7-2022

Credit Card Fraud Detection Using Machine Learning

Meera AlEmad
mma1930@rit.edu

Credit Card Fraud Detection Using Machine Learning

by

Meera AlEmad

A Capstone Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Professional Studies: Data Analytics

Department of Graduate Programs & Research

Rochester Institute of Technology

RIT Dubai

July 2022

RIT

Master of Science in Professional Studies:

Data Analytics

Graduate Capstone Approval

Student Name**: Meera AlEmad**

Graduate Capstone Title**: Credit Card Fraud Detection Using Machine Learning**

Graduate Capstone Committee:

Name: Dr. Sanjay Modak **Date:**

Chair of committee

Name: Dr. Ehsan Warriach **Date:**

Member of committee

Acknowledgments

By the end of the master's journey, I would like to first thank Allah for granting me with the opportunity of going through and completing my study, I would also like to thank my family, friends and my colleagues who have supported and encouraged me. A special thanks and gratitude would also go to my mentor Dr. Ehsan Warriach who has advised me through the capstone project and helped in improving it.

Abstract

The purpose of this project is to detect the fraudulent transactions made by credit cards by the use of machine learning techniques, to stop fraudsters from the unauthorized usage of customers' accounts. The increase of credit card fraud is growing rapidly worldwide, which is the reason actions should be taken to stop fraudsters. Putting a limit for those actions would have a positive impact on the customers as their money would be recovered and retrieved back into their accounts and they won't be charged for items or services that were not purchased by them which is the main goal of the project. Detection of the fraudulent transactions will be made by using three machine learning techniques KNN, SVM and Logistic Regression, those models will be used on a credit card transaction dataset.

Keywords*: Credit Card Fraud Detection, Fraud Detection, Fraudulent Transactions, K-Nearest Neighbors, Support Vector Machine, Logistic Regression, NaïveBayes*

Table of Contents

List of Figures

List of Tables

Chapter 1

1.1 Introduction

With the increase of people using credit cards in their daily lives, credit card companies should take special care in the security and safety of the customers. According to (Credit card statistics 2021) the number of people using credit cards around the world was 2.8 billion in 2019, in addition 70% of those users own a single card at least.

Reports of Credit card fraud in the US rose by 44.7% from 271,927 in 2019 to 393,207 reports in 2020. There are two kinds of credit card fraud, the first one is by having a credit card account opened under your name by an identity thief, reports of this fraudulent behavior increased 48% from 2019 to 2020. The second type is by an identity thief uses an existing account that you created, and it's usually done by stealing the information of the credit card, reports on this type of fraud increased 9% from 2019 to 2020 (Daly, 2021). Those statistics caught my attention as the numbers are increasing drastically and rapidly throughout the years, which gave me the motive to try to resolve the issue analytically by using different machine learning methods to detect the credit card fraudulent transactions within numerous transactions.

1.2 Project goals

The main aim of this project is the detection of credit card fraudulent transactions, as it's important to figure out the fraudulent transactions so that customers don't get charged for the purchase of products that they didn't buy. The detection of the credit card fraudulent transactions will be performed with multiple ML techniques then a comparison will be made between the outcomes and results of each technique to find the best and most suited model in the detection of the credit card transaction that are fraudulent, graphs and numbers will be provided as well. In addition, exploring previous literatures and different techniques used to distinguish the fraud within a dataset.

Research question: What is the most suited machine learning model in the detection of fraudulent credit card transactions?

1.3 Research Methodology

1.4.1 CRISP-DM

I believe that taking the route of CRISP-DM will ease obtaining efficient and elite results, as it takes the project into the whole journey, starting by understanding the business and data, preparing the data then modeling it and finally evaluate the model to make sure it's performing well.

Phase 1: Business Understanding

As stated before credit card fraud is increasing drastically every year, many people are facing the problem of having their credits breached by those fraudulent people, which is impacting their daily lives, as payments using a credit card is similar to taking a loan. If the problem is not solved many people will have large amounts of loans that they cannot pay back which will make them face a hard life, and they won't be able to afford necessary products, in the long run not being able to pay back the amount might lead to them going to jail. Basically, the problem proposed is the detection of the credit card fraudulent transactions made by fraudsters to stop those breaches and to ensure customers security.

Business Objective: Identification of fraudulent transaction to prohibit deduction from effected customers' accounts.

Phase 2: Data Understanding

In the Data understanding phase, it was critical to obtain a high-quality dataset as the model is based on it, the dataset was explored by taking a closer look into it which gave the knowledge needed to confirm the quality of the dataset, additionally to reading the description of the whole dataset and each attribute. It's also important to have a dataset that contains several mixed transaction types "Fraudulent and real" and a class to clarify the type of transaction, finally, identifiers to clarify the reason behind the classification of

the transaction type. I made sure to follow all of those points during the search for the most suited dataset.

Phase 3: Data Preparation

After choosing the most suited dataset the preparation phase begins, the preparation of the dataset includes selecting the wanted attributes or variables, cleaning it by excluding Null rows, deleting duplicated variables, treating outlier if necessary, in addition to transforming data types to the wanted type, data merging can be performed as well where two or more attributes get merged. All those alterations lead to the wanted result which is to make the data ready to be modeled.

The dataset chosen for this project didn't need to go through all of the alterations mentioned earlier, as there were no missing nor duplicated variables, there was no merging needed as well. But there was some changing in the types of the data to be able to create graphs, in addition to using the application Sublime Text to be able to insert the data into Weka and perform analysis, as it needed to be altered.

Phase 4: Modeling

Four machine learning models were created in the modeling phase, KNN, SVM, Logistic Regression and Naïve Bayes. A comparison of the results will be presented later in the paper to know which technique is most suited in the credit card fraudulent transactions detection. The dataset is sectioned into a ratio of 70:30, the training set will be the 70% and remaining set will be the testing set which is the 30%. The four models were created using Weka and only two in R, KNN and Naïve Bayes. Visualizations will be provided from both tools.

Phase 5: Evaluation and Deployment

The final phase will show evaluations of the models by presenting their efficiency, the accuracies of the models will be presented in addition to any comment observed, to find the best and most suited model for detecting the fraud transactions made by credit card.

Chapter 2: Literature Review

2.1 Introduction

It is essential for credit card companies to establish credit card transactions that fraudulent from transactions that are non-fraudulent, so that their customers' accounts won't get affected and charged for products that the customers didn't buy (Maniraj et al., 2019). There are many financial Companies and institutions that lose massive amounts of money because of fraud and fraudsters that are seeking different approaches continuously to violate the rules and commit illegal actions; therefore, systems of fraud detection are essential for all banks that issue credit cards to decrease their losses (Zareapoor et al., 2012). There are multiple methods used to detect fraudulent behaviors such as Neural Network (NN), Decision Trees, K-Nearest Neighbor algorithms, and Support Vector Machines (SVM). Those ML methods can either be applied independently or can be used collectively with the addition of ensemble or meta-learning techniques to develop classifiers (Zareapoor et al., 2012).

2.2 Literature Review

Zareapoor and his research team used multiple techniques to determine the best performing model in detecting fraudulent transactions, which was established using the accuracy of the model, the speed in detecting and the cost. The models used were Neural Network, Bayesian Network, SVM, KNN and more. The comparison table provided in the research paper showed that Bayesian Network was very fast in finding the transactions that are fraudulent, with high accuracy. The NN performed well as well as the detection was fast, with a medium accuracy. KNN's speed was good with a medium accuracy, and finally SVM scored one of the lower scores, as the speed was low, and the accuracy was medium. As for the cost All models built were expansive (Zareapoor et al., 2012).

The model used by Alenzi and Aljehane to detect fraud in credit cards was Logistic Regression, their model scored 97.2% in accuracy, 97% sensitivity and 2.8% Error Rate. A comparison was performed between their model and two other classifier which are

Voting Classifier and KNN. VC scored 90% in accuracy, 88% sensitivity and 10% error rate, as for KNN where k = 1:10, the accuracy of the model was 93%, the sensitivity 94% and 7% for the error rate (Alenzi & Aljehane, 2020).

Manirajs team built a model that can recognize if any new transaction is fraud or non-fraud, their goal was to get 100% in the detection of fraudulent transactions in addition to trying to minimize the incorrectly classified fraud instances. Their model has performed well as they were able to get 99.7% of the fraudulent transactions (Maniraj et al., 2019).

The classification approach used by Dheepa and Dhanapal was the behavior-based classification approach, by using Support Vector Machine, where the behavioral patterns of the customers were analyzed to distinguish credit card fraud, such as the amount, date, time, place, and frequency of card usage. The accuracy achieved by their approach was more than 80% (Dheepa & Dhanapal, 2012).

Mailini and Pushpa proposed using KNN and Outlier detection in identifying credit card fraud, the authors found after performing their model over sampled data, that the most suited method in detecting and determining target instance anomaly is KNN which showed that its most suited in the detection of fraud with the memory limitation. As for Outlier detection the computation and memory required for the credit card fraud detection is much less in addition to its working faster and better in online large datasets. But their work and results showed that KNN was more accurate and efficient (Malini & Pushpa, 2017).

Maes and his team proposed using Bayesian and Neural Network in the credit card fraud detection. Their results showed that Bayesian performance is 8% more effective in detecting fraud than ANN, which means that in some cases BBN detects 8% more of the fraudulent transactions. In addition to the Learning times, ANN can go up to several hours whereas BBN takes only 20 minutes (Maes et al., 2002).

The team of Awoyemi compared the usage of three ML techniques in the detection of credit card fraud, the first is KNN, the second is Naïve Bayes and the third is Logistic Regression. They sampled different distributions to view the various outcomes. The top Accuracy of the 10:90 distribution is Naïve Bayes with 97.5%, then KNN with 97.1%,

Logistic regression performed poorly as the accuracy is 36.4%. Another distribution that was viewed is 34:66, KNN topped the chart with a slight increase in the accuracy 97.9%, then Naïve Bayes with 97.6%, Logistic Regression performed better in this distribution as the accuracy raised to 54.8% (Awoyemi et al., 2017).

Jain's team used several ML techniques to distinguish credit card fraud, three of them are SVM, ANN and KNN. Then to compare the outcome of each model, they calculated the true positive (TP), false negative (FN), false positive (FP), and true negative (TN) generated. ANN scored 99.71% accuracy, 99.68% precision, and 0.12% false alarm rate. SVM accuracy is 94.65%, 85.45% for the precision, and 5.2% false alarm rate. and finally, the accuracy of KNN is 97.15%, precision is 96.84% and the false alarm rate is 2.88% (Jain et al., 2019).

Gupta's team worked on implementing an automated model that uses various ML techniques to detect fraudulent instances that are related economically to users but is specializing more in credit card transactions, according to Gupta and his team Out of all the techniques that they used Naïve Bayes had an outstanding performance in distinguishing fraudulent transactions as the accuracy of it was 80.4% and the area under the curve is 96.3% (Gupta et al., 2021).

Adepoju and his team used all of the ML methods that are used in this paper, Logistic Regression , (SVM) Support Vector Machine, Naive Bayes, and (KNN) K-Nearest Neighbor, those methods were used on distorted credit card fraud data. The accuracies scored by all the models were 99.07% for Logistic Regression, Naïve Bayes scored 95.98%, 96.91% for K-nearest neighbor, and the last model (SVM) Support Vector Machine scored 97.53% (Adepoju et al., 2019).

Safa and Ganga investigated how well Logistic Regression, (KNN) K-nearest neighbor, and Naïve Bayes work on exceptionally distorted credit card dataset, they implanted their work on Python where the best method was selected using evaluation. The accuracies result of their model for Naïve Bayes is 83%, 97.69% for Logistic regression and in last place K-nearest neighbor with 54.86% (Safa & Ganga, 2019).

The team of Varmedja used multiple machine learning algorithms in their paper such as Logistic Regression, Multilayer Perception, Random Forest, and Naïve Bayes. As the dataset was quite very unbalanced Varmedja and his team SMOTE technique to oversample, feature selection, in addition to sectioning the data into a training section and a testing data section. The best scoring model during the experiment is Random Forest with 99.96%, with not many difference the model in second place is Multilayer Perceptron with 99.93%, in third place is Naïve bayes with 99.23% and in last place is Logistic regression with 97.46% (Varmedja et al., 2019).

The system to detect credit card fraud that was introduced by Sailusha and his team to detect fraudulent activities. The algorithms used in their model is adaboost and Random Forest, which scored the accuracy 93.99% and the accuracy of adaboost is 99.90% which shows that it did better than Random Forest in term of accuracy (Sailusha et al.).

The paper of Kiran and his team presents Naïve Bayes (NB) improved (KNN) K-Nearest Neighbor method for Fraud Detection of Credit Card which is (NBKNN) in short format. The outcome of the experiment illustrates the difference in the process of each classifier on the same dataset. Naïve bayes performed better than K-nearest neighbor as it scored an accuracy of 95% while KNN scored 90% (Kiran et al., 2018).

Najdat and his team's approach in detecting fraudulent transactions is (BiLSTM) BiLSTM-MaxPooling-BiGRU- MaxPooling, this approach is established upon bidirectional Long short-term memory in addition to (BiGRU) bidirectional Gated recurrent unit. In addition, the group decided to go for six ML classifiers, which are Voting, Adaboost, Random Forest, Decision Tree, Naïve bayes, and Logistic Regression. K-nearest neighbor scored an accuracy of 99.13%, and logistic regression scored 96.27%, Decision tree scored 96.40% and Naïve bayes scored 96.98% (Najadat et al., 2020).

The paper of Saheed and his group focuses on detection of Credit Card Fraud with the use of (GA) Genetic Algorithm as a feature selection technique. In feature selection the data is splitted in two parts first priority features and second priority features, and the ML techniques that the group used are The Naïve Bayes (NB), Random Forest (RF) and (SVM) Support Vector Machine. Naïve bayes scored 94.3%, SVM scored 96.3%, and Random Forest scored 96.40% which is the highest accuracy (Saheed et al., 2020).

The work of Itoo and his group uses three different ML methods the first is logistic regression, the second is Naïve bayes and the last one is K-nearest neighbors. Itoo and his group recorded the work and comparative analysis, their work is implemented on python. Logistic regression accuracy is 91.2%, Naïve bayes accuracy is 85.4% and K-nearest neighbor is last with an accuracy of 66.9% (Itoo et al., 2020).

The team of Tanouz proposed working on various ML based classification algorithms, like Naïve Bayes, Logistic Regression, Random Forest, and Decision Tree in handling datasets that are strongly imbalanced, in addition their research will have the calculations of five measures the first is accuracy, the second is precision, the third is recall, the fourth is confusion matrix, and the last one is Roc-auc score. 95.16% is the score of both Logistic Regression and Naïve Bayes, 96.77% is the score for random forest, for the last model Decision Tree scored 91.12% (Tanouz et al., 2021).

Dighe and his team used KNN, Naïve Bayes, Logistic Regression and Neural Network, Multi-Layers Perceptron and Decision Tree in their work, then evaluated the results in terms of numerous accuracy metrics. Out of all the models created the best performing one is KNN which scored 99.13%, then in second place Naïve Bayes which scored 96.98%, the third best performing model 96.40% and in last place is logistic regression with 96.27% (Dighe et al., 2018).

The paper of Bhanusri and his team implemented multiple ML techniques on an unbalanced dataset. The ML methods used are logistic regression, naïve bayes, and random forest to explain the relation of fraud and credit card. Their conclusion of the project presents the best classifier by training and testing supervised techniques in term of their work. The logistic regression model scored 99.8% accuracy, random forest scored 100% and 90.8% is scored by naïve bayes.

Sahin and Duman used four Support Vector Machine methods in detecting credit card fraud. SVM) Support Vector Machine with RBF, Polynomial, Sigmoid, and Linear Kernel, all models scored 99.87% in the training model and 83.02% in the testing part of the model (Sahin & Duman, 2011).

2.3 Literature Review Conclusion

Throughout the search I found that there were many models created by other researchers which have proven that people have been trying to solve the credit card fraud problem. I found that Najdat Team used an approach that is established upon bidirectional long/short-term memory in building their model, other researchers have tried different data splitting ratios to generate different accuracies. The team of Sahin and Duman used different Support Vector Machine methods which are (SVM) Support Vector Machine with RBF, Polynomial, Sigmoid, and Linear Kernel.

The lowest accuracy of the four models that will be studied in this research, is 54.86% for KNN and 36.40% for logistic Regression which were scored by Awoyemi and his team, as for Naïve Bayes the lowest accuracy was scored by Gupta and his team which is 80.4% and finally, SVM the lowest score was 94.65% and it was scored by Jain's team. To determine the best model out of the four models that will be studied through the research, the average of the best three accuracies of each model will be calculated, the average of the accuracy of KNN is 98.72%, the average of logistic regression is 98.11%, 98.85% for Naïve bayes and 96.16% for Support Vector Machine. So, for the best performing credit card fraud detecting model within the Literature review is the Logistic Regression model.

Chapter 3: Project Description

3.1 Introduction

In order to accomplish the objective and goal of the project which is to find the most suited model to detect credit card fraud several steps need to be taken. Finding the most suited data and preparing/preprocessing are the first and second steps, after making sure that the data is ready the modeling phase starts, where 4 models are created, K-Nearest Neighbor (KNN) , Naïve Bayes, SVM and the last one is Logistic Regression. In the KNN model two Ks were chosen K=3 and K=7. All models were created in both R and Weka programs expect SVM which was created in Weka only, in addition all visualizations are taken from both applications.

3.2 Data Source

The dataset was retrieved from an open-source website, Kaggle.com. it contains data of transactions that were made in 2013 by credit card users in Europe, in two days only. The dataset consists of 31 attributes, 284,808 rows. 28 attributes are numeric variables that due to confidentiality and privacy of the customers have been transformed using PCA transformation, the three remaining attributes are “Time” which contains the elapsed seconds between the first and other transactions of each attribute, “Amount” is the amount of each transaction, and the final attribute “Class” which contains binary variables where “1” is a case of fraudulent transaction, and “0” is not as case of fraudulent transaction.

Dataset Link: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

Chapter 4: Data Analysis

4.1 Data Preparation

The first figure bellow shows the structure of the dataset where all attributes are shown, with their type, in addition to glimpse of the variables within each attribute, as shown at the end of the figure the Class type is integer which I needed to change to factor and identify the 0 as Not Fraud and the 1 as Fraud to ease the process of creating the model and obtain visualizations

```
'data.frame':   284807 obs. of  31 variables:
 $ Time  : num  0 0 1 1 2 2 4 7 7 9 ...
 $ V1    : num  -1.36 1.192 -1.358 -0.966 -1.158 ...
 $ V2    : num  -0.0728 0.2662 -1.3402 -0.1852 0.8777 ...
 $ V3    : num  2.536 0.166 1.773 1.793 1.549 ...
 $ V4    : num  1.378 0.448 0.38 -0.863 0.403 ...
 $ V5    : num  -0.3383 0.06 -0.5032 -0.0103 -0.4072 ...
 $ V6    : num  0.4624 -0.0824 1.8005 1.2472 0.0959 ...
 $ V7    : num  0.2396 -0.0788 0.7915 0.2376 0.5929 ...
 $ V8    : num  0.0987 0.0851 0.2477 0.3774 -0.2705 ...
 $ V9    : num  0.364 -0.255 -1.515 -1.387 0.818 ...
 $ V10   : num  0.0908 -0.167 0.2076 -0.055 0.7531 ...
 $ V11   : num  -0.552 1.613 0.625 -0.226 -0.823 ...
 $ V12   : num  -0.6178 1.0652 0.0661 0.1782 0.5382 ...
 $ V13   : num  -0.991 0.489 0.717 0.508 1.346 ...
 $ V14   : num  -0.311 -0.144 -0.166 -0.288 -1.12 ...
 $ V15   : num  1.468 0.636 2.346 -0.631 0.175 ...
 $ V16   : num  -0.47 0.464 -2.89 -1.06 -0.451 ...
 $ V17   : num  0.208 -0.115 1.11 -0.684 -0.237 ...
 $ V18   : num  0.0258 -0.1834 -0.1214 1.9658 -0.0382 ...
 $ V19   : num  0.404 -0.146 -2.262 -1.233 0.803 ...
 $ V20   : num  0.2514 -0.0691 0.525 -0.208 0.4085 ...
 $ V21   : num  -0.01831 -0.22578 0.248 -0.1083 -0.00943 ...
 $ V22   : num  0.27784 -0.63867 0.77168 0.00527 0.79828 ...
 $ V23   : num  -0.11 0.101 0.909 -0.19 -0.137 ...
 $ V24   : num  0.0669 -0.3398 -0.6893 -1.1756 0.1413 ...
 $ V25   : num  0.129 0.167 -0.328 0.647 -0.206 ...
 $ V26   : num  -0.189 0.126 -0.139 -0.222 0.502 ...
 $ V27   : num  0.13356 -0.00898 -0.05535 0.06272 0.21942 ...
 $ V28   : num  -0.0211 0.0147 -0.0598 0.0615 0.2152 ...
 $ Amount: num  149.62 2.69 378.66 123.5 69.99 ...
 $ Class : int  0 0 0 0 0 0 0 0 0 0 ...
```

Figure 1 - Dataset Structure

The second figure shows the distribution of the class, the red bar which contains 284,315 variables represents the non-fraudulent transactions, and the blue bar with 492 variables represents the fraudulent transactions.

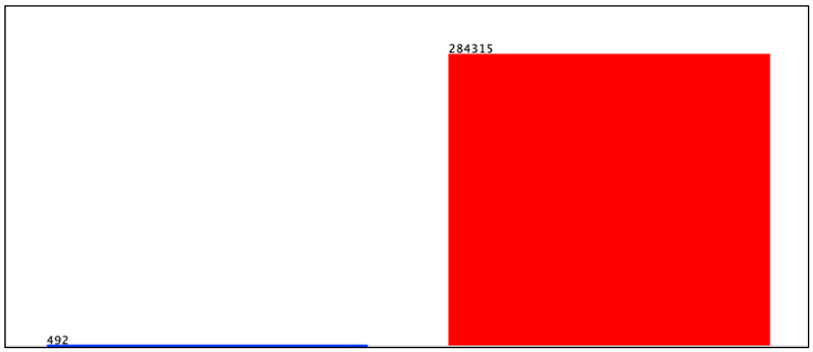

Figure 2 - Class Distribution

4.1.1 Correlation between attributes “Image from R”

The correlations between all the of the attributes within the dataset are presented in the figure below.

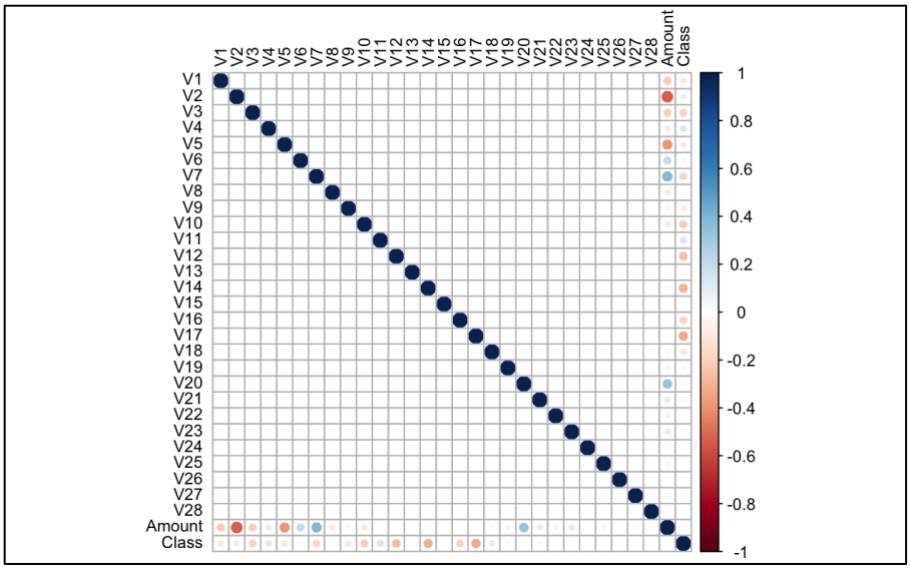

Figure 3 - Correlations

4.1.2 Attribute with the most fraud

Figure 4 below shows attribute 18 the attribute with the most credit card fraudulent transactions, the blue line represents the variable 1 which is the fraudulent transactions.

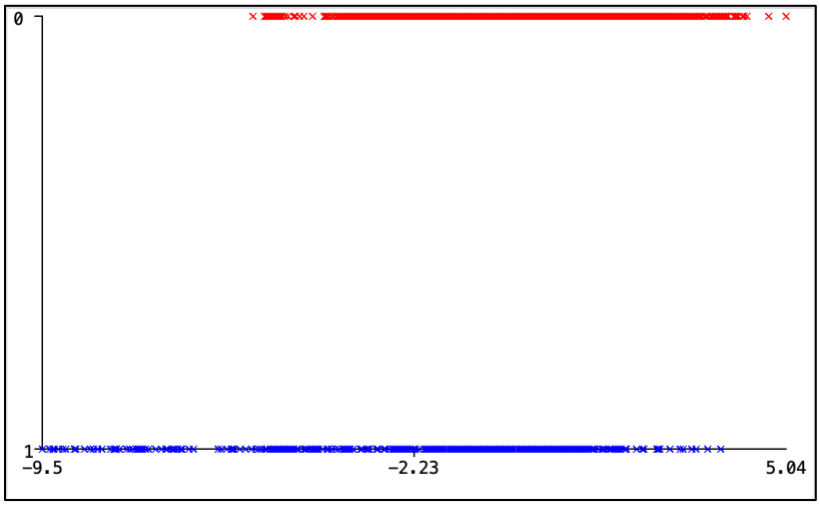

Figure 4 – Variable 18

4.1.3 Attribute with the less fraud

The figure below shows the variable that have the lowest number of fraudulent transactions, as mentioned earlier the blue line represents the fraudulent instances within the dataset.

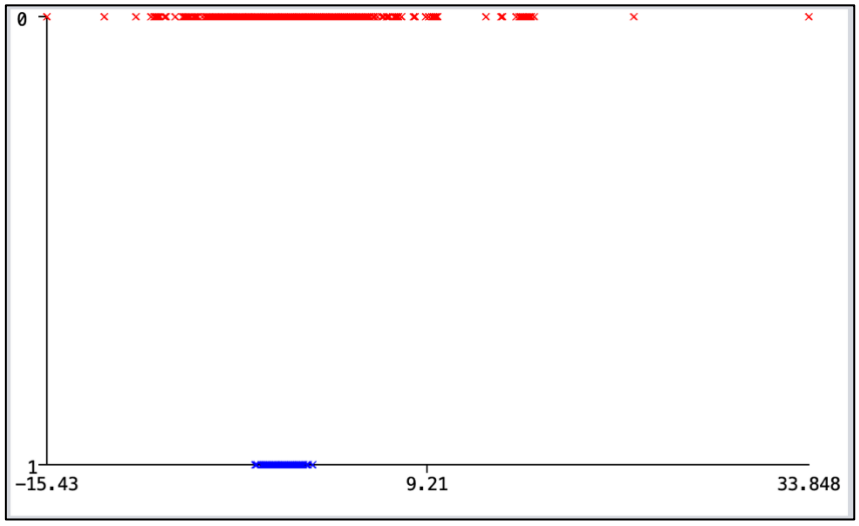

Figure 5 - Variable 28

4.2 Data Preprocessing

As there are no NAs nor duplicated variables, the preparation of the dataset was simple the first alteration that was made to be able to open the dataset on Weka program is changing the type of the class attribute from Numeric to Class and identify the class as {1,0} using the program Sublime Text. Another alteration was made on the type as well on the R program to be able to create the model and the visualization.

4.3 Data Modeling

After making sure that the data is ready to get modeled the four models were created using both Weka and R. the model SVM was created using Weka only, as for KNN, Logistic Regression and NaïveBayes they were created using R and Weka.

4.3.1 KNN

The K-Nearest Neighbor algorithm (KNN) is a supervised ML technique that can be applied in both scenario instances, classification instances along with regression instances (Mahesh, 2020).To figure the best KNN model two Ks where used K=3 and K=7, both are presented with figures from both Weka and R.

- K = 3

During the making of the KNN model, I decided to create two models where K=3 and K=7. Figure 5 shows the model created in R, the model scored an accuracy of 99.83% and managed to correctly identify 91,719 transactions and missed 155. As for the Weka program the model scored 99.94% for the accuracy and miss-classified 52 transactions. As there are different accuracies the average of the accuracies is 99.89%.

```
Correctly Classified Instances       85390               99.9391 %
Incorrectly Classified Instances        52                0.0609 %
Kappa statistic                          0.833
Mean absolute error                      0.0008
Root mean squared error                  0.024
Relative absolute error                 21.9704 %
Root relative squared error             53.2988 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.747    0.000    0.942      0.747    0.833      0.839    0.911     0.779     1
                 1.000    0.253    0.999      1.000    1.000      0.839    0.911     1.000     0
Weighted Avg.    0.999    0.252    0.999      0.999    0.999      0.839    0.911     0.999

=== Confusion Matrix ===

     a     b   <-- classified as
   130    44 |     a = 1
     8 85260 |     b = 0
```

Figure 6 - Weka K=3

```
Confusion Matrix and Statistics

                Reference
Prediction       Not Fraudulent Fraudulent
  Not Fraudulent          91702        155
  Fraudulent                  0         17

               Accuracy : 0.9983
```

Figure 7 - RStudio K=3

- K = 7

There was a slight decrease in the accuracy in the model created in R (Figure 6) as it scored 99.82% when K is 7, and the model miss classified 166 fraudulent transactions as nonfraudulent. As for Weka (Figure 7) the accuracy is the same as K=3 99.94% with 52 misclassified transactions, the only difference is within the classifications. The average of the accuracies is 99.88%

```
Confusion Matrix and Statistics

                Reference
Prediction       Not Fraudulent Fraudulent
  Not Fraudulent          91702        166
  Fraudulent                  0          6

               Accuracy : 0.9982
```

Figure 8 - RStudio K=7

```
Correctly Classified Instances       85390               99.9391 %
Incorrectly Classified Instances        52                0.0609 %
Kappa statistic                          0.8351
Mean absolute error                      0.0008
Root mean squared error                  0.0235
Relative absolute error                 22.256  %
Root relative squared error             52.1008 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.759    0.000    0.930      0.759    0.835      0.839    0.917     0.795     1
                 1.000    0.241    1.000      1.000    1.000      0.839    0.917     1.000     0
Weighted Avg.    0.999    0.241    0.999      0.999    0.999      0.839    0.917     0.999

=== Confusion Matrix ===

     a     b   <-- classified as
   132    42 |     a = 1
    10 85258 |     b = 0
```

Figure 9 - Weka K=7

4.3.2 Naïve Bayes

Naïve Bayes is a classification algorithm that consider the being of a certain trait within a class is unrelated to the being of any different feature, the main use of it is for clustering and classifications, depending on the conditional probability of happening (Mahesh, 2020).

The second model created by R is Naïve Bayes, figure 9 shows the performance of the model, it scored an accuracy of 97.77% and misclassified a total of 2,051 transactions, 33 fraudulent as nonfraudulent and 2018 nonfraudulent as fraudulent. There is a slight difference in the accuracy of the Naïve bayes model created within Weka as its 97.73% and the misclassification instances are 1,938.

```
Correctly Classified Instances       83504               97.7318 %
Incorrectly Classified Instances      1938                2.2682 %
Kappa statistic                          0.1292
Mean absolute error                      0.0227
Root mean squared error                  0.1491
Relative absolute error                626.539  %
Root relative squared error            330.6127 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.851    0.022    0.072      0.851    0.132      0.243    0.968     0.091     1
                 0.978    0.149    1.000      0.978    0.989      0.243    0.964     1.000     0
Weighted Avg.    0.977    0.149    0.998      0.977    0.987      0.243    0.964     0.998

=== Confusion Matrix ===

     a     b   <-- classified as
   148    26 |     a = 1
  1912 83356 |     b = 0
```

Figure 10 - Weka Naïve Bayes

```
Confusion Matrix and Statistics

                Reference
Prediction       Not Fraudulent Fraudulent
  Not Fraudulent          89684         33
  Fraudulent               2018        139

               Accuracy : 0.9777
```

Figure 11 - RStudio Naïve Bayes

4.3.3 Logistic Regression

Logistic Regression model is statical model where evaluations are formed of the connection among dependent qualitative variable (binary or binomial logistic regression) or variable with three values or higher (multinomial logistic regression) and one independent explanatory variable or higher whether qualitative or quantitative (Domínguez-Almendros et al., 2011).

The last model created using both R and Weka is Logistic Regression, the model managed to score and accuracy of 99.92% in R (figure 11) with 70 misclassified instances, while it scored 99.91% in Weka with 77 misclassified instances as presented in figure 10.

```
Correctly Classified Instances       85365               99.9099 %
Incorrectly Classified Instances        77                0.0901 %
Kappa statistic                          0.7256
Mean absolute error                      0.0014
Root mean squared error                  0.0276
Relative absolute error                 38.6516 %
Root relative squared error             61.1796 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.586    0.000    0.953      0.586    0.726      0.747    0.976     0.831     1
                 1.000    0.414    0.999      1.000    1.000      0.747    0.976     1.000     0
Weighted Avg.    0.999    0.413    0.999      0.999    0.999      0.747    0.976     1.000

=== Confusion Matrix ===

     a     b   <-- classified as
   102    72 |     a = 1
     5 85263 |     b = 0
```

Figure 12 - Weka Logistic Regression

```
   FALSE  TRUE
0 91693     9
1    61   111

[1] 99.92381
```

Figure 13 - RStudio Logistic Regression

4.3.4 Support Vector Machine

Support Vector machine is a supervised ML technique with connected learning algorithms which inspect data used for both classification and regression analyses, it also performs linear classification, additionally to non-linear classification by creating margins between the classes, which are created in such a fashion that the space between the margin and the classes is maximum which minimizes the error of the classification (Mahesh, 2020).

Finally, the model Support Vector Machine as show in figure 12 managed to score 99.94% for the accuracy and misclassified 51 instances.

```
Correctly Classified Instances       85391               99.9403 %
Incorrectly Classified Instances        51                0.0597 %
Kappa statistic                          0.8388
Mean absolute error                      0.0006
Root mean squared error                  0.0244
Relative absolute error                 16.4433 %
Root relative squared error             54.1917 %
Total Number of Instances            85442

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.764    0.000    0.930      0.764    0.839      0.843    0.882     0.711     1
                 1.000    0.236    1.000      1.000    1.000      0.843    0.882     1.000     0
Weighted Avg.    0.999    0.235    0.999      0.999    0.999      0.843    0.882     0.999

=== Confusion Matrix ===

     a     b   <-- classified as
   133    41 |     a = 1
    10 85258 |     b = 0
```

Figure 14 - Support Vector Machine

4.4 Evaluation and Deployment

The last stage of the CRISP-DM model is the evaluation and deployment stage, as presented in table 2 below all models are being compared to each other to figure the best model in identifying fraudulent credit card transactions.

Accuracy is the overall number of instances that are predicted correctly, accuracies are represented by confusion matrix where it showed the True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). True Positive represents the transactions that are fraudulent and was correctly classified by the model as fraudulent. True Negative represents the not fraudulent transactions that were correctly predicted by the model as Not fraudulent. The third rating is False positive which represents the transaction that are fraudulent but was misclassified as not fraudulent. And finally False Negative which are the not fraudulent transactions that were identified as fraudulent, table 1 below shows the confusion matrix.

Actual/Predicted	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Table 1 - Confusion Matrix

The table above shows all the components to calculate an accuracy of a model which is displayed in the below equation.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Model		Accuracy
KNN	K = 3	99.89%
	K = 3	
	K = 7	99.88%
	K = 7	
Naïve Bayes	Naïve Bayes	97.76%
	Naïve Bayes	
Logistic Regression	Logistic Regression	99.92%
	Logistic Regression	
Support Vector Machine	SVM	99.94%

Table 2 - Table of Accuracies

Table 2 shows all of the accuracies of all the models that were created in the project, all models performed well in detecting fraudulent transactions and managed to score high accuracies. Out of all the models the model that scored the best is Support Vector Machine as its accuracy is 99.94%, the second best is Logistic Regression, then in third place is KNN as both Ks scored similar accuracies, and the model that scored the lowest accuracy out of all models is Naïve Bayes with a score of 97.76%.

Chapter 5: Conclusion

5.1 Conclusion

In conclusion, the main objective of this project was to find the most suited model in credit card fraud detection in terms of the machine learning techniques chosen for the project, and it was met by building the four models and finding the accuracies of them all, the best model in terms of accuracies is Support Vector Machine which scored 99.94% with only 51 misclassified instances. I believe that using the model will help in decreasing the amount of credit card fraud and increase the customers satisfaction as it will provide them with better experience in addition to feeling secure.

5.2 Recommendations

There are many ways to improve the model, such as using it on different datasets with various sizes, different data types or by changing the data splitting ratio, in addition to viewing it from different algorithm perspective. An example can be merging telecom data to calculate the location of people to have better knowledge of the location of the card owner while his/her credit card is being used, this will ease the detection because if the card owner is in Dubai and a transaction of his card was made in Abu Dhabi it will easily be detected as fraud.

Chapter 6: Bibliography

[1] Adepoju, O., Wosowei, J., lawte, S., & Jaiman, H. (2019). Comparative evaluation of credit card fraud detection using machine learning techniques. 2019 Global Conference for Advancement in Technology (GCAT). https://doi.org/10.1109/gcat47503.2019.8978372

[2] Alenzi, H. Z., & Aljehane, N. O. (2020). Fraud detection in credit cards using logistic regression. International Journal of Advanced Computer Science and Applications, 11(12). https://doi.org/10.14569/ijacsa.2020.0111265

[3] Awoyemi, J. O., Adetunmbi, A. O., & Oluwadare, S. A. (2017). Credit card fraud detection using Machine Learning Techniques: A Comparative Analysis. 2017 International Conference on Computing Networking and Informatics (ICCNI). https://doi.org/10.1109/iccni.2017.8123782

[4] Bhanusri, A., Valli, K. R. S., Jyothi, P., Sai, G. V., & Rohith, R. (2020). Credit card fraud detection using Machine learning algorithms. Journal of Research in Humanities and Social Science, 8(2), 04-11.

[5] Credit card statistics. Shift Credit Card Processing. (2021, August 30). Retrieved from https://shiftprocessing.com/credit-card/

[6] Daly, L. (2021, October 27). Identity theft and credit card fraud statistics for 2021: The ascent. The Motley Fool. Retrieved from https://www.fool.com/the-ascent/research/identity-theft-credit-card-fraud-statistics/

[7] Dheepa, V., & Dhanapal, R. (2012). Behavior based credit card fraud detection using support vector machines. ICTACT Journal on Soft Computing, 02(04), 391–397. https://doi.org/10.21917/ijsc.2012.0061

[8] Dighe, D., Patil, S., & Kokate, S. (2018). Detection of credit card fraud transactions using machine learning algorithms and Neural Networks: A comparative study. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). https://doi.org/10.1109/iccubea.2018.8697799

[9] Domínguez-Almendros, S., Benítez-Parejo, N., & Gonzalez-Ramirez, A. R. (2011). Logistic regression models. Allergologia et immunopathologia, 39(5), 295-305.

[10] Gupta, A., Lohani, M. C., & Manchanda, M. (2021). Financial fraud detection using naive Bayes algorithm in highly imbalance data set. Journal of Discrete Mathematical Sciences and Cryptography, 24(5), 1559–1572. https://doi.org/10.1080/09720529.2021.1969733

[11] Itoo, F., Meenakshi, & Singh, S. (2020). Comparison and analysis of logistic regression, Naïve Bayes and Knn Machine Learning Algorithms for credit card fraud detection. International Journal of Information Technology, 13(4), 1503–1511. https://doi.org/10.1007/s41870-020-00430-y

[12] Jain, Y., NamrataTiwari, S., & Jain, S. (2019). A comparative analysis of various credit card fraud detection techniques. International Journal of Recent Technology and Engineering, 7(5S2), 402-407

[13] Kiran, S., Guru, J., Kumar, R., Kumar, N., Katariya, D., & Sharma, M. (2018). Credit card fraud detection using Naïve Bayes model based and KNN classifier. International Journal Of Advance Research, Ideas And Innovations In Technology, 4(3).

[14] Maes, S., Tuyls, K., Vanschoenwinkel, B., & Manderick, B. (2002, January). Credit card fraud detection using Bayesian and neural networks. In Proceedings of the 1st international naiso congress on neuro fuzzy technologies (pp. 261-270).

[15] Mahesh, B. (2020). Machine Learning Algorithms - A Review, 9(1). https://doi.org/10.21275/ART20203995

[16] Malini, N., & Pushpa, M. (2017). Analysis on credit card fraud identification techniques based on KNN and outlier detection. 2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB). https://doi.org/10.1109/aeeicb.2017.7972424

[17] Maniraj, S. P., Saini, A., Ahmed, S., & Sarkar, S. D. (2019). Credit card fraud detection using machine learning and Data Science. Credit Card Fraud Detection Using Machine Learning and Data Science, 08(09). https://doi.org/10.17577/ijertv8is090031

[18] Najadat, H., Altiti, O., Aqouleh, A. A., & Younes, M. (2020). Credit card fraud detection based on machine and Deep Learning. 2020 11th International Conference on Information and Communication Systems (ICICS). https://doi.org/10.1109/icics49469.2020.239524

[19] Safa, M. U., & Ganga, R. M. (2019). Credit Card Fraud Detection Using Machine Learning. International Journal of Research in Engineering, Science and Management, 2(11).

[20] Saheed, Y. K., Hambali, M. A., Arowolo, M. O., & Olasupo, Y. A. (2020). Application of ga feature selection on Naive Bayes, random forest and SVM for credit card fraud detection. 2020 International Conference on Decision Aid Sciences and Application (DASA). https://doi.org/10.1109/dasa51403.2020.9317228

[21] Sahin, Y., & Duman, E. (2011). Detecting Credit Card Fraud by Decision Trees and Support Vector Machines. Proceedings of the International MultiConference of Engineers and Computer Scientists, 1.

[22] Sailusha, R., Gnaneswar, V., Ramesh, R., & Rao, R. R. (n.d.). Credit Card Fraud Detection Using Machine Learning. Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020).

[23] Tanouz, D., Subramanian, R. R., Eswar, D., Reddy, G. V., Kumar, A. R., & Praneeth, C. H. V. (2021). Credit card fraud detection using machine learning. 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS). https://doi.org/10.1109/iciccs51141.2021.9432308

[24] Varmedja, D., Karanovic, M., Sladojevic, S., Arsenovic, M., & Anderla, A. (2019). Credit Card Fraud Detection - machine learning methods. 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH). https://doi.org/10.1109/infoteh.2019.8717766

[25] Zareapoor, M., Seeja.K.R, S. K. R., & Afshar Alam, M. (2012). Analysis on credit card fraud detection techniques: Based on certain design criteria. International Journal of Computer Applications, 52(3), 35–42. https://doi.org/10.5120/8184-1538