Exploratory Data Analysis (EDA)

November 1, 2023

1 Exploratory Data Analysis (EDA)

EDA

1.0.1 Importing the basics libraries

[2]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

1.0.2 Importing dataset

[6]:
```
df = pd.read_csv("Titanic-Dataset.csv")

print("Data shape :", df.shape)
```

```
Data shape : (891, 12)
```

1.1 Exploratory data analysis (EDA)

1.1.1 Data info

[18]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
```

```
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

1.1.2 Missing values

[19]:
```
df.isnull().sum()
```

[19]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

[20]:
```
import missingno as msno
%matplotlib inline
```

[21]:
```
msno.bar(df)
plt.title("Missing values by features", fontsize = 22, pad = 25)
```

[21]:
```
Text(0.5, 1.0, 'Missing values by features')
```

1.1.3 Descriptive statistics

[22]:
```
df.describe()
```

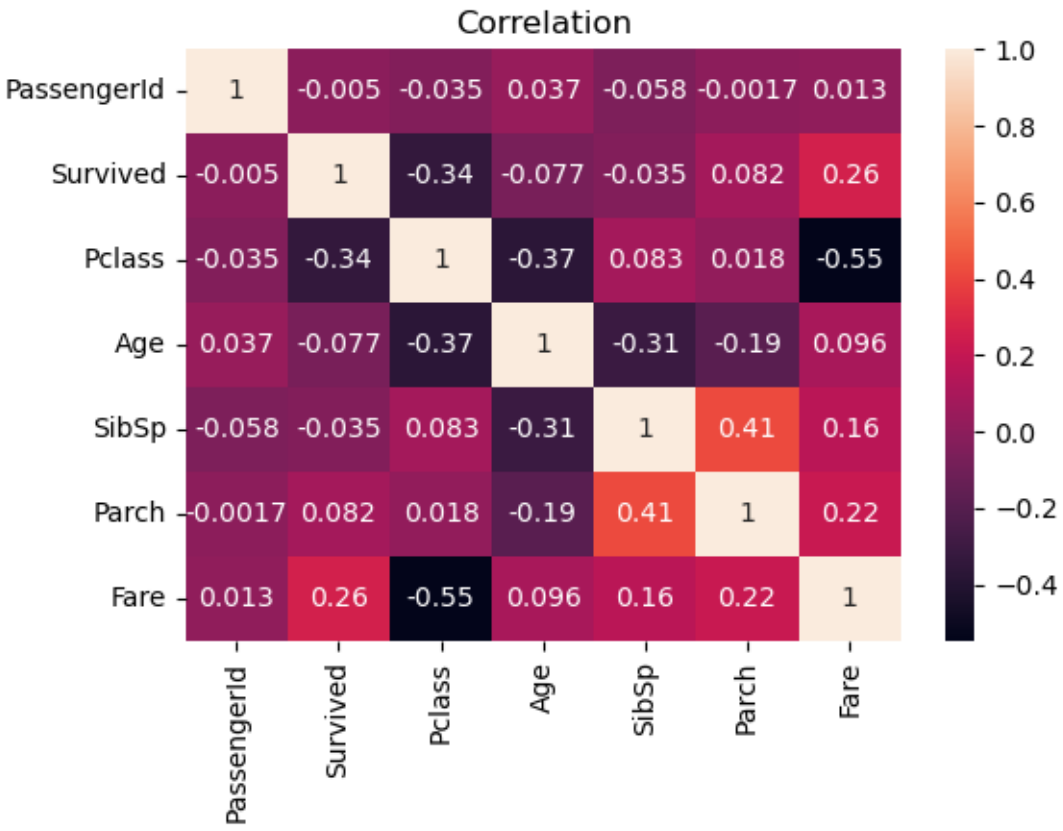
[22]:
```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
```

1.1.4 Correlation

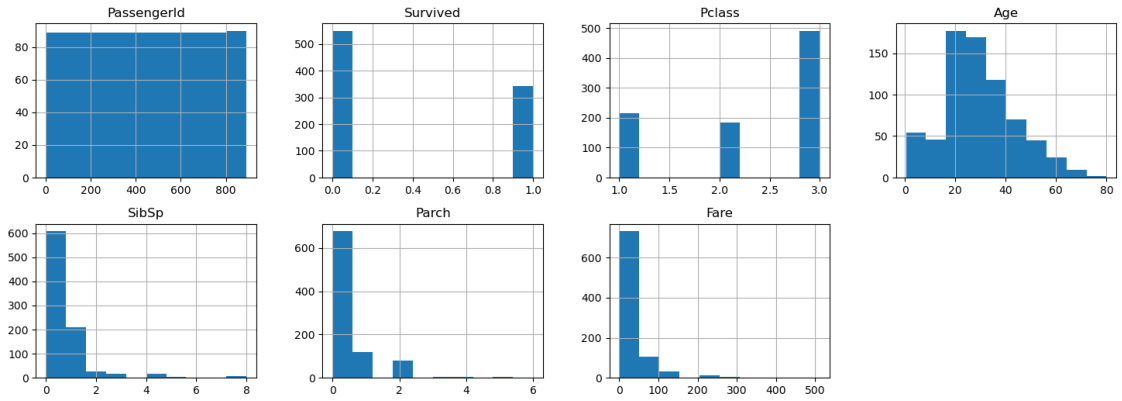
[23]:
```
plt.figure(figsize=(6, 4))
sns.heatmap(df.select_dtypes(include = np.number).corr(), annot = True)
plt.title("Correlation")
```

[23]:
```
Text(0.5, 1.0, 'Correlation')
```



1.1.5 Plot numerical values

[25]:
```
df.hist(layout=(5,4), figsize=(18,16))
plt.show()
```



1.1.6 Count plot

[53]:
```
import seaborn as sns
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt


fig, axes = plt.subplots(2, 3, figsize=(18, 10))

fig.suptitle('Plot the number of categories', fontsize = 18)

sns.countplot(ax=axes[0, 0], data=df, x='Sex')
sns.countplot(ax=axes[0, 1], data=df, x='Embarked')
sns.countplot(ax=axes[0, 2], data=df, x='Survived')
sns.countplot(ax=axes[1, 0], data=df, x='Pclass')
sns.countplot(ax=axes[1, 1], data=df, x='SibSp')
sns.countplot(ax=axes[1, 2], data=df, x='Cabin')
```

[53]:
```
<Axes: xlabel='Cabin', ylabel='count'>
```