 **BỘ NÔNG NGHIỆP VÀ PHÁT TRIỂN NÔNG THÔN**

**PHÂN HIỆU TRƯỜNG ĐẠI HỌC THỦY LỢI**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

-----🙞🙜🕮🙞🙜-----



***BÁO CÁO ĐỒ ÁN MÔN HỌC***

*DATA MINING – KHAI PHÁ DỮ LIỆU*

**Đề tài**

***RAIN IN AUSTRAILIA***

*(Dự Báo Mưa Ở Úc)*

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | ***ThS. Vũ Thị Hạnh*** |
| **Sinh viên thực hiện:** |  |
| ***2051067844 – Nguyễn Lý Hoàng Thành*** | |
| ***2051067524 – Võ Chí Nô*** | |
| ***2051067152 – Nguyễn Thị Thúy Diễm*** | |

***TP. Hồ Chí Minh***, ngày 19 tháng 10 năm 2023

Mục Lục

PHẦN 1: TỔNG QUAN ĐỀ TÀI 2

1.1. Lý do chọn đề tài 2

1.2. Mục tiêu đề tài 3

1.3. Phạm vi đề tài 3

1.4. Phương pháp nghiên cứu 3

1.5. Tài liệu tham khảo 3

1.5.1. Sách: 3

1.5.2. Link: 3

1.5.3. Source code: 4

PHẦN 2: CƠ SỞ LÝ THUYẾT 5

2.1. Khai phá dữ liệu 5

2.1.1. Tại sao lại cần khai phá dữ liệu 5

2.1.2. Khái niệm 5

2.1.3. Quy trình khám phá tri thức trong CSDL 5

2.1.4. Các chức năng chính của khai phá dữ liệu 7

2.1.5. Ứng dụng của khai phá dữ liệu 7

2.2. Data Preprocessing (Tiền xử lý dữ liệu) 8

2.2.1. Kiểu thuộc tính 8

2.2.2. Các kiểu giá trị thuộc tính 8

2.2.3. Khái niệm về tiền xử lý dữ liệu 8

2.2.4. Tổng quan về giai đoạn tiền xử lý dữ liệu 8

2.2.5. Làm sạch dữ liệu – Data Cleaning 10

2.3. Thuật toán 11

2.3.1. Các phương pháp đánh giá mô hình 11

2.3.3. Over fitting – Underfitting 21

2.3.4. Giới thiệu về Ensemble – Gradient Boosting (GBM) 28

2.3.5. XGBoost 33

PHẦN 3: TRIỂN KHAI ĐỀ TÀI 41

3.1. Mô tả bài toàn 41

3.1.1. Tổng quan bài toán 41

3.1.2. Tìm hiểu khái quát dữ liệu gốc 41

3.2. Khí hậu nước Úc 42

3.2.1. Tổng quan về khí hậu nước Úc 42

3.2.2. Khí hậu nước Úc theo mùa 43

3.2.3. Khí hậu nước Úc theo vùng 43

3.2.4. Khí hậu nước Úc theo từng địa điểm trong data 45

3.3. Xử lý dữ liệu 54

3.3.1. Khai phá dữ liệu 54

3.3.2. Xử lý dữ liệu 59

3.4. GradientBoosting 60

3.5. XGBoost 62

3.6. Đánh giá mô hình 63

3.6.1. Kết luận 65

PHẦN 4: TỔNG KẾT 67

4.1. Kết luận 67

4.2. Hướng phát triển 67

Lời Cảm Ơn

Đầu tiên, nhóm chúng em xin gửi lời cảm ơn chân thành đến tập thể quý Thầy Cô của Phân hiệu trường Đại học Thủy Lợi và quý Thầy, Cô khoa Công nghệ thông tin đã giúp cho nhóm chúng em có những kiến thức cơ bản làm nền tảng để thực hiện đề tài này.

Đặc biệt, nhóm chúng em xin gửi lời cảm ơn và lòng biết ơn sâu sắc nhất tới cô ***Th.S Vũ Thị Hạnh*** (Giảng viên môn Khai phá dữ liệu). Cô đã trực tiếp hướng dẫn tận tình, sửa chữa và đóng góp nhiều ý kiến quý báu giúp nhóm tác giả hoàn thành tốt báo cáo môn học của mình. Trong thời gian một học kỳ thực hiện đề tài, nhóm chúng em đã vận dụng những kiến thức nền tảng đã tích lũy đồng thời kết hợp với việc học hỏi và nghiên cứu những kiến thức mới. Từ đó, nhóm tác giả vận dụng tối đa những gì đã thu thập được để hoàn thành một báo cáo đồ án tốt nhất.

Tuy nhiên, trong quá trình thực hiện không tránh khỏi những thiếu sót. Chính vì vậy, nhóm chúng em rất mong nhận được những sự góp ý từ phía các Thầy Cô nhằm hoàn thiện những kiến thức mà nhóm chúng em đã học tập và là hành trang để nhóm chúng em thực hiện tiếp các đề tài khác trong tương lai.

Xin chân thành cảm ơn các quý Thầy Cô!

**Nhóm sinh viên thực hiện**

1. TỔNG QUAN ĐỀ TÀI
   1. Lý do chọn đề tài
2. Môi trường địa lý đa dạng: Úc có một môi trường địa lý phức tạp với sự hiện diện của núi non, sa mạc, rừng rậm và vùng biển. Sự đa dạng này tạo ra sự khác biệt lớn về mô hình mưa trên khắp lãnh thổ Úc. Do đó, việc nghiên cứu và dự báo mưa ở Úc đòi hỏi sự hiểu biết sâu về môi trường địa lý và các yếu tố tương quan với mô hình mưa.
3. Biến đổi khí hậu và hiện tượng cực đoan: Úc là một trong những quốc gia chịu ảnh hưởng mạnh mẽ của biến đổi khí hậu. Sự gia tăng của hiện tượng hạn hán và mưa lớn đã gây ra những thách thức lớn trong việc quản lý tài nguyên nước và đối phó với các sự kiện thời tiết cực đoan. Dự báo mưa chính xác và đáng tin cậy là cần thiết để chuẩn bị và ứng phó với những biến đổi khí hậu không thể tránh khỏi.
4. Quản lý tài nguyên nước và an ninh thực phẩm: Mưa đóng vai trò quan trọng trong chu kỳ nước và cung cấp nguồn nước cho cây trồng và động vật. Việc dự báo mưa chính xác giúp quản lý tốt hơn tài nguyên nước và hạn chế sự lãng phí. Ngoài ra, mưa cũng ảnh hưởng đến môi trường tự nhiên và đa dạng sinh học. Hiểu rõ về mô hình mưa và khả năng dự báo giúp chúng ta đưa ra các biện pháp bảo vệ môi trường hiệu quả và duy trì an ninh thực phẩm.
5. Tầm quan trọng kinh tế: Úc là một quốc gia nông nghiệp quan trọng với ngành nông, lâm, và thủy sản phát triển. Việc dự báo mưa chính xác giúp nông dân và người làm trong ngành nông nghiệp có thể lên kế hoạch sản xuất, quản lý tài nguyên và đảm bảo an toàn cho cây trồng và động vật. Ngoài ra, các ngành công nghiệp khác như du lịch và năng lượng cũng phụ thuộc vào dự báo mưa để đưa ra các quyết định kinh doanh hiệu quả.

Tóm lại, việc chọn đề tài dự báo mưa ở Úc là cần thiết vì môi trường địa lý đa dạng, biến đổi khí hậu, quản lý tài nguyên và tầm quan trọng kinh tế của mưa. Nghiên cứu và phát triển các phương pháp dự báo mưa ở Úc giúp hiểu rõ hơn về mô hình mưa và đưa ra các biện pháp ứng phó hiệuquả và bảo vệ tốt hơn môi trường, tài nguyên và kinh tế của quốc gia.

Do đó, nhóm tác giả đã quyết định chọn đề tài “Rain in Austraila” để làm báo cáo môn học “Khai phá dữ liệu” vì nhận thấy rằng đề tài có ý nghĩa thực tiễn và làm tiền đề để dự báo mưa cho khu vực mình đang sống.

Báo cáo gồm 4 phần:

***Phần 1***: Tổng quan đề tài

***Phần 2***: Cơ sở lý thuyết

***Phần 3***: Triển khai đề tài

***Phần 4***: Tổng kết

* 1. Mục tiêu đề tài

1. Tìm hiểu rõ nguyên lý hoạt động của thuật toán và cách sử dụng.
2. Hiểu và vận dụng những kiến thức tích lũy trong quá trình học để xử lý dữ liệu.
3. Tìm hiểu về thư viện pandas, numpy.
   1. Phạm vi đề tài

Trong bài báo cáo đồ án môn học “Khai phá dữ liệu” này, chỉ gói gọn trong phạm vi tìm hiểu và giải thích các vấn đề cơ bản về đề tài nghiên cứu.

Ngôn ngữ: Python.

Môi trường cài đặt: Google Colab.

* 1. Phương pháp nghiên cứu

Phương pháp nghiên cứu lý thuyết: tìm kiếm tài liệu trên internet, sách, báo. Đọc và dịch những tài liệu đã thu thập được, sau đó phân tích tổng hợp và viết lại cho hoàn chỉnh.

Phương pháp nghiên cứu chuyên gia: trình bày các kết quả nghiên cứu dưới dạng báo cáo và gửi về người phụ trách hướng dẫn đồ án để nhận về những bình luận, góp ý phản biện.

* 1. Tài liệu tham khảo
     1. Sách:

[1] DeepAI

* + 1. Link:

<https://vtitech.vn/xgboost-gioi-thieu-chuoi-bai-viet-ve-thuat-toan-xgboost/>

[Đắm mình vào học sâu — Đắm mình vào Học Sâu 0.14.4 documentation (aivivn.com)](https://d2l.aivivn.com/index.html?fbclid=IwAR0kWQ6TynnmAfxYt8RuQLVO5SLWLBbiVc0Ypjjctyl4nAGlNUkkJhRwKes)

[Các phương pháp tránh Overfitting - Regularization, Dropout (viblo.asia)](https://viblo.asia/p/cac-phuong-phap-tranh-overfitting-gDVK24AmlLj)

[L3-Tien\_xu\_ly\_du\_lieu.pdf (hnue.edu.vn)](http://ccs.hnue.edu.vn/hungtd/DM2012/NhatQuang/L3-Tien_xu_ly_du_lieu.pdf)

[2\_data\_mining\_tienxulydulieu\_chapter\_21\_564.pdf (tailieu.vn)](https://tailieu.vn/docview/tailieu/2015/20151201/kiepnaybinhyen_01/2_data_mining_tienxulydulieu_chapter_21_564.pdf)

[Địa lý 7: Các loại gió và hướng gió thổi đến lục địa oxtraylia | VFO.VN](https://vfo.vn/r/dia-ly-7-cac-loai-gio-va-huong-gio-thoi-den-luc-dia-oxtraylia.90594/" \l ":~:text=D%E1%BB%B1a%20v%C3%A0o%20h%C3%ACnh%2048.1%2C%2050.2%2C%2050.3%20%28trang%20152,T%C3%A2y%20%C3%B4n%20%C4%91%E1%BB%9Bi%20th%E1%BB%95i%20theo%20h%C6%B0%E1%BB%9Bng%20t%C3%A2y%20b%E1%BA%AFc.)

[Rain in Australia | Kaggle](https://www.kaggle.com/code/victorbnnt/rain-in-australia)

[Data Preprocessing - Techniques, Concepts and Steps to Master (projectpro.io)](https://www.projectpro.io/article/data-preprocessing-techniques-and-steps/512)

[(600) Gradient Boost Part 3 (of 4): Classification - YouTube](https://www.youtube.com/watch?v=jxuNLH5dXCs)

[(600) Gradient Boost Part 4 (of 4): Classification Details - YouTube](https://www.youtube.com/watch?v=StWY5QWMXCw&t=463s)

[(600) XGBoost Part 2 (of 4): Classification - YouTube](https://www.youtube.com/watch?v=8b1JEDvenQU)

[(600) XGBoost Part 3 (of 4): Mathematical Details - YouTube](https://www.youtube.com/watch?v=ZVFeW798-2I)

* + 1. Source code:

<https://colab.research.google.com/drive/1-Tce8SEMJAN-AUFSfTvWe-628TmUtizS?usp=sharing>

1. CƠ SỞ LÝ THUYẾT
   1. Khai phá dữ liệu
      1. Tại sao lại cần khai phá dữ liệu

Khoảng hơn một thập kỷ trở lại đây, lượng thông tin được lưu trữ trên các thiết bị điện tử (đĩa cứng, CD-ROM, băng từ, .v.v.) không ngừng tăng lên. Sự tích lũy dữ liệu này xảy ra với một tốc độ bùng nổ. Người ta ước đoán rằng lượng thông tin trên toàn cầu tăng gấp đôi sau khoảng hai năm và theo đó số lượng cũng như kích cỡ của các cơ sở dữ liệu (CSDL) cũng tăng lên một cách nhanh chóng. Nói một cách hình ảnh là chúng ta đang “ngập” trong dữ liệu nhưng lại “đói” tri thức. Câu hỏi đặt ra là liệu chúng ta có thể khai thác được gì từ những “núi” dữ liệu tưởng chừng như “bỏ đi” ấy không ?

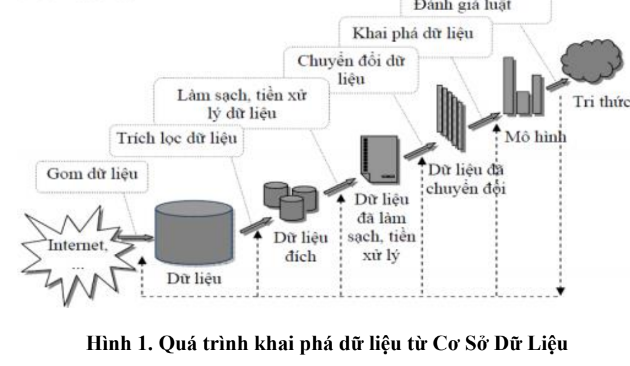
“Necessity is the mother of invention” - Data Mining ra đời như một hướng giải quyết hữu hiệu cho câu hỏi vừa đặt ra ở trên []. Khá nhiều định nghĩa về Data Mining và sẽ được đề cập ở phần sau, tuy nhiên có thể tạm hiểu rằng Data Mining như là một công nghệ tri thức giúp khai thác những thông tin hữu ích từ những kho dữ liệu được tích trữ trong suốt quá trình hoạt động của một công ty, tổ chức nào đó.

* + 1. Khái niệm

Khai phá dữ liệu (datamining) được định nghĩa như là một quá trình chắt lọc hay khai phá tri thức từ một lượng lớn dữ liệu. Một ví dụ hay được sử dụng là là việc khai thác vàng từ đá và cát, Dataming được ví như công việc "Đãi cát tìm vàng" trong một tập hợp lớn các dữ liệu cho trước. Thuật ngữ Dataming ám chỉ việc tìm kiếm một tập hợp nhỏ có giá trị từ một số lượng lớn các dữ liệu thô. Có nhiều thuật ngữ hiện được dùng cũng có nghĩa tương tự với từ Datamining như Knowledge Mining (khai phá tri thức), knowledge extraction(chắt lọc tri thức), data/patern analysis(phân tích dữ liệu/mẫu), data archaeoloogy (khảo cổ dữ liệu), datadredging(nạo vét dữ liệu),...

**Định nghĩa**: *Khai phá dữ liệu là một tập hợp các kỹ thuật được sử dụng để tự động khai thác và tìm ra các mối quan hệ lẫn nhau của dữ liệu trong một tập hợp dữ liệu khổng lồ và phức tạp, đồng thời cũng tìm ra các mẫu tiềm ẩn trong tập dữ liệu đó.*

* + 1. Quy trình khám phá tri thức trong CSDL



Khai phá dữ liệu là một bước trong bảy bước của quá trình KDD (Knowleadge Discovery in Database) và KDD được xem như 7 quá trình khác nhau theo thứ tự sau:

1. Làm sạch dữ liệu (data cleaning & preprocessing): Loại bỏ nhiễu và các dữ liệu không cần thiết.

2. Tích hợp dữ liệu: (data integration): quá trình hợp nhất dữ liệu thành những kho dữ liệu (data warehouses & data marts) sau khi đã làm sạch và tiền xử lý (data cleaning & preprocessing).

3. Trích chọn dữ liệu (data selection): trích chọn dữ liệu từ những kho dữ liệu và sau đó chuyển đổi về dạng thích hợp cho quá trình khai thác tri thức. Quá trình này bao gồm cả việc xử lý với dữ liệu nhiễu (noisy data), dữ liệu không đầy đủ (incomplete data), .v.v.

4. Chuyển đổi dữ liệu: Các dữ liệu được chuyển đổi sang các dạng phù hợp cho quá trình xử lý

5. Khai phá dữ liệu(data mining): Là một trong các bước quan trọng nhất, trong đó sử dụng những phương pháp thông minh để chắt lọc ra những mẫu dữ liệu.

6. Ước lượng mẫu (knowledge evaluation): Quá trình đánh giá các kết quả tìm được thông qua các độ đo nào đó.

7. Biểu diễn tri thức (knowledge presentation): Quá trình này sử dụng các kỹ thuật để biểu diễn và thể hiện trực quan cho người dùng.

* + 1. Các chức năng chính của khai phá dữ liệu

Data Mining được chia nhỏ thành một số hướng chính như sau:

• Mô tả khái niệm (concept description): thiên về mô tả, tổng hợp và tóm tắt khái niệm. Ví dụ: tóm tắt văn bản.

• Luật kết hợp (association rules): là dạng luật biểu diễn tri thứ ở dạng khá đơn giản. Ví dụ: “60 % nam giới vào siêu thị nếu mua bia thì có tới 80% trong số họ sẽ mua thêm thịt bò khô”. Luật kết hợp được ứng dụng nhiều trong lĩnh vực kính doanh, y học, tin-sinh, tài chính & thị trường chứng khoán, .v.v.

• Phân lớp và dự đoán (classification & prediction): xếp một đối tượng vào một trong những lớp đã biết trước. Ví dụ: phân lớp vùng địa lý theo dữ liệu thời tiết. Hướng tiếp cận này thường sử dụng một số kỹ thuật của machine learning như cây quyết định (decision tree), mạng nơ ron nhân tạo (neural network), .v.v. Người ta còn gọi phân lớp là học có giám sát (học có thầy).

• Phân cụm (clustering): xếp các đối tượng theo từng cụm (số lượng cũng như tên của cụm chưa được biết trước. Người ta còn gọi phân cụm là học không giám sát (học không thầy).

• Khai phá chuỗi (sequential/temporal patterns): tương tự như khai phá luật kết hợp nhưng có thêm tính thứ tự và tính thời gian. Hướng tiếp cận này được ứng dụng nhiều trong lĩnh vực tài chính và thị trường chứng khoán vì nó có tính dự báo cáo.

* + 1. Ứng dụng của khai phá dữ liệu

Data Mining tuy là một hướng tiếp cận mới nhưng thu hút được rất nhiều sự quan tâm của các nhà nghiên cứu và phát triển nhờ vào những ứng dụng thực tiễn của nó. Chúng ta có thể liệt kê ra đây một số ứng dụng điển hình:

• Phân tích dữ liệu và hỗ trợ ra quyết định (data analysis & decision support)

• Điều trị y học (medical treatment)

• Text mining & Web mining

• Tin-sinh (bio-informatics)

• Tài chính và thị trường chứng khoán (finance & stock market)

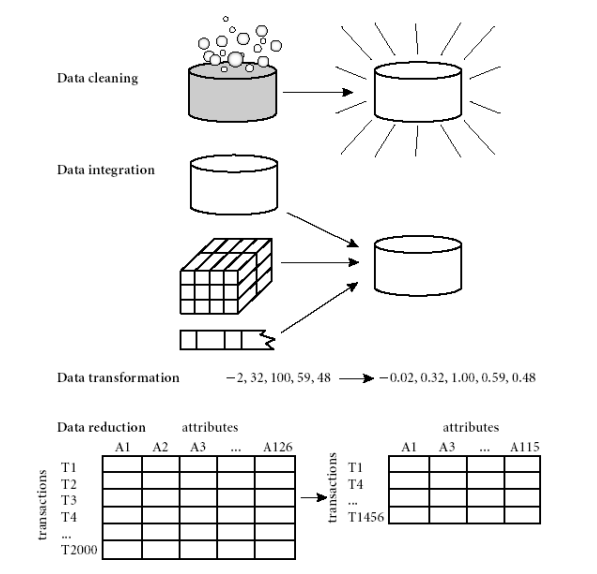
• Bảo hiểm (insurance)

• Nhận dạng (pattern recognition)

* 1. Data Preprocessing (Tiền xử lý dữ liệu)
     1. Kiểu thuộc tính
* Kiểu thuộc tính rời rạc:
* Tập các giá trị là một tập hữu hạn
* Bao gồm cả các thuộc tính có kiểu giá trị là các số nguyên
* Bao gồm cả các thuộc tính nhị phân.
* Kiểu thuộc tính liên tục:
* Các giá trị là các số thực
  + 1. Các kiểu giá trị thuộc tính
* Kiểu định danh/chuỗi (norminal): không có thứ tự
* Kiểu nhị phân (binary): là một trường hợp đặc biệt của kiểu định danh
* Kiểu có thứ tự (ordinal)
  + 1. Khái niệm về tiền xử lý dữ liệu

Tiền xử lý dữ liệu là một kỹ thuật khai thác dữ liệu bao gồm chuyển đổi dữ liệu thô thành định dạng dễ hiểu. dữ liệu trong thế giới thực thường không đầy đủ, không nhất quán hoặc thiếu một số hành vi hoặc xu hướng nhất định và có khả năng chứa nhiều lỗi. Tiền xử lý dữ liệu là một phương pháp đã được chứng minh để giải quyết các vấn đề đó.

* + 1. Tổng quan về giai đoạn tiền xử lý dữ liệu



* + - 1. Tóm tắt mô tả dữ liệu

Xác định các thuộc tính (properties) tiêu biểu của dữ liệu về xu hướng chính (central tendency) và sự phân tán (dispersion) của dữ liệu.

* Các độ đo về xu hướng chính: mean, median, mode, midraange
* Các độ đo về sự phân tán: quartiles, interquartile, range (IQR), variance

Làm nổi bật các giá trị dữ liệu được xem như nhiễu hoặc phần tử biên (outliers), cung cấp cái nhìn tổng quan về dữ liệu.

* + - * 1. Các độ đo xu hướng chính của dữ liệu

Mean: giá trị trung bình

Weighted arithmetic mean

Median: giá trị trung vị

Mode : giá trị xuất hiện thường xuyên nhất trong tập dữ liệu.

Midrange: khoảng giá trị từ giá trị lớn nhất và giá trị nhỏ nhất trong tập dữ liệu.

* + - 1. Các độ đo về sự phân tán của dữ liệu

Quartiles

* The first quartile (Q1): 25%
* The second quartile (Q2): 50% (median)
* The third quartile (Q3): 75%

Interquartile Range (IQR) = Q3 – Q1

* Outliers (the most extreme observaions): giá trị nằm cách trên Q3 hay dưới Q1 một khoảng 1.5\*IQR
  + 1. Làm sạch dữ liệu – Data Cleaning

***Xử lý các giá trị bị thiếu*** – Handing Missing Value có nhiều cách khác nhau để xử lý các giá trị bị thiếu, cụ thể là:

* Bỏ qua dữ liệu – đơn giản là chỉ cần loại bỏ các dữ liệu có thông tin bị thiếu. điều này khả thi đối với các tập dữ liệu lớn, trong đó việc bỏ qua một tỷ lệ nhỏ các bộ dữ liệu sẽ không ảnh hưởng đáng kể đến việc phân tích sâu hơn.
* Việc điền thủ công các giá trị còn thiếu có thể là một quá trình cực ký mệt mỏi nhưng có thể cần thiết, đặc biệt khi tập dữ liệu nhỏ.
* Gán một hằng số cho tất cả các giá trị thiếu
* Sự quy kết – đực thực hiện bằng cách thy thế giá trị trung bình của tất cả các mẫu hoặc bất kỳ cách logic nào khác.
* Việc thay thế giá trị còn thieeuss bằng giá trị có khả năng xảy ra cao nhất có thể đucợ thực hiện bằng cách sử dụng công thức Bayes, cây quyết định, ….

***Định dạng lại (Reformatting)***: điều này liên quan đến việc thực hiện các thay đổi định dạng dữ liệu thành định dạng chuẩn.

***Nhận dạng ngoại lệ và làm mịn dữ liệu nhiễu:*** nhiễu là một lỗi ngẫu nhiên trong một biến định lượng. một số cachs thường được sử dụng để loại bỏ nhiễu hoặc làm mịn dữ liệu như sau:

* Binning
* Hồi quy – trong phương pháp này, việc làm mịn đạt đucợ bằng cách khớp dữ liệu vào các hàm hồi quy.
* Phân tích ngoại lệ - một kỹ thuật để phát hiện các ngoại lệ là phân cụm, trong đó các giá trị nằm ngoài tập hợp các cụm bao gồm các điểm dữ liệu được coi là các ngoại lệ và sau đó bị loại bỏ
  1. Thuật toán
     1. Các phương pháp đánh giá mô hình

Đánh giá mô hình là quá trình đánh giá hiệu suất và chất lượng của một mô hình máy học (machine learning model) sau khi nó đã được huấn luyện trên dữ liệu. Mục tiêu của quá trình này là xác định khả năng của mô hình trong việc dự đoán hoặc phân loại dữ liệu mới, mà nó chưa từng thấy trong quá trình huấn luyện. Đánh giá mô hình giúp đảm bảo rằng mô hình của bạn hoạt động tốt trong thực tế và có khả năng tổng quát hóa.

* + - 1. Holdout Method:

Phương pháp "holdout" là một trong các phương pháp chia dữ liệu trong học máy và thường được sử dụng để đánh giá hiệu suất của mô hình máy học. Phương pháp này dựa trên việc chia dữ liệu thành hai phần chính: tập huấn luyện (training set) và tập kiểm tra (test set). Tập huấn luyện được sử dụng để huấn luyện mô hình, trong khi tập kiểm tra được sử dụng để đánh giá mô hình.

Thông thường, ta sẽ chia tỷ lệ dữ liệu theo phương pháp holdout là 1/3 cho test set và 2/3 cho train set, hoặc 20% cho test set và 80% cho train set.

Ưu điểm:

* Dễ triển khai và nhanh chóng: Holdout là một phương pháp đơn giản và dễ triển khai. Bạn chỉ cần chia dữ liệu thành hai phần và thực hiện huấn luyện và đánh giá.
* Phù hợp cho dữ liệu lớn: Nếu bạn có một lượng lớn dữ liệu, việc chia thành tập huấn luyện và tập kiểm tra có thể đủ để đảm bảo mô hình học cách phân loại dữ liệu một cách tốt.
* Ít tốn thời gian tính toán: So với các phương pháp cross-validation, Holdout ít tốn thời gian tính toán hơn, vì bạn chỉ cần huấn luyện và đánh giá mô hình một lần.

Nhược điểm:

* Khả năng dữ liệu không đủ lớn: Đối với các tập dữ liệu nhỏ hoặc mất cân bằng, việc chia dữ liệu thành hai phần có thể dẫn đến tập kiểm tra quá nhỏ hoặc không có đủ dữ liệu của các lớp thiểu số.
* Không đảm bảo tính tổng quát của mô hình: Một lần chia dữ liệu có thể dẫn đến việc mô hình bị ảnh hưởng bởi sự biến đổi ngẫu nhiên trong tập kiểm tra. Kết quả có thể không thể hiện sự tổng quát của mô hình trên tất cả các dữ liệu mới.
* Khả năng chọn mẫu ngẫu nhiên có thể dẫn đến sự chồng lấn: Trong trường hợp tập dữ liệu chưa được xáo trộn trước khi chia, các phần dữ liệu có thể chồng lấn nhau, làm cho tập kiểm tra không độc lập và gây ra sự thiên vị trong đánh giá mô hình.
* Không tận dụng tối ưu dữ liệu: Holdout chỉ sử dụng một phần của dữ liệu cho huấn luyện và một phần khác cho kiểm tra, không tận dụng tối ưu tất cả dữ liệu có sẵn.
  + - 1. Stratified sample

"Stratified sample" (mẫu phân tầng) là một phương pháp chọn mẫu từ một tập dữ liệu một cách sao cho mỗi lớp hoặc nhóm (class) trong dữ liệu được lấy mẫu độc lập. Mục tiêu chính của phương pháp này là đảm bảo rằng mỗi lớp hoặc nhóm đều có đại diện trong tập mẫu, và tỷ lệ của mỗi lớp trong tập mẫu gần giống với tỷ lệ ban đầu trong tập dữ liệu gốc.

Điều quan trọng nhất trong "Stratified sampling" là lấy mẫu từng lớp riêng biệt, không kết hợp dữ liệu từ các lớp lại với nhau. Điều này đảm bảo rằng mỗi lớp sẽ được đại diện và không bị thiếu trong tập mẫu. Nếu trong tập dữ liệu có mất cân bằng về số lượng mẫu giữa các lớp, "Stratified sampling" giúp giảm thiểu tình trạng lớp thiểu số bị bỏ sót trong mẫu.

Ưu điểm:

* Bảo đảm đại diện của từng lớp: Một ưu điểm quan trọng của Stratified Sampling là nó đảm bảo rằng mỗi lớp hoặc nhóm trong tập dữ liệu được đại diện trong tập mẫu. Điều này quan trọng trong trường hợp dữ liệu mất cân bằng, khi một hoặc vài lớp có ít mẫu hơn.
* Khả năng sử dụng trong phân loại và dự đoán: Stratified Sampling thường được sử dụng trong các tác vụ phân loại và dự đoán, nơi việc đảm bảo đại diện của từng lớp là quan trọng để mô hình học và đánh giá chính xác.
* Tăng độ tin cậy của kết quả: Khi bạn cần đánh giá mô hình hoặc thực hiện các thử nghiệm thống kê, Stratified Sampling giúp tạo ra kết quả đáng tin cậy hơn bằng cách đảm bảo rằng mẫu thể hiện đúng phân phối của dữ liệu gốc.

Nhược điểm:

* Cần sử dụng thêm dữ liệu: Vì Stratified Sampling đảm bảo mỗi lớp đều có mặt trong mẫu, nó thường đòi hỏi lấy mẫu nhiều hơn so với các phương pháp lấy mẫu ngẫu nhiên, đặc biệt trong trường hợp dữ liệu mất cân bằng.
* Không phù hợp cho mô hình toàn cục: Trong một số tình huống, khi bạn cần xây dựng mô hình toàn cục hoặc thực hiện phân tích dữ liệu trên toàn bộ tập dữ liệu, Stratified Sampling có thể không phù hợp.
* Tính ngẫu nhiên bị giảm: Stratified Sampling không giữ lại tính ngẫu nhiên trong việc chọn mẫu, điều này có thể làm mất đi tính khách quan trong việc nghiên cứu thống kê hoặc thực hiện kiểm định giả thuyết.
* Khó triển khai trong một số tình huống đặc biệt: Trong một số trường hợp, như trong dữ liệu dạng chuỗi thời gian, việc áp dụng Stratified Sampling có thể không phải là một nhiệm vụ đơn giản.
  + - 1. Random Subsampling

Là một phương pháp phân chia dữ liệu thường được sử dụng trong học máy để chia tập dữ liệu thành hai hoặc nhiều phần con (subsets) một cách ngẫu nhiên. Mục đích chính của phương pháp này là tạo ra các tập con dữ liệu để sử dụng cho huấn luyện và kiểm tra mô hình máy học. Nói một cách dễ hiểu, Random Sub sampling là quá trình lặp lại nhiều lần phương pháp “holdout”.

Ưu điểm:

* Dễ triển khai: Random Subsampling là phương pháp đơn giản và dễ triển khai. Không cần nhiều công đoạn phức tạp.
* Thời gian tính toán ít hơn: So với các phương pháp cross-validation, Random Subsampling tốn ít thời gian tính toán, vì bạn chỉ cần huấn luyện và đánh giá mô hình một lần.
* Phù hợp cho dữ liệu lớn: Khi bạn có một lượng lớn dữ liệu, việc chia thành tập huấn luyện và tập kiểm tra có thể đủ để đảm bảo mô hình học cách phân loại dữ liệu một cách tốt.

Nhược điểm:

* Có thể không đại diện: Do tính ngẫu nhiên của việc lựa chọn mẫu, tập kiểm tra có thể không đại diện cho tất cả trường hợp có thể xuất hiện trong thực tế. Điều này có thể dẫn đến đánh giá không chuẩn xác về hiệu suất của mô hình.
* Khả năng chồng lấn: Nếu tập dữ liệu chưa được xáo trộn hoặc xử lý trước khi chia, có khả năng tập kiểm tra và tập huấn luyện có sự chồng lấn, làm cho mô hình bị ảnh hưởng bởi sự biến đổi ngẫu nhiên trong tập kiểm tra.
* Không tận dụng tối ưu dữ liệu: Random Subsampling chỉ sử dụng một phần của dữ liệu cho huấn luyện và một phần khác cho kiểm tra, không tận dụng tối ưu tất cả dữ liệu có sẵn.
* Có thể không phù hợp cho dữ liệu mất cân bằng: Trong trường hợp dữ liệu mất cân bằng, khi một lớp có số mẫu lớn hơn đáng kể so với lớp khác, Random Subsampling có thể không đảm bảo rằng lớp thiểu số có đủ mẫu để đại diện trong tập kiểm tra.
  + - 1. K-Fold Cross-Validation

K-fold cross-validation (phân chia dữ liệu kiểm tra thành k phần) là một phương pháp đánh giá hiệu suất của mô hình máy học hoặc phân loại. K-fold cross-validation chia tập dữ liệu thành k phần (gọi là "fold"), thường với giá trị k là một số nguyên dương. Quá trình này giúp đánh giá mô hình một cách khách quan và đảm bảo khả năng tổng quát hóa của mô hình.

Cách thực hiện k-fold cross-validation:

1. Chia dữ liệu: Tập dữ liệu ban đầu được chia thành k phần bằng nhau (k fold). Các fold này không trùng lặp và không chồng lên nhau.
2. Quá trình lặp lại: Quá trình kiểm tra và đánh giá mô hình được lặp lại k lần. Trong mỗi lần lặp, một fold sẽ được dùng làm tập kiểm tra, trong khi k - 1 fold còn lại sẽ được sử dụng làm tập huấn luyện.
3. Tính toán hiệu suất: Trong mỗi lần lặp, mô hình được huấn luyện trên tập dữ liệu huấn luyện và sau đó được đánh giá trên tập dữ liệu kiểm tra. Kết quả đánh giá, chẳng hạn như độ chính xác, F1-score, hay bất kỳ thước đo nào khác, được tính toán.
4. Kết hợp kết quả: Kết quả từ k lần lặp lại được tổng hợp để đánh giá hiệu suất tổng quát của mô hình. Thường là tính trung bình của các kết quả hoặc sử dụng thước đo đặc biệt (như ROC-AUC) để tính toán một kết quả cuối cùng.

Kfold cross-validation giúp đánh giá mô hình một cách khách quan hơn và đảm bảo rằng mô hình không bị thiên vị đối với một tập kiểm tra cụ thể. Nó cũng giúp định cấu hình tối ưu cho mô hình và cung cấp một ước tính trung bình về hiệu suất của nó trên dữ liệu mới. K-fold cross-validation thường được sử dụng trong nghiên cứu và phát triển mô hình học máy.

* + - 1. Stratified cross-validation

Stratified cross-validation (cross-validation phân tầng) là một biến thể của kỹ thuật cross-validation (kiểm định chéo) được sử dụng để đánh giá hiệu suất của mô hình học máy hoặc phân loại trên dữ liệu. Mục tiêu chính của Stratified cross-validation là đảm bảo rằng tỷ lệ các lớp hoặc nhóm (classes) trong dữ liệu được duy trì cân bằng trong quá trình chia tập dữ liệu thành các tập huấn luyện và tập kiểm tra.

Cách thức thực hiện Stratified cross-validation:

1. Chia dữ liệu thành các fold: Tương tự như k-fold cross-validation, dữ liệu được chia thành một số phần (fold) dựa trên giá trị k. Mỗi fold chứa một phần của dữ liệu.
2. Phân tầng theo lớp hoặc nhóm: Quan trọng nhất, Stratified cross-validation đảm bảo rằng tỷ lệ của từng lớp hoặc nhóm trong tập dữ liệu ban đầu được duy trì trong mỗi fold. Điều này có nghĩa là mỗi fold chứa một số mẫu từ mỗi lớp, và không có sự chênh lệch quá lớn về tỷ lệ giữa các fold.
3. Lặp lại quá trình kiểm tra và đánh giá: Quá trình kiểm tra và đánh giá mô hình được lặp lại cho mỗi fold. Trong mỗi lần lặp, fold hiện tại được sử dụng làm tập kiểm tra, trong khi các fold còn lại được sử dụng làm tập huấn luyện.
4. Tính toán hiệu suất tổng quát: Kết quả từ mỗi lần lặp được tổng hợp để đánh giá hiệu suất tổng quát của mô hình trên dữ liệu.

Stratified cross-validation đặc biệt hữu ích trong trường hợp dữ liệu mất cân bằng, khi một hoặc vài lớp có số mẫu ít hơn so với các lớp khác. Bằng cách phân tầng, Stratified cross-validation đảm bảo rằng mô hình được đánh giá trên các mẫu đại diện từ mỗi lớp, giúp đánh giá tính tổng quát hóa của mô hình một cách chính xác.

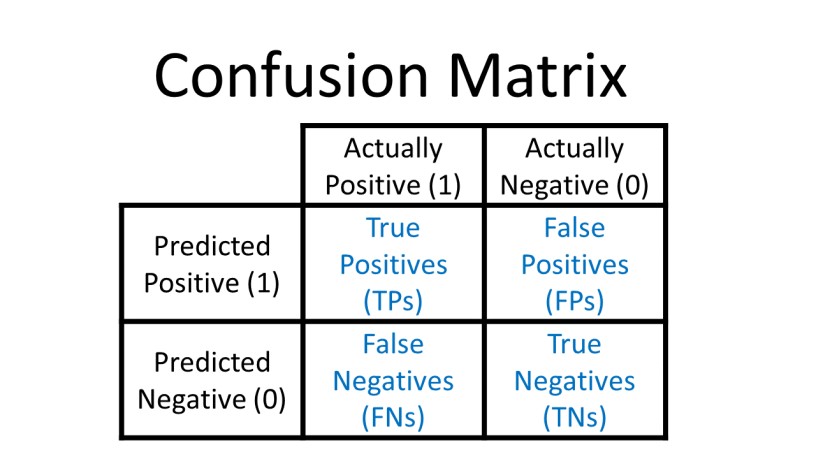
* + 1. Các thông số đánh giá mô hình:

Thông số đánh giá mô hình dùng để đo lường độ chính xác và hiệu xuất của mô hình sau khi xây dựng

* + - 1. Confusion Matrix:

Ma trận nhầm lẫn (Confusion Matrix) là một công cụ quan trọng trong đánh giá hiệu suất của mô hình phân loại trong machine learning. Nó cung cấp một cách chi tiết để biểu thị sự khác biệt giữa các dự đoán của mô hình và sự thật (actual values) của tập dữ liệu. Ma trận nhầm lẫn thường được sử dụng để tính toán các thông số đánh giá hiệu suất như Precision, Recall, và F1 Score hay Accuracy-Score.

Ma trận nhầm lẫn thường bao gồm bốn ô chính:



Positive (Tích cực): Đây là một trong các lớp hoặc kết quả mà bạn quan tâm đến hoặc muốn phát hiện. Nó thường đại diện cho một kết quả có ý nghĩa hoặc trạng thái cụ thể. Ví dụ, trong một bài toán phát hiện email rác (spam), "positive" có thể là email rác, vì bạn muốn phát hiện và lọc ra chúng.

Negative (Tiêu cực): Đây là lớp hoặc kết quả còn lại, thường là những trạng thái hoặc kết quả không có ý nghĩa hoặc là bạn không quan tâm đến. Trong ví dụ về phát hiện email rác, "negative" sẽ đại diện cho các email không phải là rác, tức là email bình thường hoặc hợp lệ.

* True Positives (TP): Số lượng trường hợp dự đoán đúng là positive và thực tế cũng là positive.
* True Negatives (TN): Số lượng trường hợp dự đoán đúng là negative và thực tế cũng là negative.
* False Positives (FP): Số lượng trường hợp dự đoán sai là positive trong khi thực tế chúng là negative.
* False Negatives (FN): Số lượng trường hợp dự đoán sai là negative trong khi thực tế chúng là positive.
  + - 1. Accuracy-Score và Error Rate:

Accuracy Score (Độ chính xác): Đây là tỷ lệ của số lượng dự đoán đúng (true predictions) trên tổng số dự đoán (total predictions).

Độ chính xác đo lường khả năng của mô hình trong việc dự đoán chính xác cả các trường hợp dương (positive) và trường hợp âm (negative). Nó thường được sử dụng khi tỷ lệ giữa các lớp dương và âm trong tập dữ liệu là cân bằng. Tuy nhiên, đối với dữ liệu mất cân bằng, độ chính xác có thể không phản ánh hiệu suất thực tế của mô hình.

Công thức của Accuracy-Score dựa vào Confusion Matrix:

Accuracy-Score =  =

Error Rate (Tỷ lệ lỗi): Đây là tỷ lệ của số lượng dự đoán sai (false predictions) trên tổng số dự đoán (total predictions).

Tỷ lệ lỗi đo lường tỷ lệ dự đoán sai của mô hình. Nó thường được sử dụng khi bạn muốn biết mức độ sai lầm của mô hình, đặc biệt là trong trường hợp dữ liệu mất cân bằng, khi mức độ sai lầm có thể quan trọng hơn.

Công thức của Error Rate:

Error Rate =

* + - 1. Precision và Recall:

Precision và Recall là hai thông số quan trọng trong đánh giá hiệu suất của mô hình phân loại, đặc biệt trong trường hợp có sự mất cân bằng giữa các lớp (class imbalance). Cả hai thông số này liên quan đến khả năng mô hình phân biệt giữa các trường hợp dương (positive) và trường hợp âm (negative) trong dự đoán.

- Precision (Độ chính xác): Precision là tỷ lệ của số lượng dự đoán dương đúng (true positives) trên tổng số lượng dự đoán dương (true positives và false positives).

Precision đo lường khả năng của mô hình trong việc đưa ra dự đoán dương mà thực sự là dương. Nó giúp xác định mức độ sai dự đoán dương và đánh giá khả năng tránh làm sai dự đoán dương khi thực tế là âm.

Công thức Precision dựa vào Confusion Matrix:

Precision =

- Recall (Tỷ lệ dự đoán dương đúng): Recall là tỷ lệ của số lượng dự đoán dương đúng (true positives) trên tổng số lượng trường hợp thực tế là dương (true positives và false negatives)

Recall đo lường khả năng của mô hình trong việc phát hiện toàn bộ các trường hợp dương thực tế. Nó giúp xác định mức độ sai dự đoán dương bị bỏ sót và đánh giá khả năng phát hiện dự đoán dương.

Công thức Recall dựa vào Confusion Matrix:

Recall =

Precision và Recall thường là hai thông số trái ngược nhau. Tăng precision có thể dẫn đến giảm recall và ngược lại. Lựa chọn giữa precision và recall phụ thuộc vào mục tiêu cụ thể của bạn trong việc đánh giá mô hình. Nếu bạn quan trọng việc tránh dự đoán sai dương, bạn có thể ưu tiên precision. Nếu bạn quan trọng việc không bỏ sót bất kỳ trường hợp dương nào, bạn có thể ưu tiên recall.

* + - 1. F1 - Measure:

Điểm F1 (F1 Score) là một thước đo tổng hợp của hiệu suất mô hình phân loại trong machine learning, và nó kết hợp cả precision và recall thành một thông số duy nhất. F1 Score hữu ích khi bạn muốn cân nhắc cả sai dự đoán dương (false positives) và trường hợp sai dự đoán âm (false negatives) trong mô hình của bạn.

F1 Score được tính bằng cách lấy trung bình điều hòa (harmonic mean) của precision và recall:

F1 Score =

* + - 1. Đường cong ROC:

Đường cong ROC (Receiver Operating Characteristic Curve) là biểu đồ thường được sử dụng để đánh giá hiệu suất của mô hình phân loại trong machine learning. Đường cong ROC biểu thị mối quan hệ giữa True Positive Rate (TPR) và False Positive Rate (FPR) của mô hình ở nhiều ngưỡng (thresholds) khác nhau

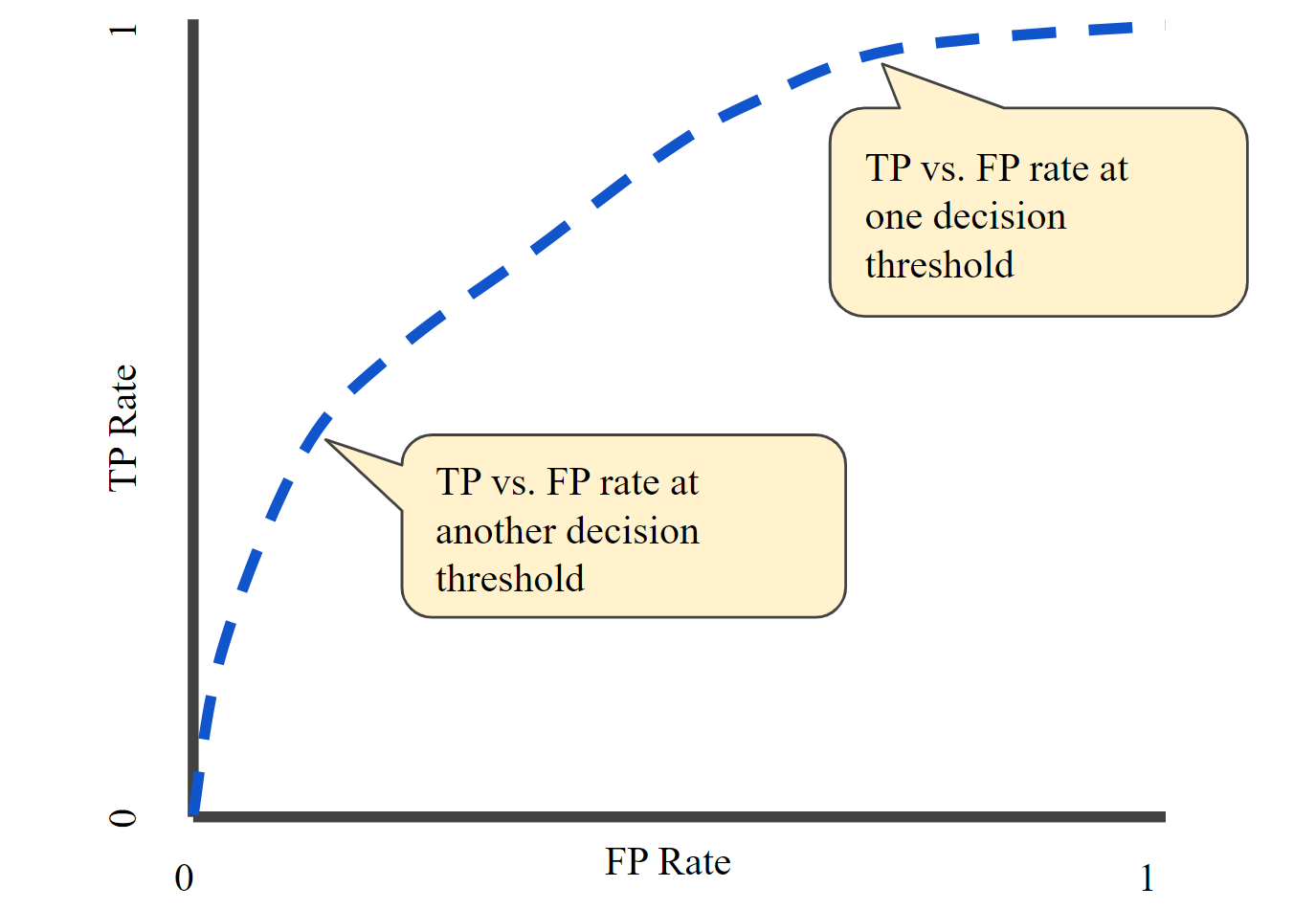
True Positive Rate (TPR), còn được gọi là Recall hoặc Sensitivity, là tỷ lệ của số lượng True Positives (dự đoán đúng là dương khi thực tế cũng là dương) trên tổng số trường hợp dương thực tế. TPR đo lường khả năng của mô hình trong việc phát hiện các trường hợp dương.

False Positive Rate (FPR) là tỷ lệ của số lượng False Positives (dự đoán sai là dương khi thực tế là âm) trên tổng số trường hợp âm thực tế. FPR đo lường khả năng của mô hình trong việc gây ra các sai dự đoán dương khi thực tế là âm.

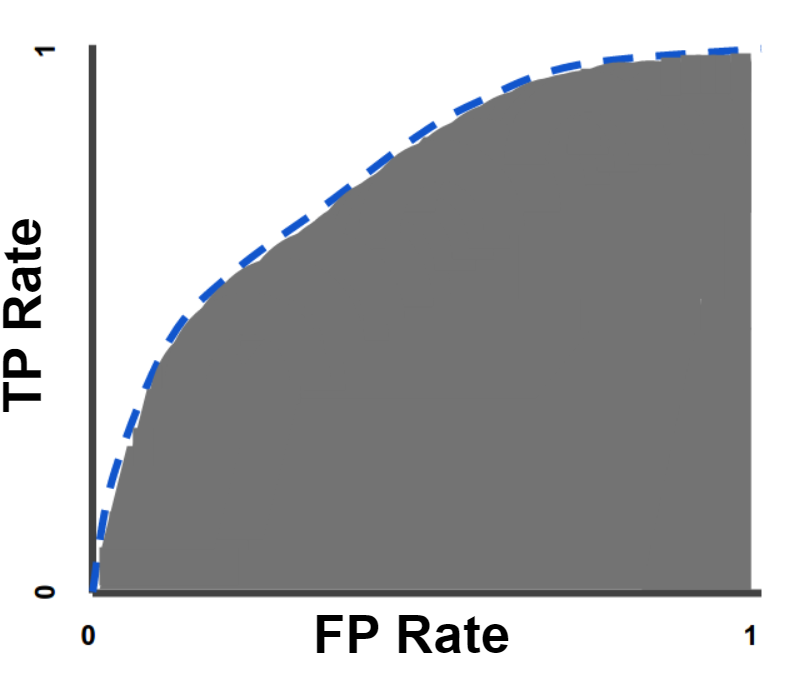
FPR = False Positives / (False Positives + True Negatives)

Đường cong ROC được tạo bằng cách biểu thị giá trị TPR trên trục tung và giá trị FPR trên trục hoành ở nhiều ngưỡng ngưỡng khác nhau. Một đường cong ROC lý tưởng là một đường chéo từ góc trái dưới đến góc phải trên, ngược với đường ngẫu nhiên.

Đường cong ROC là một công cụ quan trọng để đánh giá hiệu suất của mô hình phân loại, đặc biệt trong trường hợp có hai lớp dương và âm. Nó cho phép bạn tùy chỉnh ngưỡng để điều chỉnh cường độ độ nhạy và độ không nhạy của mô hình theo nhu cầu. Diện tích dưới đường cong ROC (AUC - Area Under the Curve) thường được sử dụng để đánh giá hiệu suất tổng thể của mô hình phân loại, với AUC càng cao thì mô hình càng tốt



*Hình ảnh: Minh họa ROC*

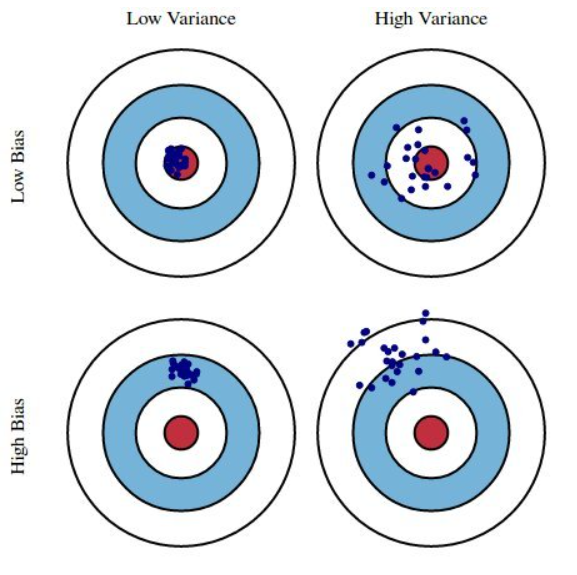


*Hình ảnh: Minh họa AUC*

* + 1. Over fitting – Underfitting
       1. Độ chênh lệch (Bias) – Phương sai (Variance)

Bias hay con gọi là độ chênh lệch chính là sự chênh lệch giữa giá trị dự báo với giá trị thật. Khi xây dựng mô hình, chúng ta mong muốn sẽ tạo ra bias thấp. Điều đó có nghĩa là giá trị dự báo sẽ gần với giá trị thật hơn.

Variance hay còn gọi là phương sai là hiện tượng giá trị dự báo của mô hình có mức độ dao động lớn nhưng thiếu tổng quát.



Độ chênh lệch thấp, phương sai thấp (Low bias, Low Variance): đây là trường hợp mô hình khớp tốt vì phân phối của giá trị dự báo trùng với phân phối của giá trị thật.

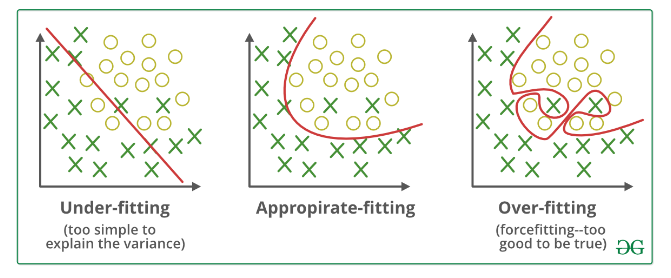
Độ chênh lệch thấp, phương sai cao (*Low Bias, High Variance*): Đây là trường hợp các giá trị dự báo sẽ giao động qua lại xung quanh ground truth. Thông thường trường hợp này sẽ xảy ra hiện tượng quá khớp (*overfitting*) mà chúng ta sẽ tìm hiểu sau.

Độ chênh lệch cao, phương sai thấp (*High Bias, Low Variance*): Đây là trường hợp mô hình dự báo bị chệch, phân phối của giá trị dự báo nằm khác xa so với phân phối của ground truth. Thông thường xảy ra khi lớp mô hình quá đơn giản. Các mô hình có đặc điểm này thường bị vị khớp (*underfitting*).

Độ chênh lệch, phương sai cao (*High Bias, High Variance*): Đặc điểm này thường thấy ở những mô hình kém khi nó vừa bị chệch và vừa dao động. Trong trường hợp này mô hình cũng bị vị khớp.

* + - 1. Khái niệm
         1. Overfitting

Khi nói đến quá khớp (overfitting) là ta nói đến trường hợp mô hình dự báo tốt trên tập huấn luyện nhưng không dự báo tốt trên tập kiểm tra. Trong trường hợp quá khớp thì mô hình chỉ học tốt trên những tập dữ liệu có cùng phân phối với tập dữ liệu huấn luyện. Đối với các trường hợp khác phân phối với tập huấn luyện mà nó chưa được học, ví dụ như tập kiểm tra khác phân phối, thì sẽ không được dự báo tốt.



Trong hình trên, hiện tượng overfitting là hình ngoài cùng bên phải, tạm gọi là hình 3. Đường biên phân chia trong hình 3 có xu hướng phân loại tốt mọi điểm dữ liệu nhưng đó là một đường biên rất phức tạp (thể hiện phương sai cao) và không khái quát về hình dạng như đường biên phân chia ở hình chính giữa.

Nói ngắn gọn, overfitting (High variance) chính là mô hình học quá mức chi tiết những thông tin không cần thiết từ tập huấn luyện, dẫn đến không thể tổng quát hóa trên tập dữ liệu mới.

* + - * 1. Underfitting

Underfitting hay còn gọi vị khớp là hiện tượng mà mô hình dự báo kém trên đồng thời cả tập huấn luyện và tập kiểm tra như hình ngoài cùng bên trái. Thông thường những mô hình quá đơn giản khi dự báo trên tập dữ liệu lớn thường dẫn tới hiện tượng vị khớp. Một mô hình vị khớp thì sẽ có độ chệch lớn nên các dự báo sẽ không thể chính xác và dẫn tới không thể áp dụng được mô hình vào thực tế.

* + - 1. Nguyên nhân dẫn đến Overfitting – Underfitting

Nguyên nhân của Overfitting và Underfitting có thể xuất phát từ mô hình quá phức tạp hoặc dữ liệu chưa đủ khái quát.

Những mô hình quá phức tạp thường có không gian biểu diễn lớn dễ dàng quá khớp được những đường biên phân chia phức tạp. Điều này tưởng chừng như là tốt cho quá trình dự báo nhưng hóa ra là không tốt vì mô hình phức tạp có xu hướng học chi tiết thay vì học quy luật tổng quát.

Hiện tượng Overfitting hay Underfitting cũng xuất phát từ dữ liệu, là hiện tượng khá phổ biến. khi dữ liệu không đủ rộng và khái quát thì mô hình không thể dự báo tốt trên tập huấn luyện.

Ngược lại, đối với những bộ dữ liệu lớn nhưng sử dụng mô hình đơn giản thì sẽ không đủ khả năng biểu diễn tốt dữ liệu.

* + - 1. Xử lý hiện tượng quá khớp (Overfitting)

Đối với hiện tượng quá khớp, có nhiều hướng xử lý khác nhau trong Machine Learning và Deep Learning. Hướng xử lý ở đây chính là phòng tránh.

* + - * 1. Phòng tránh quá khớp (Overfitting) trong mô hình Machine Learning

Cách đơn giản nhất trong phòng tránh hiện tượng quá khớp đối với mô hình Machine Learning là giảm số lượng biến và sử dụng mô hình ít phức tạp hơn.

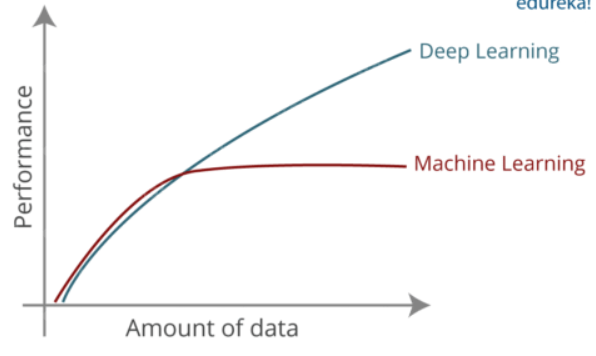
Phương pháp đề cập sau đây, chính là Regularization (phương pháp điều chuẩn).

Regularization (Phương pháp điều chuẩn)

Điều chuẩn cũng là một phương pháp năm giảm thiểu độ phức tạp của mô hình. Trong phương pháp điều chuẩn chúng ta tìm cách cộng thêm vào giá trị của hàm mất mát (loss function) một thành phần kiểm soát để làm cho mô hình có xu hướng học được một kết quả khái quát hơn trên bộ dữ liệu huấn luyện.

* + - * 1. Phòng tránh quá khớp (Overfitting) trong mô hình Deep Learning

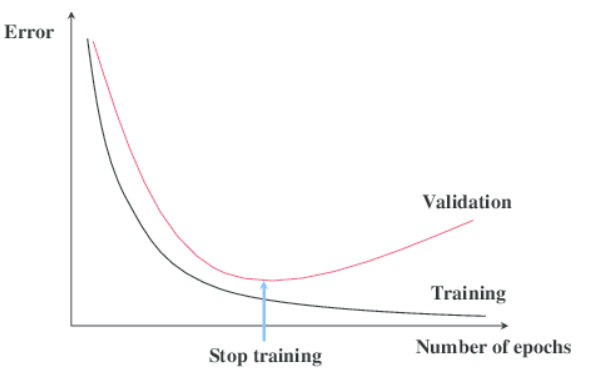
Quá khớp là hiện tượng thường gặp khi huấn luyện các mạng thần kinh nơ-ron. Một phần nguyên nhân là bởi số lượng tham số của mạng nơ-ron có thể lớn tuỳ ý nên hàm biểu diễn của mạng nơ-ron có khả năng biểu diễn lớn và có độ phức tạp cao. Thậm chí người ta còn chứng minh được rằng mạng nơ-ron có khả năng xấp xỉ mọi hàm số. Điều đó cho thấy khả năng biểu diễn của mạng nơ-ron tốt như thế nào và đây là ưu thế giúp cho hiệu suất của mô hình deep learning vượt trội hơn so với các mô hình machine learning trên những bộ dữ liệu có kích thước lớn.



Phương pháp dừng sớm (Early stopping)

Khi huấn luyện càng lâu thì giá trị hàm mất mát của mô hình trên tập huấn luyện càng nhỏ và mô hình có xu hướng khớp tốt dữ liệu trên tập huấn luyện hơn. Mặc dù sai số trên tập huấn luyện có xu hướng giảm dần theo thời gian nhưng trên tập kiểm định (validation data) điều này sẽ chưa chắc là đúng. Lý do là vì thời điểm mô hình đạt tới một độ phức tạp nhất định nó sẽ không còn khái quát hoá tốt dữ liệu kiểm định. Như vậy trên tập kiểm định tới một giai đoạn epoch nào đó sai số sẽ tăng lên.

Phương pháp dừng sớm sẽ xác định đâu là epoch được lựa chọn làm điểm dừng phù hợp căn cứ vào sai số trên tập kiểm định. Đó là thời điểm mà sai số trên tập kiểm định bắt đầu có xu hướng tăng lên. Khi đó một quyết định dừng sớm quá trình huấn luyện sẽ giúp tránh hiện tượng quá khớp.



Phương pháp Dropout

Phương pháp dropout sẽ tìm cách làm đơn giản hóa mô hình dự báo thông qua việc loại bỏ một số trọng số của mô hình thông qua mẹo thiết lập giá trị của chúng về 0 trong một vài lượt huấn luyện. Các tham số được lựa chọn để loại bỏ là ngẫu nhiên theo một tỷ lệ được xác định sẵn trên các layers được chọn, thường là những layers tại vị trí đầu tiên hoặc vị trí cuối cùng. Quá trình suy luận (inference) thì chúng ta sẽ lấy toàn bộ trọng số của mô hình mà không loại bỏ.

Mỗĩ một lượt huấn luyện chúng ta loại bỏ một tỷ lệ thiểu số các trọng số mô hình thì sẽ tạo ra một mô hình mới ít phức tạp hơn. Như vậy mô hình được huấn luyện theo kỹ thuật này sẽ là kết hợp của rất nhiều các mô hình biến thể sau dropout và chúng ta có thể xem chúng như là một phương pháp kết hợp mô hình (ensemble model) nhằm giảm quá khớp.

Trong mạng nơ ron thì dropout thường được áp dụng tại các vị trí layers đầu tiên và layers cuối cùng. Khi áp dụng chúng ta cần xác định một tỷ lệ dropout rate qui định phần trăm các trọng số sẽ bị loại bỏ khỏi từng layer. Đối với layer đầu tiên thì các low-level features còn thô (chưa tốt) nên tỷ lệ loại bỏ có thể được thiết lập cao hơn chẳng hạn từ 0.7-0.8, nhưng đối với layer cuối cùng là những high-level features tốt và cần thiết cho quá trình dự báo nên dropout rate được thiết lập thấp hơn (từ 0.1-0.5).

* + - 1. Xử lý hiện tượng vị khớp (Underfitting)
         1. Bổ sung dữ liệu cho mô hình

Bổ sung dữ liệu cho mô hình là một chiến lược lâu dài và tốn kém hơn so với việc thay đổi kiến trúc. Nhưng dường như nó lại là phương pháp mang lại hiệu quả lớn hơn so với thay đổi kiến trúc.

Sở dĩ chúng ta nói bổ sung dữ liệu tốn kém hơn so với thay đổi kiến trúc là bởi nguồn open source của các mô hình deep learning hiện tại rất dồi dào. Chúng ta có thể dễ dàng tham khảo và kế thừa lại trong các tác vụ của mình trong một thời gian ngắn. Quá trình này đơn giản và tốn ít công sức hơn nhiều so với đầu tư công phu vào làm dữ liệu.

Bên cạnh đó các mô hình deep learning nhỏ với kích thước vài triệu tham số cũng có thể đủ để biểu diễn tốt các tập dữ liệu lớn. Khi dữ liệu được cải thiện và bổ sung thì hiệu suất của những mô hình có backbone nhẹ vài triệu tham số có thể vượt xa những backbone nặng vài chục triệu hoặc thậm chí vài trăm triệu tham số nhưng chỉ được huấn luyện trên những bộ dữ liệu nhỏ. Như vậy ở thời điểm dữ liệu đang còn thiếu và ít thì tập trung vào dữ liệu sẽ mang lại hiệu suất lớn hơn so với tập trung vào cải thiện kiến trúc mô hình.

Quá trình bổ sung dữ liệu cho mô hình sẽ bao gồm thu thập và gán nhãn dữ liệu. Những dữ liệu cần thu thập nên bao quát những tình huống hiếm có (edge cases) mà mô hình cần bao phủ được để cải thiện chất lượng của chúng. Trong giai đoạn gán nhãn, để tiết kiệm chi phí gán nhãn thì chúng ta có thể sử dụng các mô hình được huấn luyện trên những backbone mạnh để có pretrained-label tốt hơn. Mô hình được dùng cho pretrained-label có thể rất lớn mà không cần quan tâm tới chi phí tính toán và khả năng triển khai trên thiết bị edge devices vì mục tiêu của chúng không phải là triển khai những mô hình này mà chỉ là tạo ra nhãn gợi ý tốt cho gán nhãn.

* + - * 1. Tăng cường dữ liệu

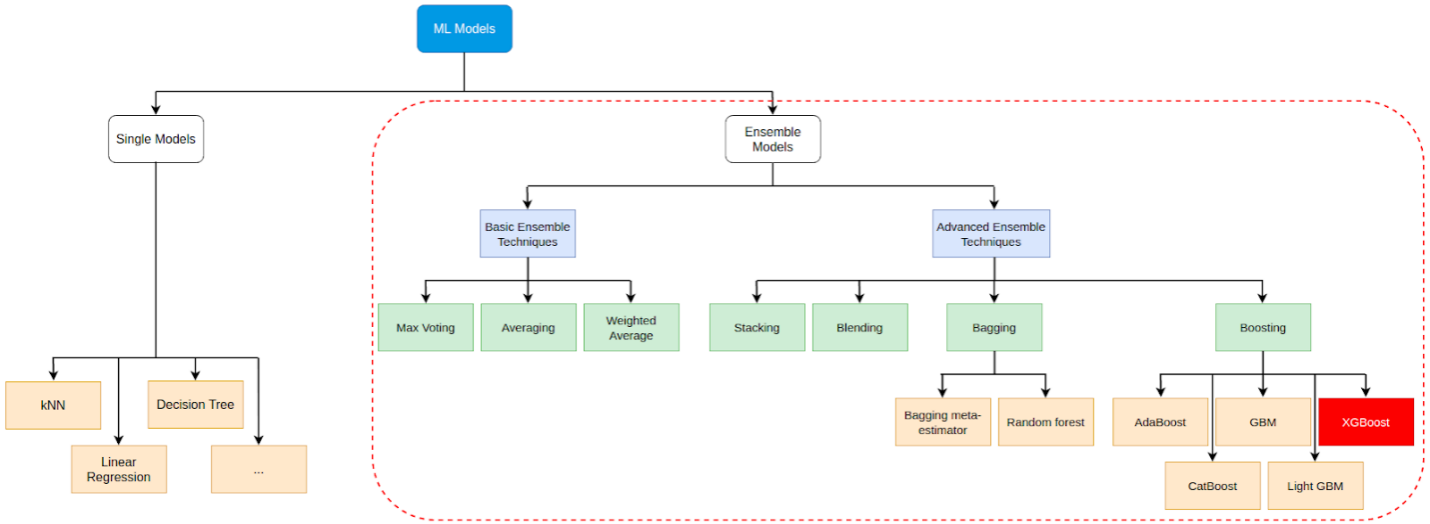
Tăng cường dữ liệu là một nguyên tắc bổ sung dữ liệu với một chi phí rẻ. Theo phương pháp này, từ một quan sát chúng ta sẽ nhân bản thành nhiều quan sát bằng cách áp dụng các kỹ thuật biến đổi mà giá trị sau biến đổi của chúng có thể mô phỏng lại một cách tương đối chính xác và đa dạng các trường hợp thực tế.

* + - * 1. Sử dụng thuật toán phức tạp hơn

Phương pháp này là một hướng cải thiện dựa trên mô hình. Đối với những bộ dữ liệu kích thước lớn mà mô hình có hiệu suất thấp thì chúng ta có thể chuyển sang những thuật toán phức tạp hơn. Trong các mô hình phân loại thuộc lớp mô hình học có giám sát của machine learning thì những mô hình được coi là phức tạp hơn thường là Random Forest, Decision Tree, MLP, SVM và ít phức tạp là Logistic, Naive Bayes, k-NN.

Đối với Deep Learning thì các kiến trúc phức tạp hơn được thể hiện qua độ sâu lớn hơn, số lượng tham số lớn hơn. Ngày nay cùng với sự phát triển mạnh mẽ của nghiên cứu, thực nghiệm và dữ liệu lớn khiến cho các kiến trúc của Deep Learning trở nên vô cùng dồi dào và đa dạng. Do đó thật khó để chúng ta nói đâu là một backbone hiệu quả nhất bởi thứ hạng chúng thường thay đổi theo thời gian. Thứ hạng chính của những backbones có thể được tìm kiếm tại leaderboard trên các tập dataset chuẩn như [ImageNet Leader Board](https://paperswithcode.com/sota/image-classification-on-imagenet).

* + 1. Giới thiệu về Ensemble – Gradient Boosting (GBM)
       1. Ensemble Learning



Ensemble Learning (học kết hợp) là một phương pháp trong lĩnh vực học máy, trong đó nhiều mô hình dự đoán yếu được kết hợp với nhau tạo ra một mô hình dự đoán mạnh mẽ hơn. Ý tưởng cơ bản của Ensemble Learning là kết hợp các dự đoán từ nhiều mô hình yếu có thể tạo ra một dự đoán chính xác hơn so với sử dụng một mô hình đơn lẻ.

Có hai phương pháp chính trong Ensemble Learning:

* Bagging (Bootstrap Aggregating): phương pháp Bagging tạo ra nhiều mô hình yếu độc lập bằng cách sử dụng các tập con dữ liệu được chọn ngẫu nhiên từ tập dữ liệu huấn luyện gốc. Mỗi mô hình được huấn luyện trên một tập dữ liệu con khác nhau và độc lập. Sau đó, kết quả của các mô hình được kết hợp bằng cách áp dụng một quy tắc voting hoặc tính trung bình để đưa ra dự đoán.
* Boosting: Phương pháp Boosting xây dựng mô hình dự đoán mạnh mẽ bằng cách lần lượt tạo ra các mô hình yếu, mỗi mô hình tập trung vào việc khắc phục những sai sót của mô hình trước đó. Trong quá trình huấn luyện, trọng số được cập nhật để tăng cường sự quan trọng của các mẫu bị dự đoán sai. Kết quả cuối cùng được tính dựa trên trọng số của các mô hình yếu và độ tin cậy của chúng.
  + - 1. Gradient Boosting Model

Phương pháp Gradient Boosting là huấn luyện liên tiếp các mô hình yếu (weak learner). Nhưng không sử dụng sai số của mô hình để tính toán trọng số cho dữ liệu huấn luyện mà sử dụng phần dư.

Xuất phát từ mô hình hiện tại, chúng ta cố gắng xây dựng một cây quyết định cố gắng khớp với phần dư từ mô hình liền trước. Điểm đặc biệt của mô hình này, đó là thay vì chúng ta cố gắng khớp giá trị biến mục tiêu thì chúng ta sẽ tìm cách khớp giá trị sai số của mô hình trước đó. Sau đó, chúng ta sẽ đưa thêm mô hình huấn luyện vào hàm dự báo để cập nhật dần dần phần dư.

Mỗi cây quyết định trong chuỗi mô hình có kích thước rất nhỏ với chỉ một vài node được xác định bởi tham số max\_depth. Như vậy, trong phương pháp Ensemble sẽ có 3 tham số hiệu chỉnh:

1. Số lượng cây.
2. Hệ số co lamda là một số dương nhỏ. Hệ số này gần giống như learning\_rate, có tác dụng kiểm soát tỷ lệ mà Ensemble cập nhật số dự.
3. Độ sâu của cây.

Trình bày ý tưởng của GBM thông qua ví dụ sau:

Cho bảng dữ liệu bên dưới:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Married** | **Gender** | **City** | **Monthly Income** | **Age (target)** |
| 1 | Y | F | Hanoi | 51.000 | 35 |
| 2 | N | M | HCM | 25.000 | 24 |
| 3 | Y | F | Hanoi | 70.000 | 38 |
| 4 | Y | M | HCM | 53.000 | 30 |
| 5 | N | M | Hanoi | 47.000 | 33 |

***Bài toán đặt ra*** là cần dự đoán tuổi dựa trên các input features như Married, Gender, City, Monthly Income.

***Bước 1***: Train decision tree model thứ nhất trên tập dữ liệu bên trên

***Bước 2***: Tính toán lỗi dựa theo sai số giữa giá trị thực tế và giá trị dự đoán

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Married** | **Gender** | **City** | **Monthly Income** | **Age (target)** | **Age (prediction 1)** | **Error 1** |
| 1 | Y | F | Hanoi | 51.000 | 35 | 32 | 3 |
| 2 | N | M | HCM | 25.000 | 24 | 32 | -8 |
| 3 | Y | F | Hanoi | 70.000 | 38 | 32 | 6 |
| 4 | Y | M | HCM | 53.000 | 30 | 32 | -2 |
| 5 | N | M | Hanoi | 47.000 | 33 | 32 | 1 |

***Bước 3***: Một decision tree model thứ 2 được tạo, sử dụng cùng input features với model trước đó, nhưng target là Error 1.

***Bước 4***: Giá trị dự đoán của model thứ 2 được cộng với giá trị dự đoán của model thứ nhất

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Age (target)** | **Age (prediction 1)** | **Error 1 (new target)** | **Prediction 2** | **Combine (Pred1+Pred2)** |
| 1 | 35 | 32 | 3 | 3 | 35 |
| 2 | 24 | 32 | -8 | -5 | 27 |
| 3 | 38 | 32 | 6 | 3 | 35 |
| 4 | 30 | 32 | -2 | -5 | 27 |
| 5 | 33 | 32 | 1 | 3 | 35 |

***Bước 5***: Giá trị kết hợp bởi bước 3 coi như là giá trị dự đoán mới. Ta tính lỗi (Error 2) dựa trên sai số giữa giá trị này và giá trị thực tế

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Age (target)** | **Age (prediction 1)** | **Error 1 (new target)** | **Prediction 2** | **Combine (Pred1+Pred2)** | **Error 2** |
| 1 | 35 | 32 | 3 | 3 | 35 | 0 |
| 2 | 24 | 32 | -8 | -5 | 27 | -3 |
| 3 | 38 | 32 | 6 | 3 | 35 | 3 |
| 4 | 30 | 32 | -2 | -5 | 27 | 3 |
| 5 | 33 | 32 | 1 | 3 | 35 | -3 |

***Bước 6***: Lặp lại bước 2-5 cho đến khi số lượng weak learner đạt được hoặc giá trị lỗi không đổi.

* + - 1. Nguyên lý hoạt động

Input: Data ; Loss function L

Step 1: Initialize model with a constant value

Step 2: for m = 1 to M:

1. Compute
2. Fit a classification tree to the values and create terminal regions for j = 1,…,Jm
3. For j = 1,…,Jm compute
4. Update
   * 1. XGBoost
        1. Định nghĩa

XGBoost (Extreme Gradient Boosting) là một thuật toán học máy thuộc lớp thuật toán Boosting. Nó được phát triển bởi Tianqi Chen vào năm 2014 và đã trở thành một trong những thuật toán học máy phổ biến và hiệu quả nhất trong cộng đồng khoa học dữ liệu và học máy.

XGBoost kết hợp giữa Gradient Boosting và Regularization để xây dựng mô hình dự đoán mạnh mẽ. Nó có khả năng học từ các mô hình yếu (weak learners) và tạo ra một mô hình tổng hợp mạnh mẽ. XGBoost sử dụng một loạt các cây quyết định như weak learners, được gọi là "Cây XGBoost" (XGBoost Trees).

* + - 1. Ưu điểm

XGBoost (*Extreme Gradient Boosting*) là phiên bản cải tiến của Gradient Boosting. Ưu điểm vượt trội của nó được chứng minh ở các khía cạnh:

* Tốc độ xử lý

XGBoost thực hiện tinh toán song song nên tốc độ xử lý có thể tăng gấp 10 lần so với GBM. Ngoài ra, XGboost còn hỗ trợ tính toán trên Hadoop.

* Overfitting

XGBoost áp dụng cơ chế Regularization nên hạn chế đáng kể hiện tượng Overfitting (GBM không có regularization).

* Sự linh hoạt

XGboost cho phép người dùng sử dụng hàm tối ưu và chỉ tiêu đánh giá của riêng họ, không hạn chế ở những hàm cung cấp sẵn.

* Xử lý missing value

XGBoost bao gồm cơ chế tự động xử lý missing value bên trong nó. Vì thế, có thể bỏ qua bước này khi chuẩn bị dữ liệu cho XGBoost.

* Tự động cắt tỉa

Tính năng tree pruning hỗ trợ việc tự động bỏ qua những leaves, nodes không mang giá trị tích cực trong quá trình mở rộng tree.

Chính vì những ưu điểm đó mà hiệu năng của XGBoost tăng lên đáng kể so với các thuật toán Ensemble Learning khác.

* + - 1. Xây dựng thuật toán

***Bước 1***: Chuẩn bị dữ liệu.

***Bước 2***: Tạo và tinh chỉnh mô hình.

***Bước 3***: Huấn luyện mô hình.

***Bước 4***: Đánh giá hiệu suất của mô hình trên dữ liệu kiểm tra bằng cách sử dụng các phép đo lỗi.

***Bước 5***: Dự đoán.

* + - * 1. Chuẩn bị dữ liệu

Chuẩn bị dữa liệu là quá trình gồm thu thập dữ liệu, tiền xử lý dữ liệu, khai phá dữ liệu có một bộ dữ liệu sạch, làm tiền đề cho bước xây dựng mô hình.

Sau đó, tiến hành chia tập dữ liệu thành các tập: tập huấn luyện (Training Set), tập kiểm tra (Validation Set) và tập đánh giá (Test Set) để đảm bảo mô hình được huấn luyện và đánh giá một cách chính xác.

Tập huấn luyện (Training Set): dùng để huấn luyện mô hình. Đây là tập dữ liệu mà mô hình sẽ học từ đó.

Tập kiểm tra (Validation Set): dùng để tinh chỉnh các hyperparameter và lựa chọn mô hình tốt nhất. Sau khi huấn luyện mô hình trên tập huấn luyện, bạn sẽ đánh giá hiệu suất của mô hình trên tập kiểm tra xem mô hình có hoạt động tốt hay không.

Tập đánh giá (Test Set): dùng để đánh giá khả năng tổng quát hóa của mô hình trên dữ liệu mới mà model chưa bao giờ nhìn thấy trong quá trình huấn luyện và tinh chỉnh hyperparameter.

* + - * 1. Tin chỉnh hyperparameters

Kỹ thuật tìm siêu tham số lưới tìm kiếm (grid search) là một phương pháp tìm kiếm và đánh giá các giá trị siêu tham số khác nhau để xác định bộ siêu tham số tốt nhất cho mô hình. Grid search hoạt động bằng cách xác định một lưới các giá trị có thể cho từng siêu tham số và sau đó kiểm tra tất cả các kết hợp có thể có của các giá trị đó.

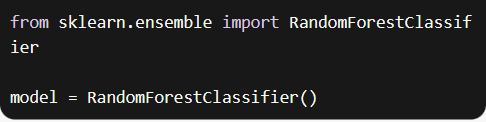
1. Xác định lưới siêu tham số: Đầu tiên, bạn cần xác định các siêu tham số mà bạn muốn tinh chỉnh và các giá trị mà muốn thử cho mỗi siêu tham số.
2. Tạo lưới tìm kiếm
3. Xây dựng và đánh giá mô hình: Với mỗi bộ siêu tham số trong lưới tìm kiếm, bạn xây dựng và huấn luyện mô hình trên tập huấn luyện. Sau đó, đánh giá hiệu suất của mô hình trên tập kiểm tra (validation set) bằng cách sử dụng các phép đo lỗi.
4. Chọn bộ siêu tham số tốt nhất.

Để thực hiện kỹ thuật tìm siêu tham số lưới tìm kiếm, có thể sử dụng thư viện máy học Scikit-Learn trong Python, có cung cấp công cụ hỗ trợ Grid Search như gridsearchcv để thực hiện quá trình kiểm tra đánh giá siêu tham số.

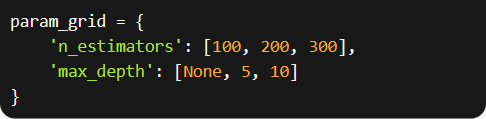
* Import thư viện cần thiết



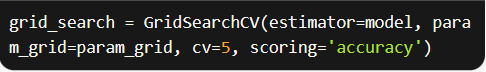
* Định nghĩa mô hình. Ví dụ: RandomForestClassifier



* Định nghĩa siêu tham số và giá trị để tìm kiếm.



* Tạo GridSearchCV: bằng cách cung cấp mô hình, siêu tham số và các thông số khác như số lần chia tập dữ liệu (cross-validation),…



Trong đó:

estimator: mô hình bạn muốn tinh chỉnh

param\_grid: Danh sách các siêu tham số và giá trị tương ứng

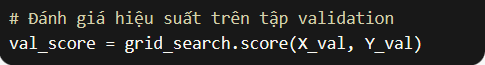
cv: số lần xác thực chéo

scoring: điểm số mà bạn muốn tối ưu

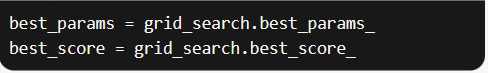
* Huấn luyện và tìm kiếm siêu tham số tốt nhất



* Đánh giá hiệu suất trên tập validation



* Chọn siêu tham số



* + - * 1. Huấn luyện mô hình và dự báo

Sau khi, tinh chỉnh hyperparameter của mô hình. Ta tạo mô hình với các hypermater đã tìm được ở trên. Đánh giá hiệu suất trên tập validation set và test set để kiểm tra high bias (underfitting) và high variance (overfitting).

* + - 1. Một số tham số quan trọng của XGBClassifier

1. n\_estimators:
   * Giải thích: Đây là số lượng cây quyết định trong mô hình.
   * Công dụng: Quy định số lượng bước lặp trong quá trình huấn luyện mô hình.
   * Cách sử dụng: Thông thường, giá trị này được thiết lập lớn để tăng độ chính xác của mô hình, nhưng cũng làm tăng thời gian huấn luyện. Giá trị mặc định là 100.
2. max\_depth:

* Giải thích: Đây là độ sâu tối đa của các cây quyết định.
* Công dụng: Quy định độ phức tạp của mô hình. Điều chỉnh giá trị này có thể giúp tránh tình trạng quá khớp (overfitting) hoặc quá đơn giản hóa mô hình (underfitting).
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 3 đến 10. Giá trị mặc định là 3.

1. learning\_rate:

* Giải thích: Đây là tốc độ học của mô hình.
* Công dụng: Quy định mức độ ảnh hưởng của mỗi cây quyết định đối với mô hình tổng thể. Giá trị nhỏ hơn sẽ yêu cầu nhiều cây quyết định hơn để đạt được kết quả tương tự và ngược lại.
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0.01 đến 0.3. Giá trị mặc định là 0.1.

1. subsample:

* Giải thích: Đây là tỷ lệ mẫu (samples) được sử dụng trong quá trình huấn luyện mỗi cây quyết định.
* Công dụng: Quy định mức độ phân chia dữ liệu huấn luyện cho các cây con. Giá trị nhỏ hơn sẽ làm giảm độ phức tạp của mô hình và ngược lại.
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0.5 đến 1. Giá trị mặc định là 1.

1. colsample\_bytree:

* Giải thích: Đây là tỷ lệ các đặc trưng (features) được sử dụng trong quá trình huấn luyện mỗi cây quyết định.
* Công dụng: Quy định mức độ phân chia đặc trưng cho các cây con. Giá trị nhỏ hơn sẽ làm giảm độ phức tạp của mô hình và ngược lại.
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0.5 đến 1. Giá trị mặc định là 1.

1. gamma:

* Giải thích: Đây là một tham số cắt tỉa (pruning) trong quá trình xây dựng cây quyết định.
* Công dụng: Quy định mức độ giảm thiểu lỗi để tiếp tục xây dựng cây. Một giá trị lớn hơn sẽ làm giảm số lượng cây quyết định và tăng tốc độ huấn luyện.
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0 đến vô cùng. Giá trị mặc định là 0.

1. reg\_lambda:

* Giải thích: Đây là hệ số điều chỉnh (regularization) L2 regularization trong quá trình huấn luyện.
* Công dụng: Quy định mức độ kiểm soát của regularization để tránh tình trạng quá khớp (overfitting).
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0 đến vô cùng. Giá trị mặc định là 1.

1. reg\_alpha:

* Giải thích: Đây là hệ số điều chỉnh (regularization) L1 regularization trong quá trình huấn luyện.
* Công dụng: Quy định mức độ kiểm soát của regularization để tránh tình trạng quá khớp (overfitting).
* Cách sử dụng: Thông thường, giá trị này được thiết lập trong khoảng từ 0 đến vô cùng. Giá trị mặc định là 0.

1. objective:

* Giải thích: Đây là hàm mục tiêu (objective function) được tối ưu trong quá trình huấn luyện.
* Công dụng: Quy định mục tiêu của bài toán hồi quy. Ví dụ: 'reg:squarederror' cho hồi quy tuyến tính, 'reg:gamma' cho hồi quy gamma, 'reg:tweedie' cho hồi quy Tweedie, v.v.
* Cách sử dụng: Chọn một hàm mục tiêu phù hợp với bài toán hồi quy của bạn. Giá trị mặc định là 'reg:squarederror'.

Dưới đây là một số giá trị phổ biến của tham số objective trong XGBRegressor:

* 'reg:squarederror': Đây là hàm mục tiêu mặc định cho bài toán hồi quy tuyến tính. Nó sử dụng độ lỗi bình phương của giá trị dự đoán và giá trị thực tế để đo lường hiệu suất mô hình.
* 'reg:linear': Đây cũng là một hàm mục tiêu cho bài toán hồi quy tuyến tính. Nó tương tự như 'reg:squarederror', nhưng không tính toán độ lỗi bình phương.
* 'reg:gamma': Đây là hàm mục tiêu cho bài toán hồi quy gamma, trong đó giá trị dự đoán được giả định là một phân phối gamma. Hàm mục tiêu này được sử dụng khi biến phụ thuộc có phân phối gamma.
* 'reg:tweedie': Đây là hàm mục tiêu cho bài toán hồi quy Tweedie, trong đó giá trị dự đoán được giả định là một phân phối Tweedie. Hàm mục tiêu này được sử dụng khi biến phụ thuộc có phân phối Tweedie.
* 'reg:poisson': Đây là hàm mục tiêu cho bài toán hồi quy Poisson, trong đó giá trị dự đoán được giả định là một phân phối Poisson. Hàm mục tiêu này được sử dụng khi biến phụ thuộc có phân phối Poisson.

1. TRIỂN KHAI ĐỀ TÀI
   1. Mô tả bài toàn
      1. Tổng quan bài toán

Lựa chọn đề tài về dự báo lượng mưa là do sự nhận thức về tầm quan trọng của khí hậu và môi trường đối với cuộc sống chúng ta. Mưa là một yếu tố quan trọng trong chu kỳ thực vật, nguồn nước và hệ sinh thái tự nhiên. Hiểu và dự báo chính xác lượng mưa có thể giúp chúng ta tối ưu hóa sử dụng tài nguyên nước, lập kế hoạch nông nghiệp, dự báo lũ lụt và xây dựng các biện pháp phòng chống thảm họa. Đề tài này cũng có thể góp phần vào việc nghiên cứu và hiểu rõ hơn về biến đổi khí hậu và ảnh hưởng của nó đến môi trường và cuộc sống của con người. Dự báo lượng mưa không chỉ giúp chúng ta chuẩn bị cho những tác động của mưa, mà còn đóng vai trò quan trọng trong việc phòng tránh và giảm thiểu thiệt hại do mưa gây ra.

Bộ dữ liệu mà nhóm chúng em sử dụng là bộ dữ liệu về lượng mưa ở Úc được quan sát tại 49 địa điểm khác nhau tại nước Úc. Bộ dữ liệu này nghiên cứu và cho ra kết luận rằng ngày mai sẽ có mưa hay không.

Link dataset:

Bộ dữ liệu gồm 145460 dòng dữ liệu và 23 thuộc tính. Thuộn tính target là RainTomorrow. Mỗi hàng đại diện cho thời tiết của ngày hôm nay tại vùng bất kỳ được dán nhãn là ngày mai mưa hay không. Có nhiều giá trị thuộc tính bị thiếu.

* + 1. Tìm hiểu khái quát dữ liệu gốc

1. Date: ngày khảo khát
2. Location: địa điểm khảo sát
3. MinTemp: Nhiệt độ thấp nhất trong 24 giờ đến 9 giờ sáng. Đơn vị là 0C
4. MaxTemp: Nhiệt độ cao nhất trong 24 giờ đến 9 giờ sáng. Đơn vị là 0C
5. Rainfall: Lượng mưa trong 24 giờ đến 9 giờ sáng. Đơn vị là mm
6. Evaporation: Sự bố hơi trong 24 giờ đến 9 giờ sáng. Đơn vị là mm
7. Sunshine: Số giờ nắng trong 24 giờ đến nửa đêm. Đơn vị là giờ
8. WindGustDir: hướng gió giật mạnh nhất trong vòng 24 giờ đến nửa đêm. Có 16 loại hướng gió
9. WindGustSpeed: Tốc độ gió giật mạnh nhất trong vòng 24 giờ đến nửa đêm. Đơn vị là km/h
10. WindDir9am: Hướng gió lúc 9 giờ sáng
11. WindDir3pm: Hướng gió lúc 3 giờ chiều
12. WindSpeed9am: Tốc độ gió lúc 9 giờ sáng. Đơn vị là km/h
13. WindSpeed3pm: Tốc độ gió lúc 3 giờ chiều. Đơn vị là km/h
14. Humidity3pm: Độ ẩm tương đối lúc 3 giờ chiều. Đơn vị là %
15. Humidity9am: Độ ẩm tương đối lúc 9 giờ sáng. Đơn vị là %
16. Pressure9am: Áp suất trung bình của khí quyển đổi với mực nước biển lúc 9 giờ sáng. Đơn vị là Hectopascals.
17. Pressure3pm: Áp suất trung bình của khí quyển đổi với mực nước biển lúc 3 giờ chiều. Đơn vị là Hectopascals.
18. Cloud9am: Một phần bầu trời bị mây che khuất lúc 9 giờ sáng.
19. Cloud3pm: Một phần bầu trời bị mây che khuất lúc 3 giờ chiều.
20. Temp9am: Nhiệt độ lúc 9 giờ sáng.
21. Temp3pm: Nhiệt độ lúc 3 giờ chiều.
22. RainToday: Ghi nhận ngày hôm nay có mưa không. Yes = Có, No = Không.
23. RainTomorrow: Dự đoán ngày mai có mưa hay không. Yes = Có; No = Không
    1. Khí hậu nước Úc
       1. Tổng quan về khí hậu nước Úc

Nước Úc có tổng diện tích khoảng 7.692.024 km2(2.969.907 dặm vuông) là nước có diện tích lớn thứ 6 trên thế giới và khí hậu của nước Úc có sự thay đổi liên tục giữa các vùng và các mùa.Nước úc gồm 6 bang và 2 vùng lãnh thổ mang đặc trưng, khí hậu cũng không có sự khác nhau nhiều.

Tuy nhiên diện tích nước Úc lớn đứng thứ 6 trên thế giới nhưng đa số 80% dân số sống ở các khu vực ven biển với thời tiết ôn hòa và dễ chịu.Gần 1/3 nước Úc là khí hậu nhiệt đới và phần còn lại là khu vực khí hậu ôn đới**.** Úc cũng là châu lục khô nhất trên Trái Đất với lượng mưa trung bình hàng năm ít hơn 600mm

Thời tiết ở Úc ôn hòa hầu như quanh năm,tuy nhiên khí hậu có thể thay đổi do kích thức rộng lớn của châu lục này**.** Hầu hết,các vùng của nước Úc có 4 mùa gồm mùa xuân, mùa hạ, mùa thu và mùa đông. Riêng các vùng như: Darwin, Kakadu, Katherine, Arnhem Land, Broome, Kimberley, Cairns, Daintree Rainforest và The Whitsundays chỉ có 2 mùa là mùa ẩm ướt và mùa khô.

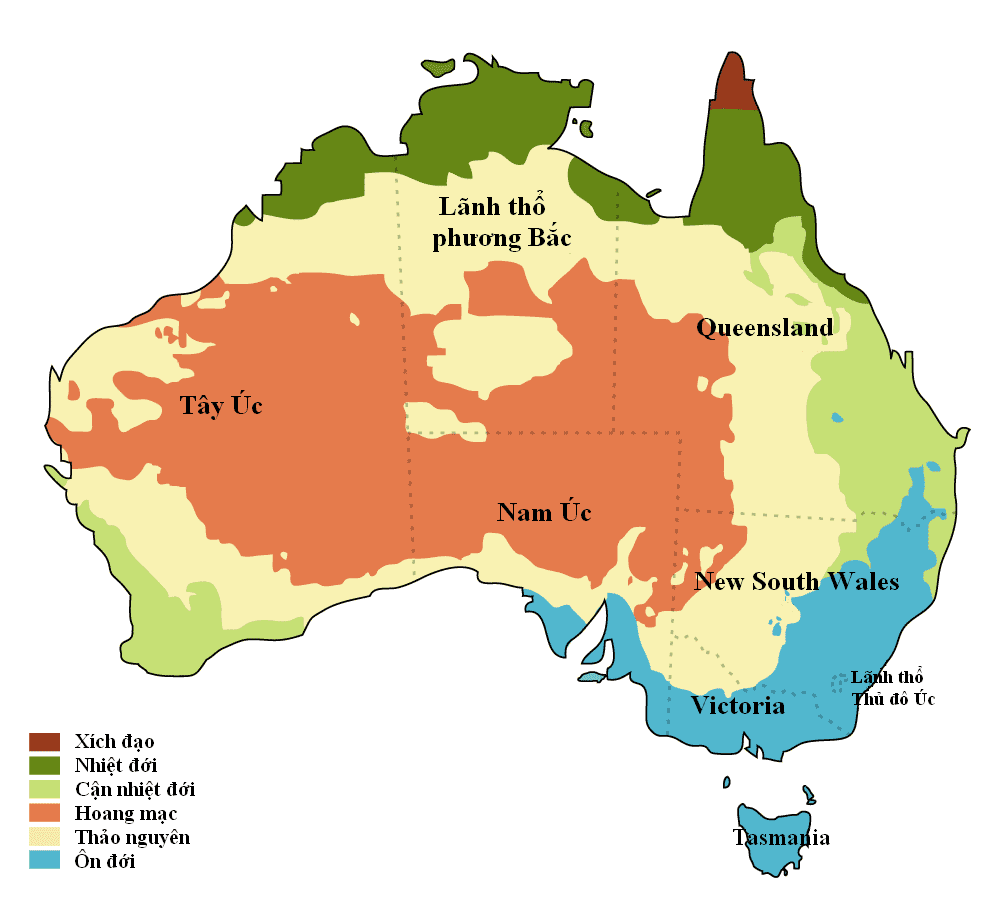
* + 1. Khí hậu nước Úc theo mùa

Khí hậu của nước 4 úc gồm 4 mùa:

* Mùa xuân(Tháng 9 đến tháng 11): Thời tiết ở Úc vào mùa này với khí hậu ngày nhiều nắng còn về ban đêm thì mát mẻ và thoải mái.
* Mùa hè(Tháng 12 đến tháng 2): Từ giữa tháng 12 đến tháng 2 là mùa hè nước Úc. Mùa hè với những ngày thời tiết đẹp và nắng ấm.
* Mùa thu(Từ tháng 3 đến tháng 5): Trừ một số vùng đặc biệt tại Úc,thì mùa thu lại là thời khắc cây cối chuyển từ màu xanh sang sắc đỏ,cam và vàng.Nhiệt độ trở nên mát mẻ và không khí luôn sôi động với các lễ hội tổ chức vào mùa thu.
* Mùa đông( Từ tháng 6 đến tháng 8): Mùa này thì điều đặc trưng của khí hậu miền Nam nước Úc với sự xuất hiện của tuyết. Khí hậu ở Nam úc lạnh và thường đóng băng cho hết mùa đông.

Hầu hết các vùng lãnh thổ ở nước Úc có sự phân chia thành 4 mùa, nhưng có một số khu vực như ở phần tổng quan về khí hậu nước úc có đề cập thì có một số khu vực đặc biệt chỉ có 2 mùa là mùa khô và mùa mưa

* Mùa mưa(Tháng 11 đến tháng 4): Nhiệt độ và độ ẩm vào ban ngày cao,xuất hiện mưa trái mùa và có lốc xoáy.
* Mùa khô (Tháng 5 đến tháng 10):Khí hậu vào mùa này thì ban ngày khí hậu ấm áp và ban đêm thì khí hậu mát mẻ.
  + 1. Khí hậu nước Úc theo vùng



*Hình 3.: bản đồ khí hậu nước Úc*

Bang New South Wales- nằm ở phía đông nam nước Úc. Khí hậu nơi đâythuộc khí hậu cận nhiệt đối và tương đối ôn hòa,mùa đông không quá lạnh và mùa hè thì không quá nóng. Nhiệt độ vào mùa đông thấp nhất không dưới 10 độ C. Ngược lại, vào mùa hè thì nhiệt độc có thể lên đến 40 độ C và có độ ẩm cao gây. Lượng mưa trung bình hàng năm từ 900mm đến 1.500mm.

Bang Queenland nằm phía đông nước Úc, nơi đây có khi hậu nhiệt đới, ấm áp và đa phần diện tích nằm trong vùng nhiệt đới. Tại đây khí hậu vào mùa hè, tức khoảng tháng 12- tháng 2 nhiệt độ trung bình khoảng 30 độ C, thường có mưa rào, thỉnh thoảng có mưa lớn. Mùa đông từ tháng 6- tháng 8 tương đối khô ấm và dễ chịu nhiệt độ trung bình khoảng 17 độ C. Lượng mưa trung bình hàng năm từ 900mm đến 1.500mm.

Bang Nam Úc, khí hậu nơi đây thuộc khí hậu địa trung hải. Mùa hè cực kì nóng và khô, nhiệt độ trung bình lên đến 38 độ C là chuyện bình thường, còn mùa đông lại ẩm ước và 2 mùa dễ chịu nhất ở đây là mùa xuân và mùa thu. Lượng mưa thấp, từ 250mm đến 500mm mỗi năm.

Tiểu bang Tasmania có khí hậu ôn đới, mùa hè ấm áp và mát mẻ, mùa đông mát lạnh. Lượng mưa phân bố khá đều trong năm, từ 500mm đến 1.500mm mỗi năm.

Tiểu bang Victoria thuộc vùng khí hậu ôn đới, mùa hè nơi đây không quá nóng, nhiệt độ chỉ vào khoảng 23 – 26oC. Còn mùa đông tương đối mát chứ không quá lạnh, nhiệt độ dao động từ 6 – 13oC. Đây là một môi trường vừa thích hợp để tham quan, du lịch vừa thích hợp để du học, định cư.Lượng mưa phân bố khá đều trong năm, từ 500mm đến 1.500mm mỗi năm.

Tiểu bang Tây Úc Khí hậu ở Tây Úc rất đa dạng, có sự dịch chuyển từ kiểu khí hậu nhiệt đới ở phía bắc đến khí hậu ôn đới ở phía Nam. Khí hậu ở phía nam tương đối khô, lượng mưa trung bình thấp, trên dưới 300 mm/năm. Về phía bắc, do ảnh hưởng của khí hậu nhiệt đới nên lượng mưa7 cao hơn, vào khoảng 500 – 1500 mm/năm. Biên độ nhiệt tương đối lớn giữa các vùng trong tiểu bang.

Vùng lãnh thổ Bắc Úc nằm trong vùng khí hậu nhiệt đới. Vùng này có mùa mưa kéo dài và mùa khô ngắn, nhiệt độ ấm quanh năm và lượng mưa hàng năm trung bình từ 1.500mm đến 2.500mm.

Vùng lãnh thổ Thủ đô Úc (ACT) nằm gần tiểu bang New South Wales và thuộc vùng khí hậu ôn đới. Vùng này có mùa hè ấm áp và mát mẻ, mùa đông mát lạnh. Lượng mưa phân bố khá đều trong năm, từ 500mm đến 1.500mm mỗi năm.

* + 1. Khí hậu nước Úc theo từng địa điểm trong data

**Albury:** Thuộc tiểu bảng New South Wales Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 600mm.

**Badgerys Creek**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 800mm.

**Cobar:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 300mm.

**Coffs Harbour:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu nhiệt đới. Mùa hè ấm áp và ẩm, mùa đông ấm mát. Lượng mưa trung bình hàng năm khoảng 1.800mm.

**Moree:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 600mm.

**Newcastle:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.200mm.

**Norah Head**:

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè ấm áp và ẩm, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.200mm.

**Norfolk Island:**

Thuộc tiểu bang: Norfolk Island (lãnh thổ hải ngoại thuộc Úc)

Khí hậu: Vùng khí hậu nhiệt đới ẩm. Mùa hè mát mẻ, mưa phân bố quanh năm. Lượng mưa trung bình hàng năm khoảng 1.400mm

**Penrith:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 650mm.

**Richmond:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 680mm.

**Sydney:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè ấm áp và mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.200mm.

**Sydney Airport:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khíhậu ôn đới. Mùa hè ấm áp và mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.200mm.

**Wagga Wagga:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 550mm.

**Williamtown:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè ấm áp và ẩm, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.200mm.

**Wollongong:**

Thuộc tiểu bang: New South Wales

Khí hậu: Vùng khí hậu ôn đới. Mùa hè ấm áp và ẩm, mùa đông mát mẻ. Lượng mưa trung bình hàng năm khoảng 1.300mm.Làm sạch dữ liệu

**Canberra:**

Thuộc tiểu bang: Australian Capital Territory

Khí hậu: Vùng khí hậu ôn đới lục địa. Mùa hè nóng và khô, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 600mm.

**Tuggeranong:**

Thuộc tiểu bang: Australian Capital Territory

Khí hậu: Vùng khí hậu ôn đới lục địa. Mùa hè nóng và khô, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 600mm.

**Mount Ginini:**

Thuộc tiểu bang: Australian Capital Territory

Khí hậu: Vùng khí hậu ôn đới với khí hậu núi cao. Mùa hè mát mẻ và mùa đông lạnh. Lượng mưa trung bình hàng năm khoảng 1.800mm.

**Ballarat:**

Thuộc tiểu bang: Victoria

Khí hậu: Vùng khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 600mm.

**Bendigo:**

Thuộc tiểu bang: Victoria

Khí hậu: Vùng khí hậu ôn đới. Mùa hè nóng và khô, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 500mm.

**Sale**

Thuộc tiểu bang: Victoria

Khí hậu: Vùng khí hậu ôn đới. Mùa hè mát mẻ và ẩm, mùa đông lạnh và ẩm. Lượng mưa trung bình hàng năm khoảng 600mm.

**Melbourne Airport:**

Thuộc tiểu bang: Victoria

Khí hậu: Vùng khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 650mm.

**Melbourne:**

Thuộc tiểu bang: Victoria

Khí hậu: Vùng khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và khô. Lượng mưa trung bình hàng năm khoảng 650mm.

**Mildura**:

Thuộc tiểu bang: Victoria

Khí hậu: Thành phố Mildura có khí hậu ôn đới nóng. Mùa hè nóng và khô, mùa đông mát mẻ.

Lượng mưa hàng năm: Trung bình khoảng 280mm.

**Nhil**:

Thuộc tiểu bang: Victoria

Khí hậu: Thành phố Nhill có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông lạnh và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Portland**:

Thuộc tiểu bang: Victoria

Khí hậu: Thành phố Portland có khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và ẩm.

Lượng mưa hàng năm: Trung bình khoảng 660mm.

**Watsonia**:

Thuộc tiểu bang: Victoria

Khí hậu: Thành phố Watsonia có khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Dartmoor**:

Thuộc tiểu bang: Victoria

Khí hậu: Thành phố Dartmoor có khí hậu ôn đới. Mùa hè mát mẻ và ẩm, mùa đông lạnh và ẩm.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Brisbane**:

Thuộc tiểu bang: Queensland

Khí hậu: Thành phố Brisbane có khí hậu nhiệt đới ẩm. Mùa hè nóng và ẩm, mùa đông ấm áp và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Cairns**:

Thuộc tiểu bang: Queensland

Khí hậu: Thành phố Cairns có khí hậu nhiệt đới ẩm. Mùa hè nóng và ẩm, mùa đông ấm áp và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Gold Coast:**

Thuộc tiểu bang: Queensland

Khí hậu: Thành phố Gold Coast có khí hậu nhiệt đới ẩm. Mùa hè ấm áp và ẩm, mùa đông ấm mát.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Townsville**:

Thuộc tiểu bang: Queensland

Khí hậu: Thành phố Townsville có khí hậu nhiệt đới ẩm. Mùa hè nóng và ẩm, mùa đông ấm áp và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Adelaide**:

Thuộc tiểu bang: South Australia

Khí hậu: Thành phố Adelaide có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và ẩm.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Mount Gambier**:

Thuộc tiểu bang: South Australia

Khí hậu: Thành phố Mount Gambier có khí hậu ôn đới. Mùađông mát mẻ và ẩm, mùa hè mát mẻ và mưa nhiều.

Lượng mưa hàng năm: Trung bình khoảng 900mm.

**Nuriootpa**:

Thuộc tiểu bang: South Australia

Khí hậu: Thành phố Nuriootpa có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Woomera**:

Thuộc tiểu bang: South Australia

Khí hậu: Thành phố Woomera có khí hậu ôn đới nóng. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Albany**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Albany có khí hậu ôn đới. Mùa hè mát mẻ và ẩm, mùa đông mát mẻ và mưa nhiều.

Lượng mưa hàng năm: Trung bình khoảng 860mm.

**Witchcliffe**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Witchcliffe có khí hậu ôn đới. Mùa hè mát mẻ và khô, mùa đông mát mẻ và mưa nhiều.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Pearce RAAF**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Pearce RAAF có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Perth Airport**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Perth Airport có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Perth**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Perth có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Salmon Gums**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Salmon Gums có khí hậu ôn đới. Mùa hè nóng và khô, mùa đông mát mẻ và khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Walpole**:

Thuộc tiểu bang: Western Australia

Khí hậu: Thành phố Walpole có khí hậu ôn đới. Mùa hè mát mẻ và ẩm, mùa đông mát mẻ và mưa nhiều.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Hobart**:

Thuộc tiểu bang: Tasmania

Khí hậu: Thành phố Hobart có khí hậu ôn đới. Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và ẩm.

Lượng mưa hàng năm: Trung bình khoảng 600mm.

**Launceston**:

Thuộc tiểu bang: Tasmania

Khí hậu: Thành phố Launceston có khí hậu ôn đới.Mùa hè mát mẻ và mưa nhiều, mùa đông lạnh và ẩm.

Lượng mưa hàng năm: Trung bình khoảng 600mm.

**Alice Springs**:

Thuộc tiểu bang: Northern Territory

Khí hậu: Alice Springs có khí hậu sa mạc nhiệt đới.Mùa hè nóng và khô, mùa đông mát mẻ và hanh khô.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Darwin**:

Thuộc tiểu bang: Northern Territory

Khí hậu: Thành phố Darwin có khí hậu nhiệt đới gió mùa.Mùa mưa và mùa khô rõ rệt, nhiệt độ cao quanh năm.

Lượng mưa hàng năm: Trung bình khoảng 1700mm.

**Katherine**:

Thuộc tiểu bang: Northern Territory

Khí hậu: Thành phố Katherine có khí hậu nhiệt đới gió mùa.Mùa mưa và mùa khô rõ rệt, nhiệt độ cao quanh năm.

Lượng mưa hàng năm: Thông tin không có sẵn.

**Uluru**:

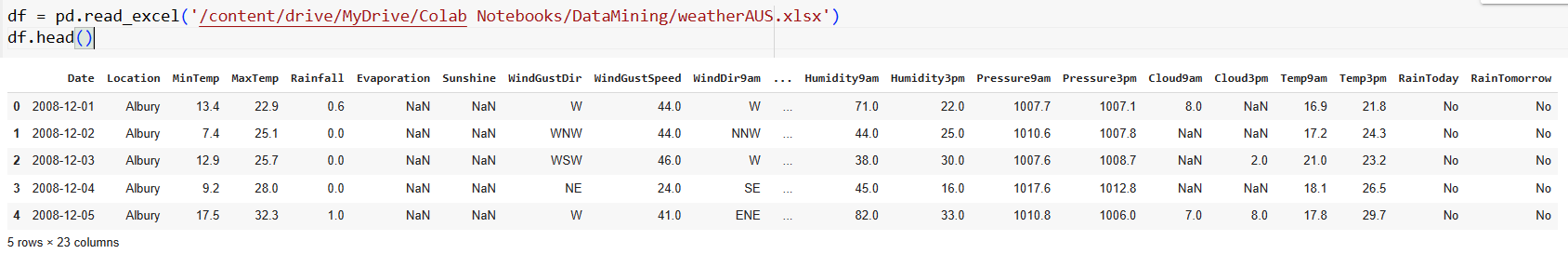
Thuộc tiểu bang: Northern Territory

Khí hậu: Khu vực xung quanh Uluru có khí hậu sa mạc nhiệt đới.Mùa hè nóng và khô, mùa đông mát mẻ.

Lượng mưa hàng năm: Trung bình rất thấp, khoảng 250mm.

* 1. Xử lý dữ liệu
     1. Khai phá dữ liệu

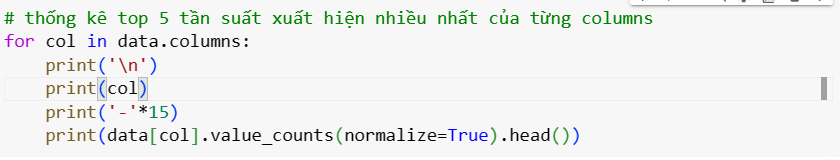
Đầu tiên, load tập dữ liệu lên Google Colab và đọc thông tin bằng cách sử dụng thư viên Pandas trong Python. Tập dữ liệu sẽ được lưu trữ dưới dạng DataFrame.



*Hình 3.1: Hình ảnh một phần code đọc thông tin về tập dữ liệu.*

Sau khi, xem thông tin của tập dữ liệu. Nhận xét:

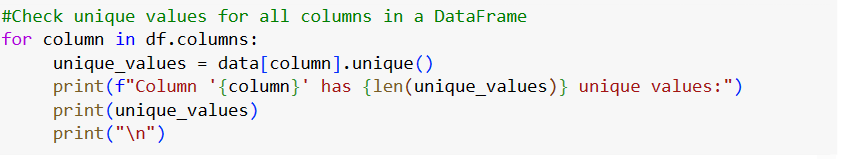
* Có 145460 dòng dữ liệu, được đánh số tự 0 đến 145459, gồm 23 thuộc tính.
* Cột ‘’Date’’ không cần chuyển đổi sang datetime.
* Có 2 kiểu dữ liệu: float64 và object.
* Xuất hiện khá nhiều giá trị null. Các thuộc tính Evaporation, Sunshine, Cloud9am, Cloud3pm có giá trị null trên 35%. Ngoài Date và Location, tất cả các thuộc tính còn lại đều có giá trị null.
* Không có dòng dữ liệu nào bị trùng lặp.



*Hình 3.2: Code tống kê top 5 tần suất xuất hiện nhiều nhất của từng thuộc tính.*

Thống kê top 5 tần suất xuất hiện nhiều nhất của các thuộc tính. Nhận xét:

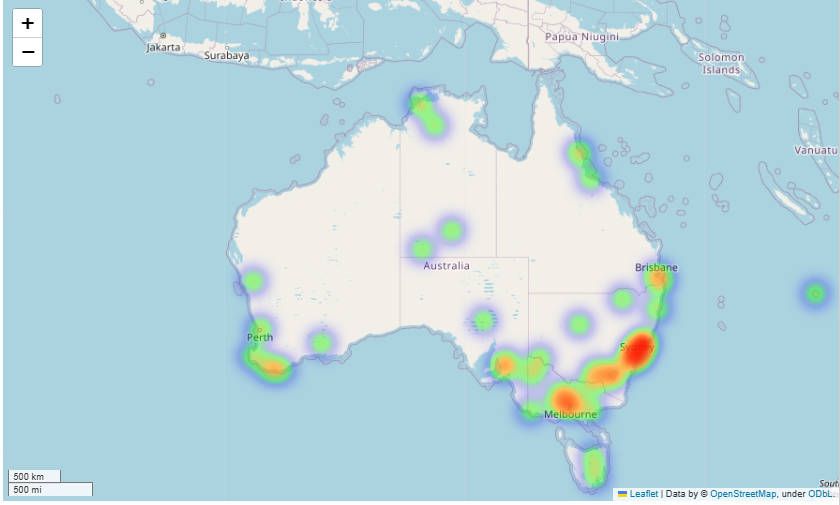
* Có sự ngắt kết nối giữa số lượng giá trị thuộc tính Rainfall và số lượng giá trị thuộc tính RainToday/RainTomorrow. Trong khi, các ngày quan sát lượng mưa của thuộc tính Rainfall có giá trị bằng 0 (mm) đạt đến 64% thì số ngày không có mưa của RainToday/RainTomorrow khoảng 75%. Sự khác biệt này có thể là do sự khác biệt về số lượng giá trị bị thiếu cho mỗi cột.



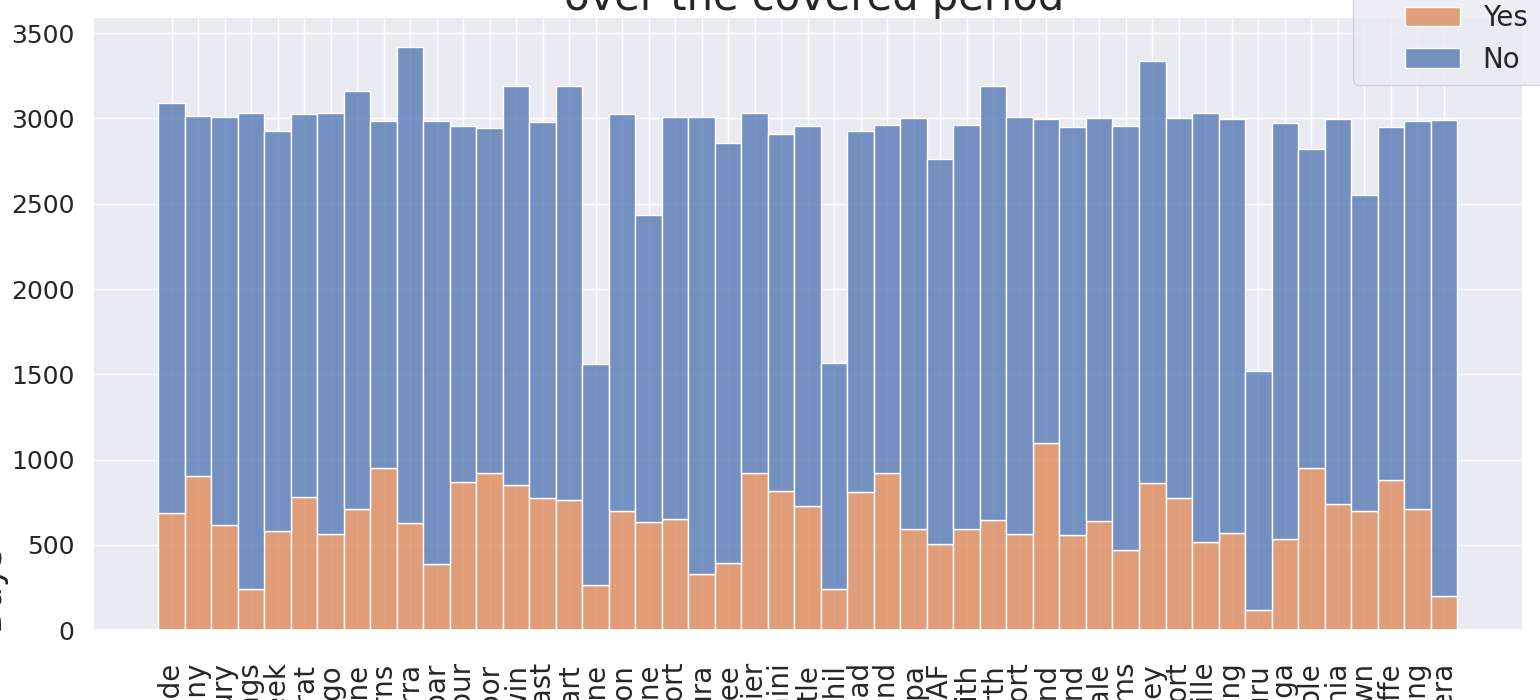
*Hình 3.3: Code kiểm tra giá trị unique*

Kiểm tra các giá trị riêng biệt của từng thuộc tính, ta thấy rằng:

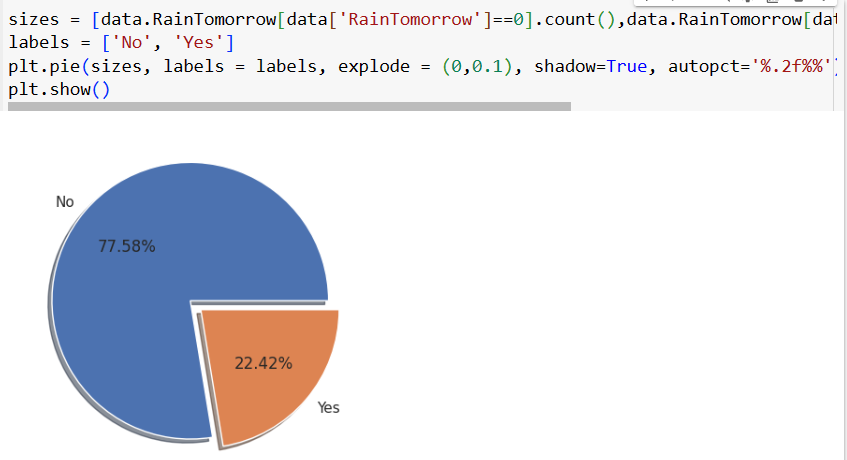
* Giá trị “Date” có sự bất thường không giống với mô tả ban đầu là khảo sát 10 năm.
* Ngày bắt đầu khảo sát là 01-11-2007 đến ngày 25-06-2017.
* Có 49 vùng được khảo sát tại nước Úc. Với ngày bắt dầu khảo sát của 49 vùng đều khác nhau, nhưng có chung ngày kết thúc. 49 vùng có thể gom nhóm vào 9 khu vực theo bản đồ địa lý của nước Úc. (Hình 3.4)
* Số ngày mưa hay không mưa vào ngày mai của từng khu vực cũng có sự khác nhau.



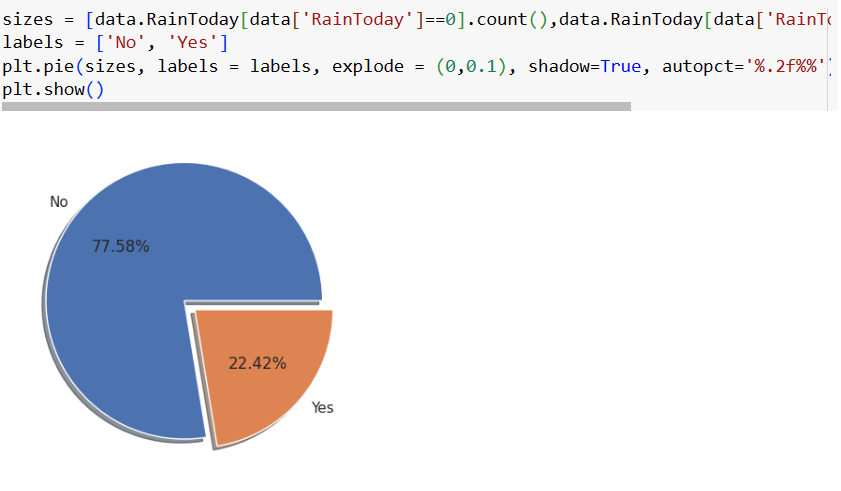
*Hình 3.4: Bản đồ khu vực nước Úc*



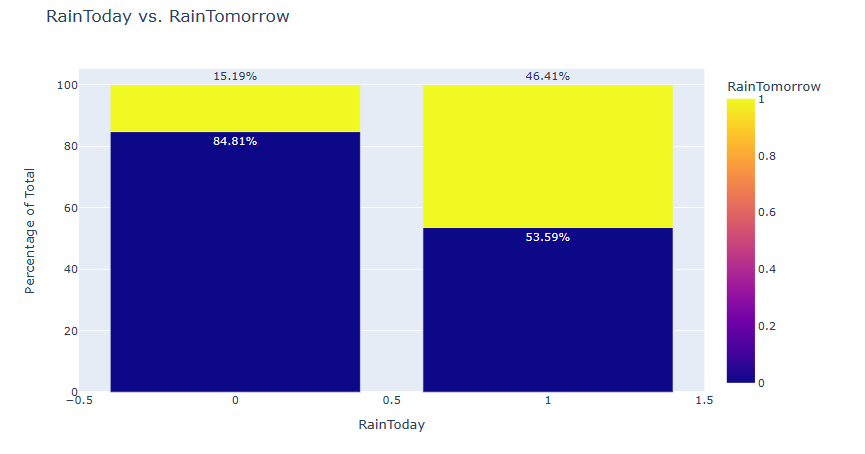
*Hình 3.5: Biểu đồ thể hiện số ngày quan sát của tất cả các vùng*



*Hình 3.7: Biểu đồ thể hiện phần trăm số ngày mưa và không mưa của thuộc tính RainTomorrow*



*Hình 3.7: Biểu đồ thể hiện phần trăm số ngày mưa và không mưa của thuộc tính RainToday*



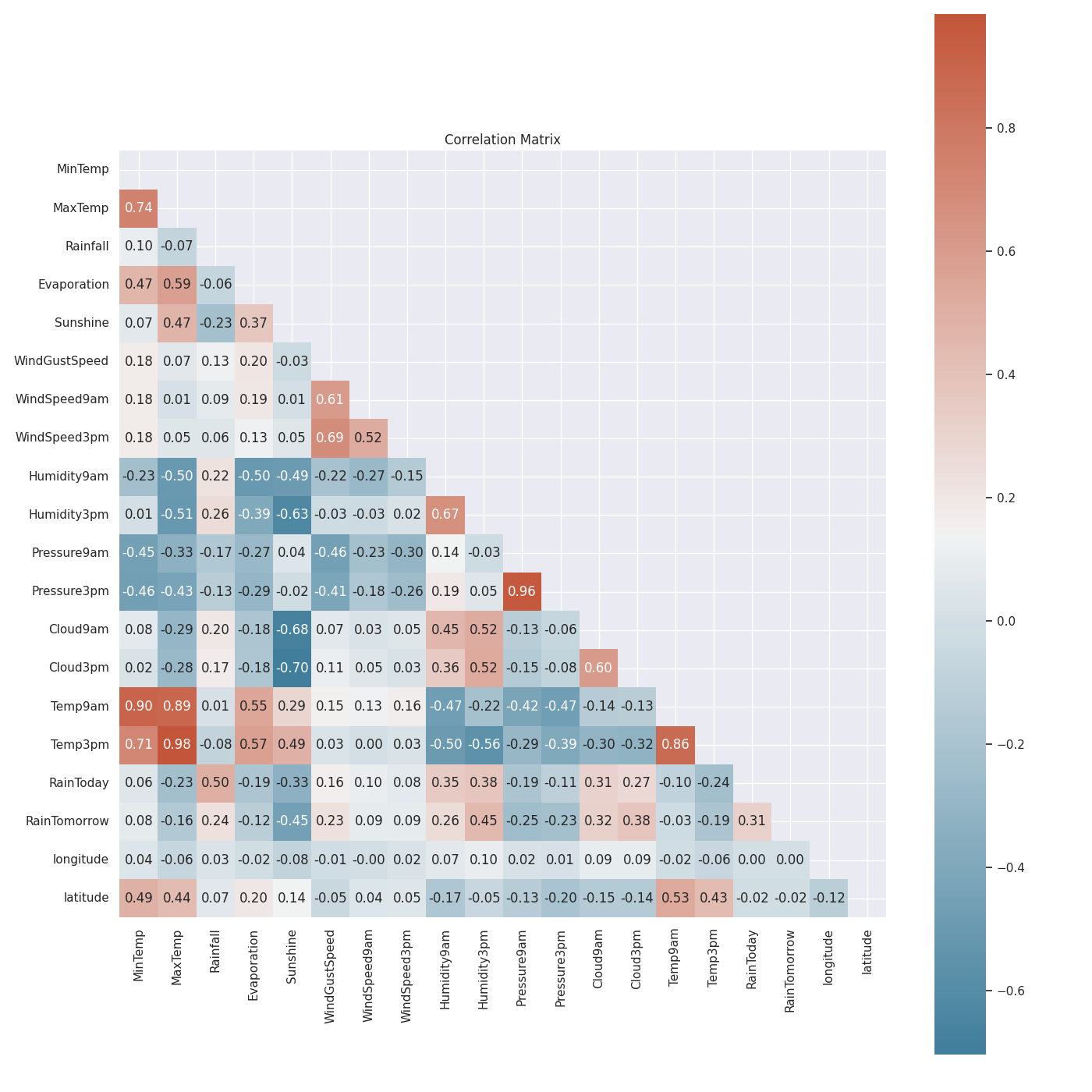
*Hình 3.8: Biểu đồ thể hiện mối quan hệ giữa RainToday – RainTomorrow*

Nhìn vào hai biểu đồ hình tròn trên, cho ta thấy số lượng ngày mưa và ngày không mưa của hai thuộc tính raintoday và raintomorrow là như nhau (77.58% không có mưa trên tổng số ngày quan sát, 22.42% có mưa trên tổng số ngày quan sát). Đều này, không chứng minh là raintoday và raintomorrow tương ứng với nhau. Khi vẽ biểu đồ hình cột thể hiện mối quan hệ giữa thuộc tính raintoday với raintomorrow thì ta thấy, ngày hôm nay không có mưa (raintoday = No) thì ngày mai sẽ có 84.81% không mưa (raintomorrow = No). Ngược lại, nều ngày hôm nay có mưa (raintoday = Yes) thì ngày mai sẽ có 53.59% không mưa (RainTomorrow = No).

Trực quan hóa toàn bộ dữ liệu, nhận xét:

* Hầu hết các thuộc tính dều phân phối chuẩn như mong đợi.
* Sự phân bố của Rainfall cần được xem xét vì sự ngoại lai khá nhiều và rõ rệt.
* Sự phân bố của Sunshine rất thú vị:
  + Tần suất xuất hiện của giá trị 0 cao hơn đều đó có nghĩa là có nhiều ngày bị mây che mờ.
  + Sự suy giảm đột ngột của các giá trị sau 11 giờ là do sự hạn chế của những ngày được chiếu sang trong năm.
* Sự phân bố của Humidity9am có sự chuyển biến mạnh mẽ, tăng đột ngột 100%.

Nhìn vào mối tương quan của tập dữ liệu, ta thấy biến target RainTomorrow có tương quan đến thuộc tính Humidity3pm là cao nhất.



*Hình 3.9: Biểu đồ Heatmap*

* + 1. Xử lý dữ liệu

Sau khi tìm hiểu sơ bộ về tập dữ liệu, tiến hành xử lý dữ liệu để xây dựng model. Kỹ thuật xử lý dữ liệu mà nhóm tác giả thực hiện là điền những giá trị còn thiếu trong tập dữ liệu bằng giá trị trung bình đối với biến numerical và giá trị xuất hiện nhiều nhất đối với biến categorical; xóa đi những thuộc tính không cần thiết và những dòng bị thiếu giá trị mà không thể điền được nữa. Các bước thực hiện như sau:

*Bước 1*: Tạo thêm thuộc tính “month”, “year”, “quarter” cho tập dữ liệu dựa trên thuộc tính “Date”.

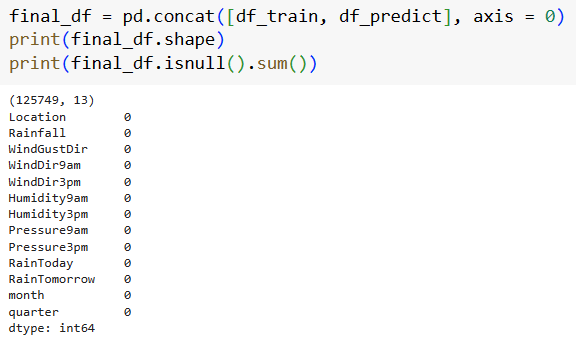
*Bước 2:* Tính toán phần trăm giá trị bị thiếu của từng thuộc tính trong tập dữ liệu, xóa đi những thuộc tính mà phần trăm giá trị bị thiếu là quá nhiều.

*Bước 3*: Chọn ra những thuộc tính có kiểu dữ liệu là mumerical để điền giá trị bị thiếu bằng giá trị trung bình theo tháng.

*Bước 4*: Chọn ra những thuộc tính có kiểu dữ liệu là categorical (trừ thuộc tính target RainTomorrow và RainToday) để điền giá trị bị thiếu bằng giá trị xuất hiện nhiều nhất của tháng.

*Bước 5*: Xử lý dữ liệu bị thiếu của thuộc tính target bằng thuật toán DecisionTree.

Kết quả sau khi thực hiện các bước trên, thu được tập dữ liệu gồm 125749 dòng dữ liệu và 13 thuộc tính bao gồm như hình sau đây:

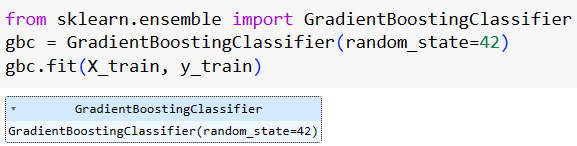


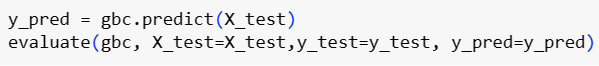
* 1. GradientBoosting

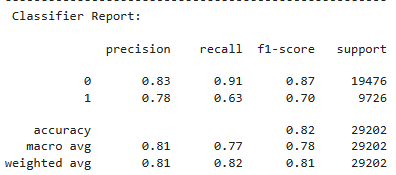
Bước 1:

* Sử dụng kĩ thuật SMOTE để gia tăng class thiểu số (50%)
* Chuẩn hoá dữ liệu bằng MinMaxScaler
* Chia tập dữ liệu thành 80% training và 20% test
* Dữ liệu ban dầu: 145460 dòng và 23 cột.
* Dữ liệu sau khi qua xử lý: 146010 dòng và 13 cột.
* Train set: 116808 dòng và 13 cột
* Test set: 29202 dòng và 13 cột

Bước 2: Tiến hành xây dựng model Gradient Boosting Classifier bằng thư viện Scikit learn để dự đoán và ước tính kết quả.

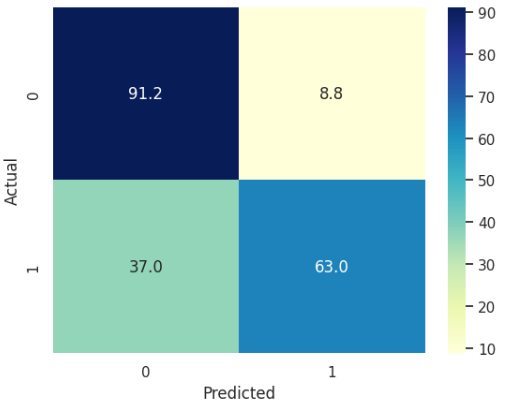


 *Classifier Report:*



*Nhận xét:*

Nhìn vào số liệu bên trên, ta thấy mô hình Gradient Boosting Classifier có độ phân loại tốt trên tập dữ liệu Rain in Austrailia với f1-score tương đối ổn.



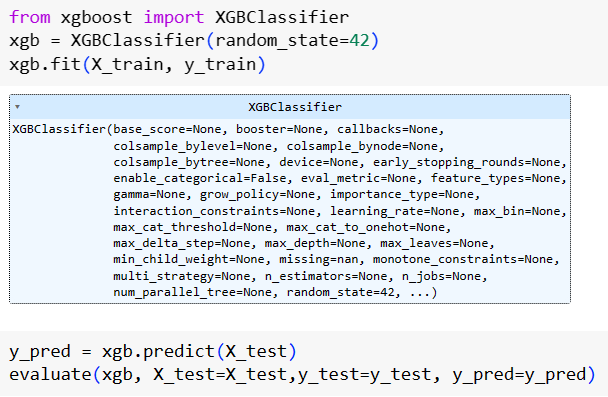
Dựa vào Confusion Matrix, ta có một số nhận xét:

* Dự đoán chính xác ngày mai không có mưa là 91.2%.
* Dự đoán sai lệch ngày mai sẽ có mưa là 8.8%.
* Dự đoán chính xác ngày mai sẽ có mưa là 37%.
* Dự đoán sai lệch ngày mai sẽ có mưa là 63%.
  1. XGBoost

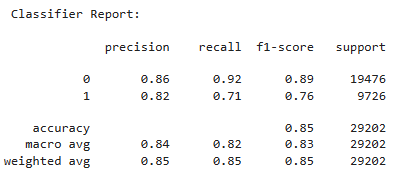
Bước 1:

* Sử dụng kĩ thuật SMOTE để gia tăng class thiểu số (50%)
* Chuẩn hóa dữ liệu bằng MinMaxScaler
* Chia tập dữ liệu thành 80% training và 20% test
* Dữ liệu ban dầu: 145460 dòng và 23 cột.
* Dữ liệu sau khi qua xử lý: 146010 dòng và 13 cột.
* Train set: 116808 dòng và 13 cột
* Test set: 29202 dòng và 13 cột

Bước 2: Tiến hành xây dựng model XGBoost Classifier bằng thư viện Scikit learn để dự đoán và ước tính kết quả.

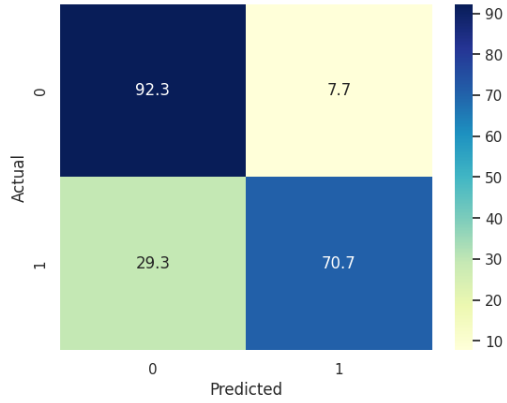


*Classifier Report:*



*Nhận xét:*

Nhìn vào số liệu bên trên, ta thấy mô hình XGBoost Classifier có độ phân loại tốt trên tập dữ liệu Rain in Austrailia với f1-score ổn định hơn so với GBoost.



Dựa vào Confusion Matrix, ta có một số nhận xét:

* Dự đoán chính xác ngày mai không có mưa là 92.3%.
* Dự đoán sai lệch ngày mai sẽ có mưa là 7.7%.
* Dự đoán chính xác ngày mai sẽ có mưa là 70.7%.
* Dự đoán sai lệch ngày mai sẽ không có mưa là 29.3%.
  1. Tunning Hyperparameter

Đối với quá trình tinh chỉnh tham số, nhóm đã sử dụng những phương pháp sau:

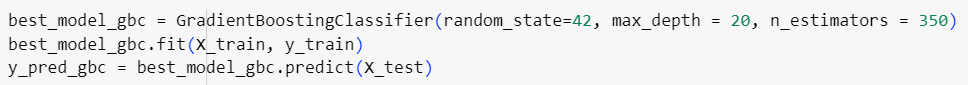
* GridSearchCV
* Kfold – cross validation

Các Hyper parameter được sử dụng cho cả hai mô hình:

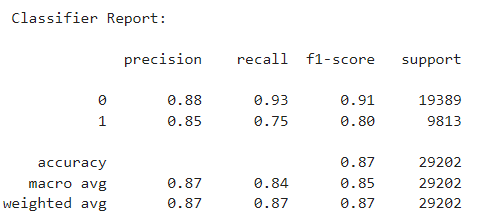
* n\_estimators
* max\_depth
  + 1. Gradient Boosting

Kết quả tốt nhất tìm được trên GridSearchCV được sử dụng để đi xây dựng model

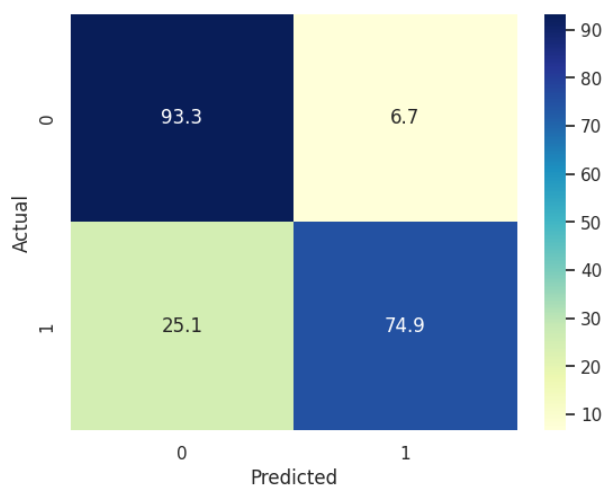
*Final model:*



Kết quả:



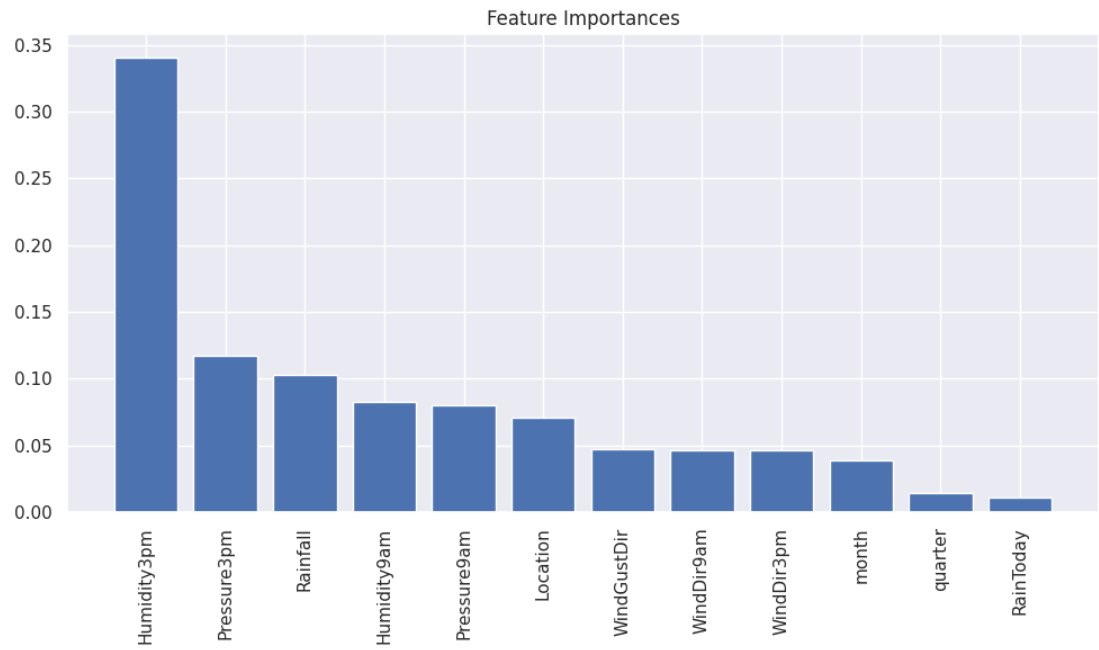
*Confusion matrix:*



*Accuracy:*



*Features Importance:*

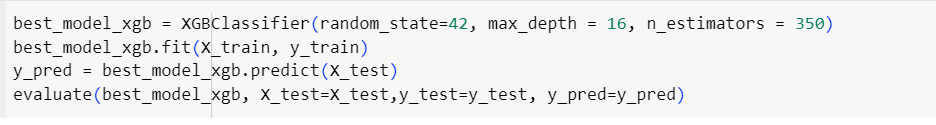


Nhận xét

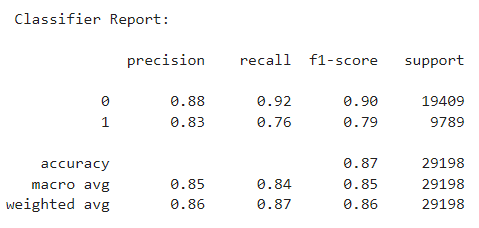
* Đối với f1-score và confusion matrix, model với thông số tốt nhất từ GridSearchCV cho kết quả khá tốt, có cải thiện trong vấn đề imbalance dataset.
* Accuracy score tương đối cao: 85.21%, mô hình lúc này đã khá ổn nhưng vẫn có thể tiếp tục điều chỉnh để kết quả tốt hơn.
* Features Importance cho thấy rằng, humidity3am là một thuộc tính quan trọng nhất trong model, điều này đúng như phán đoán ban đầu trong quá trình khai phá.
  + 1. XGBoost:

Kết quả tốt nhất tìm được trên GridSearchCV được sử dụng để đi xây dựng model

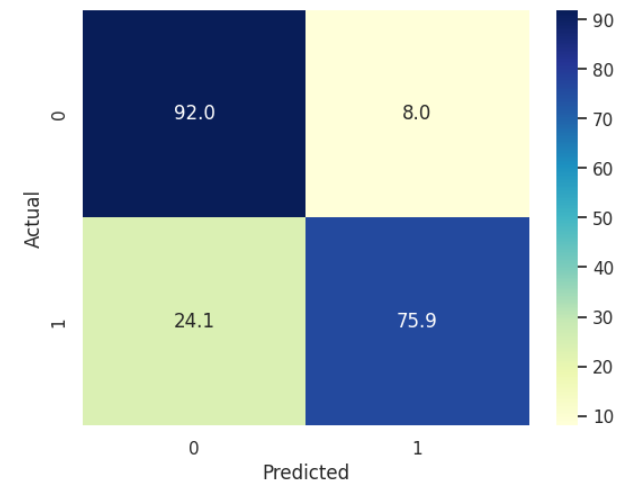
*Final model:*



Kết quả:



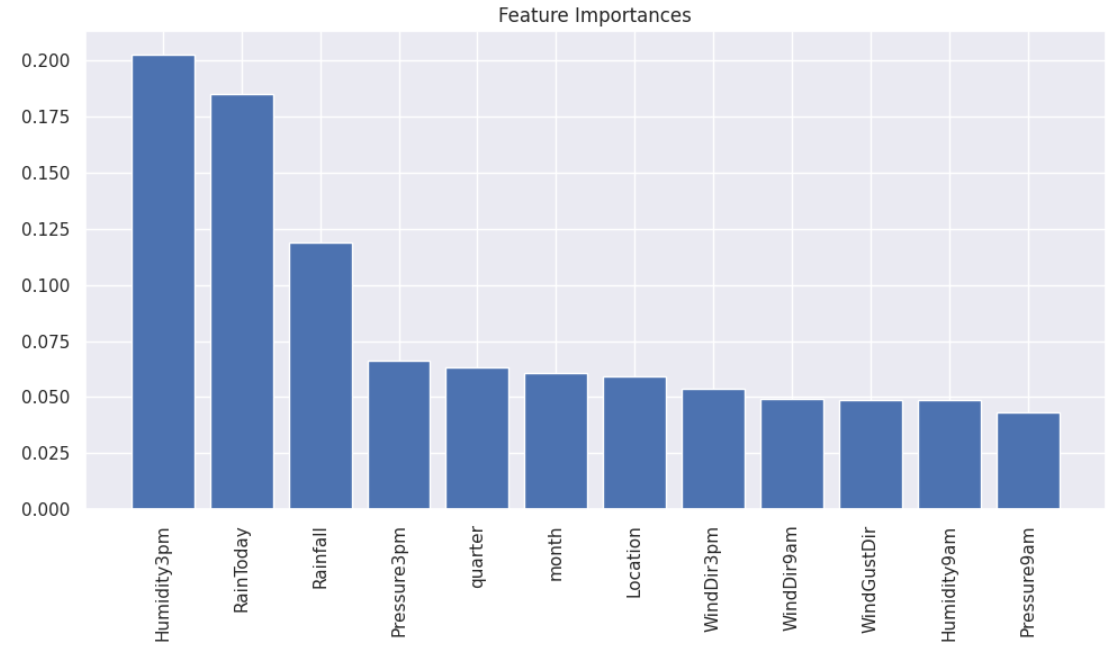
*Confusion matrix:*



*Accuracy:*



*Features Importance:*



Nhận xét

* Đối với f1-score và confusion matrix, model với thông số tốt nhất từ GridSearchCV cho kết quả khá tốt, có cải thiện trong vấn đề imbalance dataset.
* Accuracy score tương đối cao: 87.04%, mô hình lúc này đã khá ổn nhưng vẫn có thể tiếp tục điều chỉnh để kết quả tốt hơn.
* Features Importance cho thấy rằng, humidity3am là một thuộc tính quan trọng nhất trong model, điều này đúng như phán đoán ban đầu trong quá trình khai phá. Các thuộc tính quan trọng khác là RainToday và RainFall lần lượt là thuộc tính quan trọng tiếp theo. Ngoài ra, những thuộc tính còn lại đều nằm ở mức ngang nhau về mức quan trọng, không quá đề cao vào duy nhất một thuộc tính như ở GBoost.
  1. Đánh giá mô hình

Nhóm tác giả đã lựa chọn phương pháp Xác thực chéo (Cross-validation) để ước tính độ chính xác và lựa chọn mô hình ROC để lựa chọn mô hình của trình phân loại.

**Xác thực chéo:**

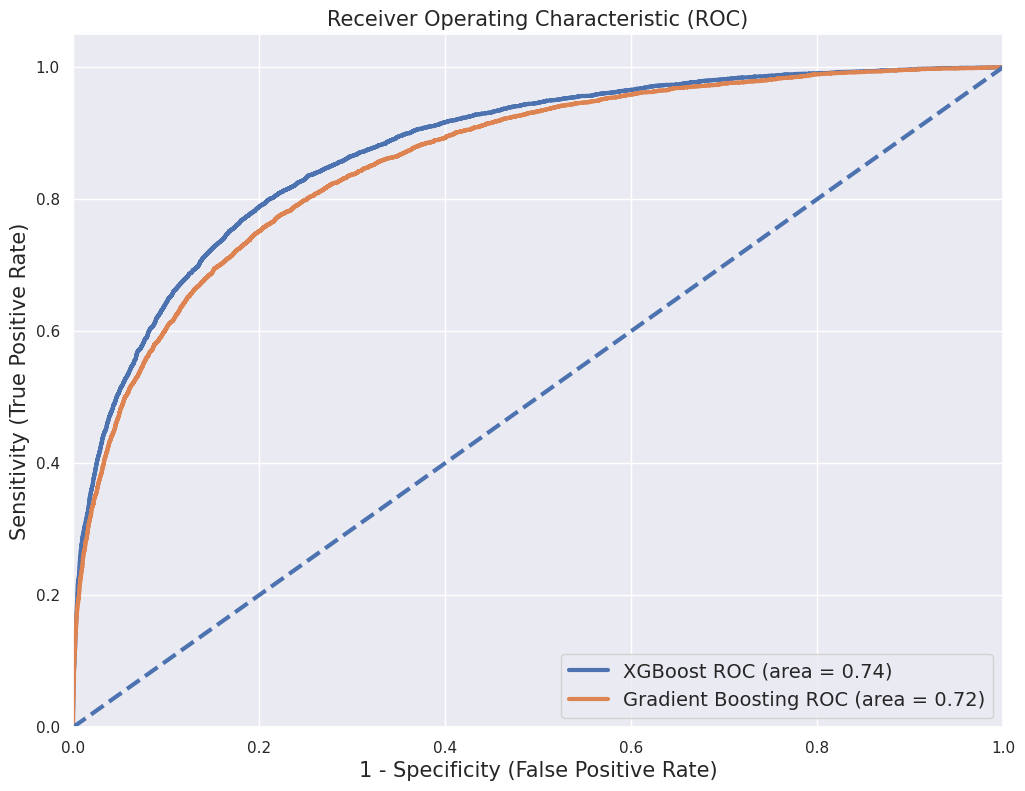
* Phân vùng dữ liệu ngẫu nhiên thành k tập con loại trừ lẫn nhau, mỗi tập có kích thước xấp xỉ bằng nhau
* Ở lần lặp thứ i, sử dụng Di làm bộ kiểm tra và các bộ khác làm bộ huấn luyện
* Bỏ lại 1 lần: k lần gấp, k = số bộ dữ liệu, đối với dữ liệu có kích thước nhỏ
* Xác nhận chéo phân tầng: Các nếp gấp được phân tầng để phân lớp, mỗi nếp gấp là khoảng giống như trong dữ liệu ban đầu



*Hình 3.10: Hình ảnh minh hoa cho Cross-validation.*

**Mô hình ROC** (Receiver Operating Characteristics): để so sánh trực quan các mô hình phân loại

* Bắt nguồn từ lý thuyết phát hiện tín hiệu
* Cho thấy sự cân bằng giữa tỷ lệ dương tính thực sự và tỷ lệ dương tính giả
* Diện tích dưới đường cong ROC là thước đo độ chính xác của mô hình
* Xếp hạng các bộ giá trị thử nghiệm theo thứ tự giảm dần: bộ giá trị có nhiều khả năng thuộc về lớp tích cực nhất xuất hiện ở đầu danh sách
* Càng gần đường chéo (tức là vùng càng gần 0,5), mô hình càng kém chính xác
* Trục dọc biểu thị tỷ lệ dương thực sự
* Đại diện trục ngang biểu thị tỷ lệ dương tính giả
* Biểu đồ cũng cho thấy một đường chéo
* Một mô hình với độ chính xác hoàn hảo sẽ có diện tích là 1,0
  + 1. Kết luận



Dựa vào kết quả tính toán bên trên, chúng em đã tổng hợp thống kê ra bảng số liệu so sánh các mô hình thuật toán như sau:

|  |  |  |
| --- | --- | --- |
| So sánh Thuật toán | Gradient Boosting Classifier | XGBoost Classifier |
| Average Accuracy |  | 0.8367 |
| Standard Deviation |  | 0.0052 |
| TP |  | 75.9% |
| FP |  | 8% |
| TN |  | 92% |
| FN |  | 24.1% |
| Tổng điểm (%) |  | 87.04% |

*Chú thích:*

* Average Accuracy: Độ chính xác trung bình
* Standard Deviation: Độ lệch chuẩn
* TP (True Positive): Số lượng dự đoán chính xác ngày mai không có mưa.
* FP (False Positive): Số lượng dự đoán sai lệch ngày mai không có mưa.
* TN (True Negative): Số lượng dự đoán chính xác ngày mai sẽ có mưa.
* FN (False Negative): Số lượng dự đoán sai lệch ngày mai sẽ có mưa.

*Nhận xét:*

Nhìn vào bảng số liệu so sánh, nhận xét chung các thuật toán phân loại đều cho ra

độ chính xác cao trên mô hình Water Quality, cụ thể:

* Với 2 mô hình:
  + Gradient Boosting Classifier có điểm số đạt 84.12%
  + XGBoost Classifier có điểm số đạt 84.72%
* Do mô hình XGBoost Classifier đạt điểm số cao nhất trong bảng số liệu vừa so sánh. Nên có thể nói đây là mô hình phân loại tối ưu nhất trên tập dữ liệu Rain in Austrailia.

1. TỔNG KẾT
   1. Kết luận

Classification là lĩnh vực khá quan trọng trong khai phá dữ liệu nhằm đưa ra những dự đoán, xu hướng trong tương lai, nó được ứng dụng trong nhiều ngành như y tế, thương mại, ngân hàng, giáo dục, nông nghiệp ... Trong quá trình tìm hiểu và hoàn thành bài tập lớn với đề tài “Rain in Australia”. Nhóm em đã đạt được một số kết quả như sau:

* Tìm hiểu tổng quan về khai phá dữ liệu, bài toán dự đoán, qua các mô hình hồi quy thuộc Ensemble như XGBoost, Gradient Boosting. Hỗ trợ dự đoán ngày mai mưa hay không.
* Thu thập dữ liệu trên Kaggle, tiền xử lý dữ liệu bằng thư viện pandas trong python. Xây dựng nên mô hình phân lớp trên Google Collaboration ...
* Đánh giá các mô hình phân lớp khác nhau.
* Tìm hiểu về ngôn ngữ Python các các thư viện hỗ trợ như Pandas, Matplotlib, Sci-learn.

***Hạn chế***:

Thời gian triển khai ngắn, nên chúng em còn nhiều sơ sót và kiến thức của chúng em về mặt toán học và khả năng lập trình còn hạn chế.

* 1. Hướng phát triển

Xây dựng, cải tiến mô hình dự đoán với phương pháp học máy khác như CNN,...Tích hợp các mô hình học máy hiện đại, nhằm dự đoán qua hình ảnh theo không gian và thời gian, qua video realtime.

Trong quá trình hoàn thành bài tập lớn, chúng em đã cố gắng tìm hiểu và tham khảo các tài liệu liên quan. Tuy nhiên, thời gian có hạn nên chúng em sẽ không tránh khỏi những thiếu sót, rất mong nhận được sự đóng góp ý kiến của quý thầy cô và các bạn để báo cáo và kỹ năng của chúng em ngày được hoàn thiện hơn và có thế áp dụng trong thực tiễn.