import pandas as pd

df=pd.read\_csv("https://raw.githubusercontent.com/sakshi2k/Social\_Network\_Ads/master/Social\_Network\_Ads.csv")

df

#input data

x=df[['Age','EstimatedSalary']]

#output data

y=df["Purchased"]

from sklearn.preprocessing import MinMaxScaler

sc=MinMaxScaler()

x\_scaled=sc.fit\_transform(x)

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

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(x\_scaled,y,random\_state=42,test\_size=0.25)

Y\_train

from sklearn.linear\_model import LogisticRegression

import seaborn as sns

import matplotlib.pyplot as plt

sns.countplot(x=y)

y.value\_counts()

classifier=LogisticRegression()

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

Y\_pred=classifier.predict(X\_test)

print("Shape of X\_train : ",X\_train.shape)

print("Shape of Y\_train : ",Y\_train.shape)

Y\_pred

Y\_test

#Scatter Plot

plt.xlabel('Age')

plt.ylabel('Salary')

plt.scatter(x['Age'],x['EstimatedSalary'],c=y)

pd.DataFrame(x\_scaled).describe()

plt.xlabel('Age')

plt.ylabel('Salary')

plt.scatter(x\_scaled[:,0],x\_scaled[:,1],c=y)

from sklearn.metrics import confusion\_matrix

confusion\_matrix(Y\_test,Y\_pred)

Y\_test.value\_counts()

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

fig,ax=plt.subplots(figsize=(6,6))

# Create a heatmap of the confusion matrix with a custom color map

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', ax=ax)

# Set the title of the confusion matrix

ax.set\_title('Confusion Matrix')

# Set the x and y axis labels

ax.set\_xlabel('Predicted')

ax.set\_ylabel('True')

# Display the confusion matrix

plt.show()

from sklearn.metrics import accuracy\_score

from sklearn.metrics import classification\_report

print("Accuracy Score: \n", accuracy\_score(Y\_test,Y\_pred))

print("\n\n Classification Report: \n", classification\_report(Y\_test,Y\_pred))