**A Comparative Study of Machine Learning for Predicting Multiple Diseases**

This research is presented to the Department of Computer Science and Engineering at Jahangirnagar University as a partial fulfillment of the requirement for the degree of Bachelor of Science and Engineering.

***Submitted By***

**Mehadi Hasan**

Exam Roll: 180696

**Sohanur Rahman**

Exam Roll: 1806

***Supervised By***

**Dr. Israt Jahan**

Professor

Department of Computer Science and Engineering

Jahangirnagar University.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

JAHANGIRNAGAR UNIVERSITY.

SAVAR, DHAKA 1342.

MAY-2023

**Abstract**

Our everyday lives' most crucial aspect is how well we are mentally and physically. Heart disease, diabetes, and pneumonia are merely a few of the health issues that have recently become more common in our daily lives and in the field of healthcare. And this expenses humanity an enormous quantity of time. Based on the available data, machine learning can be used to detect such situations. The work, which is called "Multiple Disease Prediction" attempts to demonstrate how machine learning can be used to model the collection of data. The model is then applied to identify if a person is suffering from the disease or not. We employed a variety of methods, including decision trees, K-nearest neighbors (KNN), support vector machines (SVM), naïve bayes, and linear regression. Results of these algorithms are compared using their accuracy, precision, recall, and F1-score. The confusion matrix is used to plot the ROC curve. The technique with the best accuracy, precision, recall, and F1-score is taken into consideration for determining the optimum algorithm for illness detection after these algorithms are compared for accuracy, precision, recall, and F1-score.

# Declaration

The project work **" A Comparative Study of Machine Learning for Predicting Multiple Disease"** iscompleted at Department of Computer Science and Engineering, Jahangirnagar University is unique and conforms to the university's regulations.

We are aware of the University's plagiarism policy and certify that no component of this project has been plagiarized or previously submitted for the granting of any degree or diploma.

----------------------------------- ------------------------------------

**Mehadi Hasan** **Sohanur Rahman**

Exam Roll: 180696 Exam Roll: 1806

-----------------------------

**Dr. Israt Jahan**

Professor, Department of Computer Science and Engineering.

# Acknowledgement

All thanks and praise are due to Allah, who made it possible for us to complete this project successfully with the help of his heavenly blessing.

We are grateful to Dr. Israt Jahan, our esteemed supervisor, who is a professor in the department of computer science and engineering at Jahangirnagar University, for her patient counsel, astute direction, insightful instruction, and creative suggestion throughout the project's duration. This undertaking would not be possible without her invaluable aid. She has consistently been a source of inspiration and encouragement for us to put in a lot of effort.

Finally, we had want to give a shout-out to all of our friends who are truly dear to our hearts. We will never be willing to find the perfect words to express our gratitude to our loving parents, who have committed moral support and fortification in the completion of the project.

**Contents**

1. **Introduction 1**
2. **Problem Definition and Algorithm 3**

2.1 Task Definition . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

2.2 Algorithm Definition . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.2.1 Decision Tree Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.2.2 K-Nearest Neighbor Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

2.2.3 Support Vector Machine Algorithm . . . . . . . . . . . . . . . . . . . . . . . 5

2.2.4 Naïve Bayes Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

2.2.5 Logistic Regression Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

1. **Experimental Evaluation 8**
   1. Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

3.1.1 Experimental Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

3.1.2 Performance Evaluation Metrics . . . . . . . . . . . . . . . . . . . . . . . . . . . 15

3.2 Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19  
  
 3.3 Discussion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25

1. **Related Work 26**

**5 Future Work 28**

**6 Conclusion 29**

**List of Figures**

**1.1** Diagram of a Credit Card Fraud Detection Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

**3.1** Approached Credit Card Fraud Detection System . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

**3.1.1.1** Sample Dataset for Credit Card Fraud Detection Model . . . . . . . . . . . . . . . . . . . . . . . 10

**3.1.1.2** Graphing the Age of Credit Card by Gender . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

**3.1.1.3** Frequency Distribution of Credit Card’s Customer Gender . . . . . . . . . . . . . . . . . . . . . 12

**3.1.1.4** Gender Distribution of Customers as a Pie Chart . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

**3.1.1.5** Frequency Distribution of Credit Card Age . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

**3.1.1.6** Age Distribution of Credit Card as a Pie Chart . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14

**3.1.1.7** Visual Illustration of Two Outputs with the Numbers 0 and 1

Indicating Whether They Are Legitimate or Fraudulent . . . . . . . . . . . . . . . . . . . . . . . 15

**3.1.2** A 22 Confusion Matrix . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

**3.2.1** Classification Report for Decision Tree Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19

**3.2.2** ROC Curve for Decision Tree Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20

**3.2.3** Classification Report for K-Nearest Neighbor Algorithm . . . . . . . . . . . . . . . . . . . . . . . . 20

**3.2.4** ROC Curve for K-Nearest Neighbor Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

**3.2.5** Classification Report for Support Vector Machine Algorithm . . . . . . . . . . . . . . . . . . . 21

**3.2.6** ROC Curve for Support Vector Machine Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . 22

**3.2.7** Classification Report for Naive Bayes Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22

**3.2.8** ROC Curve for Naive Bayes Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

**3.2.9** Classification Report for Logistic Regression Algorithm . . . . . . . . . . . . . . . . . . . . . . . 23

**3.2.10** ROC Curve for Logistic Regression Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24

**List of Table**

**3.3** A comparative result of our proposed model using machine learning algorithm **25**

**Chapter 1**

**Introduction**

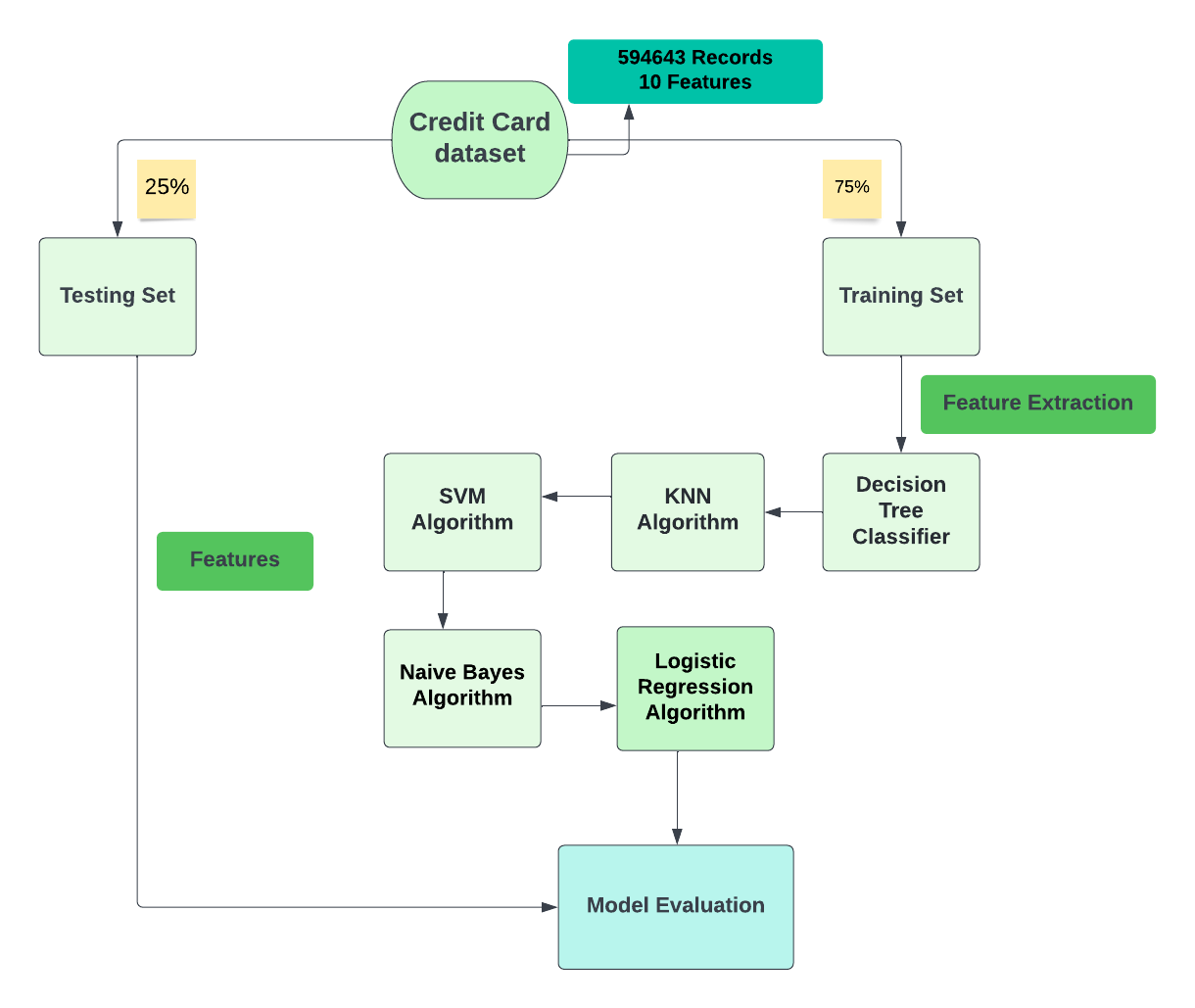
The goal of the rapidly expanding field of healthcare known as disease detection using machine learning is to create accurate and effective algorithms for diagnosing diseases based on patient data. It is now possible to leverage a lot of data to create models based on machine learning for disease identification, thanks to the expanding availability of electronic health records, medical imaging, and genetic data. On the basis of this data, machine learning algorithms may be trained to identify patterns and connections between the characteristics of the diseases and the attributes. These algorithms can then be used to forecast whether a disease will manifest in a new patient.

Machine learning-based identification of illnesses has considerable potential advantages. Machine learning models can assist healthcare professionals in making more precise and swifter diagnoses, which is essential for the effective treatment of diseases. Additionally, risk indicators for diseases can be found, and particular to the patient treatment plans may be developed using machine learning.

Data quality, data privacy, model interpretability, and model generalization are just a few of the difficulties that come with utilizing machine learning to detect diseases. The ability to overcome these obstacles and create efficient disease detection systems is now possible because to developments in machine learning algorithms and methods. In light of this, disease identification via machine learning is an interesting area of research that has the potential to revolutionize healthcare and enhance patient outcomes.

This study can teach us about a reliable feedback machine learning based medical disease detection system. With the aid of this feedback technique, the classifier's detection rate and performance are enhanced. Examine the performance of a number of classification algorithms, such as Decision Tree (DT) classifiers, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes (NB), and Linear Regression methods, on a completely unbalanced medical disease detection database.

In our research to identify numerous diseases, the proposed model is visually shown. The dataset for a person's disease is read by our algorithm using data downloaded from Kaggle.

****

**Figure 1.1:** Diagram of multiple disease detection model

This is the approach we suggest using to assess the model.

**Chapter 2**

**Problem Definition and Algorithm**

**2.1 Task Definition**

It may end up in many different kinds of health problems in the future if a particular individual has been experiencing a few symptoms but is unaware of the illness they are experiencing. This disease prediction will be highly helpful to a variety of people, including children, teens, adults, and elderly citizens, in order to prevent this and learn about the condition in the very beginning stages of the symptoms. In order to acquire the suitable treatment needed to address the condition, the person might adopt preventive measures or seek expert medical guidance.

The scope of a disease prediction system is particularly broad, illustrating how the world continues to evolve and how advancements in technology bring with them numerous disadvantages, which include a variety of food adulterations, inadequate nutrient supply to the body, unhealthy lifestyles involving improper consumption of food, as well as issues like obesity or unhealthy weight. Many different diseases go along with all of this.

Offering the finest quality services to all patients in the medical or healthcare fields is a significant task, and only those who can afford it can profit from it. There is a sizable amount of healthcare data that is not being mined in a more trustworthy and effective way to find hidden information for making good decisions. Methods based on data mining are used in the proposed framework to find chronic diseases early. Programming computers to come up with better results based on examples or past data has become known as machine learning. Machine learning pertains to the study of computer systems that learn from information and experience. The predictive machine learning algorithm includes two stages: training and testing.

Unfortunately, people often overlook their health because they are too preoccupied with their everyday tasks. Children and elderly people are both capable of ignoring or failing to recognize the crucial indicators that can later lead to more serious problems. It is advisable to get treatment before the illness worsens and progresses. Such individuals can benefit from preventative care and early detection of their health conditions with the use of a prediction system. This facilitates access to primary healthcare in isolated regions.

It is feasible to anticipate more than one disease at once when using the multiple disease prediction method. In order to anticipate the ailments, the user does not have to visit many sites. We are focusing on the disorders of the lung, diabetes, and the heart. because the three illnesses are related to one another. We're going to use Streamlit and machine learning methods to implement multiple illness analyses. When a user accesses this API, they must send the disease's parameters as well as the name of the disease. The appropriate model will be called by Streamlit, which then delivers the patient's state.

**2.2 Algorithm Definition**

An algorithmic and mathematical framework known as machine learning enables a computer system to learn from its prior experiences without being explicitly instructed what to do. Learning is concerned with the mimicking of human behavior by computers as well as the improvement of learning via the use of historical data. It also wants to create a data-driven system that is more predictive and adaptable. Here are some illustrations of machine learning techniques.

**2.2.1 Decision Tree Algorithm**

An example of supervised learning [1] is a decision tree, which can be applied to problems like classification and regression. It has the capacity to work with numerical and discrete data. It has a tree-like design with nodes and branches beginning at the tree's base and extending on subsequent branches till reaching the leaf node. The central node symbolizes the properties of the dataset, the branches the properties of the rules, and the leaf nodes the qualities of the solution to the issue. Decision tree algorithms are used in the real world for things like differentiating between cancerous and non-cancerous cells and giving customers car-buying recommendations.

There are several different data mining methodologies, which include ID3, CART, J48, NB Tree, REP Tree, and others. A broad algorithmic approach that has been often utilized to construct classification models is a tree structure. Most decision tree induction approaches employ a greedy top-down recursive partitioning methodology for tree development.

**2.2.2 K nearest neighbors Algorithm**

Both classification and regression concerns may be solved using the supervised learning method K-Nearest Neighbor. The new piece of data and the existing data points must be similar, according to this procedure. The new information points are given to the most comparable categories based on their relationship. It is sometimes referred to as the lazy learner algorithm since it preserves all accessible datasets and classifies each new instance using K-neighbors. The distance between data sets will be determined using any distance measure, and the new instance will be put into the category with the most similarities. The distance measure might be minkowski, euclidean, hamming, or manhattan, depending on the specifications.

**Implementation in Steps:**

**Step-1:** Calculate the Cosine Similarity of each document.

A o B = x1\*x2 + y1\*y2

dist(A,0) = sqrt((xa-x0)^2 + (ya-y0)^2) == |A|

As a result,

cos t = A o B/|A|x|B = sim(A,B).

**Step-2:** Loop

i) Select a Centroid

ii) Compare findings

iii) Continue if sim(A,B)Threshold is exceeded;

**Step-3:** Retrieve Cluster

**2.2.3 Support Vector Machine Algorithm**

A supervised learning method for regression and classification issues is the support vector machine (SVM). However, it is primarily employed to address categorization issues. SVM is used to establish a decision boundary or collection of points, that categorizes data.

Because support vectors are the data sets that help define the higher dimensional space, the technology is known as a support vector machine. SVM may be utilized for a variety of tasks, including medication development, picture classification, and face identification.

**Pseudo code:**

• Importing all of the necessary packages

For example, import pandas as pd.

• SVM defense

**STEP-1:** Start

**STEP-2:** Reading the dataset # reads the pd.read.csv dataset (file name)

**STEP-3:** Cleaning and preparation of data. Relevant Data is resized as normal and fraudster classes, with normal = 0 and fraudster = 1 in the normal and fraudster classes, respectively.

• Data is under saturated;

• Data is scaled (null values are deleted); and data is normalized

• Using the split () function on the training phase, the data is split into two sets: training dataset and testing dataset.

**Step-4:** Train the dataset with the Support Vector Machine algorithm.

• classifier.predict () # is an SVM classifier that forecasts whether or not a transaction is fraud.

**Step-5:** Computing the number of fraudulent and genuine transactions, and also recall, precision, and accuracy, and placing the findings in the proper places.

**STEP-6:** Stop

**2.2.4 Naive Bayes Algorithm**

The supervised learning methodology The approach is known as Naive Bayes since it is based on the Bayes theorem and operates on the naive assumption that the variables are independent of one another. Predictions are generated based on the item's value using this classifier.

The Bayes theorem is predicated on conditional probability, which is the chance of event (A) happening if event (B) has already happened.

The equation for Bayes theorem is given as: P(A|B)=

**2.2.5 Linear Regression Algorithm**

The link between a dependent variable and one or more independent variables may be modeled statistically using linear regression. It presumes that the variables have a linear connection, which means that changes to one or more of the independent variables will result in proportionate changes to the dependent variable.

In basic linear regression, there is just one independent variable, and a straight line is used to represent the connection. Finding the best-fitting line that illustrates the connection between the variables is the aim of linear regression. To do this, the difference between the observed values of the dependent variable and the expected values based on the independent variable is squared, and the difference is minimized.

When there are several independent variables, multiple linear regression is an extension of simple linear regression. In this instance, a hyperplane in multidimensional space is used to model the connection.

In several disciplines, including economics, finance, engineering, and social sciences, linear regression is often employed. In addition to determining the degree and direction of the link between variables, it is frequently used for forecasting and prediction. Numerous machine learning algorithms, including neural networks and support vector machines, are based on linear regression.

**Chapter 3**

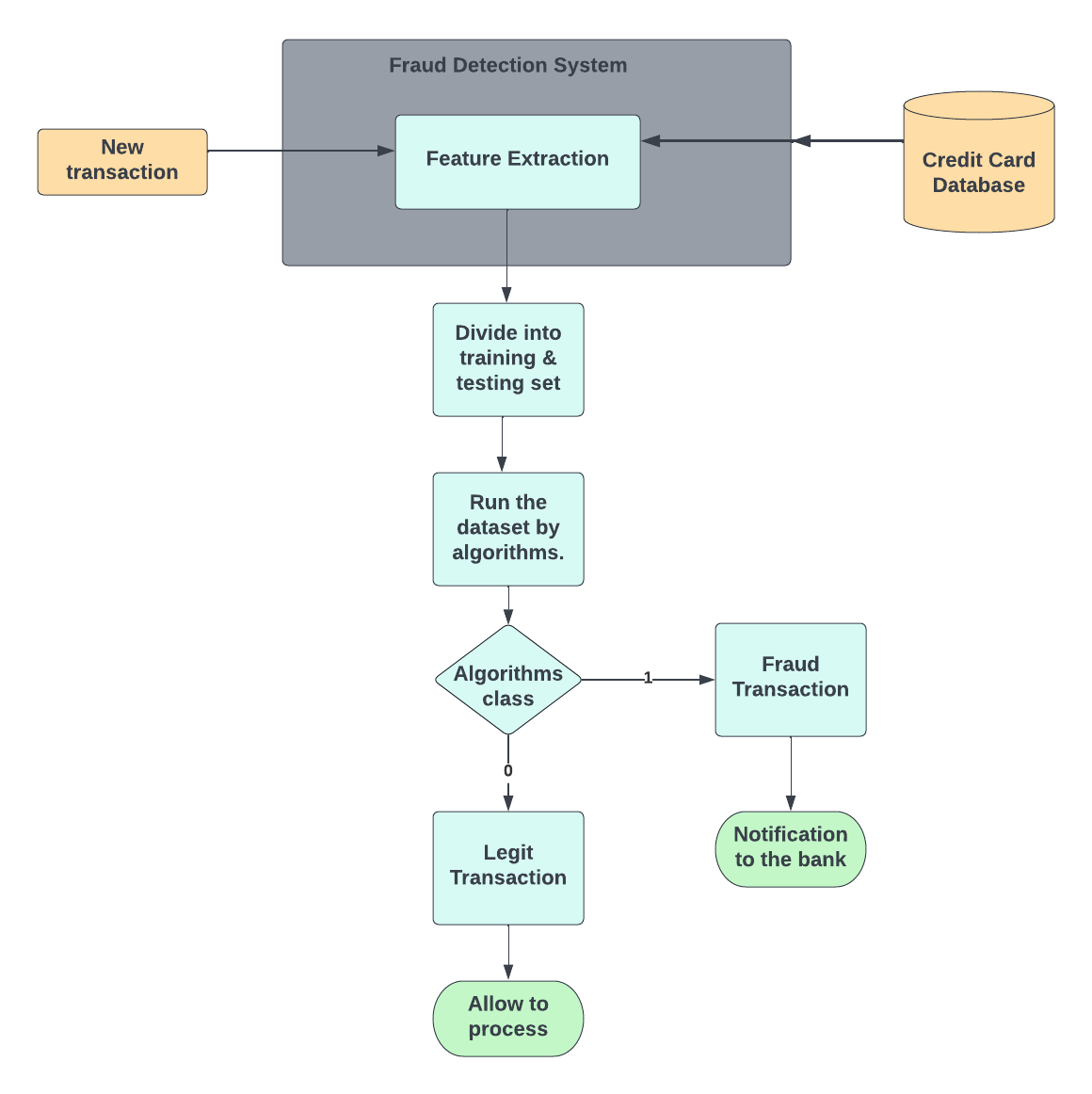
**Experimental Evaluation**

This section discusses the research process, findings, and conclusions from our investigation of numerous illness prediction systems. Because of its ability to handle both numerical and image data, we chose this assessment method when gathering our data. Three distinct forms of data will be included because of the three different types of research we have done. During the application step, the data set was divided into two groups. These two categories are, respectively, the training set and the testing set. We used a total of 25% of the data for testing and 75% of the data for training in order to produce a more effective model. The model was built using machine learning techniques such as Decision Tree (DT), K Nearest Neighbors (KNN), Support Vector Machines (SVM), Naive Bayes (NB) and Linear Regression. The best at identifying the abnormality was the Linear Regression (SVM).

**3.1 Methodology**

A goal-based evaluation methodology underpins our approach to detecting credit card fraud. Goal-based evaluations assess whether or not objectives have been met. Machine learning algorithms will extract data from two different sources in the fraud detection system. One is a new transaction initiated by the users, and the other is a credit card data set from the chosen bank. After that, the dataset is divided into two sections: a training and testing set. 75 percent of the data was utilized for training, while 25 percent was used for testing. It's critical to properly train the data set so that if a new fraudulent transaction occurs, the algorithm can recognize it immediately and send a warning to the credit card authorities. [2]

Graphic representation of our project design is drawn below:

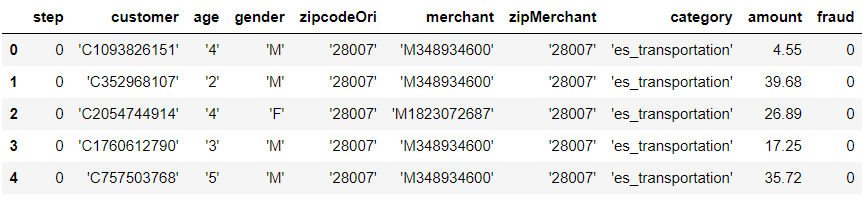
****

**Figure 3.1:** Approached credit card fraud detection system

**3.1.1 Experimental Dataset**

We used a dataset called fraudData.csv in our credit card fraud detection using machine learning study. We use the dataset as an ideal one for the goal of identifying fraud so that we may construct a perfect model that can reliably predict future fraudulent transactions.

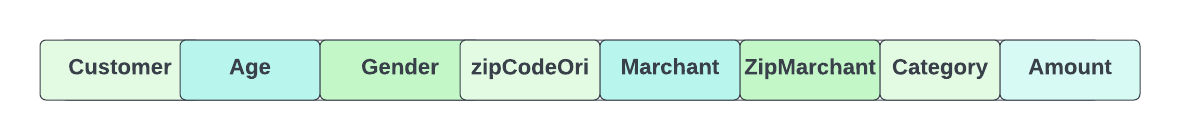
This project makes use of a Kaggle dataset that includes 594,643 pieces of data, 587,443 genuine transactions, and 7,200 fraudsters and ten features. BankSim transaction simulation software was used to generate the data. Here's where the sample set with several entries is mentioned.



**Figure 3.1.1.1:** Sample Data set for credit card fraud detection model

The data collection has a total of ten features. Eight of these characteristics are independent variables, while one is a dependent variable. The terms "independent set" and "dependent set" are mentioned here.

**Independent variables**

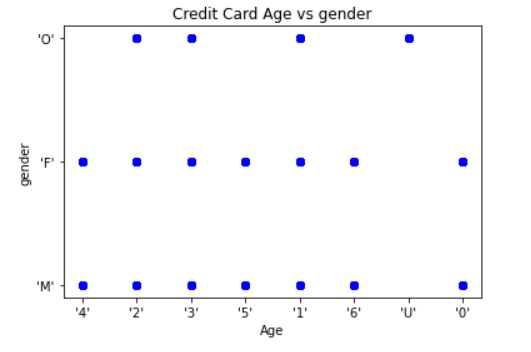
****

**Dependent variables**

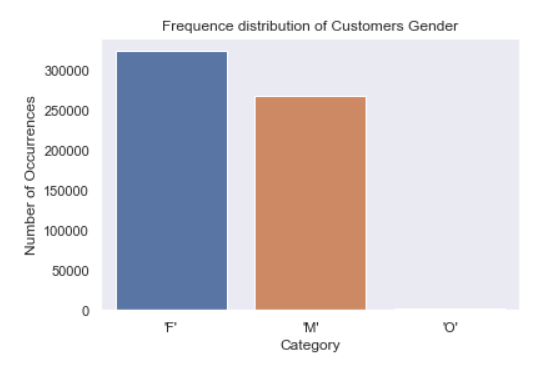
****

Seven out of the eight independent variables are categorical data. Using the python matplotlib library, we attempted to characterize some of them. Using matplotlib, we plotted the age and gender attributes of the data set to compare them. Except for the transaction amount attribute, the remaining seven categorical data must be converted to numerical data because machine learning algorithms cannot work with categorical data because they lack a numeric value.

We used some of the encoding techniques available in Python to transform categorical data to numeric data. Ordinal encoding and label order encoding are the encoding methods applied. Here is a graphic representation of certain categorical attributes.

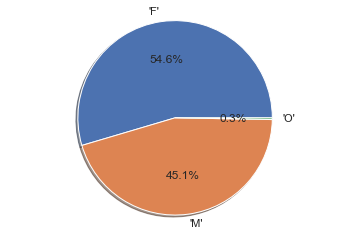
****

**Figure 3.1.1.2:** Graphing the age of credit cards by gender



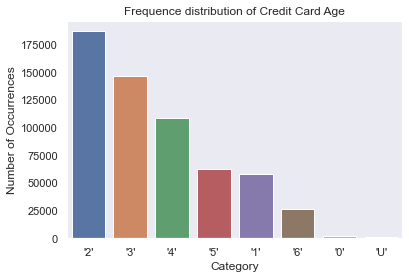
**Figure 3.1.1.3:** Frequency distribution of credit card’s customer gender

We also used a pie chart to show the percentage of different gender types in the dataset.



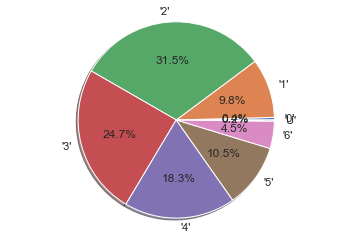
**Figure 3.1.1.4:** Gender distribution of customers as a pie chart

The age of the credit card is another crucial feature we used in our dataset. The credit card dataset contains a total of eight different types of outcomes. They are a numeric value ranging from 0 to 6, as well as a category value 'U,' which denotes an unused credit card.

****

**Figure 3.1.1.5:** Frequency distribution of credit card age

In the dataset, there is also a pie chart that shows the percentage of different credit card age categories.



**Figure 3.1.1.6:** Age distribution of credit card as a pie chart

Because our Kaggle database has 594,643 sets of data with 10 characteristics, including 587,443 legitimate transactions and 7,200 fraudsters, it's a big deal.

We attempted to categorize them into two groups: 'legit' and 'fraud'. Legitimate transactions have a class of 0 and fraudulent transactions have a class of 1. Then we categorized them as 'LEGIT' and 'FRAUD' and used a histogram to visually illustrate them. Below is a visual illustration of the two classes legit and fraud.

****

**Figure 3.1.1.7:** Visual illustration of two outputs with the numbers 0 and 1

Indicating whether they are legitimate or fraudulent.

**3.1.2 Performance Evaluation Metrics**

The performance of a machine learning model can be measured. They can present us with the progress of our machine learning algorithms, as well as a numerical value to go with it. A performance metric is required for all machine learning models, whether linear regression or classification. Whenever a new transaction occurs in our project, we will determine if it is fraudulent or not. The machine learning algorithm will provide us with both a predicted and real value. By comparing these two, we can obtain the confusion metrics, which is a performance evaluation indicator. Accuracy, precision, recall, f1-score, ROC curve, and other metrics can be used to describe the end outcome.

To characterize a categorization problem, a variety of phrases are utilized as performance evaluation metrics. We need a metric that compares discrete classes in some way because classification models produce discrete output. Categorization Metrics assess a model's performance and tell us whether the classification is excellent or poor, but they do it in diverse ways.

So, in order to assess Classification models, we'll go over the following measures in depth:

* Accuracy
* Precision
* Recall
* F1-score
* Confusion Matrix
* AUC-ROC Curve.

**Accuracy**

One of the evaluation metrics is accuracy. For each class, the percentage of correctly categorized examples among all the testing examples is:

%

**Precision**

Precision is a metric that measures how many correct positive predictions have been made. The ratio of accurately predicted true positives divided by the total number of true positives predicted is used to compute it[3].

**Recall**

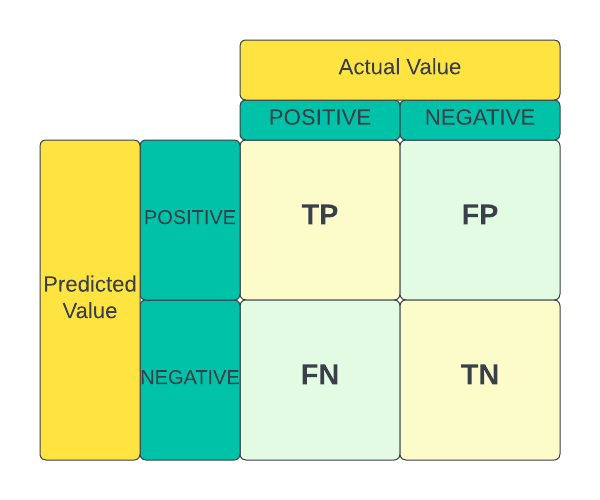
Recall is also known as true positive rate. The percentage of cases of the positive class that are well-classified among all positive examples. Recall is averaged when considering a multi-class classification problem[3].

**F-1 Score**

Because the F1 score is indeed the mean of precision and recall, precision and recall are given equal weight. As a result, for unbalanced classification problems, this metric, which takes values between 0 and 1, is frequently considered an acceptable by default metric[3]. The metric is constructed using the following equation as the overall average of precision and recall:

**Confusion Matrix**

A Confusion matrix is indeed an N-by-N matrix used to assess the effectiveness of a classification algorithm, where N is the number of class labels. The matrix compares actual class labels to the predictions of the machine learning model[3]. As shown below, we used to have a two-by-two matrix containing four values:



**Figure 3.1.2:** A 22 Confusion Matrix

**AUC-ROC**

Receiver Operating Curve or “ROC Curve” is a graph that depicts how a classification model works at various levels of categorization. Here on curve, 2 parameters are mapped: True Positive Rate (TPR) and False Positive Rate (FPR).

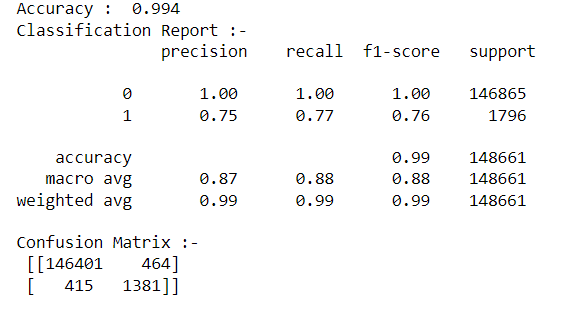
"AUC" is the abbreviation for "Area under the ROC Curve." AUC, in other words, assesses the full two-dimensional partially obscuring the entire ROC curve (imagine integral calculus) across (0, 0) to (1, 0). (1, 1).

**3.2 Results**

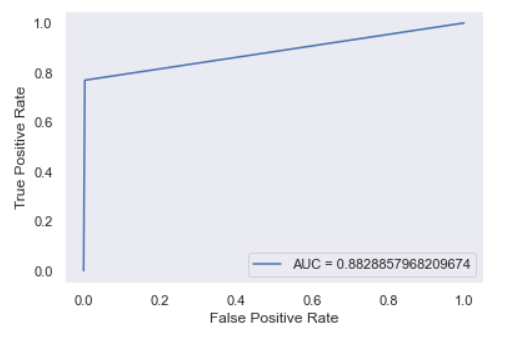
We utilized five distinct machine learning techniques, divided into two groups: classification and regression. They are included here, along with the results of each algorithm's classification report.

**i. Decision Tree Algorithm**

We have 99.4% accuracy in detecting credit card fraud using the decision tree method. It's a fantastic result.

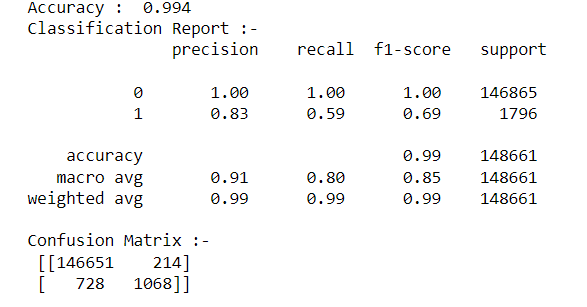


**Figure 3.2.1:** Classification report for Decision Tree Algorithm

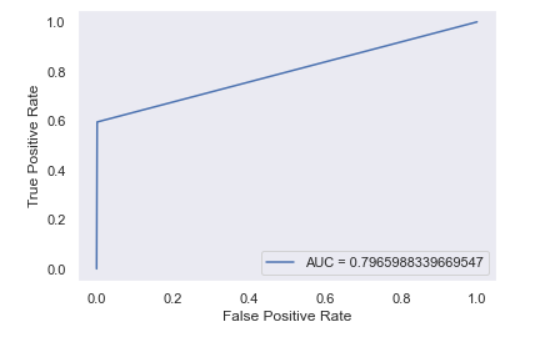


**Figure 3.2.2:** ROC Curve for Decision Tree Algorithm

**ii. K-Nearest Neighbor algorithm**

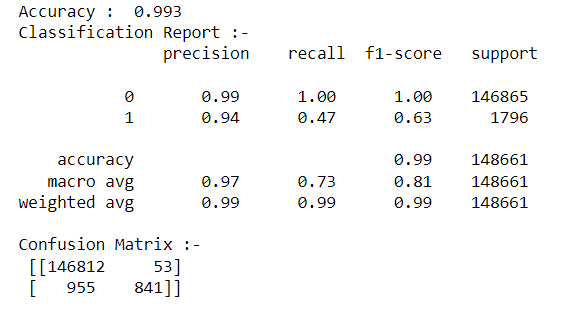
****

**Figure 3.2.3:** Classification report for K-Nearest Neighbor Algorithm

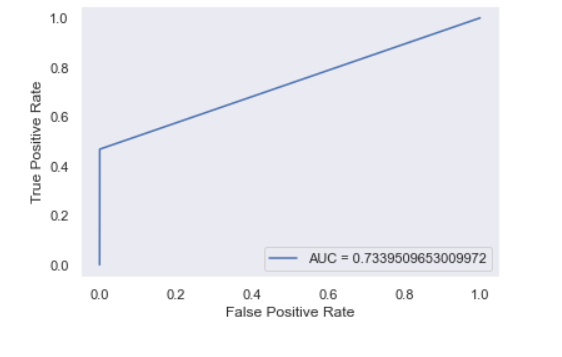


**Figure 3.2.4:** ROC Curve for K-Nearest Neighbor Algorithm

**iii. Support Vector Machine Algorithm**

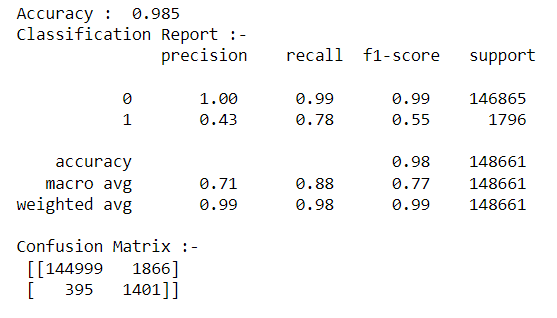
****

**Figure 3.2.5:** Classification report for Support Vector Machine Algorithm

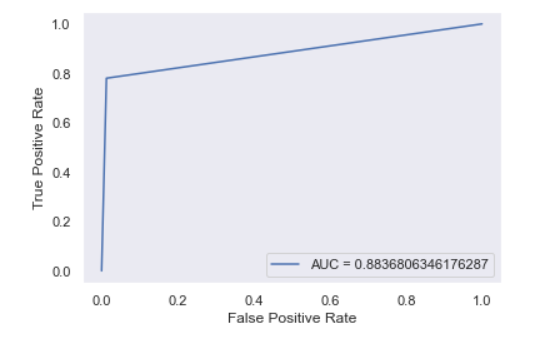


**Figure 3.2.6:** ROC Curve for Support Vector Machine Algorithm

**iv. Naive Bayes Algorithm**

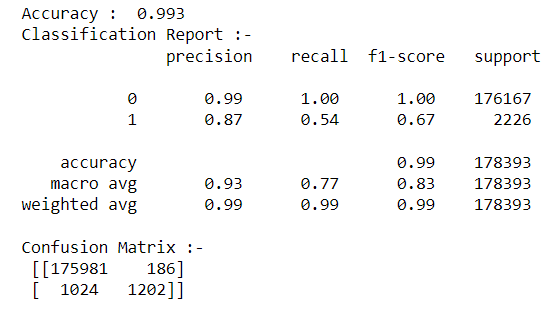


**Figure 3.2.7:** Classification report for Naive Bayes Algorithm

****

**Figure 3.2.8:** ROC Curve for Naive Bayes Algorithm

**v. Logistic Regression Algorithm**

****

**Figure 3.2.9:** Classification report for Logistic Regression Algorithm

****

**Figure 3.2.10:** ROC Curve for Logistic Regression Algorithm

That's all there is to the five algorithms we employed to detect credit card fraud in our project. Our data collection is significantly unbalanced, with over 98 percent of legitimate transactions and only 2 percent of fraudulent transactions.

As a result, when we utilize these algorithms, we receive a wide range of results. In comparison to the actual value, all of these algorithms yield the number of true positive and true negative rates. The accuracy score and precision of the algorithms are calculated using these rates.

**3.3 Discussion**

The hypothesis we offered in our study is really well-fitting. Because our data set is well-trained, and we applied a new machine learning technique, we were able to detect fraud transactions with a better rate of accuracy. The data set we used is severely unbalanced, with just 7200 fraudulent transactions out of 594,643 total. That is why we attempted to use an algorithm that would deliver a greater rate of accuracy.

We gain 99.4 percent overall accuracy using the decision tree technique, but only 75 percent precision when detecting fraudulent transactions. We get about 85 percent fraud transaction precision using the KNN algorithm, 94 percent fraud transaction precision using SVM, 43 percent fraud transaction precision using naive bayes, and 87 percent fraud transaction precision using logistic regression.

As a result, decision trees, KNN, Naive Bayes, and logistic regression algorithms did not work any better in the face of a highly skewed dataset. However, when dealing with a severely unbalanced credit card fraud data set, the SVM algorithm performs substantially better. As a result, we can put the SVM algorithm in charge of detecting credit card fraud in the real world.

A comparative result of our proposed model is mentioned here:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **Support** | **AUC** |
| Decision Tree | 99.4% | 75% | 77% | 76% | 1796 | 88.28% |
| KNN | 99.4% | 83% | 59% | 69% | 1796 | 79.65% |
| SVM | 99.3% | 94% | 47% | 63% | 1796 | 73.39% |
| Naïve Bayes | 98.5% | 43% | 78% | 55% | 1796 | 88.36% |
| Logistic Regression | 99.3% | 87% | 54% | 67% | 2226 | 76.94% |

**Table 3.3:** A comparative result of our proposed model using machine learning algorithm

A comparison of the results of our suggested model using five different classification and regression algorithms. Among these algorithms, SVM has a greater precision rate and a higher accuracy rate. As a result, our model can determine if a new transaction is fraudulent and send a message to the card issuer.

**Chapter 4**

**Related Work**

In this section, we go through some of the earlier work that has been done on Fraud Detection.

Machine learning techniques play a critical role in a variety of efficient data processing domains, one of which is the detection of card fraud. Several ways for detecting fraud were recommended in prior studies, including supervised methods, unsupervised methods, and a hybrid strategy; this necessitates knowing some technology involved in recognizing credit card fraud, as well as a deeper understanding of the sorts of card fraud. Many other solutions were proposed and tested. The majority of them will be discussed in the following paragraphs.

Yashvi Jain, Namrata Tiwari, Shripriya Dubey, and Sarika Jain researched SVM, ANN, Hidden Markov Model, Bayesian Networks, KNN, Fuzzy Logic system and Decision Trees in 2019. In their research, they discovered that the algorithms k-nearest neighbor, decision trees, and SVM provide a medium level of accuracy. Fuzzy Logic and Logistic Regression have the lowest accuracy of all the methods. Neural networks, naive bayes, fuzzy systems, and KNN all have a high detention rate. The LogisticRegression, SVM, decision trees have a high detection rate at the middle level. Both ANN and Nave Bayesian Networks outperform each other across all parameters. It is highly expensive to train these people. Each algorithm has a unique set of parameters. They give great results with one type of dataset, but they produce horrible results with another type of dataset.[1]

The work of Navanushu Khare and Saad Yunus Sait on decision trees, random forests, support vector machines (SVM), and logistic regression was published in 2018. They had to deal with a dataset that was considerably biased. Accuracy, sensitivity, specificity, and precision are used to evaluate performance. Overall accuracy of Logistic Regression is 97.7%, Decision Trees is 95.5 percent, Random Forest is 98.6%, and SVM classifier is 97.5 percent, according the statistics. Among the various ways, they determined that the Random Forest approach has the highest accuracy and is the best algorithm for detecting fraud. They also revealed that the support vector machine (SVM) approach has a data imbalance problem and performs poorly in detecting credit card fraud.[1]

Developments in E-Commerce and Communication technology, according to Altab Althar Taha and Sareef Jameel Malbery, have created credit card usage a far more frequent mode of payment, and transactional fraud is on the rise. Researchers used the high optical support vector machine, which combines Bayesian-based high energy optimization with parameter adjustment of the light gradient boosting machine (LightGBM). They employed a real-world public dataset that included both fraudulent and non-fraudulent transactions in their technique. Their suggested approach exceeded other strategies in terms of accuracy when compared to other techniques. The suggested system achieves a 98.40 percent accuracy, a 92.88 percent area under the receiver operating characteristics curve (AUC), a 97.34 percent precision, and a 56.95 percent F1-score.[4]

Debachudamani Prusti and Santhnu Kumar Rath designed an application employing machine learning methods such as decision trees, K-nearest neighbor (KNN), multilayer perceptron (MLP), and support vector machine to determine the accuracy in fraud detection (SVM). They proposed a model that included decision trees, support vector machines, and k-nearest neighbor algorithms. They used two web-based protocols for successful data exchange across several diverse platforms: simple object access protocol (SOAP) and representational state transfer (REST). They used the accuracy metric to compare the results of machine learning algorithms. The support vector machine (SVM) outscored other algorithms by 81.63 percent, while the hybrid system they presented had an accuracy of 82.58 percent.[4]

Because fraud detection must be very flexible in order to track the continuous growth of fraud over time and the appearance of unknown anomalies, H.Tran and K.P.Tran used anomaly detection techniques for credit card fraud detection. As data-driven techniques, they recommended a T2 control chart and a one-class support vector machine OCVM with optimum kernel parameter selection. The approach was tested utilizing a huge real-time data set of online e-commerce transactions from European credit card holders, which contained a total of 284807 non-fraud transactions. Simulators were also used to create fraudulent transactions, with 284000 transactions used for training and 200 fraud and non-fraud transactions used for testing. To examine the outcomes obtained by the techniques, they used accuracy, F1-score, Recall (DR), FPR, and Precision matrices. The experimental results show that OCVM surpasses the T2 flowchart with accuracy of 96.6 percent, FPR of 8.5 percent, and F-score of 100 percent. The two proposed techniques, on the other hand, have shown that they can accurately detect credit card fraud with a low false rate.[5]

**Chapter 5**

**Future Work**

Based on the results of the above research, we were unable to achieve our aim of 100 percent accuracy in fraud detection, indicating that there is still space for improvement. For detecting fraud, machine learning techniques are applied, although the results are not accurate.

There are certain approaches that can be used to correctly identify credit card fraud. As a result, implementing deep learning algorithms will improve the accuracy of detecting credit card fraud.

Another alternative for improvement is to expand the amount of the dataset; this would enhance the precision of the used algorithm. These will provide considerably more precise results. As an outcome, more data will almost certainly enhance the model's effectiveness in fraud prevention while also reducing the false positive rate. However this needs support (by giving more personal information about users) from the banks themselves.

**Chapter 6**

**Conclusion**

As credit cards carry not only money but also intellectual property, credit card fraud is a serious crime. To lessen the severity of this criminal crime. Machine learning can be a useful tool for detecting fraudulent transactions. We applied several classification and regression algorithms in our credit card fraud detection project. These algorithms give us a considerably greater level of accuracy in detecting fraudulent transactions. Because the SVM algorithm reacts significantly better in response to our severely imbalanced dataset, it fared the best among these algorithms.

The SVM (Support Vector Machine) technique gives a 99.3 percent accuracy rate and a 94 percent precision rate in the fraud class. In the actual world, that is a far higher accuracy rate for detecting fraud transactions. When a new transaction occurs, it will be processed using our proposed machine learning model, which will use the SVM method to determine if the transaction is fraudulent or not.

We may not be able to detect fraud transactions with 100% accuracy, but we can use these datasets in conjunction with real-world data. After that, we can reach 100% accuracy by integrating some machine learning techniques. Despite the fact that credit card data is confidential, different banks will not provide their genuine credit card user data set.

**Bibliography**

[1] R. Tyagi, R. Ranjan, and S. Priya, “Credit Card Fraud Detection Using Machine Learning Algorithms,” *Proc. 5th Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud), I-SMAC 2021*, no. Iciccs, pp. 334–341, 2021, doi: 10.1109/I-SMAC52330.2021.9640822.

[2] S P Maniraj, Aditya Saini, Shadab Ahmed, and Swarna Deep Sarkar, “Credit Card Fraud Detection using Machine Learning and Data Science,” *Int. J. Eng. Res.*, vol. 08, no. 09, pp. 110–115, 2019, doi: 10.17577/ijertv8is090031.

[3] N. K. Trivedi, S. Simaiya, U. K. Lilhore, and S. K. Sharma, “An efficient credit card fraud detection model based on machine learning methods,” *Int. J. Adv. Sci. Technol.*, vol. 29, no. 5, pp. 3414–3424, 2020.

[4] A. RB and S. K. KR, “Credit card fraud detection using artificial neural network,” *Glob. Transitions Proc.*, vol. 2, no. 1, pp. 35–41, 2021, doi: 10.1016/j.gltp.2021.01.006.

[5] H. Najadat, O. Altiti, A. A. Aqouleh, and M. Younes, “Credit Card Fraud Detection Based on Machine and Deep Learning,” *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, no. Section IX, pp. 204–208, 2020, doi: 10.1109/ICICS49469.2020.239524.