Assignment-2

Python Programming

Assignment Date	19 September 2022
Student Name	U.Manju
Student Roll Number	962719106019
Maximum Marks	2Marks

Question-1:

Download the dataset

Link:- https://drive.google.com/file/d/15dFx93Pnri_PIPTMGyrs_9d8jcqKPuzF/view?usp=sharing

Question-2:

Load the dataset

Solution:

```python
df=pd.read_csv("/content/Churn_Modelling.csv")
print(df)
df.info()
df.describe()
import matplotlib.pyplot as plt
import seaborn as sns
import math
df.isna().sum()
df.isnull().sum()
```

Share

+ Code   + Text   Copy to Drive   ··· Connecting   Edit

Files

2.Load the dataset

..

sample_data

Churn_Modelling.csv

```
df=pd.read_csv("/content/Churn_Modelling.csv")
print(df)
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender  Age  \
0             1    15634602   Hargrave          619    France  Female   42
1             2    15647311       Hill          608     Spain  Female   41
2             3    15619304       Onio          502    France  Female   42
3             4    15701354       Boni          699    France  Female   39
4             5    15737888   Mitchell          850     Spain  Female   43
...         ...         ...        ...          ...       ...     ...  ...
9995       9996    15606229   Obijiaku          771    France    Male   39
9996       9997    15569892  Johnstone          516    France    Male   35
9997       9998    15584532        Liu          709    France  Female   36
9998       9999    15682355  Sabbatini          772   Germany    Male   42
9999      10000    15628319     Walker          792    France  Female   28

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
```

✓ 1m 3s   completed at 9:46 PM

✓ 0s completed at 1:12 PM

Welcome To Colaboratory

File Edit View Insert Runtime Tools Help Cannot save changes

Share ⚙

+ Code + Text ⬦ Copy to Drive

[34]
```
 3   CreditScore     10001 non-null  object
 4   Geography       10001 non-null  object
 5   Gender          10001 non-null  object
 6   Age             10001 non-null  object
 7   Tenure          10001 non-null  object
 8   Balance         10001 non-null  object
 9   NumOfProducts   10001 non-null  object
 10  HasCrCard       10001 non-null  object
 11  IsActiveMember  10001 non-null  object
 12  EstimatedSalary 10001 non-null  object
 13  Exited          10001 non-null  object
dtypes: object(14)
memory usage: 1.1+ MB
```

[33] `df.describe()`

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrC
count	10001	10001	10001	10001	10001	10001	10001	10001	10001	10001	10
unique	10001	10001	2933	461	4	3	71	12	6383	5	
top	RowNumber	CustomerId	Smith	850	France	Male	37	2	0	1	
freq	1	1	32	233	5014	5457	478	1048	3617	5084	7

≡ Files

＋ Code ＋ Text △ Copy to Drive

✓ RAM ▬ Disk ▬ ▾ ✎ Editin

[x]

⬆ ⟲ ▲ 👁

⬆ ..
📁 sample_data
📄 Churn_Modelling.csv

✓ [31] `import math`

✓ ▶ `df.isna().sum()`

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

↑ ↓ ⊖ ⚙ ▯

▶ `df.isnull().sum()`

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
```

<>
▭
▶ Disk ▬▬▬▬▬ 70.45 GB available

✓ 0s completed at 7:25 PM

≡ Files

＋ Code ＋ Text △ Copy to Drive

✓ RAM ▬ Disk ▬ ▾ ✎ Ec

[x]

⬆ ⟲ ▲ 👁

⬆ ..
📁 sample_data
📄 Churn_Modelling.csv

✓ [32] EstimatedSalary   0
```
Exited             0
dtype: int64
```

↑ ↓ ⊖ ⚙

▶ `df.isnull().sum()`

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

3.Perform Below Visualizations

Univariate Analysis

<>
▭
▶ Disk ▬▬▬▬▬ 70.45 GB available

[ ] `sns.histplot(df.EstimatedSalary,kde=True)`

✓ 0s completed at 7:25 PM

Question-3:

Perform Below Visualizations

   *Univariate Analysis

Solution:

`sns.histplot(df.EstimatedSalary,kde=True)`
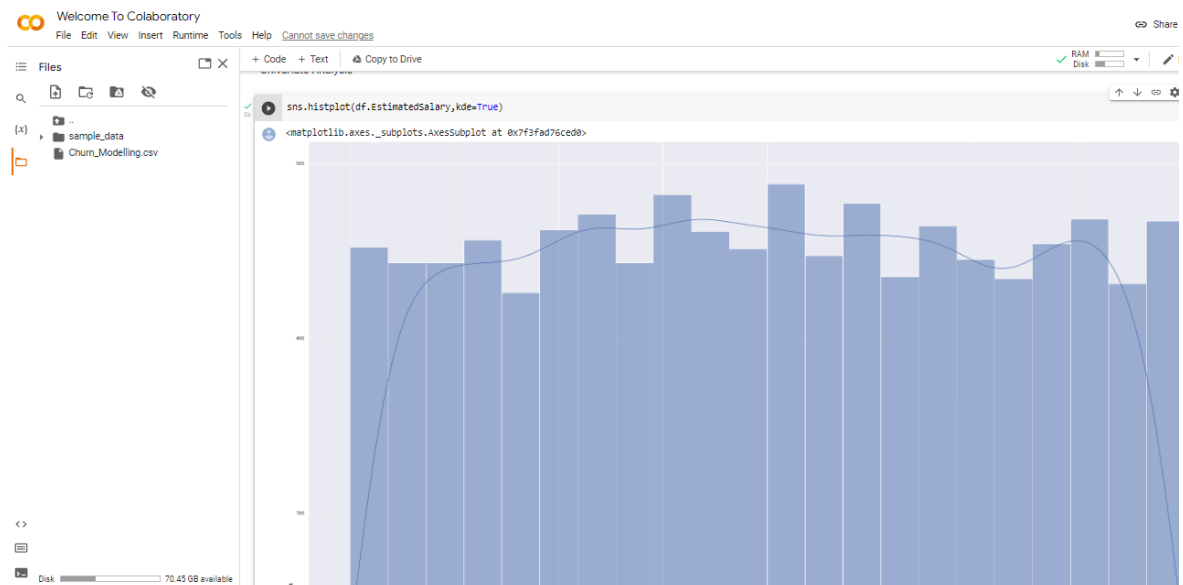
*Bivariate Analysis

Solution:

```
sns.scatterplot(df.Balance,df.EstimatedSalary)

plt.ylim(0,15000)
```
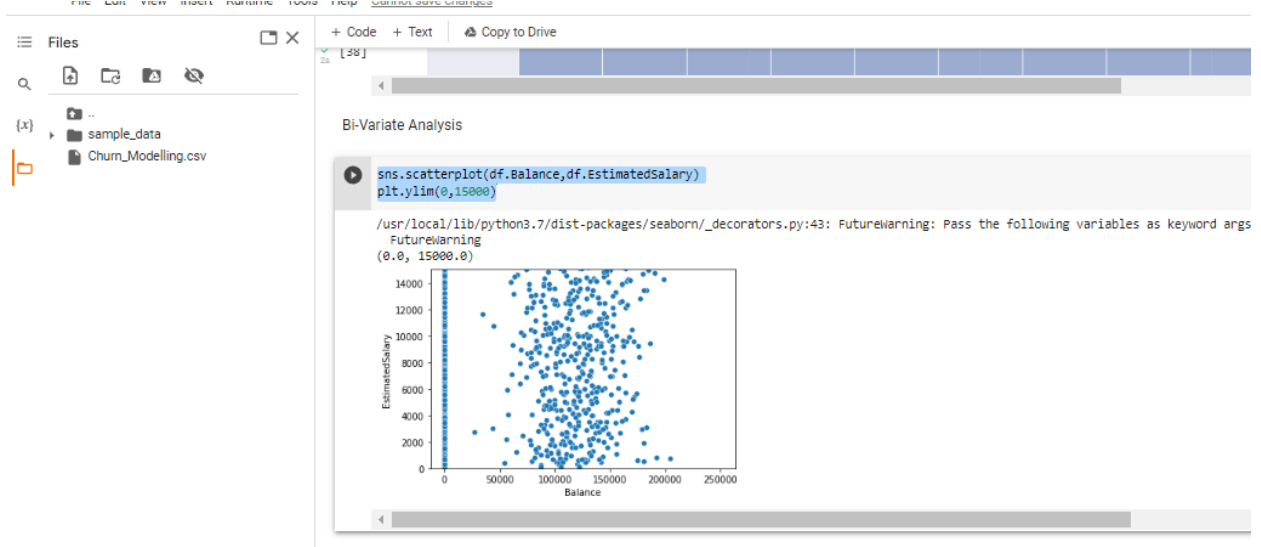
*Multi-Variate Analysis

Solution:

```
df=pd.read_csv("/content/Churn_Modelling.csv")

sns.pairplot(df)
```

4.Perform descriptive statistics on the dataset
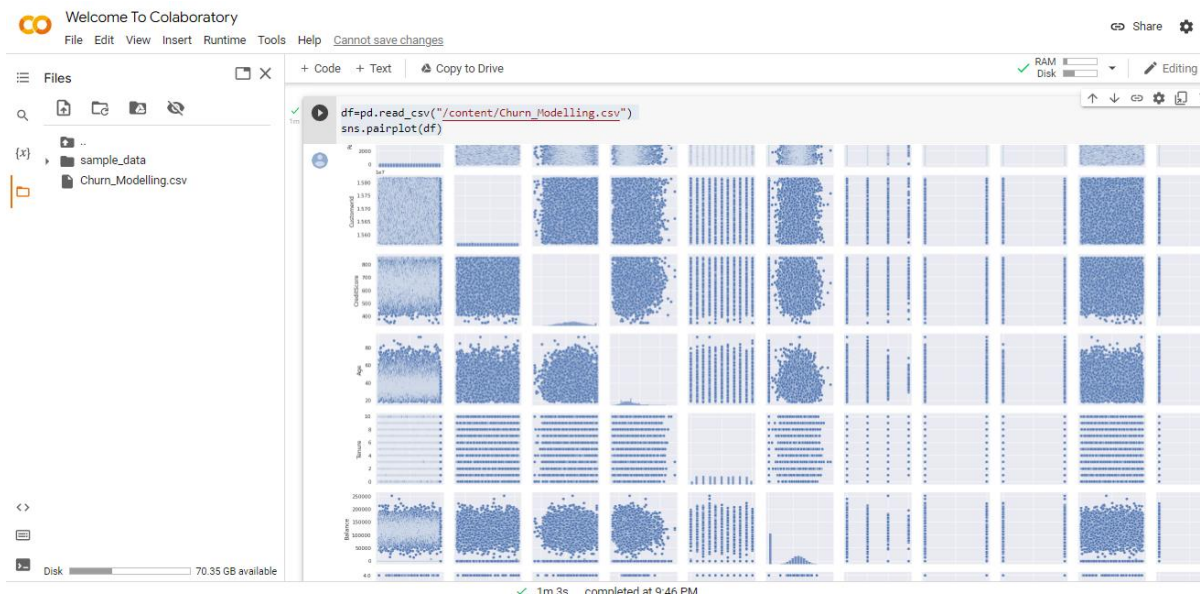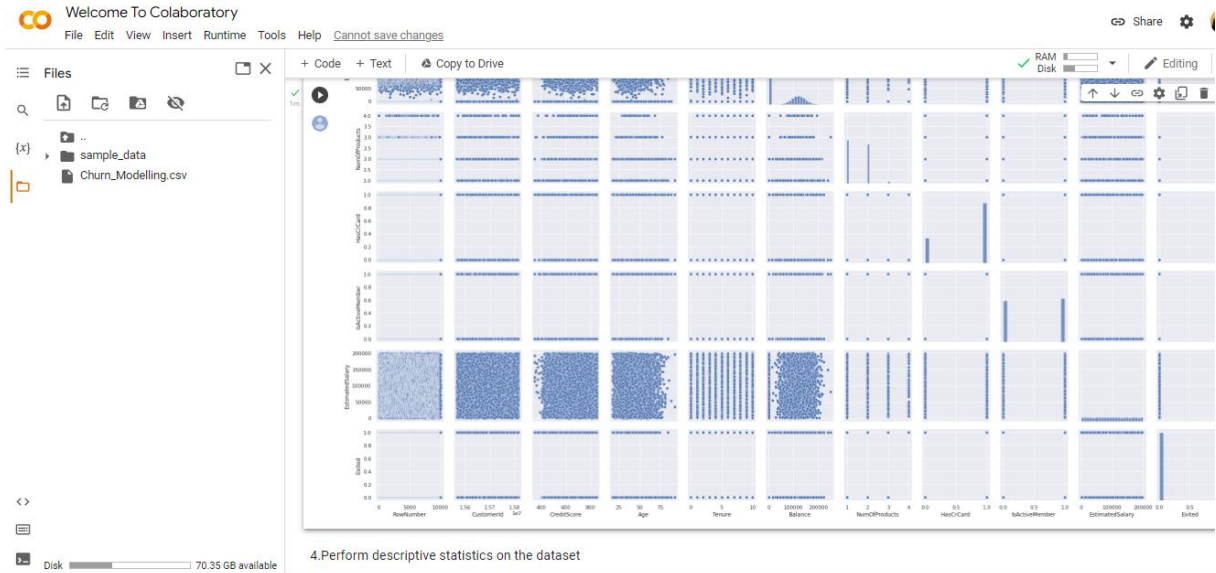
Question-4:

Perform descriptive statistics on the dataset

Solution:

```
df=pd.read_csv("/content/Churn_Modelling.csv")
df.describe(include='all')
```

+ Code + Text   △ Copy to Drive   ✓ RAM   ▾   ✏ Editing
                                                            Disk

4.Perform descriptive statistics on the dataset

```python
df=pd.read_csv("/content/Churn_Modelling.csv")
df.describe(include='all')
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	Has
count	10000.00000	1.000000e+04	10000	10000.000000	10000	10000	10000.000000	10000.000000	10000.000000	10000.000000	1000
unique	NaN	NaN	2932	NaN	3	2	NaN	NaN	NaN	NaN	
top	NaN	NaN	Smith	NaN	France	Male	NaN	NaN	NaN	NaN	
freq	NaN	NaN	32	NaN	5014	5457	NaN	NaN	NaN	NaN	
mean	5000.50000	1.569094e+07	NaN	650.528800	NaN	NaN	38.921800	5.012800	76485.889288	1.530200	
std	2886.89568	7.193619e+04	NaN	96.653299	NaN	NaN	10.487806	2.892174	62397.405202	0.581654	
min	1.00000	1.556570e+07	NaN	350.000000	NaN	NaN	18.000000	0.000000	0.000000	1.000000	
25%	2500.75000	1.562853e+07	NaN	584.000000	NaN	NaN	32.000000	3.000000	0.000000	1.000000	
50%	5000.50000	1.569074e+07	NaN	652.000000	NaN	NaN	37.000000	5.000000	97198.540000	1.000000	
75%	7500.25000	1.575323e+07	NaN	718.000000	NaN	NaN	44.000000	7.000000	127644.240000	2.000000	
max	10000.00000	1.581569e+07	NaN	850.000000	NaN	NaN	92.000000	10.000000	250898.090000	4.000000	

Question-5:

Handle the Missing Values

Solution:

```python
from ast import increment_lineno
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(color_codes=True)
df=pd.read_csv("/content/Churn_Modelling.csv")
df.head()
```

Question-6:

Find the outliers and replace the outliers

Solution:

```python
import pandas as pd
import matplotlib
from matplotlib import pyplot as pyplot
%matplotlib inline
matplotlib.rcParams['figure.figsize']=(10,6)
df=pd.read_csv("/content/Churn_Modelling.csv")
df.sample(5)
```

Question-7:

Check for Categorical columns and perform encoding

Solution:

```python
df=pd.read_csv("/content/Churn_Modelling.csv")
df.columns
import pandas as pd
import numpy as np
headers=['RowNumber', 'CustomerId', 'Surname', 'CreditScore', 'Geography',
        'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard',
        'IsActiveMember', 'EstimatedSalary', 'Exited']
df=pd.read_csv("/content/Churn_Modelling.csv",header=None,names=headers,na
_values="?")
df.head()
```

Question-8:

Split the data into dependent and independent variables

Solution:

```
x=df.iloc[:,:-1].values
print(x)
y=df.iloc[:,-1].values
print(y)
```
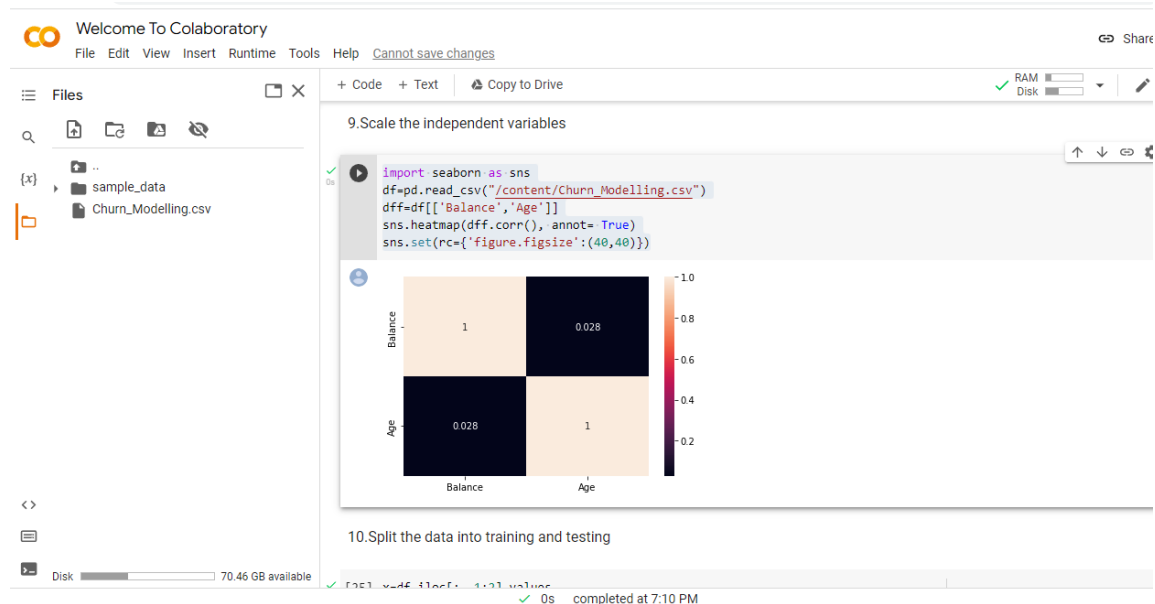
Question-9:

Scale the independent variables

Solution:

```
import seaborn as sns
df=pd.read_csv("/content/Churn_Modelling.csv")
dff=df[['Balance','Age']]
sns.heatmap(dff.corr(), annot= True)
sns.set(rc={'figure.figsize':(40,40)})
```

Question-10

Split the data into training and testing

Solution:

```python
x=df.iloc[:, 1:2].values
y=df.iloc[:,2].values
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test=train_test_split(x,y,test_size=0.2,random
_state=0)
print('Row count of x_train table'+'-'+str(f"{len(x_train):,}"))
print('Row count of y_train table'+'-'+str(f"{len(y_train):,}"))
print('Row count of x_test table'+'-'+str(f"{len(x_test):,}"))
print('Row count of y_test table'+'-'+str(f"{len(y_test):,}"))
```
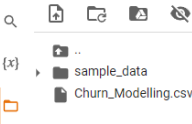
Welcome To Colaboratory

File   Edit   View   Insert   Runtime   Tools   Help   Cannot save changes

Share

+ Code   + Text   Copy to Drive

RAM
Disk

Editing

[19]

<__array_function__ internals> in nanmin(*args, **kwargs)

/usr/local/lib/python3.7/dist-packages/numpy/lib/nanfunctions.py in nanmin(a, axis, out, keepdims)
    317   # Fast, but not safe for subclasses of ndarray, or object arrays,
    318   # which do not implement isnan (gh-9009), or fmin correctly (gh-8975)
--> 319   res = np.fmin.reduce(a, axis=axis, out=out, **kwargs)
    320   if np.isnan(res).any():
    321   warnings.warn("All-NaN slice encountered", RuntimeWarning,

ValueError: zero-size array to reduction operation fmin which has no identity

SEARCH STACK OVERFLOW

10.Split the data into training and testing

```
x=df.iloc[:, 1:2].values
y=df.iloc[:,2].values
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test=train_test_split(x,y,test_size=0.2,random_state=0)
print('Row count of x_train table'+'-'+str(f"{len(x_train):,}"))
print('Row count of y_train table'+'-'+str(f"{len(y_train):,}"))
print('Row count of x_test table'+'-'+str(f"{len(x_test):,}"))
print('Row count of y_test table'+'-'+str(f"{len(y_test):,}"))
```

✓   0s   completed at 6:59 PM
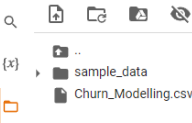
```
print('Row count of x_test table'+'-'+str(f"{len(x_test):,}"))
print('Row count of y_test table'+'-'+str(f"{len(y_test):,}"))
```

```
Row count of x_train table-8,000
Row count of y_train table-8,000
Row count of x_test table-2,001
Row count of y_test table-2,001
```