HW

summary report for the Titanic dataset

Tsung-Jiun Huang

2025-02-26

目錄

一、讀取資料

```
# R Interface to Python
library(reticulate)               # Make R and Python interoperable, allowing R to call Python code.
use_python("C:/Users/user/anaconda3/python.exe", required = TRUE)  # Finding Anaconda's Python path
py_config()
```

python: C:/Users/user/anaconda3/python.exe libpython: C:/Users/user/anaconda3/python312.dll pythonhome: C:/Users/user/anaconda3 version: 3.12.7 | packaged by Anaconda, Inc. | (main, Oct 4 2024, 13:17:27) [MSC v.1929 64 bit (AMD64)] Architecture: 64bit numpy: C:/Users/user/anaconda3/Lib/site-packages/numpy numpy_version: 1.26.4

NOTE: Python version was forced by use_python() function

```
library(Hmisc)                    # data analysis and report tools
library(ggplot2)

# read Titanic dataset
titanic_data <- read.csv("C:/Users/user/Downloads/Titanic.csv")
```

二、資料集圖表呈現

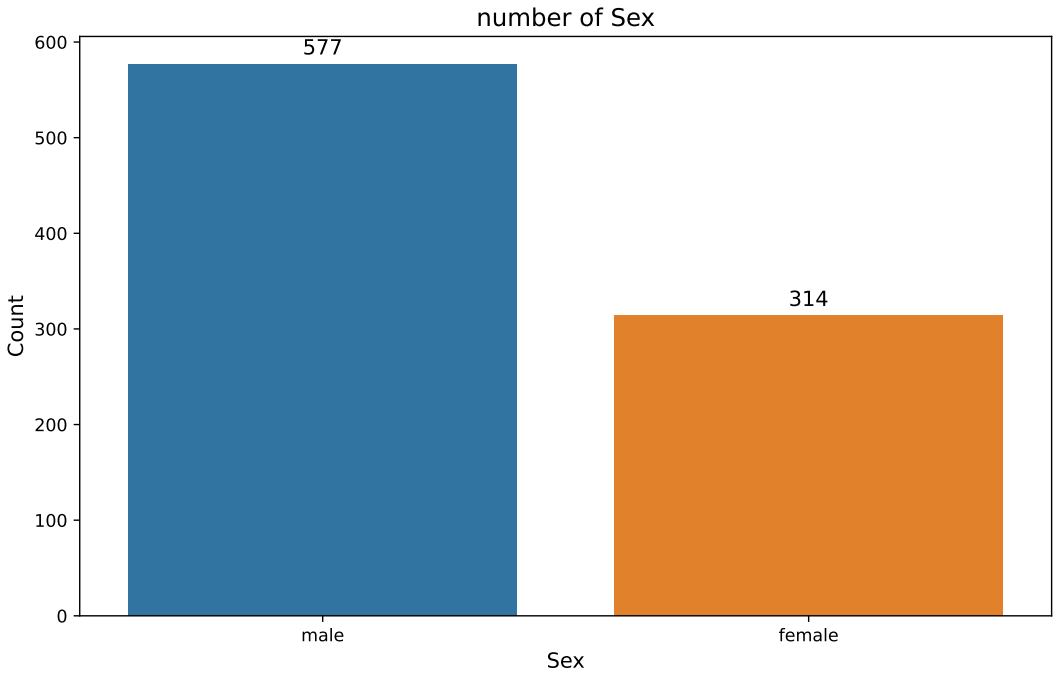
```
# use the command below to install package (run it once)
# ! pip install matplotlib seaborn # (remove # to run it)
# ! pip install shap
```

二-0、讀入套件

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
import shap
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
```

二-1、性別特徵分析

```
titanic_data = pd.read_csv("C:/Users/user/Downloads/Titanic.csv")

# making 'Sex' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='Sex', data=titanic_data, palette='tab10')
# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)
plt.title('number of Sex', fontsize=14)
plt.xlabel('Sex', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.show()
```



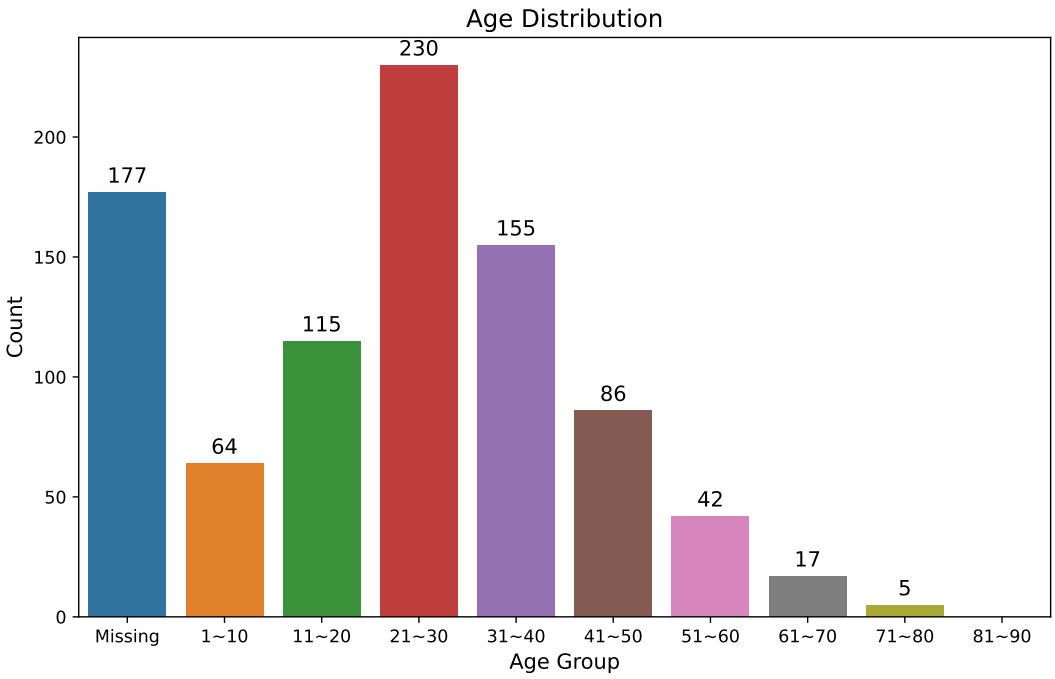
From the chart, there are 577 males and 314 females. We can see that the number of males on the Titanic was higher than that of females, with a male-to-female ratio of approximately 183:100.

二-2、年齡特徵分析

```
# Setting age range (each 10 years old is a group)
bins = list(range(0, 91, 10))  # 0~10, 11~20, ..., 81~90
labels = [f"{i+1}~{i+10}" for i in bins[:-1]]  # Generate corresponding labels

# make NaN values as a class
titanic_data['AgeGroup'] = np.where(
    titanic_data['Age'].isna(), 'Missing',  # Set NaN to 'Missing'
    pd.cut(titanic_data['Age'], bins=bins, labels=labels, right=True)
)

# Plotting the categorized age chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='AgeGroup', data=titanic_data, order=['Missing'] + labels, palette='tab10')

# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)
```

```
[Text(0, 3, '177')]
[Text(0, 3, '64')]
[Text(0, 3, '115')]
[Text(0, 3, '230')]
[Text(0, 3, '155')]
[Text(0, 3, '86')]
[Text(0, 3, '42')]
[Text(0, 3, '17')]
[Text(0, 3, '5')]
[]
```

```
plt.title('Age Distribution', fontsize=14)
plt.xlabel('Age Group', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.show()
```

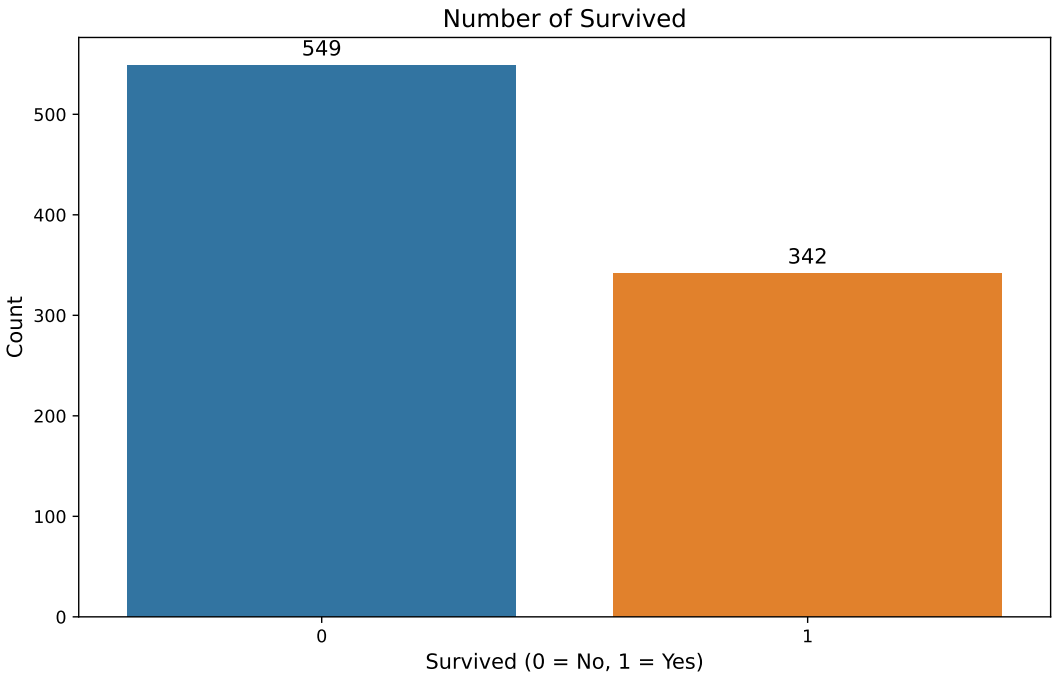


From the chart, there are a total of 891 datas, with 177 missing (NA) values, accounting for nearly 20%. The remaining data shows that passengers aged 21–30 and 31–40 made up the majority, comprising approximately 43% in total.

Additionally, we can see that children aged 1–10 and passengers aged 61 and above accounted for about 10%. This distribution of passengers may have influenced the survival probability on the Titanic.Further analysis can be conducted to determine whether age influenced survival probability.

二-3、生存特徵分析

```
# making 'Survived' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='Survived', data=titanic_data, palette='tab10')

# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)

plt.title('Number of Survived', fontsize=14)
plt.xlabel('Survived (0 = No, 1 = Yes)', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.show()
```



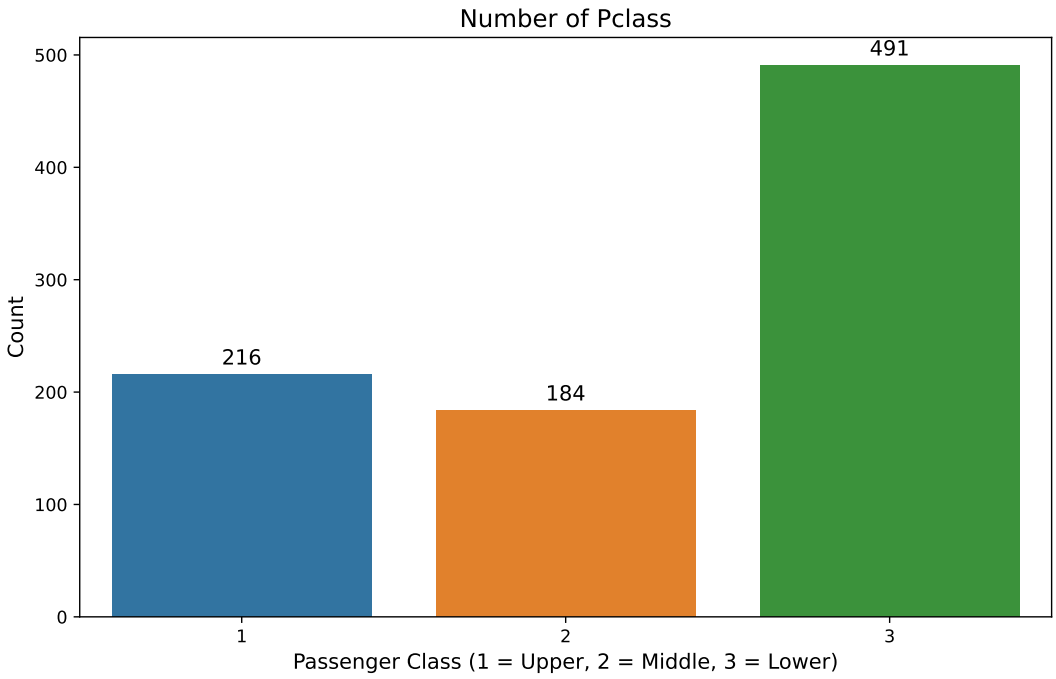
The key for "Survival" is mean for 0 = No and 1 = Yes.

According to the chart, among the 891 passengers on the Titanic, 549 tragically lost their lives, while only 342 survived. This indicates that the overall survival rate in this disaster was merely 38.38%.

We will continue to study whether different characteristics affect the chance of survival.

二-4、社經地位特徵分析

```
# making 'Pclass' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='Pclass', data=titanic_data, palette='tab10')

# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)

plt.title('Number of Pclass', fontsize=14)
plt.xlabel('Passenger Class (1 = Upper, 2 = Middle, 3 = Lower)', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.show()
```

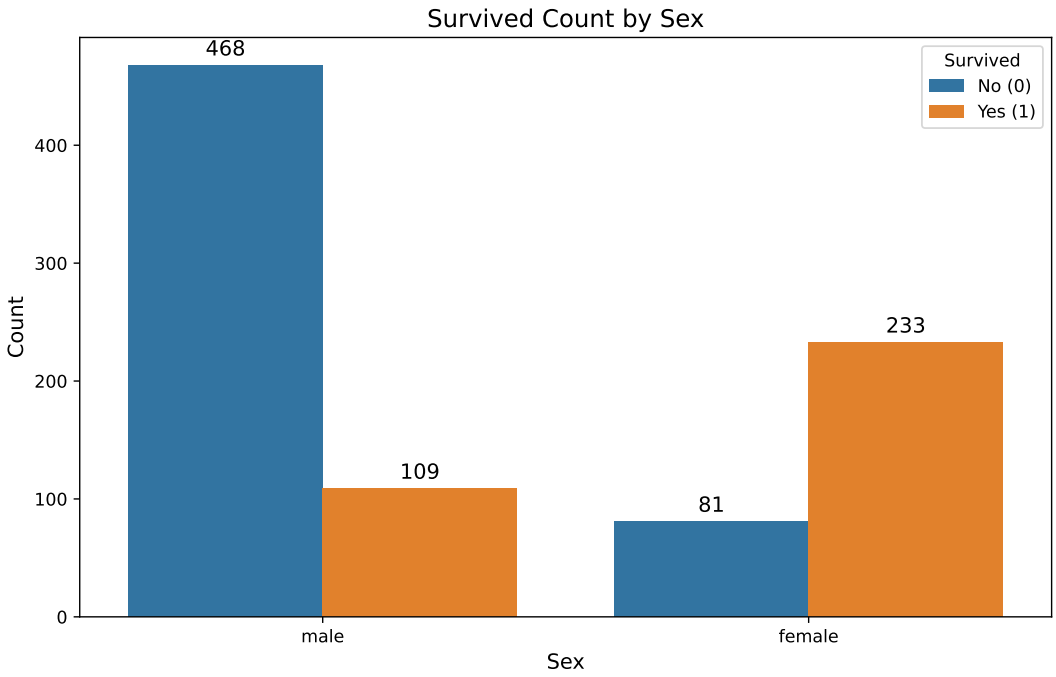


"Pclass" serves as a proxy for socio-economic status (SES), where 1st = Upper, 2nd = Middle, and 3rd = Lower.

We can observe that 55% of the passengers belonged to the Lower class, while the Upper class accounted for 24% and the Middle class for 20%. Further analysis can be conducted to determine whether socio-economic status influenced survival probability.

二-5、性別及存活機率關係分析

```
# making 'Sex' & 'Survived' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='Sex', data=titanic_data, hue='Survived', palette='tab10')

# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)

plt.title("Survived Count by Sex", fontsize=14)
plt.xlabel("Sex", fontsize=12)
plt.ylabel("Count", fontsize=12)
plt.legend(title="Survived", labels=["No (0)", "Yes (1)"])
plt.show()
```



性別與存活率關係

```
# Calculate and display the relationship between sex and survived
survival_Sex_rates = titanic_data.groupby("Sex", as_index=False)["Survived"].mean().round(3)
survival_Sex_rates
```

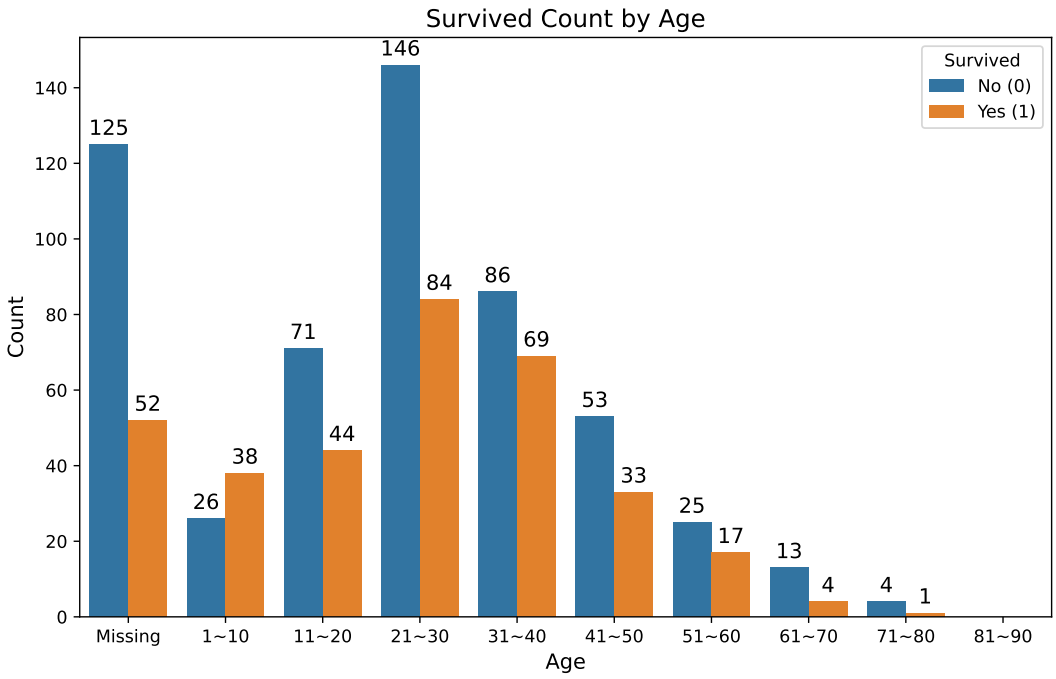
```
      Sex  Survived
0  female     0.742
1    male     0.189
```

By analyzing the relationship between Sex and Survived rate, we can see that the survival rate for females was as high as 74.2%, while for males, it was only 19%.

This suggests that during the rescue efforts, priority may have been given to women, leading to this outcome. Further analysis could be conducted to explore this phenomenon in greater depth.

二-6、年齡及存活機率關係分析

```
# making 'Age' & 'Survived' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='AgeGroup', data=titanic_data, order=['Missing'] + labels, hue='Survived', palette=

# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)

plt.title("Survived Count by Age", fontsize=14)
plt.xlabel("Age", fontsize=12)
plt.ylabel("Count", fontsize=12)
plt.legend(title="Survived", labels=["No (0)", "Yes (1)"])
plt.show()
```



年齡與存活率關係

```
# Calculate and display the relationship between age and survived
survival_Age_rates = titanic_data.groupby("AgeGroup", as_index=False)["Survived"].mean().round(3)
survival_Age_rates
```

```
  AgeGroup  Survived
0    11~20     0.383
1     1~10     0.594
2    21~30     0.365
3    31~40     0.445
4    41~50     0.384
5    51~60     0.405
6    61~70     0.235
7    71~80     0.200
8  Missing     0.294
```

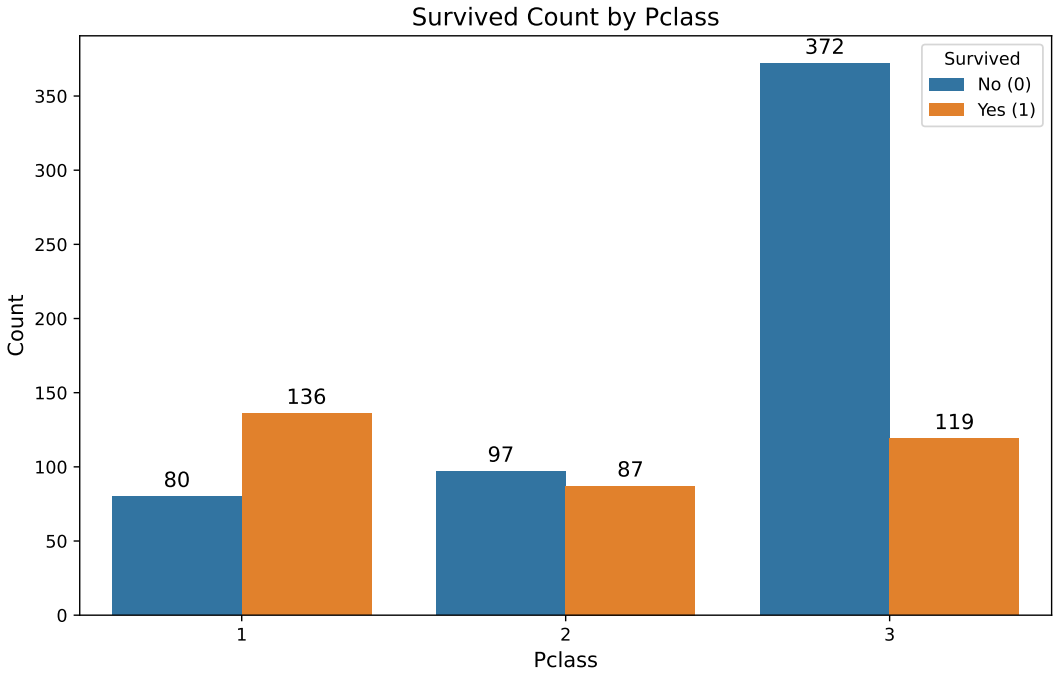
By analyzing the relationship between age and survival rate, we observe that the 1–10 age group had the highest survival rate, with nearly 60% surviving. This suggests that children may have been prioritized during the rescue efforts. For other age groups, there does not appear to be a clear correlation with survival probability.

However, due to the significant number of missing values in this feature, further analysis could incorporate additional features or attempt to impute the missing values to improve accuracy.

二-7、社經地位及存活機率關係分析

```
# making 'Pclass' & 'Survived' chart
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='Pclass', data=titanic_data, hue='Survived', palette='tab10')
```

```
# Display the value on each bar
for container in ax.containers:
    ax.bar_label(container, fmt='%d', label_type='edge', fontsize=12, padding=3)

plt.title("Survived Count by Pclass", fontsize=14)
plt.xlabel("Pclass", fontsize=12)
plt.ylabel("Count", fontsize=12)
plt.legend(title="Survived", labels=["No (0)", "Yes (1)"])
plt.show()
```



年齡與存活率關係

```
# Calculate and display the relationship between Pclass and survived
survival_Pclass_rates = titanic_data.groupby("Pclass", as_index=False)["Survived"].mean().round(3)
survival_Pclass_rates
```

```
   Pclass  Survived
0       1     0.630
1       2     0.473
2       3     0.242
```

By analyzing the relationship between socio-economic status and survival rate, where 1 represents the highest status and 3 represents the lowest.

We observe that higher socio-economic status was associated with a higher survival rate. This may be significantly related to cabin location or rescue priority.

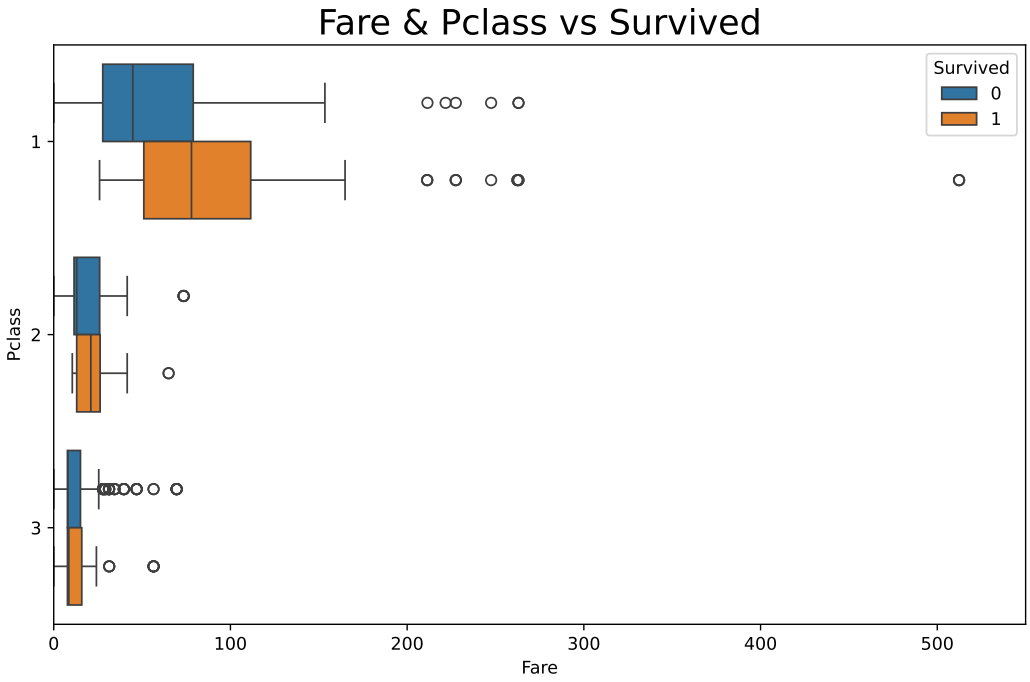
Passengers in the first and second classes had a survival rate exceeding 50%, while the survival rate for third-class passengers was only 24%.

二-8、船票價格、社經地位及存活率關係分析

```
# making 'Fare' & 'Pclass' & 'Survived' chart
fig, ax = plt.subplots(figsize=(10, 6))
sns.boxplot(y='Pclass', x='Fare', hue='Survived', data=titanic_data, orient='h', ax=ax, palette="tab10")

# setting chart's title
ax.set_title('Fare & Pclass vs Survived', fontsize=20)

# Limit the X-axis range to make the chart more visually appealing
ax.set_xlim(0, 550)
```

```
(0.0, 550.0)
```

```
# Calculate the median
fare_median_table = pd.pivot_table(titanic_data, values=['Fare'], index=['Pclass'], columns=['Survived']
plt.show()
```



船票價格、社經地位及存活機率關係

```
# print the table
print(fare_median_table)
```

```
           Fare
Survived      0       1
Pclass
1         44.75  77.958
2         13.00  21.000
3          8.05   8.517
```

By analyzing the relationship between Fare, Pclass, and survival rate, we can see that survivors paid higher fares. This suggests that passengers who paid more or had a higher social status were more

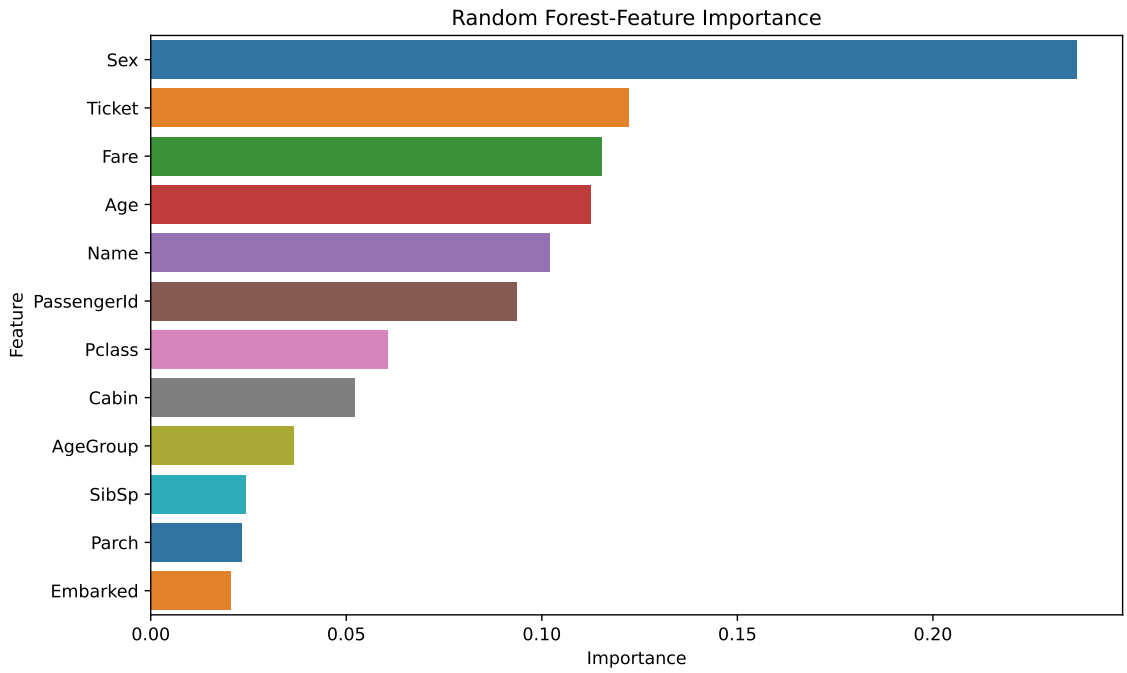
likely to survive the disaster.

三、挑選出對於存活機率來說，最重要與最不重要的特徵。

```
df = titanic_data.copy()

# use labelencoder to handle categorical_cols
categorical_cols = df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    df[col] = LabelEncoder().fit_transform(df[col].astype(str))

# defined x & y (predict survived or not)
X = df.drop(columns=['Survived'])
y = df['Survived']

# split training data and test data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=2)

# trianing random forest model
rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
```

```
RandomForestClassifier(random_state=42)
```

```
# earning feature_importance
feature_importance = rf.feature_importances_
feature_names = X.columns

# DataFrame and sort
importance_df = pd.DataFrame({'Feature': feature_names, 'Importance': feature_importance})
importance_df = importance_df.sort_values(by='Importance', ascending=False)

# show the importance chart
plt.figure(figsize=(10, 6))
sns.barplot(x='Importance', y='Feature', data=importance_df, palette="tab10")
plt.title('Random Forest-Feature Importance')
plt.show()
```



From the Feature Importance chart of the Random Forest model, we can see that "sex" has a significant impact on survival rate, with an importance value greater than 0.2. Next, features such as "Ticket," "Fare," and "Age" also have importance values above 0.1.

This indicates that "sex" is the most important feature in this dataset, greatly influencing survival probability. Features like "ticket", "fare" and "age" also hold considerable importance and should be considered, while other features appear to be less significant.