Task5

June 2, 2025

[5]:
```python
import pandas as pd
df = pd.read_csv('train (1).csv')
```

[6]:
```python
df.head()
```

[6]:
```
   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
```

[7]:
```python
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
```

[8]:
```
df.describe()
```

[8]:
```
       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
```

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

[9]:
```
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
```

[10]:
```python
df['Survived'].value_counts()
df['Sex'].value_counts()
df['Pclass'].value_counts()
df['Embarked'].value_counts()
```

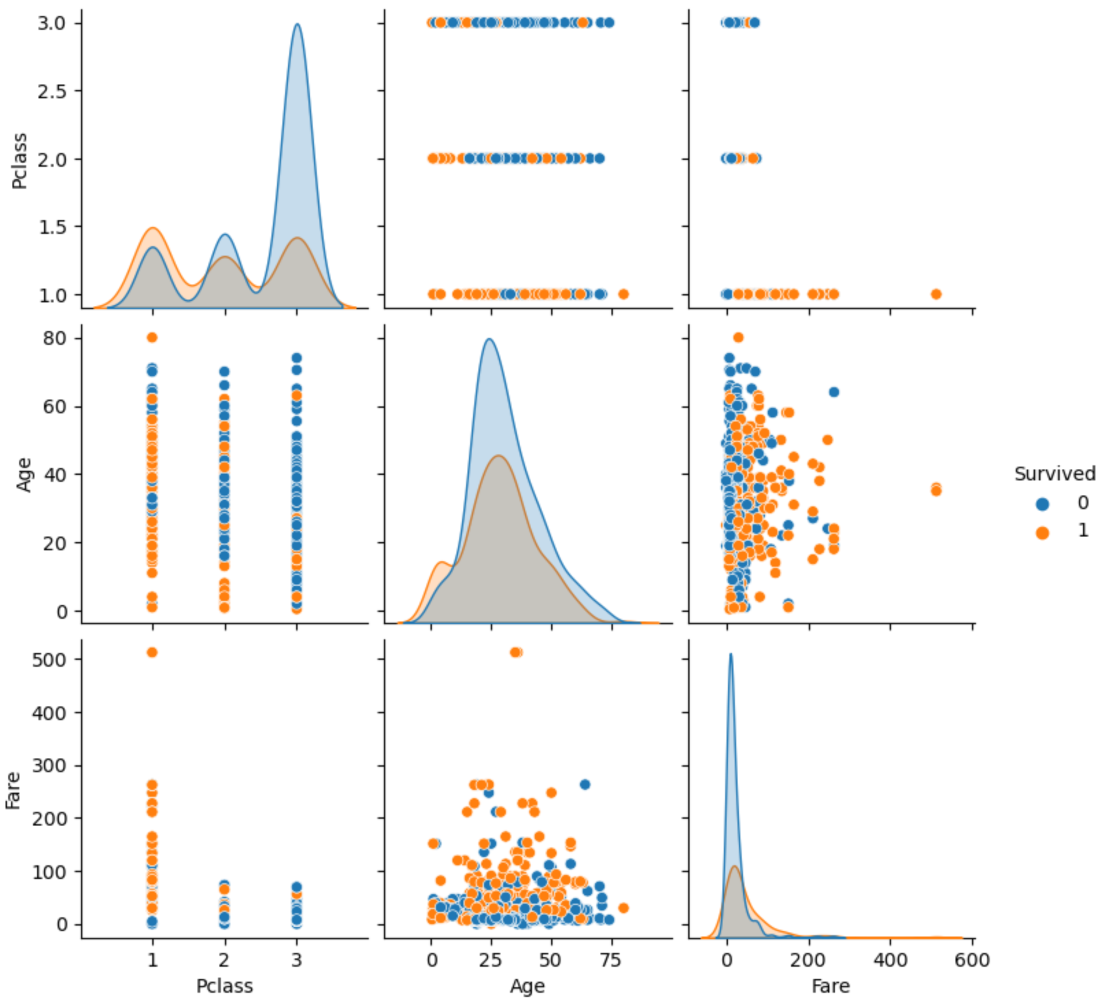
[10]:
```
Embarked
S    644
C    168
Q     77
Name: count, dtype: int64
```

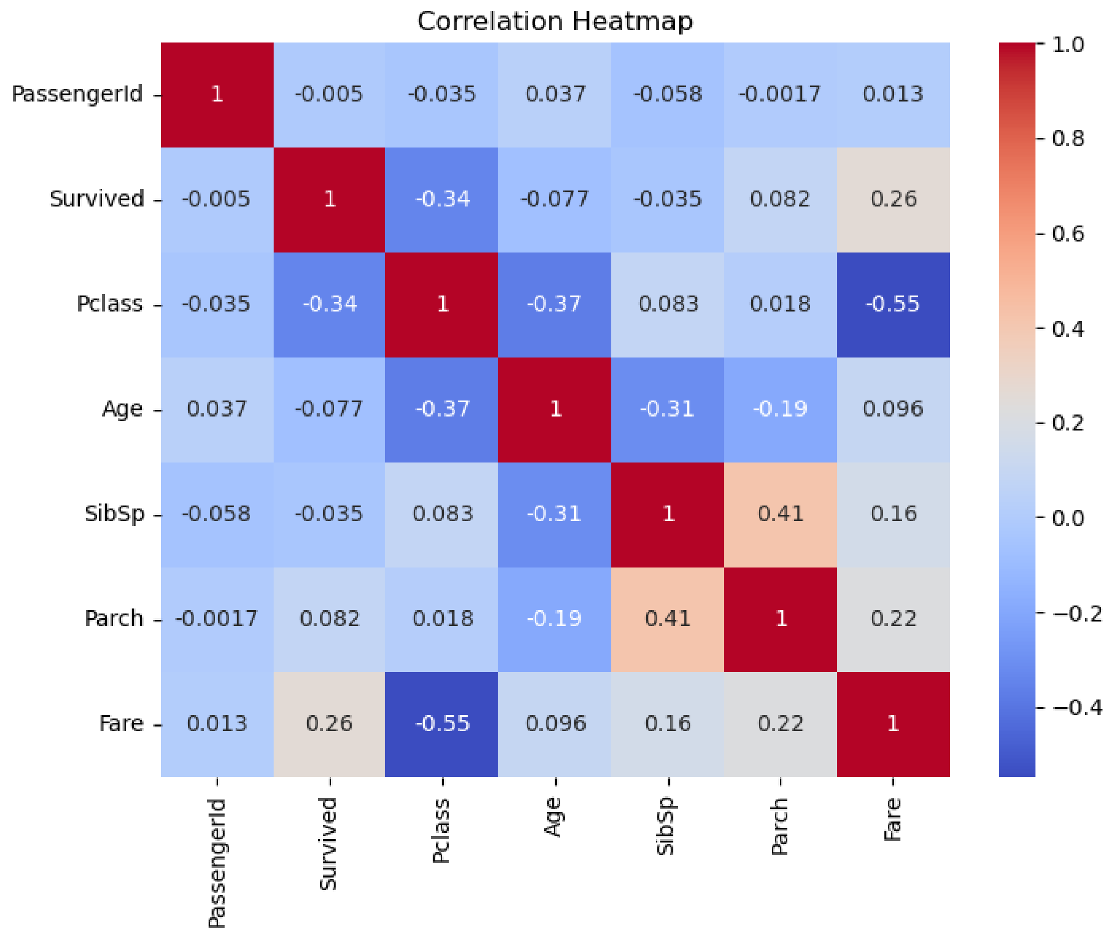
1.Pairplot Observation: Clear separation in survival patterns—passengers with higher fare and lower Pclass had better survival.

2.Heatmap Observation: Fare and Pclass are negatively correlated. Age and Fare are slightly correlated.

[11]:
```python
import seaborn as sns
import matplotlib.pyplot as plt

# Pairplot
sns.pairplot(df[['Survived', 'Pclass', 'Age', 'Fare']], hue='Survived')
plt.show()

# Correlation heatmap for numeric data
numeric_df = df.select_dtypes(include='number')

plt.figure(figsize=(8, 6))
sns.heatmap(numeric_df.corr(), annot=True, cmap='coolwarm')
plt.title("Correlation Heatmap")
plt.show()
```

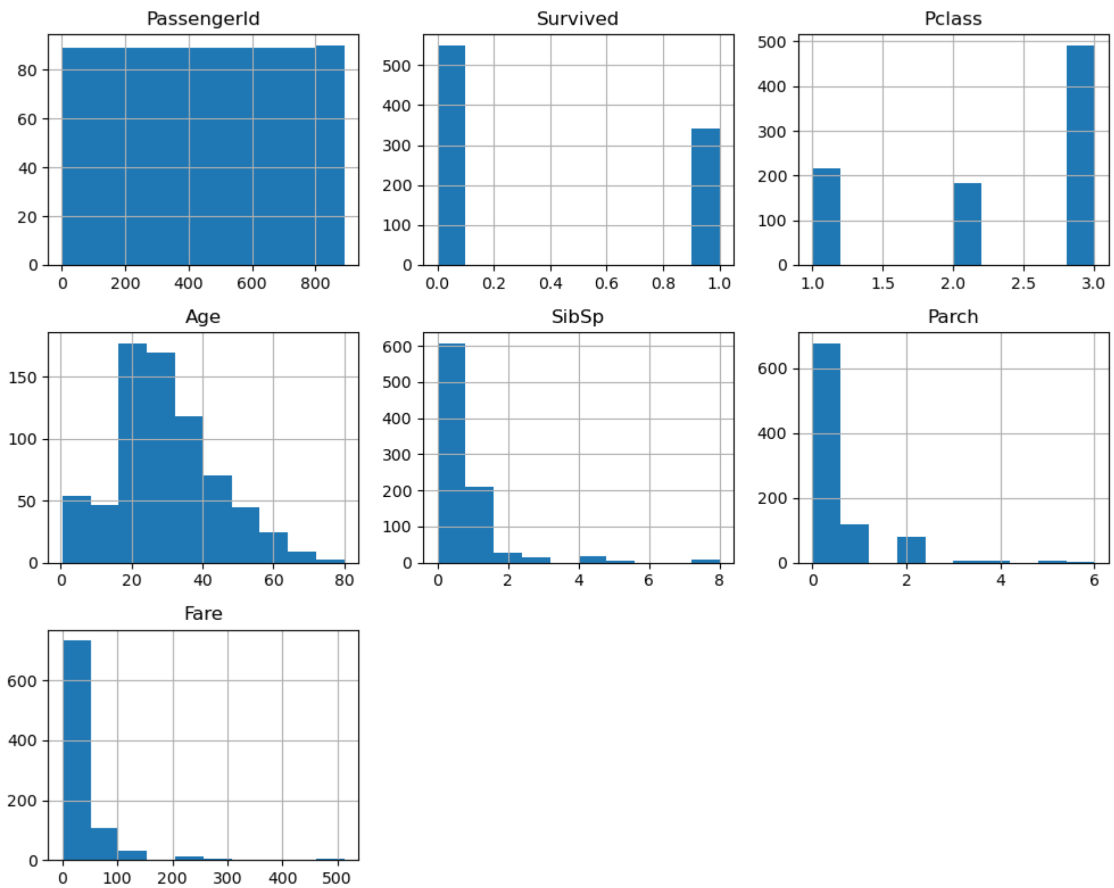
```
/opt/conda/envs/anaconda-panel-2023.05-py310/lib/python3.11/site-
packages/seaborn/axisgrid.py:118: UserWarning: The figure layout has changed to
tight
  self._figure.tight_layout(*args, **kwargs)
```


Pclass
Age
Fare
Survived
0
1



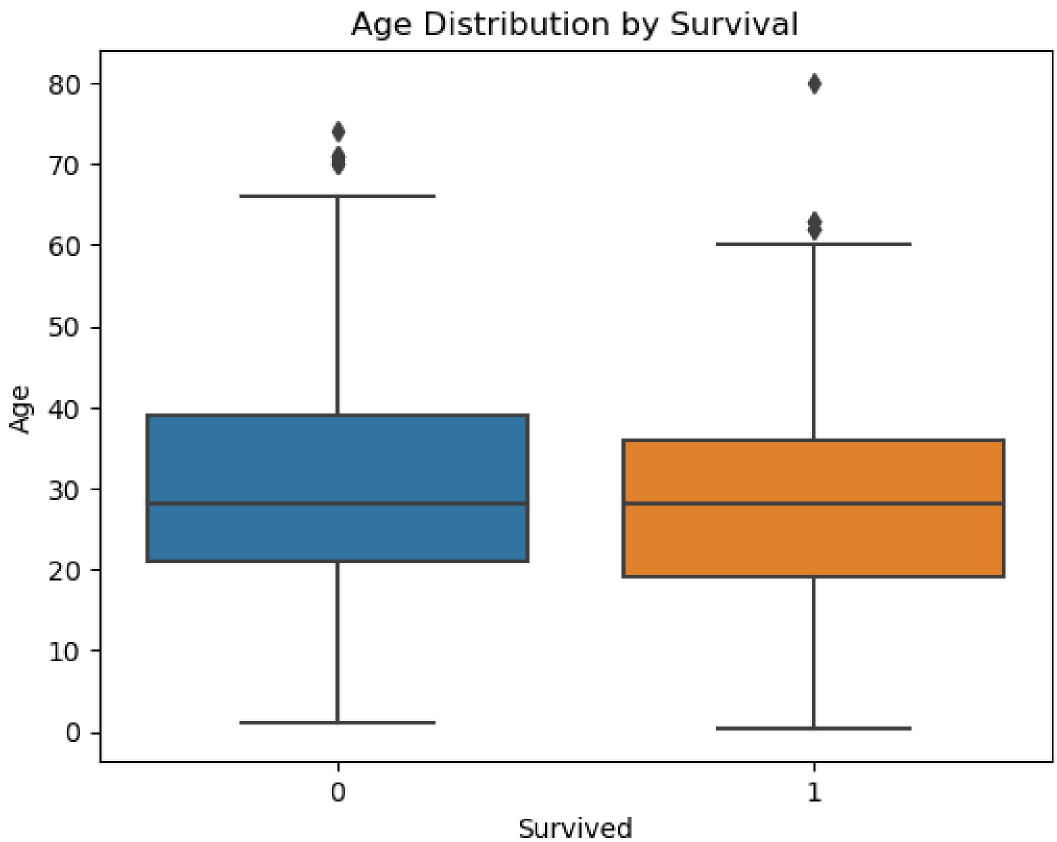
[ ]:
```
Histogram Observation: Most passengers were in their 20s-30s. Fare distribution is right-skewed.
```

[12]:
```
df.hist(figsize=(10, 8))
plt.tight_layout()
plt.show()
```



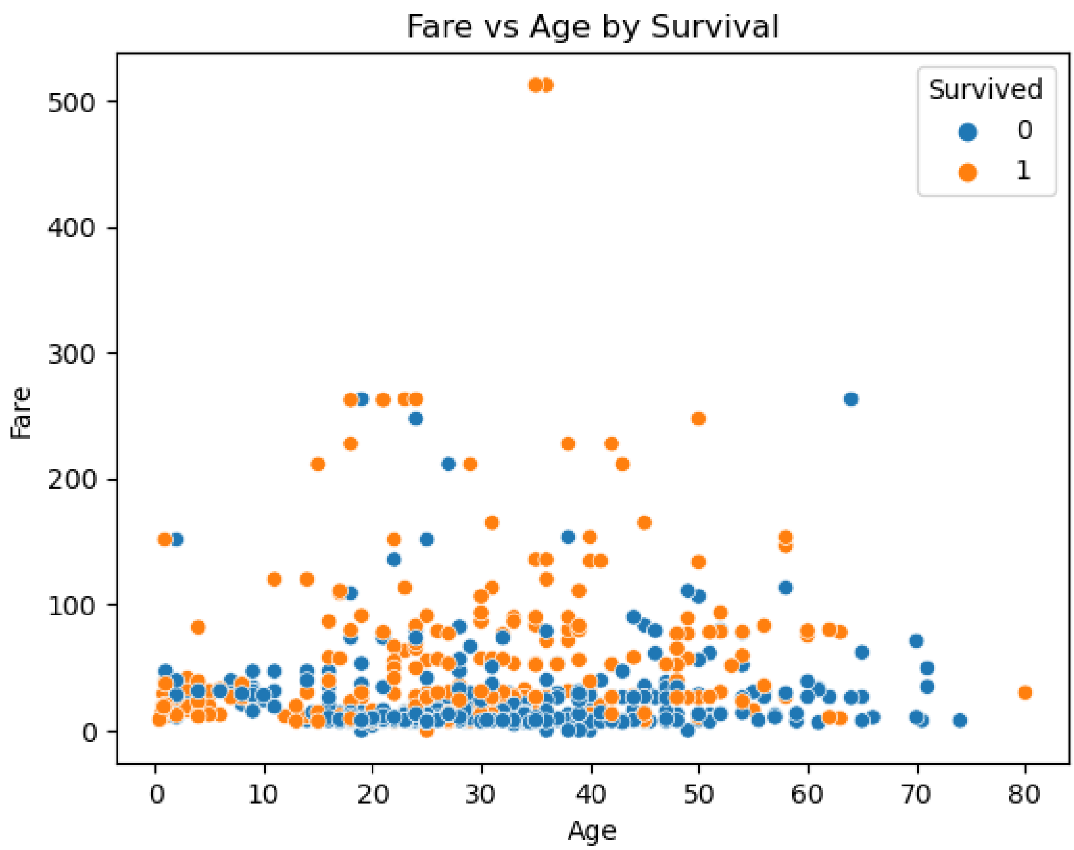

```
[ ]: Boxplot Observation: Median age of survivors is slightly lower than␣
     ↪non-survivors.
```

```
[13]: # Boxplot of age by survival
      sns.boxplot(x='Survived', y='Age', data=df)
      plt.title("Age Distribution by Survival")
      plt.show()
```




```
[ ]: Scatterplot Observation: Higher fares are generally associated with survivors␣
     ↪(especially young and wealthy).
```

```
[14]: sns.scatterplot(x='Age', y='Fare', hue='Survived', data=df)
      plt.title("Fare vs Age by Survival")
      plt.show()
```



Summary of EDA Findings: -*Passenger class (Pclass strongly affects survival: 1st class had more survivors. **S** is a major factor: females survived at a much higher rate. **Fe** is positively correlated with survival—wealthier passengers had better chances. - My **missing vaes** exist in `Cabin` and some in `Age`; this must be handle preprocessing.ing. - The dataset is slightly imbalanced but usable for classification ton trees).