**Mục lục**

[1. Giới thiệu về Machine Learning 2](#_Toc197444668)

[2. Cài đặt môi trường làm việc 2](#_Toc197444669)

[3. Tải và khám phá bộ dữ liệu 2](#_Toc197444670)

[4. Phân tích dữ liệu (Data Exploration) 2](#_Toc197444671)

[5. Làm sạch và xử lý dữ liệu (Data Cleaning & Preprocessing) 2](#_Toc197444672)

[6. Chia dữ liệu train/test 3](#_Toc197444673)

[7. Lựa chọn thuật toán Machine Learning 3](#_Toc197444674)

[8. Huấn luyện mô hình (Model Training) 3](#_Toc197444675)

[9. Đánh giá mô hình với validation set (Cross-validation) 3](#_Toc197444676)

[10. So sánh nhiều thuật toán ML 4](#_Toc197444677)

[11. Đánh giá bằng nhiều chỉ số 4](#_Toc197444678)

[12. Tối ưu hyperparameter (GridSearchCV) 4](#_Toc197444679)

[13. Lưu và tải mô hình (pickle/joblib) 5](#_Toc197444680)

[14. Dự đoán với dữ liệu mới (Inference) 5](#_Toc197444681)

[15. Đánh giá mô hình với dữ liệu thực tế 5](#_Toc197444682)

[16. Triển khai mô hình: Automation & API cơ bản 5](#_Toc197444683)

[17. Tổng kết & Ứng dụng thực tế 5](#_Toc197444684)

[Kỹ năng đạt được 6](#_Toc197444685)

[Hướng phát triển 6](#_Toc197444686)

# 1. Giới thiệu về Machine Learning

* Hiểu khái niệm cơ bản về Machine Learning (ML), sự khác biệt giữa supervised, unsupervised, và các loại bài toán (classification, regression, clustering).
* Nắm được quy trình tổng quát của một dự án ML: thu thập dữ liệu, tiền xử lý, xây dựng mô hình, đánh giá, triển khai.

# 2. Cài đặt môi trường làm việc

* Cài đặt Python, Jupyter Notebook, các thư viện cần thiết: numpy, pandas, matplotlib, scikit-learn.
* Làm quen với giao diện Jupyter Notebook, cách tạo, chạy cell code và markdown.

# 3. Tải và khám phá bộ dữ liệu

* Biết cách tải dữ liệu từ file CSV hoặc internet.
* Sử dụng pandas để đọc dữ liệu (pd.read\_csv), xem thông tin tổng quan (head, shape, info, describe).
* Kiểm tra dữ liệu thiếu, kiểu dữ liệu, phân tích sơ bộ các cột.

# 4. Phân tích dữ liệu (Data Exploration)

* Thực hiện phân tích thống kê mô tả: giá trị trung bình, độ lệch chuẩn, min, max, giá trị phân vị.
* Vẽ biểu đồ phân phối (histogram), boxplot để phát hiện outlier, kiểm tra phân phối của các thuộc tính.
* Phân tích mối liên hệ giữa các thuộc tính và nhãn (label).

# 5. Làm sạch và xử lý dữ liệu (Data Cleaning & Preprocessing)

* Xử lý giá trị thiếu: loại bỏ, thay thế bằng mean/median/mode.
* Xử lý outlier: loại bỏ hoặc thay thế.
* Chuyển đổi dữ liệu: chuẩn hóa (StandardScaler, MinMaxScaler), mã hóa biến phân loại (OneHotEncoder, LabelEncoder).
* Tách thuộc tính (X) và nhãn (y).

# 6. Chia dữ liệu train/test

* Chia dữ liệu thành hai tập: train (dùng để huấn luyện mô hình) và test (dùng để kiểm tra mô hình).
* Thường dùng tỉ lệ 70/30 hoặc 80/20.
* Sử dụng train\_test\_split của sklearn để chia dữ liệu một cách ngẫu nhiên và có thể tái lập.

# 7. Lựa chọn thuật toán Machine Learning

* Hiểu biết về các thuật toán phổ biến:
* Hồi quy logistic (Logistic Regression): cho bài toán phân loại nhị phân.
* K-Nearest Neighbors (KNN): phân loại dựa trên láng giềng gần nhất.
* Decision Tree: cây quyết định cho phân loại và hồi quy.
* Support Vector Machine (SVM): phân loại với biên tối ưu.
* Naive Bayes: phân loại theo xác suất Bayes.
* Random Forest: tập hợp các cây quyết định.
* Biết ưu nhược điểm cơ bản của từng thuật toán.

# 8. Huấn luyện mô hình (Model Training)

* Sử dụng các class model của sklearn để fit dữ liệu train (model.fit(X\_train, y\_train)).
* Hiểu ý nghĩa của quá trình huấn luyện: tìm các tham số tối ưu để mô hình dự đoán tốt nhất trên tập train.

# 9. Đánh giá mô hình với validation set (Cross-validation)

* Sử dụng kỹ thuật K-Fold cross-validation để đánh giá mô hình trên nhiều phần khác nhau của tập train.
* Sử dụng cross\_val\_score, KFold để tự động hóa quá trình chia và đánh giá.
* Hiểu rằng cross-validation giúp kết quả đánh giá ổn định, khách quan hơn, giảm nguy cơ overfitting.

# 10. So sánh nhiều thuật toán ML

* Chạy thử nhiều model khác nhau với cùng quy trình cross-validation để so sánh công bằng.
* Lưu lại các chỉ số đánh giá (mean accuracy, std).
* Trực quan hóa kết quả bằng boxplot để so sánh độ ổn định và hiệu năng trung bình của từng mô hình.

# 11. Đánh giá bằng nhiều chỉ số

* Không chỉ dùng accuracy mà còn sử dụng các chỉ số khác:
* Confusion matrix: ma trận nhầm lẫn thể hiện số lượng dự đoán đúng/sai cho từng lớp.
* Precision, recall, f1-score: đo lường mức độ chính xác, độ bao phủ và cân bằng giữa hai yếu tố trên từng lớp.
* Sử dụng classification\_report, confusion\_matrix của sklearn để tổng hợp các chỉ số này.

# 12. Tối ưu hyperparameter (GridSearchCV)

* Hiểu sự khác biệt giữa tham số mô hình (parameter) và siêu tham số (hyperparameter).
* Sử dụng GridSearchCV để thử nghiệm nhiều giá trị hyperparameter, tìm ra bộ tham số tốt nhất.
* Hiểu ý nghĩa của scoring, cross-validation trong quá trình grid search.

# 13. Lưu và tải mô hình (pickle/joblib)

* Lưu model đã huấn luyện để tái sử dụng mà không cần train lại (dùng pickle hoặc joblib).
* Biết cách load lại model và sử dụng để dự đoán dữ liệu mới.
* Lưu ý phải lưu cả các bước tiền xử lý (scaler, encoder) nếu có.

# 14. Dự đoán với dữ liệu mới (Inference)

* Sử dụng model đã lưu để dự đoán dữ liệu mới chưa từng gặp.
* Đảm bảo dữ liệu mới được tiền xử lý giống như lúc train (scaling, encoding).
* Hiểu sự khác biệt giữa training, validation, test và inference.

# 15. Đánh giá mô hình với dữ liệu thực tế

* Kiểm tra model trên dữ liệu thực tế (out-of-sample), không dùng trong quá trình train/test/validation.
* So sánh kết quả với kỳ vọng, kiểm tra mô hình có bị overfitting không.
* Nếu có nhãn thực tế, sử dụng các chỉ số accuracy, confusion matrix để đánh giá.

# 16. Triển khai mô hình: Automation & API cơ bản

* Tự động hóa dự đoán với script Python (đọc file input, dự đoán, lưu kết quả).
* Biết cách xây dựng API đơn giản với Flask hoặc FastAPI để phục vụ mô hình cho các hệ thống khác.
* Hiểu quy trình triển khai mô hình vào ứng dụng thực tế (deployment).

# 17. Tổng kết & Ứng dụng thực tế

* Nắm vững quy trình chuẩn của một dự án ML: từ khám phá, xử lý, modeling, đánh giá tới triển khai.
* Có thể tự tin thực hiện các dự án ML thực tế từ A-Z.
* Sẵn sàng tích hợp mô hình vào ứng dụng thực tế, dashboard, hoặc sản phẩm phần mềm.
* Định hướng phát triển thêm: học về feature engineering, ensemble, deep learning, MLOps, triển khai trên cloud.

# Kỹ năng đạt được

* Làm chủ quy trình xây dựng và đánh giá mô hình Machine Learning với Python.
* Kỹ năng thực hành với pandas, numpy, matplotlib, scikit-learn.
* Hiểu sâu về các bước tiền xử lý, modeling, đánh giá, tối ưu và triển khai mô hình.

# Hướng phát triển

* Feature engineering (tạo đặc trưng mới)
* Ensemble learning (bagging, boosting)
* Deep learning (TensorFlow, PyTorch)
* MLOps, triển khai cloud, tự động hóa pipeline