A4 - part 1: Deriving Backpropagation for Batch Gradient Descent

1. Review of Stochastic Gradient Descent:

In Stochastic Gradient Descent (SGD), we update the weights of a neural network iteratively for each training example. The update rule for the weights in each iteration is given by:

Where:

represents the weight between neuron in layer and neuron *j* in layer +1.

is the learning rate.

is the gradient of the loss function with respect to the weight .

is the true target, and ​ is the predicted target.

1. Understand Batch Gradient Descent:

Batch Gradient Descent (BGD) differs from SGD in that it computes the gradient of the loss function with respect to the weights using the entire training dataset in each iteration. It calculates the average gradient over the entire dataset and then updates the weights. In BGD, you sum the gradients of the loss function over all training examples and then update the weights. The update rule for BGD is similar to SGD but with the average gradient:

Where:

is the number of training examples.

is the gradient of the loss for the -nth training example.

1. Derive Backpropagation for Batch Gradient Descent:

To derive backpropagation equations for BGD, you need to compute the gradient of the loss function with respect to each weight for the entire dataset and then update the weights using the BGD update rule.

**Gradient Calculation for Output Layer (MSE Loss)**:

**Gradient Calculation for Hidden Layers (MSE Loss)**: