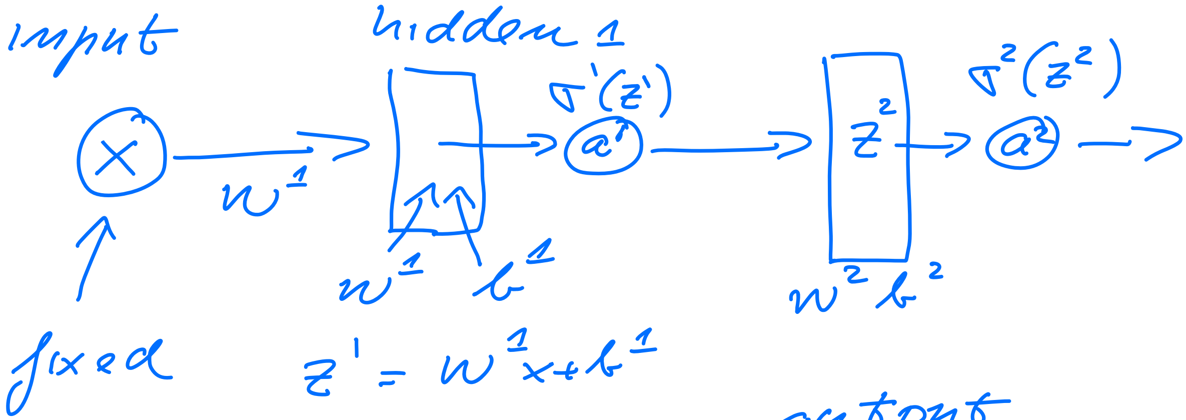
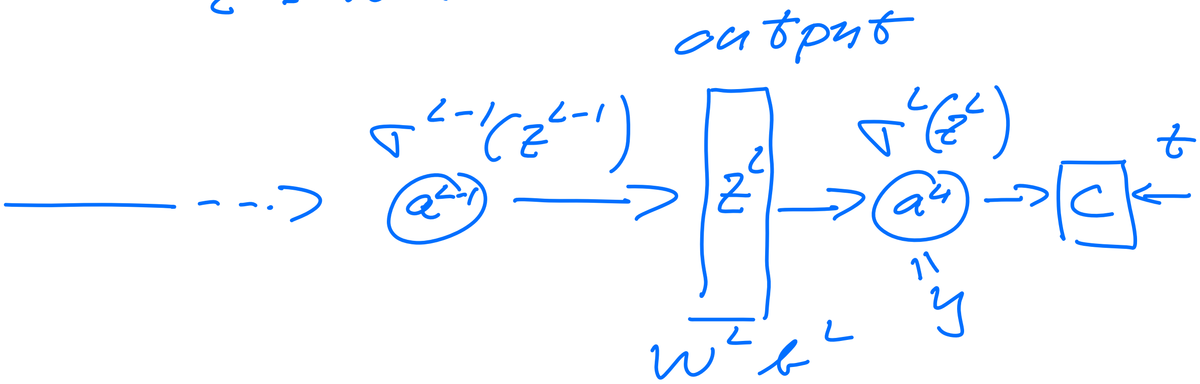
# Comp Sci program, JAN 31, 2023

## FFNN




$$y = a^L(\Theta, x) = \sigma^L(z^L)$$

$$\Theta = \{ w^1, b^1, w^2, b^2 \ldots w^L, b^L \}$$

$$\sigma^L(z^L) = \sigma^L(\sigma^{L-1}(\sigma^{L-2}(\ldots \sigma^1(z^1)\ldots)))$$

$$C(\Theta) = \| t - a^L(\Theta; x) \|_2^2$$

(MSE in regression)

$$\boxed{\frac{\partial C}{\partial \theta^L} = 0}$$

$$\frac{\partial C}{\partial \theta^{L-1}} = \frac{\partial C}{\partial a^L} \frac{\partial a^L}{\partial \theta^{L-1}}$$

$$\frac{\partial C}{\partial \theta^{L-2}} = \frac{\partial C}{\partial a^L} \frac{\partial a^L}{\partial a^{L-1}} \frac{\partial a^{L-1}}{\partial \theta^{L-2}}$$

$$\frac{\partial C}{\partial \theta^\ell} = \frac{\partial C}{\partial a^L} \frac{\partial a^L}{\partial a^{L-1}} \frac{\partial a^{L-1}}{\partial a^{L-2}} \cdots \frac{\partial a^{\ell+1}}{\partial \theta^\ell}$$

$$\theta^\ell_{(k+1)} \leftarrow \theta^\ell_{(k)} - \eta_{(k)} \left.\frac{\partial C}{\partial \theta^\ell}\right|_{\theta^\ell = \theta^\ell_{(k)}}$$

Automatic differentiation

---

$$f(x) = \sqrt{x^2 + \exp(x^2)}$$

$x \cdot x = 1$ FLOP

$\exp(x^2) = \exp(x \cdot x) = 2$ FLOP

$x^2 + \exp(x^2) = 1$ FLOP

$$\frac{\text{SQRT}(\cdot\cdot) = 1 \text{ FLOP}}{5 \text{ FLOP}}$$

$$\frac{df}{dx} = \frac{x(1+\exp(x^2))}{\sqrt{x^2+\exp(x^2)}}$$

10 FLOPS

automatic diff:

$a = x^2$

$b = \exp(a)$

$c = a+b$

$d = \sqrt{c} = f(x)$

$$\frac{da}{dx} = 2x \qquad \frac{db}{dx} = \frac{db}{da}\frac{da}{dx}$$

$$= 2x\exp(x^2)$$

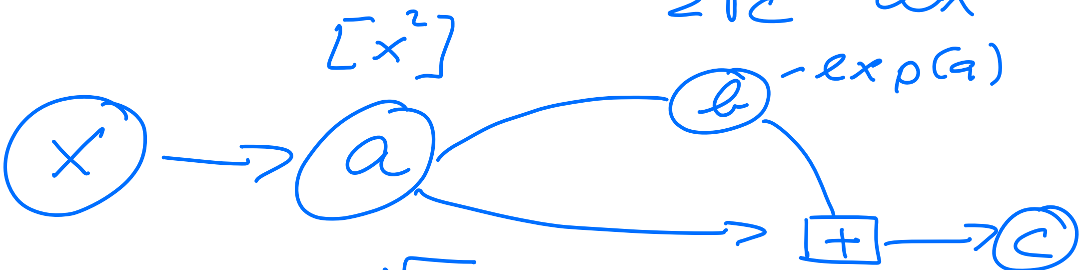
$$\frac{dc}{dx} = \left[\frac{dc}{da}\frac{da}{dx} + \frac{dc}{db}\frac{db}{dx}\right]$$

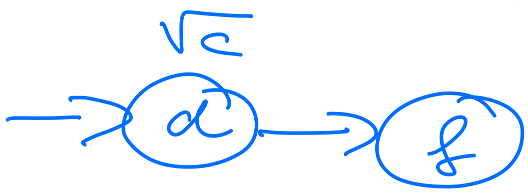
$$= \left[\frac{dc}{da}\frac{da}{dx} + \frac{dc}{db}\frac{db}{da}\frac{da}{dx}\right]$$

$$\frac{dd}{dc} = \frac{1}{2\sqrt{c}}$$

$$\frac{dd}{dx} = \frac{df}{dx} = \frac{dd}{dc}\,\frac{dc}{dx}$$

$$= \frac{1}{2\sqrt{c}}\,\frac{dc}{dx}$$




$$\frac{df}{dd} = \underline{1}$$

compute $\frac{df}{dx}$ in backward/reverse mode

$$\frac{df}{dc} = \underbrace{\frac{df}{dd}}_{=\underline{1}}\,\frac{dd}{dc} = \frac{1}{2\sqrt{c}}$$

$$\frac{df}{db} = \frac{df}{dc}\,\frac{dc}{db} = \frac{1}{2\sqrt{c}}$$

$c = a+b \quad \frac{dc}{db} = \underline{1}$

$$\frac{df}{da} = \frac{df}{db}\frac{db}{da} + \frac{df}{dc}\frac{dc}{da}$$

$$= \frac{1}{2\sqrt{c}}\left[1 + \exp(a)\right]$$

$$\frac{df}{dx} = \frac{df}{da}\frac{da}{dx}$$

$$= \frac{x\left(1+\exp(x^2)\right)}{\sqrt{x^2+\exp(x^2)}}$$

$$\frac{df}{dx} = \frac{x(1+b)}{\sqrt{a+b}}$$

$b$ is calculated in $f$

numerator : 2 Flops

denominator : 2 Flops

+ Division : 5 Flops

$$f(x) = \sqrt{x^2+\exp(x^2)} = \sqrt{a+b}$$

$$\frac{df}{dx} = \frac{x(1+b)}{f(x)}$$

Formalization

assume we have $x_1, x_2 \dots x_d$
input variables to $f$,

$x_{d+1}, x_{d+2} \dots x_D$ intermediate
variables $x_D$ = output
variable

in previous example

$x_1 = x \qquad d = 1$

$x_2 = a \qquad x_3 = b \qquad x_4 = c$

$x_D = d = f$

For $i = d+1, \dots D$

$$x_i = g_i(x_{pa(x_i)})$$

$g_i$ are elementary functions
and $x_{pa(x_i)}$ are the parent
nodes of the variable $x_i$

$$g_2 = (x \cdot x)^2 = a$$

$$g_3 = \exp(\;\;) = b$$

$$g_4 = c = a + b$$

$$g_5 = \sqrt{c} = d = f$$

By def $\quad \dfrac{df}{dx_D} = \underline{1}$

Reverse mode or Backprop

$$\frac{df}{dx_i'} = \sum_{\substack{x_j \\ x_i' = Pa(x_j)}} \frac{df}{dx_j} \frac{dg_j}{dx_i'}$$

$Pa(x_j)$ = set of parent nodes of $x_i'$

$$\frac{df}{dd} = \underline{1}$$

$$\frac{df}{dc} = \underbrace{\frac{df}{dd}}_{=1} \frac{dd}{dc} = \frac{1}{2\sqrt{c}}$$

$$\frac{df}{db} = \frac{df}{db}\frac{db}{da} + \frac{df}{dc}\frac{dc}{da}$$

$$\frac{df}{dx} = \frac{df}{da}\frac{da}{dx} = \frac{x(1+b)}{d}$$

Simple neural network example:


$$a = f(x, w_1) \qquad y = g(a, w_2)$$

$$= g(f(x, w_1), w_2)$$

$$C = \frac{1}{2}(t-y)^2$$

$$w_1^{(k+1)} \Leftarrow w_1^{(k)} - \delta \frac{\partial C}{\partial w_1}\Big|_{w_1 = w_1^{(k)}}$$

$$w_2^{(k+1)} \Leftarrow w_2^{(k)} - \delta \frac{\partial C}{\partial w_2}\Big|_{w_2 = w_2^{(k)}}$$

linear activation function

$$f(x, w_1) = w_1 x = a$$

$$g(a, w_2) = w_2 \cdot a = w_2 f(x_1, w_1)$$

$$\frac{\partial C}{\partial w_1} = -(t-y)\frac{\partial y}{\partial a}\frac{\partial a}{\partial w_1}$$

$$\frac{\partial C}{\partial w_2} = -(t-y)\frac{\partial y}{\partial w_2}$$

$$\frac{\partial C}{\partial w_1} = -(t-y) \times w_2$$

$$\frac{\partial C}{\partial w_2} = -(t-y) w_1 x$$

$$\frac{\partial C}{\partial w_1} = -(t-y)\frac{\partial y}{\partial a_L}\frac{\partial a_L}{\partial a_{L-1}} \cdots \frac{\partial a_2}{\partial a_1}\frac{\partial a_1}{\partial w_1}$$

Ingredients for an NN-code

- architecture (model)
  - # layers
  - # nodes
  - # activation functions and their derivatives
- cost function
- regularization & optimization
  - regularization parameter $\lambda$ with $l_1$ or $l_2$
  - gradient descent methods (GD)
    - GD with momentum
    - SGD with & without momentum
    - learning rates
      - AdaGrad
      - RMSprop
      - ADAM