Chương 18

Mạng sâu lan truyền thẳng (thực hành)

Trong chương này, ta thực hiện những ví dụ cơ bản về thực hành học sâu trong các bài toán hồi quy tuyến tính, hồi quy tuyến tính, và phân loại đa lớp. Việc hỗ trợ tính toán trên ma trận được cung cấp bởi thư viện `numpy`. Các ví dụ này cũng khá đơn giản đến mức nhiều phép toán có thể thực hiện thủ công.

Các mạng nơron đa lớp trong học sâu được đề cập trong chương này chủ yếu được thiết kế như sau: đối với mỗi nút của lớp ẩn, ta sẽ sử dụng phép biến đổi affine đối với các nút ở lớp trước, rồi tác động cùng một hàm kích hoạt tại các nút trên lớp đó. Lớp đầu ra là đơn trị, vẫn tiến hành phép biến đổi affine đối với các nút ẩn ở lớp trước, còn hàm kích hoạt sẽ tùy thuộc vào định dạng đầu ra của bài toán. Trong trường hợp hồi quy tuyến tính, sẽ không có hàm kích hoạt trên lớp đầu ra.

Phương pháp thực hiện học sâu được triển khai là thuật toán lan truyền ngược. Để tiến hành lan truyền ngược, trước hết ta cần khởi tạo tham số của mô hình, tiến hành bước lan truyền xuôi để tính tất cả giá trị của các nút tại các lớp ẩn và lớp đầu ra, và giá trị của hàm mất mát. Sau đó mới áp dụng thuật toán lan truyền ngược, tính gradient của hàm mất mát theo ma trận giá trị tại các lớp và theo các tham số. Cuối cùng sau khi tính được gradient của hàm mất mát theo các tham số, ta có thể áp dụng các thuật toán học, chẳng hạn phương pháp hướng giảm, để cập nhật lại tham số trong bài toán cực tiểu hóa hàm mất mát.

Quá trình lan truyền xuôi được tóm tắt trong Thuật toán 6.3, và thủ tục lan truyền ngược được mô tả trong Thuật toán 6.4. Mục tiêu của chương này là minh họa cụ thể các bước của thuật toán trong từng ví dụ cụ thể.

Các bài toán được đề cập trong chương này thuộc dạng học có giám sát, với

dữ liệu huấn luyện được tóm tắt dưới dạng bảng

$\boldsymbol{x}^{(i)\top}$	x_1	x_2	...	x_n	y
$\boldsymbol{x}^{(1)\top}$	$x_1^{(1)}$	$x_1^{(1)}$	...	$x_n^{(1)}$	y_1
$\boldsymbol{x}^{(2)\top}$	$x_1^{(2)}$	$x_1^{(2)}$	...	$x_n^{(2)}$	y_2
...	...	...	...	...	...
$\boldsymbol{x}^{(m)\top}$	$x_1^{(m)}$	$x_1^{(m)}$	...	$x_n^{(m)}$	y_m

trong đó $\boldsymbol{x} = [x_1, \dots, x_n]^\top$ là các đặc trưng của đầu vào, y là nhãn gán cho đầu vào; $\boldsymbol{x}^{(i)} = \left[x_1^{(i)}, \dots, x_n^{(i)}\right]^\top$ là đầu vào thứ i, và y_i là nhãn của đầu vào thứ i đó.

Trong chương này, ký hiệu $\boldsymbol{X}$ là ma trận cỡ $m \times n$ chứa tất cả các giá trị tại lớp đầu vào, mỗi giá trị được lưu trên một hàng, và $\boldsymbol{Y}$ là ma trận cỡ $m \times 1$ lưu các nhãn đã gán tương ứng:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}^{(i)\top} \\ \boldsymbol{x}^{(2)\top} \\ \vdots \\ \boldsymbol{x}^{(m)\top} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_1^{(1)} & \dots & x_n^{(1)} \\ x_1^{(2)} & x_1^{(2)} & \dots & x_n^{(2)} \\ & \cdots\cdots\cdots & & \\ x_1^{(m)} & x_1^{(m)} & \dots & x_n^{(m)} \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}. \tag{18.1}$$

Tại lớp ẩn thứ ℓ với độ rộng p, ký hiệu $\boldsymbol{z}^\ell = \left[z_1^\ell, \dots, z_p^\ell\right]^\top$ là véctơ giá trị trước kích hoạt, $\boldsymbol{Z}^\ell$ là ma trận chứa tất cả các véctơ giá trị trước kích hoạt; $\boldsymbol{h}^\ell = \left[h_1^\ell, \dots, h_p^\ell\right]^\top$ là véctơ giá trị kích hoạt, $\boldsymbol{H}^\ell$ lưu tất cả các véctơ giá trị kích hoạt được thực hiện tương ứng trên các véctơ giá trị trước kích hoạt.

Tại lớp đầu ra, $\widehat{\boldsymbol{Y}} = \left[\widehat{y}_1, \dots, \widehat{y}_m\right]^\top$ lưu các giá trị dự đoán cho các đầu vào tương ứng.

Đầu vào trực tiếp của $\boldsymbol{z}^\ell$ là $\boldsymbol{h}^{\ell-1}$ có độ rộng n, với các tham số là ma trận trọng số $\boldsymbol{W}^\ell$ cỡ $n \times p$, và véctơ các độ dời $\boldsymbol{b}^\ell$ có số chiều bằng p.

$$\boldsymbol{z}^{(\ell)} = \boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1)} + \boldsymbol{b}^{(\ell)}. \tag{18.2}$$

Ứng với đầu vào thứ i, véctơ trước kích hoạt nhận giá trị

$$\boldsymbol{z}^{(\ell,i)} = \boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1,i)} + \boldsymbol{b}^{(\ell)}. \tag{18.3}$$

Khi viết dưới dạng ma trận

$$\left[\boldsymbol{z}^{(\ell,1)}, \dots, \boldsymbol{z}^{(\ell,m)}\right] = \left[\boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1,1)} + \boldsymbol{b}^{(\ell)}, \dots, \boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1,m)} + \boldsymbol{b}^{(\ell)}\right] \tag{18.4}$$

$$= \left[\boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1,1)}, \dots, \boldsymbol{W}^{(\ell)\top} \boldsymbol{h}^{(\ell-1,m)}\right] + \boldsymbol{b}^{(\ell)} \tag{18.5}$$

$$= \boldsymbol{W}^{(\ell)^\top} \left[\boldsymbol{h}^{(\ell-1,1)}, \ldots, \boldsymbol{h}^{(\ell-1,m)}\right] + \boldsymbol{b}^{(\ell)}, \tag{18.6}$$

ở đây trong đẳng thức thứ hai và thứ ba, phép cộng ma trận với véctơ được ngầm hiểu là cộng từng hàng của ma trận với véctơ đó (đã được nhắc đến trong Mục 2.1). Phương trình trên có thể viết dưới dạng ma trận

$$\boldsymbol{Z}^{(\ell)^\top} = \boldsymbol{W}^{(\ell)^\top} \boldsymbol{H}^{(\ell-1)^\top} + \boldsymbol{b}^{(\ell)}. \tag{18.7}$$

Lấy chuyển vị hai vế, ta được

$$\boldsymbol{Z}^{(\ell)} = \boldsymbol{H}^{(\ell-1)} \boldsymbol{W}^{(\ell)} + \boldsymbol{b}^{(\ell)^\top} \tag{18.8}$$

Đầu vào trực tiếp của lớp ẩn $\boldsymbol{h}^{(\ell)}$ là véctơ trước kích hoạt $\boldsymbol{h}^{(\ell)}$, được thực hiện bởi hàm kích hoạt f trên từng phần tử:

$$h_j^{(\ell)} = f\left(z_j^{(\ell)}\right). \tag{18.9}$$

Ta viết lại phương trình trên dưới dạng véctơ

$$\boldsymbol{h}^{(\ell)} = \begin{bmatrix} h_1^{(\ell)} \\ \vdots \\ h_p^{(\ell)} \end{bmatrix} = \begin{bmatrix} f\left(z_1^{(\ell)}\right) \\ \vdots \\ f\left(z_p^{(\ell)}\right) \end{bmatrix} = f\left(\begin{bmatrix} z_1^{(\ell)} \\ \vdots \\ z_p^{(\ell)} \end{bmatrix}\right) = f\left(\boldsymbol{z}^{(\ell)}\right). \tag{18.10}$$

Ứng với đầu vào thứ i, lớp ẩn nhận giá trị

$$\boldsymbol{h}^{(\ell,i)} = f\left(\boldsymbol{z}^{(\ell,i)}\right). \tag{18.11}$$

Ma trận chứa các véctơ giá trị kích hoạt (véctơ hàng) tại lớp ẩn ứng với các đầu vào là

$$\boldsymbol{H}^{(\ell)} = \begin{bmatrix} \boldsymbol{h}^{(\ell,1)^\top} \\ \vdots \\ \boldsymbol{h}^{(\ell,m)^\top} \end{bmatrix} = \begin{bmatrix} f\left(\boldsymbol{z}^{(\ell,1)^\top}\right) \\ \vdots \\ f\left(\boldsymbol{z}^{(\ell,m)^\top}\right) \end{bmatrix} = f\left(\begin{bmatrix} \boldsymbol{z}^{(\ell,1)^\top} \\ \vdots \\ \boldsymbol{z}^{(\ell,m)^\top} \end{bmatrix}\right) = f\left(\boldsymbol{Z}^{(\ell)}\right). \tag{18.12}$$

Ta đã xây dựng xong các công thức cho quá trình lan truyền xuôi. Giai đoạn lan truyền ngược cần một số kết quả. Kết quả thứ nhất, nếu $\boldsymbol{Z}$ là ma trận trước kích hoạt được cung cấp bởi ma trận $\boldsymbol{H}$ (có thể là ma trận đầu vào hoặc ma trận kích hoạt ở lớp trước) và được điều khiển bởi ma trận trọng số $\boldsymbol{W}$ và véctơ độ dời $\boldsymbol{b}$, tức là

$$\boldsymbol{Z} = \boldsymbol{H}\boldsymbol{W} + \boldsymbol{b}^\top, \tag{18.13}$$

thì ta có thể tính được gradient của vô hướng J theo mỗi $\boldsymbol{H}$, $\boldsymbol{W}$ và $\boldsymbol{b}$ thông qua gradient theo trung gian $\boldsymbol{Z}$:

$$\nabla_{\boldsymbol{H}} J = (\nabla_{\boldsymbol{Z}} J)\, \boldsymbol{W}^\top \tag{18.14}$$

$$\nabla_{\boldsymbol{W}} J = \boldsymbol{H}^\top (\nabla_{\boldsymbol{Z}} J) \tag{18.15}$$

$$\nabla_{\boldsymbol{b}} J = (\nabla_{\boldsymbol{Z}} J)^\top \mathbf{1}_{m\times 1}. \tag{18.16}$$

Các công thức (18.14), (18.15) có thể dẫn xuất từ kết luận trong Mục 6.5.6, chỉ khác đôi chút về ký hiệu.

Kết quả thứ hai, với ma trận trước kích hoạt $\boldsymbol{Z}$ là ma trận, f là hàm kích hoạt tác động lên từng phần tử của $\boldsymbol{Z}$ để được ma trận kích hoạt

$$\boldsymbol{H} = f(\boldsymbol{Z}), \tag{18.17}$$

ta có công thức tính gradient của vô hướng J theo $\boldsymbol{Z}$ thông qua gradient trung gian theo $\boldsymbol{H}$:

$$\nabla_{\boldsymbol{Z}} J = \nabla_{\boldsymbol{H}} J \odot f'(\boldsymbol{Z}), \tag{18.18}$$

trong đó f' là đạo hàm của f, là hàm tác động lên từng phần tử của $\boldsymbol{Z}$, phép nhân $\odot$ được thực hiện tương ứng theo vị trí từng phần tử của hai ma trận cùng cỡ (cũng đã được nhắc đến trong Mục 2.1).

Để tiến hành một lần lan truyền ngược, có thể ta cần sử dụng nhiều lần các công thức (18.14), (18.15), (18.16) và (18.18) tùy thuộc vào độ sâu của mô hình.

Cuối cùng, sau khi thực hiện xong lan truyền ngược, ta cập nhật lại các tham số theo các phương pháp tối ưu dựa trên gradient, nhằm tối ưu hàm mục tiêu, chẳng hạn phương pháp hướng giảm để cực tiểu hóa hàm chi phí $J(\boldsymbol{\theta})$ với tốc độ học η:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}). \tag{18.19}$$

Quá trình ba giai đoạn lan truyền xuôi – lan truyền ngược – cập nhật tham số có thể cần lặp lại nhiều lần để đảm bảo điều kiện dừng, thường ta mong muốn gradient đủ nhỏ:

$$\left\|\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})\right\| < \varepsilon. \tag{18.20}$$

18.1 Hồi quy logistic

Trong bài toán hồi quy logistic, hàm mất mát thường xét là entropy chéo

$$J = J\left(\boldsymbol{W}^{(1)}, \boldsymbol{b}^{(1)}, \ldots, \boldsymbol{W}^{(\ell)}, \boldsymbol{b}^{(\ell)}\right) = \frac{1}{m}\sum_{i=1}^{m}\left(-y_i \log \widehat{y}_i - (1-y_i)\log\left(1-\widehat{y}_i\right)\right). \tag{18.21}$$

Các hạng tử trong tổng trên có thể tổ hợp lại trong biểu thức dạng véctơ $-\boldsymbol{Y}\log\widehat{\boldsymbol{Y}} - (1-\boldsymbol{Y})\log\left(1-\widehat{\boldsymbol{Y}}\right)$. Từ đó suy ra J là trung bình cộng của véctơ này. Ngoài ra lưu ý rằng y_i chỉ nhận một trong hai giá trị 0 hoặc 1, nên trong tổng trên, mỗi hạng tử có hai số hạng, luôn có một trong hai số hạng bằng 0. Ta có

$$J = \text{mean}\left(-\boldsymbol{Y}\log\widehat{\boldsymbol{Y}} - (1-\boldsymbol{Y})\log\left(1-\widehat{\boldsymbol{Y}}\right)\right). \tag{18.22}$$

Để phục vụ cho thủ tục lan truyền ngược được thực hiện sau này, ta cần xác định gradient của J theo $\widehat{\boldsymbol{Y}}$. Từ (18.21) ta xác định được đạo hàm riêng của J theo từng $\widehat{y}_i$:

$$\frac{\partial J}{\partial \widehat{y}_i} = \frac{1}{m}\left(-y_i\frac{1}{\widehat{y}_i} - (1-y_i)\frac{1}{1-\widehat{y}_i}(-1)\right) = \frac{1}{4}\left(-\frac{y_i}{\widehat{y}_i} + \frac{1-y_i}{1-\widehat{y}_i}\right). \tag{18.23}$$

Gradient của J theo $\widehat{\boldsymbol{Y}}$ là véctơ cột chứa các thành phần đạo hàm riêng ở trên. Ta cũng có thể biểu diễn gradient này dưới dạng véctơ, trong đó các phép toán được thực hiện theo nghĩa tương ứng từng phần tử:

$$\nabla_{\widehat{\boldsymbol{Y}}} J = \frac{1}{m}\begin{bmatrix} -\dfrac{y_1}{\widehat{y}_1} + \dfrac{1-y_1}{1-\widehat{y}_1} \\ \vdots \\ -\dfrac{y_4}{\widehat{y}_m} + \dfrac{1-y_m}{1-\widehat{y}_m} \end{bmatrix} = \frac{1}{m}\left(-\frac{\boldsymbol{Y}}{\widehat{\boldsymbol{Y}}} + \frac{1-\boldsymbol{Y}}{1-\widehat{\boldsymbol{Y}}}\right). \tag{18.24}$$

Một dạng hàm mất mát khác là sai số bình phương trung bình:

$$J = J\left(\boldsymbol{W}^{(1)}, \boldsymbol{b}^{(1)}, \ldots, \boldsymbol{W}^{(\ell)}, \boldsymbol{b}^{(\ell)}\right) = \frac{1}{m}\sum_{i=1}^{m}\left(\widehat{y}_i - y_i\right)^2 = \frac{1}{m}\left\|\widehat{\boldsymbol{Y}} - \boldsymbol{Y}\right\|_2^2. \tag{18.25}$$

Đạo hàm riêng của J theo $\widehat{y}_i$ là

$$\frac{\partial J}{\partial \widehat{y}_i} = \frac{2}{m}\left(\widehat{y}_i - y_i\right), \tag{18.26}$$

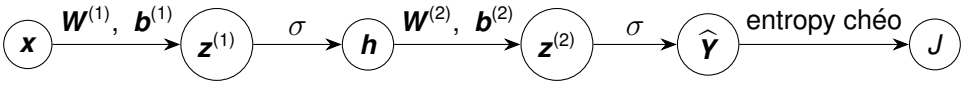
và ta suy ra được gradient của J theo $\widehat{\boldsymbol{Y}}$:

$$\nabla_{\widehat{\boldsymbol{Y}}} J = \frac{2}{m}\begin{bmatrix} \widehat{y}_1 - y_1 \\ \vdots \\ \widehat{y}_m - y_m \end{bmatrix} = \frac{2}{m}\left(\widehat{\boldsymbol{Y}} - \boldsymbol{Y}\right). \tag{18.27}$$

Ta đã đủ kiến thức để có thể triển khai một ví dụ đơn giản về mô hình hồi quy logistic bằng học sâu. Xét bài toán phân loại nhị phân với tập dữ liệu huấn luyện được cho bởi bảng sau.

$\boldsymbol{x}^{(i)\top}$	x_1	x_2	y
$\boldsymbol{x}^{(1)\top}$	1	2	1
$\boldsymbol{x}^{(2)\top}$	2	1	0
$\boldsymbol{x}^{(3)\top}$	3	1	0
$\boldsymbol{x}^{(4)\top}$	4	3	1

Mạng nơron chỉ bao gồm một lớp ẩn có ba nút. Các lớp ẩn và lớp đầu ra được trang bị hàm kích hoạt sigmoid. Hàm mất mát được xác định bởi entropy chéo.



Các bước lan truyền xuôi tính giá trị tại các lớp ẩn, lớp đầu ra và hàm mục tiêu theo dữ liệu đầu vào theo giá trị hiện tại của tham số như sau:

1. Khởi tạo tham số cho lớp ẩn và lớp đầu ra:

$$\boldsymbol{W}^{(1)} = \begin{bmatrix} 1 & 0 & -2 \\ -1 & 3 & 2 \end{bmatrix}, \ \boldsymbol{b}^{(1)} = \begin{bmatrix} 4 \\ 5 \\ -3 \end{bmatrix}; \quad \boldsymbol{W}^{(2)} = \begin{bmatrix} 2 \\ 0 \\ -3 \end{bmatrix}, \ \boldsymbol{b}^{(2)} = 1. \tag{18.28}$$

2. Ma trận giá trị trước kích hoạt của lớp ẩn:

$$\boldsymbol{Z}^{(1)} = \boldsymbol{X}\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)\top} \tag{18.29}$$

$$= \begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 1 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 & -2 \\ -1 & 3 & 2 \end{bmatrix} + \begin{bmatrix} 4 & 5 & -3 \end{bmatrix} \tag{18.30}$$

$$= \begin{bmatrix} -1 & 6 & 2 \\ 1 & 3 & -2 \\ 2 & 3 & -4 \\ 1 & 9 & -2 \end{bmatrix} + \begin{bmatrix} 4 & 5 & -3 \end{bmatrix} = \begin{bmatrix} 3 & 11 & -1 \\ 5 & 8 & -5 \\ 6 & 8 & -7 \\ 5 & 14 & -5 \end{bmatrix} \tag{18.31}$$

3. Ma trận giá trị kích hoạt tại lớp ẩn:

$$\boldsymbol{H} = \sigma\left(\boldsymbol{Z}^{(1)}\right) = \begin{bmatrix} 0.952574 & 0.999983 & 0.268941 \\ 0.993307 & 0.999665 & 6.69285 \times 10^{-3} \\ 0.997527 & 0.999665 & 9.11051 \times 10^{-4} \\ 0.993307 & 0.999999 & 6.69285 \times 10^{-3} \end{bmatrix}. \tag{18.32}$$

4. Ma trận giá trị trước kích hoạt của lớp đầu ra:

$$\boldsymbol{Z}^{(2)} = \boldsymbol{H}\boldsymbol{W}^{(2)} + \boldsymbol{b}^{(2)\top} = \begin{bmatrix} 2.09832 \\ 2.96654 \\ 2.99232 \\ 2.96654 \end{bmatrix} \tag{18.33}$$

5. Ma trận (lúc này là véctơ cột) giá trị kích hoạt tại lớp đầu ra:

$$\widehat{\boldsymbol{Y}} = \sigma\left(\boldsymbol{Z}^{(2)}\right) = \begin{bmatrix} 0.890740 \\ 0.951039 \\ 0.952226 \\ 0.951039 \end{bmatrix} \tag{18.34}$$

6. Hàm mất mát entropy chéo

$$J = \text{mean}\left(-\boldsymbol{Y}\log\widehat{\boldsymbol{Y}} - (1-\boldsymbol{Y})\log\left(1-\widehat{\boldsymbol{Y}}\right)\right) \tag{18.35}$$

$$= \frac{1}{4}\left(-\log 0.890740 - \log(1-0.951039) - \log(1-0.952226)\right.$$

$$\left. - \log 0.951039\right) \tag{18.36}$$

$$= 1.55598. \tag{18.37}$$

Như vậy ta đã thực hiện xong một thủ tục lan truyền xuôi. Bây giờ, ta tiếp tục tiến hành quá trình lan truyền ngược để tìm gradient của hàm mục tiêu theo các ma trận giá trị và các tham số.

7. Gradient của J theo véctơ giá trị lớp đầu ra $\widehat{\boldsymbol{Y}}$:

$$\nabla_{\widehat{\boldsymbol{Y}}} J = \frac{1}{m}\left(-\frac{\boldsymbol{Y}}{\widehat{\boldsymbol{Y}}} + \frac{1-\boldsymbol{Y}}{1-\widehat{\boldsymbol{Y}}}\right) = \begin{bmatrix} -0.280665 \\ 5.10613 \\ 5.23298 \\ -0.262870 \end{bmatrix}. \tag{18.38}$$

8. Gradient của J theo véctơ giá trị trước kích hoạt $\boldsymbol{Z}^{(2)}$ của lớp đầu ra:

$$\nabla_{\boldsymbol{Z}^{(2)}} J = \nabla_{\widehat{\boldsymbol{Y}}} J \odot f'\left(\boldsymbol{Z}^{(2)}\right) = \begin{bmatrix} -0.28066 \\ 5.10613 \\ 5.23298 \\ -0.26287 \end{bmatrix} \odot \begin{bmatrix} 0.0973221 \\ 0.0465636 \\ 0.0454916 \\ 0.0465636 \end{bmatrix} = \begin{bmatrix} -0.027315 \\ 0.237760 \\ 0.238057 \\ -0.012240 \end{bmatrix}. \tag{18.39}$$

9. Gradient của J theo các tham số của lớp đầu ra.

$$\nabla_{\boldsymbol{W}^{(2)}} J = \boldsymbol{H}^\top \left(\nabla_{\boldsymbol{Z}^{(2)}} J\right) = \begin{bmatrix} 0.435459 \\ 0.436102 \\ -0.00561987 \end{bmatrix}, \tag{18.40}$$

$$\nabla_{\boldsymbol{b}^{(2)}} J = \left(\nabla_{\boldsymbol{Z}^{(2)}} J\right)^\top \mathbf{1}_{m\times 1} = 0.436261. \tag{18.41}$$

10. Gradient của J theo ma trận giá trị của lớp ẩn:

$$\nabla_{\boldsymbol{H}} J = \left(\nabla_{\boldsymbol{Z}^{(2)}} J\right) \boldsymbol{W}^{(2)^\top} = \begin{bmatrix} -0.0546299 & 0 & 0.0819449 \\ 0.475520 & 0 & -0.713279 \\ 0.476113 & 0 & -0.714170 \\ -0.0244804 & 0 & 0.0367206 \end{bmatrix}. \tag{18.42}$$

11. Gradient của J theo ma trận trước kích hoạt của lớp ẩn:

$$\nabla_{\boldsymbol{Z}^{(1)}} J = \nabla_{\boldsymbol{H}} J \odot f'\left(\boldsymbol{Z}^{(1)}\right) = \begin{bmatrix} -0.002468 & 0 & 0.0161113 \\ 0.00316128 & 0 & -0.00474192 \\ 0.00117434 & 0 & -0.00065005 \\ -0.00016275 & 0 & 0.00024412 \end{bmatrix}. \tag{18.43}$$

12. Bước cuối cùng trong lan truyền ngược, tính grandient của J theo các tham số của lớp ẩn:

$$\nabla_{\boldsymbol{W}^{(1)}} J = \boldsymbol{H}^\top \left(\nabla_{\boldsymbol{Z}^{(1)}} J\right) = \begin{bmatrix} 0.00672659 & 0 & 0.00565382 \\ -0.00108862 & 0 & 0.0275631 \end{bmatrix}, \tag{18.44}$$

$$\nabla_{\boldsymbol{b}^{(1)}} J = \left(\nabla_{\boldsymbol{Z}^{(1)}} J\right)^\top \mathbf{1}_{m\times 1} = \begin{bmatrix} 0.00170487 \\ 0 \\ 0.0109635 \end{bmatrix}. \tag{18.45}$$

Sau khi thực hiện xong lan truyền ngược, ta cập nhật lại các tham số theo các phương pháp hướng giảm, với tốc độ học $\eta = 0.1$. Các tham số có số chiều khác nhau, nên ta có thể viết công thức cập nhật tường minh cho từng tham số. Việc thực hiện tham số chỉ nên thực hiện khi điều kiện dừng chưa thỏa mãn. Chẳng hạn điều kiện dừng là chuẩn của các gradient đều nhỏ hơn $\varepsilon = 10^{-5}$.

13. Kiểm tra điều kiện dừng bằng cách tính các chuẩn.

$$\left\|\nabla_{\boldsymbol{W}^{(1)}} J\right\| = 0.0289503, \quad \left\|\nabla_{\boldsymbol{b}^{(1)}} J\right\| = 0.0110953, \tag{18.46}$$

$$\left\|\nabla_{\boldsymbol{W}^{(2)}} J\right\| = 0.616312, \quad \left\|\nabla_{\boldsymbol{b}^{(2)}} J\right\| = 0.436261. \tag{18.47}$$

Như vậy điều kiện dừng chưa thỏa mãn, nên ta thực hiện bước cập nhật tham số dưới đây.

14. Cập nhật tham số $\boldsymbol{W}^{(1)}$, $\boldsymbol{b}^{(1)}$, $\boldsymbol{W}^{(2)}$, $\boldsymbol{b}^{(2)}$:

$$\boldsymbol{W}^{(1)} \leftarrow \boldsymbol{W}^{(1)} - \eta \nabla_{\boldsymbol{W}^{(1)}} J = \begin{bmatrix} 0.999327 & 0 & -2.00057 \\ -0.999891 & 3 & 1.99724 \end{bmatrix} \tag{18.48}$$

$$\boldsymbol{b}^{(1)} \leftarrow \boldsymbol{b}^{(1)} - \eta \nabla_{\boldsymbol{b}^{(1)}} J = \begin{bmatrix} 3.99983 \\ 5 \\ -3.00110 \end{bmatrix} \tag{18.49}$$

$$\boldsymbol{W}^{(2)} \leftarrow \boldsymbol{W}^{(2)} - \eta \nabla_{\boldsymbol{W}^{(2)}} J = \begin{bmatrix} 1.95645 \\ -0.0436102 \\ -2.99944 \end{bmatrix} \tag{18.50}$$

$$\boldsymbol{b}^{(2)} \leftarrow \boldsymbol{b}^{(2)} - \eta \nabla_{\boldsymbol{b}^{(2)}} J = 0.956374. \tag{18.51}$$

Ta lại quay về bước (2) để thực hiện tiếp quy trình ba giai đoạn lan truyền xuôi – lan truyền ngược – cập nhật tham số.

Thực hiện quy trình trên kèm theo điều kiện số bước lặp tối đa $N = 1\,000$ để đề phòng mãi không thỏa mãn điều kiện dừng, ta thu được kết quả sau:

- Số bước lặp chương trình đã thực hiện là 1000, chính là số bước lặp tối đa, nên hầu như chắc chắn điều kiện dừng về gradient đủ nhỏ không được đảm bảo

- Chuẩn L^2 của từng gradient của hàm mất mát theo các tham số

$$\left\|\nabla_{\boldsymbol{W}^{(1)}} J\right\| = 4.48751 \times 10^{-3}, \quad \left\|\nabla_{\boldsymbol{b}^{(1)}} J\right\| = 1.92980 \times 10^{-3}, \tag{18.52}$$

$$\left\|\nabla_{\boldsymbol{W}^{(2)}} J\right\| = 1.13802 \times 10^{-3}, \quad \left\|\nabla_{\boldsymbol{b}^{(2)}} J\right\| = 3.65930 \times 10^{-4}, \tag{18.53}$$

  tuy chưa đạt điều kiện dừng nhưng cũng đã giảm khá nhiều so với các giá trị ban đầu của chúng ở bước (13).

- Hàm mất mát cũng giảm dần từ 1.55598 xuống còn 0.695883.

- Hàm dự báo mà mô hình học được, được xây dựng từng bước theo quy tắc hàm hợp.

$$\boldsymbol{z}^{(1)} = \boldsymbol{W}^{(1)\top}\boldsymbol{x} + \boldsymbol{b}^{(1)} \tag{18.54}$$

$$\boldsymbol{h} = \sigma\left(\boldsymbol{z}^{(1)}\right) = \sigma\left(\boldsymbol{W}^{(1)\top}\boldsymbol{x} + \boldsymbol{b}^{(1)}\right) \tag{18.55}$$

$$\boldsymbol{z}^{(2)} = \boldsymbol{W}^{(2)\top}\boldsymbol{h} + \boldsymbol{b}^{(2)} = \boldsymbol{W}^{(2)\top}\sigma\left(\boldsymbol{W}^{(1)\top}\boldsymbol{x} + \boldsymbol{b}^{(1)}\right) + \boldsymbol{b}^{(2)} \tag{18.56}$$

$$\Rightarrow \widehat{y} = \sigma\left(\boldsymbol{z}^{(2)}\right) = \sigma\left(\boldsymbol{W}^{(2)\top}\sigma\left(\boldsymbol{W}^{(1)\top}\boldsymbol{x} + \boldsymbol{b}^{(1)}\right) + \boldsymbol{b}^{(2)}\right). \tag{18.57}$$

- Đường phân tách hai lớp ứng với đường mức $\widehat{y} = 0.5$.