Therefore for us to calculate the change in weight, W_{k1l1}, we would proceed as follows:

$$\frac{\partial E_1}{\partial W_{k1l1}} = \frac{\partial E_1}{\partial O_{out1}} * \frac{\partial O_{ut1}}{\partial O_{in1}} * \frac{\partial O_{in1}}{\partial W_{k1l1}}$$ from the chain rule in Calculus

From here we can derive the equations for the other weights as errors for the second hidden layer δ_{KL} is given by:

$$\delta W_{KL} = \begin{bmatrix} \frac{\partial E_1}{\partial W_{k1l1}} & \frac{\partial E_2}{\partial W_{k1l2}} & \frac{\partial E_3}{\partial W_{k1l3}} \\ \frac{\partial E_1}{\partial W_{k2l1}} & \frac{\partial E_2}{\partial W_{k2l2}} & \frac{\partial E_3}{\partial W_{k2l3}} \\ \frac{\partial E_1}{\partial W_{k3l1}} & \frac{\partial E_2}{\partial W_{k3l2}} & \frac{\partial E_3}{\partial W_{k3l3}} \end{bmatrix} = \begin{bmatrix} \frac{\partial E_1}{\partial O_{out1}} \times \frac{\partial O_{out1}}{\partial O_{in1}} * \frac{\partial O_{in1}}{\partial W_{k1l1}} & \frac{\partial E_2}{\partial O_{out2}} * \frac{\partial O_{out2}}{\partial O_{in2}} * \frac{\partial O_{in2}}{\partial W_{k1l2}} & \frac{\partial E_3}{\partial O_{out3}} * \frac{\partial O_{out3}}{\partial O_{in3}} * \frac{\partial O_{in3}}{\partial W_{k1l3}} \\ \frac{\partial E_1}{\partial O_{out1}} \times \frac{\partial O_{out1}}{\partial O_{in1}} \times \frac{\partial O_{in1}}{\partial W_{k2l1}} & \frac{\partial E_2}{\partial O_{out2}} * \frac{\partial O_{out2}}{\partial O_{in2}} * \frac{\partial O_{in2}}{\partial W_{k2l2}} & \frac{\partial E_3}{\partial O_{out3}} * \frac{\partial O_{out3}}{\partial O_{in3}} * \frac{\partial O_{in3}}{\partial W_{k2l3}} \\ \frac{\partial E_1}{\partial O_{out1}} \times \frac{\partial O_{out1}}{\partial O_{in1}} * \frac{\partial O_{in1}}{\partial W_{k3l1}} & \frac{\partial E_2}{\partial O_{out2}} * \frac{\partial O_{out2}}{\partial O_{in2}} * \frac{\partial O_{in2}}{\partial W_{k3l2}} & \frac{\partial E_3}{\partial O_{out3}} * \frac{\partial O_{out3}}{\partial O_{in3}} * \frac{\partial O_{in3}}{\partial W_{k3l3}} \end{bmatrix}$$

$$\delta W_{KL} = \begin{bmatrix} \delta W_{k1l1} & \delta W_{k1l2} & \delta W_{k1l3} \\ \delta W_{k2l1} & \delta W_{k2l2} & \delta W_{k2l3} \\ \delta W_{k3l1} & \delta W_{k3l2} & \delta W_{k3l3} \end{bmatrix} = \begin{bmatrix} -5.036 \times 0.1591 \times 0.938 & -1.399 \times 0.2025 \times 0.938 & -2.065 \times 0.2497 \times 0.938 \\ -5.036 \times 0.1591 \times 0.94 & -1.399 \times 0.2025 \times 0.94 & -2.065 \times 0.2497 \times 0.94 \\ -5.036 \times 0.1591 \times 0.98 & -1.399 \times 0.2025 \times 0.98 & -2.065 \times 0.2497 \times 0.98 \end{bmatrix}$$

$$\delta W_{KL} = \begin{bmatrix} -0.7516 & -0.2657 & -0.4837 \\ -0.7532 & -0.2663 & -0.4847 \\ -0.7852 & -0.2776 & -0.5053 \end{bmatrix}$$

These results shows how much we should reduce our weights by. Typically, a learning rate is applied which ranges from 0.1 to 0.001. It helps with convergence of the algorithm.

We use the equation $W = W - (\text{learning rate}(\alpha) \times \delta W)$

New weight matrix for

$$\hat{W}_{KL} = \begin{bmatrix} W_{k1l1} - (\alpha * \delta W_{k1l1}) & W_{k1l2} - (\alpha * \delta W_{k1l2}) & W_{k1l3} - (\alpha * \delta W_{k1l3}) \\ W_{k2l1} - (\alpha * \delta W_{k2l1}) & W_{k2l1} - (\alpha * \delta W_{k2l1}) & W_{k2l3} - (\alpha * \delta W_{k2l3}) \\ W_{k3l1} - (\alpha * \delta W_{k3l1}) & W_{k3l1} - (\alpha * \delta W_{k3l1}) & W_{k3l3} - (\alpha * \delta W_{k3l3}) \end{bmatrix}$$

$$= \begin{bmatrix} 0.1 - (0.01)(-0.7516) & 0.4 - (0.01)(-0.2657) & 0.8 - (0.01)(-0.4837) \\ 0.3 - (0.01)(-0.7532) & 0.7 - (0.01)(-0.2663) & 0.2 - (0.01)(-0.4847) \\ & \ldots(0.01)(-0.2776) & 0.9 - (0.01)(-0.5053) \end{bmatrix} =$$