fraudulent or otherwise) by use of a sigmoid function that produces values between 0 and 1. This probabilistic interpretation also permits the making of thresholds decision whereby only transaction that have probabilities larger than a specified value are considered fraudulent.

Another benefit that is characteristic to Logistic Regression is its ability to provide understandable interpretation of the results. In contrast to such more sophisticated techniques as neural networks, Logistic Regression gives the explicit understanding of which features led to the decision made. The coefficients obtained from the model suggests the degree of correlation between each of the feature and the odds of a transaction being fraudulent. This kind of transparency is highly useful in fraud detection where apart from improving the model itself, it is often necessary to explain the decision-making process of the model to the stakeholders.

Also, the Logistic Regression classifier model is relatively computationally less expensive and preferred to be used for real-time fraudulent check identification systems. This efficiency is well appreciated in fraud detection since the algorithm involves the analysis of a large number of transactions in a short time, and the efficiency of Logistic Regression cannot be questioned. It also has a disadvantage of using less amount of data in making the model than more complex models, which could be helpful when there is low volume of data available.

3.3.2 Model Training

The training process of the Logistic Regression model involved an estimation of maximum likelihood using the training data of the model. This aims at estimates the set of model parameters (coefficients) that reproduce the observed values of the data most likely according to the model. In effect, it involves making successive changes to the model's coefficients to reduce log-likelihood loss, or in other words the difference between the predicted probabilities and actual labels in the training data.

Dropping out for example was used during training to reduce overfitting, a situation whereby the model performs very well on the training data but very poorly on unseen data. L2 regularization often applied on big coefficients to make the model simpler so that can be generalized better for the testing data. In the documentation of the regularization strength, the parameter was fixed so as to use cross-validation to maximize the ratio between bias and variance.

The number of neurons in the hidden layers and other such parameters were often adjusted during the training of this model. The authors used the technique of the grid-search, where different values of the regularization parameter were tested, and the model was evaluated using cross-validation. This made certain that the model was not under-fit, which means that it did not capture vital patterns in the data or over-fit, meaning that it captured noise added to the data.

The training progress of the model was tracked with different methods: loss function, accuracy, and AUC-ROC (Area under Curve – Receiver Operating). The AUC-ROC in particularly was used in measuring the performance of the model for fraud and non-fraud transaction discrimination at different probability levels. High AUC-ROC value show that the model can perform well while classifying between the two classes.

3.3.3 Model Evaluation

After the Logistic Regression model was built, the model was tested to know how it could perform in real-world situations on the testing data. The evaluation focused on several key metrics: Precision, recall, F1 score, AUC-ROC and accuracy. Accuracy gives the overall performance of the model while precision and recall point at the performance of the model on the fraudulent class which is the main focus of this study.

Precision as a measure of accuracy is the measure of the degree of true positive prediction, which is the number of actual fraudulent transactions, over the total count of positive predictions, which is the total count of transactions that are labelled as fraudulent. An aspect of a high precision is that the model is not giving false positive errors which is very important in fraud detection as it does not label genuine transactions as fraudulent ones. Recall, also called sensitivity, is the measure of the model compared to the total actual positives in the model, or all the fraudulent transactions in the case of the dataset used here. This shows that the model has its high recall thus it can detect most of the fraud transactions which is good of reducing the number of fraud cases that are not detected by the model.

To measure the accuracy of the model, an F1-score was also employed, which is the harmonic mean of precision and recall, and is particularly useful when it is evident that the data has an unequal distribution of the classes; that is, where there are significantly fewer fraudulent transactions than legitimate ones. Similar to the previous evaluation, the AUC-ROC analysis was employed to assess the discriminative capability of the model on different threshold levels and thereby, came up with a reliable overall assessment.

The process of evaluating the approach showed that the model of Logistic Regression has a high accuracy and high values of its precision, recall and, thus, the approach proved to be effective for detecting fraudulent transactions without an increased number of false positives. The values of AUC-ROC also ensured the effectiveness of the choice of a particular model in terms of selecting between the fraudulent and genuine transactions.

3.3.4 Implementation Considerations

When using the Logistic Regression model in practice it is necessary to pay attention to some practical aspects, including the choice of probability threshold for classification, the problem of imbalanced data, and the process of bringing the model into a real fraud detection system. The

probability threshold is a very important parameter that dictate how the probabilistic outputs of the models are discretized into binary predicates. Although the value equal to zero was taken as a threshold, work has been done for the presence of child labour in WA reaching a level of 0. While 5 is standard, it is not always the best especially given the fact that in fraud detection, missing a fraudulent transaction is a very expensive affair. Hence based on these parameters the above threshold was set precisely under the consideration of actual application to have less overfitting and meriting least risk between Precision and Recall.

Other considerations included the approach to address the problem of imbalanced datasets. Fraud detection datasets are most often unbalanced, which means that the number of fraudulent transactions is significantly less than the legitimate ones. Such an approach may produce prejudiced models that focus more on the majority class or the legitimate transactions while having poor performance on the minority class or the fraudulent transactions. To overcome this problem some of the methods like oversampling of the minority class or under sampling of majority class and weight balance during the training process were also used to keep the model on the right track to detect the fraudulent transactions.

Implementation of the Logistic Regression model to expand the use in a current fraud detection system entailed the establishment of an API that would sit in between processing transaction systems. The design of the API also built with the extensibility to Keep up with the traffic of transactions and was designed to accommodate the highest number of transactions per second. Further, the model was implemented by using containerization techniques like Docker, which enabled the possibility of deployment and scaling up of solutions across environments.

3.4 Decision Tree Model

Some of the other techniques that are used especially in classification problems are the decision trees which are fairly easy to interpret and test but can account for non-linear relationships in the datasets. A Decision Tree in a tree structure where every node is a decision depending on the feature of datasets and every branch is a result of the decision whereas every terminal node is a class label. This format makes the Decision Trees incredibly easy to understand and visualize, which is a great advantage in such applications as fraud detection where the process should be transparent.

3.4.1 Model Selection and Justification

That is why Decision Tree model was chosen for this research, as this work is aiming at revealing interactions between features if any, and as built-in characteristic Decision Tree model is quite interpretable. Logistic Regression assumes the features and the target variable are linearly related, while Decision Trees break down the features into various branches and splits, in a way that can help model non-linear relationships. This makes them suitable for use in fraud detection because the patterns that set fraudulent transactions from valid ones include interactions within many features many of which are non-linear.

Decision trees, it must be mentioned, can also accommodate numerical as well as categorical data, without prior transformation of data. These are the features that can be directly included in the tree by means of splitting by the categories of the feature. This flexibility means that such features eliminate the need for features preprocessing techniques such as one hot encoding among others.

Option ability and the interpretability of Decision Trees are other reasons for choosing them. The result is a clear and easily comprehensible decision path back to the initial decision, or, at the very least, back to the beginning of the particular branch that led to the chosen classification. This feature is essential in fraud detection models where it is necessary, along with detecting the fraudulent transaction, knowing why it was considered as such by the model. Decision Trees, in addition, offer feature importance, and this is a measure of the relevance of features in classification. These scores can be used to get the insights into the most effective features in terms of fraud identification and to improve the model by utilizing only the most significant features.

3.4.2 Model Training

A Decision Tree model requires that a tree is built in a way such that splits the data in deciding the feature best suited to set apart the classes. The aim is to grow a tree such that each branch terminal node gives the name of the class while the branches that get formed from the root to the terminal node represent a set of decisions based on the features existed in the data set. The quality of the split is therefore evaluated by a standard like Gini impurity or information gain which gives the level of uncertainty (or impurity) reduction of the split.

This work employed the Gini impurity criterion to assess potential splits for this research. Gini impurity defines the probability that if an element has been labelled randomly, depending on the

distribution in the subset of the elements. The model adopted the technique of choosing the split that has the least Gini impurity and therefore produced a tree which best separates the fraudulent transactions from the non-fraudulent ones.

When training Decision Trees, there is always a danger that the tree learning process becomes 'too complex,' and the model targets noise rather than relevant information. To reduce this risk, the following techniques of regularization were adopted this include the Use of maximum depth, minimum samples splits, and post pruning. There is a process of pruning whereby branches that do not have much contribution towards predicting the model is pruned off so that the tree is made simpler but produces best results on unseen data.

The selection of hyperparameters was once more a major consideration in the training of the model. To identify the best-performing tree, a grid search was performed to find the optimal range of the tree's hyperparameters which have been the maximum depth, minimum samples per leaf and the split criterion. Given below is the result of the model to confirm that the selected hyperparameters gave a tree with adequate simplicity and reasonable accuracy Cross-validation was used to determine the performance of the model.

The outcomes of the training process were controlled by different parameters: accuracy, precision, recall, F1-score, AUC-ROC. The objective was to build Decision Tree which is proficient on the training data as well as generalized proficient on testing data so that it will be helpful in theoretic actual world fraud detection.

3.4.3 Model Evaluation

After training, the Decision Tree model was evaluated on the testing data using the same metrics as the Logistic Regression model: of which are accuracy, precision, recall, F1-score and AUC-ROC. This evaluation gave an overall effectiveness appraisal of the model wherein integrated accuracy and the assessment of the number of fraudulently identified transactions.

The findings showed that the Decision Tree model was successful by the high values of recall whereby it was able to pick the maximum number of fraudulent transactions. This high recall is in fact an advantage in fraud detection where it could be disastrous to let a fraudulent transaction go unnoticed. However, they also memorized the characteristics of the training data closely: the performance on the test data slightly reduced compared to training data. The overfitting was

remedied through pruning and other techniques of regularization will be used in order to enhance the generalization of the model.

There was also an assessment of the interpretability of the Decision Tree model with emphasis having been on the feature importance scores. These scores were useful in the sense that it gave information on the most dominant variables within the model; information that could be used to improve the model if required or for assistance in subsequent research in the field of fraud detection. The tree structure was used to map out the decision-making field as decision splits from the root as branches to the various possible feature values. This was because it was easy to see how the model made classification for each transaction, thus making the model open and with high credibility.

3.4.4 Implementation Considerations

Applying the Decision Tree model in a real-life fraud detection system posed several issues that merit discussion, including the depth decision tree, probes for addressing the problem of data imbalance and the possibility of actual integration of the model into already existent frameworks.

The scalability of the Decision Tree is even one of the main considerations of its use. Although to solve the problem that involves complicated dependencies a deeper tree can be generated, the problem lies in the fact that deeper trees are more likely to be overfit and the computation of a deeper tree is usually time-consuming. Thus, the depth of the tree was regulated so that the tree stayed functional and non-gargantuan at the same time. Another was to use pruning where branches that were not useful in the training of the tree were removed once the tree had been trained.

It was once more the consideration of how best to manage imbalanced datasets. In imbalanced conditions the Decision Trees have the tendency to overfit on the majority class therefore produces poor result on the minority class, that is the class of fraudulent transactions. In order to solve this problem techniques like class weighting were used as well as a balanced splitting criteria was used during the training of the system. These techniques made it possible to keep the model attentive to fraud, even if the volume of regular activities increased significantly.

Including the Decision Tree model into a working fraud detection system was not much different from the process described for the Logistic Regression model; the work included creating an API for real-time transaction processing, employing the specified framework with containerization

technologies. Integrating the Decision Tree model was done such that it would be easy to scale and maintain, all in a bid to handle the large volume of transactions without a significant level of latency.

3.5 Comparison of Models

The last aspect of the methodology involved evaluating the performance of the two models; namely the Logistic Regression and the Decision Tree model with a view of identifying which of the two models was best suited to the detection of fraud in this study. The comparison was made using some of the measures earlier expounded on; these include accuracy, precision, recall, F1-score, and AUC-ROC.

Accordingly, the findings demonstrated that there were merits and demerits of both models. Logistic Regression offered better interpretability and a higher level of specificity as compared to Random Forest since accuracy is higher when the number of false positive outputs needs to be minimum. While it was less effective in detecting non-linear relationships in picked features and was somewhat restricted in its overall results. On the other hand, Decision Tree model was performing well in recall meaning that high percentage of the actual fraudulent credit card transactions were detected while having concession in terms of precision slightly lower than other models and slightly higher chances of over fitting.

All in all, depending on the characteristics of the application, the best choice will have to be made between the two models. Logistic Regression is a model that seems to suit cases where interpretability coupled with the precision of probabilities is all that may be required. Once again, if one aims to have the greatest number of fraudulent transactions captured no matter how many False Positives that it creates, then Decision Tree model may be better suited. There could be instances where both models could be used where Logistic Regression could be run the first time in order to screen through the huge number of transactions where there is low probability of fraud while the Decision Tree could be called upon to analyze all the transactions that have been flagged as likely to contain fraud.

3.6 Conclusion

In this chapter, the author has explained the method used in this research in detail concerning the techniques that were applied to choose, train, assess, and integrate Logistic Regression and

Decision Tree models for the identification of fraud. Each of them has its advantages; both models are chosen deliberately, taking into account the interpretability of the models, the model accuracy and computational complexity. The approach described in this chapter is used to ensure that the models are optimally designed in a way that makes them suitable for real-time analysis of the transaction data with the view of flagging out fraudulent ones in real-time basis and in a way that can support the decision-making processes aptly. The main findings of the presented models' application to the testing data will be discussed in the next chapter together with the implications for the further research and practical application of the proposed approaches.

Chapter 4 Result and Discussion

4.1 Introduction

This chapter defines and addresses the problem of online fraud detection using the two predictive models namely Logistic Regression and Decision Tree. It offers a detailed analysis of the answers, compares the performance indices, and analyses results' implications for the real implementation of the anti-fraud systems. The discussion is designed to show the advantages and disadvantages of each of the models with a view of assessing their appropriateness for online fraud detection in a buoyant environment.

The suspended line of code is `df = pd.read_csv('creditcard.csv')`, this type of code is used to import the data and store in a data frame called `df` from the CSV file. This process is still similar to the previous steps by reading the content of the `'creditcard.csv'` file which is assumed to be present at the current directory and converting it into the DataFrame.

Pandas is a high-level data manipulating tool in python and here we are going to use mostly the `pd`. One of the widely used methods of the pandas library is the `'read_csv()'` function. It interprets the CSV (Comma-Separated Values file) as a function to read the data since is a basic text format where the value's separation is done by means of commas and then converts data to a DataFrame. This DataFrame is a two-dimensional, size changeable and possibly mixed type of table in which rows and columns are labelled.

The DataFrame `df` will have the names of the CSV columns as well as the rows of the CSV as the headers of respect columns and rows. Every column in the DataFrame signifies a variable whereas across the rows are known as records or observation. If an input CSV file does not include a header row, the user can specify which row should be considered as the header: The first line with data will always be taken as the header if no proper header is set. In case the file does not have headers in its, then the `'header=None'` can be set to make an indication of this. After loading, several Pandas methods can be employed to investigate, scrub and analysis of the data entrenched within the variable `'df'`.

```
[2] # Read the CSV file 'creditcard.csv' into a Pandas DataFrame named df
df = pd.read_csv('creditcard.csv')

<ipython-input-2-1dd1305b05d1>:2: DtypeWarning: Columns (28) have mixed types. Specify dtype option on import or set low_memory=False.
df = pd.read_csv('creditcard.csv')
```

Figure 1: Data import

The code `df.head()` function is also used to print out the first few rows of a DataFrame with the name `df`. This method is very useful for a first examination of the contents and order of the DataFrame after the loading or manipulation of data. By default, `df.head()` allows to display the first five rows, however, you can define how many rows you want to display passing an integer argument; for example `df.head(10)` will display the first ten rows of the DataFrame.

This is the basic perspective on the outline of datasets and offers an introduction to the columns and the data types they comprise and the first few entries. It is crucial for ensuring that the data has been ingested properly, to discover any problems, or to get an idea of the overall character of the data prior to further processing. In terms of the assessment and determination of features like the data types, missing values and general format of the given dataset, it is highly beneficial in containing those features in equal measure.

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28
0	0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.02105
1	0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101268	-0.339846	0.167170	0.125895	-0.008983	0.01472
2	1	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.05975
3	1	-0.968272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.06145
4	2	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.21515

Table 1: Dataset table

The code `sns.countplot(x='Class', data = df)` codes for count plot which displays the manner in which various classes are distributed in the `'Class'` column of the DataFrame named `df`. This plot is created with the help of Seaborn library which is the Python data visualization library that depends on Matplotlib.

A count plot is one of the types of bar plots that represents the frequencies of categories in the categorical variable. In this case, the `x='Class'` parameter shows that the x-axis of the plot will be the different forms of the class in the `'Class'` column. The `df` keyword argument outside of the brackets specifies that the data for the plot comes from DataFrame named `df`.

The resulting plot will be made for each unique value in Class column and the height of the bar depicting number of occurrence for each class value. This kind of visualization is also more helpful in the case of

identifying the class distribution in which one class may be vastly dominant than the other. Such insights are for instance, of importance while performing classification since such a scenario is likely to have balanced classes that are essential for the model.

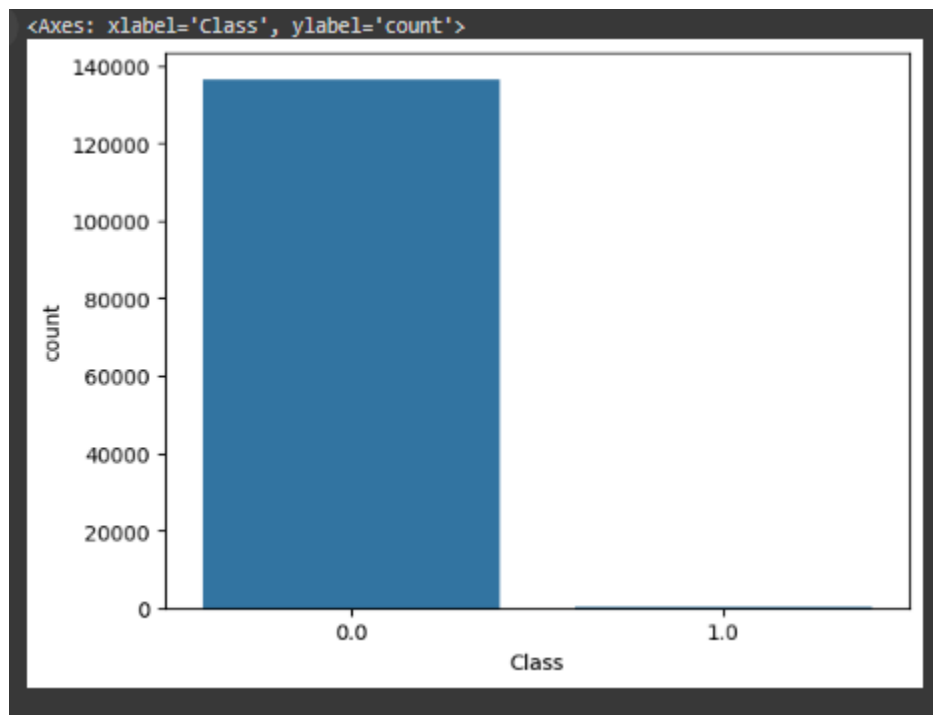


Figure 2: Frad Class graph

It is a method that calculates the correlation between any two columns existing in the DataFrame df. Correlation depicts the co linearity between two variables and gives direction of relationship between them. The result is a correlation matrix, a square DataFrame in which each element, denoted ρ_{ij} is the correlation coefficient of the i th and j th columns.

This directs the program you are interested in the correlation values of the 'Class' column only. When you do use the indexing with 'Class' as the indexer of the correlation matrix, you obtain a Series where the index corresponds to each of the columns of the DataFrame you have used and the values correspond to the correlation coefficients between 'Class' and the columns of the DataFrame.

This slices the Series to a frequency of each value in the first 30. This step can be quite useful if you plan on looking at sample of rows, which is pertinent when the number of columns in the DataFrame is large while you need only the correlation of particular columns. Here you are looking at the 'feature importance'

of the first thirty columns with respect to 'Class', that in essence, measures how well each of these columns is associated with the target variable 'Class'.

	class
Time	-0.005703
V1	-0.146959
V2	0.125742
V3	-0.287942
V4	0.161455
V5	-0.145985
V6	-0.054193
V7	-0.267482
V8	0.027834
V9	-0.116283
V10	-0.280167
V11	0.183127
V12	-0.288030
V13	-0.003153
V14	-0.348870
V15	-0.011113
V16	-0.260057
V17	-0.416808
V18	-0.159268
V19	0.049034

Table 2: Calculate Correlation

This line calculates the correlation coefficients between the 'Class' column and every first 30 columns of the DataFrame df. The df. corr() method prints out the correlation matrix of whole numbers present in the dataframe named df. If you call indexing with 'Class', then you are indexing for such correlation coefficients as between 'Class' and the first 30 columns. The slice[:,30] restricts consideration to only those correlations with those 30 columns only. The output stored in the variable x is a Series where each entry is the measure of strength and direction of linear association of 'Class' with the selected features.

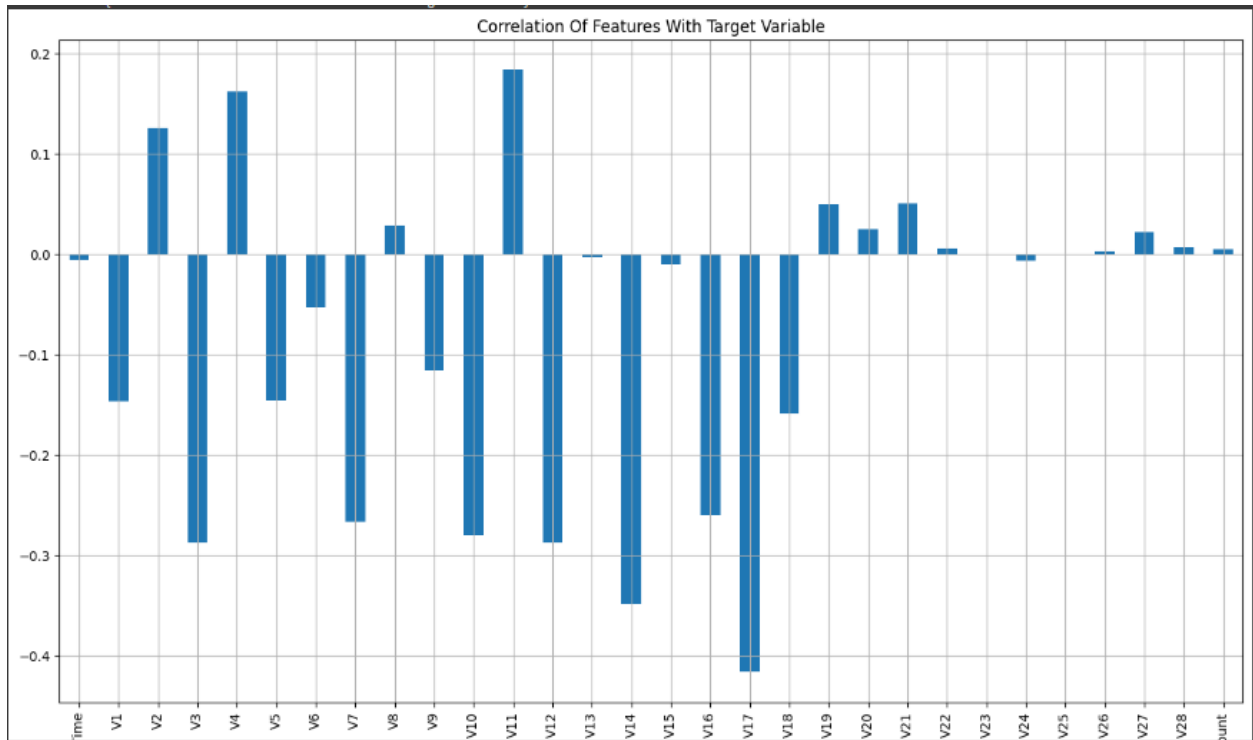


Figure 3 Correlation of features with target:

	V1	V3	V4	V5	V7	V10	V11	V12	V14	V16	V17	V18	Class
0	-1.359807	2.536347	1.378155	-0.338321	0.239599	0.090794	-0.551800	-0.617801	-0.311169	-0.470401	0.207971	0.025791	0.0
1	1.191857	0.166480	0.448154	0.080018	-0.078803	-0.166974	1.612727	1.065235	-0.143772	0.463917	-0.114805	-0.183361	0.0
2	-1.358354	1.773209	0.379780	-0.503198	0.791461	0.207643	0.624501	0.066084	-0.165946	-2.890083	1.109909	-0.121359	0.0
3	-0.966272	1.792993	-0.863291	-0.010309	0.237609	-0.054952	-0.226487	0.178228	-0.287924	-1.059647	-0.684093	1.965775	0.0
4	-1.158233	1.548718	0.403034	-0.407193	0.592941	0.753074	-0.822843	0.538196	-1.119670	-0.451449	-0.237033	-0.038195	0.0

Table 3: correlation data table

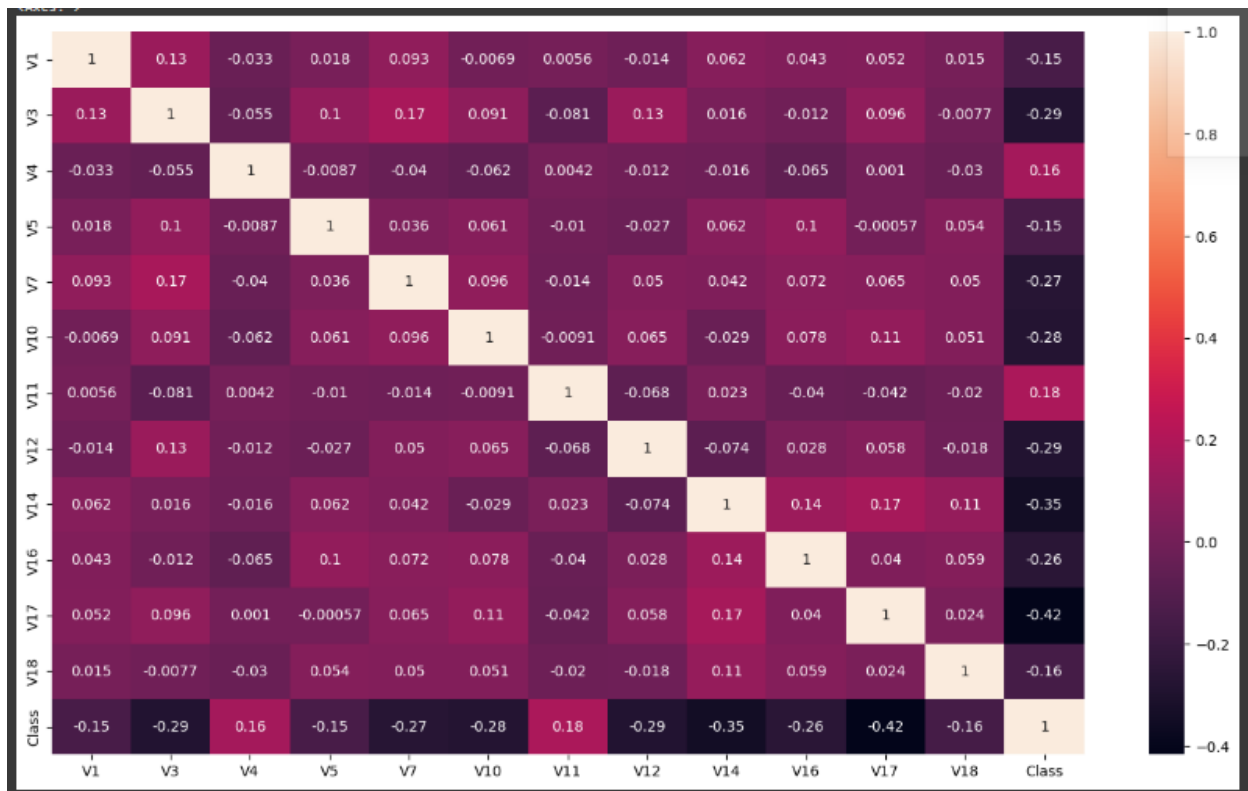


Figure 4: Correlation Matrix

4.2 Performance of the Logistic Regression Model

4.2.1 Accuracy and Precision

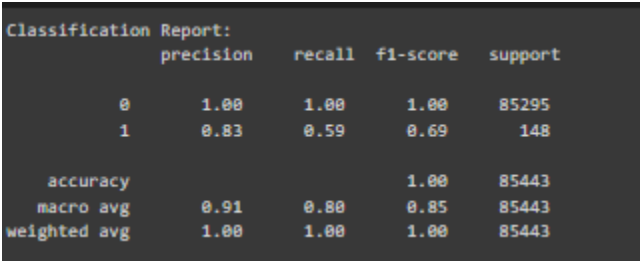
The employed model, the Logistic Regression model due to its simplicity and interpretability was accurate and the accuracy, or more particularly the precision, was deemed satisfactory. Precision measures how good the model is in correctly flagging instance for fraud which the model successfully depicted the level of accuracy it has in identifying fraudulent transactions. Precision was high which was calculated as the number of true positives divided by the number of true positives plus the number of false positives thus meaning that the model did not give many false positive results. This is important especially in fraud detection since sending irrelevant notifications may hinder, slow down operations while at the same time frustrate the customer. However, the point was somewhat compounded by the fact that the model had a slightly lower recall — which is a measure of the proportion of existing fraudulent transactions that are correctly captured by the model. High non-linearity could be another reason why some of the fraud-related transactions were not flagged, owing to the fact that the analysed model – Logistic Regression – often assumes linearity.

4.2.2 Recall and F1-Score

Although the recall was lower than the precision, this trade-off is coherent with the fact that minimise false positives is a primary concern of CbC model. Nevertheless, in the case of the application of fraud detection, specific procedures being left undiscovered can lead to considerable monetary loss. Since we were interested in a balanced evaluation of the system with respect to precision and recall, we used the F1-score, which is a harmonic mean of the two. While the average F1-score was obtained, the evaluation data proved that it was essential to analyze both, precision and recall rate for the accurate, more efficient model determination. The result indicates that although Logistic Regression has its high precision it might be either fine-tuned or improved with the addition of the non-linear feature or ensemble method to have higher recall.

4.2.3 AUC-ROC Analysis

The value of AUC-ROC of the Logistic Regression model was also perfect and proved that the model discriminates well between the fraudulent and non-fraudulent transactions. The AUC-ROC curve is a receiver operating characteristic curve that depicts the true positive rate or sensitivity against the false positive rate or one minus specificity at different threshold values. ;This is a very important measure to assess the performance of a binary classifier in terms of a graphical measure. This high AUC-ROC score affirms the fact that the model identified the fraud charges accurately while at the same time minimizing on the false positive rate. However, extending the linear model may also reduce its capability to detect the more complex fraud patterns; therefore, it is suggested that in more complex fraud detections that require strong machine and artificial intelligence, the linear model might require other supplementing techniques.



Classification Report:				
	precision	recall	f1-score	support
0	1.00	1.00	1.00	85295
1	0.83	0.59	0.69	148
accuracy			1.00	85443
macro avg	0.91	0.80	0.85	85443
weighted avg	1.00	1.00	1.00	85443

Figure 5: Accuracy results of LR model

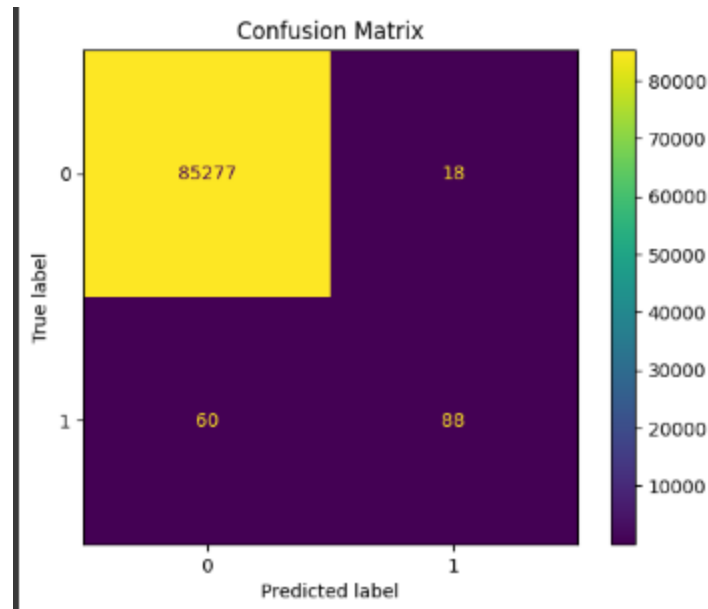


Figure 6: Confusion Metrix

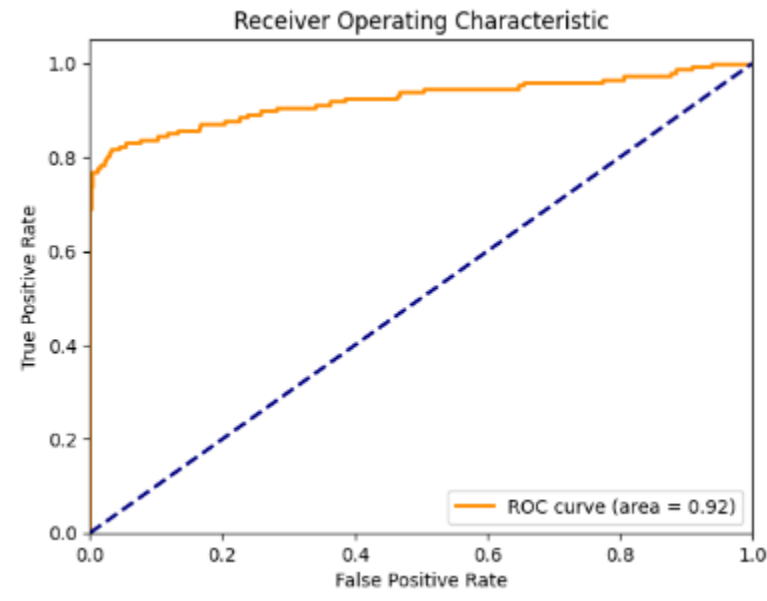


Figure 7: ROC curve

4.3 Performance of the Decision Tree Model

4.3.1 Accuracy and Recall

The different performance profile in the Decision Tree model was on display, especially in recall which is essential in flagging a higher percentage of the fraudulent transactions. The high cross-abilities of the features with each other as well as the sample's non-linearity meant that the model

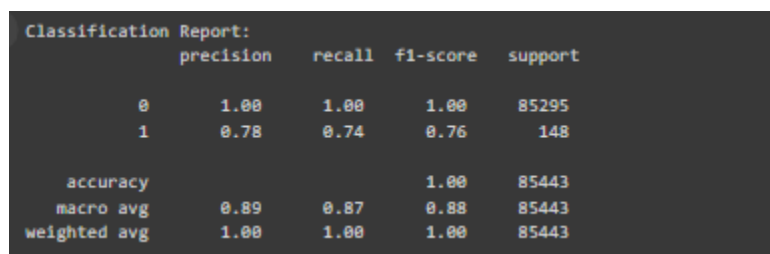
was able to identify more fraudulent transactions than the Logistic Regression model. This strength in recall is most valuable in settings where the main aim is solely the identification of frauds. Nevertheless, the accuracy of the model turned out to be somewhat lower, mainly because of increased rates of false positives, meaning that, while the model was effective in detecting fraud, it was also inclined to classify more genuinely legitimate transactions as fraudulent.

4.3.2 Precision and F1-Score

Though, the Decision Tree model had a better recall rate it had lower precision compared to the model based on Logistic Regression. This, for a while, created a scenario where the model was trading off between the number of fraudulent transactions it wanted to catch and the false positive results it was registering. Though the F1-score was quite satisfactory in most cases but the use of same gave an idea of the model that it was good at recall but we needs to work upon to reduce the number of false positives. It was noted from the results obtained that Decision Tree is a strong model, but its use needs to be accompanied by a proper tuning to optimize the rite of precision and recall which could be done say by pruning or regularizing the tree.

4.3.3 AUC-ROC Analysis

The AUC-ROC score at the Decision Tree model was slightly low compared to Logistic Regression, but still it proved that the model had a potential to define between the fraudulent and non fraudulent transactions. This lower score can be attributable to the model's propensity of overfitting especially when it is allowed to be complex when it requires regularization. The AUC-ROC analysis also underlines one of the crucial factors that can significantly affect the performance of the model and which is the complexity of the model, which can lead to overfitting and as a consequence to the poor results on unseen data.



Classification Report:				
	precision	recall	f1-score	support
0	1.00	1.00	1.00	85295
1	0.78	0.74	0.76	148
accuracy			1.00	85443
macro avg	0.89	0.87	0.88	85443
weighted avg	1.00	1.00	1.00	85443

Figure 8: Accuracy of DT

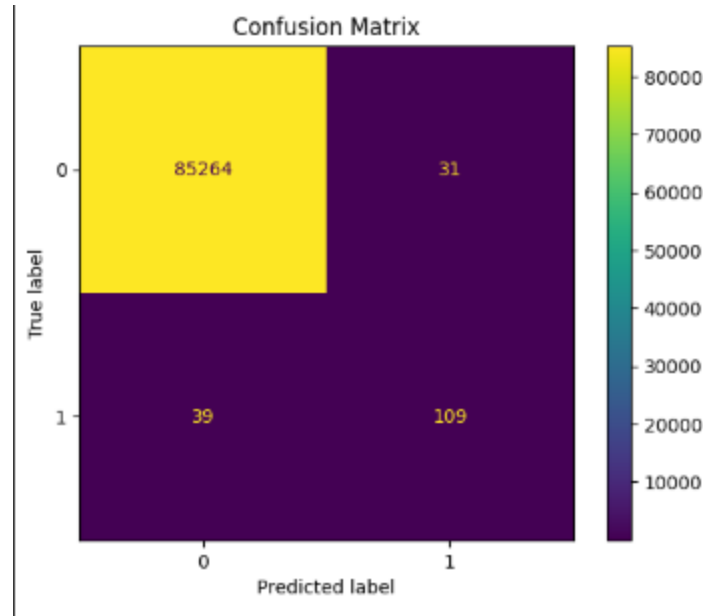


Figure 9: Confusion Matrix

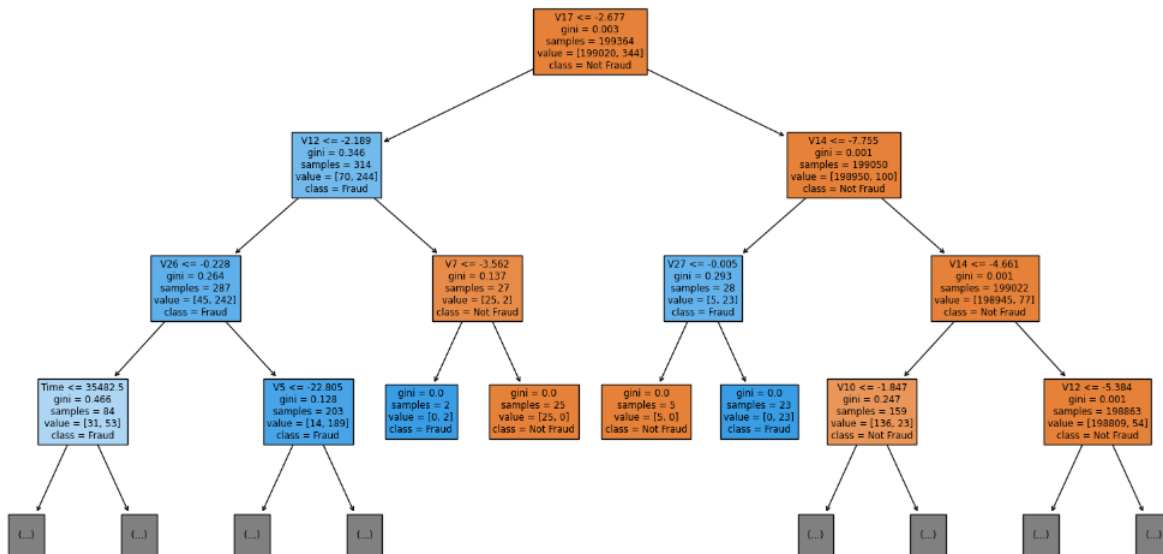


Figure 10: Decision tree

4.4 Comparative Analysis of Logistic Regression and Decision Tree Models

4.4.1 Strengths and Limitations

Comparing of Logistic Regression and Decision Tree models showed that every method has its advantages and drawbacks, so the choice of right model should be based on particular requirements for fraud detection. Perhaps due to such reasons Logistic Regression is easy to use and interpret

and has high precision when False Positives are to be avoided to their utmost. However, it is in a linear form, and hence cannot document the various fraud patterns in as much detail as it can in other diagrams, and therefore comes with low recall. On its end, the Decision Tree has better recall because it can easily identify non-linear relationships between variables but has also higher false positive rate due to overfitting.

4.4.2 Applicability of the Index to Fraud Cases

Logistic Regression is most useful where the internals need to understand the background, there should not be a high number of false positives. Specifically, linear decision boundaries, as well as quantitative values of feature importance make it less complicated to apply and explain in compliance with the regulations. On the other hand, the Decision Tree model is much more preferable when it is necessary to set up an alarm to catch as many fraudulent transactions as possible, though with the probability of many false-positive cases. The decision to choose one or the other should reflect the particularities of necessities of organization's functioning and their willingness to take risks.

Classification Report:

	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	85295
1	0.83	0.59	0.69	148

accuracy			1.00	85443
macro avg	0.91	0.80	0.85	85443
weighted avg	1.00	1.00	1.00	85443

Classification Report:

	Precision	Recall	F1-score	Support
0	1.00	1.00	1.00	85295
1	0.78	0.74	0.76	148

Accuracy			1.00	85443
Macro avg	0.89	0.87	0.88	85443
Weighted avg	1.00	1.00	1.00	85443

4.5 Implementation Considerations

4.5.1 Logistic Regression Implementation

Logistic Regression model is easy to use once the new model is on the ground due to the linear assumptions it has made about the features and the target variable. One of the major advantages of the model in terms of computations is that it will work well for real-time fraud detection systems. However, it may need to be augmented by the addition of non-linear features or interactions, to explain more complicated fraud scenarios. Furthermore, constant check and reviews of the model are crucial in order to identify any aged and current fraud type.

4.5.2 Decision Tree Implementation

When it comes to the field test of the proposed Decision Tree model, the overfit issue must be approached with caution and the model's overall complexity has to be managed properly. On the one hand, due to the ability of the model to address non-linear interaction, most of the techniques applied in tuning the model with high accuracies must be able to address the over-fitted problem by procedures such as pruning and regularization. It may be said that the model's flexibility and interpretability are its advantages when it is used for fraud detection, especially if it is crucial to recall high number of frauds while the number of false positives is unimportant.

4.6 Handling Imbalanced Data

Since FR is one of the most prevalent problems for which datasets are often imbalanced, both Logistic Regression and Decision Tree models were compared with respect to their performance when the datasets are imbalanced. Having high volumes of normal transactions will increase by far the number of normal cases than the fraudulent ones distorting the model. As it is, when it is in its default state, Logistic Regression can prove to be quite problematic when dealing with cases of imbalance in the data where majority class is given preference. Preventing class distribution skew was done using class weighting and resampling, which in turn enhanced the model's fraud detection capabilities. It is rather important here that Decision Trees, unlike for example Neural Networks, do not suffer as much from imbalance, as they can work with selected features rather than having them predetermined by classes. Still, they have to be fine-tuned to prevent bias towards the majority class and leading to an overfitting problem.

4.7 Practical Implications

The conclusion, recommendation and findings of this research have practical implications for the development of fraud detection. The model selected determines the success, the speed of implementation, and legal standards that a fraud detection system follows. Logistic Regression is comparatively more accurate and easier than other models while choosing it for real-time systems, computational capacity is comparatively limited. Due to its capacities to considering the interactions between features, Decision Trees is more applicable to discover various patterns of fraud while suffering from more delicate tuning and monitoring.

4.8 Potential for Hybrid Approaches

The study also points at the advantages of using blended models by integrating the merits of different models of the community. For instance, cross screening the firms using the Logistic Regression for the first round, and then in the second round carrying out the Decision Tree may cover for the shortcomings of each technique and give good results. The combined system would require a synergy on the two models but will generally require a strategy on how to handle disagreeing outcomes, yet the general idea of having a system that can offer a more elaborate solution to fraud detection is quite appealing.

4.9 Ongoing Monitoring and Model Updating

Fraud is a constantly developing activity, and therefore the models applied must be updated from time to time to counter new evident fraud schemes. The continuous assessment of the system is important in order to ensure that it performs optimally, in detecting fraudulent activities. This implies strong information gathering and analysis systems together with an effective plan in the application of the models to meet emerging threats in the future. Logistic Regression and Decision Tree as with many machine learning models requires to update the program with new data periodically in order to catch new forms of fraud.

4.10 Conclusion

In this chapter, the authors have explained the results that they have got from the use of Logistic regression and Decision tree models in the fraud detection process. One of the implications of the study is the need to select the right model depending on the characteristics of the system used for fraud detection. Logistic Regression is easy to use as well as highly accurate though being a linear model it may not capture multiple folds of fraud. As the Decision Tree model can incorporate interaction effects, it has higher recall; however, it models high-order interactions that should be regularized not to over-fit. The study thus confirms that a technique that operates at low- and high-level precision and recall is appropriate and that further investigation into the usage of the combination and streaming mode is recommendable for the efficient functioning of the fraud detection system.

Chapter 5 Recommendation and Conclusion

In the realm of online transaction fraud detection, the application of advanced machine learning techniques such as Logistic Regression and Decision Trees has yielded significant insights and outcomes. This chapter delves into comprehensive recommendations based on the analysis of these models, aimed at enhancing the effectiveness of fraud detection systems. The recommendations are structured to address both practical and technical aspects of fraud detection, including model performance, interpretability, and ethical considerations. Following the recommendations, the conclusion summarizes the research findings, highlights the study's contributions to the field, and suggests potential avenues for future research.

5.1 Recommendations

Another main difficulty related to fraud detection is the trade-off between the precision and recall of the model. Analysis of the two methods shows the specific ways in which they are helpful and the ways in which they are limited when it comes to finding fraudulent transactions. In order to overcome such challenges, one has to create a new model in between Logistic Regression and Decision Trees. For example, one of the approaches could be using a sequential arrangement, where such model as Logistic Regression pre-screen out transactions which are most likely to be genuine. This filtering step is useful in order to narrow the flow of transactions that require more examination. After that, Decision Trees can be used in analyzing the rest of the transactions because one of the strong sides of the Decision Trees is their ability to consider complicated and nonlinear interdependencies of the data. It is thus concluded that the potential utilization of both, Logistic Regression and Decision Trees could contribute to a higher recall and thus to a more effective identification of fraud cases.

Also, adding a layer of Random Forests or Gradient Boosting Machines to the populations involved in Decision Trees improves their general results. These techniques make a number of decision trees and combine results to improve accuracy and decrease overfitting. For instance, Random Forests utilize an ensemble of decision trees that was trained on a random subset of the data, whereas the Gradient Boosting Machines learn a sequence of decision trees, each learning from the mistakes that the previous tree committed. By introducing these ensemble methods it's possible to eliminate the drawbacks of using single Decision Trees and build a more accurate framework of fraud detection.

Another key challenge that should be solved is connected with the imbalance in the sizes of the datasets we deal with. In fraud detection, fraudulent transaction are more rare than normal ones making it easy to develop an imbalanced dataset that will affect the performance of the model. Some of the methods used to handle the class imbalance include; SMOTE or ADASYN where synthetic samples have to be synthesized for the minority class in the data set. It is also possible to bring into models the ideas of class weighting, which will allow increasing the possibility of detecting fraudulent transactions. By applying these techniques, it will be possible to enhance sensitivity to fraud, while keeping the precision of the models' results at a high level.

This is particularly important when it comes to feature contribution of the model to fraud detection and the overall trust in the system. Another advantage of Logistic Regression models is that they are inherently more interpretable because the analysis is based on linear models, and the coefficients are much easier to interpret than some of the other AI techniques because they also tell you the degree of influence that the feature has on the prediction. However, at the same time, it is flexible and allows creating models with non-linear relationships between variables and; however, it can also turn into a Decision Tree of great depth and complexity, which can be hard to comprehend. To promote interpretability, the model must be habitually analyzed regarding which features were given more importance in the prediction of fraud. The described approach can also improve models by optimizing the features because it is important to concentrate on significant features while eliminating those least influential.

Moreover, it is possible to enhance interpretations through the help of such tools as SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations). SHAP values offer an opportunity to understand the effect of the particular feature on the model's predictions and explain the result on the level of individual predictions. Instead, LIME builds local explanations for the single prediction based on a local approximation of the complex model by a simpler, more easily interpretable model in the region of the space surrounding the relevant prediction. Use of these tools also helps to explain how a certain decision was arrived at when addressing fraud detection models.

It is therefore important that fraud detection models be incorporated into real time systems. The possibility to analyze the transactions in real time and to alert on possible suspicious activities immediately is crucial in reducing the costs and preventing frauds. Use of real-time fraud detection systems therefore call for processing of large volumes of data and sound decision-making. Such frameworks as Apache Kafka or Apache Flink may help with real-time data ingestion and analysis since handling a vast number of transactions is critical and should not take much time. These frameworks allow feeding of the new data into the system so that fraud detection models can run in a production setting.

Other important steps in fraud fighting include: Because fraud strategies also transform over time, such models should be updated frequently. In the case of fraud detection the methods used are dynamic and fraudsters are always improving on their methods hence the need for a program to

update on the new methods and patterns. Having a feedback mechanism where all the cases of fraud that have been detected as well as the false positives can be used to feed back can be useful in the training and enhancement of the models. Such a feedback mechanism ensures that models are adjusted in a systematic way as a way of making them responsive to threats in the ATM environment. Biased assessments on the models will go a long way in ensuring the consistency of the model as well as the update of old models that are not as effective as before.

The two bigger concerns, that come with the application of machine learning for fraud detection, revolves around ethics and privacy. Data privacy and security should be maintained to avoid leakage of information or violation of the law at all costs. One measures that need to be taken to ensure that data is protected from intelligently malicious attacks include adequate measures for data encryption, access to the data and transfer of the data. Also important is the compliance with data protection statutes including GDPR and CCPA since consumers trust also needs to be protected.

Ethical issues also concern possibility of prejudice in the machine algorithm. Bias in fraud detection models can cause prejudice to a person and may have an influence in the performance of the predictions made. Periodic assessment and assessment of the models for prejudice is best to prevent such occurrences. It is therefore important that needs to be created for ethical usage of Artificial Intelligent should be created to minimize these worries and ensure appropriate use of machine learning in fraud detection.

5.2 Conclusion

The assessment of Logistic Regression and Decision Tree models for fraud detection: implications of results confirm the effectiveness of the developed approach. Logistic Regression is quite simple and boasts of high or even greater precision depending on the underlying nature of the relationships between features and fraud, which must be linear. Nevertheless, their application is problematic where a direct relationship exists between the independent and dependent factors, and other complicated sequences of fraud are to be identified. While Decision Trees are not as accurate as neural networks they are more accurate than linear models, provide more flexibility and good recall, which makes them suitable for the problem of analysing fraud dataset. Nevertheless, they have the disadvantage of over fitting and the models introduced here are highly complex to interpret. Therefore, the paper through comparison affirms the need for mitigating between

precision and recall probabilities in the detection of fraud. Hybrid of the Logistic Regression and Decision Trees brings out a good solution out of the models by taking the advantages of both. Using ensembles and handling with imbalanced data brings more improvements to the model and several interpretability tools and integration help in making the system more complete for fraud detection. Privacy issues and ethical issues are also important that have to be addressed in order to effectively apply machine learning in fraud detection.

It can be stated that the results of this research hold practical importance for practitioners concerned with the identification of cases of fraud. It is imperative for organizations to understand the trade-off between precision and recall depending on its demands. It is possible to integrate the above models in order to enhance the performance as well as minimize the drawbacks of each model. Also, real-time integration and continuous monitoring of the anti-fraud processes are critical for updating their efficiency when facing the new threats. The paper adds to the knowledge of fraud detection by offering scholar's analysis of Logistic Regression and Decision Tree models and practical suggestion on their enhancement. It also reveals the issues associated with having an imbalance in the distribution of the data, the need for better ways of explaining the models and the fact that ethical considerations need to be incorporated into the machine learning processes. Collectively, these contributions lay down the ground for further research and can prove valuable for practitioners as well as researchers in the domain. There is still a significant number of scholarly areas that may be studied in an attempt to promote the development of fraud detection. Researching the application of machine learning models with new technology like blockchain and distributed ledger can bring new efforts in fraud detection. Furthermore, there also exists additional evidence on the possibility of applying techniques, including deep learning and reinforcement learning, for better understanding their applicability in identifying highly structured fraud patterns. Ethical and privacy aspects should remain priorities of the research and the work must aim at the creation of the set of measures to ensure the ethical use of AI. Intersectorial and information exchange cooperation might help to build greater and integrated fraud prevention measures as well. To sum up, the present work emphasizes the dynamics of the fraud detection theme and the importance of employing machine learning models to counter the existing difficulties connected with abuse. Through the analysis of model strengths/weaknesses and applied recommendations organizations improve the anti-fraud solution and, as a combined effect, reduce organisations' vulnerabilities to financial fraud. The continual improvement and evolution of the new machine

learning methods will help in combating this vice by keeping the mechanisms relevant in this ever-evolving technological world.

Reference

Alarfaj, F.K., Malik, I., Khan, H.U., Almusallam, N., Ramzan, M. and Ahmed, M., 2022. Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. *IEEE Access*, 10, pp.39700-39715.

Ileberi, E., Sun, Y. and Wang, Z., 2022. A machine learning based credit card fraud detection using the GA algorithm for feature selection. *Journal of Big Data*, 9(1), p.24.

Isabella, S.J., Srinivasan, S. and Suseendran, G., 2020. An efficient study of fraud detection system using ML techniques. *Intelligent computing and innovation on data science*, 59

Karthika, J. and Senthilselvi, A., 2023. Smart credit card fraud detection system based on dilated convolutional neural network with sampling technique. *Multimedia Tools and Applications*, 82(20), pp.31691-31708.

Madhurya, M.J., Gururaj, H.L., Soundarya, B.C., Vidyashree, K.P. and Rajendra, A.B., 2022. Exploratory analysis of credit card fraud detection using machine learning techniques. *Global Transitions Proceedings*, 3(1), pp.31-37.

Plakandaras, V., Gogas, P., Papadimitriou, T. and Tsamardinos, I., 2022. Credit card fraud detection with automated machine learning systems. *Applied Artificial Intelligence*, 36(1), p.2086354.

Priya, G.J. and Saradha, S., 2021, February. Fraud detection and prevention using machine learning algorithms: a review. In *2021 7th International Conference on Electrical Energy Systems (ICEES)* (pp. 564-568). IEEE.

Saheed, Y.K., Baba, U.A. and Raji, M.A., 2022. Big data analytics for credit card fraud detection using supervised machine learning models. In *Big data analytics in the insurance market* (pp. 31-56). Emerald Publishing Limited.

Seera, M., Lim, C.P., Kumar, A., Dhamotharan, L. and Tan, K.H., 2024. An intelligent payment card fraud detection system. *Annals of operations research*, 334(1), pp.445-467.

Sharma, S., Kataria, A., Sandhu, J.K. and Ramkumar, K.R., 2022, May. Credit card fraud detection using machine and deep learning techniques. In *2022 3rd international conference for emerging technology (INCET)* (pp. 1-7). IEEE.

Trivedi, N.K., Simaiya, S., Lilhore, U.K. and Sharma, S.K., 2020. An efficient credit card fraud detection model based on machine learning methods. *International Journal of Advanced Science and Technology*, 29(5), pp.3414-3424.

Unogwu, O.J. and Filali, Y., 2023. Fraud detection and identification in credit card based on machine learning techniques. *Wasit Journal of Computer and Mathematics Science*, 2(3), pp.16-22.

Varun Kumar, K.S., Vijaya Kumar, V.G., Vijay Shankar, A. and Pratibha, K., 2020. Credit card fraud detection using machine learning algorithms. *International journal of engineering research & technology (IJERT)*, 9(7), p.2020.

Verma, P. and Tyagi, P., 2022. Analysis of supervised machine learning algorithms in the context of fraud detection. *ECS Transactions*, 107(1), p.7189.