task_8.3

Machine Learning (WiSe 2025/2026)

Author: Suvansh Shukla
Matriculation No. 256245

Assignment 8 Task 3

Here's a table with the with the input values:

	a	b	t
Example 1	1	0	1
Example 2	0	1	0

given is that all weights are initialized with 0.1 and the learning rate η is 0.3

Formula for input to the hidden layer (here it is Sigmoid function):

$$z_m = w_{a,m}.\,a + w_{b,m}.\,b + w_{1,m}.1$$

Formula for output of a Sigmoid function:

$$o_m = g(z_m) = \frac{1}{1 + e^{-z}}$$

where, z is the input value and e is Euler's number (≈ 2.718)

Calculation of network output for Example 1

Calculation of the processed input to the hidden unit using the formula:

$$o_1 = 0.1(1) + 0.1(0) + 0.1(1) = 0.2$$

Calculating the output of the hidden unit:

$$\frac{1}{1 + (2.718^{-0.2})} = 0.8$$

This obtained output o_1 is used to calculate the input for the output Sigmoid unit.

To calculate the input for the output Sigmoid unit we use the input function formula again:

$$z_n = w_{m,n}.\,o_m + w_{1,n}.1 z_n = (0.1)(0.8) + 0.1(1) \approx 0.18$$

Total final output by the entire network (hidden Sigmoid + output Sigmoid):

$$o_n = g(z_n) = \frac{1}{1 + e^{-0.18}} \approx 0.87$$

Calculation of Backpropagation Error (methodology used in Example 1 & 2)

We have the net output of the network (consisting of hidden sigmoid and output sigmoid), we compare this to the target value and calculate the error. Then we back-propagate the adjustment.

Using the Binary Cross-Entropy (BCE) Loss Function:

$$L(y, \hat{y}) = -[ylog(\hat{y}) + (1-y)log(1-\hat{y})]$$

Where,

- y: true value
- $\hat{y}$: predicted value

We cannot simply plug values in this fomula and calculate the error. We need to get the derivative of the loss function w.r.t the net input of the output sigmoid unit, i.e. $\frac{\partial L}{\partial z_n}$, this is also known as error signal $_n$.

In our case of using a sigmoid output unit and BSE loss function, it becomes:

$$_n = \frac{\partial L}{\partial z_n} = \hat{y} - y = o_n - t_n = o_n - t = 0.87 - 1 = -0.1$$

Calculation of Weight change for Example 1

Now that we have an error, we can use it to update our weights. This requires calculating the change in weights for both examples then using their sum for the calculation of the total update for each weight. There are two weight change formulae, one for output unit and one for hidden unit.

Change rule for output unit:

$$\triangle w_{j,n} = \eta . _n . o_j$$

Change rule for hidden unit:

$$\triangle w_{i,m} = \eta . _m . x_i$$

where, $_m$ is the backward propagated error from the output unit. This is calculated like so:

$$_m = o_m(1-o_m) . w_{m,n} . _n$$

We start by calculating the change for the weights of the output unit

$$\triangle w_{1,n} = (0.).(-0.1).(1) = -0.18\triangle w_{m,n} = (0.).(-0.1).(0.8) = -0.071$$

Then we calculate the propagated error for use in calculating change in weights of the hidden unit

$$_m = (0.8)(1-0.8)\ (0.1)\ (-0.1) = -0.011$$

Calculating the change of weights for the hidden unit:

$$\triangle w_{a,m} = \eta . _m . a = (0.)\ (-0.011)\ (1) = -0.00\triangle w_{b,m} = \eta . _m . b = (0.)\ (-0.011)\ (0) = 0\triangle w_{1,m} = \eta . _m . 1 = (0.)\ (-0.011)\ (1) =$$

Calculation of network output for Example 2

Using the processed input to the hidden unit:

$$z_m = 0.1(0) + 0.1(1) + 0.1(1) = 0.2$$

This input is used to calculate the output of the hidden unit:

$$o_m = g(z_m) = \frac{1}{1 + (2.718^{-0.2})} = 0.8$$

This is one of the inputs fed into the output sigmoid function in this second example. The output of the network includes the static weight and input (bias) as well, i.e. $w_{1,n}$. Therefore the net input received by the output unit is:

$$z_n = (0.1)(0.8) + (0.1)(1) = 0.18$$

The final network output for Example 2 is:

$$o_n = g(z_n) = \frac{1}{1 + e^{-0.18}} \approx 0.87$$

Calculation of Backpropagation of Example 2

We start by taking the calculating the error of the ouput of example 2, which is:

$$_n = o_n - t = 0.87 - 0 = 0.87$$

Calculation of weight change for Example 2

This error is used in calculating weight change for the output sigmoid function:

$$\triangle w_{1,n} = (0.).(-0.87).(1) = 0.11 \triangle w_{m,n} = (0.).(-0.87).(0.8) = 0.0888$$

Then we calculate the propagated error for use in calculating change in weights of the hidden unit

$$_m = (0.8)(1 - 0.8)\ (0.1)\ (-0.87) = 0.01$$

Calculating the change of weights for the hidden unit:

$$\triangle w_{a,m} = \eta._m.a = (0.)\ (0.01)\ (0) = 0 \triangle w_{b,m} = \eta._m.b = (0.)\ (0.01)\ (1) = 0.000 \triangle w_{1,m} = \eta._m.1 = (0.)\ (0.01)\ (1) = 0.000$$

Final weight updated values

weight	initial value	weight updates Ex 1	weight updates Ex 2	total	final
$w_{m,n}$	0.1	-0.0761	0.0888	0.0127	0.1127
$w_{1,n}$	0.1	-0.1384	0.1616	0.0232	0.1232
$w_{a,m}$	0.1	-0.0034	0	-0.0034	0.0966
$w_{b,m}$	0.1	0	0.0040	0.0040	0.1040
$w_{1,m}$	0.1	-0.0034	0.0040	0.0006	0.1006