



신경망이 학습하는 것을 순서대로 나열해보면, Input layer로부터 각 layer를 지나치며 weight 들을 곱하고, activation function을 지난 다음, 마지막 output layer에서 오차 값을 계산한 이후 (feed forward) / 업데이트를 위해 순차적으로 거꾸로 (back) 돌아가야 (propagation) 합니다. 따라서 돌아갈 때 순서대로 가장 바깥 쪽 output과 hidden layer 사이의 weight부터 바꿔보겠습니다. i 번째 output인 x_i 에 대한 오차값에 대해 w_{ji} 가 바뀌어야 하는 정도를 계산하려면 다음과 같은 수식이 필요합니다:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial x_i} \frac{\partial x_i}{\partial s_i} \frac{\partial s_i}{\partial w_{ji}}$$

1. $\frac{\partial E}{\partial x_i} = \frac{-t_i}{x_i} + \frac{1-t_i}{1-x_i} = \frac{x_i-t_i}{x_i(1-x_i)}$
2. $\frac{\partial x_i}{\partial s_i} = x_i(1-x_i)$
3. $\frac{\partial s_i}{\partial w_{ji}} = x_j$

$$\therefore \frac{\partial E}{\partial w_{ji}} = (x_i - t_i)x_j$$