purushothamreddy-assignment-3

September 20, 2023

[74]:

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

[75]:

```
df=pd.read_csv('/titanic.csv')
df.head()
```

[75]:

```
   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
```

[76]:

```
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
```

[77]:
```
df.describe()
```

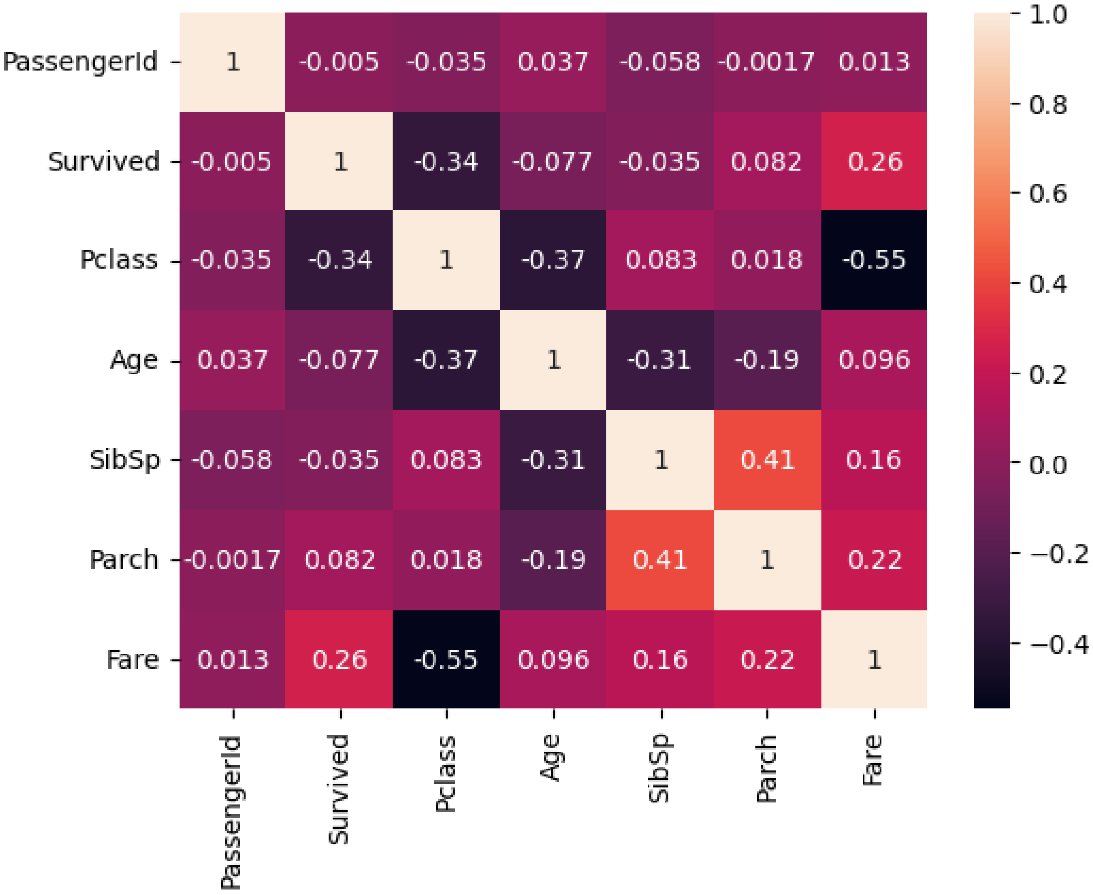
[77]:
```
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    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
```

[78]:
```
corr=df.corr()
corr
sns.heatmap(corr,annot=True)
```

```
<ipython-input-78-f6e6d731016f>: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()
```

[78]:
```
<Axes: >
```



[79]:
```
df.Cabin.value_counts()
```

[79]:
```
B96 B98        4
G6             4
C23 C25 C27    4
C22 C26        3
F33            3
              ..
E34            1
C7             1
C54            1
E36            1
C148           1
Name: Cabin, Length: 147, dtype: int64
```

[80]:
```
df.Embarked.value_counts()
```

[80]:

```
S    644
C    168
Q     77
Name: Embarked, dtype: int64
```

[81]:

```
df.Parch.value_counts()
```

[81]:

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

[82]:

```
df.isnull().any()
```

[82]:

```
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
```

[83]:

```
df.isnull().sum()
```

[83]:

```
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
```

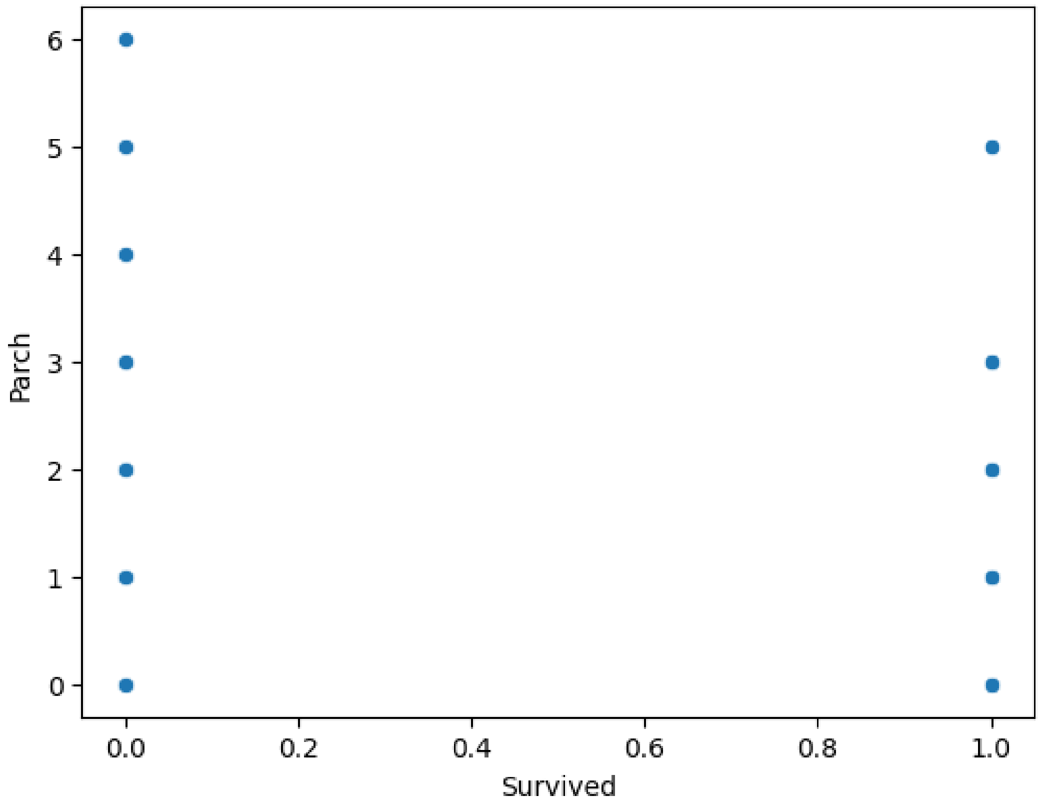
[84]:
```
df["Age"].fillna(df["Age"].mean(),inplace=True)
df["Cabin"].fillna(df["Cabin"].mode()[0],inplace=True)
df["Embarked"].fillna(df["Embarked"].mode()[0],inplace=True)
```

[85]:
```
df.isnull().sum()#I removed all null values
```

[85]:
```
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
```

[86]:
```
sns.scatterplot(x=df["Survived"],y=df["Parch"])
```

[86]:
```
<Axes: xlabel='Survived', ylabel='Parch'>
```

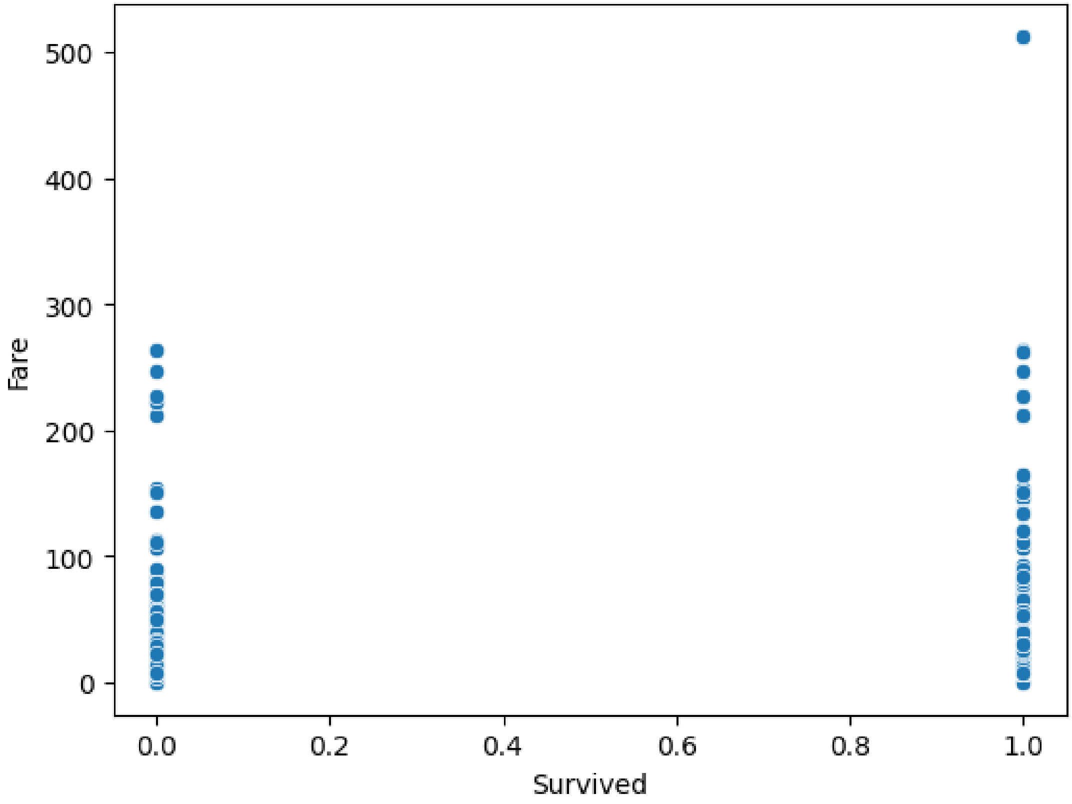


[87]:

```
sns.scatterplot(x=df["Survived"],y=df["Fare"])
```

[87]:

```
<Axes: xlabel='Survived', ylabel='Fare'>
```

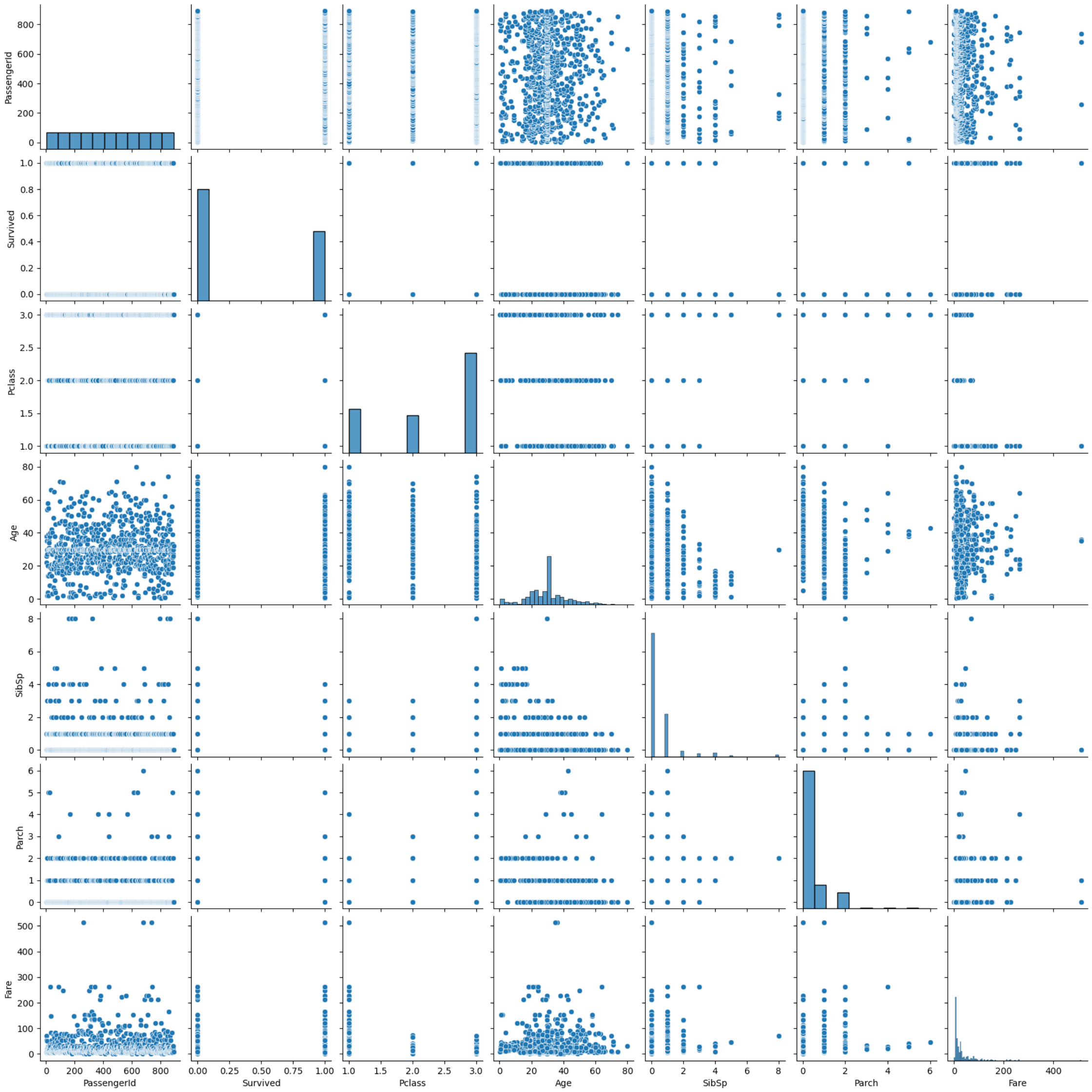


[88]:

```
sns.pairplot(df)
```

[88]:

```
<seaborn.axisgrid.PairGrid at 0x79e37a885060>
```



```
[89]: from sklearn.preprocessing import LabelEncoder
      le=LabelEncoder()
```

```
[90]: df["Sex"]=le.fit_transform(df["Sex"])
```

```
[91]: df["Embarked"]=le.fit_transform(df["Embarked"])
```

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

```
[92]:    PassengerId  Survived  Pclass  \
      0            1         0       3
      1            2         1       1
      2            3         1       3
```

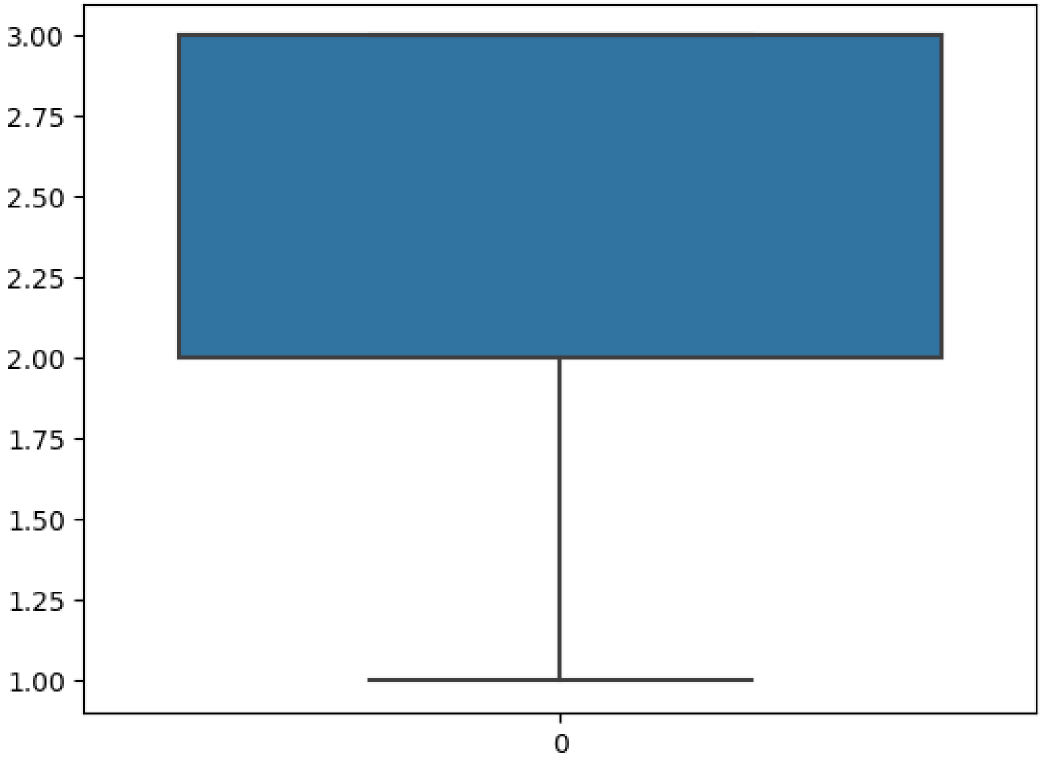
```
3            4         1       1
4            5         0       3

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

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

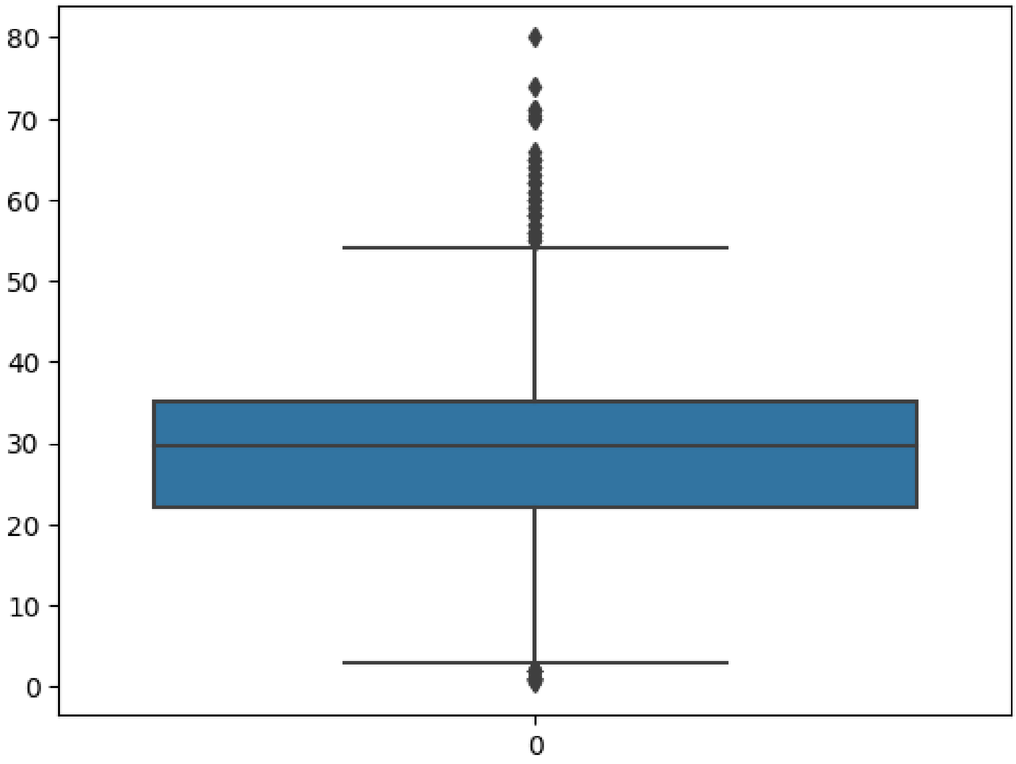
[93]:
```
sns.boxplot(df['Pclass'])
```

[93]:
```
<Axes: >
```



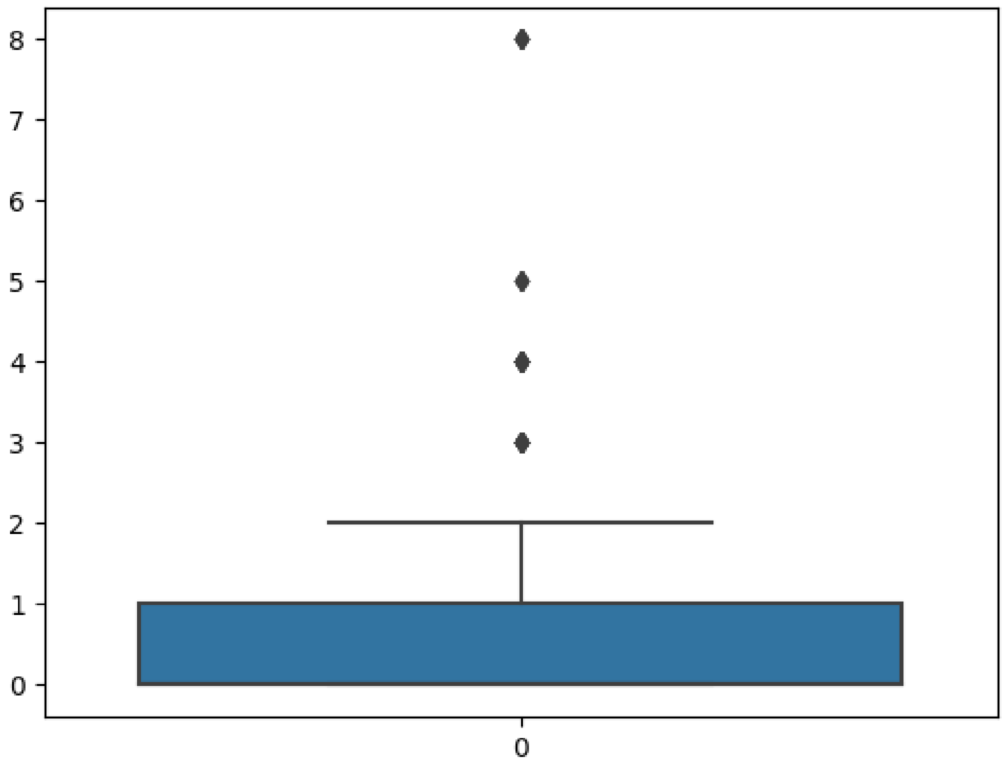
[94]:
```
sns.boxplot(df['Age'])
```

```
[94]: <Axes: >
```



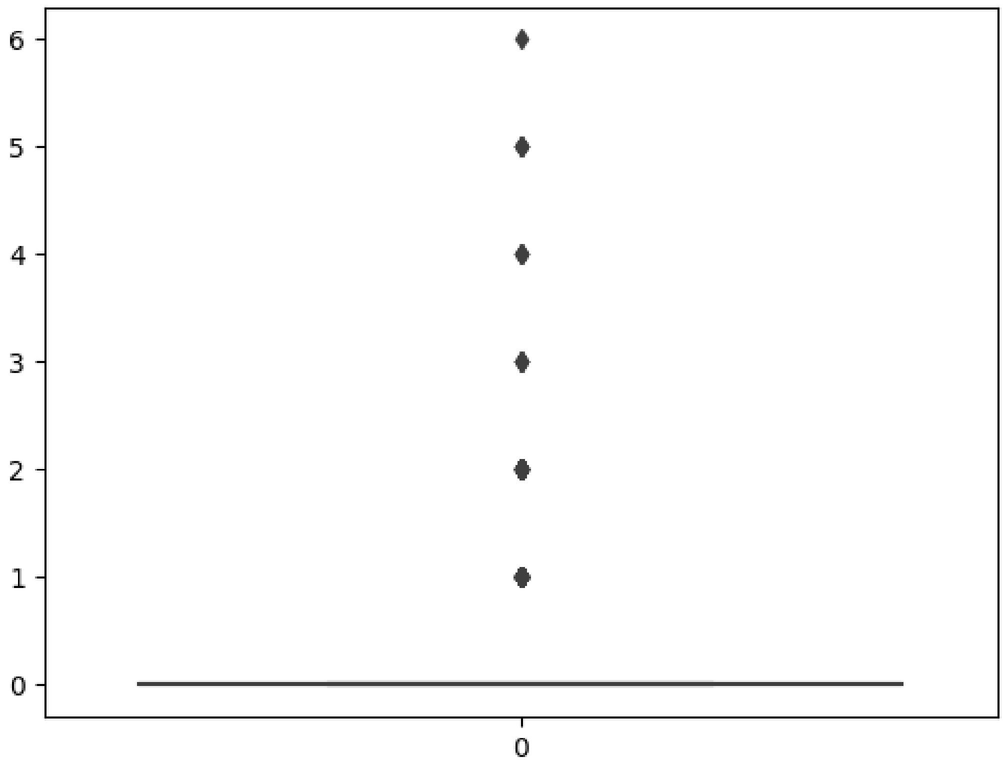
```
[95]: sns.boxplot(df['SibSp'])
```

```
[95]: <Axes: >
```



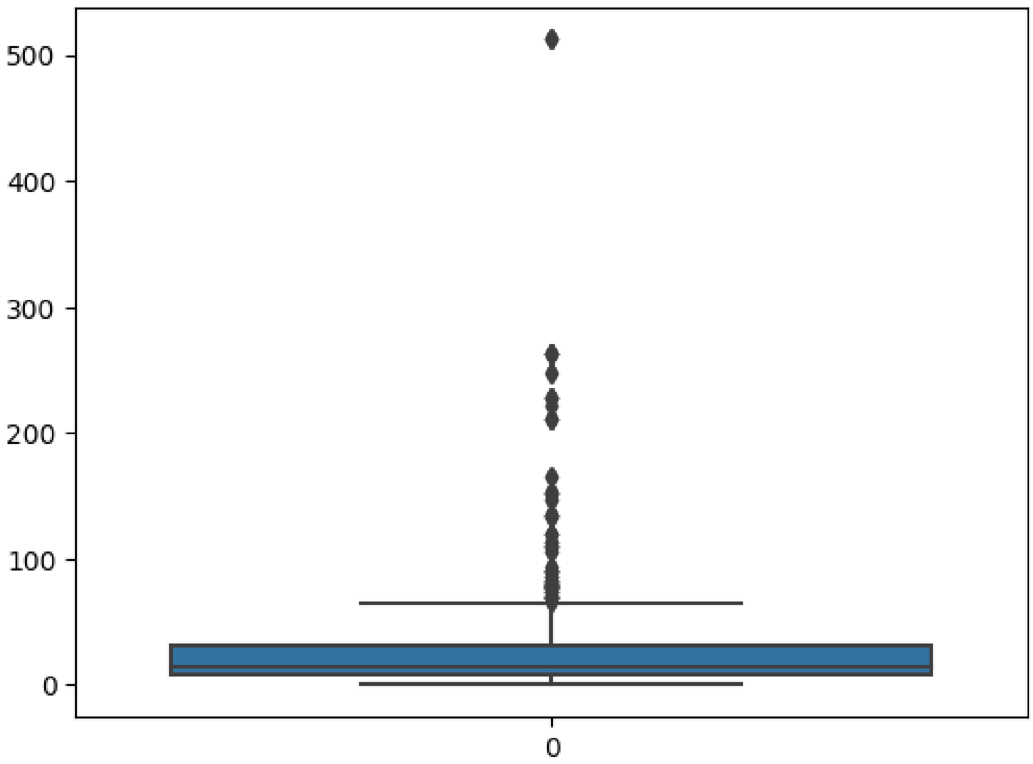
[96]:
```
sns.boxplot(df['Parch'])
```

[96]:
```
<Axes: >
```



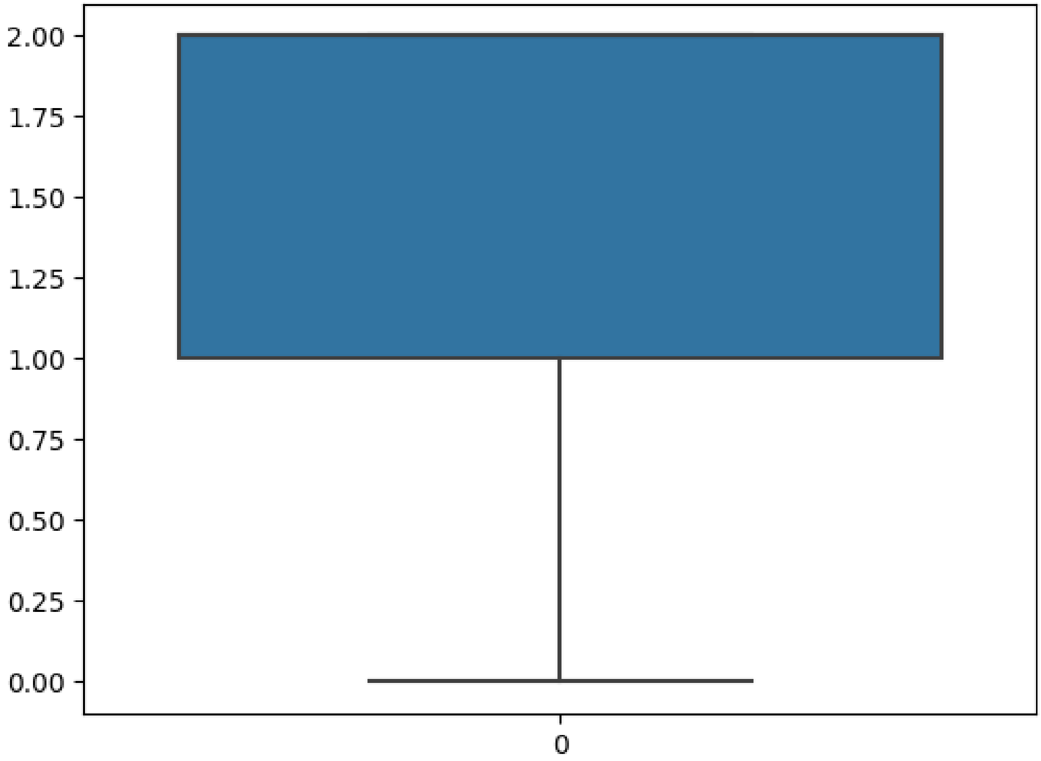
[97]:
```
sns.boxplot(df['Fare'])
```

[97]:
```
<Axes: >
```



```
[98]: sns.boxplot(df['Embarked'])
```

```
[98]: <Axes: >
```



[99]:
```
q1=df.Age.quantile(0.25)
q3=df.Age.quantile(0.75)
print(q1)
print(q3)
```

```
22.0
35.0
```

[100]:
```
iqr=q3-q1
iqr
```

[100]:
```
13.0
```

[101]:
```
upperlimit = q3+1.5*iqr
upperlimit
```

[101]:
```
54.5
```

[102]:
```
lowerlimit=q1-1.5*iqr
lowerlimit
```

[102]:
```
2.5
```

[103]:

```
df.median()
```

```
<ipython-input-103-6d467abf240d>: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()
```

[103]:

```
PassengerId    446.000000
Survived         0.000000
Pclass           3.000000
Sex              1.000000
Age             29.699118
SibSp            0.000000
Parch            0.000000
Fare            14.454200
Embarked         2.000000
dtype: float64
```

[104]:

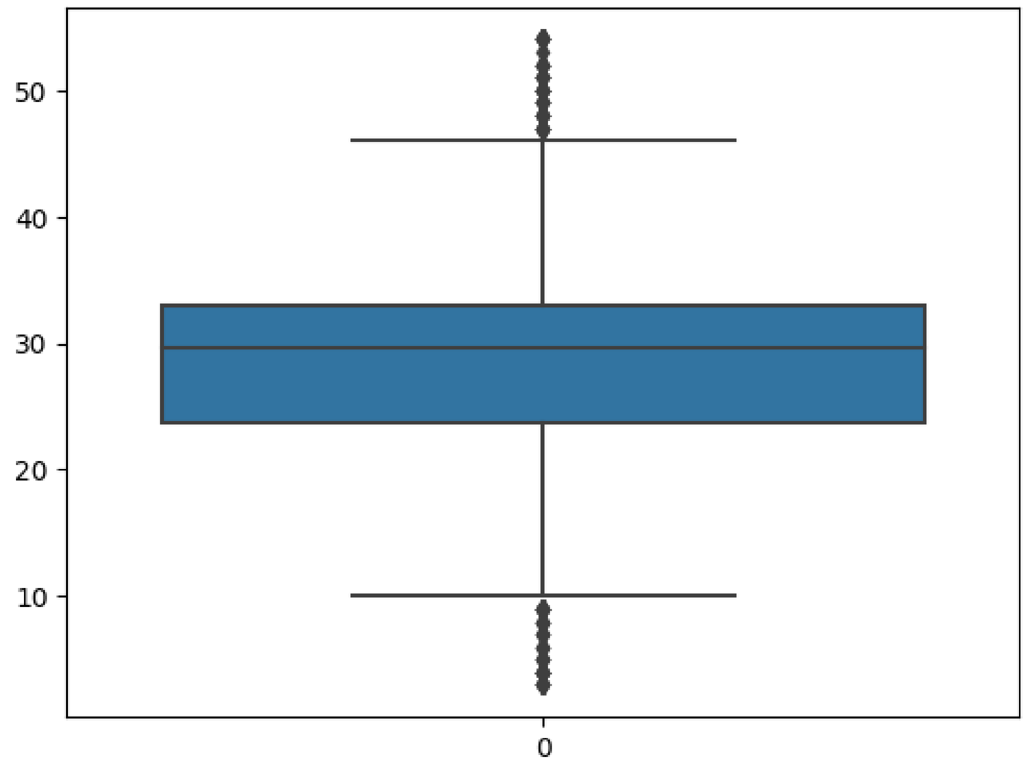
```
df['Age']=np.where(df['Age']>upperlimit,29.699118,df['Age'])
df['Age'] = np.where(df['Age'] < lowerlimit,29.699118, df['Age'])
```

[105]:

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

[105]:

```
<Axes: >
```



```
[106]: q1=df.SibSp.quantile(0.25)
       q3=df.SibSp.quantile(0.75)
       print(q1)
       print(q3)
```

```
0.0
1.0
```

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

```
[107]: 1.0
```

```
[108]: upperlimit = q3+1.5*iqr
       upperlimit
```

```
[108]: 2.5
```

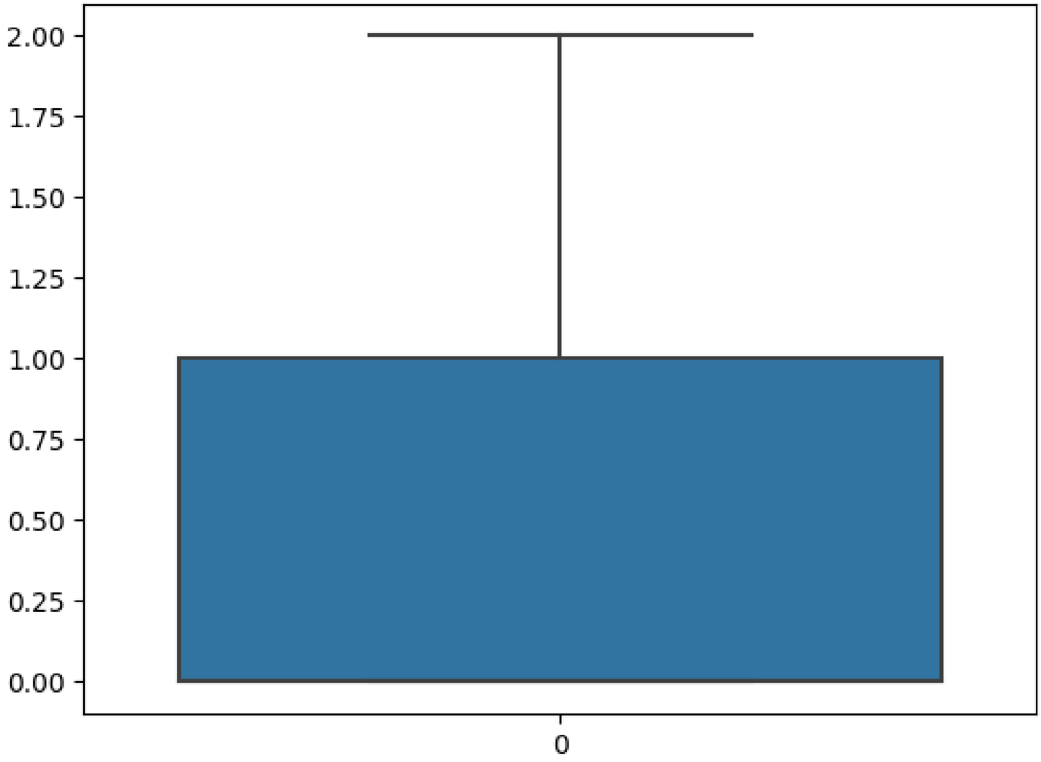
```
[109]: lowerlimit=q1-1.5*iqr
       lowerlimit
```

```
[109]: -1.5
```

[110]:
```
df['SibSp']=np.where(df['SibSp']>upperlimit,0.000000,df['SibSp'])
```

[111]:
```
sns.boxplot(df['SibSp'])
```

[111]:
```
<Axes: >
```



[112]:
```
q1=df.Parch.quantile(0.25)
q3=df.Parch.quantile(0.75)
print(q1)
print(q3)
```

```
0.0
0.0
```

[113]:
```
iqr=q3-q1
iqr
```

[113]:
```
0.0
```

[114]:
```
upperlimit = q3+1.5*iqr
upperlimit
```

[114]: 0.0

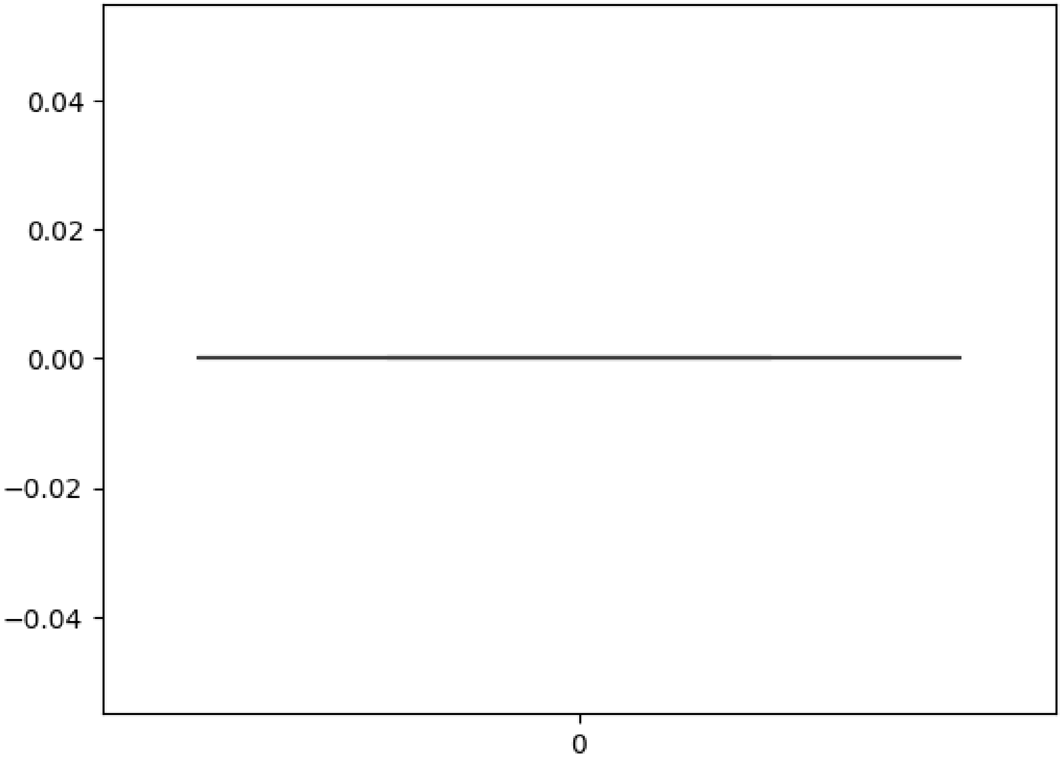
[115]:
```
lowerlimit=q1-1.5*iqr
lowerlimit
```

[115]: 0.0

[116]:
```
df['Parch']=np.where(df['Parch']>upperlimit,0.000000,df['Parch'])
```

[117]:
```
sns.boxplot(df['Parch'])
```

[117]: <Axes: >



[118]:
```
q1=df.Fare.quantile(0.25)
q3=df.Fare.quantile(0.75)
print(q1)
print(q3)
```

```
7.9104
31.0
```

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

```
[119]: 23.0896
```

```
[120]: upperlimit = q3+1.5*iqr
       upperlimit
```

```
[120]: 65.6344
```

```
[121]: lowerlimit=q1-1.5*iqr
       lowerlimit
```

```
[121]: -26.724
```

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

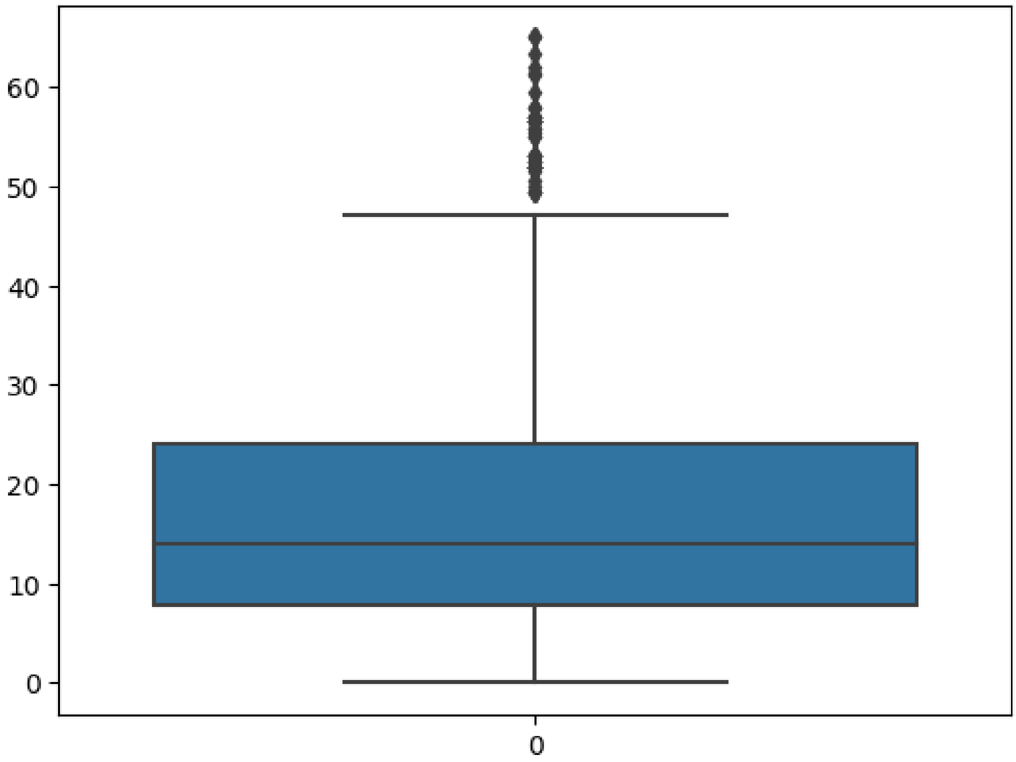
```
<ipython-input-122-6d467abf240d>:1: FutureWarning: The default value of
numeric_only in DataFrame.median is deprecated. In a future version, it will
default to False. In addition, specifying 'numeric_only=None' is deprecated.
Select only valid columns or specify the value of numeric_only to silence this
warning.
  df.median()
```

```
[122]: PassengerId    446.000000
       Survived         0.000000
       Pclass           3.000000
       Sex              1.000000
       Age             29.699118
       SibSp            0.000000
       Parch            0.000000
       Fare            14.454200
       Embarked         2.000000
       dtype: float64
```

```
[123]: df['Fare']=np.where(df['Fare']>upperlimit,14.054150,df['Fare'])
```

```
[124]: sns.boxplot(df.Fare)
```

```
[124]: <Axes: >
```



```
[125]: y=df["Survived"]
```

```
[126]: X=df.drop(columns=["Name","PassengerId","Survived","Ticket","Cabin"],axis=1)
```

```
[127]: y.head()
```

```
[127]: 0    0
       1    1
       2    1
       3    1
       4    0
       Name: Survived, dtype: int64
```

```
[128]: from sklearn.preprocessing import MinMaxScaler
       ms=MinMaxScaler()
```

```
[129]: X_Scaled=ms.fit_transform(X)
```

```
[130]: X_Scaled=pd.DataFrame(ms.fit_transform(X),columns=X.columns)
```

```
[131]: X_Scaled.head()
```

[131]:

```
   Pclass  Sex       Age  SibSp  Parch      Fare  Embarked
0     1.0  1.0  0.372549    0.5    0.0  0.111538       1.0
1     0.0  0.0  0.686275    0.5    0.0  0.216218       0.0
2     1.0  0.0  0.450980    0.0    0.0  0.121923       1.0
3     0.0  0.0  0.627451    0.5    0.0  0.816923       1.0
4     1.0  1.0  0.627451    0.0    0.0  0.123846       1.0
```

[132]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(X_Scaled,y,test_size =0.
 ↪2,random_state =0)
```

[133]:

```
print(x_train.shape,x_test.shape,y_train.shape,y_test.shape)
```

```
(712, 7) (179, 7) (712,) (179,)
```

[133]:

[133]:

[133]:

[133]:

[133]:

[133]:

[133]: