titanic-dataset

September 20, 2023

1 Data Preprocessing on Titanic Dataset

1.1 Import the libraries

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

1.2 Import the Titanic Dataset

[3]:
```
df=pd.read_csv("Titanic-Dataset.csv")
```

[4]:
```
df.head()
```

[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

[5]:
```
df.info
```

[5]:
```
<bound method DataFrame.info of      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]>
```

[6]:
```
df.describe
```

[6]:
```
<bound method NDFrame.describe of      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]>
```

[7]:
```
df.corr()
```

```
C:\Users\sbkom\AppData\Local\Temp\ipykernel_31760\1134722465.py: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.
  df.corr()
```

[7]:
```
             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
```

[8]:
```
df.corr().Fare.sort_values(ascending=False)
```

```
C:\Users\sbkom\AppData\Local\Temp\ipykernel_31760\60082530.py: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.
  df.corr().Fare.sort_values(ascending=False)
```

[8]:
```
Fare           1.000000
Survived       0.257307
Parch          0.216225
SibSp          0.159651
Age            0.096067
PassengerId    0.012658
Pclass        -0.549500
Name: Fare, dtype: float64
```

1.3 Checking for Null Values

[9]:
```
df.isnull().any()
```

[9]:
```
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
```

[10]:
```
df.isnull().sum()
```

[10]:
```
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
```

[11]:
```
df["Age"].fillna(df["Age"].mean(),inplace=True)
```

[12]:
```
df["Cabin"].fillna(df["Cabin"].mode()[0],inplace=True)
```

[13]:
```
df["Embarked"].fillna(df["Embarked"].mode()[0],inplace=True)
```

[14]:
```
df.isnull().any()
```

[14]:
```
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
```

[15]:
```
df.isnull().sum()
```

[15]:
```
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
```

[16]:
```
df.Embarked.nunique()
```

[16]:
```
3
```

[17]:
```
df.Embarked.unique()
```

[17]:
```
array(['S', 'C', 'Q'], dtype=object)
```

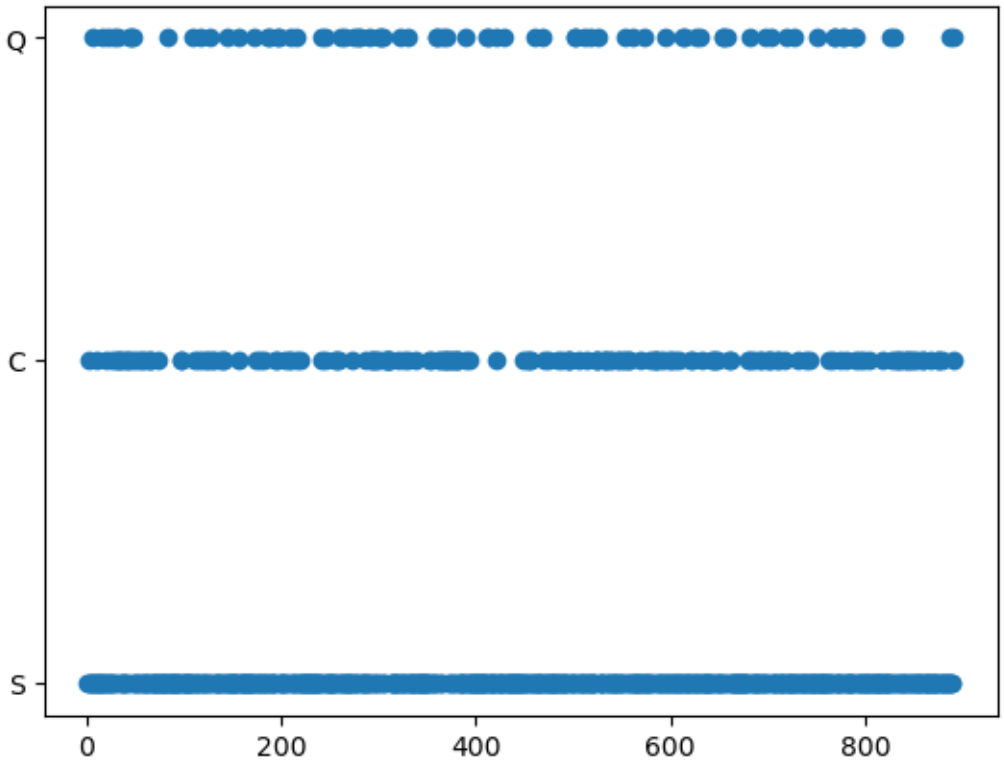
[18]:
```
df.Embarked.value_counts()
```

[18]:
```
S    646
C    168
Q     77
Name: Embarked, dtype: int64
```

2 Data Visualization

[19]:
```
plt.scatter(df["PassengerId"],df["Embarked"])
```

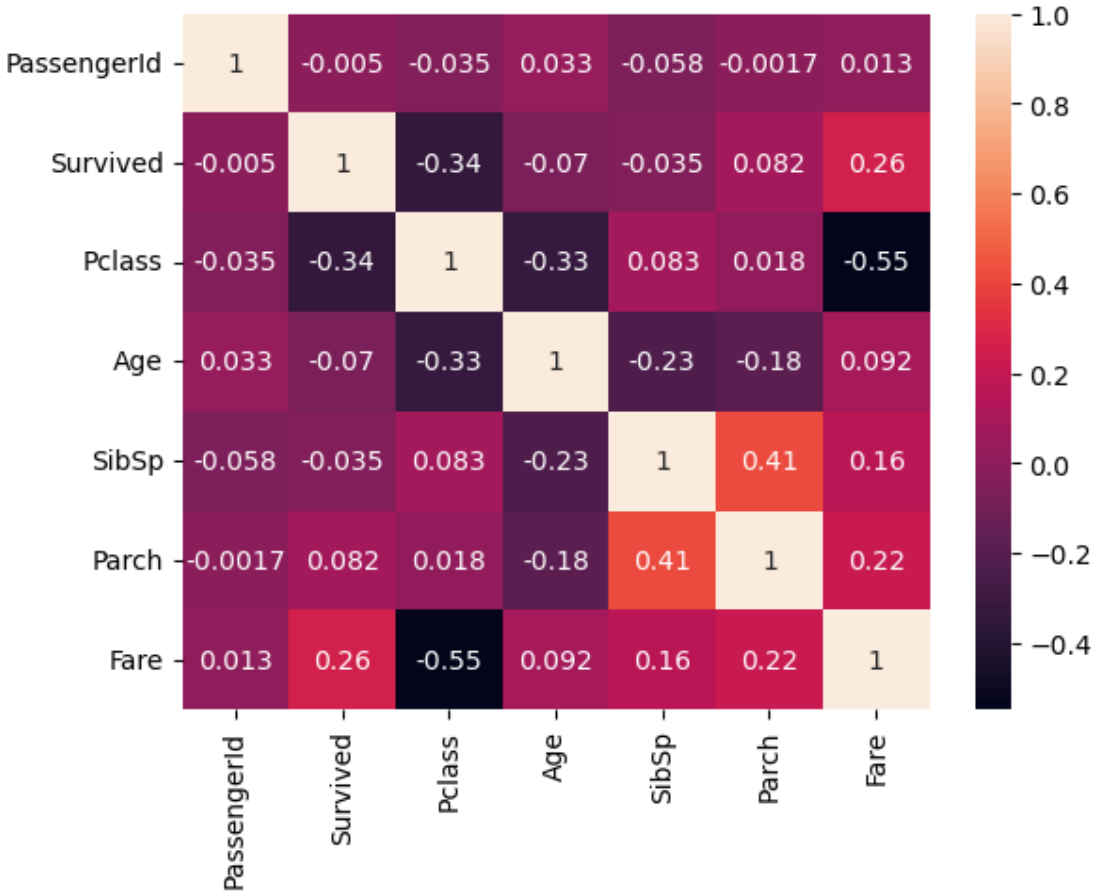
[19]:
```
<matplotlib.collections.PathCollection at 0x2107ad5fe50>
```



[20]:
```
sns.heatmap(df.corr(), annot=True)
```

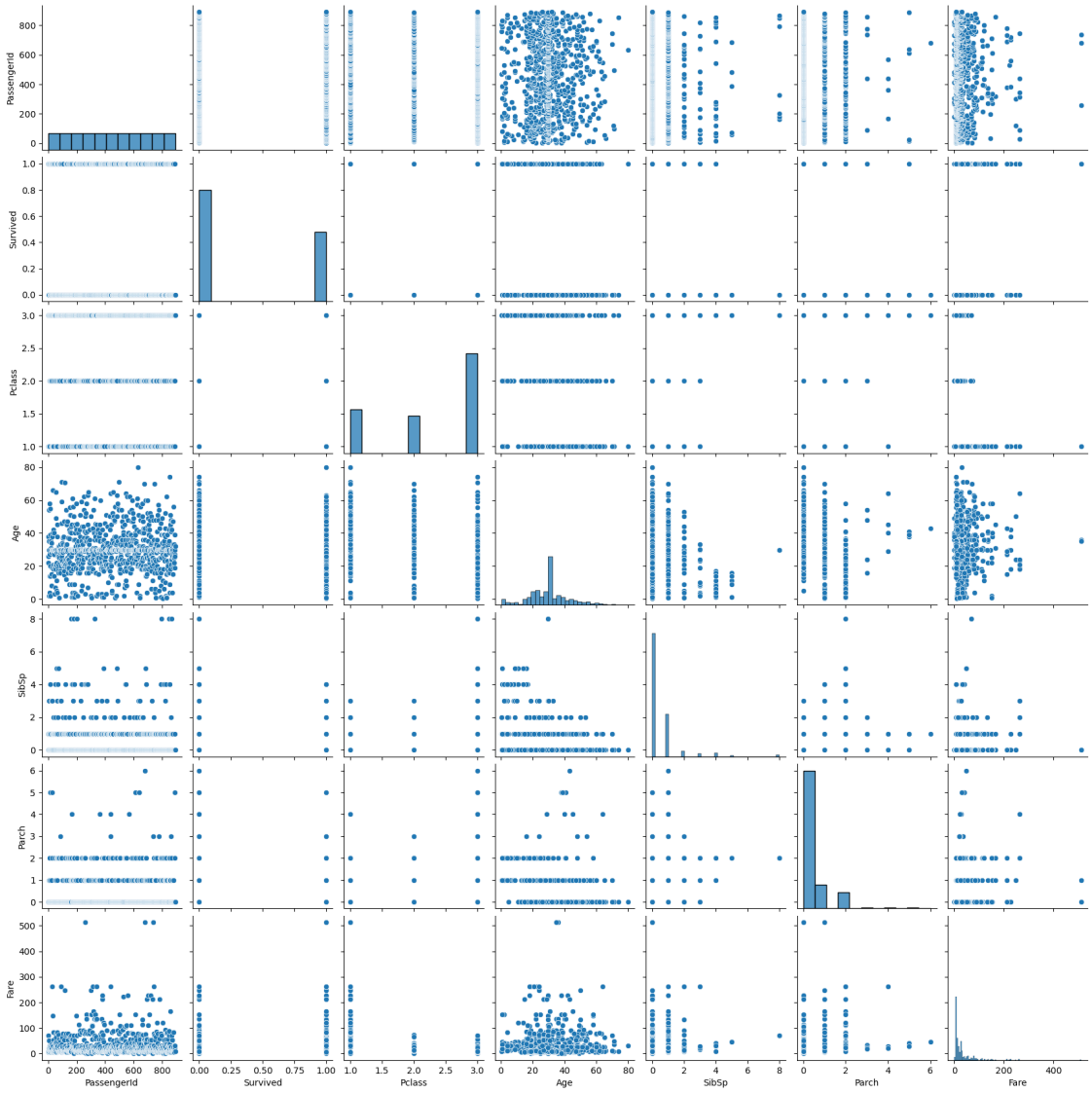
```
C:\Users\sbkom\AppData\Local\Temp\ipykernel_31760\621126171.py:1: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future
version, it will default to False. Select only valid columns or specify the
value of numeric_only to silence this warning.
  sns.heatmap(df.corr(), annot=True)
```

[20]: <Axes: >



[21]:
```
sns.pairplot(df)
```

[21]: <seaborn.axisgrid.PairGrid at 0x2107b7c39a0>



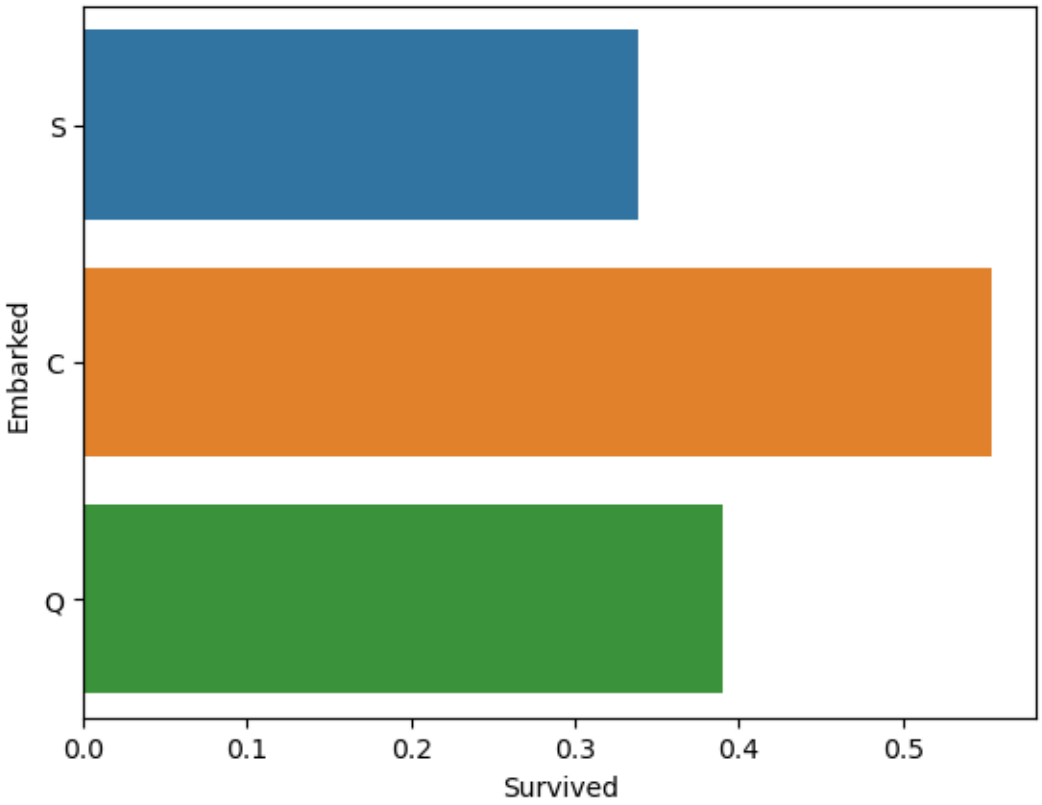
[22]:
```
sns.barplot(x=df["Survived"],y=df["Embarked"],ci=0)
```

```
C:\Users\sbkom\AppData\Local\Temp\ipykernel_31760\1646919353.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

  sns.barplot(x=df["Survived"],y=df["Embarked"],ci=0)
```

[22]: <Axes: xlabel='Survived', ylabel='Embarked'>



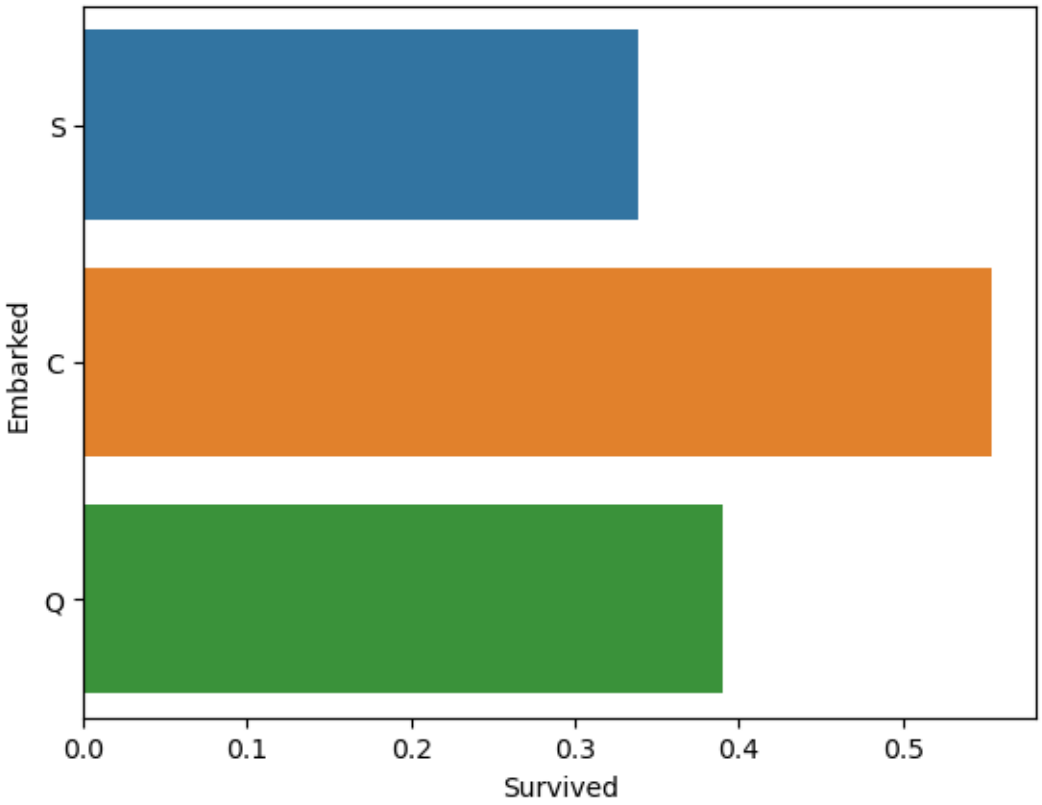
[23]:
```
sns.barplot(x=df["Survived"],y=df["Embarked"],ci=0)
```

```
C:\Users\sbkom\AppData\Local\Temp\ipykernel_31760\1646919353.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

  sns.barplot(x=df["Survived"],y=df["Embarked"],ci=0)
```

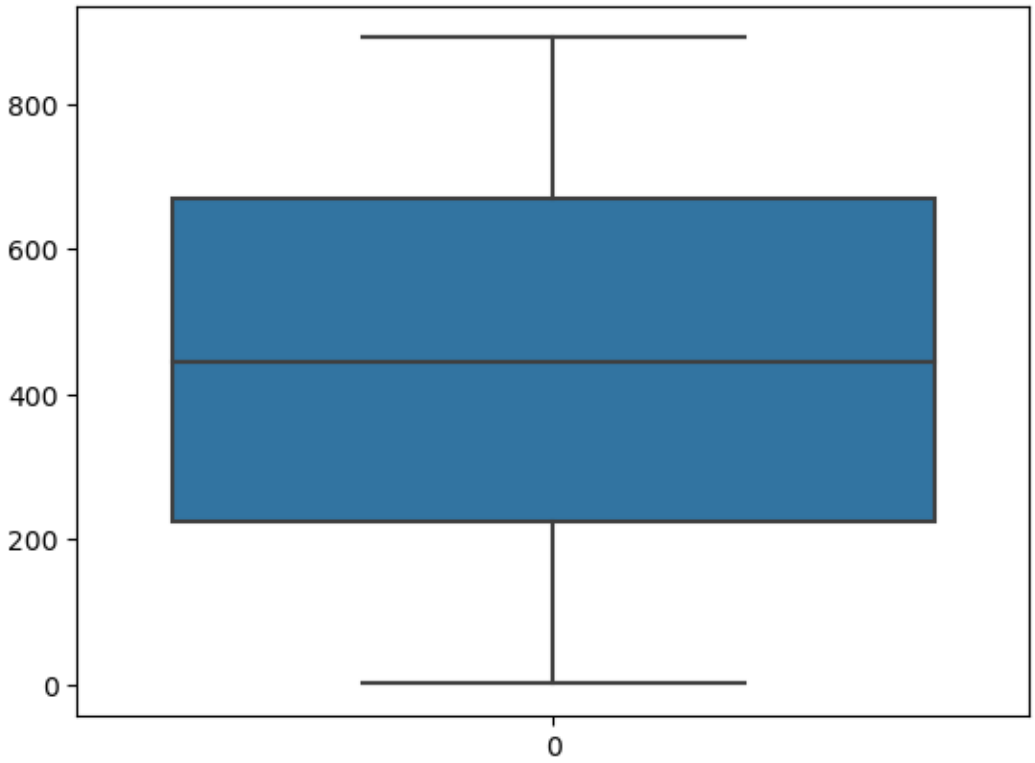
[23]:
```
<Axes: xlabel='Survived', ylabel='Embarked'>
```



3 Outlier Detection

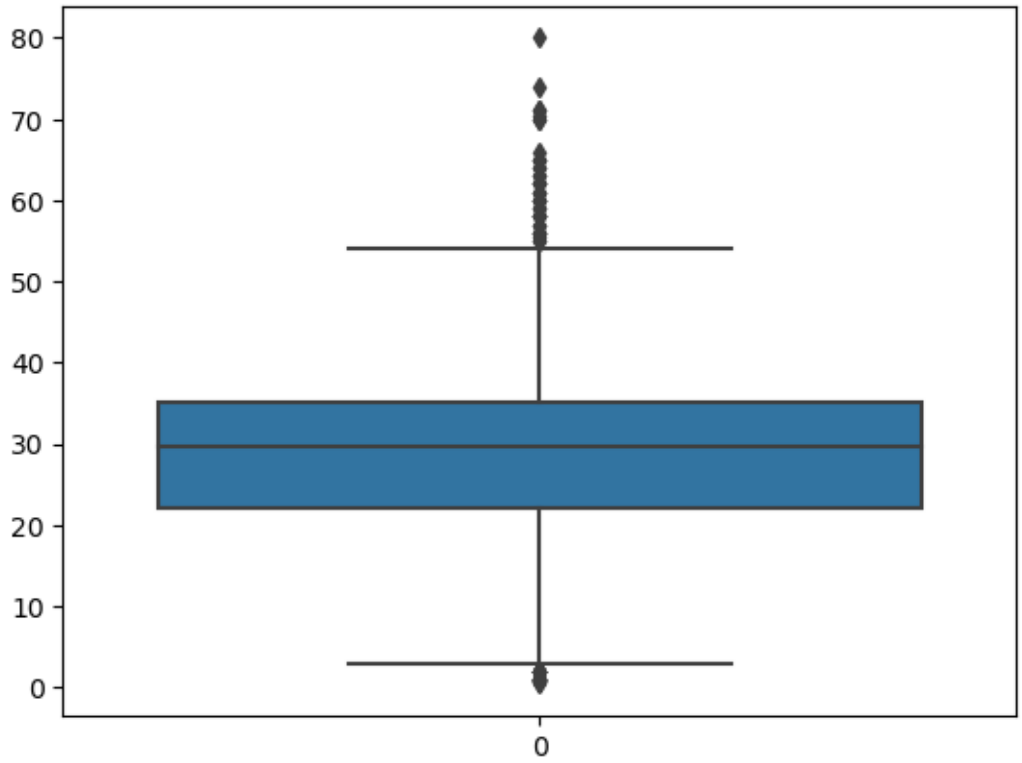
[52]:
```
sns.boxplot(df["PassengerId"])
```

[52]: <Axes: >



[24]:
```
sns.boxplot(df.Age)
```

[24]: <Axes: >



[25]:
```
q1 = df.Age.quantile(0.25) #qi mean 25 percentage of data
q3 = df.Age.quantile(0.75)
```

[26]:
```
IQR = q3-q1
print(IQR)
```

```
13.0
```

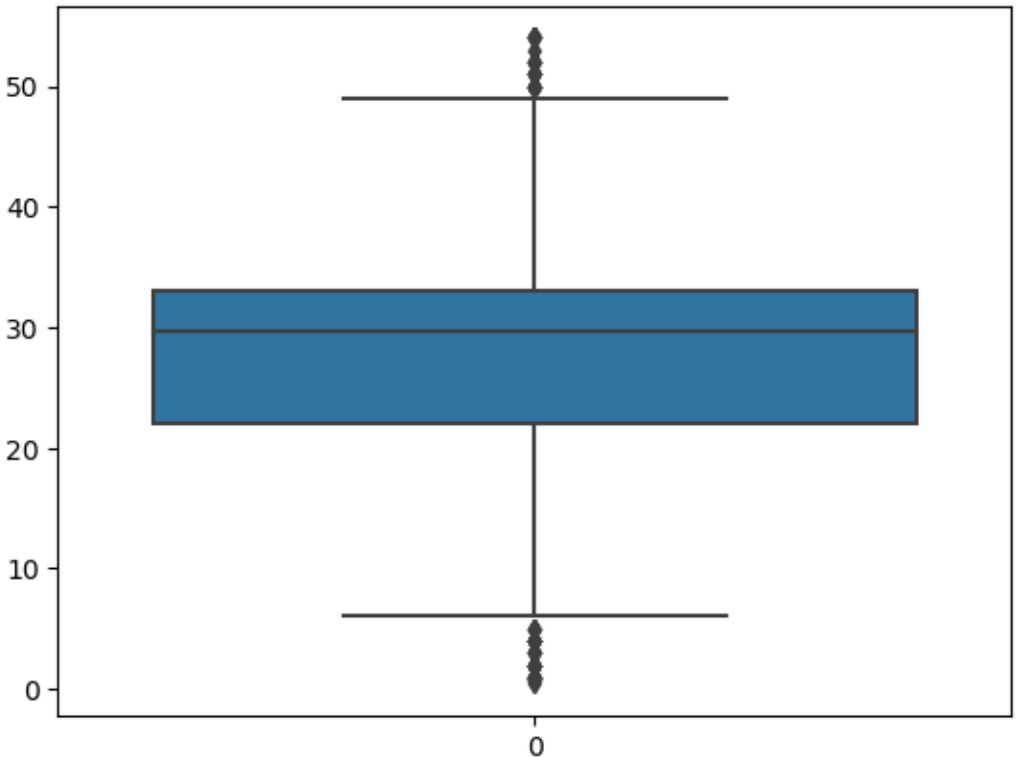
[27]:
```
upper_limit = q3+1.5*IQR
print(upper_limit)
```

```
54.5
```

[28]:
```
df = df[df.Age<upper_limit]
```

[29]:
```
sns.boxplot(df.Age)
```

[29]: <Axes: >



3.1 Splitting Dependent and Independent Variables

3.1.1 Method-I

[30]:
```
df.head()
```

[30]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	B96 B98	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1					
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0					
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1					
4	5	0	3	Allen, Mr. William Henry	male	35.0	0					

```
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
```

[31]:
```
X=df.drop(columns=["Survived"],axis=1)
X.head()
```

[31]:
```
   PassengerId  Pclass                                                Name  \
0            1       3                             Braund, Mr. Owen Harris
1            2       1  Cumings, Mrs. John Bradley (Florence Briggs Th…
2            3       3                              Heikkinen, Miss. Laina
3            4       1       Futrelle, Mrs. Jacques Heath (Lily May Peel)
4            5       3                           Allen, Mr. William Henry

      Sex   Age  SibSp  Parch            Ticket     Fare    Cabin Embarked
0    male  22.0      1      0         A/5 21171   7.2500  B96 B98        S
1  female  38.0      1      0          PC 17599  71.2833      C85        C
2  female  26.0      0      0  STON/O2. 3101282   7.9250  B96 B98        S
3  female  35.0      1      0            113803  53.1000     C123        S
4    male  35.0      0      0            373450   8.0500  B96 B98        S
```

[32]:
```
X=df.drop(columns=["Pclass"],axis=1)
X.head()
```

[32]:
```
   PassengerId  Survived                                                Name  \
0            1         0                             Braund, Mr. Owen Harris
1            2         1  Cumings, Mrs. John Bradley (Florence Briggs Th…
2            3         1                              Heikkinen, Miss. Laina
3            4         1       Futrelle, Mrs. Jacques Heath (Lily May Peel)
4            5         0                           Allen, Mr. William Henry

      Sex   Age  SibSp  Parch            Ticket     Fare    Cabin Embarked
0    male  22.0      1      0         A/5 21171   7.2500  B96 B98        S
1  female  38.0      1      0          PC 17599  71.2833      C85        C
2  female  26.0      0      0  STON/O2. 3101282   7.9250  B96 B98        S
3  female  35.0      1      0            113803  53.1000     C123        S
4    male  35.0      0      0            373450   8.0500  B96 B98        S
```

[33]:
```
X.shape
```

[33]:
```
(849, 11)
```

[34]:
```
type(X)
```

[34]:
```
pandas.core.frame.DataFrame
```

[35]:
```
y=df["Embarked"]
y.head()
```

[35]:
```
0    S
1    C
2    S
3    S
4    S
Name: Embarked, dtype: object
```

3.1.2 Method-II

[36]:
```
x=df.iloc[:,3:13]
x
```

[36]:

	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	Braund, Mr. Owen Harris	male	22.000000	1	0	A/5 21171	7.2500	B96 B98	S
1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.000000	1	0	PC 17599	71.2833	C85	C
2	Heikkinen, Miss. Laina	female	26.000000	0	0	STON/O2. 3101282	7.9250	B96 B98	S
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.000000	1	0	113803	53.1000	C123	S
4	Allen, Mr. William Henry	male	35.000000	0	0	373450	8.0500	B96 B98	S
..	…	…	…	…	…	…	…	…	…
886	Montvila, Rev. Juozas	male	27.000000	0	0	211536	13.0000	B96 B98	S
887	Graham, Miss. Margaret Edith	female	19.000000	0	0	112053	30.0000	B42	S
888	Johnston, Miss. Catherine Helen "Carrie"	female	29.699118	1	2	W./C. 6607	23.4500	B96 B98	S
889	Behr, Mr. Karl Howell	male	26.000000	0	0	111369	30.0000	C148	C
890	Dooley, Mr. Patrick	male	32.000000	0	0	370376	7.7500	B96 B98	Q

[849 rows x 9 columns]

[37]:
```
y=df.iloc[:,13:14]
y
```

[37]:
```
Empty DataFrame
Columns: []
Index: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 16, 17, 18, 19, 20, 21,
22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 34, 35, 36, 37, 38, 39, 40, 41, 42,
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 55, 56, 57, 58, 59, 60, 61, 62, 63,
64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83,
84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 95, 97, 98, 99, 100, 101, 102, 103, 104,
105, …]

[849 rows x 0 columns]
```

[38]:
```
x.shape
```

[38]:
```
(849, 9)
```

[39]:
```
y.shape
```

[39]:
```
(849, 0)
```

4 Encoding

[40]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

[41]:
```
X["Sex"]=le.fit_transform(X["Sex"])
```

[42]:
```
X["Cabin"]=le.fit_transform(X["Cabin"])
```

[43]:
```
X.head()
```

[43]:

	PassengerId	Survived	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	Braund, Mr. Owen Harris	1	22.0	1	0	A/5 21171	7.2500	38	S
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	0	38.0	1	0	PC 17599	71.2833	69	C
2	3	1	Heikkinen, Miss. Laina	0	26.0	0	0	STON/O2. 3101282	7.9250	38	S
3	4	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	0	35.0	1	0	113803	53.1000	45	S
4	5	0	Allen, Mr. William Henry	1	35.0	0	0	373450	8.0500	38	S

[44]:
```
print(le.classes_)
```

```
['A10' 'A14' 'A16' 'A19' 'A20' 'A24' 'A31' 'A32' 'A34' 'A36' 'A6' 'B101'
 'B102' 'B18' 'B20' 'B22' 'B28' 'B3' 'B35' 'B38' 'B39' 'B4' 'B42' 'B49'
 'B5' 'B50' 'B51 B53 B55' 'B57 B59 B63 B66' 'B58 B60' 'B69' 'B71' 'B73'
 'B77' 'B78' 'B79' 'B82 B84' 'B86' 'B94' 'B96 B98' 'C101' 'C104' 'C106'
 'C110' 'C111' 'C118' 'C123' 'C124' 'C125' 'C126' 'C128' 'C148' 'C2'
 'C22 C26' 'C23 C25 C27' 'C32' 'C45' 'C46' 'C47' 'C49' 'C52' 'C54'
 'C62 C64' 'C65' 'C68' 'C7' 'C70' 'C78' 'C82' 'C83' 'C85' 'C86' 'C90'
 'C91' 'C92' 'C93' 'C95' 'C99' 'D' 'D10 D12' 'D11' 'D15' 'D17' 'D19' 'D20'
 'D21' 'D26' 'D28' 'D30' 'D33' 'D35' 'D36' 'D45' 'D46' 'D47' 'D49' 'D56'
 'D6' 'D9' 'E10' 'E101' 'E12' 'E121' 'E17' 'E24' 'E25' 'E31' 'E33' 'E34'
 'E36' 'E40' 'E44' 'E46' 'E49' 'E50' 'E58' 'E63' 'E67' 'E68' 'E8' 'F E69'
 'F G63' 'F G73' 'F2' 'F33' 'F38' 'F4' 'G6' 'T']
```

[45]:
```
mapping=dict(zip(le.classes_,range(len(le.classes_))))
mapping
```

[45]:
```
{'A10': 0,
 'A14': 1,
 'A16': 2,
 'A19': 3,
 'A20': 4,
 'A24': 5,
 'A31': 6,
 'A32': 7,
 'A34': 8,
 'A36': 9,
 'A6': 10,
 'B101': 11,
 'B102': 12,
 'B18': 13,
 'B20': 14,
 'B22': 15,
 'B28': 16,
 'B3': 17,
 'B35': 18,
 'B38': 19,
 'B39': 20,
 'B4': 21,
 'B42': 22,
 'B49': 23,
 'B5': 24,
 'B50': 25,
 'B51 B53 B55': 26,
 'B57 B59 B63 B66': 27,
 'B58 B60': 28,
 'B69': 29,
 'B71': 30,
```

```
'B73': 31,
'B77': 32,
'B78': 33,
'B79': 34,
'B82 B84': 35,
'B86': 36,
'B94': 37,
'B96 B98': 38,
'C101': 39,
'C104': 40,
'C106': 41,
'C110': 42,
'C111': 43,
'C118': 44,
'C123': 45,
'C124': 46,
'C125': 47,
'C126': 48,
'C128': 49,
'C148': 50,
'C2': 51,
'C22 C26': 52,
'C23 C25 C27': 53,
'C32': 54,
'C45': 55,
'C46': 56,
'C47': 57,
'C49': 58,
'C52': 59,
'C54': 60,
'C62 C64': 61,
'C65': 62,
'C68': 63,
'C7': 64,
'C70': 65,
'C78': 66,
'C82': 67,
'C83': 68,
'C85': 69,
'C86': 70,
'C90': 71,
'C91': 72,
'C92': 73,
'C93': 74,
'C95': 75,
'C99': 76,
'D': 77,
```

```
'D10 D12': 78,
'D11': 79,
'D15': 80,
'D17': 81,
'D19': 82,
'D20': 83,
'D21': 84,
'D26': 85,
'D28': 86,
'D30': 87,
'D33': 88,
'D35': 89,
'D36': 90,
'D45': 91,
'D46': 92,
'D47': 93,
'D49': 94,
'D56': 95,
'D6': 96,
'D9': 97,
'E10': 98,
'E101': 99,
'E12': 100,
'E121': 101,
'E17': 102,
'E24': 103,
'E25': 104,
'E31': 105,
'E33': 106,
'E34': 107,
'E36': 108,
'E40': 109,
'E44': 110,
'E46': 111,
'E49': 112,
'E50': 113,
'E58': 114,
'E63': 115,
'E67': 116,
'E68': 117,
'E8': 118,
'F E69': 119,
'F G63': 120,
'F G73': 121,
'F2': 122,
'F33': 123,
'F38': 124,
```

```
 'F4': 125,
 'G6': 126,
 'T': 127}
```

5 Feature Scaling

[46]:
```
from sklearn.preprocessing import MinMaxScaler
ms=MinMaxScaler()
```

[47]:
```
df.dtypes
```

[47]:
```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

[48]:
```
X.dtypes
```

[48]:
```
PassengerId      int64
Survived         int64
Name            object
Sex              int32
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin            int32
Embarked        object
dtype: object
```

6 Splitting Data into Train and Test Dataset

[49]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=0)
```

[50]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

[50]: ((594, 9), (255, 9), (594, 0), (255, 0))