**Chương II: Cơ sở lý thuyết**

2.1. Máy học (Machine Learning)

Các thuật toán phân loại cũng đối diện với vấn đề underfitting và overfitting tương tự như trong bài toán hồi quy. Vì vậy, việc thử nhiều phương pháp với các cách chọn đặc trưng và mức độ chính quy hóa khác nhau là cần thiết để tìm ra phương án tốt nhất.

Có nhiều cách để đánh giá hiệu suất của một phương pháp phân loại. Một cách đơn giản là sử dụng một tập dữ liệu thử nghiệm và tính tỉ lệ dự đoán đúng trên tập này. Phương pháp nào có tỉ lệ dự đoán đúng cao nhất sẽ được ưu tiên lựa chọn.

Từ đó nhóm đã đưa ra 4 mô hình máy học để sử dụng:

2.1.1. Hồi quy logistic (Logistic Regression)

Logistic Regression là một thuật toán phân loại được sử dụng trong Machine Learning để dự đoán hoặc phân loại các đối tượng vào một tập hợp giá trị rời rạc. Thường, thuật toán này được áp dụng trong các bài toán phân loại nhị phân, tức là phân loại đối tượng vào hai nhóm khác nhau.

Thuật toán Logistic Regression hoạt động bằng cách tính toán giá trị dự đoán dựa trên các biến đầu vào (các biến độc lập). Kết quả dự đoán được biểu diễn dưới dạng xác suất rơi vào một nhóm cụ thể. Sau đó, giá trị dự đoán xác suất này được chuyển đổi thành các giá trị phân loại rời rạc (như 0 hoặc 1) thông qua một ngưỡng xác định trước.

Phương trình toán học của Logistic Regression cho bài toán phân loại nhị phân (hai lớp) có dạng như sau:

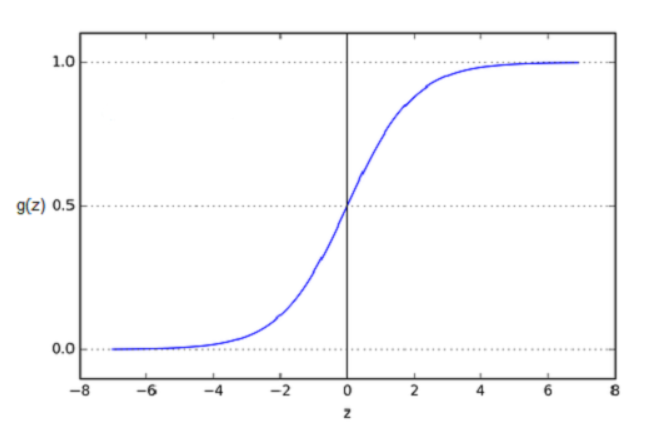
Trong đó:

- g(z) là xác suất dự đoán rằng mẫu thuộc lớp 1 dựa trên đầu vào x.

- w là vector trọng số (weight), là các tham số của mô hình cần được học từ dữ liệu huấn luyện.

- x là vector đặc trưng (feature) của mẫu.

Hàm sigmoid có dạng cong S và chuyển đổi giá trị của z từ khoảng -∞ đến ∞ thành một khoảng giá trị từ 0 đến 1, biểu thị xác suất.



Đồ thị hàm sigmoid

Để xây dựng một mô hình Logistic Regression, chúng ta cần sử dụng một tập dữ liệu huấn luyện đã được gán nhãn trước. Tập dữ liệu này bao gồm các biến đầu vào cùng với kết quả phân loại tương ứng. Quá trình huấn luyện mô hình tìm kiếm các tham số tối ưu để dự đoán kết quả phân loại cho các dữ liệu mới. Các tham số này thường được tối ưu thông qua các phương pháp như Gradient Descent.

Trong quá trình huấn luyện, mục tiêu là điều chỉnh vector trọng số w để tối ưu hóa việc dự đoán xác suất theo thực tế và giá trị dự đoán. Để làm điều này, thường sử dụng các thuật toán tối ưu hóa như gradient descent để tìm giá trị w tối ưu.

Đối với bài toán phân loại đa lớp, Logistic Regression có thể được mở rộng bằng cách sử dụng một phương pháp như One-vs-Rest (OvR) để phân loại vào nhiều lớp khác nhau.

2.1.2. Bộ phân loại Rigde (Ridge Classifier)

Ridge Classifier là một thuật toán phân loại quan trọng trong lĩnh vực Machine Learning. Thuật toán này thường được sử dụng để dự đoán hoặc phân loại các đối tượng vào các lớp khác nhau dựa trên thông tin từ các biến đầu vào. Ridge Classifier không chỉ có khả năng phân loại đa lớp mà còn giúp giảm thiểu quá khớp và cải thiện hiệu suất của mô hình phân loại.

Ridge Classifier hoạt động bằng cách tính toán giá trị dự đoán cho mỗi lớp dựa trên các biến đầu vào. Thay vì dự đoán một giá trị nhị phân như trong Logistic Regression, Ridge Classifier tính toán một vector chứa các giá trị xác suất rơi vào mỗi lớp cụ thể. Điều này giúp mô hình thể hiện sự không chắc chắn trong việc phân loại và tránh tình trạng quá tự tin.

Phương trình toán học của Ridge Classifier cho bài toán phân loại đa lớp có dạng tương tự như trong Logistic Regression. Đối với mỗi lớp, Ridge Classifier tính toán một hàm chuyển đổi xác suất, thường là hàm softmax, để biểu diễn khả năng rơi vào từng lớp.

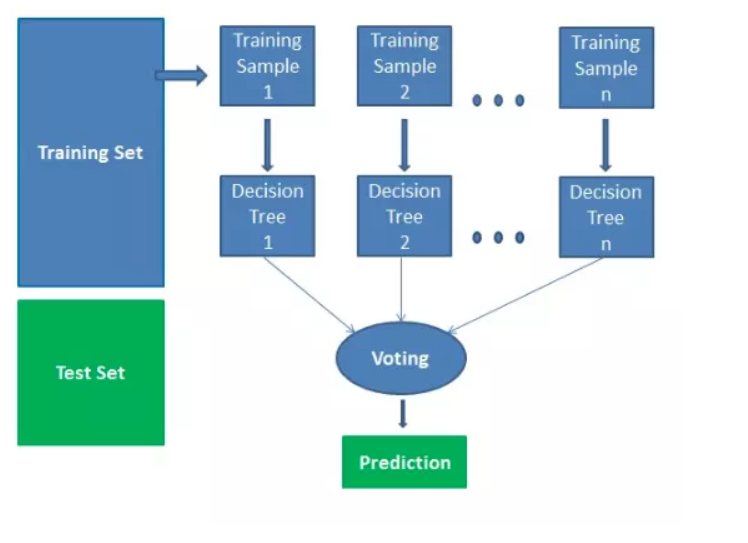
Quá trình huấn luyện mô hình Ridge Classifier liên quan đến tối ưu hóa các tham số để tối đa hóa khả năng dự đoán đúng trên tập dữ liệu huấn luyện. Điều này bao gồm việc điều chỉnh các trọng số của mô hình cho mỗi biến đầu vào và tham số chính quy (regularization parameters) để tránh quá khớp. Thuật toán thường sử dụng các phương pháp tối ưu hóa như Gradient Descent để tìm các giá trị tham số tối ưu.

2.1.3. Random Forest Classifier

Random Forest Classifier là một thuật toán phân loại trong lĩnh vực học máy. Nó là một biến thể của mô hình Random Forest, một phương pháp mạnh mẽ để giải quyết bài toán phân loại và hồi quy. Mô hình này dựa trên nguyên tắc của "rừng cây quyết định" (decision tree ensemble), trong đó nhiều cây quyết định được kết hợp để tạo thành một mô hình phân loại mạnh mẽ và ổn định.

Thuật toán hoạt động theo 4 bước:

1. Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.
2. Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi quyết định cây.
3. Bãy bỏ phiếu cho mỗi kết quả dự đoán.
4. Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.

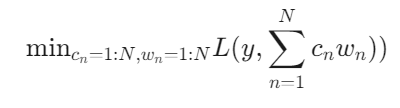


2.1.4. Gradient Boosting Classifier

Boosting (Mục tiêu là giảm bias - áp dụng cho các model có variance thấp và bị bias cao): Xây dựng một lượng lớn các model (thường là cùng loại). Mỗi model sau sẽ học cách sửa những errors của model trước (dữ liệu mà model trước dự đoán sai) -> tạo thành một chuỗi các model mà model sau sẽ tốt hơn model trước bởi trọng số được update qua mỗi model (cụ thể ở đây là trọng số của những dữ liệu dự đoán đúng sẽ không đổi, còn trọng số của những dữ liệu dự đoán sai sẽ được tăng thêm) . Chúng ta sẽ lấy kết quả của model cuối cùng trong chuỗi model này làm kết quả trả về.

Gradient Boosting Classifier là một mô hình phân loại trong lĩnh vực học máy, thuộc vào họ thuật toán boosting. Mô hình Gradient Boosting Classifier là một phương pháp mạnh mẽ để giải quyết các bài toán phân loại, nơi mục tiêu là xây dựng một mô hình dự đoán tốt hơn bằng cách tập trung vào việc cải thiện các điểm yếu của mô hình trước đó.

Gradient Boosting đều xây dựng thuật toán nhằm giải quyết bài toán tối ưu sau:



Trong đó :

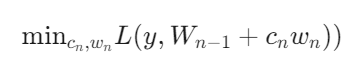
L : giá trị loss function

y : label

cn : confidence score của weak learner thứ n (hay còn gọi là trọng số)

wn : weak learner thứ n

Thay vì cố gằng quét tìm tất cả các giá trị cn, wn để tìm nghiệm tối ưu toàn cục - một công việc tốn nhiều thời gian và tài nguyên, chúng ta sẽ cố gắng tìm các giá trị nghiệm cục bộ sau khi thêm mỗi một mô hình mới vào chuỗi mô hình với mong muốn dần đi đến nghiệm toàn cục.



Trong đó: A black and white math symbols

Description automatically generated with medium confidence

2.1.5. So sánh

Logistic Regression (Hồi quy Logistic):

- Ưu điểm: Đơn giản, dễ hiểu, thường hoạt động tốt khi các lớp dữ liệu là tách biệt tốt và không có sự phức tạp lớn.

- Hạn chế: Khả năng tách biệt lớp không tốt, không thể xử lý tốt với dữ liệu phi tuyến.

Ridge Classifier

- Ưu điểm: Giúp kiểm soát overfitting, phù hợp với các bộ dữ liệu có nhiễu, là một biến thể của Logistic Regression nhưng có khả năng xử lý dữ liệu tốt hơn.

- Hạn chế: Vẫn có khả năng không phù hợp với các bài toán có độ phức tạp cao.

Random Forest Classifier:

- Ưu điểm: Mạnh mẽ, khả năng xử lý dữ liệu phi tuyến, có khả năng xử lý nhiễu, không yêu cầu chuẩn hóa dữ liệu.

- Hạn chế: Dễ bị overfitting trên dữ liệu nhỏ, khó để hiểu cách hoạt động nội tại của cây quyết định.

Gradient Boosting Classifier:

- Ưu điểm: Hiệu suất cao, khả năng xử lý dữ liệu tốt, tạo ra một mô hình mạnh bằng cách kết hợp nhiều cây quyết định yếu.

- Hạn chế: Dễ bị overfitting trên dữ liệu nhỏ, đòi hỏi nhiều thời gian và tài nguyên tính toán hơn.

**Chương IV: Kết quả**

