ĐẠI HỌC HUẾ
KHOA KỸ THUẬT VÀ CÔNG NGHỆ

BÁO CÁO ĐỒ ÁN
HỌC KÌ II, năm học 2023-2024

Học phần:
HỌC MÁY 2

Số phách **(**Do hội đồng chấm ghi thi)

Thừa Thiên Huế, tháng 05 năm 2024.

ĐẠI HỌC HUẾ
KHOA KỸ THUẬT VÀ CÔNG NGHỆ

BÁO CÁO ĐỒ ÁN
HỌC KÌ II, năm học 2023-2024

Học phần:
HỌC MÁY 1

Giảng viên hướng dẫn: T.s Nguyễn Đăng Trị
Lớp: Khoa học dữ liệu và Trí tuệ nhân tạo khóa 3.
Sinh viên thực hiện: Phạm Phước Bảo Tín_22E1020021
Trần Tùng Dương_22E1010001 .
(ký và ghi rõ họ tên)

Số phách **(**Do hội đồng chấm ghi thi)

Thừa Thiên Huế, tháng 05 năm 2024.

LỜI CẢM ƠN

DANH MỤC HÌNH ẢNH

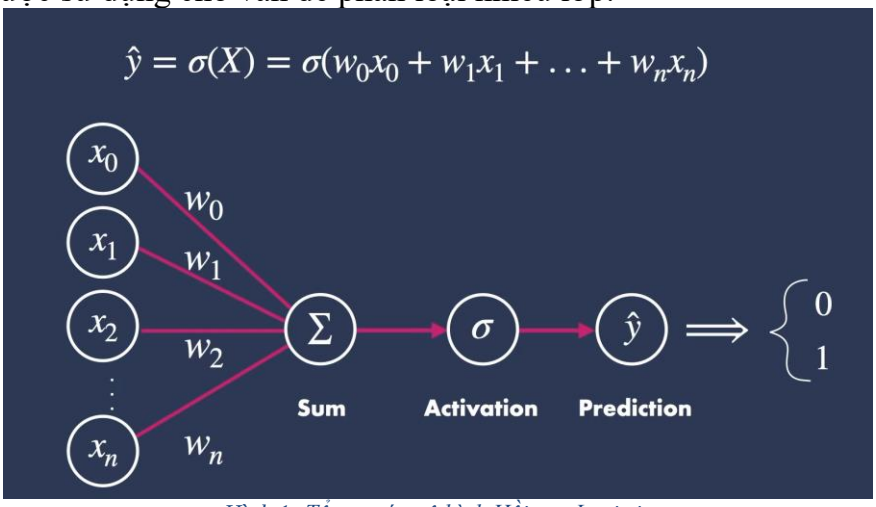
MỤC LỤC

MỞ ĐẦU

PHẦN 1: GIỚI THIỆU VÀ CƠ SỞ LÝ THUYẾT HỒI QUY LOGISTIC

1.1 Định nghĩa Hồi quy Logistics

Hồi quy logistic là một thuật toán học máy có giám sát được sử dụng rộng rãi cho các vấn đề liên quan đến phân loại. Ở dạng cơ bản, nó được sử dụng cho bài toán phân loại nhị phân chỉ có hai lớp để dự đoán. Tuy nhiên, với một chút mở rộng hồi quy logistic có thể dễ dàng được sử dụng cho vấn đề phân loại nhiều lớp.



Hình 1: Tổng quát mô hình Hồi quy Logistic

1.2 Mô hình toán

1.2.1 Hàm Sigmoid (Sigmoid Function)

Mô hình hồi quy logistic sử dụng hàm logistic để ép đầu ra của một phương trình tuyến tính có giá trị trong khoảng (0,1)

Công thức của hàm Sigmoid:

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

Trong đó:

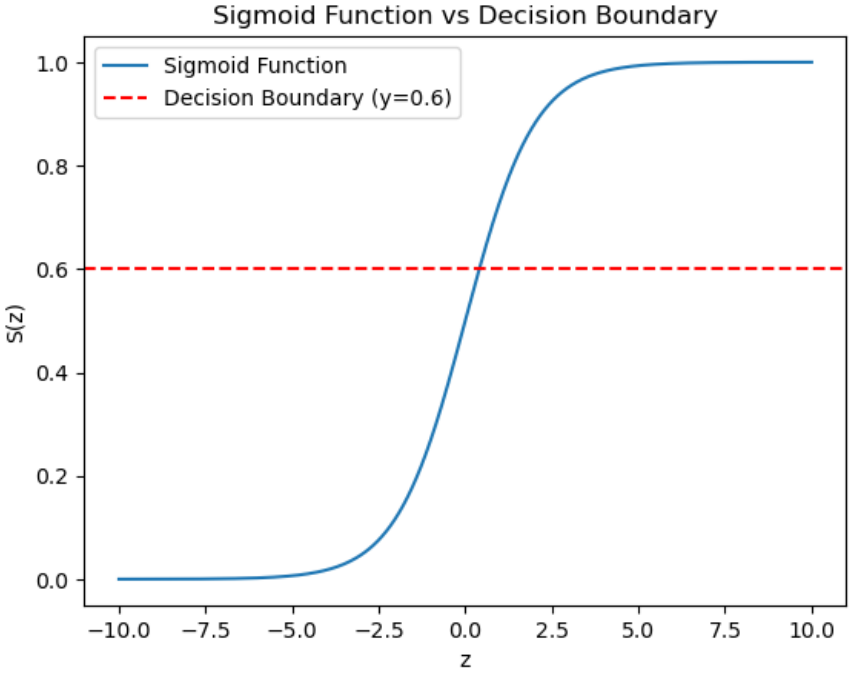
- $z = w^T x$: là phép tổ hợp tuyến tính đơn biến hoặc đa biến của biến đầu vào x với trọng số w.
- $\lim_{z\to\infty}(S(z) = 1, \lim_{z\to-\infty}(S(z) = 0$
- $S(z)$ là giá trị xác suất được dự đoán nằm trong khoảng từ 0 đến 1.

1.2.2 Ranh giới quyết định (Decision Boundary)

Hàm dự đoán trả về giá trị xác suất trong khoảng từ 0 đến 1. Để có thể phân loại các danh mục rời rạc (đỗ/trượt,cho vay/không cho vay,...), ta cần chọn giá trị ngưỡng để nếu xác suất lớn hơn giá trị này thì sẽ phân loại vào danh mục đó, còn thấp hơn thì phân loại vào danh mục còn lại đối với hồi quy Logistic nhị phân.

$$p \geq 0.6, class = 1$$
$$p < 0.6, class = 0$$

Ví dụ, nếu chọn giá trị ngưỡng là 0.6 và giá trị hàm dự đoán trả về 0.7 thì ta có thể phân loại điểm dữ liệu đó là *đỗ*. Nếu giá trị hàm dự đoán trả về 0.3 thì ta có thể phân loại điểm dữ liệu đó là *trượt*.



Hình 2: Sigmoid Funciton vs Decision Boundary

1.2.3 Hàm mất mát (Loss Funciton)

Hàm mất mát Logistic Regression**,** hay còn gọi là Entropy chéo hai lớp (Binary Cross Entropy), là một hàm số quan trọng trong học máy, được sử dụng để đánh giá mức độ sai lệch giữa dự đoán của mô hình và giá trị thực tế của dữ liệu trong bài toán phân loại nhị phân.

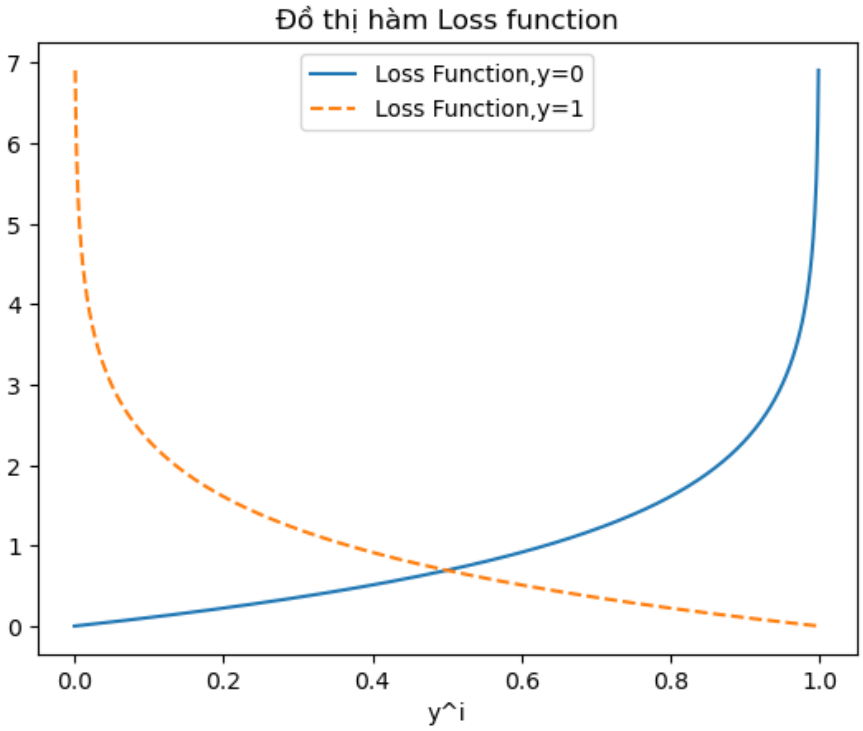
Mục tiêu chính của hàm mất mát Logistic Regression là tối ưu hóa để tăng hiệu suất của mô hình phân loại. Quá trình này được thực hiện bằng cách điều chỉnh các tham số của mô hình sao cho giá trị hàm mất mát được giảm thiểu, dẫn đến dự đoán chính xác hơn cho dữ liệu mới.

Thay vì sử dụng Mean Squared Error (MSE) sai số bình phương trung bình giữa giá trị được dự đoán và thực tế như trong hồi quy tuyến tính thì sử dụng hàm Cross-Entropy (hàm mất mát Log).

Với mỗi điểm dữ liệu $(x^{(i)}, y_i)$, công thức tổng quát của hàm mất mát:

$$L = -\big(y_i * log(\hat{y}_i) + (1 - y_i) * log(1 - \hat{y}_i)\big) \quad (2)$$

- Nếu $y_i = 1$ => $L = -log(\hat{y}_i)$
- Nếu $y_i = 0$ => $L = -\big(log(1 - \hat{y}_i)\big)$



Hình 3: Đồ thị hàm Loss Function

Nhận xét:

- Đối với $y_i = 0$, khi mô hình dự đoán $\hat{y}_i$ gần về 0, có nghĩa là giá trị dự đoán gần với giá trị thật thì giá trị hàm mất mát xấp xỉ bằng 0.
- Ngược lại, khi $y_i = 1$, khi mô hình dự đoán $\hat{y}_i$ gần về 0 thì giá trị hàm mất mát lúc này rất lớn và khi $\hat{y}_i$ gần về 1 tức là gần với giá trị thực, lúc này hàm mất mát có giá trị nhỏ xấp xỉ bằng 0.

Vậy với trên toàn bộ dữ liệu, hàm Loss function có công thức như sau:

$$L = -\frac{1}{n}\sum_{i=1}^{n}(y_i * \log \hat{y}_i + (1 - y_i) * \log(1 - \hat{y}_i)) \quad (3)$$

1.3 Ý nghĩa bài toán trong thực tế

Hồi quy logistic là một phương pháp thống kê được sử dụng để phân loại các đối tượng dựa trên một hoặc nhiều biến độc lập. Ứng dụng của hồi quy logistic rất đa dạng và được áp dụng trong nhiều lĩnh vực khác nhau. Dưới đây là một số ví dụ về ứng dụng của hồi quy logistic trong thực tế:

1.3.1 Y tế

- Chẩn đoán bệnh: Hồi quy logistic được sử dụng để dự đoán khả năng mắc bệnh của một bệnh nhân dựa trên các yếu tố nguy cơ như tuổi tác, chỉ số BMI, tiền sử bệnh lý, v.v. Ví dụ, dự đoán nguy cơ mắc bệnh tim dựa trên huyết áp, cholesterol và các yếu tố khác.
- Hiệu quả điều trị: Dự đoán khả năng thành công của một phương pháp điều trị dựa trên các đặc điểm của bệnh nhân và lịch sử y tế của họ.

1.3.2 Tài chính – Ngân hàng

- Dự đoán rủi ro tín dụng: Sử dụng hồi quy Logistic để dự đoán khả năng một khách hàng sẽ vỡ nợ dựa trên các thông tin tài chính và hành vi tiêu dùng của họ
- Phát hiện gian lận: Xác định các giao dịch có khả năng là gian lận dựa trên các mẫu giao dịch và hành vi của người dùng.

1.3.3 Marketing

- Phân loại khách hàng tiềm năng: Xác định khách hàng có khả năng mua sản phẩm hoặc dịch vụ dựa trên hành vi mua hàng trước đây, sở thích và các yếu tố khác.
- Dự đoán hành vi khách hàng: Dự đoán khả năng khách hàng sẽ hủy dịch vụ hoặc tham gia chương trình khuyến mãi dựa trên các dữ liệu lịch sử.

1.3.4 Giáo dục

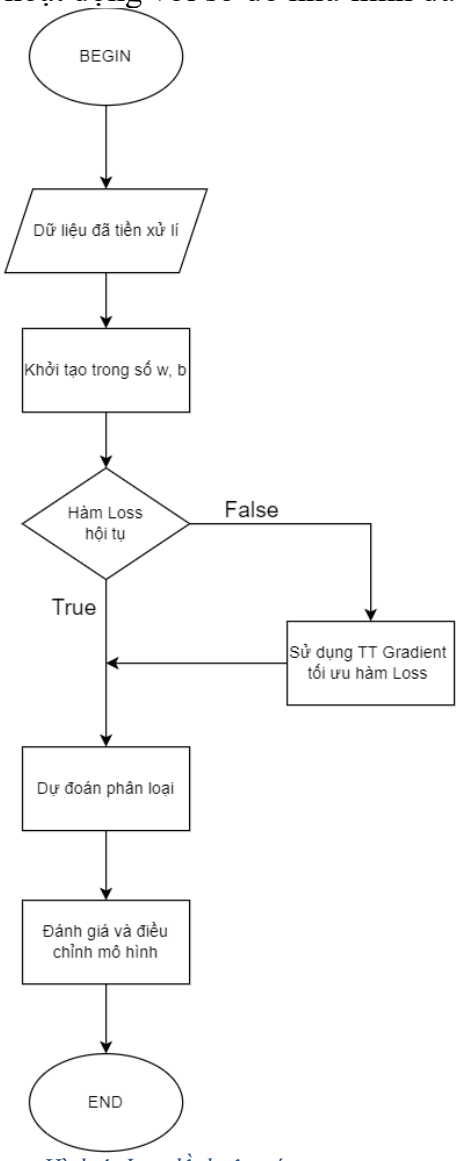
- Dự đoán kết quả học tập: Sử dụng hồi quy Logistic để dự đoán khả năng một học sinh sẽ hoàn thành khóa học dựa trên điểm số, mức độ tham gia và các yếu tố khác.
- Phân loại học sinh: Xác định học sinh cần hỗ trợ thêm dựa trên tình hình học tập và các yếu tố khác.

PHẦN 2: PHÂN TÍCH VÀ CHỨNG MINH THUẬT TOÁN

2.2 Phân tích thuật toán

2.2.1 Sơ đồ thuật toán

Mô hình hồi quy Logistic hoạt động với sơ đồ như hình dưới đây.



Hình 4: Lưu đồ thuật toán

2.2.2 Các bước thực hiện

1. Chuẩn bị các hàm:
   a. Hàm Sigmoid

b. Hàm chi phí
c. Hàm tối ưu
d. Hàm dự đoán

2. Khởi tạo và cập nhật trọng số
   a. Khởi tạo trọng số w,b ban đầu của các biến đầu vào.
   b. Chọn learning rate (η), chọn giá trị η quá lớn có thể dẫn đến việc mô hình không hội tụ và quá nhỏ có thể làm cho thuật toán mất thời gian để hội tụ.
   c. Áp dụng Gradient Descent để cập nhật trọng số.
3. Dự đoán và đánh giá hiệu quả mô hình
   a. Dự đoán: Sử dụng trọng số đã học được từ mô hình để dự đoán xác suất, phân loại nhãn dữ liệu mới
   b. Đánh giá mô hình: Đánh giá hiệu suất của mô hình bằng cách sử dụng các độ đo như accurary,recall, F1,... Tùy chỉnh các tham số (learning rate, số lần lặp,..) nếu cần để cải thiện hiệu suất của mô hình

2.2 Chứng minh cách cập nhật hệ số trong mô hình

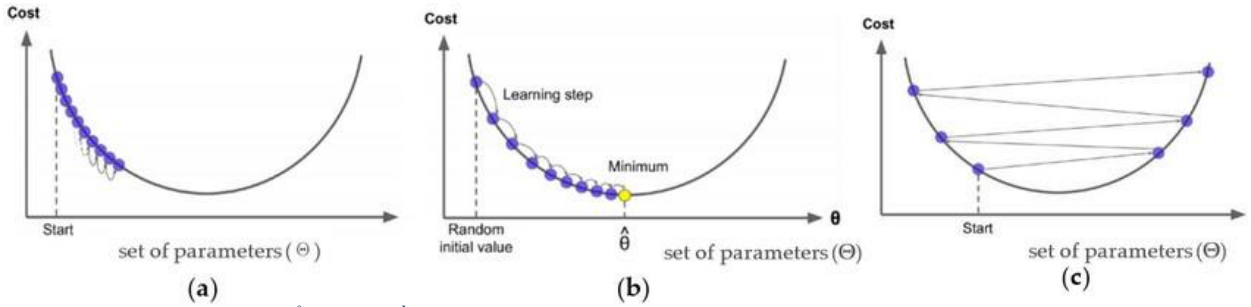
2.2.1 Sử dụng thuật toán Gradient Descent tối ưu hàm Loss function

Gradient Descent (GD) là thuật toán tìm tối ưu chung cho các hàm số. Ý tưởng chung của GD là điều chỉnh các tham số để lặp đi lặp lại thông qua mỗi dữ liệu huấn luyện để giảm thiểu hàm chi phí.

Để tối ưu hàm chi phí cần cập nhật các trọng số.

Nếu đạo hàm của hàm Loss tại $x_i : f'(x_i) < 0, x_i$ lúc này sẽ nằm phía bên trái điểm cực tiểu x_i^* hoặc $f'(x_i) > 0$ x_i sẽ nằm phía bên phải điểm cực tiểu x_i^*. Lúc này cần di chuyển gần hơn đến điểm x_i^* có nghĩa là di chuyển ngược hướng với đạo hàm.

$w_{n+1} = x_n - \eta * f'(w_n)$, trong đó η là tốc độ học của mô hình.



Hình 5: Tổng quan về thuật toán Gradient Descent

2.2.2 Đạo hàm riêng hàm Sigmoid

$$f(x_i) = \hat{y}_i = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2+\cdots+w_nx_n)}} \quad (4)$$

$$\frac{d\hat{y}_i}{dw_0} = \frac{d(1/(1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}))}{dw_0}$$

$$\frac{d\hat{y}_i}{dw_0} = \frac{e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}}{(1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)})^2}$$

$$\frac{d\hat{y}_i}{dw_0} = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}} * \frac{e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}}{1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}}$$

$$\frac{d\hat{y}_i}{dw_0} = \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}} * \left(1 - \frac{1}{1+e^{-(w_0+w_1x_1+w_2x_2+\ldots+w_nx_n)}}\right)$$

$$\frac{d\hat{y}_i}{dw_0} = \hat{y}_i(1-\hat{y}_i) \quad (5)$$

Tương tự:

$$\frac{d\hat{y}_i}{dw_1} = x_1\hat{y}_i(1-\hat{y}_i)$$

$$\frac{d\hat{y}_i}{dw_2} = x_2\hat{y}_i(1-\hat{y}_i)$$

$$\frac{d\hat{y}_i}{dw_n} = x_n\hat{y}_i(1-\hat{y}_i) \quad (6)$$

2.2.3 Đạo hàm riêng hàm Loss

Đạo hàm riêng hàm loss để cập nhật trọng số

$$L = -\left(y_i * log\hat{y}_i + (1-y_i) * log(1-\hat{y}_i)\right)$$

$$\frac{dL}{d\hat{y}_i} = -\left(\frac{y_i}{\hat{y}_i} - \frac{1-y_i}{1-\hat{y}_i}\right) \quad (7)$$

Áp dụng chain rule (nối chuỗi):

Cụ thể ta có hàm $f(x) = g(h(x))$, trong đó $\begin{cases} u = h(x) \text{ là hàm thành phần} \\ g(u) \text{ và } h(x) \text{ là các hàm khác nhau'} \end{cases}$

Khi đó, qui tắc chuỗi cho phép chúng ta tính đạo hàm của $f(x)$ theo x bằng cách nhân đạo hàm của $g(u)$ theo u với đạo hàm của $h(x)$ theo x.

$$\frac{d(f(x))}{dx} = \frac{d(g(u))}{du} * \frac{d(h(x)}{dx} \quad (8)$$

Áp dụng cho độ dời w_0 (bias) và trọng số w_1

w_0	w_1
$\frac{dL}{dw_0} = \frac{dL}{d\hat{y}_i} * \frac{d\hat{y}_i}{dw_0}$ $\frac{dL}{dw_0} = -\left(\frac{\hat{y}_i}{y_i} - \frac{1-y_i}{1-\hat{y}_i}\right) * \hat{y}_i(1-\hat{y}_i)$ $\frac{dL}{dw_0} = \hat{y}_i - y_i$ (9)	$\frac{dL}{dw_1} = \frac{dL}{d\hat{y}_i} * \frac{d\hat{y}_i}{dw_1}$ $\frac{dL}{dw_1} = -\left(\frac{y_i}{\hat{y}_i} - \frac{1-y_i}{1-\hat{y}_i}\right) * x_1^i\hat{y}_i(1-\hat{y}_i)$ $\frac{dL}{dw_1} = x_1^i(\hat{y}_i - y_i)$ (10)

Tổng quát trên toàn bộ dữ liệu:

$$\frac{dL}{dw_0} = \sum_{i=1}^{n}(\hat{y}_i - y_i), \text{tương tự}$$

$$\frac{dL}{dw_1} = \sum_{i=1}^{n} x_1^i(\hat{y}_i - y_i),$$

$$\frac{dL}{dw_n} = \sum_{i=1}^{n} x_n^i(\hat{y}_i - y_i) \quad (11)$$

Như vậy cập nhật nghiệm theo gradient descent được rút ngắn xuống thành:

$$\begin{cases} b = b - \eta * (\hat{y}_i - y_i) \\ w_n = w_n - \eta * x_i(\hat{y}_i - y_i) \end{cases}$$

PHẦN 3: ỨNG DỤNG VÀ MÔ PHỎNG MÔ HÌNH HỒI QUY LOGISTIC

3.1 Đề xuất ứng dụng

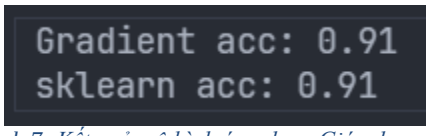
3.1.1 Ứng dụng mô hình phục vụ Giáo dục

Dự đoán kết quả kì thi dựa trên số giờ học với chất lượng học tương đương nhau.Tổng quan về dữ liệu:

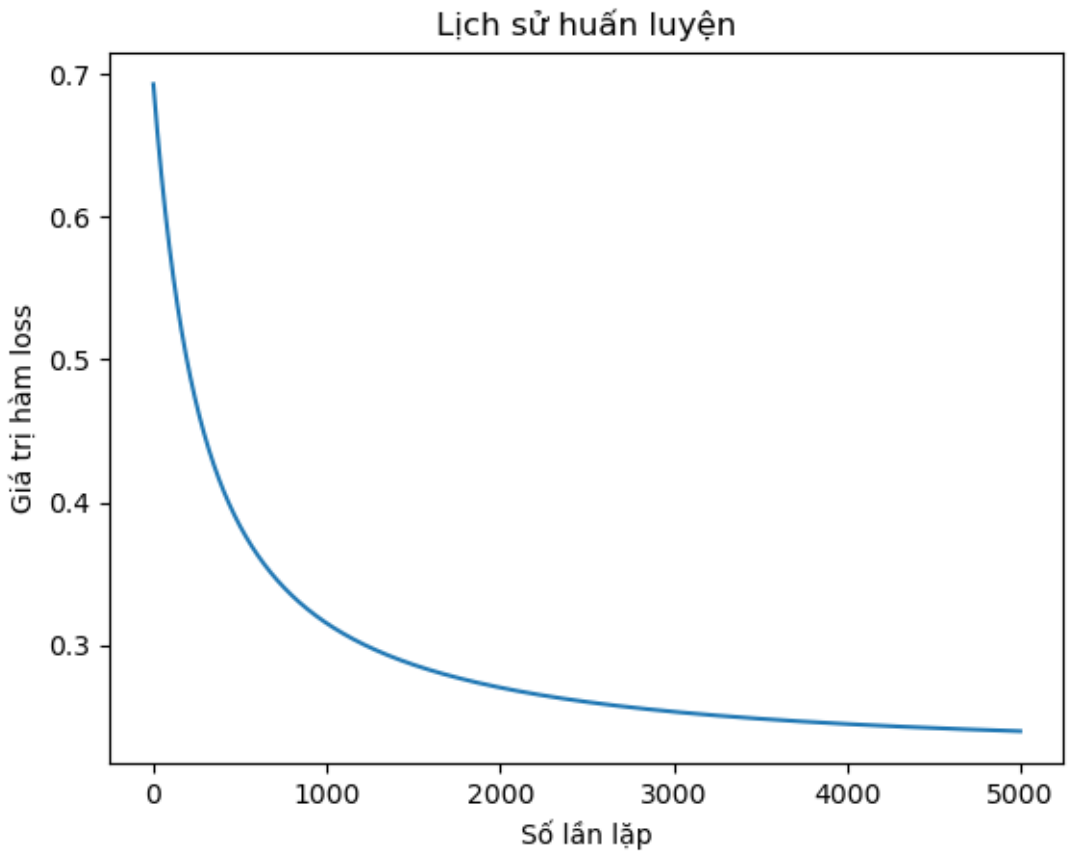
Hours_Study	Pass_Or_Fail
7	1
2	0
7	1
8	1
3	0
5	0
1	0
7	1
8	1
4	0
4	0
8	1
5	1
3	0
8	1
3	0
1	0
3	0

Hình 6: Tổng quan dữ liệu ứng dụng Giáo dục

Kết quả mô hình sử dụng Gradient Descent triển khai và thư viện Sklearn

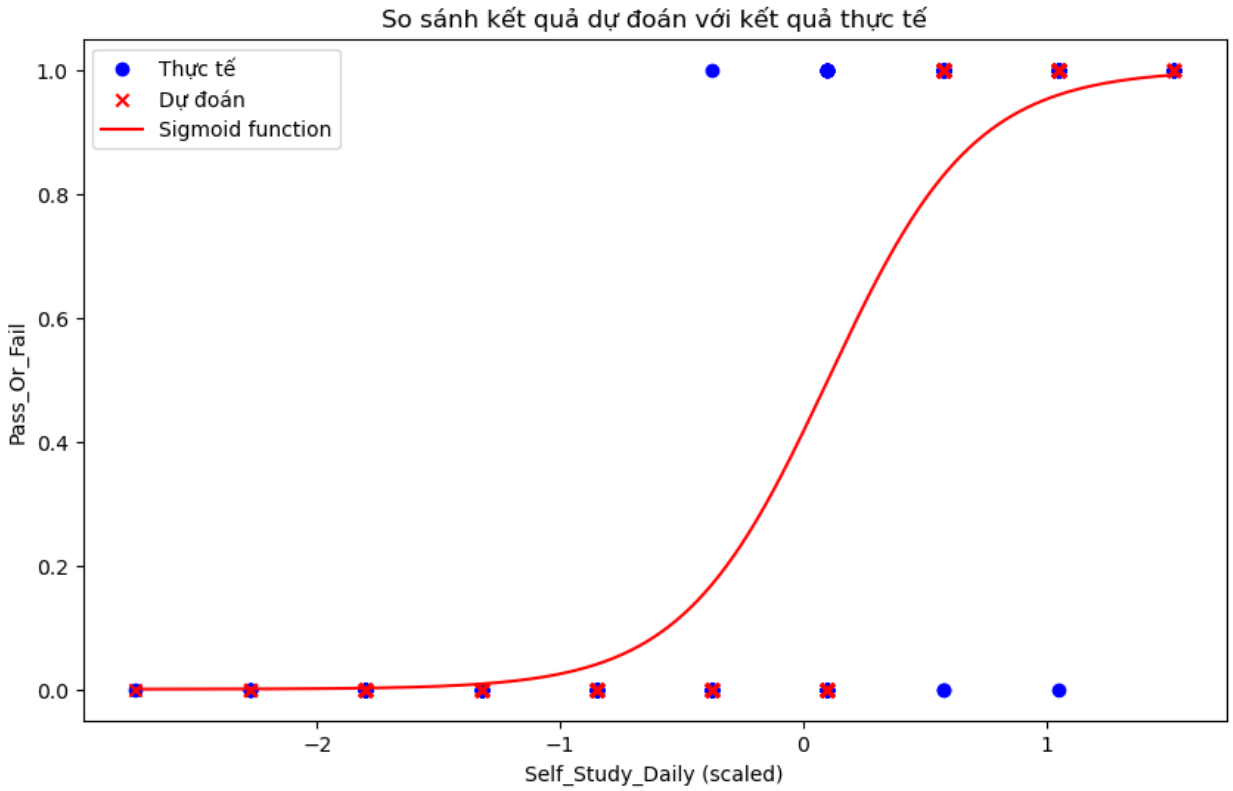


Hình 7: Kết quả mô hình ứng dụng Giáo dục



Hình 8: Lịch sử huấn luyện sử dụng Gradient Descent

Dưới đây mô phỏng kết dự đoán và kết quả từ mô hình triển khai sử dụng Gradient Descent.



Hình 9: Mô phỏng kết quả

3.1.2 Ứng dụng mô hình phục vụ Y tế

Dự đoán bệnh nhân mắc bệnh tim dựa vào các chỉ số như: Tuổi, Giới tính, Nhịp tim tối đa đạt được, đau thắt ngực do gắng sức, loại đau ngực,... Dưới đây là tổng quan về dữ liệu:

age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	170	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1
54	1	0	140	239	0	1	160	0	1.2	2	0	2	1
48	0	2	130	275	0	1	139	0	0.2	2	0	2	1
49	1	1	130	266	0	1	171	0	0.6	2	0	2	1
64	1	3	110	211	0	0	144	1	1.8	1	0	2	1
58	0	3	150	283	1	0	162	0	1	2	0	2	1
50	0	2	120	219	0	1	158	0	1.6	1	0	2	1

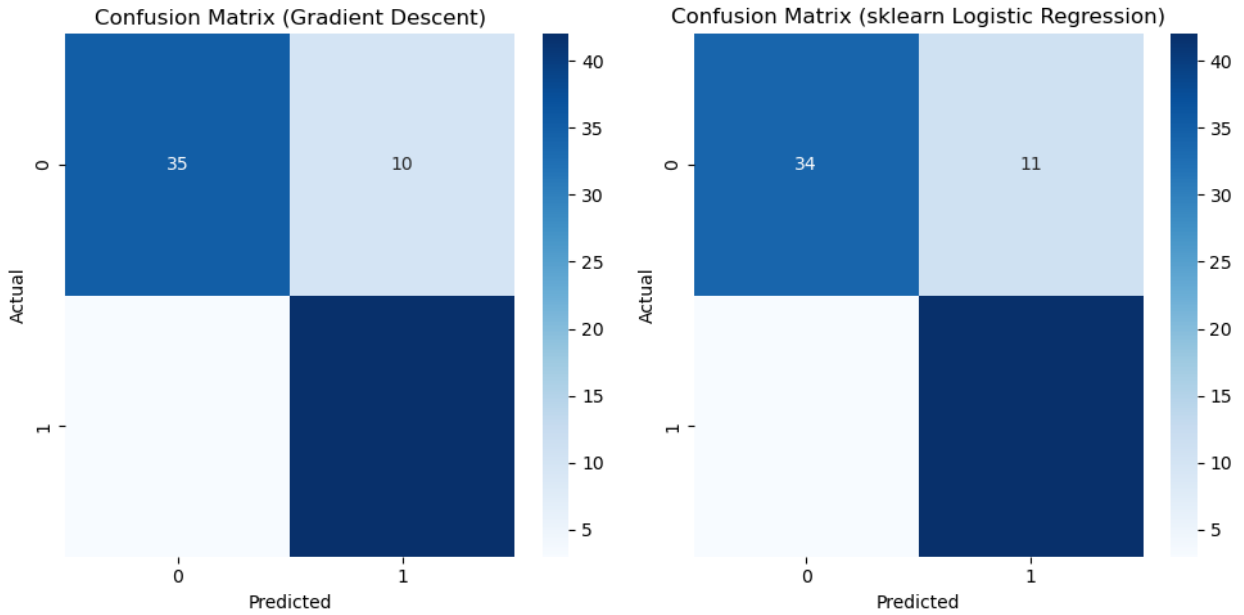
Hình 10: Tổng quan dữ liệu ứng dụng Y tế

Kết quả sau khi thực hiện mô hình nhóm em đã triển khai và sử dụng thư viện Sklearn:

```
accurary score Gradient: 0.84
accurary score sklearn: 0.84
```

Hình 11: Kết quả mô hình ứng dụng Y tế

Ma trận nhầm lẫn (Confusion Matrix) của mô hình hồi quy Logistic thực hiện theo 2 cách trên:



Hình 12: Ma trận nhầm lẫn

3.2 Mô phỏng thuật toán

Mã nguồn mô phỏng thuật toán sử dụng Gradient Descent để triển khai mô hình và sử dụng thư viện Sklearn: Click here.

TÀI LIỆU THAM KHẢO

Chúng em sẽ cập nhật sau khi hoàn thành đồ án.