REPUBLIC OF THE PHILIPPINES
POLYTECHNIC UNIVERSITY OF THE PHILIPPINES
COLLEGE OF ENGINEERING
BACHELOR OF SCIENCE IN COMPUTER ENGINEERING

CMPE 363: EXPLORATORY DATA ANALYSIS OF THE TITANIC DATASET

ZURBITO, PIERRE VICTOR T.

APRIL 2025

Introduction to the Dataset

The Titanic dataset contains detailed information about the passengers onboard the RMS Titanic, which sank in April 14, 1912 after colliding with an iceberg during its maiden voyage en route to New York from Southampton. Of the 2,224 passengers and crew on board, 1,500 people tragically died, making it one of the deadliest peacetime disasters in history.

The dataset contains key demographic and socio-economic information about the passengers which include the age, survival status, ticket class, fare costs, place of embarkation, and their respective cabins. The said dataset is one of the introductory datasets used in data science education as a beginner-friendly dataset for exploratory data analysis.

This data analysis explores the factors which influenced a passenger's survival chances and uncovering patterns that may offer insights on the decisions made during the ship's evacuation.

Data Dictionary

The dataset contains 891 rows and 12 variables. The following contains the main variables present in the dataset:

Table 1

Data Dictionary of the Titanic Dataset

Column Name	Data Type	Description
`PassengerID`	Int64	Unique Identifier on the passengers on the ship.
`Survived`	Int64	Shows the survival status of the passengers. *(0 = perished, 1 = survived)*
`Pclass`	Int64	Shows the person's ticket class. *(1 = first class, 2 = second class, 3 = third class)*
`Name`	Object	Name of the passengers of the ship.
`Sex`	Object	The gender of the passenger.
`Age`	Float64	The age of the passenger in years.
`SibSp`	Int64	Number of siblings / spouses on board.
`Parch`	Int64	Number of parents / children on board.

`Ticket`	Object	The ticket number of the passenger.
`Fare`	Float64	The cost of tickets.
`Cabin`	Object	The cabin number of the passenger.
`Embarked`	Object	The place of embarkation of the passenger. *(C = Cherbourg, Q = Queenstown, S = Southampton)*

Guide Questions Regarding the Dataset

Before the exploration of the dataset, the following questions served as a guide for the analyst:

1. How dispersed is the dataset from each other?
2. What is the breakdown of the dataset with regards to:
   a. Their sex;
   b. Their point of origin;
   c. And the passenger class?
3. What is the distribution of survival rates among the passengers?
4. What are the correlation between the variables?
   a. If there is/are strong correlation/s in the variables, what does it tell us?
5. What are the survival rates per class of passengers?

Data Cleaning, Analysis, and Visualization

Upon further inspection of the dataset, the following are seen:

- The `cabin` has the most of the missing data, for which there are only 204 non-null entries out of 891 rows. For this instance, the analyst has dropped the cabin column due to the number of missing data and complexities surrounding the data.
- This is then followed by the `age` which has 714 non-null entries. With this, the `NaN` parts were filled with the median age of everyone on the ship as it is more resistant to outliers (the dataset is very skewed).
- Upon filling the age column with the median values, the following measures of central tendency and variability were seen:

Table 2

Measures of Central Tendency in the Age and Fares.

Column Name	Mean	Median	Mode	Standard Deviation
`Age`	29.4	28	28	14.5
`Fare`	32.2	14.5	8.1	49.7

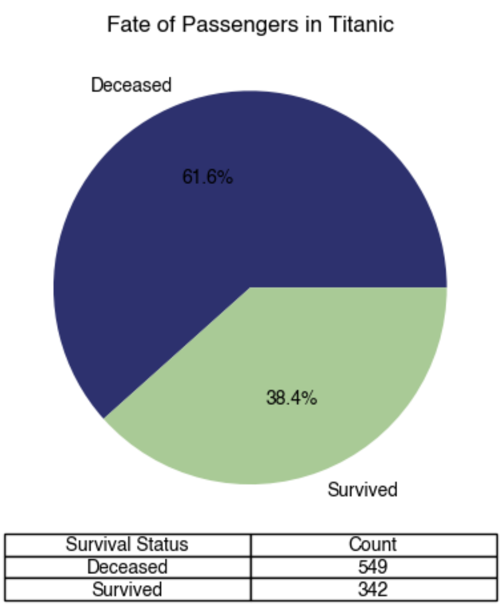


Figure 1.0. Pie chart on the passengers' survival rate.

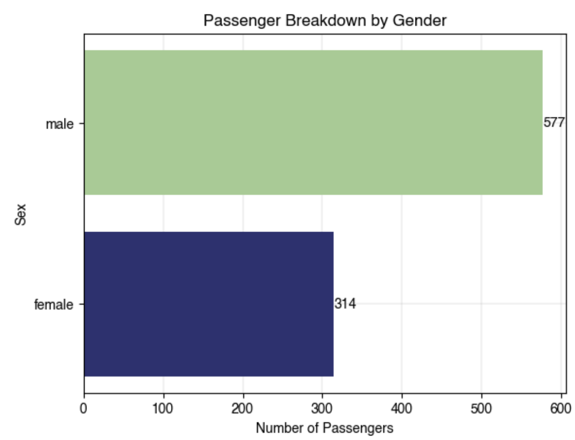


Figure 1.1. Passenger Breakdown by Gender

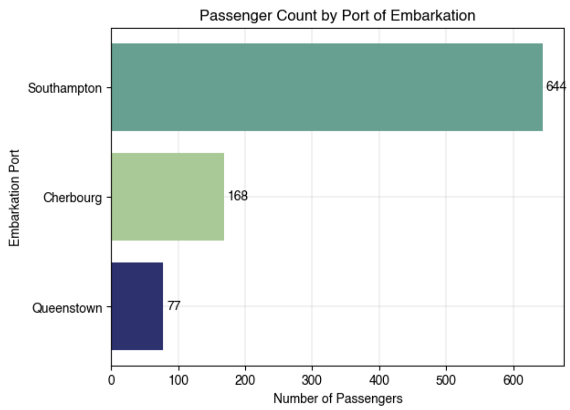


Figure 1.2. Passenger count by Port of Embarkation

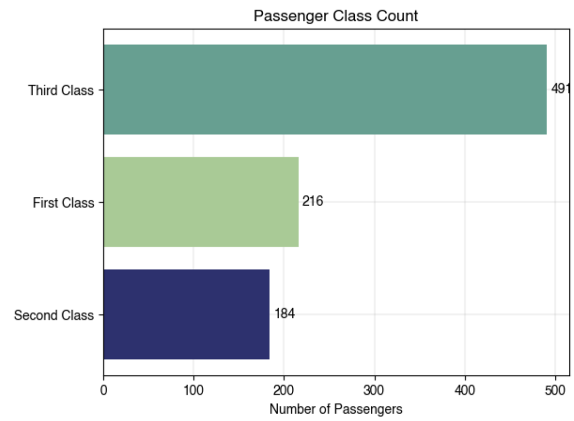


Figure 1.3. Passenger count by their classes.

Univariate Analysis and Visualization

- Looking into the fate of the passengers in Titanic, only 38.4% survived, while the remaining 61.6% has perished.
- Of the total passengers of the ship, 577 are men and 314 passengers are women.
- Southampton port has the most number of port embarkations, followed by Cherbourg and Queenstown, respectively.

Correlational Analysis

- Testing the Pearson correlation between numeric variables yield the following results:
  - There is a very weak correlation between the passenger fare and age, which indicates that a passenger's ticket is completely independent of their age;
  - There is a weak negative correlation between the passenger's survival and its

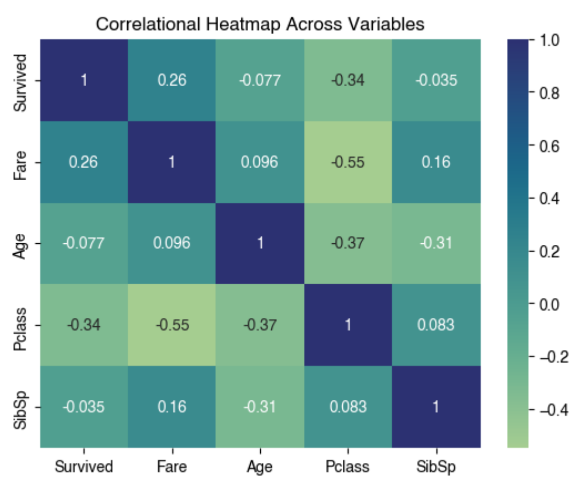


Figure 1.4. Correlation heatmap across variables in the dataset.

class, indicating that as the passenger's class increases, their chances of survival slightly decreases;

- There is a moderately negative correlation between the fare and the passenger's class, which means that as ticket prices go up, the passenger's class goes down;
- And, there is a weak positive correlation between the passenger fares and their survival, indicating that there is a slightly better chance of survival for passengers who paid higher fares.

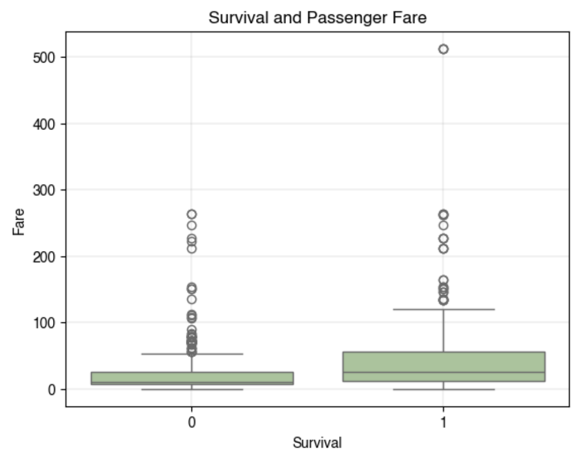


Figure 1.5. Boxplot between the survival and fare

- The relationship between the passengers' survival and fares are further explored in a boxplot, where it is observed that:
  - the median (horizontal line inside the rectangles) of the graph is closer to the bottom, indicating a positively skewed data for both the deceased and survived.
  - There are more outliers in the deceased section than on the survivors, which indicate that there are wider range of fares among the passengers who did not survive. This is then supported by the wideness of the whiskers on both charts.

- The relationship between the passengers' survival and their respective classes have their weak negative correlation on the heatmap as well. Diving deeper yields the following:
    - First class passengers have the highest rate of survival with 62.96%, followed by the second class passengers with 47.28%, and lastly are the third class passengers with 24.24%.
- The relationship between the passengers' sex and survival rate were also taken into consideration, and it is found out that female passengers have higher rates of survival than those of male passengers.
    - Female passengers have 74.20% chance and male passengers have a significantly lower chance of only 18.89%.

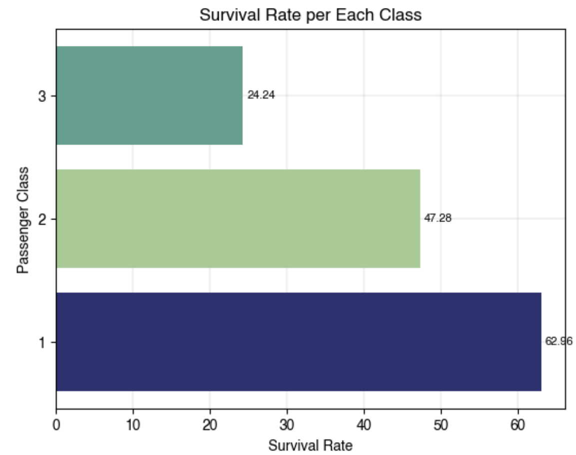


Figure 1.6. Boxplot between the survival and passenger class.

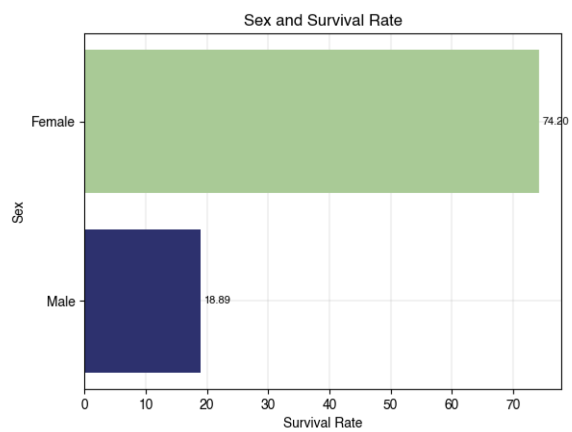


Figure 1.7. Boxplot between the survival and sex.

Conclusion and Recommendations

The dataset has given us meaningful insights about the factors which influenced the survival of passengers during the sinking of Titanic. Several key findings were observed:

- There was a correlation between the passenger classes and their survival, and it is seen that first-class passengers have higher chances of survival.
- There was an advantage observed with respect to female passengers than those of male.

- There was a weak correlation identified between the passengers' fares prices and their survival rates. Also, numerous outliers are seen on the following as a lot of exceptions were seen.
- There are a lot of missing data seen on the cabin and age columns, and it was filled with median values as the dataset is skewed.

With these findings in mind, the analyst recommends the following:

- Multivariate analysis is recommended to further understand the survival rates of passengers of the ship.
- Carefully add the ages of the passengers as opposed to only using the median, because although median is more resistant to skewed data, it is also not guaranteed than carefully approximating for each of the ages.
- Dive deeper on the other columns of the dataset, such as looking further into the SibSp column to determine family sizes, and comparing it to their survival rates.

The repository used on these can also be accessed through GitHub with the following link: https://github.com/pvtzurbito/EDA-Titanic

Bibliography

National Oceanic and Atmospheric Administration. (2024, July 18). *R.M.S Titanic - History and Significance*. Retrieved from NOAA: https://www.noaa.gov/office-of-general-counsel/gc-international-section/rms-titanic-history-and-significance

Salkind, Neil J. (2017). *Statistics for People Who Think They Hate Statistics*. SAGE Publications Inc.

ANNEX A

PEARSON CORRELATION COEFFICIENT AND INTERPRETATION

The interpretation of Pearson's correlation coefficient used in this analysis follows the commonly accepted guidelines where values close to 0 indicate weak correlation, and values close to 1 or -1 indicate strong correlation.

The Pearson correlation coefficient is a statistical measure that is used to describe the relationship of two variables. It quantifies the strength as well as the direction of the relationships of the variables being compared. The values of Pearson coefficients range from -1 to 1, and is described in the table below:

Table 1

Pearson correlation coefficient and their interpretation.

Correlation Interval	Relationship Level
0.8 – 1.0 (-0.8 to – 1.0)	Very Strong (Negative)
0.6 – 0.79 (-0.6 to -0.79)	Strong (Negative)
0.4 – 0.59 (-0.4 to -0.59)	Moderate (Negative)
0.2 – 0.39 (-0.2 to -0.39)	Weak (Negative)
0.0 – 0.19 (0.00 to -0.19)	Very Weak (Negative)

Note: Values adopted from Salkind (2017).

SCRIPTS USED FOR DATA ANALYSIS

Dataset Import

```
import kagglehub

# Download latest version
path = kagglehub.dataset_download("yasserh/titanic-dataset")

print("Path to dataset files:", path)
```

```
Warning: Looks like you're using an outdated `kagglehub` version
(installed: 0.3.6), please consider upgrading to the latest version
(0.3.12).
Path to dataset files:
/Users/pierrezurbito/.cache/kagglehub/datasets/yasserh/titanic-
dataset/versions/1
```

Dataset Shape

```
import pandas as pd
df = pd.read_csv('Titanic-Dataset.csv')

print(df.shape)
df.head(10)
#df.dropna()
#df.drop_duplicates()
```

```
(891, 12)

   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3
5            6         0       3
6            7         0       1
7            8         0       3
8            9         1       3
9           10         1       2

                                                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
5                                     Moran, Mr. James    male   NaN      0
6                              McCarthy, Mr. Timothy J    male  54.0      0
7                       Palsson, Master. Gosta Leonard    male   2.0      3
8  Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  female  27.0      0
9                  Nasser, Mrs. Nicholas (Adele Achem)  female  14.0      1

   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
5      0            330877   8.4583   NaN        Q
6      0             17463  51.8625   E46        S
7      1            349909  21.0750   NaN        S
8      2            347742  11.1333   NaN        S
9      0            237736  30.0708   NaN        C
```

```
df.dtypes
```

```
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
```

```
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
```

```
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  
```

```
median_age = df['Age'].median()
mean_age = df['Age'].mean()
mode_age = df['Age'].mode()

median_fare= df['Fare'].median()
mean_fare = df['Fare'].mean()
mode_fare = df['Fare'].mode()

print(df['Survived'].count())

print(mean_age, median_age, mode_age)
print(mean_fare, median_fare, mode_fare)
```

```
891
29.69911764705882 28.0 0    24.0
Name: Age, dtype: float64
32.204207968574636 14.4542 0    8.05
Name: Fare, dtype: float64
```

```
df['Age'] = df['Age'].fillna(df['Age'].median())
df['Age'].isnull().sum()
```

```
np.int64(0)
```

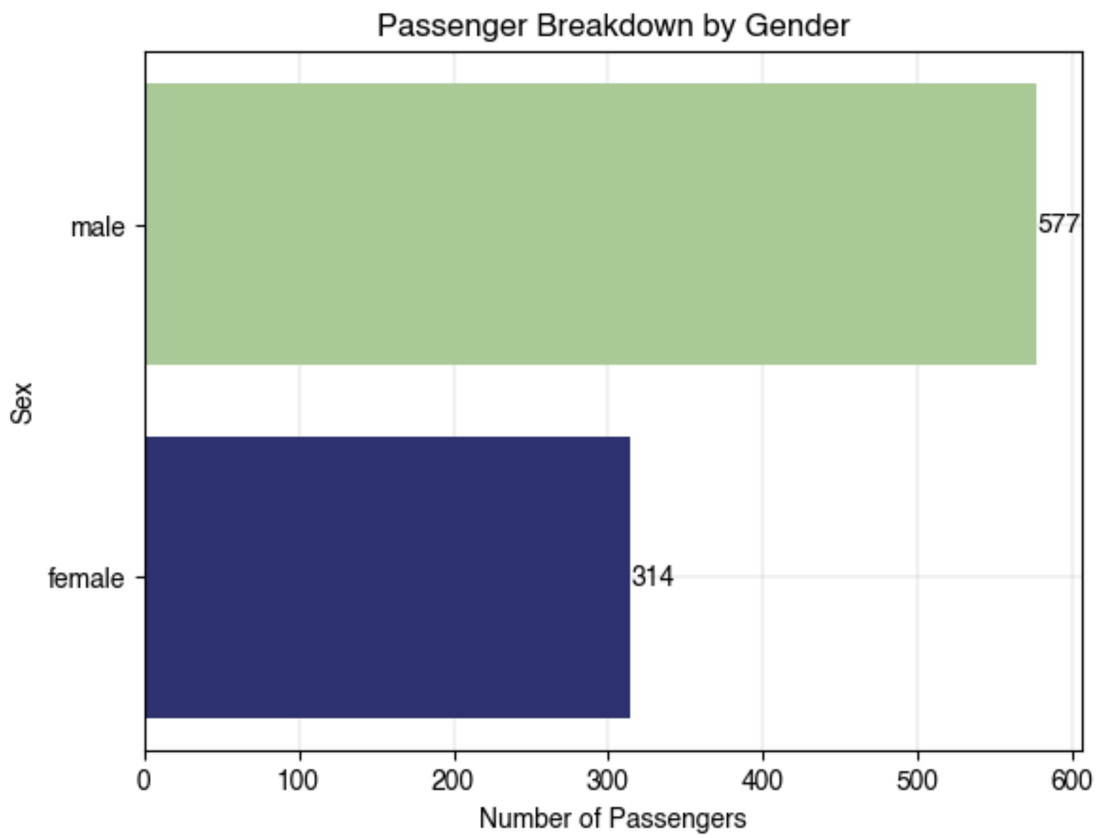
```
survival_count = df["Survived"].astype(str).eq('1').sum()
print("Number of survivors: ", survival_count)

ave_survival_age = df["Age"].mean()
print("The average age of everyone in the ship is: ",
int(ave_survival_age))
```

```
Number of survivors:  342
The average age of everyone in the ship is:  29
```

Count of Passenger by Gender

```
import matplotlib.pyplot as plt

plt.rcParams['font.family'] = 'Helvetica'
sex = df["Sex"].value_counts().sort_index()
fig, ax = plt.subplots(1,1)

colors = ['#2d316e', '#a9ca96', '#679f91']
x = sex.index
y = sex.values

bars = plt.barh(x, y, color = colors)

for bar in bars:
    width = bar.get_width()
    plt.text(width + 0.5,
             bar.get_y() + bar.get_height()/2,
             f'{width:.0f}',
             ha='left', va='center', fontsize=10)

ax.set_axisbelow(True)
plt.grid(linewidth = 0.25)
```

```
plt.xlabel('Number of Passengers')
plt.xticks()
plt.ylabel('Sex')
plt.title("Passenger Breakdown by Gender")

plt.show()
```

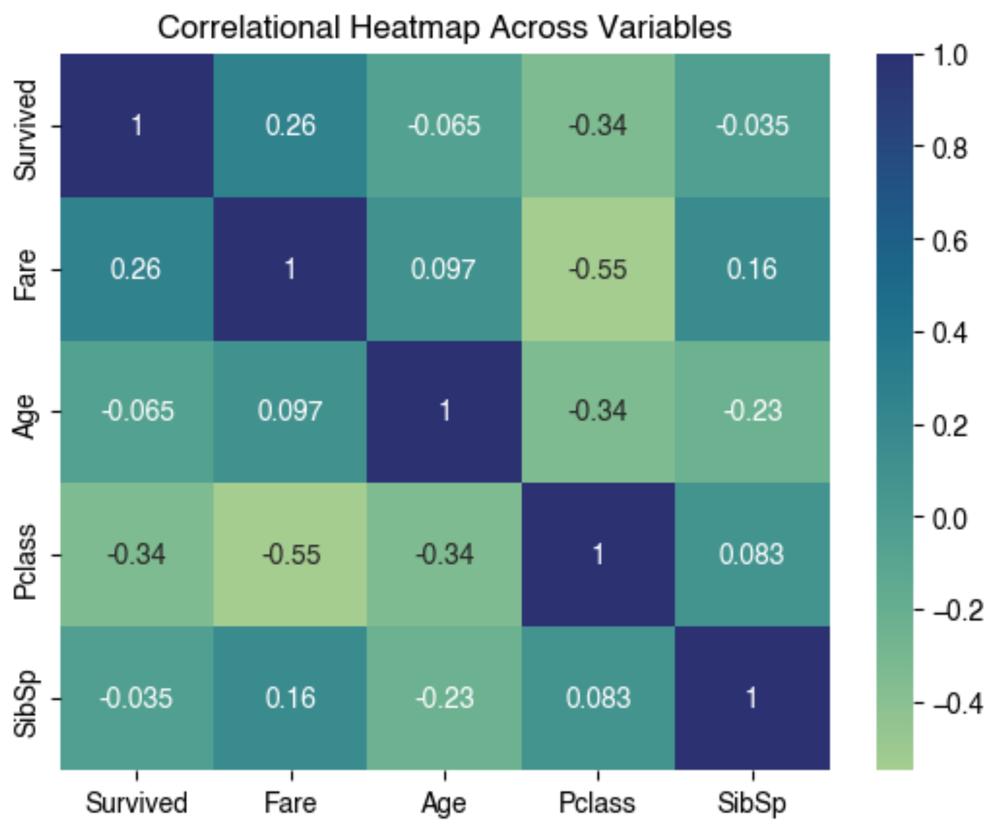



```
correlation = df['Fare'].corr(df['Survived'])
print(correlation)
```

```
0.25730652238496243
```

```
import seaborn as sns
import matplotlib.pyplot as plt

array= ['Survived', 'Fare', 'Age', 'Pclass', 'SibSp']
correlation_matrix = df[array].corr(method='pearson')

plt.rcParams['font.family'] = 'Helvetica'
```

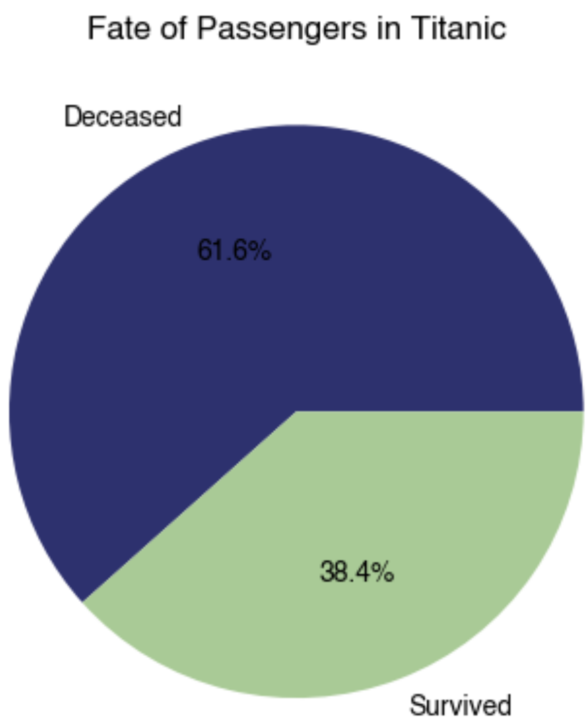
```
plt.title('Correlational Heatmap Across Variables')
sns.heatmap(correlation_matrix, cmap='crest', annot=True)
plt.show()
```




```
import matplotlib.pyplot as plt
plt.rcParams['font.family'] = 'Helvetica'
survival_counts = df['Survived'].value_counts().sort_index()
survival_counts.plot(kind = 'pie', labels = ['Deceased', 'Survived'],
autopct = '%1.1f%%', colors = colors)
plt.title('Fate of Passengers in Titanic')
table_data = [
    ['Deceased', survival_counts[0]],
    ['Survived', survival_counts[1]]
]

table = plt.table(
    cellText = table_data,
    colLabels= ['Survival Status', 'Count'],
    cellLoc='center',
    loc='bottom',
```

```
)
plt.ylabel('')
```

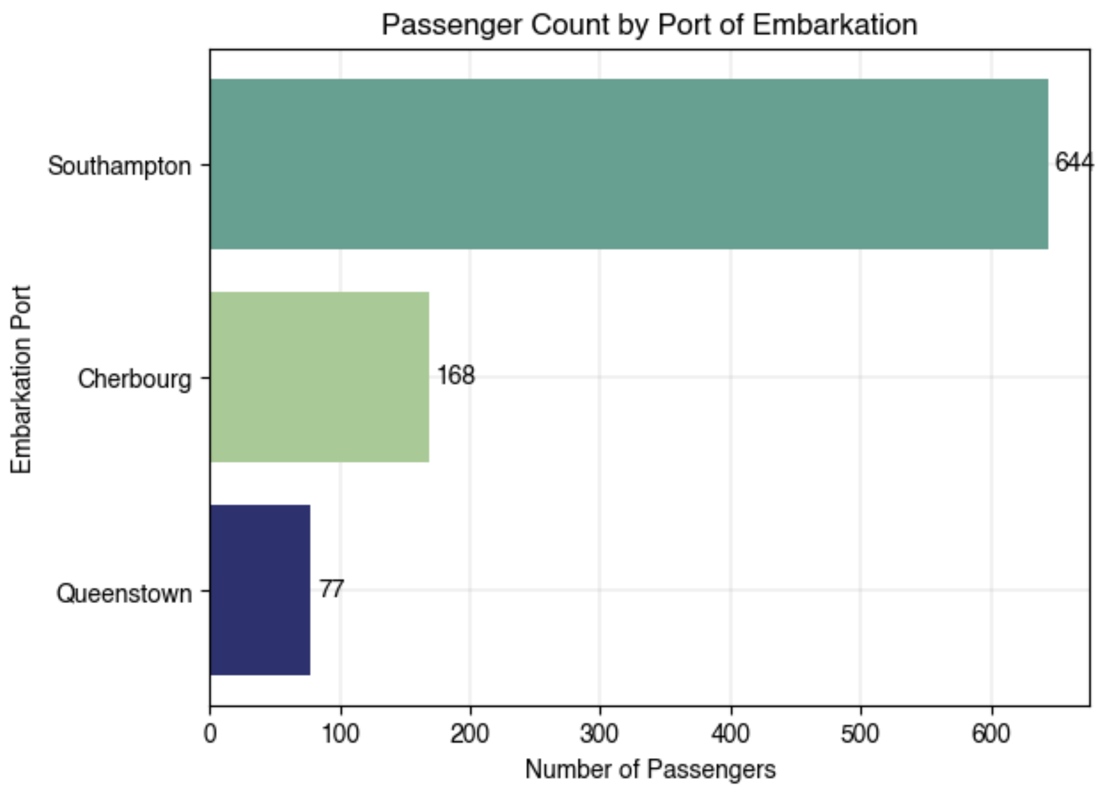
```
Text(0, 0.5, '')
```



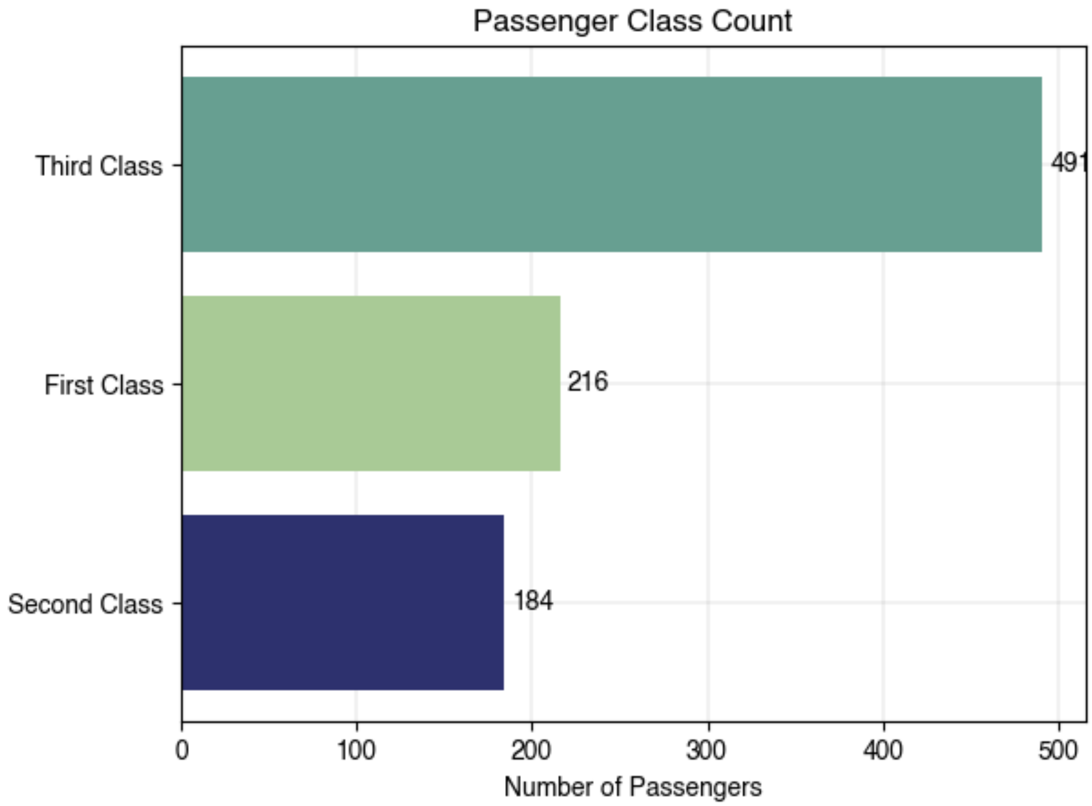
Survival Status	Count
Deceased	549
Survived	342

```python
import matplotlib.pyplot as plt

# Set global font
plt.rcParams['font.family'] = 'Helvetica'

# Data
embark_count = df['Embarked'].value_counts(ascending=True)
embark_count = embark_count.rename(index={'C': 'Cherbourg', 'Q':
'Queenstown', 'S': 'Southampton'})

# Create the plot
```

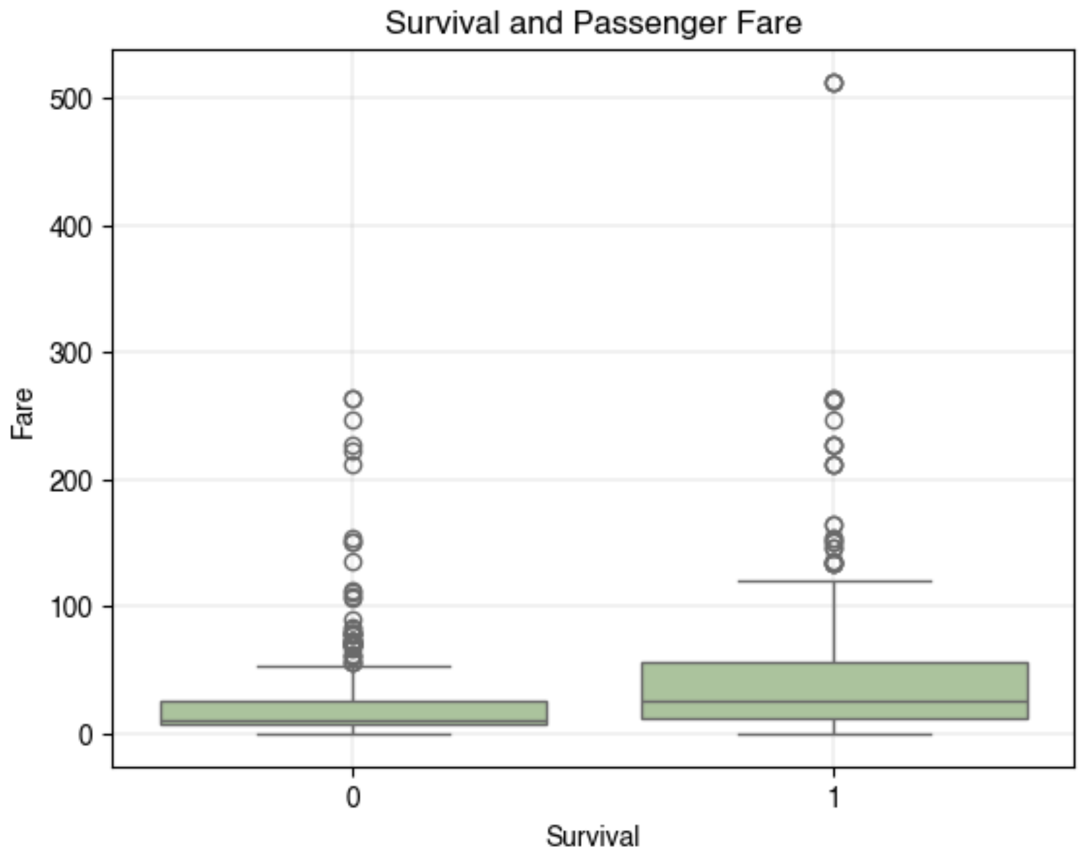
```
fig, ax = plt.subplots()

# Horizontal bar chart
bars = ax.barh(embark_count.index, embark_count.values, color = colors)

# Add value labels beside each bar
for bar in bars:
    width = bar.get_width()
    ax.annotate(f'{int(width)}',            # Value text
                xy=(width, bar.get_y() + bar.get_height()/2),
                xytext=(3, 0),              # Offset (x, y)
                textcoords="offset points",
                ha='left', va='center', fontsize=10)

#Titles and Labels
ax.set_title('Passenger Count by Port of Embarkation')
ax.set_xlabel('Number of Passengers')
ax.set_ylabel('Embarkation Port')
ax.set_axisbelow(True)
plt.grid(linewidth = 0.25)

# Show the plot
plt.show()
```




```
pclass_count = df['Pclass'].value_counts(ascending = True)
pclass_count = pclass_count.rename(index={1 : 'First Class', 2 : 'Second
Class', 3 : 'Third Class'})

fig, ax = plt.subplots()
bars = plt.barh(pclass_count.index, pclass_count.values, color=colors)
for bar in bars:
    width = bar.get_width()
    plt.annotate(f'{int(width)}',
                xy=(width, bar.get_y() + bar.get_height()/2),
                xytext=(3, 0),
                textcoords="offset points",
                ha='left', va='center', fontsize=10)

plt.title('Passenger Class Count')
plt.xlabel('Number of Passengers')
ax.set_axisbelow(True)
plt.grid(linewidth = 0.25)

plt.show()
```

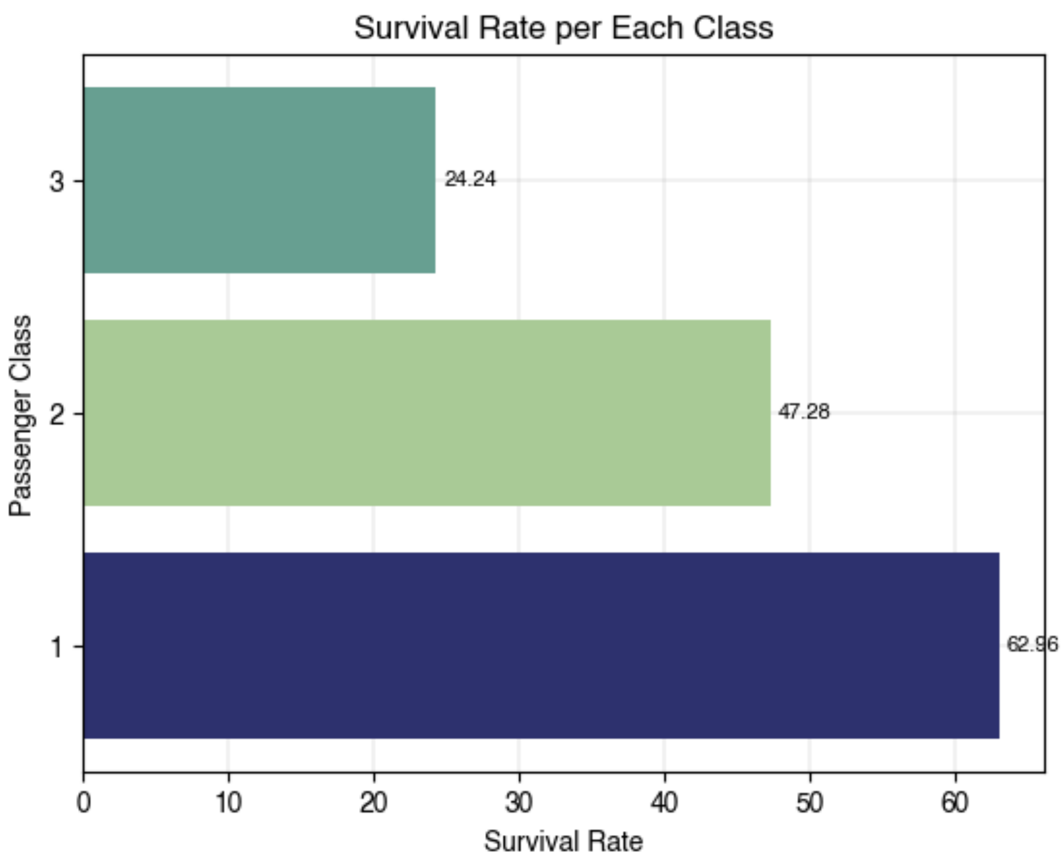



```
sns.boxplot(x = 'Survived', y='Fare', data=df, color = '#a9ca96')
plt.title('Survival and Passenger Fare')
plt.xlabel('Survival')
plt.ylabel('Fare')
plt.grid(linewidth = 0.25)
```




```
survival_pclass = df.groupby('Pclass')['Survived'].mean()
survival_pclass = survival_pclass*100
print(survival_pclass)

fig, ax = plt.subplots()
ax.set_axisbelow(True)
plt.grid(linewidth = 0.25)
bars = plt.barh(survival_pclass.index, survival_pclass.values, color =
colors)
for bar in bars:
    width = bar.get_width()
    plt.annotate(f'{width:.2f}',
            xy=(width, bar.get_y() + bar.get_height()/2),
            xytext=(3, 0),
            textcoords="offset points",
            ha='left', va='center', fontsize=8)

plt.xlabel('Survival Rate')
plt.ylabel('Passenger Class')
plt.title('Survival Rate per Each Class')
```

```
plt.yticks([3, 2, 1])

plt.show()

Pclass
1    62.962963
2    47.282609
3    24.236253
Name: Survived, dtype: float64
```




```
gender_survival=df.groupby('Sex')['Survived'].mean()
gender_survival = gender_survival.sort_values(ascending=True)
gender_survival = gender_survival*100
gender_survival = gender_survival.rename(index={'female': 'Female', 'male':
'Male'})


fig, ax = plt.subplots()
ax.set_axisbelow(True)
plt.grid(linewidth = 0.25)
```

```
bars = plt.barh(gender_survival.index, gender_survival.values,
color=colors)
for bar in bars:
    width = bar.get_width()
    plt.annotate(f'{width:.2f}',
            xy=(width, bar.get_y() + bar.get_height()/2),
            xytext=(3, 0),
            textcoords="offset points",
            ha='left', va='center', fontsize=8)
plt.title('Sex and Survival Rate')
plt.xlabel('Survival Rate')
plt.ylabel('Sex')
```

```
Text(0, 0.5, 'Gender')
```