Assignment -2 Artificial Intelligence

Assignment Date	19 September 2022
Student Name	Mr. AKASH M
Student Roll Number	730419104005
Maximum Marks	2 Marks

Question-1:

- 1. Download the dataset:
- 2. Load the dataset.

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

url =
'https://drive.google.com/file/d/1_HcM0K8wt4b7FMLkc1V1dv0y6I_9ULzy/
view?usp=sharing'
path = 'https://drive.google.com/uc?
export=download&id='+url.split('/')[-2]
df = pd.read_csv(path)
```

df.sample(20)

7. ~ ~	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
Age 8075 58	8076	15745250	Simpson	850	France	Male
4957 39	4958	15600478	Watson	752	France	Male
6841 26	6842	15793491	Cherkasova	714	Germany	Male
4965 36	4966	15729515	McCarthy	782	France	Male
2828 33	2829	15716449	Fraser	527	Spain	Male
4732 53	4733	15653937	McIntyre	638	Germany	Female
6210 30	6211	15592197	Simmons	522	Spain	Male
5505 53	5506	15802466	Donaldson	534	France	Female
6450 28	6451	15781409	Lazarev	834	France	Female
5407 37	5408	15714431	Yeh	561	France	Male
7529 33	7530	15575430	Robson	579	France	Female
1887 34	1888	15680918	Freeman	613	Spain	Male
1590 39	1591	15651802	Day	632	Spain	Female
7578	7579	15656417	Marsh	582	France	Female

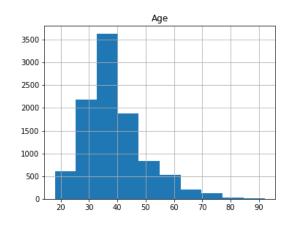
Male	France	751	Prokhorova	15736274	2693	39 2692 31
Male	Spain	478	Okechukwu	15580914	7032	7031 48
Female	France	731	Bird	15685706	2159	2158

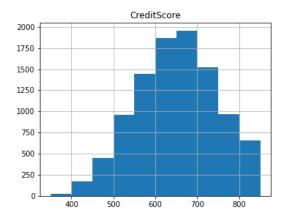
40							
3549	35	550 156	47725	Napolitano	6	75 France	Female
61 3772	27	73 156	99486	Johnson	7	45 Spain	Male
34	3 /	/3 136	199400	JOHNSON	/ ·	45 Spain	Male
5328	53	29 156	80234	Bray	6	67 Germany	Male
27					_		
	Tenure	Balanc		OfProducts	HasCrCard	IsActiveMemb	per \
8075	8	156652.1		1	0		0
4957	3	0.0		1	1		0
6841	3	119545.4		2	1		0
4965		148795.1		2	1		1
2828		132168.2		1	0		0
4732	1	123916.6		1	1		0
6210	3 7	0.0		2 2	1 1		0
5505 6450	6	0.0		1	1		1 0
5407	1	100443.3		2	0		1
7529	1	118392.7		1	1		1
1887	8	117300.0		1	1		0
1590		97854.3		2	1		0
7578	1	132077.4		2	1		0
2692	8	0.0		2	0		0
7031		83287.0		2	0		1
2158	7	118991.7		1	1		1
3549	5	62055.1	.7	3	1		0
3772	7	132944.5	3	1	1		1
5328	2	138032.1	.5	1	1		0
		edSalary	Exite				
8075		25899.21		1			
4957	1	.88187.05		0			
6841	_	65482.94		0			
4965	1	95681.43		0			
2828		98734.15		0			
4732 6210	1	16657.68		1 0			
5505	Τ	80619.17		0			
6450		74287.53		0			
5407	1	.01693.73		0			
7529		57564.75		0			
1887		39410.08		0			
1590	_	93536.38		0			
7578	1	92255.15		0			
2692		17550.49		0			
7031		44147.95		1			
2158	1	56048.64		0			
3549	1	66305.16		1			

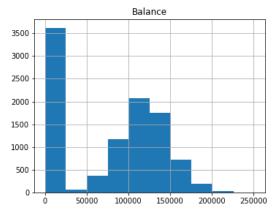
```
3772 31802.92 0
5328 166317.71 0
```

Perform Below Visualizations Univariate Analysis

```
features =['Age', 'CreditScore', 'Balance']
df[features].hist(figsize=(13, 10));
```



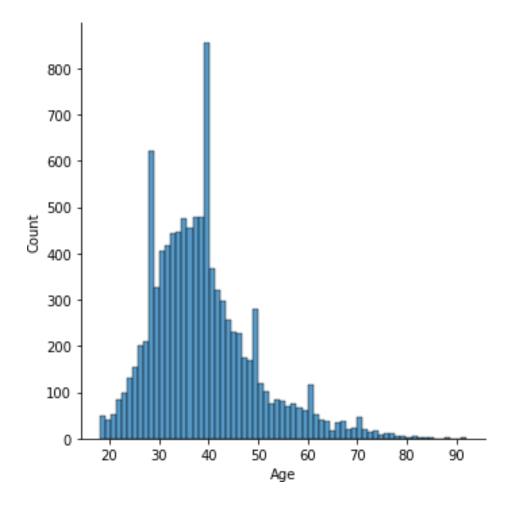




import seaborn as sns

sns.displot(df["Age"])

<seaborn.axisgrid.FacetGrid at 0x7fc07c40a350>

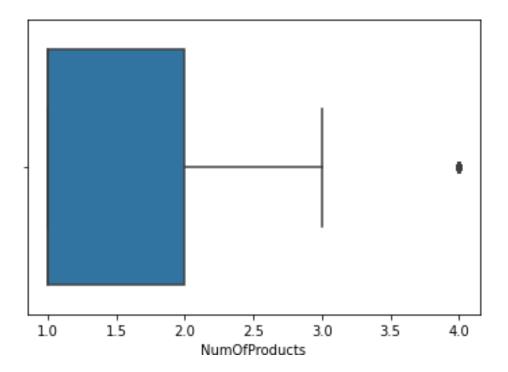


sns.boxplot(df["NumOfProducts"])

/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.

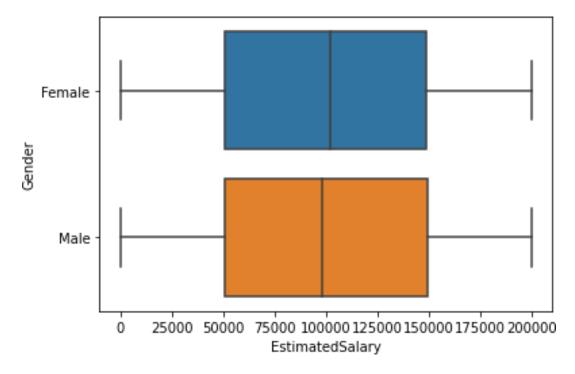
FutureWarning

<matplotlib.axes._subplots.AxesSubplot at 0x7fc0889c6a90>

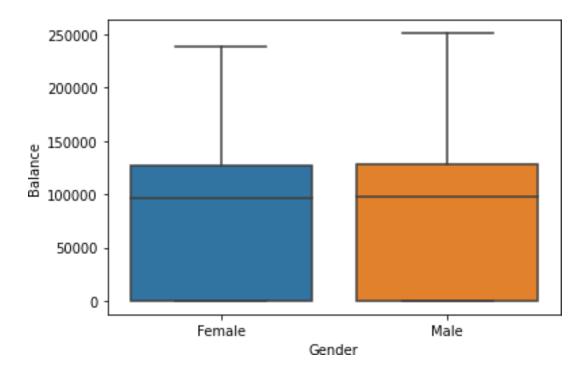


Bivariate Analysis

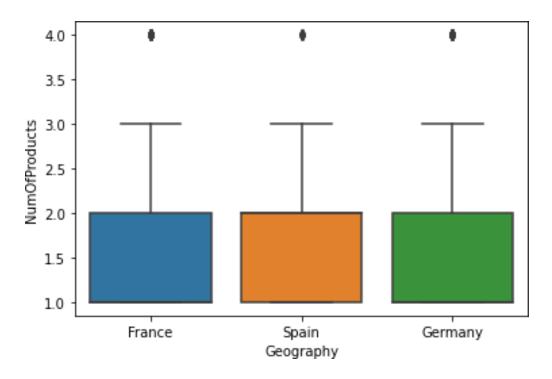
```
import seaborn as sns
sns.boxplot(x = df['EstimatedSalary'], y = df['Gender'] );
```



sns.boxplot(x=df['Gender'], y=df['Balance']);



sns.boxplot(x=df['Geography'],y=df['NumOfProducts']);



Multivariate Analysis

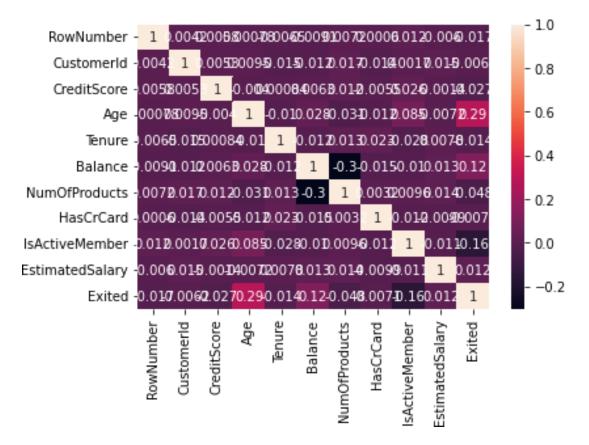
```
df_1 =
pd.DataFrame(df,columns=['NumOfProducts','EstimatedSalary','Balance'])
corrMatrix = df_1.corr()
```

sns.heatmap(corrMatrix, annot=True)
plt.show()



sns.heatmap(df.corr(),annot = True)

<matplotlib.axes._subplots.AxesSubplot at 0x7fc079668750>



4. Perform descriptive statistics on the dataset.

df.describe(include=['object'])

rue)

France

geography_counts

value counts

5014

Surname Geography Gender

```
10000
                   10000 10000
count
          2932
unique
         Smith
                  France
                           Male
top
            32
                    5014
                           5457
freq
df['CreditScore'].value counts()
df['CreditScore'].value counts().to frame()
df['Geography'].value counts()
France
           5014
Germany
           2509
Spain
           2477
Name: Geography, dtype: int64
geography counts=df['Geography'].value counts().to frame()
geography counts.rename(columns={'Geography':'value counts'},inplace=T
```

Germany 2509 Spain 2477

5. Handle the Missing values.

df.shape

(10000, 14)

df.isnull()

Age	RowNumb	er Custo	omerId	Surname	CreditScore	e Geography	Gender
0 False	Fal	se	False	False	False	e False	False
1 False	Fal	se	False	False	False	e False	False
2 False	Fal	se	False	False	False	e False	False
3 False	Fal	se	False	False	False	e False	False
4 False	Fal	se	False	False	False	e False	False
		••	• • •	• • •	• •		•••
9995 False	Fal	se	False	False	False	e False	False
9996 False	Fal	se	False	False	False	e False	False
9997 False	Fal	se	False	False	False	e False	False
9998 False	Fal	se	False	False	False	e False	False
9999 False	Fal	se	False	False	False	e False	False
	Tenure	Balance	NumOf	Products	HasCrCard	IsActiveMemb	er \
0	False	False		False	False	Fal	
1	False	False		False	False	Fal	
2	False	False		False	False	Fal	
3	False	False		False	False	Fal	
4	False	False		False	False	Fal	
9995	False	False		False	False	Fal	
9996	False	False		False	False	Fal	se
9997	False	False		False	False	Fal	se
9998	False	False		False	False	Fal	se
9999	False	False		False	False	Fal	se

1	False	False
2	False	False
3	False	False
4	False	False
9995	False	False
9996	False	False
9997	False	False
9998	False	False
9999	False	False

df.notnull()

Age	RowNumb	er Custo	merId	Surname	CreditScore	e Geography	Gender
0 True	Tr	ue	True	True	True	e True	True
1 True	Tr	rue	True	True	True	e True	True
2 True	Tr	ue	True	True	True	e True	True
3 True	Tr	ue	True	True	Tru	e True	True
4 True	Tr	ue	True	True	True	e True	True
•••		• •	• • •	• • •	• •		• • •
9995 True	Tr	ue	True	True	True	e True	True
9996 True	Tr	ue	True	True	True	e True	True
9997 True	Tr	ue	True	True	True	e True	True
9998 True	Tr	ue	True	True	True	e True	True
9999 True	Tr	rue	True	True	Tru	e True	True
	Tenure	Balance	NumOf	Products	HasCrCard	IsActiveMemb	er \
0	True	True		True	True	Tr	ue
1	True	True		True	True	Tr	ue
2	True	True		True	True	Tr	
3	True	True		True	True	Tr	ue
4	True	True		True	True	Tr	ue
• • •							• •
9995	True	True		True	True	Tr	
9996	True	True		True	True	Tr	
9997	True	True		True	True	Tr	ue

9998 9999	True True	True True		True True	True True	True True
2222	iluc	TTUC		iiuc	iiuc	iiuc
	EstimatedSa	alary	Exited			
0		True	True			
1		True	True			
2		True	True			
3		True	True			
4		True	True			
9995		True	True			
9996		True	True			
9997		True	True			
9998		True	True			
9999		True	True			

df.fillna(df.mean())

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.

"""Entry point for launching an IPython kernel.

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
Age 0	1	15634602	Hargrave	619	France	Female
42 1 41	2	15647311	Hill	608	Spain	Female
2 42	3	15619304	Onio	502	France	Female
3	4	15701354	Boni	699	France	Female
4 4 3	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	HasCrCard	IsActiveMember	\
2	0.00	1	1	1	
1	83807.86	1	0	1	
8	159660.80	3	1	0	
1	0.00	2	0	0	
2	125510.82	1	1	1	
				• • •	
5	0.00	2	1	0	
10	57369.61	1	1	1	
7	0.00	1	0	1	
3	75075.31	2	1	0	
4	130142.79	1	1	0	
	2 1 8 1 2 5 10 7 3	2 0.00 1 83807.86 8 159660.80 1 0.00 2 125510.82 5 0.00 10 57369.61 7 0.00 3 75075.31	2 0.00 1 1 83807.86 1 8 159660.80 3 1 0.00 2 2 125510.82 1 5 0.00 2 10 57369.61 1 7 0.00 1 3 75075.31 2	2 0.00 1 1 1 83807.86 1 0 8 159660.80 3 1 1 0.00 2 0 2 125510.82 1 1 5 0.00 2 1 10 57369.61 1 1 7 0.00 1 0 3 75075.31 2 1	2 0.00 1 1 1 1 83807.86 1 0 1 8 159660.80 3 1 0 1 0.00 2 0 0 2 125510.82 1 1 1 1 5 0.00 2 1 0 10 57369.61 1 1 1 7 0.00 1 0 1 3 75075.31 2 1 0

EstimatedSalary	Exited
101348.88	1
112542.58	0
113931.57	1
93826.63	0
79084.10	0
96270.64	0
101699.77	0
42085.58	1
92888.52	1
38190.78	0
	101348.88 112542.58 113931.57 93826.63 79084.10 96270.64 101699.77 42085.58 92888.52

df.fillna(df.median())

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=None') is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the reduction.

"""Entry point for launching an IPython kernel.

	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	9996	15606229	Obijiaku	77	1 France	Male		
39 9996	9997	15569892	Johnstone	51	6 France	Male		
35 9997	9998	15584532	Liu	70	9 France	Female		
36 9998	9999	15682355	Sabbatini	77	2 Germany	Male		
42 9999 28	10000	15628319	Walker	79	2 France	Female		
0 1 2	2 1 838	alance Nur 0.00 307.86 560.80	nOfProducts 1 1 3	HasCrCard 1 0 1	IsActiveMen	nber \ 1 1 0		
3	1	0.00 510.82	2	0		0 1		
9995 9996 9997 9998 9999	7 3 750	0.00 369.61 0.00 075.31	2 1 1 2 1	1 1 0 1		0 1 1 0 0		
	38190) rows x 14 c	3.88 2.58 1.57 6.63 4.10 0.64 9.77 5.58 3.52 0.78	ed 1 0 1 0 0 0 1 1 0 0 0 1 1 0 0 0 0 1 1 1					
df.isr	df.isnull().sum							
	d method NDF: nber Custome		_			: Age		
0	False	False	False	False	False	False		
False 1 False	False	False	False	False	False	False		

2	Fal	se	False	False	Fals	e False	False
False	Fal	se	False	False	Fals	e False	False
False	Fal	se	False	False	Fals	e False	False
False 							
9995	Fal	se	False	False	Fals	e False	False
False 9996	Fal	se	False	False	Fals	e False	False
False 9997	Fal	se	False	False	Fals	e False	False
False 9998	Fal	se	False	False	Fals	e False	False
False 9999	Fal	se	False	False	Fals	e False	False
False							
0 1 2 3 4 9995 9996 9997	Tenure False False False False False False False False False	Balance False False False False False False False False False	NumOfPr	roducts False	HasCrCard False False False False False False False False False	Fal Fal Fal Fal	.se .se .se .se .se .se
9998 9999	False False	False False		False False	False False	Fal Fal	se
9999		edSalary	Exited	raise	raise	rai	.se
0 1 2 3 4		False	False False False False False				
9995 9996 9997 9998 9999		False False False False False	False False False False False				

df[df.CreditScore.isnull()]

Empty DataFrame

Columns: [RowNumber, CustomerId, Surname, CreditScore, Geography,

Gender, Age, Tenure, Balance, NumOfProducts, HasCrCard,

IsActiveMember, EstimatedSalary, Exited]

Index: []

df.dropna(how='any').shape

(10000, 14)

df.dropna(subset=['CreditScore','Tenure'],how='any').shape

(10000, 14)

df.dropna(subset=['CreditScore','Tenure'],how='any')

ar.ar	opiia (Sabs	cc-[creare	DCOIC , ICHA.	ic j, now- any	,	
Age	RowNumbe	r Customer	Id Surnam	e CreditScore	e Geography	Gender
0 42		1 156346	02 Hargrav	e 619	France	Female
1 41		2 156473	11 Hil	1 608	Spain	Female
2 42		3 156193	004 Oni	502	France	Female
3		4 157013	Bon	i 699	France	Female
4 43		5 157378	88 Mitchel	1 850	Spain	Female
• • •						
9995 39	999	6 156062	29 Obijiak	u 771	France	Male
9996 35	999	7 155698	92 Johnston	e 516	France	Male
9997 36	999	8 155845	32 Li	u 709	France	Female
9998 42	999	9 156823	55 Sabbatin	i 772	Germany	Male
9999	1000	0 156283	19 Walke	r 792	? France	Female
0 1 2 3 4	1	Balance 0.00 83807.86 159660.80 0.00 125510.82		1 1 1 1 3 3 1 2 0 1 1	IsActiveMem	1 1 0 0
9995 9996 9997	5 10 7	0.00 57369.61 0.00				0 1 1

```
9998
        3 75075.31
                                             1
                                                            0
9999
         4 130142.79
                                  1
                                             1
                                                            \Omega
     EstimatedSalary Exited
0
           101348.88
                          1
1
           112542.58
                          Ω
2
           113931.57
                          1
            93826.63
3
                          0
            79084.10
                          0
. . .
                 . . .
                        . . .
           96270.64
9995
                          0
9996
           101699.77
                          0
9997
            42085.58
                          1
9998
            92888.52
9999
            38190.78
                          0
[10000 rows x 14 columns]
df.dropna(subset=['CreditScore','Tenure'],how='all').shape
(10000, 14)
df.dropna(subset=['CreditScore', 'Tenure'], how='all')
     RowNumber CustomerId Surname CreditScore Geography Gender
Age \
                 15634602 Hargrave
                                             619
0
             1
                                                  France Female
42
1
                 15647311
                               Hill
                                             608
                                                   Spain Female
41
2
             3
                                             502
                                                  France Female
                 15619304
                               Onio
42
3
             4
                 15701354
                                             699 France Female
                               Boni
39
             5
                 15737888 Mitchell
                                             850
4
                                                   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
```

	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	\
Ω	2	0 00	1	1	1	

772

792

Germany

France Female

Male

15682355 Sabbatini

15628319 Walker

36 9998

42 9999

28

9999

10000

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

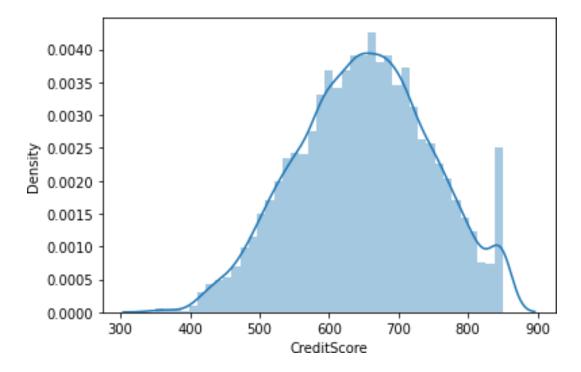
6. Find the outliers **and** replace the outliers

sns.distplot(df['CreditScore'])

/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 0x7fc0797203d0>

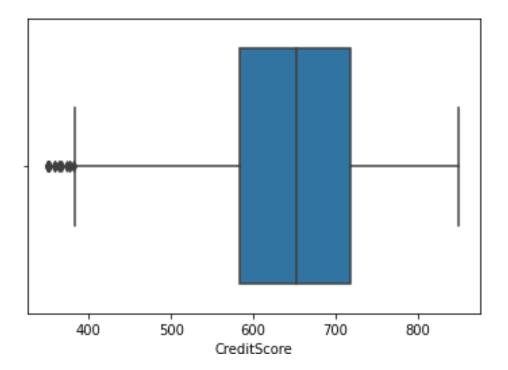


sns.boxplot(df['CreditScore'])

/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 0x7fc07989acd0>



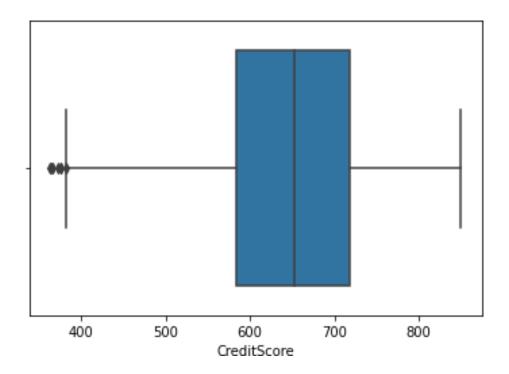
```
upper_limit = df['CreditScore'].mean() + 3*df['CreditScore'].std()
lower_limit = df['CreditScore'].mean() - 3*df['CreditScore'].std()
print('upper limit:', upper_limit)
print('lower limit:', lower limit)
```

upper limit: 940.488696208391
lower limit: 360.568903791609

df.loc[(df['CreditScore'] > upper_limit) | (df['CreditScore'] <
lower_limit)]</pre>

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
\					
1406	15612494	Panicucci	359	France	Female
1632	15685372	Azubuike	350	Spain	Male
1839	15758813	Campbell	350	Germany	Male
1963	15692416	Aikenhead	358	Spain	Female
2474	15679249	Chou	351	Germany	Female
8724	15803202	Onyekachi	350	France	Male
8763	15765173	Lin	350	France	Female
	1406 1632 1839 1963 2474 8724	1406 15612494 1632 15685372 1839 15758813 1963 15692416 2474 15679249 8724 15803202	1406 15612494 Panicucci 1632 15685372 Azubuike 1839 15758813 Campbell 1963 15692416 Aikenhead 2474 15679249 Chou 8724 15803202 Onyekachi	1406 15612494 Panicucci 359 1632 15685372 Azubuike 350 1839 15758813 Campbell 350 1963 15692416 Aikenhead 358 2474 15679249 Chou 351 8724 15803202 Onyekachi 350	1406 15612494 Panicucci 359 France 1632 15685372 Azubuike 350 Spain 1839 15758813 Campbell 350 Germany 1963 15692416 Aikenhead 358 Spain 2474 15679249 Chou 351 Germany 8724 15803202 Onyekachi 350 France

```
60
9624
          9625
                                               350
                  15668309
                               Maslow
                                                      France Female
40
      Tenure
              Balance NumOfProducts HasCrCard IsActiveMember
           6 128747.69
1405
                                    1
                                               1
1631
           1 152677.48
                                    1
                                               1
                                                               1
                                     2
1838
          0 109733.20
                                               0
                                                               0
1962
          8 143542.36
                                    3
                                               1
                                                               0
2473
          4 163146.46
                                    1
                                               1
                                                               0
8723
         10
                   0.00
                                    1
                                               1
                                                               1
8762
         3
                   0.00
                                    1
                                               0
                                                               0
9624
         0 111098.85
                                    1
                                               1
                                                               1
     EstimatedSalary Exited
1405
           146955.71
1631
           191973.49
                            1
           123602.11
                           1
1838
           141959.11
1962
                            1
2473
           169621.69
                            1
8723
           125823.79
                            1
8762
           113796.15
                            1
9624
          172321.21
new df = df.loc[(df['CreditScore'] <= upper limit) &</pre>
(df['CreditScore'] >= lower limit)]
print('before removing outliers:', len(df))
print('after removing outliers:',len(new df))
print('outliers:', len(df)-len(new df))
before removing outliers: 10000
after removing outliers: 9992
outliers: 8
sns.boxplot(new df['CreditScore'])
/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 0x7fc0797e5310>
```

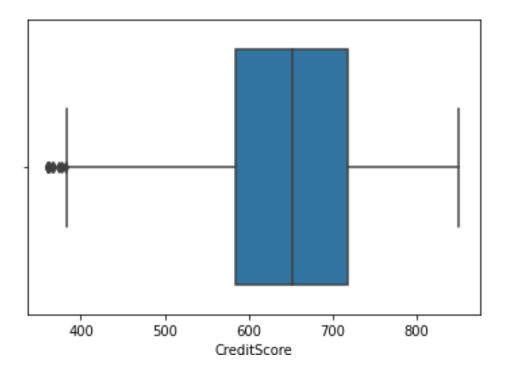


```
new_df = df.copy()
new_df.loc[(new_df['CreditScore']>=upper_limit), 'CreditScore'] =
upper_limit
new_df.loc[(new_df['CreditScore']<=lower_limit), 'CreditScore'] =
lower_limit
sns.boxplot(new_df['CreditScore'])</pre>
```

/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 0x7fc077c76a50>

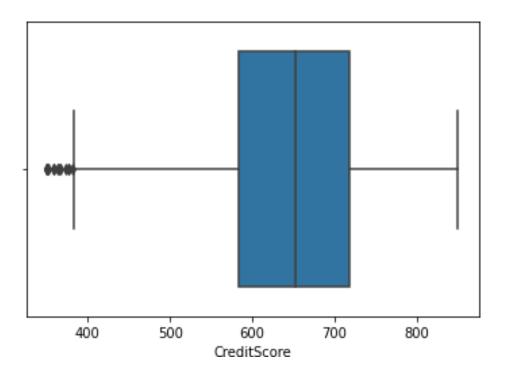


```
upper_limit = df['CreditScore'].quantile(0.99)
lower_limit = df['CreditScore'].quantile(0.01)
print('upper limit:', upper_limit)
print('lower limit:', lower_limit)
upper limit: 850.0
lower limit: 432.0
sns.boxplot(df['CreditScore'])
```

/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 0x7fc077c4bd90>



df.loc[(df['CreditScore'] > upper_limit) | (df['CreditScore'] <
lower_limit)]</pre>

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender
Age	\					
7	8	15656148	Obinna	376	Germany	Female
29	2.0	15656200	.	411	-	
29 29	30	15656300	Lucciano	411	France	Male
79	80	15803136	Postle	416	Germany	Female
41	0.0	10000100	100010	110	ccimarry	remare
99	100	15633059	Fanucci	413	France	Male
34						
149	150	15794413	Harris	416	France	Male
32						
• • •	• • •	• • •	• • •	• • •	• • •	• • •
9357	9358	15814405	Chesnokova	418	France	Female
46	3330	13011103	CITEBITOROVA	110	TTance	remare
9407	9408	15652835	Liang	419	Spain	Female
27			,		-	
9522	9523	15664504	Beede	418	France	Male
35						
9624	9625	15668309	Maslow	350	France	Female
40 9930	0021	15712604	Doggi	405	Compani	Mala
40	9931	15713604	Rossi	425	Germany	Male
10						

Tenure Balance NumOfProducts HasCrCard IsActiveMember \

```
7
         4 115046.74
                                           1
                                                          0
29
         0 59697.17
                                2
                                           1
                                                          1
79
         10 122189.66
                                2
                                           1
                                                         0
        9
99
                0.00
                                 2
                                           0
                                                          0
        0
                                2
                                           0
149
                 0.00
                                                         1
                               . . .
. . .
        . . .
                 . . .
                                          . . .
                                                        . . .
        9
9357
                 0.00
                                1
                                           1
                                                         1
                                1
9407
        2 121580.42
                                           0
                                                         1
9522
         7
                 0.00
                                 2
                                           1
                                                          1
9624
        0 111098.85
                                1
                                           1
                                                         1
9930
        9 166776.60
                                2
                                           0
                                                          1
     EstimatedSalary Exited
7
         119346.88
29
                         0
           53483.21
                         0
79
           98301.61
           6534.18
                         0
99
149
             878.87
                         0
. . .
                . . .
                       . . .
         81014.50
9357
                        1
                        0
9407
         134720.51
                        0
9522
          88878.15
9624
          172321.21
                        1
9930
          172646.88
                        0
[99 rows x 14 columns]
new df = df.loc[(df['CreditScore'] <= upper limit) &</pre>
(df['CreditScore'] >= lower limit)]
print('before removing outliers:', len(df))
```

```
print('after removing outliers:',len(new df))
print('outliers:', len(df)-len(new df))
```

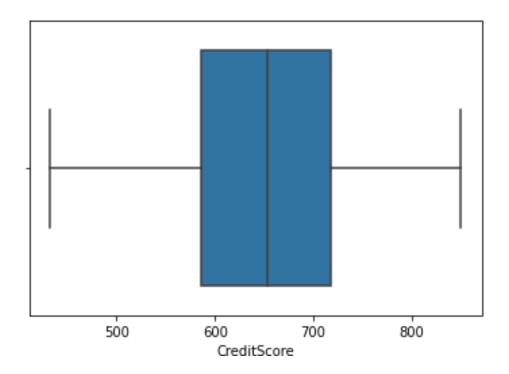
before removing outliers: 10000 after removing outliers: 9901 outliers: 99

sns.boxplot(new df['CreditScore'])

/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 0x7fc077bc8550>

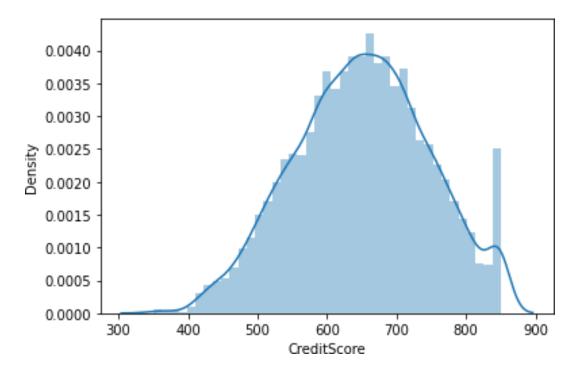


sns.distplot(df['CreditScore'])

/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 0x7fc077b2d510>

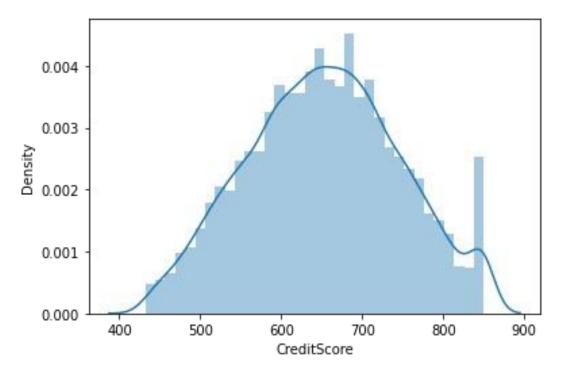


sns.distplot(new_df['CreditScore'])

/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 0x7fc077c61990>



```
7.) Check for Categorical columns and perform encoding.
df=df.iloc[:,:].values
df
array([[1, 15634602, 'Hargrave', ..., 1, 101348.88, 1],
       [2, 15647311, 'Hill', ..., 1, 112542.58, 0],
       [3, 15619304, 'Onio', ..., 0, 113931.57, 1],
       . . . ,
       [9998, 15584532, 'Liu', ..., 1, 42085.58, 1],
       [9999, 15682355, 'Sabbatini', ..., 0, 92888.52, 1],
       [10000, 15628319, 'Walker', ..., 0, 38190.78, 0]],
dtype=object)
8. Split the data into dependent and independent variables
'https://drive.google.com/file/d/1 HcM0K8wt4b7FMLkc1V1dv0y6I 9ULzy/
view?usp=sharing'
path = 'https://drive.google.com/uc?
export=download&id='+url.split('/')[-2]
df = pd.read csv(path)
x=df.iloc[:,4:7]
X
     Geography Gender
                        Age
                         42
0
        France Female
1
         Spain Female
                         41
        France Female
                         42
```

```
France Female
3
                         39
4
         Spain Female 43
                  . . .
           . . .
                         . . .
. . .
9995
                  Male
        France
                         39
9996
       France
                 Male
                       35
9997
        France Female
                         36
                         42
9998
       Germany
                  Male
       France Female
                         28
9999
[10000 rows x 3 columns]
y=df.iloc[:,7]
У
         2
0
1
         1
2
         8
3
         1
         2
        . .
9995
        5
        10
9996
         7
9997
9998
         3
9999
         4
Name: Tenure, Length: 10000, dtype: int64
9. Scale the independent variables
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
array([[1, 15634602, 'Hargrave', ..., 1, 101348.88, 1],
       [2, 15647311, 'Hill', ..., 1, 112542.58, 0],
       [3, 15619304, 'Onio', ..., 0, 113931.57, 1],
       [9998, 15584532, 'Liu', ..., 1, 42085.58, 1],
       [9999, 15682355, 'Sabbatini', ..., 0, 92888.52, 1],
       [10000, 15628319, 'Walker', ..., 0, 38190.78, 0]],
dtype=object)
from sklearn.preprocessing import scale
x = scale(X)
Х
names=X.columns
names
```

10. Splitting the data into Training and Testing

```
x=np.array(df['CreditScore']).reshape(-1,1)
x.shape
(10000, 1)
print(x)
[[619]
[608]
 [502]
 . . .
 [709]
 [772]
 [792]]
y.shape
(10000,)
print(y)
0
         2
1
         1
         8
3
         1
         2
9995
        5
9996
       10
9997
         7
         3
9998
9999
         4
Name: Tenure, Length: 10000, dtype: int64
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)
y_train.shape
(7000,)
y_test.shape
(3000,)
print(y_train.shape)
(7000,)
print(y test.shape)
```