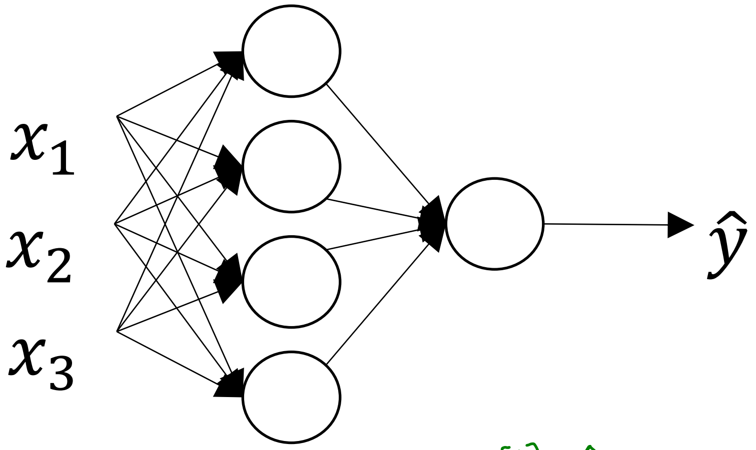
deeplearning.ai

One hidden layer Neural Network

Vectorizing across multiple examples

Vectorizing across multiple examples



$$z^{[1]} = W^{[1]}x + b^{[1]}$$
$$a^{[1]} = \sigma(z^{[1]})$$
$$z^{[2]} = W^{[2]}a^{[1]} + b^{[2]}$$
$$a^{[2]} = \sigma(z^{[2]})$$

$X \longrightarrow a^{[2]} = \hat{y}$

$x^{(1)} \longrightarrow a^{[2](1)} = \hat{y}^{(1)}$

$x^{2} \longrightarrow a^{[2](2)} = \hat{y}^{(2)}$

$x^{3} \longrightarrow a^{[2](3)} = \hat{y}^{(3)}$

$\vdots$

$x^{m} \longrightarrow a^{[2](m)} = \hat{y}^{(m)}$

$a^{[2](i)}$ — $[2]$: layer 2; (i): example i

For m samples

for $i = 1$ to m

$z^{[1](i)} = W^{[1]T} \cdot x^{(i)} + b^{[1]}$

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

$z^{[2](i)} = W^{[2]T} \cdot a^{[1](i)} + b^{[2]}$

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

Andrew Ng

Vectorizing across multiple examples

```
for i = 1 to m:
```

$$z^{[1](i)} = W^{[1]}x^{(i)} + b^{[1]}$$

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

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

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

No For loops

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

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

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

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

Note
w & b remain the same for all the training examples
$w^{[1]}$ is the same for $x^{(1)}$ or $x^{(2)}$....
$w^{[2]}$ is the same for $a^{(1)}$ or $a^{(2)}$....

where

$$Z^{[1]} = \begin{bmatrix} | & | & | & & | \\ z^{[1](1)} & z^{[1](2)} & z^{[1](3)} & \cdots & z^{[1](m)} \\ | & | & | & & | \end{bmatrix}$$

$(m \times m)$ why? → do dim Analysis of $z^{[1]} = w^{[1]T} \cdot x + b^{[1]}$

$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ | & | & & | \end{bmatrix}$$

(n_x, m)

Q = 2nd training example
1st layer
1st member of the layer

P = 1st training eg
1st layer
1st member of that layer

$$A^{[1]} = \begin{bmatrix} | & | & | & & | \\ a^{[1](1)} & a^{[1](2)} & a^{[1](3)} & \cdots & a^{[1](m)} \\ | & | & | & & | \end{bmatrix}$$

dim same as Z $(m \times m)$

hidden unit in layer "L"

training eg.

Andrew Ng

One hidden layer Neural Network

deeplearning.ai

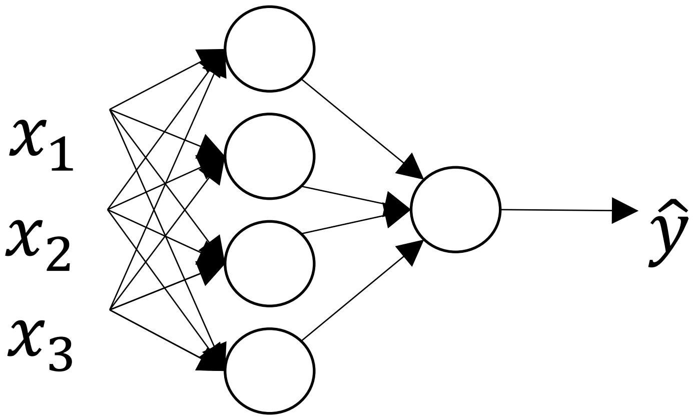
Explanation for vectorized implementation

Justification for vectorized implementation

PTO

Andrew Ng

Recap of vectorizing across multiple examples



$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ | & | & & | \end{bmatrix}$$

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

```
for i = 1 to m
```

$$\boxed{z^{[1](i)} = W^{[1]}x^{(i)} + b^{[1]}}$$

$$\boxed{a^{[1](i)} = \sigma(z^{[1](i)})}$$

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

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

vectorize

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

vectorize

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

etc.

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

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

Andrew Ng