

```
In [1]: import sklearn
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [2]: bank = pd.read_csv('bank.csv')
```

```
In [3]: # Data Exploration
```

```
In [4]: bank.head()
```

```
Out[4]:
```

	RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Bal
0	1	15634602	Hargrave	619	France	Female	42	2	
1	2	15647311	Hill	608	Spain	Female	41	1	838
2	3	15619304	Onio	502	France	Female	42	8	1596
3	4	15701354	Boni	699	France	Female	39	1	
4	5	15737888	Mitchell	850	Spain	Female	43	2	1255

```
In [5]: bank.shape
```

```
Out[5]: (10000, 14)
```

```
In [6]: # check data info
bank.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
#   Column                Non-Null Count  Dtype
---  -
0   RowNumber              10000 non-null  int64
1   CustomerId             10000 non-null  int64
2   Surname                10000 non-null  object
3   CreditScore            10000 non-null  int64
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
9   NumOfProducts          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
```

```
In [6]: # check the unique values for each column
bank.nunique()
```

```
Out[6]: RowNumber          10000
CustomerId          10000
Surname              2932
CreditScore         460
Geography            3
Gender               2
Age                  70
Tenure               11
Balance              6382
NumOfProducts        4
HasCrCard             2
IsActiveMember       2
EstimatedSalary      9999
Exited               2
dtype: int64
```

```
In [7]: # check missing values
bank.isnull().sum()
```

```
Out[7]: 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
```

```
In [9]: # understand Numerical feature
        # discrete/continuous
        # 'CreditScore', 'Age', 'Tenure', 'NumberOfProducts'
        # 'Balance', 'EstimatedSalary'
        bank[['CreditScore', 'Age', 'Tenure', 'NumOfProducts', 'Balance', 'EstimatedSalary']]
```

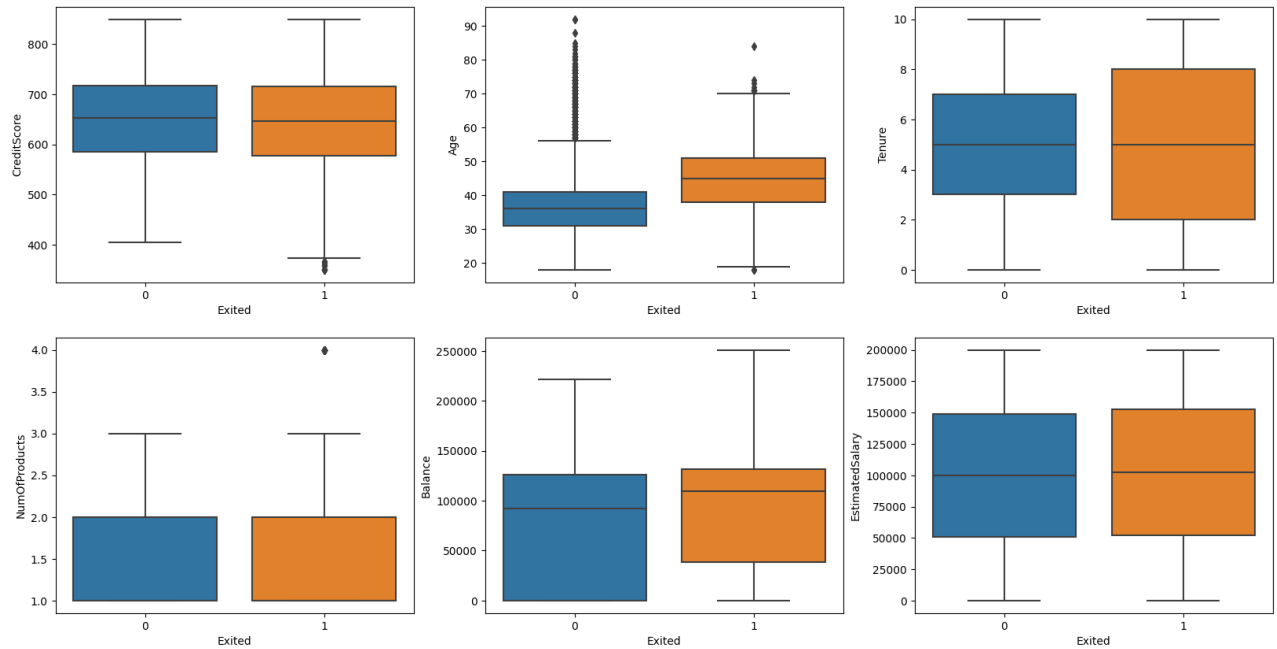
```
Out[9]:
```

	CreditScore	Age	Tenure	NumOfProducts	Balance	EstimatedSalary
<b>count</b>	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
<b>mean</b>	650.528800	38.921800	5.012800	1.530200	76485.889288	10009.599500
<b>std</b>	96.653299	10.487806	2.892174	0.581654	62397.405202	5751.557700
<b>min</b>	350.000000	18.000000	0.000000	1.000000	0.000000	1000.000000
<b>25%</b>	584.000000	32.000000	3.000000	1.000000	0.000000	5100.000000
<b>50%</b>	652.000000	37.000000	5.000000	1.000000	97198.540000	10019.599500
<b>75%</b>	718.000000	44.000000	7.000000	2.000000	127644.240000	14938.599500
<b>max</b>	850.000000	92.000000	10.000000	4.000000	250898.090000	19999.599500

```
In [10]: import matplotlib.pyplot as plt
         import seaborn as sns
```

```
In [11]: # boxplot for numerical feature
_, axss = plt.subplots(2, 3, figsize=[20, 10])
sns.boxplot(x='Exited', y='CreditScore', data=bank, ax=axss[0][0])
sns.boxplot(x='Exited', y='Age', data=bank, ax=axss[0][1])
sns.boxplot(x='Exited', y='Tenure', data=bank, ax=axss[0][2])
sns.boxplot(x='Exited', y='NumOfProducts', data=bank, ax=axss[1][0])
sns.boxplot(x='Exited', y='Balance', data=bank, ax=axss[1][1])
sns.boxplot(x='Exited', y='EstimatedSalary', data=bank, ax=axss[1][2])
```

Out[11]: <AxesSubplot: xlabel='Exited', ylabel='EstimatedSalary'>



```
In [12]: # understand categorical feature
# 'Geography', 'Gender'
# 'HasCrCard', 'IsActiveMember'
_,axss = plt.subplots(2,2, figsize=[20,10])
sns.countplot(x='Exited', hue='Geography', data=bank, ax=axss[0][0])
sns.countplot(x='Exited', hue='Gender', data=bank, ax=axss[0][1])
sns.countplot(x='Exited', hue='HasCrCard', data=bank, ax=axss[1][0])
sns.countplot(x='Exited', hue='IsActiveMember', data=bank, ax=axss[1][1])
```

Out[12]: <AxesSubplot: xlabel='Exited', ylabel='count'>

