**CÁC DỮ LIỆU CẦN CHO PHÂN TÍCH**

1. covid\_19\_clean\_complete.csv: Dữ liệu **toàn thế giới theo từng quốc gia mỗi ngày**
2. day\_wise.csv: Dữ liệu **toàn thế giới theo từng ngày**
3. full\_grouped.csv: Dữ liệu **từng quốc gia theo từng ngày kèm số ca mới**
4. usa\_county\_wise.csv: Dữ liệu **chi tiết từng quận ở Mỹ**
5. worldometer\_data.csv: Dữ liệu **tổng quan theo quốc gia** (thống kê dân số, số ca/mật độ test...)

**✅ THỨ TỰ CÁC BƯỚC PHÂN TÍCH DỰ ÁN MACHINE LEARNING**

**🔹 Bước 1: Xác định mục tiêu phân tích**

Ví dụ:

* Dự đoán số ca nhiễm/tử vong trong tương lai cho từng quốc gia (Time Series)
* Phân nhóm quốc gia theo mức độ nguy cơ (Clustering)
* Phân loại quốc gia có nguy cơ tử vong cao/thấp (Classification)
* Dự đoán số ca tử vong dựa vào các yếu tố dân số, test, ca nhiễm (Regression)

**🔹 Bước 2: Tiền xử lý & Khám phá dữ liệu (EDA)**

**🧼 Làm sạch dữ liệu:**

* Kiểm tra null, duplicated trong các file
* Chuyển định dạng Date → datetime
* Chuyển đổi kiểu dữ liệu phù hợp (số, chuỗi, ngày)

**📊 Khám phá dữ liệu:**

* Vẽ biểu đồ xu hướng số ca nhiễm, tử vong (file day\_wise.csv)
* Top 10 quốc gia theo tổng ca tử vong, tử vong/triệu dân (file worldometer\_data.csv)
* Biểu đồ tăng trưởng của Việt Nam, Mỹ, Brazil (file full\_grouped.csv)
* Phân bố ca nhiễm theo lục địa (từ worldometer\_data.csv)
* Heatmap tương quan các biến đầu vào

**🔹 Bước 3: Feature Engineering (tạo biến đặc trưng)**

Từ worldometer\_data.csv:

* Tạo biến “Tỉ lệ tử vong”: TotalDeaths / TotalCases
* “Tỉ lệ test”: TotalTests / Population

Từ full\_grouped.csv:

* Tạo thêm biến Growth Rate, Death Rate, Recovery Rate
* Dùng rolling mean để làm trơn chuỗi thời gian

**🔹 Bước 4: Áp dụng mô hình học máy (ML)**

**📘 A. Hồi quy (Regression)**

* **Dự đoán số ca tử vong** dựa vào: dân số, ca nhiễm, số lần test...
* Dữ liệu: worldometer\_data.csv
* Mô hình: Linear Regression, Ridge, Lasso

**📘 B. Chuỗi thời gian (Time Series Forecasting)**

* **Dự báo số ca nhiễm của Việt Nam 7 ngày tới**
* Dữ liệu: covid\_19\_clean\_complete.csv lọc Country == "Vietnam"
* Mô hình: Prophet, ARIMA

**📘 C. Phân cụm (Clustering)**

* **Nhóm quốc gia theo mức độ nghiêm trọng**
* Dữ liệu: worldometer\_data.csv, chuẩn hóa các cột:
  + TotalCases, Deaths/1M pop, ActiveCases, Tests/1M pop
* Mô hình: KMeans, DBSCAN, Hierarchical

**📘 D. Phân loại (Classification)**

* **Phân loại quốc gia có nguy cơ tử vong cao/thấp**
* Tạo nhãn: tử vong > 1000 → High Risk
* Mô hình: Random Forest, Logistic Regression

**🔹 Bước 5: Đánh giá mô hình**

| **Mô hình** | **Thước đo đánh giá** | **Lưu ý** |
| --- | --- | --- |
| Hồi quy | RMSE, MAE, R² | Dùng Ridge/Lasso để chống overfitting |
| Time Series | MAPE, biểu đồ dự báo | Dùng Prophet để xử lý missing |
| Phân cụm | Silhouette Score | Vẽ biểu đồ cụm để giải thích kết quả |
| Phân loại | Accuracy, F1, ROC-AUC | Cân nhắc nếu dữ liệu mất cân bằng |

**🔹 Bước 6: Trực quan hóa & báo cáo**

* Dùng matplotlib, seaborn, plotly, Prophet.plot() để minh họa
* Tạo Dashboard tổng quan hoặc xuất ra file báo cáo (.pdf, .html)

**🔹 Bước 7: Triển khai (nếu có yêu cầu)**

* Đóng gói mô hình → Flask app / Streamlit để người dùng nhập thông số và nhận dự đoán

**📄 covid\_19\_clean\_complete.csv**

**🔍 Mục tiêu sử dụng:**

* Phân tích xu hướng COVID-19 theo từng quốc gia mỗi ngày.
* Là dữ liệu gốc để dự báo số ca nhiễm/tử vong cho một quốc gia cụ thể.

**🧰 Các công việc thực hiện:**

* Chuyển đổi cột Date sang định dạng thời gian (Date type trong R).
* Lọc dữ liệu theo quốc gia (ví dụ: Việt Nam) để dùng cho **Time Series Forecasting**.
* Tính toán các biến phụ: tỷ lệ tử vong, tỷ lệ hồi phục, tỷ lệ ca đang điều trị, rolling mean.

**📘 Mô hình liên quan:**

* Prophet hoặc ARIMA/SARIMA để dự báo số ca bệnh tương lai.
* Phân tích xu hướng theo từng quốc gia (EDA)

**📄 day\_wise.csv**

**🔍 Mục tiêu sử dụng:**

* Khám phá tổng thể diễn biến dịch trên toàn thế giới qua từng ngày.

**🧰 Các công việc thực hiện:**

* Vẽ biểu đồ xu hướng toàn cầu: tổng số ca nhiễm, tử vong, hồi phục.
* Tính các chỉ số tăng trưởng theo thời gian.
* Kiểm tra tính mùa vụ hoặc các dịp đặc biệt (ví dụ: lễ tết) trong dữ liệu.

**📘 Mô hình liên quan:**

* **Time Series Analysis** (dữ liệu tổng hợp toàn cầu)
* Dùng Prophet với các dịp lễ/tết để dự đoán ảnh hưởng đến dịch bệnh

**📄 full\_grouped.csv**

**🔍 Mục tiêu sử dụng:**

* Dữ liệu tổng hợp theo quốc gia mỗi ngày, có thêm cột “New cases”, “New deaths”...

**🧰 Các công việc thực hiện:**

* Phân tích tăng trưởng của từng quốc gia (top quốc gia tăng nhanh nhất).
* So sánh diễn biến dịch tại các quốc gia cụ thể: Việt Nam, Mỹ, Brazil,...
* Tính growth rate, recovery rate, death rate cho từng quốc gia.

**📘 Mô hình liên quan:**

* Time Series Forecasting
* Feature Engineering cho các mô hình hồi quy hoặc phân loại

**📄 worldometer\_data.csv**

**🔍 Mục tiêu sử dụng:**

* Dữ liệu tổng hợp theo quốc gia tại một thời điểm cụ thể: dân số, mật độ test, tổng ca...

**🧰 Các công việc thực hiện:**

* Tạo các đặc trưng (feature) như:
  + Tỷ lệ tử vong: TotalDeaths / TotalCases
  + Tỷ lệ test: TotalTests / Population
  + Ca nhiễm trên 1 triệu dân: TotCases/1M pop
* Chuẩn hóa dữ liệu để áp dụng phân cụm (Clustering).
* Phân loại quốc gia có nguy cơ tử vong cao/thấp (Classification).

**📘 Mô hình liên quan:**

* **Clustering**: KMeans, DBSCAN
* **Classification**: Random Forest, Logistic Regression
* **Regression**: Dự đoán số ca tử vong theo dân số và ca nhiễm

**📄 usa\_county\_wise.csv**

**🔍 Mục tiêu sử dụng:**

* Dữ liệu chi tiết theo quận của Mỹ (địa phương).

**🧰 Các công việc thực hiện:**

* Phân tích dịch ở cấp độ nhỏ hơn (sub-national level).
* Có thể dùng để:
  + Dự báo ca bệnh cho từng quận (Time Series per county)
  + Xác định quận có nguy cơ cao bằng phân loại (Classification)

**📘 Mô hình liên quan:**

* **Time Series Forecasting** tại cấp độ quận
* **Classification** / **Regression** ở cấp độ địa phương

**Trước khi xử lí cần phải lọc sạch sẽ trước các giá trị (Na) và những giá trị khác nữa.**

**✅ 1. Tất cả các file đều nên được "tiền xử lý" (cleaning), không chỉ mỗi covid\_19\_clean\_complete.csv**

Trong Machine Learning, **dữ liệu đầu vào quyết định chất lượng mô hình**, nên bước **làm sạch dữ liệu (data cleaning)** là bắt buộc với mọi file bạn sử dụng, dù chỉ là để trực quan hóa hay để huấn luyện mô hình.

**🔍 Cụ thể cho từng file:**

**📄 covid\_19\_clean\_complete.csv**

**🔧 Bắt buộc phải xử lý sạch:**

* Chuyển Date sang định dạng Date
* Kiểm tra và xử lý giá trị null, NaN (đặc biệt là các cột như Confirmed, Deaths, Recovered)
* Gộp dữ liệu nếu cùng một quốc gia có nhiều dòng (do phân chia theo tỉnh/thành)
* Có thể cần group\_by(Country/Region, Date) rồi summarise()
* **Chuẩn hóa tên cột**: chuyển về snake\_case để xử lý dễ hơn (Country/Region → country\_region)
* **Loại bỏ cột không cần thiết** nếu không dùng trong mô hình (ví dụ: Province/State, Latitude, Longitude nếu đã tổng hợp theo quốc gia)
* **Chuyển đổi kiểu dữ liệu** đúng (nhiều khi các cột số đọc vào thành chuỗi nếu có ký hiệu lạ)
* **Sắp xếp dữ liệu** theo country và date để đảm bảo chuỗi thời gian đúng

**📄 day\_wise.csv**

**🧹 Cần kiểm tra:**

* Kiểm tra thiếu giá trị (NA) ở các cột số liệu
* Chuyển cột Date sang định dạng Date
* Kiểm tra tính liên tục của thời gian (không bị thiếu ngày)
* Đảm bảo các số liệu cộng dồn không bị giảm ngược (giá trị âm)

**Bổ sung:**

* **Tính các biến mới (feature engineering)** như:
  + Tỷ lệ tăng trưởng hàng ngày (growth\_rate = Confirmed.diff() / Confirmed.shift(1))
  + Tỷ lệ tử vong (Deaths / Confirmed)
* **Phát hiện outliers**: nếu số ca tăng đột biến bất thường → kiểm tra lại (ví dụ: tăng gấp 100 lần trong 1 ngày)

**📄 full\_grouped.csv**

**🧼 Bắt buộc xử lý:**

* Tương tự covid\_19\_clean\_complete.csv
* Đặc biệt lưu ý các cột như New cases, New deaths → có thể có giá trị âm do lỗi ghi nhận
* Làm trơn chuỗi số liệu bằng rolling mean nếu làm Time Series
* **Tính rolling mean/rolling std** để làm trơn chuỗi (ví dụ: rollmean(New\_cases, k=3))
* **Điền giá trị 0 nếu thiếu New Cases hoặc New Deaths**: nhiều quốc gia ghi nhận trễ nhưng không bị mất dữ liệu
* **Kiểm tra consistency** giữa các cột: Confirmed = sum(New\_cases) trong khoảng thời gian

**📄 worldometer\_data.csv**

**📊 Cần xử lý:**

* Các giá trị thiếu như TotalTests, Population → nên xử lý cẩn thận (thay thế, bỏ dòng, hoặc điền trung bình)
* Chuyển các số có dấu phẩy/thập phân thành số ("1,234" → 1234)
* Kiểm tra các cột chứa tỷ lệ phần trăm ("12.5%") và chuyển về số (0.125)
* **Chuẩn hóa tên quốc gia** để phù hợp với các file khác (United States vs US)
* **Xử lý định dạng số kiểu châu Âu hoặc có ký tự lạ**: "1.234,56" → 1234.56
* **Tạo biến phân loại (label)** nếu làm classification:
* High Risk nếu TotalDeaths > 1000
* **Scale/normalize** dữ liệu nếu dùng cho Clustering hoặc Regression (chuẩn hóa theo z-score, min-max...)

**📄 usa\_county\_wise.csv**

**📍 Nên làm sạch:**

* Tên quận có thể trùng → kiểm tra Province\_State + Admin2 để phân biệt
* Chuyển định dạng số, bỏ các ký hiệu , hoặc %
* Kiểm tra null và trùng lặp dữ liệu

 **Tạo mã định danh duy nhất** cho từng quận: ghép Admin2 + Province\_State

 **Xử lý dữ liệu thiếu theo nhóm**: nếu 1 quận thiếu dân số, thử lấy trung bình theo bang

 **Tạo cột Region phân nhóm bang theo miền (Đông, Tây, Nam, Trung Tây...) nếu cần**

**SƠ ĐỒ CÂY (PHÂN TÍCH ĐẠI DỊCH CO-VID)**

Quy trình phân tích

│

├── 1. Chuẩn bị môi trường

│ ├── Script (.R) [1 file: finalization\_deployment.R]

│ │ ├── Tải thư viện: tidyverse, lubridate, prophet, caret, cluster

│ └── Markdown (.Rmd) [1 file: data\_exploration.Rmd]

│ ├── Mô tả: Phân tích dịch COVID-19 toàn cầu

│ └── Dữ liệu: covid\_19\_clean\_complete.csv, day\_wise.csv, full\_grouped.csv, worldometer\_data.csv, usa\_county\_wise.csv

│

├── 2. Bước 1: Xác định mục tiêu

│ └── Markdown (.Rmd) [1 file: data\_exploration.Rmd]

│ ├── Time Series: Dự đoán ca nhiễm Việt Nam (covid\_19\_clean\_complete.csv)

│ ├── Clustering: Phân nhóm quốc gia (worldometer\_data.csv)

│ ├── Classification: Phân loại nguy cơ tử vong (worldometer\_data.csv)

│ ├── Regression: Dự đoán ca tử vong (worldometer\_data.csv)

│ └── Mục tiêu chính: Time Series cho Việt Nam

│

├── 3. Bước 2: Tiền xử lý & EDA

│ ├── Script (.R) [1 file: data\_preprocessing.R]

│ │ ├── Đọc dữ liệu: Tất cả file CSV

│ │ ├── Làm sạch: Kiểm tra NA, đổi kiểu Date, chuẩn hóa dữ liệu

│ │ └── Lưu: cleaned\_day\_wise.csv, cleaned\_worldometer.csv

│ └── Markdown (.Rmd) [1 file: data\_exploration.Rmd]

│ ├── Mô tả: Hiểu dữ liệu, phát hiện xu hướng

│ ├── Trực quan: Xu hướng toàn cầu, Top 10 tử vong, Tăng trưởng VN/Mỹ/Brazil, Phân bố lục địa, Heatmap tương quan

│ └── Nhận xét: Xu hướng tăng, Mỹ-Brazil dẫn đầu tử vong

│

├── 4. Bước 3: Feature Engineering

│ ├── Script (.R) [1 file: data\_preprocessing.R]

│ │ ├── worldometer\_data.csv: Death\_Rate, Test\_Rate

│ │ ├── full\_grouped.csv: Growth\_Rate, Death\_Rate, Recovery\_Rate, Rolling\_Mean

│ │ └── Lưu: worldometer\_with\_features.csv, full\_grouped\_with\_features.csv

│ └── Markdown (.Rmd) [1 file: data\_exploration.Rmd]

│ ├── Mô tả: Tạo biến mới

│ └── Kiểm tra: Xem biến mới

│

├── 5. Bước 4: Áp dụng mô hình ML

│ ├── Script (.R) [1 file: model\_training.R]

│ │ ├── Regression, Time Series, Clustering, Classification

│ │ └── Lưu: regression\_model.rds, prophet\_model.rds, kmeans\_model.rds, rf\_model.rds

│ └── Markdown (.Rmd) [1 file: model\_results.Rmd]

│ ├── Mô tả: Các mô hình

│ └── Kết quả: Hiển thị dự đoán, phân cụm

│

├── 6. Bước 5: Đánh giá mô hình

│ ├── Script (.R) [1 file: model\_training.R]

│ │ ├── Regression: RMSE, MAE, R²

│ │ ├── Time Series: MAPE

│ │ ├── Clustering: Silhouette Score

│ │ ├── Classification: Accuracy, F1

│ │ └── Lưu: Metrics vào CSV

│ └── Markdown (.Rmd) [1 file: model\_results.Rmd]

│ ├── Mô tả: Thước đo đánh giá

│ └── Kết quả: Hiển thị metrics, biểu đồ dự báo

│

├── 7. Bước 6: Trực quan hóa & Báo cáo

│ ├── Script (.R) [1 file: finalization\_deployment.R]

│ │ ├── Lưu biểu đồ: trend\_plot.png, top\_deaths\_plot.png, prophet\_forecast\_plot.png

│ │ └── Lưu dữ liệu: final\_day\_wise.csv, final\_worldometer.csv

│ └── Markdown (.Rmd) [1 file: final\_report.Rmd]

│ ├── Tổng hợp biểu đồ: Xu hướng, Top 10, Tăng trưởng, Phân bố, Heatmap, Dự báo

│ ├── Báo cáo: Tóm tắt kết quả

│ └── Xuất: HTML/PDF hoặc dashboard Shiny

│

└── 8. Bước 7: Triển khai

├── Script (.R) [1 file: finalization\_deployment.R]

│ ├── Đóng gói: final\_prophet\_model.rds, final\_rf\_model.rds

│ └── Ứng dụng: Shiny app

└── Markdown (.Rmd) [1 file: final\_report.Rmd]

├── Mô tả: Dự đoán ca nhiễm qua app

└── Kết luận: Tóm tắt dự án, đề xuất cải tiến