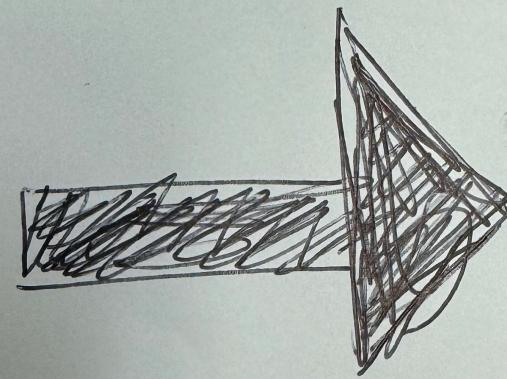
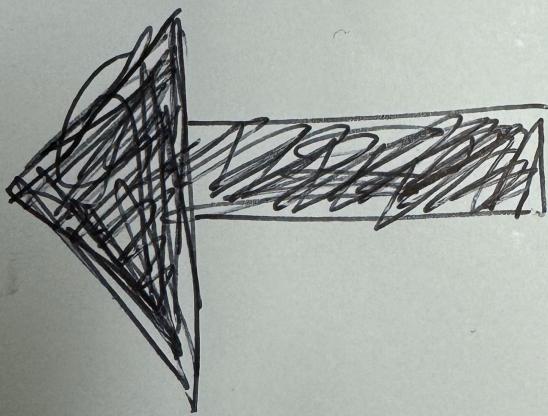


Softmax regression



Multiclass classification: generalization of binary classification

	Logistic regression	Softmax regression
Output	Binary	Multiclass
Activation function	Sigmoid	Softmax
Prediction	Threshold	Argmax
Loss function	Binary cross-entropy	Cross-entropy

1. **Modeling**
2. **Prediction (forward propagation)**
3. Loss & cost function
4. Gradient descent (backward propagation)

From binary to multiclass classification

Logistic regression

2 possible outcomes

- $a_1 = g(z) = \frac{1}{1+e^{-z}} = P(y = 1|x)$
- $a_2 = 1 - a_1 = P(y = 0|x)$

Softmax regression

K possible outcomes

- $a_1 = \text{softmax}(z_1) = P(y = 1|x)$
- $a_2 = \text{softmax}(z_2) = P(y = 2|x)$
- \vdots
- $a_K = \text{softmax}(z_K) = P(y = K|x)$

Softmax function

$$a_k = \text{softmax}(z_k) = \frac{e^{z_k}}{\sum_{k=1}^K e^{z_k}} = P(y = k|x)$$

For class $k = 1, 2, \dots, K$

$$a_1 = \text{softmax}(z_1) = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + \dots + e^{z_K}} = P(y = 1|x)$$

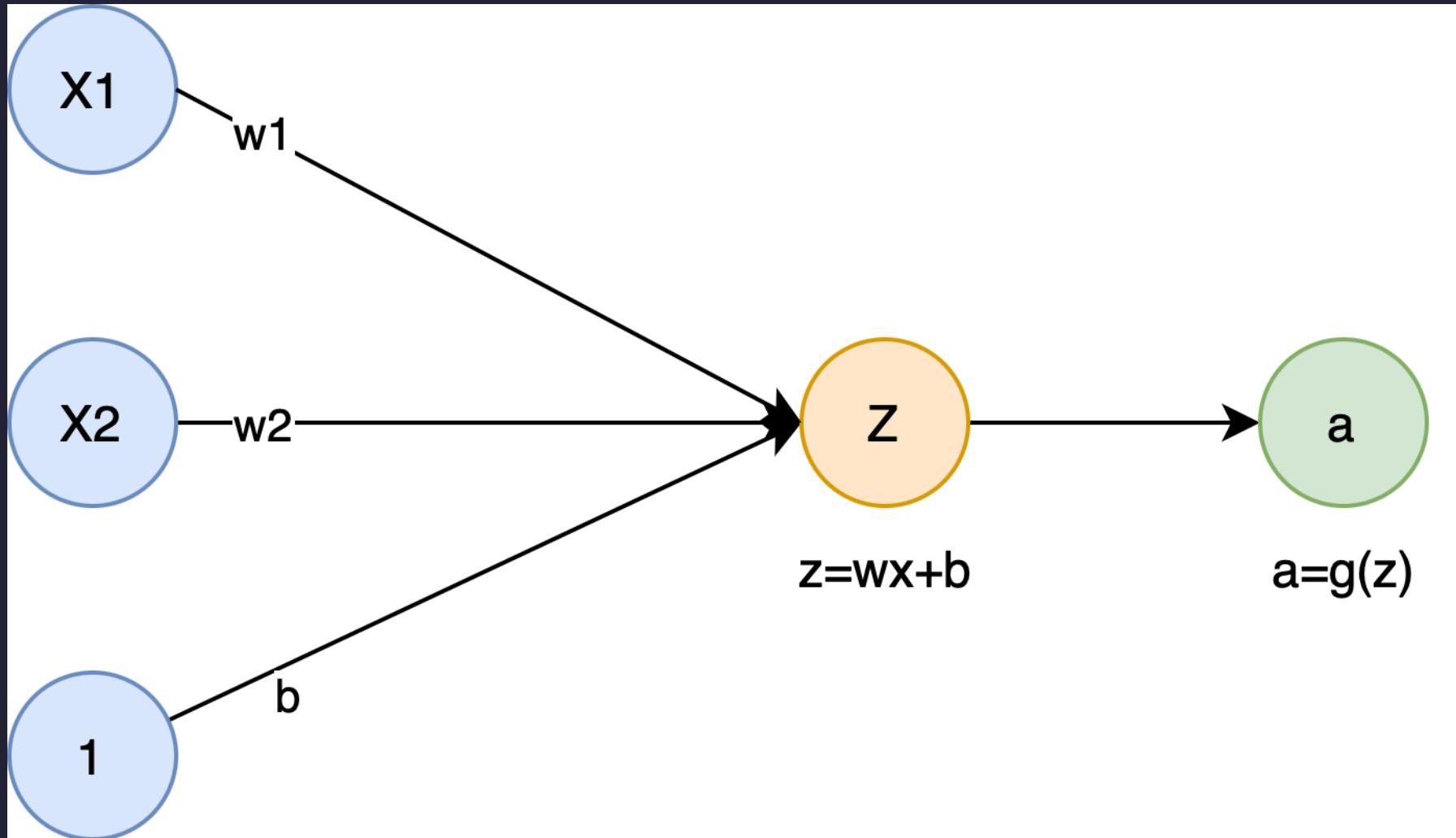
$$a_2 = \text{softmax}(z_2) = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + \dots + e^{z_K}} = P(y = 2|x)$$

⋮

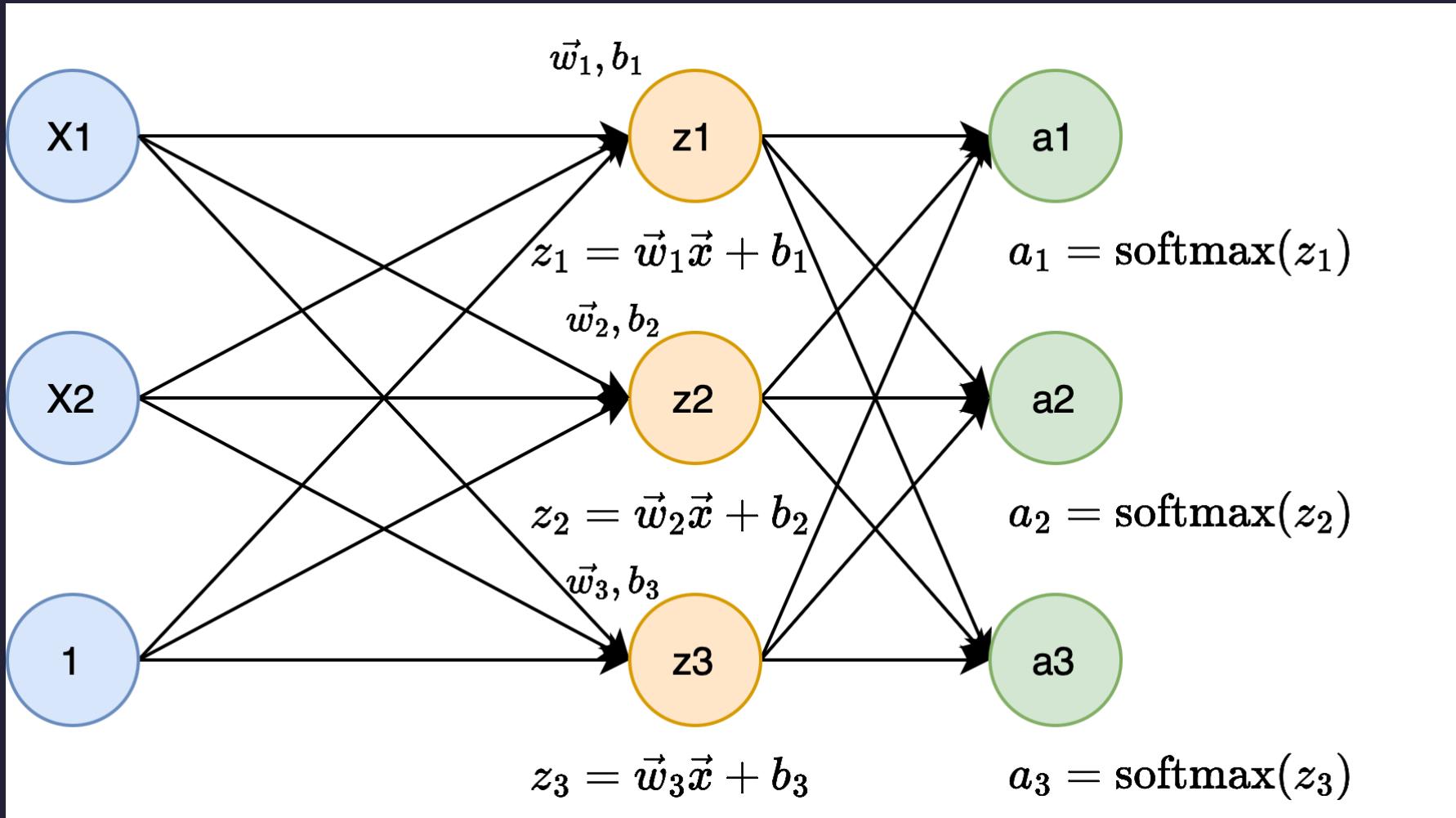
$$a_K = \text{softmax}(z_K) = \frac{e^{z_K}}{e^{z_1} + e^{z_2} + \dots + e^{z_K}} = P(y = N|x)$$

logistic(sigmoid) function is a special case of softmax function. See page 23.

Logistic regression for two classes



Softmax regression for three classes



Compute sigmoid probability for two classes

Model: $w = 2, b = 1$

Input: $x = 1$

The probability of class 1, 2 for the input?

$$z = wx + b = 2 \cdot 1 + 1 = 3$$

$$a = g(z) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-3}} = 0.9526$$

$$1 - a = 1 - 0.9526 = 0.0474$$

Compute softmax probability for three classes

Model: $\vec{w} = [2, 3, 5], \vec{b} = [1, 1, 1]$

Input: $x = 1$

The probability of class 1, 2, 3 for the input?

$$z_1 = w_1 x + b_1 = 2 \cdot 1 + 1 = 3$$

$$z_2 = w_2 x + b_2 = 3 \cdot 1 + 1 = 4$$

$$z_3 = w_3 x + b_3 = 5 \cdot 1 + 1 = 6$$

$$a_1 = \text{softmax}(z_1) = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}} = \frac{e^3}{e^3 + e^4 + e^6} = 0.0420$$

$$a_2 = \text{softmax}(z_2) = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}} = \frac{e^4}{e^3 + e^4 + e^6} = 0.1141$$

$$a_3 = \text{softmax}(z_3) = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}} = \frac{e^6}{e^3 + e^4 + e^6} = 0.8437$$

Prediction

Select the class with the highest probability

$$\begin{aligned}\hat{y} &= \operatorname{argmax}_k P(y = k|x) \\ &= \operatorname{argmax}_k a_k\end{aligned}$$

- a_k : predicted probability of class k
- k : class index ($1, 2, \dots, K$)
- e.g., $\hat{y} = 3$ if $a_1 = 0.0420, a_2 = 0.1141, a_3 = 0.8437$

1. Modeling
2. Prediction (forward propagation)
- 3. Loss & cost function**
4. Gradient descent (backward propagation)

Loss function: cross-entropy loss

Logistic regression

binary cross-entropy loss (log loss)

$$\begin{aligned} L &= \begin{cases} -\log(a_1) & \text{if } y = 1 \\ -\log(1 - a_1) = -\log(a_2) & \text{if } y = 0 \end{cases} \\ &= -y \log(a_1) - (1 - y) \log(1 - a_1) \\ &= -y_1 \log(a_1) - y_2 \log(a_2) \\ &= -\sum_{k=1}^2 y_k \log(a_k) \end{aligned}$$

Softmax regression

cross-entropy loss

$$\begin{aligned} L &= \begin{cases} -\log(a_1) & \text{if } y = 1 \\ -\log(a_2) & \text{if } y = 2 \\ \vdots & \\ -\log(a_K) & \text{if } y = K \end{cases} \\ &= -y_1 \log(a_1) - y_2 \log(a_2) - \dots \\ &= -\sum_{k=1}^K y_k \log(a_k) \end{aligned}$$

Cost function: average loss over all samples

Logistic regression

$$J = -\frac{1}{m} \sum_{i=1}^m \left[\sum_{k=1}^2 y_k^{(i)} \log(a_k^{(i)}) \right]$$

Softmax regression

$$J = -\frac{1}{m} \sum_{i=1}^m \left[\sum_{k=1}^K y_k^{(i)} \log(a_k^{(i)}) \right]$$

$y_k^{(i)}$: output label of class k for sample i
 $a_k^{(i)}$: predicted probability of class k for sample i

m : number of samples

K : number of classes

1. Modeling
2. Prediction (forward propagation)
3. Loss & cost function
4. **Gradient descent (backward propagation)**

Gradient descent

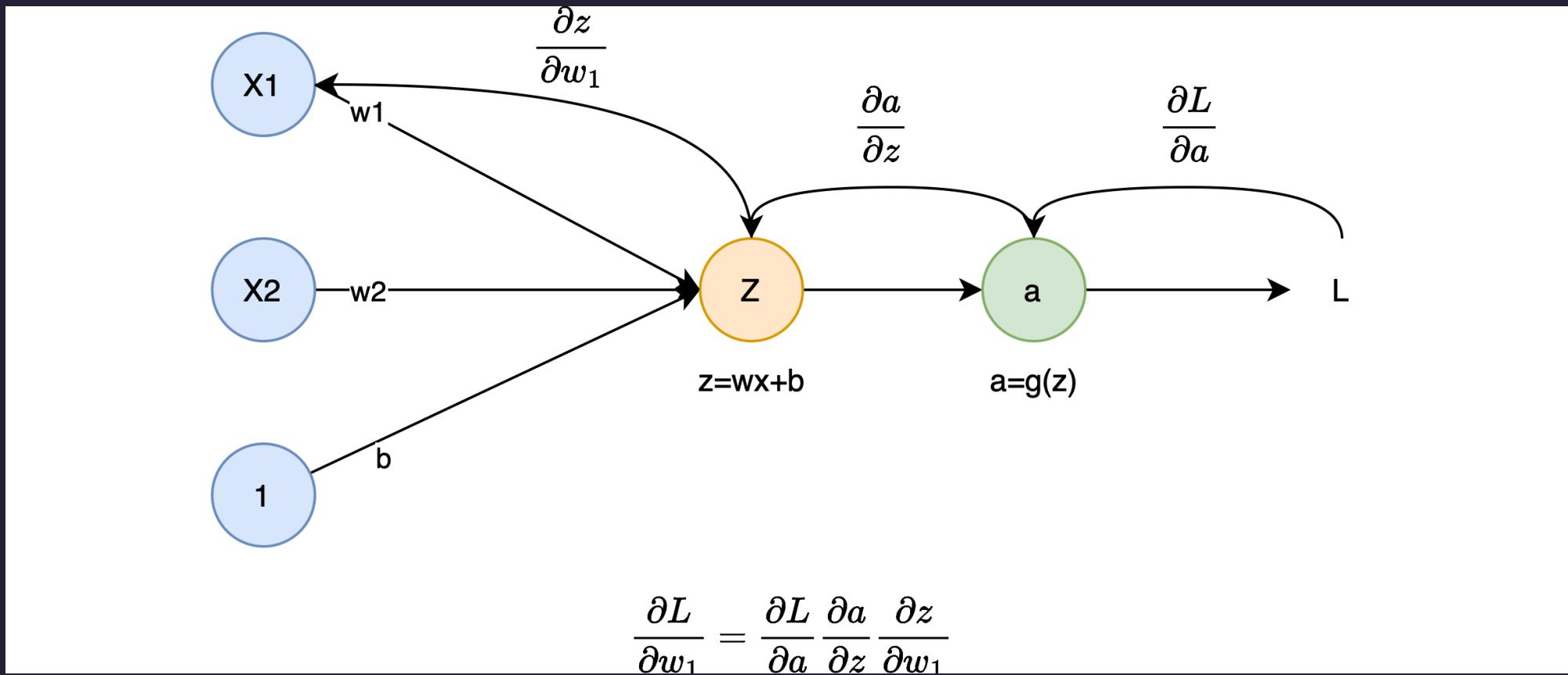
Repeat until convergence

$$w_{k,j} = w_{k,j} - \alpha \frac{\partial J}{\partial w_{k,j}}$$

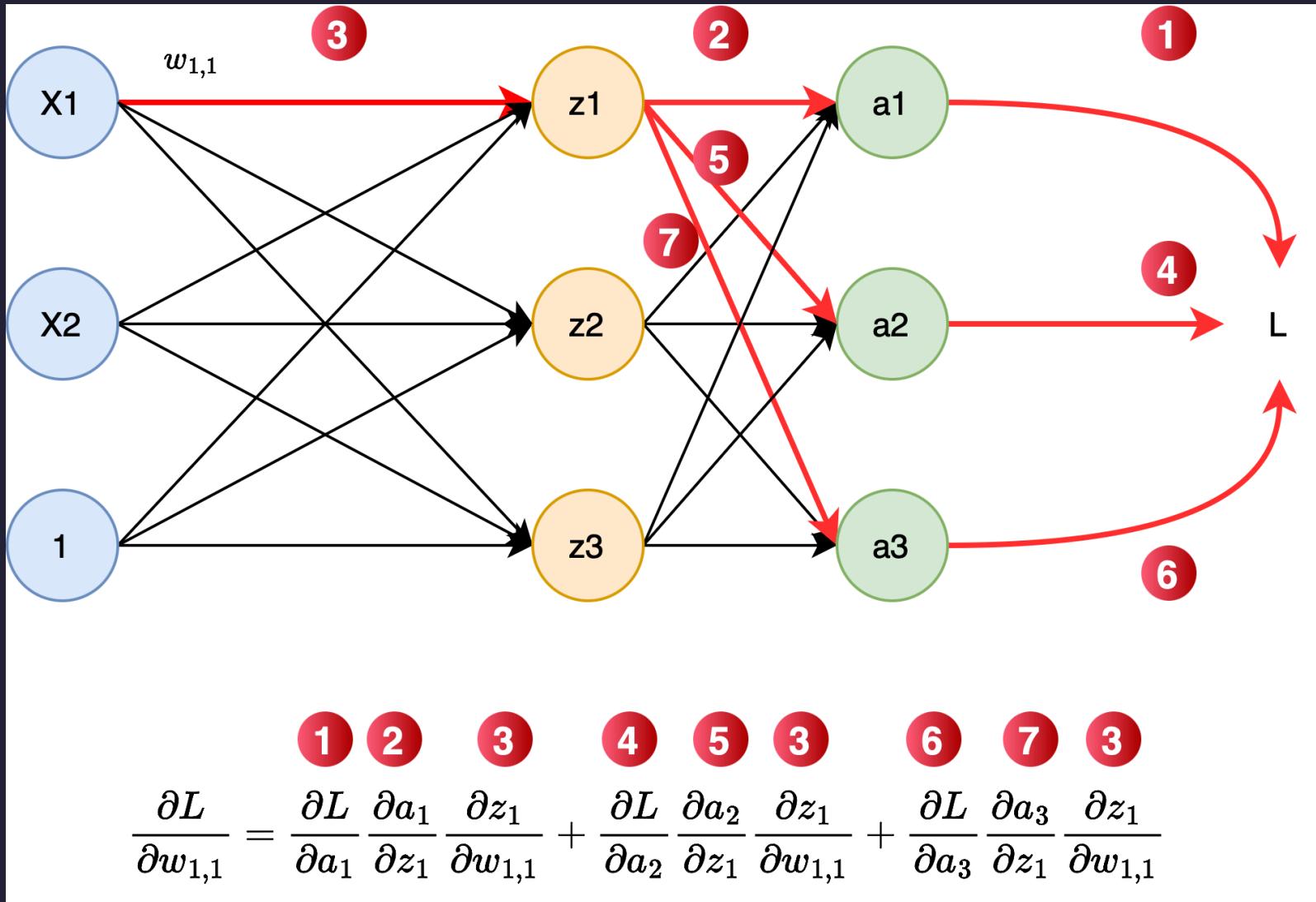
$$b_k = b_k - \alpha \frac{\partial J}{\partial b_k}$$

- k : class index ($1, 2, \dots, K$)
- j : feature index ($1, 2, \dots, N$)

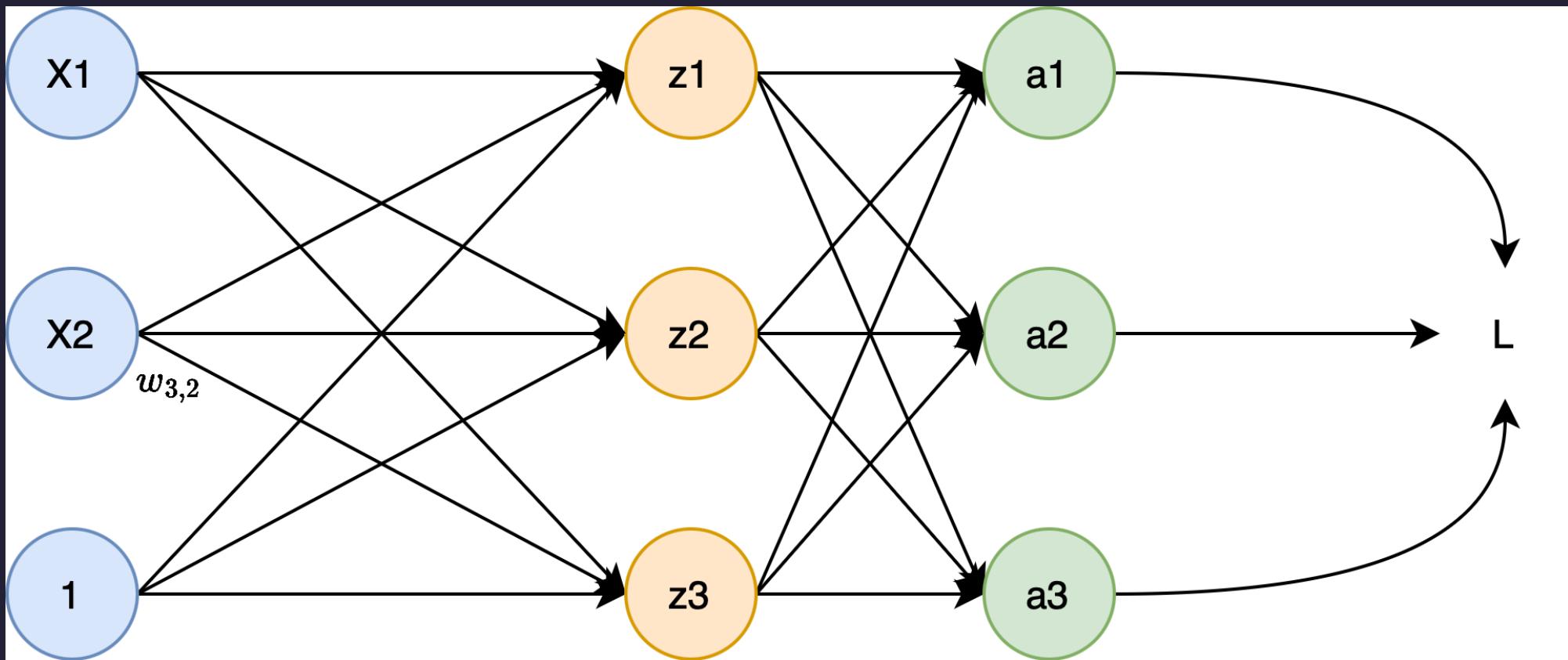
$\partial L / \partial w_1$: how changes in w_1 affect the loss L



$\partial L / \partial w_{1,1}$: how changes in $w_{1,1}$ affect the loss L



$\partial L / \partial w_{3,2}?$



Gradient with respect to weight $w_{1,1}$

$w_{1,1}$: weight for class 1 and feature 1

$$z_1 = w_{1,1}x_1 + w_{1,2}x_2 + b_1$$

$$a_1 = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_2 = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_3 = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

$$L = -y_1 \log(a_1) - y_2 \log(a_2) - y_3 \log(a_3)$$

$$\begin{aligned}\frac{\partial L}{\partial w_{1,1}} &= \frac{\partial L}{\partial a_1} \frac{\partial a_1}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} + \frac{\partial L}{\partial a_2} \frac{\partial a_2}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} + \frac{\partial L}{\partial a_3} \frac{\partial a_3}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} \\ &= (a_1 - y_1)x_1\end{aligned}$$

See page 20 for the derivation

Gradient with respect to bias b_1

b_1 : bias for class 1

$$\begin{aligned}\frac{\partial L}{\partial b_1} &= \frac{\partial L}{\partial a_1} \frac{\partial a_1}{\partial z_1} \frac{\partial z_1}{\partial b_1} + \frac{\partial L}{\partial a_2} \frac{\partial a_2}{\partial z_1} \frac{\partial z_1}{\partial b_1} + \frac{\partial L}{\partial a_3} \frac{\partial a_3}{\partial z_1} \frac{\partial z_1}{\partial b_1} \\ &= (a_1 - y_1)\end{aligned}$$

See page 21 for the derivation

Gradient of the cost function

$w_{k,j}$: weight for class k and feature j

b_k : bias for class k

Partial derivative of the cost function with respect to $w_{k,j}$

$$\frac{\partial J}{\partial w_{k,j}} = \frac{1}{m} \sum_{i=1}^m (a_k^{(i)} - y_k^{(i)}) x_j^{(i)}$$

Partial derivative of the cost function with respect to b_k

$$\frac{\partial J}{\partial b_k} = \frac{1}{m} \sum_{i=1}^m (a_k^{(i)} - y_k^{(i)})$$

Comparison of gradients

Linear regression

$$\frac{\partial J}{\partial w_j} = \frac{2}{m} \sum_{i=1}^m (a^{(i)} - y^{(i)}) x_j^{(i)}$$

Logistic regression

$$\frac{\partial J}{\partial w_j} = \frac{1}{m} \sum_{i=1}^m (a^{(i)} - y^{(i)}) x_j^{(i)}$$

Softmax regression

$$\frac{\partial J}{\partial w_{k,j}} = \frac{1}{m} \sum_{i=1}^m (a_k^{(i)} - y_k^{(i)}) x_j^{(i)}$$



Softmax regression

Gradient for weight $w_{1,1}$

$$\begin{aligned}L &= -y_1 \log(a_1) - y_2 \log(a_2) - y_3 \log(a_3) \\a_1 &= \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_2 = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_3 = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}} \\z_1 &= w_{1,1}x_1 + w_{1,2}x_2 + b_1 \\\frac{\partial L}{\partial w_{1,1}} &= \frac{\partial L}{\partial a_1} \frac{\partial a_1}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} + \frac{\partial L}{\partial a_2} \frac{\partial a_2}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} + \frac{\partial L}{\partial a_3} \frac{\partial a_3}{\partial z_1} \frac{\partial z_1}{\partial w_{1,1}} \\&= -\frac{y_1}{a_1} a_1(1 - a_1)x_1 - \frac{y_2}{a_2} (-a_1a_2)x_1 - \frac{y_3}{a_3} (-a_1a_3)x_1 \\&= -y_1x_1 + a_1x_1(y_1 + y_2 + y_3) \\&= -y_1x_1 + a_1x_1 \\&= (a_1 - y_1)x_1\end{aligned}$$

Gradient for bias b_1

$$\begin{aligned}L &= -y_1 \log(a_1) - y_2 \log(a_2) - y_3 \log(a_3) \\a_1 &= \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_2 = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}, a_3 = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}} \\z_1 &= w_{1,1}x_1 + w_{1,2}x_2 + b_1 \\\frac{\partial L}{\partial b_1} &= \frac{\partial L}{\partial a_1} \frac{\partial a_1}{\partial z_1} \frac{\partial z_1}{\partial b_1} + \frac{\partial L}{\partial a_2} \frac{\partial a_2}{\partial z_1} \frac{\partial z_1}{\partial b_1} + \frac{\partial L}{\partial a_3} \frac{\partial a_3}{\partial z_1} \frac{\partial z_1}{\partial b_1} \\&= -\frac{y_1}{a_1} a_1(1 - a_1) - \frac{y_2}{a_2} (-a_1 a_2) - \frac{y_3}{a_3} (-a_1 a_3) \\&= -y_1 + a_1(y_1 + y_2 + y_3) \\&= a_1 - y_1\end{aligned}$$

$$\partial a_1 / \partial z_1$$

Use Quotient rule ($\frac{d}{dx} \frac{f(x)}{g(x)} = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$)

$$\begin{aligned} a_1 &= \frac{e^{z_1}}{\sum_{k=1}^K e^{z_k}} \\ \frac{\partial a_1}{\partial z_1} &= \frac{e^{z_1} \sum_{k=1}^K e^{z_k} - e^{z_1} e^{z_1}}{(\sum_{k=1}^K e^{z_k})^2} \\ &= \frac{e^{z_1} (\sum_{k=1}^K e^{z_k} - e^{z_1})}{(\sum_{k=1}^K e^{z_k})^2} \\ &= \frac{e^{z_1}}{\sum_{k=1}^K e^{z_k}} \left(1 - \frac{e^{z_1}}{\sum_{k=1}^K e^{z_k}} \right) \\ &= a_1 (1 - a_1) \end{aligned}$$

$$\partial a_2 / \partial z_1$$

Use Quotient rule ($\frac{d}{dx} \frac{f(x)}{g(x)} = \frac{f'(x)g(x) - f(x)g'(x)}{(g(x))^2}$)

$$\begin{aligned} a_2 &= \frac{e^{z_2}}{\sum_{k=1}^K e^{z_k}} \\ \frac{\partial a_2}{\partial z_1} &= \frac{0 \cdot \sum_{k=1}^K e^{z_k} - e^{z_2} e^{z_1}}{\left(\sum_{k=1}^K e^{z_k}\right)^2} \\ &= -\frac{e^{z_2} e^{z_1}}{\left(\sum_{k=1}^K e^{z_k}\right)^2} \\ &= -a_2 a_1 \end{aligned}$$

Sigmoid function as a special case of softmax

$$\begin{aligned} a_1 &= \frac{e^{z_1}}{e^{z_1} + e^{z_2}} \\ &= \frac{\frac{e^{z_1}}{e^{z_1}}}{\frac{e^{z_1}}{e^{z_1}} + \frac{e^{z_2}}{e^{z_1}}} \\ &= \frac{1}{1 + e^{-(z_1 - z_2)}} \\ &= \frac{1}{1 + e^{-z'}} \end{aligned}$$