Labwork 5: Backprop

Phi Doan Minh Luong - 2440046

May 2025

1 Back propagation implementation

1.1 Gradient Calculation for Output Layer

- For each neuron in the output layer, the loss gradient with respect to the neuron's output (dL_dz) is calculated using the derivative of the sigmoid activation function and the difference between the predicted and expected values
- Gradients for weights (grad_w) and biases (grad_b) are computed using the chain rule

1.2 Gradient Calculation for Hidden Layers

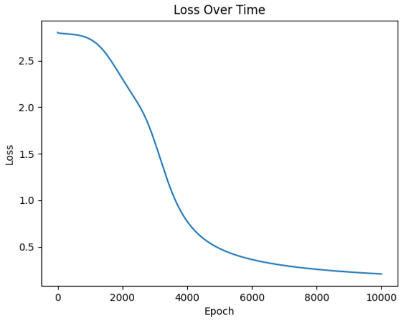
- For each neuron in the hidden layers, the gradient is propagated backward from the next layer
- The downstream gradient is calculated as the weighted sum of gradients from the next layer
- Gradients for weights and biases are computed using the chain rule

1.3 Weight and Bias updates

- After calculating gradients, the weights and biases are updated using the learning rate (lr)

2 Result on the XOR example

- The initial weights and biases are random. After 10000 epochs with a learning rate equals to 0.1, the loss decreased from 2.8 to 0.2


Loss Over Time
Loss
2.5
2.0
1.5
1.0
0.5
0
2000
4000
6000
8000
10000
Epoch