VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING


## Machine Learning (CO1007)

---

## Assignment

# *"Machine Learning trong dự báo thời tiết"*

---

**Instructor(s)**: Võ Thanh Hùng

**Students**: Lưu Văn Huy - 2211199

HO CHI MINH CITY, April 2025

# Contents

# Member list & Workload

| No. | Fullname | Student ID | Problems | % done |
|---|---|---|---|---|
| 1 | Lưu Văn Huy | 2211199 | All | 100% |

Table 1: Member list & workload

# 1 Tìm hiểu về đề tài

## 1.1 Động lực của đề tài

Hiện nay AI nói chung và Machine Learning nói riêng đã được ứng dụng vào rất nhiều lĩnh vực trong đời sống thực tế, có thể kể đến như nhận dạng các hình ảnh, hỗ trợ kinh doanh, buôn bán, áp dụng vào các công tác y tế. Trong bài tập này chúng ta sẽ tìm hiểu về ứng dụng của Machine Learning trong việc dự báo thời tiết.

Việc áp dụng machine learning vào dự báo thời tiết đang tạo ra những bước tiến lớn trong việc hiểu và phân tích các hiện tượng tự nhiên, cho phép các hệ thống học hỏi từ lượng dữ liệu thời tiết khổng lồ, bao gồm thông tin về nhiệt độ, áp suất, độ ẩm và gió, để đưa ra các dự báo chính xác hơn. Nhờ khả năng nhận diện mô hình và xử lý dữ liệu phức tạp, các thuật toán này có thể dự đoán thời tiết ở quy mô địa phương lẫn toàn cầu, từ đó hỗ trợ trong các lĩnh vực như nông nghiệp, vận tải và quản lý thiên tai. Machine learning không chỉ cải thiện độ chính xác, mà còn tăng cường khả năng dự báo trong thời gian ngắn và dài hạn, đóng vai trò quan trọng trong việc đối phó với những thách thức của biến đổi khí hậu.

## 1.2 Mô tả đề tài

### 1.2.1 Mục tiêu

Mục tiêu chính của đề tài này là nghiên cứu và ứng dụng các mô hình Machine Learning khác nhau vào việc dự đoán và phân tích hiện tượng thời tiết, cụ thể là xác định ngày mưa dựa trên các tệp dữ liệu thời tiết đã thu thập. Các dữ liệu có thể bao gồm thông tin về nhiệt độ, độ ẩm, áp suất khí quyển, lượng mưa trong quá khứ và các yếu tố khí tượng khác.

Việc sử dụng các thuật toán Machine Learning cho phép xây dựng các mô hình phân loại chính xác, từ đó dự đoán xem ngày mai có phải là một ngày mưa hay không. Đề tài này cũng hướng tới việc đánh giá hiệu suất của các mô hình, so sánh tỷ lệ dự đoán chính xác giữa các phương pháp tiếp cận khác nhau, nhằm chọn ra mô hình tối ưu nhất. Ngoài ra, nghiên cứu còn có thể mở rộng để phân tích xu hướng và biến đổi khí hậu, cung cấp thông tin hữu ích cho các lĩnh vực như nông

nghiệp, quản lý nguồn nước và phòng chống thiên tai.

### 1.2.2 Lựa chọn tập dữ liệu

Trong đề tài này, nhóm sẽ sử dụng tập dữ liệu "Rain in Australia". Bộ dữ liệu này bao gồm khoảng 10 năm quan sát thời tiết hàng ngày từ nhiều địa điểm trên khắp nước Úc.

Lý do chọn tập dữ liệu này vì đây là một tập dữ liệu trực quan và đủ lớn để ta có thể huấn luyện các mô hình Machine Learning một cách chính xác, tránh được trường hợp bị overfitting và underfitting trong quá trình huấn luyện. Tệp dữ liệu này được public tại

https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package

# 2 Hướng tiếp cận sử dụng Machine Learning

Trong đề tài này yêu cầu sử dụng ít nhất hai mô hình Machine Learning để thực hiện phân loại và đánh giá tập dữ liệu.

**Tìm hiểu về các mô hình Machine Learning**

## 2.1 Logistic regression

Logistic regression được định nghĩa là một thuật toán học máy có giám sát, thực hiện các nhiệm vụ phân loại nhị phân bằng cách dự đoán xác suất của một kết quả, sự kiện hoặc quan sát. Logistic regression phân tích mối quan hệ giữa một hoặc nhiều biến độc lập và phân loại dữ liệu thành các nhóm riêng biệt. Nó được sử dụng rộng rãi trong mô hình dự đoán, nơi mô hình ước tính xác suất toán học về việc liệu một trường hợp có thuộc một danh mục cụ thể hay không. Một ví dụ điển hình của Logistic regression là phân loại Email như email công việc, gia đình, email spam,... Xác định xem giao dịch trực tuyến có an toàn hay không an toàn, khối u lành tính hay ác tính. Thuật toán trên dùng hàm sigmoid logistic để đưa ra đánh giá theo xác suất. Ví dụ: Khối u này 80% là lành tính, giao dịch này 90% là lừa đảo...

**Nguyên lý hoạt động:**

- Hàm sigmoid: Biến đổi đầu ra thành giá trị nằm trong khoảng [0, 1] thông qua hàm sigmoid:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

  trong đó $z = b_0 + b_1 x_1 + \cdots + b_n x_n$ (tổ hợp tuyến tính của các biến đầu vào).

- Ngưỡng quyết định (Threshold): Thường chọn 0.5 để phân loại (ví dụ: 0.5 là lớp 1, <0.5 là lớp 0).

**Ưu điểm và hạn chế:**

- Ưu điểm:
  - . Dễ triển khai, hiệu quả với tập dữ liệu nhỏ.
  - . Kết quả dễ giải thích (xác suất và hệ số hồi quy phản ánh mức độ ảnh hưởng của từng biến).
- Hạn chế:
  - . Giả định mối quan hệ tuyến tính giữa biến độc lập và log-odds.
  - . Không phù hợp với bài toán phi tuyến hoặc dữ liệu phức tạp.

## 2.2 Decision Trees

Cây quyết định (Decision Trees) là một thuật toán học có giám sát (supervised learning) được sử dụng cho cả bài toán phân loại (classification) và hồi quy (regression). Tuy nhiên trong đề tài này chúng ta sẽ đề cập đến ứng dụng của Decision Trees nhiều hơn. Cây Quyết định xây dựng một mô hình dạng cây phân cấp, mô phỏng quá trình đưa ra quyết định dựa trên các câu hỏi liên tiếp về đặc trưng của dữ liệu. Mỗi nút trong cây đại diện cho một đặc trưng, mỗi nhánh thể hiện quy tắc phân chia, và mỗi lá chứa kết quả dự đoán cuối cùng.

**Nguyên lý hoạt động:**

- Cấu trúc cây:

  Nút gốc (**Root Node**): Chứa toàn bộ dữ liệu ban đầu.

  Nút trong (**Internal Nodes**): Đặt câu hỏi về giá trị của một đặc trưng

  Lá (**Leaf Nodes**): Kết luận phân loại

- Tiêu chí phân chia:

  Độ không thuần nhất Gini (**Gini Impurity**): Đo lường xác suất phân loại sai một mẫu ngẫu

nhiên.

$$Gini = 1 - \sum_{i=1}^{C} p_i^2$$

Trong đó $p_i$ là tỷ lệ mẫu thuộc lớp $i$

Độ lợi thông tin(**Information Gain**): Dựa trên Entropy, đánh giá mức độ giảm "hỗn loạn" sau khi phân chia.

- Quy trình xây dựng:

  Lặp lại việc chọn đặc trưng và ngưỡng phân chia tối ưu nhất để tạo các nút con.

  Dừng khi đạt điều kiện như độ sâu tối đa ($max\_depth$), số mẫu tối thiểu tại nút ($min\_samples\_split$), hoặc nút đã "thuần nhất" (chỉ chứa một lớp).

**Ưu điểm và hạn chế:**

- Ưu điểm:

  . Đơn giản và dễ hiểu: Cây quyết định rất dễ tiếp cận và dễ hiểu. Bạn có thể hình dung chúng như một biểu đồ dòng chảy, giúp dễ dàng theo dõi cách các quyết định được thực hiện.

  . Đa dụng: Có nghĩa là chúng có thể được sử dụng cho nhiều loại tác vụ khác nhau và hoạt động tốt cả trong phân loại lẫn hồi quy.

  . Không cần chuẩn hóa hoặc chia tỷ lệ dữ liệu: Cây quyết định không yêu cầu bạn phải chuẩn hóa hay chia tỷ lệ dữ liệu của mình.

- Hạn chế:

  . Không ổn định: Tính không ổn định có nghĩa là mô hình có thể không đáng tin cậy; những biến đổi nhỏ trong đầu vào có thể dẫn đến sự khác biệt lớn trong dự đoán.

  . Dễ bị**(overfitting)**: Xảy ra khi cây quyết định bắt được cả nhiễu và chi tiết trong dữ liệu huấn luyện, dẫn đến hiệu suất kém trên dữ liệu mới.

# 3 Các kết quả thực nghiệm

## 3.1 Logistic regression

Kết quả Hiện thực Model bằng Python:

```
# Logistic Regression
from sklearn.linear_model import LogisticRegression
params_lr = {'penalty': 'l1', 'solver':'liblinear'}
model_lr = LogisticRegression(**params_lr)
model_lr, accuracy_lr, roc_auc_lr, coh_kap_lr, tt_lr = run_model(model_lr, X_train,
    y_train, X_test, y_test)
```


Figure 1: Logistic regression result

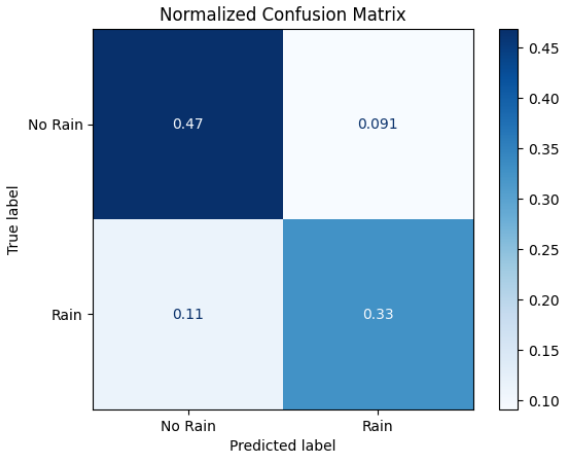


Figure 2: Illustration of confusion matrix

## 3.2 Decision Trees

Kết quả Hiện thực Model bằng Python:

```python
# Decision Tree
from sklearn.tree import DecisionTreeClassifier
params_dt = {'max_depth': 16,
             'max_features': "sqrt"}
model_dt = DecisionTreeClassifier(**params_dt)
model_dt, accuracy_dt, roc_auc_dt, coh_kap_dt, tt_dt = run_model(model_dt, X_train,
    y_train, X_test, y_test)
```

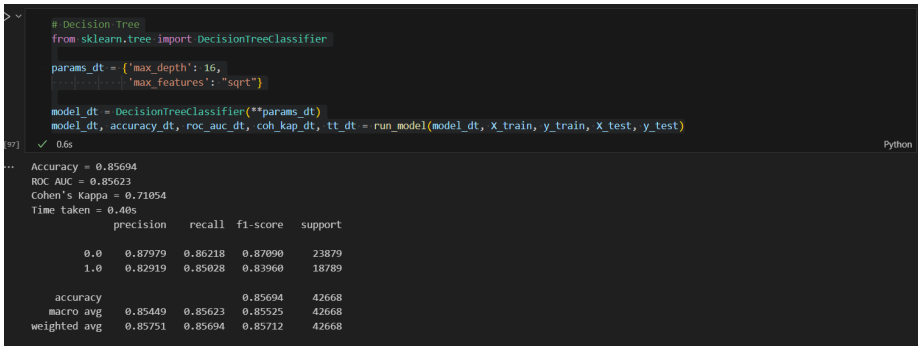


Figure 3: Decision Trees result

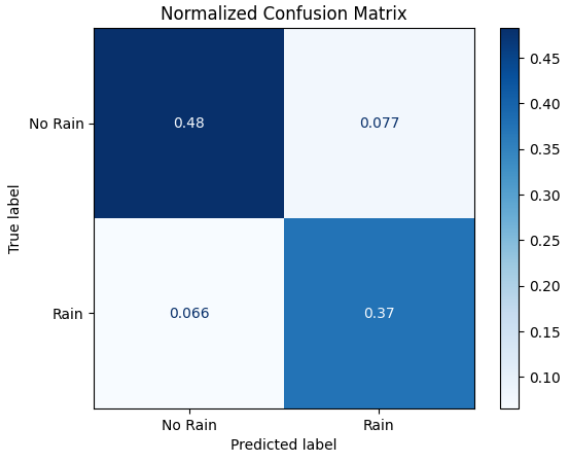


Figure 4: Illustration of confusion matrix

### 3.3 So sánh

So sánh dựa trên độ chính xác dự đoán và thời gian thực thi của mô hình:

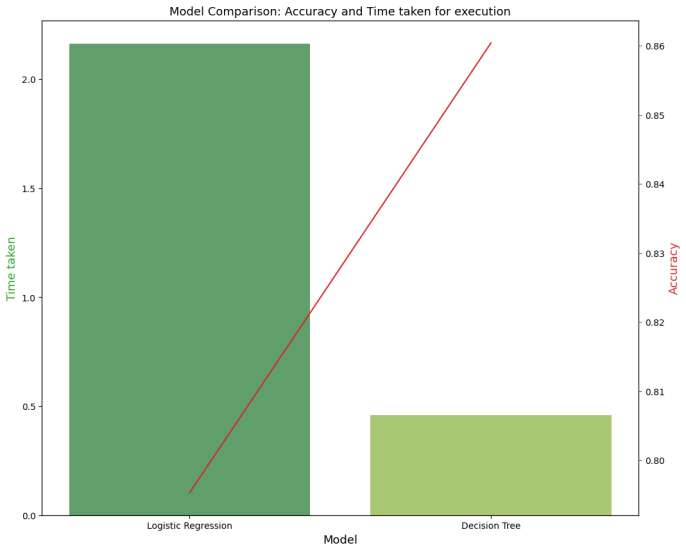


Figure 5: Model comparison

**Toàn bộ source code của dự án này được public tại đây** here

## 4 Tổng kết

Qua quá trình nghiên cứu và thực nghiệm so sánh giữa hai mô hình Machine Learning (Logistic Regression) và (Decision Trees) trong bài toán dự báo thời tiết. Ta có thể thấy rằng cả hai thuật toán đều đơn giản nhưng lại hiệu quả trong các bài toán phân loại, nhờ sự phù hợp với từng tình huống cụ thể.

Nhìn chung, việc lựa chọn giữa hai mô hình này phụ thuộc vào đặc điểm của dữ liệu và mục tiêu của bài toán. Nếu cần giải thích hoặc mở rộng thêm về bất kỳ khía cạnh nào, mình rất sẵn lòng hỗ trợ!

# 5 References

- Logistic regression Scikit-learn here
- Decision Trees Scikit-learn here
- Dataset here