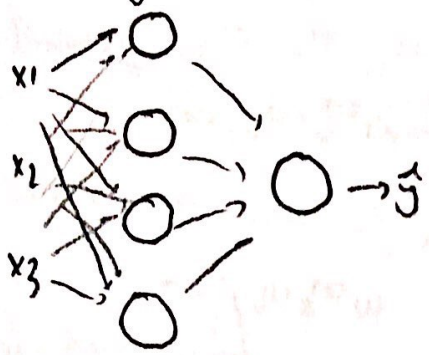
Vectorizing across multiple examples



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

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

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

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

$\vdots$

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

$a^{[2](i)}$ → example i

layer 2

→ for i = 1 to m,

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

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

(n_x, m)

$$Z^{[1]} = W^{[1]} X + b^{[1]}$$
$$A^{[1]} = \sigma(Z^{[1]})$$
$$Z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$$
$$A^{[2]} = \sigma(Z^{[2]})$$

training examples →

hidden units ↓

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

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

ication for vectorized implementation

$z^{[1](1)} = w^{[1]} x^{(1)} + b^{[1]}$ $z^{[1](2)} = w^{[1]} x^{(2)} + b^{[1]}$ $z^{[1](3)} = w^{[1]} x^{(3)} + b^{[1]}$

$w^{[1]} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$ $w^{[1]} x^{(1)} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$ $w^{[1]} x^{(2)} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$ $w^{[1]} x^{(3)} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$

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

X

Gradient descent for neural networks

Parameters: $W^{[1]}, b^{[1]}, W^{[2]}, b^{[2]}$ $\quad n_x = n^{[0]}, n^{[1]}, n^{[2]} = 1$

$(n^{[1]}, n^{[0]}) \; (n^{[1]}, 1) \; (n^{[2]}, n^{[1]}) \; (n^{[2]}, 1)$

Cost function: $J(W^{[1]}, b^{[1]}, W^{[2]}, b^{[2]}) = \frac{1}{m} \sum_{i=1}^{m} \mathcal{L}(\hat{y}, y)$ $\quad \hat{y} = a^{[2]}$

Gradient descent:

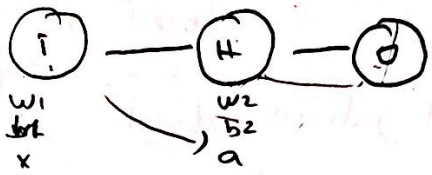
→ Repeat {

Compute predi. $(\hat{y}^{(i)}, i = 1, \dots m)$

$dW^{[1]} = \frac{dJ}{dW^{[1]}}, \quad db^{[1]} = \frac{dJ}{db^{[1]}}$

$W^{[1]} = W^{[1]} - \alpha \, dW^{[1]}$

$b^{[1]} = b^{[1]} - \alpha \, db^{[1]}$



Formulas for computing derivatives

Forward propagation:

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

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

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

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

Backward propagation:

$dZ^{[2]} = A^{[2]} - Y$ $\quad Y = [y^{(1)}, y^{(2)} \dots y^{(m)}]$

$dW^{[2]} = \frac{1}{m} dZ^{[2]} \cdot A^{[1]T}$ $\quad (n^{[2]}, n^{[1]})$

$db^{[2]} = \frac{1}{m} (np.sum(dZ^{[2]}, axis=1, keepdim=True))$ $\quad (n^{[2]}, 1)$

$dZ^{[1]} = W^{[2]T} dZ^{[2]} * g^{[1]\prime}(Z^{[1]})$

element-wise product $(n^{[1]}, m)$

$dW^{[1]} = \frac{1}{m} dZ^{[1]} X^T$

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

$dz = a - y$
$dw_1 = x \, dz$
$db = dz$

X
W_1
b_1

$Z^{[1]} = W_1 X + b_1$ → $A^{[1]} = g(Z^{[1]})$ → $Z^{[2]} = W_2 A^{[1]} + b_2$ → $A^{[2]} = g(Z^{[2]})$ → $\mathcal{L}(A^{[2]}, y)$

Network gradients

x, $w^{[1]}$, $b^{[1]}$ → $z^{[1]} = w^{[1]}x + b^{[1]}$ → $a^{[1]} = \sigma(z^{[1]})$ → $z^{[2]} = w^{[2]}a^{[1]} + b^{[2]}$ (with $w^{[2]}$, $b^{[2]}$) → $a^{[2]} = \sigma(z^{[2]})$ → $\mathcal{L}(a^{[2]}, y)$

$dz^{[1]} = W^{[2]T} \cdot dz^{[2]}$
element wise ← $* \; g^{[1]'}(z^{[1]})$

$dz^{[2]} = a^{[2]} - y$
$dw^{[2]} = dz^{[2]} a^{[1]}$ → $dw = dz \cdot x$
$db^{[2]} = dz^{[2]}$

→ $z^{[2]}, dz^{[2]}$
$z^{[1]} \; dz^{[1]}$

$dw^{[1]} = dz^{[1]} x^T$
$db^{[1]} = dz^{[1]}$

$dz^{[1]} = W^{[2]T} dz^{[2]} * g^{[1]'}(z^{[1]})$
$(n^{[1]}, 1) \quad (n^{[1]}, n^{[2]}) \; (n^{[2]}, 1) * (n^{[1]}, 1)$

Summary of gradient descent

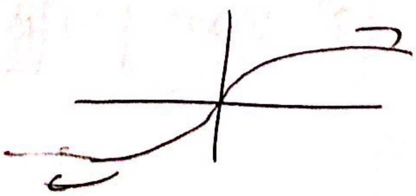
$dz^{[2]} = a^{[2]} - y$
$dw^{[2]} = dz^{[2]} a^{[1]T}$
$db^{[2]} = dz^{[2]}$
$dz^{[1]} = W^{[2]T} dz^{[2]} * g^{[1]'}(z^{[1]})$
$dw^{[1]} = dz^{[1]} x^T$
$db^{[1]} = dz^{[1]}$

$dZ^{[2]} = A^{[2]} - Y$
$dW^{[2]} = \frac{1}{m} dZ^{[2]} A^{[1]T}$
$db^{[2]} = \frac{1}{m}$ np.sum($dZ^{[2]}$, axis=1, k... = True)
$dZ^{[1]} = W^{[2]T} dZ^{[2]} * g^{[1]'}(Z^{[1]})$
$dW^{[1]} = \frac{1}{m} dZ^{[1]} X^T$
$db^{[1]} = \frac{1}{m}$ np.sum($dZ^{[1]}$, axis=1, k... = True)

Random Initialization

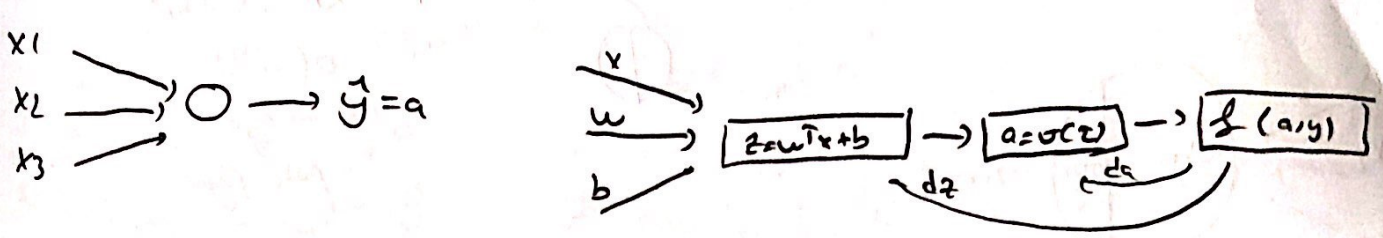
$w^{[1]}$ = np.random.randn((2,2)) * 0,001 (gaussian) → neden ten eğrisine saldığında çok büyük ve çok küçük değerler 1 veya -1 e gidiliy

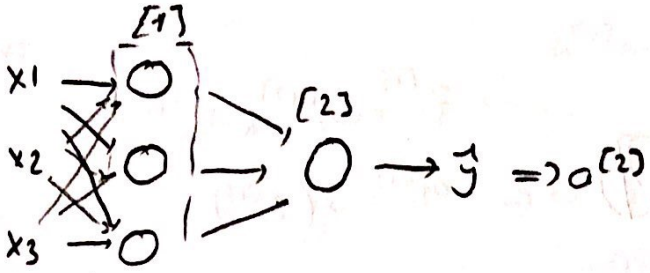
$b^{[1]}$ = np.zeros((2,1))
$w^{[2]}$ = np.random.randn((1,2)) * 0.001
$b^{[2]} = 0$

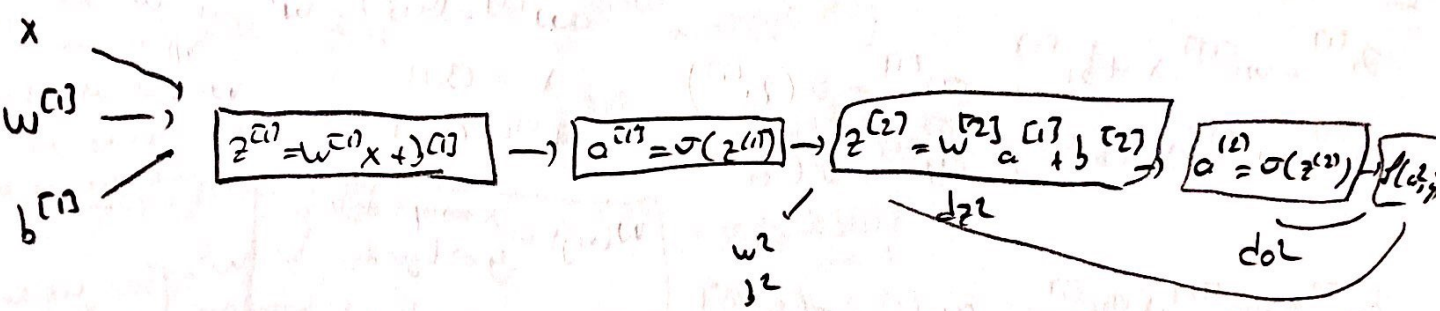


What is a Neural Network?

26 January



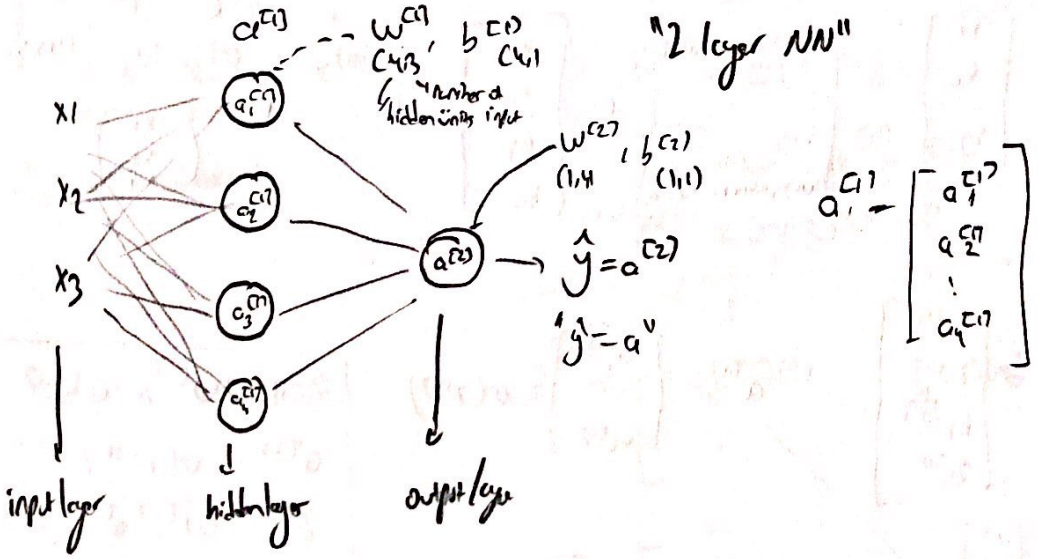




Neural Network Representation

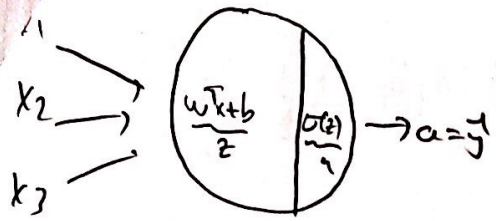
→ hidden layer

"2 layer NN"



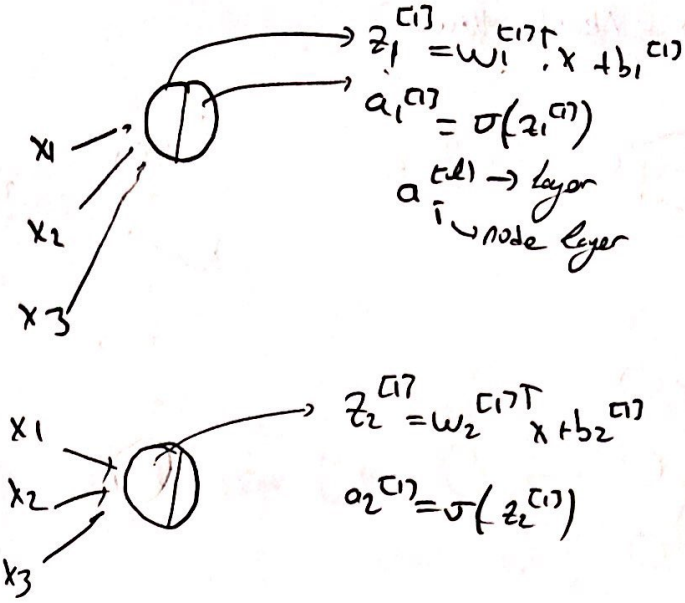
$$a^{[1]} = \begin{bmatrix} a_1^{[1]} \\ a_2^{[1]} \\ \vdots \\ a_4^{[1]} \end{bmatrix}$$

Network Representation



$z = w^T x + b$

$a = \sigma(z)$



$z_1^{[1]} = w_1^{[1]T} \cdot x + b_1^{[1]}, \quad a_1^{[1]} = \sigma(z_1^{[1]})$

$z_2^{[1]} = w_2^{[1]T} \cdot x + b_2^{[1]}, \quad a_2^{[1]} = \sigma(z_2^{[1]})$

$\vdots$

$z_4^{[1]} = w_4^{[1]T} \cdot x + b_4^{[1]}, \quad a_4^{[1]} = \sigma(z_4^{[1]})$

$w^{[1]} = (4,3) = w^T, \quad w = (3,4)$

$x = (3,1)$

$b^{[1]} = (4,1)$

$W_{(x,y)}$: $x \Rightarrow$ input layer's
$y \Rightarrow$ layer deps

$b_{(z,1)}$: $z \Rightarrow$ layer deps

$w = \begin{bmatrix} w_1 & w_2 & w_3 \\ w_1 & w_2 & w_3 \\ w_1 & w_2 & w_3 \\ w_1 & w_2 & w_3 \end{bmatrix}_{3\times 4}$

$w^T = \begin{bmatrix} w_1 & w_1 & w_1 \\ w_2 & w_2 & w_2 \\ w_3 & w_3 & w_3 \\ w_4 & w_4 & w_4 \end{bmatrix}_{(4\times 3)}$

$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$

$w^{[1]T} \cdot x = \begin{bmatrix} w_1 & w_1 & w_1 \\ w_2 & w_2 & w_2 \\ w_3 & w_3 & w_3 \\ w_4 & w_4 & w_4 \end{bmatrix}_{4\times 3} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}_{3\times 1} = \begin{bmatrix} w_1 x_1 + w_1 x_2 + w_1 x_3 \\ \vdots \\ w_4 x_1 + w_4 x_2 + w_4 x_3 \end{bmatrix}_{4\times 1} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$

Transpose of this bol

$= \begin{bmatrix} w_1^{[1]T} x + b_1^{[1]} \\ w_2^{[1]T} x + b_2^{[1]} \\ w_3^{[1]T} x + b_3^{[1]} \\ w_4^{[1]T} x + b_4^{[1]} \end{bmatrix} = \begin{bmatrix} z_1^{[1]} \\ z_2^{[1]} \\ z_3^{[1]} \\ z_4^{[1]} \end{bmatrix}$

$a^{[1]} = \begin{bmatrix} a_1^{[1]} \\ \vdots \\ a_4^{[1]} \end{bmatrix} = \sigma(z^{[1]})$

$w_{(4,1)} = \begin{bmatrix} w_1 \\ w_1 \\ w_1 \\ w_1 \end{bmatrix}_{(4,1)}$

$w^T \cdot x + b$

$w^T \cdot a + b$ $_{(4,1)\ (1,1)}$

$z^{[1]} = w^{[1]} x + b^{[1]}$

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

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

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

$\begin{bmatrix} w_1 & w_1 & w_1 & w_1 \end{bmatrix} \begin{bmatrix} a_1^{[1]} \\ a_2^{[1]} \\ a_3^{[1]} \\ a_4^{[1]} \end{bmatrix} = (a_1^{[1]} \cdot w_1 + a_2^{[1]} \cdot w_1 + a_3^{[1]} \cdot w_1 + a_4^{[1]} \cdot w_1) + b = \underline{z^{[1]}}$

$\underline{a^{[2]} = \sigma(z^{[1]})}$