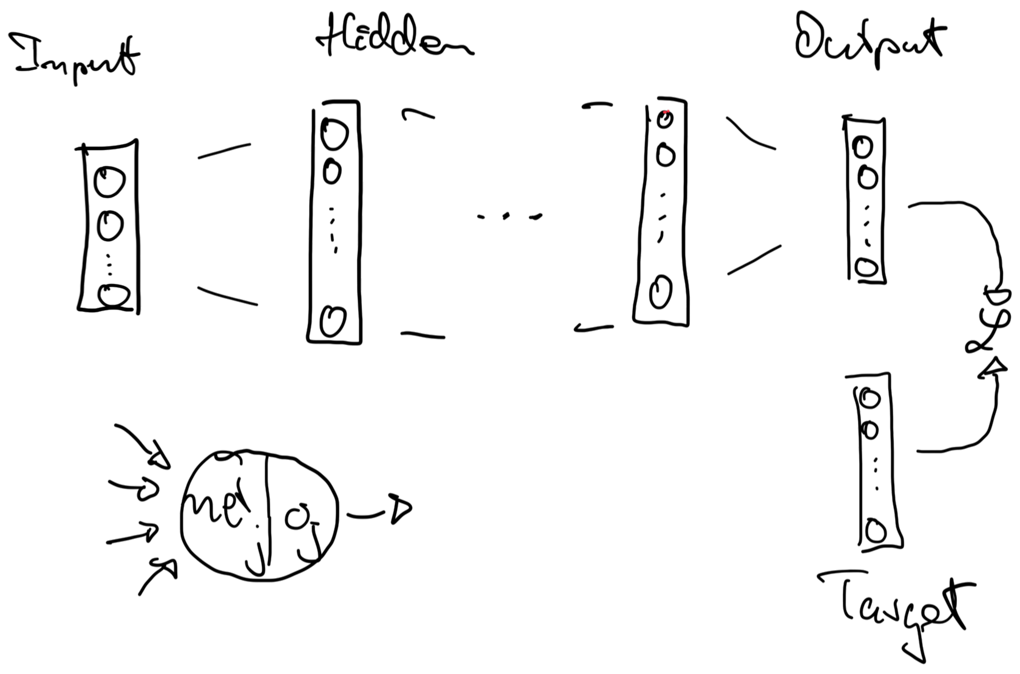
Backpropagation by hand



$$E = \mathcal{L}(t, o) = \frac{1}{2}(o_j - t_j)^2$$

$$o_j = \varphi(net_j)$$

$$net_j = \sum_{k=1}^{n} w_{kj} o_k$$

$$\frac{\partial E}{\partial w_{ij}} = \overbrace{\frac{1}{2}\Big(\overbrace{\varphi\big(\overbrace{\textstyle\sum_k w_{kj} o_k}^{net_j}\big)}^{o_j} - t_j\Big)^2}^{E}$$

$$= \underset{\text{I}}{\frac{\partial E}{\partial o_j}} \; \underset{\text{II}}{\frac{\partial o_j}{\partial net_j}} \; \underset{\text{III}}{\frac{\partial net_j}{\partial w_{ij}}}$$

$$\delta_j := \frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j}$$

I Case A: j is output layer

$$\frac{\partial E}{\partial o_j} = \frac{\partial}{\partial o_j} \frac{1}{2}(o_j - t_j)^2$$

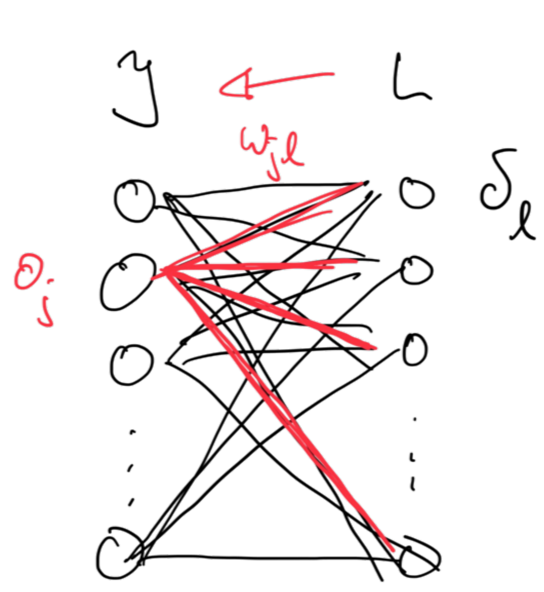
$$= \frac{\partial}{\partial o_j} \frac{1}{2}\left[o_j^2 - 2 o_j t_j + t_j^2\right]$$

$$= \frac{1}{2}\left[2 o_j - 2 t_j\right]$$

$$\boxed{\frac{\partial E}{\partial o_j} = o_j - t_j}$$

Case B: j is hidden layer

$$\boxed{\frac{\partial E}{\partial o_j} = \sum_{l}^{L} w_{jl} \delta_l}$$



II

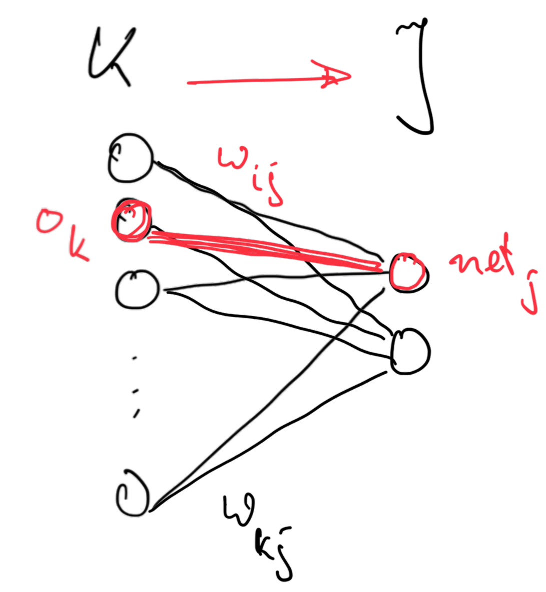
$$\frac{\partial o_j}{\partial net_j} = \frac{\partial}{\partial net_j} \varphi(net_j)$$

$$\boxed{\frac{\partial o_j}{\partial net_j} = \varphi'(net_j)}$$

III

$$\frac{\partial net_j}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \sum_{k=1}^{n} w_{kj} o_k$$

$$\boxed{\frac{\partial net_j}{\partial w_{ij}} = o_k = o_i}$$



Putting it together

$$\frac{\partial E}{\partial w_{ij}} = \begin{cases} (o_j - t_j)\, \varphi'(net_j)\, o_i & \text{if } j \text{ is output layer} \\ \underbrace{\left(\sum_{l}^{L} w_{jl} \delta_l\right) \varphi'(net_j)}_{\delta_j}\, o_i & \text{else} \end{cases}$$

Steps for gradient computation

1. Compute deltas starting from output layer backwards to input layer.
2.) Compute gradients with respect to each weight.