UNIVERSITY OF CALIFORNIA

Los Angeles

Credit Card Fraud Detection Using Logistic Regression and Machine Learning Algorithms

A thesis submitted in partial satisfaction of

the requirements for the degree

Master of Applied Statistics

by

Haoyi Cheng

2023

ABSTRACT OF THE THESIS

Credit Card Fraud Detection Using Logistic Regression and Machine Learning Algorithms

by

Haoyi Cheng

Master of Applied Statistics

University of California, Los Angeles, 2023

Professor Frederic R Paik Schoenberg, Chair

This thesis is focused on detecting the probability of credit card fraud occurrence according to seven relative independent variables by using logistic regression, support vector machine, decision tree, and k-NN models. The dataset provided by Dhanush Narayanan R from Kaggle contains one million of data [1]. The final goal is to compare these four models. The outcomes indicate that all models perform well and the most perfect model is the decision tree model, which reaches nearly 100% accuracy.

The thesis of Haoyi Cheng is approved.

Yingnian Wu

Hongquan Xu

Frederic R Paik Schoenberg, Committee Chair

University of California, Los Angeles

2023

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

With the development and popularity of credit cards, more and more people hold credit cards in current society. According to a credit card processing company, there are 2.8 billion credit cards worldwide [2]. On average, every three people in the world own a credit card. The United States is the origin of credit cards and the most developed country in the industry. The major credit card companies are from America, such as Visa, Mastercard, American Express, etc. Americans held an average of 3.84 credit card accounts in the third quarter of 2020 [3]. In addition, US credit card issuers' transaction volume and credit card loans also maintained a leading position in the world.

1.01 billion credit card transactions occurred every day worldwide in 2018 [4]. However, everything is double-sided. While credit cards bring convenience to daily life, card fraud incidents also occur more and more frequently. The Federal Trade Commission indicated that in 2021, 389,737 credit card fraud reports happened in the United States. “Credit card fraud is the second most reported type of identity theft in the US.” [5].

The reason for choosing this topic is that this is related to every aspect of public life because most consumption nowadays is generated by cards, and few people use cash or checks in daily life. Therefore, it is still a card transaction, even if it is bound to PayPal or other platforms. Nevertheless, on the other hand, many people have experienced credit card fraud. Thus, they would probably like to know the characteristics of credit card fraud or what type of transaction is most prone to credit card fraud. In addition, if the machine can identify fraudulent credit card transactions using models, credit card companies or banks can avoid unnecessary losses.

Since fraud detection is a classification question, logistic regression is the primary regression option to be chosen. In addition, machine learning methods like the k-NN algorithm, SVM, and decision tree can also be considered.

In this study, the following steps are performed to fulfill the purpose: download the original card transaction data from Kaggle, exercise the exploratory data analysis, apply different kinds of models, compare the models, and make conclusions.

Chapter 2

Exploratory Data Analysis

2.1 Data Information

The card transaction dataset was obtained from Kaggle and uploaded by Dhanush Narayanan R in May 2022. The uploader states this dataset was sourced by some unnamed institute. It contains 1,000,000 historical card transaction records, which should be persuasive due to the enormous sample size. Moreover, this dataset includes eight variables, seven independent variables and one dependent variable.

2.2 Variable Introduction

All variables are numerical variables

DISTANCE FROM HOME (DH): the distance between the location of the transaction happened and the home of the card owner

DISTANCE FROM LAST TRANSACTION (DLT): the distance between the location of this transaction happened and the last transaction that happened

RATIO TO MEDIA PURCHASE PRICE (RATIO): ratio of purchased price to the median purchase price

REPEAT RETAILER (RR): whether this transaction happened from the same retailer or not

0: not from the same retailer

1: from the same retailer

USED CHIP (CHIP): whether this transaction is used through the chip since normally credit card contains three ways to pay, use a chip, swipe or tap the card

0: using other ways to pay

1: using the chip credit card

USED PIN NUMBER (PIN): whether this transaction is used the PIN number

0: no need for the PIN number

1: need the PIN number to pay

ONLINE ORDER (OO): whether this transaction is an online order or a physical payment

0: physical payment

1: online order

Fraud: whether this transaction is fraudulent or not

0: normal transaction

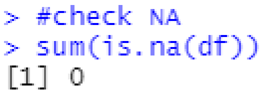
1: fraud transaction

2.3 Research Questions

It involves several study questions. For example, under what circumstances are credit card fraud more frequently occurring? Furthermore, the most crucial part is that the model can predict fraud.

2.4 Exploratory Data Analysis

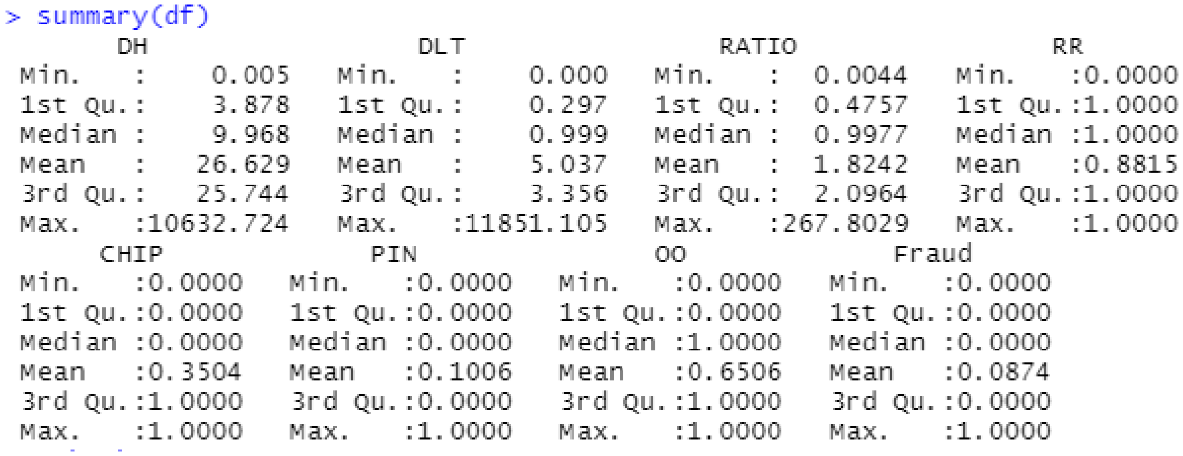
Figure 2.1: Missing Values



```
> #check NA
> sum(is.na(df))
[1] 0
```

The above result (Figure 2.1) indicates there does not exist any missing values in the original dataset. Therefore, replacing or cleaning any value in the data frame is temporarily unnecessary.

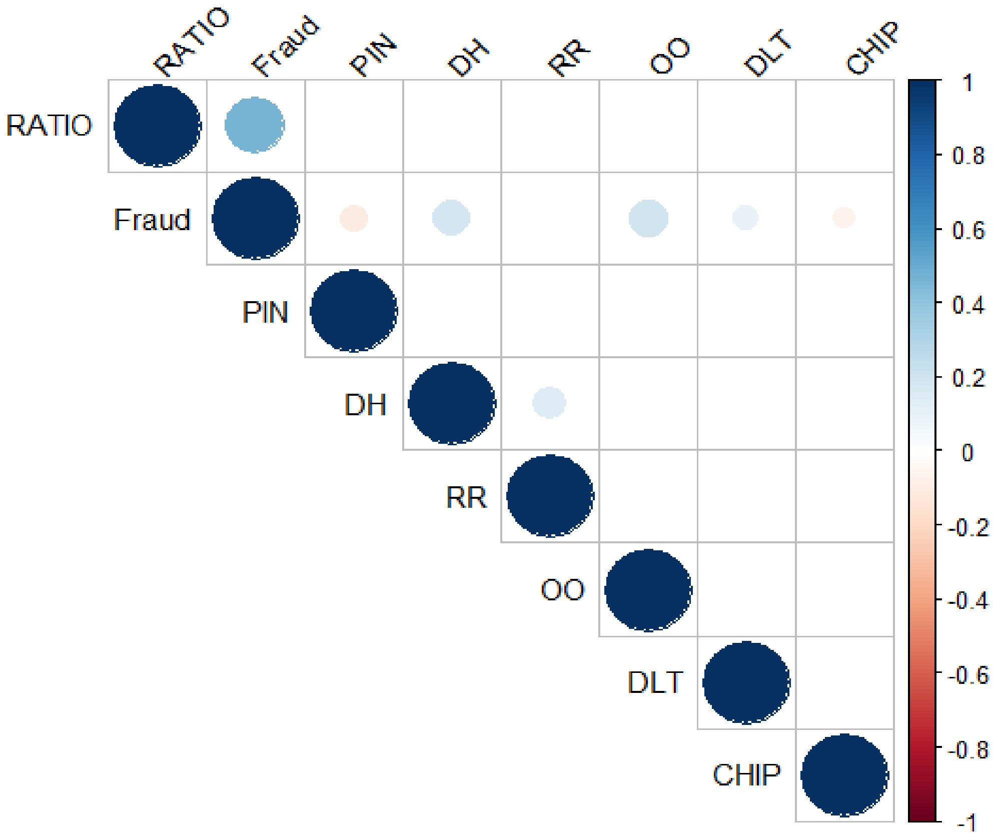
Figure 2.2: Dataset Summary



```
> summary(df)
       DH                DLT               RATIO                RR        
 Min.   :    0.005   Min.   :    0.000   Min.   :  0.0044   Min.   :0.0000  
 1st Qu.:    3.878   1st Qu.:    0.297   1st Qu.:  0.4757   1st Qu.:1.0000  
 Median :    9.968   Median :    0.999   Median :  0.9977   Median :1.0000  
 Mean   :   26.629   Mean   :    5.037   Mean   :  1.8242   Mean   :0.8815  
 3rd Qu.:   25.744   3rd Qu.:    3.356   3rd Qu.:  2.0964   3rd Qu.:1.0000  
 Max.   :10632.724   Max.   :11851.105   Max.   :267.8029   Max.   :1.0000  
      CHIP             PIN               OO             Fraud       
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
 Median :0.0000   Median :0.0000   Median :1.0000   Median :0.0000  
 Mean   :0.3504   Mean   :0.1006   Mean   :0.6506   Mean   :0.0874  
 3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.0000  
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000  
```

The dataset summary (Figure 2.2) shows that the range for the two distance relative variables is extensive. For example, the differences between the minimum and maximum distances reach more than 10,000 for both variables. However, the mean and median for these two distance variables are minor. That can indicate only a few extreme values in the two variables that do not severely affect the median and mean. Also, the range for the ratio to the median purchase price is relatively high.

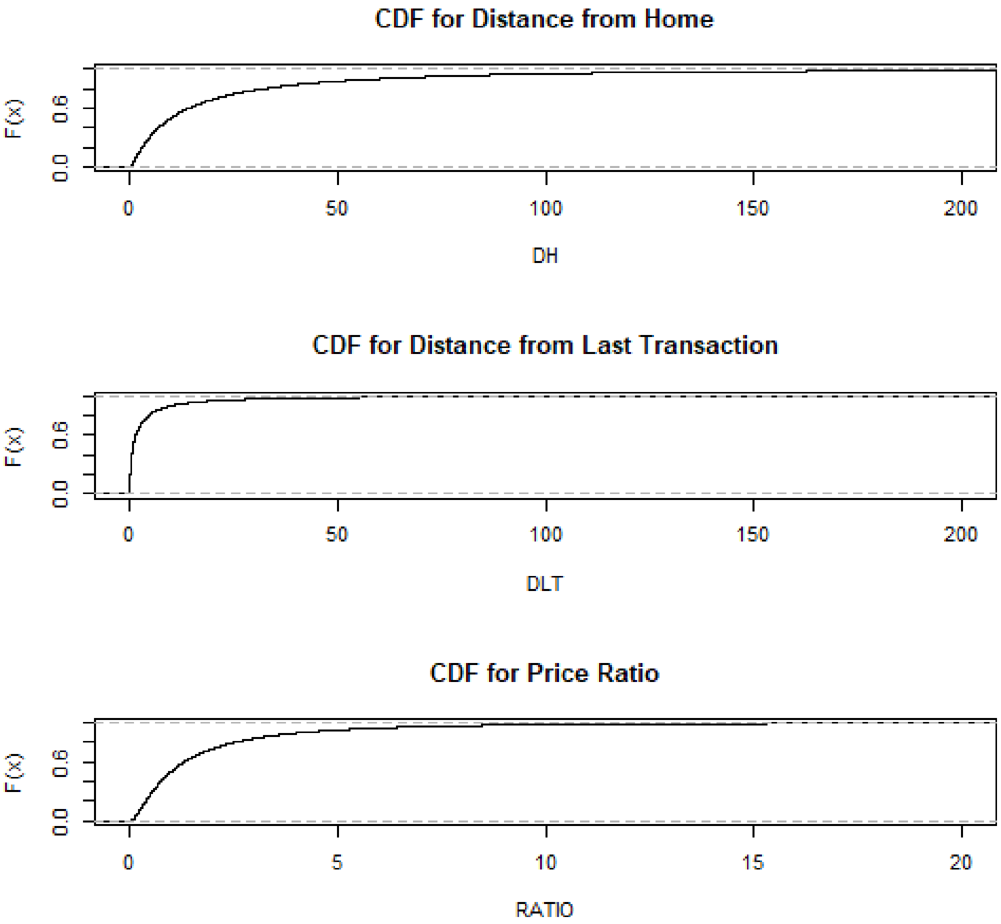
Figure 2.3: Correlation Matrix Plot



The correlation matrix plot (Figure 2.3) indicated that the ratio to the median price variable is the most correlated variable with fraud. Moreover, they should be positively related, like as the price ratio goes higher, this card transaction seems to be fraud is higher. On the one hand, distance from home, online order, and distance from the last transaction may lead to fraud to a certain extent. On the other hand, using the PIN and chip possibly reduces the risk of fraud since they are negatively related. Meanwhile, the repeat retailer seems not to have any relationship with the dependent variable, but it does have a positive relationship with an independent variable, distance from home.

For other figures in EDA, the variables are divided into three groups. First, put the non-binary independent variables together, as shown below. Then put the independent variables that only contain zero and one, which are the binary variables, and make them into bar plots since there are only two possible values. Finally, variable fraud, as an outcome, is shown separately.

Figure 2.4: CDF Plots Before Transforming



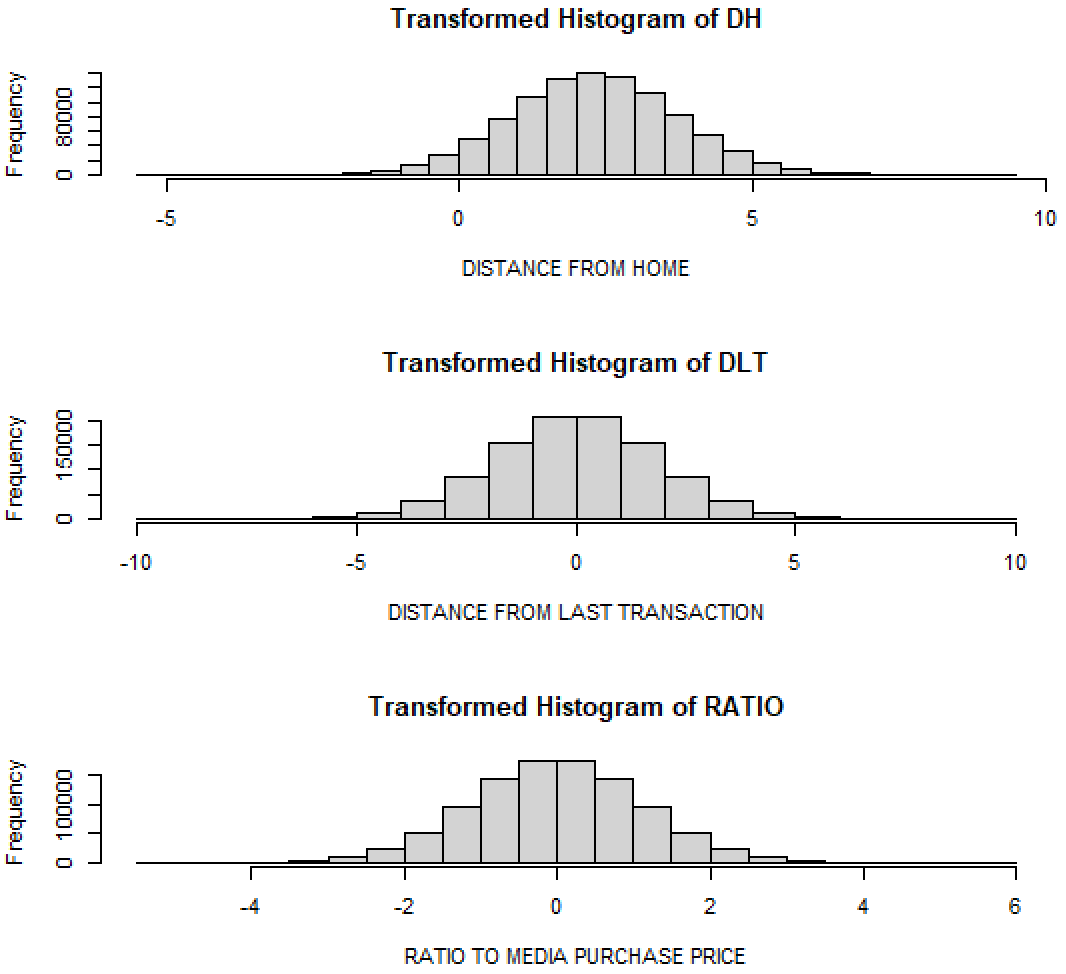
CDF for Distance from Home

CDF for Distance from Last Transaction

CDF for Price Ratio

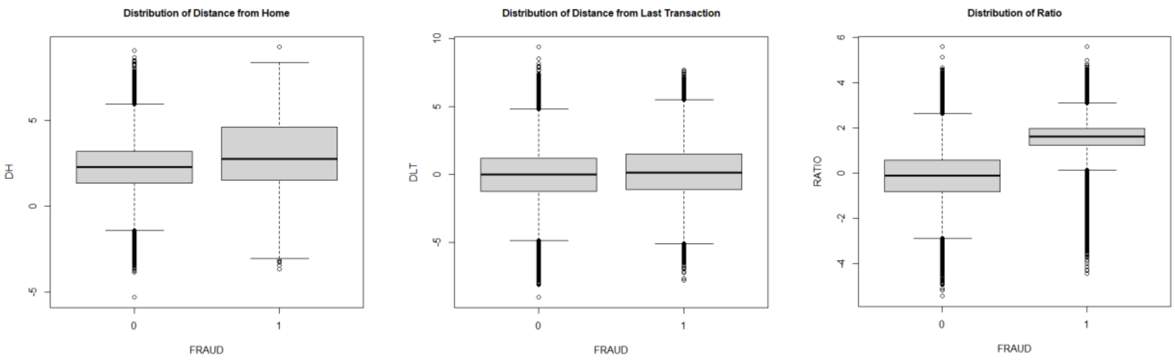
From the above three CDF plots (Figure 2.4), to see the plots more clearly, the x range of the two distances is controlled from 0 to 200, and the price ratio ranges from 0 to 20. Because basically, most distances are within 200 meters, and the ratio is within 20. In other words, the values for the distance less than 200 and the ratio below 20 accounts for almost 100%.

Figure 2.5: Histograms After Transforming



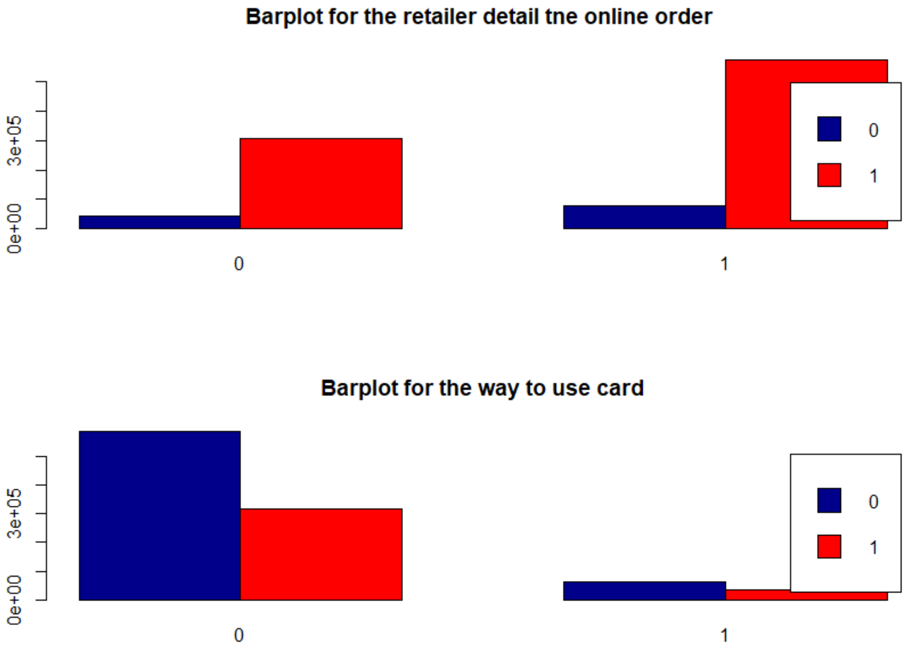
Since the original histograms are generally not distributed, transformation to the variables is applied. After using the log transformation, all three variables look more normally distributed. (Figure 2.5) Especially for the distance from home variable can be counted as the most standard normal distribution among these variables, which is a bell shape.

Figure 2.6: Box Plots



(Figure 2.6) The left box plot shows that the fraud transaction has a more extensive range of distance from home than the non-fraud transaction, so fraud may be more likely to occur far from home. The central figure demonstrates that the situation is similar for both fraud cases, that fraud may occur at any distance from the last transaction. For the right box plot, the fraud case has a smaller range, but the mean is higher than the non-fraud case, which means that overall, the fraud transaction tends to happen in a more excellent ratio to the median price.

Figure 2.7: Bar Plots for Variables



The reason for separating the four variables is to facilitate comparison. These two variables, chip and PIN, belong to the statistics of card usage. The repeat retailer and online order are statistics on the transaction mode. From the bar plots result (Figure 2.7), most card transactions in this dataset happened in a repeat retailer or online order. And most of them did not use the chip or PIN. It fits expectations since according to personal credit card transaction experience in daily life, it is true that in most cases do not need to enter the pin number when swiping a credit card.

Figure 2.8: Bar Plot for Fraud

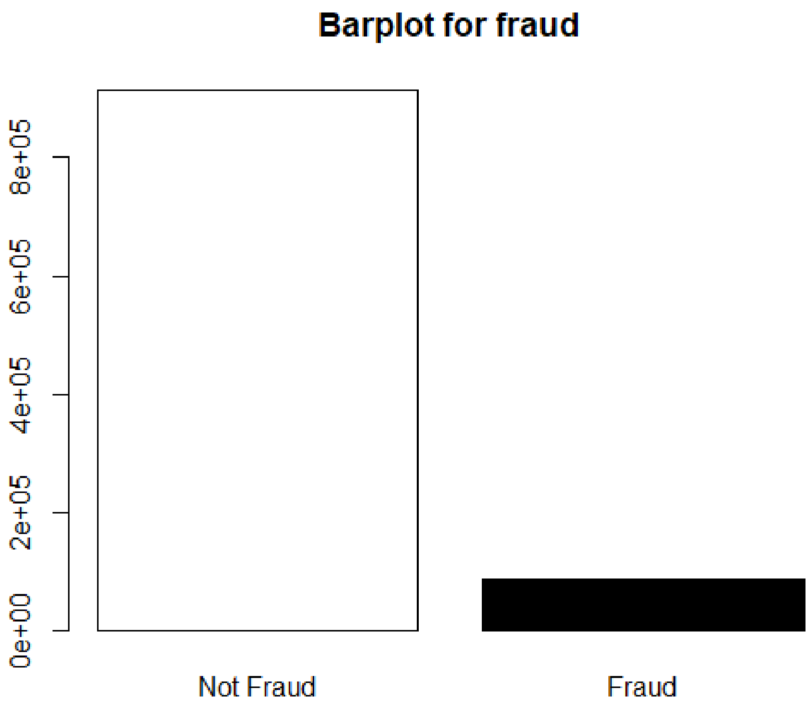


Table 2.1: Fraud Count

Table	Normal Transaction	Fraud
Count	912597	87403

The bar plot for the outcome (Figure 2.8) shows that most credit card transactions are not a fraud. (Table 2.1) Out of one million card transactions, only 87,403 transactions are fraud. The probability of card fraud is less than nine percent in this dataset.

Chapter 3

Logistic Regression

3.1 Introduction of Logistic Regression

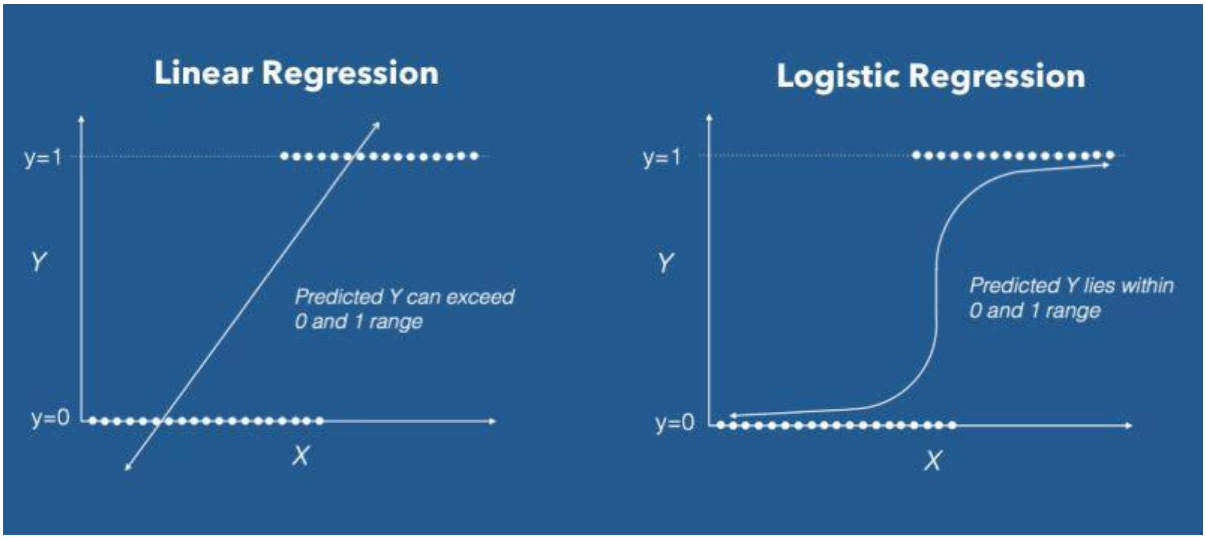
Logistic regression is based on the sigmoid function and its result is from zero to one. The sigmoid for logistic regression is

$$S_i = \sum_{j=1}^{p} x_{ij}\beta_j = x_i^T\beta$$

The function of the logistic regression is

$$f(x) = \frac{1}{1 + e^{-(\beta_0+\beta_1 x)}}$$

Figure 3.1: Regression Comparison [6]

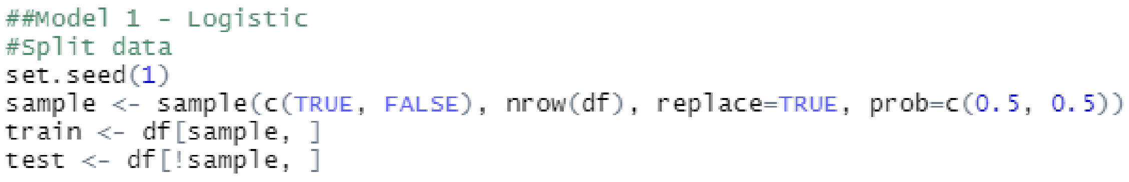


Logistic regression is similar to linear regression. It is a particular case of the Generalized Linear Model. The main difference between these two regressions is whether the outcome Y is binary. The linear regression allows the outcome to exceed the zero to one range, while the logistic regression is more applicable to the outcome limited in zero to one. Another difference is that logistic regression calculates the maximum likelihood equations from the probability distribution of the dependent variables.

3.2 Model Analysis

First of all, try to use glm() in R to fit the data since the outcome is binary in the dataset. Before applying the model, splitting the data into training and testing subgroups is essential because it is a large dataset. Splitting the dataset effectively can protect against overfitting. (Figure 3.2) Also, since the training data establish the model, it can help to predict testing data.

Figure 3.2: Split Dataset

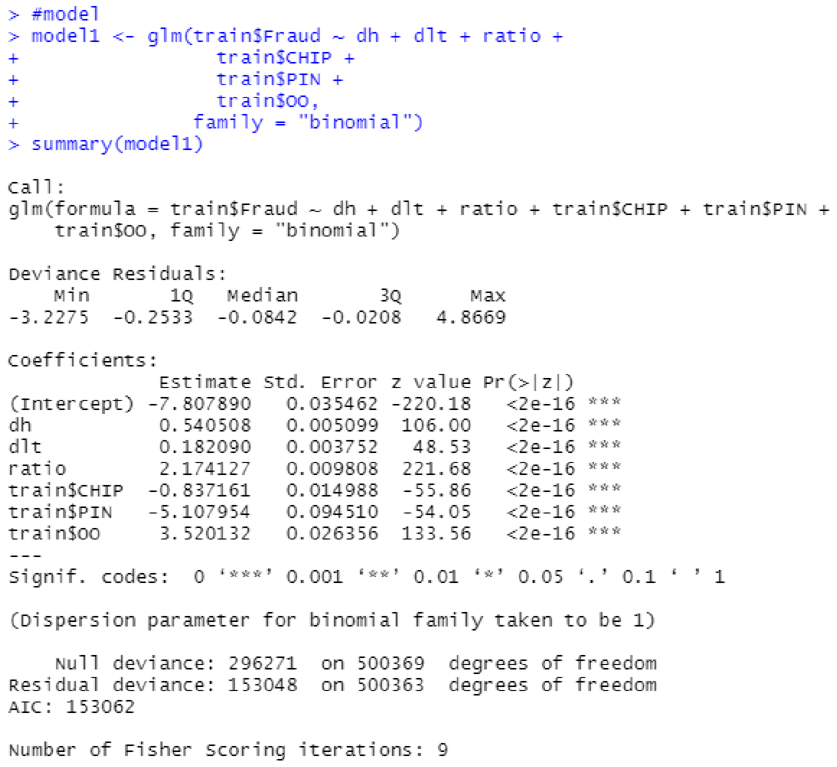


```
##Model 1 - Logistic
#Split data
set.seed(1)
sample <- sample(c(TRUE, FALSE), nrow(df), replace=TRUE, prob=c(0.5, 0.5))
train <- df[sample, ]
test <- df[!sample, ]
```

After randomly splitting data by fifty percent, model one is built based on the training data:

$$\text{Model One} = \text{FRAUD} \sim \log(\text{DH}) + \log(\text{DLT}) + \log(\text{RATIO}) + \text{CHIP} + \text{PIN} + \text{OO}$$

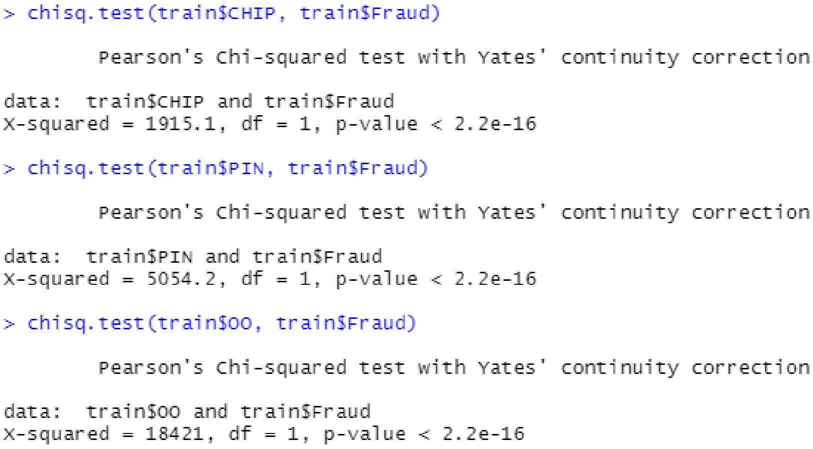
Figure 3.3: Logistic Regression Model One Summary



```
> #model
> model1 <- glm(train$Fraud ~ dh + dlt + ratio +
+                 train$CHIP +
+                 train$PIN +
+                 train$OO,
+               family = "binomial")
> summary(model1)

Call:
glm(formula = train$Fraud ~ dh + dlt + ratio + train$CHIP + train$PIN +
    train$OO, family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2275  -0.2533  -0.0842  -0.0208   4.8669

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -7.807890   0.035462 -220.18   <2e-16 ***
dh           0.540508   0.005099  106.00   <2e-16 ***
dlt          0.182090   0.003752   48.53   <2e-16 ***
ratio        2.174127   0.009808  221.68   <2e-16 ***
train$CHIP  -0.837161   0.014988  -55.86   <2e-16 ***
train$PIN   -5.107954   0.094510  -54.05   <2e-16 ***
train$OO     3.520132   0.026356  133.56   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 296271  on 500369  degrees of freedom
Residual deviance: 153048  on 500363  degrees of freedom
AIC: 153062

Number of Fisher Scoring iterations: 9
```

From the model one summary (Figure 3.3), the p-value for all variables is less than 0.05, typically considered statistically significant. Since the repeat retailer variable has a p-value greater than 0.05, which is not significant and is not considered this variable in model one. Moreover, from the result of slopes, chip and pin have a negative relationship with fraud, as the previous correlation matrix plot shows from the exploratory data analysis part. Therefore, using the PIN is a safer way to avoid fraud than using a credit card chip. Furthermore, online ordering and a high purchase price may increase the probability of fraud.

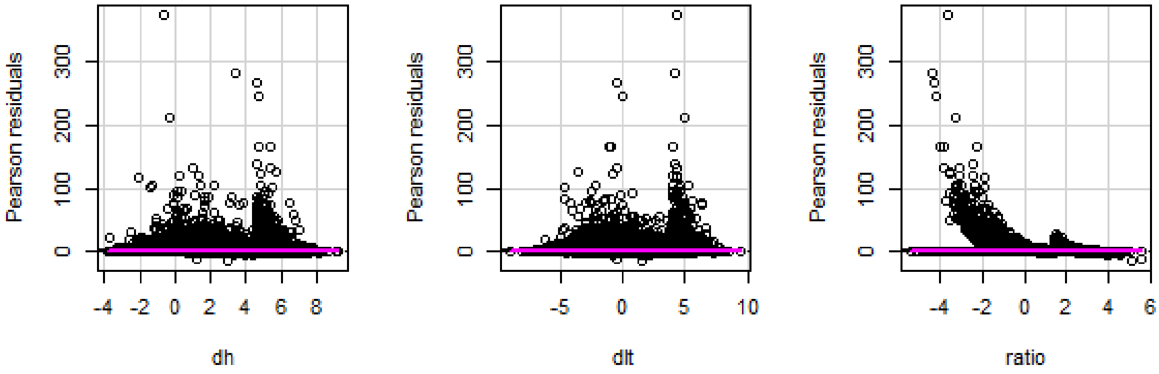
Figure 3.4: Chi-square Test



```
> chisq.test(train$CHIP, train$Fraud)

        Pearson's Chi-squared test with Yates' continuity correction

data:  train$CHIP and train$Fraud
X-squared = 1915.1, df = 1, p-value < 2.2e-16

> chisq.test(train$PIN, train$Fraud)

        Pearson's Chi-squared test with Yates' continuity correction

data:  train$PIN and train$Fraud
X-squared = 5054.2, df = 1, p-value < 2.2e-16

> chisq.test(train$OO, train$Fraud)

        Pearson's Chi-squared test with Yates' continuity correction

data:  train$OO and train$Fraud
X-squared = 18421, df = 1, p-value < 2.2e-16
```

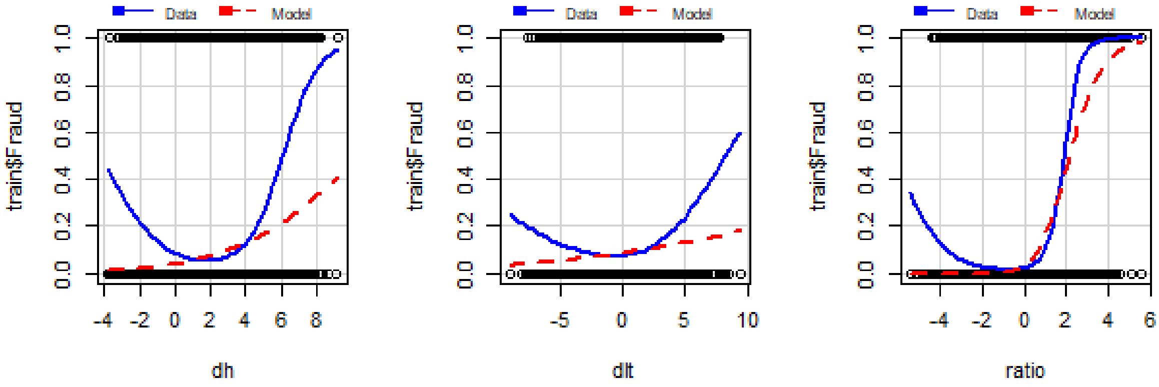
The chi-square test (Figure 3.4) is generally used to determine the differences between categorical variables in the dataset. Above are three groups of chi-square test results. The chi-square value for each variable and the outcome is 1915.1, 5054.2, and 18421, respectively, with one degree of freedom. Hence, duplicates conclude that all three variables have statistically significant fraud-related relationships.

Figure 3.5: Pearson Residuals Plot



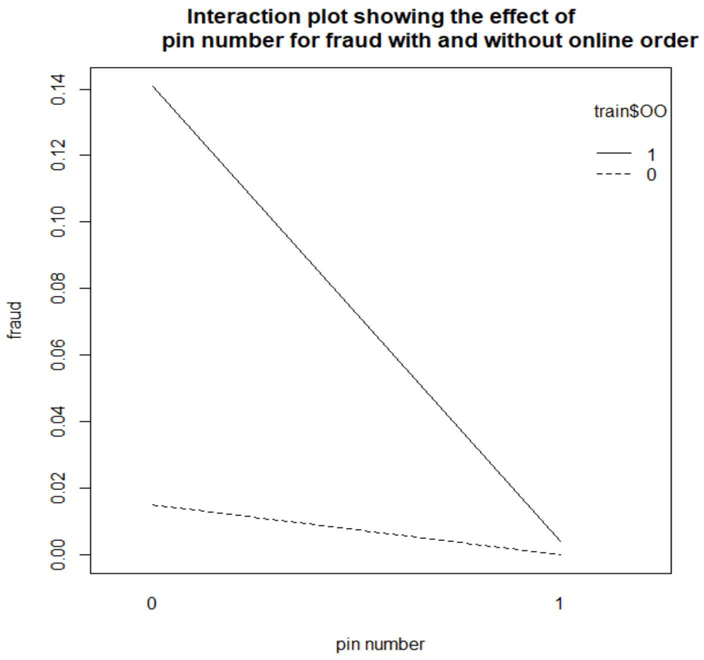
The Pearson Residuals Plots (Figure 3.5) show residuals in model one exist. However, the dark pink lines for these variables are all horizontal, which should be reasonable.

Figure 3.6: Marginal Model Plot



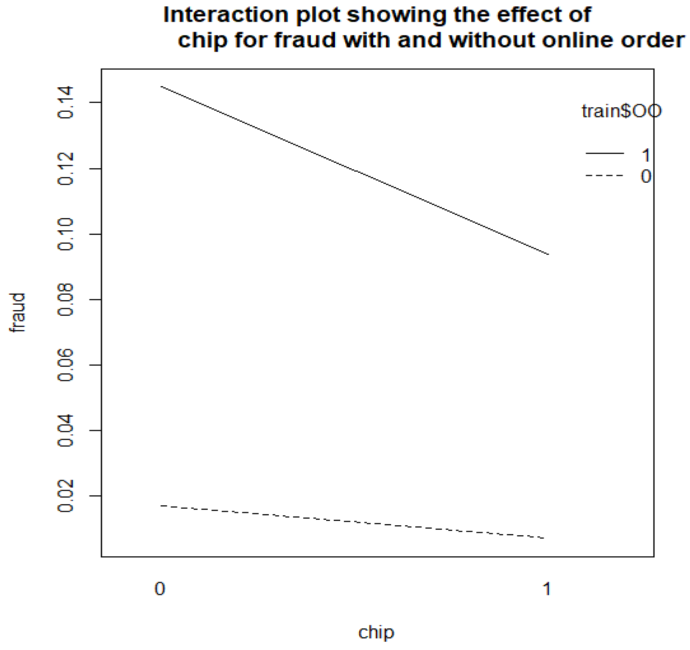
By comparing three marginal model plots (Figure 3.6), the variable ratio, or the right figure, is the fittest one among the three variables since the data line and the model line are almost coincident.

Figure 3.7: Interaction Plot One



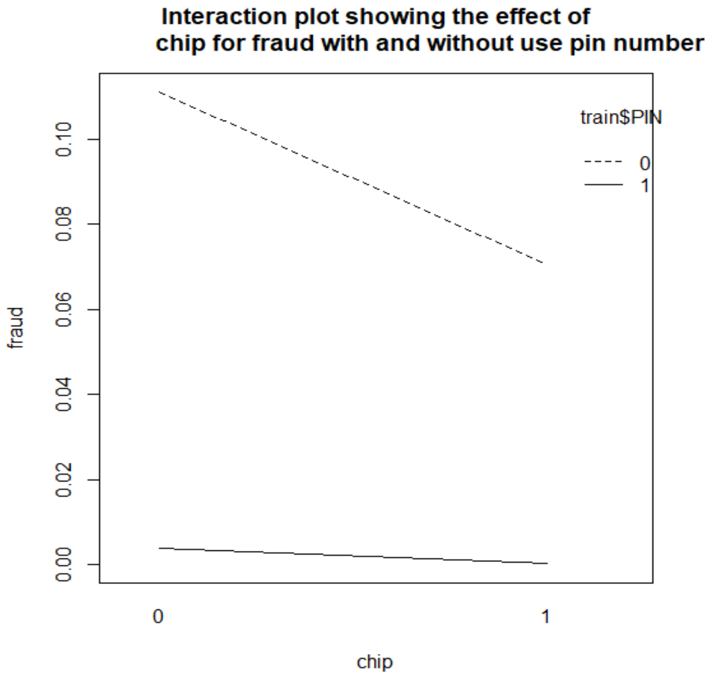
The two lines are not parallel when choosing two variables with the top two estimate values from model one to make the interaction effect plot. Therefore, the effect of a fraudulent transaction on a PIN varies with whether it is an online order. (Figure 3.7)

Figure 3.8: Interaction Plot Two



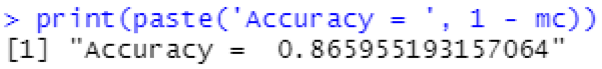
The two lines between the chip and the online order (Figure 3.8) are more parallel than the PIN and the online order, which may imply that the effect of fraud in a chip way does not vary with if it is an online order or not. However, using a chip may not sharply reduce the risk of fraud if it is an online order. That can be explained because most online order does not use the chip. In other words, a chip is usually used in a point-of-sale (POS) terminal or a physical situation. Thus, online orders and chips are generally not interacted.

Figure 3.9: Interaction Plot Three



The interaction plots (Figure 3.9) that include variable PIN indicate that using the PIN during the transaction should be one of the best ways to avoid fraud. For example, the above plot shows that the fraud rate is meager if the transaction is with the PIN. And the chip should also be a valid way to reduce the risk of fraud.

Figure 3.10: Accuracy for Model One

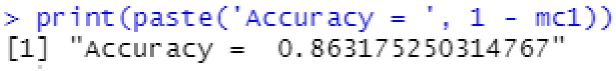


```
> print(paste('Accuracy = ', 1 - mc))
[1] "Accuracy =  0.865955193157064"
```

Once the training model is used to predict the testing data, it is time to evaluate the prediction accuracy. Setting the decision boundary to 0.5, if P (Y = 1 | X) is more significant than 0.5, then Y = 1. The accuracy of predicting Y is near 86.60% (Figure 3.10), which is a high accuracy rate. The correlation matrix plot vaguely shows a specific relationship between DH and RR. Hence adding the DH * RR to model two based on model one may be helpful, which keeps other variables the same.

Model Two = FRAUD ~ log(DH) + log(DLT) + log(RATIO) + CHIP + PIN + OO + (DH * RR)

Figure 3.11: Accuracy for Model Two



```
> print(paste('Accuracy = ', 1 - mc1))
[1] "Accuracy =  0.863175250314767"
```

The accuracy result for model two (Figure 3.11) is 86.32%, slightly lower than model one. Therefore, adding the combination of DH and RR seems unnecessary, and model one should be acceptable.

Chapter 4

Machine Learning

4.1 Introduction of Machine Learning

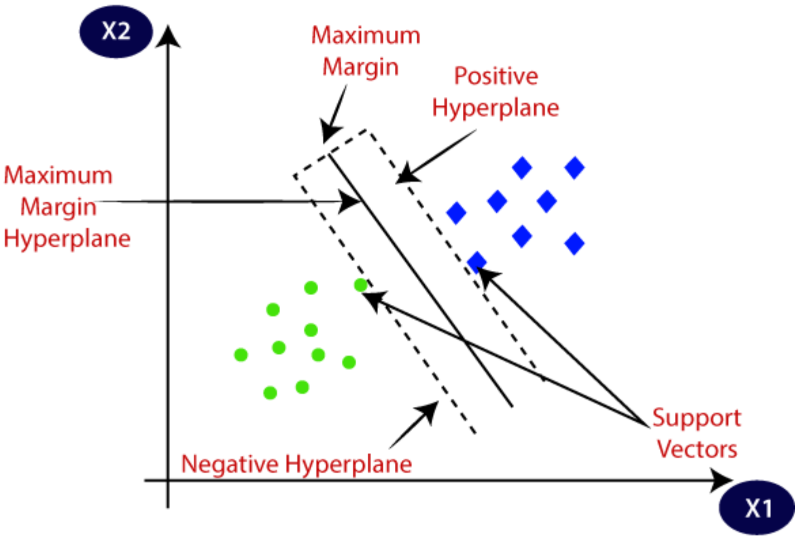
Machine learning has been an emerging direction or subject in recent years. Its popularity is like data science at the end of the last century. It mainly aims to improve the accuracy of machines by using big data and algorithms to let machines imitate the way of human learning. Machine learning has three major types: supervised, unsupervised, and reinforcement learning.

The significant difference between supervised learning and unsupervised learning is that the former uses labeled data to predict outcomes, while unsupervised focuses on the inherent structure of unlabeled data. According to specific data problems, such as the structure and quantity of data, and research goals, choosing the most suitable learning type should be reasonable. As mentioned earlier, credit card fraud detection is a typical classification problem. Therefore, supervised learning should be the most rational approach.

4.2 Support Vector Machine

Support Vector Machine (SVM) is a supervised learning model that aims to deal with classification and regression, which should be fitted to the fraud detection problem. Basically, “the objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.” [7]. In order to separate two kinds of data, try to find an optimal hyperplane from multiple hyperplanes which has the maximum margin.

Figure 4.1: Explanation of SVM [8]



In geometry way,

$$S_i = x_i^T \beta = x_i^T w + b$$

$$w\ stands\ for\ weights\ or\ coefficients; b\ stands\ for\ bias\ or\ interception$$

In general, maximum margin means to find the $\min(w, b)\ \frac{1}{2}|w|^2$

$$\begin{cases} y_i = +1, < x_i, w > +b \geq 1 \\ y_i = -1, < x_i, w > +b \leq -1 \end{cases} \Rightarrow y_i(< x_i, w > +b) \geq 1$$

Similar to the logistic regression analysis, for the data preparation part, keep applying the log transformation to three variables, DH, DLT, and RATIO. After that, we still need to split the entire dataset into X_train, X_test, y_train, and y_test by fifty percent.

Implementing the linear SVM and the gaussian kernel SVM to the training data by changing the C and gamma degree and reporting the training and testing loss to compare each model.

For the linear SVM,

Table 4.1: Accuracy Results for Linear SVM

Training accuracy 1	0.96282
Training accuracy 2 (C = 2)	0.962824
Training accuracy 3 (C = 3)	0.962832
Testing accuracy 1	0.185834
Testing accuracy 2 (C = 2)	0.185862
Testing accuracy 3 (C = 2)	0.185872

The linear SVM model is not fitted with the dataset correctly since the accuracy results for all three testing data are low. (Table 4.1) Therefore, this model will not be further considered.

For the gaussian kernel SVM,

Table 4.2: Accuracy Results for Gaussian Kernel SVM

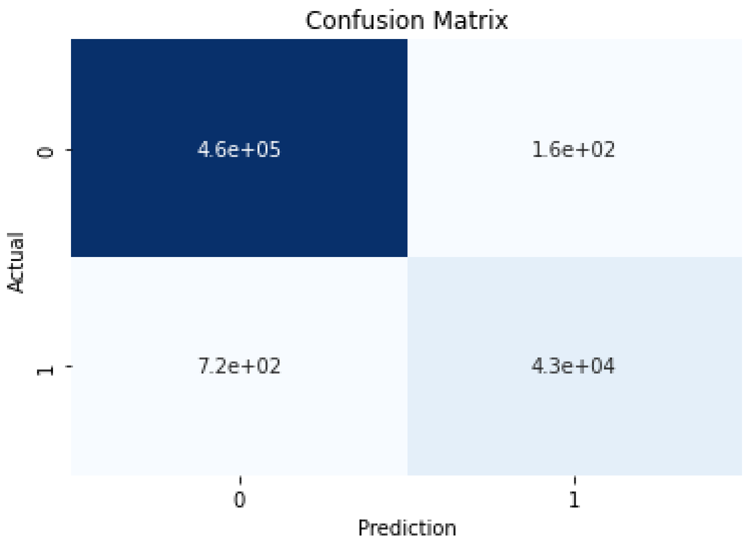
Training accuracy 1	0.9993
Training accuracy 2 (C = 4, gamma = 2)	0.999636
Training accuracy 3 (C = 5, gamma = 3)	0.999716
Testing accuracy 1	0.541282

Testing accuracy 2 (C = 4, gamma = 2)	0.737062
Testing accuracy 3 (C = 5, gamma = 3)	0.872564

The previous results (Table 4.2) clearly indicate that the gaussian kernel SVM performs better than the linear SVM for testing accuracy. Furthermore, and according to the table, it is obvious to find that the testing accuracy increased significantly as the C and gamma increased. In short, choosing model three, with C = 5 and gamma = 3, applies the model evaluation since it has the highest accuracy in training and testing data.

The confusion matrix can identify the difference between actual values and predicted values. It contains true positive, true negative, false positive, and false negative. In other words, a type one error is a false positive, while a type two error is a false negative.

Figure 4.2: Confusion Matrix for SVM



Furthermore, these four results can calculate recall and precision. The recall represents the correctly predicted rate from all the positive classes. And precision indicates the real positive rate from all the predicted positive classes.

$$Recall = \frac{TP}{TP + FN} = 98.35\%$$

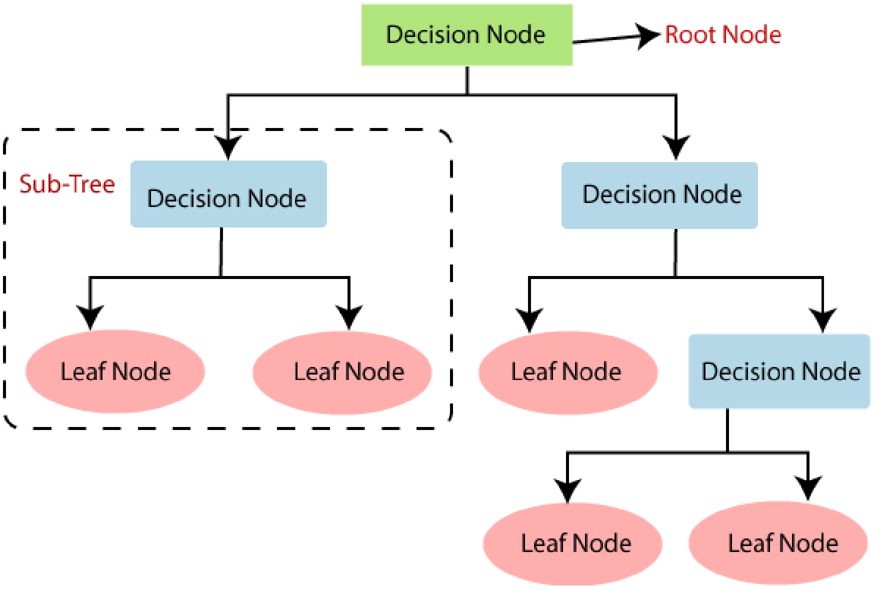
$$Precision = \frac{TP}{TP + FP} = 99.63\%$$

Generally, the results of recall and precision should be as high as possible. In this case, these two results are nearly to 100%, which means the gaussian kernel SVM with C = 5 and gamma = 3 perform ideally in this dataset.

4.3 Decision Tree

The decision tree is also a supervised learning model that splits the data continuously by a certain parameter. It is also used to solve regression and classification problems. The below figure illustrates: “The tree can be explained by two entities, namely decision nodes, and leaves. The leaves are the decisions or the final outcomes. And the decision nodes are where the data is split.” [9]. Each decision node has a characteristic condition and the next node is decided by the condition of the previous node.

Figure 4.3: Decision Tree Basic Structure [10]

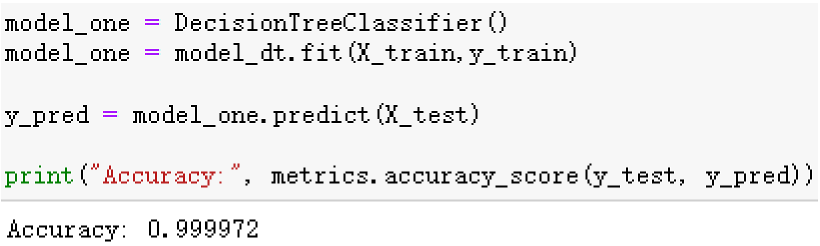


As the comparison between the decision tree and logistic regression model or support vector machine used in the previous chapter, the decision tree can deal with the collinearity better than the logistic regression and SVM, and it is more suitable for the categorical value. However, one of the disadvantages is that the decision tree needs to derive the significance of features.

The data preparation part is similar to the support vector machine since we still have to make a log transformation to the three independent variables and split the dataset by using the "train_test_split" function from the "sklearn" package into X training and testing, y training and testing by an equal percentage.

Model one contains all seven independent variables and applies the "decision tree classifier" function. The decision tree computes easier and faster than the support vector machine. Getting the model accuracy by using the training data to predict the testing data:

Figure 4.4: Accuracy for Decision Tree Model One



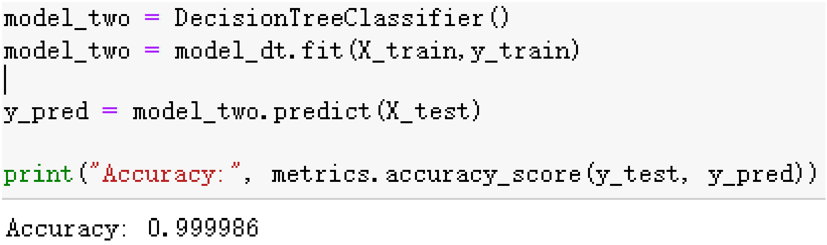
```
model_one = DecisionTreeClassifier()
model_one = model_dt.fit(X_train,y_train)

y_pred = model_one.predict(X_test)

print("Accuracy:", metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.999972
```

The accuracy is almost 100%, which fits the dataset. (Figure 4.4)

Model two is similar to model one except for deleting the repeated retailer variable. Because in the logistic regression model analysis, the repeated retailer has a large p-value and is not as significant as other variables. By erasing the non-significant variable to see whether the new model has a higher accuracy, although model one already has a remarkable accuracy result.

Figure 4.5: Accuracy for Decision Tree Model Two



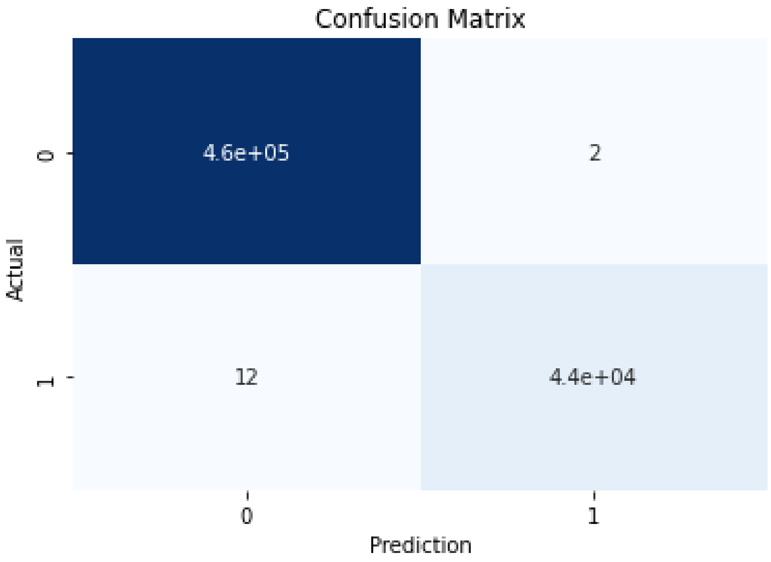
```
model_two = DecisionTreeClassifier()
model_two = model_dt.fit(X_train,y_train)

y_pred = model_two.predict(X_test)

print("Accuracy:", metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.999986
```

From the outcome, the new model's accuracy (Figure 4.5) is similar to model one and increases by 0.0014%.

Like the support vector machine evaluation, the confusion matrix is a straightforward way to evaluate the models. Below is the confusion matrix for the decision tree model one:

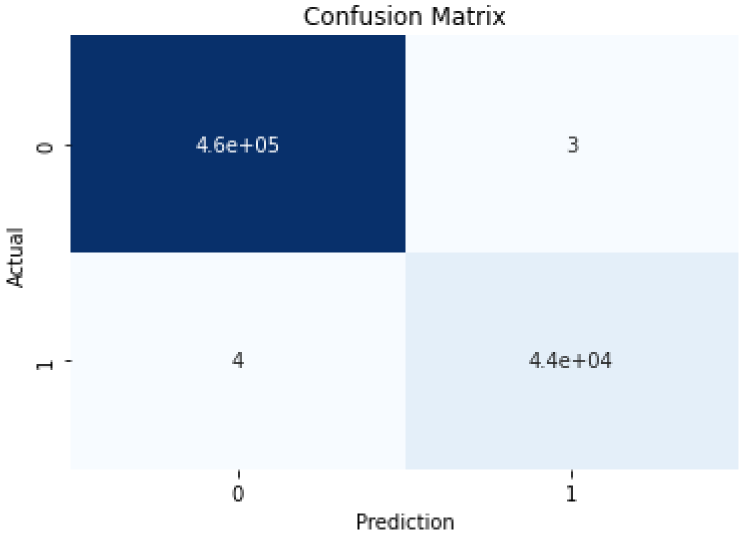
Figure 4.6: Confusion Matrix for Decision Tree Model One



The result is outstanding. There are only a few type one and two errors: 2 false positives and 12 false negatives. The precision achieves 99.9954% in this model one.

Here is the confusion matrix for the decision tree model two:

Figure 4.7: Confusion Matrix for Decision Tree Model Two

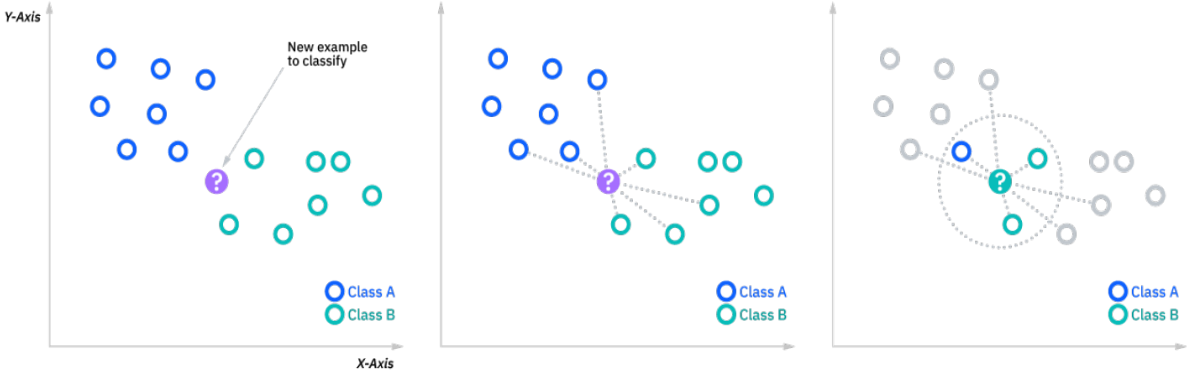


The outcome is even better for model two than model one since it contains fewer errors. There only have seven errors in total from the significant predictions. Moreover, both decision tree models perform better than the support vector machine model from the previous chapter.

4.4 k-NN

The k-nearest neighbor is also a supervised machine learning algorithm and a classic way to solve classification and regression problems. “KNN works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label.” [11]. In other words, k-NN is used to infer results from similar or close things, like its name.

Figure 4.8: The Procedure for k-NN [12]

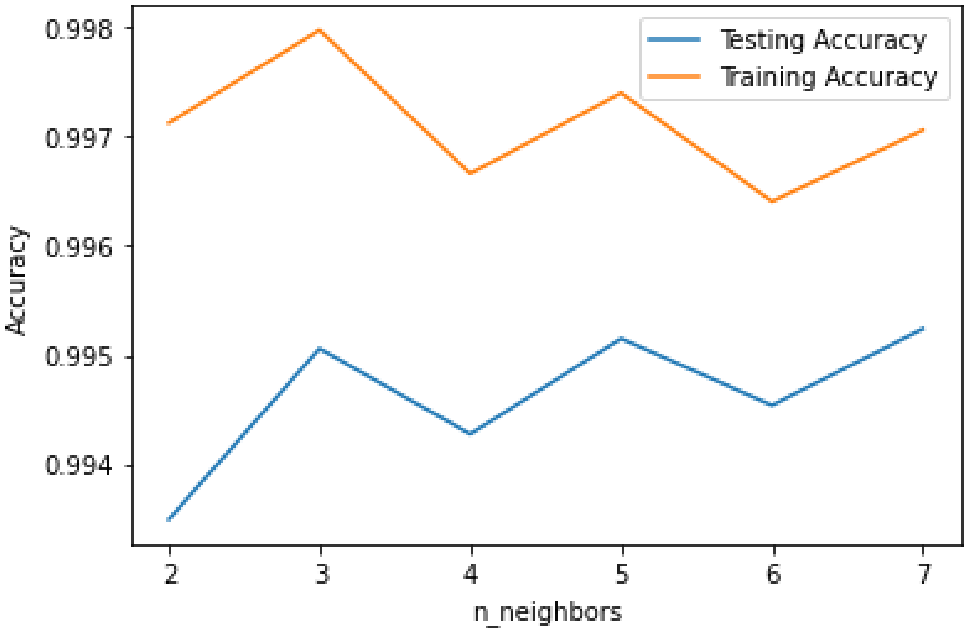


As the figure shows (Figure 4.8), label the existing examples first. When a new example appears, initialize K to the number of neighbors. Then calculate the distance between the new and known examples, and adjust the k value by sorting and ordering the distances. Finally, return the mode or mean of the K labels.

The beginning part of the k-NN analysis is the same as the previous model analysis. Letting the seven independent variables as X and fraud as Y. The non-binary variables need to be transformed into the log first, and afterward, continued to split the data into test and train groups by fifty percent each.

The foremost step is to find the best k value by creating a loop and making a range to allow the machine to try each k value. And the minimum value is set to two to avoid the overfitting issue. From the neighbor result plot, k equals three best fits the dataset. (Figure 4.9)

Figure 4.9: Accuracy for Different K Values



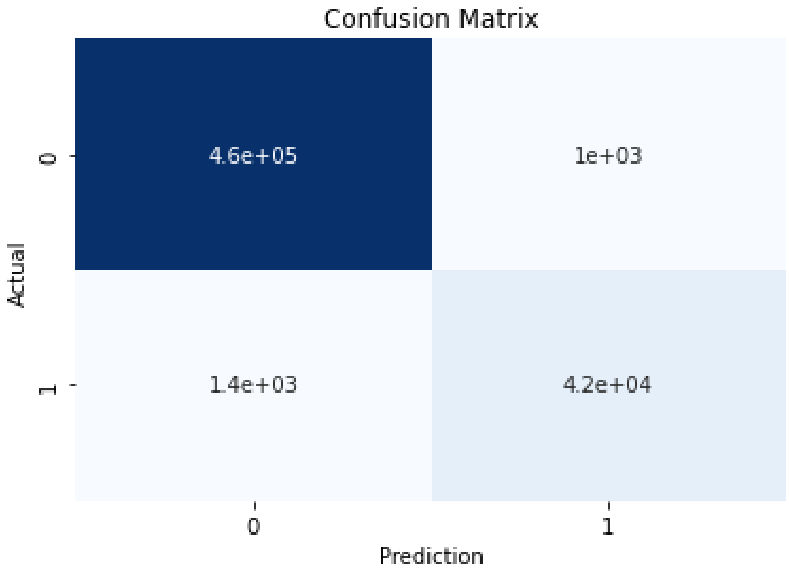
After determining the k value, there has a function named "KneighborsClassifier" from the "sklearn" package that can contribute to the k-NN model. The model accuracy is around 99.50%. (Table 4.3)

Table 4.3: Accuracy Results for k-NN

Training Accuracy	0.997918
Testing Accuracy	0.994992

Applying the confusion matrix for the k-NN like the previous steps. (Figure 4.10)

Figure 4.10: Confusion Matrix for k-NN



The result is not particularly ideal for the KNN when compared to the excellent results of the previous two models. It contains 1025 false positives and 1445 false negatives. However, the precision for this model still hits 97.63%, which is a high exactness.

Chapter 5
Conclusion

5.1 Conclusion

With the popularity of credit cards, cash has gradually faded out of public life, and credit card transactions have become one of the most common transaction methods worldwide. Overall, the principal target of this study is to figure out what variables are relative to credit card fraud and use logistic regression as the primary statistical model and three supervised machine learning models to predict the probability of fraud occurring after cleaning and transforming the significant independent variables.

The original seven independent variables contain binary variables, like repeat retailer, used chip, pin number, and online order, and three numerical variables. As can be seen from the Exploratory Data Analysis section, the higher or more outrageous the amount of credit card transactions, and the farther the transaction takes place from home or from the place where the previous transaction is, the more likely it is a fraud transaction. Also, online orders are more prone to fraud. The use of pins and chips can effectively reduce the occurrence of fraud. Additionally, the correlation plot indicates that RR has no relationship with fraud. The bar plot of the fraud statistics proved that credit card is generally a safe purchase. After all, the fraud rate is only 8.74%, which means that the number of fraudulent transactions is less than one out of every ten credit card transactions. However, the number of frauds remains particularly terrifying due to the large base of credit card transactions worldwide, such as 10 billion.

Table 5.1: Model Comparison

Model	Accuracy
Logistic Regression	86.60%
Support Vector Machine	87.26%
Decision Tree	99.99%
K-Nearest Neighbors	99.50%

To compare all models, it is clear that the decision tree's accuracy is surprisingly high. Meanwhile, the testing result for k-NN is unexpectedly higher than SVM. On the contrary, the accuracy of logistic regression is the lowest. However, the accuracy of all models is sufficiently high and should all be fitted methods to predict credit card transaction fraud.

For the logistic regression model, only apply six variables from seven since the repeat retailer has a relatively significant p-value. In addition, although there is a relationship between DH and RR, adding the combination of them to the model does not improve the accuracy of the model. For the three machine learning models, apply all seven variables at the beginning to check whether the accuracy is considerable. From the accuracy results, linear SVM is unsuitable for this dataset. Although the accuracy of training is adequately high, after using training data to predict testing data, the accuracy is less than 20%. The Gaussian Kernel SVM fits the dataset better. Furthermore, the accuracy has been significantly improved with the adjustment of the c and gamma values. The most incredible part of the whole study is that the decision tree model terrifically fits the dataset. Although all variables have been inferred with extremely high accuracy, the accuracy can be further improved after deleting RR in the model, like logistics regression. However, there is little difference between the two models since the rounding accuracy rates are 100%. Furthermore, the application of k-NN is also perfect, although it is less extraordinary than the decision tree. After finding the best k value, the model accuracy is also close to 100%. Last but not least, including RR has little effect on the three machine learning models, and their precision rates are all over 97%.

According to the current results, the following points may help avoid fraud:

1. It is best to reduce online shopping frequency, mainly when using a public or unsecured network, since it is difficult for the public to know whether their computers are safe.
2. Better to apply for a credit card with a chip. Also, it will be more secure if the PIN is requested during the transaction.

5.2 Limitation and Further Enhancement

First of all, although the number of samples is as many as one million, compared to the billions of credit card transactions that occur worldwide every day, this sample only accounts for less than one-thousandth of the global daily number. Thus, the sample size is still not large enough. Also, the background of the dataset needs to be adequate. The data uploader should provide a detailed introduction, including but not limited to which country and region does these records originated from, how they were obtained, etc.

Secondly, the dataset is relatively clean when it is downloaded. For instance, there are no incomplete values such as zero or NA. Moreover, categorical binary variables have been converted into numerical variables to facilitate analysis. Almost all independent variables are particularly significant. Mining or getting source data from real life will definitely be more complicated. In addition, there may have more variables to be considered. For example, the ratio to the mean or mode purchase price may also be significant. And the store type of the transaction may be necessary, like a small store on the street may be more prone to fraud than a large chain store. Meanwhile, different countries may have different credit card fraud rates. Finally, more models, such as the random forest, can be applied during analysis. In addition, sensitivity, specificity, and AUC tests can be added for further analysis of the logistic regression model. Another improvement is for the Gaussian Kernel SVM, since the accuracy is significantly improved after applying different c and gamma, trying more combinations of c and gamma may further increase the model accuracy.

Reference

[1] “Credit Card Fraud”. Dhanush Narayanan R, 2022, https://www.kaggle.com/datasets/dhanushnarayananr/credit-card-fraud.

[2] “Credit Card Statistics”. Shift, 2021, https://shiftprocessing.com/credit-card.

[3] “What Is the Average Number of Credit Cards per US Consumer?”. Stefan Lembo Stolba, 2021, https://www.experian.com/blogs/ask-experian/average-number-of-credit-cards-a-person-has.

[4] “Global Network Cards — Purchase Transactions”. Nilson Report, 2018, https://nilsonreport.com/research_featured_chart.php.

[5] “24 Eye-Opening Credit Card Statistics for 2023”. Radovan Sekulic, https://moneytransfers.com/news/2022/08/09/credit-card-statistics.

[6] “A Comparison between Linear and Logistic Regression”. Medium, 2019, https://medium.com/@maithilijoshi6/a-comparison-between-linear-and-logistic-regression-8aea40867e2d.

[7] “Support Vector Machine — Introduction to Machine Learning Algorithms”. Medium, 2018, https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

[8] “Support Vector Machine Algorithm”. Javatpoint, https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm.

[9] “Decision Trees for Classification: A Machine Learning Algorithm”. Xoriant, https://www.xoriant.com/blog/decision-trees-for-classification-a-machine-learning-algorithm#:~:te

xt=Introduction%20Decision%20Trees%20are%20a,namely%20decision%20nodes%20and%20leaves.

[10] “Decision Tree Classification Algorithm”. Javatpoint, https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm.

[11] “Machine Learning Basics with the K-Nearest Neighbors Algorithm”. Medium, 2018, https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761.

[12] “K-Nearest Neighbors Algorithm”. IBM, https://www.ibm.com/topics/knn.