Tutorial 06 — Answers

(Version 1.1)

1. The weight update for the error correction method is:

$$w_i \leftarrow w_i + \alpha(t - g(s))x_i$$

where t is the target, or desired value, $g(s)$ is the weighted sum s passed through the transfer function, which in this case is:

$$g(s) = \begin{cases} 1 & \text{if } s \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

and α is the learning rate.

We are told that $\alpha = 0.5$, and the initial weights are $w_1 = 1$, and $w_2 = 2$.

To this we add, as always, a bias which we model with $x_0 = 1$ (in all cases) and an initial $w_0 = 1$.

- First example

$$\begin{aligned} s &= \sum w_i x_i \\ &= 1 \times 1 + 1 \times 1 + 2 \times 2 \\ &= 6 \\ g(s) &= 1 \end{aligned}$$

Since $t = 1$ we have:

$$\begin{aligned} w_0 &\leftarrow w_0 + \alpha(t - g(s))x_0 \\ &\leftarrow 1 + 0.5(1 - 1)1 \\ &\leftarrow 1 + 0 \\ w_0 &= 1 \end{aligned}$$

Since $t = g(s)$ in this case, the updates for all the weights will be zero and after the first iteration we will have:

$$\begin{aligned} w_0 &= 1 \\ w_1 &= 1 \\ w_2 &= 2 \end{aligned}$$

- Second example

This time we have:

$$\begin{aligned} s &= \sum w_i x_i \\ &= 1 \times 1 + 1 \times 3 + 2 \times 1 \\ &= 6 \\ g(s) &= 1 \end{aligned}$$

Now $t = 0$ so we have:

$$\begin{aligned} w_0 &\leftarrow w_0 + \alpha(t - g(s))x_0 \\ &\leftarrow 1 + 0.5(0 - 1)1 \\ &\leftarrow 1 - 0.5 \\ w_0 &= 0.5 \end{aligned}$$

Similarly:

$$\begin{aligned} w_1 &\leftarrow w_1 + \alpha(t - g(s))x_1 \\ &\leftarrow 1 + 0.5(0 - 1)3 \\ &\leftarrow 1 - 1.5 \\ w_1 &= -0.5 \\ w_2 &\leftarrow w_2 + \alpha(t - g(s))x_2 \\ &\leftarrow 2 + 0.5(0 - 1)1 \\ &\leftarrow 2 - 0.5 \\ w_2 &= 1.5 \end{aligned}$$

after the second iteration we will have:

$$\begin{aligned} w_0 &= 0.5 \\ w_1 &= -0.5 \\ w_2 &= 1.5 \end{aligned}$$

- Third example

$$\begin{aligned} s &= 0.5 \times 1 + (-0.5) \times 1 + 1.5 \times 1 \\ &= 1.5 \\ g(s) &= 1 \end{aligned}$$

Since $t = 1$, just as for the first example we will not have any updates, so after the third iteration we will have:

$$\begin{aligned} w_0 &= 0.5 \\ w_1 &= -0.5 \\ w_2 &= 1.5 \end{aligned}$$

- Fourth example

$$\begin{aligned} s &= 0.5 \times 1 + (-0.5) \times 2 + 1.5 \times 0 \\ &= -0.5 \\ g(s) &= 0 \end{aligned}$$

and since $t = 0$ again there will be no updates, and we have:

$$\begin{aligned} w_0 &= 0.5 \\ w_1 &= -0.5 \\ w_2 &= 1.5 \end{aligned}$$

2. Now we solve the some problem using the delta rule. This has the update rule:

$$w_i \leftarrow w_i + \alpha(t - s)x_i$$

The big difference between this and the error correction rule is that we compare t with s not $g(s)$, so we will adjust whenever the weighted sum is different from t. This means we update more often.

- First example

$$\begin{aligned} s &= \sum w_i x_i \\ &= 1 \times 1 + 1 \times 1 + 2 \times 2 \\ &= 6 \end{aligned}$$

Since $t = 1$ we have:

$$\begin{aligned} w_0 &\leftarrow w_0 + \alpha(t - s)x_0 \\ &\leftarrow 1 + 0.5(1 - 6)1 \\ &\leftarrow 1 + (-2.5) \\ w_0 &= -1.5 \end{aligned}$$

Similarly:

$$\begin{aligned} w_1 &\leftarrow w_1 + \alpha(t - s)x_1 \\ &\leftarrow 1 + 0.5 \times (1 - 6) \times 1 \\ &\leftarrow 1 + (-2.5) \\ w_1 &= -1.5 \\ w_2 &\leftarrow w_2 + \alpha(t - s)x_2 \\ &\leftarrow 2 + 0.5 \times (1 - 6) \times 2 \\ &\leftarrow 2 + (-5) \\ w_2 &= -3 \end{aligned}$$

and after the first example:

$$\begin{aligned} w_0 &= -1.5 \\ w_1 &= -1.5 \\ w_2 &= -3 \end{aligned}$$

So, even though the output of the perceptron was correct ($g(s) = 1$), because the value of s was much bigger than t, we took a big chunk off each weight to reduce s.

- Second example

$$\begin{aligned} s &= (-1.5) \times 1 + (-1.5) \times 3 + (-3) \times 1 \\ &= (-1.5) + (-4.5) + (-3) \\ &= -9 \end{aligned}$$

Then:

$$\begin{aligned} w_0 &\leftarrow w_0 + \alpha(t - s)x_0 \\ &\leftarrow -1.5 + 0.5 \times (0 - (-9)) \times 1 \\ &\leftarrow -1.5 + (4.5) \\ w_0 &= 3 \\ w_1 &\leftarrow w_1 + \alpha(t - s)x_1 \\ &\leftarrow -1.5 + 0.5 \times (0 - (-9)) \times 3 \\ &\leftarrow -1.5 + 13.5 \\ w_1 &= 12 \\ w_2 &\leftarrow w_2 + \alpha(t - s)x_2 \\ &\leftarrow -3 + 0.5 \times (0 - (-9)) \times 1 \\ &\leftarrow -3 + 4.5 \\ w_2 &= 1.5 \end{aligned}$$

So:

$$\begin{aligned} w_0 &= 3 \\ w_1 &= 12 \\ w_2 &= 1.5 \end{aligned}$$

and you can see why we typically keep α small — to reduce big swings in value when we use stochastic gradient descent.

- Third example

$$\begin{aligned} s &= 3 \times 1 + 12 \times 1 + 1.5 \times 1 \\ &= 3 + 12 + 1.5 \\ &= 16.5 \end{aligned}$$

Then:

$$\begin{aligned}
w_0 &\leftarrow 3 + 0.5 \times (1 - 16.5) \times 1 \\
&\leftarrow 3 - 7.75 \\
w_0 &= -4.75 \\
w_1 &\leftarrow 12 + 0.5 \times (1 - 16.5) \times 1 \\
&\leftarrow 12 - 7.75 \\
w_1 &= 4.25 \\
w_2 &\leftarrow 1.5 + 0.5 \times (1 - 16.5) \times 1 \\
&\leftarrow 1.5 - 7.75 \\
w_2 &= -6.25
\end{aligned}$$

So:

$$\begin{aligned}
w_0 &= -4.75 \\
w_1 &= 4.25 \\
w_2 &= -6.25
\end{aligned}$$

- Fourth example

  Finally:

$$\begin{aligned}
s &= (-4.75) \times 1 + 4.25 \times 2 + (-6.25) \times 0 \\
&= (-4.75) + 8.5 + 0 \\
&= 3.75
\end{aligned}$$

And:

$$\begin{aligned}
w_0 &\leftarrow -4.75 + 0.5 \times (0 - 3.75) \times 1 \\
&\leftarrow -4.75 + (-1.875) \\
w_0 &= -6.625 \\
w_1 &\leftarrow 4.25 + 0.5 \times (0 - 3.75) \times 2 \\
&\leftarrow 4.25 - 3.75 \\
w_1 &= 0.5 \\
w_2 &\leftarrow -6.25 + 0.5 \times (0 - 3.75) \times 0 \\
&\leftarrow -6.25 + 0 \\
w_2 &= -6.25
\end{aligned}$$

So:

$$\begin{aligned}
w_0 &= -6.625 \\
w_1 &= 0.5 \\
w_2 &= -6.25
\end{aligned}$$

3. For the generalised delta rule, with the sigmoid transfer function, we have the update rule:

$$w_i \leftarrow w_i + \alpha(t - g(s))g(s)(1 - g(s))x_i$$

where:

$$g(s) = \frac{1}{1 + e^{-s}}$$

So, running through the example using the generalised delta rule, we have:

- First example

$$\begin{aligned} s &= \sum w_i x_i \\ &= 1 \times 1 + 1 \times 1 + 2 \times 2 \\ &= 6 \\ g(s) &= 0.998 \end{aligned}$$

Since $s \geq 0$, $g(s) \approx 1$.
Since $t = 1$ we have:

$$\begin{aligned} w_0 &\leftarrow w_0 + \alpha(t - g(s))g(s)(1 - g(s))x_i \\ &\leftarrow 1 + 0.5 \times (1 - 0.998) \times 0.998 \times (1 - 0.998) \times 1 \\ w_0 &= 1.000002 \end{aligned}$$

So, although $g(s) \approx t$ the weights increase, but only by a very small amount. Similarly:

$$\begin{aligned} w_1 &\leftarrow 1 + 0.5 \times (1 - 0.998) \times 0.998 \times (1 - 0.998) \times 1 \\ w_1 &= 1.000002 \\ w_2 &\leftarrow 2 + 0.5 \times (1 - 0.998) \times 0.998 \times (1 - 0.998) \times 2 \\ w_2 &= 2.000004 \end{aligned}$$

Thus all the weights increase slightly to give:

$$\begin{aligned} w_0 &= 1.000002 \\ w_1 &= 1.000002 \\ w_2 &= 2.000004 \end{aligned}$$

- Second example.

$$\begin{aligned} s &= 1.000002 \times 1 + 1.000002 \times 3 + 2.000004 \times 1 \\ &= 6.000012 \\ g(s) &= 0.998 \end{aligned}$$

(Here we are using the rounded values of the weights).
So:

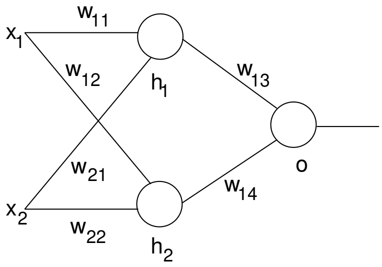
$$\begin{aligned} w_0 &\leftarrow 1.000002 + 0.5 \times (0 - 0.998) \times 0.998 \times (1 - 0.998) \times 1 \\ w_0 &= 0.999 \\ w_1 &\leftarrow 1.000002 + 0.5 \times (0 - 0.998) \times 0.998 \times (1 - 0.998) \times 3 \\ w_1 &= 0.997 \\ w_2 &\leftarrow 2.000004 + 0.5 \times (0 - 0.998) \times 0.998 \times (1 - 0.998) \times 1 \\ w_2 &= 1.999 \end{aligned}$$

and:

$$\begin{aligned} w_0 &= 0.999 \\ w_1 &= 0.997 \\ w_2 &= 1.999 \end{aligned}$$

and all the weights have decreased, with the biggest decrease being in w_1 because x_1 was the largest input value.

4. What we will do here is to derive the expressions for the various weight updates, taking the general solution given in the lecture and creating the specific updates for this example:



All of our training examples will have values for x_1 and x_2, and a value y (adopting the notation we have been using for `scikit-learn` in the practicals) which is the target value.

Then:

$$Error = y - g(o)$$

where $g(\cdot)$ is the sigmoid function. Then we have:

$$\Delta = (y - g(o))g(o)(1 - g(o))$$

and so the update rules for the weights into the output unit are:

$$\begin{aligned} w_{13} &\leftarrow w_{13} + \alpha\Delta g(h_1) \\ w_{14} &\leftarrow w_{14} + \alpha\Delta g(h_2) \end{aligned}$$

where h_1 is the weighted sum of the inputs using w_{11} and w_{12} and h_2 is the weighted sum of the inputs for the bottom hidden unit.

We also have to consider the bias weight of the output unit. If we call this w_o, then we update it with:

$$w_o \leftarrow w_o + \alpha\Delta 1$$

Then let's consider the update of the weights into h_1. We have:

$$\Delta_{h1} = g(h_1)(1 - g(h_1))w_{13}\Delta$$

and, then updates:

$$\begin{aligned} w_{h1} &\leftarrow w_{h1} + \alpha\Delta_{h1} 1 \\ w_{11} &\leftarrow w_{11} + \alpha\Delta_{h1} x_1 \\ w_{21} &\leftarrow w_{21} + \alpha\Delta_{h1} x_2 \end{aligned}$$

where w_{h1} is the bias weight for h_1. Similar updates are easily derived for h_2.

Version list

- Version 1.0, January 30th 2020.
- Version 1.1, February 4th 2021.