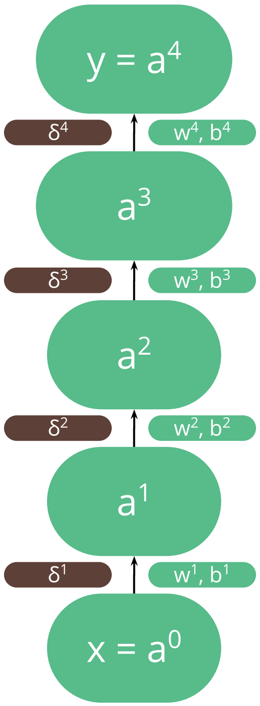
Notation

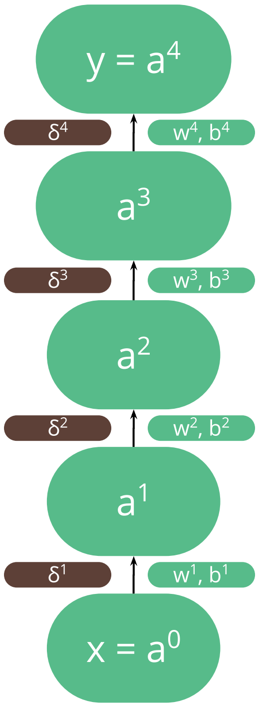
- t -> Ground Truth Output
- C -> Cost Function
- δ -> Gradient



The Cost Function

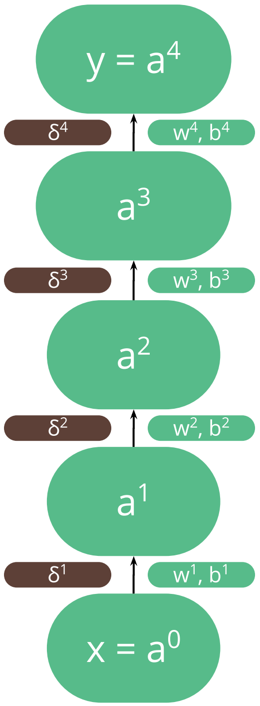
For a scalar output

- Mean Squared Error: $C = ½ * (y - t)^2$
- Cross Entropy: $C = t * \ln(y) + (1-t) * \ln(1-y)$



Backpropagation

- Goal: Compute $\partial C / \partial w$ and $\partial C / \partial b$
- Why: Use them for Stochastic Gradient Descent
- Define: $\delta^l = \partial C / \partial z^l$



Backward Pass

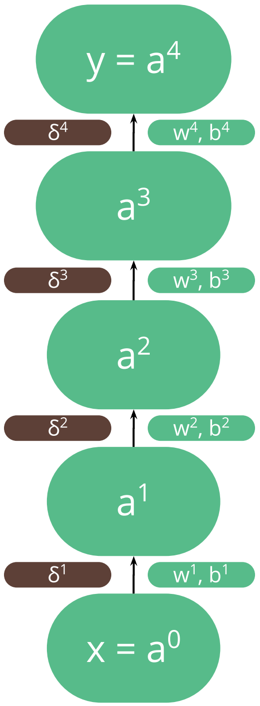
$\delta^4 = \partial C/\partial z^4 = \partial C/\partial y * \partial y/\partial z^4$

Now

- $\partial C/\partial y = (y - t)$
- $\partial y/\partial z^4 = \partial a^4/\partial z^4 = f'(z^4)$

where f'(.) is derivative of f(.) w.r.t (.)

$\Rightarrow \delta^4 = (y - t) * f'(z^4)$



Backward Pass

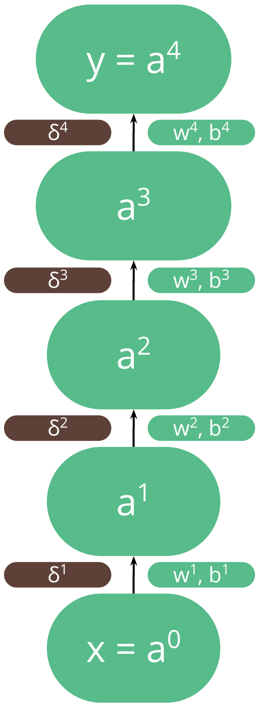
$\delta^3 = \partial C/\partial z^3$

Now

- $z^4_1 = \ldots + w^4_{1j} * f(z^3_j) + \ldots$
- $z^4_k = \ldots + w^4_{kj} * f(z^3_j) + \ldots$

i.e. all elements of z^4 depend on z^3_j

Thus, by chain rule we can say that
$\delta^3_j = \partial C/\partial z^3_j = \sum_k \partial C/\partial z^4_k * \partial z^4_k/\partial z^3_j$



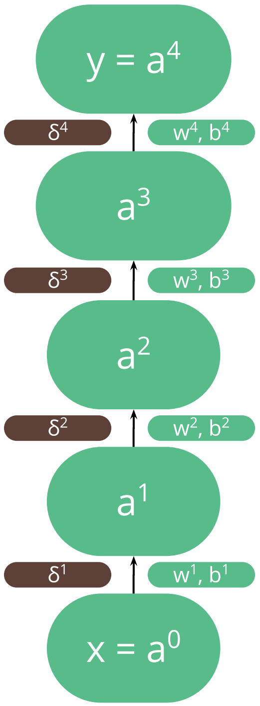
Backward Pass

$\delta^3_j = \partial C/\partial z^3_j = \sum_k \partial C/\partial z^4_k * \partial z^4_k/\partial z^3_j$
$\Rightarrow \delta^3_j = \sum_k \partial C/\partial z^4_k * \partial z^4_k/\partial a^3_j * \partial a^3_j/\partial z^3_j$

Now

- $\partial C/\partial z^4_k = \delta^4_k$
- $\partial z^4_k/\partial a^3_j = w^4_{kj}$ [As $z^4_k = \ldots + w^4_{kj} * a^3_j + \ldots$]
- $\partial a^3_j/\partial z^3_j = f'(z^3_j)$

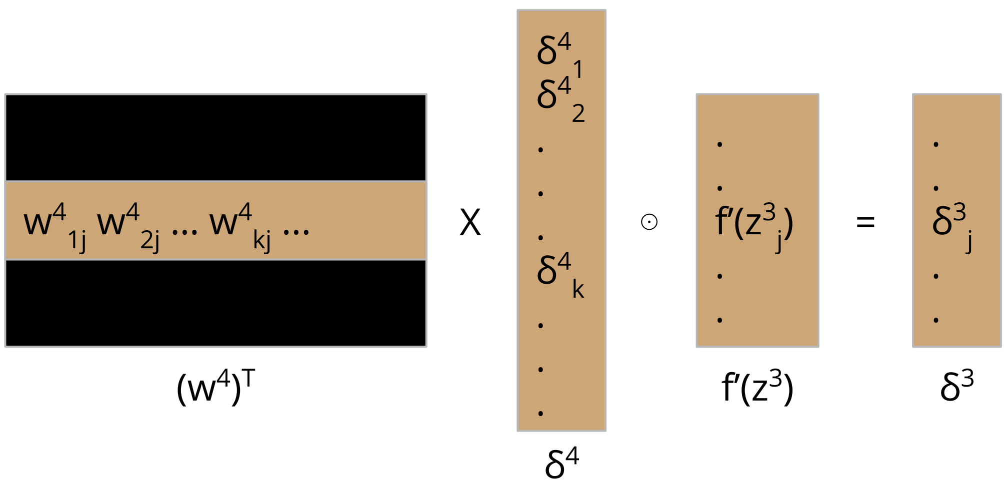
$\Rightarrow \delta^3_j = (\sum_k \delta^4_k * w^4_{kj}) * f'(z^3_j)$

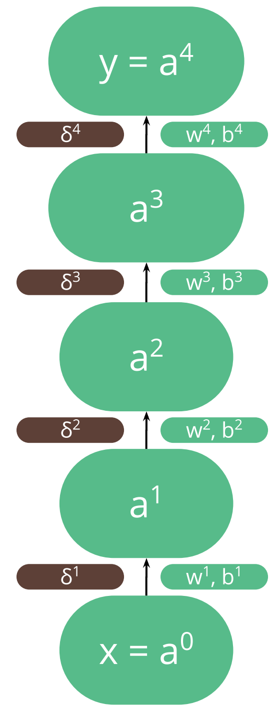


Backward Pass

$\delta^3_j = (\sum_k \delta^4_k * w^4_{kj}) * f'(z^3_j)$
$\Rightarrow \delta^3 = (w^4)^T * \delta^4 \odot f'(z^3)$

where $\odot$ = Element-wise product





Backward Pass

Hence we have

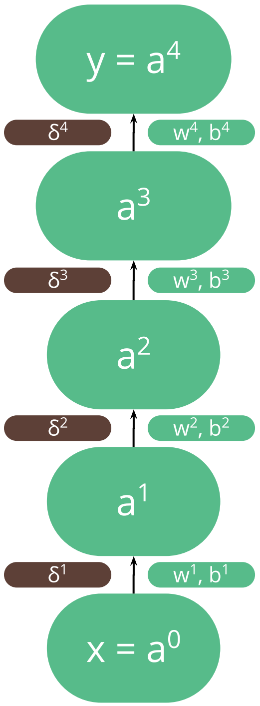
- $\delta^4 = (y - t) * f'(z^4)$
- $\delta^3 = (w^4)^T * \delta^4 \odot f'(\ z^3\ )$
- $\delta^2 = (w^3)^T * \delta^3 \odot f'(\ z^2\ )$
- $\delta^1 = (w^2)^T * \delta^2 \odot f'(\ z^1\ )$

Or in general

$\delta^l = (w^{l+1})^T * \delta^{l+1} \odot f'(\ z^l\ )$ for $l = 1, 2, \ldots, L-1$

$\delta^L = \nabla_y C \odot f'(\ z^L\ )$

where $\nabla_y C$ is derivative of cost wrt output



Backward Pass

Now for our main objectives: $\partial C/\partial w^l_{jk}$ and $\partial C/\partial b^l_j$

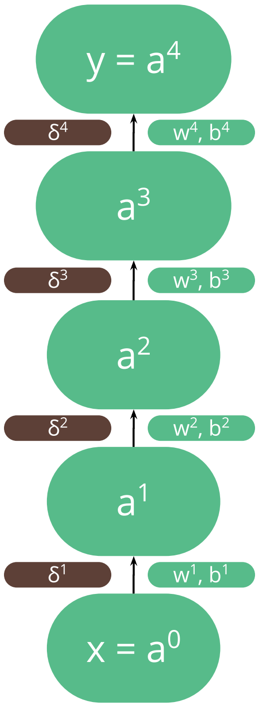
$\partial C/\partial w^l_{jk} = \partial C/\partial z^l_j * \partial z^l_j / \partial w^l_{jk}$

Since

- $\partial C/\partial z^l_j = \delta^l_j$
- $\partial z^l_j / \partial w^l_{jk} = a^{l-1}_k$

[As $z^l_j = \ldots + w^l_{jk} * a^{l-1}_k + \ldots$]

=> $\partial C/\partial w^l_{jk} = \delta^l_j\, a^{l-1}_k$

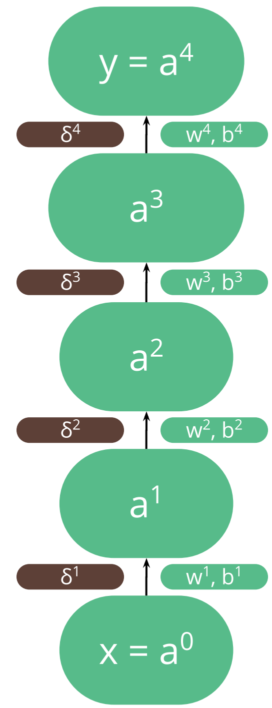


Backward Pass

$\partial C/\partial w^l_{jk} = \delta^l_j \, a^{l-1}_k$

Or in general
$\partial C/\partial w^l = \delta^l * (a^{l-1})^T$ for $l = 1, \ldots, L$

$$\underbrace{\begin{bmatrix} \delta^l_1 a^{l-1}_1 & \ldots & \delta^l_1 a^{l-1}_m \\ \vdots & & \vdots \\ \delta^l_n a^{l-1}_1 & \ldots & \delta^l_n a^{l-1}_m \end{bmatrix}}_{\partial C/\partial w^l} = \underbrace{\begin{bmatrix} \delta^l_1 \\ \vdots \\ \delta^l_n \end{bmatrix}}_{\delta^l} \times \underbrace{\begin{bmatrix} a^{l-1}_1 & \ldots & a^{l-1}_m \end{bmatrix}}_{(a^{l-1})^T}$$



Backward Pass

Also

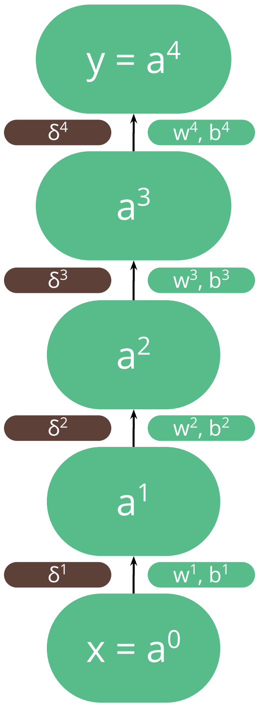
$\partial C/\partial b^l_j = \partial C/\partial z^l_j * \partial z^l_j / \partial b^l_j$

Since

- $\partial C/\partial z^l_j = \delta^l_j$
- $\partial z^l_j / \partial b^l_j = 1$ [As $z^l_j = \ldots + b^l_j$]

=> $\partial C/\partial b^l_j = \delta^l_j$

Or in general $\partial C/\partial b^l = \delta^l$ for $l = 1, \ldots, L$

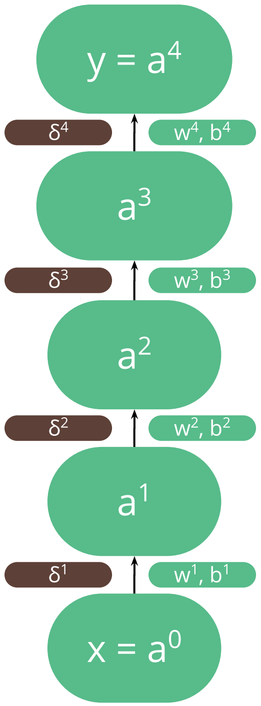


Backward Pass

In general:
$\delta^L = \nabla_y C \odot f'(z^L)$ where $\nabla_y C$ is derivative of cost wrt output

Then for $l = 1, 2, \ldots, L-1$
$\delta^l = (w^{l+1})^T * \delta^{l+1} \odot f'(z^l)$ where $\odot$ stands for element wise product

Finally for $l = 1, \ldots, L$
$\partial C/\partial w^l = \delta^l * (a^{l-1})^T$
$\partial C/\partial b^l = \delta^l$



Summary

Forward Pass

The Input

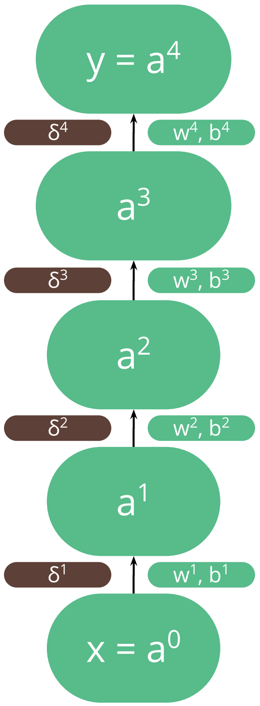
- $a^0 = x$

For l = 1, ... , L layers

- $z^l = w^l * a^{l-1} + b^l$
- $a^l = f(z^l)$

Finally

- $y = a^L$



Backward Pass

$\delta^L = \nabla_y C \odot f'(z^L)$
where $\nabla_y C$ is derivative of cost wrt output

Then for l = 1, 2, ..., L-1
$\delta^l = (w^{l+1})^T * \delta^{l+1} \odot f'(z^l)$
where $\odot$ stands for element wise product

Finally for l = 1, ..., L
$\partial C/\partial w^l = \delta^l * (a^{l-1})^T$
$\partial C/\partial b^l = \delta^l$