Artificial Neural Networks

Symbols

1. i^{th} input: x_i
2. Expected output: y
3. Weight from k^{th} neuron in $(l-1)$ to j^{th} neuron in l^{th} layer: w_{jk}^{l}
4. Summation of j^{th} neuron in l^{th} layer: z_j^l
5. Summation of j^{th} neuron in the output layer: z_j^L
6. Activation of j^{th} neuron in l^{th} layer: a_j^l
7. Activation of j^{th} neuron in the output layer: a_j^L
8. Bias of j^{th} neuron in l^{th} layer: b_j^l
9. Activation function: $\sigma(x)$
10. Cost of j^{th} output: C_j
11. Learning rate: η
12. Mini batch size: m

Equations

1. $z_j^l = \sum_k a_k^{l-1} w_{jk}^l + b_j^l$
2. $a_j^l = \sigma\left(z_j^l\right)$
3. $C_j = \frac{1}{2}\left(a_j^L - y\right)^2$

Definitions

1. Rate of change of cost relative to sum (delta): $\delta_j^L = \frac{\partial C_j}{\partial z_j^L}$

Derivations

1. $C_j = \frac{1}{2}\left(a_j^L - y\right)^2$
   $$\frac{\partial C_j}{\partial a_j^L} = \left(a_j^L - y\right)$$

Four Fundamental Equations

1. $\mathit{Output\ error} = \delta_j^L = \frac{\partial C_j}{\partial a_j^L}\, \sigma'\left(z_j^L\right)$

Proof:

We start with the definition of δ_j^L

$$\delta_j^L = \frac{\partial C_j}{\partial z_j^L}$$

$$\delta_j^L = \frac{\partial C_j}{\partial a_j^L}\frac{\partial a_j^L}{\partial z_j^L}$$

$$a_j^L = \sigma\left(z_j^L\right)$$

$$\delta_j^L = \frac{\partial C_j}{\partial a_j^L}\frac{\sigma\left(z_j^L\right)}{\partial z_j^L}$$

$$\delta_j^L = \frac{\partial C_j}{\partial a_j^L}\sigma'\left(z_j^L\right)$$

2. $\mathit{Error\ in\ } l^{th} \mathit{layer\ in\ terms\ of\ error\ in\ } (l+1)\mathit{\ layer} = \delta_j^l = \sum_p w_{pj}^{l+1}\, \delta_p^{l+1}\, \sigma'\left(z_j^l\right)$

Proof:

$$\delta_j^l = \frac{\partial C_i}{\partial z_j^l}$$

$$= \sum_p \frac{\partial C_i}{\partial z_p^{l+1}} \frac{\partial z_p^{l+1}}{\partial z_j^l}$$

$$= \sum_p \delta_p^{l+1} \frac{\partial z_p^{l+1}}{\partial z_j^l}$$

$$z_p^{l+1} = \sum_j a_j^l w_{pj}^{l+1} + b_p^{l+1}$$

$$= \sum_j \sigma\left(z_j^l\right) w_{pj}^{l+1} + b_p^{l+1}$$

By differentiating,

$$\frac{\partial z_p^{l+1}}{\partial z_j^l} = \sigma'\left(z_j^l\right) w_{pj}^{l+1}$$

Substituting,

$$\delta_j^l = \sum_p w_{pj}^{l+1}\, \delta_p^{l+1}\, \sigma'\left(z_j^l\right)$$

3. $Rate\ of\ change\ of\ cost\ relative\ bias = \frac{\partial C_i}{\partial b_j^l} = \delta_j^l$

Proof:

$$\frac{\partial C_i}{\partial b_j^l} = \frac{\partial C_i}{\partial z_j^l} \frac{\partial z_j^l}{\partial b_j^l}$$

$$= \frac{\partial C_i}{\partial z_j^l} \frac{\partial z_j^l}{\partial b_j^l}$$

$$= \frac{\partial C_i}{\partial z_j^l} \frac{\partial\left[\sum_k a_k^{l-1} w_{jk}^l + b_j^l\right]}{\partial b_j^l}$$

$$= \frac{\partial C_i}{\partial z_j^l} \frac{\partial b_j^l}{\partial b_j^l}$$

$$= \delta_j^l$$

4. $Rate\ of\ change\ of\ cost\ relative\ to\ weight = \frac{\partial C_i}{\partial w_{jk}^l} = a_k^{l-1} \delta_j^l$

Proof:

$$\frac{\partial C_i}{\partial w_{jk}^l} = \frac{\partial C_i}{\partial z_j^l} \frac{\partial z_j^l}{\partial w_{jk}^l}$$

$$= \frac{\partial C_i}{\partial z_j^l} \frac{\partial\left[\sum_k a_k^{l-1} w_{jk}^l + b_j^l\right]}{\partial w_{jk}^l}$$

$$= \frac{\partial C_i}{\partial z_j^l} a_k^{l-1}$$

$$= a_k^{l-1} \delta_j^l$$

Algorithm

1. Initialize weights and biases w, b
2. Input x
3. Feedforward
   $a^1 = x$
   $for\ l = 2,3, \dots, L$
   $z^l = w^l . a^{l-1} + b^l$
   $a^l = \sigma(z^l)$
4. Output error
   $\delta^L = (a^L - y) * \sigma'(z^L)$
5. Backpropagate error and calculate gradient
   $for\ l = L-1, L-2, \dots, 2$

   $$\delta^l = ((w^{l+1})^T . \delta^{l+1}) * \sigma'(z^l)$$

6. Adjust weights and biases for every mini batches
   $for\ mini\_batch : mini\_batches$
   $w^{l,t} = w^{l,t-1} - \frac{\eta}{m} \sum_x \delta^{x,l} \left(a^{x,l-1}\right)^T$
   $$b^{l,t} = b^{l,t-1} - \frac{\eta}{m} \sum_x \delta^{x,l}$$