Exploratory Data Analysis (EDA)

Written by: M.Danish Azeem Date: 08.12.2023 Email: danishazeem365@gmail.com

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

1- import dataset

```
df = sns.load_dataset("titanic")
df
```

```
     survived  pclass     sex   age  sibsp  parch     fare embarked
class  \
0           0       3    male  22.0      1      0   7.2500        S
Third
1           1       1  female  38.0      1      0  71.2833        C
First
2           1       3  female  26.0      0      0   7.9250        S
Third
3           1       1  female  35.0      1      0  53.1000        S
First
4           0       3    male  35.0      0      0   8.0500        S
Third
..        ...     ...     ...   ...    ...    ...      ...      ...
...
886         0       2    male  27.0      0      0  13.0000        S
Second
887         1       1  female  19.0      0      0  30.0000        S
First
888         0       3  female   NaN      1      2  23.4500        S
Third
889         1       1    male  26.0      0      0  30.0000        C
First
890         0       3    male  32.0      0      0   7.7500        Q
Third

       who  adult_male deck  embark_town alive  alone
0      man        True  NaN  Southampton    no  False
1    woman       False    C    Cherbourg   yes  False
2    woman       False  NaN  Southampton   yes   True
3    woman       False    C  Southampton   yes  False
```

```
4      man        True  NaN  Southampton    no   True
..     ...         ...  ...          ...   ...    ...
886    man        True  NaN  Southampton    no   True
887  woman       False    B  Southampton   yes   True
888  woman       False  NaN  Southampton    no  False
889    man        True    C    Cherbourg   yes   True
890    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

1- Bigger picture of data

```
df.dtypes
```

```
survived          int64
pclass            int64
sex              object
age             float64
sibsp             int64
parch             int64
fare            float64
embarked         object
class          category
who              object
adult_male         bool
deck           category
embark_town      object
alive            object
alone              bool
dtype: object
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     891 non-null    int64
 1   pclass       891 non-null    int64
 2   sex          891 non-null    object
 3   age          714 non-null    float64
 4   sibsp        891 non-null    int64
 5   parch        891 non-null    int64
 6   fare         891 non-null    float64
 7   embarked     889 non-null    object
 8   class        891 non-null    category
 9   who          891 non-null    object
 10  adult_male   891 non-null    bool
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object
 13  alive        891 non-null    object
 14  alone        891 non-null    bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

```
df.isnull().sum()   # finding missing and null values
```

```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

```
df2 = df
```

```
df2
```

```
     survived  pclass     sex   age  sibsp  parch     fare embarked
class  \
0           0       3    male  22.0      1      0   7.2500        S
Third
1           1       1  female  38.0      1      0  71.2833        C
First
2           1       3  female  26.0      0      0   7.9250        S
Third
3           1       1  female  35.0      1      0  53.1000        S
First
4           0       3    male  35.0      0      0   8.0500        S
Third
..        ...     ...     ...   ...    ...    ...      ...      ...
...
886         0       2    male  27.0      0      0  13.0000        S
Second
887         1       1  female  19.0      0      0  30.0000        S
First
888         0       3  female   NaN      1      2  23.4500        S
```

```
Third
889         1       1    male  26.0         0      0  30.0000        C
First
890         0       3    male  32.0         0      0   7.7500        Q
Third

       who  adult_male deck  embark_town alive  alone
0      man        True  NaN  Southampton    no  False
1    woman       False    C    Cherbourg   yes  False
2    woman       False  NaN  Southampton   yes   True
3    woman       False    C  Southampton   yes  False
4      man        True  NaN  Southampton    no   True
..     ...         ...  ...          ...   ...    ...
886    man        True  NaN  Southampton    no   True
887  woman       False    B  Southampton   yes   True
888  woman       False  NaN  Southampton    no  False
889    man        True    C    Cherbourg   yes   True
890    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

```
df2.isnull().sum() / len(df2) * 100
```

```
survived        0.000000
pclass          0.000000
sex             0.000000
age            19.865320
sibsp           0.000000
parch           0.000000
fare            0.000000
embarked        0.224467
class           0.000000
who             0.000000
adult_male      0.000000
deck           77.216611
embark_town     0.224467
alive           0.000000
alone           0.000000
dtype: float64
```

#1 Assighnment : How to deal with missing values in EDA Analysis. like categorical/object, numeric(int,float) , boolean etc

```
print(df2["age"].max())
print(df2["age"].min())
print(df2["age"].mean())
```

```
80.0
0.42
29.69911764705882
```

```
df2['age'].unique()
```

```
array([22.  , 38.  , 26.  , 35.  ,   nan, 54.  ,  2.  , 27.  , 14.  ,
        4.  , 58.  , 20.  , 39.  , 55.  , 31.  , 34.  , 15.  , 28.  ,
        8.  , 19.  , 40.  , 66.  , 42.  , 21.  , 18.  ,  3.  ,  7.  ,
       49.  , 29.  , 65.  , 28.5 ,  5.  , 11.  , 45.  , 17.  , 32.  ,
       16.  , 25.  ,  0.83, 30.  , 33.  , 23.  , 24.  , 46.  , 59.  ,
       71.  , 37.  , 47.  , 14.5 , 70.5 , 32.5 , 12.  ,  9.  , 36.5 ,
       51.  , 55.5 , 40.5 , 44.  ,  1.  , 61.  , 56.  , 50.  , 36.  ,
       45.5 , 20.5 , 62.  , 41.  , 52.  , 63.  , 23.5 ,  0.92, 43.  ,
       60.  , 10.  , 64.  , 13.  , 48.  ,  0.75, 53.  , 57.  , 80.  ,
       70.  , 24.5 ,  6.  ,  0.67, 30.5 ,  0.42, 34.5 , 74.  ])
```

```
df2['age'].fillna(df['age'].mean(), inplace=True)
```

```
df2
```

```
     survived  pclass     sex        age  sibsp  parch     fare
embarked  \
0           0       3    male  22.000000      1      0   7.2500
S
1           1       1  female  38.000000      1      0  71.2833
C
2           1       3  female  26.000000      0      0   7.9250
S
3           1       1  female  35.000000      1      0  53.1000
S
4           0       3    male  35.000000      0      0   8.0500
S
..        ...     ...     ...        ...    ...    ...      ...       .
..
886         0       2    male  27.000000      0      0  13.0000
S
887         1       1  female  19.000000      0      0  30.0000
S
888         0       3  female  29.699118      1      2  23.4500
S
889         1       1    male  26.000000      0      0  30.0000
C
890         0       3    male  32.000000      0      0   7.7500
Q

      class    who  adult_male deck  embark_town alive  alone
0     Third    man        True  NaN  Southampton    no  False
1     First  woman       False    C    Cherbourg   yes  False
2     Third  woman       False  NaN  Southampton   yes   True
3     First  woman       False    C  Southampton   yes  False
4     Third    man        True  NaN  Southampton    no   True
..      ...    ...         ...  ...          ...   ...    ...
886  Second    man        True  NaN  Southampton    no   True
```

```
887   First  woman       False    B  Southampton   yes   True
888   Third  woman       False  NaN  Southampton    no  False
889   First    man        True    C    Cherbourg   yes   True
890   Third    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

```
df2.isnull().sum()
```

```
survived         0
pclass           0
sex              0
age              0
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

```
df2
```

```
     survived  pclass      sex         age  sibsp  parch      fare
embarked  \
0           0       3     male  22.000000      1      0    7.2500
S
1           1       1   female  38.000000      1      0   71.2833
C
2           1       3   female  26.000000      0      0    7.9250
S
3           1       1   female  35.000000      1      0   53.1000
S
4           0       3     male  35.000000      0      0    8.0500
S
..        ...     ...     ...        ...    ...    ...      ...      .
..
886         0       2     male  27.000000      0      0   13.0000
S
887         1       1   female  19.000000      0      0   30.0000
S
888         0       3   female  29.699118      1      2   23.4500
S
889         1       1     male  26.000000      0      0   30.0000
C
```

```
890         0       3    male  32.000000      0      0   7.7500
Q

      class    who  adult_male deck  embark_town alive  alone
0     Third    man        True  NaN  Southampton    no  False
1     First  woman       False    C    Cherbourg   yes  False
2     Third  woman       False  NaN  Southampton   yes   True
3     First  woman       False    C  Southampton   yes  False
4     Third    man        True  NaN  Southampton    no   True
..      ...    ...         ...  ...          ...   ...    ...
886  Second    man        True  NaN  Southampton    no   True
887   First  woman       False    B  Southampton   yes   True
888   Third  woman       False  NaN  Southampton    no  False
889   First    man        True    C    Cherbourg   yes   True
890   Third    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

```
df3 = df2
```

```
df3
```

```
     survived  pclass     sex        age  sibsp  parch     fare
embarked  \
0           0       3    male  22.000000      1      0   7.2500
S
1           1       1  female  38.000000      1      0  71.2833
C
2           1       3  female  26.000000      0      0   7.9250
S
3           1       1  female  35.000000      1      0  53.1000
S
4           0       3    male  35.000000      0      0   8.0500
S
..        ...     ...     ...        ...    ...    ...      ...       .
..
886         0       2    male  27.000000      0      0  13.0000
S
887         1       1  female  19.000000      0      0  30.0000
S
888         0       3  female  29.699118      1      2  23.4500
S
889         1       1    male  26.000000      0      0  30.0000
C
890         0       3    male  32.000000      0      0   7.7500
Q

      class    who  adult_male deck  embark_town alive  alone
0     Third    man        True  NaN  Southampton    no  False
1     First  woman       False    C    Cherbourg   yes  False
```

```
2     Third  woman       False  NaN  Southampton   yes   True
3     First  woman       False    C  Southampton   yes  False
4     Third    man        True  NaN  Southampton    no   True
..      ...    ...         ...  ...          ...   ...    ...
886  Second    man        True  NaN  Southampton    no   True
887   First  woman       False    B  Southampton   yes   True
888   Third  woman       False  NaN  Southampton    no  False
889   First    man        True    C    Cherbourg   yes   True
890   Third    man        True  NaN   Queenstown    no   True

[891 rows x 15 columns]
```

```
df3.isnull().sum()
```

```
survived         0
pclass           0
sex              0
age              0
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

```
df3 = df3.drop(columns=["deck"])
```

```
df3.isnull().sum()
```

```
survived       0
pclass         0
sex            0
age            0
sibsp          0
parch          0
fare           0
embarked       2
class          0
who            0
adult_male     0
embark_town    2
alive          0
alone          0
dtype: int64
```

```
df4 = df3

df4["embarked"].fillna(df["embarked"].mode()[0], inplace=True)
df4["embark_town"].fillna(df["embark_town"].mode()[0], inplace=True)

df4.isnull().sum()
```

```
survived       0
pclass         0
sex            0
age            0
sibsp          0
parch          0
fare           0
embarked       0
class          0
who            0
adult_male     0
embark_town    0
alive          0
alone          0
dtype: int64
```

```
print("mean = ",df['age'].mean())
print("median = ",df['age'].median())
print("mode = ",df['embarked'].mode())
```

```
mean =  29.69911764705882
median =  29.69911764705882
mode =  0    S
Name: embarked, dtype: object
```

steps :Data wranglings (EDA)

1. Import labireses
2. Import database
3. Explore your data
    a. information
    b. Datatype
    c. Missing values
    d. Take sence of your data
4. Understanding the variables
5. Relationship between the variables Analysts (heatmap, pairplot, correlation)
6. Brainstorming
    a. Normalize (Technics # asignment)
    b. Removing outliers # Asighnment
7. Tidy data, clean data

8. Ready for statstitical Analystis
9. Ready for Predection
10. Ready for machin learning
11. Ready for DL.