Neural Network Backpropagation: 3-4-3 Architecture

September 2, 2025

1 Network Architecture

Input Layer: 3 neurons (x_1, x_2, x_3)
Hidden Layer: 4 neurons (h_1, h_2, h_3, h_4)
Output Layer: 3 neurons (y_1, y_2, y_3)

2 Weight Matrices and Bias Vectors

2.1 Input to Hidden Layer

Weight matrix $W^{(1)}$ (4×3):

$$W^{(1)} = \begin{bmatrix} 0.5 & -0.3 & 0.7 \\ -0.2 & 0.8 & 0.1 \\ 0.9 & -0.6 & 0.4 \\ 0.3 & 0.2 & -0.5 \end{bmatrix}$$

Bias vector $b^{(1)}$ (4×1):

$$b^{(1)} = \begin{bmatrix} 0.1 \\ -0.2 \\ 0.3 \\ 0.0 \end{bmatrix}$$

2.2 Hidden to Output Layer

Weight matrix $W^{(2)}$ (3×4):

$$W^{(2)} = \begin{bmatrix} 0.6 & -0.4 & 0.2 & 0.8 \\ -0.1 & 0.7 & -0.3 & 0.5 \\ 0.4 & 0.1 & 0.9 & -0.2 \end{bmatrix}$$

Bias vector $b^{(2)}$ (3×1):

$$b^{(2)} = \begin{bmatrix} 0.2 \\ -0.1 \\ 0.4 \end{bmatrix}$$

3 Forward Pass

3.1 Input to Hidden Layer

$$z_j^{(1)} = \sum_{i=1}^{3} W_{ji}^{(1)} x_i + b_j^{(1)} \quad \text{for } j = 1, 2, 3, 4 \tag{1}$$

$$h_j = \sigma(z_j^{(1)}) = \frac{1}{1 + e^{-z_j^{(1)}}} \tag{2}$$

3.2 Hidden to Output Layer

$$z_k^{(2)} = \sum_{j=1}^{4} W_{kj}^{(2)} h_j + b_k^{(2)} \quad \text{for } k = 1, 2, 3 \tag{3}$$

$$y_k = \sigma(z_k^{(2)}) = \frac{1}{1 + e^{-z_k^{(2)}}} \tag{4}$$

4 General Mathematical Framework

4.1 Loss Function

For mean squared error loss:

$$L = \frac{1}{2} \sum_{k=1}^{3} (y_k - t_k)^2$$

where t_k is the target output for neuron k.

4.2 Activation Function and Its Derivative

Sigmoid activation function:

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

Derivative of sigmoid:

$$\sigma'(z) = \sigma(z)(1 - \sigma(z))$$

5 General Formulas for Delta (Error Terms)

5.1 Output Layer Delta - General Formula

The delta for output layer neuron k is:

$$\delta_k^{(2)} = \frac{\partial L}{\partial z_k^{(2)}}$$

Using the chain rule:

$$\delta_k^{(2)} = \frac{\partial L}{\partial z_k^{(2)}} = \frac{\partial L}{\partial y_k} \cdot \frac{\partial y_k}{\partial z_k^{(2)}}$$

For MSE loss:

$$\frac{\partial L}{\partial y_k} = \frac{\partial}{\partial y_k} \left[\frac{1}{2} \sum_{i=1}^{3} (y_i - t_i)^2 \right] = (y_k - t_k) \tag{5}$$

$$\frac{\partial y_k}{\partial z_k^{(2)}} = \frac{\partial}{\partial z_k^{(2)}} \sigma(z_k^{(2)}) = y_k(1 - y_k) \tag{6}$$

Therefore:

$$\boxed{\delta_k^{(2)} = (y_k - t_k) \cdot y_k(1 - y_k)}$$

5.2 Hidden Layer Delta - General Formula

The delta for hidden layer neuron j is:

$$\delta_j^{(1)} = \frac{\partial L}{\partial z_j^{(1)}}$$

Using the chain rule through all output neurons that this hidden neuron connects to:

$$\delta_j^{(1)} = \frac{\partial L}{\partial z_j^{(1)}} = \sum_{k=1}^{3} \frac{\partial L}{\partial z_k^{(2)}} \cdot \frac{\partial z_k^{(2)}}{\partial h_j} \cdot \frac{\partial h_j}{\partial z_j^{(1)}}$$

Since:

$$\frac{\partial L}{\partial z_k^{(2)}} = \delta_k^{(2)} \tag{7}$$

$$\frac{\partial z_k^{(2)}}{\partial h_j} = \frac{\partial}{\partial h_j}\left[\sum_{i=1}^{4} W_{ki}^{(2)} h_i + b_k^{(2)}\right] = W_{kj}^{(2)} \tag{8}$$

$$\frac{\partial h_j}{\partial z_j^{(1)}} = \frac{\partial}{\partial z_j^{(1)}}\sigma(z_j^{(1)}) = h_j(1-h_j) \tag{9}$$

Therefore:

$$\boxed{\delta_j^{(1)} = \left(\sum_{k=1}^{3} \delta_k^{(2)} W_{kj}^{(2)}\right) \cdot h_j(1-h_j)}$$

5.3 General Gradient Formulas

Once we have the deltas, the gradients are:

For output layer:

$$\frac{\partial L}{\partial W_{kj}^{(2)}} = \delta_k^{(2)} \cdot h_j \tag{10}$$

$$\frac{\partial L}{\partial b_k^{(2)}} = \delta_k^{(2)} \tag{11}$$

For hidden layer:

$$\frac{\partial L}{\partial W_{ji}^{(1)}} = \delta_j^{(1)} \cdot x_i \tag{12}$$

$$\frac{\partial L}{\partial b_j^{(1)}} = \delta_j^{(1)} \tag{13}$$

6 Concrete Example Computation

Let's use input $\mathbf{x} = [1.0, 0.5, -0.3]^T$ and target $\mathbf{t} = [0.8, 0.2, 0.6]^T$

6.1 Forward Pass Calculations

Hidden layer activations:

$$z_1^{(1)} = 0.5(1.0) + (-0.3)(0.5) + 0.7(-0.3) + 0.1 = 0.19 \tag{14}$$

$$z_2^{(1)} = (-0.2)(1.0) + 0.8(0.5) + 0.1(-0.3) + (-0.2) = -0.03 \tag{15}$$

$$z_3^{(1)} = 0.9(1.0) + (-0.6)(0.5) + 0.4(-0.3) + 0.3 = 0.78 \tag{16}$$

$$z_4^{(1)} = 0.3(1.0) + 0.2(0.5) + (-0.5)(-0.3) + 0.0 = 0.55 \tag{17}$$

$$h_1 = \sigma(0.19) = 0.547 \tag{18}$$
$$h_2 = \sigma(-0.03) = 0.493 \tag{19}$$
$$h_3 = \sigma(0.78) = 0.686 \tag{20}$$
$$h_4 = \sigma(0.55) = 0.634 \tag{21}$$

Output layer activations:

$$z_1^{(2)} = 0.6(0.547) + (-0.4)(0.493) + 0.2(0.686) + 0.8(0.634) + 0.2 = 0.768 \tag{22}$$
$$z_2^{(2)} = (-0.1)(0.547) + 0.7(0.493) + (-0.3)(0.686) + 0.5(0.634) + (-0.1) = 0.364 \tag{23}$$
$$z_3^{(2)} = 0.4(0.547) + 0.1(0.493) + 0.9(0.686) + (-0.2)(0.634) + 0.4 = 1.090 \tag{24}$$

$$y_1 = \sigma(0.768) = 0.683 \tag{25}$$
$$y_2 = \sigma(0.364) = 0.590 \tag{26}$$
$$y_3 = \sigma(1.090) = 0.748 \tag{27}$$

6.2 Backward Pass - Delta Calculations

Output layer deltas:

$$\delta_1^{(2)} = (0.683 - 0.8) \times 0.683 \times (1 - 0.683) = -0.117 \times 0.216 = -0.025 \tag{28}$$
$$\delta_2^{(2)} = (0.590 - 0.2) \times 0.590 \times (1 - 0.590) = 0.390 \times 0.242 = 0.094 \tag{29}$$
$$\delta_3^{(2)} = (0.748 - 0.6) \times 0.748 \times (1 - 0.748) = 0.148 \times 0.189 = 0.028 \tag{30}$$

Hidden layer deltas:

$$\delta_1^{(1)} = [(-0.025)(0.6) + (0.094)(-0.1) + (0.028)(0.4)] \times 0.547 \times (1 - 0.547) \tag{31}$$
$$= [-0.015 - 0.009 + 0.011] \times 0.247 = -0.013 \times 0.247 = -0.003 \tag{32}$$

$$\delta_2^{(1)} = [(-0.025)(-0.4) + (0.094)(0.7) + (0.028)(0.1)] \times 0.493 \times (1 - 0.493) \tag{33}$$
$$= [0.010 + 0.066 + 0.003] \times 0.250 = 0.079 \times 0.250 = 0.020 \tag{34}$$

$$\delta_3^{(1)} = [(-0.025)(0.2) + (0.094)(-0.3) + (0.028)(0.9)] \times 0.686 \times (1 - 0.686) \tag{35}$$
$$= [-0.005 - 0.028 + 0.025] \times 0.215 = -0.008 \times 0.215 = -0.002 \tag{36}$$

$$\delta_4^{(1)} = [(-0.025)(0.8) + (0.094)(0.5) + (0.028)(-0.2)] \times 0.634 \times (1 - 0.634) \tag{37}$$
$$= [-0.020 + 0.047 - 0.006] \times 0.232 = 0.021 \times 0.232 = 0.005 \tag{38}$$

7 Gradient Computations

7.1 Gradients for Output Layer Weights and Biases

$$\frac{\partial L}{\partial W_{kj}^{(2)}} = \delta_k^{(2)} \cdot h_j \tag{39}$$
$$\frac{\partial L}{\partial b_k^{(2)}} = \delta_k^{(2)} \tag{40}$$

Weight gradients $W^{(2)}$:

$$\frac{\partial L}{\partial W^{(2)}} = \begin{bmatrix} -0.025 \times 0.547 & -0.025 \times 0.493 & -0.025 \times 0.686 & -0.025 \times 0.634 \\ 0.094 \times 0.547 & 0.094 \times 0.493 & 0.094 \times 0.686 & 0.094 \times 0.634 \\ 0.028 \times 0.547 & 0.028 \times 0.493 & 0.028 \times 0.686 & 0.028 \times 0.634 \end{bmatrix}$$

$$= \begin{bmatrix} -0.014 & -0.012 & -0.017 & -0.016 \\ 0.051 & 0.046 & 0.064 & 0.060 \\ 0.015 & 0.014 & 0.019 & 0.018 \end{bmatrix}$$

Bias gradients $b^{(2)}$:

$$\frac{\partial L}{\partial b^{(2)}} = \begin{bmatrix} -0.025 \\ 0.094 \\ 0.028 \end{bmatrix}$$

7.2 Gradients for Hidden Layer Weights and Biases

$$\frac{\partial L}{\partial W_{ji}^{(1)}} = \delta_j^{(1)} \cdot x_i \tag{41}$$

$$\frac{\partial L}{\partial b_j^{(1)}} = \delta_j^{(1)} \tag{42}$$

Weight gradients $W^{(1)}$:

$$\frac{\partial L}{\partial W^{(1)}} = \begin{bmatrix} -0.003 \times 1.0 & -0.003 \times 0.5 & -0.003 \times (-0.3) \\ 0.020 \times 1.0 & 0.020 \times 0.5 & 0.020 \times (-0.3) \\ -0.002 \times 1.0 & -0.002 \times 0.5 & -0.002 \times (-0.3) \\ 0.005 \times 1.0 & 0.005 \times 0.5 & 0.005 \times (-0.3) \end{bmatrix}$$

$$= \begin{bmatrix} -0.003 & -0.002 & 0.001 \\ 0.020 & 0.010 & -0.006 \\ -0.002 & -0.001 & 0.001 \\ 0.005 & 0.003 & -0.002 \end{bmatrix}$$

Bias gradients $b^{(1)}$:

$$\frac{\partial L}{\partial b^{(1)}} = \begin{bmatrix} -0.003 \\ 0.020 \\ -0.002 \\ 0.005 \end{bmatrix}$$

8 Summary of All Gradients

All computed gradients ready for parameter updates:

- $\frac{\partial L}{\partial W^{(2)}}$: 3×4 matrix with values computed above
- $\frac{\partial L}{\partial b^{(2)}}$: 3×1 vector with values computed above
- $\frac{\partial L}{\partial W^{(1)}}$: 4×3 matrix with values computed above
- $\frac{\partial L}{\partial b^{(1)}}$: 4×1 vector with values computed above

These gradients can now be used with gradient descent: $\theta_{new} = \theta_{old} - \alpha \frac{\partial L}{\partial \theta}$