1 INTRODUCTION

1.1 Overview

The main objective of this project is to perform predictive analysis on credit card transaction dataset using machine learning techniques and detect the fraudulent transactions from the given dataset. The focus is to identify if a transaction comes under normal class or fraudulent class using predictive models. Machine learning algorithms logistic regression will be implemented on the dataset, and the results will be reported.

1.2 Purpose

E-commerce has come a long way since its inception. It has become an essential means for most organizations, companies, and government agencies to increase their productivity in global trade. One of the main reasons for the success of e-commerce is the easy online credit card transaction. Whenever we talk about monetary transactions, we also have to take financial fraud into consideration. . As credit card transactions are the most common method of payment in recent years, the fraud activities have increased rapidly.

2 LITERATURE SURVEY

2.1 Existing problem

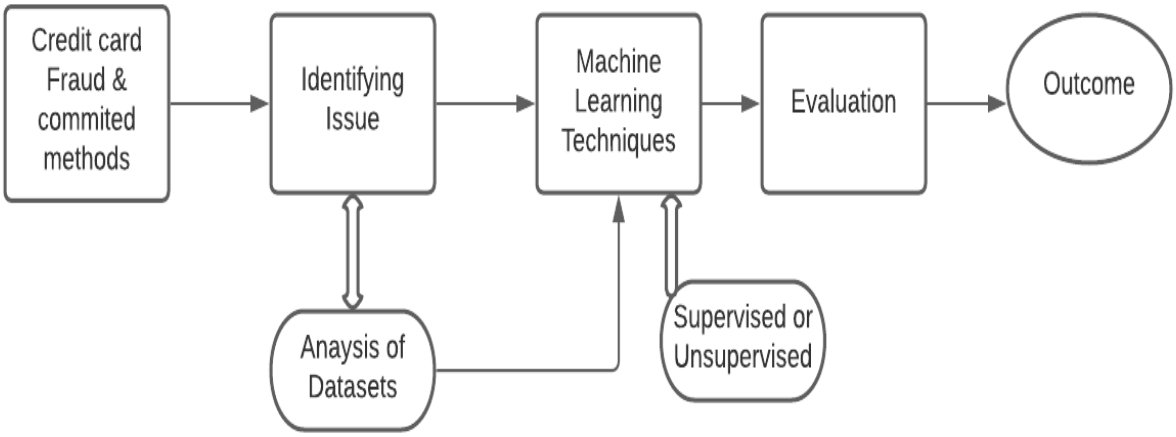
Credit card fraud is an ever-growing problem in today's financial market. There has been a rapid increase in the rate of fraudulent activities in recent years causing a substantial financial loss to many organizations, companies, and government agencies. The numbers are expected to increase in the future, because of which, many researchers have focused on detecting fraudulent behaviours early using advanced machine learning techniques. Predictive models such as logistic regression in combination with different resampling techniques have been applied to predict if a transaction is fraudulent or genuine.

2.2 Proposed solution

Multiple Supervised learning techniques are used for fraud detection. Different Supervised machine learning algorithms like Naive Bayes Classification, Least Squares Regression, Logistic Regression are used to detect fraudulent transactions in real-time datasets.

3.THEORITICAL ANALYSIS

3.1.Block diagram



Credit card fraud can occur online in a variety of ways. Online fraud is where a fraudster commits the fraud via the phone or the Internet with the card details. Offline fraud is committed when a stolen card is used physically to pay for goods or services.

Identify issues and challenges**:**

1. Enormous Data is processed every day and the model build must be fast enough to respond to the scam in time.
2. Data availability as the data is mostly private.
3. Misclassified Data can be another major issue, as not every fraudulent transaction is caught and reported.

Evaluation **:** Precision is the rate of true positives divided by the sum of true positives and false positives. Having a high precision means having a high measure of relevant results returned with limited irrelevant results. Recall, on the other hand, is the number of true positives divided by the sum of true positives and false negatives. A high recall indicates the model is able to successfully identify relevant results without mislabelling them as irrelevant. Depending on use case, one must evaluate whether to prioritize precision or recall.

Analysis of datasets**:** It contains only numerical input variables which are the result of a PCA transformation. Feature 'Fraud_risk' is the response variable and it takes value 1 in case of fraud and 0 otherwise."

Supervised and unsupervised**:** Supervised learning techniques are widely employed in credit card fraud detection, as they make use of the assumption that fraudulent patterns can be learned from an analysis of past transactions. Unsupervised learning techniques are employes to replace any missing values in data set and removal of subsets of datasets.

3.2.Hardware / Software designing

Eclipse - An Integrated development environment used for Java programming
Weka 3.8.5 - Collection of visualization tools and algorithms for data analysis and predictive modeling

4 EXPERIMENTAL INVESTIGATIONS

4.1 Machine Learning

In general context, machine learning can be defined as an artificial intelligence that provides the system the capability to learn from the experience automatically without human intervention and aims to predict the future outcomes as accurately as possible utilizing various algorithmic models. Machine Learning is very different from conventional computation approaches, where systems are explicitly programmed to calculate or solve a problem. Machine learning deals with the input data that are used to train a model where the model learns different patterns in the input data and uses that knowledge to predict unknown results.

4.2 Supervised Learning

Supervised learning is a machine learning approach in which both input and output labels are provided to the model to train. The supervised model uses the input and output labelled data for training, and it extracts the patterns from the input data.

Supervised learning can be formally represented as follows : Y = f(x) where x represents the input variables, Y denotes an output variable and f(X) is a mapping

function. The goal is to approximate a mapping function such that when an unseen input is given to the mapping function, it can predict the output variable (Y) correctly. Furthermore, supervised learning has two sub-categories: classification and regression. In a classification problem, the output variable is a category (e.g., fraud or genuine, rainy or sunny, etc.). In a regression problem, the output variable is a real value, (e.g., the price of a house, temperature, etc.). This thesis only deals with the classification problem.

4.3 Classification

Classification problem in machine learning can be done as the task of predicting the class label of a given data point. For example, fraud detection can be identified to be classification problem. In this case, the goal is to predict if a given transaction is fraud or genuine.

4.4. Logistic regression

Logistic regression is one of the most popular machine learning algorithms that is used for classification. Although the term ’regression’ appears in the name, it is not a regression algorithm. Logistic regression has its name as it was built on another very popular machine learning algorithm, linear regression, which is used for regression problems. In logistic regression, the prediction is expressed in terms of probability of outcome belonging to each class.

Since it should predict the probability of the outcome of belonging to each class, it uses a sigmoid function or a logistic function to squash the predicted real values between the range of 0 and 1.

$$\text{sigma}(z) = 1/_{1 + e\ z}$$

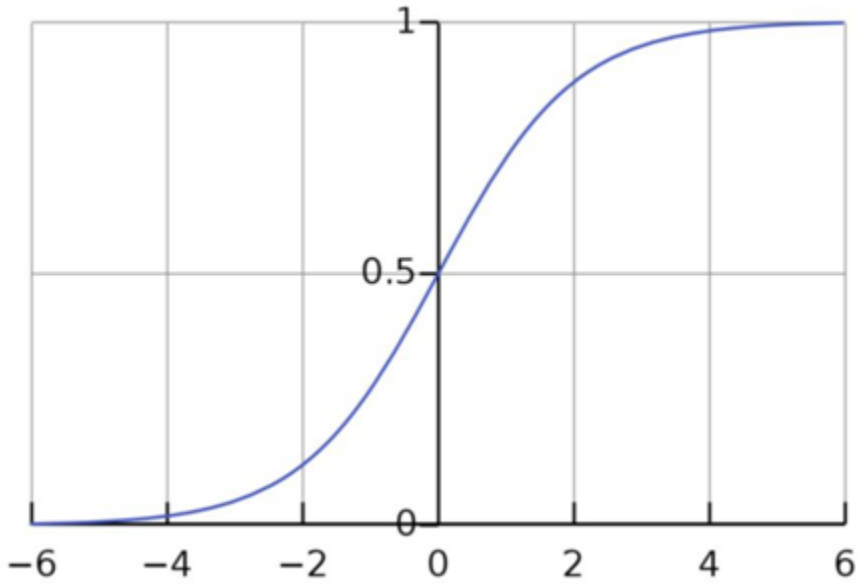


Figure : Sigmoid function graph

Figure shows how the sigmoid function looks like. In a classification problem where we have one independent variable ’x’ and one dependent variable ’y’, logistic regression can be represented as in equation 3. By default, logistic regression uses a threshold of 0.5 such that any probability below 0.5 is classified as class 0, and any probability above 0.5 is classified as class. This threshold can be adjusted according to the needs

4.5 Evaluation metrics

In machine learning, we train the model with the training data, and then we check the generalization capability of the model. In simple terms, we examine how the model performs when tested on data that was unseen. So how do we measure the performance of the model? We use evaluation metrics for evaluating the performance of the model depending on the nature of the problem (whether it is a regression or classification). In this section, we will only discuss the evaluation metrics related to the classification problem.

4.6 Confusion matrix

It is the most commonly used evaluation metrics in predictive analysis mainly because it is very easy to understand and it can be used to compute other essential metrics such as accuracy, recall, precision, etc. It is an n*n matrix that describes the overall performance of a model when used on some dataset, where n is the number of class labels in the classification problem. For binary classification, we have a 2x2 confusion matrix as shown in figure.

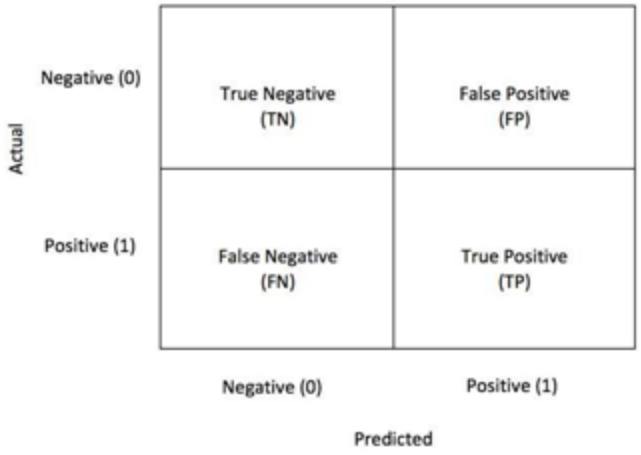


Figure : Confusion matrix

A confusion matrix is composed of statistics such as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) which are calculated using the combination of actual and predicted values.

True Positive (TP) is a case where the actual value was positive (e.g., fraud) and the predicted value is also positive

False Positive (FP) is a case where the actual value was negative (e.g., normal) but the predicted value is positive.

True Negative (TN) is a case where the actual value was negative (e.g., normal) and the predicted value is also negative.

False Negative (FN) is a case where the actual value was positive (e.g., fraud) but the predicted value is negative.

4.7 Recall

Recall, also known as sensitivity, is the fraction of true positives to the actual positive cases, which is shown in equation. In simple terms, recall is how many of true positives were found (recalled) out of all the true positive cases.

Recall =TP/(TP+FN)

4.8 Precision

It is the fraction of true positives over the true positives and false positives, which is shown in equation . In simple terms, precision is how many of the found cases were true positives

Precision:TP/(TP+FP)

4.9 F_1 Score

F_1 Score also called F score or F-measure is the harmonic mean of the recall and precision. Its value ranges from 0 to 1, where 0 is considered worst, and 1 is considered best.

F1 score = 2*precision*recall/(precision+recall)

4.10. Area Under Receiver Operating Characteristic curve

Area Under Receiver Operating Characteristic curve is one of the most widely used evaluation metrics in predictive analysis. It tells us how good a model performs when used at different probability thresholds. By default, a probability threshold of 0.5 is used for the classification problem. It is a plot between True positive rate (TPR), which is also called sensitivity and False Positive Rate (FPR). From the ROC curve, we can calculate the area under the curve which is the probability that a model will rank a randomly chosen positive instance higher than a

randomly chosen negative on. Figure shows an example of a ROC curve where the blue curve shows the ROC of the model and AUC is 0.966. Whereas, the red dotted line shows the ROC of a random model whose AUC is 0.5.

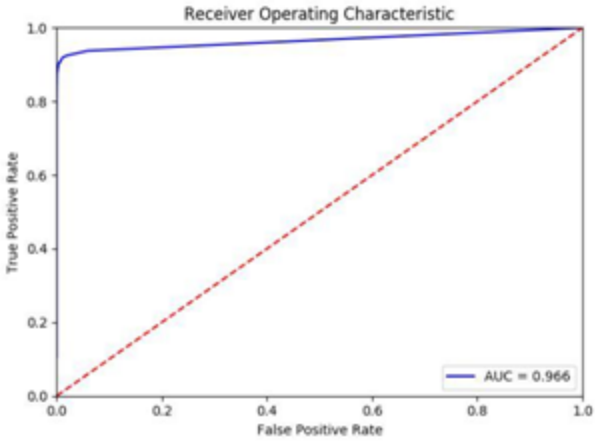
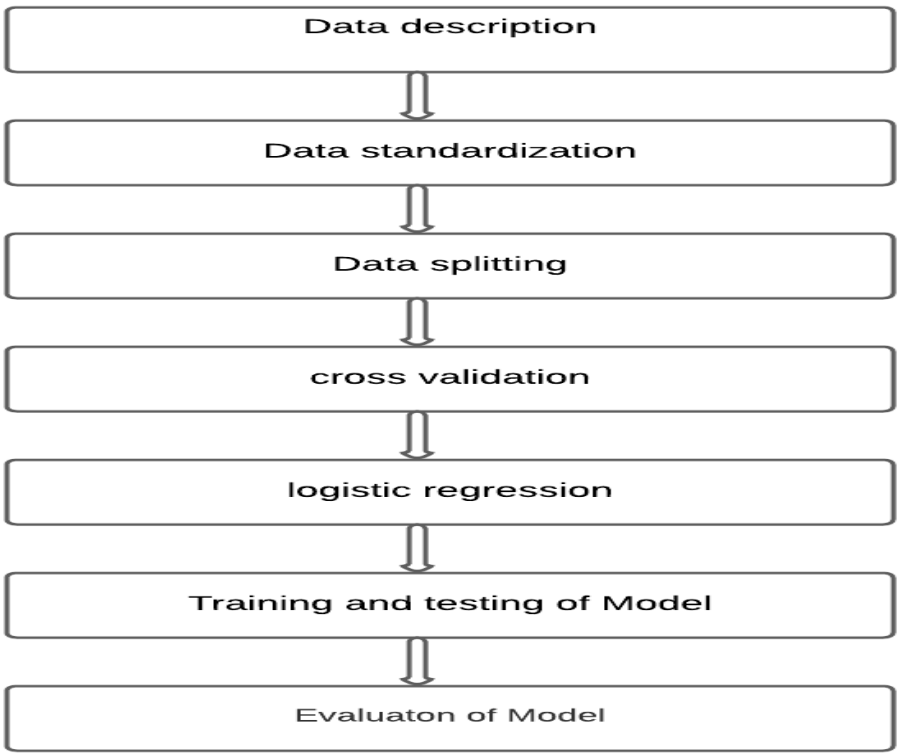


Figure: An example of ROC curve

5. FLOWCHART



6. RESULT

Classification report is given in the output as follows, where class 0 means the transaction was determined to be valid and 1 means it was determined as a fraud transaction. In this experiment after the classification of the data sets using logistic regression, we calculate four metrics used in evaluation those are:

True Positive Rate (TPR): The true positive rate is the probability that an actual positive will test positive.

True Negative Rate (TNR): The true negative rate is the probability that an actual negative will test negative.

False Positive Rate (FPR): The false positive rate is the probability that an actual positive will test negative.

False Negative Rate (FNR): The false negative rate is the probability that an actual negative will test positive.

Performance of logistic regression classifiers are evaluated based on:

Accuracy: The quality or state of being correct or precise. The degree to which the result of a measurement, calculation, or specification conforms to the correct value or a standard.

Accuracy = (TP+TN)/(TP+TN+FP+FN) = 0.93

Recall: Recall is an absolute quantity, the smallest absolute amount of change that can be detected by a measurement.

Recall= TP/(TP+FN) = 0.91

Precision: Precision is defined as the quality of being exact and refers to how close two more measurements are to each other, regardless of whether those measurements are accurate or not.

Precision = TP/(TP+FP) = 1.0

F1 score: F score is defined as harmonic mean of the recall and precision

F1 score = 2*precision*recall/(precision+recall) = 0.95

Area under the curve : It tells us how good a model performs when used at different probability thresholds. AUC = 0.97

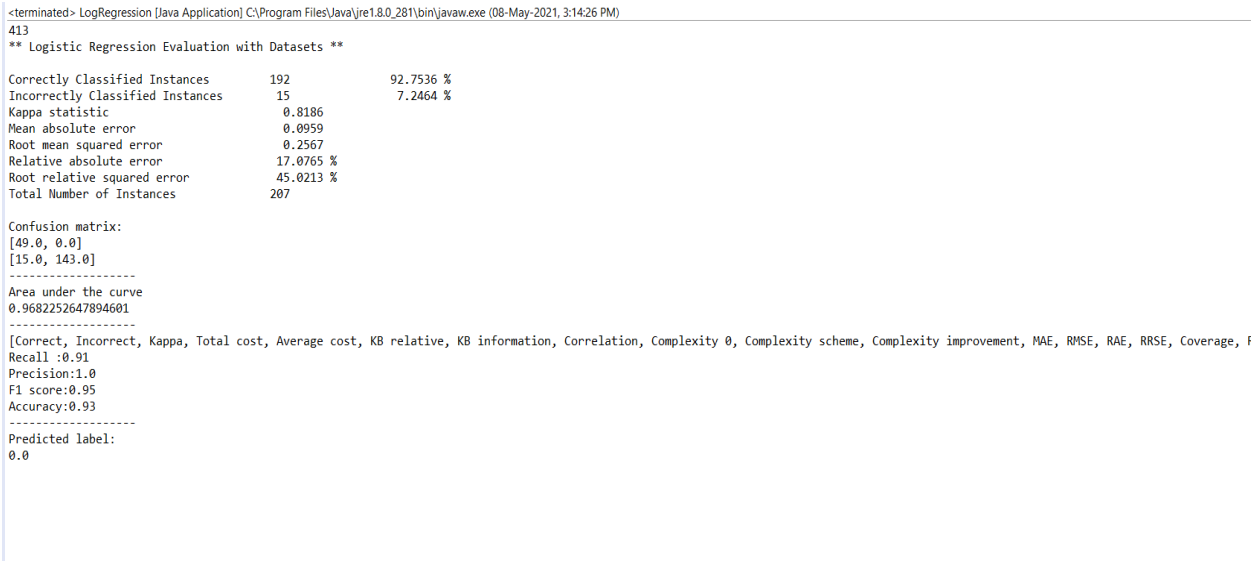


Figure : Results

7.1.ADVANTAGES

It has become an essential means for most organizations, companies, and government agencies to increase their productivity in global trade. One of the main reasons for the success of e-commerce is the easy online credit card transaction. Whenever we talk about monetary transactions, we also have to take financial fraud into consideration.

7.2. DISADVANTAGES

When providing input data of a highly unbalanced class distribution to the predictive model, the model tends to be biased towards the majority samples. As a result, it tends to misrepresent a fraudulent transaction as a genuine transaction

8. APPLICATIONS

Credit card fraud costs consumers and the financial company billions of dollars annually, and fraudsters continuously try to find new rules and tactics to commit illegal actions. Thus, fraud detection systems have become essential for banks and financial institution, to minimize losses.

9. CONCLUSION

In this project, we applied machine learning techniques to predict whether a credit card transaction is fraudulent or not. For this, we collected a publicly available dataset. In this project, machine learning technique logistic regression used to detect the fraud in credit card system. Here customers are grouped based on their transactions and extract behavioral patterns to develop a profile for every card holder.

10. FUTURE SCOPE

A cost-sensitive learning approach can be implemented by considering the misclassification costs. The cost for misclassifying a fraudulent class as a legitimate class which corresponds to the cost related to analysing the transaction and contacting the cardholder. So, this type of learning deals with classifying an example into a class that has the minimum expected cost.

11. BIBILOGRAPHY

John O. Awoyemi, Adebayo Olusola Adetunmbi, and Samuel Adebayo Oluwadare. Credit card fraud detection using machine learning techniques: A comparative analysis. 2017 International Conference on Computing Networking and Informatics (ICCNI)

APPENDIX

Data Anlayisis Code:

```
package org.kl;
import java.io.IOException;
import tech.tablesaw.api.Table;
import tech.tablesaw.plotly.Plot;
import tech.tablesaw.plotly.components.Figure;
import tech.tablesaw.plotly.components.Layout;
import tech.tablesaw.plotly.traces.BoxTrace;
import tech.tablesaw.plotly.traces.HistogramTrace;
public class DataAnalysis {
```

```
public static void main(String args[])
{
System.out.println("data Analysis");
        try
        {
        Tablebank_data=
Table.read().csv("D:\\eclipse-workspace\\org.kl\\src\\main\\java\\org\\fraud_dataset.csv");
        System.out.println(bank_data.shape());
        System.out.println(bank_data.first(7));
        System.out.println(bank_data.structure());
        System.out.println(bank_data.summary());
        Layout layout1 = Layout.builder().title("Distribution of Loan").build();
        HistogramTrace                                  trace1                                  =
HistogramTrace.builder(bank_data.nCol("Loan_Amount_Term")).build();
        Plot.show(new Figure(layout1, trace1));
Layout layout2 = Layout.builder().title("Distribution of Loan").build();
        BoxTrace trace2 =BoxTrace.builder(bank_data.categoricalColumn("Loan_Status"),
                        bank_data.nCol("Loan_Amount_Term")).build();
Plot.show(new Figure(layout2, trace2));

        }
catch(IOException e)
        {
        e.printStackTrace();
        }
}
}
```

Logistic Regression code:

```
package org.kl;
import java.util.Arrays;
import weka.classifiers.Classifier;
import weka.classifiers.evaluation.Evaluation;
import weka.core.Instance;
import weka.core.Instances;
import weka.core.converters.ConverterUtils.DataSource;
public class LogRegression {
	public static Instances getInstances (String filename)
	{DataSource source;
		Instances dataset = null;
		try {
			source = new DataSource(filename);
			dataset = source.getDataSet();
			dataset.setClassIndex(dataset.numAttributes()-1);
			} catch (Exception e) {
			e.printStackTrace();}
		return dataset;
	}
public static void main(String[] args) throws Exception{
Instances train_data =
getInstances("D:\\eclipse-workspace\\org.kl\\src\\main\\java\\org\\kl\\train.arff");
		Instances test_data =
getInstances("D:\\eclipse-workspace\\org.kl\\src\\main\\java\\org\\kl\\test.arff");
		System.out.println(train_data.size());
	Classifier classifier = new weka.classifiers.functions.Logistic();
```

```
        classifier.buildClassifier(train_data);
Evaluation eval = new Evaluation(train_data);
        eval.evaluateModel(classifier,test_data);
        System.out.println("** Logistic Regression Evaluation with Datasets **");
        System.out.println(eval.toSummaryString());
double confusion[][] = eval.confusionMatrix();
        System.out.println("Confusion matrix:");
        for (double[] row : confusion)
                System.out.println(     Arrays.toString(row));
        System.out.println("--------------------");
System.out.println("Area under the curve");
        System.out.println( eval.areaUnderROC(0));
        System.out.println("--------------------");
        System.out.println(Evaluation.getAllEvaluationMetricNames());
        System.out.print("Recall :");
        System.out.println(Math.round(eval.recall(1)*100.0)/100.0);
        System.out.print("Precision:");
        System.out.println(Math.round(eval.precision(1)*100.0)/100.0);
        System.out.print("F1 score:");
        System.out.println(Math.round(eval.fMeasure(1)*100.0)/100.0);
        System.out.print("Accuracy:");
        double acc = eval.correct()/(eval.correct()+ eval.incorrect());
        System.out.println(Math.round(acc*100.0)/100.0);
        System.out.println("--------------------");
        Instance predicationDataSet = test_data.get(2);
        double value = classifier.classifyInstance(predicationDataSet);
        /** Prediction Output */
        System.out.println("Predicted label:");
        System.out.print(value);    }}
```