DATA PROCESSING

1.DOWNLOAD THE DATASET

from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive

The given dataset has been downloaded successfully

2.LOAD THE DATASET

import pandas as pd
data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")
data.head()

,	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

	Tenure	Balance	NumOfProducts	${\sf HasCrCard}$	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	

EstimatedSalary	Exited
101348.88	1
112542.58	0
113931.57	1
93826.63	0
79084.10	0
	112542.58 113931.57 93826.63

New Section

3 A)UNI VARIATE ANALYSIS

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

data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")
data.head()

,	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

	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	\
0	2	0.00	1	1	1	
1	1	83807.86	1	Θ	1	
2	8	159660.80	3	1	0	
3	1	0.00	2	Θ	0	
4	2	125510.82	1	1	1	

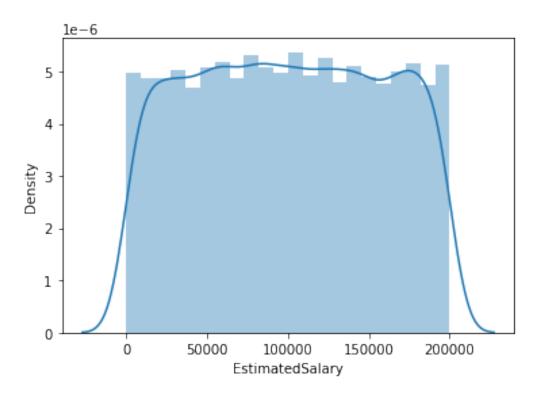
	EstimatedSalary	Exited
0	101348.88	1
1	112542.58	0
2	113931.57	1
3	93826.63	0
4	79084.10	0

sns.distplot(data['EstimatedSalary'])

/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)

<matplotlib.axes._subplots.AxesSubplot at 0x7ff6bf6f60d0>

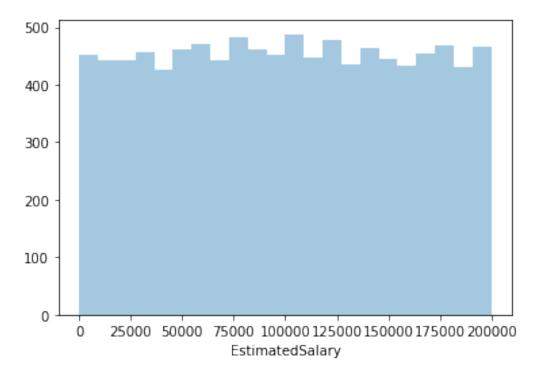


sns.distplot(data['EstimatedSalary'],kde=False)

/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)

<matplotlib.axes._subplots.AxesSubplot at 0x7ff6bf5aac10>

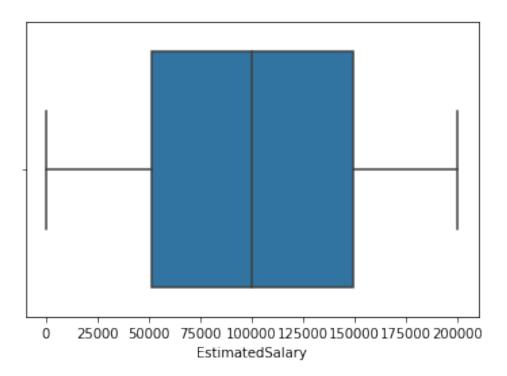


sns.boxplot(data['EstimatedSalary']),

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

FutureWarning

(<matplotlib.axes. subplots.AxesSubplot at 0x7ff6bf07ebd0>,)

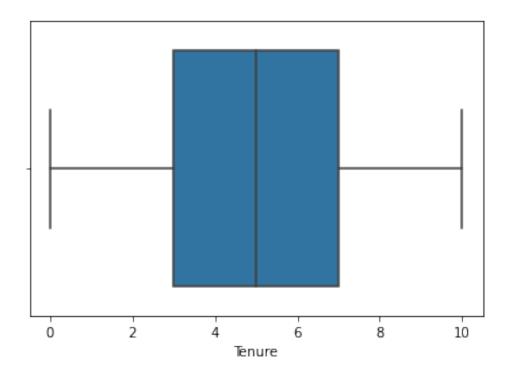


sns.boxplot(data['Tenure']),

/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

FutureWarning

(<matplotlib.axes. subplots.AxesSubplot at 0x7ff6bf00d690>,)



3 B)BI-VARIATE ANALYSIS

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")

data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")

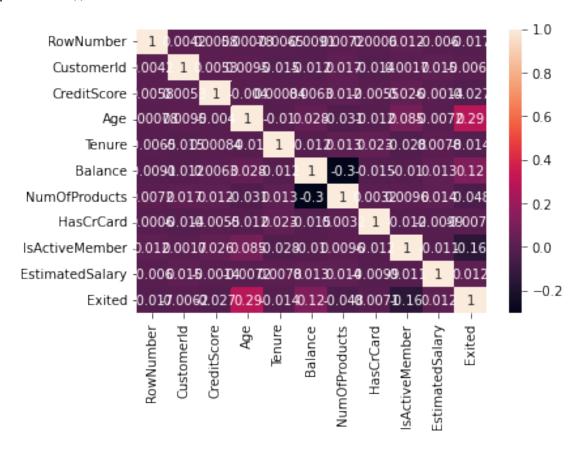
data.head()

	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

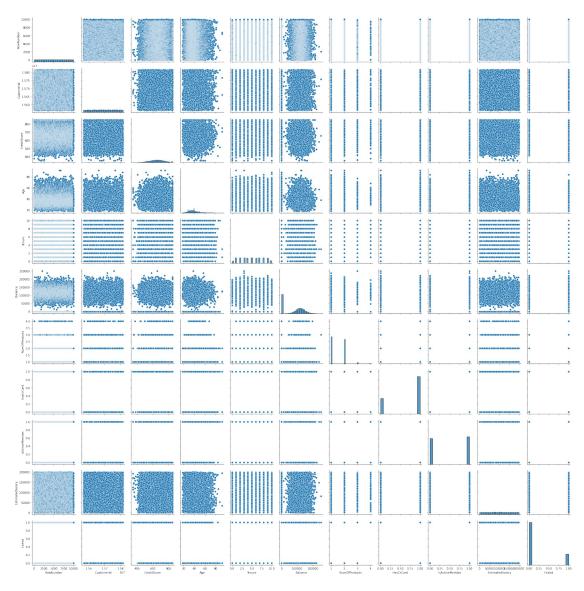
	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	\
0	2	0.00	1	1	1	

1 2 3 4	1 83807.80 8 159660.80 1 0.00 2 125510.83	9 9	1 3 2 1	0 1 0 1	1 0 0 1
	EstimatedSalary	Exited			
0	101348.88	1			
1	112542.58	Θ			
2	113931.57	1			
3	93826.63	Θ			
4	79084.10	0			

sns.heatmap(data.corr(),annot=True)
plt.show()



sns.pairplot(data)
plt.show()



3 C)MULTI-VARIATE ANALYSIS

```
from pydoc import help
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import scale
from sklearn.decomposition import PCA
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from scipy import stats
from IPython.display import display,HTML
%matplotlib inline
np.set_printoptions(suppress=True)
pd.set_option('display.max_rows',20)
import os
print(os.listdir("../NT project/"))
```

```
data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")
data.head()
```

da	ta.head()							
,	RowNumber	Custome	erId	Surname	CreditScore	Geography	Gender	Age
0	1	15634	1602	Hargrave	619	France	Female	42
1	2	2 15647	'311	Hill	608	Spain	Female	41
2	3	3 15619	304	Onio	502	France	Female	42
3	4	15701	354	Boni	699	France	Female	39
4	5	5 15737	888	Mitchell	850	Spain	Female	43
0 1 2 3 4	8 1 1 2 1	Balance 0.00 83807.86 59660.80 0.00 125510.82		1 1 3 2 1	HasCrCard 1 0 1 0 1	IsActiveMe	mber \ 1	
0 1 2 3 4	101 112 113 93	ISalary E 1348.88 2542.58 3931.57 3826.63 9084.10		d 1 0 1 0				
da	ta.columns	5						
'G	eography', 'Genc asCrCard', 'IsAc	ler', 'Age	e', ' er',	Tenure', '	Surname', 'Cı Balance', 'Nı Salary', 'Exi	umOfProduct		
da	ta.info()							
Ra	ngeIndex: ta columns	10000 ent	ries 14 co	.DataFrame , 0 to 9999 lumns): -Null Count	9			
0 1 2 3	Custome Surname	erId e	100 100	00 non-null 00 non-null 00 non-null 00 non-null	l int64 l object			

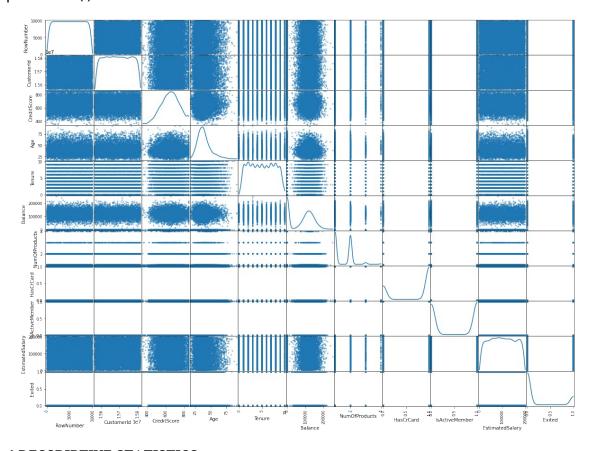
```
4
    Geography
                    10000 non-null object
5
    Gender
                    10000 non-null object
6
    Age
                    10000 non-null int64
7
    Tenure
                    10000 non-null int64
8
   Balance
                    10000 non-null float64
   NumOfProducts
9
                    10000 non-null int64
10 HasCrCard
                    10000 non-null int64
11
   IsActiveMember
                    10000 non-null int64
12
   EstimatedSalary 10000 non-null float64
13 Exited
                    10000 non-null int64
```

dtypes: float64(2), int64(9), object(3)

memory usage: 1.1+ MB

MATRIX SCATTERPLOT

pd.plotting.scatter_matrix(data.loc[:,"RowNumber":"Exited"],diagonal="
kde",figsize=(20,15))
plt.show()



4.DESCRIPTIVE STATISTICS

import numpy as np
import pandas as pd
from pandas import Series,DataFrame
import scipy
from scipy import stats

data=pd.read_csv("/content/drive/MyDrive/Colab Notebooks/Churn_Modelling.csv") data.head()

`	RowNumb	er	Custome	rId	Surname	CreditScore	Geography	Gender	Age
0		1	156346	502	Hargrave	619	France	Female	42
1		2	156473	311	Hill	608	Spain	Female	41
2		3	156193	304	Onio	502	France	Female	42
3		4	157013	354	Boni	699	France	Female	39
4		5	157378	888	Mitchell	850	Spain	Female	43
	_	_	_						
_	Tenure	ı	Balance	Num	OfProducts	_	IsActiveMe	_	
0	2		0.00	Num	1	1	IsActiveMe	1	
1		83	0.00 3807.86	Num	1 1	1 0	IsActiveMe	_	
	2 1 8 1	83	0.00	Num	1 1 3	1 0 1 0	IsActiveMe	1	
1 2	2 1 8	83 159	0.00 3807.86 9660.80	Num	1 1	1 0 1	IsActiveMe	1 1 0	
1 2 3	2 1 8 1 2	83 159 125	0.00 3807.86 9660.80 0.00 5510.82		1 1 3 2 1	1 0 1 0	IsActiveMe	1 1 0 0	
1 2 3	2 1 8 1 2	83 159 12! edSa	0.00 3807.86 9660.80 0.00 5510.82	kite	1 1 3 2 1	1 0 1 0	IsActiveMe	1 1 0 0	
1 2 3 4	2 1 8 1 2 Estimat 1	83 159 125 edSa 0134 1254	0.00 3807.86 9660.80 0.00 5510.82 alary Ex 48.88	kite	1 1 3 2 1 d 1	1 0 1 0	IsActiveMe	1 1 0 0	
1 2 3 4	2 1 8 1 2 Estimat 1 1	83 159 125 edSa 0134 1254 1393	0.00 3807.86 9660.80 0.00 5510.82 alary Ex 48.88 42.58	kite	1 1 3 2 1 d 1 0	1 0 1 0	IsActiveMe	1 1 0 0	
1 2 3 4	2 1 8 1 2 Estimat 1 1	83 159 125 edSa 0134 1254 1393 9382	0.00 3807.86 9660.80 0.00 5510.82 alary Ex 48.88	kite	1 1 3 2 1 d 1	1 0 1 0	IsActiveMe	1 1 0 0	

LOOKING AT SUMMARY STATISTICS THAT DESCRIBE A VARIABLE'S NUMERIC VALUES

data.sum()

RowNumber CustomerId	50005000 156909405694
Surname CreditScore	HargraveHillOnioBoniMitchellChuBartlettObinnaH 6505288
Geography	FranceSpainFranceFranceSpainSpainFranceGermany
Gender	FemaleFemaleFemaleFemaleMaleMaleFemaleMa
Age	389218
Tenure	50128
Balance	764858892.88
NumOfProducts	15302
HasCrCard	7055
IsActiveMember	5151
EstimatedSalary	1000902398.81
Exited	2037

dtype: object

```
data.sum(axis=1)
0
        15736618.88
1
        15844315.44
2
        15893456.37
3
        15795925.63
4
        15943385.92
9995
        15713313.64
9996
        15739522.38
9997
        15637370.58
        15861138.83
9998
9999
        15807478.57
Length: 10000, dtype: float64
data.median()
RowNumber
                    5.000500e+03
CustomerId
                    1.569074e+07
CreditScore
                    6.520000e+02
                    3.700000e+01
Age
Tenure
                    5.000000e+00
Balance
                    9.719854e+04
NumOfProducts
                    1.000000e+00
HasCrCard
                    1.000000e+00
IsActiveMember
                    1.000000e+00
EstimatedSalary
                    1.001939e+05
                    0.000000e+00
Exited
dtype: float64
data.mean()
RowNumber
                    5.000500e+03
CustomerId
                    1.569094e+07
CreditScore
                    6.505288e+02
                    3.892180e+01
Age
Tenure
                    5.012800e+00
Balance
                    7.648589e+04
NumOfProducts
                    1.530200e+00
HasCrCard
                    7.055000e-01
IsActiveMember
                    5.151000e-01
EstimatedSalary
                    1.000902e+05
                    2.037000e-01
Exited
dtype: float64
data.max()
RowNumber
                        10000
CustomerId
                     15815690
Surname
                       Zuyeva
CreditScore
                          850
Geography
                        Spain
```

Gender Male Age 92 Tenure 10 Balance 250898.09 NumOfProducts 1 HasCrCard IsActiveMember 1 EstimatedSalary 199992.48 Exited

dtype: object

 ${\tt mpg=data.EstimatedSalary}$

mpg.idxmax()

6646

LOOKING AT SUMMARY STATISTICS THAT DESCRIBE VARIABLE DISTRIBUTION

data.std()

RowNumber	2886.895680
CustomerId	71936.186123
CreditScore	96.653299
Age	10.487806
Tenure	2.892174
Balance	62397.405202
NumOfProducts	0.581654
HasCrCard	0.455840
IsActiveMember	0.499797
EstimatedSalary	57510.492818
Exited	0.402769

dtype: float64

data.var()

8.334167e+06
5.174815e+09
9.341860e+03
1.099941e+02
8.364673e+00
3.893436e+09
3.383218e-01
2.077905e-01
2.497970e-01
3.307457e+09
1.622225e-01

num=data.NumOfProducts
num.value_counts()

dtype: float64

1 5084 2 4590 3 266 4 60

Name: NumOfProducts, dtype: int64

data.describe()

_	RowNumber	CustomerId	CreditScore	Age	
	10000.00000	1.000000e+04	10000.000000	10000.000000	
10000.00 mean	5000.50000	1.569094e+07	650.528800	38.921800	
5.012800 std 2.892174	2886.89568	7.193619e+04	96.653299	10.487806	
min 0.000000	1.00000	1.556570e+07	350.000000	18.000000	
25% 3.000000	2500.75000	1.562853e+07	584.000000	32.000000	
50% 5.000000	5000.50000	1.569074e+07	652.000000	37.000000	
75% 7.000000	7500.25000	1.575323e+07	718.000000	44.000000	
max 10.00000	10000.00000 90	1.581569e+07	850.000000	92.000000	
max 2	Balance 10000.000000 76485.88928 62397.40520 0.000000 97198.540000 127644.240000 250898.090000 EstimatedSala 10000.0000 100090.2390 57510.4920 11.5800 51002.1100 100193.9150	0 10000.0000 8 1.5302 2 0.5816 0 1.0000 0 1.0000 0 2.0000 0 4.0000 ary Exi 000 10000.000 881 0.203 818 0.402 000 0.000	1000 10000.0000 100 0.7055 154 0.4558 100 0.0000 100 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000	0 10000.000000 0 0.515100 4 0.499797 0 0.000000 0 0.000000 0 1.000000 0 1.000000	
75% max	149388.247 199992.480	500 0.000			

5.HANDLE MISSING VALUE

import pandas as pd

data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")
data.head()

`	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

	Tenure	Balance	NumOfProducts	${\sf HasCrCard}$	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	

	EstimatedSalary	Exited
0	101348.88	1
1	112542.58	0
2	113931.57	1
3	93826.63	0
4	79084.10	Θ

data.shape

(10000, 14)

data.isnull()

Age \ 0 False	
0 False False False False False 1 False False False False	ender
1 False False False False	False
Falaa	False
	False
	False
False False False False False False	False

9995 False	Fal	se	False	False	Fals	е	False	False
9996 False	Fal	se	False	False	Fals	е	False	False
9997 False	Fal	se	False	False	Fals	е	False	False
9998	Fal	se	False	False	Fals	е	False	False
False 9999 False	Fal	se	False	False	Fals	е	False	False
0 1 2 3 4	Tenure False False False False	False False		False False False False False	False False False False False	IsAct	Fals Fals Fals Fals	se se se se
9995 9996 9997 9998 9999	False False False False False	False False False		False False False False False	False False False False False		Fals Fals Fals Fals Fals	se se se
0 1 2 3 4 9995 9996 9997 9998 9999	Estimat	edSalary False	False False False False False					
[10000 rows x 14 columns]								
<pre>data.isnull().sum()</pre>								
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
data.isnull().sum().sum()
0
FILLING NULL VALUES
df=data.fillna(value=0)
df
      RowNumber CustomerId
                                Surname CreditScore Geography
Age
              1
                    15634602
                               Hargrave
42
              2
                    15647311
                                   Hill
1
41
              3
2
                    15619304
                                   Onio
42
3
              4
                    15701354
                                   Boni
39
              5
                    15737888
                               Mitchell
4
43
                         . . .
                                     . . .
            . . .
                               Obijiaku
9995
           9996
                    15606229
39
9996
           9997
                    15569892
                              Johnstone
35
9997
           9998
                    15584532
                                    Liu
36
```

9998

42 9999

28

9999

10000

	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	Θ	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	

Walker

15682355 Sabbatini

15628319

Gender

Female

Female

. . .

Male

Male

Male

France

Spain

France Female

France Female

Spain Female

France Female

France Female

. . .

France

France

Germany

619

608

502

699

850

. . .

771

516

709

772

792

9998 9999	3 75075.31 4 130142.79	2 1	1 1	0 0
0 1 2 3 4	EstimatedSalary Exited 101348.88 1 112542.58 0 113931.57 1 93826.63 0 79084.10 0			
9995 9996 9997 9998 9999	96270.64 0 101699.77 0 42085.58 1 92888.52 1 38190.78 0			
_	orows x 14 columns]			

0
df1=data.fillna(value=5)
df1

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

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
9998	3	75075.31	2	1	0
9999	4	130142.79	1	1	0

	EstimatedSalary	Exited
0	101348.88	1
1	112542.58	0
2	113931.57	1
3	93826.63	0
4	79084.10	0
9995	96270.64	0
9996	101699.77	0
9997	42085.58	1
9998	92888.52	1
9999	38190.78	0

[10000 rows x 14 columns]

FILLING NULL VALUES WITH A PREVIOUS VALUE

df2=data.fillna(method='pad')
df2

_	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
Age 0 42	1	15634602	Hargrave	619	France	Female
1	2	15647311	Hill	608	Spain	Female
41 2 42	3	15619304	Onio	502	France	Female
3	4	15701354	Boni	699	France	Female
39 4 43	5	15737888	Mitchell	850	Spain	Female
9995 39	9996	15606229	0bijiaku	771	France	Male
9996 35	9997	15569892	Johnstone	516	France	Male
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
                            NumOfProducts
                                            HasCrCard
                                                         IsActiveMember
                  Balance
0
                     0.00
            2
                                         1
                                                      1
                                                                        1
            1
                                         1
                                                      0
                                                                        1
1
                83807.86
               159660.80
2
            8
                                         3
                                                      1
                                                                        0
3
                                         2
            1
                     0.00
                                                      0
                                                                        0
4
            2
                                         1
                                                      1
                                                                        1
               125510.82
            5
                                         2
                                                      1
                                                                        0
9995
                     0.00
                                                      1
9996
           10
                 57369.61
                                         1
                                                                        1
                                                                        1
9997
                     0.00
                                         1
                                                      0
            7
9998
            3
                 75075.31
                                         2
                                                      1
                                                                        0
9999
            4
               130142.79
                                         1
                                                      1
                                                                        0
      EstimatedSalary
                         Exited
0
             101348.88
                               1
1
             112542.58
                               0
2
             113931.57
                               1
3
              93826.63
                               0
4
              79084.10
                               0
9995
              96270.64
                               0
             101699.77
                               0
9996
9997
              42085.58
                               1
                               1
9998
              92888.52
9999
              38190.78
[10000 \text{ rows } \times 14 \text{ columns}]
df2.isnull().sum().sum()
0
####Filling NULL values with the next value
df3=data.fillna(method='bfill')
df3
      RowNumber CustomerId
                                   Surname CreditScore Geography
                                                                       Gender
Age
               1
                     15634602
                                 Hargrave
                                                      619
                                                              France
                                                                       Female
0
42
               2
                                      Hill
1
                     15647311
                                                      608
                                                               Spain
                                                                       Female
41
2
               3
                     15619304
                                      Onio
                                                      502
                                                              France
                                                                      Female
42
```

3		4 1570	L354	Boni	69	9 France	Female
39 4		5 15737	7888	Mitchell	85	0 Spain	Female
43							
 9995 39	99	96 15606	5229	0bijiaku	77	'1 France	Male
9996 35	99	97 15569	9892	Johnstone	51	.6 France	Male
9997	99	98 15584	1532	Liu	70	9 France	Female
36 9998	99	99 15682	2355	Sabbatini	77	'2 Germany	Male
42 9999 28	100	00 15628	3319	Walker	79	2 France	Female
0 1 2 3 4 9995 9996 9997 9998	Tenure 2 1 8 1 2 5 10 7 3 4	Balance 0.00 83807.86 159660.80 0.00 125510.82 0.00 57369.61 0.00 75075.31 130142.79	Num	1 1 3 2 1 2 1 1 2	HasCrCard 1 0 1 0 1 1 1 0 1 1	IsActiveMe	mber \ 1 0 0 1 0 1 1 0 0
0 1 2 3 4 9995 9996 9997 9998 9999	1 1 1	edSalary 01348.88 12542.58 13931.57 93826.63 79084.10 96270.64 01699.77 42085.58 92888.52 38190.78	Exite	ed 1 0 1 0 0 0 1 1 1 0 0			
[10000 rows v 1/ columns]							

[10000 rows x 14 columns]

DROPPING NULL VALUES

df4=data.dropna()
df4

	RowNumbe	r Custome	erId	Surname	CreditScore	Geography	Gender
Age 0	\	1 15634	602	Hargrave	619	France	Female
42 1	:	2 15647	311	Hill	608	Spain	Female
41 2	;	3 15619	304	Onio	502	France	Female
42 3		4 15701	.354	Boni	699	France	Female
39 4	!	5 15737	888	Mitchell	850	Spain	Female
43 							
 9995	9990	6 15606	229	0bijiaku	771	France	Male
39 9996	999	7 15569	892	Johnstone	516	France	Male
35 9997	9998	8 15584	532	Liu	709	France	Female
36 9998	9999	9 15682	355	Sabbatini	772	Germany	Male
42 9999	1000	0 15628	319	Walker	792	France	Female
28							
0	Tenure 2	Balance 0.00	NumC)fProducts 1	HasCrCard 1	IsActiveMem	ber \ 1
1 2 3	1 8	83807.86 159660.80		1 3	0 1		1 0
3 4	1 2	0.00 125510.82		2 1	0 1		0 1
 9995	 5	0.00					0
9996 9997	10 7	57369.61 0.00		1 1	1 0		1 1
9998 9999	3 4	75075.31 130142.79		2 1	1 1		0 0
	Estimate	dSalary E	xited	j			
0 1		1348.88 2542.58	1				
2 3		3931.57 3826.63	1				
4		9084.10					
9995 9996		6270.64 1699.77	6				
9997 9998	42	2085.58 2888.52]	L			
9999		8190.78	6				

[10000 rows x 14 columns]
df5=data.dropna(how='any')

	_			
ᅬ	£	_		
01		_		

Age 0 42 1	RowNumber	Customer]	d Surn	ame	CreditScore	e Geography	Gender
	1	1563466)2 Hargr	ave	619	9 France	Female
	2	1564731	.1 н	Iill	608	Spain	Female
41 2	3	1561936	04 0	nio	502	2 France	Female
42 3	4	1570135	54 B	Boni	699) France	Female
39 4 43	5	1573788	88 Mitch	ell	850) Spain	Female
9995 39	9996	1560622	9 Obiji	.aku	77:	l France	Male
9996	9997	1556989	2 Johnst	one	510	5 France	Male
35 9997	9998	1558453	32	Liu	709	9 France	Female
36 9998	9999	1568235	5 Sabbat	ini	772	2 Germany	Male
42 9999 28	10000	1562831	.9 Wal	.ker	792	2 France	Female
0 1 2 3 4 9995 9996 9997 9998 9999	8 15 1 2 12 5 10 5 7 3 5	Balance 0.00 83807.86 59660.80 0.00 25510.82 0.00 57369.61 0.00 75075.31	lumOfProdu	1 1 3 2 1 2 1 1 2	HasCrCard 1 0 1 0 1 1 0 1 1 1	IsActiveMe	mber \ 1
0 1 2 3 4	1125 1139 938	Salary Exi 348.88 542.58 931.57 826.63 084.10	1 0 1 0 0				

```
96270.64
9995
                               0
9996
             101699.77
                               0
9997
              42085.58
                               1
              92888.52
                               1
9998
9999
              38190.78
[10000 \text{ rows } \times 14 \text{ columns}]
replace()
import numpy as np
df6=df.replace(to replace=np.nan,value=8763)
df6
                   CustomerId
                                   Surname CreditScore Geography
                                                                       Gender
      RowNumber
Age
                     15634602
               1
                                  Hargrave
                                                      619
                                                              France
                                                                       Female
42
               2
                                      Hill
1
                     15647311
                                                      608
                                                               Spain
                                                                       Female
41
2
               3
                     15619304
                                      Onio
                                                      502
                                                              France
                                                                       Female
42
3
               4
                     15701354
                                      Boni
                                                      699
                                                              France
                                                                       Female
39
               5
                                 Mitchell
4
                     15737888
                                                      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
                            NumOfProducts
                                            HasCrCard
      Tenure
                  Balance
                                                         IsActiveMember \
0
                     0.00
            2
                                          1
                                                      1
                                                                        1
1
                 83807.86
                                         1
                                                                        1
            1
                                                      0
2
            8
               159660.80
                                         3
                                                      1
                                                                        0
3
                                         2
            1
                     0.00
                                                                        0
                                                      0
4
            2
               125510.82
                                         1
                                                      1
                                                                        1
          . . .
                                        . . .
                                                    . . .
                                                                      . . .
            5
                                         2
9995
                     0.00
                                                      1
                                                                        0
9996
           10
                 57369.61
                                         1
                                                      1
                                                                        1
                                         1
                                                                        1
9997
            7
                     0.00
                                                      0
```

75075.31

```
9999
           4 130142.79
                                       1
                                                   1
                                                                    0
      EstimatedSalary Exited
0
             101348.88
                              1
                              0
1
             112542.58
2
             113931.57
                              1
3
                              0
              93826.63
4
              79084.10
                              0
              96270.64
9995
                              0
9996
             101699.77
                              0
              42085.58
                              1
9997
9998
              92888.52
                              1
             38190.78
9999
[10000 rows x 14 columns]
interpolate()
data['EstimatedSalary']=data['EstimatedSalary'].interpolate(method='li
near')
data
      RowNumber CustomerId
                                 Surname CreditScore Geography
                                                                   Gender
Age
               1
                    15634602
                                Hargrave
                                                   619
                                                           France
                                                                   Female
0
42
               2
1
                    15647311
                                    Hill
                                                   608
                                                            Spain
                                                                   Female
41
               3
                                    Onio
                                                                   Female
2
                    15619304
                                                   502
                                                           France
42
3
               4
                    15701354
                                    Boni
                                                   699
                                                           France
                                                                   Female
39
                               Mitchell
4
               5
                    15737888
                                                   850
                                                            Spain
                                                                   Female
43
. . .
                          . . .
                                                   . . .
             . . .
           9996
9995
                    15606229
                                Obijiaku
                                                   771
                                                           France
                                                                     Male
39
9996
           9997
                    15569892
                               Johnstone
                                                   516
                                                                     Male
                                                           France
35
9997
           9998
                    15584532
                                     Liu
                                                   709
                                                                   Female
                                                           France
36
9998
           9999
                    15682355
                               Sabbatini
                                                   772
                                                         Germany
                                                                     Male
42
9999
          10000
                    15628319
                                  Walker
                                                   792
                                                           France
                                                                   Female
28
                          NumOfProducts HasCrCard IsActiveMember
      Tenure
                 Balance
0
           2
                    0.00
                                       1
                                                   1
                                                                    1
```

```
83807.86
                                                   0
                                                                    1
1
           1
                                       1
2
                                       3
                                                   1
                                                                    0
           8
              159660.80
3
                                       2
           1
                    0.00
                                                   0
                                                                    0
           2
4
              125510.82
                                       1
                                                   1
                                                                    1
         . . .
                     . . .
9995
           5
                    0.00
                                       2
                                                   1
                                                                    0
               57369.61
                                       1
                                                   1
                                                                    1
9996
          10
9997
           7
                    0.00
                                       1
                                                   0
                                                                    1
9998
           3
               75075.31
                                       2
                                                   1
                                                                    0
                                       1
                                                                    0
9999
           4
              130142.79
                                                   1
      EstimatedSalary Exited
0
            101348.88
                             1
1
                             0
            112542.58
2
            113931.57
                             1
3
                             0
             93826.63
4
                             0
             79084.10
9995
             96270.64
                             0
9996
            101699.77
                             0
9997
                             1
             42085.58
             92888.52
                             1
9998
9999
                             0
             38190.78
[10000 rows x 14 columns]
6.FIND THE OUTLIERS AND REPLACE THE OUTLIERS
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
data=pd.read csv("/content/drive/MyDrive/Colab
Notebooks/Churn Modelling.csv")
data1=data["CreditScore"]
outliers=[]
def detect outliers(data):
    threshold=3
    mean=np.mean(data)
    std=np.std(data)
    for i in data:
        z score=(i-mean)/std
        if np.abs(z score)>threshold:
            outliers.append(z score)
    return outliers
outlier_pt=detect_outliers(data1)
outlier pt
```

```
[-3.0163831068948417,
 -3.1095040882937757,
 -3.1095040882937757,
 -3.026729882605834,
 -3.0991573125827827,
 -3.1095040882937757,
 -3.1095040882937757,
 -3.1095040882937757]
INTERQUANTILE RANGE
sorted(data1)
[350,
 350,
 350,
 350,
 350,
 351,
 358,
 359,
 363,
 365,
 367,
 373,
 376,
 376,
 382,
 383,
 386,
 395,
 399,
 401,
 404,
 405,
 405,
 407,
 408,
 408,
 410,
 410,
 410,
 411,
 411,
 411,
 411,
 412,
 413,
 413,
```

413, 414,

415,

415,

415,

416,

416,

416,

416,

417,

418,

418,

418, 418,

418,

418,

419,

420,

420,

420,

421, 421,

421,

422,

422,

422,

422,

423,

424,

425,

425,

425,

425,

426,

426,

427,

427,

427,

427,

427, 427,

428,

428,

428,

429,

429,

429,

429,

430,

430,

430,

430,

430,

430,

430,

431,

431,

431,

431,

431, 432,

432,

432,

432,

432,

433,

433,

433,

434,

434,

434,

434, 434,

434,

435,

435,

435,

435,

435,

436,

436,

437,

437,

437,

437, 437,

437,

438,

438,

438,

438,

438,

438,

439,

439,

439,

439,

439,

440,

441,

442,

443,

443,

443,

443,

443,

443,

443,

443,

444,

444,

444,

444,

444, 444,

445,

445,

445,

445,

445,

445, 445,

445,

445,

445,

445,

445,

446,

446,

446,

446,

446,

447,

447,

447,

447,

448,

448,

449,

449,

449,

449,

449,

449, 450,

450,

451,

451,

451,

451,

451,

452,

452,

452,

452,

452,

452,

452,

452,

452,

452,

453,

453, 453,

453,

453,

453,

454,

454,

454,

454,

455,

455,

455,

455,

456,

456,

456,

456,

456,

457,

457,

457, 457,

457,

457,

457,

458, 458,

458,

458,

458,

458,

458,

459,

459,

459,

460,

460,

460,

460,

461,

461,

461,

461,

461,

461,

461,

461,

462,

462,

462, 462,

463,

463,

463,

463,

464,

464,

464,

464,

464,

464,

464,

465,

465,

465,

465,

465,

465,

465,

465, 465,

466, 466,

466,

466,

466,

466,

466,

466, 466,

467,

467,

467,

468,

468,

468, 468,

468,

468,

468,

468,

468,

468,

468,

469,

469,

469,

469,

469,

469, 469,

470,

470,

470,

470,

470,

470,

470,

470,

470,

471,

471,

471,

471,

471, 471,

471,

472,

472,

472,

472,

472,

472,

472,

472,

473, 473,

473,

473,

473,

473,

474,

474,

474, 474,

474,

474,

474,

474,

474,

475,

475,

475,

475,

475,

475,

476,

476,

476,

476,

476, 476,

476,

476,

477,

477, 477,

477,

477,

477,

477,

477,

477, 477,

477,

477,

477,

478,

478,

478,

478,

478,

478,

478,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

479,

480,

480,

480, 480,

480,

480,

480,

480,

480, 481,

481,

481,

481,

481,

481,

481,

481,

481,

481,

482,

482,

482,

482,

482, 482,

483,

483,

483,

483,

483,

483,

483,

484,

484,

484,

484,

484,

484,

484,

484,

484,

485,

485,

485,

485,

485,

485, 485,

485,

485, 485,

485,

485,

486,

486,

486,

486,

486,

486,

486,

486,

486,

487,

487,

487,

487,

487,

487,

487,

487,

487,

487, 487,

487,

487,

487,

488,

488,

488,

488,

488, 488,

488,

488,

489,

489,

489,

489,

489,

489,

489,

489,

489,

489,

489,

489,

490,

490, 490,

490,

490,

490,

490,

490,

490, 490,

490,

490,

490,

491,

491,

491,

491, 491,

491,

491,

491,

491,

491,

492,

492,

492,

492,

492,

492,

492, 492,

492,

492,

492,

492,

492,

493,

493,

493,

493,

493,

493,

493,

493,

493,

493,

493,

493, 494,

494,

494,

494,

494,

494, 494,

494,

494,

494,

494,

494,

494,

495,

495,

495,

495,

495,

495,

495,

495,

496,

496,

496, 496,

496,

496,

496,

496,

496,

496,

496,

496,

497,

497,

497,

497,

497,

497,

497,

497,

497,

497,

497,

497,

497,

497,

498,

498,

498,

498,

498,

498,

498,

498, 498,

498,

498,

498,

498,

498,

498,

498,

499,

499,

499,

499,

499,

499,

499, 499,

499,

499,

499,

499,

499,

499,

499,

500,

500,

500, 500,

500,

500, 500,

500,

500,

500,

501,

501,

501,

501,

501,

501,

501,

501,

501,

501, 501,

501,

501,

501,

501,

501, 501,

501,

501,

502,

502,

502, 502, 502,

502,

502,

502,

502,

502, 502,

502, 502,

502,

503,

503,

503,

503, 503,

503,

503,

503,

503, 503,

503, 503,

504,

504,

504,

504,

504,

504,

504, 504,

504,

504,

504,

504,

504, 504,

505,

505,

505,

505, 505,

505,

505,

505,

505,

505,

505,

505,

505,

505,

505,

506,

506,

506, 506,

506,

506,

506,

506,

506,

506,

506,

506, 506,

506,

506,

506,

507, 507,

507, 507,

507, 507,

507, 507,

507,

507,

507,

507, 507, 507,

507,

507,

507, 507, 508,

508,

508, 508,

508,

508,

508,

508,

508,

508,

508, 508,

508,

508, 508,

508,

509,

509,

509,

509,

509,

509,

509,

509,

509,

509, 509,

509, 509,

510,

510,

510,

510, 510,

510,

510,

510,

510,

510,

510,

510,

511,

511,

511,

511,

511,

511,

511,

511, 511,

511,

511,

511,

511, 511,

511,

511,

512,

512,

512,

512,

512, 512,

512,

512,

512,

512,

512, 512,

512,

512,

512,

512,

512,

512,

512, 512,

512,

512,

512,

512, 513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

513,

514,

514,

514,

514,

514, 514,

514,

514,

514,

514, 514,

515,

515,

515,

515,

515,

515,

515,

515,

515, 515,

515,

515,

515,

515,

515,

515,

516, 516,

516,

516,

516,

516, 516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516,

516, 516,

516,

517, 517,

517,

517,

517,

517, 517, 517, 517,

517,

517,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518,

518, 519,

519,

519,

519,

519,

519,

519,

519,

519,

519,

519,

519,

520,

520,

520,

520, 520,

520,

520,

520,

520,

520,

520,

520,

520, 520,

520,

520,

520,

520,

520,

520,

520,

520,

520,

520,

520,

520, 520,

521,

521,

521,

521,

521,

521,

521,

521,

521, 521,

521,

```
521,
 521,
 521,
 521,
 521,
 521,
 521,
 521,
 521,
 521,
 521,
 . . . ]
quantile1, quantile3=np.percentile(data1, [25,75])
print(quantile1,quantile3)
584.0 718.0
iqr value=quantile3-quantile1
print(iqr value)
134.0
lower bound val=quantile1-(1.5*iqr value)
upper bound val=quantile3+(1.5*iqr value)
print(lower_bound_val,upper_bound_val)
383.0 919.0
  1. CHECK FOR CATEGORICAL COLUMNS AND PERFORM ENCODING
import pandas as pd
import numpy as np
import seaborn as sns
%matplotlib inline
METHOD I
data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn Modelling.csv")
NEW DataM1=data
datal=pd.get dummies(NEW DataM1["Gender"])
data1.head()
   Female Male
0
        1
              0
1
        1
              0
2
        1
              0
3
        1
              0
              0
```

NEW_DataM1.drop('Gender',axis='columns')

	owNumber	CustomerId	Surname	CreditScore	Geography	Age
Tenure 0	1	15634602	Hargrave	619	France	42
2	2	15647311	Hill	608	Spain	41
1 2	3	15619304	Onio	502	France	42
8	4	15701354	Boni	699	France	39
1 4	5	15737888	Mitchell	850	Spain	43
2						
9995	9996	15606229	Obijiaku	771	France	39
5 9996	9997	15569892	Johnstone	516	France	35
10 9997	9998	15584532	Liu	709	France	36
7 9998	9999	15682355	Sabbatini	772	Germany	42
3 9999 4	10000	15628319	Walker	792	France	28
0 101348.1 112542.2 1113931.3 93826.6 4 1 79084.1 9995 96270.6 9996 101699.9997	83807.86 58 59660.80 57 0.00 3 25510.82 0 0.00 4 57369.61 77 0.00	NumOfProduc	ts HasCrCa 1 1 3 2 1 2 1	IsActiveN 1 0 1 1 1 0	1 1 0 0 1 0 1	
42085.5 9998 92888.5	75075.31		2	1	0	
	30142.79		1	1	0	

```
38190.78
```

```
Exited
0
               1
               0
1
2
               1
3
               0
               0
4
9995
               0
9996
               0
9997
               1
               1
9998
9999
               0
[10000 rows x 13 columns]
NEW_DataM1["Male"]=data1["Male"].to_list()
NEW_DataM1["Female"]=data1["Female"].to_list()
```

NEW_DataM1

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

	Tenure	Balance	NumOfProducts	${\sf HasCrCard}$	IsActiveMember	\
0	2	0.00	1	1	1	
1	1	83807.86	1	0	1	

```
2
               159660.80
                                                                        0
                                         3
                                                      1
3
                                          2
                                                      0
                                                                        0
            1
                     0.00
            2
4
               125510.82
                                         1
                                                      1
                                                                        1
9995
            5
                     0.00
                                         2
                                                                        0
                                                      1
9996
           10
                 57369.61
                                         1
                                                      1
                                                                        1
            7
                                         1
                                                      0
                                                                        1
9997
                     0.00
            3
9998
                 75075.31
                                         2
                                                      1
                                                                        0
9999
            4
               130142.79
                                         1
                                                      1
                                                                        0
      EstimatedSalary
                         Exited
                                  Male
                                         Female
0
             101348.88
                               1
                                      0
                                               1
1
             112542.58
                               0
                                      0
                                               1
2
                               1
                                      0
                                               1
             113931.57
3
              93826.63
                               0
                                      0
                                               1
4
                               0
                                      0
                                               1
              79084.10
9995
              96270.64
                               0
                                      1
                                               0
             101699.77
                               0
9996
                                      1
                                               0
9997
              42085.58
                               1
                                      0
                                               1
9998
                               1
                                      1
                                               0
              92888.52
              38190.78
                                               1
                               0
                                      0
9999
[10000 \text{ rows } \times 16 \text{ columns}]
NEW DataM1.head(2)
   RowNumber CustomerId
                              Surname
                                        CreditScore Geography
                                                                  Gender
                                                                           Age
0
            1
                  15634602
                             Hargrave
                                                 619
                                                         France
                                                                  Female
                                                                             42
1
            2
                  15647311
                                 Hill
                                                 608
                                                          Spain Female
                                                                            41
   Tenure
             Balance
                       NumOfProducts
                                        HasCrCard
                                                    IsActiveMember
0
         2
                 0.00
                                     1
                                                 1
                                                                   1
         1
            83807.86
                                     1
                                                 0
                                                                   1
1
   EstimatedSalary
                      Exited
                               Male
                                      Female
0
          101348.88
                            1
                                   0
                                            1
                                            1
1
          112542.58
                            0
                                   0
METHOD II
from sklearn.preprocessing import LabelEncoder
data=pd.read csv("/content/drive/MyDrive/Colab
Notebooks/Churn Modelling.csv")
13=LabelEncoder()
label=l3.fit transform(data["Gender"])
```

l3.classes_
array(['Female', 'Male'], dtype=object)
Data=NEW_DataM1.drop("Gender",axis='columns')
Data

	CustomerId	Surname	CreditScore	Geography	Age
1	15634602	Hargrave	619	France	42
2	15647311	Hill	608	Spain	41
3	15619304	Onio	502	France	42
4	15701354	Boni	699	France	39
5	15737888	Mitchell	850	Spain	43
_					
9996	15606229	0bijiaku	771	France	39
9997	15569892	Johnstone	516	France	35
9998	15584532	Liu	709	France	36
9999	15682355	Sabbatini	772	Germany	42
10000	15628319	Walker	792	France	28
Balance		ts HasCrCa	rd IsActive	Member	
0.00	\	1	1	1	
83807.86		1	0	1	
159660.80		3	1	Θ	
.57 0.00		2	0	0	
		1	1	1	
L0					
	•				
0.00		2	1	0	
57369.61		1	1	1	
0.00		1	0	1	
	2 3 4 5 9996 9997 9998 9999 10000 Balance tedSalary 0.00 .88 83807.86 .58 159660.80 .57 0.00 .3 125510.82 10 0.00 64 57369.61	1 15634602 2 15647311 3 15619304 4 15701354 5 15737888 9996 15606229 9997 15569892 9998 15584532 9999 15682355 10000 15628319 Balance NumOfProductedSalary 0.00 .88 83807.86 .58 159660.80 .57 0.00 .53 125510.82 10 0.00 .64 57369.61 .77	1 15634602 Hargrave 2 15647311 Hill 3 15619304 Onio 4 15701354 Boni 5 15737888 Mitchell 9996 15606229 Obijiaku 9997 15569892 Johnstone 9998 15584532 Liu 9999 15682355 Sabbatini 10000 15628319 Walker Balance NumOfProducts HasCrCa tedSalary 0.00 1 88 83807.86 1 .58 159660.80 3 .57 0.00 2 .63 1255510.82 1 10 0.00 2 .64 57369.61 1	1 15634602 Hargrave 619 2 15647311 Hill 608 3 15619304 Onio 502 4 15701354 Boni 699 5 15737888 Mitchell 850 9996 15606229 Obijiaku 771 9997 15569892 Johnstone 516 9998 15584532 Liu 709 9999 15682355 Sabbatini 772 10000 15628319 Walker 792 Balance NumOfProducts HasCrCard IsActively 188 83807.86 1 0 .58 159660.80 3 1 .57 0.00 2 0 63 1255510.82 1 1 10 0.00 2 1 64 57369.61 1 1	1 15634602 Hargrave 619 France 2 15647311 Hill 608 Spain 3 15619304 Onio 502 France 4 15701354 Boni 699 France 5 15737888 Mitchell 850 Spain 9996 15606229 Obijiaku 771 France 9997 15569892 Johnstone 516 France 9998 15584532 Liu 709 France 9999 15682355 Sabbatini 772 Germany 10000 15628319 Walker 792 France Balance tedSalary 0.00 1 1 1 1 188 83807.86 1 0 1 0 1 188 83807.86 1 0 0 1 158 159660.80 3 1 0 158 159660.80 3 1 0 158 159660.80 3 1 1 0 158 159660.80 3 1 1 0 158 158 159660.80 3 1 0 158 158 159660.80 3 1 0 158 158 159660.80 3 1 0 158 158 159660.80 3 1 0 158 158 159660.80 3 1 0 158 158 159660.80 3 1 0 158 158 159660.80 1 1 0 0 158 158 159660.80 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

42085	. 58			
9998	75075.31	2	1	0
92888	. 52			
9999	130142.79	1	1	0
38190.	. 78			

	Exited	Male	Female
0	1	0	1
1	0	0	1
2	1	0	1
3	0	0	1
4	0	0	1
9995	0	1	0
9996	0	1	0
9997	1	0	1
9998	1	1	0
9999	Θ	0	1

[10000 rows x 15 columns]

Data["Gender"]=label Data

Tonur	RowNumber	CustomerId	Surname	CreditScore	Geography	Age
Tenur 0	re \ 1	15634602	Hargrave	619	France	42
2 1	2	15647311	Hill	608	Spain	41
1 2	3	15619304	Onio	502	France	42
8 3 1	4	15701354	Boni	699	France	39
4	5	15737888	Mitchell	850	Spain	43
9995 5	9996	15606229	0bijiaku	771	France	39
9996 10	9997	15569892	Johnstone	516	France	35
9997 7	9998	15584532	Liu	709	France	36
9998 3	9999	15682355	Sabbatini	772	Germany	42
9999 4	10000	15628319	Walker	792	France	28

Balance NumOfProducts HasCrCard IsActiveMember

	lary 0.00	\		1	1	1	l
101348.88 1 8380	7.86			1	Θ	1	L
112542.58 2 159660	0.80			3	1	(9
	0.00			2	0	()
93826.63 4 125510 79084.10	0.82			1	1	1	L
				•			
	0.00			2	1	()
96270.64 9996 57369	9.61			1	1	1	L
	0.00			1	0	1	l
42085.58 9998 7507	5.31			2	1	(9
92888.52 9999 13014 38190.78	2.79			1	1	(9
1 2 3	d Mal 1 0 1 0	e Fema 0 0 0 0 0	le (1 1 1 1	Gender 0 0 0 0 0			
9996 9997 9998		1 1 0 1 0	0 0 1 0	1 1 0 1 0			
[10000 rows x 16 columns]							
8.SPLIT THE D	8.SPLIT THE DATA INTO DEPENDENT AND INDEPENDENT VARIABLES						
import mathlotlih nynlot as nlt							

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")

```
X=data.iloc[:,2:9]
X
```

```
Gender
                  CreditScore Geography
                                                         Tenure
                                                                   Balance
        Surname
                                                   Age
                                                    42
0
       Hargrave
                           619
                                  France
                                           Female
                                                              2
                                                                       0.00
                                   Spain
                                                                  83807.86
1
           Hill
                           608
                                           Female
                                                    41
                                                              1
2
           Onio
                          502
                                  France
                                          Female
                                                    42
                                                              8
                                                                 159660.80
3
                                          Female
                          699
                                                    39
                                                              1
           Boni
                                  France
                                                                       0.00
                                                              2
4
       Mitchell
                          850
                                   Spain
                                          Female
                                                    43
                                                                 125510.82
                           . . .
                                     . . .
                                              . . .
                                                    . . .
                                                            . . .
       Obijiaku
9995
                          771
                                                    39
                                                              5
                                  France
                                             Male
                                                                       0.00
9996
      Johnstone
                          516
                                  France
                                             Male
                                                    35
                                                             10
                                                                  57369.61
                                  France
9997
            Liu
                          709
                                           Female
                                                    36
                                                              7
                                                                       0.00
                                                                  75075.31
9998
      Sabbatini
                          772
                                 Germany
                                             Male
                                                    42
                                                              3
9999
         Walker
                          792
                                  France
                                           Female
                                                    28
                                                              4
                                                                 130142.79
[10000 \text{ rows } \times 7 \text{ columns}]
Y=data.iloc[:,9]
Υ
0
        1
        1
1
2
        3
3
        2
4
        1
9995
        2
9996
        1
9997
        1
        2
9998
9999
        1
Name: NumOfProducts, Length: 10000, dtype: int64
9.SCALE THE INDEPENDENT VARIABLES
import numpy as np
import pandas as pd
from pandas import Series, DataFrame
import matplotlib.pyplot as plt
from pylab import rcParams
import seaborn as sb
import scipy
import sklearn
from sklearn import preprocessing
from sklearn.preprocessing import scale
%matplotlib inline
rcParams['figure.figsize']=5,4
sb.set style('whitegrid')
Normalizing and transfroming features with MinMaxScalar() and fit transform()
data=pd.read csv("/content/drive/MyDrive/Colab
Notebooks/Churn Modelling.csv")
```

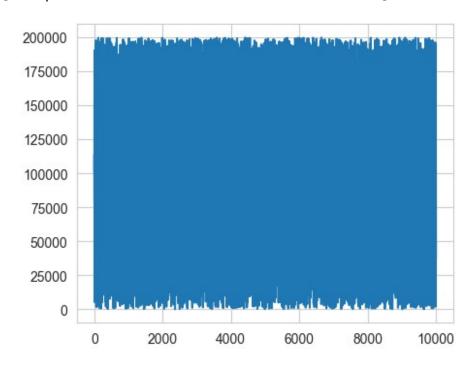
data.head()

RowNumb	er	CustomerId	Surname	 IsActiveMember
EstimatedS	ala	ry Exited		
0	1	15634602	Hargrave	 1
101348.88		1		
1	2	15647311	Hill	 1
112542.58		0		
2	3	15619304	Onio	 0
113931.57		1		
3	4	15701354	Boni	 0
93826.63		0		
4	5	15737888	Mitchell	 1
79084.10		0	_	

[5 rows x 14 columns]

tenure=data.EstimatedSalary
plt.plot(tenure)

[<matplotlib.lines.Line2D at 0x14ec8f2b400>]



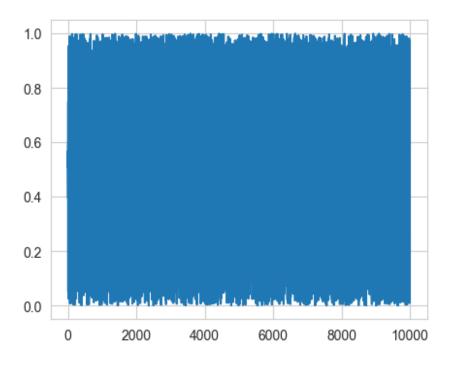
data[['Tenure']].describe()

	Tenure
count	10000.000000
mean	5.012800
std	2.892174
min	0.000000
25%	3.000000

```
50% 5.000000
75% 7.000000
max 10.000000
```

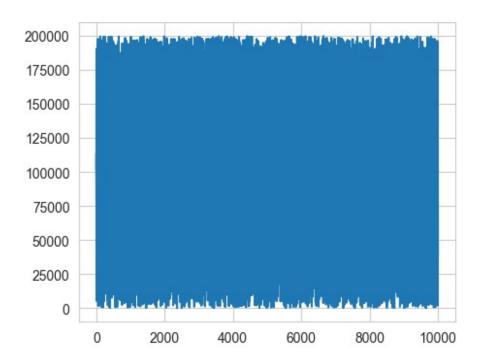
tenure_matrix=tenure.values.reshape(-1,1)
scaled=preprocessing.MinMaxScaler()
scaled_tenure=scaled.fit_transform(tenure_matrix)
plt.plot(scaled_tenure)

[<matplotlib.lines.Line2D at 0x14ec8dc02b0>]



std_tenure=scale(tenure,axis=0,with_mean=False,with_std=False)
plt.plot(std_tenure)

[<matplotlib.lines.Line2D at 0x14ec8ed07f0>]



10.SPLIT THE DATA INTO TRAINING AND TESTING

y=np.array(data["EstimatedSalary"])

```
import pandas as pd
data=pd.read_csv("/content/drive/MyDrive/Colab
Notebooks/Churn_Modelling.csv")
```

data.describe()

(10000, 1)

y.shape

(10000,)

```
RowNumber
                       CustomerId
                                         EstimatedSalary
                                                                 Exited
                                                           10000.000000
count
       10000.00000
                     1.000000e+04
                                            10000.000000
        5000.50000
                                           100090.239881
                                                               0.203700
mean
                     1.569094e+07
        2886.89568
                     7.193619e+04
                                            57510.492818
                                                               0.402769
std
                                                               0.00000
min
           1.00000
                     1.556570e+07
                                               11.580000
25%
        2500.75000
                     1.562853e+07
                                            51002.110000
                                                               0.00000
50%
        5000.50000
                     1.569074e+07
                                           100193.915000
                                                               0.00000
75%
        7500.25000
                     1.575323e+07
                                           149388.247500
                                                               0.00000
                                           199992.480000
       10000.00000
                     1.581569e+07
                                                               1.000000
max
[8 rows x 11 columns]
import numpy as np
x=np.array(data["CustomerId"]).reshape(-1,1)
x.shape
```

```
print(y)
[101348.88 112542.58 113931.57 ... 42085.58 92888.52 38190.78]
print(type(x))
<class 'numpy.ndarray'>
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30)
x_train.shape
(7000, 1)
x_test.shape
(3000, 1)
y_train.shape
(7000,)
y.shape
(10000,)
print(y_train.shape)
(7000,)
print(y_test.shape)
(3000,)
```