

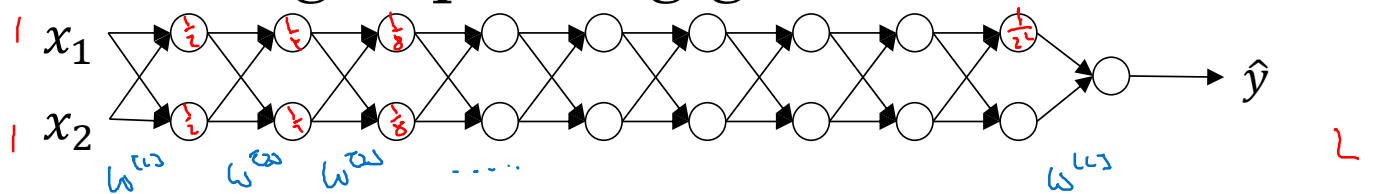


deeplearning.ai

Setting up your
optimization problem

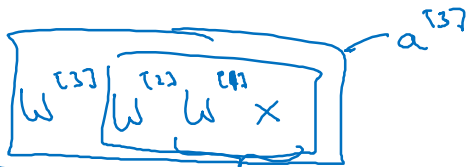
Vanishing/exploding
gradients

Vanishing/exploding gradients



$g(z) = z$ $b^{(1)} = 0$

$\hat{y} = w^{(L)}$



1.5^L
 0.5^L

$w^{(1)} > I$

$w^{(2)} < I$ $\begin{bmatrix} 0.9 & \\ & 0.9 \end{bmatrix}$

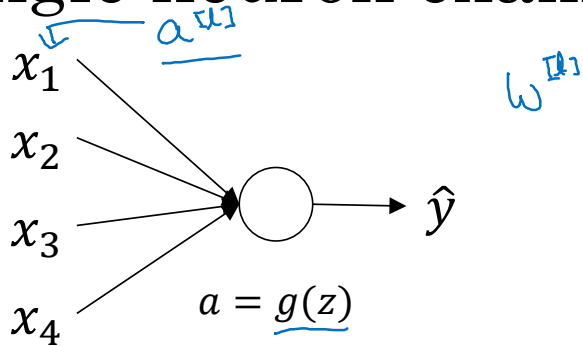
$w^{(L)} = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix}$

$\hat{y} = w^{(L)} \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix}^{L-1} x$

$1.5^{L-1} x$
 $0.5^{L-1} x$

指数级增长, 下降

Single neuron example



$$z = w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

large $n \rightarrow$ Smaller w_i

$$\text{Var}(w_i) = \frac{1}{n} \frac{2}{n}$$

$$\underline{w^{[1]}} = \text{np.random.randn}(\text{shape}) * \text{np.sqrt}\left(\frac{2}{n^{[1-1]}}\right)$$

ReLU $g^{[1]}(z) = \text{ReLU}(z)$

Other variants:

tanh

$$\frac{1}{n^{[1-1]}}$$

Xavier initialization \uparrow

$$\sqrt{\frac{2}{n^{[1-1]} + n^{[1]}}}$$

\uparrow