

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import confusion_matrix, accuracy_score
import matplotlib.pyplot as plt
import seaborn as sns
```

```
Out[26]: dataset=pd.read_csv('C:\Users\Thaarani\Downloads\Churn_Modelling.csv')
dataset.head()
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634002	Hargrave	619	France	Male	39	5	0.00	1	1	1	101348.88	1
1	2	15647311	Hill	608	Spain	Female	41	1	83907.86	1	0	1	112542.58	0
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0

```
Out[27]: dataset.tail()
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
9995	9996	15666229	Oyjaku	771	France	Male	39	5	0.00	2	1	0	96270.64	0
9996	9997	15669802	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0
9997	9998	15684532	Liu	709	France	Female	36	7	0.00	1	0	1	42085.58	1
9998	9999	15620985	Sabbadini	772	Germany	Male	42	3	75075.31	2	1	0	93808.52	1
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0

```
Out[28]: dataset.describe()
```

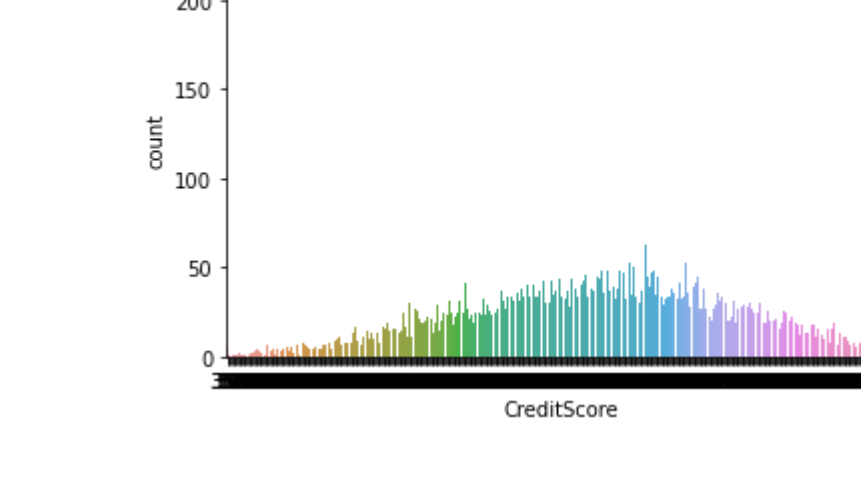
	RowNumber	CustomerId	CreditScore	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
count	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000	10000.00000
mean	5009.88568	7190316.04	65.653299	10.487806	3.892100	76495.899288	1.530000	0.70590	0.515100	100909.239861	0.207700
min	1.00000	15667067.07	350.000000	18.000000	0.000000	0.000000	1.000000	0.000000	0.000000	511.560000	0.000000
25%	2500.75000	15629367.07	652.000000	32.000000	0.000000	0.000000	1.000000	0.000000	0.000000	51002.110000	0.000000
50%	5000.50000	15609741.07	684.000000	37.000000	5.000000	87198.540000	1.000000	1.000000	1.000000	100191.910000	0.000000
75%	7500.25000	15752326.07	718.000000	44.000000	7.000000	127644.240000	2.000000	1.000000	1.000000	148394.247500	0.000000
max	10000.00000	15815696.07	850.000000	10.000000	10.000000	250898.090000	4.000000	1.000000	1.000000	199999.480000	1.000000

univariate analysis

```
Out[29]: sns.histplot(dataset['CustomerId'], kde=False)
```

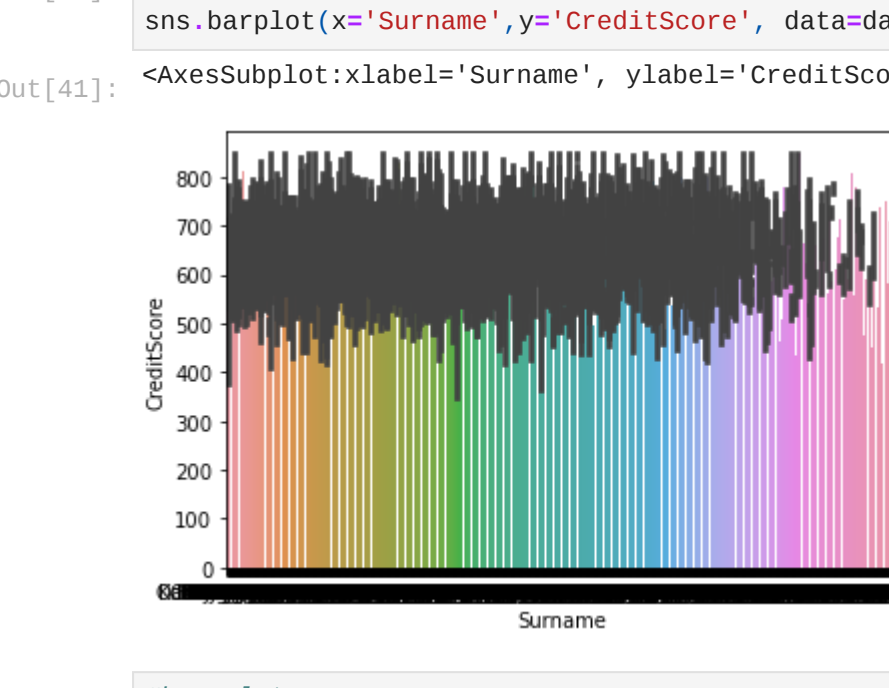
C:\Users\Thaarani\Anaconda3\lib\site-packages\seaborn\distributions.py:2129: FutureWarning: 'displot' is a deprecated function and will be removed in a future version. Please adapt your code to use either 'displot' (a figure-level function with similar flexibility) or 'histplot' (an axes-level function for histograms).

```
Out[29]: warnings.warn(msg, FutureWarning)
<AxesSubplot: xlabel='CustomerId'>
```



```
Out[30]: sns.countplot(x='CreditScore', data=dataset)
```

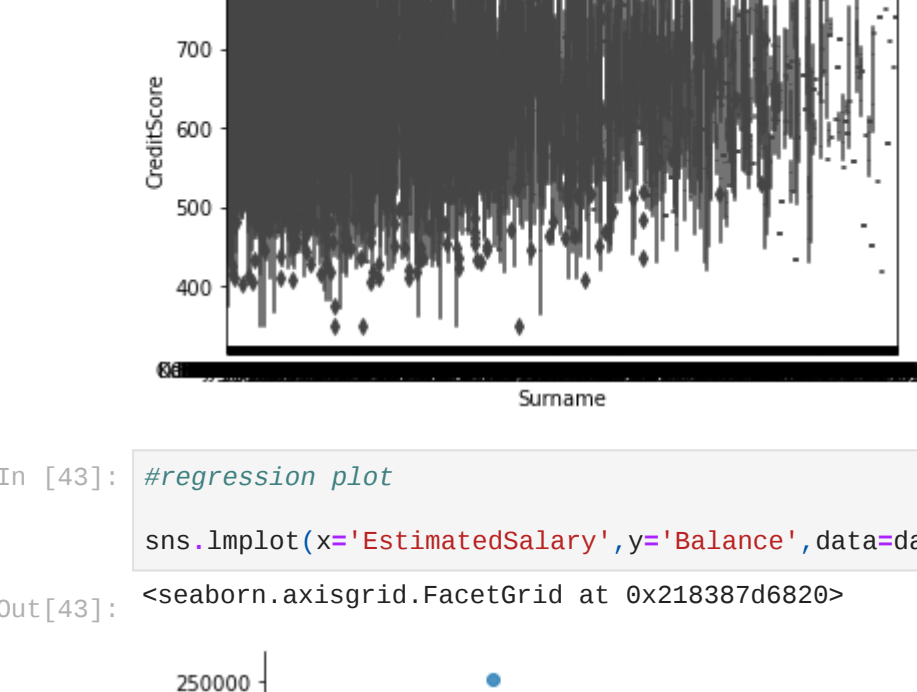
```
Out[30]: <AxesSubplot: xlabel='CreditScore', ylabel='count'>
```



bivariate analysis

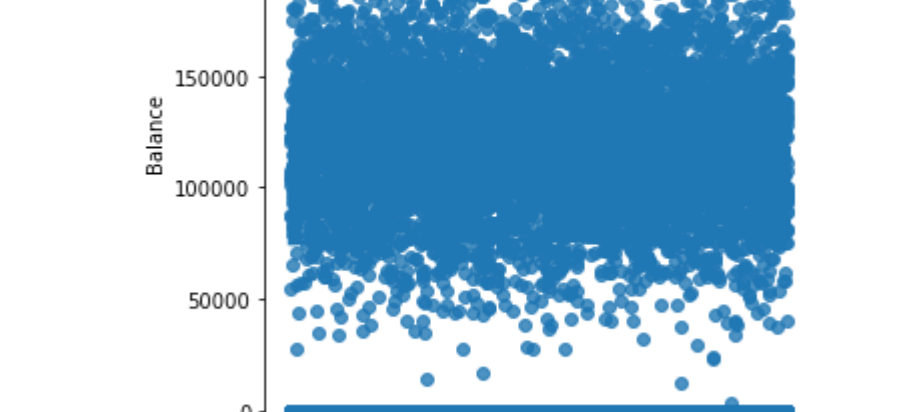
```
Out[41]: sns.barplot(x='Surname', y='CreditScore', data=dataset)
```

```
Out[41]: <AxesSubplot: xlabel='Surname', ylabel='CreditScore'>
```



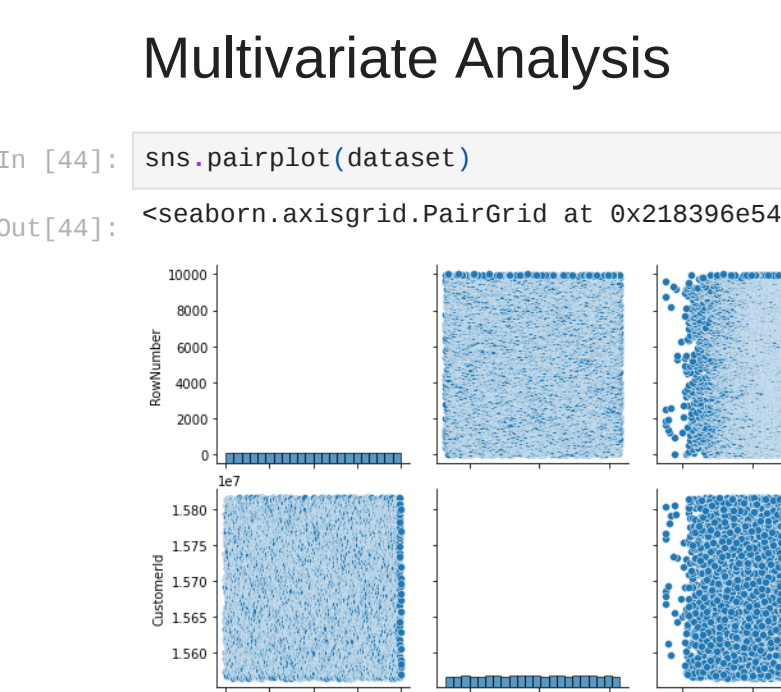
```
Out[42]: sns.boxplot(data=dataset, x='Surname', y='CreditScore')
```

```
Out[42]: <AxesSubplot: xlabel='Surname', ylabel='CreditScore'>
```



```
Out[43]: #regression plot
sns.lmplot(x='EstimatedSalary', y='Balance', data=dataset)
```

```
Out[43]: <seaborn.axisgrid.FacetGrid at 0x218387d6826>
```



Multivariate Analysis

```
Out[44]: sns.pairplot(dataset)
```

```
Out[44]: <seaborn.axisgrid.PairGrid at 0x218386e5480>
```



descriptive statistics on the dataset

```
Out[46]: dataset = pd.read_csv('C:\Users\Thaarani\Downloads\Churn_Modelling.csv')
```

```
Out[46]: dataset.head()
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
0	1	15634002	Hargrave	619	France	Male	39	5	0.00	1	1	1	101348.88	1
1	2	15647311	Hill	608	Spain	Female	41	1	83907.86	1	0	1	112542.58	0
2	3	15619304	Onio	502	France	Female	42	8	159660.80	3	1	0	113931.57	1
3	4	15701354	Boni	699	France	Female	39	1	0.00	2	0	0	93826.63	0
4	5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.10	0

```
Out[47]: dataset.tail()
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
9995	9996	15666229	Oyjaku	771	France	Male	39	5	0.00	2	1	0	96270.64	0
9996	9997	15669802	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0
9997	9998	15684532	Liu	709	France	Female	36	7	0.00	1	0	1	42085.58	1
9998	9999	15620985	Sabbadini	772	Germany	Male	42	3	75075.31	2	1	0	93808.52	1
9999	10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0

```
Out[48]: dataset.info()
```

```
Out[48]: <class 'pandas.core.frame.DataFrame'>
```

```
Out[48]: RangeIndex: 10000 entries, 0 to 9999
```

```
Out[48]: Data columns (total 14 columns):
```

```
Out[48]: # Column Non-null Count Dtype
```

```
Out[48]: # RowNumber 10000 non-null int64
```

```
Out[48]: # CustomerId 10000 non-null int64
```

```
Out[48]: # Surname 10000 non-null object
```

```
Out[48]: # CreditScore 10000 non-null int64
```

```
Out[48]: # Geography 10000 non-null object
```

```
Out[48]: # Gender 10000 non-null object
```

```
Out[48]: # Age 10000 non-null int64
```

```
Out[48]: # Tenure 10000 non-null int64
```

```
Out[48]: # Balance 10000 non-null float64
```

```
Out[48]: # NumOfProducts 10000 non-null int64
```

```
Out[48]: # HasCrCard 10000 non-null int64
```

```
Out[48]: # IsActiveMember 10000 non-null int64
```

```
Out[48]: # EstimatedSalary 10000 non-null float64
```

```
Out[48]: # Exited 10000 non-null int64
```

```
Out[48]: dtypes: float64(2), int64(1), object(3)
```

```
Out[48]: memory usage: 1.1+ MB
```

```
Out[50]: #mean
dataset.mean()
```

```
Out[50]: RowNumber 1.000000e+00
```

```
Out[50]: CustomerId 1.569994e+07
```

```
Out[50]: CreditScore 6.585288e+02
```

```
Out[50]: Age 3.882188e+01
```

```
Out[50]: Tenure 5.012806e+00
```

```
Out[50]: Balance 7.645688e+04
```

```
Out[50]: NumOfProducts 1.530208e+00
```

```
Out[50]: HasCrCard 0.705900e+00
```

```
Out[50]: IsActiveMember 0.515100e+01
```

```
Out[50]: EstimatedSalary 1.009092e+05
```

```
Out[50]: Exited 0.207700e+01
```

```
Out[50]: dtype: float64
```

```
Out[51]: #median
dataset.median()
```

```
Out[51]: RowNumber 1.000000e+00
```

```
Out[51]: CustomerId 1.569994e+07
```

```
Out[51]: CreditScore 6.520000e+02
```

```
Out[51]: Age 3.881149e+01
```

```
Out[51]: Tenure 5.009889e+00
```

```
Out[51]: Balance 7.719054e+04
```

```
Out[51]: NumOfProducts 1.000000e+00
```

```
Out[51]: HasCrCard 1.000000e+00
```

```
Out[51]: IsActiveMember 0.000000e+00
```

```
Out[51]: EstimatedSalary 1.081939e+05
```

```
Out[51]: Exited 0.000000e+00
```

```
Out[51]: dtype: float64
```

```
Out[52]: #mode
dataset.mode()
```

```
Out[52]: RowNumber 0
```

```
Out[52]: CustomerId 1
```

```
Out[52]: Surname NaN
```

```
Out[52]: CreditScore NaN
```

```
Out[52]: Geography NaN
```

```
Out[52]: Gender NaN
```

```
Out[52]: Age NaN
```

```
Out[52]: Tenure NaN
```

```
Out[52]: Balance NaN
```

```
Out[52]: NumOfProducts NaN
```

```
Out[52]: HasCrCard NaN
```

```
Out[52]: IsActiveMember NaN
```

```
Out[52]: EstimatedSalary NaN
```

```
Out[52]: Exited NaN
```

```
Out[52]: dtype: object
```

```
Out[52]: 10000 rows x 14 columns
```

handle the missing values

```
Out[53]: dataset.isna()
```

```
Out[53]: RowNumber 0
```

```
Out[53]: CustomerId 0
```

```
Out[53]: Surname 0
```

```
Out[53]: CreditScore 0
```

```
Out[53]: Geography 0
```

```
Out[53]: Gender 0
```

```
Out[53]: Age 0
```

```
Out[53]: Tenure 0
```

```
Out[53]: Balance 0
```

```
Out[53]: NumOfProducts 0
```

```
Out[53]: HasCrCard 0
```

```
Out[53]: IsActiveMember 0
```

```
Out[53]: EstimatedSalary 0
```

```
Out[53]: Exited 0
```

```
Out[53]: dtype: object
```

```
Out[53]: 10000 rows x 14 columns
```

```
Out[54]: dataset.isna().any()
```

```
Out[54]: RowNumber False
```

```
Out[54]: CustomerId False
```

```
Out[54]: Surname False
```

```
Out[54]: CreditScore False
```

```
Out[54]: Geography False
```

```
Out[54]: Gender False
```

```
Out[54]: Age False
```

```
Out[54]: Tenure False
```

```
Out[54]: Balance False
```

```
Out[54]: NumOfProducts False
```

```
Out[54]: HasCrCard False
```

```
Out[54]: IsActiveMember False
```

```
Out[54]: EstimatedSalary False
```

```
Out[54]: Exited False
```

```
Out[54]: dtype: bool
```

```
Out[55]: #skewness
dataset.skew()
```

```
Out[55]: RowNumber 0.000000
```

```
Out[55]: CustomerId 0.081149
```

```
Out[55]: CreditScore -0.071467
```

```
Out[55]: Age 0.011278
```

```
Out[55]: Tenure 0.149199
```

```
Out[55]: Balance 0.745688
```

```
Out[55]: NumOfProducts 0.000000
```

```
Out[55]: HasCrCard -0.060437
```

```
Out[55]: IsActiveMember 0.002885
```

```
Out[55]: EstimatedSalary 1.471811
```

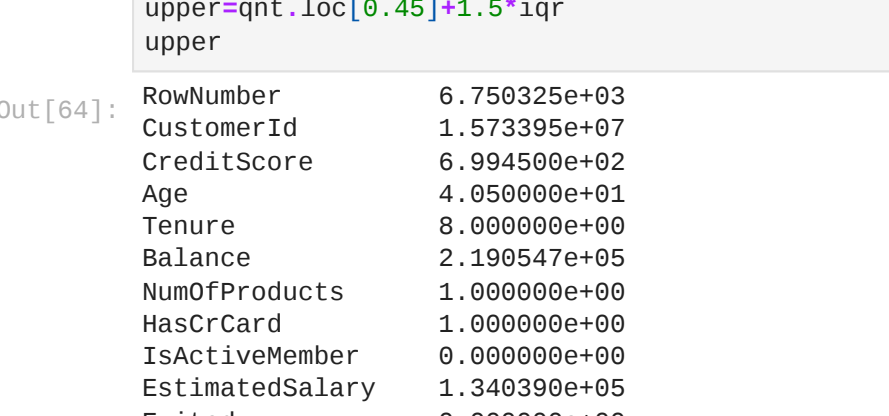
```
Out[55]: dtype: float64
```

```
Out[56]: print(sns.displot(dataset['Age']))
```

```
Out[56]: C:\Users\Thaarani\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: 'displot' 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).
```

```
Out[56]: warnings.warn(msg, FutureWarning)
```

```
Out[56]: AxesSubplot(0.125,0.125,0.775x0.755)
```



```
Out[57]: dataset.kurt()
```

```
Out[57]: C:\Users\Thaarani\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: 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.
```

```
Out[57]: dataset.kurt()
```

```
Out[57]: RowNumber 1.280610
```

```
Out[57]: CustomerId -1.196113
```

```
Out[57]: CreditScore -0.425726
```

```
Out[57]: Age -1.365476
```

```
Out[57]: Tenure -1.165225
```

```
Out[57]: Balance -1.485412
```

```
Out[57]: NumOfProducts 0.582981
```

```
Out[57]: HasCrCard -1.189973
```

```
Out[57]: IsActiveMember -1.399747
```

```
Out[57]: EstimatedSalary -1.181518
```

```
Out[57]: Exited 0.136571
```

```
Out[57]: dtype: float64
```

```
Out[58]: dataset.var()
```

```
Out[58]: C:\Users\Thaarani\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: 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.
```

```
Out[58]: dataset.var()
```

```
Out[58]: RowNumber 8.334167e+06
```

```
Out[58]: CustomerId 9.345358e+09
```

```
Out[58]: CreditScore 9.109941e+02
```

```
Out[58]: Age 0.364785e+03
```

```
Out[58]: Tenure 0.389348e+00
```