LOGISTIC REGRESSION

I Giới thiệu về hồi quy Logistic

Hồi quy Logistic được sử dụng trong phân loại nhị phân. Xét bài toán phân loại nhị phân: Với mỗi đối tượng $x \in X$ - biến quan sát, ta cần phân loại X vào 1 trong hai lớp $y \in \{0, 1\}$
Mô hình dự báo $h_\theta(x)$ được chọn như sau:

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

Hàm số: $g(s) = \frac{1}{1 + e^{-s}}$ được gọi là hàm Sigmoid, hoặc hàm Logistic
Một số tính chất của hàm Sigmoid:

1. $\lim_{s \to +\infty} g(s) = 1, \lim_{s \to -\infty} g(s) = 0$
2. $g(s) \in [0, 1]$
3. $g'(s) = g(s)(1 - g(s))$

Do y chỉ nhận hai giá trị 0 và 1 nên ta có mô hình hồi quy Logistic như sau:

$$P(y = 1|x) = h_\theta(x) \quad P(y = 0|x) = 1 - h_\theta(x)$$

trong đó θ là tham số của mô hình.
Khi đã xác định được tham số $\theta = \hat{\theta}$, ta sử dụng mô hình để phân loại như sau:
Đối tượng x thuộc về lớp 1 nếu: $P(y = 1|x) > P(y = 0|x) \Leftrightarrow h_{\hat{\theta}}(x) > \frac{1}{2} \Leftrightarrow \hat{\theta}^T x > 0$
Đối tượng x thuộc về lớp 0 nếu: $P(y = 1|x) < P(y = 0|x) \Leftrightarrow h_{\hat{\theta}}(x) < \frac{1}{2} \Leftrightarrow \hat{\theta}^T x < 0$

II Xác định tham số θ của mô hình hồi quy Logistic

1 Xây dựng hàm hợp lý cực đại

Xác suất của lớp y có thể được viết thành 1 phương trình dạng:

$$P(y|x) = (h_\theta(x))^y (1 - h_\theta(x))^{1-y}$$

Xét bộ dữ liệu training: $X = \{x_1, x_2, ..., x_N\} \subset \mathbb{R}^{d \times N}, y = \{y_1, y_2, ..., y_N\}$. Ta sử dụng phương pháp ước lượng hợp lý cực đại để xác định tham số $\hat{\theta}$ của mô hình:

$$\hat{\theta} = \arg\max_\theta P(y|x)$$

Giả sử các bộ dữ liệu trong Training set là độc lập, khi đó hàm hợp lý của dữ liệu với tham số θ là:

$$L(\theta) = P(y|X) = \prod_{i=1}^{N} P(y_i|x_i) = \prod_{i=1}^{N} (h_\theta(x_i))^{y_i}(1 - h_\theta(x_i))^{1-y_i}$$

Lấy log của hàm hợp lý, ta được:

$$l(\theta) = log(L(\theta)) = \sum_{i=1}^{N} [y_i log(h_\theta(x_i)) + (1 - y_i) log(1 - h_\theta(x_i))]$$

Do $h_\theta(x_i) \in [0, 1]$ và $y \in [0, 1]$ nên $l(\theta) < 0$. Vì vậy thay vì cực đại hóa $l(\theta)$ ta sẽ cực tiểu hóa hàm $-l(\theta)$

Để tăng tính ổn định khi giải bài toán cực trị $-l(\theta) \to \min$ và tránh overfitting, chúng ta bổ sung thêm phần hiệu chỉnh $\alpha R(\theta)$. Cụ thể ta giải bài toán cực tiểu hóa hàm mục tiêu sau:

$$J(\theta) = (-l(\theta) + \alpha R(\theta)) \to min$$

trong đó α là tham số hiệu chỉnh, $R(\theta)$ là phiếm hàm hiệu chỉnh.

$R(\theta)$ thường được chọn như sau:

$$R(\theta) = ||\theta||_2^2 = \sum_{i=0}^{d} \theta_i^2$$

Mô hình với hiệu chỉnh $R(\theta)$ như trên được gọi là mô hình Logistic với hiệu chỉnh L_2

Như vậy, chúng ta cần tìm:

$$\hat{\theta} = \arg\min_\theta \{-l(\theta) + \alpha R(\theta)\} = \arg\min_\theta \{\sum_{i=1}^{N} [y_i log(h_\theta(x_i)) + (1 - y_i) log(1 - h_\theta(x_i))] + \alpha R(\theta)\}$$

2 Phương pháp Gradient Descent

Giả sử từ khởi đầu θ^0 bất kì, hiện tại ta có xấp xỉ θ. Khi đó θ ở bước tiếp theo được cập nhật qua công thức:

$$\theta := \theta - \alpha \nabla J(\theta)$$

trong đó: α là hệ số học và $\nabla J(\theta) = \left(\frac{\partial J}{\partial \theta_0}, \frac{\partial J}{\partial \theta_1}, ..., \frac{\partial J}{\partial \theta_d} \right)$

Mỗi thành phần của tham số θ được cập nhật theo công thức:

$$\theta_j = \theta_j - \alpha \frac{\partial J}{\partial \theta_j}, \quad j = 0, 1, ..., d$$

$$= \theta_j - \alpha (h_\theta(x) - y) x_j$$

Giả sử tập dữ liệu Training có N cặp dữ liệu $\{(x_i, y_i)\}_{i=1}^N$, ta thực hiện cập nhật θ_j như sau:

$$\theta_j = \theta_j - \alpha (h_\theta(x_i) - y_I) x_{ij}, \quad j = 0, 1, ..., d$$

- Lần lượt thực hiện với $i = 1, 2, ..., N$
- Sau khi sử dụng hết N cặp dữ liệu cần xáo trộn lại thứ tự các cặp (x_i, y_i) và thực hiện lại từ đầu.

Phương pháp trên được gọi là phương pháp Gradient ngẫu nhiên - Stochastic Gradient Descent.

Áp dụng hiệu chỉnh L_2, ta được công thức cập nhật θ_j như sau:

$$\theta_j = \theta_j - \alpha(h_\theta(x_i) - y_i)x_{ij} - \alpha\theta_j, \quad j = 0, 1, ..., d$$

Thuật toán Stochastic Gradient Descent:

stochasticGradientDescent(X, y, α):

$[N, d] =$ **size**(X)

$\theta =$ **zeros**$(d, 1)$

do:

$\{k\} = Permutation(N)$

for $i_k = 1 \to N$ **do**:

for $j = 0 \to d$ **do**:

$\theta_j = \theta_j - \alpha(h_\theta(x_{i_k}) - y_{i_k})x_{i_k j} - \alpha\theta_j$

while (condition = **True**)

return θ

III Huấn luyện mô hình hồi quy Logistic bằng thư viện sklearn

+ Sử dụng thư viện:

from sklearn.linear_model **import** LogisticRegression

from sklearn.linear_metrics **import** accuracy_score

+ Huấn luyện mô hình:

log_regr = LogisticRegression()

log_regr.fit(Xbar, y_train)

+ Hiển thị hệ số của mô hình hồi quy:

log_regr.coef_

+ Chạy mô hình trên tập test:

y_pred = log_regr.predict(X_test)

+ Đánh giá mô hình bằng accuracy_score:

accuracy_score(y_test, y_pred)

IV Ví dụ

Cho dữ liệu tuyển sinh của một trường đại học (master) của Ấn độ trong tệp Admission_Predict.csv đính kèm. Các trường dữ liệu như sau:

- GRE (Graduate Record Exam) Scores (0..340): bảng điểm học tập đại học
- TOEFL Scores (0.. 120): Điểm tiếng Anh (toefl)
- University Rating (0.. 5): Điểm xếp loại đại học
- SOP (Statement of Purpose) Strength (0..5): Điểm bài viết tự giới thiệu
- LOR (Letter of Recommendation) Strength (0..5): Điểm cho thư giới thiệu
- Undergraduate GPA - CGPA (0..10): Điểm trung bình ĐH
- Research Experience (0 hoặc 1): kinh nghiệm nghiên cứu (chỉ 1 – có hoặc 0 – không)
- Chance of Admit (Số thực 0 .. 1): Khả năng được chọn

Giả thiết rằng 'Chance of Admit' ≥ 0.75 thì ứng viên tương ứng được chọn (thuộc class 1); ngược lại sẽ không được chọn (thuộc class 0).

Đọc dữ liệu và chọn ra 350 dòng đầu làm dữ liệu Training, còn lại là dữ liệu Test. Sử dụng mô hình hồi quy Logistic để xây dựng mô hình trên tập Training. Sau đó áp dụng dự báo trên tập Test và đánh giá độ chính xác của mô hình bằng Accuracy, Precision và Recall.

Các tham số của mô hình được xác định như sau:

Biến	Hệ số
Intercept	0.02977765
GRE	0.07072223
TOEFL scores	0.10958369
University Rating	0.50249676
SOP	0.51339916
LOR	0.66945905
Undergraduate	2.40227154
Research Experience	0.62220491

Đánh giá độ chính xác của mô hình:

$Accuracy = 0.88$

$Precision = 0.9444444444444444$

$Recall = 0.7727272727272727$