1.DOWNLOAD THE DATA SET:

import numpy as np import pandas as pd import seaborn as sns import matplotlib.pyplot as plt import sklearn

2.LOAD THE DATASET:

data = pd.read\_csv(r"file:///C:\Users\Christo\Downloads\Churn\_Modelling.csv")

1. VISUALIZATIONS:

sns.histplot(data["CreditScore"]) sns.distplot(data["Age"]) sns.boxplot(data['Age']) (ii) BI-VARIATE ANALYSIS:

sns.lineplot(x=data.CreditScore, y=data.EstimatedSalary) sns.barplot(x=data.CreditScore, y=data.Age) plt.figure(figsize=(15,15)) sns.barplot(x=data.Age , y=data.CreditScore) sns.scatterplot((data['Age'], data['Tenure'])) (iii) MULTI-VARIATE ANALYSIS:

sns.pairplot(data) data.corr() sns.heatmap(data.corr(), annot = True)

1. DESCRIPTIVE STATISTICS:

data.mean() data.median() data.mode() data.var() data.std() data.describe()

5.HANDLE THE MISSING VALUES:

data.isnull().any() data.isnull().sum()

6. FINDING OUTLIERS AND REPLACING THEM:

sns.boxplot(x=data['EstimatedSalary']) Q1= data['EstimatedSalary'].quantile(0.25) Q2=data['EstimatedSalary'].quantile(0.75) print(Q1,Q2)

IQR=Q2-Q1 IQRv

upper\_limit =Q2 + 1.5\*IQR lower\_limit =Q1 - 1.5\*IQR upper\_limit lower\_limit data=data[data['EstimatedSalary']<upper\_limit] data=data[data['EstimatedSalary']>lower\_limit] sns.boxplot(x=data['EstimatedSalary']) p99= data['EstimatedSalary'].quantile(0.99) p99

data = data[data['EstimatedSalary']<=p99] sns.boxplot(x=data['EstimatedSalary']) data['EstimatedSalary'] = np.where(data['EstimatedSalary']>upper\_limit,652,data['EstimatedSalary']) data.shape

---------------------------------------------------------------------------

NameError Traceback (most recent call last)

<ipython-input-7-047ed65ff157> in <module>

----> 1 data.shape

NameError: name 'data' is not defined

7. CHECK FOR CATERGORICAL COLUMNS AND PERFORM ENCODING:

from sklearn.preprocessing import LabelEncoder, OneHotEncoder le = LabelEncoder() oneh = OneHotEncoder() data['Gender'] = le.fit\_transform(data['Gender'])

---------------------------------------------------------------------------

NameError Traceback (most recent call last) <ipython-input-6-cdac9c1b5bfa> in <module>

1. le = LabelEncoder()
2. oneh = OneHotEncoder()

----> 4 data['Gender'] = le.fit\_transform(data['Gender'])

NameError: name 'data' is not defined data.head()

SPLIT THE DATA INTO DEPENDENT AND INDEPENDENT VARIABLE

X=data.drop(columns=['EstimatedSalary'],axis=1)

X.head()

Y=data['EstimatedSalary']

Y

1. SCALE THE INDEPENDENT VARIABLES:

from sklearn.preprocessing import scale

X=data.drop(columns=['Surname','Geography','Gender'],axis=1)

X.head()

X\_scaled=pd.DataFrame(scale(X),columns=X.columns)

X\_scaled.head()

1. SPLIT THE DATA INTO TRAINING AND TEST DATA:

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

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X\_scaled, Y, test\_size = 0.3, random\_state = 0)

X\_train

X\_train.shape

Y\_train.shape

X\_test

X\_test.shape Y\_test

Y\_test.shape