Backpropagation

Recall: The chain rule

$$\frac{d}{dw} C(z(w))$$

$$\frac{dC}{dw} = \frac{dC}{dz}\frac{dz}{dw}$$

$$\frac{d}{dw} C(z_1(w), z_2(w), \ldots, z_N(w))$$

$$\frac{dC}{dw} = \sum_{i=1}^{N} \frac{dC}{dz_i}\frac{dz_i}{dw}$$

Definitions:

L: # of layers

N^m: Dimensionality of layer m

W^m: weight matrix for layer m $\quad \mathbb{R}^{N^m \times N^{m-1}}$

b^m: bias vector for layer m $\quad \mathbb{R}^{N^m}$

σ^m: nonlinearity for layer m

z^m: "preactivation" for layer m $\quad z^m = W^m a^{m-1} + b^m$

a^m: "activations" for layer m $\quad a^m = \sigma^m(z^m)$

a^0: input to the model (x)

y: target output

C: Cost function $\quad C(a^L, y)$
(loss)

Want $\frac{dC}{dw^m}$ for all m.

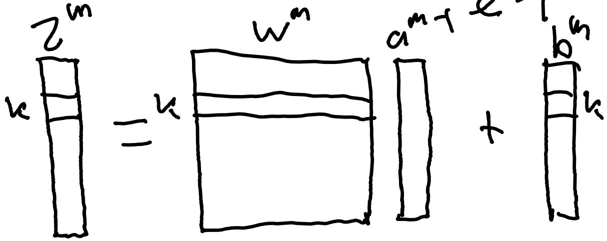
Back prop will give us $\frac{dC}{dw^m}$ given:

$$\frac{dC}{da^L} \quad \text{and} \quad \frac{da^m}{dz^m}$$

↳ immediately known based on design

$$\frac{dC}{dw_{ij}} = \sum_{k=1}^{N^m} \frac{dC}{dz_k^m} \frac{dz_k^m}{dw_{ij}^m}$$

Recall: $z_k^m = \sum_{\ell=1}^{N^{m-1}} w_{k\ell}^m a_\ell^{m-1} + b_k^m$



if $i \neq k$, $\frac{dz_k^m}{dw_{ij}^m} = 0$

When $i=k$

$$\frac{dz_i^m}{dW_{ij}^m} = \frac{d}{dW_{ij}}\left(\sum_{e=1}^{N^{m-1}} W_{ie}^m a_e^{m-1} + b_i^m\right)$$

$$= a_j^{m-1}$$

$$\rightarrow \frac{dz_k^m}{dW_{ij}^m} = \begin{cases} 0, & i \neq k \\ a_j^{m-1}, & i = k \end{cases}$$

$$\rightarrow \frac{dC}{dW_{ij}^m} = \frac{dC}{dz_i^m} a_j^{m-1}$$

$$\rightarrow \frac{dC}{dW^m} = \frac{dC}{dz^m} a^{m-1^T}$$

We still need $\frac{dC}{dz^m}$.

For $m = L$, $\frac{dC}{dz_k^L} = \underbrace{\frac{dC}{da_k^L} \frac{da_k^L}{dz_k^L}}_{\text{both known}}$

For $m < L$, we have

$$\frac{dC}{dz_k^m} = \frac{dC}{da_k^m} \frac{da_k^m}{dz_k^m}$$

known

$$= \left(\sum_{e=1}^{N^m} \frac{dC}{dz_e^{m+1}} \frac{dz_e^{m+1}}{da_k^m} \right) \frac{da_k^m}{dz_k^m}$$

$$= \left(\sum_{e=1}^{N^m} \frac{dC}{dz_e^{m+1}} \frac{d}{da_k^m} \left(\sum_{h=1}^{N^m} W_{eh}^{m+1} a_h^m + b_e^{m+1} \right) \right) \frac{da_k^m}{dz_k^m}$$

$$= \left(\sum_{e=1}^{N^m} \frac{dC}{dz_e^{m+1}} W_{ek}^{m+1} \right) \frac{da_k^m}{dz_k^m}$$

$$\frac{dC}{dz^m} = \left(W^{m+1^T} \frac{dC}{dz^{m+1}} \right) \circ \frac{da^m}{dz^m}$$

How to compute $\frac{dC}{d\text{any weight}}$:

1) $\frac{dC}{dz^L} = \frac{dC}{da^L}\frac{da^L}{dz^L}$ (all known)

2) Recursively compute $\frac{dC}{dz^m}$ for $m = L-1, L-2, \ldots$

$$\frac{dC}{dz^m} = \Big(\underbrace{W^{m+1^T}}_{\text{known}} \underbrace{\frac{dC}{dz^{m+1}}}_{\text{previous step}}\Big) \circ \underbrace{\frac{da^m}{dz^m}}_{\text{known}}$$

3) $\frac{dC}{dW^m} = \underbrace{\frac{dC}{dz^m}}_{\text{from step2}} \underbrace{a^{m-1^T}}_{\text{from forward pass}}$

ex. $a^m = \text{sigmoid}(z^m) = \frac{1}{1+\exp(-z^m)}$ (all m, including $m=L$)

$$C(y, a^L) = (y-1)\log(1-a^L) - y\log(a^L)$$

$$\frac{dC}{dz^L} = \boxed{\frac{dC}{da^L}}\boxed{\frac{da^L}{dz^L}}$$

deriv. of sigmoid

deriv. of x-entropy

$$= \frac{a^L - y}{a^L(1-a^L)} a^L(1-a^L)$$

$$= a^L - y$$

$$\frac{dC}{dz^{L-1}} = \left(W^{L^T} \frac{dC}{dz^L}\right) \circ \frac{da^{L-1}}{dz^{L-1}}$$

$$= W^{L^T}(a^L - y) \circ a^{L-1}(1-a^{L-1})$$

$$\frac{dC}{dW^{L-1}} = \frac{dC}{dz^{L-1}} a^{L-2^T}$$

$$= W^{L^T}(a^L - y) \circ a^{L-1}(1-a^{L-1}) a^{L-2^T}$$

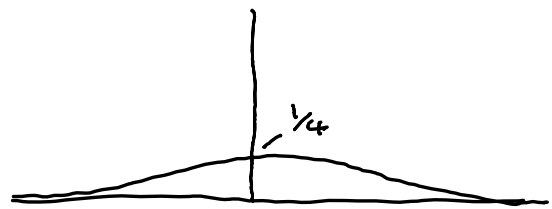
$$\frac{dC}{dz^m} = \left(W^{m+1^T} \frac{dC}{dz^{m+1}} \right) \circ \frac{da^m}{dz^m}$$

repeated matrix multiplies

repeated multiplication of nonlinearity derivative

sigmoid (x)

d/dx sigmoid(x)



This can cause to

"explode" (go to ∞) or "vanish" (go to 0)

as we propagate back.

Parameter Initialization

$$\frac{dC}{dz^m} = \left(W^{m+1^T} \frac{dC}{dz^{m+1}} \right) \circ \frac{da^m}{dz^m}$$

Want to avoid vanishing/exploding gradients.
Can control the weight matrix initialization.
"initialize" → choose a distribution to sample initial values from.

Assume: $o = wx$ $\quad x \in \mathbb{R}^{n_{in}}, o \in \mathbb{R}^{n_{out}}$

1. No nonlinearities
2. $x_i \sim \mathcal{N}(0, 1)$
3. $w_i \sim \mathcal{N}(0, \sigma^2)$

what should we use for sigma?

$$o = \sum_i w_i x_i$$

$$\mathbb{E}[o] = 0$$

$$\begin{aligned} Var[o] &= \mathbb{E}[o^2] - \cancel{(\mathbb{E}[o])^2} \\ &= \sum_i \mathbb{E}[w_i^2 x_i^2] \\ &= \sum_i \mathbb{E}[w_i^2]\,\mathbb{E}[x_i^2] = n_{in}\sigma^2 \end{aligned}$$

Forward pass: If we want the variance of activations to be ≈ 1, set $n_{in}\sigma^2 = 1$

Backward pass: Want $n_{out}\sigma^2 = 1$

Can't satisfy both if $n_{in} \neq n_{out}$.

So, we average them as a compromise:

$$(n_{in}\sigma^2 + n_{out}\sigma^2)/2 = 1$$

$$\rightarrow \sigma = \sqrt{\frac{2}{n_{in} + n_{out}}}$$

"Xavier" or "Glorot" initialization

Autograd

ex/ $h = \mathrm{ReLU}(W_h x + b_h)$

$o = W_o h + b_o$

$L = (y - o)^2$

We want $\frac{dL}{dW_o}, \frac{dL}{db_o}, \frac{dL}{dW_h}, \frac{dL}{db_h}$

We can always use the chain rule.

Break down into individual operations.

$m_h = W_h x$

$z_h = m_h + b_h$

$h = \mathrm{ReLU}(z_h)$

$m_o = W_o h$

$o = m_o + b_o$

$e = y - o$

$L = e^2$

Now we can write:

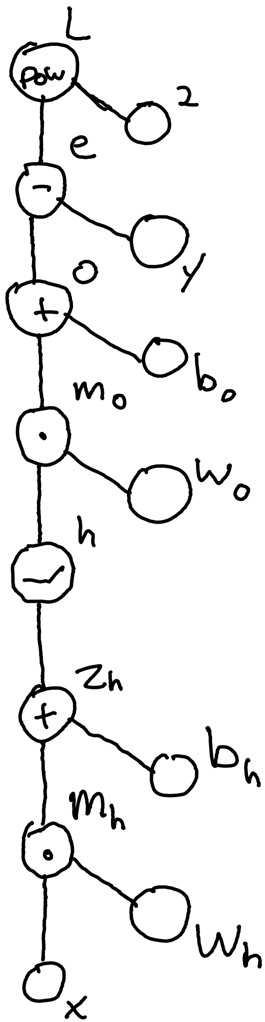
$$\frac{dL}{dw_n} = \frac{dL}{de}\frac{de}{do}\frac{do}{m_o}\frac{dm_o}{dh}\frac{dh}{dz_n}\frac{dz_n}{dm_n}\frac{dm_n}{dw_n}$$

At each step, we're computing the derivative of a single simple operation.

1. What operations were applied?
2. Derivatives of those operations.

So, we:

1. Define a set of operations
2. Define their derivatives
3. Keep track of what operations were appled in a "compution"



L
pow
2
e
y
o
b_o
m_o
w_o
h
z_h
b_h
m_h
W_h
x