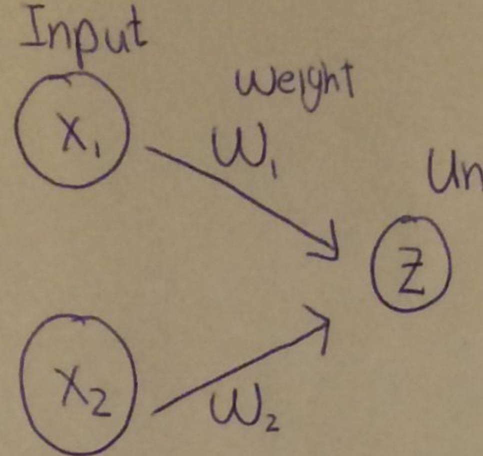
①



activation function

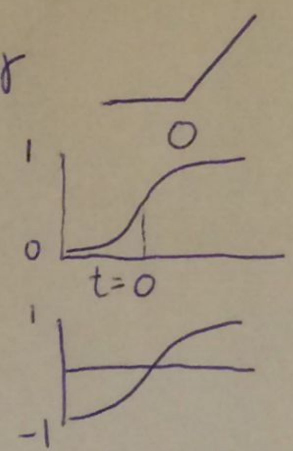
$$\mathcal{Z} = \mathcal{Z}(\underbrace{\underline{w}}_{\binom{w_1}{w_2}} \cdot \underline{x} + \underbrace{b}_{\text{bias}})$$

Possible act. functions

$\mathcal{Z}(t) =$ rectified linear $\quad \max(0,t)$

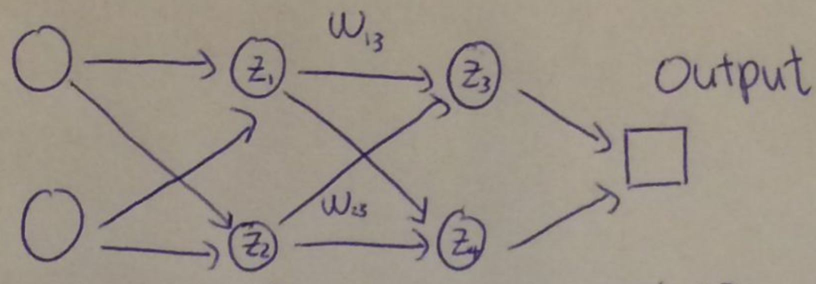
Sigmoid $\dfrac{1}{1+e^{-t}}$

tanh $\dfrac{e^{2x}-1}{e^{2x}+1}$



$\tanh(x) = 2\,\mathrm{Sig}(2x) - 1$

②



Recursively define $z_3 = \mathcal{Z}\left(\binom{z_1}{z_2} \cdot \binom{w_{13}}{w_{23}} + b_3\right)$

Let $\underline{z}^L$ denote the last layer of nodes

and $\underline{w}^L$ denote the last layer of weights.

Output: $\square$

- linear regression

$$Y = \underline{w}^L \cdot \underline{z}^L + b^L$$

- Classification

$$Y_i = \exp[w^{L_i} \cdot z^L] / \sum_j \exp[w^{L_j} \cdot z^L].$$

③ Non-linear function approximation

$Y = f^*(x) + \text{noise}$, f^* is the "true" mapping

$Y = f(x;\ \underbrace{\underline{w}, \underline{b}}_{\text{param. } \theta})$.

Cost function $C = \frac{1}{n} \sum_x [Y(x) - f(x;\theta)]^2$

Learning θ

- Init θ_0
- Iterate $\theta_t = \theta_{t-1} - \alpha_t \underbrace{\nabla_\theta C}_{-\frac{2}{n}\sum_x [Y(x) - f(x;\theta)] \times \begin{pmatrix} \frac{\partial f}{\partial w_1} \\ \frac{\partial f}{\partial w_2} \\ \vdots \end{pmatrix}}$

Example.

$$\begin{tikzcd} z_1 \arrow[r, "w_1"] & z_2 \arrow[r, "w_2"] & z_3 \arrow[r, "w_3"] & \square \end{tikzcd}$$

$out = w_3 \cdot z_3$.

$f = w_3 \cdot z_3(w_2 \cdot z_2(w_1 \cdot z_1))$

$$\frac{\partial f}{\partial w_1} = \frac{\partial f}{\partial z_3} \frac{\partial z_3}{\partial z_2} \frac{\partial z_2}{\partial w_1}$$

$$\begin{tikzcd} z_1 \arrow[r, "w_1"] & z_2 \arrow[r] \arrow[dr] & z_3 \arrow[r] & \square \\ & & z_4 \arrow[ur] & \end{tikzcd}$$

$$\frac{\partial f}{\partial w_1} = \frac{\partial f}{\partial z_3} \frac{\partial z_3}{\partial z_2} \frac{\partial z_2}{\partial w_1} + \frac{\partial f}{\partial z_4} \frac{\partial z_4}{\partial z_2} \frac{\partial z_2}{\partial w_1}$$

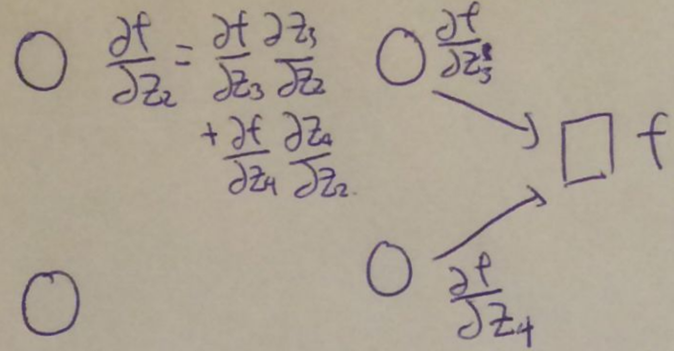
2 path of influence from w_1 to output $\square$.

④ Backprop = chain rule + dynamic programming

Reuse computation.

- If $z_i \to z_j$, compute $\frac{\partial z_j}{\partial z_i}$

- Start from the last layer and go backward.



In general, If $z_k \to z_{k1}, \ldots, z_{km}$ $\quad \frac{\partial f}{\partial z_k} = \sum_{j=1}^{m} \frac{\partial f}{\partial z_{kj}} \frac{\partial z_{kj}}{\partial z_k}$

- Now we have all $\frac{\partial f}{\partial z}$

  If $\xrightarrow{w} z$ $\quad \frac{\partial f}{\partial w} = \frac{\partial f}{\partial z}\frac{\partial z}{\partial w}$

Cost of computing $\nabla_\theta C$ is $O(\#$ of edges$)$