

Back Propagation Algorithm

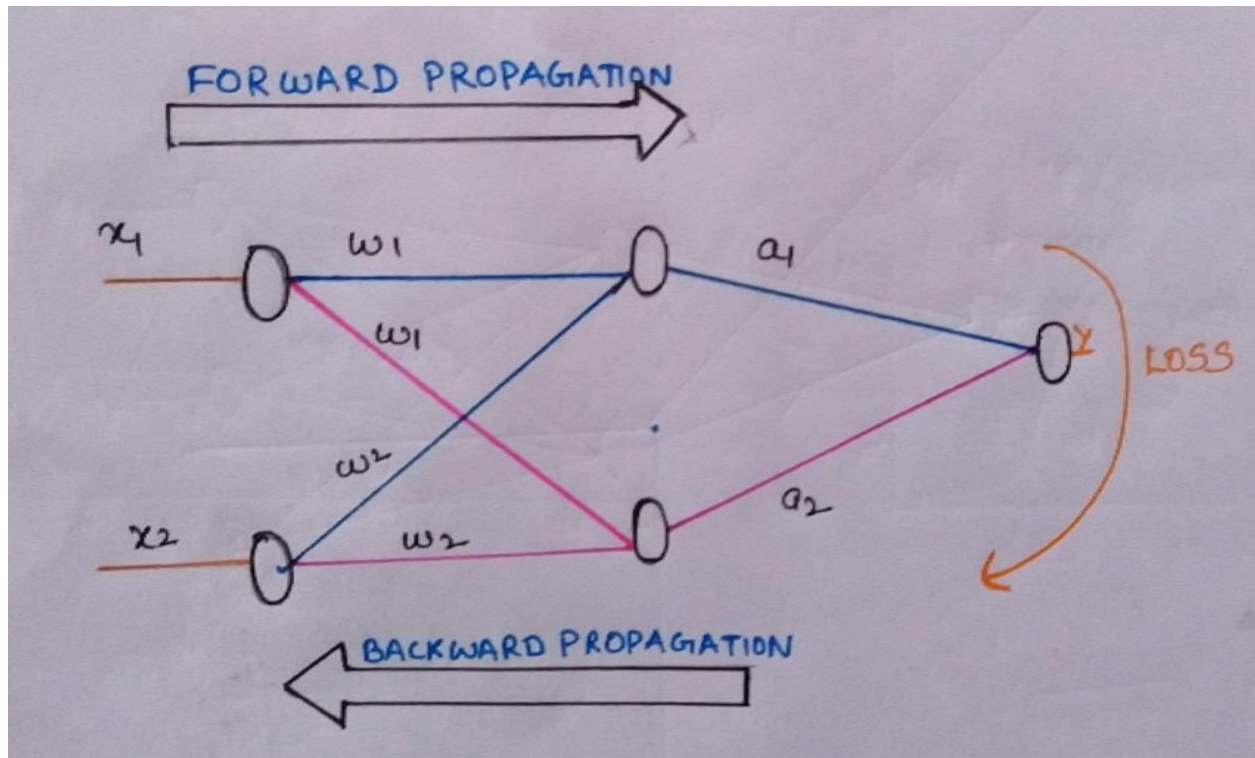


Figure: Forward and Backward Propagation

Learning a neural network means learning the value of parameter w and b (weights and bias). This is an iterative process which is forward and backward propagation. During forward propagation, information is transferred layer wise i.e from input to hidden and then from hidden to output. Computations are done at hidden and output layers. Predicted value is calculated as an output. Since, we are interested in supervised learning (where actual labels are known), loss between actual and predicted value is calculated and is propagated backward. The aim is to make the cost (**loss**) to be zero that is no divergence between actual and predicted value.

Let us suppose we have the output as 0.70 and actual output is 1.00. This means cost or error is

$$\text{cost or error} = (\text{predicted} - \text{actual})^2$$

$$\text{cost} = (0.70 - 1)^2$$

This is transmitted back as a gradient with respect to weights so that at each node new values are updated and predicted comes close to the actual value. Please have a look at the below screenshot to understand the back propagation with mathematics

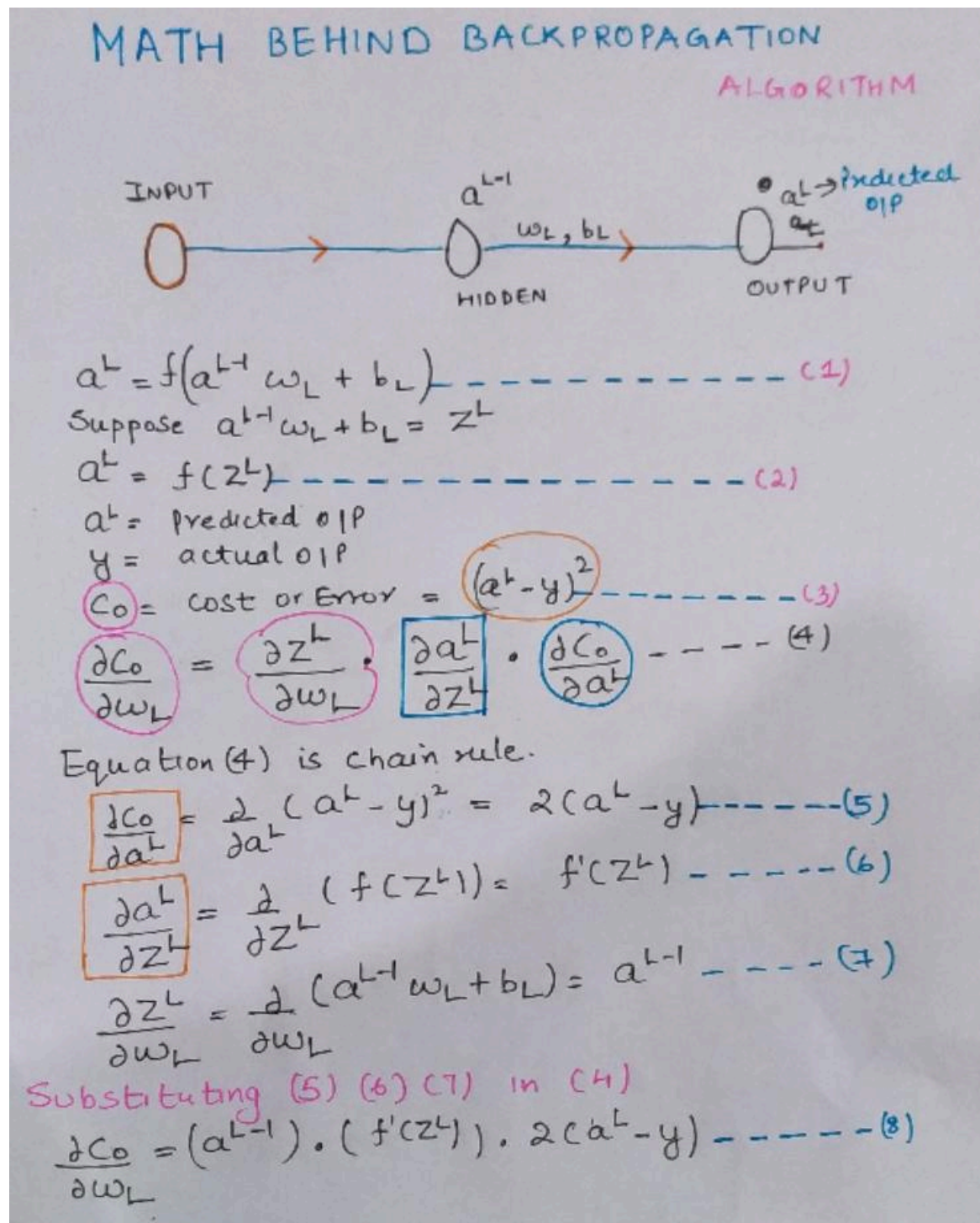


Figure: Maths behind Back Propagation Algorithm