pi-ds-task02

July 13, 2024

[ ]:
```
import pandas as pd
import numpy as np
```

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

1 Reading the dataset

[ ]:
```
data= pd.read_csv('tested.csv');
```

2 Converting to dataframe

[ ]:
```
df= pd.DataFrame(data)
```

[ ]:
```
print(df.head)   #returns first 5 rows of the dataframe
```

```
<bound method NDFrame.head of      PassengerId  Survived  Pclass  \
0            892         0       3
1            893         1       3
2            894         0       2
3            895         0       3
4            896         1       3
..           …       …     …
413         1305         0       3
414         1306         1       1
415         1307         0       3
416         1308         0       3
417         1309         0       3

                                          Name     Sex   Age  SibSp  Parch  \
0                              Kelly, Mr. James    male  34.5      0      0
1              Wilkes, Mrs. James (Ellen Needs)  female  47.0      1      0
2                     Myles, Mr. Thomas Francis    male  62.0      0      0
3                              Wirz, Mr. Albert    male  27.0      0      0
4    Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female  22.0      1      1
..                                           …     …   …    …    …
```

```
413                            Spector, Mr. Woolf    male   NaN      0      0
414                  Oliva y Ocana, Dona. Fermina  female  39.0      0      0
415                  Saether, Mr. Simon Sivertsen    male  38.5      0      0
416                           Ware, Mr. Frederick    male   NaN      0      0
417                      Peter, Master. Michael J    male   NaN      1      1

                 Ticket      Fare Cabin Embarked
0                330911    7.8292   NaN        Q
1                363272    7.0000   NaN        S
2                240276    9.6875   NaN        Q
3                315154    8.6625   NaN        S
4               3101298   12.2875   NaN        S
..                  …       …   …      …
413           A.5. 3236    8.0500   NaN        S
414            PC 17758  108.9000  C105        C
415  SOTON/O.Q. 3101262    7.2500   NaN        S
416              359309    8.0500   NaN        S
417                2668   22.3583   NaN        C

[418 rows x 12 columns]>
```

[ ]:
```
df.shape  #displays the no.of obs and features
```

[ ]:
```
(418, 12)
```

[ ]:
```
df.tail()  #displays the last 5 rows of the dataframe
```

[ ]:
```
     PassengerId  Survived  Pclass                            Name     Sex  \
413         1305         0       3              Spector, Mr. Woolf    male
414         1306         1       1    Oliva y Ocana, Dona. Fermina  female
415         1307         0       3    Saether, Mr. Simon Sivertsen    male
416         1308         0       3             Ware, Mr. Frederick    male
417         1309         0       3        Peter, Master. Michael J    male

      Age  SibSp  Parch              Ticket      Fare Cabin Embarked
413   NaN      0      0           A.5. 3236    8.0500   NaN        S
414  39.0      0      0            PC 17758  108.9000  C105        C
415  38.5      0      0  SOTON/O.Q. 3101262    7.2500   NaN        S
416   NaN      0      0              359309    8.0500   NaN        S
417   NaN      1      1                2668   22.3583   NaN        C
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
```

```
 0   PassengerId  418 non-null    int64
 1   Survived     418 non-null    int64
 2   Pclass       418 non-null    int64
 3   Name         418 non-null    object
 4   Sex          418 non-null    object
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64
 7   Parch        418 non-null    int64
 8   Ticket       418 non-null    object
 9   Fare         417 non-null    float64
 10  Cabin        91 non-null     object
 11  Embarked     418 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 39.3+ KB
```

[ ]:
```
df.isnull().sum()  #will return the no.of missing records in each column
```

[ ]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin          327
Embarked         0
dtype: int64
```

[ ]:
```
df.fillna(0)  #fills missing value/null values with 0
```

[ ]:
```
     PassengerId  Survived  Pclass  \
0            892         0       3
1            893         1       3
2            894         0       2
3            895         0       3
4            896         1       3
..           …       …     …
413         1305         0       3
414         1306         1       1
415         1307         0       3
416         1308         0       3
417         1309         0       3

                                          Name     Sex   Age  SibSp  Parch  \
```

```
0                                Kelly, Mr. James    male  34.5      0      0
1                Wilkes, Mrs. James (Ellen Needs)  female  47.0      1      0
2                       Myles, Mr. Thomas Francis    male  62.0      0      0
3                                Wirz, Mr. Albert    male  27.0      0      0
4    Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female  22.0      1      1
..                                            …     …   …    …    …
413                           Spector, Mr. Woolf    male   0.0      0      0
414                 Oliva y Ocana, Dona. Fermina  female  39.0      0      0
415                 Saether, Mr. Simon Sivertsen    male  38.5      0      0
416                         Ware, Mr. Frederick    male   0.0      0      0
417                   Peter, Master. Michael J    male   0.0      1      1

                 Ticket      Fare Cabin Embarked
0                330911    7.8292     0        Q
1                363272    7.0000     0        S
2                240276    9.6875     0        Q
3                315154    8.6625     0        S
4               3101298   12.2875     0        S
..                  …       …   …      …
413          A.5. 3236    8.0500     0        S
414           PC 17758  108.9000  C105        C
415  SOTON/O.Q. 3101262    7.2500     0        S
416              359309    8.0500     0        S
417                2668   22.3583     0        C

[418 rows x 12 columns]
```

3 Summary Statistics

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

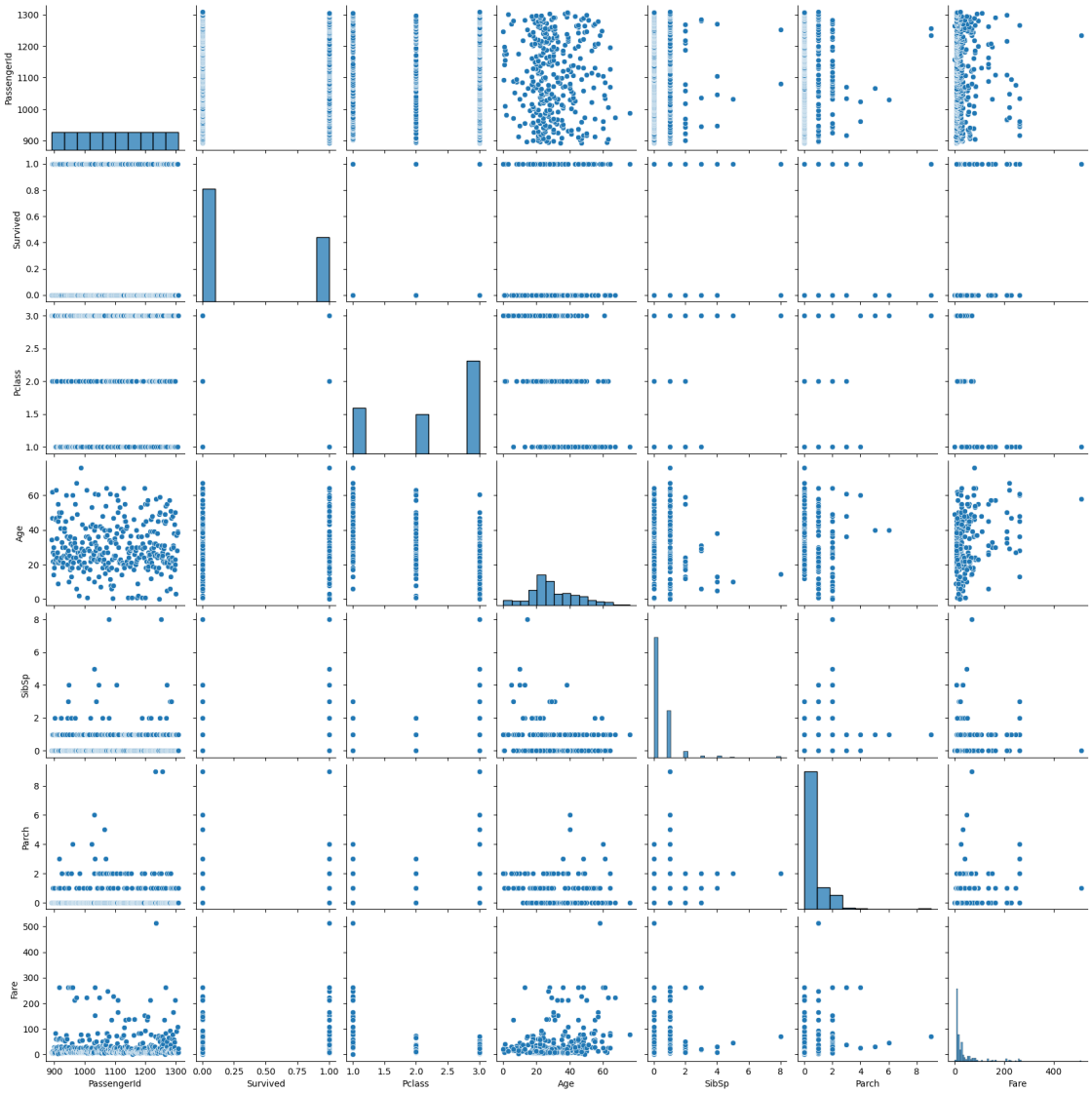
[ ]:
```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   418.000000  418.000000  418.000000  332.000000  418.000000
mean   1100.500000    0.363636    2.265550   30.272590    0.447368
std     120.810458    0.481622    0.841838   14.181209    0.896760
min     892.000000    0.000000    1.000000    0.170000    0.000000
25%     996.250000    0.000000    1.000000   21.000000    0.000000
50%    1100.500000    0.000000    3.000000   27.000000    0.000000
75%    1204.750000    1.000000    3.000000   39.000000    1.000000
max    1309.000000    1.000000    3.000000   76.000000    8.000000

            Parch        Fare
count  418.000000  417.000000
mean     0.392344   35.627188
std      0.981429   55.907576
min      0.000000    0.000000
```

```
25%       0.000000     7.895800
50%       0.000000    14.454200
75%       0.000000    31.500000
max       9.000000   512.329200
```

3.1 Pair Plot: showing relationship between two categorical values

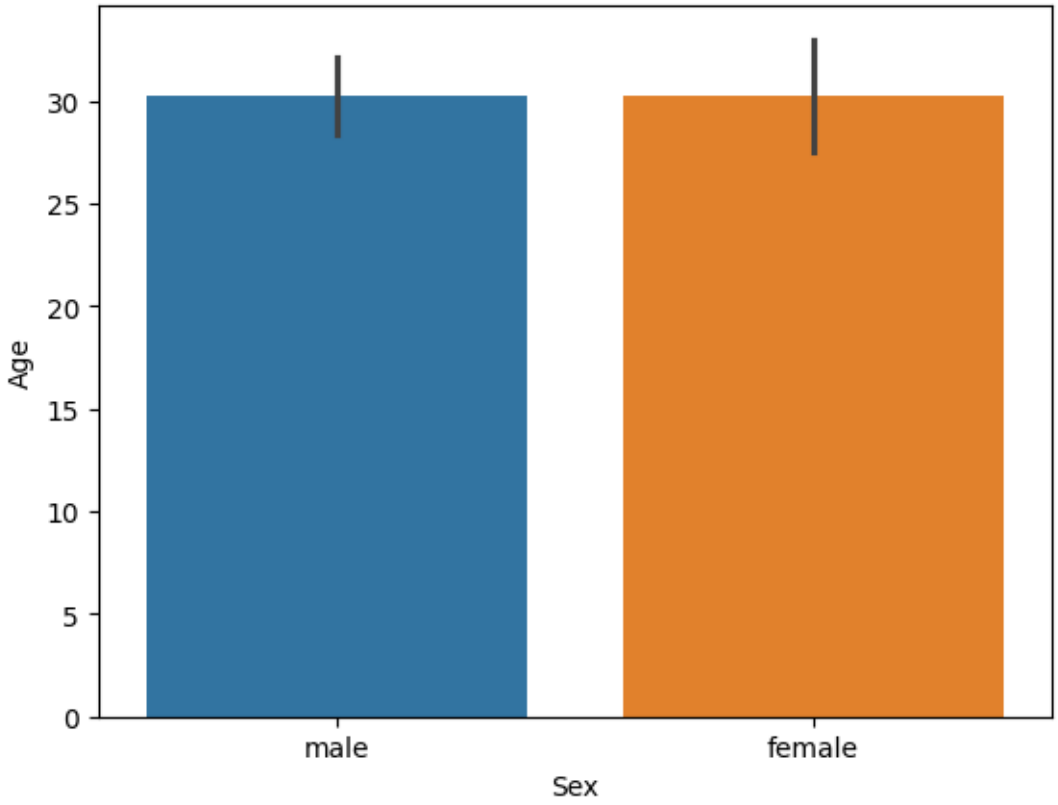
[ ]:
```
plt.figure(figsize=(13,17))
sns.pairplot(data=data.drop(['Sex','Embarked'],axis=1))
plt.show()
```

```
<Figure size 1300x1700 with 0 Axes>
```



3.2 Bar Plot: showing relationship between categorical variables and continous variables

[ ]:
```
sns.barplot(x='Sex',y='Age',data=df, hue='Sex')
plt.show()
```



3.3 Heatmap: showing correlation between variables

[ ]:
```
df1=df.drop(['Name','Cabin','Sex', 'Ticket','Embarked'],axis=1)
```

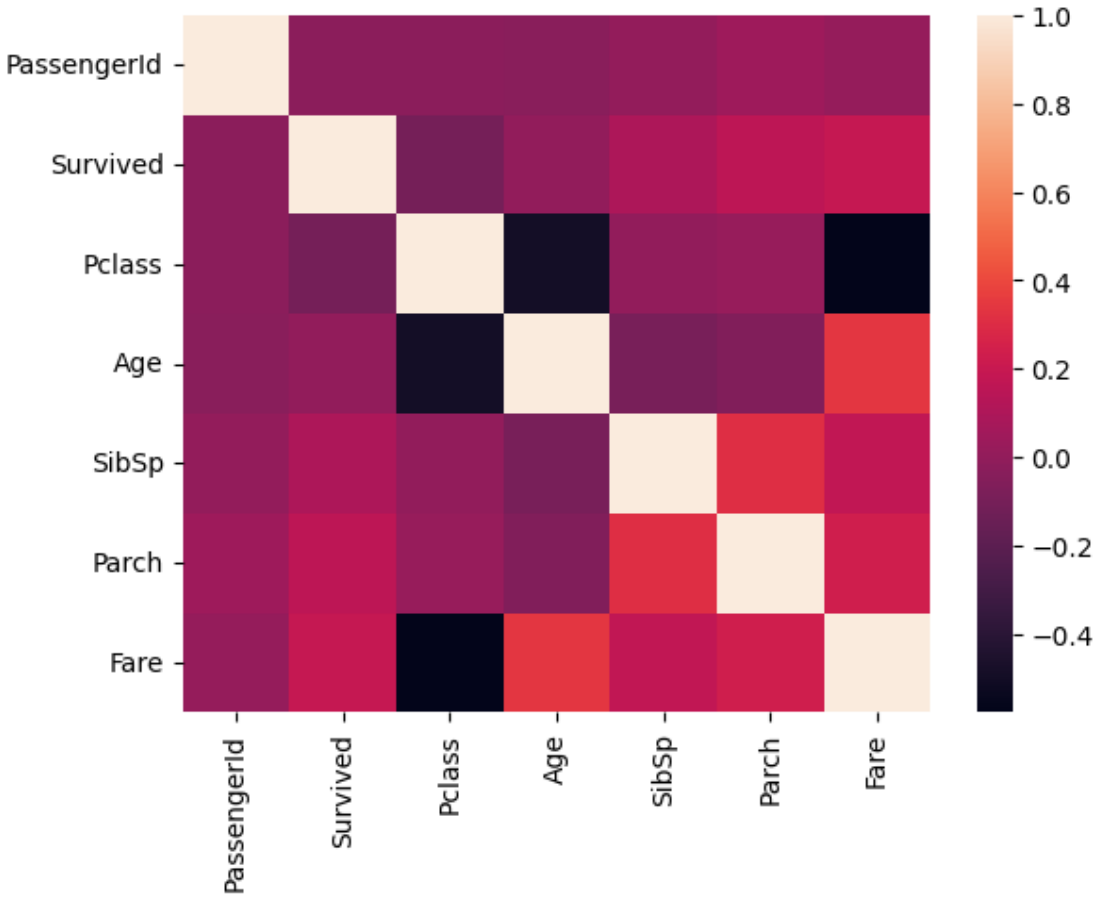
[ ]:
```
df1
```

[ ]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
0	892	0	3	34.5	0	0	7.8292
1	893	1	3	47.0	1	0	7.0000
2	894	0	2	62.0	0	0	9.6875
3	895	0	3	27.0	0	0	8.6625
4	896	1	3	22.0	1	1	12.2875
..	…	…	…	…	…	…	…
413	1305	0	3	NaN	0	0	8.0500
414	1306	1	1	39.0	0	0	108.9000

```
415          1307          0        3  38.5       0       0    7.2500
416          1308          0        3   NaN       0       0    8.0500
417          1309          0        3   NaN       1       1   22.3583

[418 rows x 7 columns]
```

[ ]:
```
sns.heatmap(df1.corr())
plt.show()
```



[ ]:
```

```