Vanishing or exploding gradients

=> very deep networks with L layers

$$\boxed{\bar{y} = w^{L} w^{L-1} \ldots w^{2} w^{1} x}$$

$w > 1$

$$w = \begin{bmatrix} 1.5 & 0 \\ 0 & 1.5 \end{bmatrix}$$

$\bar{y}$ will be very large number

$w < 1$

$$w = \begin{bmatrix} 0.8 & 0 \\ 0 & 0.8 \end{bmatrix}$$

$\bar{y}$ will be very small number

Hard to train

Weight initialization

n : # of input features

w : weight matrices

$\text{variance}(w) = \frac{1}{n}$ can be hyperparameter

Relu activation

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

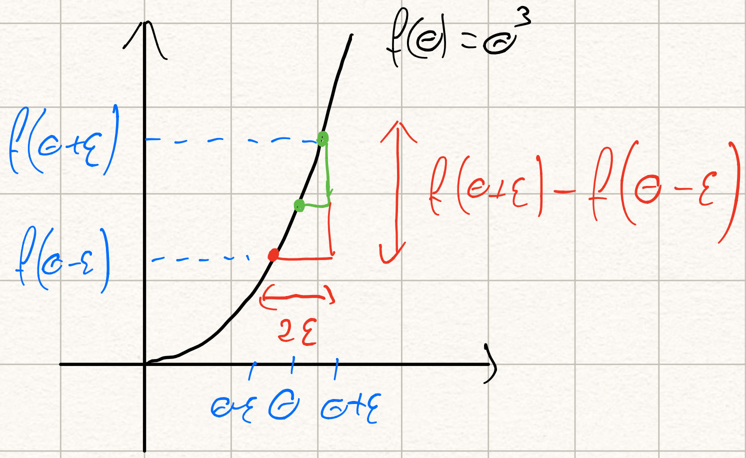
use 2 with relu

n input features from layer $l-1$

Tanh activation

$$sqrt\left(\frac{1}{n^{l-1}}\right)$$ "xavier initialization"

Numerical approximation of gradients



$$\boxed{f'(\theta) = \lim_{\varepsilon \to 0} \frac{f(\theta+\varepsilon) - f(\theta-\varepsilon)}{2\varepsilon}}$$

approximation error:

$\varepsilon := 0{,}01 \Rightarrow \dfrac{1.01^3 - 0.99^3}{2 \cdot 0.01} \approx 3.0001$

$\theta := 1 \Rightarrow$

$f'(\theta) = 3 \cdot \theta^2 = 3$

very close

With one side approximation:

$$\frac{f(\theta+\varepsilon) - f(\theta)}{\varepsilon} \Rightarrow \frac{1.01^3 - 1^3}{0{,}01} \approx \frac{0{,}0303}{0.01} = 3.03$$

approximation error

=> using two side calculation => more accurate approximation

Gradient checking

Reshape $w^1, b^1, w^2, b^2 \dots w^L b^L$ into a big vector θ

- $J(w^1, b^1 \dots w^L, b^L) = J(\theta)$ concatenate

Reshape $dw^1, db^1 \dots dw^L, db^L$ into a big vector $d\theta$

Is $d\theta$ the gradient of J?

$\Downarrow$

for each i:

$$d\theta_{approx}[i] = \frac{J(\theta_1 \dots \theta_{i+\varepsilon}, \dots \theta_L) - J(\theta_1 \dots \theta_{i-\varepsilon} \dots \theta_L)}{2\varepsilon}$$

$$\approx d\theta[i] = \frac{\partial J}{\partial \theta_i}$$

Is $d\theta_{approx} \approx d\theta$?

if $\varepsilon = 10^{-7}$?

check : $$\frac{\|d\theta_{approx} - d\theta\|_2}{\|d\theta_{approx}\|_2 + \|d\theta\|_2} \approx$$

10^{-7} great

$\vdots$

10^{-5} maybe

$\vdots$

10^{-3} worry

Don't use gradient checking in training only to debug

Doesn't work with dropout