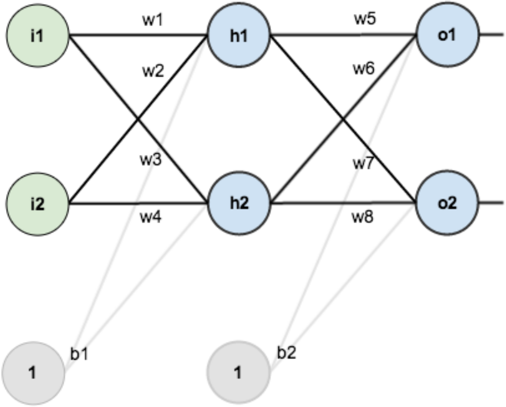
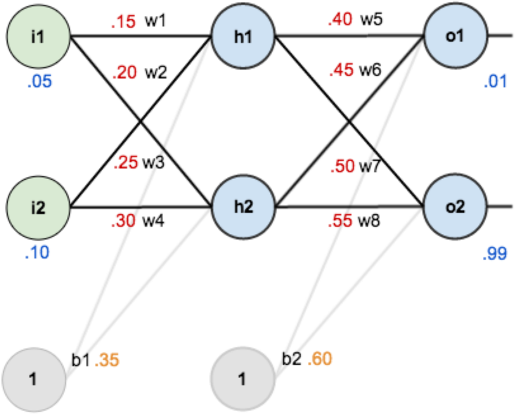
Backpropagation: Numerical Example

Backpropagation: Numerical example



source: https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/

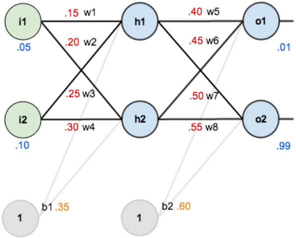
Backpropagation: Numerical example



source: https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/

Backpropagation: Numerical example

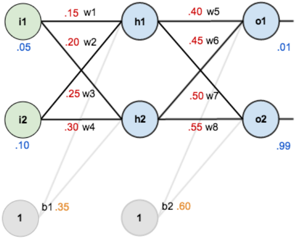
- two inputs
- two hidden neurons
- two output neurons
- parameters of the model:
  - Weight matrices W_1, W_2,
  - bias b_1, b_2



Backpropagation: Forward Pass

hidden layer neurons

- at $\mathbf{h} \in \mathbb{R}^2$, we compute:
  - $net_h = W_1^T \mathbf{x}$
  - $out_h = \sigma(W_1^T \mathbf{x})$ (activation)
- $net_{h1} = w_1 \times i_1 + w_2 \times i_2 + b_1 \times 1$

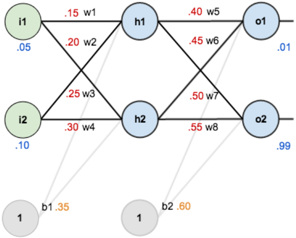


- $net_{h1} = 0.15 \times 0.005 + 0.2 \times 0.1 + 0.35 \times 1 = 0.3775$
- $out_{h1} = \sigma(net_{h1}) = \frac{1}{1+e^{-net_{h1}}} = \frac{1}{1+e^{0.3775}} = 0.59326992$
- Similarly we get $out_{h2} = 0.596884378$

Backpropagation: Forward Pass

output layer neurons

- at $\mathbf{o} \in \mathbb{R}^2$, we compute:
    - $net_o = W_2^T \mathbf{h}$
    - $out_o = \sigma(W_2^T \mathbf{h})$
- $net_{o1} =$
  $w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$



- $out_{o1} = 0.4 \times 0.5932 + 0.45 \times 0.5968 + 0.6 \times 1 = 1.1059$
- $out_{o1} = \sigma(net_{o1}) = \frac{1}{1+e^{-net_{o1}}} = \frac{1}{1+e^{1.1059}} = 0.7513$
- Similarly we get $out_{o2} = 0.7729$

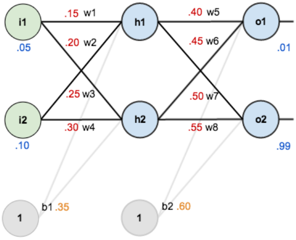
Backpropagation: Total Error

- $E_{total} = \sum \frac{1}{2}(target - output)^2$
- error at $o1$:

  $E_{o1} = \frac{1}{2}(target_{o1} - output_{o1})^2$

  $E_{o1} = \frac{1}{2}(0.01 - 0.7513)^2 = 0.2748$
- similarly $E_{o2} = 0.02356$
- Total Error: $E_{total} = E_{o1} + E_{o2}$

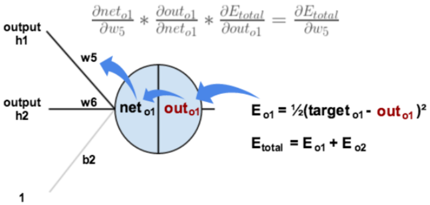
  $E_{total} = 0.2748 + 0.0235 = 0.2983$



Backpropagation: Backwards Pass

- update weights so that they cause actual output to be closer to target output
- specifically, we will consider $\frac{\partial E_{total}}{\partial w_5}$ (change in w_5 that affects total error):

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial w_5}$$



Backpropagation: Backwards Pass

1. **compute:** $\frac{\partial E_{total}}{\partial out_{o1}}$

- $E_{total} = \frac{1}{2}(target_{o1} - out_{o1})^2 + \frac{1}{2}(target_{o2} - out_{o2})^2$

  $\frac{\partial E_{total}}{\partial out_{o1}} = 2 \times \frac{1}{2}(target_{o1} - out_{o1})^{2-1} \times (-1) + 0$

  $\frac{\partial E_{total}}{\partial out_{o1}} = -(target_{o1} - out_{o1})$

  $\frac{\partial E_{total}}{\partial out_{o1}} = -(0.01 - 0.7513) = 0.7413$

Backpropagation: Backwards Pass

2. compute: $\frac{\partial out_{01}}{\partial net_{o1}}$

$$out_{o1} = \sigma(net_{o1}) = \frac{1}{1+e^{-net_{o1}}}$$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}(1 - out_{o1}) \text{ (see logistic regression lecture notes)}$$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = 0.7513(1 - 0.7513) = 0.1868$$

3. compute: $\frac{\partial net_{01}}{\partial w_5}$

$$net_{o1} = w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$$

$$\frac{\partial net_{01}}{\partial w_5} = out_{h1} = 0.5932$$

Backpropagation: Backwards Pass

Putting it together:

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial w_5}$$

$$\frac{\partial E_{total}}{\partial w_5} = 0.7413 \times 0.1868 \times 0.5932 = 0.0821$$

- To decrease the error, we subtract $\frac{\partial E_{total}}{\partial w_5}$ from current weight:

  $$w_5^+ = w_5 - \eta \frac{\partial E_{total}}{\partial w_5} = 0.4 - 0.5 \times 0.0821 = 0.3589$$

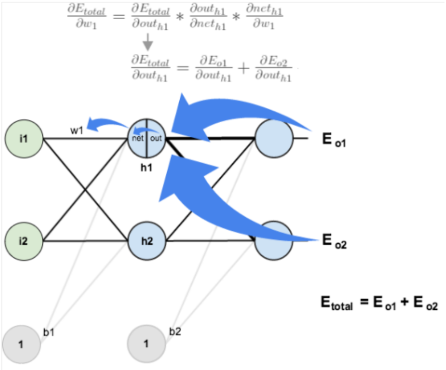
  Similarly:

  $$w_6^+ = 0.4086; w_7^+ = 0.51130; w_8^+ = 0.5613$$

Backpropagation: Backwards Pass

Hidden Layer:

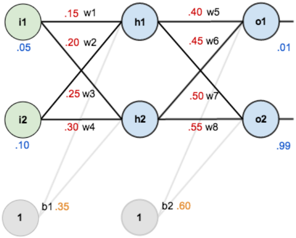
- compute w_1, w_2, w_3, w_4: $\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$



Backpropagation: Backwards Pass

$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$

$$\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial out_{h1}}$$



1. compute $\frac{\partial E_{o1}}{\partial out_{o1}}$

$$E_{o1} = \frac{1}{2}(target_{o1} - out_{o1})^2$$

$$\frac{\partial E_{o1}}{\partial out_{o1}} = -(target_{o1} - out_{o1}) = -(0.01 - 0.7513) = 0.7413$$

2. compute $\frac{\partial out_{o1}}{\partial net_{o1}}$

$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}(1 - out_{o1}) = 0.7513(1 - 0.7513) = 0.1868$$

Backpropagation: Backwards Pass

3. compute: $\frac{\partial net_{o1}}{\partial out_{h1}}$

$net_{o1} = w_5 \times out_{h1} + w_6 \times out_{h2} + b_2 \times 1$

$\frac{\partial net_{o1}}{\partial out_{h1}} = w_5 = 0.40$

Substituting for $\frac{\partial E_{o1}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{o1}} \times \frac{\partial out_{o1}}{\partial net_{o1}} \times \frac{\partial net_{o1}}{\partial out_{h1}}$, we get

$\frac{\partial E_{o1}}{\partial out_{h1}} = 0.7413 \times 0.1868 \times 0.40 = 0.0553$

Following the same process: $\frac{\partial E_{o2}}{\partial out_{h1}} = -0.0190$

Thus, $\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}} = 0.0553 + (-0.0190) = 0.0363$

Backpropagation: Backwards Pass

- We had: $\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$
- We computed $\frac{\partial E_{total}}{\partial out_{h1}}$, we need to compute: $\frac{\partial out_{h1}}{\partial net_{h1}}; \frac{\partial net_{h1}}{\partial w_1}$

$$out_{h1} = \sigma(net_{h1}) = \frac{1}{1+e^{-net_{h1}}}$$

$$\frac{\partial out_{h1}}{\partial net_{h1}} = out_{h1}(1 - out_{h1})$$

$$\frac{\partial out_{h1}}{\partial net_{h1}} = 0.5932(1 - 0.5932) = 0.2413$$

Backpropagation: Backwards Pass

- We had: $\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} \times \frac{\partial out_{h1}}{\partial net_{h1}} \times \frac{\partial net_{h1}}{\partial w_1}$
- We computed $\frac{\partial E_{total}}{\partial out_{h1}}$; $\frac{\partial out_{h1}}{\partial net_{h1}}$, we need to compute: $\frac{\partial net_{h1}}{\partial w_1}$

  $net_{h1} = w_1 \times i_1 + w_2 \times i_2 + b_1 \times 1$

  $\frac{\partial net_{h1}}{\partial w_1} = i_1 = 0.05$
- Putting it together:

  $\frac{\partial E_{total}}{\partial w_1} = 0.0363 \times 0.2413 \times 0.05 = 0.00043$
- update w_1:
  $w_1^+ = w_1 - \eta \frac{\partial E_{total}}{\partial w_1} = 0.15 - 0.5 \times 0.000438 = 0.1497$

  Similarly: $w_2^+ = 0.1995$; $w_3^+ = 0.2497$; $w_4^+ = 0.2995$

Backpropagation: Backwards Pass

- with inputs 0.05 and 0.1 - initial error 0.298371109
- after first round of backpropagation - error: 0.291027924
- after repeating the process 10,000 times - error: 0.0000351085
- the output neurons generates:
  - 0.015912196 (vs 0.01 target)
  - 0.984065734 (vs 0.99 target)

.