NNDL:
Problem Set #2

Table of Contents

Problem 1 2

Problem 2 5

Problem 3 7

Part A: Forward Propagation 7

Part B: Determine the Error 8

Part C: Calculate the Gradients 8

Part D: Update the Weights 15

Part A Redux: Forward Propagation 16

Part B Redux: Determine the Error 17

Alejandro C Parra Garcia

Problem 1

Sigmoid function, as define by Max Bramer is:

$$sigmoid(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

Now let's start the derivation:

$$\frac{d\,\sigma(x)}{d\,x} = \frac{d}{d\,x}\frac{1}{1 + e^{-x}}$$

Now we use the Reciprocal Rule, and we get the next result:

$$\frac{d}{d\,x}\frac{1}{u(x)} = \frac{-u'(x)}{u(x)^2}$$

$$\frac{d}{d\,x}\frac{1}{1 + e^{-x}} = \frac{-\frac{d}{d\,x}(1 + e^{-x})}{(1 + e^{-x})^2} = -(1 + e^{-x})^{-2} * \frac{d}{d\,x}(1 + e^{-x})$$

We have derived part of the equation, now we derivate the other part. Since the other part is a sum, we need to use the sum rule, and do the derivative of its part. The derivative of a constant is 0:

$$\frac{d}{d\,x}(1 + e^{-x}) = \frac{d}{d\,x}(1) + \frac{d}{d\,x}(e^{-x}) = 0 + \frac{d}{d\,x}(e^{-x}) = \frac{d}{d\,x}(e^{-x})$$

Now we do use the exponential rule:

$$\frac{d}{d\,x}e^{u(x)} = e^{u(x)} * \frac{d}{d\,x}u(x)$$

$$\frac{d}{d\,x}(e^{-x}) = e^{-x} * \frac{d}{d\,x}(-x)$$

We need to do one last derivative, using the linearity rule:

$$\frac{d}{d\,x}c * x = c$$

$$\frac{d}{d\,x}(-x) = \frac{d}{d\,x}(-1 * x) = -1$$

Lastly, we join all the derivatives

$$\frac{d}{d\,x}\frac{1}{1 + e^{-x}} = -(1 + e^{-x})^{-2} * \frac{d}{d\,x}(1 + e^{-x})$$

$$\frac{d}{d\,x}(1 + e^{-x}) = \frac{d}{d\,x}(e^{-x})$$

$$\frac{d}{d\,x}(e^{-x}) = e^{-x} * \frac{d}{d\,x}(-x)$$

$$\frac{d}{d\,x}(-x) = -1$$

Giving us the final derivate as:

$$\frac{d}{d\,x}\frac{1}{1+\ e^{-x}}=\ -(1+\ e^{-x})^{-2}*\ e^{-x}*-1$$

If we now rewrite the formula as:

$$\frac{d}{d\,x}\frac{1}{1+\ e^{-x}}=\ (1+\ e^{-x})^{-2}*\ e^{-x}=\frac{e^{-x}}{(1+\ e^{-x})^{2}}$$

The question we to show that the derivative of the sigmoid function is given by the next formula:

$$\frac{d}{d\,x}\sigma(x)=\sigma(x)(1-\sigma(x))$$

Right now, we have seen that the derivative of the sigmoid function is equal to the next formula, so now we need to do some transformation:

$$\frac{d}{d\,x}\ \sigma(x)=\frac{e^{-x}}{(1+\ e^{-x})^{2}}$$

So, we need to proof that:

$$\sigma(x)\big(1-\sigma(x)\big)=\frac{e^{-x}}{(1+\ e^{-x})^{2}}$$

Let's transform the $\sigma(x)\big(1-\sigma(x)\big)$ and see where we go.

First let's substitute the $\sigma(x)$ with the original sigmoid function:

$$\frac{d}{d\,x}\sigma(x)=\sigma(x)\big(1-\sigma(x)\big)=\frac{1}{1\ +\ e^{-x}}\ *\ (1\ -\ \frac{1}{1\ +\ e^{-x}})$$

Now let's multiply its terms:

$$\frac{d}{d\,x}\sigma(x)=\frac{1}{1\ +\ e^{-x}}*\left(1\ -\ \frac{1}{1\ +\ e^{-x}}\right)=\frac{1}{1\ +\ e^{-x}}-\frac{1}{1\ +\ e^{-x}}*\frac{1}{1\ +\ e^{-x}}$$

$$\frac{1}{1\ +\ e^{-x}}-\frac{1}{1\ +\ e^{-x}}*\frac{1}{1\ +\ e^{-x}}=\frac{1}{1\ +\ e^{-x}}-\frac{1}{(1\ +\ e^{-x})*(1\ +\ e^{-x})}$$

$$\frac{1}{1\ +\ e^{-x}}-\frac{1}{(1\ +\ e^{-x})*(1\ +\ e^{-x})}=\frac{1}{1\ +\ e^{-x}}-\frac{1}{(1\ +\ e^{-x})^{2}}$$

Now let's do the subtraction:

$$\frac{1}{1\ +\ e^{-x}}-\frac{1}{(1\ +\ e^{-x})^{2}}=\frac{1\ +\ e^{-x}}{(1\ +\ e^{-x})^{2}}-\frac{1}{(1\ +\ e^{-x})^{2}}$$

$$\frac{1\ +\ e^{-x}}{(1\ +\ e^{-x})^{2}}-\frac{1}{(1\ +\ e^{-x})^{2}}=\frac{1\ +\ e^{-x}\ -\ 1}{(1\ +\ e^{-x})^{2}}=\frac{e^{-x}}{(1\ +\ e^{-x})^{2}}$$

$$\frac{d}{d\,x}\sigma(x)=\sigma(x)\big(1-\sigma(x)\big)=\frac{e^{-x}}{(1+\ e^{-x})^{2}}$$

As we can see, doing some simple transformation to the equation, we have arrived at the original derivative, that we have calculated before.

Problem 2

First lest write the notation I'm going to use, based on Bramer's notation:

Inp_i: ith input node

hid_j: jth hidden node

out_k: kth output node

$Whid_j$: Weighted sum value for hid_j

$Thid_j$: Transformed value for hid_j

$Wout_k$: Weighted sum value for out_k

$Tout_k$: Transformed value for out_k

$bias_H$: Bias weight of hidden layer

$bias_O$: Bias weight of output layer

w_{ij}: Weight between input node i and hidden node j

W_{jk}: Weight between hidden node j and ouput node k

Now we can write the equation regarding de different layers, that way we arrived at the following set of equations. With 3 input nodes, 3 hidden nodes, and 2 output nodes:

$$Whid_{j\,(1\leq j\leq 3)} = w_{1j} * inp_1 + w_{2j} * inp_2 + w_{3j} * inp_3 + bias_H$$

$$Whid_j = Thid_j$$

$$Wout_{k(1\leq k\leq 2)} = W_{1k} * Thid_1 + W_{2k} * Thid_2 + W_{3k} * Thid_3 + bias_O$$

$$Wout_k = Tout_k$$

Now we need to clear for $Tout_1$ and $Tout_2$.

First, we can simplify the calculation by groping all the adding parts:

$$Whid_{j\,(1\leq j\leq 3)} = \sum_{i=1}^{3}(w_{ij} * inp_i) + bias_H$$

$$Wout_{k(1\leq k\leq 2)} = \sum_{j=1}^{3}(W_{jk} * Thid_j) + bias_O$$

Now we start to clear the equation:

$$Tout_{k\,(1\leq k\leq 2)} = Wout_k = \sum_{j=1}^{3}(W_{jk} * Thid_j) + bias_O$$

$$Tout_{k\ (1\le k\le 2)} = \sum_{j=1}^{3}(W_{jk} * Thid_j) + bias_O = \sum_{j=1}^{3}(W_{jk} * Whid_j) + bias_O$$

$$Tout_{k\ (1\le k\le 2)} = \sum_{j=1}^{3}(W_{jk} * Whid_j) + bias_O = \sum_{j=1}^{3}\left(W_{jk} * \left[\sum_{i=1}^{3}(w_{ij} * inp_i) + bias_H\right]\right) + bias_O$$

$$Tout_{k\ (1\le k\le 2)} = \sum_{j=1}^{3}\left(W_{jk} * \left[\sum_{i=1}^{3}(w_{ij} * inp_i) + bias_H\right]\right) + bias_O$$

As we can see the $Tout_1$ and $Tout_2$, as a linear combination of the input's values, the weights, and the biases.

$$Tout_1 = \sum_{j=1}^{3}\left(W_{j1} * \left[\sum_{i=1}^{3}(w_{ij} * inp_i) + bias_H\right]\right) + bias_O$$

$$Tout_1 = \sum_{j=1}^{3}(W_{j1} * [w_{1j} * inp_1 + w_{2j} * inp_2 + w_{3j} * inp_3 + bias_H]) + bias_O$$

$$\begin{aligned} Tout_1 = \quad & (W_{11} * [w_{11} * inp_1 + w_{21} * inp_2 + w_{31} * inp_3 + bias_H]) \\ & + (W_{21} * [w_{12} * inp_1 + w_{22} * inp_2 + w_{32} * inp_3 + bias_H]) \\ & + (W_{31} * [w_{13} * inp_1 + w_{23} * inp_2 + w_{33} * inp_3 + bias_H]) \\ & + bias_O \end{aligned}$$

$$Tout_2 = \sum_{j=1}^{3}\left(W_{j2} * \left[\sum_{i=1}^{3}(w_{ij} * inp_i) + bias_H\right]\right) + bias_O$$

$$Tout_2 = \sum_{j=1}^{3}(W_{j2} * [w_{1j} * inp_1 + w_{2j} * inp_2 + w_{3j} * inp_3 + bias_H]) + bias_O$$

$$\begin{aligned} Tout_2 = \quad & (W_{12} * [w_{11} * inp_1 + w_{21} * inp_2 + w_{31} * inp_3 + bias_H]) \\ & + (W_{22} * [w_{12} * inp_1 + w_{22} * inp_2 + w_{32} * inp_3 + bias_H]) \\ & + (W_{32} * [w_{13} * inp_1 + w_{23} * inp_2 + w_{33} * inp_3 + bias_H]) \\ & + bias_O \end{aligned}$$

Problem 3

Disclaimer: (Calculations are done with the original values, while the presented values on this document are rounded)

Part A: Forward Propagation

First let's calculate the hidden layer values:

$$Whid_{j\,(1\le j\le 3)} = w_{1j} * inp_1 + w_{2j} * inp_2 + w_{3j} * inp_3 + bias_H$$

$$Whid_1 = 0.1 * 4 + (-0.4) * 8 + (-0.1) * 6 + 0.2 = -3.2$$

$$Whid_2 = 0.3 * 4 + 0.1 * 8 + (-0.2) * 6 + 0.2 = 1$$

$$Whid_3 = (-0.2) * 4 + 0.2 * 8 + 0.4 * 6 + 0.2 = 3.4$$

Then let's apply the sigmoid function:

$$Thid_j = \sigma\left(Whid_j\right)$$

$$Thid_1 = \sigma(-3.2) = \frac{1}{1 + e^{-(-3.2)}} = 0.03916572$$

$$Thid_2 = \sigma(1) = \frac{1}{1 + e^{-(1)}} = 0.73105858$$

$$Thid_3 = \sigma(3.4) = \frac{1}{1 + e^{-(3.4)}} = 0.96770453$$

We now know the hidden layer values. Now we need to do the same for the output

$$Wout_{k(1\le k\le 2)} = W_{1k} * Thid_1 + W_{2k} * Thid_2 + W_{3k} * Thid_3 + bias_O$$

$$Wout_1 = 0.5 * 0.03916572 + (-0.3) * 0.73105858 + 0.2 * 0.96770453 + 0.3 = 0.29380619$$

$$Wout_2 = 0.2 * 0.03916572 + (-0.3) * 0.73105858 + 0.1 * 0.96770453 + 0.3 = 0.18528602$$

Now let's apply the sigmoid function:

$$Tout_k = \sigma(Wout_k)$$

$$Tout_1 = \sigma(0.29380619) = \frac{1}{1 + e^{-(0.29380619)}} = 0.572927696$$

$$Tout_2 = \sigma(0.18528602) = \frac{1}{1 + e^{-(0.18528602)}} = 0.546189438$$

Input 1	Input 2	Input 3	Hidden 1	Hidden 2	Hidden 3	Output 1	Output 2
4	8	6	0.039165	0.731058	0.967704	0.572927	0.546189

Part B: Determine the Error

Let's calculate the error between the output values and the target values

$$E = 0.5 * \sum_{k} (targ_k - Tout_k)^2$$

$$E = 0.5 * \sum_{k=1}^{2} (targ_k - Tout_k)^2 = 0.5 * [(targ_1 - Tout_1)^2 + (targ_2 - Tout_2)^2]$$

$$E = 0.5 * [(0.65 - 0.572927)^2 + (0.4 - 0.546189)^2 = 0.01365575$$

Part C: Calculate the Gradients

1. **Show that for the weights from hidden to output nodes, the gradient is given by:**

$$g(E, W_{jk}) = (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * Thid_j$$

First let's do: g(E,Tout$_k$):

$$E = 0.5 * \sum_{k} (targ_k - Tout_k)^2$$

$$\frac{dE}{dTout_k} 0.5 * \sum_{k} (targ_k - Tout_k)^2 = 0.5 * \frac{dE}{dTout_k} \sum_{k} (targ_k - Tout_k)^2$$

$$0.5 * \frac{dE}{dTout_k} \sum_{k} (targ_k - Tout_k)^2 = 0.5 * \frac{dE}{dTout_k} (targ_k - Tout_k)^2$$

$$0.5 * \frac{dE}{dTout_k} (targ_k - Tout_k)^2 = 0.5 * \left[2 * (targ_k - Tout_k) * \frac{dE}{dTout_k} (targ_k - Tout_k)\right]$$

$$0.5 * \left[2 * (targ_k - Tout_k) * \frac{dE}{dTout_k} (targ_k - Tout_k)\right]$$
$$= (targ_k - Tout_k) * \frac{dE}{dTout_k} (targ_k - Tout_k)$$

$$\frac{dE}{dTout_k} (targ_k - Tout_k) = \frac{dE}{dTout_k} targ_k - \frac{dE}{dTout_k} Tout_k$$

$$\frac{dE}{dTout_k} targ_k - \frac{dE}{dTout_k} Tout_k = 0 - 1$$

$$\frac{dE}{dTout_k} 0.5 * \sum_{k} (targ_k - Tout_k)^2 = (targ_k - Tout_k) * (0 - 1) = Tout_k - targ_k$$

$$g(E, Tout_k) = Tout_k - targ_k$$

Now let's do: g($Tout_k$, $Wout_k$): (We have already calculated the derivative of the sigmoid function)

$$Tout_k = \sigma(Wout_k) = \frac{1}{(1 + e^{-Wout_k})}$$

$$\frac{dTout_k}{dWout_k}\sigma(Wout_k) = \sigma(Wout_k)(1 - \sigma(Wout_k))$$

$$\frac{dTout_k}{dWout_k}\sigma(Wout_k) = Tout_k(1 - Tout_k)$$

$$g(Tout_k, Wout_k) = Tout_k(1 - Tout_k)$$

Now let's do: g($Wout_k$,W_{jk}):

$$Wout_k = \sum_j (Thid_j * W_{jk}) + bias_O$$

$$\frac{dWout_k}{dW_{jk}}\sum_j (Thid_j * W_{jk}) + bias_O = \frac{dWout_k}{dW_{jk}}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dW_{jk}} bias_O$$

$$\frac{dWout_k}{dW_{jk}}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dW_{jk}} bias_O = \frac{dWout_k}{dW_{jk}}(Thid_j * W_{jk}) + 0$$

$$\frac{dWout_k}{dW_{jk}}(Thid_j * W_{jk}) = Thid_j$$

$$\frac{dWout_k}{dW_{jk}}\sum_j (Thid_j * W_{jk}) + bias_O = Thid_j$$

$$g(Wout_k, W_{jk}) = Thid_j$$

Now we have done all the derivative that we need, now we use the chain rule to combine them:

$$g(E, W_{jk}) = g\left(E, g(Tout_k, g(Wout_k, W_{jk})\right)$$

$$g\left(E, g(Tout_k, g(Wout_k, W_{jk})\right) = (Tout_k - targ_k) * \left(Tout_k(1 - Tout_k)\right) * \left(Thid_j\right)$$

As we can see we have arrived at the same equation.

2. **Show that for the bias term for the output nodes, the gradient is given by:**

$$g(E, bias_O) = \sum_k [(Tout_k - targ_k) * Tout_k * (1 - Tout_k)]$$

We already have:

$$g(E, Tout_k) = Tout_k - targ_k$$

$$g(Tout_k, Wout_k) = Tout_k(1 - Tout_k)$$

$$g(Wout_k, W_{jk}) = Thid_j$$

First let's do: g($Wout_k$,$bias_O$):

$$Wout_k = \sum_j (Thid_j * W_{jk}) + bias_O$$

$$\frac{dWout_k}{dbias_O}\sum_j (Thid_j * W_{jk}) + bias_O = \frac{dWout_k}{dbias_O}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dbias_O} bias_O$$

$$\frac{dWout_k}{dbias_O}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dbias_O} bias_O = 0 + 1$$

$$\frac{dWout_k}{dbias_O}\sum_j (Thid_j * W_{jk}) + bias_O = 0 + 1$$

$$g(Wout_k, bias_O) = 1$$

Now we can apply the chain rule

$$g(E, bias_O) = g\big(E, g(Tout_k, g(Wout_k, bias_O))\big)$$

$$g\big(E, g(Tout_k, g(Wout_k, bias_O))\big) = \sum_k (Tout_k - \ targ_k) * \big(Tout_k(1 - Tout_k)\big) * (1)$$

As we can see we have arrived at the same equation.

3. **Show that for the weights from input to hidden nodes, the gradient is given by:**

$$g(E, w_{ij}) = \sum_k \big[(Tout_k - \ targ_k) * Tout_k(1 - Tout_k) * W_{jk}\big] * Thid_j * \big(1 - Thid_j\big) * inp_i$$

We already have:

$$g(E, Tout_k) = Tout_k - \ targ_k$$

$$g(Tout_k, Wout_k) = Tout_k(1 - Tout_k)$$

$$g\big(Wout_k, W_{jk}\big) = Thid_j$$

$$g(Wout_k, bias_O) = 1$$

First let's do g($Wout_k$,$Thid_j$):

$$Wout_k = \sum_j (Thid_j * W_{jk}) + bias_O$$

$$\frac{dWout_k}{dThid_j}\sum_j (Thid_j * W_{jk}) + bias_O = \frac{dWout_k}{dThid_j}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dThid_j} bias_O$$

$$\frac{dWout_k}{dThid_j}\sum_j (Thid_j * W_{jk}) + \frac{dWout_k}{dThid_j} bias_O = \frac{dWout_k}{dThid_j}(Thid_j * W_{jk}) + 0$$

$$\frac{dWout_k}{dThid_j}(Thid_j * W_{jk}) = W_{jk}$$

$$\frac{dWout_k}{dThid_j}\sum_j (Thid_j * W_{jk}) + bias_O = W_{jk}$$

$$g(Wout_k, Thid_j) = W_{jk}$$

Now we do the g($Thid_j$, $Whid_j$): (We have already calculated the derivative of the sigmoid function)

$$\mathrm{Thid}_j = \sigma(Whid_j) = \frac{1}{(1 + e^{-Whid_j})}$$

$$\frac{d\mathrm{Thid}_j}{dWhid_j}\sigma(Whid_j) = \sigma(Whid_j)\left(1 - \sigma(Whid_j)\right)$$

$$\frac{d\mathrm{Thid}_j}{dWhid_j}\sigma(Whid_j) = \mathrm{Thid}_j(1 - \mathrm{Thid}_j)$$

$$g(\mathrm{Thid}_j, Whid_j) = \mathrm{Thid}_j(1 - \mathrm{Thid}_j)$$

Now we do g($Whid_j$,w_{ij}):

$$Whid_j = \sum_j (inp_i * w_{ij}) + bias_H$$

$$\frac{dWhid_i}{dw_{ij}}\sum_j (inp_i * w_{ij}) + bias_H = \frac{dWhid_i}{dw_{ij}}\sum_j (inp_i * w_{ij}) + \frac{dWhid_i}{dw_{ij}} bias_H$$

$$\frac{dWhid_i}{dw_{ij}}\sum_j (inp_i * w_{ij}) + \frac{dWhid_i}{dw_{ij}} bias_H = \frac{dWhid_i}{dw_{ij}}(inp_i * w_{ij}) + 0$$

$$\frac{dWhid_i}{dw_{ij}}(inp_i * w_{ij}) = inp_i$$

$$\frac{dWhid_i}{dw_{ij}}\sum_j (inp_i * w_{ij}) + bias_H = inp_i$$

$$g(Whid_j, w_{ij}) = inp_i$$

Now we have:

$$g(E, Tout_k) = Tout_k - targ_k$$

$$g(Tout_k, Wout_k) = Tout_k(1 - Tout_k)$$

$$g(Wout_k, W_{jk}) = Thid_j$$

$$g(Wout_k, bias_O) = 1$$

$$g(Wout_k, Thid_j) = W_{jk}$$

$$g\left(\mathrm{Thidj}_j, Whid_j\right) = \mathrm{Thid}_j\left(1 - \mathrm{Thid}_j\right)$$

$$g(Whid_j, w_{ij}) = inp_i$$

Now we can apply the chain rule

$$g(E, w_{ij}) = g\left(E, g\left(Tout_k, g\left(Wout_k, g\left(\mathrm{Thid}_j, g(Whid_j, w_{ij})\right)\right)\right)\right)$$

$$g\left(E, g\left(Tout_k, g\left(Wout_k, g\left(\mathrm{Thidj}_j, g(Whid_j, w_{ij})\right)\right)\right)\right)$$
$$= \sum_k [(Tout_k - targ_k) * (Tout_k(1 - Tout_k)) * W_{jk} * \left(\mathrm{Thid}_j\left(1 - \mathrm{Thid}_j\right)\right) * inp_i]$$

As we can see we have arrived at the same equation.

4. Show that for the bias term for the hidden nodes, the gradient is given by:

$$(E, bias_H) = \sum_k [(Tout_k - targ_k) * Tout_k(1 - Tout_k) * W_{jk}] * \mathrm{Thid}_j\left(1 - \mathrm{Thid}_j\right)$$

We already have:

$$g(E, Tout_k) = Tout_k - targ_k$$

$$g(Tout_k, Wout_k) = Tout_k(1 - Tout_k)$$

$$g\left(Wout_k, W_{jk}\right) = Thid_j$$

$$g(Wout_k, bias_O) = 1$$

$$g(Wout_k, Thid_j) = W_{jk}$$

$$g\left(\mathrm{Thidj}_j, Whid_j\right) = \mathrm{Thid}_j\left(1 - \mathrm{Thid}_j\right)$$

$$g(Whid_j, w_{ij}) = inp_i$$

First we need g(Whid$_j$,bias$_H$):

$$Whid_j = \sum_j \left(inp_i * w_{ij}\right) + bias_H$$

$$\frac{dWhid_i}{dbias_H}\sum_j \left(inp_i * w_{ij}\right) + bias_H = \frac{dWhid_i}{dbias_H}\sum_j \left(inp_i * w_{ij}\right) + \frac{dWhid_i}{dbias_H}\ bias_H$$

$$\frac{dWhid_i}{dbias_H}\sum_j \left(inp_i * w_{ij}\right) + \frac{dWhid_i}{dbias_H}\ bias_H = 0 + 1$$

$$g(Whid_j, bias_H) = 1$$

Now we can apply the chain rule

$$g(E, w_{ij}) = g\left(E, g\left(Tout_k, g\left(Wout_k, g(\text{Thid}_j, g(Whid_j, bias_H))\right)\right)\right)$$

$$g\left(E, g\left(Tout_k, g\left(Wout_k, g(\text{Thid}_j, g(Whid_j, bias_H))\right)\right)\right) = \sum_k \left[(Tout_k - targ_k) * \left(Tout_k(1 - Tout_k)\right) * W_{jk} * \left(\text{Thid}_j(1 - \text{Thid}_j)\right) * (1)\right]$$

As we can see we have arrived at the same equation.

We can calculate the gradient:

$$g(E, W_{jk}) = (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * Thid_j$$

$$g(E, W_{11}) = (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * Thid_1 = (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * 0.0396 = -0.000738594$$

$$g(E, W_{12}) = (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * Thid_1 = (0.5461 - 0.4) * 0.5461 * (1 - 0.5461) * 0.0396 = 0.001419188$$

$$g(E, W_{21}) = (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * Thid_2 = (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * 0.7310 = -0.013786428$$

$$g(E, W_{22}) = (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * Thid_2 = (0.5461 - 0.4) * 0.5461 * (1 - 0.5461) * 0.7310 = 0.026490251$$

$$g(E, W_{31}) = (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * Thid_3 = (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * 0.9677 = -0.018249137$$

$$g(E, W_{32}) = (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * Thid_3 = (0.5461 - 0.4) * 0.5461 * (1 - 0.5461) * 0.9677 = 0.035065228$$

Now we do the same for the bias of the output layer:

$$g(E, bias_O) = \sum_k [(Tout_k - targ_k) * Tout_k * (1 - Tout_k)]$$

$$g(E, bias_O) = [(Tout_1 - targ_1) * Tout_1 * (1 - Tout_1)] + [(Tout_2 - targ_2) * Tout_2 * (1 - Tout_2)]$$

$$g(E, bias_O) = [(0.5729 - 0.65) * 0.5729 * (1 - 0.5729)] + [(0.5461 - 0.4) * 0.5461 * (1 - 0.5461)] = 0.017377299$$

link from node	Link to node	Gradient
Input 1	Hidden 1	
Input 1	Hidden 2	
Input 1	Hidden 3	
Input 2	Hidden 1	
Input 2	Hidden 2	
Input 2	Hidden 3	
Input 3	Hidden 1	
Input 3	Hidden 2	

Input 3	Hidden 3	
Hidden 1	Out 1	-0.00073859
Hidden 1	Out 2	0.00141919
Hidden 2	Out 1	-0.01378643
Hidden 2	Out 2	0.02649025
Hidden 3	Out 1	-0.01824914
Hidden 3	Out 2	0.03506523
	Layer	(Bias)
	Hidden	
	Output	0.017377299

Now lets calculate the other weights, for that it will be better to first calculate g(E,Thid$_j$)

$$g(E, Thid_j) = \sum_k (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * W_{jk}$$

$$\begin{aligned} g(E, Thid_1) &= \sum_k (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * W_{1k} \\ &= (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * W_{11} \\ &+ (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * W_{12} \end{aligned}$$

$$\begin{aligned} g(E, Thid_1) = \quad & (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * 0.5 \\ & + (0.5461 - 0.40) * 0.5461 * (1 - 0.5461) * 0.2 = -0.002181992 \end{aligned}$$

$$\begin{aligned} g(E, Thid_2) &= \sum_k (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * W_{2k} \\ &= (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * W_{21} \\ &+ (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * W_{22} \end{aligned}$$

$$\begin{aligned} g(E, Thid_2) = \quad & (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * (-0.3) \\ & + (0.5461 - 0.40) * 0.5461 * (1 - 0.5461) * (-0.3) = -0.00521319 \end{aligned}$$

$$\begin{aligned} g(E, Thid_3) &= \sum_k (Tout_k - targ_k) * Tout_k * (1 - Tout_k) * W_{3k} \\ &= (Tout_1 - targ_1) * Tout_1 * (1 - Tout_1) * W_{31} \\ &+ (Tout_2 - targ_2) * Tout_2 * (1 - Tout_2) * W_{32} \end{aligned}$$

$$\begin{aligned} g(E, Thid_3) = \quad & (0.5729 - 0.65) * 0.5729 * (1 - 0.5729) * 0.2 \\ & + (0.5461 - 0.40) * 0.5461 * (1 - 0.5461) * 0.1 = -0.000148087 \end{aligned}$$

Now let's get the gradient

$$g(E, w_{ij}) = g(E, Thid_j) * Thid_j * (1 - Thid_j) * inp_i$$

$$g(E, w_{11}) = g(E, Thid_1) * Thid_1 * (1 - Thid_1) * inp_1$$

$$\begin{aligned} g(E, w_{11}) = & -0.002181992 * 0.039165723 * (1 - 0.039165723) * 4 \\ & = -0.000328449 \end{aligned}$$

$$g(E, w_{12}) = g(E, Thid_2) * Thid_2 * (1 - Thid_2) * inp_1 = -0.004099901$$

$$g(E,w_{13}) = g(E,Thid_3) * Thid_3 * (1 - Thid_3) * inp_1 = -1.85124E - 05$$
$$g(E,w_{21}) = g(E,Thid_1) * Thid_1 * (1 - Thid_1) * inp_2 = -0.000656898$$
$$g(E,w_{22}) = g(E,Thid_2) * Thid_2 * (1 - Thid_2) * inp_2 = -0.008199802$$
$$g(E,w_{23}) = g(E,Thid_3) * Thid_3 * (1 - Thid_3) * inp_2 = -3.70247E - 05$$
$$g(E,w_{31}) = g(E,Thid_1) * Thid_1 * (1 - Thid_1) * inp_3 = -0.,000492673$$
$$g(E,w_{32}) = g(E,Thid_2) * Thid_2 * (1 - Thid_2) * inp_3 = -0.006149852$$
$$g(E,w_{33}) = g(E,Thid_3) * Thid_3 * (1 - Thid_3) * inp_3 = -2.77685E - 05$$

Now we do the same for the bias of the hidden layer:

$$g(E,bias_H) = \sum_j g(E,Thid_j) * \mathrm{Thid}_j * (1 - \mathrm{Thid}_j)$$

$$\begin{aligned} g(E,bias_H) = &\quad g(E,Thid_1) * \mathrm{Thid}_1 * (1 - \mathrm{Thid}_1) \\ &+ g(E,Thid_2) * \mathrm{Thid}_2 * (1 - \mathrm{Thid}_2) \\ &+ g(E,Thid_3) * \mathrm{Thid}_3 * (1 - \mathrm{Thid}_3) \end{aligned}$$

$$\begin{aligned} g(E,bias_H) = &\quad -0.002181992 * 0.0391 * (1 - 0.0391) \\ &+ -0.00521319 * 0.7310 * (1 - 0.7310) \\ &+ -0.000148087 * 0.9677 * (1 - 0.9677) = -0{,}001111716 \end{aligned}$$

We now can complete the table:

link from node	Link to node	Gradient
Input 1	Hidden 1	-0,000328449
Input 1	Hidden 2	-0,004099901
Input 1	Hidden 3	-1,85124E-05
Input 2	Hidden 1	-0,000656898
Input 2	Hidden 2	-0,008199802
Input 2	Hidden 3	-3,70247E-05
Input 3	Hidden 1	-0,000492673
Input 3	Hidden 2	-0,006149852
Input 3	Hidden 3	-2,77685E-05
Hidden 1	Out 1	-0.00073859
Hidden 1	Out 2	0.00141919
Hidden 2	Out 1	-0.01378643
Hidden 2	Out 2	0.02649025
Hidden 3	Out 1	-0.01824914
Hidden 3	Out 2	0.03506523
	Layer	(Bias)
	Hidden	-0,001111716
	Output	0.017377299

Part D: Update the Weights

We need to make use of these equations to update the weights:

$$Nw_{ij} = Ow_{ij} - \alpha * \ g(E, Ow_{ij})$$

$$Nbias_H = Obias_H - \alpha * \ g(E, Obias_H)$$

$$NW_{ij} = OW_{ij} - \alpha * \ g(E, OW_{ij})$$

$$Nbias_O = Obias_O - \alpha * \ g(E, Obias_O)$$

And we get the next new weights:

NEW WEIGHTS		
link from node	**Link to node**	**Weight**
Input 1	Hidden 1	0,10006569
Input 1	Hidden 2	0,30081998
Input 1	Hidden 3	-0,1999963
Input 2	Hidden 1	-0,39986862
Input 2	Hidden 2	0,10163996
Input 2	Hidden 3	0,2000074
Input 3	Hidden 1	-0,09990147
Input 3	Hidden 2	-0,19877003
Input 3	Hidden 3	0,40000555
Hidden 1	Out 1	0,50014772
Hidden 1	Out 2	0,19971616
Hidden 2	Out 1	-0,29724271
Hidden 2	Out 2	-0,30529805
Hidden 3	Out 1	0,20364983
Hidden 3	Out 2	0,09298695
NEW BIAS WEIGHTS		
	Layer	**Weight**
	Hidden	0,20022234
	Output	0,29652454

Part A Redux: Forward Propagation

First let's calculate the hidden layer values:

$$Whid_{j\,(1 \le j \le 3)} = \ w_{1j} * inp_1 + w_{2j} * inp_2 + w_{3j} * inp_3 + bias_H$$

$$Whid_1 = \ 0.10006 * 4 + (-0.39986) * 8 + (-0.09990) * 6 + 0.20022 \ = \ -3.19787$$

$$Whid_2 = \ 0.30081 * 4 + 0.10163 * 8 + (-0.19877) * 6 + 0.20022 \ = \ 1.02400$$

$$Whid_3 = (-0.19999) * 4 + 0.20000 * 8 + 0.40000 * 6 + 0.20022 = \ 3.40032$$

Then let's apply the sigmoid function:

$$Thid_j = \sigma(Whid_j)$$

$$Thid_1 = \sigma(-3.19787) = \frac{1}{1 + e^{-(-3.19787)}} = 0.039245857$$

$$Thid_2 = \sigma(1.02400) = \frac{1}{1 + e^{-(1.02400)}} = 0.735751362$$

$$Thid_3 = \sigma(3.40032) = \frac{1}{1 + e^{-(3.40032)}} = 0.967714838$$

Now we need to do the same for the output

$$Wout_{k(1 \le k \le 2)} = W_{1k} * Thid_1 + W_{2k} * Thid_2 + W_{3k} * Thid_3 + bias_O$$

$$Wout_1 = 0.294531494$$

$$Wout_2 = 0.169723971$$

Now let's apply the sigmoid function:

$$Tout_k = \sigma(Wout_k)$$

$$Tout_1 = \sigma(0.294531494) = \frac{1}{1 + e^{-(0.294531494)}} = 0.573105154$$

$$Tout_2 = \sigma(0.169723971) = \frac{1}{1 + e^{-(0.169723971)}} = 0.542329429$$

Input 1	Input 2	Input 3	Hidden 1	Hidden 2	Hidden 3	Output 1	Output 2
4	8	6	0.039245	0.735751	0.967714	0.573105	0.542329

Part B Redux: Determine the Error

$$E = 0.5 * \sum_k (targ_k - Tout_k)^2$$

$$E = 0.5 * \sum_{k=1}^{2} (targ_k - Tout_k)^2 = 0.5 * [(targ_1 - Tout_1)^2 + (targ_2 - Tout_2)^2]$$

$$E = 0.5 * [(0.65 - 0.573105)^2 + (0.4 - 0.542329)^2 = 0.013085242$$

Now we can compere the two errors.

The first error was: 0.01365575

The second error was: 0.013085242

As we can see the error has decreased by 0.000570504