



deeplearning.ai

Deep Neural Networks

Forward and backward
propagation

Forward propagation for layer l

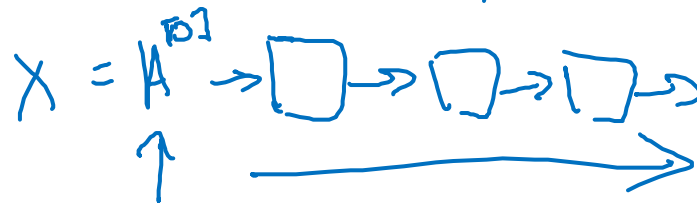
→ Input $a^{[l-1]}$ ←

→ Output $a^{[l]}$, cache ($z^{[l]}$)

$$z^{[l]} = W^{[l]} \cdot a^{[l-1]} + b^{[l]}$$

$$a^{[l]} = g^{[l]}(z^{[l]})$$

$a^{[0]}$
 $A^{[0]}$



Vectorized:

$$z^{[l]} = W^{[l]} \cdot A^{[l-1]} + b^{[l]}$$

$$A^{[l]} = g^{[l]}(z^{[l]})$$

Backward propagation for layer l

→ Input $da^{[l]}$

→ Output $da^{[l-1]}$, $dW^{[l]}$, $db^{[l]}$

$$dz^{[l]} = da^{[l]} * g^{[l]'}(z^{[l]})$$

$$dW^{[l]} = dz^{[l]} \cdot \underline{a^{[l-1]}}$$

$$db^{[l]} = dz^{[l]}$$

$$\underline{da^{[l-1]}} = W^{[l]T} \cdot dz^{[l]}$$

$$\underline{dz^{[l+1]}} = W^{[l+1]T} dz^{[l]} * g^{[l+1]'}(z^{[l+1]})$$

$$dz^{[l]} = \underline{dA^{[l]}} * g^{[l]'}(z^{[l]})$$

$$\underline{dW^{[l]}} = \frac{1}{n} dz^{[l]} \cdot A^{[l-1]T}$$

$$\underline{db^{[l]}} = \frac{1}{n} \text{np.sum}(dz^{[l]}, \text{axis}=1, \text{keepdims}=\text{True})$$

$$\underline{dA^{[l-1]}} = W^{[l]T} \cdot dz^{[l]}$$

Summary

