Titanic Dataset EDA

Importing the dataset

```
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)
warnings.filterwarnings("ignore")

import pandas as pd
titanic_df= pd.read_csv("C:/Users/arunj/Downloads/Titanic-
Dataset.csv")
print(titanic_df)
titanic_df.columns
```

```
     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]

Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age',
'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

Exploring the summary statitics

Numeric columns

```
titanic_df.describe()

       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
```

Categorical Columns

```
titanic_df.describe(include=['object'])
```

```
                           Name   Sex  Ticket Cabin Embarked
count                       891   891     891   204      889
unique                      891     2     681   147        3
top     Dooley, Mr. Patrick  male  347082    G6        S
freq                          1   577       7     4      644
```

Visualizations

Categorical Columns

1. Survived Column

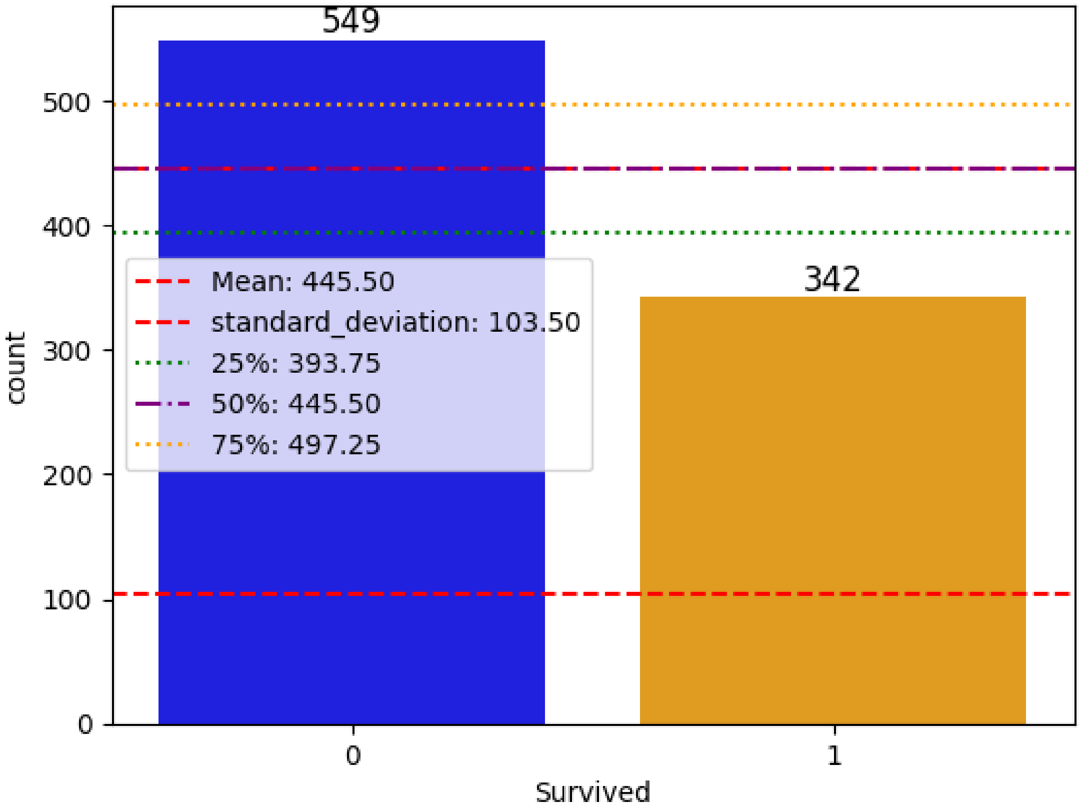
Bar Plot

As there are only 2 unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['Survived'].value_counts()
sns.countplot(x=titanic_df['Survived'], palette=['blue', 'orange'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\3660258745.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
```

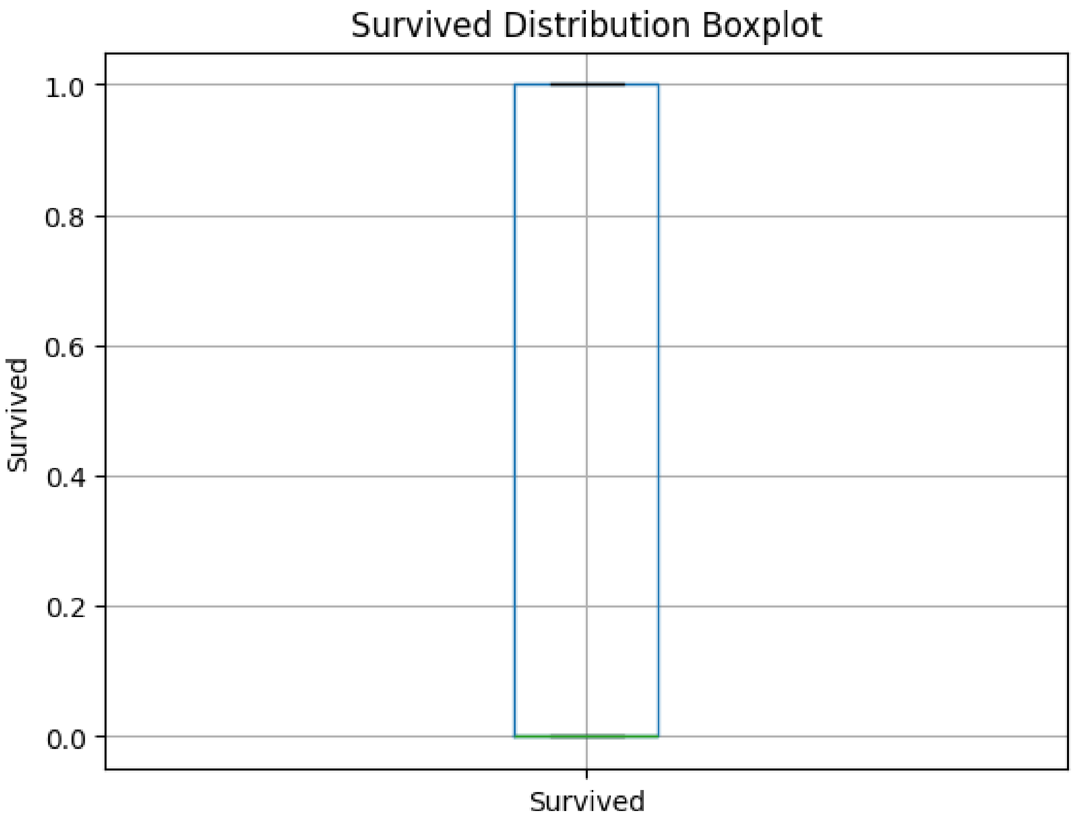
```
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['Survived'], palette=['blue', 'orange'])
```



The bar plot shows that the majority of passengers did not survive.

Box Plot

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='Survived')
plt.title('Survived Distribution Boxplot')
plt.ylabel('Survived')
plt.show()
```



The box plot reveals the distribution of survival statuses.

2. Pclass column

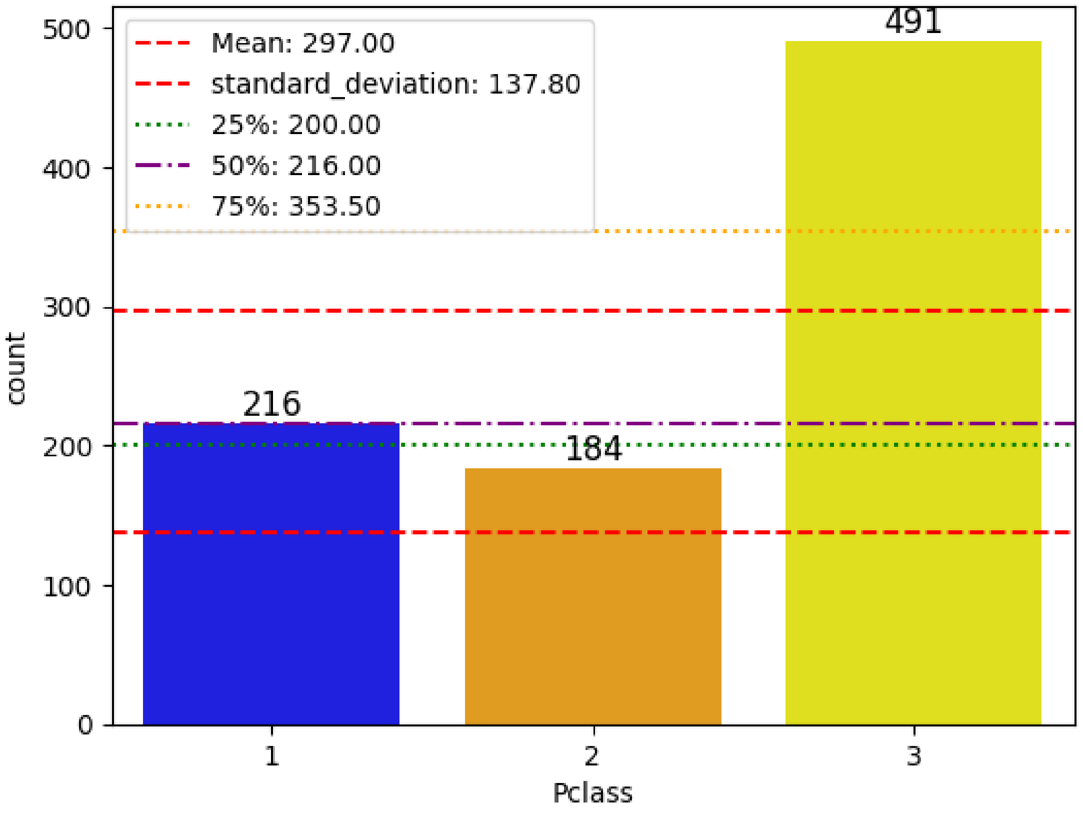
Bar Plot

As there are only 3 unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['Pclass'].value_counts()
sns.countplot(x=titanic_df['Pclass'], palette=['blue',
'orange','Yellow'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
```

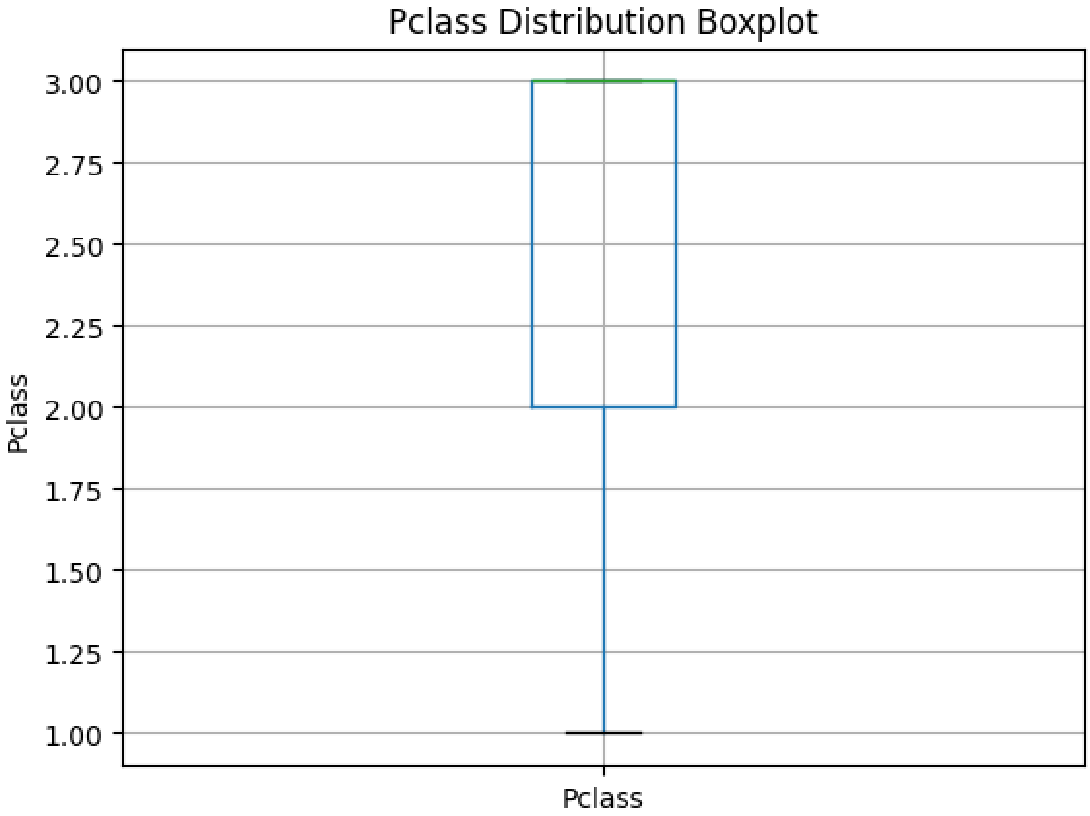
```
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\3798396356.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['Pclass'], palette=['blue',
'orange','Yellow'])
```



The bar plot indicates that third-class passengers were the largest group.

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='Pclass')
plt.title('Pclass Distribution Boxplot')
plt.ylabel('Pclass')
plt.show()
```



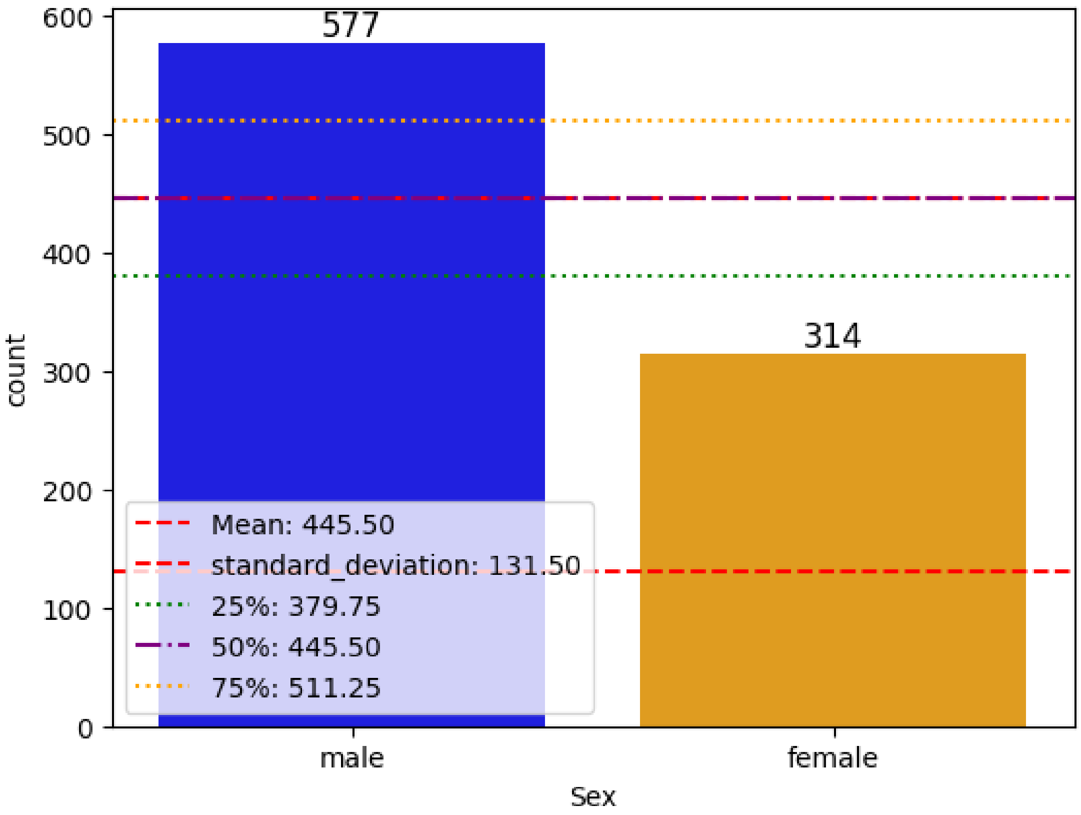
3. Sex Column

Bar Plot

As there are only 2 unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['Sex'].value_counts()
sns.countplot(x=titanic_df['Sex'], palette=['blue', 'orange'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\1928085125.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['Sex'], palette=['blue', 'orange'])
```



The 'Sex' bar plot shows that there were significantly more male passengers than female passengers.

The 'Survived' bar plot, when analyzed with gender, indicates that a higher proportion of female passengers survived compared to male passengers.

The count labels on the bars further reinforce that female passengers had a noticeably higher survival rate, showing the disparity in survival rates between genders.

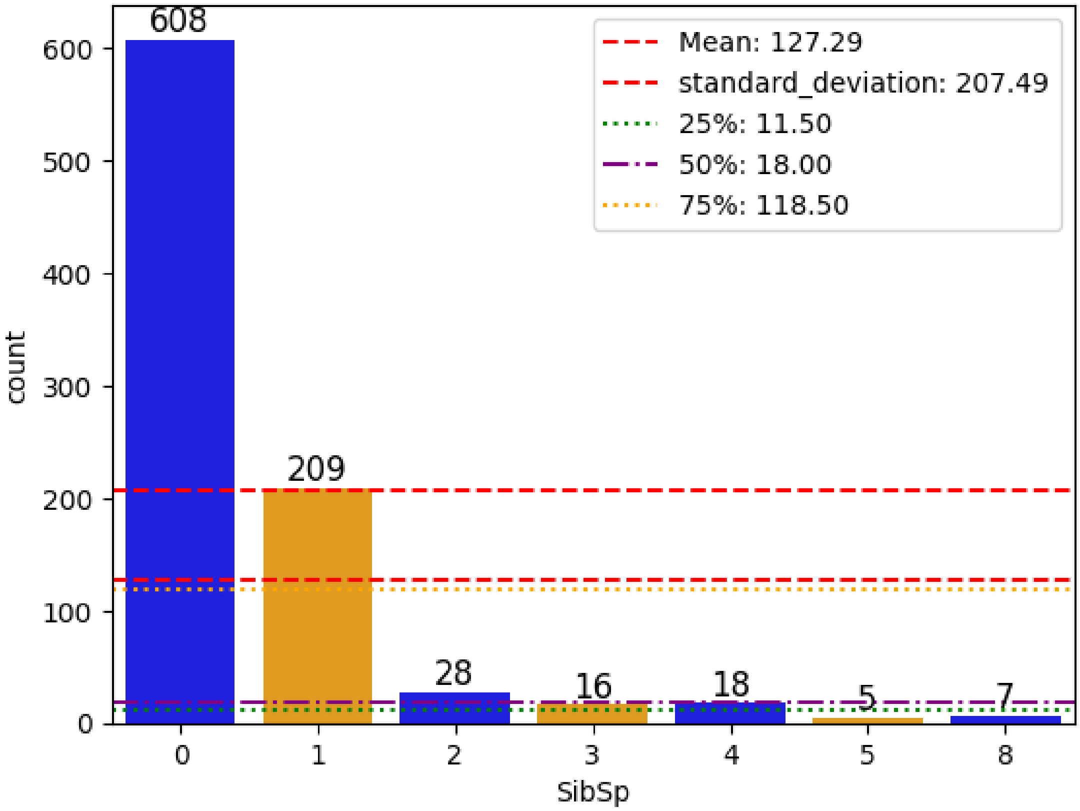
4. SibSp Column

Bar Plot

As there are only a few unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['SibSp'].value_counts()
sns.countplot(x=titanic_df['SibSp'], palette=['blue', 'orange'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\1367556401.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['SibSp'], palette=['blue', 'orange'])
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\1367556401.py:5:
UserWarning:
The palette list has fewer values (2) than needed (7) and will cycle,
which may produce an uninterpretable plot.
  sns.countplot(x=titanic_df['SibSp'], palette=['blue', 'orange'])
```



These show the count of passengers with different numbers of family members onboard.

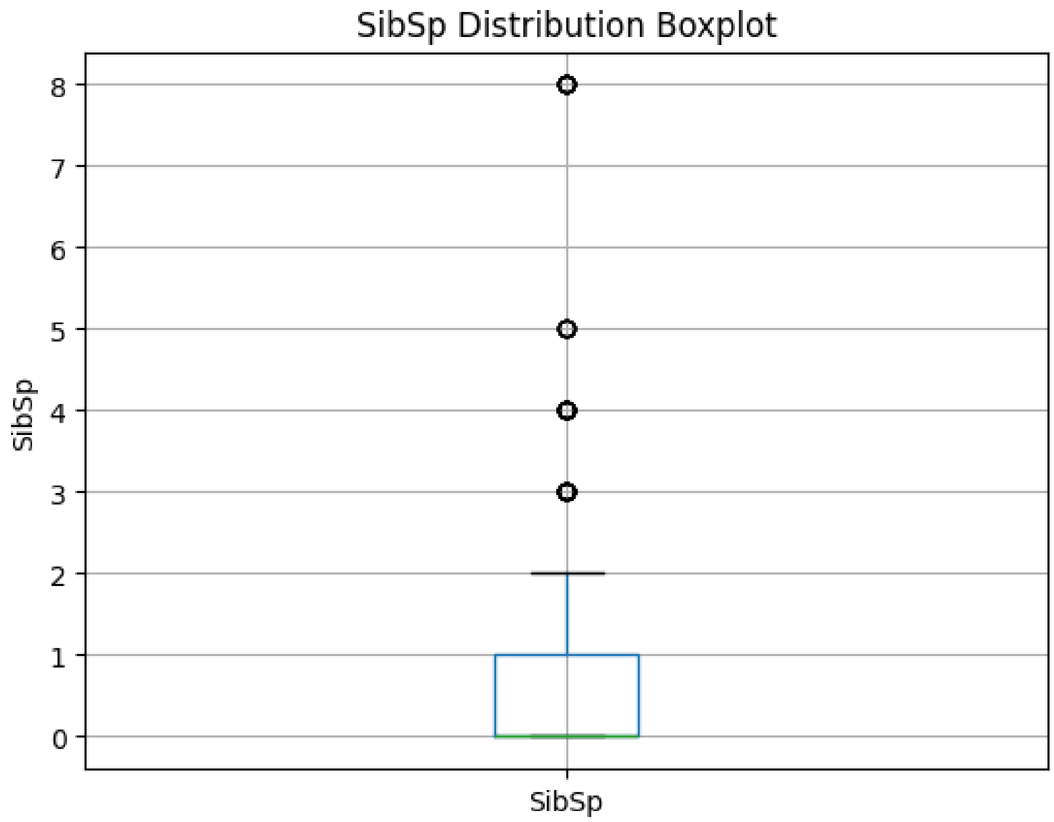
By comparing them with the Survived bar plot, you can infer which groups had better survival chances.

Passengers with low SibSp tend to have higher survival rates.

The bar plots show that most passengers traveled alone.

Box plot

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='SibSp')
plt.title('SibSp Distribution Boxplot')
plt.ylabel('SibSp')
plt.show()
```



The box plot of 'SibSp' (Sibling/Spouse count) shows the distribution of passengers traveling alone versus those who had family onboard.

The survival rate appears higher for passengers with a small number of family members.

The box plots suggest that those with family members onboard had slightly better survival chances.

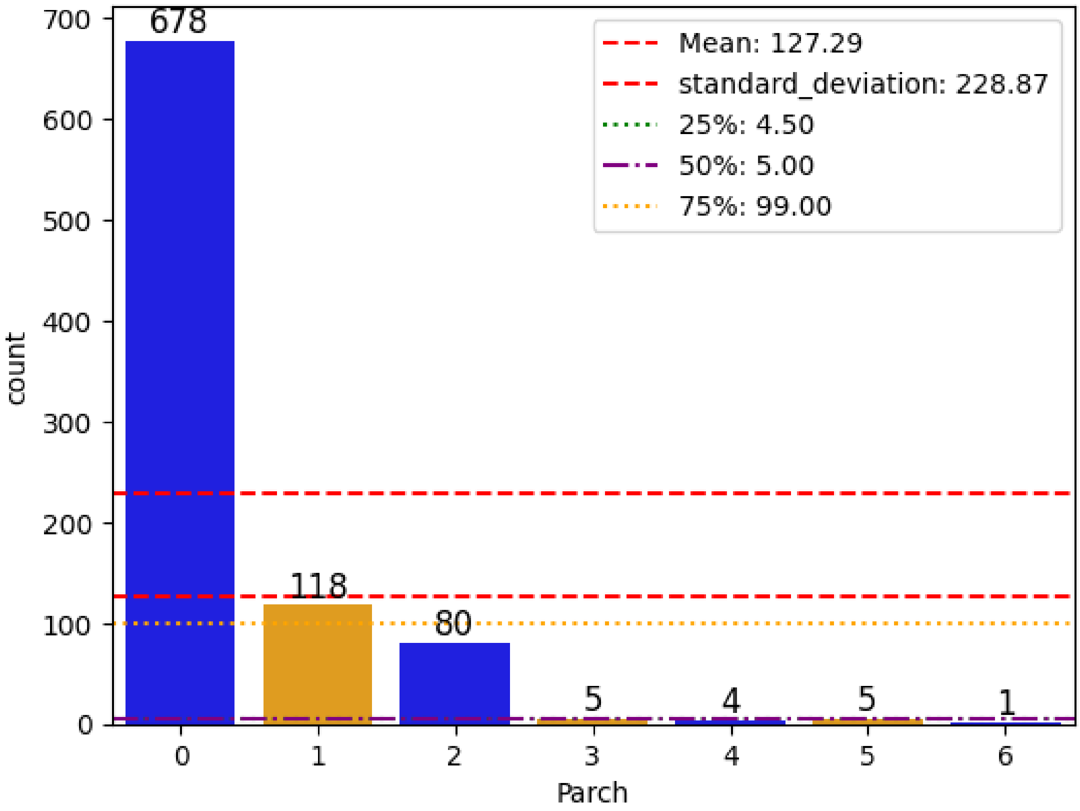
5. Parch Column

Bar Plot

As there are only a few unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['Parch'].value_counts()
sns.countplot(x=titanic_df['Parch'], palette=['blue', 'orange'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\2859125797.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['Parch'], palette=['blue', 'orange'])
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\2859125797.py:5:
UserWarning:
The palette list has fewer values (2) than needed (7) and will cycle,
which may produce an uninterpretable plot.
  sns.countplot(x=titanic_df['Parch'], palette=['blue', 'orange'])
```



These show the count of passengers with different numbers of family members onboard.

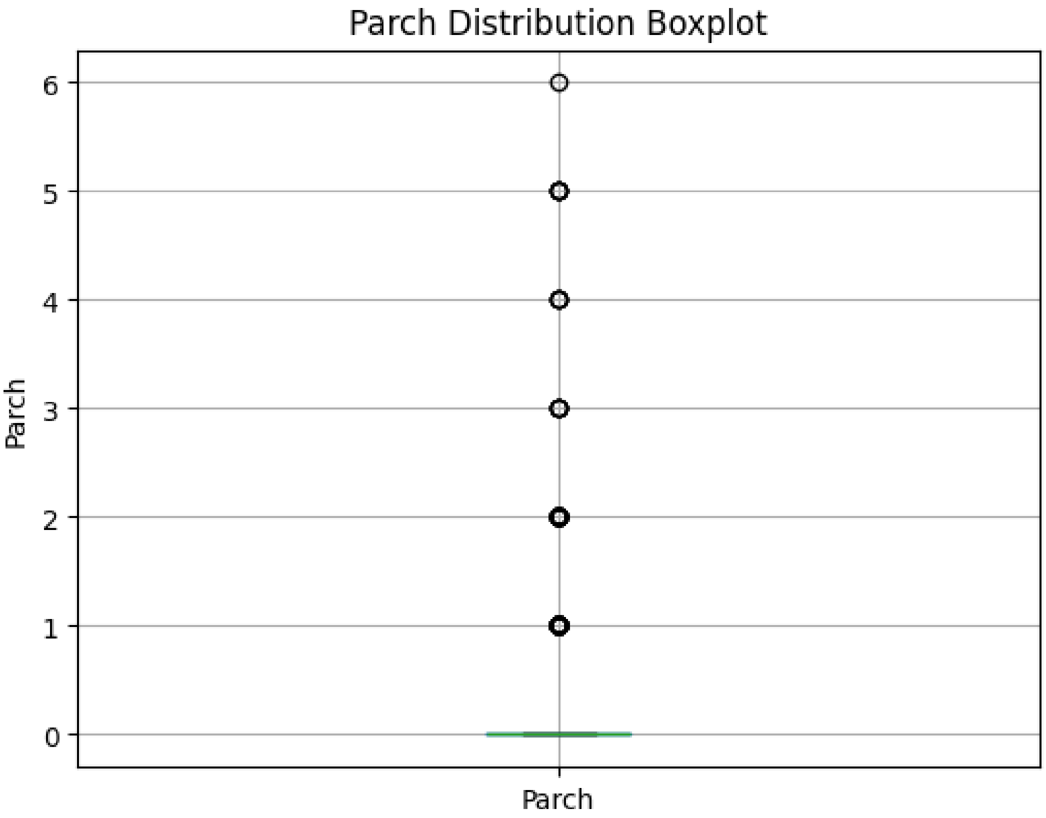
By comparing them with the Survived bar plot, you can infer which groups had better survival chances.

Passengers with low SibSp and Parch values tend to have higher survival rates.

The bar plots indicate that most passengers traveled alone.

Box Plot

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='Parch')
plt.title('Parch Distribution Boxplot')
plt.ylabel('Parch')
plt.show()
```



The box plot of 'Parch' (Parent/Children count) reveals that passengers who traveled with at least one family member had a slightly better chance of survival compared to those who were alone.

The median survival rate is slightly higher for passengers with family, suggesting some advantage in survival, possibly due to assistance during the emergency.

6. Cabin Column

```
print(f"The number of null values in the Cabin column is:
{titanic_df['Cabin'].isnull().sum()}")
print(f"Number of Unique values:{titanic_df['Cabin'].nunique()}")

The number of null values in the Cabin column is: 687
Number of Unique values:147
```

The Cabin column has a high number of missing values.

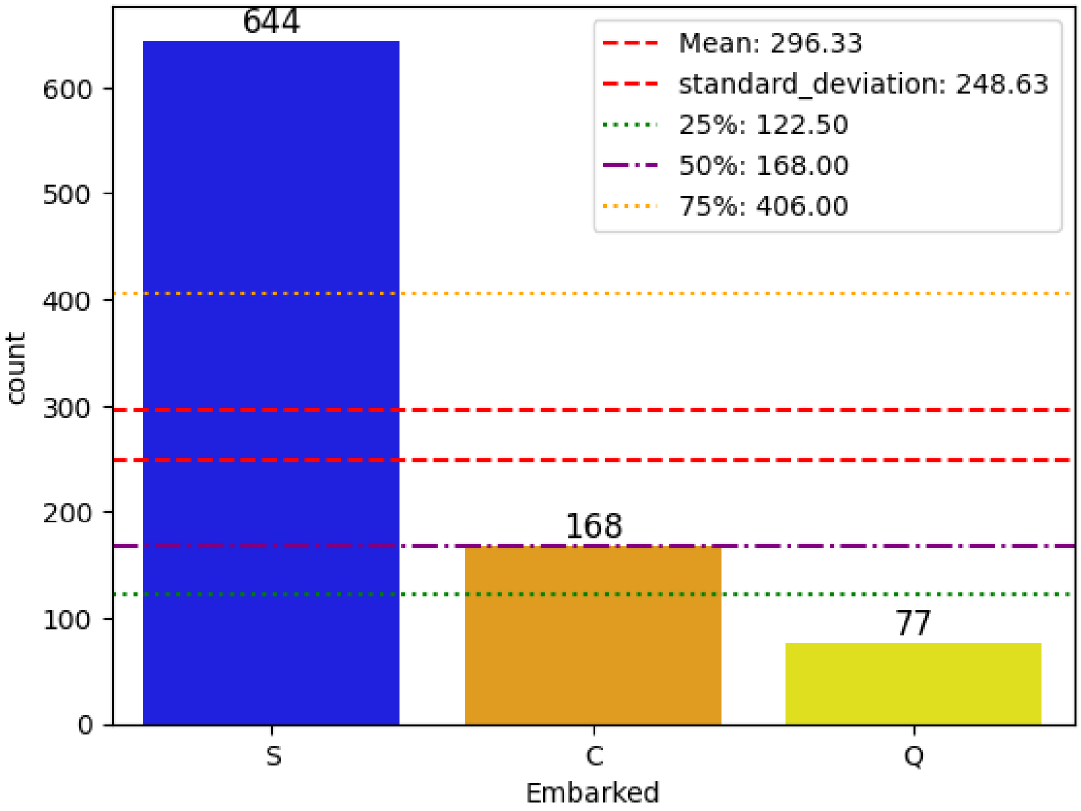
7. Embarked column

Bar Plot

As there are only a few unique values in the column, bar plot is more suitable than histogram.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
value_counts=titanic_df['Embarked'].value_counts()
sns.countplot(x=titanic_df['Embarked'], palette=['blue',
'orange','Yellow'])
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
for container in plt.gca().containers:
    plt.gca().bar_label(container, fontsize=12)
plt.axhline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.axhline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axhline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50%: {percentiles[1]:.2f}')
plt.axhline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.legend()
plt.show()
```

```
C:\Users\arunj\AppData\Local\Temp\ipykernel_21812\1366599108.py:5:
FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.countplot(x=titanic_df['Embarked'], palette=['blue',
'orange','Yellow'])
```

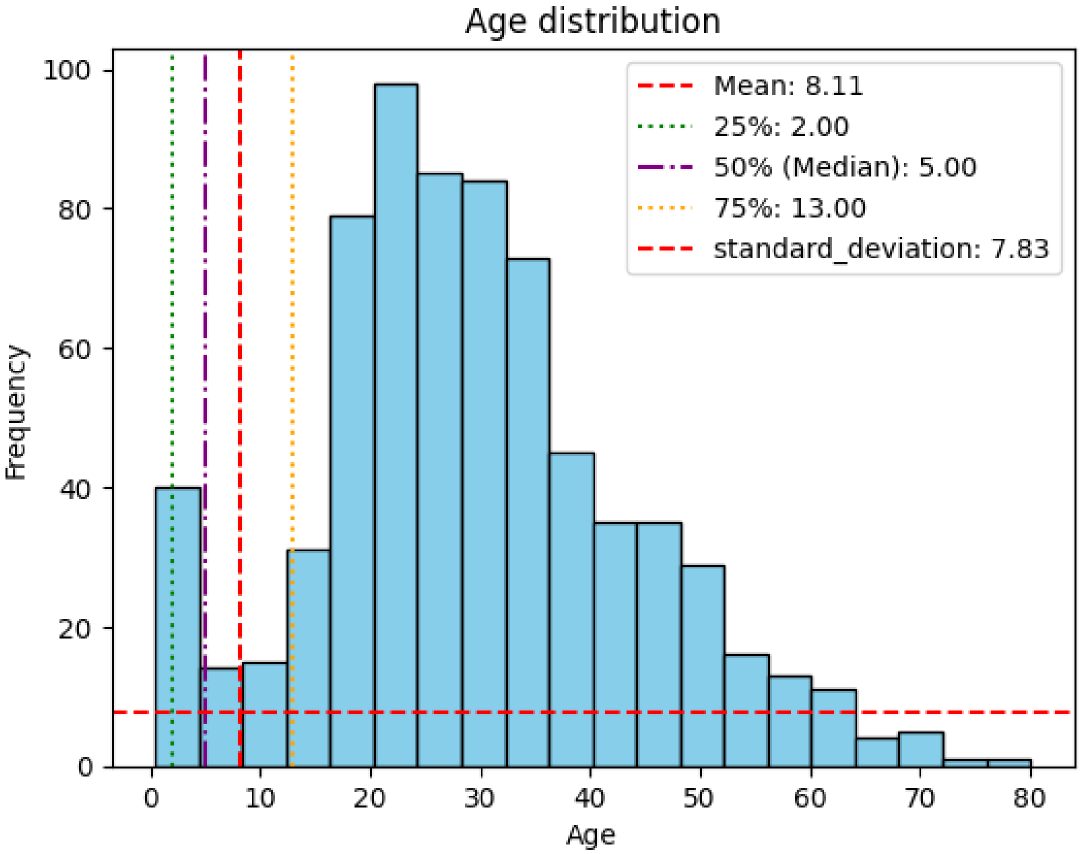


The bar plot for Embarked shows that Southampton had the highest number of passengers.

Numeric Data

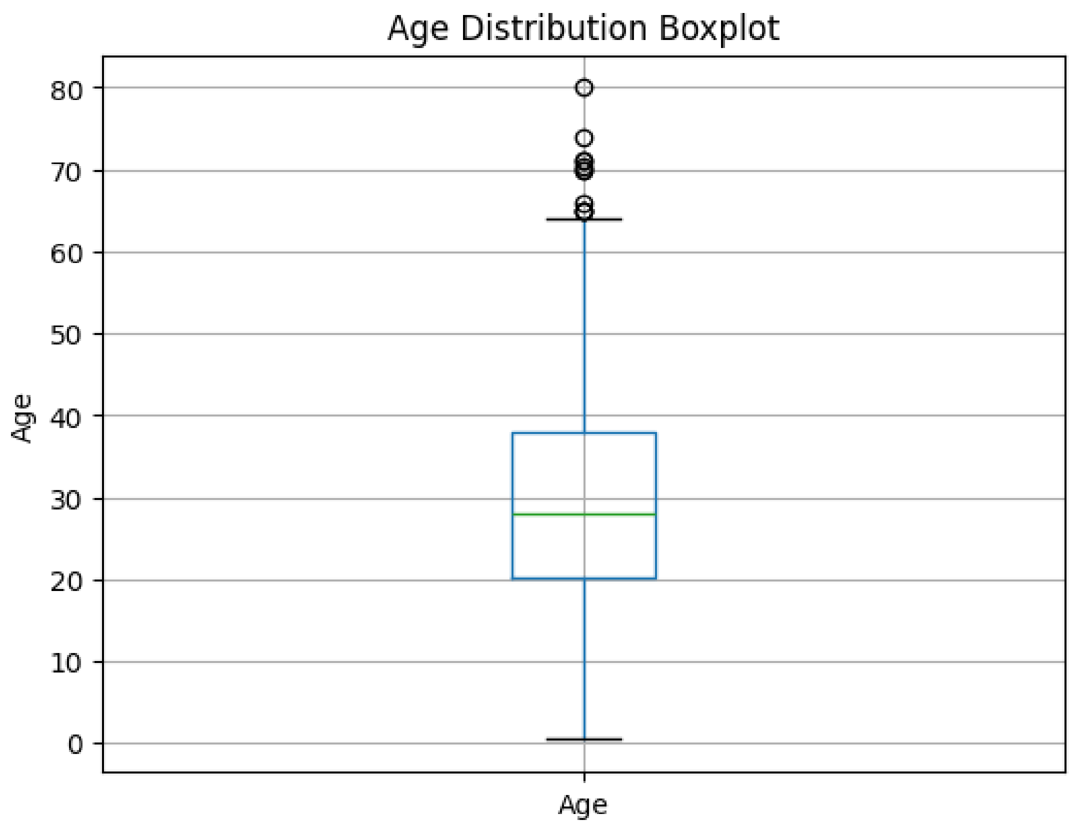
1. Age Distrubution

```
import pandas as pd
import matplotlib.pyplot as plt
value_counts=titanic_df['Age'].value_counts()
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
plt.hist(titanic_df['Age'],bins=20, color='skyblue',
edgecolor='black')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.title('Age distribution')
plt.axvline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axvline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
plt.axvline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50% (Median): {percentiles[1]:.2f}')
plt.axvline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.legend()
plt.show()
```



The histograms indicate that most passengers were young adults and that higher fares were linked to higher survival rates.

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='Age')
plt.title('Age Distribution Boxplot')
plt.ylabel('Age')
plt.show()
```



The box plot highlights the overall distribution, including:

- Median age around 28 years.

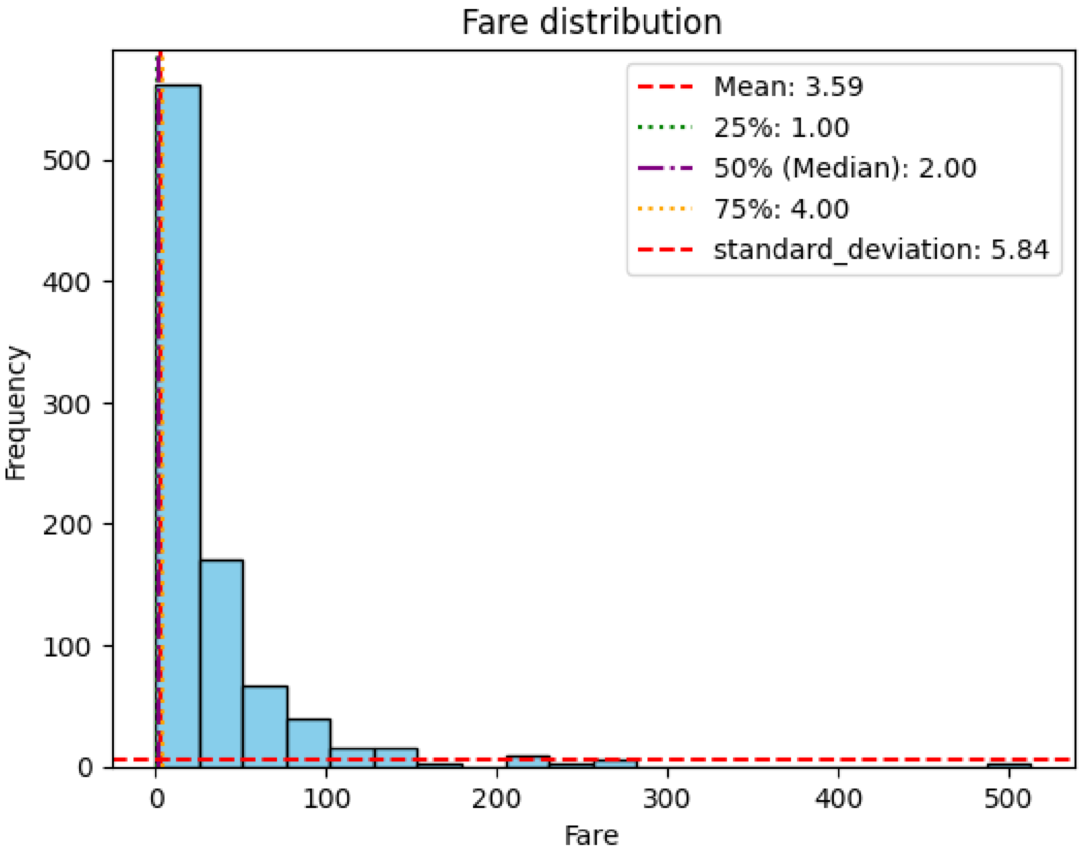
- Interquartile range (IQR) suggests most passengers were between 20 and 38 years.

- Outliers exist in older age groups, indicating a few passengers were over 60–80 years.

2. Fare distribution

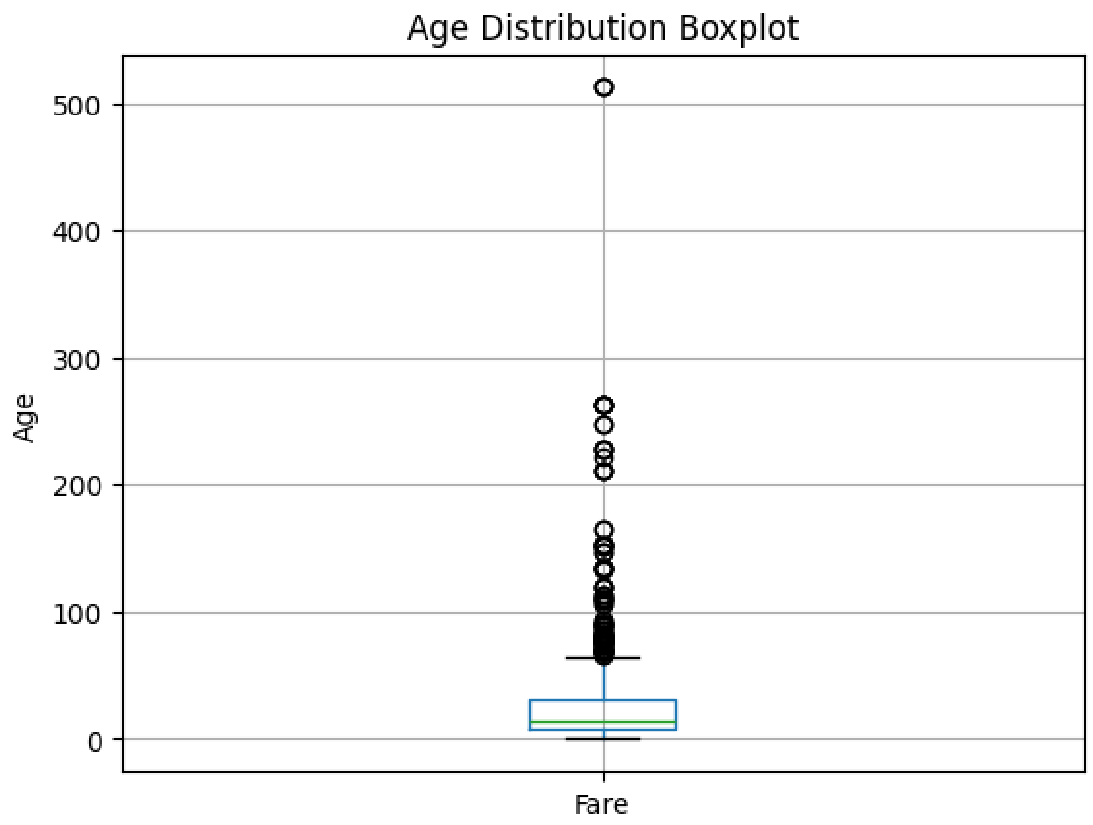
```
import pandas as pd
import matplotlib.pyplot as plt
value_counts=titanic_df['Fare'].value_counts()
mean = np.mean(value_counts)
std_dev = np.std(value_counts)
percentiles = np.percentile(value_counts, [25, 50, 75])
plt.hist(titanic_df['Fare'],bins=20, color='skyblue',
edgecolor='black')
plt.xlabel('Fare')
plt.ylabel('Frequency')
plt.title('Fare distribution')
plt.axvline(mean, color='red', linestyle='dashed', label=f'Mean:
{mean:.2f}')
plt.axvline(percentiles[0], color='green', linestyle='dotted',
label=f'25%: {percentiles[0]:.2f}')
```

```
plt.axvline(percentiles[1], color='purple', linestyle='dashdot',
label=f'50% (Median): {percentiles[1]:.2f}')
plt.axvline(percentiles[2], color='orange', linestyle='dotted',
label=f'75%: {percentiles[2]:.2f}')
plt.axhline(std_dev, color='red', linestyle='dashed',
label=f'standard_deviation: {std_dev:.2f}')
plt.legend()
plt.show()
```



The histogram illustrates a right-skewed distribution, meaning a large number of passengers paid lower fares, while fewer paid higher fares.

```
import pandas as pd
import matplotlib.pyplot as plt
titanic_df.boxplot(column='Fare')
plt.title('Age Distribution Boxplot')
plt.ylabel('Age')
plt.show()
```



The box plot reveals:

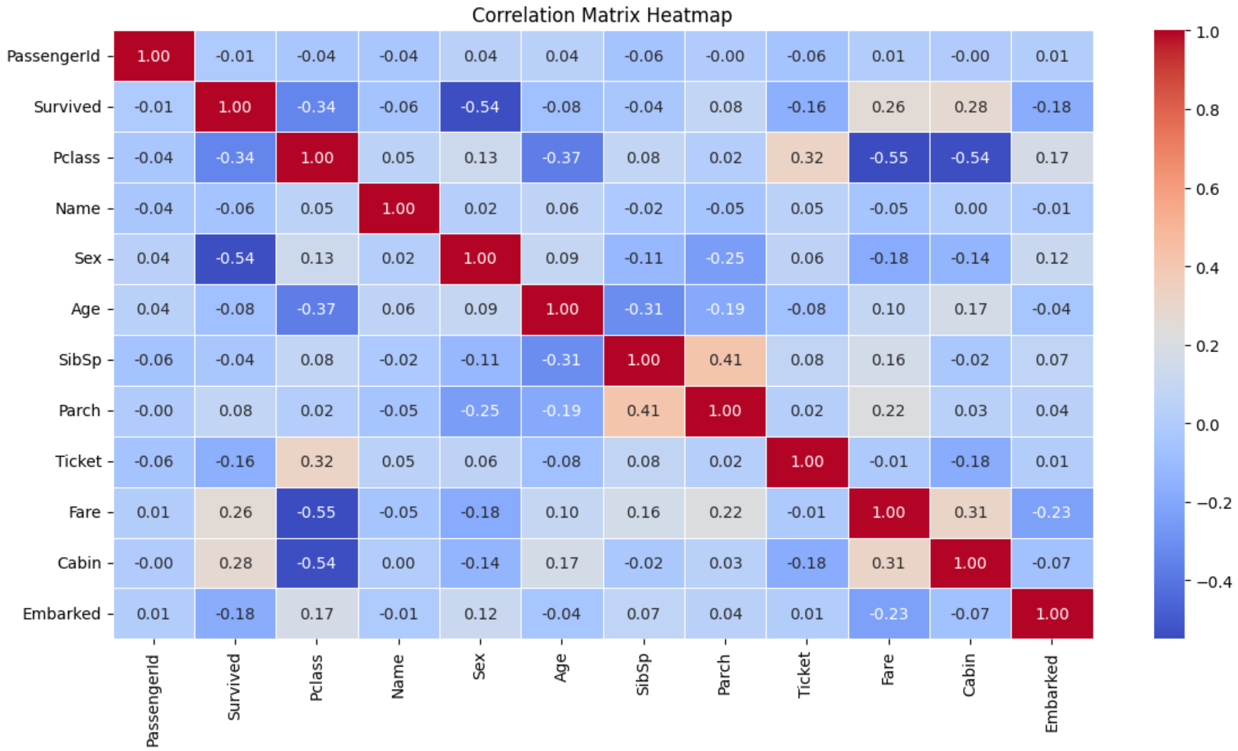
- Median fare around 14–15 currency units.

- Interquartile range (IQR) shows most fares were between 7 and 31 units.

- Outliers exist where some passengers paid extremely high fares (~512 units), likely first-class passengers.

PairPlot for feature relationships

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
df_encoded = titanic_df.apply(lambda x: x.astype('category').cat.codes
if x.dtype == 'object' else x)
corr_matrix=df_encoded.corr()
plt.figure(figsize=(14,7))
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f",
linewidths=0.5)
plt.title("Correlation Matrix Heatmap")
plt.show()
```



	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId	1.00	-0.01	-0.04	-0.04	0.04	0.04	-0.06	-0.00	-0.06	0.01	-0.00	0.01
Survived	-0.01	1.00	-0.34	-0.06	-0.54	-0.08	-0.04	0.08	-0.16	0.26	0.28	-0.18
Pclass	-0.04	-0.34	1.00	0.05	0.13	-0.37	0.08	0.02	0.32	-0.55	-0.54	0.17
Name	-0.04	-0.06	0.05	1.00	0.02	0.06	-0.02	-0.05	0.05	-0.05	0.00	-0.01
Sex	0.04	-0.54	0.13	0.02	1.00	0.09	-0.11	-0.25	0.06	-0.18	-0.14	0.12
Age	0.04	-0.08	-0.37	0.06	0.09	1.00	-0.31	-0.19	-0.08	0.10	0.17	-0.04
SibSp	-0.06	-0.04	0.08	-0.02	-0.11	-0.31	1.00	0.41	0.08	0.16	-0.02	0.07
Parch	-0.00	0.08	0.02	-0.05	-0.25	-0.19	0.41	1.00	0.02	0.22	0.03	0.04
Ticket	-0.06	-0.16	0.32	0.05	0.06	-0.08	0.08	0.02	1.00	-0.01	-0.18	0.01
Fare	0.01	0.26	-0.55	-0.05	-0.18	0.10	0.16	0.22	-0.01	1.00	0.31	-0.23
Cabin	-0.00	0.28	-0.54	0.00	-0.14	0.17	-0.02	0.03	-0.18	0.31	1.00	-0.07
Embarked	0.01	-0.18	0.17	-0.01	0.12	-0.04	0.07	0.04	0.01	-0.23	-0.07	1.00

The heatmap highlights strong correlations between survival and passenger class, fare, and gender.