**PRACTICAL:5**

**AIM:**

Implement logistic regression and calculate the different evaluation measure (F-measures, Confusion Matrix etc.) for the same. Also implement gradient descent and observe the cost with logistic regression using gradient descent. (Hint: Confusion Matrix and F-measures involve use of True Negatives, True Positives, False Negatives and False Positives). Also implement Cross- Validation.

**CODE:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sb

predict=pd.read\_csv("Admission\_Predict.csv")

predict

predict.shape

predict.describe()

predict.corr()

predict['TOEFL Score']

sb.jointplot(x='GRE Score',y='Chance of Admit ',data=predict,kind='kde',fill=True,cmap='mako')

plt.title('GRE VS CHANCE OF ADMIT')

plt.show()

def zerone(val):

  if val>0.7:

    return 1

  else:

    return 0

predict['Chance of Admit ']=predict['Chance of Admit '].apply(zerone)

predict['Chance of Admit ']

sb.countplot(data=predict,x='Chance of Admit ')

features=['GRE Score','TOEFL Score','University Rating','SOP','LOR ','CGPA']

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import confusion\_matrix

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

x=predict[features]

y=predict['Chance of Admit ']

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,random\_state=9,test\_size=0.2)

LR=LogisticRegression(C=0.01, solver='liblinear').fit(x\_train,y\_train)

LR

yhat=LR.predict(x\_test)

yhat

yhat\_prob=LR.predict\_proba(x\_test)

yhat\_prob

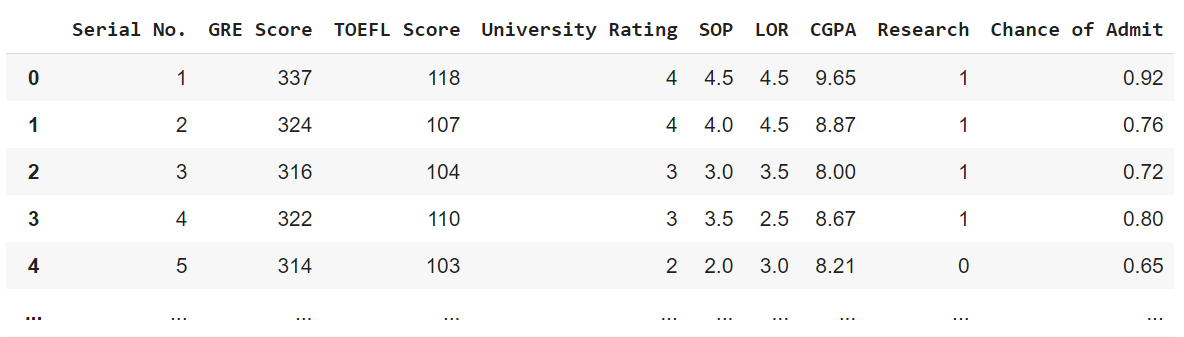
from sklearn.metrics import classification\_report,confusion\_matrix

import itertools

print(classification\_report(y\_test,yhat))

confusion\_matrix(y\_test,yhat)

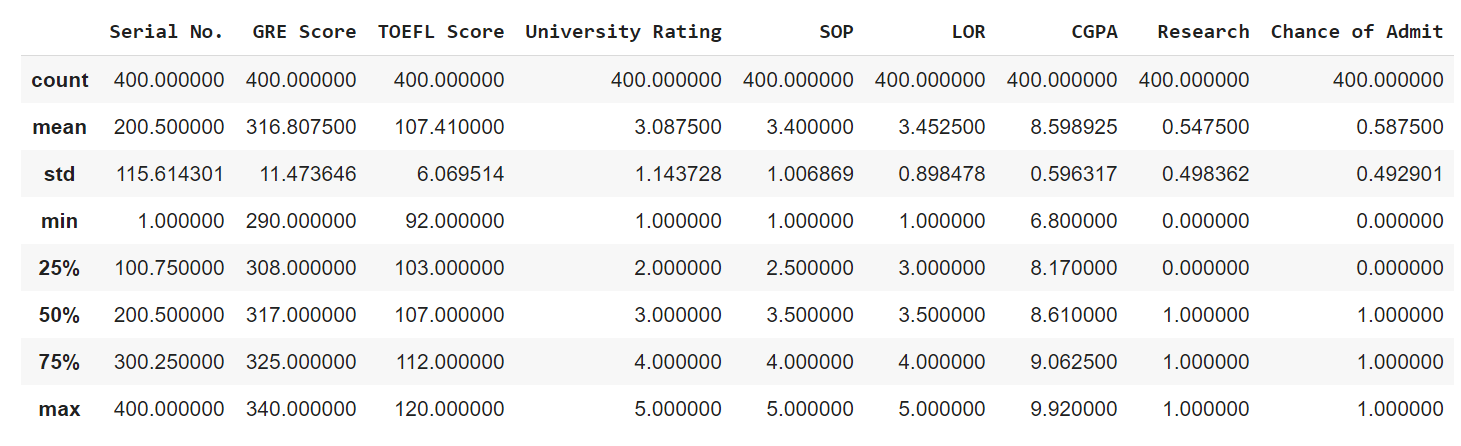
**OUTPUT:**



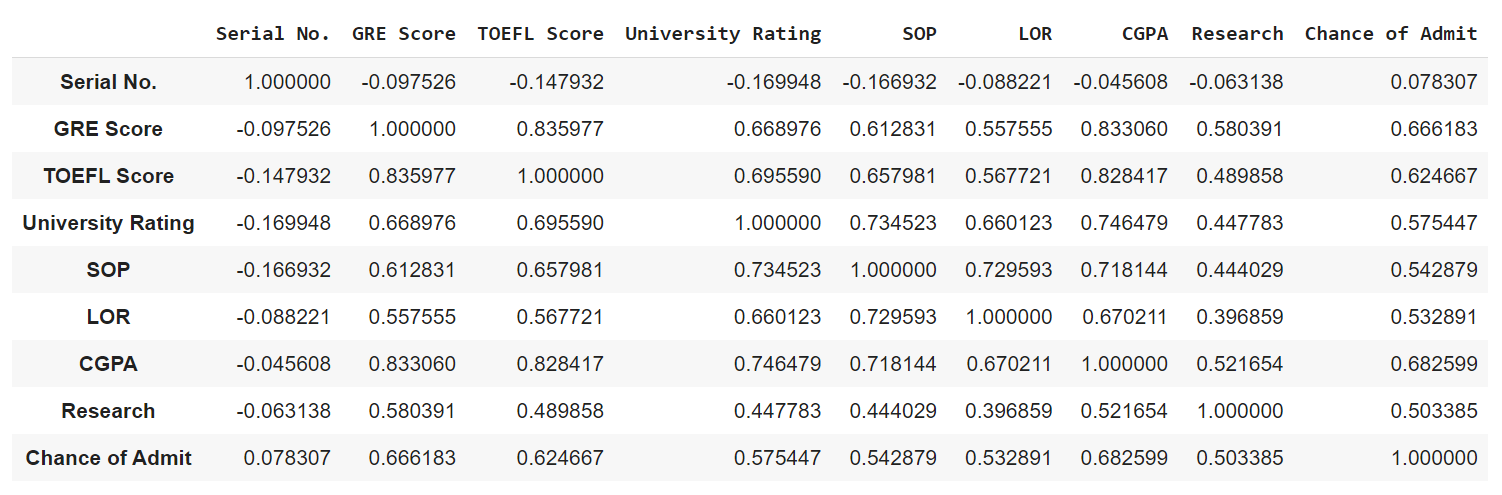
*Glimpses of dataset*



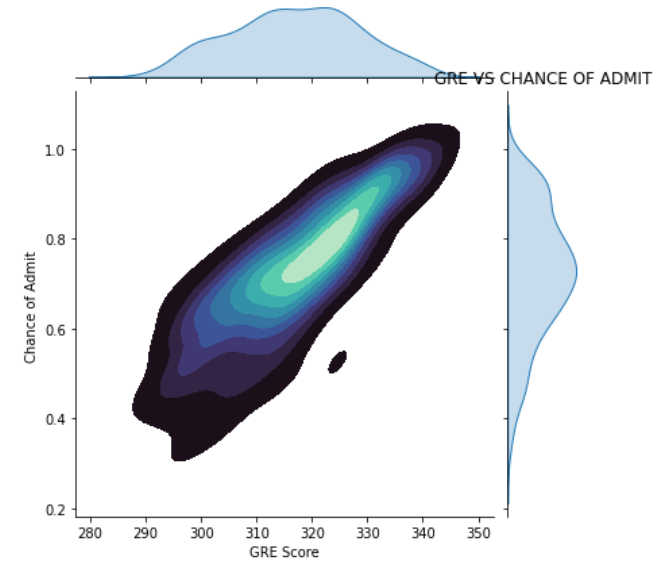
*Shape of the df*



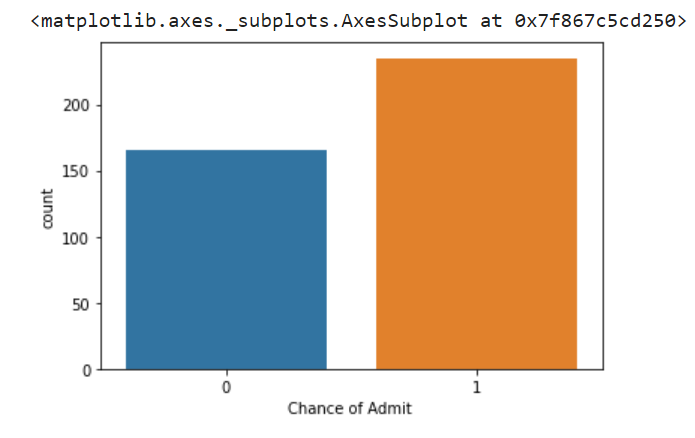
*Output of describe()*



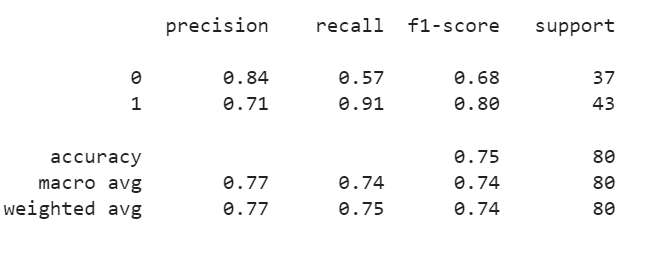
*Output of corr()*



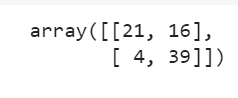
*Jointplot*



*Countplot*



*Classification report*



Confusion matrix