Report on Machine Learning

Topic - Credit Card Scam Classification

Problem Statement : The objective of this study is to develop a robust and effective classification model for detecting potential credit card fraud. This task entails analyzing a dataset containing various features associated with credit card transactions and accurately classifying whether a given transaction is a scam or not.

Algorithm Used : There are various classification algorithms like Random Forest, Logistic Regression & Support Vector Machine but we have used Naïve Bayes Algorithm to perform the learning task.

This algorithm is known for its simplicity, efficiency, and effectiveness in handling classification problems, making it a suitable choice for our particular scenario.

Naïve Bayes Algorithm leverages Bayes theorem to estimate the probability of a given data frame belonging to a specific class, in this case, whether a credit card transaction is a scam or not.

Libraries Used : We have used some libraries to perform the tasks given below :-

1. Pandas : Used to import the data file and read it.
2. Numpy: Used to print confusion Matrix.
3. train_test_split: To split the data into training and testing units.

4. GaussianNB : We used Gaussian distribution so that the probability distribution is symmetric about the mean, showing that data near the mean are more frequent in occurrence than data far from the mean.

5. accuracy_score, confusion_matrix: To print accuracy and confusion matrix.

Dataset : The data set used is downloaded from kaggle.com

Link: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

Due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, ... V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'.

Working of Algorithm : We use Bayes Theorem to calculate the probability of outcome to occur. The formula to calculate the probability is :-

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Here we calculate the probabilities of an event occurring and not occurring for each specific feature within the dataset. This involves estimating the likelihood of a given feature value given the event's occurrence (e.g., fraud) and the likelihood of the same feature value given the event's non-occurrence (e.g., non-fraud). These conditional probabilities are calculated independently for each feature in the dataset.

Furthermore, we determine the overall probabilities of the event occurring and not occurring based on the frequency of these events within the dataset. Specifically, we compute the prior probabilities of the event occurring and not occurring by dividing the number of occurrences and non-occurrences by the total number of instances in the dataset.

To classify a new data point with specific feature values, we utilize these calculated probabilities within the Naïve Bayes formula. This formula combines the prior probabilities and the conditional probabilities for each feature to estimate the probability of the event (e.g., credit card fraud) occurring given the observed feature values. The calculated probability allows us to make an informed classification decision for the new data point, determining whether the event is likely to occur or not based on the available information.

Train & Test data Size : We have divided our data into 8:2 where 80% data is used in training and 20% is used in testing.

Result of Model:

At last we calculated the Accuracy and Confusion Matrix.

Confusion Matrix : [[56552 319]
[29 61]]

Accuracy : 99.38905566966872

Conclusion : In conclusion, the development and implementation of our machine learning model, which utilizes the Naïve Bayes algorithm for credit card fraud detection, have proven to be a valuable asset in enhancing the security and reliability of credit card transactions. The model's ability to accurately classify transactions as either fraudulent or non-fraudulent is crucial in safeguarding financial institutions, businesses, and consumers against potential fraudulent activities.

Code of Machine learning Model

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix

# Load the credit card fraud dataset (replace with your dataset)
data = pd.read_csv('creditcardtest.csv')
print(type(data))

# Showing Data
pd.options.display.max_rows = 3
data = pd.read_csv('creditcardtest.csv')
print(data)

X = data.drop(['Class'], axis=1)
y = data['Class']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2)

# Create a Gaussian Naive Bayes classifier
clf = GaussianNB()

# Train the classifier on the training data
clf.fit(X_train, y_train)
```

```
# Make predictions on the test data
y_pred = clf.predict(X_test)

# Create and display the confusion matrix & accuracy
conf_matrix = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(conf_matrix)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy*100}")
```