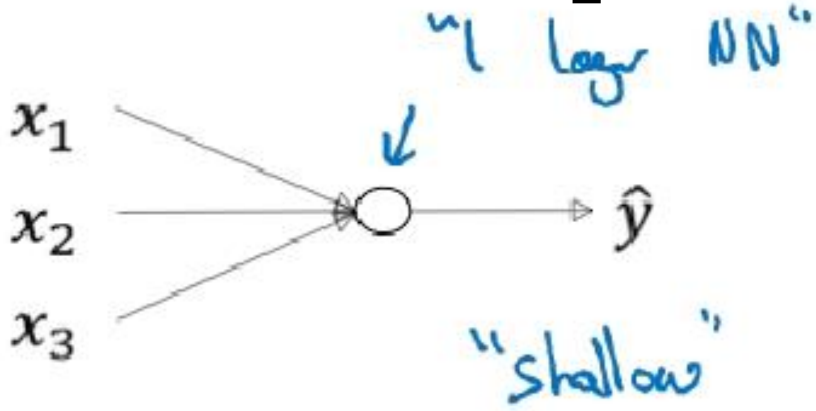
deeplearning.ai

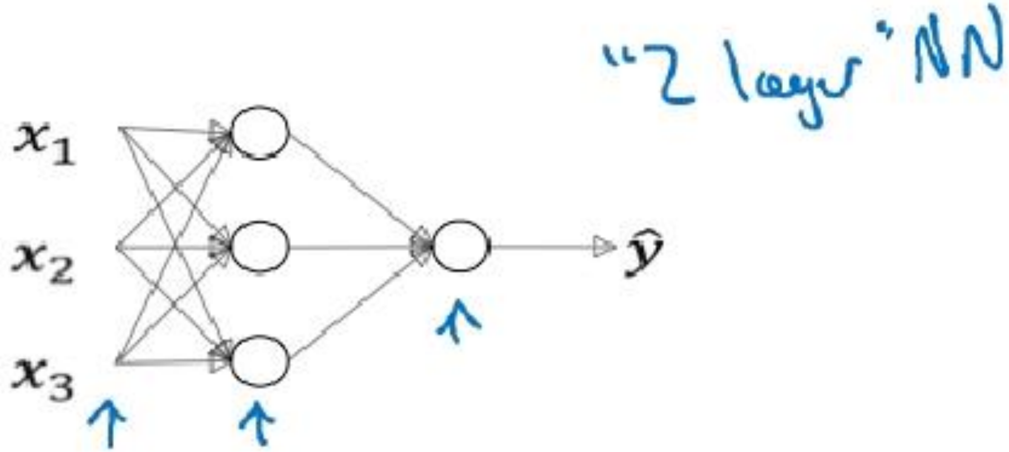
Deep Neural Networks

Deep L-layer Neural network

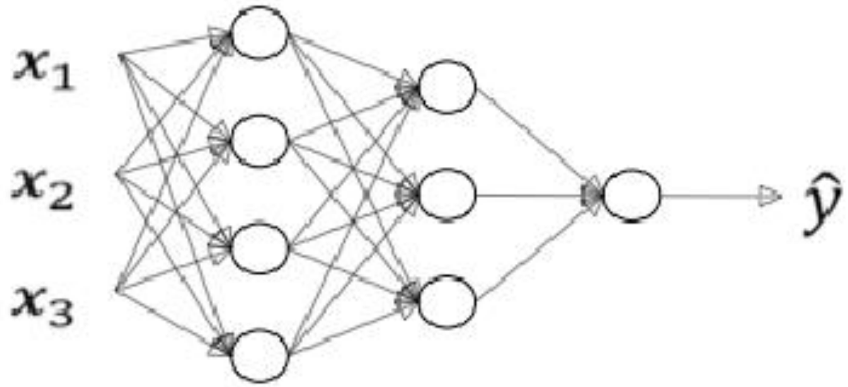
What is a deep neural network?



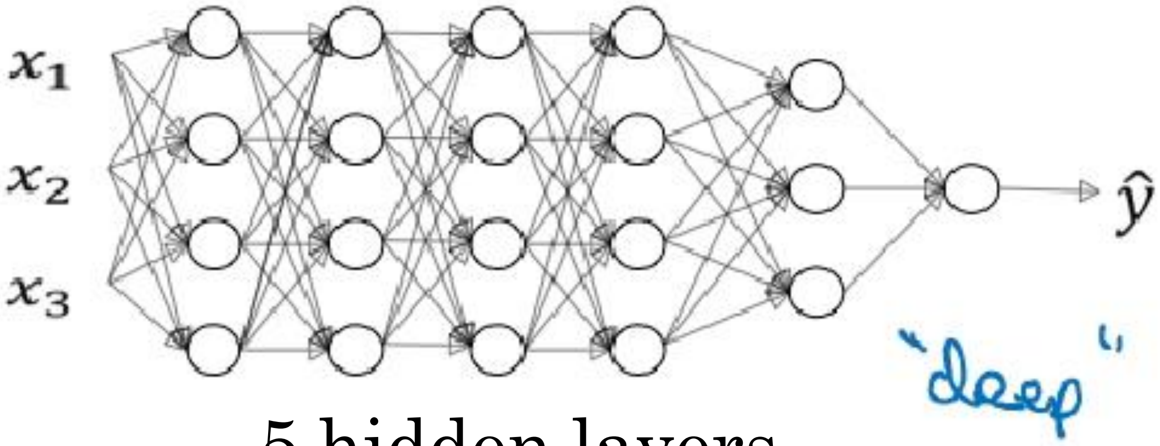
logistic regression



1 hidden layer



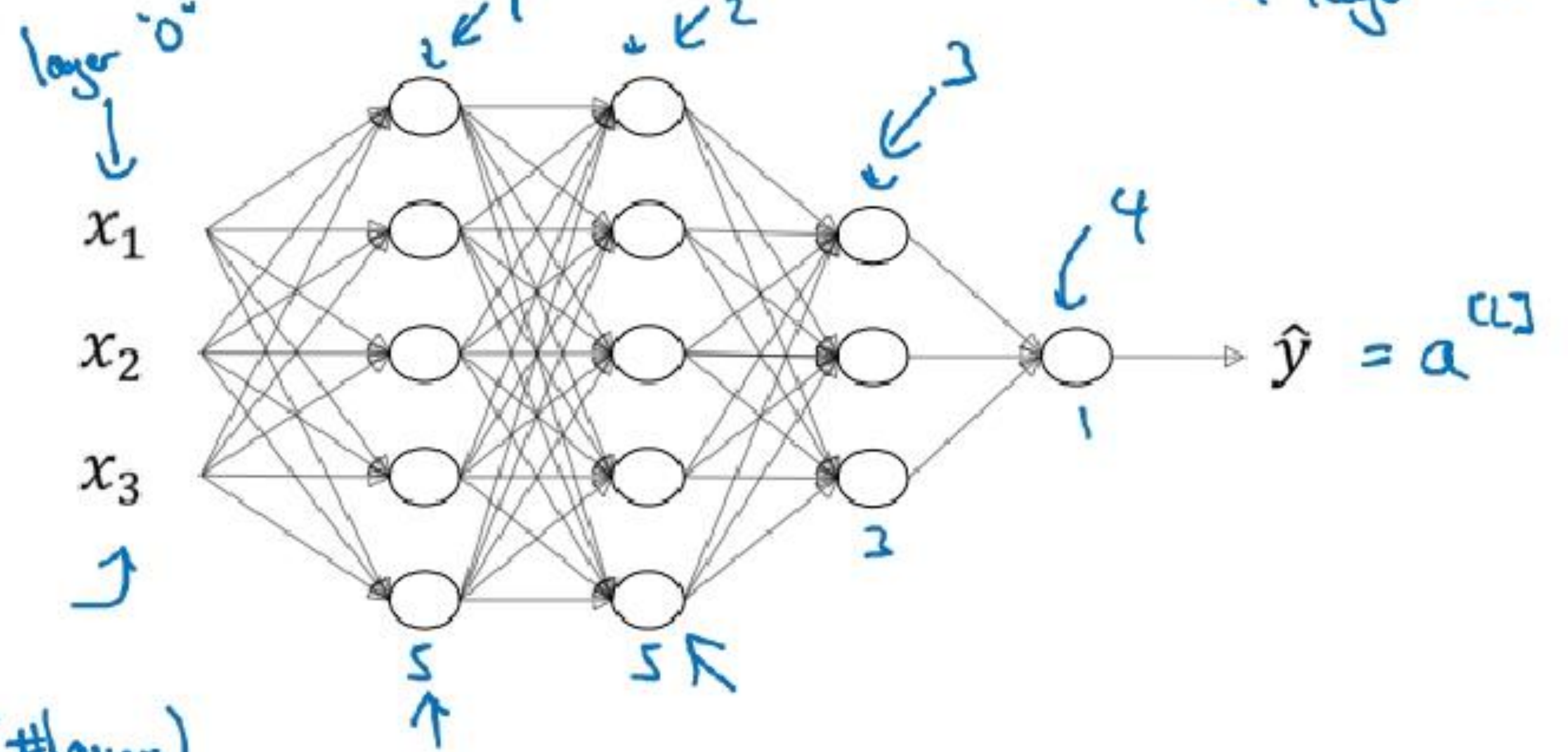
2 hidden layers



5 hidden layers

Deep neural network notation

4 layer NN



$X = a^{[0]}$

$L = 4$ (#layers)

$n^{[l]}$ = #units in layer l

$n^{[1]} = 5$, $n^{[2]} = 5$, $n^{[3]} = 3$, $n^{[4]} = n^{[L]} = 1$

$n^{[0]} = n_x = 3$

$a^{[l]}$ = activations in layer l

$a^{[l]} = g^{[l]}(z^{[l]})$, $W^{[l]}$ = weights for $\underline{z^{[l]}}$

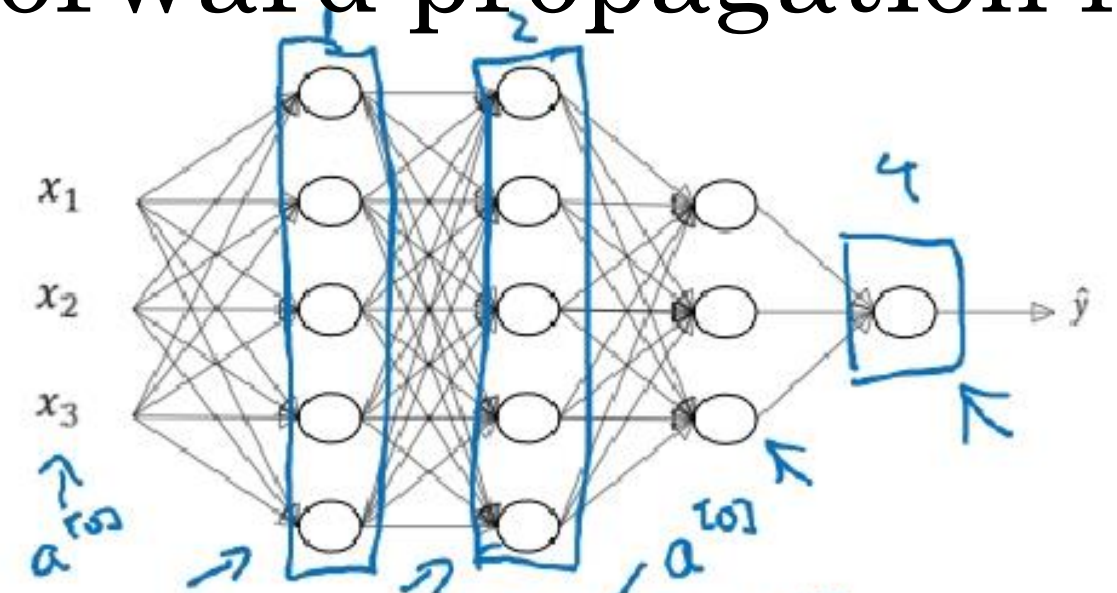
$b^{[l]}$

deeplearning.ai

Deep Neural Networks

Forward Propagation in a Deep Network

Forward propagation in a deep network



$a^{[0]}$

x : $z^{[1]} = W^{[1]} a^{[0]} + b^{[1]}$

$a^{[1]} = g^{[1]}(z^{[1]})$

$z^{[2]} = W^{[2]} a^{[1]} + b^{[2]}$

$a^{[2]} = g^{[2]}(z^{[2]})$

....

$z^{[4]} = W^{[4]} a^{[3]} + b^{[4]}$, $a^{[4]} = g^{[4]}(z^{[4]}) = \hat{y}$

$$\begin{bmatrix} | & | & & | \\ z^{[1](1)} & z^{[1](2)} & \cdots & z^{[1](m)} \\ | & | & & | \end{bmatrix}$$

$A^{[0]} = X$

$$\boxed{\begin{aligned} Z^{[l]} &= W^{[l]} A^{[l-1]} + b^{[l]} \\ A^{[l]} &= g^{[l]}(Z^{[l]}) \end{aligned}}$$

Vectorized:

$X = A^{[0]}$

for $l = 1..4$

$Z^{[1]} = W^{[1]} \cancel{X} A^{[0]} + b^{[1]}$

$A^{[1]} = g^{[1]}(Z^{[1]})$

$Z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$

$A^{[2]} = g^{[2]}(Z^{[2]})$

$\hat{Y} = g(Z^{[4]}) = A^{[4]}$

Andrew Ng

Deep Neural Networks

Getting your matrix dimensions right

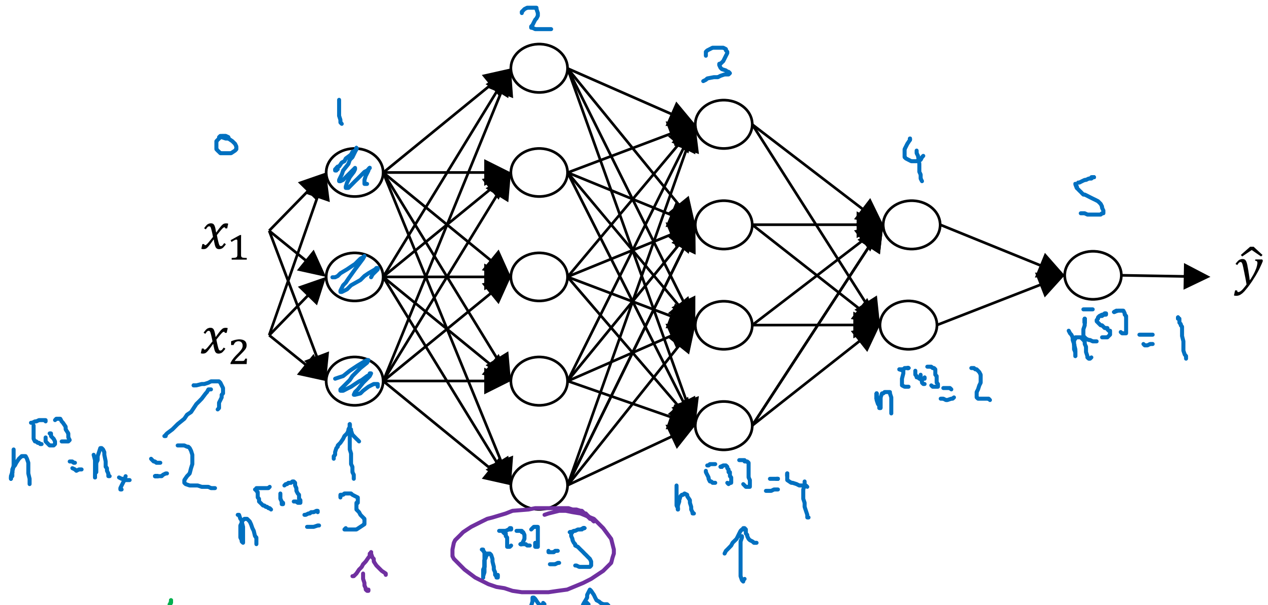
deeplearning.ai

Parameters $W^{[l]}$ and $b^{[l]}$

$L = 5$

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

$a^{[l]}$



$$W^{[l]} : (n^{[l]}, n^{[l-1]})$$
$$b^{[l]} : (n^{[l]}, 1)$$
$$dW^{[l]} : (n^{[l]}, n^{[l-1]})$$
$$db^{[l]} : (n^{[l]}, 1)$$

$$z^{[1]} = \boxed{W^{[1]} \cdot x} + \boxed{b^{[1]}}$$

$(3,1) \leftarrow (3,2)\quad(2,1)\quad(3,1)$

$(n^{[1]},1)\quad(n^{[1]},n^{[0]})\quad(n^{[0]},1)\quad(n^{[1]},1)$

$$\begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \end{bmatrix}$$

$W^{[1]} : (n^{[1]}, n^{[0]})$

$W^{[2]} : (5, 3)\quad (n^{[2]}, n^{[1]})$

$$z^{[2]} = \boxed{W^{[2]} \cdot a^{[1]}} + \boxed{b^{[2]}}$$

$\rightarrow (5,1)\quad(5,3)\quad(3,1)\quad(5,1)$

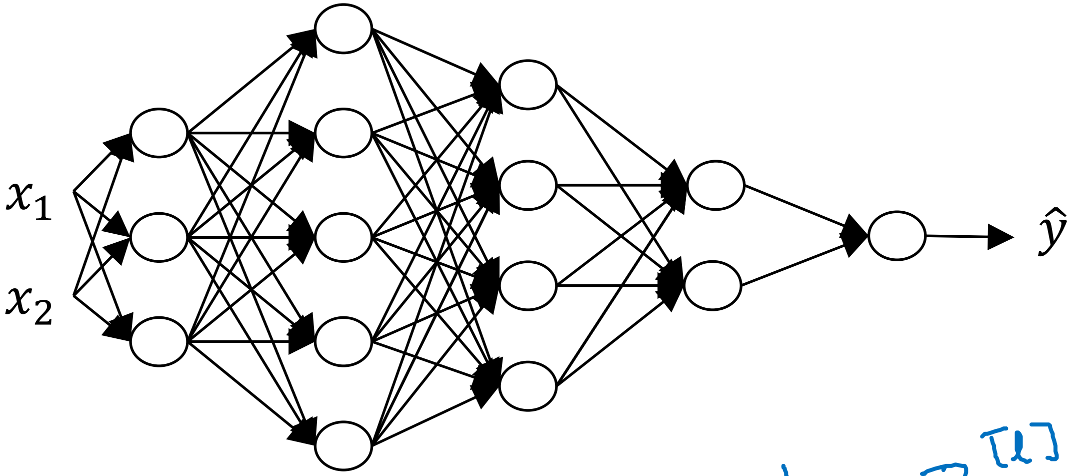
$(n^{[2]},1)$

$W^{[3]} : (4, 5)$

$W^{[4]} : (2, 4)\quad,\quad W^{[5]} : (1, 2)$

Andrew Ng

Vectorized implementation



$z^{[1]} = W^{[1]} \cdot x + b^{[1]}$

$(n^{[1]}, 1) \quad (n^{[1]}, n^{[0]}) \quad \underline{(n^{[0]}, 1)} \quad (n^{[1]}, 1)$

$\left[z^{[1](1)} \; z^{[1](2)} \; \cdots \; z^{[1](m)} \right]$

$Z^{[1]} = W^{[1]} \cdot X + b^{[1]}$

$\underline{(n^{[1]}, m)} \quad \underbrace{(n^{[1]}, \underline{n^{[0]}}) \quad (\underline{n^{[0]}}, m)} \quad \underline{(n^{[1]}, 1)}$

$(n^{[1]}, m)$

$z^{[l]}, a^{[l]} : (n^{[l]}, 1)$

$Z^{[l]}, A^{[l]} : (n^{[l]}, m)$

$l = 0 \quad A^{[0]} = X = (n^{[0]}, m)$

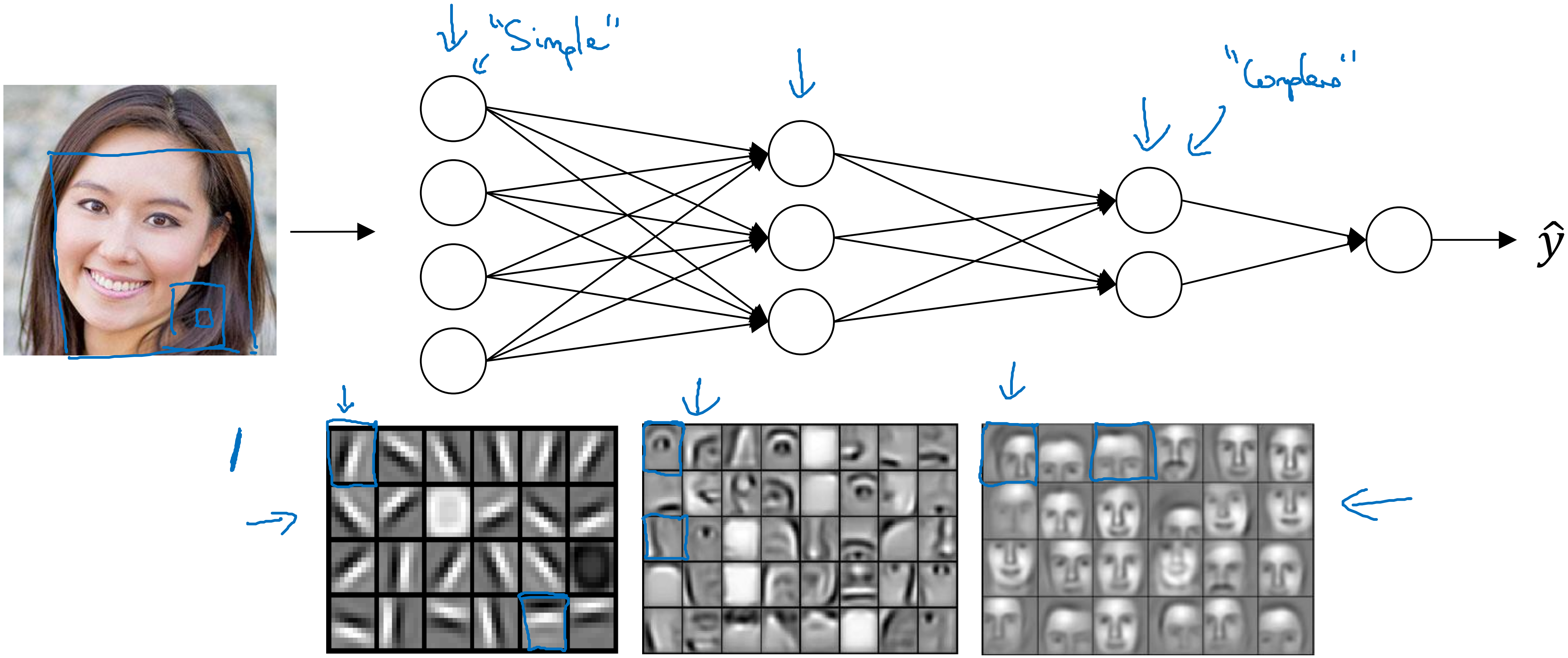
$dZ^{[l]}, dA^{[l]} : (n^{[l]}, m)$

deeplearning.ai

Deep Neural Networks

Why deep representations?

Intuition about deep representation



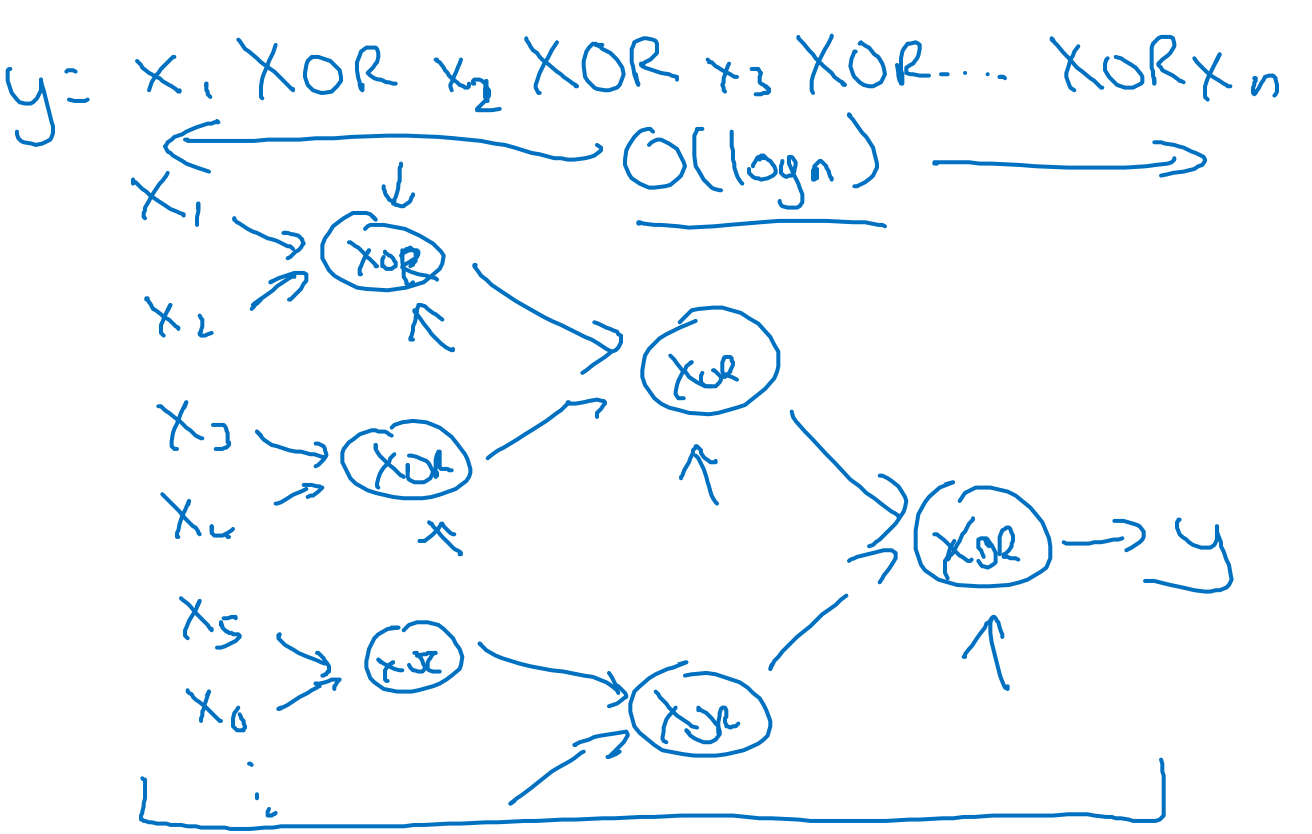
Audio → low level audio waveform → Phonemes C A T → Words → Sentence phrase

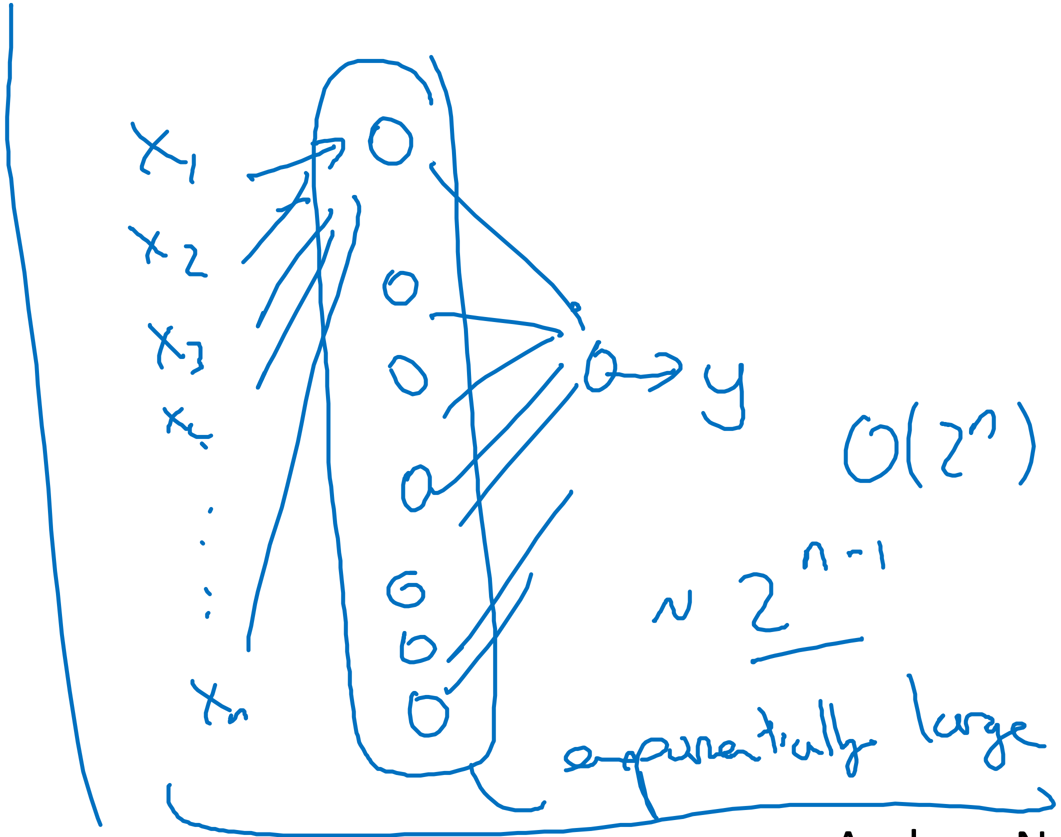
Andrew Ng

Circuit theory and deep learning

Informally: There are functions you can compute with a "small" L-layer deep neural network that shallower networks require exponentially more hidden units to compute.

$y = x_1 \text{ XOR } x_2 \text{ XOR } x_3 \text{ XOR} \dots \text{ XOR } x_n$





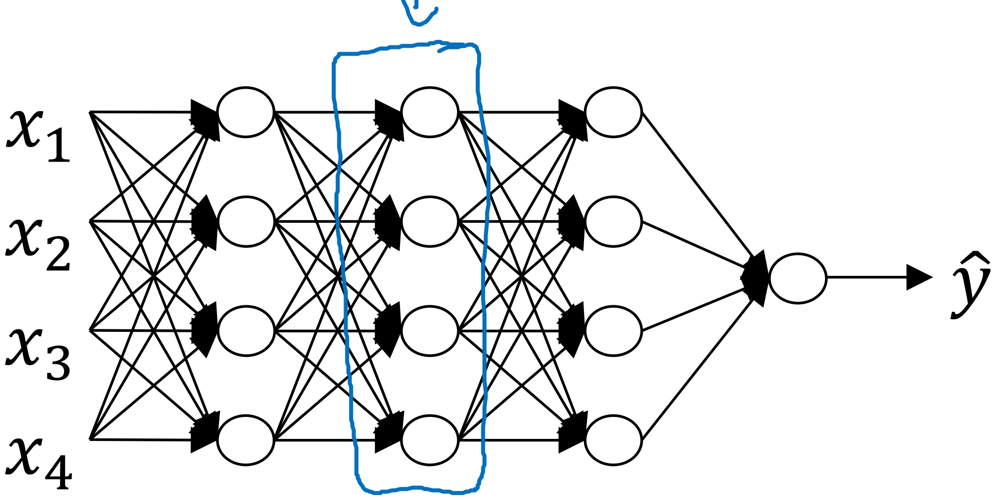
Andrew Ng

deeplearning.ai

Deep Neural Networks

Building blocks of deep neural networks

Forward and backward functions



layer l: $W^{[l]}, b^{[l]}$

→ Forward: Input $a^{[l-1]}$, output $a^{[l]}$

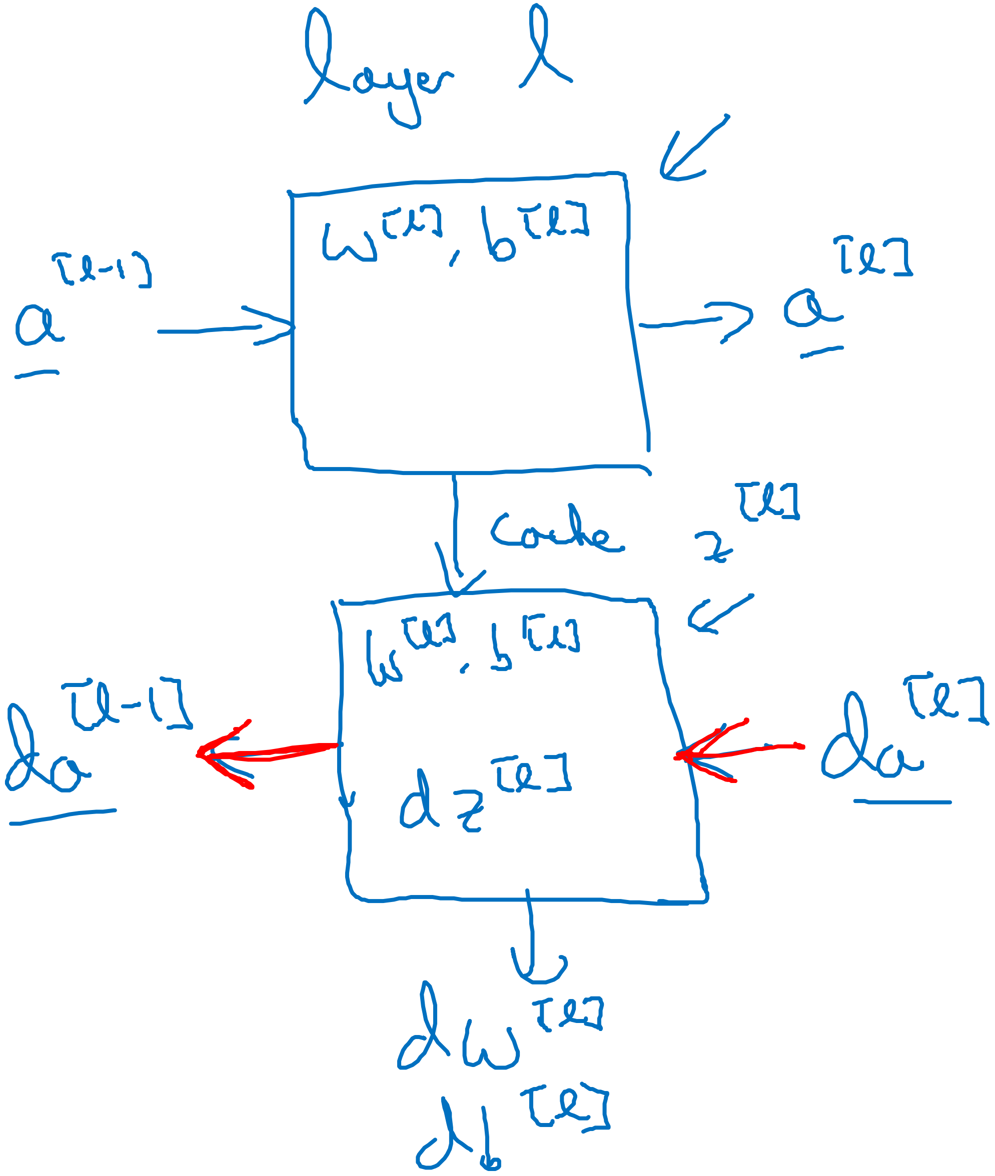
$z^{[l]} = W^{[l]} a^{[l-1]} + b^{[l]}$ cache $z^{[l]}$

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

→ Backward: Input $da^{[l]}$, cache($z^{[l]}$) output $da^{[l-1]}$

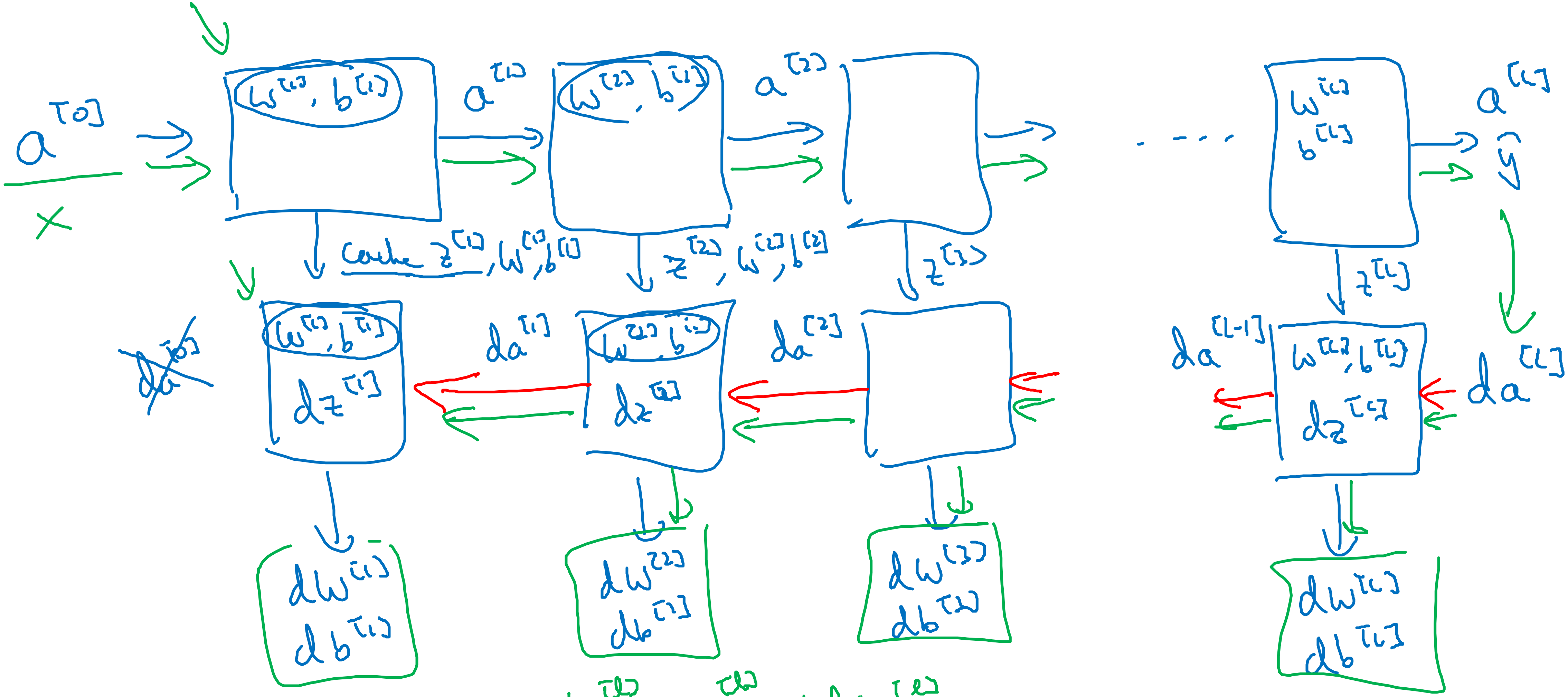
$dW^{[l]}$

$db^{[l]}$



Andrew Ng

Forward and backward functions



$$W^{[l]} := W^{[l]} - \alpha \, dW^{[l]}$$

$$b^{[l]} := b^{[l]} - \alpha \, db^{[l]}$$

Andrew Ng

deeplearning.ai

Deep Neural Networks

Forward and backward propagation

Forward propagation for layer l

→ Input $a^{[l-1]}$ ←

$W^{[l]}, b^{[l]}$

→ Output $a^{[l]}$, cache $(\underline{z^{[l]}})$

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

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

Vectorized:

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

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

$a^{[0]}$

$A^{[0]}$

$$X = A^{[0]} \rightarrow \square \rightarrow \square \rightarrow \square \rightarrow$$

Backward propagation for layer l

- → Input $\underline{da^{[l]}}$
- → Output $\boxed{da^{[l-1]}}$, $dW^{[l]}$, $db^{[l]}$

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

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

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

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

$$\underline{dz^{[l]} = W^{[l+1]T} dz^{[l+1]} * g^{[l]\prime}(z^{[l]})}$$

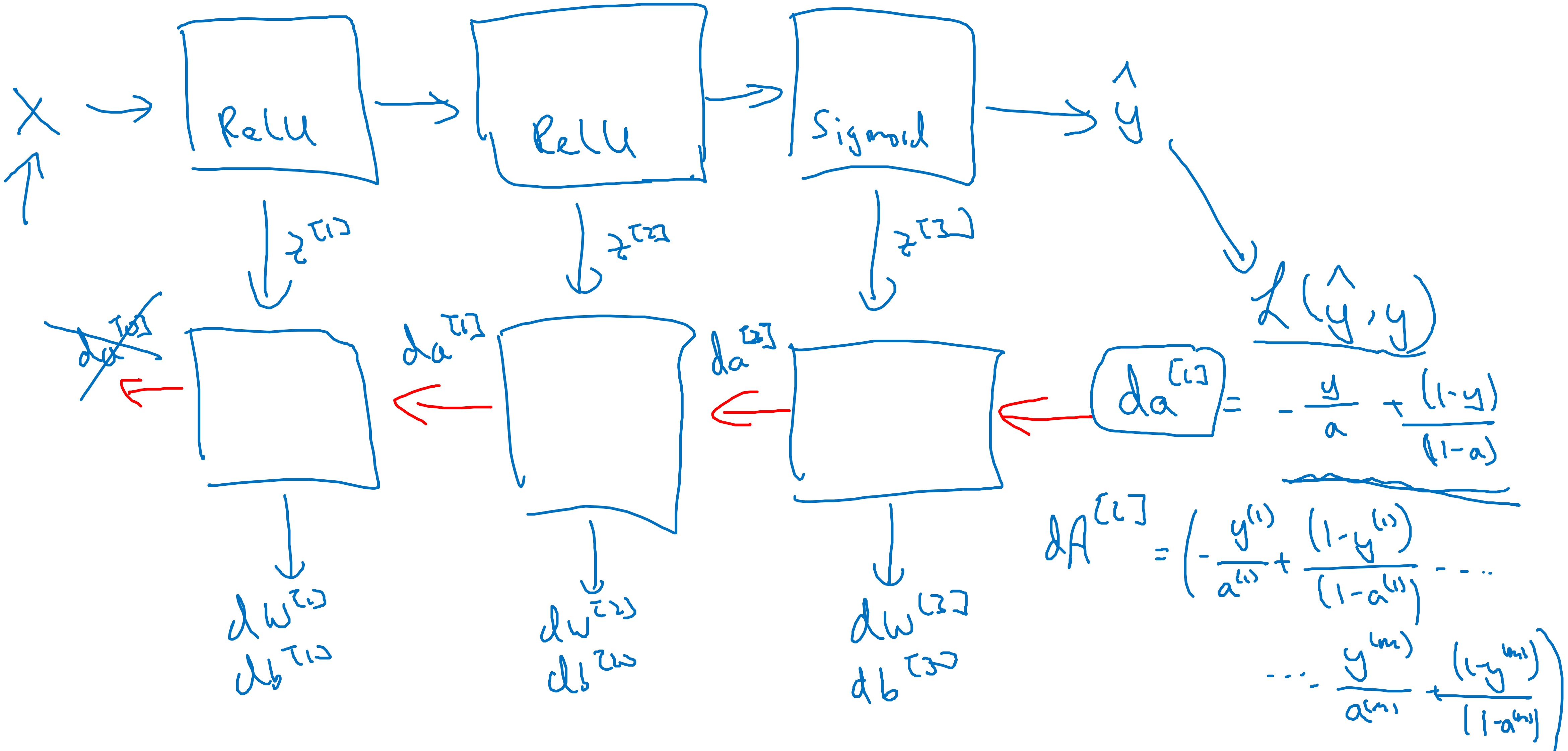
$$dZ^{[l]} = dA^{[l]} * g^{[l]\prime}(Z^{[l]})$$

$$dW^{[l]} = \frac{1}{m} dZ^{[l]} \cdot A^{[l-1]T}$$

$$db^{[l]} = \frac{1}{m} np.sum(dZ^{[l]}, axis=1, keepdims=True)$$

$$\boxed{dA^{[l-1]}} = W^{[l]T} \cdot dZ^{[l]}$$

Summary



$$\boxed{da^{[l]}} = -\frac{y}{a} + \frac{(1-y)}{(1-a)}$$

$$dA^{[l]} = \left(-\frac{y^{(1)}}{a^{(1)}} + \frac{(1-y^{(1)})}{(1-a^{(1)})} \cdots \cdots -\frac{y^{(m)}}{a^{(m)}} + \frac{(1-y^{(m)})}{(1-a^{(m)})}\right)$$

deeplearning.ai

Deep Neural Networks

Parameters vs Hyperparameters

What are hyperparameters?

Parameters: $\underline{W^{[1]}, b^{[1]}, W^{[2]}, b^{[2]}, W^{[3]}, b^{[3]}}$ ...

Hyperparameters: learning rate α

#iterations

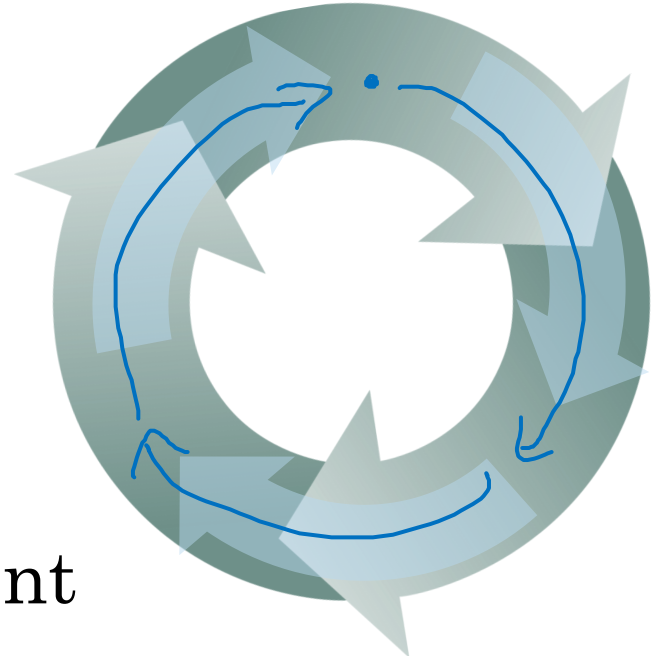
#hidden layers L

#hidden units $n^{[1]}, n^{[2]}, \ldots$

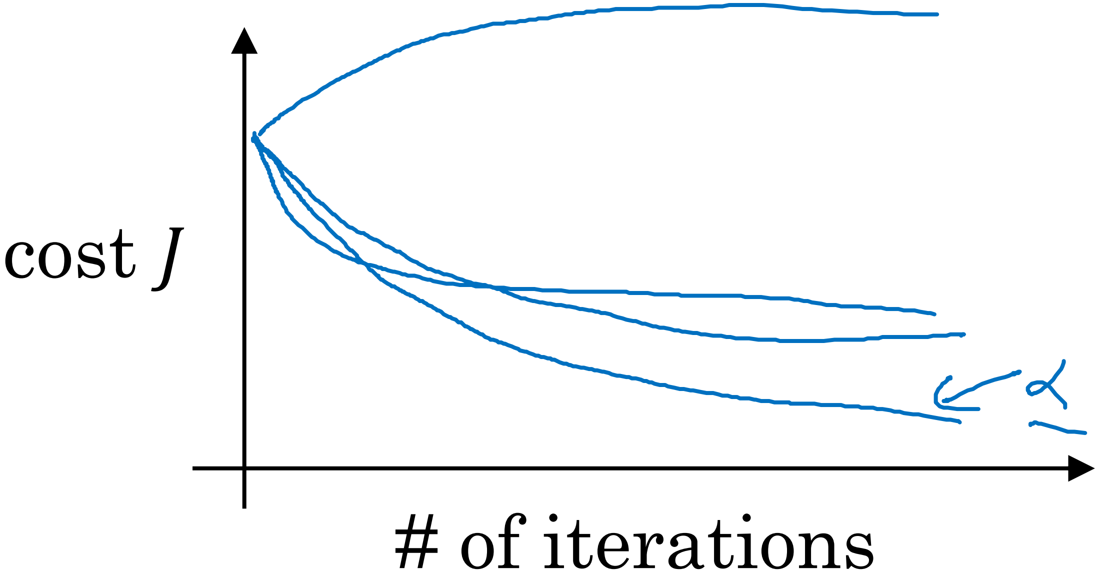
choice of activation function

Later: Momentum, mini-batch size, regularizations, ...

Applied deep learning is a very empirical process



$\alpha =$ ~~0.01~~ 0.05



Vision, Speech, NLP, Ad, Search, Recommendation.

deeplearning.ai

Deep Neural Networks

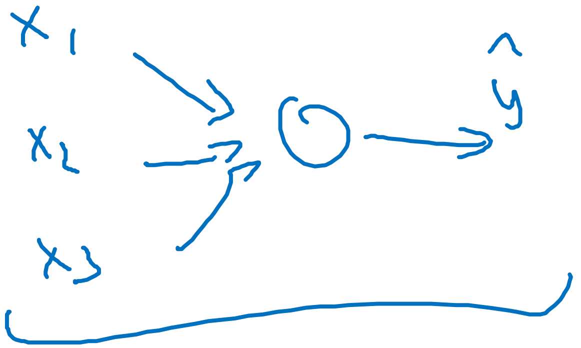
What does this have to do with the brain?

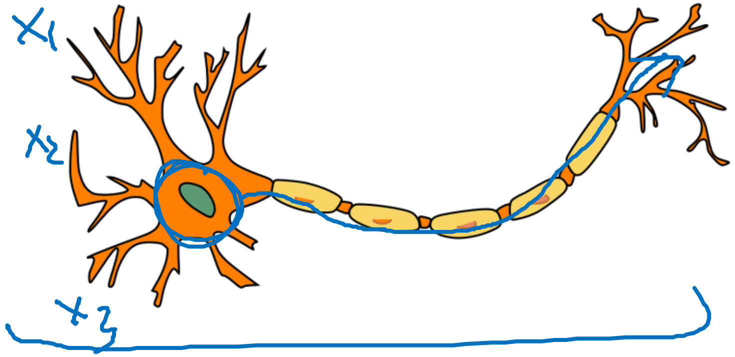
Forward and backward propagation

$$
\boxed{\begin{aligned}
Z^{[1]} &= W^{[1]}X + b^{[1]} \\
A^{[1]} &= g^{[1]}(Z^{[1]}) \\
Z^{[2]} &= W^{[2]}A^{[1]} + b^{[2]} \\
A^{[2]} &= g^{[2]}(Z^{[2]}) \\
&\ \vdots \\
A^{[L]} &= g^{[L]}\left(Z^{[L]}\right) = \hat{Y}
\end{aligned}}
$$

$$
\boxed{\begin{aligned}
dZ^{[L]} &= A^{[L]} - Y \\
dW^{[L]} &= \frac{1}{m} dZ^{[L]} A^{[L-1]^T} \\
db^{[L]} &= \frac{1}{m} np.\mathrm{sum}(\mathrm{d}Z^{[L]}, axis = 1, keepdims = True) \\
dZ^{[L-1]} &= W^{[L]^T} dZ^{[L]} g'^{[L-1]}(Z^{[L-1]}) \\
&\ \vdots \\
dZ^{[1]} &= W^{[2]^T} dZ^{[2]} g'^{[1]}(Z^{[1]}) \\
dW^{[1]} &= \frac{1}{m} dZ^{[1]} A^{[0]^T} \\
db^{[1]} &= \frac{1}{m} np.\mathrm{sum}(\mathrm{d}Z^{[1]}, axis = 1, keepdims = True)
\end{aligned}}
$$

"It's like the brain"





$X \longrightarrow y$

Andrew Ng