**PROJECT 1: CREDIT CARD FRAUD DETECTION**

Phase 3: Development part 2

Data set:



500row X 31 columns

**Necessary steps to follow**:

**1.Import Libraries:**

start by importing the necessary libraries:

**Program:**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seborn as sns

import warnings

warnings.filterwarnings(‘ignore’)

Setting up options.

pd.set\_option(‘display.max\_rows’,None)

pd.set\_option(‘display.max\_columns’,None)

For Scaling

from sklearn.preprocessing import StandardScaler

Train-Test Splitting.

from sklearn.model\_selection import train\_test\_split

For Pipelines

from sklearn.pipeline import Pipeline

Model Building

from sklearn.linear\_model import LogisticRegression

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier, GradientBoostingClassifier

from sklearn.svm import SVC,LinearSVC

Metric Evaluation.

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

import os

import

for dirname,\_,filenames in os.walk(‘\kaggle\input’):

for filename in filenames:

print(os.path.join(dirname,filename))

**2.Load the Dataset:**

Load your dataset into a pandas dataframe.

**Program:**

Df=pd.read\_csv(‘’)

Pd.read()

**3.Exploratory Data Analysis (EDA):**

df = data\_import.drop('id' , axis = 1

features = df.drop('Class' , axis = 1)

label = df['Class']

x\_train , x\_test , y\_train , y\_test = train\_test\_split(features , label , test\_size = 0.2 , stratify = label , random\_state = 42)

print(f"x train shape : **{**x\_train.shape**}\n**y train shape : **{**y\_train.shape**}\n**x test shape : **{**x\_test.shape**}\n**y test shape : **{**y\_test.shape**}**")

x train shape : (454904, 29)

y train shape : (454904,)

x test shape : (113726, 29)

y test shape : (113726,)

linkcode

lr\_pipe = Pipeline(steps =[

('scaler' , StandardScaler())

('Logistic Regression' , LogisticRegression())

])

tree\_pipe = Pipeline(steps = [

('scaler' , StandardScaler()),

('Decision Tree' , DecisionTreeClassifier())

])

forest\_pipe = Pipeline(steps = [

('scaler' , StandardScaler()),

('Random Forest' , RandomForestClassifier())

])

ada\_pipe = Pipeline(steps = [

('scaler' , StandardScaler()),

('AdaBoost' , AdaBoostClassifier())

])

gradient\_pipe = Pipeline(steps = [

('scaler' , StandardScaler()),

('Gradient Boosting' , GradientBoostingClassifier())

])

svc\_pipe = Pipeline(steps = [

('scaler' , StandardScaler()),

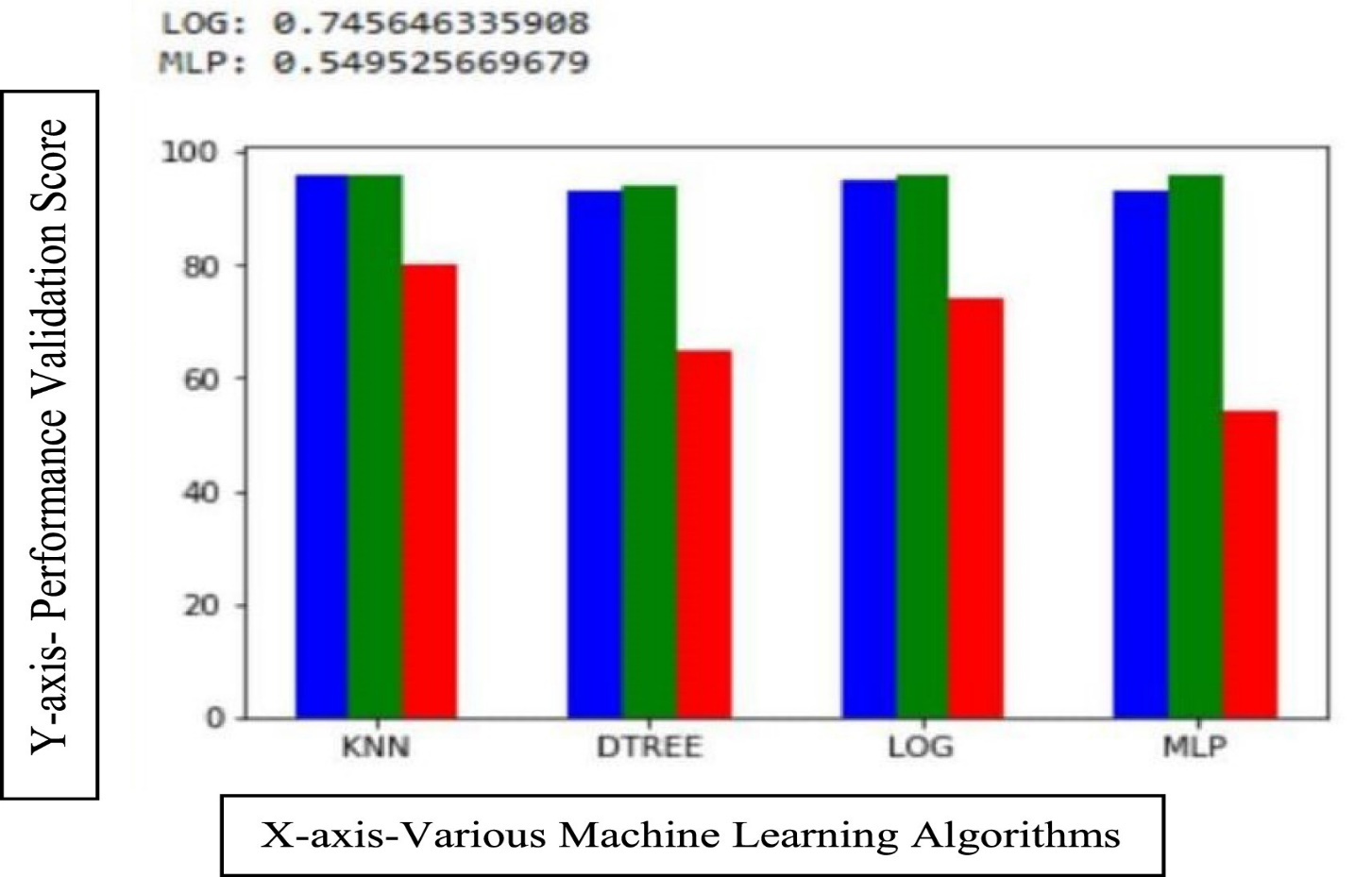
('Support Vector Machines', SVC())

])

linear\_svc\_pipe = Pipeline(steps = [

('Linear Support Vector Machines ' , LinearSVC())

])



**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 chosen for the project, and it was met by building the four models and finding the accuracies of them all, the best model in terms of accuracies is Support Vector Machine which scored 99.94% with only 51 misclassified instances. I believe that using the model will help in decreasing the amount of credit card fraud and increase the customers satisfaction as it will provide them with better experience in addition to feeling secure.