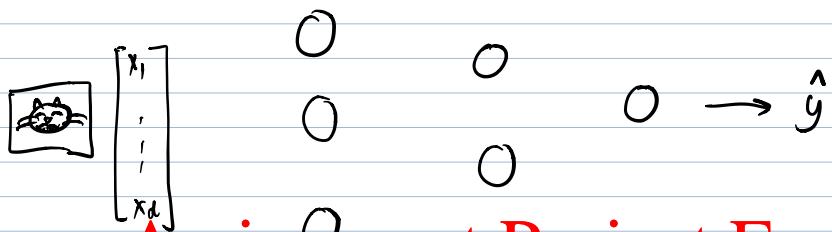


Deep Learning

- ① Logistic Regression with a NN mindset
- ② Neural Networks
- ③ Backpropagation
- ④ Improving your NN



Assignment Project Exam Help

$$z^{(1)} = w^{(1)}x + b^{(1)}$$

$$a^{(1)} = \sigma(z^{(1)})$$

$$z^{(2)} = w^{(2)}a^{(1)} + b^{(2)}$$

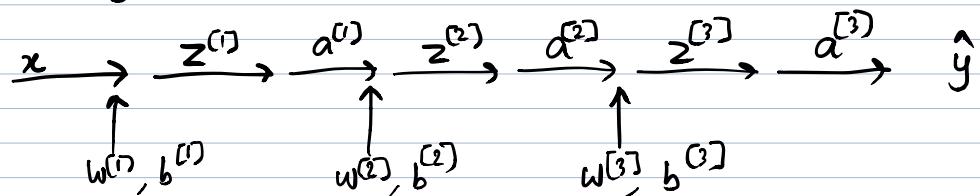
$a^{(2)} = \sigma(z^{(2)})$

$$z^{(3)} = w^{(3)}a^{(2)} + b^{(3)}$$

$$a^{(3)} = \sigma(z^{(3)})$$

$$\hat{y} = a^{(3)}$$

Optimizing $w^{(1)}, w^{(2)}, w^{(3)}, b^{(1)}, b^{(2)}, b^{(3)}$



$$\text{Loss fn } J(\hat{y}, y) = \frac{1}{n} \sum_{i=1}^n \ell^{(i)}$$

$$\mathcal{L}^{(l)} = -[y^{(l)} \log \hat{g}^{(l)} + (1-y^{(l)}) \log (1-\hat{g}^{(l)})]$$

$$\frac{\partial J}{\partial w^{(2)}}$$

$$\frac{\partial \mathcal{L}}{\partial w^{(2)}}$$

Backward Propagation

$l = 1 \dots 3$

$$w^{(l)} = w^{(l)} - \alpha \frac{\partial J}{\partial w^{(l)}}$$

$$b^{(l)} = b^{(l)} - \alpha \frac{\partial J}{\partial b^{(l)}}$$

$\frac{\partial J}{\partial w^{(3)}} \cdot \frac{\partial J}{\partial a^{(3)}} \cdot \frac{\partial a^{(3)}}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial w^{(2)}}$

$$\frac{\partial J}{\partial w^{(2)}} = \frac{\partial J}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial a^{(2)}} \cdot \frac{\partial a^{(2)}}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial w^{(1)}}$$

$$\frac{\partial J}{\partial w^{(1)}} = \frac{\partial J}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial a^{(1)}} \cdot \frac{\partial a^{(1)}}{\partial z^{(0)}} \cdot \frac{\partial z^{(0)}}{\partial w^{(0)}}$$

$$\cancel{\frac{\partial J}{\partial w^{(3)}}} \frac{\partial \mathcal{L}^{(l)}}{\partial w^{(3)}} =$$

$$\mathcal{L}^{(l)} = -[y^{(l)} \log \hat{g}^{(l)} + (1-y^{(l)}) \log (1-\hat{g}^{(l)})]$$

$$\frac{\partial \mathcal{L}^{(l)}}{\partial w^{(3)}} = - \left[y^{(l)} \frac{\partial}{\partial w^{(3)}} (\log \sigma(w^{(3)} a^{(2)} + b^{(3)})) \right]$$

$$+ (1-y^{(l)}) \frac{\partial}{\partial w^{(3)}} (\log (1-\sigma(w^{(3)} a^{(2)} + b^{(3)}))) \right]$$

$$= - \left[y^{(l)} \cdot \frac{1}{\sigma^{(3)}} a^{(3)} (1-a^{(3)}) a^{(2)T} \right]$$

$$+ (1-y^{(l)}) \frac{1}{1-\sigma^{(3)}} (-1) a^{(3)} (1-a^{(3)}) a^{(2)T} \right]$$

$$\log' x = \frac{1}{x}, \quad \sigma'(x) = \sigma(x)(1-\sigma(x))$$

$$\frac{\partial \log \sigma(\cdot)}{\partial w^{(3)}} = \frac{1}{\sigma(\cdot)} \cdot \frac{\partial \sigma(\cdot)}{\partial w^{(3)}}$$

$$\frac{\partial \sigma(w^{(3)}a^{(2)} + b^{(3)})}{\partial w^{(3)}} = a^{(3)}(1-a^{(3)}) \cdot \frac{\partial (w^{(3)}a^{(2)} + b^{(3)})}{\partial w^{(3)}}$$

$$(1,2) \quad \frac{\partial}{\partial w^{(3)}} \underbrace{(w^{(3)}a^{(2)} + b^{(3)})}_{(1,1)} = a^{(2)T}$$

$$\frac{\partial \lambda^{(i)}}{\partial w^{(3)}} = -[y^{(i)}(1-a^{(2)})a^{(2)T} - (1-y^{(i)})a^{(3)}a^{(2)T}]$$

Assignment Project Exam Help

$$\frac{\partial J}{\partial w^{(3)}} = -\frac{1}{n} \sum_{i=1}^n [y^{(i)} - a^{(3)}] a^{(2)T}$$

$$\frac{\partial \lambda}{\partial w^{(2)}} = \frac{\partial \lambda}{\partial a^{(2)}} \cdot \frac{\partial a^{(2)}}{\partial z^{(3)}} \cdot \frac{\partial z^{(3)}}{\partial a^{(2)}} \cdot \frac{\partial a^{(2)}}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial w^{(2)}}$$

$$\frac{\partial \lambda}{\partial w^{(3)}} = \frac{\partial \lambda}{\partial z^{(3)}} \cdot \frac{\partial z^{(3)}}{\partial w^{(3)}} \quad a^{(2)T}$$

$$\frac{\partial \lambda^{(i)}}{\partial w^{(2)}} = \frac{\partial \lambda}{\partial a^{(2)}} \cdot \frac{\partial a^{(2)}}{\partial z^{(3)}} \cdot \frac{\partial z^{(3)}}{\partial w^{(3)}} = (a^{(3)} - y^{(i)}) w^{(3)T} a^{(2)}(1-a^{(2)}) a^{(1)T}$$

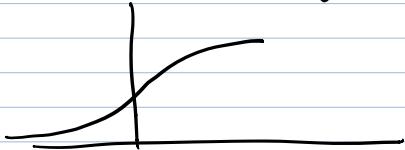
$$= w^{(3)T} * a^{(2)}(1-a^{(2)}) (a^{(3)} - y^{(i)}) \cdot a^{(1)T}$$

\uparrow
 element
 wise
 product

$$\frac{\partial J}{\partial w^{(2)}} = \frac{1}{n} \sum_{i=1}^n \frac{\partial L^{(i)}}{\partial w^{(2)}}$$

③ Improving your NN

A) Activation functions



Sigmoid: $\sigma(z) = \frac{1}{1+e^{-z}}$

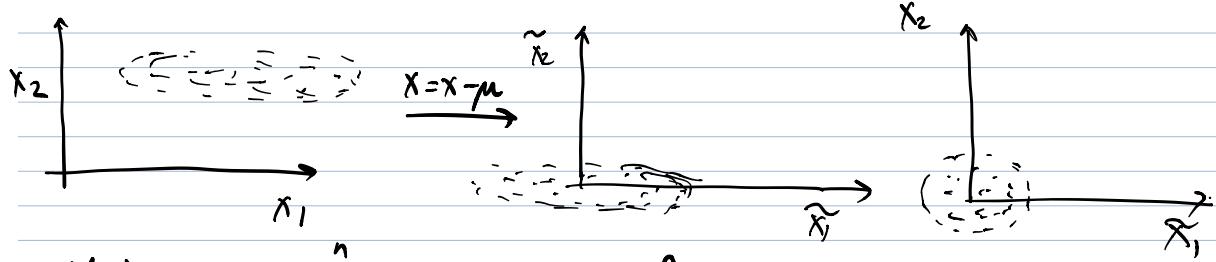
$$\sigma'(z) = \sigma(z)(1-\sigma(z))$$

~~$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$~~

<https://powcoder.com>

B) Initialization methods

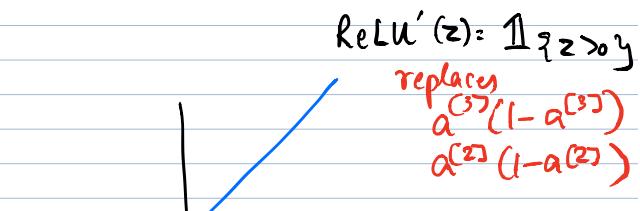
Normalizing input



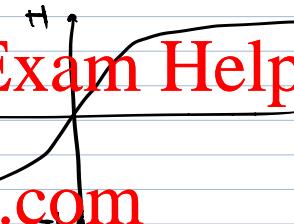
$$\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = \mu = \frac{1}{n} \sum_{i=1}^n x^{(i)}$$

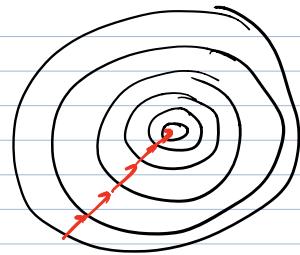
$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (x^{(i)})^2$$

$$X = \frac{X}{\sigma} \quad X = \Sigma^{-1} X$$

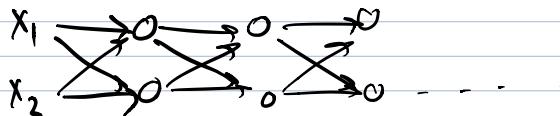


$$\text{ReLU}(z) = \begin{cases} 0 & \text{if } z \leq 0 \\ z & \text{if } z > 0 \end{cases}$$





Vanishing / Exploding gradients



Assume initialization $b=0$

Activation f^n id $z \mapsto z$

$$\hat{y} = w^{[L]} a^{[L-1]} = w^{[L]} \cdot b^{[L-1]} \cdot a^{[L-2]}$$

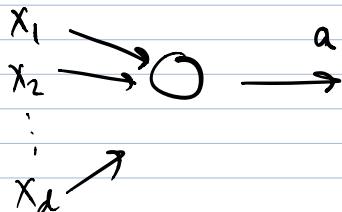
Assignment Project Exam Help

$$w^{[L]} = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \text{ Add WeChat powcoder}$$

- avoid by initializing weights close to 1

Example with 1 neuron



$$a = \sigma(z)$$

$$z = w_1 x_1 + \dots + w_d x_d$$

large $d \rightarrow$ small w_i

$$w_i \sim \frac{1}{d}$$

$$w_i = np.random.randn(\text{shape}) * np.sqrt\left(\frac{1}{n^{(a-1)}}\right)$$

for sigmoid

for ReLU : 2 instead of 1

Xavier Initialization

$$w^{[L]} \sim \sqrt{\frac{1}{n^{[L-1]}}} \text{ for tanh}$$

He Initialization

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

↑ ↗
backward forward
prop. prop.

Assignment Project Exam Help

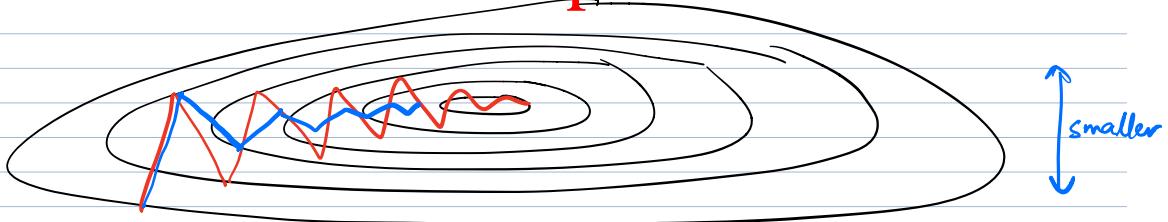
Gradient Descent

Stochastic Gradient Descent

<https://powcoder.com>

Mini Batch gradient Descent

Add WeChat powcoder



Momentum

$$w = w - \alpha \frac{\partial L}{\partial w}$$

$$\bar{U} = \beta U + (1-\beta) \frac{\partial L}{\partial w}$$

$$w = w - \alpha \bar{U}$$

larger

current gradient : $(1-\beta)$

last gradient : $\beta(1-\beta)$

2nd

$\beta^2(1-\beta)$

⋮