import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

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

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import accuracy\_score

# loading  the dataset to a panads DataFrame

credit\_card\_data=pd.read\_csv('/content/creditcard[1].csv')

credit\_card\_data.head()

credit\_card\_data.info()

#distribuation of legit transaction & fraudulent transaction

credit\_card\_data['Class'].value\_counts()

# Seperating the data for analysis

legit=credit\_card\_data[credit\_card\_data.Class==0]

fraud=credit\_card\_data[credit\_card\_data.Class==1]

#checking the number of missing values in each column

credit\_card\_data.isnull().sum()

print(legit.shape)

print(fraud.shape)

# Statastical measures of data

legit.Amount.describe()

fraud.Amount.describe()

# compare the values of both trasacations

credit\_card\_data.groupby('Class').mean()

legit\_sample=legit.sample(n=103)

new\_dataset=pd.concat([legit\_sample,fraud],axis=0)

new\_dataset.head()

new\_dataset.tail()

new\_dataset['Class'].value\_counts

new\_dataset.groupby('Class').mean()

X=new\_dataset.drop(columns='Class',axis=1)

Y=new\_dataset['Class']

print(X)

print(Y)

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,test\_size=0.2,stratify=Y,random\_state=2)

print(X.shape,X\_train.shape,X\_test.shape)

print(X.shape,X\_train.shape,X\_test.shape)

model=LogisticRegression()

# tranining the Logistic Regression model with training data

model.fit(X\_train,Y\_train)

#accuracy on training data

X\_train\_prediction=model.predict(X\_train)

training\_data\_accuracy=accuracy\_score(X\_train\_prediction,Y\_train)

print('Accuracy of Training data:',training\_data\_accuracy)

#accuracy on test data

X\_test\_prediction=model.predict(X\_test)

test\_data\_accuracy=accuracy\_score(X\_test\_prediction,Y\_test)

print('Accuracy score of  test data:',test\_data\_accuracy)

plt.plot(X\_test,Y\_test)

plt.xlabel('Data Points')

plt.ylabel('Accuracy')

plt.show()