ALTINBAS UNIVERSITY
GRADUATE SCHOOL OF SCIENCES ENGINEERING

Four Classification Methods Naïve Bayesian, Support Vector Machine, K-Nearest Neighbors and Random Forest Are Tested For Credit Card Fraud Detection

LAYTH RAFEA HAZIM

Master of Information Technology

Thesis Supervisor:
Asst. Prof. Dr. Oğuz ATA

Istanbul, 2018

Four Classification Methods Naïve Bayesian,
Support Vector Machine, K-Nearest Neighbors and Random Forest
Are Tested For Credit Card Fraud Detection

by

LAYTH RAFEA HAZIM

Altinbas University

Submitted to the Graduate School of
Science and Engineering in partial fulfillment
Of the requirements for the degree of
Master of Information Technology

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Oğuz ATA
Supervisor

Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)

(Asst. Prof. Dr. Oğuz ATA)
(Assoc. Prof. Dr. Oğuz BAYAT)
(Prof. Dr. Hasan Hüseyin BALIK)

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Asst. Prof. Oğuz ATA
Head of Department

Approval of [Institution] ____/____/____

Assoc. Prof. Dr. Oğuz BAYAT
Director

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

LAYTH RAFEA HAZIM

ACKNOWLEDGMENTS

بسم الله الرحمن الرحيم

In the name of God, the Most Gracious, the Most Merciful.

All praises and thanks to the Almighty, Allah (SWT), who helps me to finish this study, Allah gives me the opportunity, strength and the ability to complete my study for a Master degree after a long time of continuous work. No volume of words is enough to express my gratitude towards my guides, Dr. Oğuz Ata without his knowledge and assistance plus his recommendations this study would not have been successful, he has helped me explore this topic in an organized manner and provided me with all the ideas on how to work towards a research-oriented venture.

Finally, it would not be possible for me to complete the study and this project without the help of Allah and then supporting and encourage from my family and friends. First and foremost, my gratitude goes to my father, my mother and my wife to motivate me and for their endless support for me, may Allah bless them.

Thanks for all persons who helped or contributed to finish my Master program.

ABSTRACT

Four Classification Methods Naïve Bayesian, Support Vector Machine, K-Nearest Neighbors and Random Forest Are Tested For Credit Card Fraud Detection

LAYTH RAFEA HAZIM,

M.S., Information Technology, Altinbas University,

Supervisor: Asst. Prof. Dr. Oğuz ATA

Date: March 2018

Banks suffer multimillion money losses each year for several reasons, the most important of which is due to credit card fraud. In actuality, the issue is how to cope the challenges we face with this kind of fraud. Skewed “class imbalance” is a very important challenge with regard to this kind of fraud. Therefore, in this study, we explore four data mining techniques, namely 'naïve Bayesian (NB)', 'Support Vector Machine (SVM)', 'K-Nearest Neighbor (KNN)' and Random 'Forest (RF)', on actual credit card transactions from European cardholders. This paper offers four major contributions. First, we used under-sampling to balance the dataset because of the high imbalance class, implying skewed distribution. Second, we applied well-known models (NB, SVM, KNN and RF) to our under-sampled class to classify the transactions into fraudulent and genuine followed by testing the performance measures using a “confusion matrix” and comparing them. Third, we adopted cross validation (CV) with 10 folds to test the accuracy of our models with a standard deviation followed by comparing the results for all our models. Next, we examined four models against the entire dataset (skewed) using the confusion matrix and AUC (‘Area Under the ROC Curve’) ranking measure in order to conclude the final results to determine which would be the best model for us to use with a particular type of fraud. In our work, is used the Python programming language. The results showing the best accuracy for the

NB, SVM, KNN and RF classifiers are 97.46%, 95.04%, 97.55% and 97.7%, respectively. The comparative results display that RF performs better than NB, SVM and KNN, and the results, when utilized our proposed study on the entire dataset ('skewed'), achieved preferable outcomes than the undersampled dataset.

ÖZET

Kredi Kartı Dolandırıcılık Tespiti için Dört Sınıflandırma Yöntemi Test Edilmiştir: (NAİVE BAYESİAN, DESTEK VEKTÖR MAKİNESİ, K-EN YAKIN KOMŞU ve RASTGELE ORMAN)

LAYTH RAFEA HAZIM,

Yüksek Lisans, Bilişim Teknolojisi, ALTINBAŞ Üniversitesi

Tez Yöneticisi: Yrd. Prof. Dr. Oğuz ATA

Tarih: Mart 2018

Bankalar, her yıl birkaç nedenden dolayı milyonlarca para kaybına maruz kalmaktadır; bunların en önemlisi kredi kartı sahtekarlığıdır. Aslında, mesele, bu tür bir sahtekârlıkla karşılaştığımız zorluklarla nasıl başa çıkılacağından ibarettir. Yönelimli "sınıf dengesizliği" bu tür sahtekarlık konusunda çok önemli bir sorun oluşturmaktadır. Bu nedenle, bu çalışmada, Avrupalı kart sahiplerine ilişken gerçek kredi kartı işlemleri üzerine dört veri madenciliği tekniğini araştırıyoruz, bunlar: NAİVE BAYESİAN (NB), DESTEK VEKTÖR MAKİNESİ (SVM), K-EN YAKIN KOMŞU (KNN) ve RASTGELE ORMAN (RF). Bu makale dört önemli nokta sunmaktadır. İlk olarak, çarpık dağılımı gösteren yüksek dengesizlik sınıfı nedeniyle veri kümesini dengelemek için alt örneklemeyi kullandık. İkinci adımda, işlemlerin sahte ve gerçek olarak sınıflandırılması için alt örneklenmiş sınıflarımıza iyi bilinen modeller uyguladık, ardından bir "karışıklık matrisi" kullanarak performans ölçümlerini test ettik ve bunları karşılaştırdık. Üçüncüsü, Modellerimizin doğruluğunu standart sapma ile test etmek ve sonuçları tüm modellerimiz ile karşılaştırmak için 10 katlamayla çapraz validasyonu (CV) uyguladık. Daha sonra, belirli bir dolandırıcılık türü ile hangi modelin kullanılmasının en iyi model olacağını belirlemek için sonuçların sonuçlandırılması amacıyla karışıklık matrisi ve AUC (ROC eğrisinin altındaki alan) sıralama hatası kullanılarak tüm veri kümesine (çarpık) karşı dört model incelendi. Araştırmamızda Python programlama dilli kullandık. Dört sınıflandırma yöntemi (NB, SVM, KNN ve DF) için en iyi doğruluğu gösteren sonuçlar sırasıyla, %97,46, %95.04, %97,55 ve %97,7'dir. Karşılaştırmalı sonuçlar RF'nin NB, SVM ve KNN'den daha iyi performans gösterdiğini göstermekte ve bu sonuçlar, tüm veri seti (çarpık) üzerinde önerilen çalışmamızı kullandığımızda, örneklenmiş veri kümesinden daha iyi sonuçlar elde etmiştir.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

The first chapter in this study includes some of sections. First, an overview of credit card fraud problems. It is followed by previous work within detection of fraudulent transactions using machine learning (related work), problem statement and definition, our study research questions, project objectives and aims, scope of this study and lastly, the main structure of this thesis will be discussed.

1.1 Overview

In today's increasingly internet dependent community the use of 'credit cards' has become comfortable and necessary. 'Credit card fraud' is a field with criminals performing illegal acts that may affect other individuals or companies negatively [1]. 'Credit card transactions' have become the actually standard for 'Internet ecommerce'.

In '2026' global trademark 'credit, debit, and prepaid cards' are expected to reach '767' billion purchase transactions for goods and services worldwide. Global brand cards are Visa, MasterCard, Union Pay, American Express, Discover/Diners Club, and JCB [2], as we shown in the fig (1.1) [2]. In 2016 Global card fraud reached '$22.80' billion, as shown in the fig (1.2) below, that figure amounted to '7.15¢' per each '$100' in combined purchase and cash volume of '$31.878' trillion [2].

In the last years, there is an increasing amount of literature on the credit cards fraud detection. The issue of credit card fraud has been studied in [3], [4], [5], [6] and [7]. Fraud detection has become a very important tool to maintain the payment system viability, and to make sure that losses are reduced to a minimum. A secure and reliable banking network for e-commerce needs fast verification and authentication methods allowing genuine users easy access in order to conduct their business, while preventing fraud transactions by others. Presently, 'financial institutions' use a third party 'neural network' based fraud detection system called the 'Falcon Fraud Manager(FFM)' to detect fraudulent credit card transactions [8]. Fraud is a severe problem faced by 'credit card' issuers and can cause great financial losses.

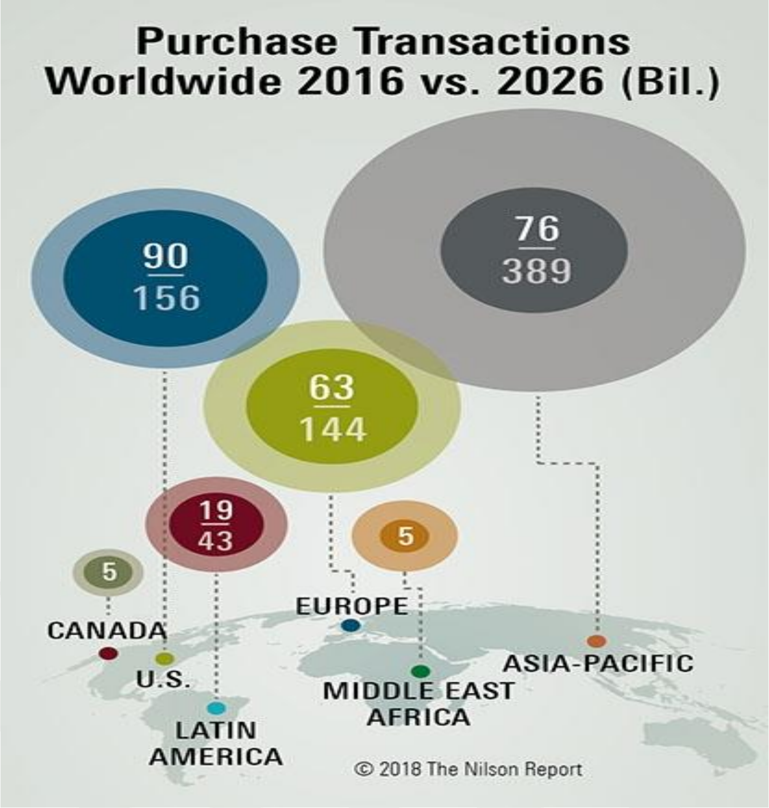


Figure 1. 1: Purchase Transactions for Goods and Services worldwide

Source (The Nilson Report 2018**)**

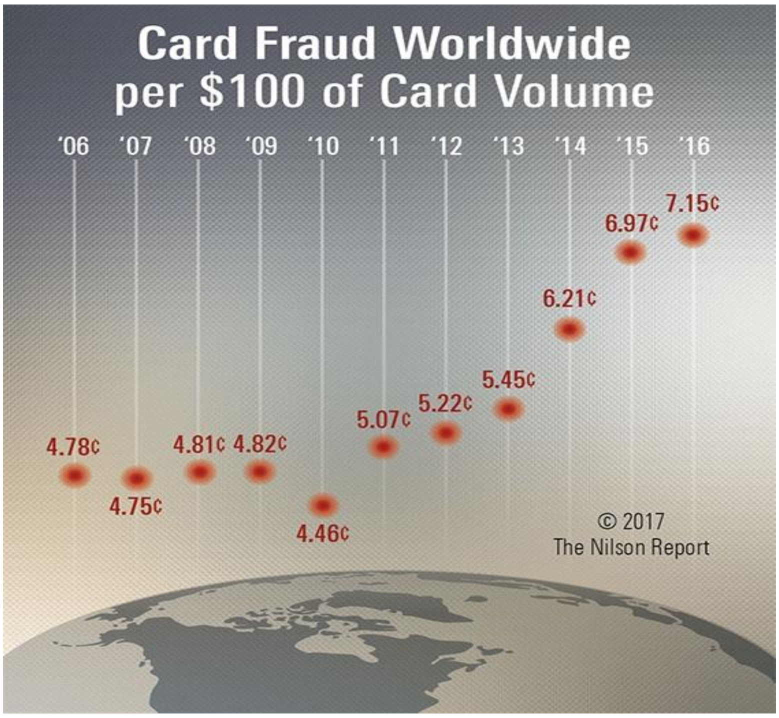


Figure 1. 2 : Credit Card Fraud

Source (The Nilson Report 2018**)**

Credit card fraud is split to two kinds: 1- "Offline fraud is committed" with the use of a stolen physical card anywhere else such as call center, 2- "Online fraud is committed" by phone, shopping, internet, web, or in absence of card holder. Statistical learning means understanding data via statistical and computer analysis and its outputs are detected patterns and knowledge in the data that cannot be obtained using conventional statistical analysis.

Fraud detection may be either supervised or unsupervised [9]. In the first type fraud detection approaches, utilize a data-base of known “fraudulent/genuine” transactions, for the classification of new transactions as fraudulent or genuine. In unsupervised methods, use when there are no prior sets of 'genuine and fraudulent' observations, that means unusual or outliers transactions which are identified as a potential case of fraudulent transactions [10]. Two fraud detection approaches perform a prediction of the possibility of fraud in any of the new transactions [11]. "Credit card fraud detection" depends on the analysis of the 'cardholder' spending behavior. Most of data mining techniques are used to apply on credit card fraud detection, support vector machine [10], [12], [13], [14], many of researcher have used artificial neural network and genetic algorithm [15], [16], [17], [18], [19], [20], credit card fraud detection comparative analysis using logistic regression , k-nearest neighbors and naïve bayesian [21], credit card fraud detection using k-nearest neighbor algorithm [22], [23], hybrid approaches for detecting credit card fraud [7] using random forest, bayesian network, decision tree, naïve bayesian, K* models and support vector machine, implementation of credit card fraud detection [24] is based on bagging ensemble classifier, as well as some used hidden markov model (HMM) [25], [26], migrating birds optimization algorithm [27], Real-time credit card fraud detection with the use of computational intelligence self-organizing map (SOM) [1], [28].

Classification issues that are of the most widely known prediction issues in supervised learning, were traditionally tackled with the data mining methods. The objective taken in those methods is a statistical one in which the aim is minimizing the number of falsely classified records. For binary classification both classes are referred to as positive (P) and negative (N). If a P record is properly classified as P, it is a true positive (TP) and if an N record is properly classified as N, it is a true negative (TN). The other 'two metrics' are incorrect ones (false positive: 'FP' and false negative: 'FN') [29].

The implementation of machine learning on credit card usage has its advantages and disadvantages. Due to a greatly increased number of transactions during the past decade, the credit frauds gained an increasing trend as well [30]. Therefore, databases now store a huge amount of data concerning whether transactions are fraudulent or genuine. This research, we tested three advanced data mining methods, 'naïve bayesian (NB)', 'support vector machines (SVM)' and 'k-nearest neighbors (KNN)', with the well-known 'random forest (RF)'. This research depends on “real-life data” of transactions from an European cardholders.

1.2 Literature Review

The detection of Credit card fraud is a 'binary' classification task in which a credit card transaction is labelled as either fraudulent or genuine. "Data mining approaches" are useful to this type of fraud detection because of their ability to identify small anomalies in huge data sets. In this section, we reviewed some previous research relevant with our study [3].

1.2.1 Under-sampling Approach

Under sampling imbalanced class means deleting part of the data in the majority class or the negative class (genuine) [12] . Many researchers have used the under sampling approach to balance the training data for fraud detection systems [3]. The under sampling approach used for [31] they have used two sampling approaches oversampling and undersampling commonly used in machine learning algorithms to imbalanced (skewed) classes and costs for misclassification their study cost curves to explore the interaction of undersampling and oversampling with the learner C4.5 of decision tree, where they concluded that under-sampling results in a reasonable sensitivity to variations in the costs of misclassification and class distributions and Over-sampling has shown a little sensitivity, the paper [32] they employed the three algorithms logistic regression, C4.5 and random forest for cost sensitive credit card fraud detection they applied those algorithms on the full dataset and on the undersampled dataset, when it applied the undersampling the best results are found, a comparative study [10] is included test different levels undersampling class distributions by data mining techniques, comparison results showed undersampling generally perform better, the hybrid undersampling and oversampling for credit card transactions using machine learning techniques [21] they also gave the same assessment of this approach the achieved two sets distributions (10:90 and 34:64) for analysis, the paper [33] effectiveness of undersampling on unbalanced classification, that proposed an integrated analysis for two objects having the biggest effect on the efficiency of an under-sampling approach: the increasing of the variance because of reducing the number of samples and counterfeiting (i.e. warping) of the posterior distribution because of the variations of priori possibilities as well as their impact on the result of accuracy, they conclude two main influences: 1- It raises the classifier’s variance and 2-Results in counterfeited (i.e. warped) posterior possibilities. Usually, the first influence is addressed using averaging methods for reducing the variability and the second needs the calibration of the possibility to the new priors of testing.

1.2.2 Credit Card Fraud Detection

Growing use online payment by credit card, as a final result, fraudsters are also increasing to get money. Through the significant contribution of researchers in recent years in finding the best ways to reduce frauds by using the data mining techniques or artificial intelligence machine learning.

'Data mining' for 'credit card fraud' [10] utilized comparative to three methods 'support vector machines(SVM)', 'random forest(RF)' and 'logistic regression(LR)' to evaluate the best one depending on performance measures, they used undersampling class imbalanced for their real transactions dataset from international company with various proportions and they divided in two subsets, after then applied three proposed techniques with cross validation performance the results were, respectively SVM (93.8 accuracy, 52.4 sensitivity, 98.4 specificity), RF (96.2 accuracy, 72.7 sensitivity, 98.7 specificity) and LR (94.7 accuracy, 65.4 sensitivity, 97.9 specificity). The authors in [7] proposed a hybrid approaches of six well-known data mining techniques, namely, decision tree (DT), random forest (RF), Bayesian network (BN), Naïve Bayes (NB), support vector machine (SVM), and their proposed model is (K*) employed these models to detecting credit card fraud, they are combined ensemble of artificial intelligence (AI) models are applied into real life transactions from a leading bank in Turkey, the results in terms of performance measures were, respectively DT (95.19 accuracy, 52.53 sensitivity, 97.35 specificity), RF (95.81 accuracy, 50.84

sensitivity, 98.09 specificity), BN (96.92 accuracy, 50.00 sensitivity, 99.30 specificity), NB (94.10 accuracy, 92.57 sensitivity, 94.18 specificity), SVM (94.17 accuracy, 66.89 sensitivity, 95.55 specificity) and K* (91.37 accuracy, 73.14 sensitivity, 92.67 specificity). They investigated in [34] the efficiency of personalized models in comparison with the aggregated structures in identify fraud for various people, authors used two techniques for comparable are random forest (RF) and naive Bayesian (NB), the dataset collected from actual transactions and some other information via an on-line questionnaire, the performance results for their proposed showed (RF) is of a more efficient performance than the (NB) for the aggregated model whereas (NB) is of a more efficient performance in the personalized models, respectively RF (91.09 accuracy, 91.1 sensitivity, 91.9 precision), NB (96.04 accuracy, 96.00 sensitivity, 95.9 precision) and RF (96.18 accuracy, 96.00 sensitivity, 96.00 precision), NB (95.08 accuracy, 95.00 sensitivity, 95.00 precision). Researchers in [24] proposed the three techniques for credit card fraud detection are naive Bayesian (NB), support vector machine (SVM) and k-nearest neighbors (KNN), not alone but they used these models with collaboration ensemble learning methods, the evaluation of performance is done on a real dataset transaction from UCSD-FICO competition, and the authors showed the bagging classifier based on decision tree, as the best one for fraud model. The study [35] used classification methods are artificial neural networks (ANN) and logistic regression (LR) for create best model to detecting credit card fraud, where they concluded the genetic algorithm is the best in their literature and they proposed to apply its on bank to predicted fraud soon after credit card transactions. The paper [36] employed three supervised methods to predicting credit card fraud are logistic regression (LR), gradient boosted trees (GBT), and deep learning (DL), authors researched also explores the benefits according to features by used domain expertise and feature engineering to compares with the three techniques mentioned above, they concluded using domain expertise for feature engineering is the best and their results after applied cross validation with 5 fold were, respectively LR (83.8), GBT (87.4) and DL (86.2). Presented [37] a survey of two techniques are Hidden Markov Model (HMM) and k-means clustering, they adopted to the analysis spending behavior for cardholders, (HMM) categorized the cardholder's profile into low, medium and high, and then made clustering by used k-means clustering for the categorized cardholder behavior, HMM has ability to detect the new arriving transaction is fraudulent or genuine completely.

Previously, we have historically reviewed comparative studies for credit card fraud detection, now we will review some historical studies for machine learning and features engineering. The study [30] showed, it is the way of extracting the proper traits from the transactions for constructing credit card fraud detection approach, by aggregating the transactions, and they expanded the transaction aggregation strategy, as proposed creating a new group of properties according to analysis of the time of transaction by employing the "von Mises" distribution. Topological pattern in [38] discovered the 'topological patterns' of 'fraudulent financial reporting' FFR via dual 'GHSOM' ('Growing Hierarchical Self-Organizing Map') approach, as well as presented an expert competitive feature extraction mechanism, revealed accurate to detect the fraudulent and genuine by used the topological patterns for FFR and feature extraction. On the other hand, the authors in [39] proposed a linear discriminate as a fisher discriminant function to detecting credit card fraud for the first time, their experiment resulted from the fisher discriminant function more profit for fraudulent/genuine classifier. The study [40] proposed combines of intrinsic features derived and network based features for cardholder behavior merchants, their results for both two types combination are two strongly tangled, and leads to the best performance models where the 'AUC' reach higher than 0.98. A new cost sensitive decision tree in [41] compared the traditional popular classification method with the performance like precision and true positive rate to minimize the sum of misclassification costs, the outputs showed that the cost sensitive decision tree may be ready and implemented in real transactions to avoid fraud for credit card transactions. The study [42] applied k-nearest neighbors (KNN) method and outlier detecting approach to put the optimal solution for credit card fraud issue, where those two methods minimizing the false alarm rates and minimizing the fraud detecting rate to prevent the fraudulent transaction. In this paper [43] implemented the self-organizing map (SOM) for credit card fraud detection because this approach it is very efficient, is a part of neural network and unsupervised learning, focuses on real time credit card fraud detection, they concluded the SOM better accurate for detecting fraud because of used clustering with that model.

1.3 Problem Statements

During the growing in credit card transactions, such as the electronic payment system, there was an increase in “credit card fraud”, and (70 percent) of US customers are most concerned about identity fraud [10], [44]. The “Federal Trade Commission’s” on-line data-base of customer complaints has received (13) million complaints from year 2012 to 2016, with 3 m. in 2016 alone. Of them almost, 42% were related to fraud, and 13% were complaints concerning identity theft [45]. Thus, banks are trying to decrease their losses from card fraud.

Turkey is a continuously growing e-commerce market, with a high rate of card penetration and potentials. The main resource of payment fraud is “card-not-present” fraud and robust fraud preventing measurements will be required of ensuring profitable business expanding in that market. With 57 m. credit cards in circulation, Turkey is of a high rate of card penetration (nearly 75 percent of population) and there has been a fast increasing in card use [46]. With the expanding of the market of credit card, criminals devised numerous methods for getting around improved security measurements, like the magnetic stripes and holograms. Fraud analysts are security officers trained to examine the cardholder's historical behavior and by considering different factors determine the potential risk associated with the flagged accounts. As well as this problem of fraud it got with my father also almost a year ago, where the amount of his personal account was stolen in Ziraat bank, it turns out there is a fraudster in Istanbul in that day stole many accounts.

Actually, consideration should be taken to the development of fraud detection methods such as data mining techniques, because fraudsters develops also their fraud practices for avoiding detection [9]. Hence, credit card fraud detecting techniques require continuous innovation. This research evaluates four techniques, including "naïve bayesian, support vector machines, k-nearest neighbor algorithm and random forests" to try detecting credit card fraud. It examines the performance for these techniques, but we faced many of challenges for this study, where the 'fraudulent' behavior look like the 'genuine', real datasets transactions aren’t made available and results are typically not declared to the public and even if we found will be high imbalanced ('skewed') [3]. So, feature selection is a problem with this study because large disparity in measurement and high dimensions of fraud dataset and presence of numbers of 'features' /'attributes' /'inputs' make to apply of "data mining" techniques and detection very difficult and complicated, choose existing performance measures for the aggregating techniques we used them that very importante, there are four most commonly used are accuracy, sensitivity, specificity and precision all of them depend on true positives, false positives, true negatives and false negatives [6] , in the case of classifying the incoming credit card transactions as fraudulent or genuine, the cost of a FN (i.e. missing to name a fraudulent transaction as fraudulent) is much larger than the cost of a FP (false alert), which is typically variable. These performance measures are affected by the type of sampling used for data set. We investigated in this study the effect of aggregating sampling on performance of fraud detection techniques are "naïve baysian, support vector machines, k-nearest neighbor and random forest" classifiers on high imbalanced credit card fraud transactions ('skewed'), as well as their impact on used of undersampling fraud transactions.

1.4 Research Methodology

The objectives of exploring the efficiency of traditional algorithms of data mining in dealing with the data management needs of credit card fraud problems (CCFPs) [47], have been evaluated by means of using python. As pointed out earlier, the verification of suspicious transactions with the cardholder is a major part of fraud investigation and cannot be eliminated. Therefore, any solution that refines the investigation selection process by reducing the number of unnecessary calls is welcomed by the world banks.

To overcome this problem with that field (CCFPs), we used in this study analytical comparison and investigated of credit card fraud detection using NB, SVM, KNN and RF techniques on high imbalanced data (skewed) based on accuracy, sensitivity, specificity, precision and finally region under the ROC curve (AUC) is utilized as a standard measurement of classification performance [14], where we used (AUC), finally to examine all techniques with skewed credit card transactions to obtain the best technique even we can advise to use with that type of fraud. In this study enhances the handling of high imbalanced credit card fraud data in [48]. This study used high imbalanced dataset transactions which contains about 0.172% of fraud transactions is sampled in aggregating approaches. The fraud transactions indicates to positive class while the negative class (genuine), by using the undersampling approach to overcome the skewed or imbalanced transactions as a part of preprocessing dataset because of the small fraudulent credit card transactions percentages of total number of the transactions, balancing handling mechanism is desired to make this data balanced with distribution of '1:1' between 'genuine' and 'fraudulent' class to reshape class imbalance [12], [49], where the distribution is in the format of '50:50'. Applied four techniques to the undersampled dataset using confusion matrix to calculate the accuracy, sensitivity, specificity and precision to comparison the performance of the four techniques, after then to verify for the performance measures more we applied a cross validation with 10 fold and Grid Search of the aggregating techniques and comparison the performance. Finally, applied our aggregating techniques to imbalanced dataset (skewed) and calculate the accuracy, sensitivity, specificity, precision and AUC to comparison for each technique to get the most accurate technique in this field of fraud.

1.5 Project's Objectives

The purpose of this study is to classify credit card transactions as fraudulent or genuine, where supervised learning algorithms are utilized. Thus, each individual transaction in the provided dataset is already assigned to one of the known classes (fraudulent or genuine). Before onset of this particular research, the analyzer has hold out through screening of the problem focus, so that a proper aim could be devised. Accordingly, the study is aimed at inspecting the performance effectiveness of traditional data mining techniques in dealing with the credit card fraud problems (CCFPs). Consequently, the analyzer has devised the following objectives for attaining the aim of the study:

- To explore data mining algorithms.
- Identify the best credit card fraud detection (CCFD) technique for classifying real life transactions.
- To inspect the credit card fraud (CCF).
- To derive the challenges hidden in data mining of (CCFPs) with traditional algorithms.
- To construe the asserted allegation of inefficiency of data mining algorithms with (CCFPs).

1.6 Thesis Structure

The presented models have been utilized in all of the aforesaid implementation scopes of classification and have been compared to the related works in order to display the contributions of our models taking under consideration the gross net profit which is the main aim of this thesis. The whole thesis put up with the below mentioned structure:

Chapter 1: This chapter named (introduction) presents the overview about our topic of background related description. In touch to the background context, the chapter proceeds with the statement of the problem on the foundation of which the aim and objectives of the research have been formulated. Thus, the chosen research methodology for the maturity of the formulated objectives is briefly described; consequently, displaying the entire theme of the study that decide the credibility of the study.

Chapter 2: introduces the notions and tools that will be considered in the thesis. The chapter is divided in three main parts, the first section provides the reader prefatory knowledge about challenges of credit cards fraud detection system, the trouble of classification and describes the main layers of a fraud detection system. The second section is devoted to the machine learning and what meaning of supervised and unsupervised. The third section is clarify some of data mining techniques are used with this field of fraud.

Chapter 3: Introduces the existing fraud solution approaches and adopted research methodology is described in detail.

Chapter 4: The researcher has given the pertinent results for leading the research towards conclusion.

Chapter 5: In this part, presents the conclusions of this research and offers suggestions for further study.

CHAPTER TWO

FRAUD DETECTION MATRIALS

Several of authorization methods have been utilized for preventing credit card fraud situations, like signatures, credit card number, ID number, cardholder's address, expiry date, and so on. Nevertheless, those methods are not sufficient for hindering credit card fraud. Thus, there have to be fraud detection methods that analyze data, which in turn can discover and eliminate the cases of credit card fraud [50]. Credit card fraud prevention is the first line of defense in reducing expenses that are associated with credit card fraud. Once fraud prevention fails, it is essential for fraud detection methods to identify fraud as soon as possible. Data mining techniques are relevant to fraud detection because there is a need for fast and efficient algorithms to search for patterns in large databases [51]. This chapter presents detailed descriptions to how handling the challenges of credit cards fraud detection system such as (techniques for unbalanced classification (skewed) class distributions and features augmentation) and descriptions of data mining methods are outlined, beginning with the introduction of the three main challenges after then, explains the two classes of machine learning – which are namely, the supervised and the unsupervised. The algorithms for various fraud detection approaches are discussed, and the four common techniques in combining multiple algorithms are described.

"Credit card fraud detection" is a binary classification problem in which a 'credit card transaction' is classified as either a genuine transaction ('negative class') or a fraudulent transaction ('positive class'), Older fraud detection software tools have their roots in statistics (cluster analysis), whereas the more recent tools are based in data mining (due to increased power of modern computers and massive datasets) [52]. Data mining is an operation of obtaining patterns from data, and a procedure of analyzing data from various points of view and then summarizing it into useful information which may be utilized for increasing revenue, cut expenses, or both.

Ordinarily, data mining is the procedure of detecting correlations or patterns among a large number of fields in large 'data-bases' [53], the data mining gives the users chance of analyzing data from several of various dimensions or angles, classify it, and epitomize the detected correlations. Generally, machine learning is categorized into two basic types, supervised and unsupervised learning [53].

2.1 Challenges of Credit Cards Fraud Detection System

Fraud detection is a complicated field, it can be found that a fraud detection system is under the threat if failing, it is of low precision degree or reports many alarms that are false [49]. It is obvious that there are still facets of 'intelligent fraud detection' that have not been investigated and extremely difficult for 'e-commerce systems' to handle fraud problems putting them in the position of incurring massive losses. This occurs due to the fact that fraud detection systems must deal with several issues. In this section we present several challenging and problems that are associated with credit card fraud detection and which the systems must deal with.

2.1.1 Concept Drift

Concept drift in data mining indicates the phenomenon that the underlying structure or concept are changing over time [54]. Fraud detection systems work in dynamic environments in which the behavior of a genuine user or fraudster keeps varying is known as the phenomenon drift concept [55]. Due to the fact that credit card holders are continuously changing their behavior that may happen because of certain circumstances (such as, Christmas holidays), and in this situations, the user purchase power will raise. In the case where fraud detection system doesn't treat this as normal change, it will be treated as fraudulent case and alarms will be triggered, which will lead to locking the transaction of the card-holder, and that results in a regression of the reputation of the bank. Hence, the fraud detection system requires effectively discriminating and classifying fraudulent and genuine transactions. Moreover, credit card fraud detection system has to be capable of capturing and adapting the card-holder’s drifting behavior, updating detecting models for that behavior throughout time. For that reason, credit card fraud detection system has to have high detection accuracy and low false alerts.

Consequently, there are several approaches used from researchers in order to handle concept drift in credit card systems, namely the first one developing based method and the second one regulated based method, the majority of those existing adaptive fraud detection systems which are handling concept drift are using evolving based approach which includes Adaptive ensembles and Base model specific methods [56] as shown in Fig (2.1).

2.1.1.1 Evolving based method

The learner is capable of automatically adapting its behavior in staying updated with the stream dynamics [5].

2.1.1.2 Regulated based approach

The concept drift and classification are taken care of as standalone problems [5].

Several researchers are utilized evolving learning approach under adaptive ensemble classifier technique for dealing with concept drift. The study in [56] proposed a sufficient credit card fraud detecting structure which is mining concept drifting data series with the use of a weighted group classifiers. They have trained aggregating of classification approaches C4.5, RIPPER, naïve Bayesian, and others, from sequential bulks of credit card data, as well [5] presented an adaptive structure for fraud detection system specialized for credit cards.

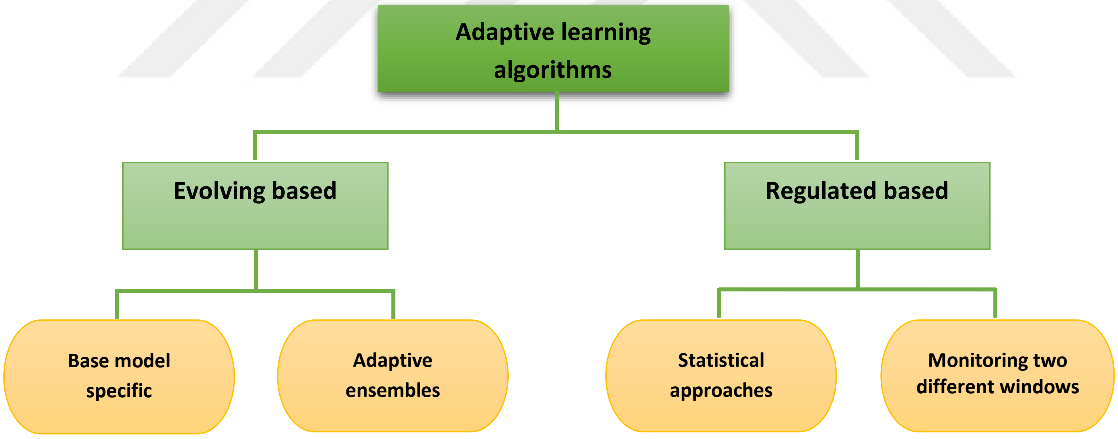


Figure 2. 1: Adaptive learning algorithms

2.1.2 Skewed Class Distribution

Imbalanced class distribution (skewed) is one of the most prominent problems that are faced by fraud detection system. In general, the imbalanced class issues is the case in which there are noticeably less fraudulent transaction samples than genuine ones [57]. In credit card fraudulent transactions are of quite little proportion of the overall number of the transactions, and that may be causing obstacles for the efficiency of the "fraud detection system (FDS)". Specifically, in credit card systems the false classification of a genuine transaction causes the customers being dissatisfied, and that is of more harm than fraud itself.

There is a number of approaches dealing with this issue and it is possible to distinguish them from one another, especially ones operating at first the data levels and second algorithmic levels [58]. At the first level, the imbalanced strategies are utilized as a preprocessing stage for balancing the data-set or removing noise between the two classes, prior to applying any algorithm. In the second level, algorithms are themselves adjusted to take care of the minority class detection.

All the methods presented in the following section will discuss the unbalanced problem as referred to between class imbalance, i.e. imbalance in class frequency. However, class imbalance can exist also within the class [59] (due to small clusters within one class), and this problem is often linked to the presence of rare cases [60], Fig (2.2) shows the balanced approaches and techniques.

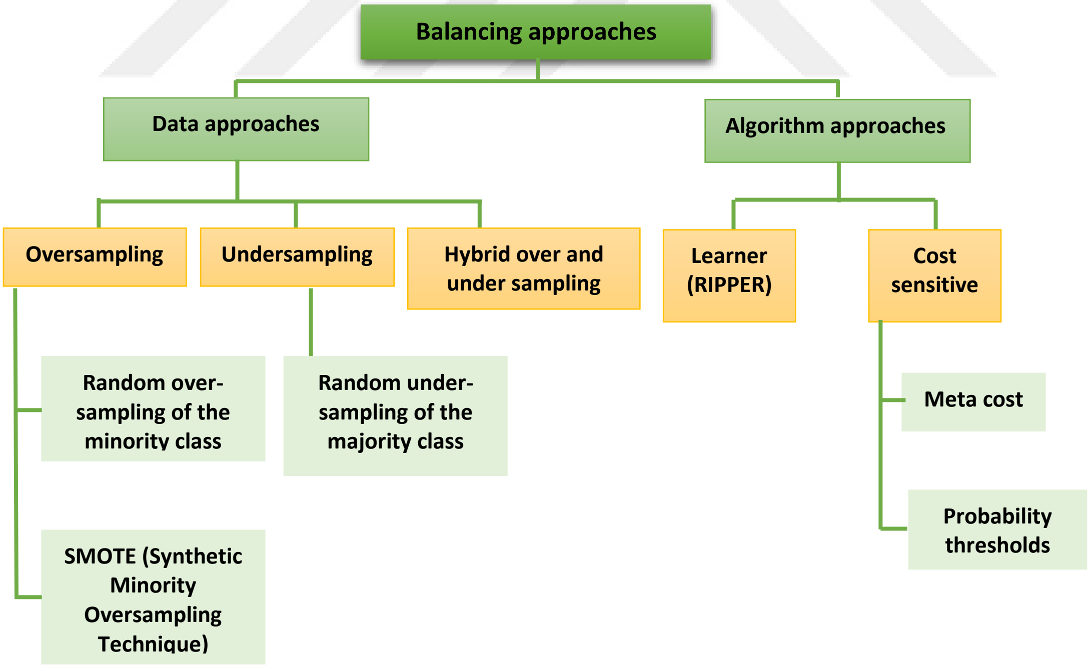

Balancing approaches
Data approaches
Algorithm approaches
Oversampling
Undersampling
Hybrid over and under sampling
Learner (RIPPER)
Cost sensitive
Random over-sampling of the minority class
Random under-sampling of the majority class
Meta cost
Probability thresholds
SMOTE (Synthetic Minority Oversampling Technique)

Figure 2. 2: Balancing class distribution techniques

2.1.2.1 Data level methods

In fraud detection system resources, the majority of the researchers employed data level balancing approaches into three main categories: under-sampling, oversampling and SMOTE (which is an acronym for “Synthetic Minority Over-Sampling Technique”).

2.1.2.1.1 Under-sampling

Under-sampling consists in downsizing the majority class via the elimination of the part of observations at random. In the imbalanced problems it makes sense assuming that several observations of the majority class are repeated and that via the random elimination of a part of them the producing distribution should not be a lot different. On the other hand, the risk of the elimination of suitable observations from the data-set still exists, due to the fact that the elimination is performed in an unsupervised way. This method is often utilized due to its simplicity and accelerates the speed of the learning stage [31].

2.1.2.1.2 Over-sampling

Over-sampling refers to randomly up-sizing the small classes (i.e. the minority) diminishing the class imbalance degree. Via the replication of the minority class up to the point where the two classes are equally frequent, over sampling approach is rarely used because raises the risk of over-fitting through biasing the model towards the minority class. Other disadvantages of this method lie in the fact that it doesn’t add new informative minority cases and that it slows down the training. This may be specifically of no effect in the case where the original data-set is quite big [31].

2.1.2.1.3 SMOTE

Oversamples the minority class via the generation of synthetic samples in the area that surrounds the monitored ones. The concept is forming new minority examples via interpolating between same class samples. Which has the influence of producing clusters that surround every one of the minority observations. By creating synthetic observations the classifier builds larger decision areas containing surrounding samples from the minority class. SMOTE has shown to improve the performances of a base classifier in many applications, but it has also some drawbacks [61]. Synthetic observations are generated without considering neighboring examples, leading to an increase of overlap between the two classes [62]. Borderline-SMOTE [63] and ADASYN [64] were suggested for overcoming this issue.

2.1.2.2 Algorithmic level methods

Depending on their applications we distinguish between cost-sensitive learning and imbalanced learning. Algorithm oriented methods are essentially a modification of existing classification algorithms for unbalanced tasks. In first case, the goal is to improve accuracy of the minority class, while in the second step the objective is to minimize the cost associated to the classification task. There are several classification algorithm types dealing with fraudulent classes [65]. The algorithm level approach deployed:

2.1.2.2.1 Cost-Sensitive learning

This type of learning deals with distributions of skewed class. Cost-sensitive learning puts a cost variable to mis-classification of various classes with the assumption of the availability of a cost-matrix for the various error types. In unbalanced classification tasks, it's typically of higher importance to correctly predict positive (minority class) transactions than negative (majority class) transactions. This is often achieved by associating different costs to erroneous predictions of each class. Cost-based methods operating at the algorithmic level are able to consider misclassification costs in the learning phase without the need of sampling the two classes. Such as these classifiers are cost-sensitive boosting [66], [67] SVM [68] and Neural Network [69].

Cost-based splitting criteria in the family of decision tree classifiers are used to minimize costs, or cost information determine whether a subtree should be pruned [70]. Generally, pruning allows improving the generalization of a tree classifier since it removes leaves with few samples on which we expect poor probability estimates. In fraud detection system resources, there are two basic methods which were suggested for the utilization of cost-sensitive learning for imbalanced classes, namely, Meta-cost and Thresholds or usage of learners that aren't sensitive to the issues of class imbalance [71]. Those approaches are utilized often in fraud detecting system for the balancing of the training data. Use Metacost Domingos proposed a general framework that allows transforming any non-cost-sensitive classifier into a cost-sensitive one and similar with thresholding [72] allows using cost-insensitive algorithms for cost minimization via different classification thresholds.

2.1.2.2.2 Imbalance learning

The use of the learner for handling skewed distribution that is one other algorithmic approach that is utilized in the resources of fraud detection systems. Those learners either resist class imbalance issue via inherent learner features, such as in the situation of the "Repeated Incremental Pruning to Produce Error Reduction" (RIPPER) method as stated in [73]. Moreover, learners are hardened against the issue via internal modifying such as in the situation of KNN or the support vector machine learners.

A SVM optimized in terms of F-measure is presented by [74], while the [75] use SVM with RBF kernels as base classifier for AdaBoost. In the family of lazy learning classifiers, the study in [76] proposed a K-nearest neighbor weighting method designed to handle the issue of class unbalance. The algorithm, called CCW-KNN (Class Confidence Weights KNN), is capable of correcting the inherent bias towards the majority class in existing classifiers of the K-nearest neighbor.

Finally, data approaches are more efficient than algorithmic approaches [65]. Because data approaches are easier to be implemented and don't result in increasing the time of training time or the required resources. Thus, the majority of the fraud detection system resources utilize data level balancing approaches [5].

2.1.3 Reducing Large Data Amounts

High dimensions and large-scale of fraud dataset and existence of numbers of /inputs /attributes /features / variables make the data mining procedures and detecting very hard and complex [77]. The methods of data reduction include first once **dimensionality reducing** and second **numerousity reducing** [49]. Reduced data highly affects the efficiency of the fraud detection system. It is of high importance in credit card systems due to the fact that it performs a reduction of processing time of transactions in addition to the complexity in transaction processing. Credit card features are utilized for deciding the consuming habits of the card-holders that are massively correlated with the cardholder's features. There are nearly 30 features detected in a credit card, such as cardholder's age, cardholder's profession, cardholder's income, Credit card type, Number of the used cards, Credit grade, balance, the frequency of card usage, over-draft frequency, Time bracket, Credit line, overdraft but not bad debt frequency, bad debt frequency, times of Card usage, shopping Growth rate, Average daily spendings, Overdraft rate and so on, use of data reduction approach in the above to solve this problem.

2.1.3.1 Dimensionality reduction

This approach includes several of strategies, which are **data compressing**, **property construction** and **property selection** are the most often utilized strategies of fraud detection systems, Data compressing strategy performs a compression of the original data representation via using data compressing approaches like in [78]. Meanwhile, property generation is in which a small group of more beneficial properties are obtained from the genuine group and properties selection is one other dimensionality reducing approach, the most important and useful properties are selected for usage in structure generation. Three property selecting approaches are utilized in fraud detection system: filter approaches, wrapper approaches and embedded approaches. This approach of Dimensionality reducing was deployed in credit card fraud detection system via applying Principal Component Analysis (PCA) [78], for the sake of reducing credit card training data-set dimension.

2.1.3.2 Numerosity reduction

This approach is the data is replaced via smaller representations such as the use of data aggregations [5] [14], and that is a non-parametric approach for the aggregation of credit card transactions for the sake of capturing customer buying behaviors before every one of the transactions and utilized those 'aggregations' for model estimations for the detection of fraudulent transactions. 'Dimensionality reduction' and 'Numerosity reduction' for 'Data reduction' approaches contain as presented in Fig (2.3).

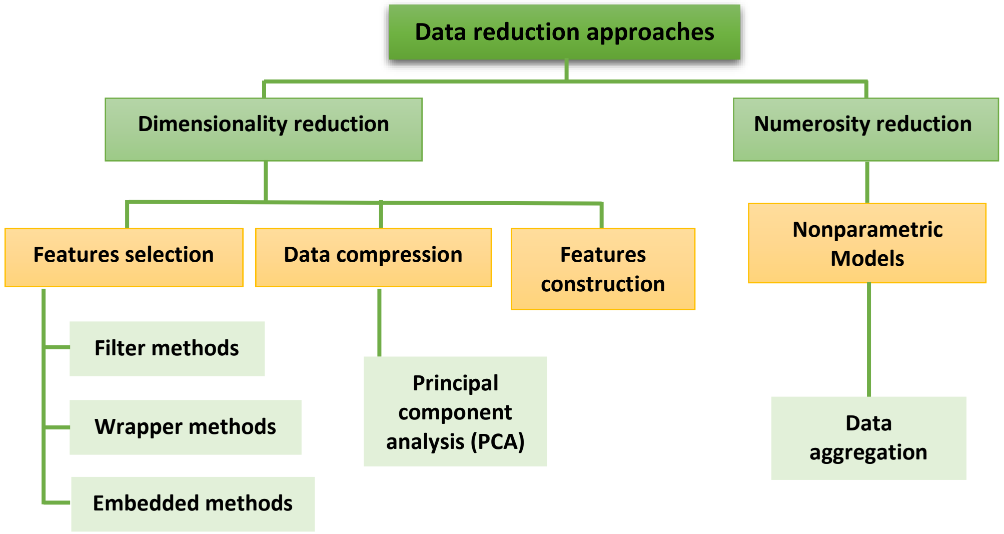


Figure 2. 3: Data reduction strategies

2.2 Supervised and Unsupervised Learning

Fraud detecting approaches may be divided to supervised or unsupervised learning. The supervised type of learning in the fraud detection is an approach which applies algorithms on each of fraudulent and original instances for the construction of models which grant new observations to one of the two classes, those classes being either fraudulent or genuine. The aim of the supervised method is building an accurate structure of distributing class labels concerning the properties of the predictor [52]. In supervised methods fraudulent and genuine examples are utilized for the prediction of the class of a new observation, the resultant from classifier is afterwards utilized for assigning class labels to the test examples in which the predictor features values are known, but the class label values are unknown.

Unsupervised methods are applied when there are no prior sets of 'genuine' and 'fraudulent' supervisions. Since they are not based on examples of fraud or genuine transactions, this learning method simply decides which of the observations are least similar to the norm. Unsupervised algorithms look for similarity in the training data for the determination if the instances may be featured as producing a set. Unsupervised strategies have the advantage of being independent of their selection, and are able in theory to discover frauds still unobserved, that have not been detected by an expert, therefore unsupervised learning is usually referred to as “cluster analysis” and has the aim of grouping data for automatically developing classification labels [79].

Inductive learning or classification, takes place when a learner or classifier, example (neural network, decision tree, rule-learners and SVM is applied to some data to produce a hypothesis explaining a target concept, the search for a good hypothesis is dependent on fixed bias embedded by the learner [80]. The algorithm is said to be able to learn due to the fact that the quality of the hypothesis typically gets better as the number of instances increases. On the other hand, since the bias of the learner is fixed, successive implementations of the algorithm on the same data always results in the same hypothesis, invariant of the performance; no knowledge is commonly obtained across tasks or domains.

2.3 Base Classifier Techniques

This thesis utilizes the supervised methods learning. As mentioned above, this is a machine learning approach utilizing a training data-set with known target classes for the production of an inferred function which performs pairing of an input into a wanted value of an output. This inferred function, called a (classifier), should almost the correct output even for examples that have not been shown during training. There are five main supervised data mining techniques: logic based approaches (decision trees), statistic approaches (Bayes/Regression), instance based learners (KNN), perceptron based techniques (NNs) and SVMs. It is preferable to use Logic based when dealing with discrete or categorical features while multidimensions and continuous properties support vector machines and NNs are mining approaches of choice. SVMs and Neural network models need large training data-set sizes in order to achieve the maximum prediction precision, while the Bayes method merely needs quite smaller data-set size [81]. Non-useful features are of a large negative effect on the procedure of training process of the K-nearest neighbor and NN approaches, and due to those irrelevant features the training of classifiers according to those approaches may regularly be insufficient and in some cases impractical [81].

Since there are strengths and weaknesses for each algorithm, a strategy is required to determine the best base classifiers to use in the credit card domain. Following sections detailed and descriptions of seven data mining techniques that were used in experimentation are presented.

2.3.1 Random Forests

The decision tree popularity in data mining is due to them being easy to use, flexible and interpretable according to dealing with different data feature kinds. On the other hand, single trees, may be unstable and are of high sensitivity to certain training data. Ensemble approaches aim at addressing this issue via improving a collection of aggregating and models their predictions in deciding the class label for each one of the data points. A random forest [82] technique is a set of regression (or classification) trees. Those sets are efficient if specific members are not similar, and arbitrary forests get variation amongst separate trees with the use of 2 randomness resources: first, every one of the trees is generated on individual bootstrapped examples of the training data and second, only an arbitrarily chosen sub-set of data features is taken under consideration at every one of the nodes in the construction of the separate trees. As such, arbitrary combine the ideas of bagging, in which separate models in a set are improved via sampling with replacement from the training data, and the arbitrary sub-space approach, in which every one of the trees in a set is constructed from an arbitrary sub-set of features. Considering a training dataset of R samples characterized by N attributes, every one of the trees in the set is constructed in the following manner:

- Find a bootstrap sample of R samples
- At every one of the nodes, arbitrarily choose a sub-set of n<N features. Select the optimal split at the node from this reduced group of b attributes
- Complete the entire tree with no pruning

Random forests gained popularity in implementations of the last decade. Due to their ease of use, with merely two parameters that are adjustable, the number of trees (T) in the set and the size of the attribute sub-set (n), with robust performance noted for typical parameter values [82].

Applying random forests for fraud detection is a rather new area, with a limited number of submitted researches. This study [14] produces random forests in order to produce superior performance in credit card fraud detection.

Random forests are computationally sufficient due to the fact that every one of the trees is constructed separately from the rest. With a big number of trees in the set, moreover, it should be noted that they are robust to over-fitting and noise.

2.3.2 Bayes Network

Bayes belief networks are efficient modeling tools to condense everything known concerning reasons and influences to a compact network of possibilities. A Bayes network is a graphic structure for probabilistic relations amongst a collection of variables. The 'Bayes network' became a common representation to encode uncertain expert knowledge in 'expert systems' [83]. Bayes net-works is capable of readily handling incomplete datasets and learning about causal relations. Bayes belief net-works are of high efficiency to model cases in which information concerning the previous and/or the current case is not clear, not complete, conflicting, and not certain, while rule-based models produce insufficient or inaccurate predictions when the data is not certain or not available. The 'Bayes belief net-work' was first introduced by [84]. In a Bayes Network graphic model every one of the nodes represents an arbitrary variable, and the oriented edges of the graph denote conditional dependence assumptions. Therefore, they give a compact illustration of joint possibility distributions.

The possibility of joint events may be identified using the following equation:

$$P(X_1, X_2) = P(X_1) \cdot P((X_2|X_1)$$

$P(X_1)$ denotes the possibility that event1 is true, $P((X_2|X_1)$ denotes the marginal possibility that event2 is true taking under consideration the case in which event1 is true as well, and $P(X_1, X_2)$ denotes the possibility that each of the events happens. The Bayes Net-work diagram is constructed for the sake of showing the marginal and joint events possibilities.

2.3.3 Naive Bayesian

This classifier is an efficient probabilistic approach utilizing class information from training samples for the prediction of the class of future examples, so this is a form of Bayesian Networks, in which the conditional attribute independence (except for the class attribute) is assumed. This method was first presented by [85] and it’s b in its better concerning the speed of learning at the same time it maintains the accuracy in predictive power. Studies on real-world data have often proved that the Naive Bayesian classifiers are of better performance comparably to more sophisticated induction approaches. The study in [85] showed that the use of a kernel density estimation rather than the Gauss distribution, the Naive Bayesian classifier is also of equally efficient performance and in some situations more efficient than the decision tree method C 4.5. The study in [86] showed that Bayes classifiers are equally accurate when compared with rule induction approaches like the CN2 and ID3 methods in medical areas. Hence, this technique is known as (Naive) due to the fact that it naively presumes independence of the features given the class. Classification is afterwards done via applying Bayesian rule for computing the possibility of the proper class taking under consideration the specific features of the credit card transaction:

$$P(C_i|X) = P(X|C_i) \cdot P(C_i)/P(X)$$

We can apply this equation with our study as follow [51]:

$$P(Fraud|Proof) = P(Proof|Fraud) \cdot P(Fraud)/P(Proof)$$

Where $P(Fraud|Proof)$ denotes the posterior possibility, the possibility of the hypothesis (i.e. the transaction being a fraud) post taking under consideration the influence of the proof (i.e. the attribute values according to training samples). $P(Fraud)$ Represents the a-priori possibility, the possibility of the hypothesis having only previous experiences at the same time ignoring any attribute value. $P(Proof|Fraud)$ is famous as the likelihood. Which is the possibility of the proof knowing that the hypothesis is actually a fraud and that previous experiences are correct. The likelihood, $P(Proof|Fraud)$ is computed using the following equation:

$$P(Proof|Fraud) = \prod_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{n}} P(Proof_K|Fraud)$$

Where n represents how many attributes there are in the data-set.

The aim of classification is the correct prediction of the value of a specific discrete class variable considering a vector of predictors or attributes [87]. On the other hand, the Naive Bayesian classifier is a Bayes net-work in which the class does not have parents and every one of the attributes has the class as its only parent.

2.3.4 Support Vector Machines (SVM)

SVM are statistic learning approaches [88] which were found to be of a high rate of success in different classification job. A number of distinct properties of those methods make them specifically proper for binary classification tasks such fraud detection. 'Support vector machines' are linear classifiers working in a "high-dimensional" property space which is a non-linear representation of the input space of the task at hand. It has been initially presented by [89], this method detects a specific type of linear structure, which is the **maximum margin hyperplane**, and it performs a classification of each training instance correctly via the separation of those instances into correct classes using a hyper-plane (i.e. a linear model). The instances which are the nearest to the maximal margin hyper-plane, the ones with the smallest distance to it – are known as **support vectors**. There's always a minimum of a single support vector for every one of the classes, and typically there are even more [52]. The maximum margin hyperplane is the one giving the optimal separation between the classes, it never comes nearer to any class than it should.

The optimal hyperplane is found by maximizing the width of the 'margin'. As shown in Fig (2.4), the margin is the distance between the separating hyper-plane and the nearest positive class and negative class.

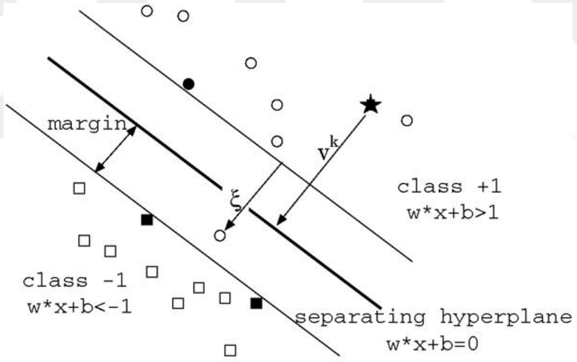


Figure 2. 4: SVM algorithm (separating hyperplane)

Source: (Edda Leopold and Jörg Kindermann**)**

In situations that the classes are not perfectly separable, The support vector machine approach has the aim of maximizing the margin that surrounds a hyper-plane separating a positive class (denoted by circles) from a negative one (denoted by squares). While minimizing the misclassified instances using a slack variable. The slack variable, (ξ) denotes the distance of the mis-classified instance from its margin hyper-plane, as depicted in Fig. 2.4, the support vector machines method performs a minimization of the summation of the distances of the slack variables from the margin hyper-planes at the same time increasing the width of the 'margin'. This is performed via solving the following formula with the use of the Quadratic Programming:

$$\text{Minimize: } \frac{1}{2}ww + C\sum_{i=1}^{l}\xi_i$$

$$\text{Subject to: } \forall_{i=1}^{l}: y_i(w \cdot x_i + b) \geq 1 - \xi_i$$

$$\forall_{i=1}^{l}: \xi_i \geq 0$$

Where ξ represents the slack variable representing the outliers, C represents a parameter allowing the selection of the complexity of the model and w,b are parameters learned with the use of the training data. The larger the value of C is the less training errors are acceptable and the more complicated the predictive model turns.

There are cases when a non-linear area are capable of separating the classes more effectively. Instead of fitting non-linear curves to the data, support vector machines determines a dividing line by using simple trick named is a **kernel function** to map the data into a varying space in which a hyper-plane may be utilized in order to make a linear separation. The idea of the kernel mapping function is quite powerful due to the fact that it permits support vector machines models for performing separations even with highly complicated edges. An infinite number of kernel mapping functions may be utilized like the, **linear kernel (L)**, **second and third order polynomial kernel (P(d))**, **Gaussian Radial Basis Function kernel (RBF)** and **Sigmoidal kernel (S)**, but the (RBF) was discovered to work efficiently for many applications such as credit card fraud [90]. The transforming to a high-dimension space is performed via the replacement of each dot product in the support vector machines method with the Gauss radial basis function kernel, the equations for kernel functions as follows:

Linear Kernel:

$$K(x_i, x_j) = x_i \cdot x_j$$

Second and third order polynomial kernel:

$$K(x_i, x_j) = (x_i \cdot x_j)^d$$

Sigmoidal kernel:

$$K(x_i, x_j) = \tanh(x_i \cdot x_j)$$

Gauss Radial Basis Function kernel:

$$K(x_i, x_j) = exp(-\gamma \; \|x_i - x_j\|^2) \,, \gamma \geq 0$$

$$K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j)$$

Where $K(x_i, x_j)$, represents the kernel function and *φ(x)* is the transform function.

2.3.5 Neural Networks

Artificial neural network (ANN) can be defined as a mathematic representing of data processing in biological NNs of the human body [91]. NNs are of a couple of main kinds: single-layer perceptron and multi-layer perceptron (MLP). Single layer ones is a linear discriminant and it's limited in mapping the property space and those are the first generation of artificial neural networks. This mathematic model is made up of inter-connected artificial neurons (i.e. nodes) which are capable of receiving a minimum of one input and sums them for producing a prediction (i.e. output). A neuron has a couple of procedure modes: training, and usage modes. In the usage one, in the case where a taught input pattern is found via the neuron its associated predictions is outputted while in training mode, the neuron may be taught for the association a specific prediction with an input pattern.

The effectively of every one of the input contributions to the ultimate prediction depends on the weight of the specific input. The most commonly utilized approach for the determination of the best connection weights is known as **back-propagation**. Back-propagation uses a mathematic method known as gradient descent that adjusts a function's parameters in an iterative way for the sake of minimizing the squared error function of the network's output. In the case where the function has numerous minima the gradient descent approach might not detect the optimal one. For the determination of an NN that is a precise predictor, proper weights for the connections have to be selected. The NN approach has been presented by [92] and throughout their work ANN study gained recognition in machine learning. Sigmoid function was utilized for the calculation of the output of every net-work layer and is represented below:

$$f(x) = \frac{1}{1 + e^{-x}}$$

The squared error function is defined as below:

$$E = \frac{1}{2}(y - f(x))^2$$

Where $f(x)$ represents the net-work’s predicting derived from the output unit and (y) represents the instance’s class label. Simple instance of an ('NN') is depicted in Fig (2.5).

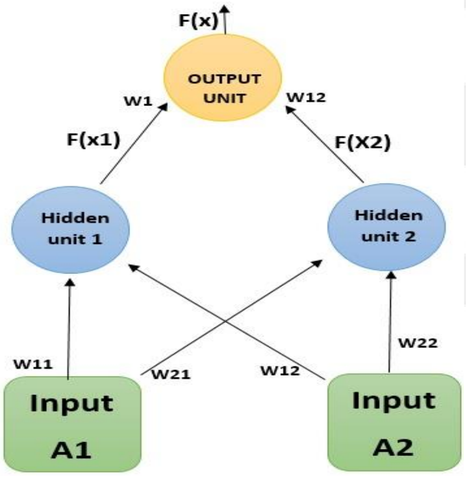


Figure 2. 5: NN with a single hidden layer

To find the weights minimizing the error function, it is needed to deriving the squared error function output according to every weight [51].

$$\frac{dE}{dw_i} = (y - f(x)) \cdot f'(x)a_i$$

Where (x) the weighted summation of the inputs, (w_i) are the weights for the ith input variable and (a_i) are the inputs to the NN. This calculation is performed for every one of the training instances and the changes that are in association with a specific weight (w_i) are added up, multiplied by the learning rate (which is a small constant value) and then subtracted from (w_i)'s current value. This is performed until the variations of the weights become very little.

2.3.6 K-Nearest Neighbors

This approach is a simple algorithm that saves every available instance and classifies new cases according to a similarity measure. This algorithm is an instance based learning which carries out its classification based on a similarity measure, like **Manhattan**, **Euclidean** and **Minkowski** distance functions, equations for these distance as below:

- The Manhattan Distance is the distance between a couples of points which is measured along axes at 90 degrees angles.

$$d(x,y) = \sum_{i=0}^{n-1} |x_i - y_i|$$

- The Euclidean distance can be defined as the straight line distance between a couple of points in Euclidean space:

$$d(x,y) = \sqrt{\sum_{i=0}^{n-1} |x_i - y_i|^2}$$

- The Minkowski distance is defined as the distance between a couple of points in a normalized vector space:

$$d(x,y) = \left(\sum_{i=0}^{n-1} |x_i - y_i|^p\right)^{1/p}$$

The KNN method directly performs a search through every training example via computing the distances between the test sample and the entire the training data for the sake of identifying its nearest neighbors and generate the classification result [93]. This technique is a sample of an instance-based learner, can see how work this algorithm in fig (2.6). In another meaning, all of the other learning methods are also instance-based, due to the fact that they start with a group of examples as the initial training data. Hence, for instance-based learners the instances are utilized for representing everything learned, instead of utilizing the instances for inferring a rule group or decision tree.

The classification approach of the **nearest-neighbor** is when every one of the new instances is compared against existing instances with the use of a distance measure, and the nearest available instance is utilized for assigning the class to the new one.

Typically, over one nearest neighbor is utilized, and the majority class of the nearest K neighbors (or the distance weighted average, in the case where the class is of numerical values) is given to the new instance. The idea of the instance based KNN method was first presented by [94].

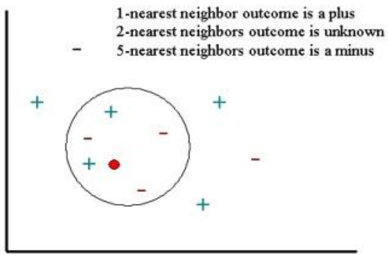

1-nearest neighbor outcome is a plus
2-nearest neighbors outcome is unknown
5-nearest neighbors outcome is a minus

Figure 2. 6: KNN algorithm (instances classification)

Source: (Statistica**)**

The most widely used of the distance functions is the Euclidean distance. Therefore, this gives the assumption that the features are normalized and are equally important (determine the important features is one of the most considerable matters in the process of learning). For instances in which nominal attributes are present, such as comparing the values of the attribute of the types of credit cards, which are: Gold and Platinum, Classic, a distance equal to 0 is assigned if the values are identical, otherwise, the assigned distance is equal to 1. Thus the distance between platinum and platinum equals 0 but the distance between platinum and gold is equal to 1.

There are important for some attributes than others, and this is sometimes seen in the distance measurement by some type of attribute weighing. The derivation of relevant attribute weights from the training group is one of the main problems in instance-based type of learning. In this method the instances don't actually give a description of the patterns in data. However, the instances combined with distance measures for carving out boundaries in instance space which distribution a class from the other, which is a type of direct knowledge representation.

2.3.7 Logistic Regression

Logistic Regression which uses a functional approach for the estimation of the possibility of a binary response according to one or more variables (features). It is often used when the dependent variable takes only two values and the independent variables are continuous, categorical, or both. Logistic regression technique is ideal when classifying outcomes that only have two values because the logistic curve is limited to values between 1 and 0. The technique utilized in this thesis is based on the work done by [95]. It finds the best fit parameters to a nonlinear function called the sigmoid. The function performs mapping of any real value to another value in the range (0-1). In machine learning, used sigmoid for mapping predictions to probabilities, as shown in the equations below:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Where $\sigma(z)$ the output between '0' and '1' ('probability estimate'), (z) is the input to the function and (e) is the base of natural log.

$$z = w_0 z_0 + w_1 z_1 + \cdots + w_n z_n$$

The vector (z) is input data and the best coefficients are (w) the multiplied together multiply each element and adds up to get one number which determines the classifier classification of the target class.

In the credit card fraud detection field the dependent variable would take on a value of 1 (fraudulent transaction) or 0 (genuine transaction). Not like ordinary linear regression however, logistical regression doesn’t presume a linear relation between the dependent variable and the independent ones, nor does it presume that the dependent variable or the error terms are distributed in a normal way, the logistic regression model is defined as below:

$$\log\left(\frac{p}{1 - p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k$$

Where $X_1, X_2, \dots, X_k$ represent the independent variables and (p) is the possibility that the dependent variable has a value of (1). (β_0) is a constant value and $\beta_1, \dots, \beta_k$ are coefficients of the independent variables. The logistic regression model looks similar to the multi linear regression equation, however, the logistic regression regresses against the logit $\log\left(\frac{p}{1-p}\right)$ and not contra the dependent variable, shown the Fig (2.7).

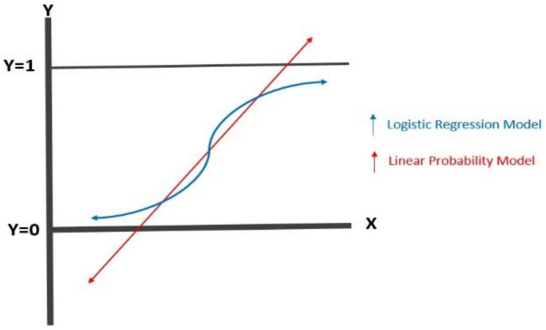


Figure 2. 7: Comparing the Logit model and linear probability model

The Maximum Likelihood Estimation (MLE) is then used to compute the beta coefficients in the logistic regression formula. The aim of 'MLE' is to find the parameter values that make the observed data most probable to be predicted. Likelihood and probability are closely related because the likelihood of the parameters given the data is equal to the possibility of the data considering the parameters [96].

Likelihood → Estimating model parameters given the observed data.

Probability → predicting an outcome given model parameters.

The 'likelihood function' is defined bellow:

$$L(a) = f(x_1; a) \cdot f(x_2; a) \cdots\cdots f(x_n; a)$$

Where $x_1, x_2, \ldots., x_n$ are the observed values of a dataset, (a) is a single unknown parameters and $f(x; a)$ is the probability distribution function, the MLE algorithm initially chooses arbitrary numbers for the 'likelihood function' is maximized. Using the beta parameters calculated by the Maximum Likelihood Estimation method and the corresponding values of the independent variables, the expected probability for a fraudulent transaction can be calculated.

There are advantages and disadvantages with applying certain algorithms to fraud detection. However, a metric is needed to determine the ideal algorithms to use in the credit card fraud field. A "diversity" value was selected as a metric to determine the optimal algorithms because it is easily calculated and the numerical score output can assist in ranking the best base algorithms to use as base classifiers.

2.3.8 Decision Tree (C4.5)

It can be defined as a 'tree structure' which seeking to split the given records into mutually exclusive 'sub-groups' [13]. This technique is rule based classifiers which utilizes a “divide and conquer” approach for the construction of a prediction rule. This approach operates via a recursive breaking down of the problem to two or even more sub-problems to the point where it’s sufficiently simple to be solved instantly. Decision trees are graphic representations of the “if-then statements” (i.e. decision rules), the decision tree method that has been utilized in this study (C4.5), has been first presented by [97]. This approach is made up of branches and nodes. The first node typically goes by the name “root node”. Every one of those node has a labeled containing a property name and every branch leading out of it is labeled with one or more possible values for that feature. Every one of the internal nodes in the tree is corresponding to test of the value of one of the properties. Every one of the nodes has merely one incoming branch, except the root, and that is designated as the start point. The node has various labels for Branches that have the potential test values. Leaves are labeled using the values of the classification properties and decide the value to be returned in the case where that leaf is reached. Via taking a group of properties and their associated values as input, a decision tree is capable of classifying a case via decision tree traversing. According to whether the result of a test is false or true, the tree branches to one of the nodes. The property of the instance that corresponds to the root label is compared against the values on the outgoing branches of the root, and the matching branch is chosen.

Node's label matching and the procedure of selecting a branch keeps going till a terminal node, also known as leaf has been reached, where the case is classified with respect to the leaf label and a decision is done on the case’s class assignment [97].

The decision tree (C4.5) approach is the most commonly implemented approach for building decision trees. This approach utilizes the idea of entropy for the determination of the optimal node for the tree to branch. In every one of the tree nodes (C4.5) selects a data attribute which with the most efficiency divides its group of instances to sub-sets enriched in one of the classes. Its criteria is the normalized data gain difference in entropy which is resulted from determining a feature for dividing data. The attribute that has the most optimal normalized data gain is selected for making the decision.

Entropy for a group of instances, (S), for a variable may be computed as the following equation:

$$E(S) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

Where (p_i) represents the probability of outcome, i is the outcome state and c is the number of output states.

Entropy for a couple of variables can be computed:

$$E(S, A) = \sum_{v \in A}^{c} \frac{|S_v|}{|S|} \cdot E(S_v)$$

Where (S_v) represents the size of the sub-set in state v, (S) is the size of the whole group, v the of the second variable state and A is the group of samples of the 2nd variable.

Finally the data gain is identifies using the following equation:

$$Gain(S, A) = E(S) - \sum_{v \in A}^{c} \frac{|S_v|}{|S|} \cdot E(S_v)$$

Thus, the attribute that has the biggest data amount gained would be chosen as the splitting property. The entropy of an attribute denotes the presumed amount of data which will would be required for specifying classifying new instances. The decision tree terminates as soon the data can't be split any more. In the idea, the procedures is performed to the point where each leaf node is pure that is when they include instances having identical classifications.

CHAPTER THREE

METHODOLOGY

Objectives from this chapter to describe the performance measures and examines the four data mining techniques Naïve Bayesian (NB), Support Vector Machine (SVM), K- nearest neighbor (KNN) and Random forest (RF) for credit card fraud detection to give the right advice for banks of which the best technique they can build their system, this chapter included different sections, as shown in fig (3.1): first section include software used for conduct the examination on classification algorithms, second section dataset description, third section sampling dataset technique and how to use the training dataset and testing dataset, forth section explains the classification methods used in this study, fifth section performance measures that used for evaluating our comparison, sixth section explains cross-validation for estimation our models used and seventh section describe evaluation technique and AUC measures.

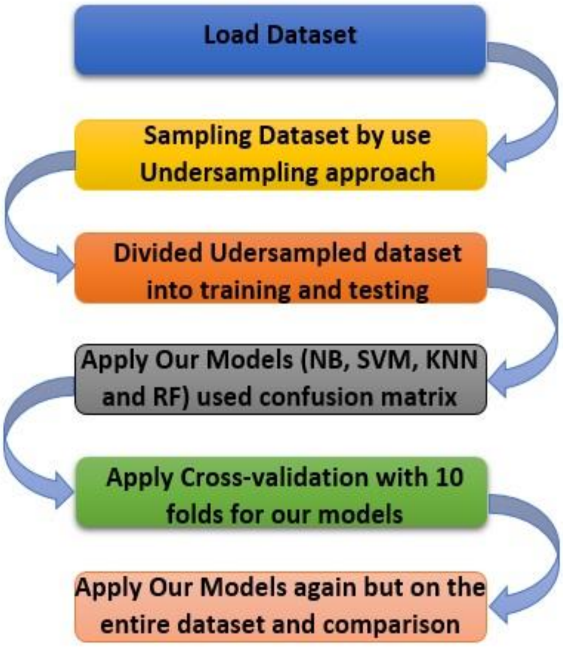

Load Dataset
Sampling Dataset by use Undersampling approach
Divided Udersampled dataset into training and testing
Apply Our Models (NB, SVM, KNN and RF) used confusion matrix
Apply Cross-validation with 10 folds for our models
Apply Our Models again but on the entire dataset and comparison

Figure 3. 1: Stages for Methodology

3.1 Software Used

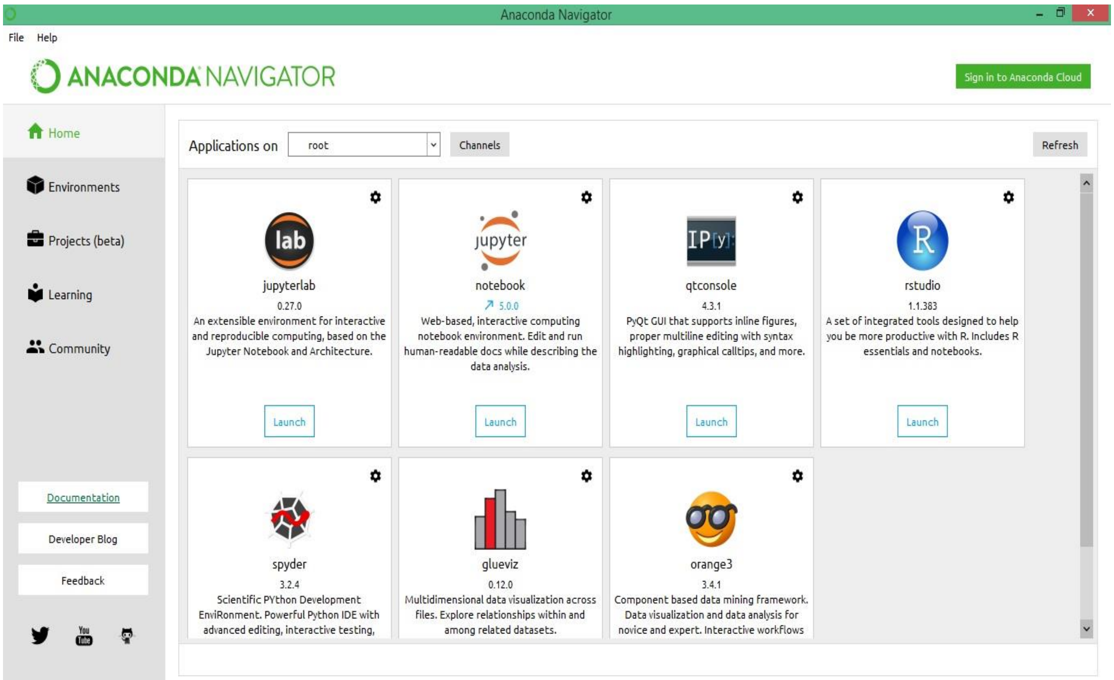
In this section presents our software used All the classification models used in this study and outputs were obtained using **Anaconda3** version **5.0.1**. Anaconda Distribution is easy and free to install package manager, environment manager and

Python distribution with a combination of more than (1,000), packages are open source with free community support, as shown in fig (3.2). Anaconda is platform agnostic, so it may be used on Windows, macOS or Linux. Anaconda Enterprise is Anaconda with world class priority enterprise support on an enterprise ready, secure and scalable data science platform empowering teams in governing data science assets, collaborate, and utilize data science projects [98].

With Anaconda Enterprise, enterprises are capable of:

- Managing and controlling versions of data science assets
- Authorizing accessing to data science projects and assets
- Accessing detailed logging of system actions for auditing
- Integrating Anaconda platform with the user's enterprise authentication
- Leveraging a security-vetted Open Data Science platform

Anaconda is considered an umbrella for all python platforms, in this study used python from notebook anaconda because it is powerful with approximate all data mining techniques and also give us the high accurate for performance measures such as (accuracy, recall, precision) .



An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing,

Multidimensional data visualization across files. Explore relationships within and among related datasets.

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows

Figure 3. 2: Interface for Anaconda Navigator

3.2 Dataset Description

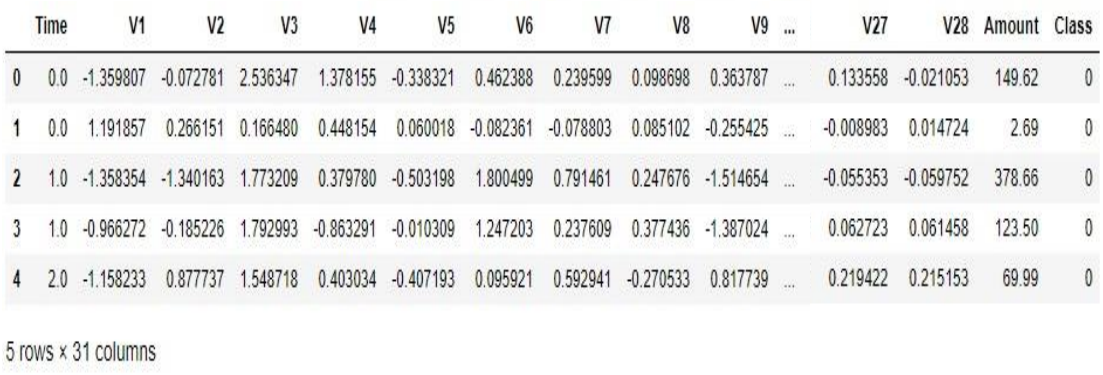
This section we will describe the provided dataset with descriptive statistics used for our study, explanation of feature variables and the distribution of fraudulent respective genuine transactions.

Highly imbalanced dataset for our study from European cardholders, this dataset is provided based on transactions from European cardholders for a two day period that have been made in September 2013 [4], so it has been published in 2016 . It was originally collected by a research collaboration of (ULB) Wordline and University Libre de Bruxelles with the aim of analyzing big data and fraudulent transactions. The total there were (284,807) transactions throughout the time span, there were (492) positive class (fraudulent) and the dependent class (fraudulent, genuine) is heavily unbalanced, Table (3.1) describes all (31) thirty-one variables in the data, where the feature variables besides Time and Amount are displayed with an unknown description due to a protection of sensitive information.

However, these are not the original variables obtained during the collection of data. They have all been transformed with principal component analysis (PCA) to protect the true information from the analyst examining the data (or other third parties that may contribute to negative consequences). In other words, V1-V28 are principal components holding the real data in some fashion. All twenty-eight (Vs) variables and Amount are categorized as numeric, while Class and Time are both integers, as shown in fig (3.3) sample of this dataset.

Table 3. 1 : Description of Dataset and Attributes

Attributes	Type	Description
Time	int	Time between each transaction
V1	num	Feature variable with unknown information
.	.	_____
.	.	_____
V28	num	Feature variable with unknown information
Amount	num	Total money spent
Class	int	Response attribute (0= genuine and 1 = Fraudulent)



	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V27	V28	Amount	Class
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	0.133558	-0.021053	149.62	0
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.008983	0.014724	2.69	0
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	-0.055353	-0.059752	378.66	0
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	0.062723	0.061458	123.50	0
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	0.219422	0.215153	69.99	0

5 rows × 31 columns

Figure 3. 3: Sample of dataset

3.3 Balancing Technique

Because of the high imbalance our dataset (skewed), as shown in fig (3.4) the original distributions of dataset between fraudulent and genuine. This section presents how to handling that skewed dataset among existing approaches and how to divided dataset into training and testing. We proposed in our study used under-sampling imbalanced class distribution.

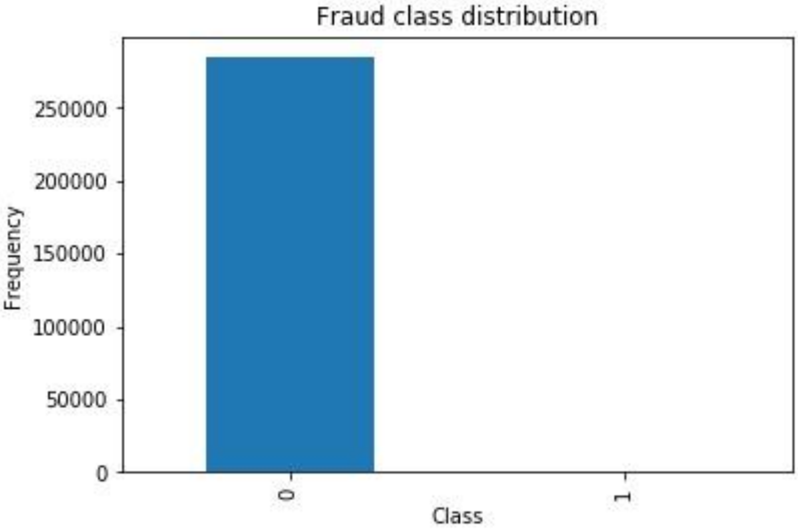


Figure 3. 4: Distribution of class (0,1)

3.3.1 Under-Sampling Approach

Under-sampling is a commonly used technique to handle imbalanced datasets to decrease the skew in class distributions [48], under-sampling used to removing observations values from majority class (genuine) randomly until the dataset reach to balanced. Standard machine learning methods which maximize general precision usually classify each observation as a majority class instance, which results in poor precision on the minority class (i.e. low recall), usually the class of interest, as we mentioned in chapter two section (2.1.2). Deterioration of the performance of classification isn’t merely associated with a proportion of high number of the majority classes in comparison to the small number of instances in the minority classes (expressed with the class imbalance ratio), but also to the minority class decomposition to small sub-clusters [99] and to the overlap between the two classes [62]. We can see that skewed distribution of dataset in fig (3.4), where showed the small percentage to class 0 (genuine) in comparison with the class 1 (fraudulent), almost make up the ratio of minority class (0.172%) percent of total transactions equivalent (492) fraud transactions from (284,807) transactions. Under-sampling work to the equal proportion among fraudulent /genuine (1:1), under-sampling is beneficial for handling the imbalanced dataset, as shown in fig (3.5) below.

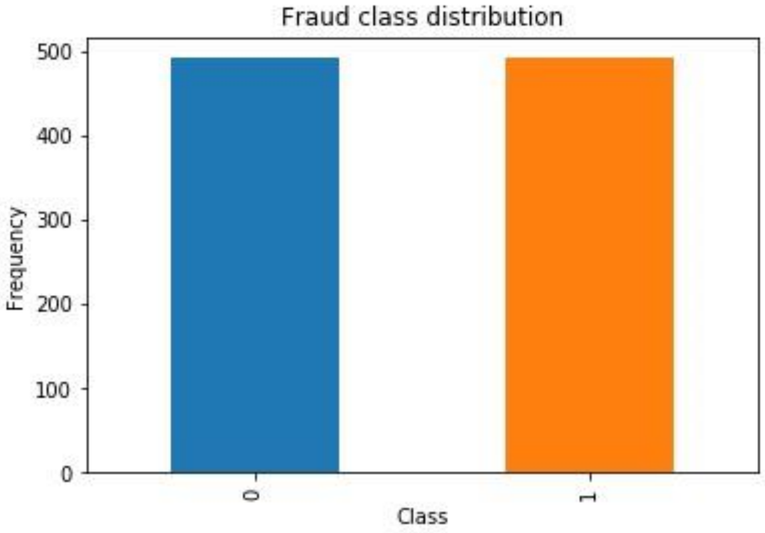


Figure 3. 5: After Apply Under-Sampling Approach

After apply our proposed approach for skewed dataset we note in fig (3.5) in above, where dataset class distributions become equal (50:50) for positive class 1 and negative class 0 . Hence, these studies don't imply that classifier models can't learn from unbalanced datasets. On the other hand, other studies have also shown that some classifier models don't enhance their efficiency when the training data-set is balanced with the use of sampling approaches [66] [59]. Therefore, for the moment the only means of knowing whether sampling is helpful for the learning procedure in running some simulations. In spite of the common used of under-sampling, it is required to remark that there’s not yet a theoretical structure that explains the way it is capable of affecting the precision of the learning procedure.

3.3.2 Training and Testing Dataset

The main function of this module is to build Train Model which receives in input some supervised transactions (feedbacks or delayed samples) and returns a predictive model built using the our models and we have to perform a training of the classifier with the use of a training set, tuning the parameters with the use of validation group and afterwards testing the efficiency of the classifier on unobserved test group, a significant point to be noted is that throughout the process of training the classifier merely the training and/or validating set is available. The testing set has to be unused throughout the process of classifier training. The testing group will only exist throughout the process of classifier testing, as we shown in fig (3.6) below. There is no one way of choosing the size of training and testing set and people apply heuristics such as 20% testing and 80% training.

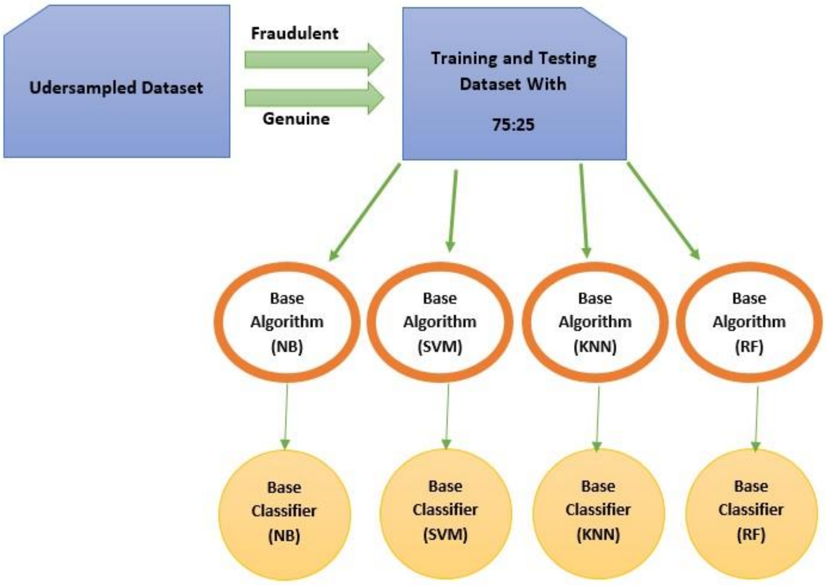


Figure 3. 6: Training and Testing using 75% respective 25%

In this study we present our proposed to divide the dataset after undersampled dataset into two groups, too, where the training group will be (75%) of dataset and the testing group will be (25%) of dataset, well as used the same numbers of split entire dataset. As we shown in above fig (3.6).

3.4 Classification Algorithms

The process of classification happens in a many people's activities. Generally, the term could relate to any of the contexts where a decision or forecast is performed on the base of currently existing data. The classification process is applied for repeatedly making decisions like those in new cases, such as a problems are often referred to as classification problems. Constructing a classification process from a collection of data for which the true classes are known was termed as "pattern recognition", "discrimination", and "supervised learning" as well. Statistically, the classification issue is often known as the prediction issue and in the area of machine learning it is typically known as concept learning [47].

In this section we present the algorithms used in our study as we mentioned previously, these are 'Naïve Bayesian (NB)', 'Support Vector Machine (SVM)', 'K-nearest neighbor(KNN)' and 'Random forest(RF)', respectively.

3.4.1 Naive Bayesian Classifier

This classifier is a highly efficient probabilistic approach for supervised classification as well as a statistical method used class data from training examples for the prediction of the future fraud class, classification performed via implementing Bayesian rule for the calculation of the possibility of the correct class given the specific features of the credit card transactions [51], we used in our study Gaussian Naive Bayes, this model extended real-valued attributes, Gaussian distribution is the easiest and merely require the estimation of the mean and the standard deviation from the training data, following the equation of Gaussian Naive Bayes [100].

$$P(c_i|f) = \frac{1}{\sqrt{2\pi\sigma_f^2}} exp\left(-\frac{(c_i-\mu_f)^2}{2\sigma_f^2}\right) \text{-------} (3.1)$$

Where i indicator either 0 for genuine transactions or 1 for fraudulent transactions from training data, this two values refers to the our classification problem is binary as we mentioned, is a probability of feature value f being in class c_i, the μ_f and σ^2 are a mean and standard deviation calculating values of every one of the input variables (c_i) for each class value.

If $P(c_1|f) > P(c_i|f)$ then the classification is $C1$

If $P(c_1|f) < P(c_i|f)$then the classification is $C2$

The class C_i is a target or predicted class for classification where C_1 is the negative class (genuine) and $C2$ is the positive class (fraudulent).

Steps of Gaussian Naïve Bayesian Algorithm:

Step1: Calculate the probabilities for each input values (x), this means the training vectors.

Step2: Calculate the mean value of every one of the input variables (x) for every class value according to equation below:

$$\mu(x) = 1/n * sum(x) \text{ ------ } (3.2)$$

Where n represents the number of instances and x are the values for an input variable in our training information.

Step3: compute the value of the standard deviation of every one of the input variables (x) for every one of the class values according to equation below:

$$Std\ (x) = sqrt(1/n * sum(x_i - mean(x)\verb|^|2\ )) \text{ ------ } (3.3)$$

Where n represents the number of examples, $sqrt()$ means the square root function, $sum()$ means the summation function, x_i is a certain value of the x variable for the $i'th$ instance and $mean(x)$ are described above, and ^2 is the square.

Step4: maintain the standard deviation and the mean values for each one of the input variables (x) for each class.

Step5: Fit Gauss Naïve Bayes according to (x, y) according to the plug in the possibilities to the abovementioned equation for making anticipations with real-valued inputs.

Step6: Target values of (y) predict.

3.4.2 Support Vector Machine (SVM)

This a supervised and statistical learning approach which has been used for a variety of classification problem successfully, suitable for binary classification problems as a credit card fraud detection, support vector machines are linear classifiers which operate in a high dimensional property space which is a nonlinear mapping of the input space of the present problem [10], SVM is make of solving non-linear classification problems, advantages for the support vector machines is a result of two significant features, they possess kernel representation and margin optimization, where the kernel function is the trick used for convert the nonlinear problem to the linear problem even we can extract optimal solutions for our problem, after then we can deal with problem to find the (hyper-plan) with maximum separation margin between both classes to avoid any risk for overfitting the training instances, there are three functions to transform the 'nonlinear' to 'linear' classification, namely, 'polynomial function', 'radial basis function (Gaussians)' and 'sigmoid (neural net activation function)'. We used in our study radial basis function (RBF) due to our dataset is nonlinear and it work with wide variety problems like credit card problems, following the equation for Gaussians RBF:

$$K(x_i, x_j) = exp(-\gamma \, \| x_i - x_j \|^2) \, , \gamma \geq 0 \text{ ------ } (3.4)$$

$$K(x_i, x_j) = \ \varphi(x_i) \cdot \varphi(x_j) \text{ ------ } (3.5)$$

Where φ: $x \rightarrow H$ is a trick to transform the input space x into a higher dimensional space H, the $K(x_i, x_j)$ is the kernel function used and $\varphi(x_i)$ is the transformation function [51].

Steps of Support Vector Machine Algorithm:

Step1: Read the dataset given.

Step2: Re-order the data in two groups as transaction class and time of transactions and difference between successive transactions.

Step3: Each transaction is making in the form of data as vector of two area.

Step4: Then make two distinct data sets referred to as positive and negative transaction groups.

Step5: Select Gaussian (RBF) of three kernels, as we mentioned above.

Step6: Train the 'SVM'.

Step7: After apply we save the performance of classifier.

Step8: Then we read the current 'transaction'.

Step9: Restart the operation from steps 1 to 3 only for current transaction data.

Step10: Replaced the saved classifier and currently produced vector in classifier.

Step11: Admit the produced decision from the classifier.

3.4.3 K-Nearest Neighbor

This algorithm is a strong and largely used in detection systems, the KNN classifier used very well in credit card fraud detection problem, that is always used as a benchmark for more complex classifiers such the 'Artificial Neural Networks(ANN)' and 'Support Vector Machines(SVM)' [42]. The KNN is a supervised learning method, in this technique the new instance query will be classified depending on the well-known KNN distance measures such as Manhattan distance, Euclidean distance and Minkowski distance. In our study we used the Euclidean distance between two instances (transactions), where each incoming transaction will be computed of its nearest point to new incoming transaction to detect fraud, following its formulation given by [21]:

$$D(x_i, x_j) = \sqrt{\sum_{k=1}^{n}(x_{ik} - x_{jk})^2} \qquad k = 1,2,\dots\dots,n \text{ ------ } (3.6)$$

The KNN algorithm basic boils are going to form a majority vote between the K most similar instances to a given “predicted” observation, the Euclidean distance between two data point are new input data point with current data point are computed, the distances that computed are sorted and arranged incrementally and select the lowest distances with k-items to the input data point, the binary classification for our study means the negative class among these items is found and the KNN classifier returns the positive class such the classification for input data point. Parameters chosen for k neighbors start from k= [1, 2, 3, 4, 5, 6, 7, …., 12] the our classifier returns the k=5 the best parameter for accuracy performance, so that parameter used in the our classifier.

Steps of 'K-Nearest Neighbor' Algorithm:

Step1: Open ('CSV') file and split the dataset into test and train datasets to handle it.

Step2: Set parameter of K which is the number of nearest neighbors, in our study supposed k=5.

Step3: Counting the distance between 'two' data instances and each training sample by using 'Euclidean distance'.

Step4: Sort the distances and choose the nearest neighbors according to the k minimal distance.

Step5: Generate a response from a set of data instances.

Step6: Deploy simple majority of the category of nearest neighbors as the prediction to output either positive= fraudulent or negative= genuine.

Step7: Conclusion the performance such the 'accuracy' of predictions.

3.4.4 Random Forest

Random forest (RF) is an aggregate of decision trees models or a combination of tree predictors [82], used average to improve the predictive accuracy and control overfitting, due to (RF) method is supervised so it trained each sub sample (tree) of the original training set input on different bootstrap and the size of sub sample same the original input data, after then using a random sub sample of all the features available, this returns a forest of decision trees that are very different from each other [65], every one of the trees in an aggregate is produced from an arbitrary subsample of features, because of many studies recommended used this technique among different data mining techniques its performance where achieved best accuracy. In our study every one of the trees in the set is constructed from a sample drawn with replacement (i.e. bootstrap sample) from the training group, as shown in above fig (3.7).

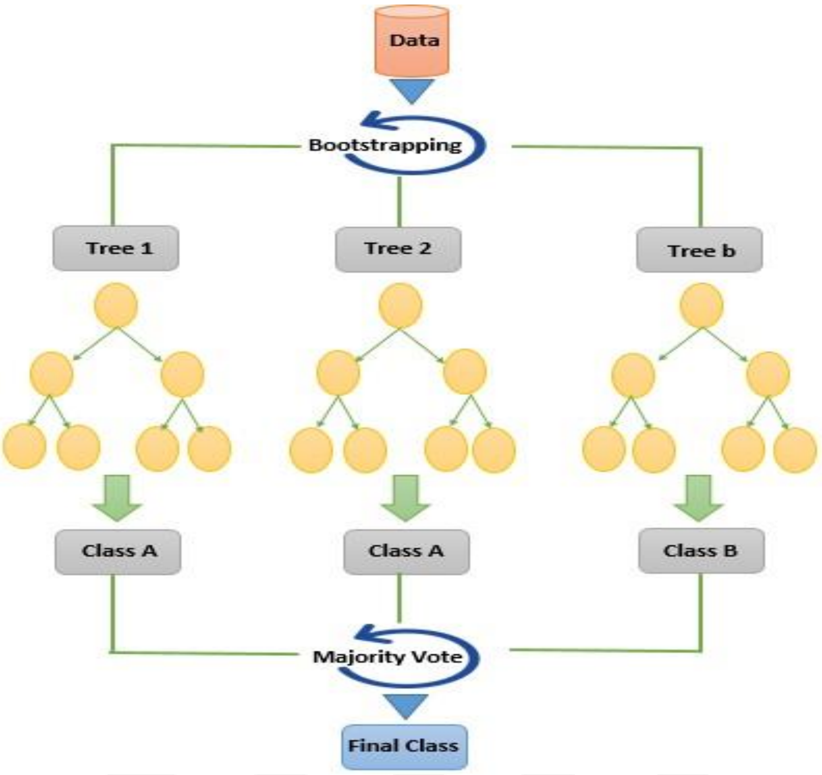


Figure 3. 7: Random Forest Algorithm

The essential parameters used in our study **estimator** parameter is the number of trees in the forest, where used random estimators start from E= [1, 10, 100, 1000] were the best accuracy we got it from E= 100, **criterion** parameter is the function to measure the quality of a split, selected from that parameter **'Gini'** impurity is the best for improvement performance, and **max features** parameter is the number of properties to take under consideration when looking for the optimal split, selected from that parameter **'auto'** is the best for performance, these parameters are the most important for our study. The RF is better technique in terms of performance for detecting fraud among four techniques we used in our study.

Steps of Random Forest Algorithm:

Step1: Open the dataset from (CSV) file and split into test and train datasets.

Step2: Arbitrarily choose (k) properties from overall (m) properties, where the k<m.

Step3: The 'node(d)' is calculated among the 'features(k)', with the use of the optimal split dot**.**

Step4: Split the node to daughter nodes with the use of the optimal split.

Step5: Redo step1 to step3 until (l) number of nodes has been produced.

Step6: A forest is built via repeating 'step1' to 'step4' for (n) number period for creating (n) number of trees.

Step7: Takes the test properties and utilize the rules of every one of the randomly produced decision trees for predicting the outcome and stores the predicted result (i.e. the target).

Step8: Compute the votes for every one of the predicted targets.

Step9: Assume the high voted expected target as the ultimate prediction from the random forest approach.

3.5 Cross-Validation

In this section we will present how work the cross validation and what type of fold used with our study. Considered one of the approaches for obtaining more credible estimates of the predictive accuracy of the classifiers is 'n-fold cross validation(CV)'. Researcher use cross validation, also referred to as rotation estimation, it is a model validation approach used to assess the way the results of a statistic analysis generalizes to an independent dataset, for this purpose, the program splits the data into a number of folds (splits) equal to a chosen number [47], if we suppose that a number of folds (n) is set. The data-set is arbitrarily rearranged and after that split to n folds of identical sizes. In every one of the iterations, a single fold is utilized to test and the rest of the n-1 folds are utilized to train the classifier. Experience on a large number of datasets has shown that the number of folds equal to 10 has achieved good results [101]. In our study used 10 fold cross validation to estimate the precision of the classifiers.

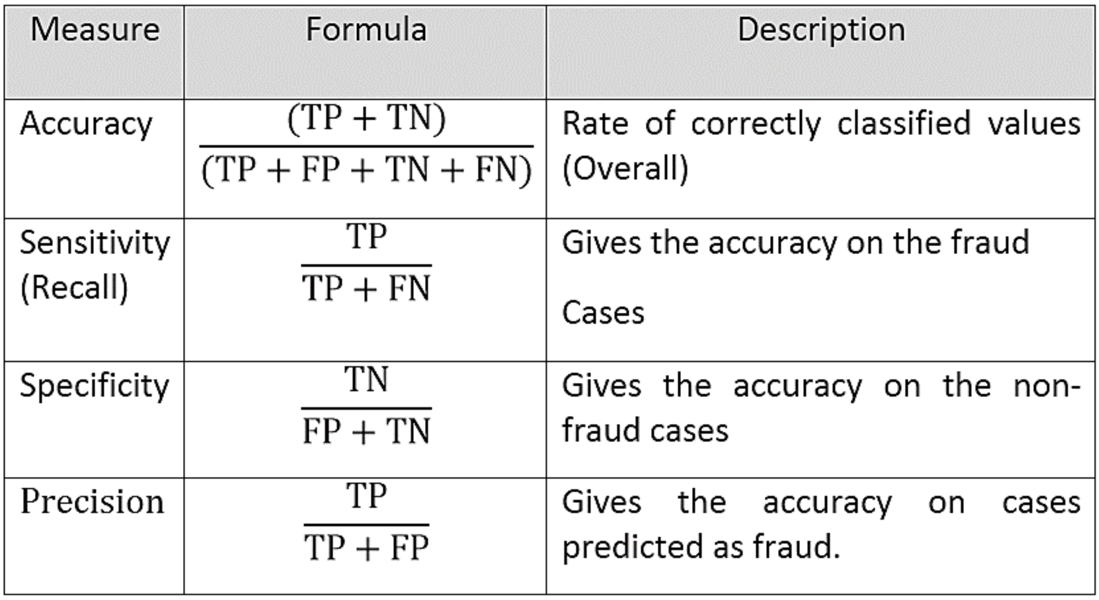
3.6 Evaluation Performance Measures

The conventional approaches of classification efficiency, do not necessarily adequately address performance requirements of certain applications. In fraud detection, cases that are expected as possible fraud are checked to be investigated or some other action involving a cost. In this section we present some important measures for testing our models.

3.6.1 Basic Measures

We will use in our study four well known measures to evaluate the methodology are accuracy, sensitivity, specificity and precision, these measures depend entirely on the four basic metrics (alarm rates), respectively true positive (TP) the number of fraudulent transactions which detected alarm true, False Positive (FP) the number of genuine transactions which detected false alarm, True Negative (TN) the number of genuine transactions are detected true alarm and False Negative (FN) the number of missed fraudulent transactions [7], positives (P) mean the number of fraudulent transactions and negatives (N) the number of genuine transactions the total of P and N that means all transactions. In the Table (3.2) below equations for each measure we used.

Table 3. 2 : Description of the evaluation formulas



Measure	Formula	Description
Accuracy	$\frac{(\mathrm{TP}+\mathrm{TN})}{(\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN})}$	Rate of correctly classified values (Overall)
Sensitivity (Recall)	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	Gives the accuracy on the fraud Cases
Specificity	$\frac{\mathrm{TN}}{\mathrm{FP}+\mathrm{TN}}$	Gives the accuracy on the non-fraud cases
Precision	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$	Gives the accuracy on cases predicted as fraud.

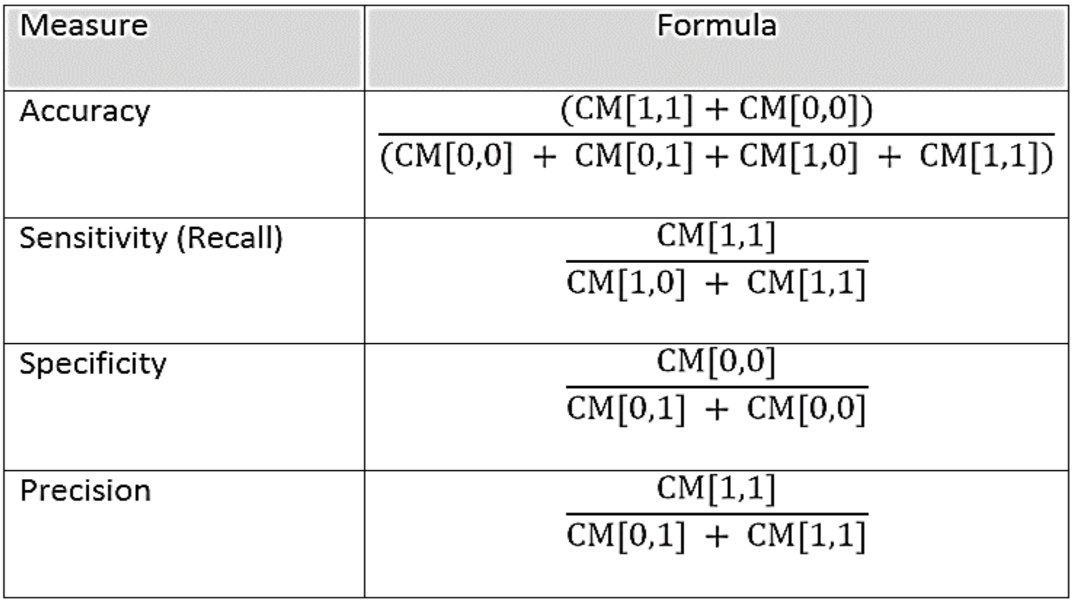
Evaluation of the classification performance needs to know what meaning of each measure, where accuracy means the proportion of true alarm rates among all alarm rates, sensitivity (recall) the proportion of the true positives, that indicates the fact that the number of fraudulent are detected correctly, specificity the proportion of true negatives, that indicates the fact that the number of genuine transactions are detected correctly too and precision measures the proportion of true positive among all positives alarms.

3.6.2 Confusion Matrix

A 'confusion matrix (error matrix)' consists of 'two rows' and 'two columns' that together represents 'true positives', 'true negatives', 'false positives' and 'false negatives' [102]. First, all values found in the true positive cell are predicted outcomes matching the actual values in the dataset. In a data transactions, this would denote a prediction that classifies as genuine when the real value is also genuine. Second, the true negatives are the exact vice versa, a predicted respectively an actual value is both labeled as fraudulent. Third, the false positives are recognized as (type two errors) that stands for outcomes classified as fraudulent but they are actually not. Finally, the false negatives are (type two errors) and represents transactions (fraudulent) that are predicted as genuine.

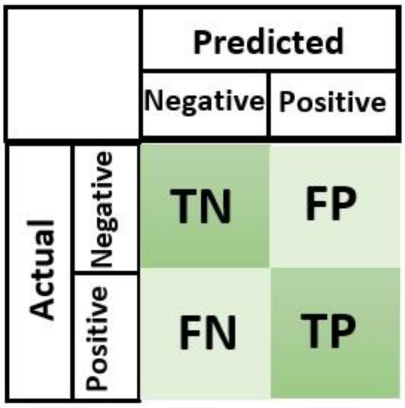
To summarize the confusion matrix specifications, one may see the off diagonal values as misclassifications and the diagonal as accurately classified outcomes. In our study use confusion matrix, because the measures in above mentioned can be computed only once a confusion matrix is available. However, Table (3.3) provides a visible illustration of the evaluation performance measures with confusion matrix. As well as the confusion matrix used twice first with udersampled dataset, second with entire dataset (skewed) and comparison of them.

Table 3. 3 : Evaluation formulas with confusion matrix



Measure	Formula
Accuracy	$\frac{(CM[1,1] + CM[0,0])}{(CM[0,0] + CM[0,1] + CM[1,0] + CM[1,1])}$
Sensitivity (Recall)	$\frac{CM[1,1]}{CM[1,0] + CM[1,1]}$
Specificity	$\frac{CM[0,0]}{CM[0,1] + CM[0,0]}$
Precision	$\frac{CM[1,1]}{CM[0,1] + CM[1,1]}$

Where different values of basic metric when we use confusion matrix due to each value will take its index in matrix, that means the (TP= $\mathrm{CM}[1,1]$), (FP= $\mathrm{CM}[0,1]$), (TN= $\mathrm{CM}[0,0]$) and (FN= $\mathrm{CM}[1,0]$). In the fig (3.8) below example about confusion matrix.



		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

Figure 3. 8: Confusion matrix

3.6.3 Area Under ROC Curve (AUC)

In our study used important type of measure is area under ROC curve (AUC), we used in undersampled dataset and entire dataset. AUC is better than accuracy measure for evaluating learning algorithms [103], AUC tested on positive class (fraudulent) FP and TP. The comparison is always done by computing the 'Area Under the ROC Curve' (AUC). AUC is also a well-accepted measure for unbalanced datasets and it has become the de facto standard in classification [104], the area under the curve (AUC) is defined by equations as below [51]:

$$AUC = \frac{U_1}{n_1 n_2} \text{------} (3.7)$$

Where (U_1) denotes the (U) value that has been computed with the use of sample 1, (n_1) denotes the size of sample 1, and (n_2) is the size of sample 2 (sample 2 are the examples not selected to be in sample 1).

CHAPTER FOUR

RESULTS AND ANALYSIS

The main results of this study are presented in this chapter. Our study performed in two ways: first, the undersampling, and second, on the entire dataset, as mentioned in Chapter Three. In Section 4.1, we explain how to apply the preprocessing approaches and distribution data. In Section 4.2, we present the evaluation performance of our proposed classifiers (NB, SVM, KNN and RF) as mentioned in the previous chapters with the confusion matrix. So in Section 4.3, we present the comparative models to determine which techniques performed better, and we also compare all four proposed classifiers by using the new measure (AUC) as well as the results of the classifiers and every comparison in terms of their accuracy, sensitivity (recall), specificity, precision and AUC.

4.1 Data Distribution

In the preceding chapter, we discussed our dataset in addition to how it performs preprocessing, as shown above in Figures (3.4) and (3.5) by using the under-sampling approach, followed by dividing the dataset into 75% training and 25% testing. In this section, we present a determination of whether there is any correlation in the variables in the dataset, as shown in Figure (4.1). We note and observe in this figure that the dataset is uncorrelated. However, we can move forward with our analysis as mentioned previously. We check for missing values and do some exploratory data analysis to determine whether any of the transactions are fraudulent or genuine, as shown in Figure (4.2). In this figure, we observe that the proportion of fraudulent transactions is very low compared to those that are genuine.

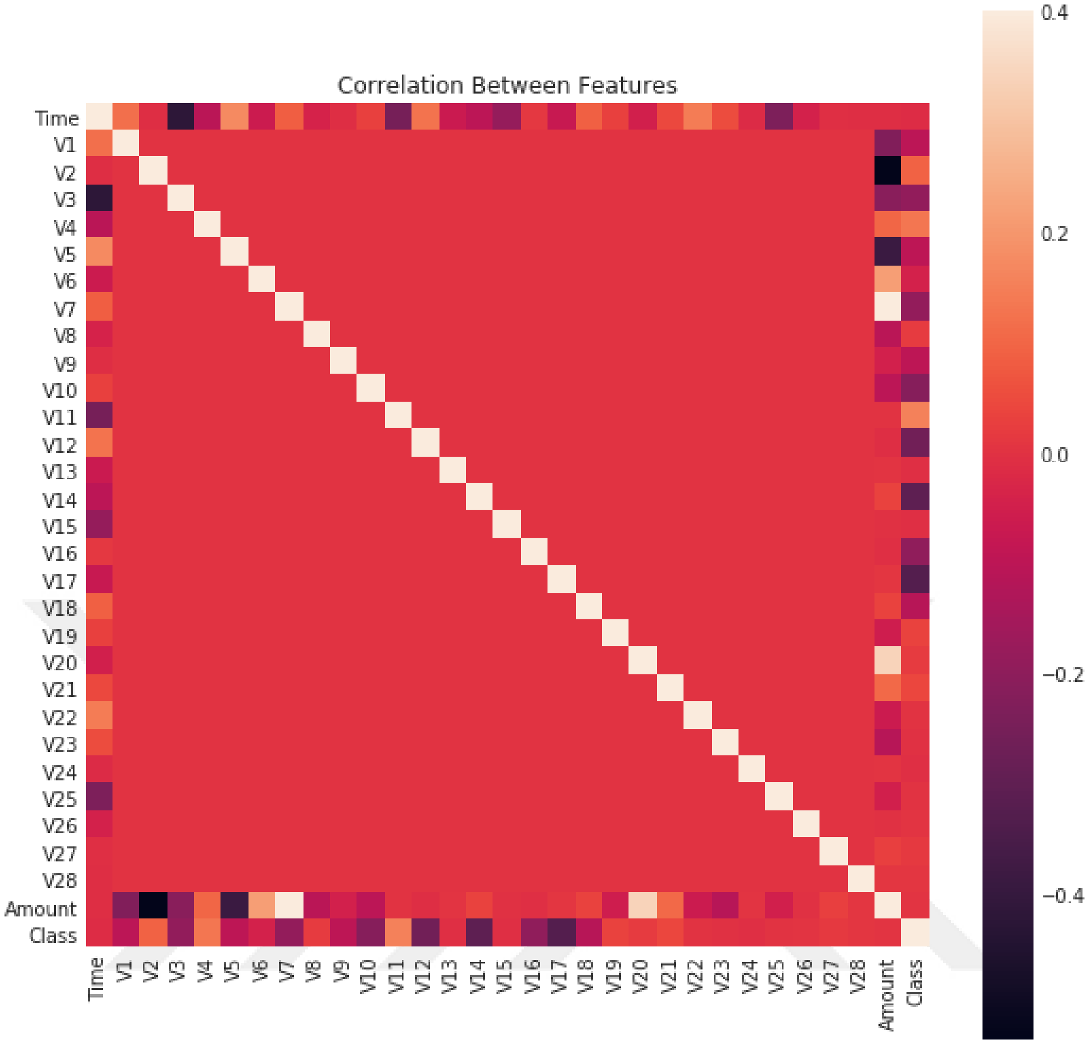


Figure 4. 1: Correlation on the Dataset

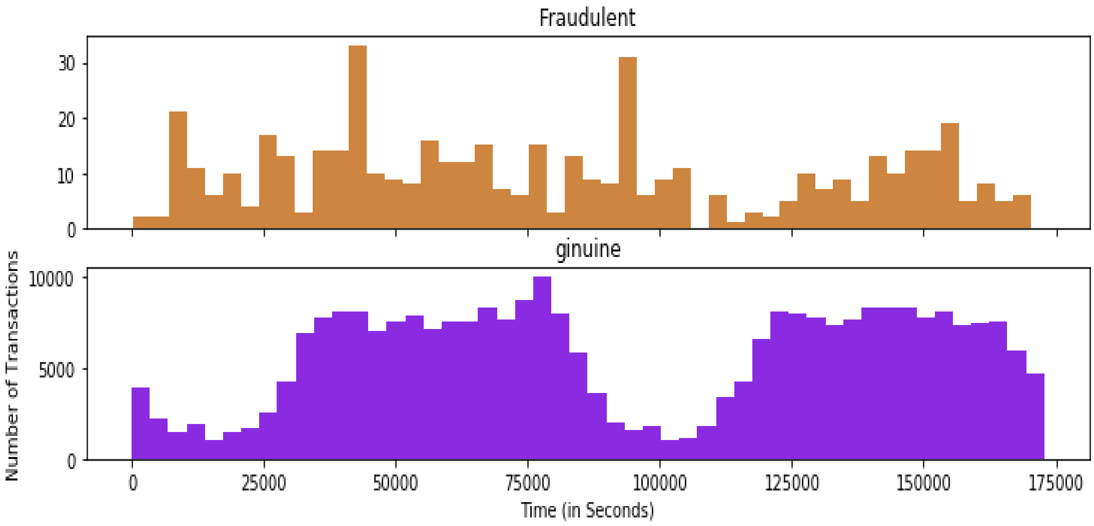


Figure 4. 2: Percentage of Fraudulent and Genuine

As mentioned above, the percentage of fraudulent transactions was 0.172% and the remaining percentage of genuine transactions was 99.828%, as in Figure (4.2).

4.2 Performance Results of Our Study

In this section, we present our results of performance measures from four data mining methods: the naïve Bayesian (NB), Support Vector Machines (SVM), K-Nearest Neighbor (KNN) and Random Forests (RF), evaluated from the training data which carry different levels of fraud cases using the confusion matrix (CM).

4.2.1 Result of (NB)

We tested the NB classifier in our study three ways. First, we took an undersampling approach as a part of preprocessing the dataset to make a division to train and test, as mentioned above, after which we applied that classifier. The result is shown in Figure (4.3) and the confusion matrix in Figure (4.4).

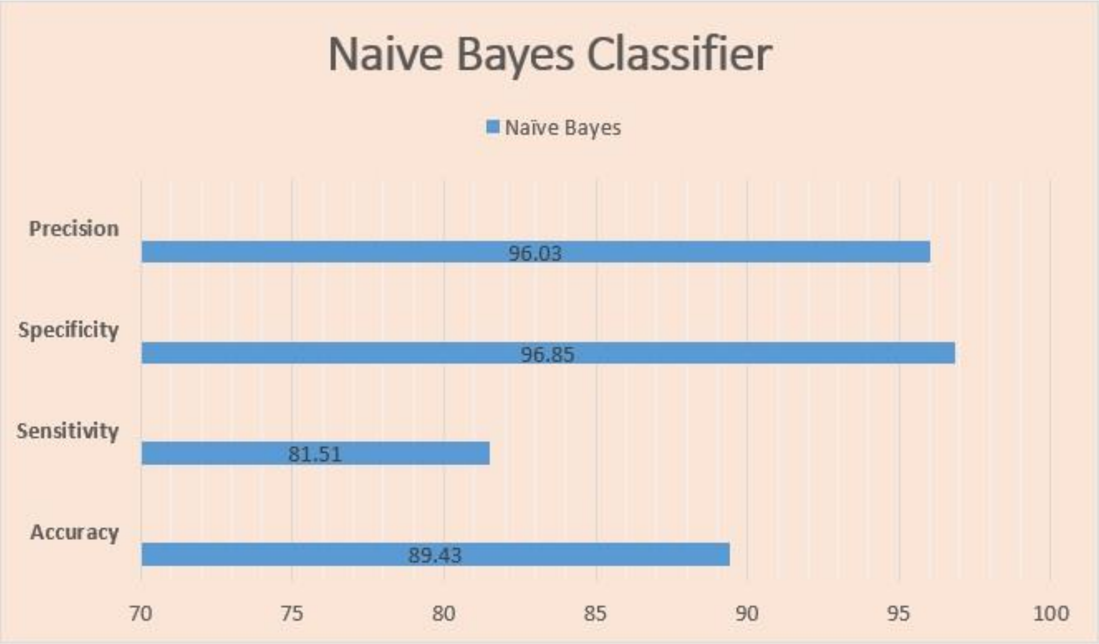


Figure 4. 3: Naïve Bayes with Undersampled Dataset

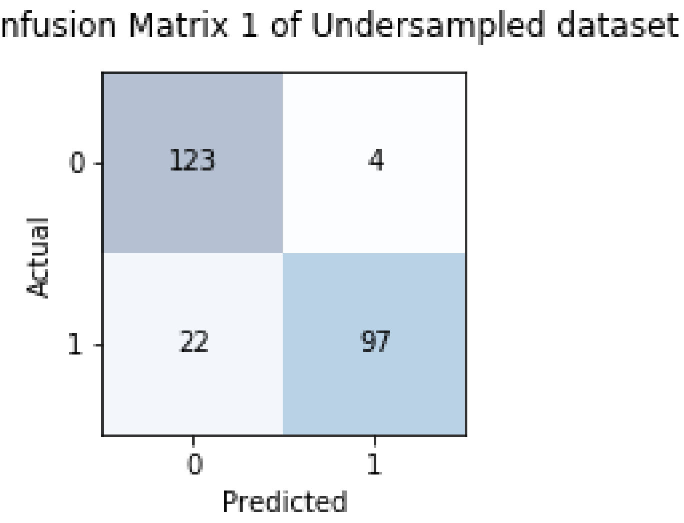


Figure 4. 4: Confusion matrix of Naïve Bayes

In Figure (4.3), accuracy, specificity and precision recorded good values; however, the sensitivity (recall) was low, which means that the sensitivity of fraudulent detection is not satisfactory where we need accuracy for that field in fraud detection. As shown in Figure (4.4), the confusion matrix contains TN = 123, FP = 4, FN = 22 and TP =97, meaning each term mentioned in Chapter Three. The second way we tested the NB classifier in our study was to apply a 10-fold cross-validation on the undersampled dataset to examine the NB classifier performance, the results of which are illustrated in Figure (4.5).

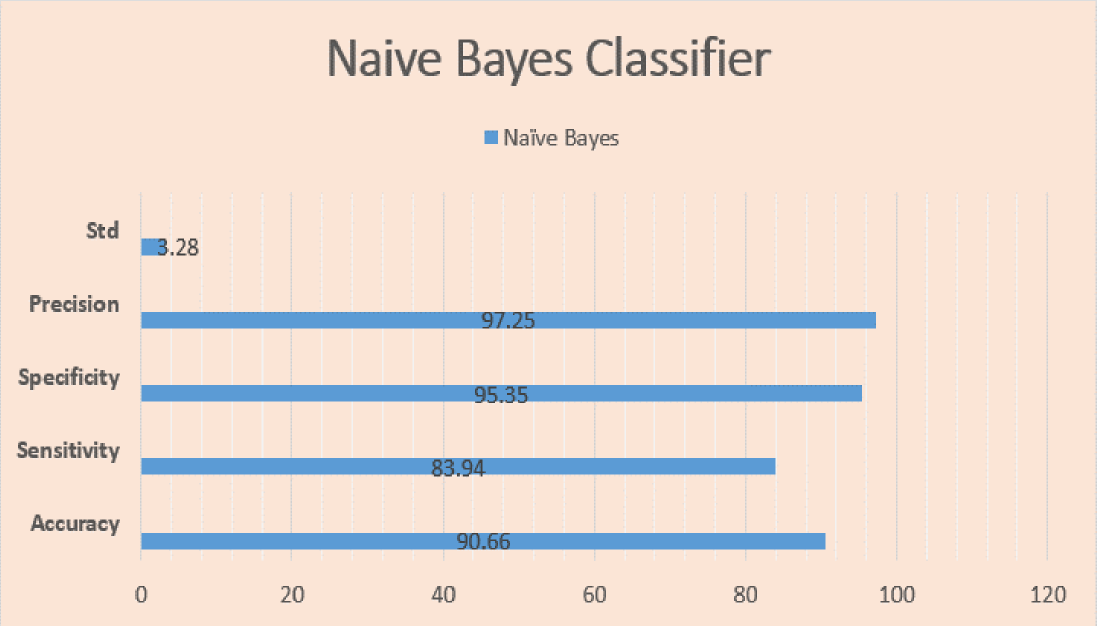


Figure 4. 5: Naïve Bayes with Undersampled Dataset and Cross-Validation

When applying the cross-validation and adding the new measure, which is the standard deviation (Std), we observed a slight increase in accuracy and sensitivity while the specificity and precision almost remained at the previous measures. The third way we tested the NB classifier in our study was to apply naïve Bayes classifier on the entire dataset (skewed), as shown in Figure (4.6), and on the confusion matrix, as in Figure (4.7).

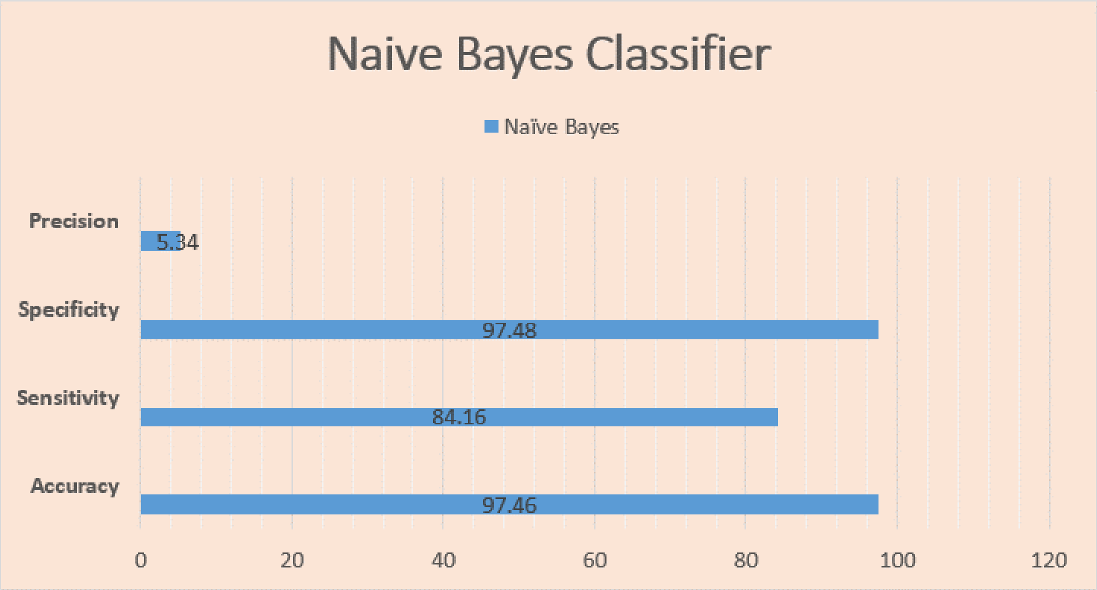


Figure 4. 6: Naïve Bayes with Full Dataset (Skewed)

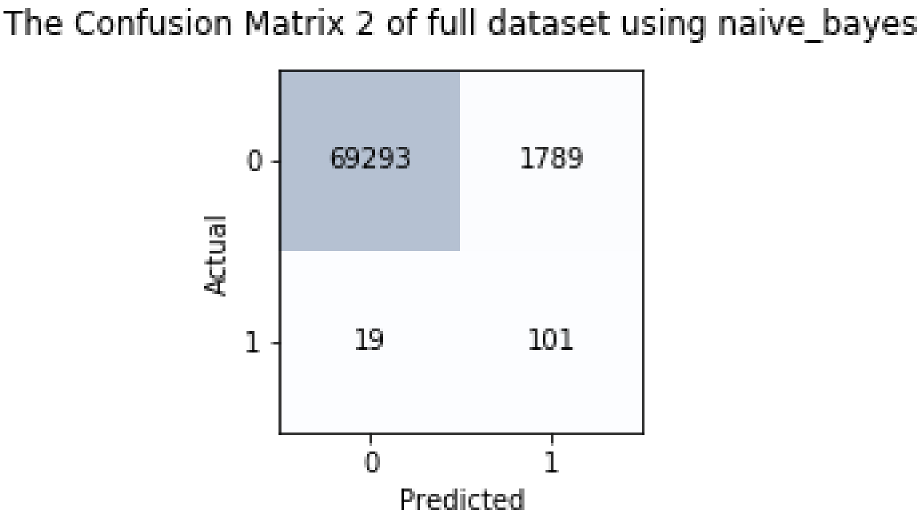


Figure 4. 7: NB Confusion Matrix of Full Dataset

We observe from the above figures (4.6 and 4.7) that the performance measures of the NB classifier are significantly increased compared with above ways, which means applying the naïve-Bayes technique on the entire dataset resulted in better performance, and contained the confusion matrix TN = 69293, FP = 1789, FN = 19 and TP = 101. It is important to note that the confusion matrix deals with the testing dataset, which means the summation of these values is 72,202. This number is 25% of full dataset of 284,807.

4.2.2 Result of (SVM)

We applied SVM classifier in our study also on the three steps, as mentioned Section 4.2.1. For the first step, we used the undersampling approach, after which we applied the classifier. The result is shown in Figure (4.8) and the confusion matrix in Figure (4.9).

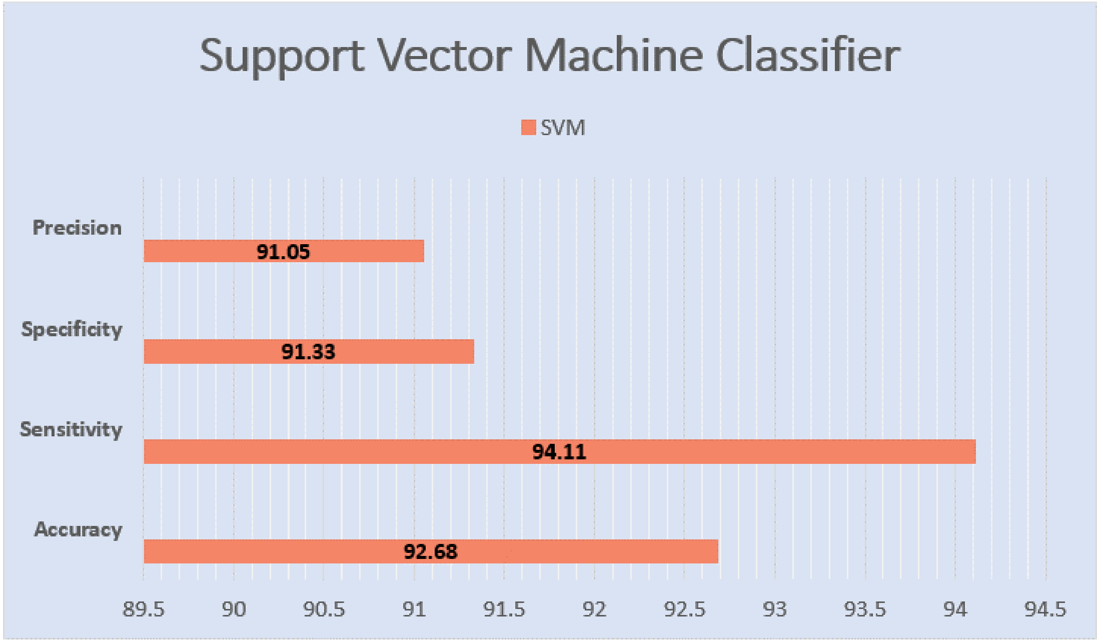


Figure 4. 8: Support Vector Machine with Undersampled Dataset

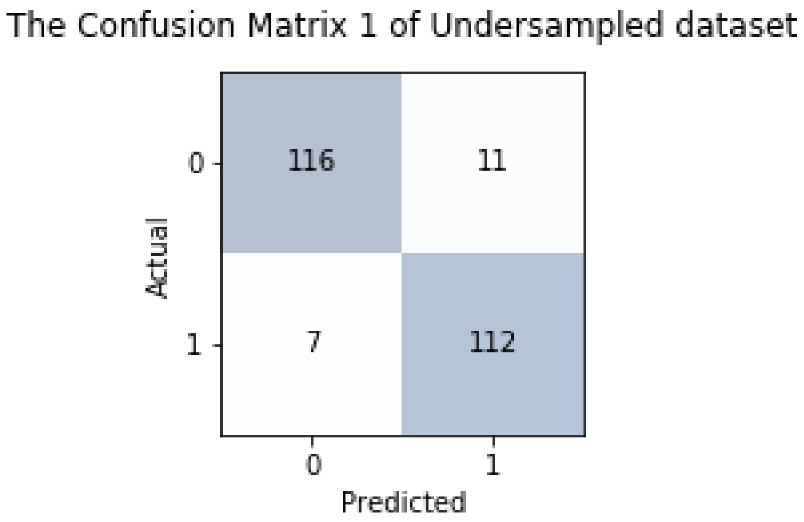


Figure 4. 9: SVM Confusion Matrix of Undersampled Dataset

In Figure (4.8), the accuracy, specificity and precision were recorded as percentages considered to be good values, while the sensitivity (recall) reached a very good value, which means the sensitivity of fraudulent detection is satisfactory for that field in fraud detection. Figure (4.9) shows the confusion matrix; the alarms rate are TN = 116, FP = 11, FN = 7 and TP = 112 respectively. For the second step, we applied 10-fold cross-validation to the undersampled dataset, as illustrated Figure (4.10).

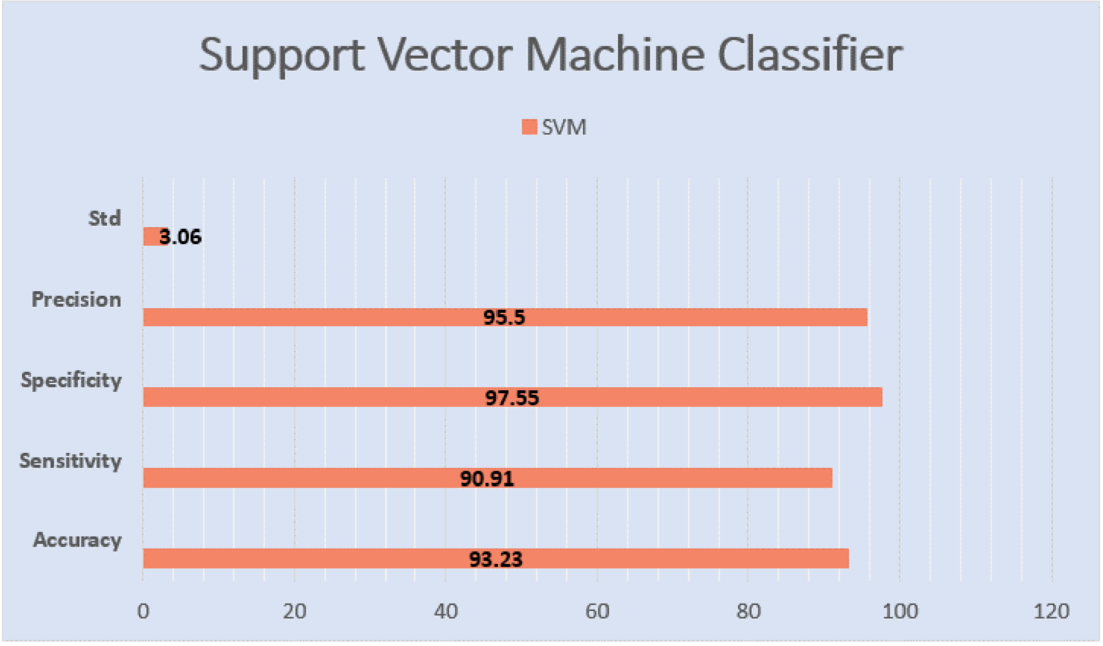


Figure 4. 10: SVM with Undersampled Dataset and Cross-Validation

In Figure (4.10), we applied the cross-validation and added the new examined measure, which is the standard deviation (Std). We observed a significant increase in accuracy, specificity and precision; however, the observation on the sensitivity decreased. In the third step, we applied our classifier to the full dataset (skewed), as shown in Figure (4.11) and in the confusion matrix in Figure (4.12).

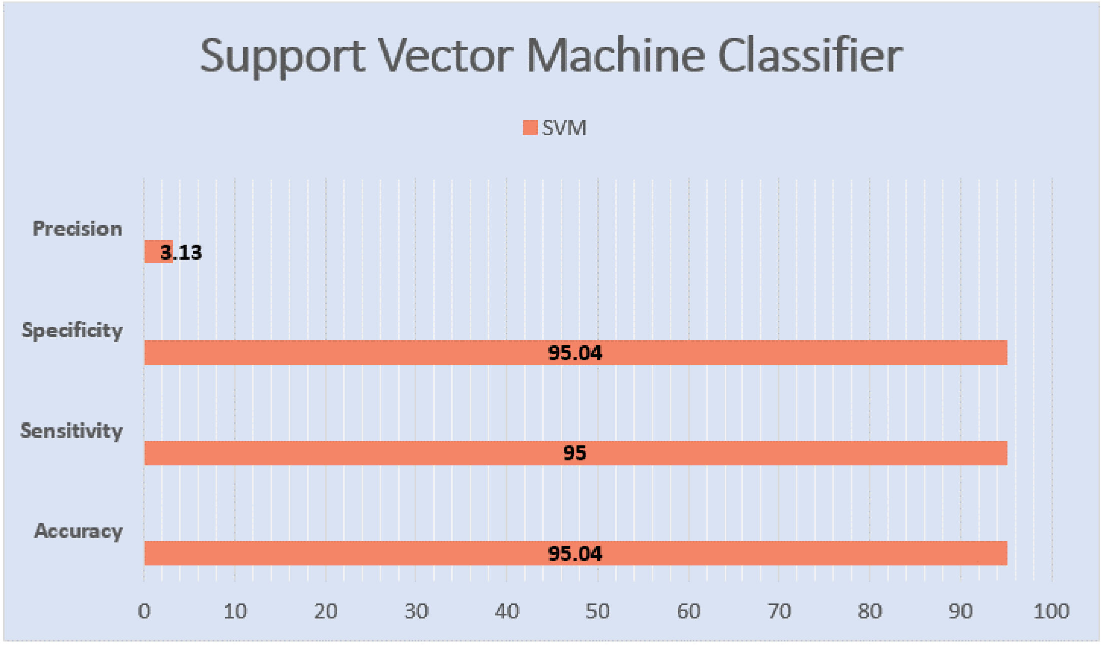


Figure 4. 11: SVM with Full Dataset

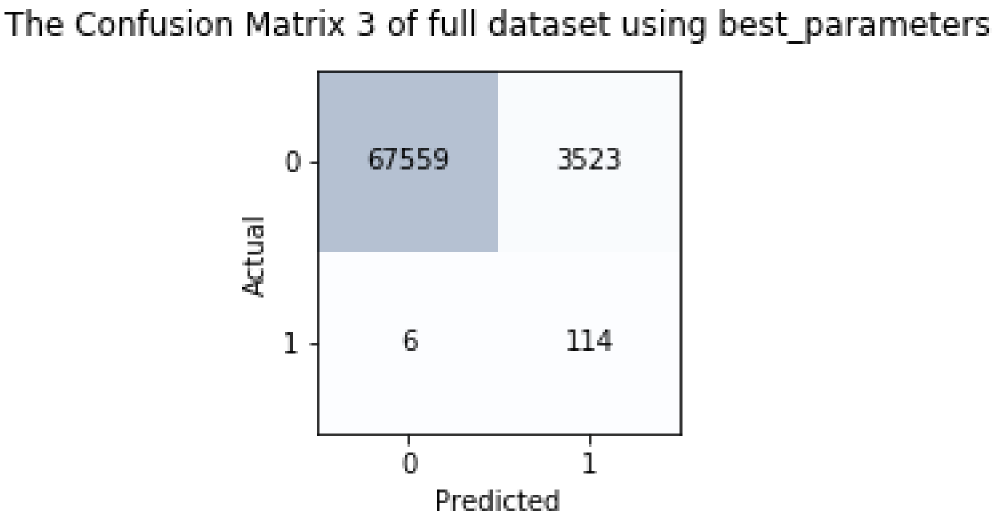


Figure 4. 12: SVM Confusion Matrix of Full Dataset

We observe in Figures (4.11 and 4.12) above that the performance measures of the SVM model increased compared with the above steps, which means applying this technique to the full dataset without preprocessing. It also performed better. The confusion matrix contains TN = 67559, FP = 3523, FN = 6 and TP = 114.

4.2.3 Result of (KNN)

We examined KNN classifier in our study in the three stages, as mentioned in Sections 4.2.1 and 4.2.2. In the first stage, we used the undersampling approach, followed by applying that proposed model. The result is shown in Figure (4.13), and in the confusion matrix in Figure (4.14).

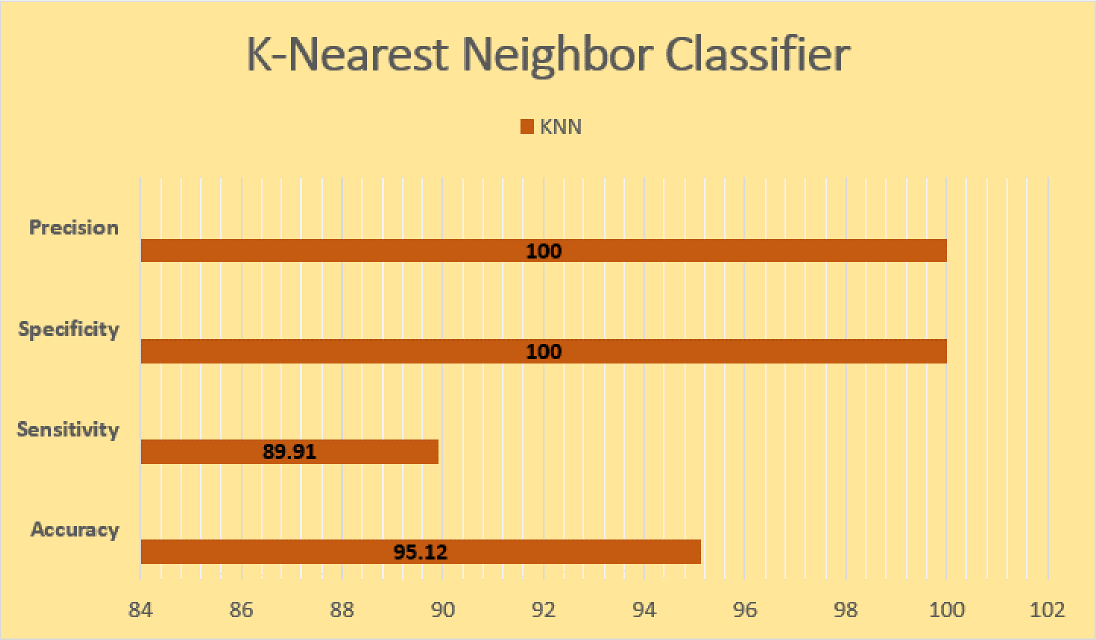


Figure 4. 13: KNN with Undersampled Dataset

In Figure (4.11), the accuracy is a very good value. Specificity and precision recorded full percentages and are considered to have reached the top value for this field of credit card fraud, while the sensitivity (recall) compared with the above values is considered a good value. However, we need a higher result.

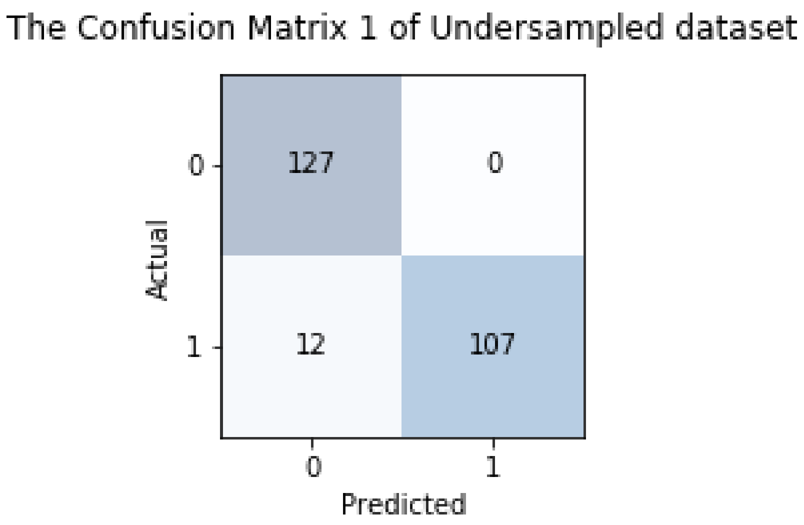


Figure 4. 14: KNN Confusion Matrix of Undersampled Dataset

As illustrated in Figure (4.14) containing the confusion matrix, the alarm rates are TN = 127, FP = 0, FN = 12 and TP = 107, respectively. Second stages, applied cross-validation with 10 fold on the undersampled dataset were result as illustrated in the Figure (4.15).

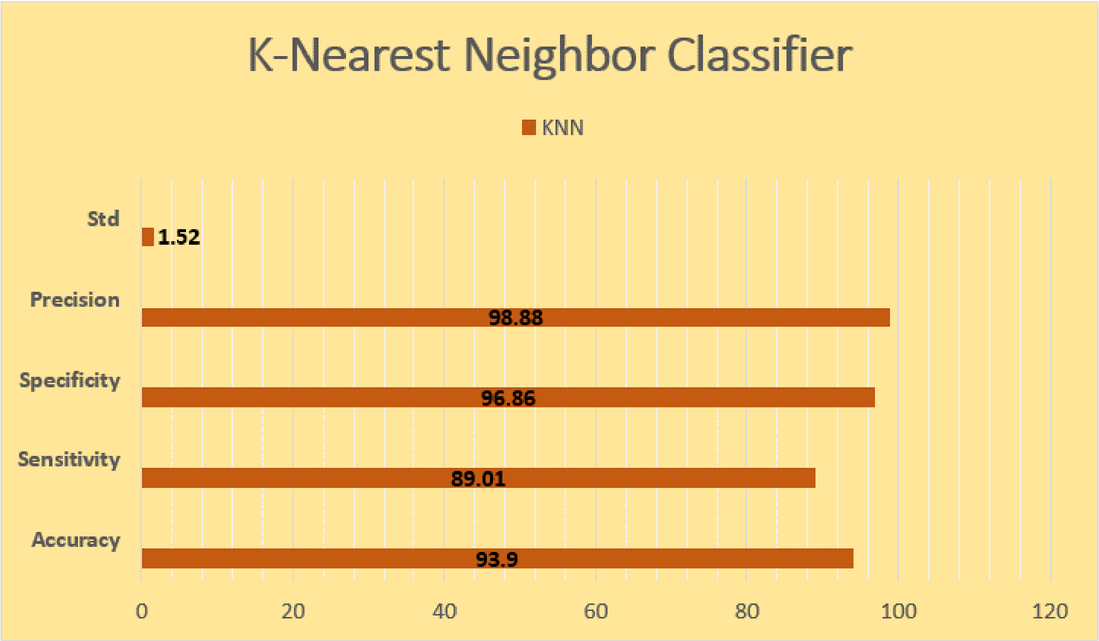


Figure 4. 15: KNN with Undersampled Dataset and Cross-Validation

In Figure (4.15), when applying the cross-validation and adding the new examined measure of the standard deviation (Std), the observation is slightly decreased in all performances.

It can be said here that there is no benefit of applying cross validation. In the third stage, we tested our technique on the full dataset, as shown in Figure (4.16), and on the confusion matrix in Figure (4.17).

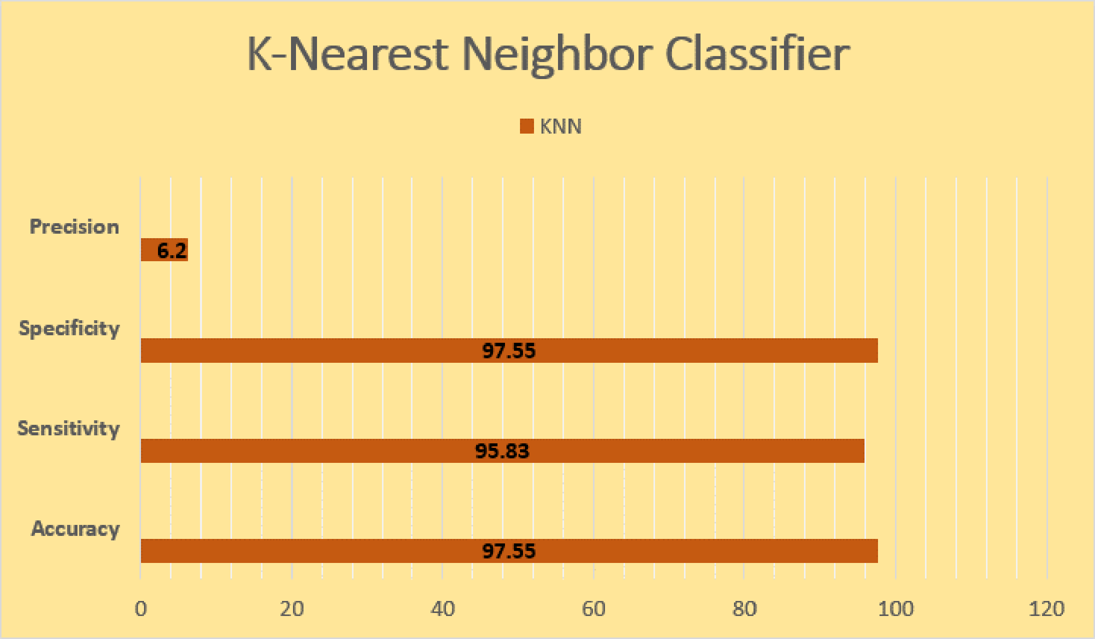


Figure 4. 16: KNN with Full Dataset

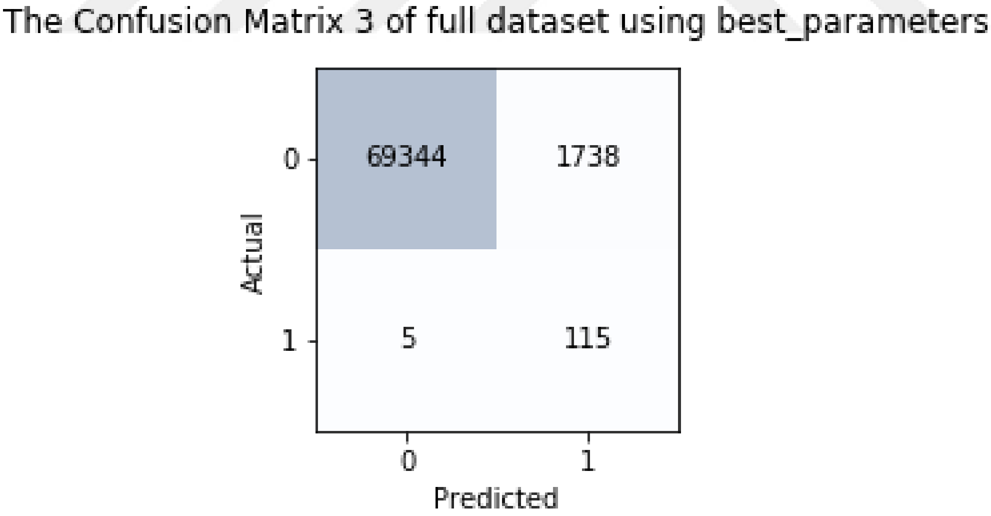


Figure 4. 17: KNN Confusion Matrix of Full Dataset

We observe in Figures (4.16 and 4.17) that the performance measures of the KNN model are increased compared with the above stages, which means that by applying this classifier to the full dataset without preprocessing, it also performed better, with the confusion matrix containing TN = 69344, FP = 1738, FN = 5 and TP = 115, as mentioned in the sections above.

4.2.4 Result of (RF)

When we applied the random forest (RF) classifier in this study also to the three stages, as mentioned in Sections 4.2.1, 4.2.2, and 4.2.3, in the first stage, we used the undersampling approach, followed by applying that model. The results are shown in Figure (4.18), and confusion matrix in Figure (4.19).

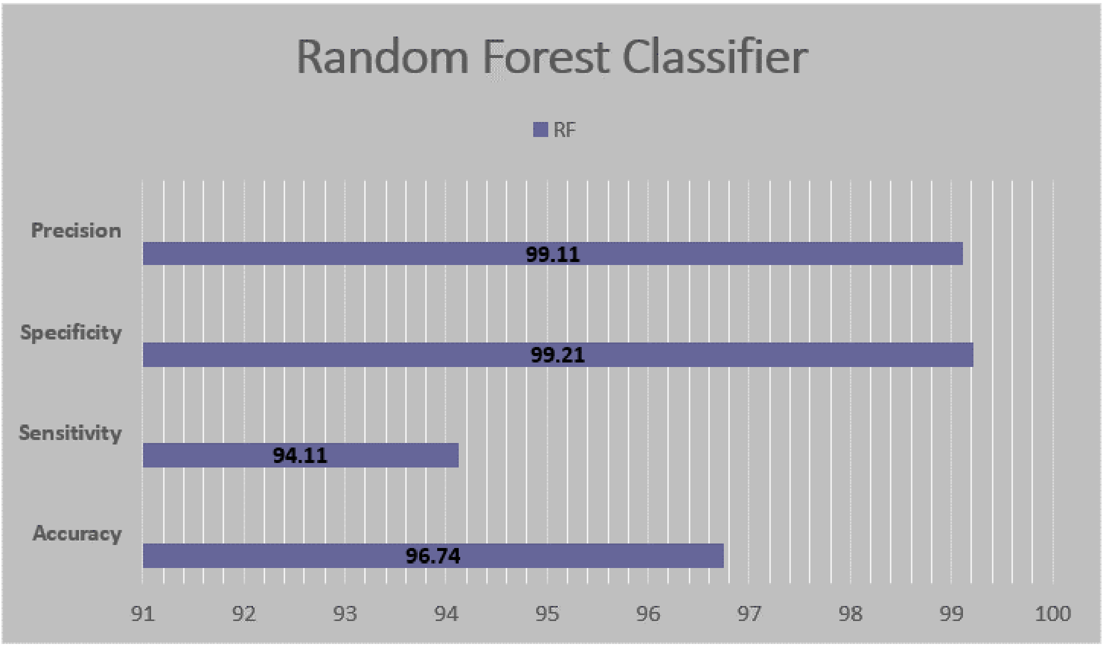


Figure 4. 18: Random Forest with Undersampled Dataset

All the measures are satisfactory in terms of accuracy, sensitivity, specificity and precision in Figure (4.18), and were recorded and achieved very good values, which is necessary in this field of credit card fraud.

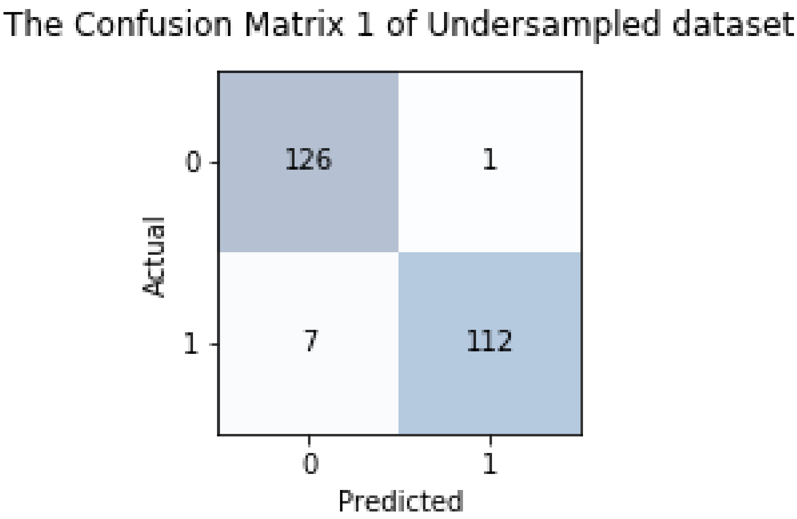


Figure 4. 19: RF Confusion Matrix of Undersampled Dataset

Figure (4.19) shows the confusion matrix, the alarm rates of which are TN = 126, FP = 1, FN = 7 and TP = 112. In the second stage, we applied a 10-fold cross-validation to the undersampled dataset, the results of which are illustrated in Figure (4.20).

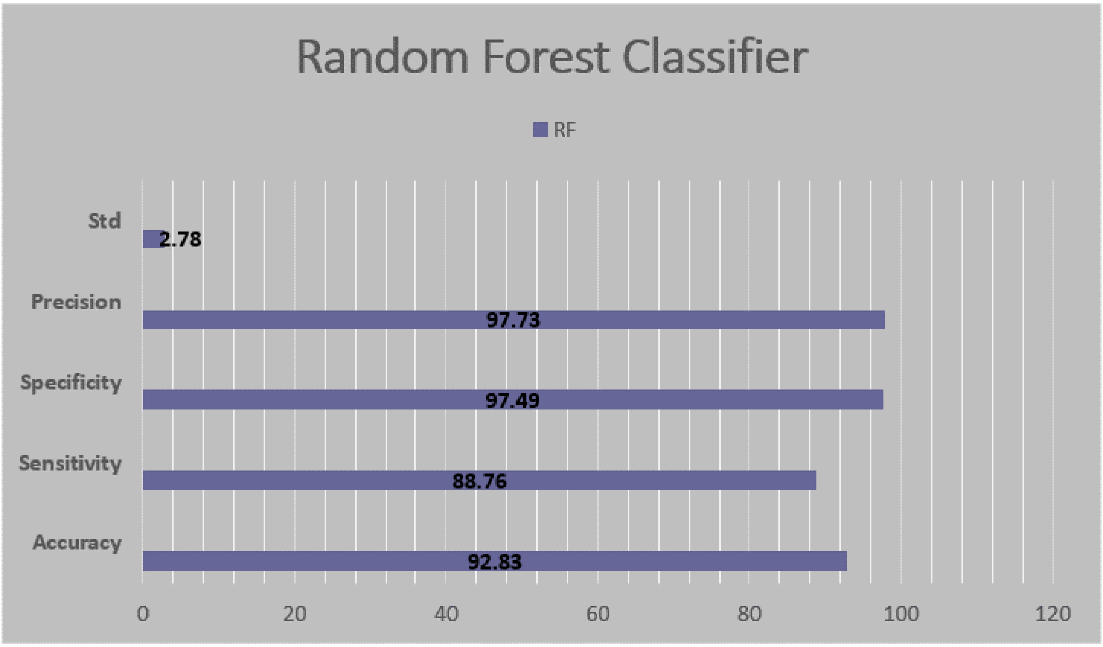


Figure 4. 20: Random Forest with Undersampled Dataset and Cross-Validation

In Figure (4.20), the cross-validation was applied and we added the new examined measure of the standard deviation (Std). We can observed a decrease in every performance measure. There is also no benefit to applying cross validation. In the third stage, we tested our technique on the entire dataset (skewed), as shown in Figure (4.21), and on the confusion matrix in Figure (4.22).

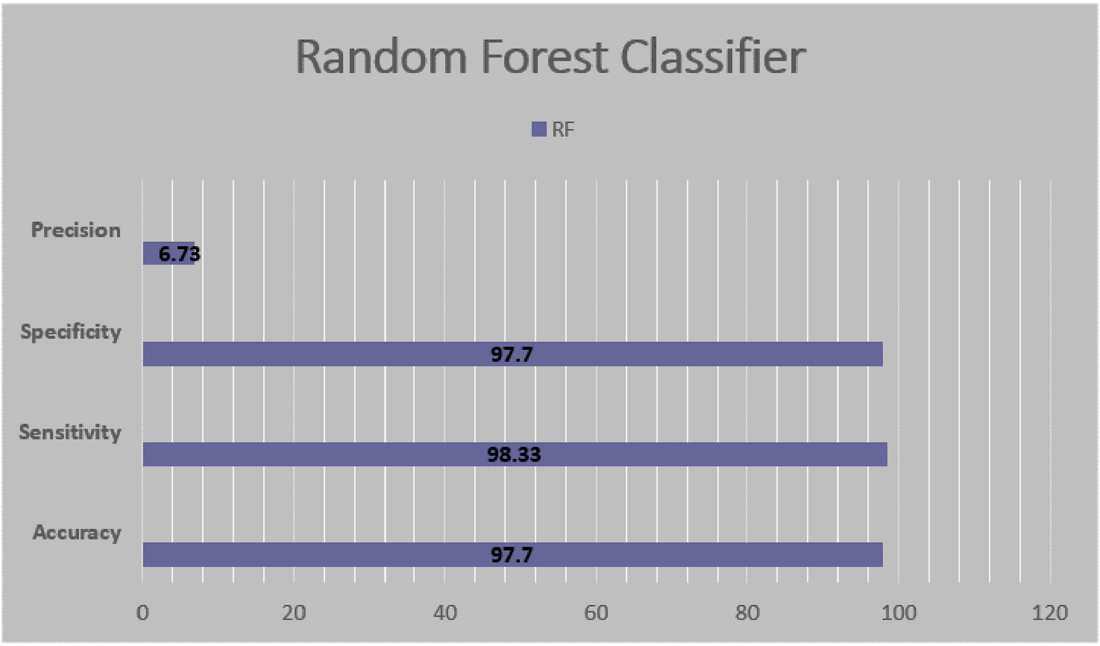


Figure 4. 21: Random Forest with Full Dataset

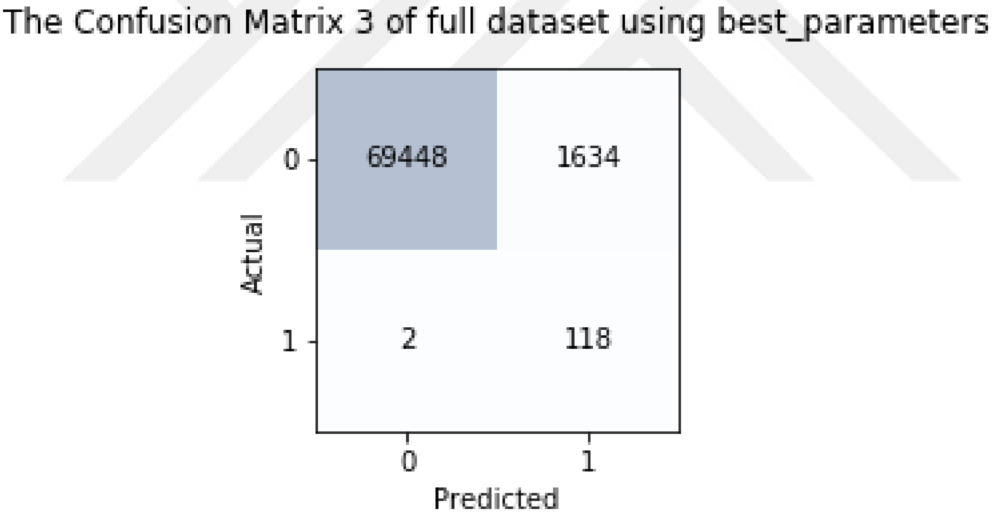


Figure 4. 22: Random Forest Confusion Matrix of Full Dataset

In Figures (4.21 and 4.22), the performances of the RF classifier are also increased, which means that by applying this classifier to the entire dataset without the under-sampling approach, it also performed better. The confusion matrix contained TN = 69448, FP = 1634, FN = 2 and TP = 118, as mentioned in the sections above.

4.3 Comparative Study of Proposed Models

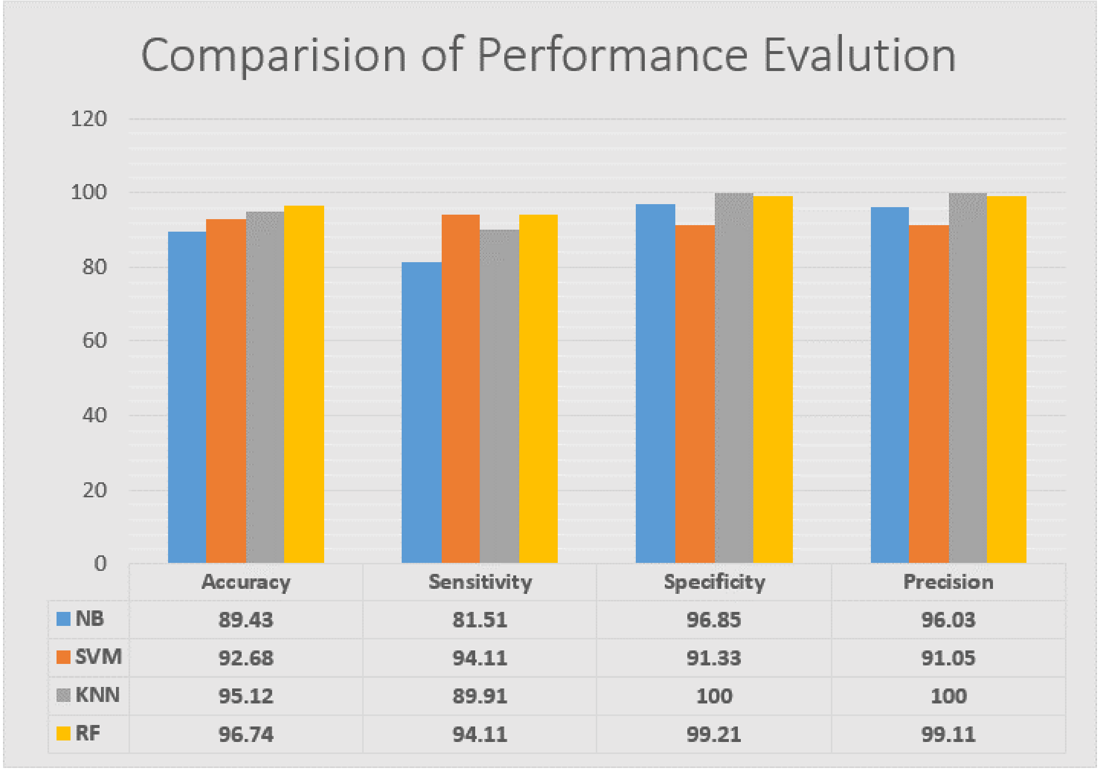
As we pointed out previously, to ensure and provide advice for which methods we have selected above, we considered the best method for the classification of credit cards between genuine and fraudulent.

4.3.1 Classification Methods Comparison

We compared the results of four methods, namely the naïve Bayes (NB), support vector machine (SVM), K-nearest neighbor (KNN) and random forest (RF). We implemented these classifiers on four classification methods. The comparison also was based on accuracy, sensitivity, specificity and precision.

First, we present in Table (4.1) our results for the performance measure accuracy, sensitivity, specificity and precision, respectively, for the four techniques after we followed the undersampling approach to balance the dataset by removing a number of genuine classes to reach the number of the minority class. This means that the majority class becomes 492 transactions, equaling the number of the minority class, where we used 75% of the undersampled dataset to train 738 transactions and 25% of the undersampled dataset to test 246 transactions. Note that in the results in Table (4.1), the RF technique produced higher accuracy and sensitivity but for specificity and precision, the KNN produced highest results reaching 100 for specificity and precision, while the NB produced less accuracy and sensitivity. Moreover, the SVM produced lower specificity and precision.

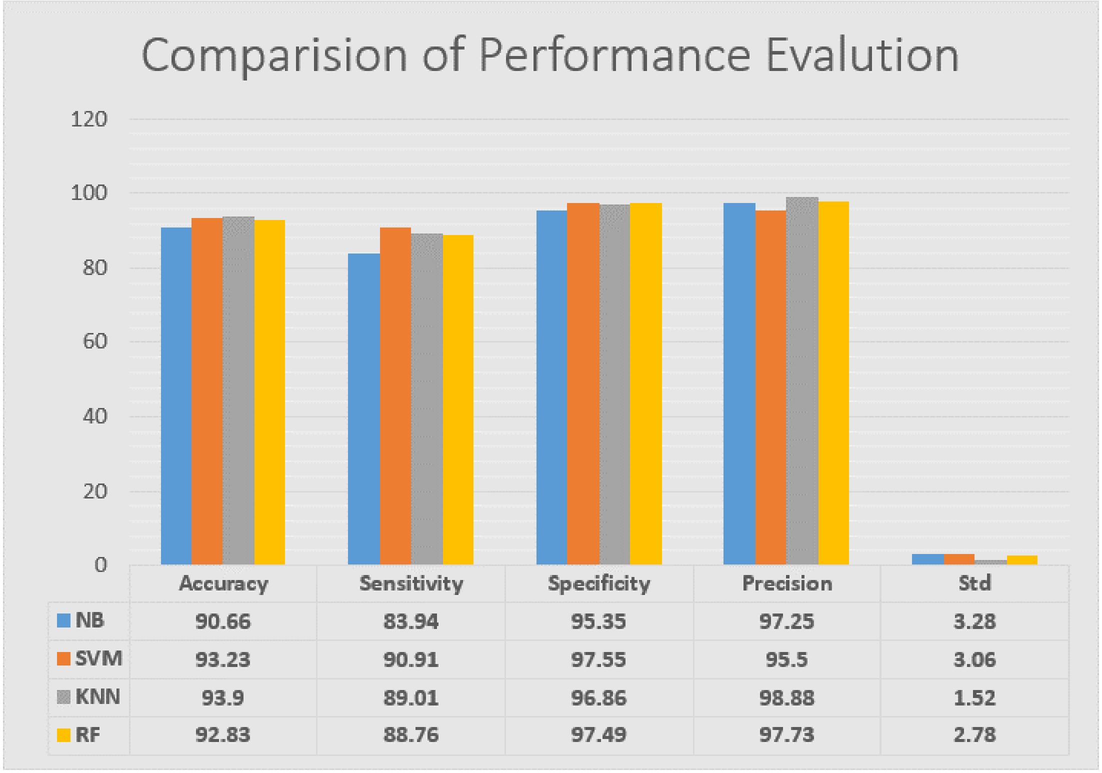
Table 4. 1 : Performance of under-sampling data set for four techniques



	Accuracy	Sensitivity	Specificity	Precision
🟦 NB	89.43	81.51	96.85	96.03
🟧 SVM	92.68	94.11	91.33	91.05
◼ KNN	95.12	89.91	100	100
🟨 RF	96.74	94.11	99.21	99.11

Second, in Table (4.2), we present the results for our models. However, here we applied cross-validation to obtain the traditional classification performance for the same size of training and testing datasets as well as for the under-sampling. In this table, we added another measure, namely a standard deviation (Std), which measures the spread of the dataset. The dataset with the smaller Std has a narrower spread of measurements around the mean and it usually has comparatively low values. As we observe in this table, there is an increase in the accuracy and sensitivity of the NB technique as well as an increase in the accuracy for the SVM. However, in the remaining measures for the NB and SVM, there is a notable decrease. While the performance measures of KNN and RF have decreased when applying the cross-validation, as mentioned in the above section. On the other hand, the standard deviation is a better record of the KNN and RF than the NB and SVM.

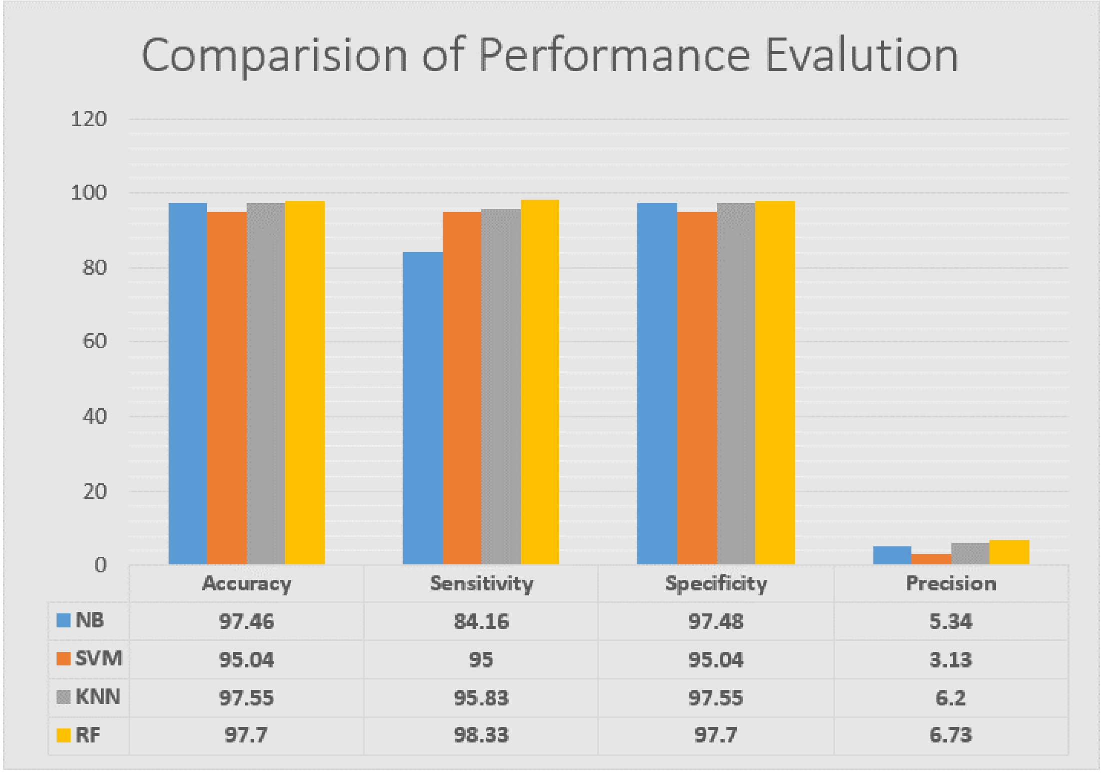
Table 4.2: Apply cross-validation performance measures of four techniques and standard deviation



	Accuracy	Sensitivity	Specificity	Precision	Std
🟦 NB	90.66	83.94	95.35	97.25	3.28
🟧 SVM	93.23	90.91	97.55	95.5	3.06
⬜ KNN	93.9	89.01	96.86	98.88	1.52
🟨 RF	92.83	88.76	97.49	97.73	2.78

For the third way, we present in Table (4.3) the results of our models when applied to the entire dataset, that is, a skewed dataset. In terms of dataset division previously, we also divided the dataset into two sampling datasets of 75% for training and 25% for testing, which means 213,605 transactions for training and 71,202 for testing. We observe that all performance measures for our models have noted increased only specificity of KNN is decreased, It is important to note that in Table (4.3) that the precision measure has different values than in the above tables and also in Figures (4.6, 4.11, 4.16 and 4.21). This difference is due to the precision measure depending on the true positive class (fraudulent) among all positives and the number of fraudulent transactions in the entire dataset being a very small approximate (0.172%) of all the transactions in comparison with the genuine transactions, as mentioned previously.

Table 4.3: Performance results for imbalanced dataset (skewed) distributions



	Accuracy	Sensitivity	Specificity	Precision
🟦 NB	97.46	84.16	97.48	5.34
🟧 SVM	95.04	95	95.04	3.13
▨ KNN	97.55	95.83	97.55	6.2
🟨 RF	97.7	98.33	97.7	6.73

After implementing these four classifiers, and comparing the detailed performance measures of them, we conclude that the RF algorithm performs better than the other classifiers.

4.3.2 AUC Ranking Comparative

Finally, we present the area under the ROC curve (AUC) measure for our experimental techniques. In order to compare between them, we propose to use the AUC measure for two types of dataset. The first use is with the undersampled data and the second use with the imbalanced data when used with undersampling data. The results for our models are illustrated in Figures 4.23, 4.24, 4.25 and 4.26, where, as usual, the highest results were RF, KNN, SVM and NB. We also observe, when used with all the data, as illustrated in Figures 4.27, 4.28, 4.29 and 4.30, that the results are also highest in RF, KNN, SVM and NB.

Figure 4.23: AUC measure for NB

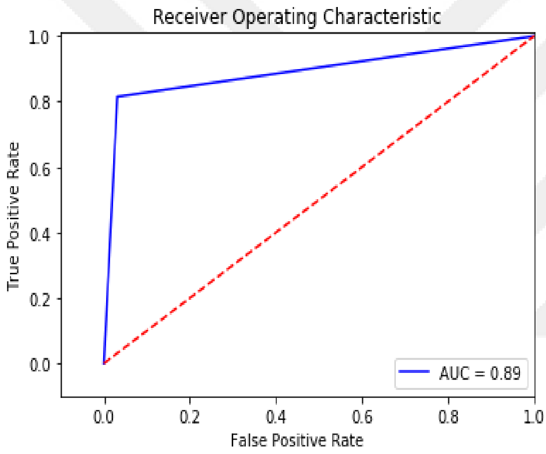


Figure 4.24: AUC measure for SVM

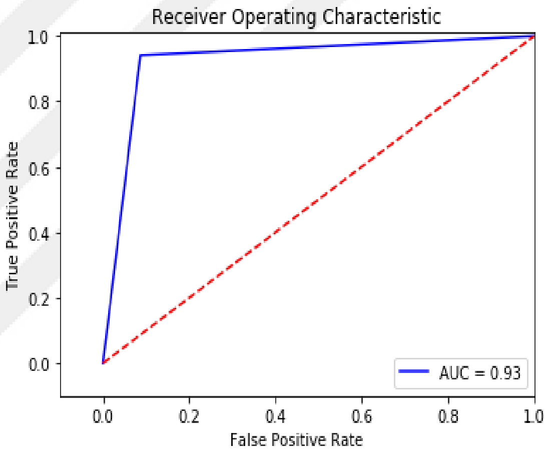


Figure 4.25: AUC measure for KNN

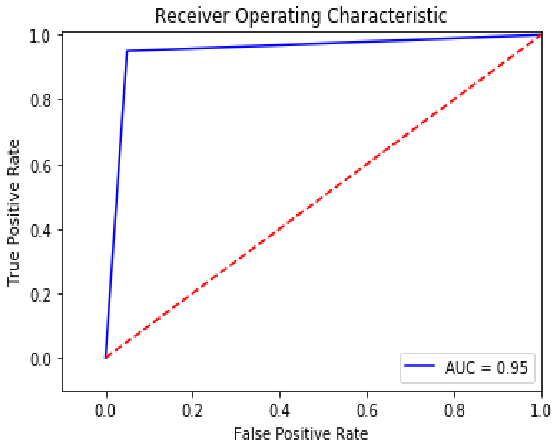


Figure 4.26: AUC measure for RF

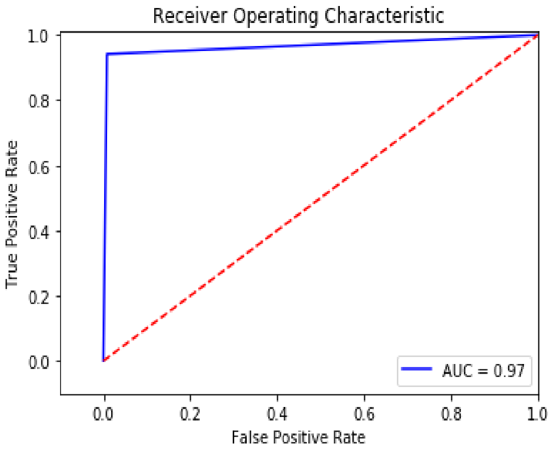


Figure 4.27: AUC measure for NB

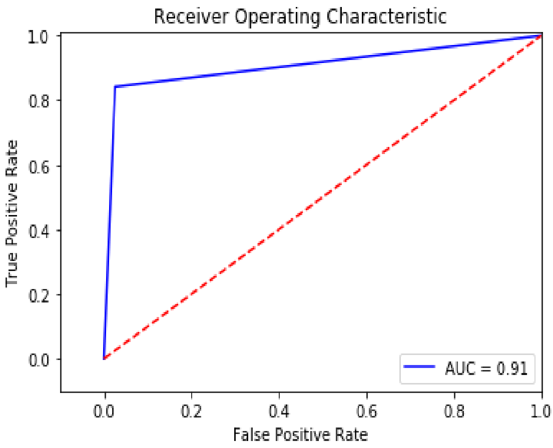


Figure 4.28: AUC measure for SVM

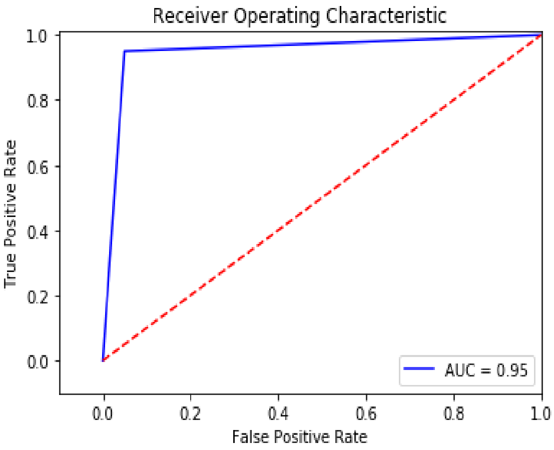


Figure 4.29: AUC measure for KNN

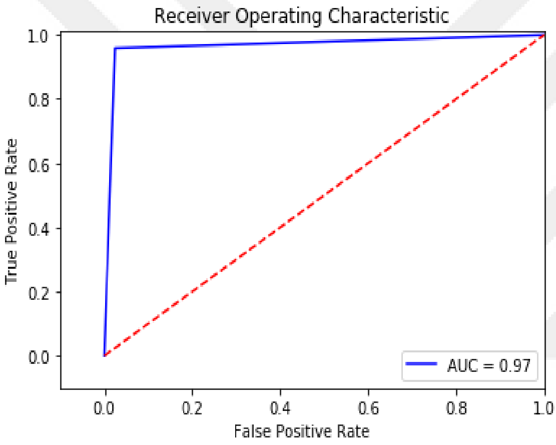
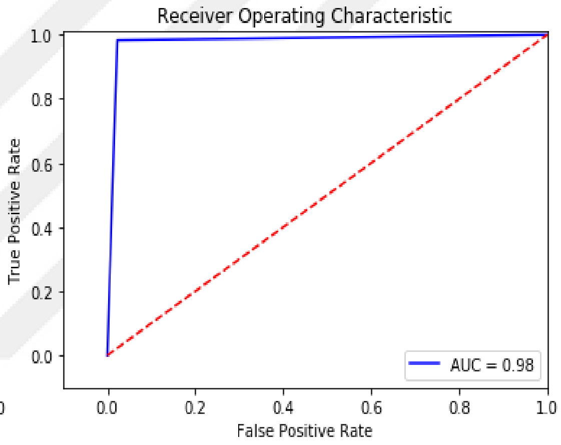


Figure 4.30: AUC measure for RF



This comparison reaffirms that the RF classifier was also trained on two distributions of the dataset and obtained the higher performance classifier for the prediction of new cases among all classifiers.

CHAPTER FIVE

CONCLUSION

The fifth chapter sums up the overall research. Firstly, we present a review of the calculated results with some reflections on the evaluation design. This is followed by deeper analysis of the feature variables, disadvantages that arise and presented the comparison with the previous works. Lastly, further studies within the field are discussed.

5.1 Discussion

In conclusion, this one-year Master thesis has provided a deeper knowledge of the field of credit card fraud detection using machine learning algorithms. We investigated and examined in this study four classifier techniques: 1- 'Naïve-Bayesian(NB)', 2- 'Support Vector Machines(SVM)', 3- 'K-Nearest Neighbor(KNN)' and 4- 'Random Forests(RF)'. Several techniques used by criminals have been documented, including the underlying terminology behind advanced statistical models that possess the capacity to prevent fraudulent behaviors, examine credit card fraud problems with binary classification as this problem has become very common in banks. Our study has contributed to three major trends. First, we tested four proposed techniques following the under-sampling dataset approach. Second, we applied cross-validation with a 10-fold iteration and comparative performance of the four methods between them. Third, an examination was made of the four classification methods while being applied to the entire dataset or skewed dataset with comparisons of their performances. All these comparisons and analyses were completed with the classifiers trained on a 25:75 ratio of fraudulent to genuine transactions.

Therefore, we conclude that the performance measures of our models increase when applied to the entire dataset than use undersampled dataset, due to the undersampling approach suffering weakness when used with a huge dataset, where remove the number of majority class even equal to minority class has great effect on results, On the other hand, when we used cross validation, some of techniques increased their performances and others decreased them. As presented in Chapter

Four, also concluded from our comparative analysis, the Random Forest (RF) technique is the best classification technique for credit card fraud problems. Which has better results for all evaluated performances on the three examination results. Therefore, we advise the use of this technique with huge datasets with **100** estimators.

The other important factor which may have a serious impact on the performance of the classifiers was the limitations of the data sets. The most important limitations were the rather small fraudulent database and the lack of FDS scores associated with the flagged transactions.

In summary, pattern recognition for genuine/fraudulent occurrences is inherently complex and since genuine cardholders’ and fraudsters’ patterns of behavior evolve through time, our study is a base for further research. Overall, the techniques used in this study demonstrate that the approach employed in this research has a very good potential for distinguishing 'genuine transactions' from 'fraudulent transactions'. This means that through the advanced results, we can deem the KNN technique to be satisfactory to deal with such problems as the credit card fraud problem needs accurate techniques of detection.

5.2 Comparison with Previous Work

As we pointed out previously, to make sure that we have chosen the best method for the classification of credit card fraud detection between genuine and fraudulent, we compare our results with previous works, as we shown in the table (5.1) below.

Table 5. 1: Comparison our study with previous works

Reference	NB	SVM	KNN	RF	Dataset Source
[10](2011)	-	93.8	-	96.2	Almost 50 million credit card transactions from an international company from 2006 to 2007, all of which occurred in a single country.
[34](2012)	96.04	-	-	91.09	The actual transactions dataset contains details about the purchases made via customers' credit cards such as amount, location, time, date and etc.
[7](2016)	94.10	94.17	-	95.81	A leading bank in Turkey provided a real‐life credit card transaction dataset for the evaluation of their models.
[21](2017)	97.69	-	97.92	-	The dataset is same as our dataset.
Our Study (2018)	97.46	95.04	97.55	97.7	Our dataset is provided based on transactions from European cardholders for a two day period that have been made in September 2013 [4].The dataset published on public since 2016.

The dataset used with our study has become on the public in 2016, for this reason their usage is still not commonly used by researchers. Note in the study [21] the NB and KNN classifiers were obtained results 97.69 for NB and 97.92 for KNN. They obtained these results after tested two data distributions, firstly %10 testing:%90 training and secondly %34 testing:%66 training, where they concluded the best result were on the second data distribution. Compared with our study that was used the under-sampling approach with %25 testing and

%75 training data, where got slightly decreased in accuracy of NB and KNN as noted in table (5.1) above.

After making this comparison with previous works, and comparing the detailed performance measures with RF algorithm, we conclude that also RF algorithm is performing better than other researches. We can advise as we mentioned in above section to use this technique and apply it on the huge dataset directly without make sampling approaches.

5.3 Future Work

Future work in this area includes:

- Investigation of other preprocessing, feature selection and feature weighting sections that can better represent credit card datasets to enhance the accuracy of classification.
- Expecting and attempting to extend our study through the use of another approach to sample datasets, such as the Synthetic Minority Over Sampling Technique (SMOTE).
- Formulation datasets with higher minority instances (example, 60:40, 70:30, 90:10 etc. of 'fraudulent/genuine' cases) in order to investigate the impcat of class distribution on the performance of classifiers, and based on this evaluation to select the best predictive classifier to use.
- Different learning techniques that can be applied to the same sample data as well as the meta-learning strategy classifiers.
- As discussed previously, the fraud environment is dynamic; therefore, the system being designed must be adaptive to a changing environment of fraud.

References

[1] Jon T.S. Quah and M. Sriganesh, "Real-time credit card fraud detection using computational intelligence," *Expert Systems with Applications*, pp. 1721–1732, NOV 2008.

[2] the nilson report, "perchas transactions worldwide 2016 vs 2026," 2018. [Online]. https://www.nilsonreport.com/publication_chart_of_the_month.php

[3] Mohd Aizaini Maarof and Anazida Zainal Aisha Abdallah, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, pp. 90–113, june 2016. [Online]. http://dx.doi.org/10.1016/j.jnca.2016.04.007

[4] Giacomo Boracchi, Olivier Caelen, Cesare Alippi, Fellow, IEEE, Andrea Dal Pozzolo, "Credit Card Fraud Detection: A Realistic Modeling a Novel Learning Strategy," *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, pp. 2162-237X, 14/09/2017. [Online]. http://www.ieee.org/publications_standards/publications/rights/index.html

[5] Olivier Caelen, Yann-Aël Le Borgne, Serge Waterschoot and Gianluca Bontempi Andrea Dal Pozzolo, "Learned lessons in credit card fraud detection from a practitioner perspective," *Expert Systems with Applications*, pp. 4915–4928, 01/08/2014. [Online]. http://dx.doi.org/10.1016/j.eswa.2014.02.026

[6] Jarrod West and Maumita Bhattacharya, "Intelligent financial fraud detection: A comprehensive review," *ScienceDirect*, pp. 47–66, 12 November 2015. [Online]. http://dx.doi.org/10.1016/j.cose.2015.09.005

[7] Yiğit Kültür and Mehmet Ufuk Çağlayan, "Hybrid approaches for detecting credit card fraud," *wiley - Expert Systems*, pp. 1–13, 28/10/2016.

[8] Duygu. Tesco Bank deploys FICO's banking solutions for risk Tavan, "fraud management," 2011. [Online]. http://www.vrl-financial-news.com/retail-banking/retail-bankerintl/issues/rbi-2011/rbi-645/tesco-bank-deploys-fico%E2%80%99s-bank.aspx.

[9] Richard J. Bolton and David J. Hand, "Unsupervised Profiling Methods for Fraud Detection," london, 2001.

[10] Sanjeev Jha, Kurian Tharakunnel and Christopher Westland Siddhartha Bhattacharyya, "Data mining for credit card fraud: A comparative study," *Decision Support Systems*, pp. 602–613, 2011.

[11] Richard J. Bolton and David J. Hand, "Statistical Fraud Detection: A Review," *Statistical Science*, pp. 235–255, 2002.

[12] TUNG-SHOU CHEN, and CHIH-CHIANG LIN RONG-CHANG CHEN, "A NEW BINARY SUPPORT VECTOR SYSTEM FOR INCREASING DETECTION RATE OF CREDIT CARD FRAUD," *International Journal of Pattern Recognition and Artificial Intelligence*, pp. 227-239, 2006. [Online]. https://doi.org/10.1142/S0218001406004624

[13] Y. Sahin and E. Duman, "Detecting Credit Card Fraud by Decision Trees and Support Vector Machines," *international multiconference of engineers and computer scientists (IMECS)*, pp. 442–447, 2011. [Online]. http://www.iaeng.org/publication/IMECS2011/IMECS2011_pp442-447.pdf

[14] C. Whitrow · D. J. Hand · P. Juszczak · D. Weston · N. M. Adams, "Transaction aggregation as a strategy for credit card fraud detection," *Data Mining and Knowledge Discovery*, pp. 30–55, 2009.

[15] D.UmaDevi K.RamaKalyani, "Fraud Detection of Credit Card Payment System by Genetic Algorithm," *International Journal of Scientific & Engineering Research*, pp. 1-6, 2012.

[16] M. Hamdi Ozcelik Ekrem Duman, "Detecting credit card fraud by genetic algorithm and scatter search," *Expert Systems with Applications*, pp. 13057–13063, 2011.

[17] Xin Li Haiying Ma, "Application of Data Mining in Preventing Credit Card Fraud," *international conference on management and service science (MASS)*, pp. 1-6, 2009.

[18] B. Freisleben, and B. Rao E. Aleskerov, "CARDWATCH: A neural network based database mining system for credit card fraud detection," *Computational Intelligence for Financial Engineering (CIFEr)*, pp. 220–226, 1997.

[19] E. Duman Y. Sahin, "Detecting Credit Card Fraud by ANN and Logistic Regression," *international symposium on innovations in intelligent systems and applications (INISTA)*, pp. 315–319, 2011.

[20] CarstenA.W. Paasch, *Credit Card Fraud Detection Using Artificial Neural Networks Tuned by Genetic Algorithms (Ph.D. thesis)*.: Hong Kong University of Science and Technology., 2008.

[21] Adebayo O. Adetunmbi and Samuel A. Oluwadare John O. Awoyemi, "Credit card fraud detection using Machine Learning: A Comparative Analysis," *Computing Networking and Informatics (ICCNI)*, pp. 1-9, 2017.

[22] SIVA NAGA PRASAD MANNEM Venkata Ratnam Ganji, "Credit card fraud detection using anti-k nearest neighbor algorithm," *International Journal on Computer Science and Engineering (IJCSE)*, pp. 1035-1039, 2012.

[23] Wen-Fang Yu and Na Wang, "Research on Credit Card Fraud Detection Model Based on Distance Sum," *International Joint Conference on Artificial Intelligence*, pp. 353 - 356, 2009.

[24] Masoumeh Zareapoora and Pourya Shamsolmoali, "Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier," *Procedia Computer Science*, pp. 679-685, 2015. [Online]. https://doi.org/10.1016/j.procs.2015.04.201

[25] V., & Patil, S. Bhusari, "Application of Hidden Markov Model in credit card fraud detection," *International Journal of Distributed and Parallel Systems (IJDPS)*, pp. 203–211, 2011.

[26] Dr. R. C. Thool Avinash Ingole, "Credit Card Fraud Detection Using Hidden Markov Model and Its Performance," *International Journal of Advanced Research in Computer Science and Software Engineering*, pp. 626-632, 2013.

[27] Ayse Buyukkaya and Ilker Elikucuk Ekrem Duman, "A Novel and Successful Credit Card Fraud Detection," *IEEE 13th International Conference on Data Mining Workshops*, pp. 1-10, 2013.

[28] Dominik Olszewski, "Fraud detection using self-organizing map visualizing the user profiles," *Knowledge-Based Systems*, pp. 324–334, 2014. [Online]. http://dx.doi.org/10.1016/j.knosys.2014.07.008

[29] Ashkan Zakaryazad, *PROFIT-DRIVEN NON-LINEAR CLASSIFICATION WITH APPLICATIONS TO CREDIT CARD FRAUD DETECTION, CHURN PREDICTION, DIRECT MARKETING AND CREDIT SCORING*. Istanbul, Turkey: Ozyegin University, 2015.

[30] Djamila Aouada, Aleksandar Stojanovic and Björn Ottersten Alejandro Correa Bahnsen, "Feature engineering strategies for credit card fraud detection," *Expert Systems With Applications*, pp. 134–142, JUN 1 2016. [Online]. http://dx.doi.org/10.1016/j.eswa.2015.12.030

[31] C., & Holte, R. C Drummond, "C4.5, class imbalance, and cost sensitivity: why Under-Sampling beats Over-Sampling," *Workshop on Learning from Imbalanced Datasets II, (ICML), Washington DC*, pp. 1-8, 2003.

[32] Aleksandar Stojanovic, Djamila Aouada and Bjorn Ottersten Alejandro Correa Bahnsen, "Cost Sensitive Credit Card Fraud Detection using Bayes Minimum Risk," *International Conference on Machine Learning and Applications*, pp. 333 - 338, 2013.

[33] Olivier Caelen and Gianluca Bontempi Andrea Dal Pozzolo, "When is Undersampling Effective in Unbalanced Classification Tasks?," *Machine Learning and Knowledge Discovery in Databases*, pp. 200-215, 2015. [Online]. https://doi.org/10.1007/978-3-319-23528-8_13

[34] Mohammed Ibrahim Alowais and Lay-Ki Soon, "Credit Card Fraud Detection: Personalized or Aggregated Model," *International Conference on Mobile, Ubiquitous, and Intelligent Computing*, pp. 114-119, 2012.

[35] Ganesh Kumar.Nune and P.Vasanth Sena, "Novel Artificial Neural Networks and Logistic Approach for Detecting Credit Card Deceit," *International Journal of Computer Science and Network Security (IJCSNS)*, pp. 58-65, 2013.

[36] Cody Stancil, Muyang Sun, Stephen Adams, Peter Beling Gabriel Rushin, "Horse Race Analysis in Credit Card Fraud—Deep Learning, Logistic Regression, and Gradient Boosted Tree," *Systems and Information Engineering Design Symposium (SIEDS)*, pp. 117 - 121, 2017.

[37] Jabir Daud Pathan and Ali Haider Ekbal Ahmed MohdAvesh Zubair Khan, "Credit Card Fraud Detection System Using Hidden Markov Model and K-Clustering," *International Journal of Advanced Research in Computer and Communication Engineering*, pp. 5458-5461, 2014.

[38] Rua-Huan Tsaih and Fang Yu Shin-Ying Huang, "Topological pattern discovery and feature extraction for fraudulent," *Expert Systems with Applications*, pp. 4360–4372, 01/07/2014. [Online]. http://dx.doi.org/10.1016/j.eswa.2014.01.012

[39] Nader Mahmoudi and Ekrem Duman, "Detecting credit card fraud by Modified Fisher Discriminant Analysis," *Expert Systems with Applications*, pp. 2510–2516, APR 1 2015. [Online]. http://dx.doi.org/10.1016/j.eswa.2014.10.037

[40] Cristián Bravo, Olivier Caelen, Tina Eliassi-Rad, Leman Akoglu, Monique Snoeck and Bart Baesens Véronique Van Vlasselaer, "APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions," *Decision Support Systems*, pp. 38–48, 2015. [Online]. https://doi.org/10.1016/j.dss.2015.04.013

[41] Serol Bulkan and Ekrem Duman Yusuf Sahin, "A cost-sensitive decision tree approach for fraud detection," *Expert Systems with Applications*, pp. 5916–5923, 2013. [Online]. http://dx.doi.org/10.1016/j.eswa.2013.05.021

[42] N.Malini and Dr.M.Pushpa, "Analysis on Credit Card Fraud Identification Techniques based on KNN and Outlier Detection," *International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics*, pp. 255 - 258, 2017.

[43] Mitali Bansal and Suman, "Credit Card Fraud Detection Using Self Organised Map," *International Journal of Information & Computation Technology*, pp. 1343-1348, 2014.

[44] S. McAlearney, "TJX Data Breach: Ignore Cost Lessons and Weep," August 07, 2008.

[45] Federal Trade Commission, *Consumer Sentinel Network Data Book*., January-December 2016" March 2017.

[46] The papers insights into payments, "Cross-border Ecommerce Report Turkey," 2014. [Online]. https://www.thepaypers.com/online-fraud-prevention/turkey/19

[47] Soheila Ehramikar, *The Enhancemeat of Credit Card Fraud Detectioa Systems*. Toronto, Canda: Center for Management of Technology and Entrepreneurship, 2000.

[48] Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson, and Gianluca Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification," *IEEE Symposium Series on Computational Intelligence*, pp. 159 - 166, 2015.

[49] Mohd Aizaini Maarof and Anazida Zainal Aisha Abdallah, "Fraud detection system: A survey," *Journal of Network and Computer Applications*, pp. 90–113, June 2016.

[50] Sherly K.K and R Nedunchezhian, "BOAT ADAPTIVE CREDIT CARD FRAUD DETECTION SYSTEM," *Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International Conference on*, pp. 1 - 7, 2010.

[51] Joseph King-Fung Pun, *Improving Credit Card Fraud Detection using a Meta-Learning Strategy*, thesis ed., Chemical Engineering and Applied Chemistry, Ed.: University of Toronto, 2011.

[52] Ian H. Witten and Eibe Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. United States of America: Elsevier, 2005.

[53] Jiawei Han and Micheline Kamber, *Data Mining:Concepts and Techniques second edition*. United States of America: Elsevier, 2006.

[54] Jaume Bacardit, Martin V. Butz, and Xavier Llor`a Hussein A. Abbass, *Online Adaptation in Learning Classifier Systems: Stream Data Mining*. Australia: Australian Defence Force Academy, 2004.

[55] Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy and Abdelhamid Bouchachia João Gama, "A Survey on Concept Drift Adaptation," *ACM Computing Surveys (CSUR)*, pp. 1-44, 2014.

[56] Wei Fan, Philip S. Yu and Jiawei Han Haixun Wang, "Mining Concept-Drifting Data Streams using Ensemble Classifiers," *ACM*, pp. 226–235, 2003.

[57] Karl Tuyls, Bram Vanschoenwinkel and Bernard Manderick Sam Maes, "Credit card fraud detection using Bayesian and neural networks," *Proceedings of the 1st international Naiso Congresson Neuro Fuzzy Technologies*, 2002.

[58] Nathalie Japkowicz and Aleksander Kol cz Nitesh V. Chawla, "Editorial: Special Issue on Learning from Imbalanced Data Sets," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 1-6, 2004.

[59] Taeho Jo and Nathalie Japkowicz, "Class imbalances versus small disjuncts," *ACM SIGKDD Explorations Newsletter - Special issue on learning from imbalanced datasets*, pp. 40-49, 2004.

[60] Gary M. Weiss, "Learning with Rare Cases and Small Disjuncts," *International Conference on Machine Learning*, pp. 558–565, 1995.

[61] Kevin W. Bowyer, Lawrence O. Hall and W. Philip Kegelmeyer Nitesh V. Chawla, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, pp. 321–357, 2002.

[62] Benjamin X. Wang and Nathalie Japkowicz, "Imbalanced data set learning with synthetic samples," *IRIS Machine Learning Workshop*, p. 19, 2004.

[63] Wen-Yuan Wang, and Bing-Huan Mao Hui Han, "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning ," *In Advances in intelligent computing, Springer*, pp. 878 – 887, 2005.

[64] Yang Bai, Edwardo A. Garcia, and Shutao Li Haibo He, "ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning," *International Joint Conference on Neural Networks (IJCNN)*, pp. 1322-1328, 2008.

[65] Andrea Dal Pozzolo, *Adaptive Machine Learning for Credit Card Fraud Detection*, Machine Learning Group, Ed.: Université Libre de Bruxelles, 2015.

[66] Mohamed S. Kamel, Andrew K.C. Wong and Yang Wang Yanmin Suna, "Cost-sensitive boosting for classification of imbalanced data," *Elsevier* , pp. 3358–3378, 2007.

[67] Salvatore J. Stolfo, Junxin Zhang and Philip K. Chan Wei Fan, "AdaCost: Misclassification Cost-sensitive Boosting," *International Conference on Machine Learning (ICML)*, pp. 97-105 , 1999.

[68] Hamed Masnadi-Shirazi and Nuno Vasconcelos, "Risk minimization, probability elicitation, and cost-sensitive svms," *International Conference on Machine Learning (ICML)*, pp. 759–766, 2010. [Online]. http://dblp.uni-trier.de/db/conf/icml/icml2010.html#Masnadi-ShiraziV10

[69] Matjaz Kukar and Igor Kononenko, "Cost-Sensitive Learning with Neural Networks," *European Conference on Artificial Intelligence*, pp. 445-449, 1998.

[70] Qiang Yang, Jianning Wang, and Shichao Zhang Charles X Ling, "Decision trees with minimal costs," *International conference on Machine learning (ICML)*, p. 69, 2004.

[71] Dr Peter Brennan, *A comprehensive survey of methods for overcoming the class imbalance problem in fraud detection*, Dr Markus Hofmann, Ed., 2012.

[72] Charles X. Ling and Victor S. Sheng, "Cost-Sensitive Learning and the Class Imbalance Problem, Springer," *Encyclopedia of Machine Learning*, pp. 1-8, 2008.

[73] W. Fan, A.L. Prodromidis and S.J. Stolfo P.K. Chan, "Distributed data mining in credit card fraud detection," *Intelligent Systems and their Applications, IEEE*, pp. 67 - 74, 1999.

[74] J. Callut and P. Dupont, "F/sub /spl beta// support vector machines," *International Joint Conference on Neural Networks,IEEE*, pp. 1443–1448, 2005.

[75] Lei Wang and Eric Sung Xuchun Il, "AdaBoost with SVM-based component classifiers," *Engineering Applications of Artificial Intelligence*, pp. 785–795, 2008.

[76] Wei Liu and Sanjay Chawla, "Class Confidence Weighted kNN Algorithms for Imbalanced Data Sets," *In Advances in Knowledge Discovery and Data Mining, Springer*, pp. 345–356, 2011.

[77] Constantinos S. Hilas and John N. Sahalos, "An Application of Decision Trees for Rule Extraction Towards Telecommunications Fraud Detection," *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, 2007. [Online]. https://doi.org/10.1007/978-3-540-74827-4_139

[78] Qibei Lu and Chunhua Ju, "Research on Credit Card Fraud Detection Model Based on Class Weighted Support Vector Machine," *Journal of Convergence Information Technology*, pp. 62-68, 2011.

[79] M. N. Murty and P. J. Flynn A. K. Jain, "Data clustering: a review," *ACM Computing Surveys (CSUR)*, pp. 264-323, 1999.

[80] Tom M. Mitchell, "The Need for Biases in Learning Generalizations," *Technical Report CMB*, pp. 1-4, 1980.

[81] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques ," *Informatica (SCRIBD)*, pp. 249-268, 2007.

[82] Leo Breiman, "Random Forests," *Manufactured in The Netherlands*, pp. 5–32, 2001.

[83] DAN GEIGER and DAVID M. CHICKERING DAVID HECKERMAN, "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data ," *Machine Learning*, pp. 197-243 , 1995.

[84] Gregory F. Cooper and Edward Herskovits, "A Bayesian method for the induction of probabilistic networks from data ," *Machine Learning, Springer*, pp. 309–347, 1992.

[85] George H. John and Pat Langley, "Estimating Continuous Distributions in Bayesian Classifiers," *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pp. 338-348, 1995.

[86] Peter Clark and Tim Niblett, "The CN2 Induction Algorithm ," *Machine Learnin*, pp. 261-283, 1989.

[87] Daniel Grossman and Pedro Domingos, "Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood," *International Conference on Machine Learning*, 2004.

[88] Vladimir N. Vapnik, *Statistical Learning Theory*., 1998.

[89] Corinna Cortes and Vladimir Vapnik, "Support-Vector Networks," *Machine Learning* , pp. 273-297, 1995.

[90] A. Dhar and K. Buescher V. Hanagandi, "Density-based clustering and radial basis function modeling to generate credit card fraud scores," *Computational Intelligence for Financial Engineering*, 1996.

[91] Frank. Rosenblatt, *Principles of neurodynamics; perceptrons and the theory of brain mechanisms*. Washington, United States of America : Washington, Spartan Books [1962], 1962.

[92] Geoffrey E. Hinton and Ronald J. Williams David E. Rumelhart, *LEARNING INTERNAL REPRESENTATIONS BY ERROR PROPAGATION*. CALIFORNIA: ICS Report 8506, 1986.

[93] Tom M. Mitchell, *Machine Learning*.: McGraw-Hill Science/Engineering/Math, 1997.

[94] Dennis Kibler and Marc K. Albert David W. Aha, "Instance-based learning algorithms," *Machine Learning*, pp. 37-66, 1991.

[95] S. Le Cessie and J. C. Van Houwelingen, "Ridge Estimators in Logistic Regression," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, pp. 191-201, 1992.

[96] Douglas C. Montgomery and George C. Runger, *Applied Statistics and Probability for Engineers*.: Wiley Books by These Authors, 2003.

[97] J. Ross Quinlan, *C4.5: programs for machine learning*.: Morgan Kaufmann , 1993.

[98] "Anaconda3,". [Online]. https://www.anaconda.com/what-is-anaconda/

[99] Jerzy Stefanowski, "Overlapping, Rare Examples and Class Decomposition in Learning Classifiers from Imbalanced Data," *Emerging Paradigms in Machine Learning*, pp. 277-306, 2013.

[100] "Naive Bayes," sklearn.naive_bayes,. [Online]. http://scikit-learn.org/stable/modules/naive_bayes.html#gaussian-naive-bayes

[101] Lei Tang and Huan Liu Payam Refaeilzadeh, "Cross-Validation," *Encyclopedia of Database Systems*, pp. 532-538, 2009. [Online]. https://doi.org/10.1007/978-0-387-39940-9_565

[102] Oded Maimon and Lior Rokach, "Data Mining and Knowledge Discovery Handbook," *Database Management & Information Retrieval*, pp. 875 - 886, 2010.

[103] J. Huang, H. Zhang C.X. Ling, "AUC: a Statistically Consistent and more Discriminating Measure than Accuracy," *International Joint Conferences on Artificial Intelligence*, pp. 519–526, 2003.

[104] Reid Johnson, Olivier Caelen, Serge Waterschoot, Nitesh V Chawla and Gianluca Bontempi Andrea Dal Pozzolo, "Using HDDT to avoid instances propagation in unbalanced and evolving data streams," *International Joint Conference on Neural Network (IJCNN)*, pp. 588–594, 2014.