AN INTERNSHIP PROJECT REPORT

On

CREDIT CARD FRAUD DETECTION

Submitted on partial fulfillment of the award of the degree

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted By

SIKHAKOLLI SHANKAR

(21021A0503)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

UNIVERSITY COLLEGE OF ENGINEERING KAKINADA

Jawaharlal Nehru Technological University Kakinada

Kakinada-533003, Andhra Pradesh, INDIA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY COLLEGE OF ENGINEERING KAKINADA
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA – 533003, ANDHRA PRADESH, INDIA

CERTIFICATE

This is to certify that this project report entitled **"CREDIT CARD FRAUD DETECTION"** is a bonafide record of the work being submitted by **S. SHANKAR** bearing the roll number **21021A0503,** in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in **COMPUTER SCIENCE AND ENGINEERING** to the **UCEK(A), JNTUK,** Kakinada, Andhra Pradesh, India. It has been found satisfactory and hereby found satisfactory and hereby approved for submission.

Signature of Head of the Department
Dr. N. RamaKrsihnah
Professor & HOD
Department of CSE
UCEK (A)
JNTU KAKINADA

HDLC TECHNOLOGIES

Certificate of Internship

This is to certify that

Sikhakolli Shankar

is hereby awarded this certificate for successfully completing 2 months Internship program in **Data Science and Machine Learning** between 02/05/2023 & 30/06/2023. During the time of Internship he has worked with commitment and successfully completed the project. We wish him all the very best for future endeavors.

Program Head

Certificate issued on 30/06/2023

AICTE INTERNSHIP_1680690356642d4cb4eab80

Certificate id : HDLC/IT/MY/2182

ABSTRACT

Credit card fraud is a significant problem, with billions of dollars lost each year. Machine learning can be used to detect credit card fraud by identifying patterns that are indicative of fraudulent transactions. Credit card fraud refers to the physical loss of a credit card or the loss of sensitive credit card information. Many machine learning algorithms can be used for detection. This project proposes to develop a machine-learning model to detect credit card fraud. The model will be trained on a dataset of historical credit card transactions and evaluated on a holdout dataset of unseen transactions.

Keywords: Credit Card Fraud Detection, Fraud Detection, Fraudulent Transactions, K-Nearest Neighbors, Support Vector Machine, Logistic Regression, Decision Tree.

Chapter 1

1.1 Introduction

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

Reports of Credit card fraud in the U.S. rose by 44.7% from 271,927 in 2019 to 393,207 words in 2020. There are two kinds of credit card fraud, and the first is having a credit card account opened under your name by an identity thief. Reports of this fraudulent behaviour increased 48% from 2019 to 2020. The second type is when an identity thief uses an existing account you created, usually by stealing the information on the credit card. Reports on this type of Fraud increased 9% from 2019 to 2020 (Daly, 2021). Those statistics caught We's attention as the numbers have increased drastically and rapidly throughout the years, which motivated We to resolve the issue analytically by using different machine learning methods to detect fraudulent credit card transactions within numerous transactions.

1.2 Project goals

The main aim of this project is the detection of fraudulent credit card transactions, as it is essential to figure out the fraudulent transactions so that customers do not get charged for the purchase of products that they did not buy. Fraudulent Credit card transactions will be detected with multiple ML techniques. Then, a comparison will be made between the outcomes and results of each method to find the best and most suited model for detecting fraudulent credit card transactions; graphs and numbers will also be provided. In addition, it explores previous literature and different techniques used to distinguish Fraud within a dataset.

Research question: What machine learning model is most suited for detecting fraudulent credit card transactions?

Chapter 2: Literature Review

2.1 Introduction

Credit card companies must distinguish fraudulent from non-fraudulent transactions so that their customers' accounts will not get affected and charged for products they did not buy (Maniraj et al.,2019). Many financial Companies and institutions 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). Multiple methods are used to detect fraudulent behaviours, such as Neural Networks (N.N.), Decision Trees, K-nearest neighbor algorithms, and Support Vector Machines (SVM). Those ML methods can be applied independently or collectively with 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 for detecting fraudulent transactions, which was established using the Accuracy of the model, the speed of detection and the cost. The models used were Neural Network, Bayesian Network, SVM, KNN and. The comparison table in theresearch paper showed that the Bayesian Network was high-speed in finding fraudulent transactions with high Accuracy. The N.N. performed well, as the detection was fast, with a medium accuracy. KNN's speed was good with medium Accuracy, andfinally, SVM scored one of the lower scores, as the speed was low, and the Accuracy wasmedium. As for the cost, All models built were expensive (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 classifiers, which are

They were voting Classifier and KNN. V.C. 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).

Maniraj's team built a model to recognize if any new transaction is Fraud or non-fraud. Their goal was to get 100% in detecting fraudulent transactions andtry to minimize the incorrectly classified fraud instances. Their model has performed well as they got 99.7% of the fraudulent transactions (Maniraj et al., 2019).

The classification approach used by Dheepa and Dhanapal was the behaviour-based classification approach, using a Support Vector Machine, where the behavioural patternsof 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 wasmore than 80% (Dheepa & Dhanapal, 2012).

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

Maes and his team proposed using Bayesian and Neural Networks to detect credit card fraud. Their results showed that Bayesian performance is 8% more effective in detecting Fraud than ANN, meaning BBN sometimes detects 8% more 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 compared the usage of three ML techniques in detecting 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%,

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 (T.P.), false Negative (F.N.), false positive (F.P.), and true negative (T.N.) 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. Moreover, finally,the Accuracy of KNN is 97.15%, the precision is 96.84%, and the false alarm rate is 2.88% (Jain et al., 2019)

Dighe and his team used KNN, Logistic Regression and Neural Networks, multilayer perceptron and Decision Tree in their work, then evaluated the results regarding numerous accuracy metrics. Of all the models created, the best performing one is KNN, which scored 99.13%, then in second place performing model at 96.40% and in last place is logistic Regression with 96.27% (Dighe et al., 2018).

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

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 mostsuited data and preparing/preprocessing are the first and second steps; after making surethat the data is ready, the modelling phase starts, where four models are created: K-NearestNeighbor (KNN), Decision Tree, 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 Jupiter Notebook programs.

3.2 Data Source

The dataset was retrieved from an open-source website, Kaggle.com. It contains data on transactions made in 2013 by European credit card users in two days only. Thedataset consists of 31 attributes and 284,808 rows. Twenty-eight attributes are numeric variables that, due to the 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 variableswhere “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 below shows the structure of the dataset where all attributes are shown, with their type, in addition to a 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 one as Fraud to ease the process of creating the modeland obtain visualizations.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 284807 entries, 0 to 284806
Data columns (total 31 columns):
 #   Column  Non-Null Count   Dtype  
---  ------  --------------   -----  
 0   Time    284807 non-null  float64
 1   V1      284807 non-null  float64
 2   V2      284807 non-null  float64
 3   V3      284807 non-null  float64
 4   V4      284807 non-null  float64
 5   V5      284807 non-null  float64
 6   V6      284807 non-null  float64
 7   V7      284807 non-null  float64
 8   V8      284807 non-null  float64
 9   V9      284807 non-null  float64
 10  V10     284807 non-null  float64
 11  V11     284807 non-null  float64
 12  V12     284807 non-null  float64
 13  V13     284807 non-null  float64
 14  V14     284807 non-null  float64
 15  V15     284807 non-null  float64
 16  V16     284807 non-null  float64
 17  V17     284807 non-null  float64
 18  V18     284807 non-null  float64
 19  V19     284807 non-null  float64
 20  V20     284807 non-null  float64
 21  V21     284807 non-null  float64
 22  V22     284807 non-null  float64
 23  V23     284807 non-null  float64
 24  V24     284807 non-null  float64
 25  V25     284807 non-null  float64
 26  V26     284807 non-null  float64
 27  V27     284807 non-null  float64
 28  V28     284807 non-null  float64
 29  Amount  284807 non-null  float64
 30  Class   284807 non-null  int64  
dtypes: float64(30), int64(1)
memory usage: 67.4 MB
```

Figure 1 - Dataset Structure

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

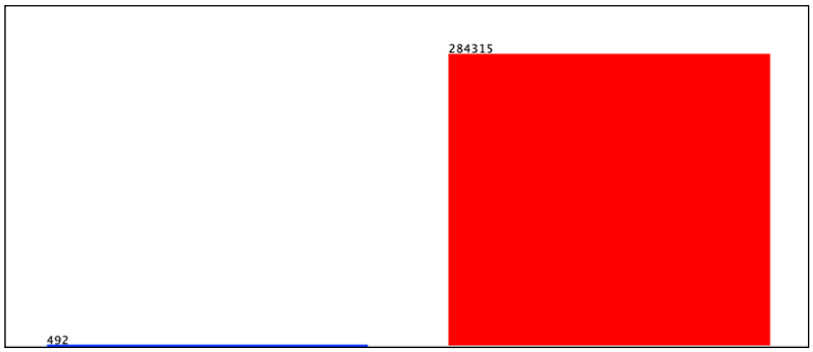

Figure 2 - Class Distribution

4.1.1 Correlation between attributes “Image from R”

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

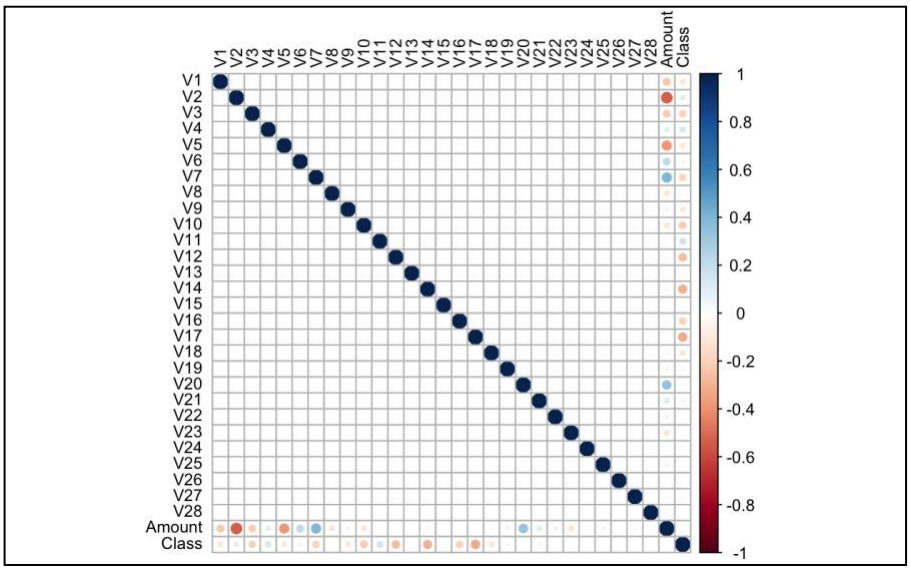

Figure 3 - Correlations

4.1.2 Attribute to the Most Fraud

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

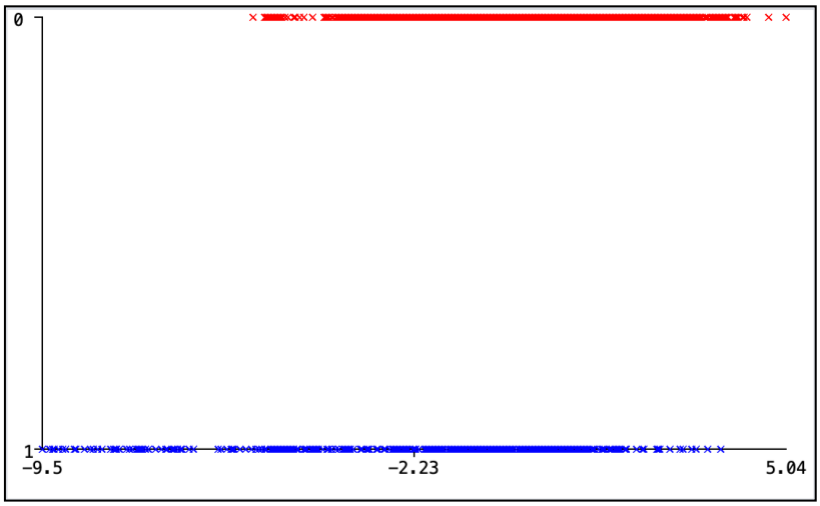

Figure 4 – Variable 18

4.1.3 Attribute with Less Fraud

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

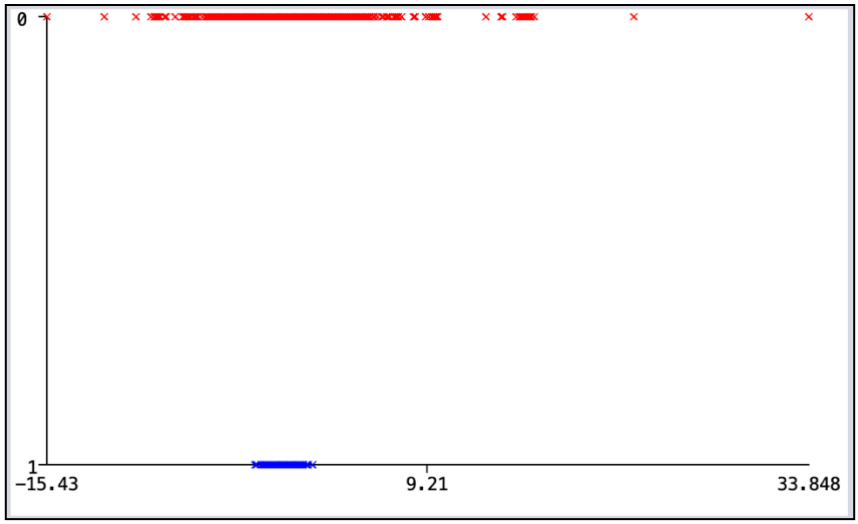

Figure 5 - Variable 28

4.2 Data Preprocessing

As there are no N.A.s nor duplicated variables, preparing the dataset was simple. The first alteration made to open the dataset on the Jupiter Notebook program is changing the type of the class attribute from Numeric to Class and identifying the class as{1, 0} using the program Sublime Text. Another alteration was made to the type and the Jupiter Notebook program to create the model and the visualization.

4.3 Data Modeling

After ensuring that the data is ready to get modelled, the four models were created for KNN, Logistic Regression and Decision Tree using Jupiter Notebook.

4.3.1 K-Nearest Neighbor (KNN):

This supervised learning technique achieves consistently high performance compared to other fraud detection techniques of supervised statistical pattern recognition. Three factors majorly affect its performance distance to identify the least distant neighbors. There are some rules to deduce a categorization from the k-nearest neighbor and the count of neighbors to label the new sample. This algorithm classifies transactions by computing the least distant point to this particular transaction. If this least distant neighbor is classified as fraudulent, the latest marketing is also labelled as fraudulent. Euclidean distance is an excellent choice to calculate the distances in this scenario. This technique is fast and results in fault alerts. Its performance can be improved by distance metric optimization.

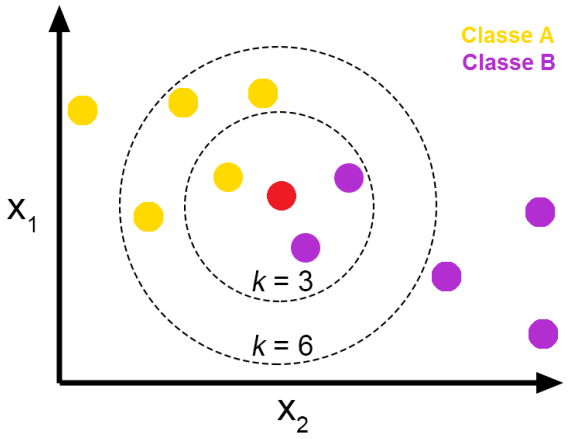

Figure 6 - Pros and Cons of K-Nearest Neighbors - From The GENESIS

Algorithm KNN

1. Let *m* be the number of training data samples. Let *p* be an unknown point that needs to be classified
2. Storing the training samples in an array of data points *arr*[]. Each element of this array denotes a tuple (*x*, *y*).
3. **for** *i* = 0 to *m* **do**
4. Calculating distance *d*(*arr*[*i*], *p*)
5. **end for**
6. They are making set *S* of *K* smallest distances achieved. Each of these distances resembles an already classified data point.
7. Returning the majority label among *S*

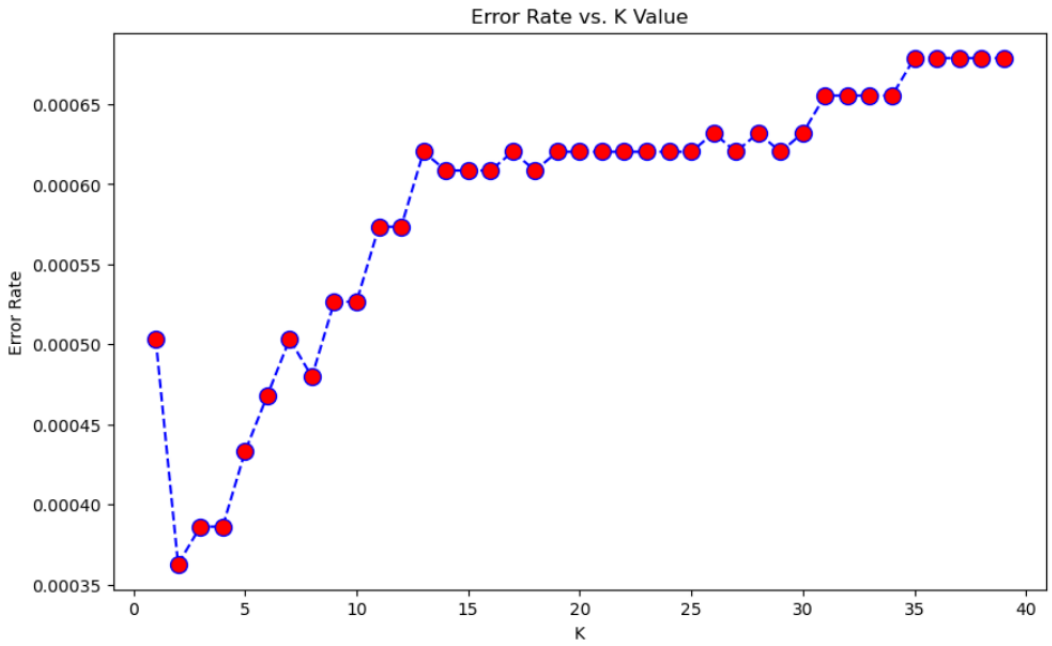

Figure 7 – Error Rate

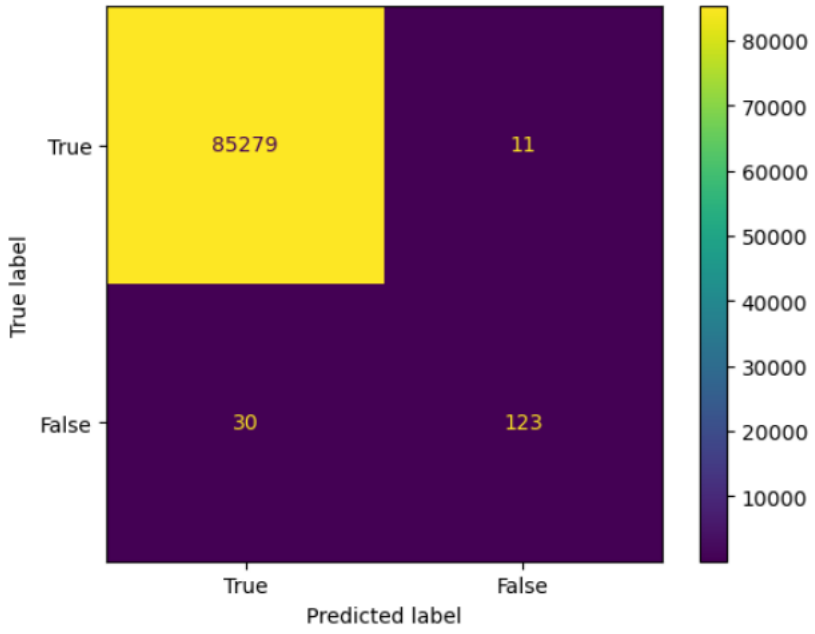

Figure 8 – Confusion Matrix

Two Ks were used to determine the best KNN model, K=3 and K =7.

- K = 3

While making the KNN model, We created two models: K =3 and K =7.Figure 5 shows the model created in Jupiter Notebook; the model scored an accuracy of 100% and identified 85,443 transactions correctly and missed 131.

```
WITH k=3


[[85307     5]
 [   28   103]]

              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85312
           1       0.95      0.79      0.86       131

    accuracy                           1.00     85443
   macro avg       0.98      0.89      0.93     85443
weighted avg       1.00      1.00      1.00     85443
```

Figure 9 - K=3

- K = 7

There was a slight decrease in the Accuracy of the model created in Jupiter Notebook (Figure 6) as it scored 100% when K is 7, and the model miss classified 131 fraudulent transactions asno fraudulent. As for the Accuracy is the same as K=3 100% with 52 misclassified transactions; the only difference is within the classifications.

```
WITH k=7


[[85300    12]
 [   31   100]]

              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85312
           1       0.89      0.76      0.82       131

    accuracy                           1.00     85443
   macro avg       0.95      0.88      0.91     85443
weighted avg       1.00      1.00      1.00     85443
```

Figure 10 - K=7

4.3.2 Logistic Regression (L.R.):

This statistical classification model based on probabilities detects Fraud using a logistic curve. Since the value of this logistic curve varies from 0 to 1, it can be used to interpret class membership probabilities. The dataset fed as input to the model is classified for training and testing the model. Post-model activity is tested for some minimum threshold cut-off value for prediction. Based on some threshold probabilities, the logistic Regression can divide the plane using a single line and divide dataset points into exactly two regions.

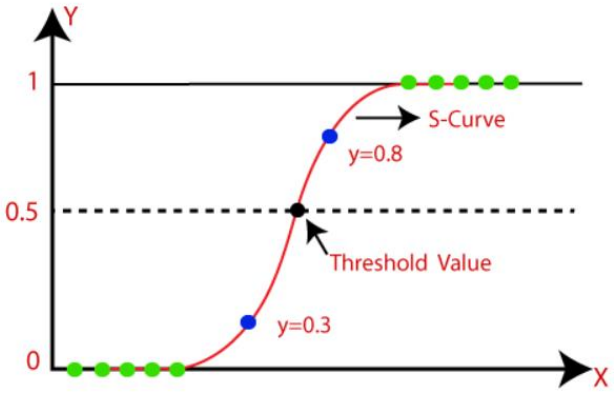

Figure 11 - The logistic regression model

The last model created using Jupiter Notebook is Logistic Regression; the model managed to score an Accuracy on Training data of 93.51% (figure 12), while it scored an Accuracy score on Test Data of 91.88%, as presented in Figure 13.

```
In [29]: print('Accuracy on Training data : ', training_data_accuracy)
         Accuracy on Training data :  0.9351969504447268
```

Figure 12 - Accuracy of Training data

```
In [31]: print('Accuracy score on Test Data : ', test_data_accuracy)
         Accuracy score on Test Data :  0.9187817258883249
```

Figure 13 - Accuracy score on Test Data

4.3.3 Support Vector Machine (SVM):

Support vector machines or SVMs are linear classifiers, as stated in that work in high dimensionality because, in high dimensions, a non-linear task in input becomes linear. Hence, this makes SVMs highly useful for detecting Fraud. Its two most important features that is a kernel function to represent the classification function in the dot product of input data point projection and the fact that it tries finding a hyperplane to maximize separation between classes while minimizing overfitting of training data; it provides a very high generalization capability.

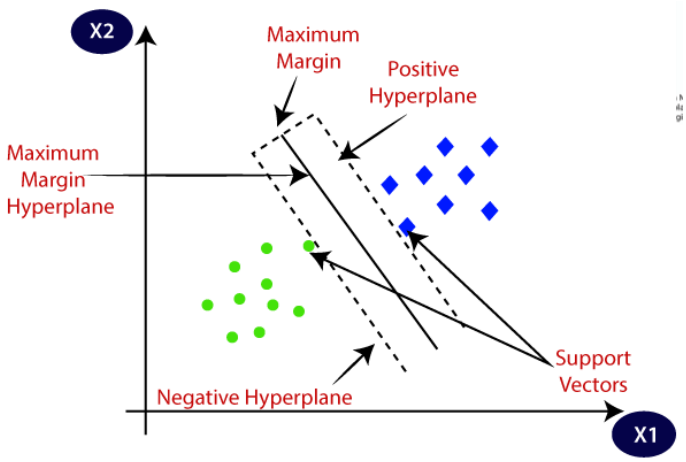

Figure 14 - Support Vector Machine algorithm.

Finally, the model Support Vector Machine, as shown in Figure 12, scored 97.59% for the Accuracy.

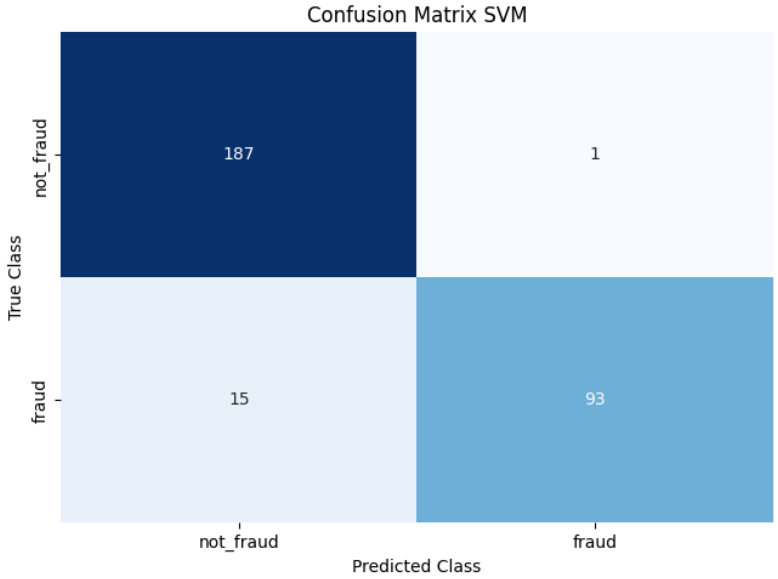

Figure 15 – Confusion Matrix of SVM

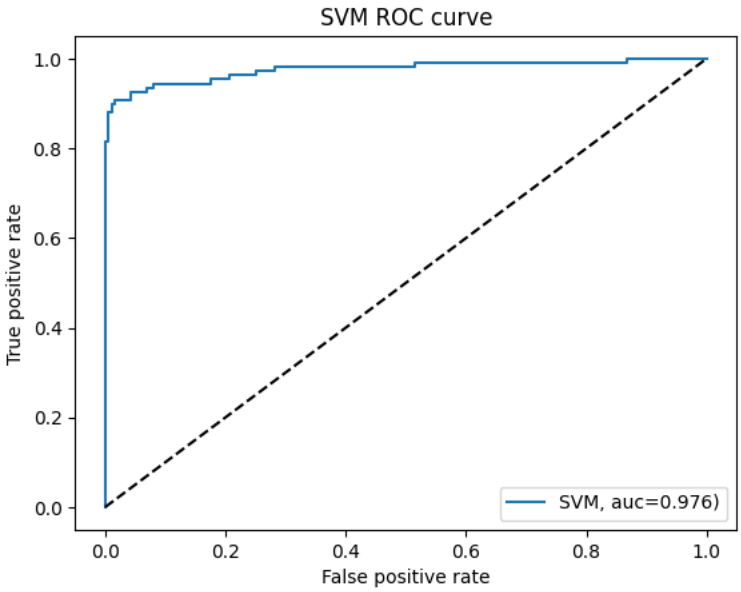

Figure 16 - Support Vector Machine ROC curve

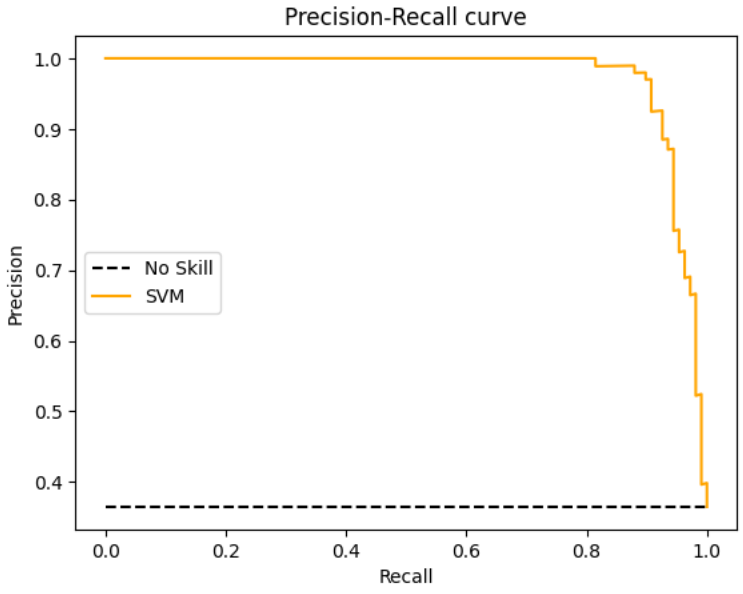

Figure 17 – Precision-Recall curve

4.3.4 Decision Tree (D.T.):

A supervised learning algorithm is a decision tree in the form of a tree structure consisting of the root node and other nodes split in a binary or multi-split manner further into child nodes, with each tree using its algorithm to perform the splitting process. With the tree growing, there may be possibilities of overfitting the training data with possible anomalies in branches, some errors or noise. Hence, pruning is used for improving classification performance of the tree by removing specific nodes. Ease in use and the flexibility that the decision trees provide to handle different data types of attributes make them quite popular.

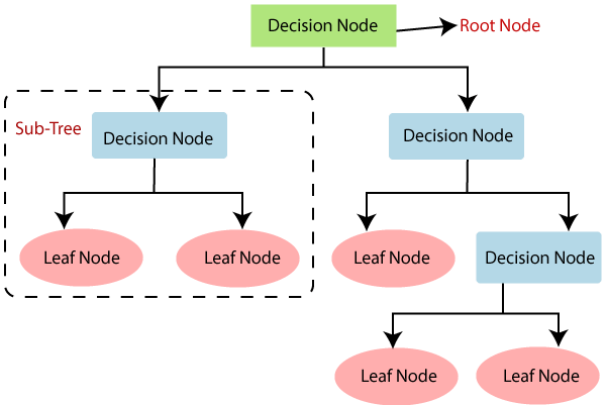

Figure 18 – Decision Tree Algorithm in Machine Learning

Algorithm D.T.

1. Create (T)
2. Calculate frequencies (Ci, T)
3. If all instances belong to the same class, the returning leaf
4. for every Attribute, a test is set for splitting criteria. An attribute that satisfies the test is test node K
5. Repeating Create (Ti) on each partition Ti.Adding those nodes as children of node K

Finally, the model Decision Tree**,** as shown in the figure, scored 100% for Accuracy.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	85285
1	0.83	0.81	0.82	158
accuracy			1.00	85443
macro avg	0.92	0.90	0.91	85443
weighted avg	1.00	1.00	1.00	85443

Figure 19 – Accuracy of Decision Tree

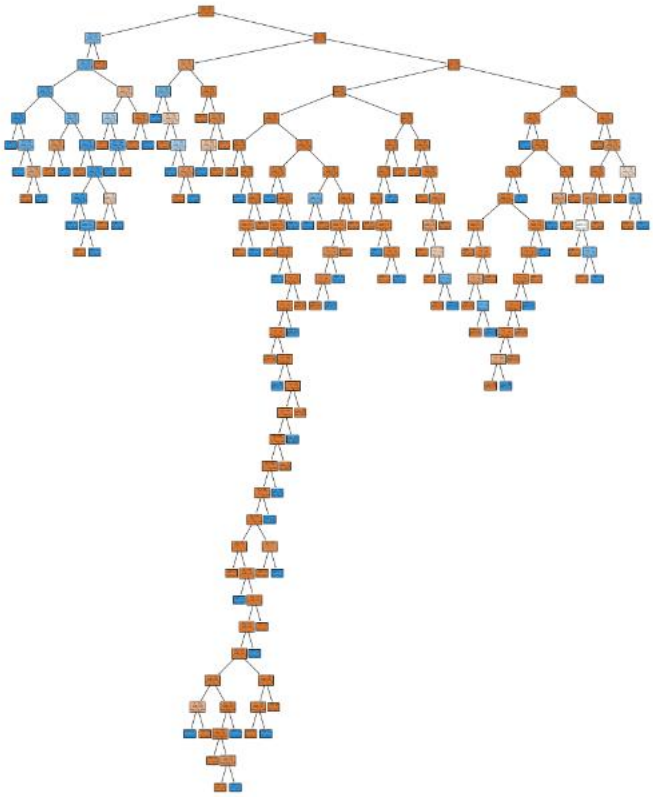

Figure 20 - Decision Tree

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 determine the best model for identifying fraudulent credit card transactions.

Accuracy is the overall number of instances that are predicted correctly; accuracies are represented by a confusion matrix where it shows the True Positive (T.P.), True Negative (T.N.), False Positive (F.P.) and False Negative (F.N.). True Positive represents the transactions that are fraudulent and were correctly classified by the model as fraudulent. True Negative represents the not fraudulent transactions that the model correctly predicted as not fraudulent. The third rating is False positive, which represents the fraudulent transaction but was misclassified as not fraudulent. Moreover, finally, FalseNegative, which are the not fraudulent transactions identified as fraudulent; Table 1 below shows the confusion matrix.

Actual/Predicted	Positive	Negative
Positive	TP	FN
Negative	F.P.	TN

Table 1 - Confusion Matrix

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

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

Model		Accuracy
KNN	K = 3	100%
	K = 7	100%
Logistic Regression	Training Data	93.51%
	Test Data	91.88%
Support Vector Machine	SVM	97.59%
Decision Tree	DT	100%

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 KNN and Decision Tree as its Accuracy is 100%, the thirdplace is the Support Vector Machine, and the model that scored the lowestAccuracy out of all models is Logistic Regression with a score of 93.51%.

Chapter 5: Conclusion

5.1 Conclusion

In conclusion, the main objective of this project was to find the most suited model for creditcard fraud detection in terms of the machine learning techniques chosen for the project. It was met by building the four models and finding the accuracies of them all; the bestmodel in terms of accuracies is KNN and Decision Tree, which scored 100%. We believe that using the model will help decrease the amount of credit card fraud and increase the customer's satisfaction as it will provide themwith a better experience and feeling secure.

5.2 Recommendations

There are many ways to improve the model, such as using it on different datasets with various sizes and data types or by changing the data splitting ratio and viewing it from a different algorithm perspective. An example can be merging telecom datato 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 easilybe detected as Fraud.

Chapter 6: Bibliography

[1] Adepoju, O., Wosowei, J., lawte, S., & Jaiman, H. (2019). Comparative evaluation ofcredit 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 Humanitiesand Social Science, 8(2), 04-11.

[5] Credit card statistics. Shift Credit Card Processing. (2021, August 30). Retrievedfrom 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). Behaviour-based credit card fraud detection usingsupport 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 usingnaive Bayes algorithm in highly imbalanced 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 andEngineering, 7(5S2), 402-407

[13] Kiran, S., Guru, J., Kumar, R., Kumar, N., Katariya, D., & Sharma, M. (2018). Creditcard 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). Creditcard fraud detection using Bayesian and neural networks. In Proceedings of the 1st International 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 of credit card fraud identification techniques based on KNN and outlier detection. 2017 Third International Conference onAdvances 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 UsingMachine 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 onInformation 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] Sahin, Y., & Duman, E. (2011). Detecting Credit Card Fraud by Decision Trees andSupport Vector Machines. Proceedings of the International MultiConference of Engineers and Computer Scientists, 1.

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

[22] 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 5thInternational Conference on Intelligent Computing and Control Systems (ICICCS). https://doi.org/10.1109/iciccs51141.2021.9432308

[23] 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

[24] 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 Journalof Computer Applications, 52(3), 35–42. https://doi.org/10.5120/8184-1538