assignment3

September 21, 2023

1 Assignment 15 sep

1.1 Data Preprocessing

1.1.1 *1.Import the Libraries.*

[998]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

1.1.2 *2.import the dataset*

[999]:
```
df=pd.read_csv("Titanic-Dataset.csv")
df
```

[999]:

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	
..	…	…	…	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

	Name	Sex	Age	SibSp	\
0	Braund, Mr. Owen Harris	male	22.0	1	
1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	
2	Heikkinen, Miss. Laina	female	26.0	0	
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	
4	Allen, Mr. William Henry	male	35.0	0	
..	…	…	…	…	
886	Montvila, Rev. Juozas	male	27.0	0	
887	Graham, Miss. Margaret Edith	female	19.0	0	

```
888          Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                             Behr, Mr. Karl Howell    male  26.0      0
890                               Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked
0        0         A/5 21171   7.2500   NaN        S
1        0          PC 17599  71.2833   C85        C
2        0  STON/O2. 3101282   7.9250   NaN        S
3        0            113803  53.1000  C123        S
4        0            373450   8.0500   NaN        S
..     …                …       …   …      …
886      0            211536  13.0000   NaN        S
887      0            112053  30.0000   B42        S
888      2        W./C. 6607  23.4500   NaN        S
889      0            111369  30.0000  C148        C
890      0            370376   7.7500   NaN        Q

[891 rows x 12 columns]
```

[1000]:
```
df.head()
```

[1000]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S
```

[1001]:
```
df.tail()
```

[1001]:
```
     PassengerId  Survived  Pclass                                      Name  \
886          887         0       2                     Montvila, Rev. Juozas
887          888         1       1              Graham, Miss. Margaret Edith
```

```
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1                     Behr, Mr. Karl Howell
890          891         0       3                       Dooley, Mr. Patrick

        Sex   Age  SibSp  Parch      Ticket   Fare Cabin Embarked
886    male  27.0      0      0      211536  13.00   NaN        S
887  female  19.0      0      0      112053  30.00   B42        S
888  female   NaN      1      2  W./C. 6607  23.45   NaN        S
889    male  26.0      0      0      111369  30.00  C148        C
890    male  32.0      0      0      370376   7.75   NaN        Q
```

[1002]:
```
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
```

[1003]:
```
df.shape
```

[1003]:
```
(891, 12)
```

[1004]:
```
df.describe()
```

[1004]:
```
       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
```

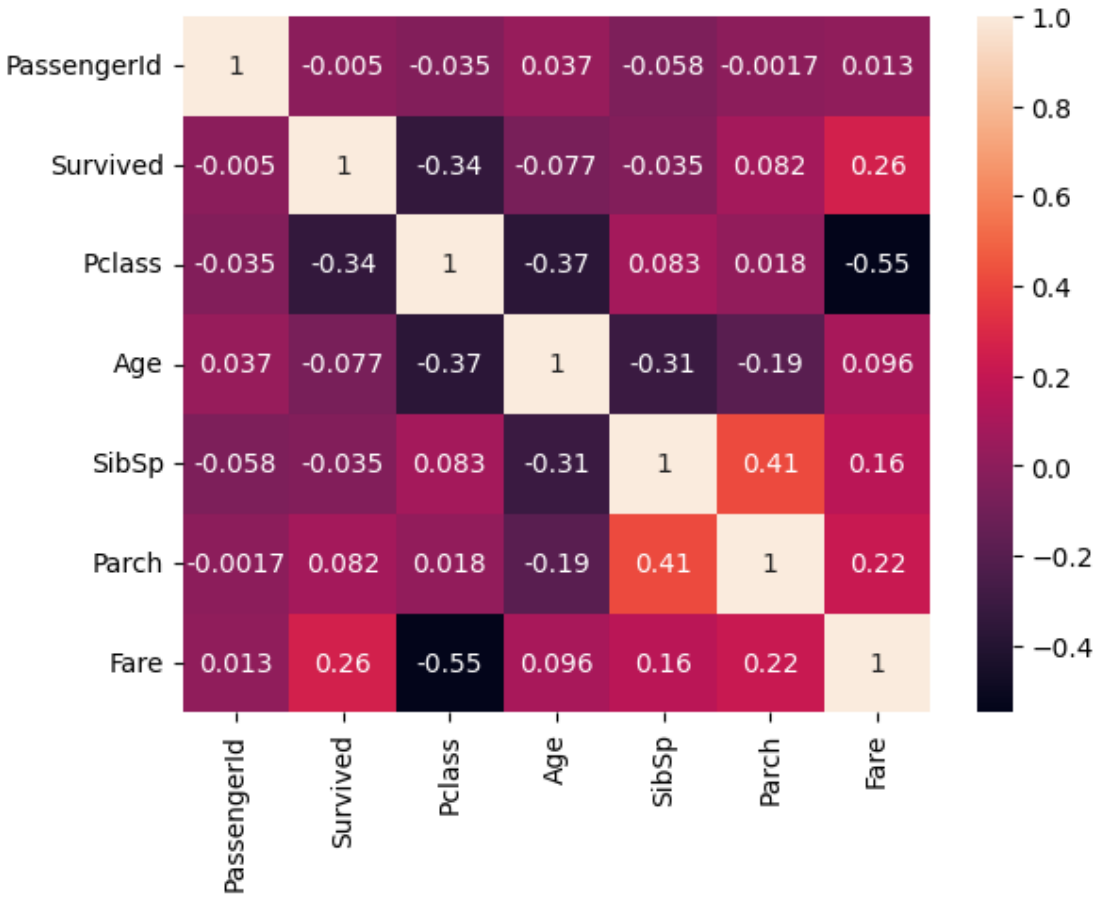
[1005]:
```
corr=df.corr()
corr
```

```
<ipython-input-1005-7d5195e2bf4d>:1: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  corr=df.corr()
```

[1005]:
```
             PassengerId  Survived    Pclass       Age     SibSp     Parch  \
PassengerId     1.000000 -0.005007 -0.035144  0.036847 -0.057527 -0.001652
Survived       -0.005007  1.000000 -0.338481 -0.077221 -0.035322  0.081629
Pclass         -0.035144 -0.338481  1.000000 -0.369226  0.083081  0.018443
Age             0.036847 -0.077221 -0.369226  1.000000 -0.308247 -0.189119
SibSp          -0.057527 -0.035322  0.083081 -0.308247  1.000000  0.414838
Parch          -0.001652  0.081629  0.018443 -0.189119  0.414838  1.000000
Fare            0.012658  0.257307 -0.549500  0.096067  0.159651  0.216225

                 Fare
PassengerId  0.012658
Survived     0.257307
Pclass      -0.549500
Age          0.096067
SibSp        0.159651
Parch        0.216225
Fare         1.000000
```

[1006]:
```
#plt.subplots(figsize=(15,15))
sns.heatmap(corr,annot=True)
```

[1006]:
```
<Axes: >
```



[1007]:
```
df.Survived.value_counts()
```

[1007]:
```
0    549
1    342
Name: Survived, dtype: int64
```

[1008]:
```
df.Parch.value_counts()
```

[1008]:
```
0    678
1    118
2     80
5      5
3      5
4      4
6      1
Name: Parch, dtype: int64
```

[1009]:
```
df.Sex.value_counts()
```

[1009]:
```
male      577
female    314
Name: Sex, dtype: int64
```

1.1.3 *3.Checking for Null Values.*

[1010]:
```
df.isnull().any()
```

[1010]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age             True
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin           True
Embarked        True
dtype: bool
```

[1011]:
```
df.isnull().sum()
```

[1011]:
```
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
```

[1012]:
```
df['Age'].fillna(df['Age'].mean(),inplace=True)
```

[1013]:
```
df['Cabin'].fillna(df['Cabin'].mode().iloc[0],inplace=True)
```

[1014]:
```
df['Embarked'].fillna(df['Embarked'].mode().iloc[0],inplace=True)
```

[1015]:
```
df.isnull().any()
```

[1015]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age            False
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin          False
Embarked       False
dtype: bool
```

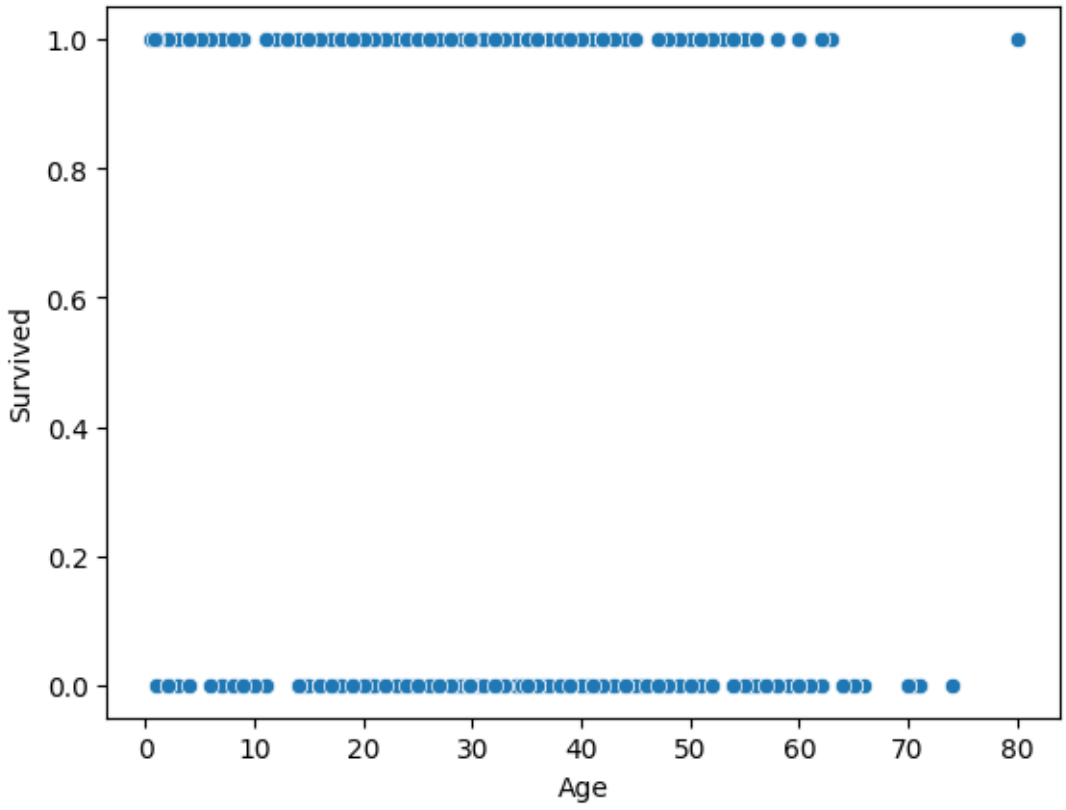
[1016]:
```
df.isnull().sum()
```

[1016]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       0
dtype: int64
```

1.1.4 *data visualization*

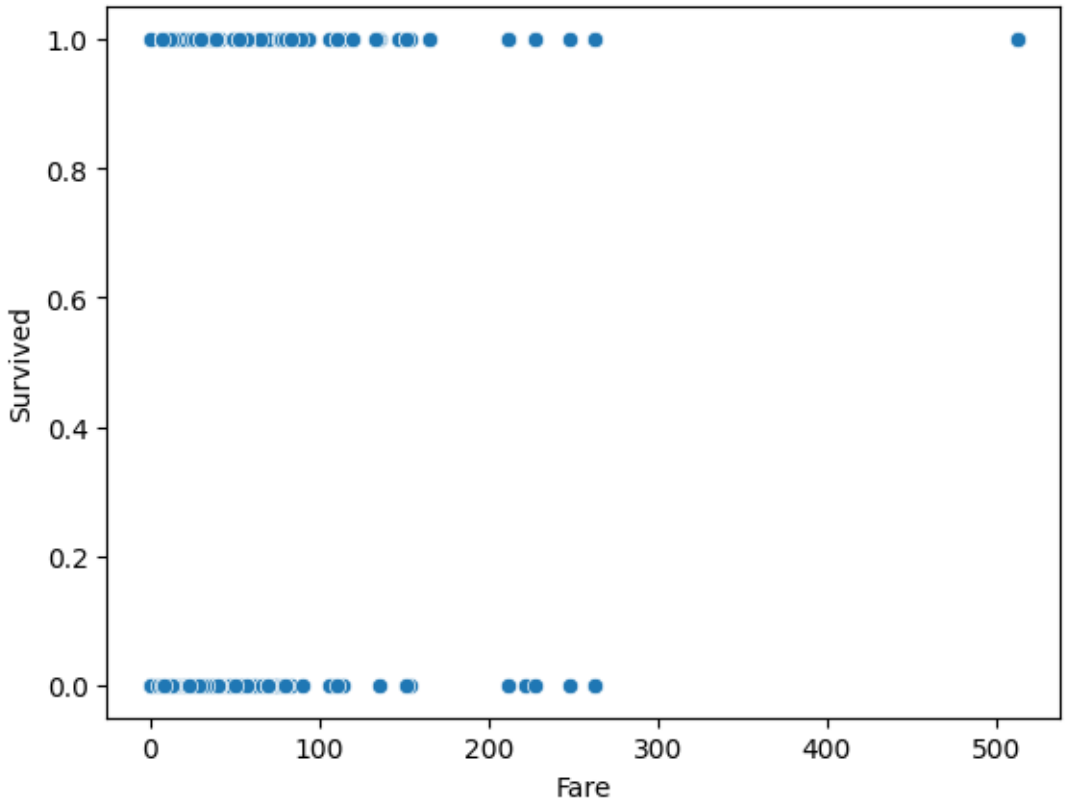
[1017]:
```
sns.scatterplot(x='Age',y='Survived',data=df)
```

[1017]:
```
<Axes: xlabel='Age', ylabel='Survived'>
```



[1018]:
```
sns.scatterplot(x='Fare',y='Survived',data=df)
```

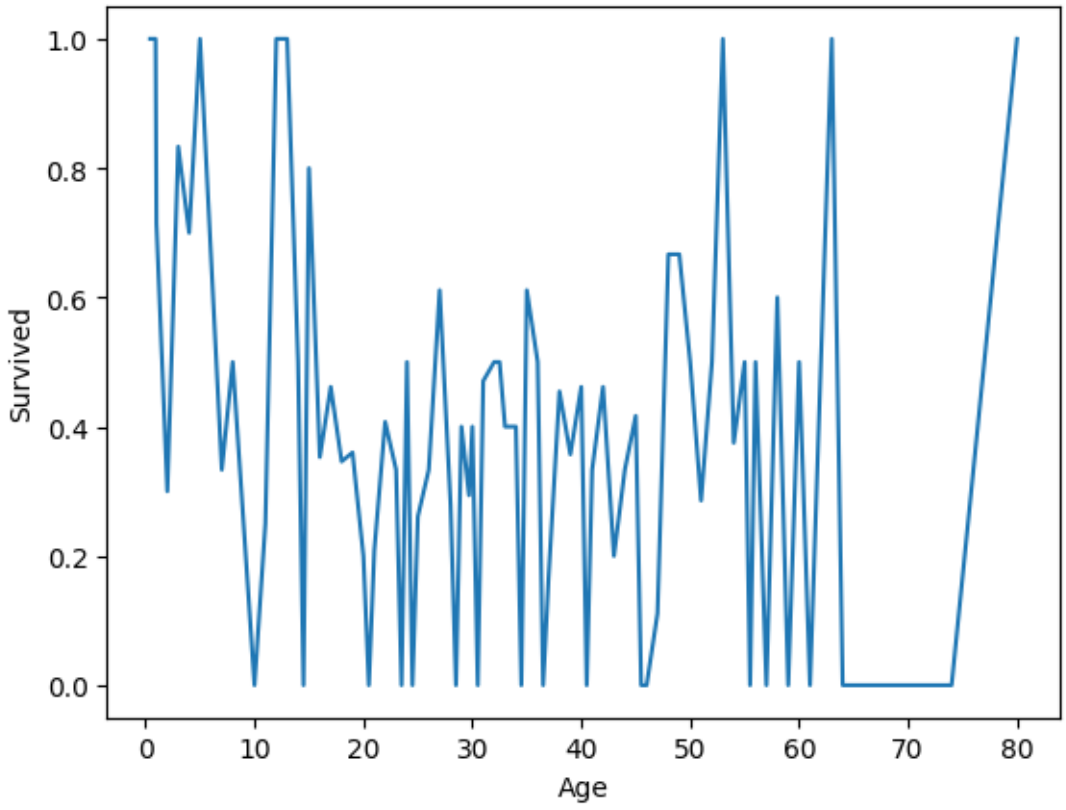
[1018]: <Axes: xlabel='Fare', ylabel='Survived'>



[1019]:
```
sns.lineplot(x='Age',y='Survived',data=df,ci=None)#ci=confidence interval
```

```
<ipython-input-1019-c9b9de42cb55>:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.lineplot(x='Age',y='Survived',data=df,ci=None)#ci=confidence interval
```

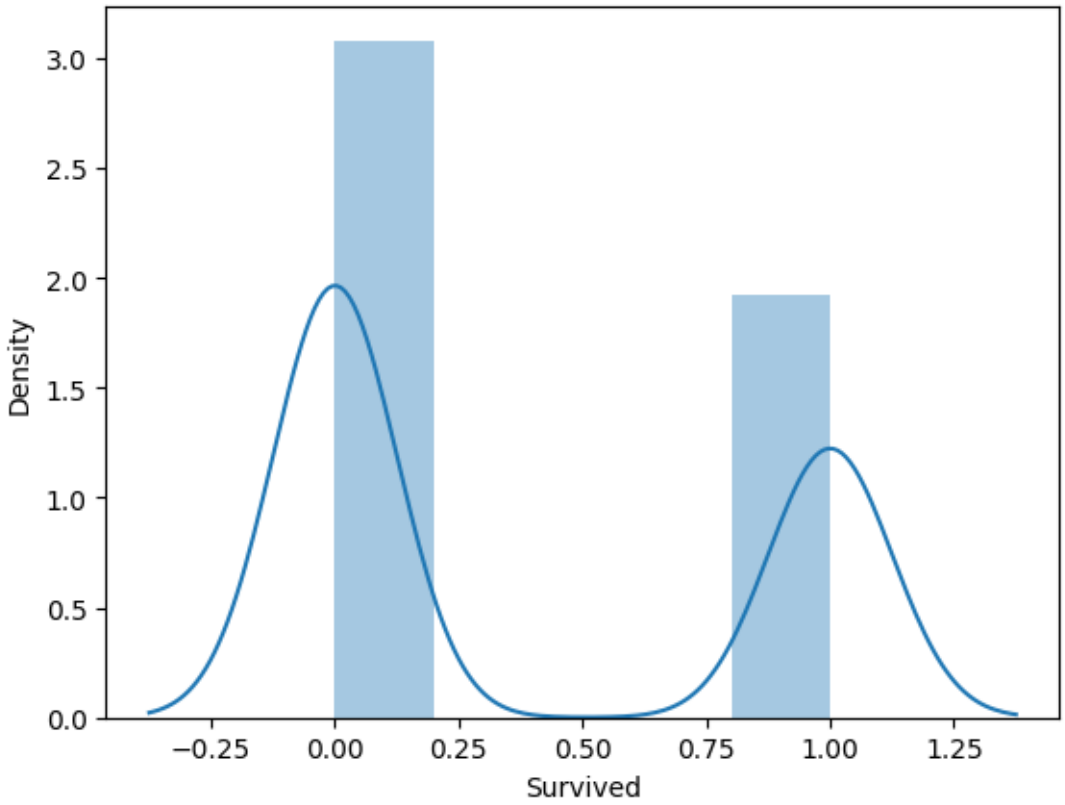
[1019]: `<Axes: xlabel='Age', ylabel='Survived'>`



[1020]:
```
sns.distplot(df['Survived'])
```

```
<ipython-input-1020-ef0f649fc5b0>:1: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df['Survived'])
```

[1020]: `<Axes: xlabel='Survived', ylabel='Density'>`



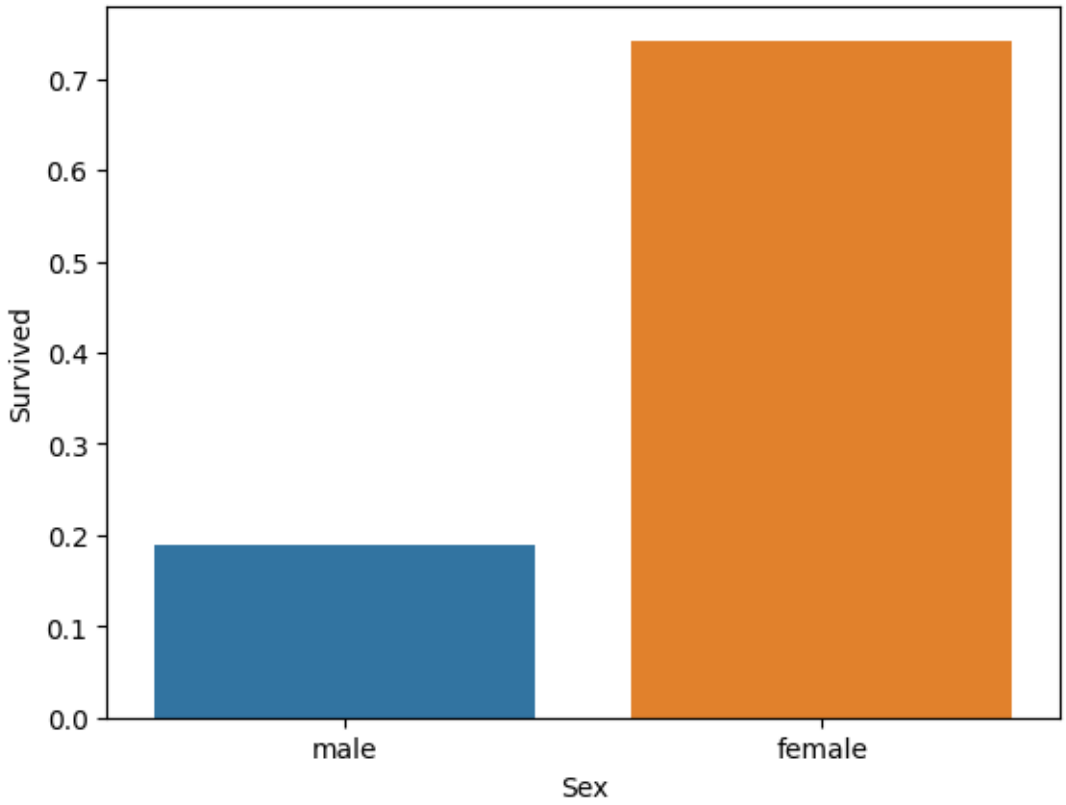
[1021]:
```
df['Sex'].value_counts()
```

[1021]:
```
male      577
female    314
Name: Sex, dtype: int64
```

[1022]:
```
sns.barplot(x='Sex',y='Survived',data=df,ci=None)
```

```
<ipython-input-1022-1c7b4b565508>:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.barplot(x='Sex',y='Survived',data=df,ci=None)
```

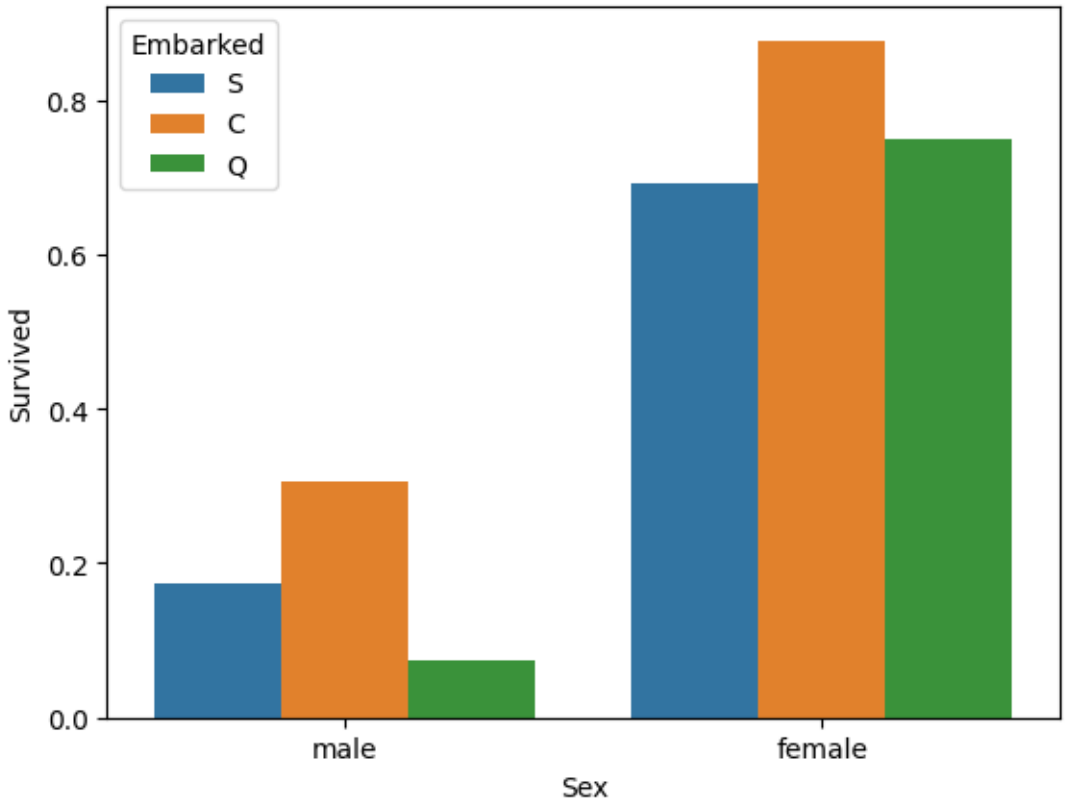
[1022]:
```
<Axes: xlabel='Sex', ylabel='Survived'>
```



[1023]:
```
sns.barplot(x='Sex',y='Survived',data=df,hue='Embarked',ci=None)
```

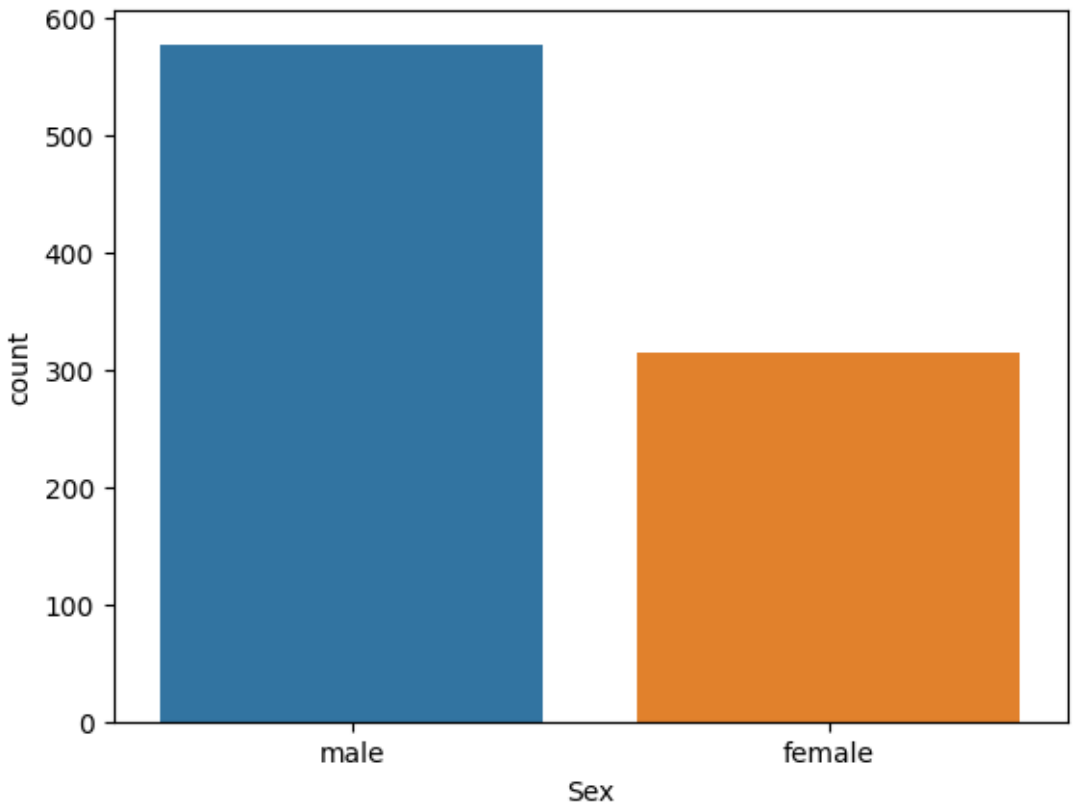
```
<ipython-input-1023-9245ea1d7de9>:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.barplot(x='Sex',y='Survived',data=df,hue='Embarked',ci=None)
```

[1023]:
```
<Axes: xlabel='Sex', ylabel='Survived'>
```



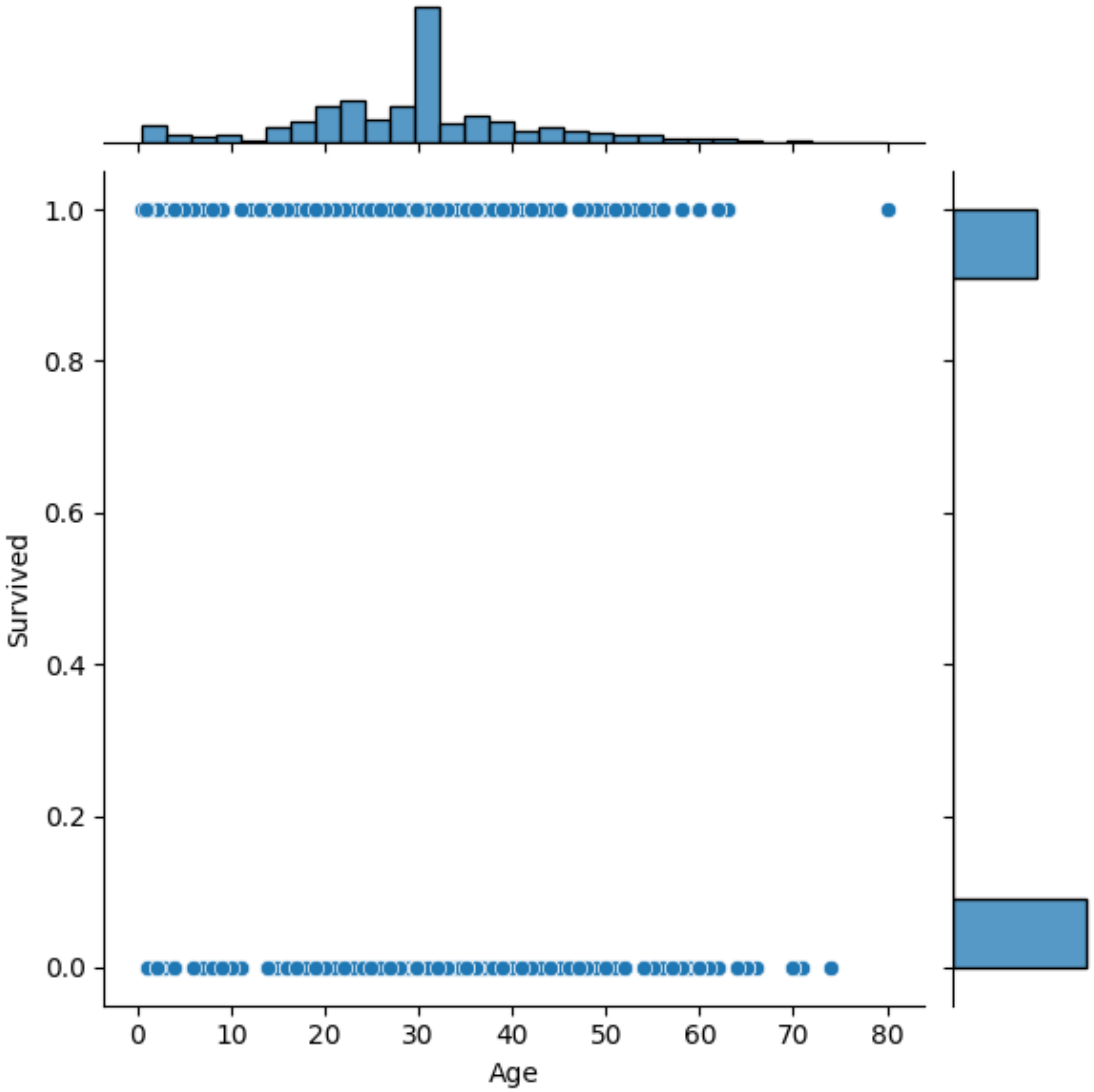
[1024]:
```
sns.countplot(x='Sex',data=df)
```

[1024]: <Axes: xlabel='Sex', ylabel='count'>



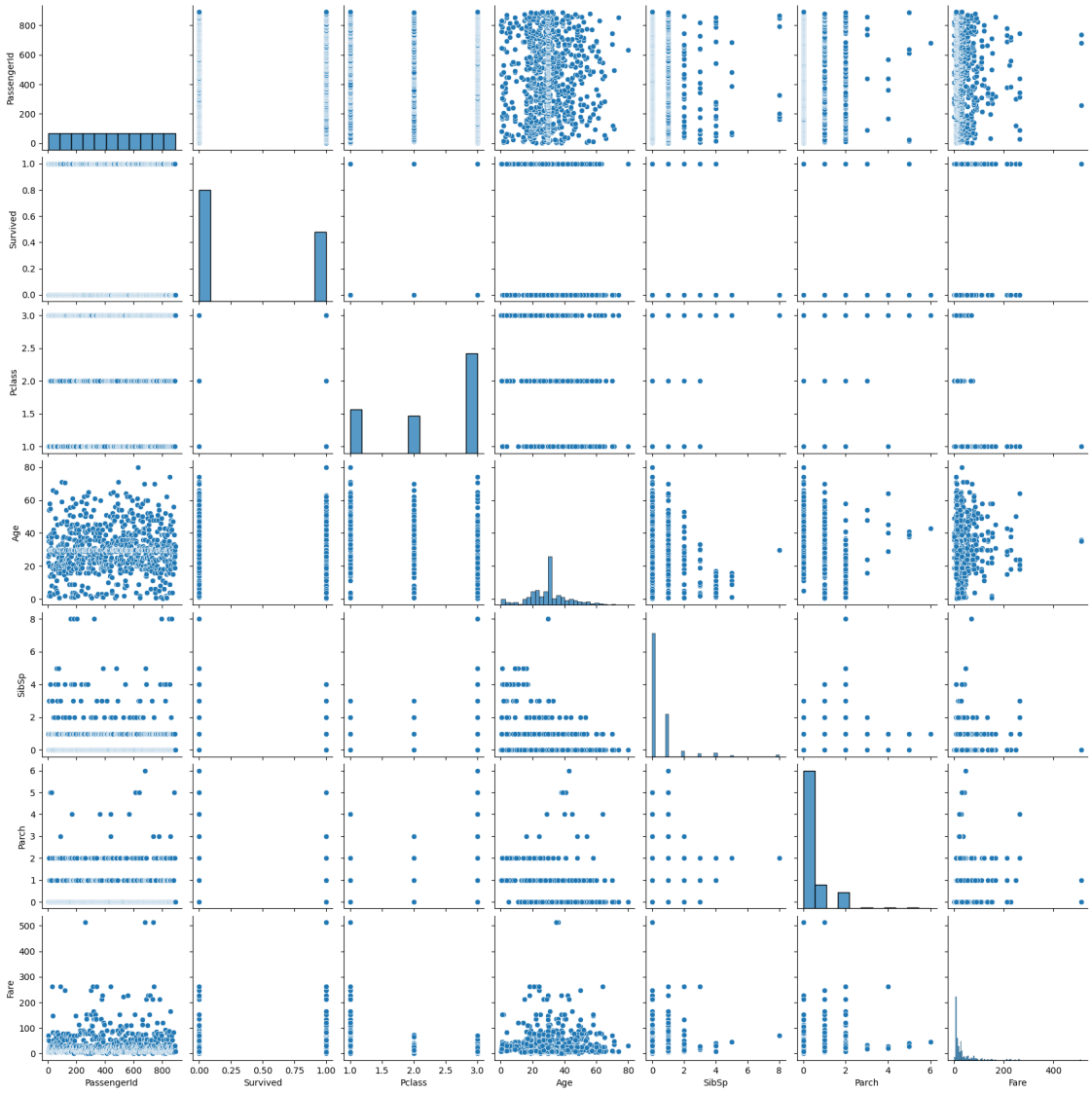
[1025]:
```
sns.jointplot(x='Age',y='Survived',data=df)
```

[1025]: <seaborn.axisgrid.JointGrid at 0x7d608f1ec9a0>



[1026]:
```
sns.pairplot(df)
```

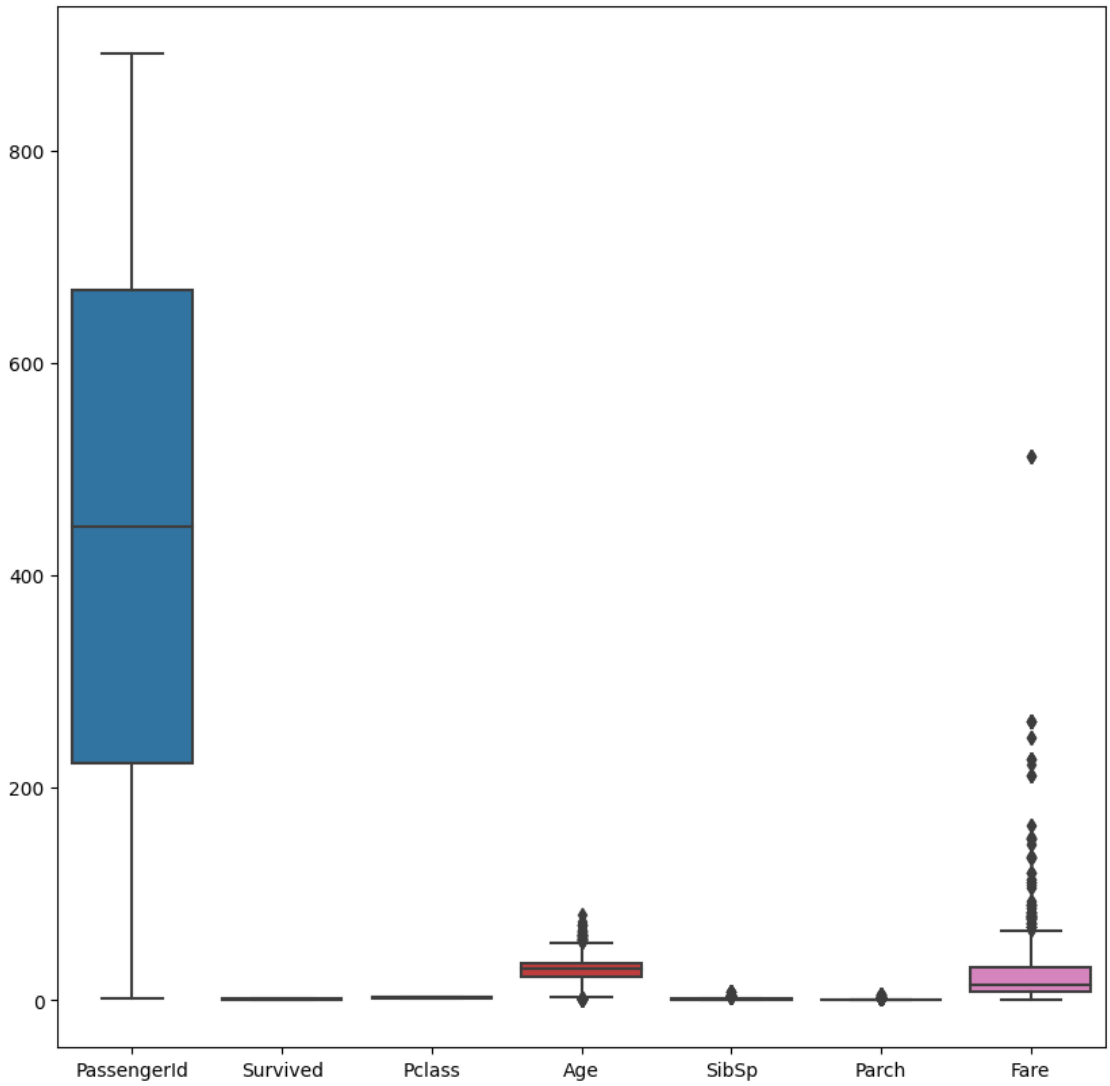
[1026]:
```
<seaborn.axisgrid.PairGrid at 0x7d6098e74c10>
```



1.1.5 *outlier detection and removal*

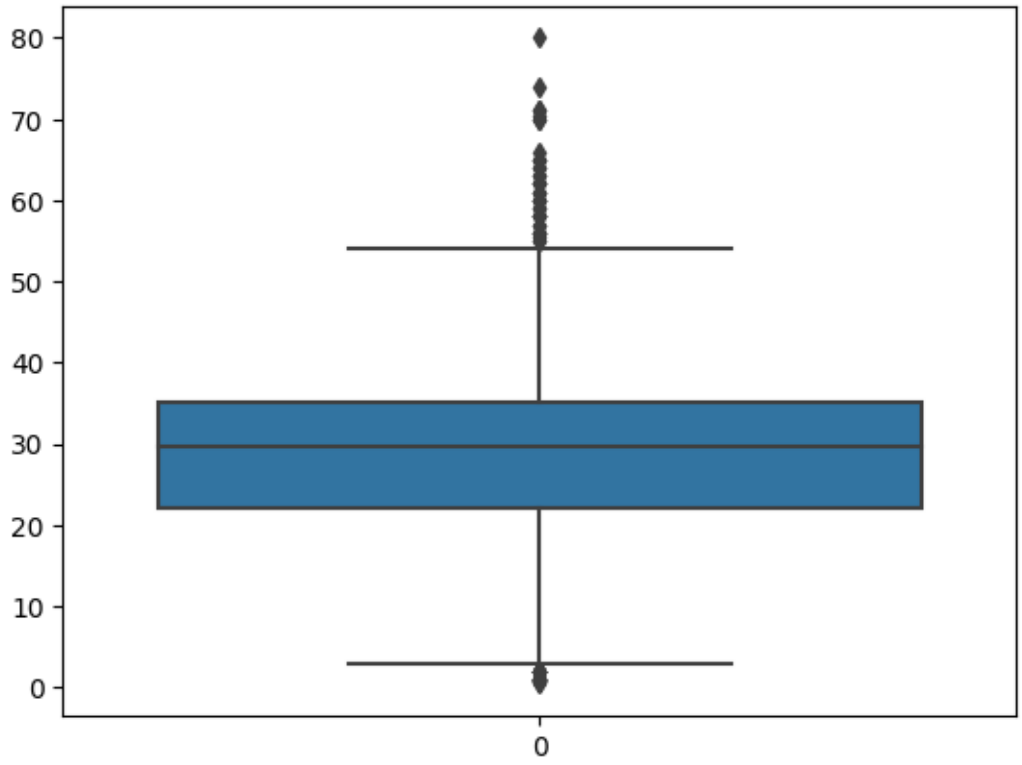
[1027]:
```
plt.subplots(figsize=(10,10))
sns.boxplot(df)
```

[1027]: <Axes: >



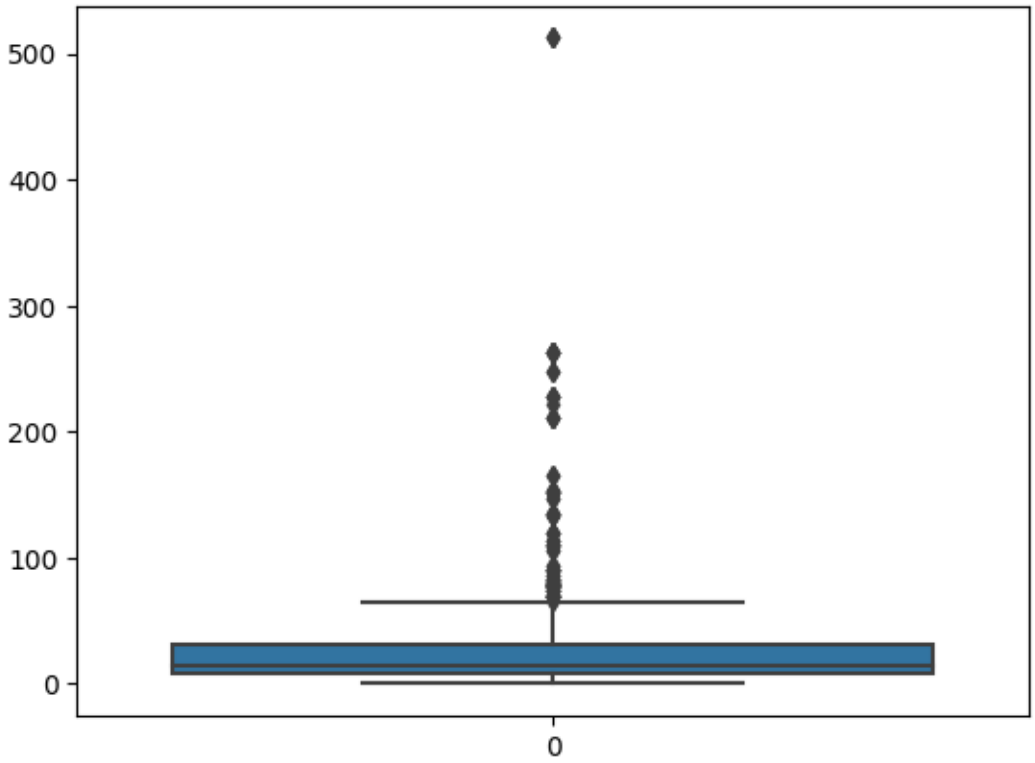
[1028]:
```
sns.boxplot(df['Age'])
```

[1028]: <Axes: >



[1029]:
```
sns.boxplot(df['Fare'])
```

[1029]: <Axes: >



[1030]:
```
df.median()
```

```
<ipython-input-1030-6d467abf240d>:1: FutureWarning: The default value of
numeric_only in DataFrame.median is deprecated. In a future version, it will
default to False. In addition, specifying 'numeric_only=None' is deprecated.
Select only valid columns or specify the value of numeric_only to silence this
warning.
  df.median()
```

[1030]:
```
PassengerId    446.000000
Survived         0.000000
Pclass           3.000000
Age             29.699118
SibSp            0.000000
Parch            0.000000
Fare            14.454200
dtype: float64
```

outliers removal for Age column

[1031]:
```
q1=df.Age.quantile(0.25)
q3=df.Age.quantile(0.75)
```

[1032]:
```
iqr=q3-q1
```

[1033]:
```
upper_limit=q3+(1.5*iqr)
```

[1034]:
```
lower_limit=q1-(1.5*iqr)
```

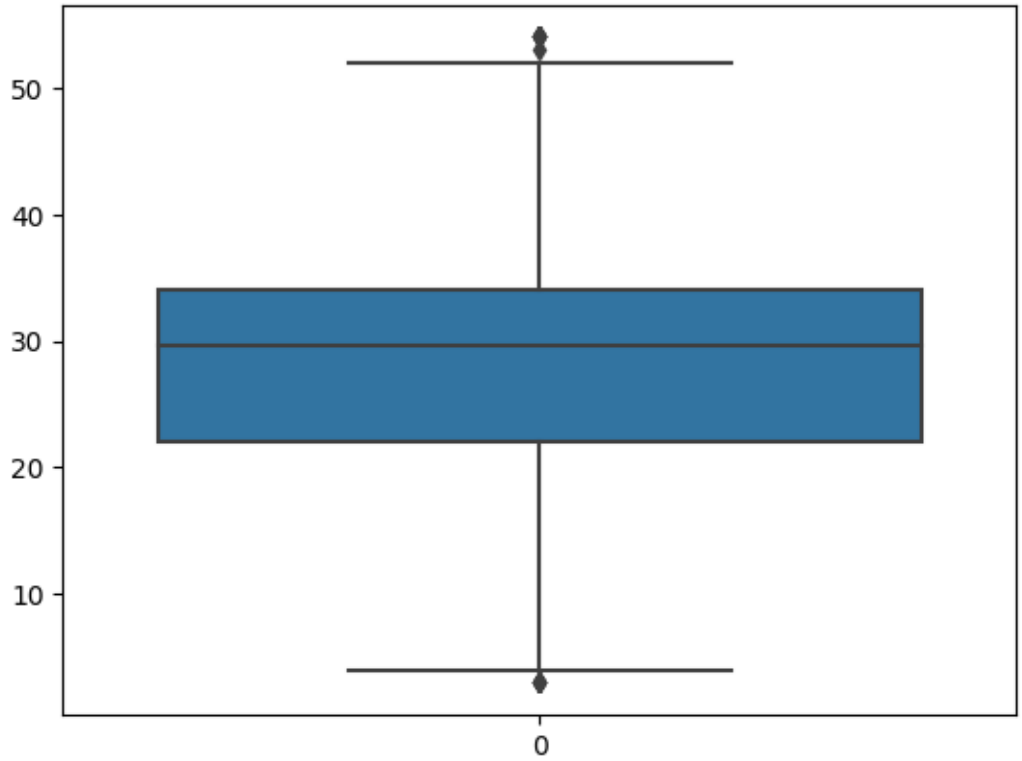
[1035]:
```
print("q1 of Age :",q1)
print("q3 of Age :",q3)
print("IQR of Age :",iqr)
print("upper_limit of Age :",upper_limit)
print("lower_limit of Age :",lower_limit)
```

```
q1 of Age : 22.0
q3 of Age : 35.0
IQR of Age : 13.0
upper_limit of Age : 54.5
lower_limit of Age : 2.5
```

[1036]:
```
df=df[(df.Age>=lower_limit) & (df.Age<=upper_limit)]
```

[1037]:
```
sns.boxplot(df['Age'])
```

[1037]: <Axes: >



1.1.6 *seperating dependent and independent varaibles*

[1038]:
```
df.head()
```

[1038]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare    Cabin Embarked
0      0         A/5 21171   7.2500  B96 B98        S
1      0          PC 17599  71.2833      C85        C
2      0  STON/O2. 3101282   7.9250  B96 B98        S
3      0            113803  53.1000     C123        S
4      0            373450   8.0500  B96 B98        S
```

[1039]:
```
x=df.iloc[:,4:12]
y=df.iloc[:,1:2]
```

[1040]:
```
x
```

[1040]:
```
        Sex        Age  SibSp  Parch            Ticket     Fare    Cabin  \
0      male  22.000000      1      0         A/5 21171   7.2500  B96 B98
1    female  38.000000      1      0          PC 17599  71.2833      C85
2    female  26.000000      0      0  STON/O2. 3101282   7.9250  B96 B98
3    female  35.000000      1      0            113803  53.1000     C123
4      male  35.000000      0      0            373450   8.0500  B96 B98
..      …        …    …    …                 …       …       …
886    male  27.000000      0      0            211536  13.0000  B96 B98
887  female  19.000000      0      0            112053  30.0000      B42
888  female  29.699118      1      2        W./C. 6607  23.4500  B96 B98
889    male  26.000000      0      0            111369  30.0000     C148
890    male  32.000000      0      0            370376   7.7500  B96 B98

    Embarked
0          S
```

```
1          C
2          S
3          S
4          S
..       …
886        S
887        S
888        S
889        C
890        Q

[825 rows x 8 columns]
```

[1041]:
```
y
```

[1041]:
```
     Survived
0            0
1            1
2            1
3            1
4            0
..         …
886          0
887          1
888          0
889          1
890          0

[825 rows x 1 columns]
```

[1042]:
```
x.shape
```

[1042]:
```
(825, 8)
```

[1043]:
```
y.shape
```

[1043]:
```
(825, 1)
```

1.2 *encoding*

[1044]:
```
from sklearn.preprocessing import LabelEncoder
```

[1045]:
```
x.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 825 entries, 0 to 890
Data columns (total 8 columns):
```

```
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Sex       825 non-null    object
 1   Age       825 non-null    float64
 2   SibSp     825 non-null    int64
 3   Parch     825 non-null    int64
 4   Ticket    825 non-null    object
 5   Fare      825 non-null    float64
 6   Cabin     825 non-null    object
 7   Embarked  825 non-null    object
dtypes: float64(2), int64(2), object(4)
memory usage: 90.3+ KB
```

[1046]:
```
le=LabelEncoder()
```

[1047]:
```
x['Sex']=le.fit_transform(x['Sex'])
```

[1048]:
```
x['Ticket']=le.fit_transform(x['Ticket'])
```

[1049]:
```
x['Cabin']=le.fit_transform(x['Cabin'])
```

[1050]:
```
x['Embarked']=le.fit_transform(x['Embarked'])
```

[1051]:
```
x.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 825 entries, 0 to 890
Data columns (total 8 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Sex       825 non-null    int64
 1   Age       825 non-null    float64
 2   SibSp     825 non-null    int64
 3   Parch     825 non-null    int64
 4   Ticket    825 non-null    int64
 5   Fare      825 non-null    float64
 6   Cabin     825 non-null    int64
 7   Embarked  825 non-null    int64
dtypes: float64(2), int64(6)
memory usage: 90.3 KB
```

[1052]:
```
x.head()
```

[1052]:

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	22.0	1	0	494	7.2500	38	2
1	0	38.0	1	0	565	71.2833	69	0
2	0	26.0	0	0	635	7.9250	38	2

```
3    0  35.0      1      0      41  53.1000     45         2
4    1  35.0      0      0     446   8.0500     38         2
```

1.3 *splitting data into training and testing data*

[1053]:
```
from sklearn.model_selection import train_test_split
```

[1054]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=0)
```

[1055]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

[1055]:
```
((577, 8), (248, 8), (577, 1), (248, 1))
```

1.3.1 *feature scaling*

[1056]:
```
from sklearn.preprocessing import StandardScaler
```

[1057]:
```
sc=StandardScaler()
```

[1058]:
```
x_train = sc.fit_transform(x_train)
```

[1059]:
```
x_test=sc.fit_transform(x_test)
```

[1060]:
```
x_train
```

[1060]:
```
array([[-1.36771589, -1.28323547, -0.46128242, …, -0.4843216 ,
        -0.28624881, -0.71856242],
       [ 0.731146  , -0.49024259, -0.46128242, …, -0.42905887,
        -0.28624881,  0.56506147],
       [ 0.731146  ,  0.07467737, -0.46128242, …, -0.47799575,
        -0.28624881,  0.56506147],
       …,
       [ 0.731146  , -0.88673903, -0.46128242, …, -0.45452602,
        -0.28624881,  0.56506147],
       [ 0.731146  ,  0.60012261, -0.46128242, …, -0.10847231,
        -0.28624881, -2.0021863 ],
       [ 0.731146  ,  0.07467737, -0.46128242, …, -0.48107578,
        -0.28624881,  0.56506147]])
```

[1060]:
```

```