Report kết quả **dntai\_titanic\_demo\_250731**

**Tên: Phan Thanh Thịnh**

**Mssv: 3123410360**

**Đọc file train.csv và lấy 5 dòng đầu**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Đọc file test.csv và lấy 5 dòng đầu**

**A screenshot of a computer

AI-generated content may be incorrect.**

**train\_data.info() thông tin tổng quan của một DataFrame**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Hàm .isna() sẽ kiểm tra từng ô trong DataFrame xem có bị kh phải số kh**

**.sum() sẽ tính tổng số giá trị bị thiếu (NaN) cho từng cột**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Vì Embarked chỉ thiếu 2 dòng, nên ta dùng giá trị phổ biến nhất để thay.**

**.mode()[0] → tìm giá trị xuất hiện nhiều nhất (mode) trong cột Embarked.**

**.fillna(most\_common\_embarked) → điền giá trị thiếu (NaN) bằng giá trị phổ biến nhất đó.**

**Vì Age thiếu 177 dòng, nên điền trung bình để không mất dữ liệu.**

**.mean() → tính giá trị trung bình của cột Age.**

**.fillna(mean) → thay giá trị bị thiếu bằng tuổi trung bình.**

**.drop cột kh cần thiết để tránh nhiễu**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Trong age có 86 dòng dữ liệu là số mà bị thiếu**

**A black rectangular object with a black stripe

AI-generated content may be incorrect.**

**điền NaN bằng giá trị trung bình age và fare . và test data kh còn cột kh phải số nữa**

**.sum() lần 2 → cộng tất cả số NaN của mọi cột thành tổng số NaN toàn DataFrame**

**A screen shot of a computer

AI-generated content may be incorrect.**

**Chọn các cột có kiểu dữ liệu chuỗi**

**A computer screen shot of a computer code

AI-generated content may be incorrect.**

**Mã hóa cột sex cho train data và set data**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**sẽ cho ra boxplot nằm ngang (theo trục X)**

**A screenshot of a computer

AI-generated content may be incorrect.**

**np.log1p(x) = log(1 + x), giúp xử lý skewed data (dữ liệu lệch về một phía).**

**StandardScaler chuẩn hóa dữ liệu về mean = 0, std = 1.**

**Làm vậy để các features có cùng thang đo (scale), tránh trường hợp Fare lớn hơn nhiều lần Age → mô hình bị lệch.**

**train\_test\_split: tách dữ liệu thành 80% train, 20% validation.**

**random\_state=42: để kết quả chia luôn cố định.**

**stratify=y: đảm bảo tỷ lệ sống/chết trong train và val giống như toàn bộ dataset → tránh bị lệch dữ liệu.**

**A screenshot of a computer code

AI-generated content may be incorrect.**

dictionary models để lưu 3 mô hình:

* **Logistic Regression** với tối đa 1000 vòng lặp.
* **Random Forest** với 200 cây, chạy song song toàn bộ CPU (n\_jobs=-1).
* **XGBoost** với thiết lập eval\_metric='logloss' và tắt cảnh báo use\_label\_encoder.

 fit() → huấn luyện mô hình.

 predict() → trả về nhãn dự đoán.

 predict\_proba() → trả về xác suất (ở đây lấy cột 1 = xác suất sống sót / bị bệnh…)

A screenshot of a computer program

AI-generated content may be incorrect.

**RandomForestClassifier**: thuật toán rừng ngẫu nhiên.

n\_estimators=200: số lượng cây trong rừng (càng nhiều cây, mô hình càng ổn định).

n\_jobs=-1: tận dụng tất cả CPU để huấn luyện nhanh hơn.

random\_state=42: để đảm bảo kết quả reproducible (chạy lại nhiều lần cho cùng kết quả).

Sau đó huấn luyện mô hình

y\_val\_pred: nhãn dự đoán của mô hình

A screenshot of a computer

AI-generated content may be incorrect.

Nạp lại file test gốc của Kaggle (test.csv) để lấy đúng cột PassengerId

Dùng best\_model (RandomForest đã huấn luyện trước đó) để dự đoán nhãn **Survived (0 hoặc 1)** cho toàn bộ tập test.

Tạo một DataFrame mới gồm 2 cột đúng format Kaggle yêu cầu:

* PassengerId: lấy từ test gốc
* Survived: kết quả dự đoán

Xuất file submission.csv để upload lên Kaggle.

A screenshot of a computer

AI-generated content may be incorrect.