* **Logistic Regression Implementation using Python Programming :**

**{**insurance data**}**

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

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

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

from sklearn.linear\_model import LogisticRegression

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

df = pd.read\_csv("/content/DIABETES.csv")

print(df)

print("\n\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

df.info()

x = df.iloc[:,:-1].values   # selecting a particular cell of the dataset,

y = df.iloc[:,1].values

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size = 0.25)

print("X shape:", x.shape)

print("y shape:", y.shape)

print("X\_train shape:", x\_train.shape)

print("X\_test shape:", x\_test.shape)

print("y\_train shape:", y\_train.shape)

print("y\_test shape:", y\_test.shape)

log\_reg = LogisticRegression()

log\_reg.fit(x\_train, y\_train)

y\_pred = log\_reg.predict(x\_test)

print("\n\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n")

print("train data's prediction --> ", log\_reg.score(x\_train, y\_train))

print("test data's prediction --> ", log\_reg.score(x\_test, y\_test))

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

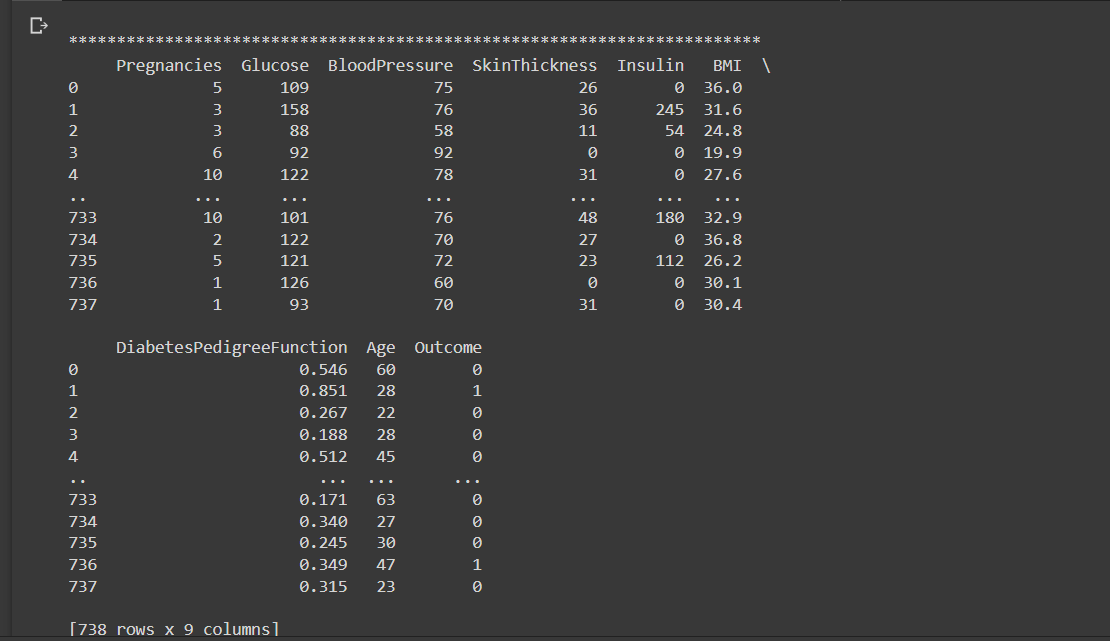
plt.plot(x\_train, y\_train, color='g')

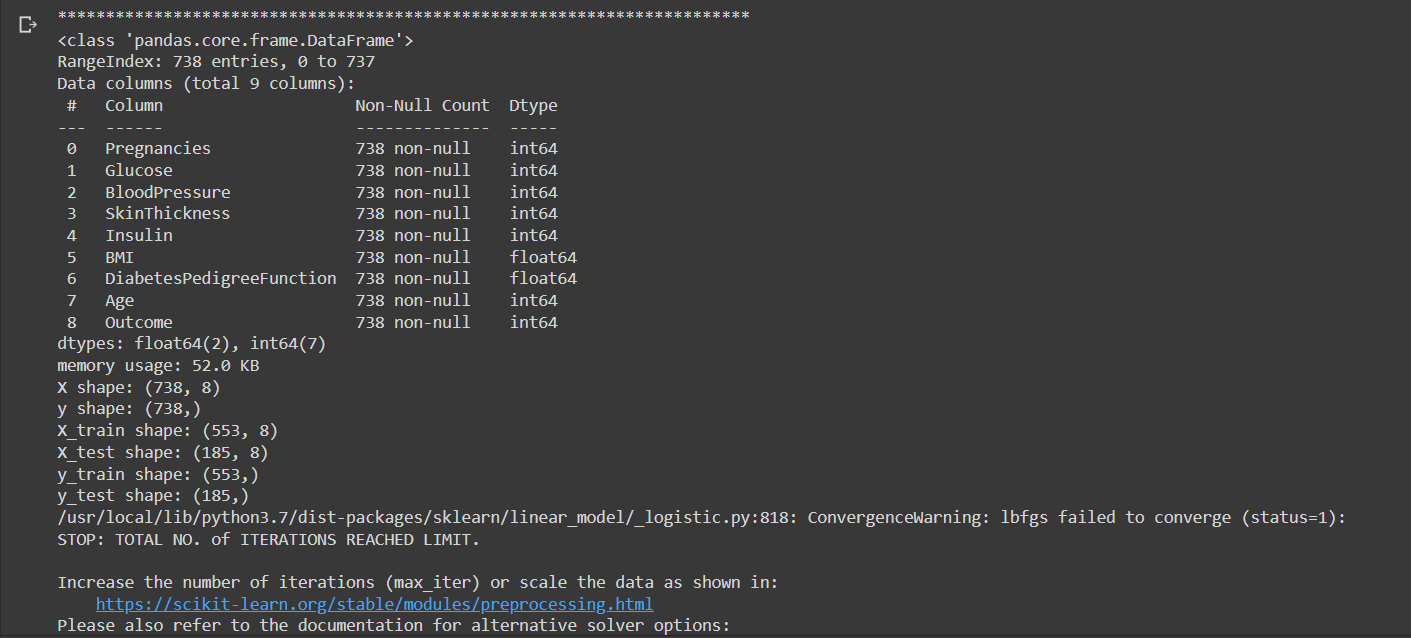
plt.show()

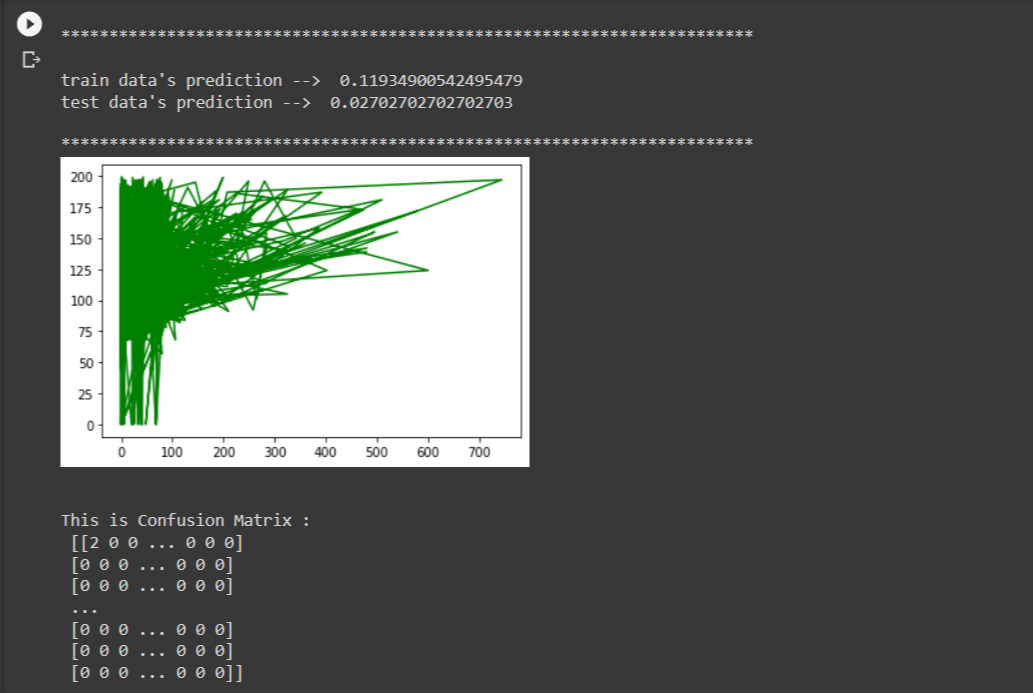
cm = confusion\_matrix(y\_test, y\_pred)

print("\n\nThis is Confusion Matrix :\n", cm)

**OUTPUT :**

****

****

****