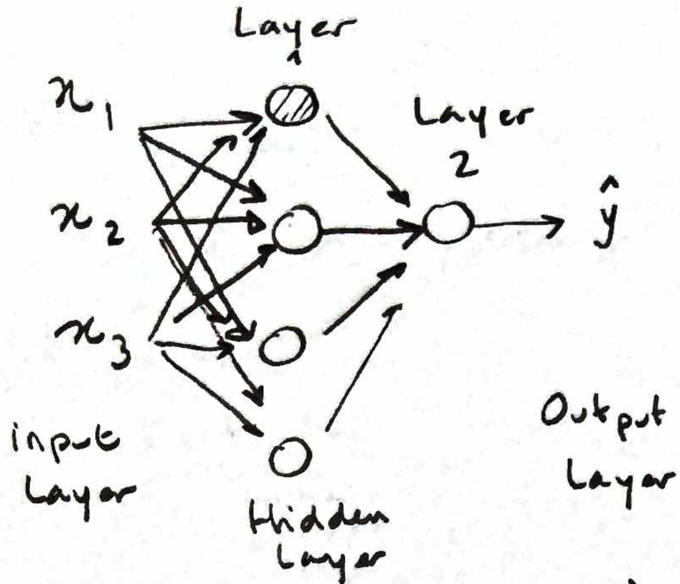
REDES NEURONALES

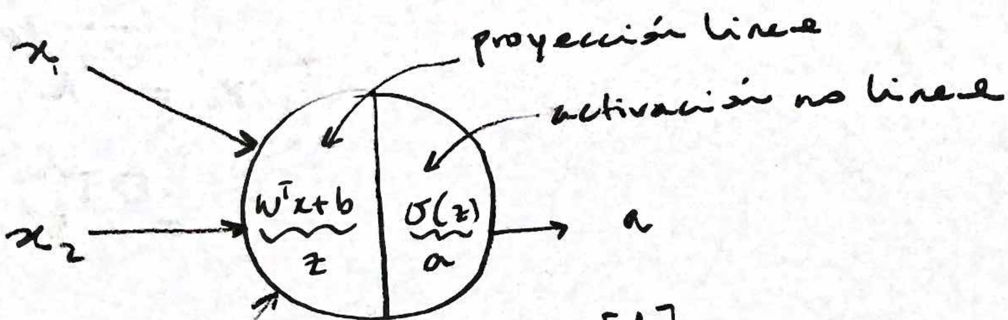


$n_0 = 3$

$n_1 = 4$

$n_2 = 1$

Nomenclatura (Andrew Ng)



$a_i^{[l]}$ l layer

i node

En el ejemplo : $l = 0, 1, 2$

$a^{[0]} = x$

$$a^{[1]} = \begin{bmatrix} a_1^{[1]} \\ a_2^{[1]} \\ a_3^{[1]} \end{bmatrix}$$

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

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

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

n_l = nodos layer l

$w^{[l]} : n_l \times n_{l-1}$

$b^{[l]} : n_l \times 1$

En la red del ejemplo

- PRIMER NODO DEL HIDDEN LAYER

$$z_1^{[1]} = w_1^{[1]T} \underline{x} + b_1^{[1]}$$

$$a_1^{[1]} = \sigma(z_1^{[1]})$$

$$\vdots$$

- OUTPUT DEL HIDDEN LAYER

$$\underset{4\times1}{z^{[1]}} = \underset{4\times3}{W^{[1]}} \underset{3\times1}{\underline{x}} + \underset{4\times1}{b^{[1]}}$$

$$\underset{4\times1}{a^{[1]}} = \sigma(z^{[1]})$$

- OUTPUT DEL SECOND LAYER

$$\underset{1\times1}{z^{[2]}} = \underset{1\times4}{W^{[2]}} \underset{4\times1}{a^{[1]}} + \underset{1\times1}{b^{[2]}}$$

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

- FUNCIONES DE ACTIVACIÓN $\quad a = g(z)$

1) Sigmoid

$$g(z) = \frac{1}{1+e^{-z}} \rightarrow g'(z) = -(1+e^{-z})^{2}(-e^{-z})$$

$$= \frac{e^{-z}}{(1+e^{-z})^{2}} = \cdots = g(z)(1-g(z))$$

$$g'(z) = a(1-a)$$

2) Tanh

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

$$\vdots$$

$$g'(z) = 1-(\tanh(z))^{2}$$

$$= 1-a^{2}$$

ENTRENAMIENTO (BACK PROPAGATION)

El entrenamiento consiste en encontrar los parámetros $\underline{W}^{[l]}$ y $b^{[l]}$

Se tiene una función de costo J que hay que minimizar, por ejemplo:

$$J = \frac{1}{m} \sum_{i=1}^{m} loss(\hat{y}^{(i)}, y^{(i)})$$

$i = 1..m$: muestras de entrenamiento

$\hat{y}^{(i)}$: output de la red neuronal

$y^{(i)}$: output ideal (supervisado)

Ejemplo: $(\hat{y}^{(i)} - y^{(i)})^2$

o bien

$-y \log(\hat{y}) - (1-y) \log(1-\hat{y})$

ALGORITMO DE ENTRENAMIENTO:

0) Se inicializan los parámetros $W^{[l]}$, $b^{[l]}$ con valores random

1) Calcular las predicciones para $i = 1 \ldots m$

$z^{[l]} = W^{[l]} a^{[l-1]} + b^{[l]}$, $a^{[l]} = \sigma(z^{[l]})$

2) Calcular las derivadas:

$dW^{[l]} = \partial J / \partial W^{[l]}$, $db^{[l]} = \partial J / \partial b^{[l]}$

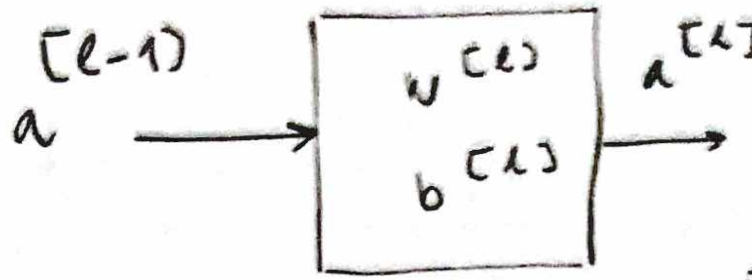
3) Actualizar parámetros:

α: Learning Rate

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

HASTA CONVERGER

PASO 1: FORWARD PROPAGATION

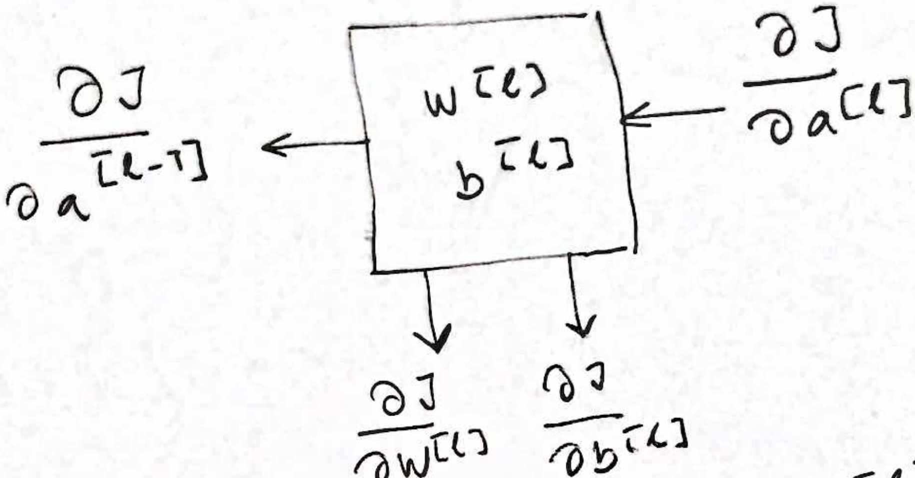


$$z^{[l]} = w^{[l]} a^{[l-1]} + b^{[l]}$$

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

$$a^{[l]} = \sigma(w^{[l]} a^{[l-1]} + b^{[l]})$$

PASO 2: BACKWARD PROPAGATION



$$\frac{\partial J}{\partial a^{[l-1]}} = \frac{\partial J}{\partial z^{[l]}} \cdot \frac{\partial z^{[l]}}{\partial a^{[l-1]}} \to w^{[l]}$$

INPUT

$$\frac{\partial J}{\partial z^{[l]}} = \frac{\partial J}{\partial a^{[l]}} \cdot \frac{\partial a^{[l]}}{\partial z^{[l]}} \to g'(z^{[l]}) = a^{[l]}(1 - a^{[l]})$$

$$\frac{\partial J}{\partial w^{[l]}} = \frac{\partial J}{\partial z^{[l]}} \cdot \frac{\partial z^{[l]}}{\partial w^{[l]}} \to a^{[l-1]}$$

$$\frac{\partial J}{\partial b^{[l]}} = \frac{\partial J}{\partial z^{[l]}} \cdot \frac{\partial z^{[l]}}{\partial b^{[l]}} \to 1$$

PARA LA ULTIMA CAPA

Por ejemplo:

$$J = \frac{1}{2} \sum (\hat{y}^i - y^i)^2 \qquad \hat{y} \leftarrow a^{[L]}$$

$$\frac{\partial J}{\partial a^{[L]}} = \frac{1}{2} \sum 2(a^{[L]} - y^i)$$

$$\frac{\partial J}{\partial a^{[L]}} = \sum (a^{[L]} - y^i)$$