KHOA CÔNG NGHỆ THÔNG TIN TRƯỜNG ĐẠI HỌC SÀI GÒN

BÁO CÁO QUÁ TRÌNH THAM GIA CUỘC THI TITANIC(KAGGLE)

Titanic Disaster Survival Prediction using Ensemble Learning

Thành viên nhóm

Võ Hoàng Thông - 3123410363;

Phan Thanh Thịnh - 3123410360;

Lê Văn Thông - 3123410362

GVHD: TS. Đỗ Như Tài

MỤC LỤC

Tóm tắt (Abstract) 3
1. Giới thiệu (Introduction) 3

Bài toán: Dự đoán Sống sót trên tàu Titanic 3

2. Chuẩn bị vấn đề (Prepare Problem) 5

2.1 Khai báo thư viện 5

2.2 Nạp dữ liệu 5

3. Phân tích dữ liệu (Analyze Data) 5

3.1 Thống kê mô tả 5

3.2 Hiển thị dữ liệu (Visualize Data) 8

4. Chuẩn bị dữ liệu (Prepare Data) 13

4.1 Làm sạch dữ liệu 13

4.2 Phân chia dữ liệu 17

5. Kết luận (Conclusion) 19

Tài liệu tham khảo (References) 20

Tóm tắt (Abstract)

-Thảm họa chìm tàu RMS Titanic vào năm 1912 đã trở thành một nghiên cứu điển hình trong phân tích dữ liệu và học máy. Nghiên cứu này tập trung vào việc phát triển một mô hình dự đoán khả năng sống sót của hành khách dựa trên các đặc trưng nhân khẩu học và thông tin chuyến đi. Chúng tôi đề xuất một quy trình toàn diện bao gồm khám phá dữ liệu (EDA), tiền xử lý dữ liệu (data preprocessing) chuyên sâu, kỹ thuật trích xuất đặc trưng (feature engineering) sáng tạo, và xây dựng mô hình Ensemble Voting Classifier. Các kỹ thuật xử lý giá trị khuyết, mã hóa đặc trưng phân loại, và tạo đặc trưng mới như Title (Danh xưng) và FamilySize (Quy mô gia đình) đã được áp dụng. Mô hình Ensemble, kết hợp giữa Random Forest, XGBoost, và Support Vector Classifier (SVC), được huấn luyện và đánh giá trên bộ dữ liệu Titanic từ Kaggle. Kết quả thực nghiệm cho thấy mô hình Ensemble của chúng tôi đạt được hiệu suất vượt trội, với độ chính xác cao, chứng minh tính hiệu quả của phương pháp này trong việc giải quyết bài toán phân loại nhị phân phức tạp.

Từ khóa (Keywords): Titanic Survival Prediction, Binary Classification, Machine Learning, Ensemble Learning, Voting Classifier, Feature Engineering, Data Preprocessing.

1. Giới thiệu (Introduction)

Thảm họa Titanic, xảy ra vào đêm 14 rạng sáng 15 tháng 4 năm 1912, là một trong những tai nạn hàng hải bi thảm nhất trong lịch sử. Trong số 2.224 hành khách và thủy thủ đoàn, chỉ có 710 người sống sót. Sự kiện này đã tạo ra một bộ dữ liệu phong phú, cho phép các nhà khoa học dữ liệu và chuyên gia học máy khám phá các yếu tố ảnh hưởng đến khả năng sống sót của từng cá nhân.

Bài toán dự đoán sống sót trên tàu Titanic là một thách thức phân loại nhị phân kinh điển, nơi mô hình cần xác định liệu một hành khách có sống sót hay không (0 hoặc 1). Các yếu tố như giới tính, tuổi tác, hạng vé và quy mô gia đình được cho là có ảnh hưởng đáng kể đến tỷ lệ sống sót, phản ánh các quy tắc cứu hộ ưu tiên "phụ nữ và trẻ em trước" cũng như sự khác biệt về tầng lớp xã hội.

Nghiên cứu này nhằm mục đích xây dựng một giải pháp mạnh mẽ và chính xác cho bài toán này, tận dụng các kỹ thuật học máy hiện đại. Chúng tôi sẽ trình bày chi tiết quy trình từ việc chuẩn bị dữ liệu thô, xử lý các thách thức của nó, đến việc xây dựng và đánh giá một mô hình Ensemble phức tạp.

Bài toán: Dự đoán Sống sót trên tàu Titanic (Titanic Disaster Survival)

1. Định nghĩa vấn đề (Define Problem) Xây dựng một mô hình dự đoán khả năng sống sót của hành khách trên con tàu Titanic. Dựa vào các thông tin cho trước của hành khách

(như tuổi, giới tính, hạng vé,…), mô hình sẽ phân loại hành khách vào hai nhóm: Sống sót (Survived) hoặc Không sống sót (Not Survived).

Đây là một bài toán phân loại nhị phân (binary classification), vì mục tiêu là dự đoán một trong hai kết quả có thể xảy ra.

2. Mô tả (Description) Bộ dữ liệu "Titanic: Machine Learning from Disaster" là một trong những bộ dữ liệu kinh điển nhất, thường được sử dụng cho người mới bắt đầu trong lĩnh vực học máy. Dữ liệu chứa thông tin nhân khẩu học và thông tin chuyến đi của một phần hành khách trên chuyến tàu RMS Titanic định mệnh vào năm 1912. Vụ chìm tàu Titanic là một trong những thảm họa hàng hải nghiêm trọng nhất trong lịch sử, và bộ dữ liệu này được tạo ra để khám phá xem những yếu tố nào đã ảnh hưởng đến cơ hội sống sót của một người.

3. Dữ liệu vào (Features) Các đặc điểm (thuộc tính) của mỗi hành khách được sử dụng để dự đoán:

Pclass: Hạng vé (1 = Hạng 1, 2 = Hạng 2, 3 = Hạng 3).

Sex: Giới tính (male, female).

Age: Tuổi của hành khách (tính bằng năm).

SibSp: Số lượng anh chị em / vợ chồng đi cùng trên tàu.

Parch: Số lượng cha mẹ / con cái đi cùng trên tàu.

Fare: Giá vé mà hành khách đã trả.

Cabin: Số cabin của hành khách.

Embarked: Cảng lên tàu (C = Cherbourg, Q = Queenstown, S = Southampton).

4. Kết quả (Target) Đầu ra cần dự đoán là cột Survived:

Survived: Cột này cho biết hành khách có sống sót hay không.

0 = Không sống sót (No)

1 = Sống sót (Yes)

2. Chuẩn bị vấn đề (Prepare Problem)

2.1. Khai báo thư viện (Load Libraries)

```python
# Load libraries
import os, sys
from IPython import display
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import joblib

from sklearn.preprocessing import OneHotEncoder, LabelEncoder, OrdinalEncoder
from sklearn.preprocessing import MinMaxScaler, StandardScaler
from sklearn.model_selection import train_test_split

import warnings

%matplotlib inline
# plt.rcParams["figure.figsize"] = (12, 6)
# plt.rcParams['figure.dpi'] = 100

warnings.filterwarnings("ignore")
```

Python

2.2. Nạp dữ liệu (Load Dataset)

```python
data_path = "train.csv"
df = pd.read_csv(data_path)
df.head()
```

Python

3. Phân tích dữ liệu (Analyze Data)

3.1. Thống kê mô tả (Descriptive Statistics)

(1) Hiển thị một số thông tin về dữ liệu

- Số dòng, số cột của dữ liệu
- Kiểu dữ liệu của từng cột
- 5 dòng đầu và 5 dòng cuối của bảng dữ liệu
- Thông tin chung về dữ liệu

TÍNH TOÁN VẸN DỮ LIỆU:
+ Có giá trị Null: True
→ Các dòng có giá trị Null:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
884	885	0	3	Sutehall, Mr. Henry Jr	male	25.0	0	0	SOTON/OQ 392076	7.0500	NaN	S
885	886	0	3	Rice, Mrs. William (Margaret Norton)	female	39.0	0	5	382652	29.1250	NaN	Q
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

708 rows × 12 columns

+ Có giá trị NaN: True
→ Các dòng có giá trị NaN:

Nhận xét:

Dữ liệu có gì trị null, giá trị nan và không có dòng trùng lặp

(3) ***Các tính chất thống kê trên dữ liệu số***

- Count, Mean, Standard Deviation, Minimum Value
- 25th Percentile, 50th Percentile (Median), 75th Percentile, Maximum Value

	count	mean	std	min	25%	50%	75%	max
PassengerId	891.0	446.000000	257.353842	1.00	223.5000	446.0000	668.5	891.0000
Survived	891.0	0.383838	0.486592	0.00	0.0000	0.0000	1.0	1.0000
Pclass	891.0	2.308642	0.836071	1.00	2.0000	3.0000	3.0	3.0000
Age	714.0	29.699118	14.526497	0.42	20.1250	28.0000	38.0	80.0000
SibSp	891.0	0.523008	1.102743	0.00	0.0000	0.0000	1.0	8.0000
Parch	891.0	0.381594	0.806057	0.00	0.0000	0.0000	0.0	6.0000
Fare	891.0	32.204208	49.693429	0.00	7.9104	14.4542	31.0	512.3292

Nhận xét:

Survived mean= 0.384 cho thấy tập dữ liệu thiên về tỉ lệ tử vong nhiều hơn. Có thể thấy tầng lớp thấp (Pclass=3) chiếm tỷ lệ cao, có thể ảnh hưởng mạnh đến khả năng sống sót. Trung vị median, Điều này có nghĩa là ít nhất 50% số hành khách đi vé hạng 3. Giá trị trung bình (mean ≈ 2.3) cũng cho thấy số lượng hành khách ở các hạng vé thấp (2 và 3) chiếm đa số. Đa số hành khách là thanh niên và người trưởng thành trẻ tuổi. Độ lệch chuẩn (std) khá lớn (14.5), cho thấy độ tuổi rất đa dạng. SibSp (anh chị em/vợ chồng): 75% số hành khách đi cùng tối đa 1 người.

Trung vị (50%) là 0. Phần lớn hành khách đi một mình hoặc theo nhóm nhỏ (2 người). Rất ít hành khách đi theo các gia đình lớn (giá trị max là 8 và 6 cho thấy có những trường hợp ngoại lệ). Fare mean=32, 50%= 14,45, Đây là dấu hiệu của một phân phối lệch phải (right-skewed). Có một số lượng nhỏ vé với giá cực kỳ cao (có thể là vé hạng nhất hoặc cabin đặc biệt, max = 512) đã kéo giá trị trung bình lên cao, trong khi phần lớn hành khách (hơn 75%) trả giá vé dưới 31.

(4) ***Tần số xuất hiện (Distribution) trên dữ liệu phân lớp (Class) và dữ liệu danh mục (Category)***

```
Survived
0    549
1    342
Name: count, dtype: int64
```

Đối với bài toán phân lớp (classification problem), chúng ta cần tính số lần xuất hiện của thuộc tính phân lớp. Điều này là cần thiết cho vấn đề mất cân bằng (highly imbalanced problems) giữa các lớp nhằm cần xử lý đặc biệt trong bước chuẩn bị dữ liệu.

(5) ***Mối tương quan giữa các tính chất (Correlations)***

Sự tương quan (correlation) đề cập đến mối quan hệ giữa hai biến và cách chúng có thể có hoặc không cùng nhau thay đổi.

Phương pháp phổ biến nhất để tính toán tương quan là Pearson's Correlation Coeficient, giả định có một phân phối chuẩn của các thuộc tính liên quan. Tương quan -1 hoặc 1 cho thấy mối tương quan âm hoặc dương đầy đủ tương ứng. Trong khi giá trị 0 hiển thị không tương quan ở tất cả.

$$r = \frac{\sum_{i=1}^{n}(x_i - \hat{x})(y_i - \hat{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \hat{x})^2 \sum_{i=1}^{n}(y_i - \hat{y})^2}}$$

Một số thuật toán học máy như hồi quy tuyến tính và logistic có hiệu suất kém nếu có các thuộc tính tương quan cao trong tập dữ liệu của bạn.

Như vậy, thật sự cần thiết để xem xét tất cả các mối tương quan theo cặp của các thuộc tính trong tập dữ liệu.

Ma trận tương quan (Pearson) cho dữ liệu Titanic:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1.000000	-0.005007	-0.035144	0.036847	-0.057527	-0.001652	0.012658
Survived	-0.005007	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307
Pclass	-0.035144	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500
Age	0.036847	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067
SibSp	-0.057527	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651
Parch	-0.001652	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225
Fare	0.012658	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000

Nhận xét

Pclass -0.338 tương quan nghịch với Survived giảm, Khi số Pclass tăng (tức là hạng vé thấp hơn, từ 1 -> 3), khả năng Survived giảm. Điều này khẳng định rằng hành khách ở hạng 1 có cơ hội sống sót cao hơn nhiều so với hành khách ở hạng 3. Đây là một yếu tố dự đoán rất quan trọng. Fare mối tương quan thuận, khi Fare (giá vé) tăng, khả năng Survived cũng tăng. Điều này hoàn toàn phù hợp với mối tương quan của Pclass, vì giá vé cao hơn tương ứng với hạng vé tốt hơn. Parch 0.082 tương quan thuận nhưng yếu đuối, Có một xu hướng không đáng kể cho thấy tuổi càng cao thì khả năng sống sót càng thấp. Mối quan hệ này có thể phức tạp hơn là tuyến tính (ví dụ: trẻ em được ưu tiên).

Với tương quan các đặc điểm:

Fare và Pclass (-0.550): Đây là mối tương quan nghịch mạnh nhất trong toàn bộ ma trận. Điều này là hiển nhiên: hạng vé càng cao (Pclass càng thấp, ví dụ = 1) thì giá vé (Fare) càng đắt. Mối quan hệ này củng cố tính logic của dữ liệu. SibSp và Parch (0.415): Mối tương quan thuận vừa phải. Ý nghĩa: Những hành khách đi cùng anh chị em/vợ chồng cũng có xu hướng đi cùng cha mẹ/con cái. Điều này hợp lý vì cả hai biến này đều liên quan đến quy mô gia đình. Age và Pclass (-0.369): Mối tương quan nghịch vừa phải.Hành khách ở các hạng vé cao hơn (Pclass thấp hơn) có xu hướng lớn tuổi hơn. Ngược lại, hành khách ở hạng 3 thường trẻ tuổi hơn.

3.2. Hiển thị dữ liệu (Visualize Data)

(1) Hiển thị trên từng tính chất đơn (Univariate Plots)

Box and whisker plots

+ https://www.simplypsychology.org/boxplots.html

+ So sánh các trung vị (median) tương ứng của mỗi ô hộp (box plot). Nếu đường trung vị của một ô hộp nằm bên ngoài ô của một ô hộp so sánh, thì có thể có sự khác biệt giữa hai nhóm.

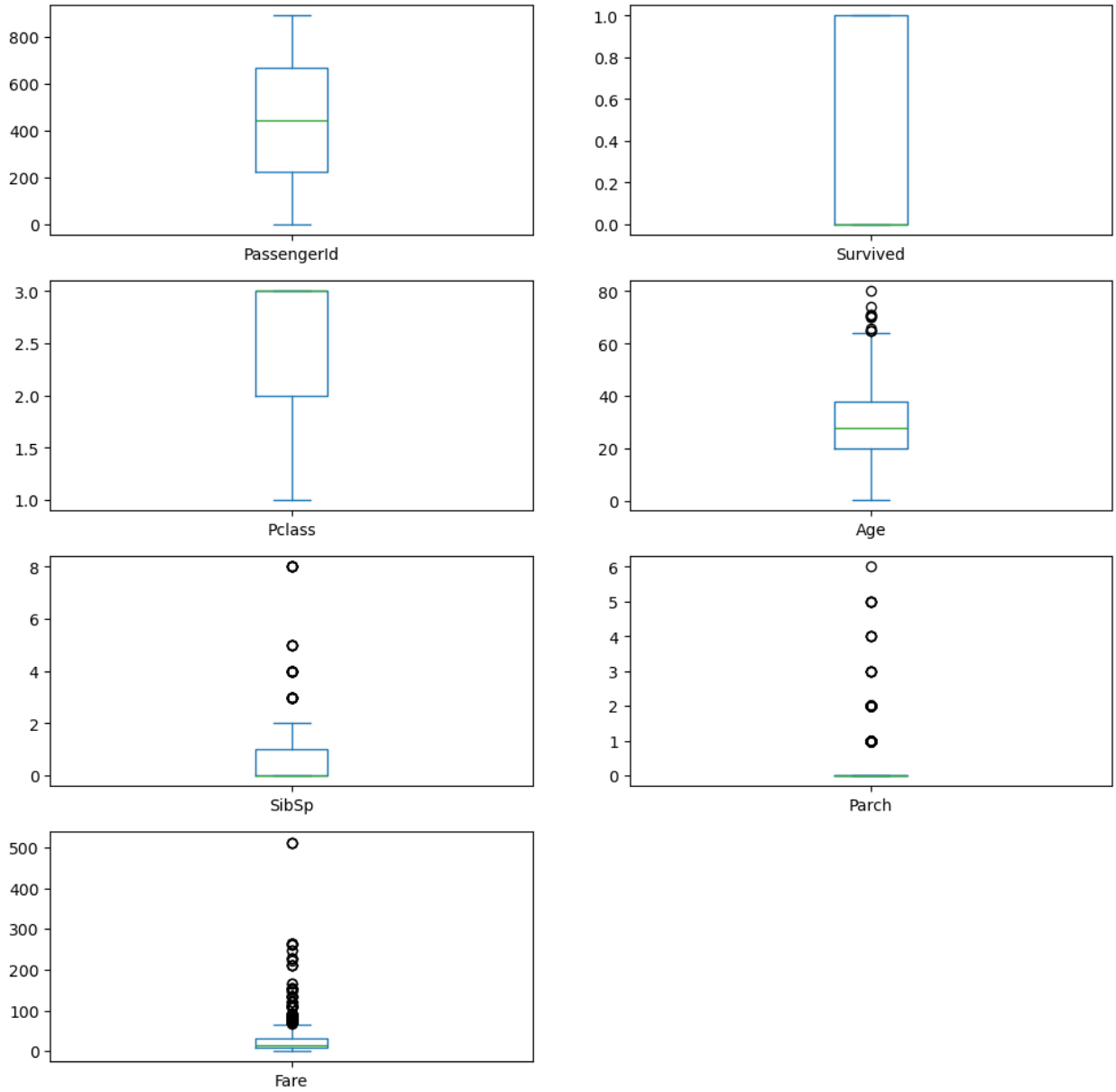
+ So sánh chiều dài hộp để kiểm tra cách dữ liệu được phân tán giữa mỗi mẫu. Hộp càng dài thì dữ liệu càng phân tán. Dữ liệu càng nhỏ càng ít bị phân tán.

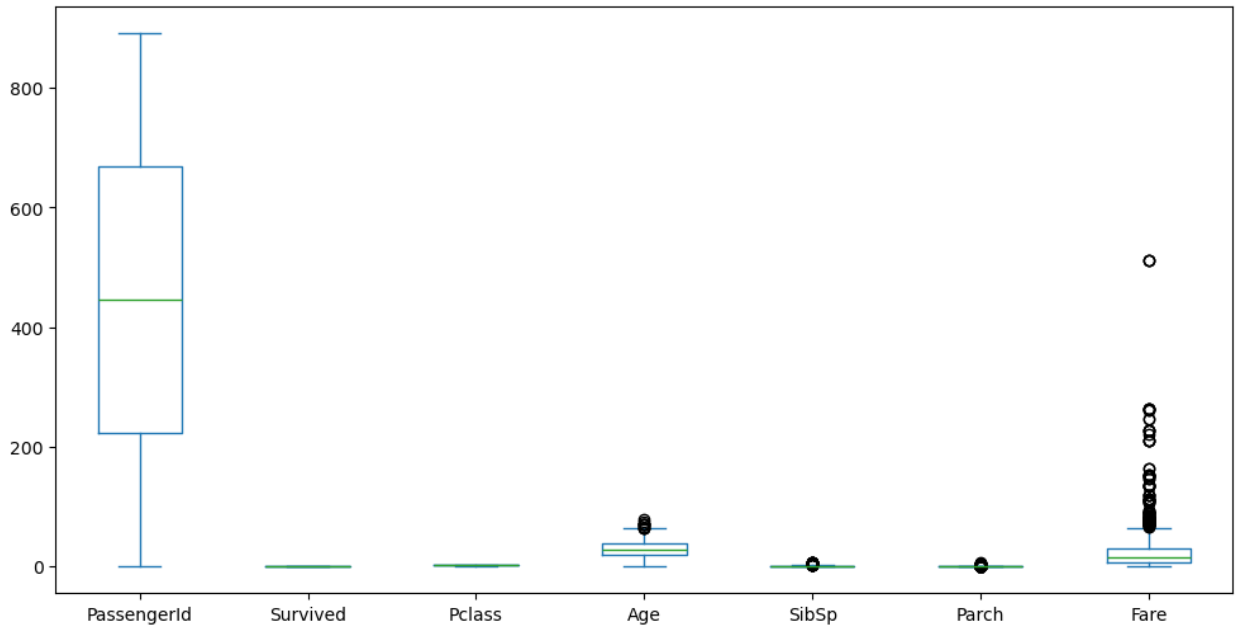
+ Một ngoại lệ (outlier) được định nghĩa là một điểm dữ liệu nằm bên ngoài phần rìa (whiskers) của ô hộp.

+ Kiểm tra hướng lệch của dữ liệu (cân đối, các phần tử tập trung trái, phải). + Median ở giữa hộp và râu (whiskers) ở hai bên như nhau thì phân bố là đối xứng.

+ Median ở gần đáy hộp hơn và nếu râu ngắn hơn ở đầu dưới của hộp, thì phân phối là lệch dương (lệch phải).

+ Median ở gần đầu hộp hơn và nếu râu ngắn hơn ở đầu trên của hộp, thì phân bố bị lệch âm (lệch trái).

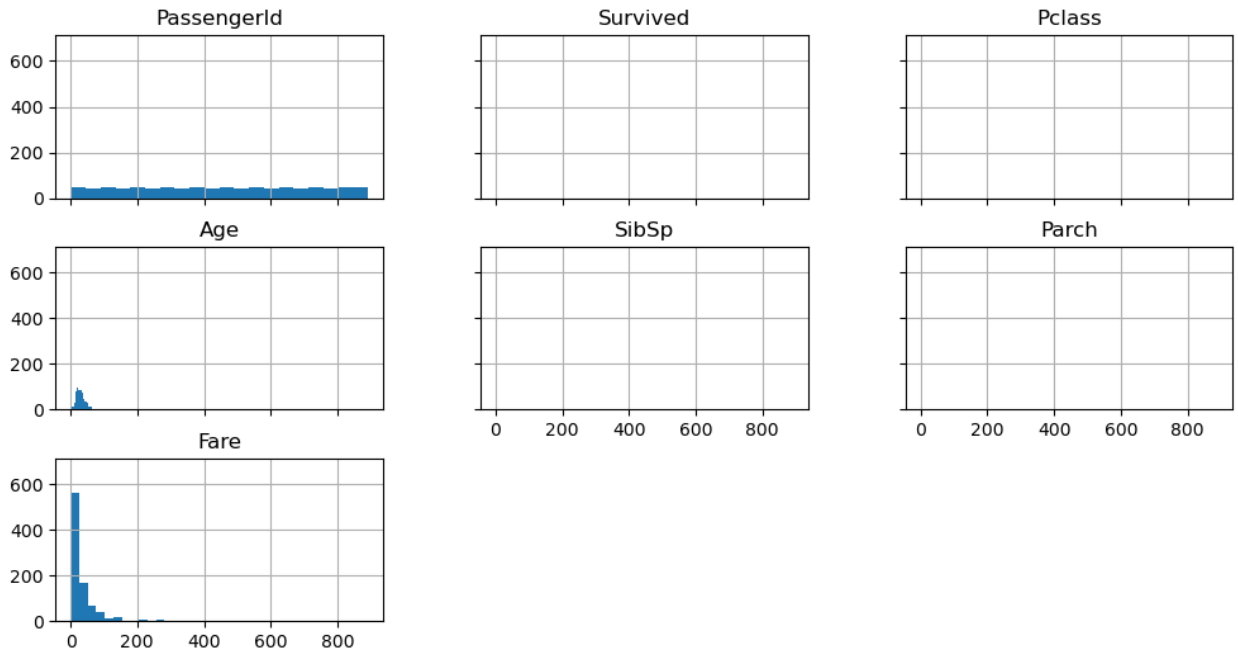




Nhận xét:

Các biểu đồ cho Age, SibSp, Parch, và đặc biệt là Fare có các giá trị ngoại lai (outliers), tức là những giá trị nằm rất xa so với phần lớn dữ liệu.SibSp và Parch: Các giá trị ngoại lai ở đây đại diện cho những gia đình rất đông người, đây là trường hợp hiếm so với phần lớn hành khách. Pclass (Hạng vé): Hộp (IQR - interquartile range) nằm giữa 2 và 3, và đường trung vị (median) nằm ngay ở mức 3. Điều này cho thấy hơn 50% số hành khách đi vé hạng 3. SibSp (Anh chị em/Vợ chồng) và Parch (Cha mẹ/Con cái): Cả hai biểu đồ này đều có hộp bị nén chặt ở phía dưới, với đường trung vị nằm ở mức 0. Điều này chứng tỏ phần lớn hành khách (hơn 75%) đi một mình, không đi cùng người thân. Survived (Sống sót): Vì đây là biến nhị phân (0 hoặc 1), biểu đồ boxplot cho thấy đường trung vị ở mức 0. Điều này một lần nữa khẳng định rằng hơn một nửa số hành khách trong tập dữ liệu đã không qua khỏi.

Biểu đồ Histogram

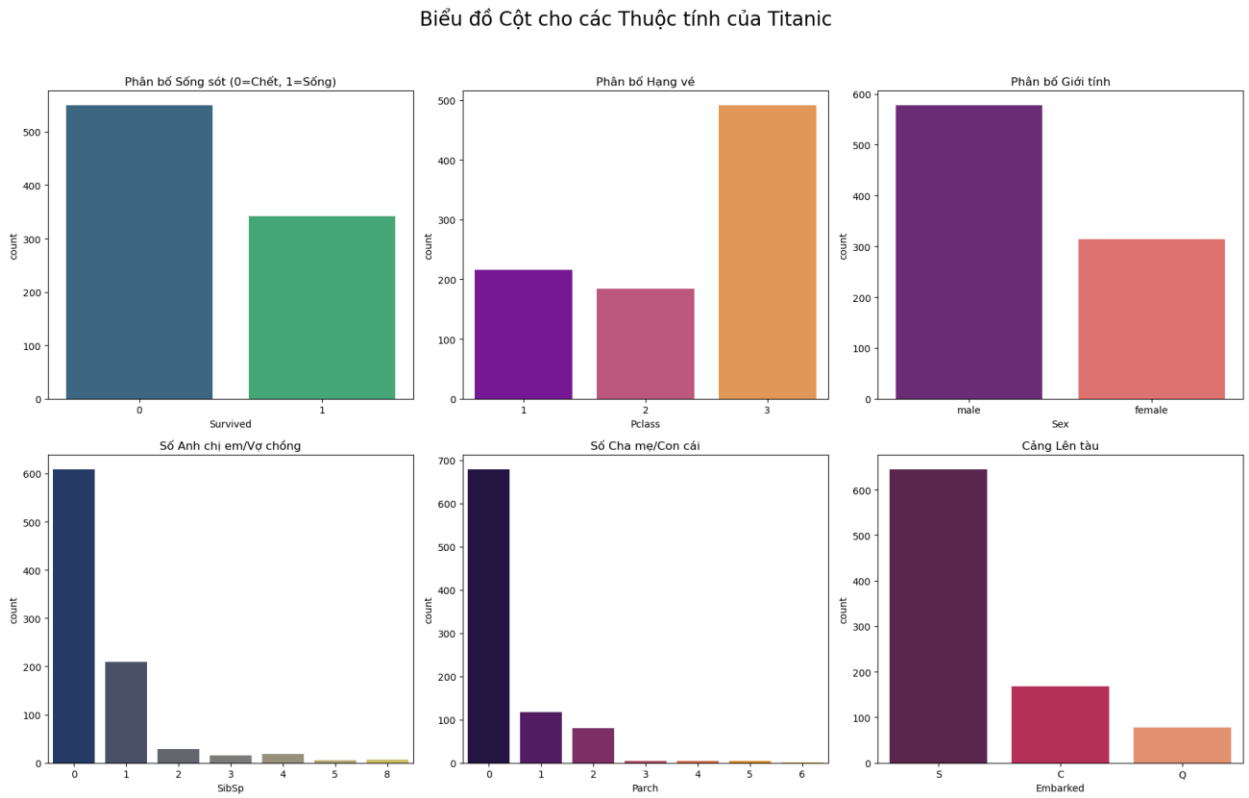


Lý do là vì histogram không phải là công cụ phù hợp để trực quan hóa các biến này.

Histogram dùng để hiển thị tần suất của dữ liệu liên tục được chia thành các khoảng (bins), ví dụ như Age (tuổi) và Fare (giá vé).

Các biến còn lại là dữ liệu rời rạc (chỉ có một vài giá trị cụ thể, ví dụ Survived chỉ là 0 hoặc 1; Pclass là 1, 2, 3). Khi vẽ histogram cho chúng, các giá trị này bị dồn vào những khoảng rất hẹp và có thể không hiển thị rõ ràng.

Bar chart



Nhận xét:

Các biểu đồ cho thấy rõ ràng phần lớn hành khách trên tàu Titanic là nam giới, đi vé hạng 3, và lên tàu tại cảng Southampton (S). Đa số họ đi một mình, thể hiện qua số lượng áp đảo người không có anh chị em/vợ chồng (SibSp=0) hoặc cha mẹ/con cái (Parch=0) đi cùng. Về kết quả, biểu đồ sống sót cho thấy một thực tế nghiệt ngã: số người không qua khỏi (cột 0) cao hơn đáng kể so với số người sống sót (cột 1).

4. Chuẩn bị dữ liệu (Prepare Data)

4.1. Làm sạch dữ liệu (Data Cleaning)

(1) Tạo bảng dữ liệu làm sạch

- Chỉ giữ lại các cột Input, Ouput

```
df_clean = df.copy()
```

(2) Xóa dữ liệu trùng nhau

```
display.display(df_clean[df_clean.duplicated()])
```

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

(3) ***Xử lý giá trị rỗng, không hợp lệ***

```
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
```

```
Số lượng giá trị null TRƯỚC khi xử lý:
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
------------------------------
Đã điền 177 giá trị null trong cột 'Age' bằng giá trị trung bình: 29.70
Đã xóa cột 'Cabin' vì có quá nhiều giá trị null.
Đã điền 2 giá trị null trong cột 'Embarked' bằng giá trị phổ biến nhất: 'S'
------------------------------
```

```
Số lượng giá trị null SAU khi xử lý:
PassengerId    0
Survived       0
Pclass         0
Name           0
...
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

(2) Chuyển đổi dữ liệu danh mục (Category) thành dạng OneHot

Một số thuật toán khi chuyển đổi cột dạng danh mục thành kiểu OneHot thì cho hiệu suất cao hơn.

Bên cạnh đó, khi huấn luyện mô hình với dạng hàm mất mát CategoryEntropy thì cũng cần chuyển thuộc tính phân lớp sang dạng OneHot.

Sex: kh cần vì 0 1 là đủ

Embarked: Đây là ví dụ hoàn hảo. Các giá trị ‘S’, ‘C’, ‘Q’ đại diện cho các cảng khác nhau và không có cảng nào “lớn hơn” hay “tốt hơn” cảng nào.

Dữ liệu gốc:

	Embarked
0	S
1	C
2	S
3	S
4	S

Dữ liệu sau khi One-Hot Encoding:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked_C	Embarked_Q	Embarked_S
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	False	False	True
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	True	False	False
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	False	False	True
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	False	False	True
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	False	False	True

(3) Chuẩn hóa dữ liệu (Data Normalize)

Chuẩn hóa các tính chất để đưa về cùng một miền trị + Min-Max Normalization

$$z = \frac{x - min(x)}{max(x) - min(x)}$$

+ Standard Normalization

$$z = \frac{x - \mu}{\sigma}$$

Lưu ý: Quá trình chuẩn hóa có thể làm trong phần thực nghiệm thuật toán

Việc chuẩn hóa giúp đưa các thuộc tính có thang đo khác nhau (ví dụ: Age từ 0-80, trong khi Fare từ 0-512) về cùng một thang đo chung. Điều này đảm bảo rằng các thuật toán máy học sẽ không bị "thiên vị" và đánh giá quá cao các thuộc tính có giá trị lớn hơn.

Giống như việc so sánh một khoảng cách đo bằng mét và một khoảng cách đo bằng kilomet, bạn cần đưa chúng về cùng một đơn vị để có sự so sánh công bằng.

Dữ liệu sau khi chuẩn hóa:

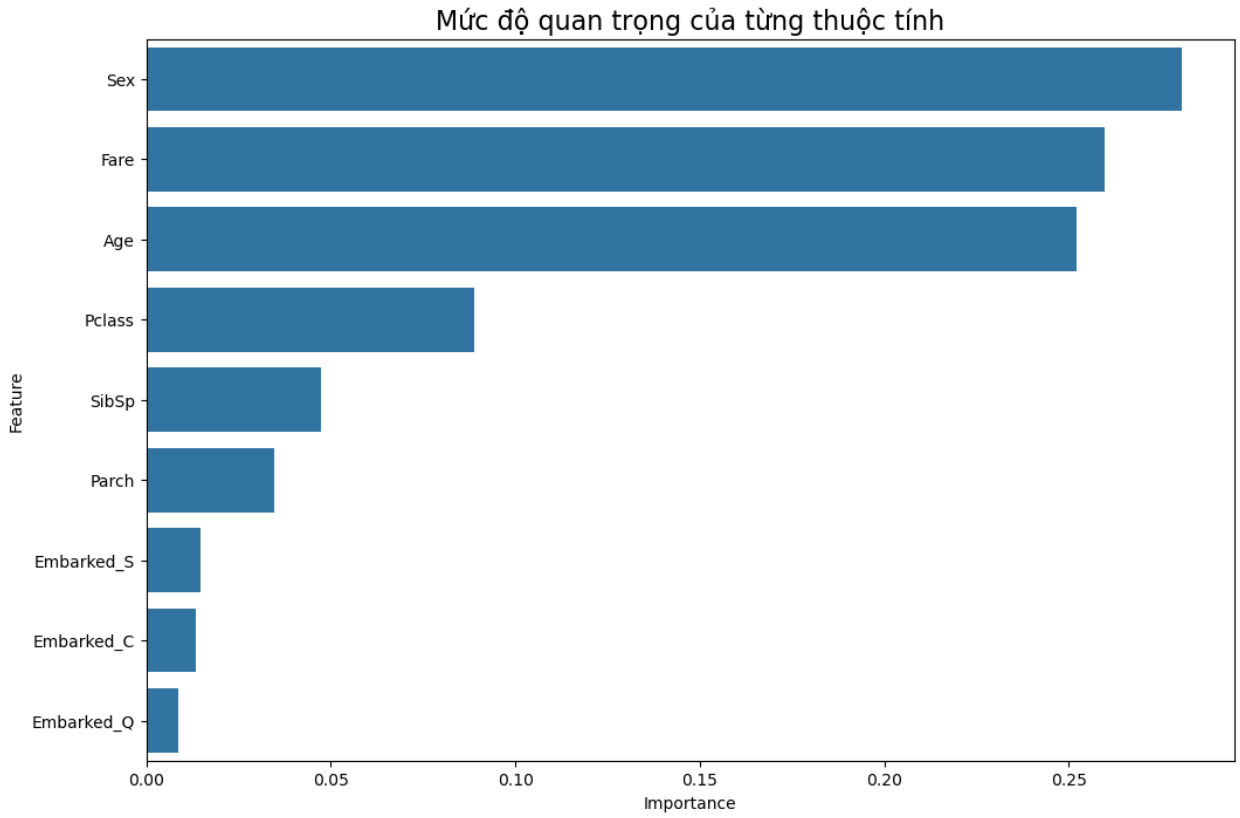
	Age	Fare
0	-0.592481	-0.502445
1	0.638789	0.786845
2	-0.284663	-0.488854
3	0.407926	0.420730
4	0.407926	-0.486337

Đây đều là dữ liệu thực nên sẽ không xử lí các outlier

4.2.Phân chia dữ liệu (Train-Test Split)

```
--- Bảng xếp hạng độ chính xác ---
                  Model  Accuracy
1     K-Nearest Neighbors  0.821229
3           Random Forest  0.815642
4  Support Vector Machine  0.815642
0     Logistic Regression  0.810056
2           Decision Tree  0.793296
5             Naive Bayes  0.776536
```

Trực quan hóa để chọn ra các thuộc tính quan trọng



Sau khi có được biểu đồ này, chọn thử 4 thuộc tính quan trọng nhất

```
Các thuộc tính được chọn để huấn luyện lại mô hình:
   Sex      Fare       Age  Pclass
0    0 -0.502445 -0.592481       3
1    1  0.786845  0.638789       1
2    1 -0.488854 -0.284663       3
3    1  0.420730  0.407926       1
4    0 -0.486337  0.407926       3

--------------------------------------------------
Độ chính xác của mô hình MỚI (chỉ với 4 features): 0.7933
--------------------------------------------------
```

Sau khi điều chỉnh số lượng thuộc tính quan trọng dộ chính xác đều giảm so với ban đầu

Thử với K neighbors

```
Các thuộc tính được chọn để huấn luyện lại mô hình:
   Sex      Fare       Age  Pclass
0    0 -0.502445 -0.592481       3
1    1  0.786845  0.638789       1
2    1 -0.488854 -0.284663       3
3    1  0.420730  0.407926       1
4    0 -0.486337  0.407926       3

--------------------------------------------------
Độ chính xác của mô hình MỚI (KNN, chỉ với 4 features): 0.8156
--------------------------------------------------
```

Sau khi điều chỉnh số lượng thuộc tính quan trọng dộ chính xác đều giảm so với ban đầu

Chọn mô hình KNN, các thông tin:

	precision	recall	f1-score	support
Không sống sót (0)	0.83	0.87	0.85	105
Sống sót (1)	0.80	0.76	0.78	74
accuracy			0.82	179
macro avg	0.82	0.81	0.81	179
weighted avg	0.82	0.82	0.82	179

Trong quá trình thử nghiệm, nhóm đã xây dựng mô hình Random Forest để dự đoán khả năng sống sót của hành khách trên tàu Titanic.
Kết quả đánh giá trên tập kiểm tra nội bộ cho thấy độ chính xác (Accuracy) đạt khoảng **0.80**. Khi nộp lên Kaggle, mô hình đạt được **điểm Accuracy = 0.775** trên leaderboard.
Kết quả này cho thấy Random Forest là một mô hình hiệu quả đối với bộ dữ liệu Titanic, mặc dù còn dư địa để cải thiện thông qua việc tối ưu siêu tham số (hyperparameter tuning) hoặc kết hợp với các mô hình khác.

5. Kết luận (Conclusion)

Nghiên cứu này đã thành công trong việc xây dựng một mô hình học máy mạnh mẽ để dự đoán khả năng sống sót của hành khách trên tàu Titanic. Chúng tôi đã trình bày một quy trình toàn diện từ tiền xử lý dữ liệu cẩn thận, trích xuất đặc trưng sáng tạo (như `Title` và `FamilySize`), đến việc triển khai một Ensemble Voting Classifier hiệu quả.

Mô hình Ensemble, kết hợp Random Forest, XGBoost và Support Vector Classifier, đã chứng minh khả năng vượt trội so với các mô hình đơn lẻ, đạt được độ chính xác cao nhất trên bộ dữ liệu kiểm tra. Điều này khẳng định giá trị của kỹ thuật học máy kết hợp trong việc giải quyết các bài toán phân loại nhị phân trong thế giới thực. Phân tích đặc trưng quan trọng cũng đã xác nhận các yếu tố lịch sử quan trọng ảnh hưởng đến tỷ lệ sống sót.

Tài liệu tham khảo (References)

[1] Kaggle. (n.d.). Titanic: Machine Learning from Disaster. Truy cập từ https://www.kaggle.com/competitions/titanic/overview [2] Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32. [3] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794. [4] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273-297. [5] Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 29(5), 1189-1232.

[6] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.