EDA

May 29, 2025

[4]:
```
import pandas as pd # Read data from CSV to a data frame named df
df = pd.read_csv('titanic.csv')
# Display the data
df
```

[4]:
```
     Survived  Pclass                                               Name  \
0           0       3                            Mr. Owen Harris Braund
1           1       1  Mrs. John Bradley (Florence Briggs Thayer) Cum…
2           1       3                       Miss. Laina Heikkinen
3           1       1       Mrs. Jacques Heath (Lily May Peel) Futrelle
4           0       3                           Mr. William Henry Allen
..        …     …                                                  …
882         0       2                          Rev. Juozas Montvila
883         1       1                   Miss. Margaret Edith Graham
884         0       3                  Miss. Catherine Helen Johnston
885         1       1                           Mr. Karl Howell Behr
886         0       3                             Mr. Patrick Dooley

        Sex   Age  Siblings/Spouses Aboard  Parents/Children Aboard     Fare
0      male  22.0                        1                        0   7.2500
1    female  38.0                        1                        0  71.2833
2    female  26.0                        0                        0   7.9250
3    female  35.0                        1                        0  53.1000
4      male  35.0                        0                        0   8.0500
..      …   …                         …                      …       …
882    male  27.0                        0                        0  13.0000
883  female  19.0                        0                        0  30.0000
884  female   7.0                        1                        2  23.4500
885    male  26.0                        0                        0  30.0000
886    male  32.0                        0                        0   7.7500

[887 rows x 8 columns]
```

[5]:
```
df.head(5)
```

[5]:
```
   Survived  Pclass                                               Name  \
0         0       3                            Mr. Owen Harris Braund
1         1       1  Mrs. John Bradley (Florence Briggs Thayer) Cum…
```

```
2         1       3                              Miss. Laina Heikkinen
3         1       1        Mrs. Jacques Heath (Lily May Peel) Futrelle
4         0       3                            Mr. William Henry Allen

      Sex   Age  Siblings/Spouses Aboard  Parents/Children Aboard     Fare
0    male  22.0                        1                        0   7.2500
1  female  38.0                        1                        0  71.2833
2  female  26.0                        0                        0   7.9250
3  female  35.0                        1                        0  53.1000
4    male  35.0                        0                        0   8.0500
```

[6]:
```
df.tail(30)
```

[6]:
```
     Survived  Pclass                                               Name  \
857         0       2                            Mr. Frederick Edward Giles
858         1       1  Mrs. Frederick Joel (Margaret Welles Barron) S…
859         0       3                             Miss. Dorothy Edith Sage
860         0       2                               Mr. John William Gill
861         1       2                             Mrs. (Karolina) Bystrom
862         1       2                          Miss. Asuncion Duran y More
863         0       1                 Mr. Washington Augustus II Roebling
864         0       3                           Mr. Philemon van Melkebeke
865         1       3                      Master. Harold Theodor Johnson
866         0       3                                   Mr. Cerin Balkic
867         1       1     Mrs. Richard Leonard (Sallie Monypeny) Beckwith
868         0       1                              Mr. Frans Olof Carlsson
869         0       3                           Mr. Victor Vander Cruyssen
870         1       2                Mrs. Samuel (Hannah Wizosky) Abelson
871         1       3                             Miss. Adele Kiamie Najib
872         0       3                        Mr. Alfred Ossian Gustafsson
873         0       3                                 Mr. Nedelio Petroff
874         0       3                                   Mr. Kristo Laleff
875         1       1        Mrs. Thomas Jr (Lily Alexenia Wilson) Potter
876         1       2         Mrs. William (Imanita Parrish Hall) Shelley
877         0       3                                   Mr. Johann Markun
878         0       3                          Miss. Gerda Ulrika Dahlberg
879         0       2                        Mr. Frederick James Banfield
880         0       3                               Mr. Henry Jr Sutehall
881         0       3                 Mrs. William (Margaret Norton) Rice
882         0       2                                Rev. Juozas Montvila
883         1       1                          Miss. Margaret Edith Graham
884         0       3                       Miss. Catherine Helen Johnston
885         1       1                                Mr. Karl Howell Behr
886         0       3                                  Mr. Patrick Dooley

        Sex   Age  Siblings/Spouses Aboard  Parents/Children Aboard     Fare
857    male  21.0                        1                        0  11.5000
```

```
 858  female  48.0                     0                     0  25.9292
 859  female  14.0                     8                     2  69.5500
 860    male  24.0                     0                     0  13.0000
 861  female  42.0                     0                     0  13.0000
 862  female  27.0                     1                     0  13.8583
 863    male  31.0                     0                     0  50.4958
 864    male  23.0                     0                     0   9.5000
 865    male   4.0                     1                     1  11.1333
 866    male  26.0                     0                     0   7.8958
 867  female  47.0                     1                     1  52.5542
 868    male  33.0                     0                     0   5.0000
 869    male  47.0                     0                     0   9.0000
 870  female  28.0                     1                     0  24.0000
 871  female  15.0                     0                     0   7.2250
 872    male  20.0                     0                     0   9.8458
 873    male  19.0                     0                     0   7.8958
 874    male  23.0                     0                     0   7.8958
 875  female  56.0                     0                     1  83.1583
 876  female  25.0                     0                     1  26.0000
 877    male  33.0                     0                     0   7.8958
 878  female  22.0                     0                     0  10.5167
 879    male  28.0                     0                     0  10.5000
 880    male  25.0                     0                     0   7.0500
 881  female  39.0                     0                     5  29.1250
 882    male  27.0                     0                     0  13.0000
 883  female  19.0                     0                     0  30.0000
 884  female   7.0                     1                     2  23.4500
 885    male  26.0                     0                     0  30.0000
 886    male  32.0                     0                     0   7.7500
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 887 entries, 0 to 886
Data columns (total 8 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Survived                 887 non-null    int64
 1   Pclass                   887 non-null    int64
 2   Name                     887 non-null    object
 3   Sex                      887 non-null    object
 4   Age                      887 non-null    float64
 5   Siblings/Spouses Aboard  887 non-null    int64
 6   Parents/Children Aboard  887 non-null    int64
 7   Fare                     887 non-null    float64
dtypes: float64(2), int64(4), object(2)
memory usage: 55.6+ KB
```

[8]:
```
df.columns
```

[8]:
```
Index(['Survived', 'Pclass', 'Name', 'Sex', 'Age', 'Siblings/Spouses Aboard',
       'Parents/Children Aboard', 'Fare'],
      dtype='object')
```

[9]:
```
# Rename 2 columns: Siblings/Spouses Aboard and Parents/Children Aboard
df.rename(columns = {
 'Siblings/Spouses Aboard': 'Siblings_Spouses',
 'Parents/Children Aboard': 'Parents_Children'
 }, inplace=True)
# Display column names
df.columns
```

[9]:
```
Index(['Survived', 'Pclass', 'Name', 'Sex', 'Age', 'Siblings_Spouses',
       'Parents_Children', 'Fare'],
      dtype='object')
```

[10]:
```
# Option 1: Set the names to each column in a list - change the column names to␣
 ↪lowercase
df.columns = ['survived', 'pclass', 'name', 'sex', 'age', 'siblings_spouses',␣
 ↪'parents_children', 'fare']
df.columns
# Option 2: Can also use a for loop to access each element in the column name␣
 ↪list
df.columns = [col.lower() for col in df]
df.columns
```

[10]:
```
Index(['survived', 'pclass', 'name', 'sex', 'age', 'siblings_spouses',
       'parents_children', 'fare'],
      dtype='object')
```

[11]:
```
# Specify the desired columns in a new selected_col list
selected_col = ['survived', 'pclass', 'sex', 'age', 'fare']
# Create a new data frame to store data from selected columns
df_new = df[selected_col]
# Display the first 5 rows in df_new
df_new.head()
```

[11]:

	survived	pclass	sex	age	fare
0	0	3	male	22.0	7.2500
1	1	1	female	38.0	71.2833
2	1	3	female	26.0	7.9250
3	1	1	female	35.0	53.1000
4	0	3	male	35.0	8.0500

[12]:
```
# Display the data types of each column
df_new.dtypes
```

[12]:
```
survived      int64
pclass        int64
sex          object
age         float64
fare        float64
dtype: object
```

[13]:
```
df.describe()
```

[13]:

	survived	pclass	age	siblings_spouses	parents_children	fare
count	887.000000	887.000000	887.000000	887.000000	887.000000	887.00000
mean	0.385569	2.305524	29.471443	0.525366	0.383315	32.30542
std	0.487004	0.836662	14.121908	1.104669	0.807466	49.78204
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.00000
25%	0.000000	2.000000	20.250000	0.000000	0.000000	7.92500
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.45420
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.13750
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.32920

[14]:
```
df.groupby('pclass').mean(numeric_only=True)
```

[14]:

pclass	survived	age	siblings_spouses	parents_children	fare
1	0.629630	38.788981	0.416667	0.356481	84.154687
2	0.472826	29.868641	0.402174	0.380435	20.662183
3	0.244353	25.188747	0.620123	0.396304	13.707707

[15]:
```
# Show basic statistics for categorical variables
df['sex'].describe()
```

[15]:
```
count      887
unique       2
top       male
freq       573
Name: sex, dtype: object
```

[16]:
```
# Count the occurrences of each class
df['sex'].value_counts()
```

[16]:
```
sex
male      573
female    314
Name: count, dtype: int64
```

[17]:
```
# Convert integer to string: survived
df['survived'] = df['survived'].astype(str)
df['survived'].describe()
```

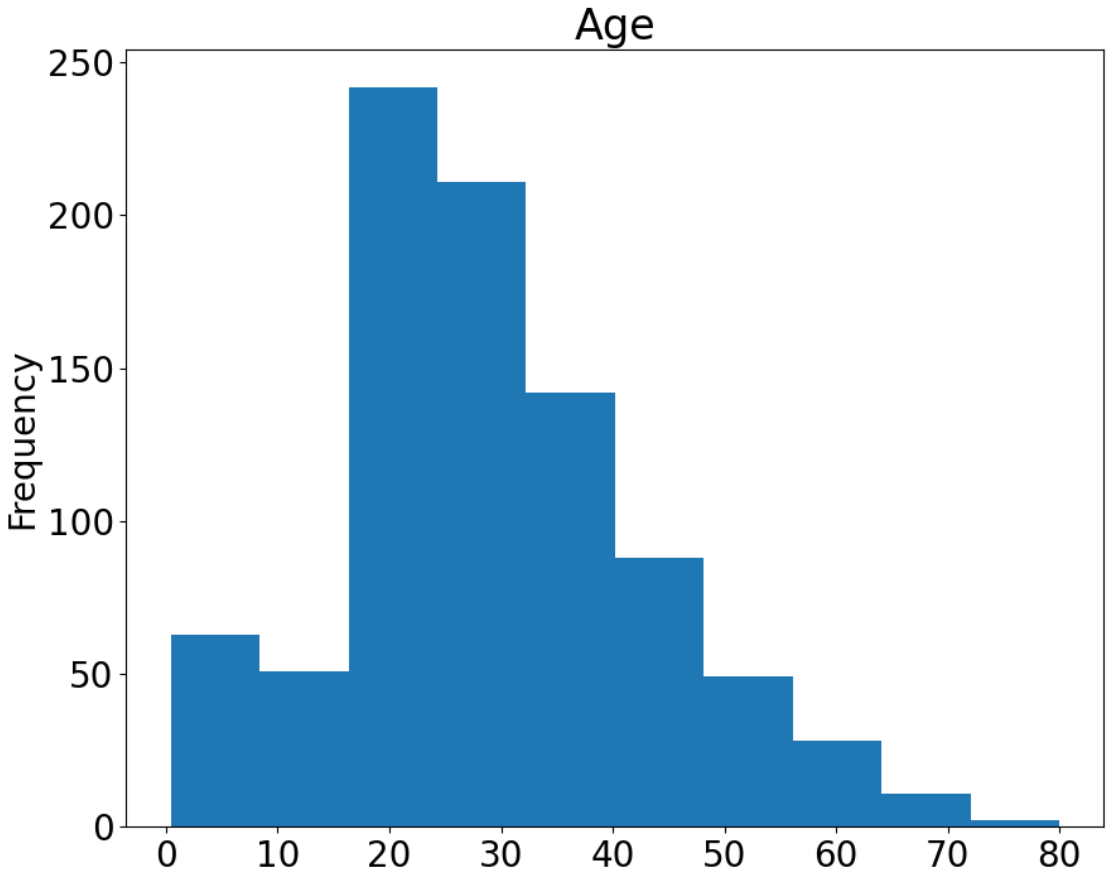
[17]:
```
count     887
unique      2
top         0
freq      545
Name: survived, dtype: object
```

[18]:
```
# Convert integer to string: pclass
df['pclass'] = df['pclass'].astype(str)
df['pclass'].describe()
```

[18]:
```
count     887
unique      3
top         3
freq      487
Name: pclass, dtype: object
```

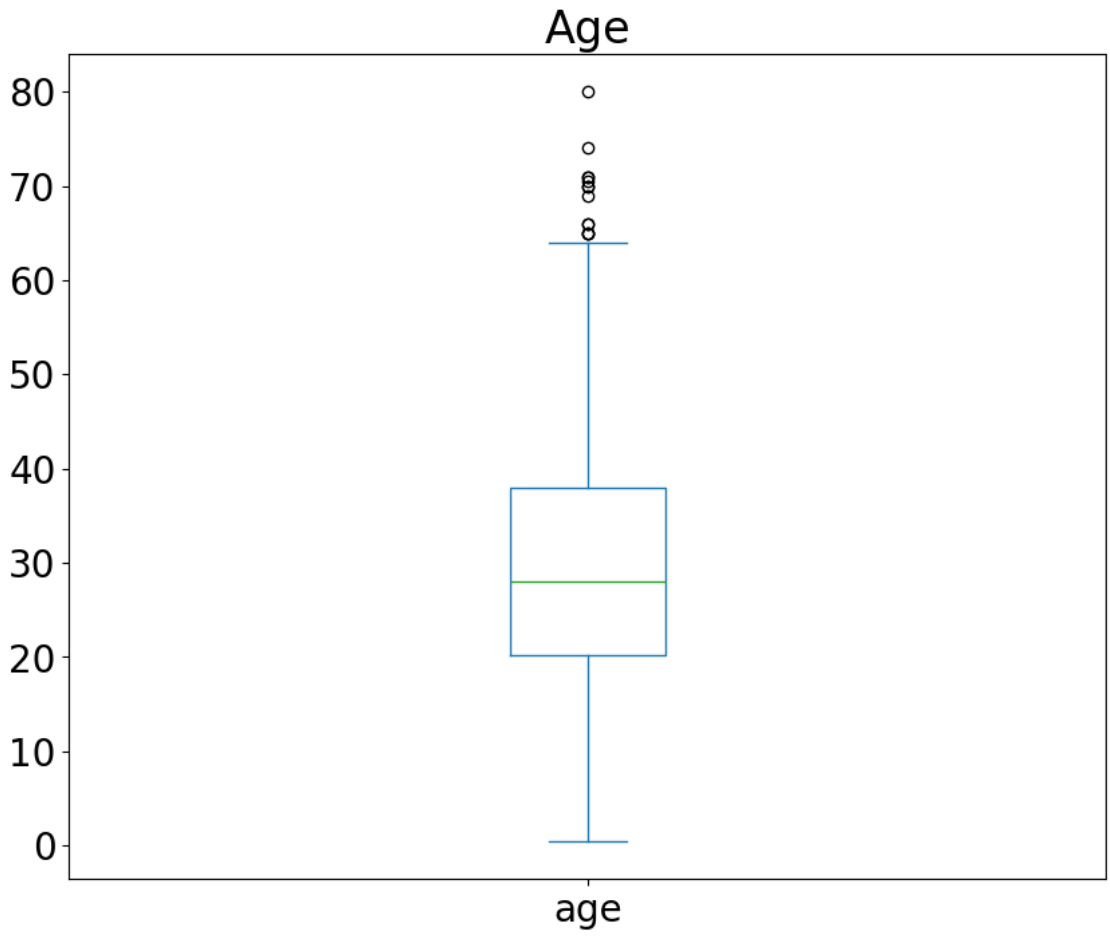
[19]:
```
# Import matplotlib library
import matplotlib.pyplot as plt
# Set font and plot size to be larger
plt.rcParams.update({'font.size': 20, 'figure.figsize': (10, 8)})
# Plot a histogram to study the distribution of age
df['age'].plot(kind='hist', title='Age')
```

[19]:
```
<Axes: title={'center': 'Age'}, ylabel='Frequency'>
```



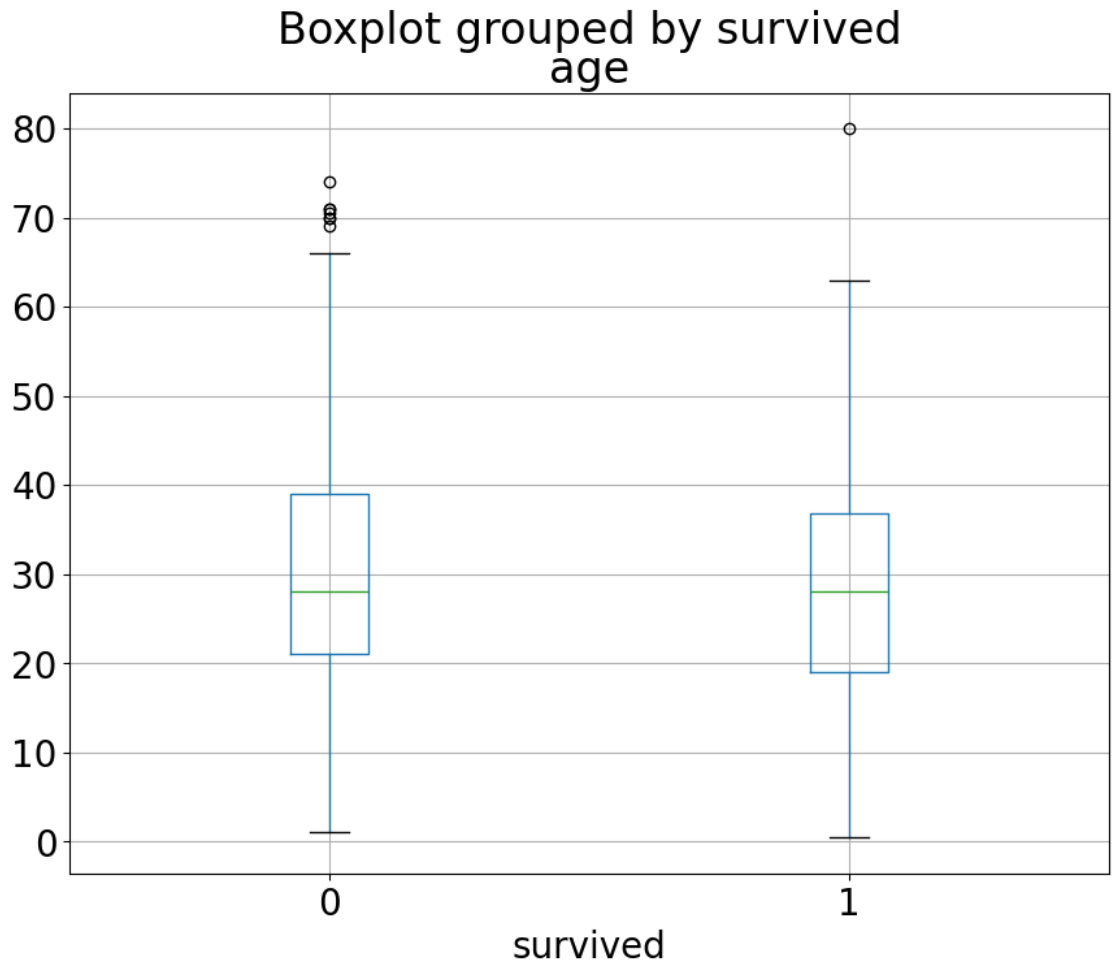
[20]:
```
df['age'].plot(kind='box', title='Age')
```

[20]: <Axes: title={'center': 'Age'}>



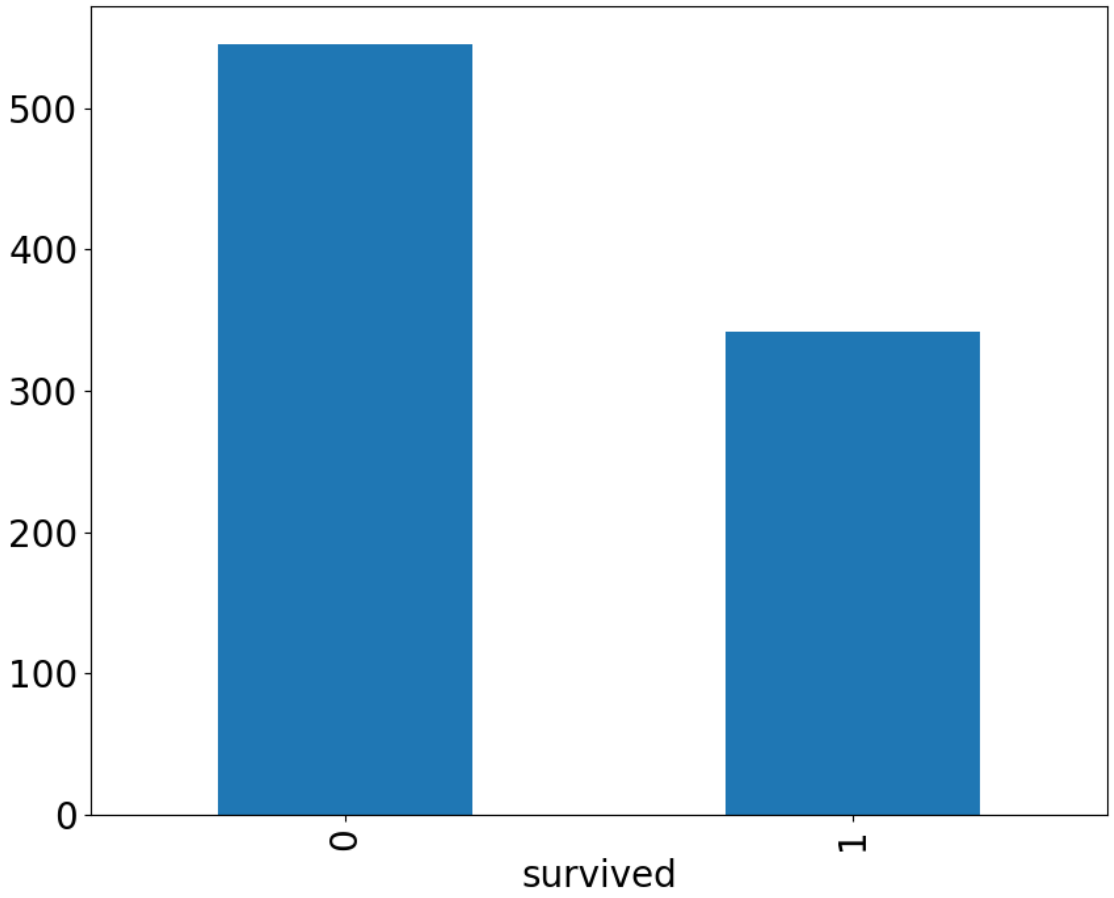
[21]:
```
df.boxplot(column='age', by='survived')
```

[21]: <Axes: title={'center': 'age'}, xlabel='survived'>



[22]:
```
# Plot a bar chart based on count of each label
df['survived'].value_counts().plot(kind='bar')
```

[22]: <Axes: xlabel='survived'>



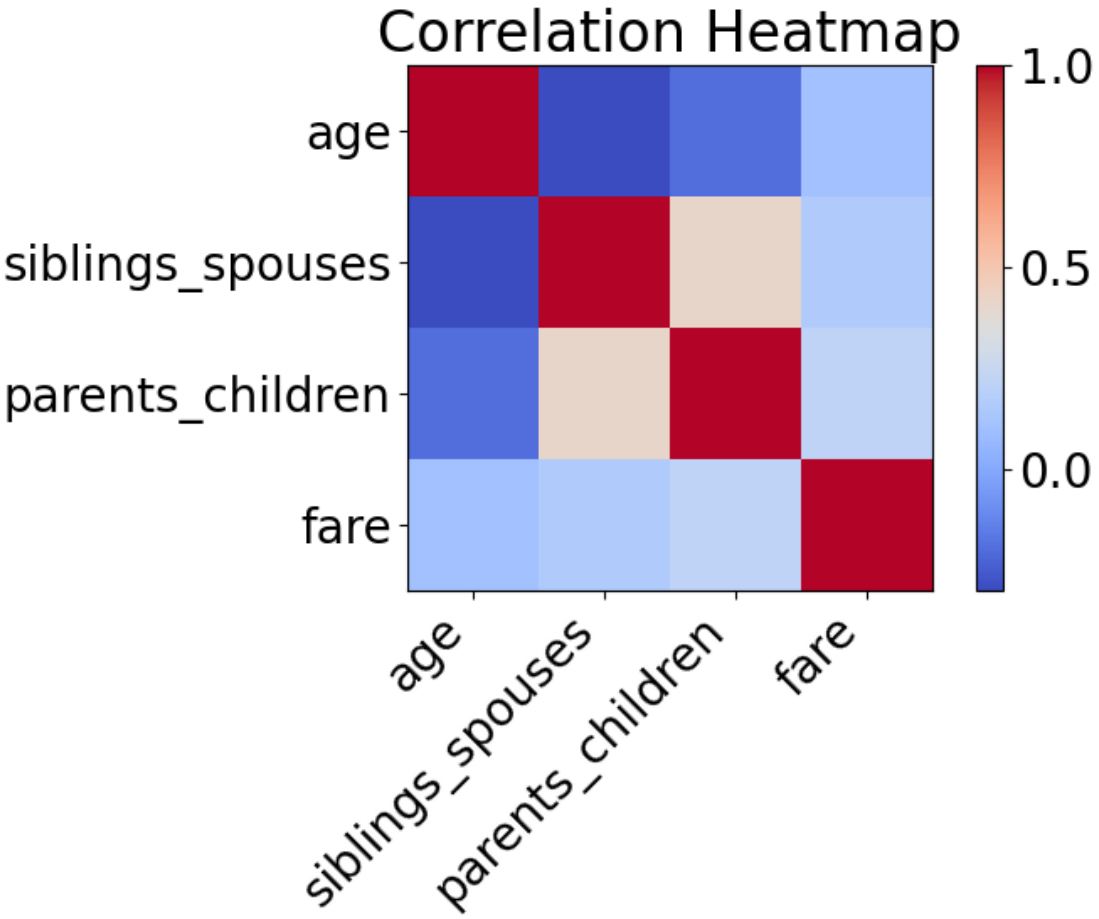
[23]:
```
df_numeric = df.select_dtypes(include='number')
correlation_matrix = df_numeric.corr()
correlation_matrix
```

[23]:

	age	siblings_spouses	parents_children	fare
age	1.000000	-0.297669	-0.193741	0.112329
siblings_spouses	-0.297669	1.000000	0.414244	0.158839
parents_children	-0.193741	0.414244	1.000000	0.215470
fare	0.112329	0.158839	0.215470	1.000000

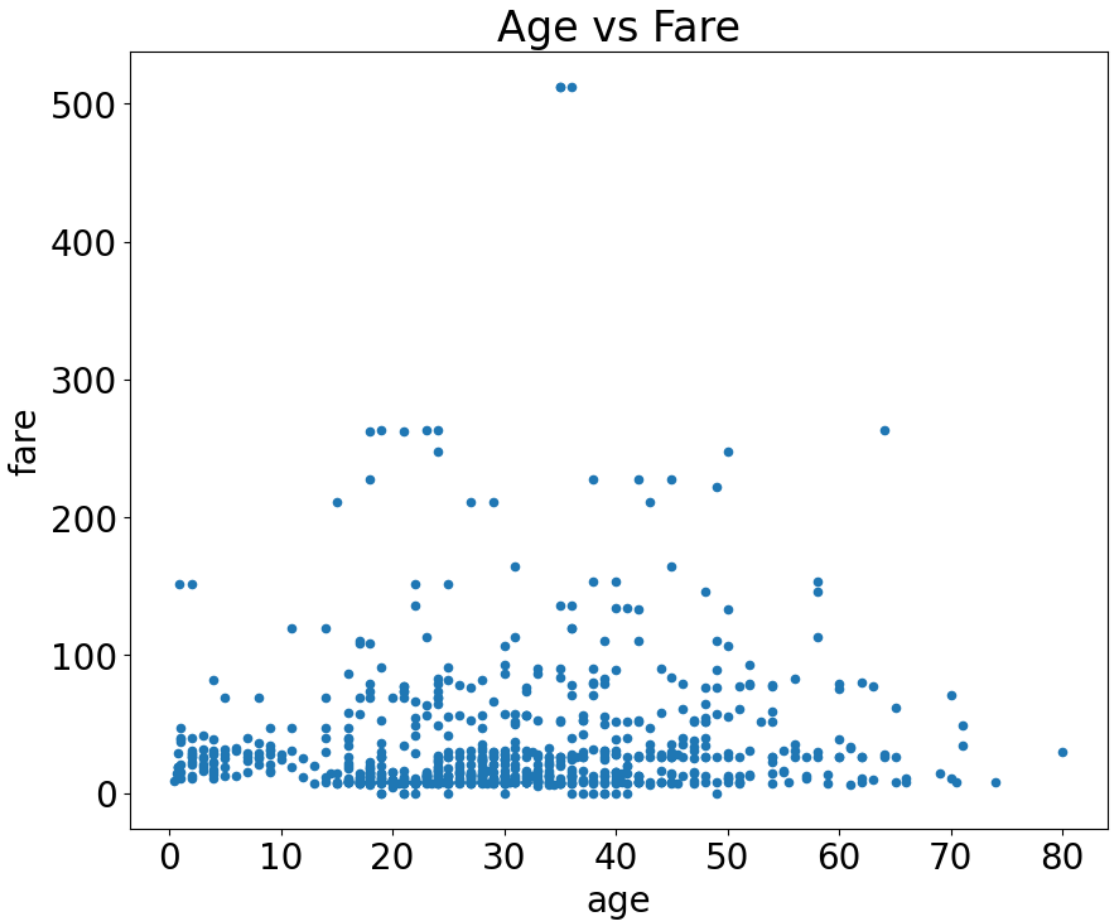
[24]:
```
# Plot heatmap
plt.figure(figsize=(8, 6))
plt.imshow(correlation_matrix, cmap='coolwarm', interpolation='none')
plt.colorbar()

# Set ticks and labels
plt.xticks(range(len(correlation_matrix.columns)), correlation_matrix.columns,␣
 ↪rotation=45, ha='right')
```

```
plt.yticks(range(len(correlation_matrix.columns)), correlation_matrix.columns)

plt.title('Correlation Heatmap')
plt.tight_layout()
plt.show()
```



[25]:
```
# Plot a scatterplot based on two variables: age and fare
df.plot(kind='scatter', x='age', y='fare', title='Age vs Fare')
```

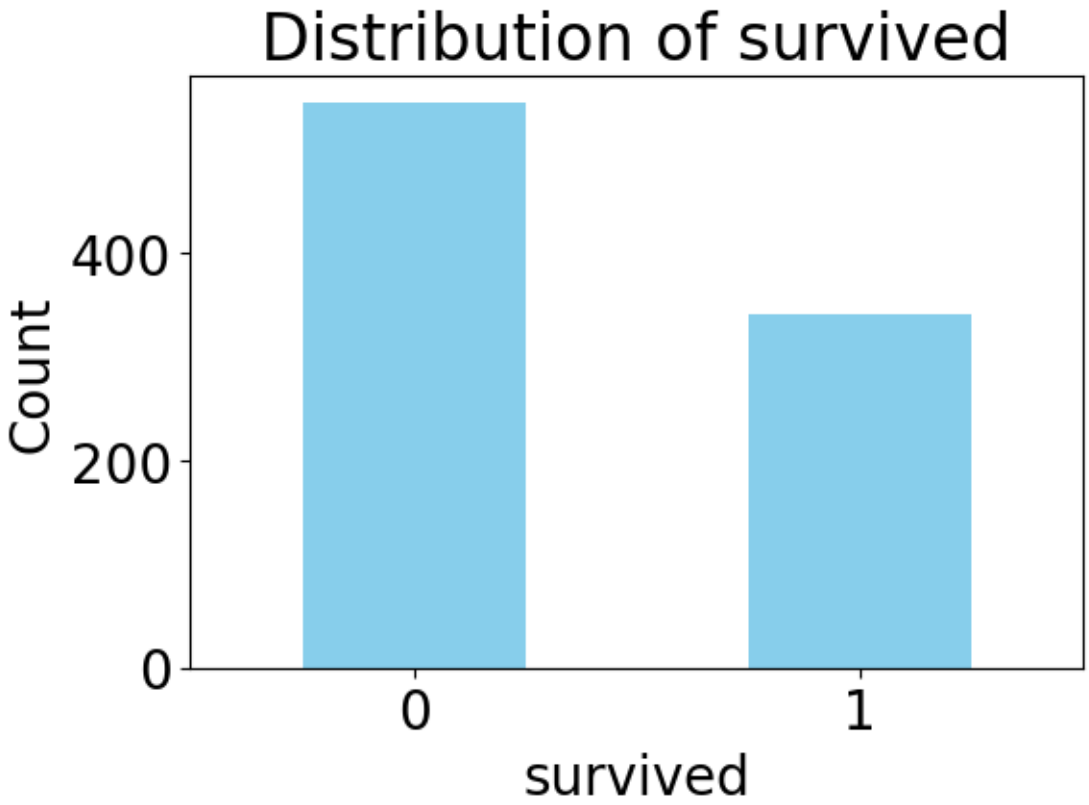
[25]: <Axes: title={'center': 'Age vs Fare'}, xlabel='age', ylabel='fare'>



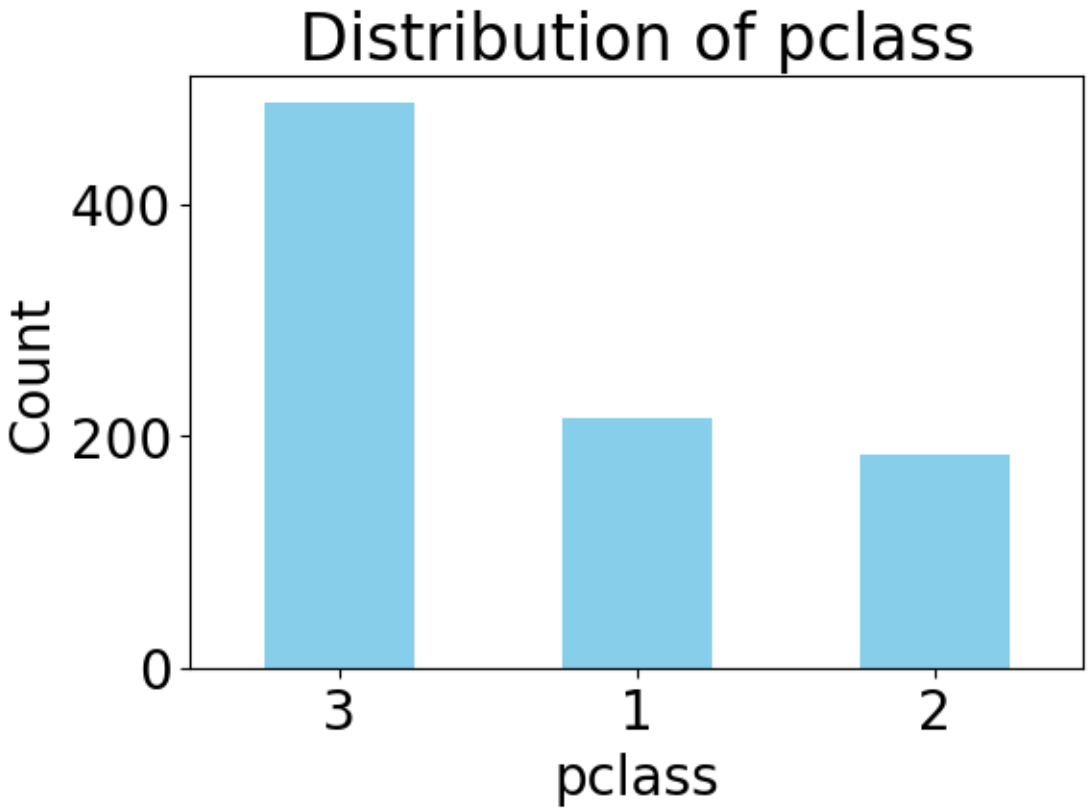
[26]:
```
#Univariate Analysis :
# 1. Categorical variables: survived, pclass, sex, sibsp, parch, embarked
categorical_cols = ['survived', 'pclass', 'sex', 'siblings_spouses',␣
 ↪'parents_children']

for col in categorical_cols:
    print(f"\nValue counts for {col}:\n", df[col].value_counts(dropna=False))
    plt.figure(figsize=(6,4))
    df[col].value_counts(dropna=False).plot(kind='bar', color='skyblue')
    plt.title(f'Distribution of {col}')
    plt.xlabel(col)
    plt.ylabel('Count')
    plt.xticks(rotation=0)
    plt.show()

# 2. Numerical variables: fare, age
numerical_cols = ['fare', 'age']
```

```python
for col in numerical_cols:
    print(f"\nSummary statistics for {col}:\n", df[col].describe())
    plt.figure(figsize=(12,4))

    # Histogram
    plt.subplot(1, 2, 1)
    df[col].hist(bins=30, color='lightgreen')
    plt.title(f'Histogram of {col}')
    plt.xlabel(col)
    plt.ylabel('Frequency')

    # Boxplot
    plt.subplot(1, 2, 2)
    df.boxplot(column=col)
    plt.title(f'Boxplot of {col}')

    plt.tight_layout()
    plt.show()
```

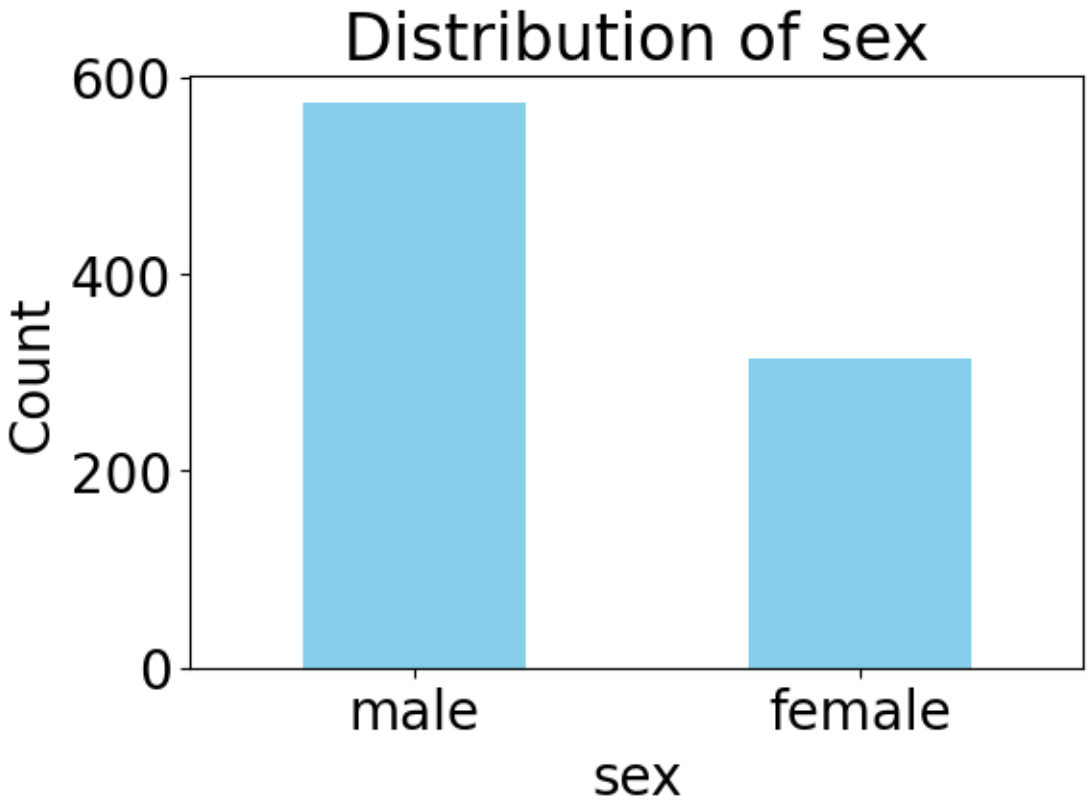
```
Value counts for survived:
 survived
0    545
1    342
Name: count, dtype: int64
```



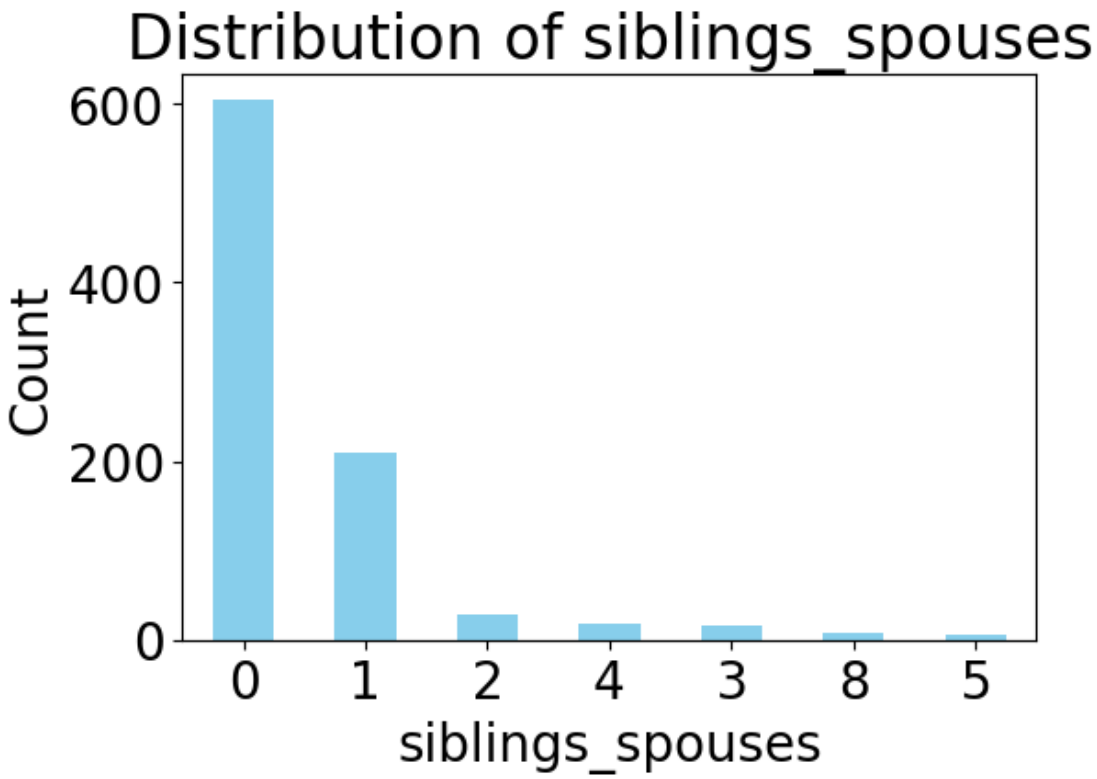

```
Value counts for pclass:
 pclass
3    487
1    216
2    184
Name: count, dtype: int64
```



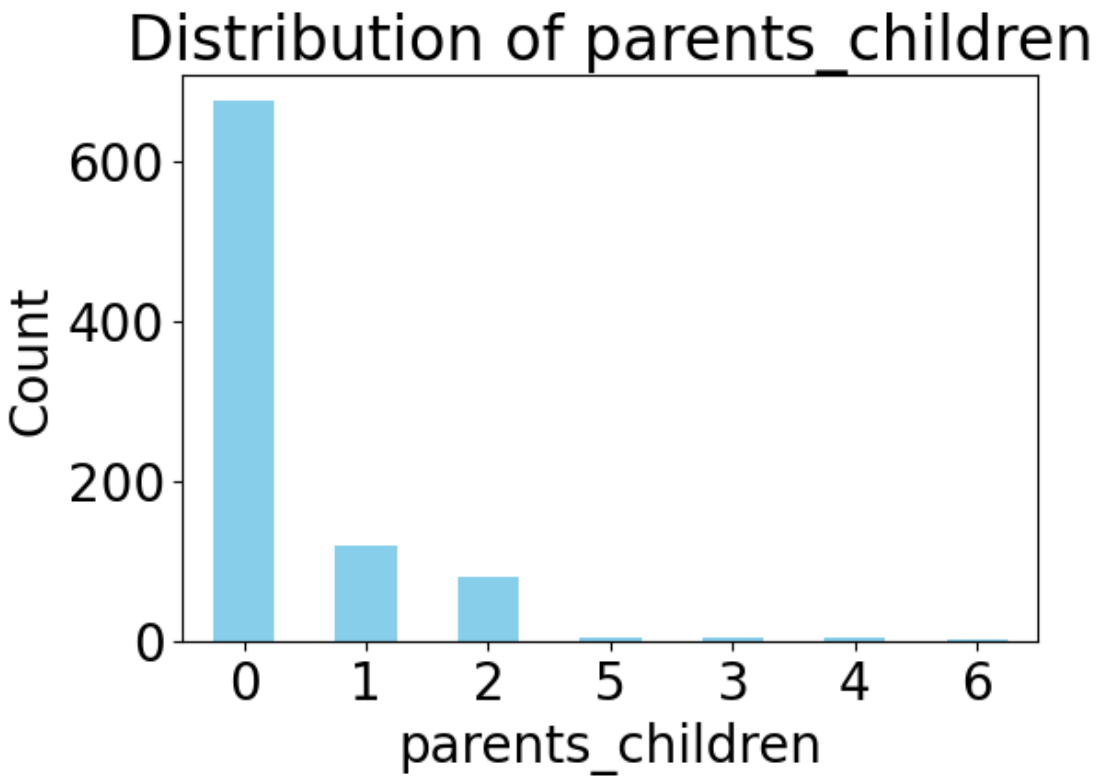

```
Value counts for sex:
 sex
male      573
female    314
Name: count, dtype: int64
```



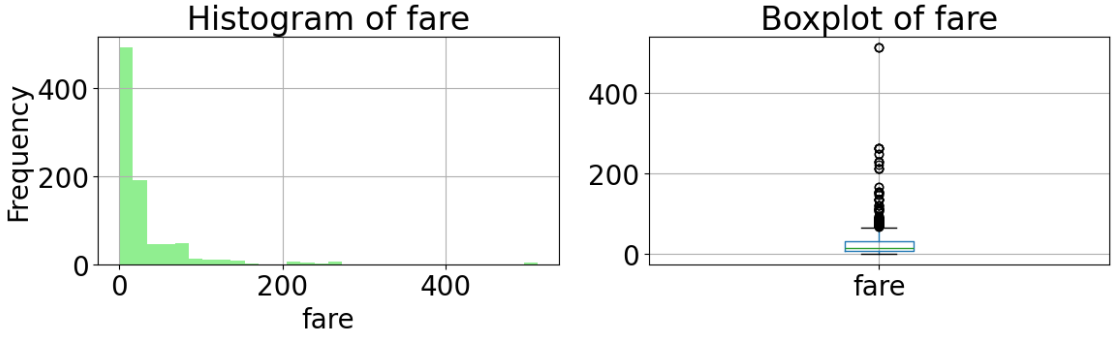

```
Value counts for siblings_spouses:
 siblings_spouses
0    604
1    209
2     28
4     18
3     16
8      7
5      5
Name: count, dtype: int64
```



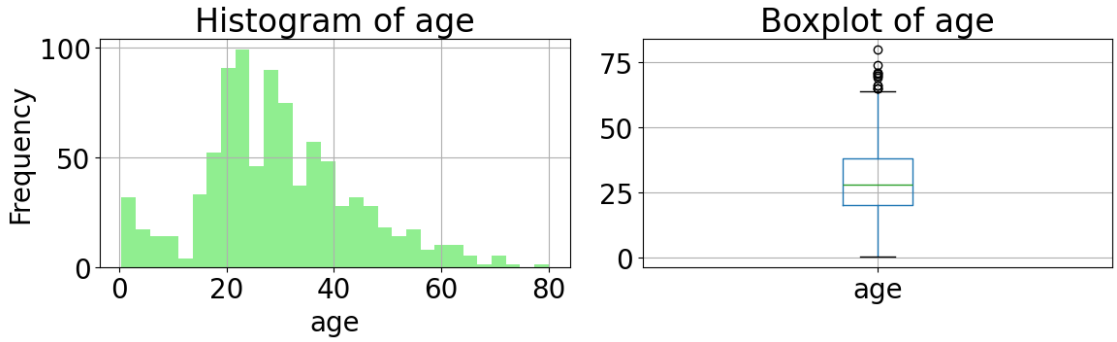

```
Value counts for parents_children:
 parents_children
0    674
1    118
2     80
5      5
3      5
4      4
6      1
Name: count, dtype: int64
```




```
Summary statistics for fare:
 count    887.00000
mean      32.30542
std       49.78204
min        0.00000
25%        7.92500
50%       14.45420
75%       31.13750
max      512.32920
Name: fare, dtype: float64
```



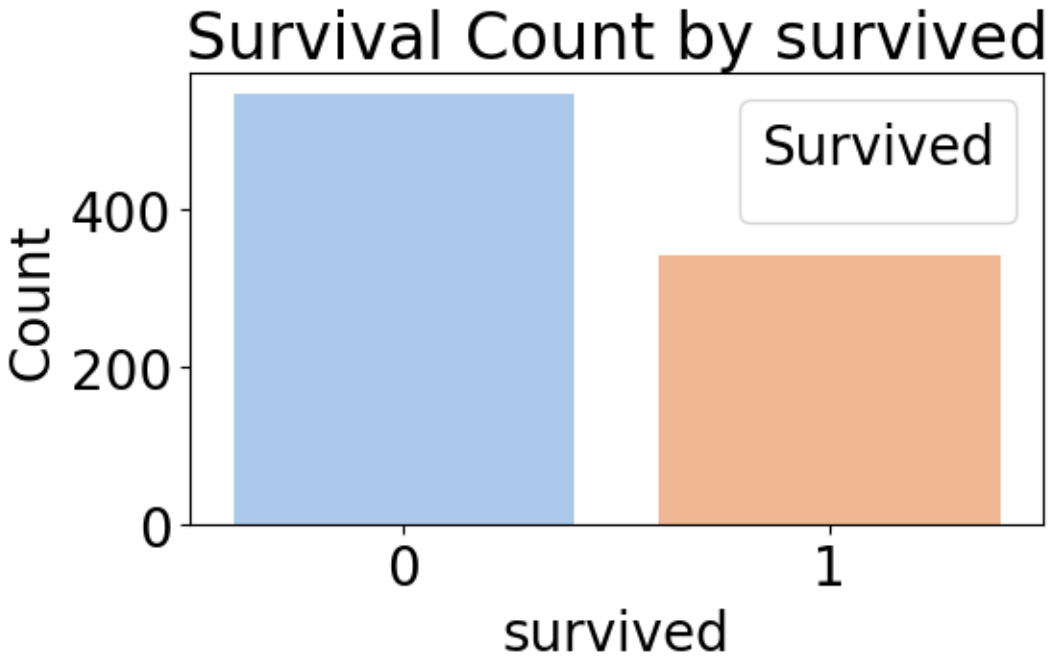
```
Summary statistics for age:
 count    887.000000
mean      29.471443
std       14.121908
min        0.420000
25%       20.250000
50%       28.000000
75%       38.000000
max       80.000000
Name: age, dtype: float64
```



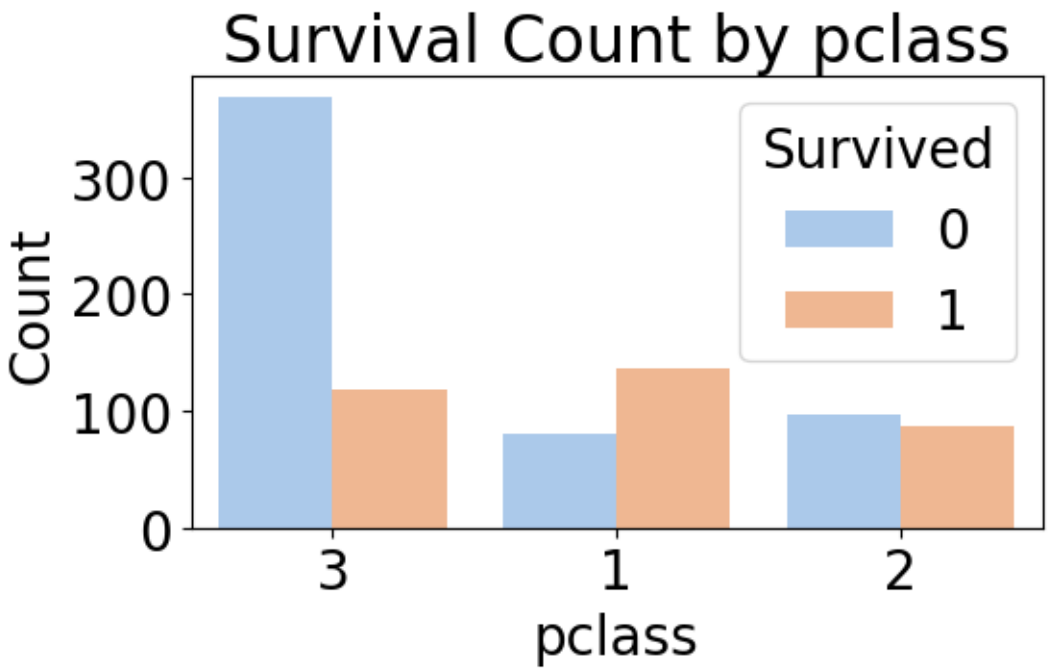
[28]:
```
import pandas as pd
import seaborn as sns
# 1. Categorical variables: survived, pclass, sex, sibsp, parch, embarked
categorical_cols = ['survived', 'pclass', 'sex', 'siblings_spouses',␣
 ↪'parents_children']
for col in categorical_cols:
    print(f"\nSurvival Rate by {col}:\n", pd.crosstab(df[col], df['survived'],␣
 ↪normalize='index'))

    plt.figure(figsize=(6,4))
    sns.countplot(data=df, x=col, hue='survived', palette='pastel')
    plt.title(f'Survival Count by {col}')
    plt.xlabel(col)
    plt.ylabel('Count')
    plt.xticks(rotation=0)
    plt.legend(title='Survived', loc='upper right')
    plt.tight_layout()
    plt.show()
```

```
Survival Rate by survived:
 survived    0    1
survived
0         1.0  0.0
1         0.0  1.0
```

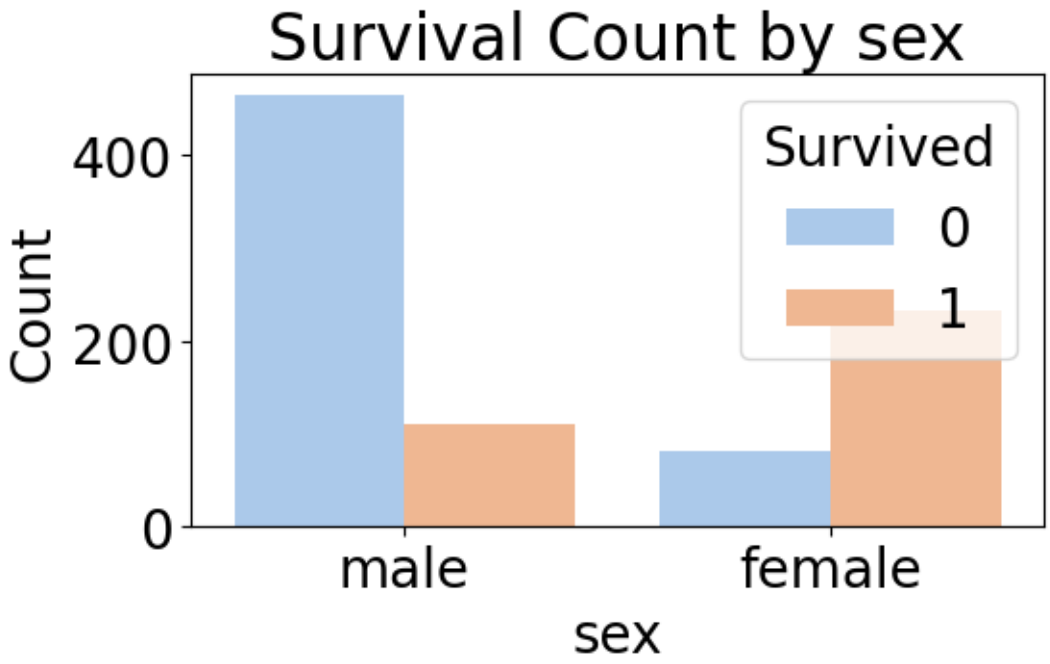
```
C:\Users\kelvchin\AppData\Local\Temp\ipykernel_46052\3267605463.py:14:
UserWarning: No artists with labels found to put in legend.  Note that artists
whose label start with an underscore are ignored when legend() is called with no
argument.
  plt.legend(title='Survived', loc='upper right')
```



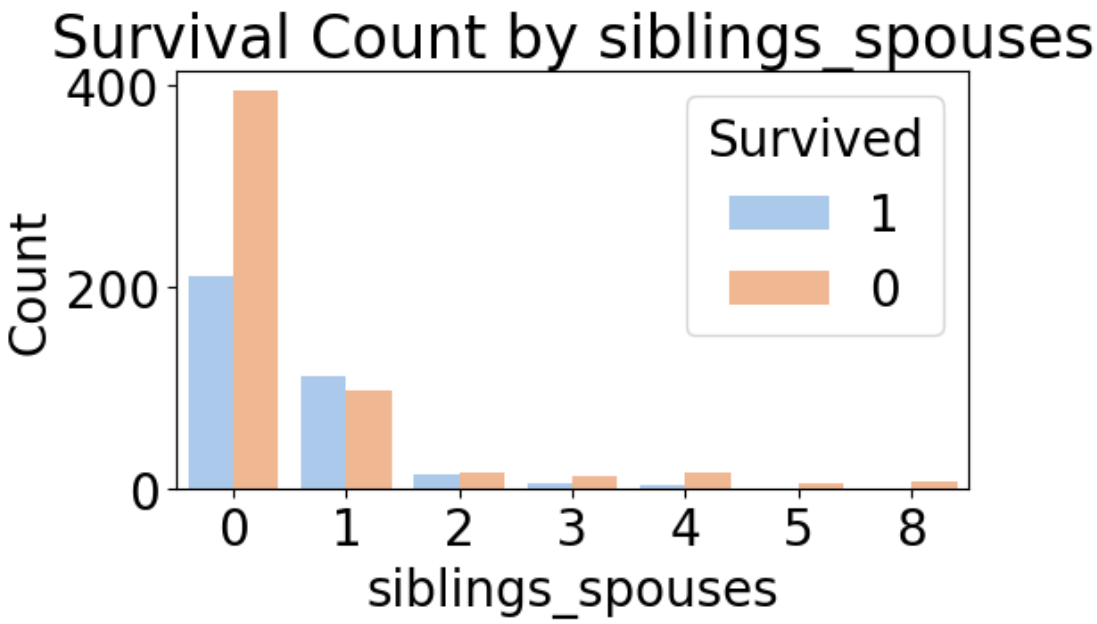

```
Survival Rate by pclass:
 survived         0         1
pclass
1         0.370370  0.629630
2         0.527174  0.472826
3         0.755647  0.244353
```




```
Survival Rate by sex:
 survived          0         1
sex
female     0.257962  0.742038
male       0.809773  0.190227
```

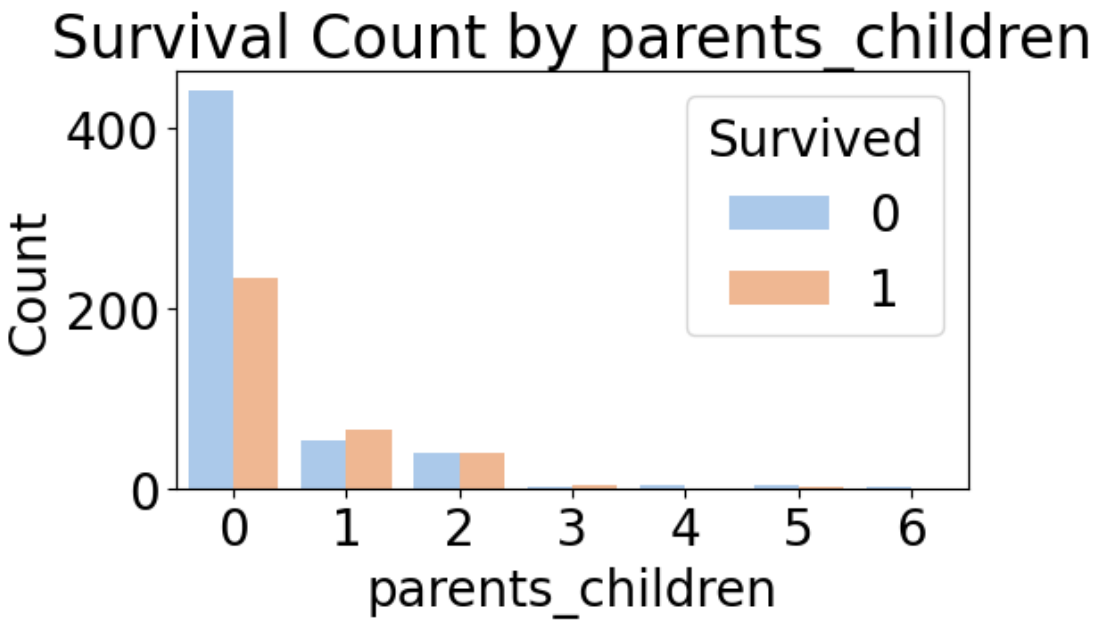



```
Survival Rate by siblings_spouses:
 survived                 0         1
siblings_spouses
0                 0.652318  0.347682
1                 0.464115  0.535885
2                 0.535714  0.464286
3                 0.750000  0.250000
4                 0.833333  0.166667
5                 1.000000  0.000000
8                 1.000000  0.000000
```

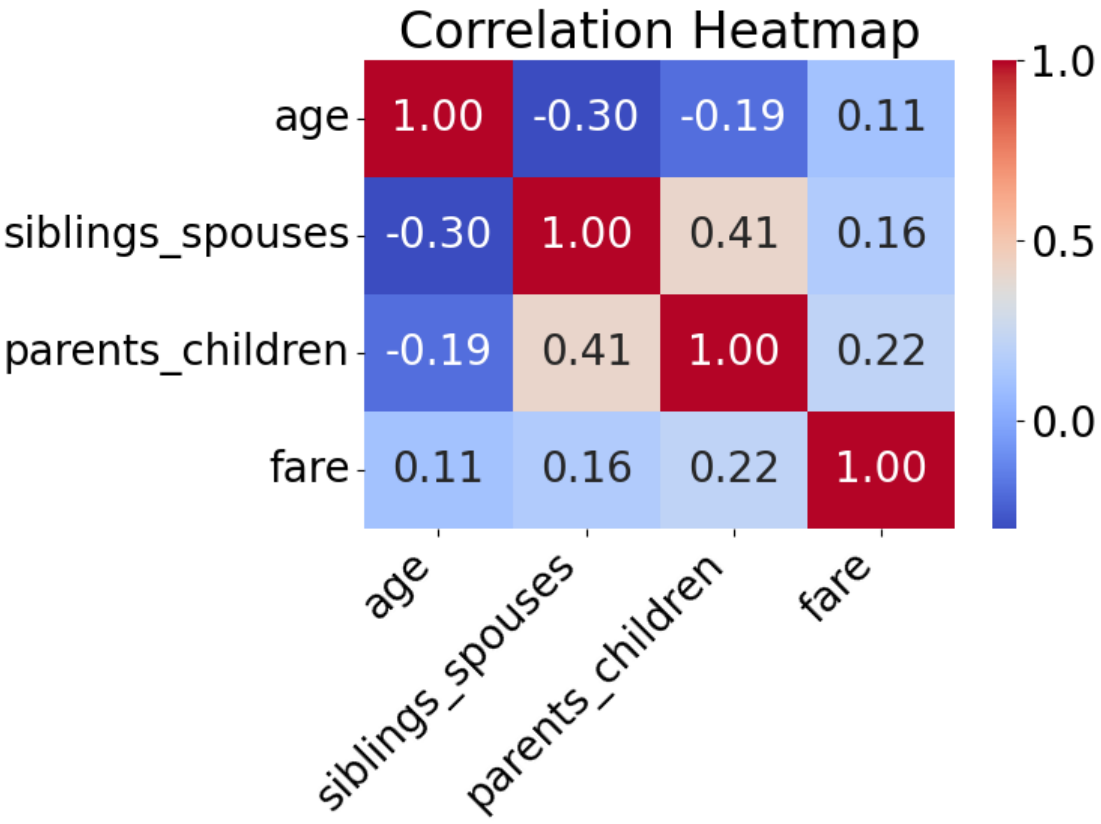


Survival Rate by parents_children:

survived	0	1
parents_children		
0	0.654303	0.345697
1	0.449153	0.550847
2	0.500000	0.500000
3	0.400000	0.600000
4	1.000000	0.000000
5	0.800000	0.200000
6	1.000000	0.000000



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

plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")

plt.title('Correlation Heatmap')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```


Correlation Heatmap
age
siblings_spouses
parents_children
fare
1.00
-0.30
-0.19
0.11
-0.30
1.00
0.41
0.16
-0.19
0.41
1.00
0.22
0.11
0.16
0.22
1.00
1.0
0.5
0.0
age
siblings_spouses
parents_children
fare