Assignment -2

DataVisualizationandPre-processing

|  |  |
| --- | --- |
| AssignmentDate | 19 September2022 |
| StudentName | Karthikeyan.R |
| StudentRollNumber | 611719104008 |
| MaximumMarks | 2Marks |

**ToPerformBelowTaskstocompletetheassignment:-**

# Step1.Downloadthedataset:[Dataset](../../../C:/Users/Administrator/Downloads/Churn_Modelling.csv)

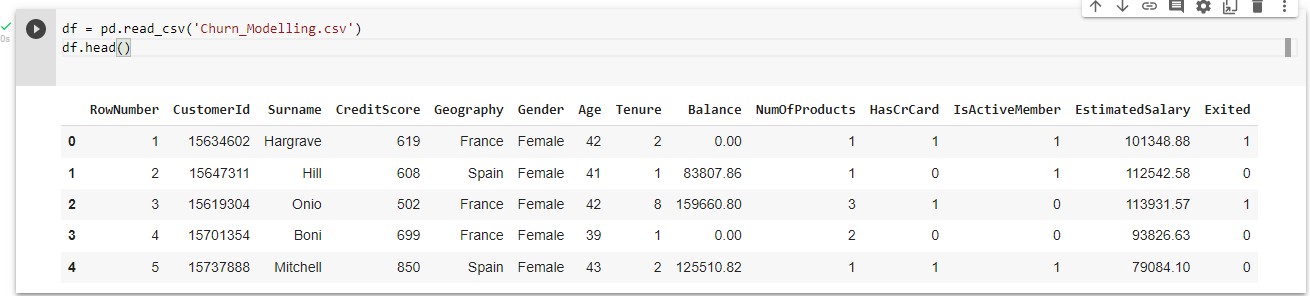
Step2.Loadthedataset.

import pandas as pdimportnumpyasnp

import matplotlib.pyplot as pltimportseaborn assns

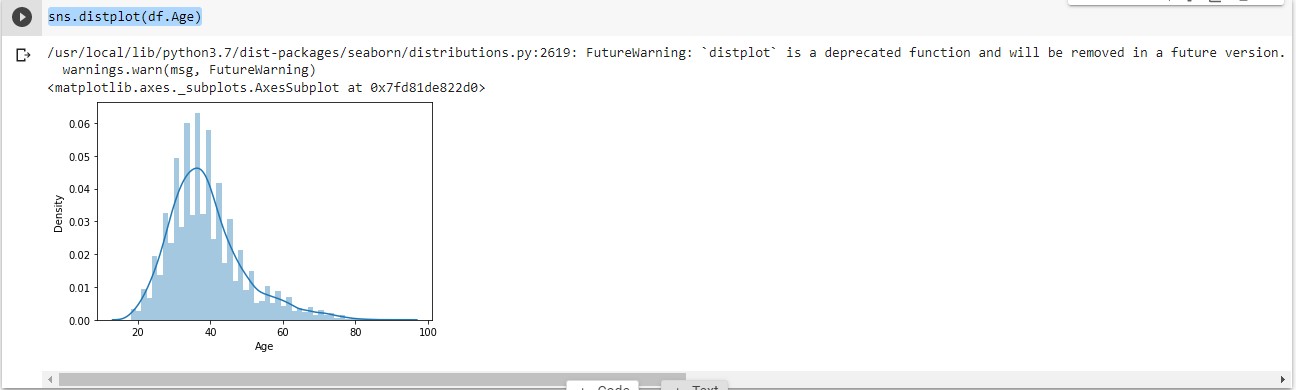
df=pd.read\_csv('Churn\_Modelling.csv')df.head()

# Output:



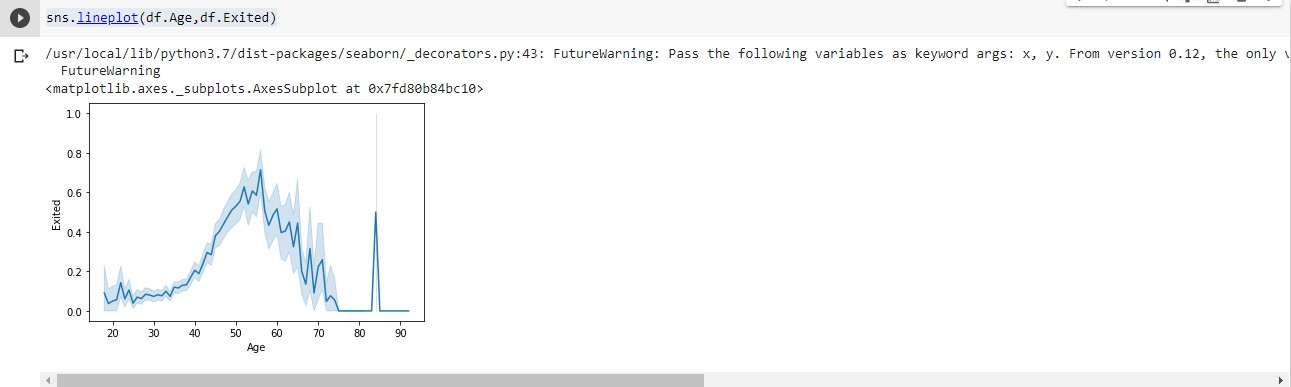
Step3.PerformBelowVisualizations.

* UnivariateAnalysissns.distplot(df.Age)Output:



sns.lineplot(df.Age,df.Exited)

# Output:



plt.pie(df.Gender.value\_counts(),[0.2,0],colors=['red','green'],labels=['Male','Female'],autopct='%1.1f%%') plt.title('GENDER')

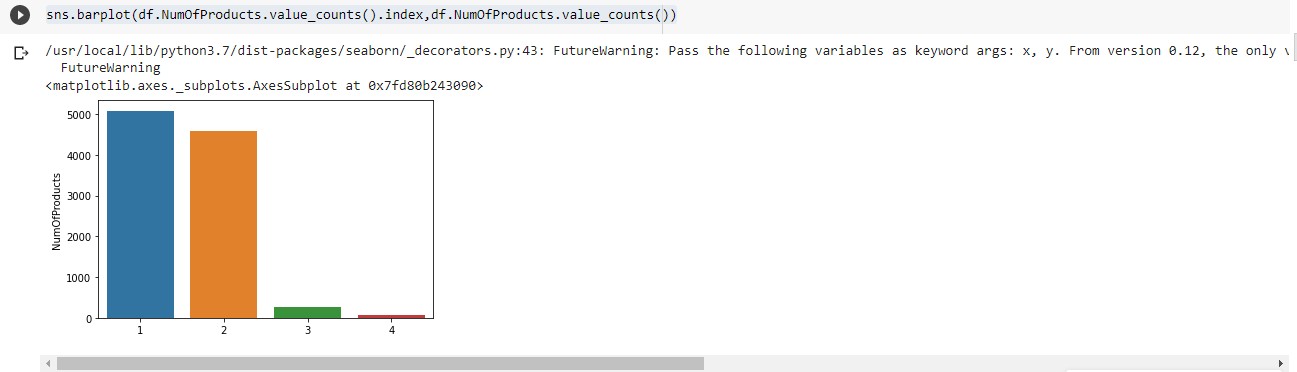
plt.show()

# Output:



sns.barplot(df.NumOfProducts.value\_counts().index,df.NumOfProducts.value\_counts())

# Output:



* Bi-VariateAnalysis

def countplot\_2(x,hue,title=None,figsize=(6,5)):plt.figure(figsize=figsize)sns.countplot(data=df[[x,hue]],x=x,hue=hue)plt.title(title)

plt.show()

countplot\_2('IsActiveMember','NumOfProducts','CreditCardHoldersProductDetails')

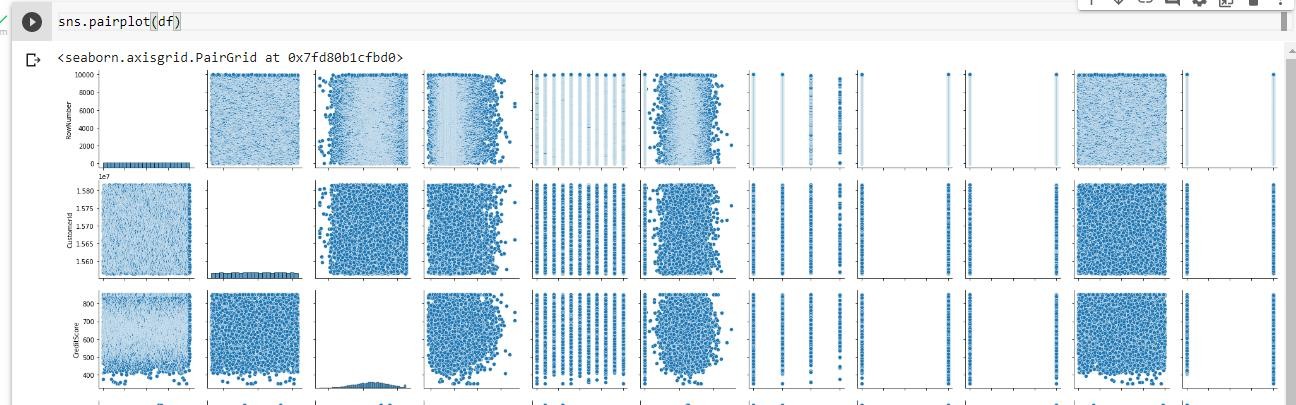
# Output:



* Multi-VariateAnalysis

sns.pairplot(df)

# Output:



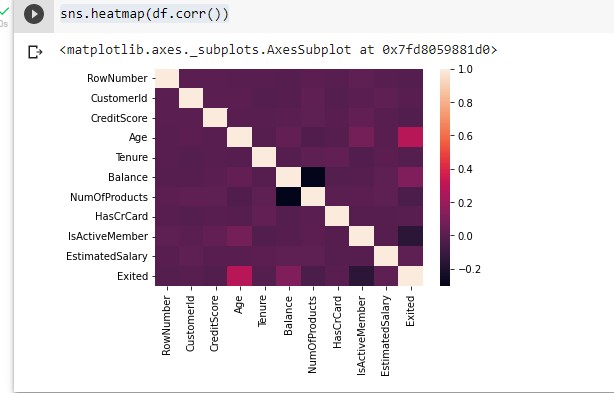
df.corr()

# Output:



sns.heatmap(df.corr())

# Output:



plt.figure(figsize=(16,15))sns.heatmap(df.corr(),annot=True)plt.show()

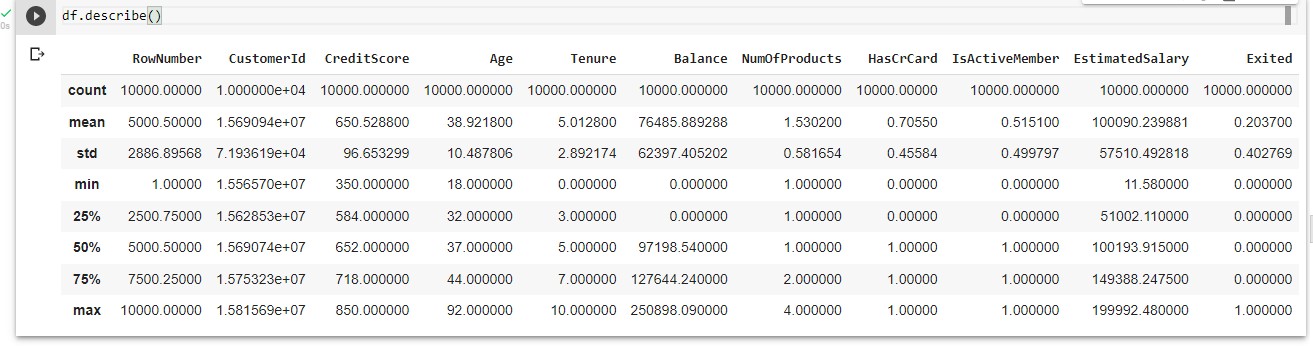
# Output:



Step4.Performdescriptivestatisticsonthedataset.

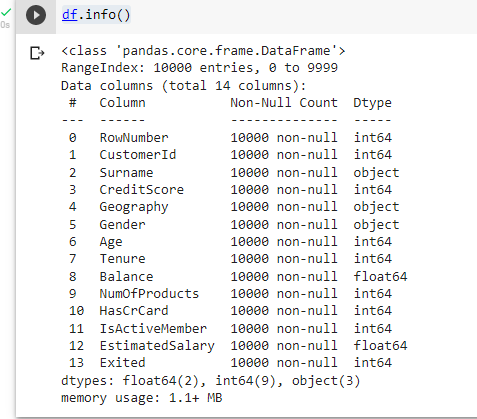
df.describe()

# Output:



df.info()

# Output:

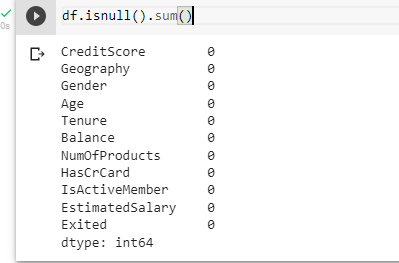


Step5. HandletheMissingvalues.

df=df.drop(columns=['RowNumber','CustomerId','Surname'])

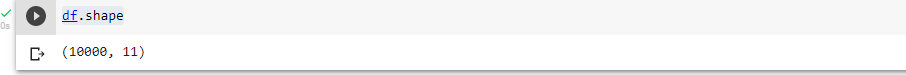
df.isnull().sum()

# Output:



df.shape

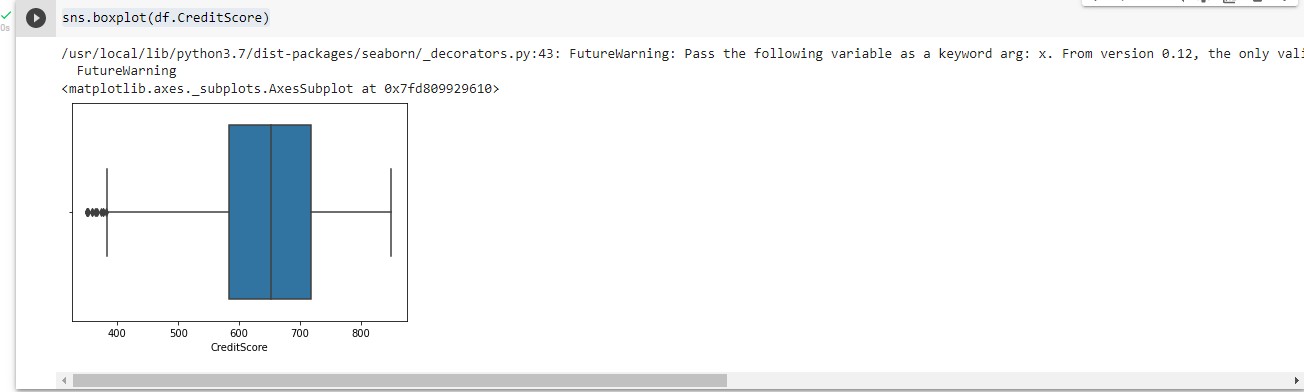
# Output:



Step6.Findtheoutliersandreplacethe outliers

sns.boxplot(df.CreditScore)

# Output:



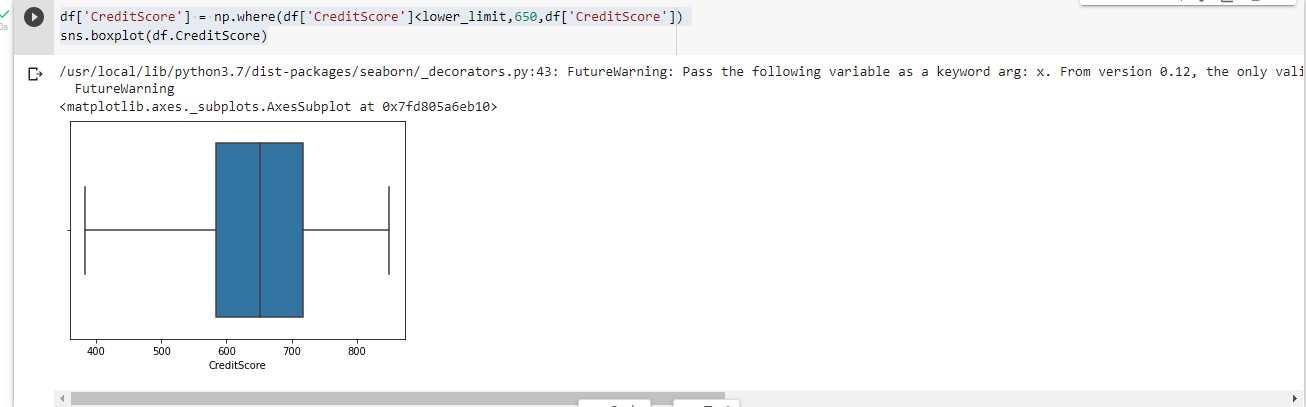
Q1 = df.CreditScore.quantile(0.25)Q3=df.CreditScore.quantile(0.75)

IQR=Q3-Q1

upper\_limit = Q3 + (1.5\*IQR)lower\_limit= Q1-(1.5\*IQR)

df['CreditScore']=np.where(df['CreditScore']<lower\_limit,650,df['CreditScore'])sns.boxplot(df.CreditScore)

# Output:



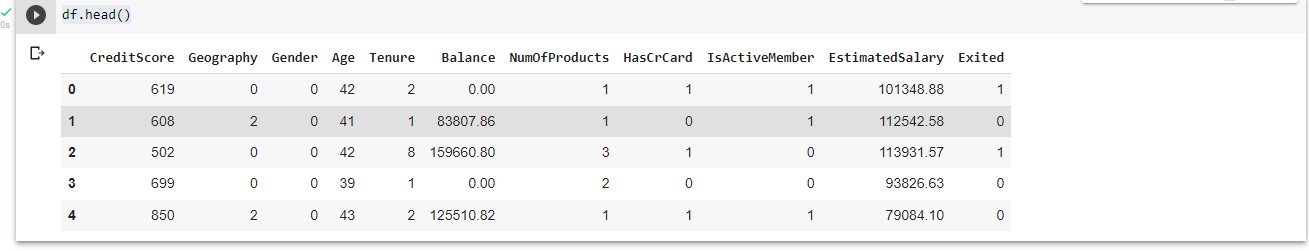
Step 7. Check for Categorical columns and perform encoding.fromsklearn.preprocessing importLabelEncoder

le= LabelEncoder()

df.Geography=le.fit\_transform(df.Geography)df.Gender=le.fit\_transform(df.Gender)

df.head()

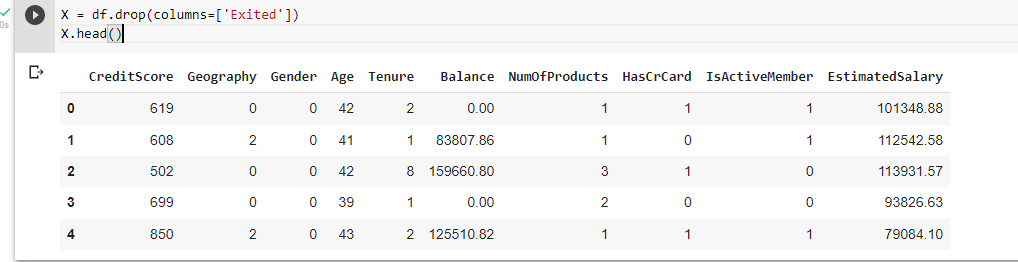
# Output:



Step8.Splitthedataintodependentandindependentvariables.

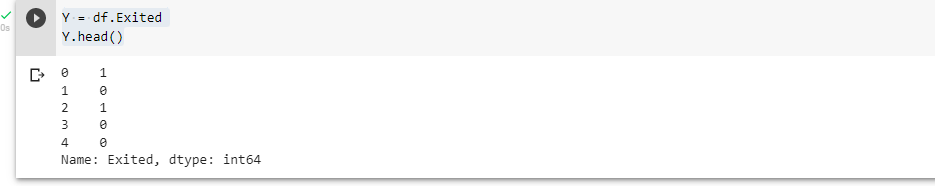
X=df.drop(columns=['Exited'])X.head()

# Output:



Y = df.ExitedY.head()

# Output:



Step9.Scaletheindependentvariables

from sklearn.preprocessing import MinMaxScalerscale=MinMaxScaler()

X\_scaled = pd.DataFrame(scale.fit\_transform(X),columns=X.columns)Step10.Splitthe data into training andtesting

fromsklearn.model\_selectionimporttrain\_test\_split

x\_train,y\_train,x\_test,y\_test=train\_test\_split(X\_scaled,Y,test\_size=0.2,random\_state=0)

# Output:

