"Credit Card Fraud Detection *using* Logistic Regression *in* SPSS"

(A Machine Learning Approach)

Presented By:

Likitha { 21-375-015 }

Nissi Divija { 21-375-023 }

Bhanuprakash { 21-375-061 }

Abstract

This work is focused on Credit Card Fraud Detection (CCFD),a serious problem in real world scenarios. As compared to the earliest times ,nowadays credit card frauds are drastically increased in numbers associated with the increasing in use of credit cards.

This effect was due to fraudsters are using fake identities to trap users in order to get money .So to eradicate or to control these frauds ,we came up with the idea of supervised machine learning model "Logistic Regression" ,which detects illegal and illicit transactions. With this we can trace the behavior and pattern of criminal activities by automating the process using respective algorithms.

From the conclusion of this entire work ,the performance of the model is based upon the Accuracy, Precision, Sensitivity and Specificity. So , our model logit classifier resulted in the best accuracy of ***97.20%*** respectively.

Keywords: Logistic regression, criminal transactions, illegal and illicit transactions,

INDEX

Introduction :

In this case study, we will explore how machine learning algorithms in SPSS were used to detect fraudulent transactions. The dataset used for this study is the Credit Card Fraud Detection dataset, which contains information on credit card transactions, including fraudulent and non-fraudulent transactions.

Credit card fraud is a major issue for financial institutions, businesses, and consumers that affects millions of people around the world each year. As the use of credit cards continues to grow, so does the prevalence of fraud.

According to a report by the Federal Trade Commission, there were over 1.4 million reports of fraud in the United States in 2020, with losses exceeding $3.3 billion.

In order to combat this issue, financial institutions and other organizations use various tools and techniques to detect and prevent fraudulent activities. This highlights the importance of developing effective methods for detecting and preventing credit card fraud.

With the increasing reliance on electronic transactions, credit card fraud is becoming more common and sophisticated. In such a way that statistical analysis techniques can help mitigate the risks associated with it.

Questionnaire:

✓ **What are the key factors that contribute to credit card fraud?**

The key factors that contribute to credit card fraud can vary depending on the specific circumstances of the fraudulent activity. However, some common factors that have been identified…

Stolen or compromised card information: This can occur through various means such as phishing scams, skimming devices, or data breaches.

Skimming: The thief uses a device to skim the information from the magnetic stripe on your credit card when you make a purchase at a retail location.

Phishing: The thief sends you an email or text message posing as a legitimate company, such as your bank or credit card issuer, and requests your credit card information.

Hacking: The thief gains access to a database containing your credit card information through a data breach or cyberattack.

Unauthorized use of a legitimate card: This can occur when someone other than the cardholder uses the card without their permission.

Counterfeit cards: This involves creating fake credit cards using stolen card information.

High-risk transactions: Transactions that are outside the cardholder's normal spending habits or occur in high-risk locations or industries (such as gambling or adult entertainment) can be indicators of fraudulent activity.

Poor security measures: Inadequate security measures on the part of the cardholder, merchant, or financial institution can make it easier for fraudsters to carry out fraudulent activities.

Identifying these factors and developing effective measures to prevent them can help reduce the incidence of credit card fraud.

Questionnaire:

- ✓ **What was the statistical hypothesis being tested in this analysis & level of significance?**

Null hypothesis (H_0): There is no significant difference in transaction patterns between legitimate and fraudulent credit card transactions.

Alternative hypothesis (H_1): There is a significant difference in transaction patterns between legitimate and fraudulent credit card transactions.

The logistic regression analysis will also provide estimates of the strength of the relationship between the *predictors* and the *outcome variable*, and will assess the statistical significance of these estimates.
If the p-value is less than the chosen level of significance (usually 0.05), the null hypothesis will be rejected in favor of the alternative hypothesis.

Overview : Binary Logistic Regression

A logistic regression (often referred to simply as Binary logistic regression), predicts the probability that an observation falls into one of two categories of a dichotomous dependent variable based on one or more independent variables that can be either continuous or categorical.

If, on the other hand, your dependent variable is a count, then Poisson regression is to be use. Alternatively, if you have more than two categories of the dependent variable, we chose multinomial logistic regression.

For example,

- To understand whether exam performance can be predicted based on revision time, test anxiety and lecture attendance (i.e., where the dependent variable is “Results", measured on a dichotomous scale –“arrears" or “no arrears" – and you have three independent variables: "revision time", "test anxiety" and "lecture attendance").

- Alternately, To understand whether drug use can be predicted based on prior criminal convictions, drug use amongst friends, income, age and gender (i.e., where the dependent variable is "drug use", measured on a dichotomous scale – "yes" or "no" – and you have five independent variables: "prior criminal convictions", "drug use amongst friends", "income", "age" and "gender").

Before we introduce you to the procedure, we need to understand the different assumptions that our data must meet in order for binary logistic regression to give you a valid result.

Assumptions :

Logistic regression is a statistical method used to model the relationship between a binary dependent variable and one or more independent variables. The following are some of the assumptions for logistic regression:

- ***Linearity***: There should be a linear relationship between the independent variables and the log odds of the dependent variable.
- ***Independence***: The observations should be independent of each other.
- ***Sample size***: The sample size should be large enough to provide reliable estimates of the coefficients.
- ***No multicollinearity***: There should be no high correlation between the independent variables.
- ***No influential outliers***: Outliers should not have a significant effect on the estimated coefficients.
- ***Binary response***: The dependent variable should be binary, i.e., it should only take two possible values.
- ***Absence of interactions***: There should be no interaction between the independent variables.
- ***No perfect separation***: There should be no subset of the data where the dependent variable is perfectly predicted by the independent variables.
- ***Model specification****:* The model should be correctly specified, including the functional form and the choice of independent variables.

Data Source - Overview:

- Credit Card Fraud Detection dataset: https://www.kaggle.com/mlg-ulb/creditcardfraud

In the Credit Card Fraud Detection dataset, the variables v1, v2, v3, ..., v28 are numeric input variables that represent the result of a principal component analysis (PCA) transformation.

PCA is a common technique used in machine learning to reduce the dimensionality of high-dimensional datasets by transforming the original features into a *new set of linearly uncorrelated variables*, called principal components.

The precise meaning of the v1, v2, ..., v28 variables in the Credit Card Fraud Detection dataset is unknown, as they were anonymized for privacy reasons. These variables may represent various features related to the credit card transactions, such as transaction amounts, locations, timestamps, or other characteristics.

In addition to the *v1-v28* variables, the dataset also includes the *Time* variable, which contains the number of seconds elapsed between each transaction and the first transaction in the dataset, and the *Amount* variable, which contains the transaction amount. Finally, the *Class* variable is a binary variable indicating whether a transaction is fraudulent (Class = 1) or legitimate (Class = 0).

In which it consists of *284807* instances with *31* attributes respectively.

Procedure :

Step 1: ***Import the dataset***

First, we need to import the Credit Card Fraud Detection dataset into SPSS. The dataset can be downloaded from Kaggle (https://www.kaggle.com/mlg-ulb/creditcardfraud). Once the dataset is downloaded, we can import it into SPSS by following these steps:

- ✓ Open SPSS and go to File > Import Data > CSV Data.
- ✓ Select the downloaded dataset and click Open.
- ✓ In the Import Wizard, select the first option "Read variable names from the first row of data."
- ✓ Click on OK button on Read CSV file dialog box.

.

Step 2: ***Data Preparation***

From the step 1 ,we can identify that our dataset variable types are in Scientific Notion type instead of Numeric type.so ,change all the Scientific type to Numeric type .

	Time	V1	V2	V3	V4	V5	V6	V7	
1	3.E+001	-5.35387763094460E-001	8.652678075752720E-001	1.35107628772237E+000	1.47575474497910E-001	4.33680212077009E-001	8.69829381161816E-002	6.93039311115721E-001	1.79
2	1.E+002	-2.42041282408364E+000	1.947885385067190E+000	5.53646042950925E-001	9.83068885733748E-001	-2.81518066165018E-001	2.40895755224446E+000	-1.40161343049739E+0...	-1.8
3	9.E+002	9.04289463743519E-001	-5.380552604101030E-001	3.96058034420560E-001	5.00679802458296E-001	-8.64473245499103E-001	-6.57198895275510E-001	2.72307766245696E-002	-2.9
4	9.E+002	1.20759584488179E+000	-3.686010004324570E-002	5.72103607295905E-001	3.73147507668388E-001	-7.09632532555591E-001	-7.13698000227395E-001	-1.81105220652192E-001	1.12
5	1.E+003	-2.44520387893208E+000	-5.054941267194030E-001	1.64511138598915E-001	-4.49657965570802E-001	4.60783925955686E-001	-5.09034851750468E-001	-4.90477677412628E-001	1.06

Credit Card Fraud Detection using Logistic Regression

	Time	V1	V2	V3	V4	V5	V6	V7	
1	26	-.53538776309446	.865267807575272	1.35107628772237	.14757547449791	.43368021207701	.08698293811618	.69303931111572	.1
2	145	-2.42041282408364	1.947885385067190	.55364604295093	.98306888573375	-.28151806616502	2.40895755224446	-1.40161343049739	-.1
3	919	.90428946374352	-.538055260410103	.39605803442056	.50067980245830	-.86447324549910	-.65719889527551	.02723077662457	-.0
4	919	1.20759584488179	-.036860100043246	.57210360729591	.37314750766839	-.70963253255559	-.71369800022740	-.18110522065219	.0
5	1074	-2.44520387893208	-.505494126719403	.16451113859892	-.44965796557080	.46078392595569	-.50903485175047	-.49047767741263	1.0

Now ,we need to prepare the dataset. This includes removing any unnecessary columns and handling missing values. In this data the "Class" column indicates whether a transaction is fraudulent or not (1 = fraudulent, 0 = non-fraudulent).

Lets look for the missing values

- Analyze > Descriptive statistics > Frequencies > chose all desired variables > Charts (Bar charts) >uncheck Frequency tables and click OK.

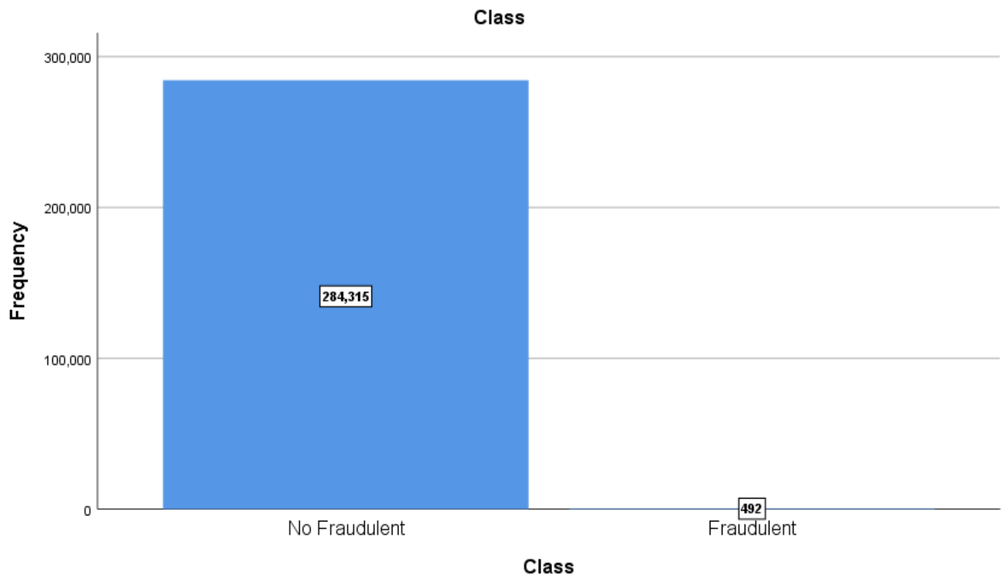
statistics

		Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13
N	Valid	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807
	Missing	0	0	0	0	0	0	0	0	0	0	0	0	0	0

statistics

		V15	V16	V17	V18	V19	V24	V26	V27	V28	V20	V21	V22	V23	V25	Amount
N	Valid	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807	284807
	Missing	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Credit Card Fraud Detection using Logistic Regression



Notice how *imbalanced* is our original dataset! Most of the transactions are non-fraud. If we use this data frame as the base for our predictive models and analysis we might get a *lot of errors* and our algorithms will probably *overfit* since it will "assume" that most transactions are not fraud. But we don't want our model to assume, we want our model to *detect patterns* that give signs of fraud!
So, We can have a *SMOTE* (Synthetic Minority Oversampling Technique) is one of the most commonly used *oversampling* methods to solve imbalance problems.

SMOTE (Synthetic Minority Over-sampling Technique) is an oversampling method that creates synthetic examples for the minority class in imbalanced datasets. we make use of *PYTHON* to implement this.

Credit Card Fraud Detection using Logistic Regression

Pycode…,

```
from imblearn.over_sampling import SMOTE
import pandas as pd

# Load your data
data = pd.read_csv("C:\\Users\\Bhanu\\OneDrive\\Desktop\\Creditcard\\creditcard.csv")

# Split the data into predictor variables (X) and outcome variable (y)
X = data.drop('Class', axis=1)
y = data['Class']

# Create a SMOTE object with the desired oversampling ratio
smote = SMOTE(sampling_strategy='auto')

# Run SMOTE on your data
X_resampled, y_resampled = smote.fit_resample(X, y)

# Combine the resampled predictor and outcome variables into a new data frame
resampled_data = pd.concat([X_resampled, y_resampled], axis=1)

# Export the resampled data to a new CSV file
resampled_data.to_csv("C:\\Users\\Bhanu\\OneDrive\\Desktop\\Creditcard\\creditresampled_data.csv",
index=False)

print("success")
```

On successful compiling ,it’ll return a balanced dataset.with this we will move forward in doing Logistic regression model building

Credit Card Fraud Detection using Logistic Regression

Analysis on SMOTE resampled data :

- Load the dataset > preprocess data >plot them.
- Now our dataset consists of **568630** instances and **31** attributes

Statistics

Indicator of each last matching case as Primary

N	Valid	568630
	Missing	0

Lets find and remove the duplicate values :

- Data > Identify Duplicate Cases >select all the desired variables for the label " Define matching cases by " > click OK
- Now all the unique values have code of (1) and Duplicate values have code of (0).

Indicator of each last matching case as Primary

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Duplicate Case	8601	1.5	1.5	1.5
	Primary Case	560029	98.5	98.5	100.0
	Total	568630	100.0	100.0	

Now chose to separate the dataset without duplicates which can be used for analysis

Steps as follows :

- Data > Select Cases > chose the radio option “ If condition is satisfied “ >enter the if condition (=> PrimaryLast = 1) >click continue & OK.
- If you want to store it as a new dataset ,click on “ Copy selected cases to new data set “ and new set consists of a column Primary Last with “Primary Case “ as 1.(i.e.., new dataset ***Unique_set1.sav***)

Step 3: ***Model Building***

Next, we will build a machine learning model to predict fraudulent transactions using the remaining columns in the dataset. we will use logistic regression as the machine learning algorithm.

To split the *Unique_set1.sav* as Training and Test sets ,follow these steps :

- Data > select cases >Random sample of cases (80%) > filter out unselect cases
- Create train and test datasets using *$filter variable* and for 80% code (1) ,for 20% code (0).
- Now we can extract the training and testing sets using respective methods.

Here, Training set had 447946 instances/cases

Testing set had 112083 instances/cases

Step 4: **Feature-Selection** :

Correlation matrices helps in understanding our data. We want to know if there are features that influence heavily in whether a specific transaction is a fraud. However, it is important to know which features have a high positive or negative correlation with regards to fraud transactions.

Credit Card Fraud Detection using Logistic Regression

From the correlation matrix ,we had

Negative Correlations: V10, V12, V14 and V17 are *negatively correlated*, the lower these values are, more likely results in a fraud transaction.

Positive Correlations: V2, V4, V11, and V19 are *positively correlated*. the higher these values are, the more likely results in a fraud transaction.

So we'll consider these variables as they are effective with the dependent variable.

To build the model, follow these steps:

- Go to File > open data > choose the training set (*Unique_trainset.sav*)> Click Open.
- Analyze > Regression > Binary Logistic.
- In the Logistic Regression dialog box, select "Class" as the dependent variable and choose correlation variables as the independent variables into "Covariates" label .
- Click on Save and choose predicted values as "Probabilities" and Continue…
- Choose Options and select Classification plots, Hosmer-Lemeshow goodness of fit, Case wise listing of residuals and CI for exp(B) .
- Click on "At last step" from Display and continue
- Choose the method as "Enter " .Click OK to run the analysis.

Result :

Logistic Regression

Case Processing Summary

Unweighted Cases[a]		N	Percent
Selected Cases	Included in Analysis	447946	100.0
	Missing Cases	0	.0
	Total	447946	100.0
Unselected Cases		0	.0
Total		447946	100.0

a. If weight is in effect, see classification table for the total number of cases.

Dependent Variable Encoding

Original Value	Internal Value
Non Fraudulent	0
Fraudulent	1

- This part of the output tells you about the cases that were *included* and *excluded* from the analysis, the coding of the *dependent* variable.

Block 0: Beginning Block

- This part of the output describes a “null model”, which is model with no predictors and just the intercept.
- This is why you will see all of the variables that you put into the model in the table titled “Variables not in the Equation”.

Variables in the Equation

		B	S.E.	Wald	df	Sig.	Exp(B)
Step 0	Constant	-.023	.003	58.610	1	.000	.977

Variables not in the Equation

			Score	df	Sig.
Step 0	Variables	V2	124109.227	1	.000
		V4	246636.903	1	.000
		V10	198379.630	1	.000
		V11	229263.517	1	.000
		V12	226071.856	1	.000
		V14	277326.429	1	.000
		V17	154622.438	1	.000
		V19	39949.130	1	.000
	Overall Statistics		293656.667	8	.000

Block 1: Method = Enter

- Its an overall test of the model (in the “Omnibus Tests of Model Coefficients” table) and the coefficients and odds ratios (in the “Variables in the Equation” table).

Omnibus Tests of Model Coefficients

		Chi-square	df	Sig.
Step 1	Step	546426.024	8	.000
	Block	546426.024	8	.000
	Model	546426.024	8	.000

The overall model is statistically significant, $\chi 2(8) = 546426.024$, $p < 0.05$

Model Summary

Step	-2 Log likelihood	Cox & Snell R Square	Nagelkerke R Square
1	74500.376[a]	.705	.940

a. Estimation terminated at iteration number 11 because parameter estimates changed by less than .001.

Variance explained:

- Using "Model Summary“ we can understand how much variation in the dependent variable can be explained by the model (the equivalent of $\boldsymbol{R^2}$ in multiple regression)..
- *Cox & Snell R Square* and *Nagelkerke R Square* values, which are both methods of calculating the explained variation. These values are sometimes referred to as *pseudo* R^2 values .However, they are interpreted in the same manner, but with more caution.
- Therefore, the explained variation in the dependent variable based on our model ranges from **70.5% to 94.0%**, depending on whether you reference the Cox & Snell R^2 or Nagelkerke R^2 methods, respectively.
- Nagelkerke R^2 is a modification of Cox & Snell R^2.

Block 1: Method = Enter

Hosmer and Lemeshow Test

Step	Chi-square	df	Sig.
1	2.513 E+ 12	8	.320

- Hosmer-Lemeshow goodness-of-fit test is a useful method to assess the fit of a logistic regression model
- A non-significant p-value (typically greater than 0.05) indicates that the model fits the data well and that the predicted probabilities of the model match the observed frequencies in each group.

Category prediction:

- Binary logistic regression estimates the probability of an event occurring (having Fraud cases). If the estimated probability of the event occurring is greater than or equal to 0.5,classifies the event as occurring (e.g., fraud is being present). If the probability is less than 0.5,classifies the event as not occurring (e.g., no fraud).

Classification Table[a]

			Predicted Class		Percentage Correct
	Observed		No Fraudulent	Fraudulent	
Step 1	Class	No Fraudulent	223716	2819	98.8
		Fraudulent	9723	211688	95.6
	Overall Percentage				97.2

a. The cut value is .500

Block 1: Method = Enter

Firstly, notice that the table has a **subscript** which states, "The cut value is .500". This means that if the probability of a case being classified into the "yes" category if greater than .500, then that particular case is classified into the "yes" category. Otherwise, the case is classified as in the "no" category .

- **Accuracy :** The percentage of cases that can be correctly classified as "no" fraud with the independent variables added (not just the overall model).
  The model now correctly classifies 97.2% of cases overall (in "Overall Percentage" row)
- **Sensitivity :** The percentage of cases that had the observed characteristic (for fraud) which were correctly predicted by the model (i.e., true positives).
  95.83 % of fraud cases were also predicted by the model to have fraud ("Percentage Correct" column in the “Fraud" row of the observed categories).

Variables in the equation :

The "Variables in the Equation" table shows the contribution of each independent variable to the model and its statistical significance. This table is shown below:

The Wald test ("**Wald**" column) is used to determine statistical significance for each of the independent variables. The statistical significance of the test is found in the "**Sig.**" column. From these results we can see that ***V4,V10, V11, V12, V14,V17,V19*** are added significantly to the model/prediction and remaining variables along with constant did not add significantly to the model.
So we can use the information in the "**Variables in the Equation**" table to predict the probability of an event occurring based on a one unit change in an independent variable when all other independent variables are kept constant.

Variables in the Equation

		B	S.E.	Wald	df	Sig.	Exp(B)	95% C.I.for EXP(B) Lower	Upper
Step 1[a]	V2	-.009	.006	2.570	1	.109	.991	.980	1.002
	V4	1.021	.008	14896.580	1	.000	2.776	2.731	2.822
	V10	-.936	.013	4966.278	1	.000	.392	.382	.403
	V11	.590	.010	3295.228	1	.000	1.804	1.768	1.841
	V12	-.997	.011	8291.973	1	.000	.369	.361	.377
	V14	-1.307	.010	15630.884	1	.000	.271	.265	.276
	V17	-.888	.013	4364.458	1	.000	.411	.401	.422
	V19	.389	.011	1279.731	1	.000	1.475	1.444	1.507
	Constant	-4.196	.017	59160.878	1	.000	.015		

a. Variable(s) entered on step 1: V2, V4, V10, V11, V12, V14, V17, V19.

Step 4: ***Model Evaluation***

Once the model is built, we need to evaluate its performance. We will use confusion matrix to evaluate the performance of the model.

To evaluate the model, follow these steps:

- To evaluate the performance of the model using a confusion matrix, go to Analyze > Descriptive Statistics > Crosstabs.
- In the Crosstabs dialog box, select "Class" as the row variable and compute a variable for “Predicted Probability > 0.5" as the column variable.
- Click OK to run the analysis.
- The resulting confusion matrix shows the number of true positives, true negatives, false positives, and false negatives.

Case Processing Summary

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
Class * Prob	447946	100.0%	0	0.0%	447946	100.0%

Class * Prob Crosstabulation

Count

		Prob		
		.00	1.00	Total
Class	No Fraudulent	223716	2819	226535
	Fraudulent	9723	211688	221411
Total		233439	214507	447946

Evaluation Metrics :

Few commonly used statistical measures in binary classification problems to evaluate the performance of a predictive model or a diagnostic test. They help to assess the accuracy, reliability, and predictive power of the model or test by comparing the predicted outcomes to the actual outcomes.

- **Accuracy**: It is the proportion of correctly classified samples out of the total number of samples. It is calculated as (True Positives +True Negatives) / (True Positives + False Positives + True Negatives + False Negatives).
- **Misclassification**: It is the proportion of incorrectly classified samples out of the total number of samples. It is calculated as (False Positives + False Negatives) / (True Positives + False Positives + True Negatives + False Negatives).
- **Sensitivity** (***True Positive Rate***): It is the proportion of true positives out of all actual positives. It is calculated as True Positives / (True Positives + False Negatives).
- **Specificity** (***True Negative Rate***): It is the proportion of true negatives out of all actual negatives. It is calculated as True Negatives / (False Positives + True Negatives).
- **Positive Predictive Value** (***PPV***): It is the proportion of true positives out of all predicted positives. It is calculated as True Positives / (True Positives + False Positives).
- **Negative Predictive Value** (***NPV***): It is the proportion of true negatives out of all predicted negatives. It is calculated as True Negatives / (False Negatives + True Negatives).

- **Positive Likelihood Ratio** (***PLR***): It is the ratio of true positives to false positives. It is calculated as Sensitivity / (1 - Specificity).
- **Negative Likelihood Ratio** (***NLR***): It is the ratio of false negatives to true negatives. It is calculated as (1 - Sensitivity) / Specificity.

Model Evaluation Statistics	Formulas	% Values for Train set	% Values for Test set
Accuracy	TP+TN/(TP+TN+FP+FN)	97.2	97.19
Misclassification	1-Accuracy	0.028	0.0281
Sensitivity(True Positive or Recall)	TP/(TP+FN)	95.83	95.85
Specificity (False Negative)	TN/(TN+FP)	98.69	98.64
Positive Predicted Value(PPV or Precession)	TP/(TP+FP)	98.7556	98.71
Negative Predicted Value(NPV)	TN/(TN+FN)	95.6086	95.62
Positive Likelihood Ratio	Sensitivity/(1-Specificity)	72.92	70.47
Negative likelihood Ratio	(1-Sensitivity)/Specificity	0.04221	0.04209
F1-Score	2*((PPV*Recall) / (PPV +Recall))	97.27081	97.25898

Credit Card Fraud Detection using Logistic Regression

Area Under the Curve

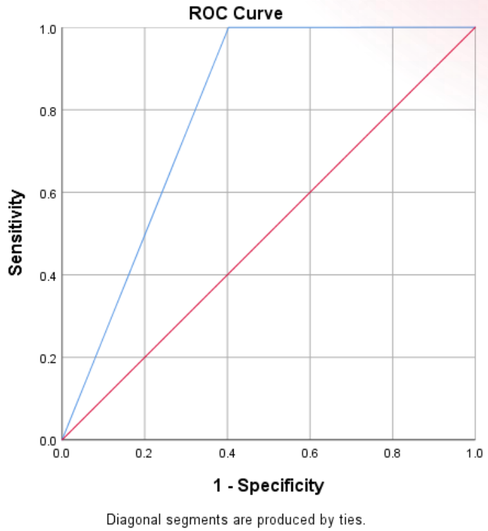
Test Result Variable(s): Class

Area	Std. Error[a]	Asymptotic Sig.[b]	Asymptotic 95% Confidence Interval	
			Lower Bound	Upper Bound
.798	.001	.000	.797	.800

The test result variable(s): Class has at least one tie between the positive actual state group and the negative actual state group. Statistics may be biased.

a. Under the nonparametric assumption

b. Null hypothesis: true area = 0.5



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.

For Train set:

The logistic regression model was statistically significant, $\chi^2(8) = 546426.064$, $p=0.00<0.05$. The model explained 94.00% (Nagelkerke R^2) of the variance and correctly classified 97.2% of cases accurately with only 12542 misclassified instances for 80% data respectively.

For Test set:

The logistic regression model was statistically significant, $\chi^2(8) = 136775.109$, $p=0.00<0.05$. The model explained 94.00% (Nagelkerke R^2) of the variance and correctly classified 97.2% of cases accurately with only 3155 misclassified for 20% of data.

References :

- ✓ 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.
- ✓ A machine learning based credit card fraud detection using the GA algorithm for feature selection Emmanuel Ileberi1*, Yanxia Sun1 and Zenghui Wang2
- ✓ 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).
- ✓ CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING Mr. Thirunavukkarasu.M1 ; Achutha Nimisha2 ; Adusumilli Jyothsna3

Thank You