Mathematics of Neural Network

Rifqi Anshari Rasyid
https://rifqiansharir.github.io/

June 9, 2025

This is the backbone of how neural network works. The implementation of this can be found on code i have published. This neural network can amazingly recognize hand-written of digits from 0 to 9.

1 Overview

The architecture of the network consists of:

- **Input layer**: 784 (Batch size $\times$ 784)
- **Hidden layer**: 128 (Batch size $\times$ 128)
- **Output layer**: 10 (Batch size $\times$ 10)

The training configuration i used is as follows:

- **Input dimension**: Flattened 28×28 grayscale image
- **Hidden layer activation**: ReLU activation
- **Output layer activation**: Softmax activation
- **Loss function**: Categorical Cross-Entropy
- **Optimization**: Gradient descent (parameter update)
- **Learning rate**: 0.001
- **Batch size**: 100
- **Number of epochs**: 10

2 Forward Pass

- **Layer 0 (Input Layer)**:

$$X \in \mathbb{R}^{m \times d} \quad \text{(sample m and feature d)}$$

- **Layer 1 (Hidden Layer)**:

$$\boxed{z_1 = XW_1 + b_1} \tag{1}$$

$$\boxed{a_1 = \text{ReLU}(z_1)} \tag{2}$$

where:

$$\text{ReLU}(z_1) = \begin{cases} z_1, & \text{if } z_1 > 0 \\ 0, & \text{if } z_1 \leq 0 \end{cases} \tag{3}$$

- **Layer 2 (Output Layer)**:

$$\boxed{z_2 = a_1 W_2 + b_2} \tag{4}$$

$$\boxed{\hat{y} = \text{Softmax}(z_2)} \tag{5}$$

where:

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}} \quad \text{for } i = 1, \ldots, K \tag{6}$$

Target label with One-Hot encoding:

$$y \in \mathbb{R}^{m \times K} \quad \text{(sample m and class K)}$$

Loss function using Cross-Entropy:

$$\boxed{\mathcal{L} = -\sum_{i=1}^{K} y_i \log(\hat{y}_i)} \tag{7}$$

$$\mathcal{L}_{mean} = \frac{1}{m} \sum_{j=1}^{m} \mathcal{L}_j \tag{8}$$

3 Backward Pass

To ensure simplicity, all gradients will be derived from single sample.

- **Layer 2**:

  Chain rule for gradient of every components in layer 2 is as follows:

First, gradient of loss w.r.t the combination of pre-activation and post-activation of layer 2 (Softmax and Cross-Entropy).

$$\frac{\partial \mathcal{L}}{\partial z_2} = \frac{\partial \mathcal{L}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial z_2} \tag{9}$$

We will derive eq. (9) by examining class of index i and j.

For every z_2 with index i affects all y, we can derive eq. (9) as follows:

$$\frac{\partial \mathcal{L}}{\partial z_i} = \sum_{j=1}^{K} \frac{\partial \mathcal{L}}{\partial \hat{y}_j} \cdot \frac{\partial \hat{y}_j}{\partial z_i} \quad \text{for } i = 1, \ldots, K \tag{10}$$

Substitute with eq. (5), (6), and (7), we get:

$$\frac{\partial \mathcal{L}}{\partial \hat{y}_j} = -\frac{y_j}{\hat{y}_j} \tag{11}$$

and

$$\frac{\partial \hat{y}_j}{\partial z_i} = \frac{\partial}{\partial z_i} \left(\frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \right) \tag{12}$$

By applying quotient rule, hence for $i = j$:

$$\left. \frac{\partial \hat{y}_j}{\partial z_i} \right|_{i=j} = \hat{y}_j(1 - \hat{y}_j) \tag{13}$$

$$\left. \frac{\partial \mathcal{L}}{\partial z_i} \right|_{i=j} = \sum_{\substack{i=1 \\ i=j}}^{K} \left(-\frac{y_j}{\hat{y}_j} \right) \cdot (\hat{y}_j(1 - \hat{y}_j)) = -y_j(1 - \hat{y}_j) \tag{14}$$

since we derived over ∂z_i and $i = j$, therefore eq. (14) can also be written:

$$\left. \frac{\partial \mathcal{L}}{\partial z_i} \right|_{i=j} = -\sum_{\substack{i=1 \\ i=j}}^{K} y_i(1 - \hat{y}_i) \tag{15}$$

when $i =$ index target label, therefore $y_i = 1$ and if $i \neq$ index target label, $y_i = 0$.

$$\left. \frac{\partial \mathcal{L}}{\partial z_i} \right|_{i=j} = \begin{cases} -y_i(1 - \hat{y}_i), & \text{if } i = \text{index target label} \\ 0, & \text{if } i \neq \text{index target label} \end{cases} \tag{16}$$

and for $i \neq j$:

$$\left. \frac{\partial \hat{y}_j}{\partial z_i} \right|_{i \neq j} = -\hat{y}_j \hat{y}_i \tag{17}$$

$$\left. \frac{\partial \mathcal{L}}{\partial z_i} \right|_{i \neq j} = \sum_{\substack{i=1 \\ i \neq j}}^{K} \left(-\frac{y_j}{\hat{y}_j} \right) \cdot (-\hat{y}_j \hat{y}_i) = y_j \hat{y}_i \tag{18}$$

and also, when $i =$ index target label, therefore $y_j = 0$ and if $i \neq$ index target label, there is one $y_j = 1$.

$$\left.\frac{\partial \mathcal{L}}{\partial z_i}\right|_{i \neq j} = \begin{cases} 0, & \text{if } i = \text{index target label} \\ y_j \hat{y}_i, & \text{if } i \neq \text{index target label} \end{cases} \tag{19}$$

Here comes the tricky part. Because we used One-Hot encoding for target label y, and the total of One-Hot encoding is 1, hence $y_j = 1 - y_i$ vice versa. This will gives us $y_j \hat{y}_i = (1 - y_i)\hat{y}_i$.

Combine eq. (16) and (19) we get:

$$\frac{\partial \mathcal{L}}{\partial z_i} = \left.\frac{\partial \mathcal{L}}{\partial z_i}\right|_{i=j} + \left.\frac{\partial \mathcal{L}}{\partial z_i}\right|_{i \neq j} \tag{20}$$

$$\frac{\partial \mathcal{L}}{\partial z_i} = (-y_i(1 - \hat{y}_i)) + ((1 - y_i)\hat{y}_i) \tag{21}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial z_2} = \hat{y} - y} \tag{22}$$

Next we can derive gradient of loss w.r.t the weight of layer 2 as follows:

$$\frac{\partial \mathcal{L}}{\partial W_2} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \frac{\partial z_2}{\partial W_2} \tag{23}$$

$$\frac{\partial \mathcal{L}}{\partial W_2} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \left(\frac{\partial (a_1 W_2 + b_2)}{\partial W_2}\right) \tag{24}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial W_2} = a_1^\top \cdot \frac{\partial \mathcal{L}}{\partial z_2}} \tag{25}$$

Lastly for layer 2 components, we can derive gradient of loss w.r.t bias of layer 2 as follows:

$$\frac{\partial \mathcal{L}}{\partial b_2} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \frac{\partial z_2}{\partial b_2} \tag{26}$$

$$\frac{\partial \mathcal{L}}{\partial b_2} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \left(\frac{\partial (a_1 W_2 + b_2)}{\partial b_2}\right) \tag{27}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial b_2} = \frac{\partial \mathcal{L}}{\partial z_2}} \tag{28}$$

For eq. (27) and (29) we need to add small adjustment because we used batch of size m, by averaging gradients of weights and biases over the batch.

- **Layer 1**:

  Chain rule for gradient of every components in layer 1 is as follows:

  First, gradient of loss w.r.t post-activation of layer 1.

$$\frac{\partial \mathcal{L}}{\partial a_1} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \frac{\partial z_2}{\partial a_1} \tag{29}$$

$$\frac{\partial \mathcal{L}}{\partial a_1} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \left(\frac{\partial (a_1 W_2 + b_2)}{\partial a_1} \right) \tag{30}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial a_1} = W_2^\top \cdot \frac{\partial \mathcal{L}}{\partial z_2}} \tag{31}$$

Second, gradient of loss w.r.t pre-activation of layer 1.

$$\frac{\partial \mathcal{L}}{\partial z_1} = \frac{\partial \mathcal{L}}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \tag{32}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial z_1} = \frac{\partial \mathcal{L}}{\partial a_1} \cdot (\mathrm{ReLU}'(z_1))} \tag{33}$$

where:

$$\mathrm{ReLU}'(z_1) = \begin{cases} 1, & \text{if } z_1 > 0 \\ 0, & \text{if } z_1 \leq 0 \end{cases} \tag{34}$$

Third, gradient of loss w.r.t weight of layer 1.

$$\frac{\partial \mathcal{L}}{\partial W_1} = \frac{\partial \mathcal{L}}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial W_1} \tag{35}$$

$$\frac{\partial \mathcal{L}}{\partial W_1} = \frac{\partial \mathcal{L}}{\partial z_1} \cdot \left(\frac{\partial (X W_1 + b_1)}{\partial W_1} \right) \tag{36}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial W_1} = X^\top \cdot \frac{\partial \mathcal{L}}{\partial z_1}} \tag{37}$$

Last, gradient of loss w.r.t bias of layer 1.

$$\frac{\partial \mathcal{L}}{\partial b_1} = \frac{\partial \mathcal{L}}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial b_1} \tag{38}$$

$$\frac{\partial \mathcal{L}}{\partial b_1} = \frac{\partial \mathcal{L}}{\partial z_1} \cdot \left(\frac{\partial (X W_1 + b_1)}{\partial b_1} \right) \tag{39}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial b_1} = \frac{\partial \mathcal{L}}{\partial z_1}} \tag{40}$$

For eq. (37) and (40) we also need to average gradients over the batch of size m.