¿Por qué necesito modelos no lineales?

x_1	x_2	$y = X_1 \text{ xor } X_2$
0	0	0
0	1	1
1	0	1
1	1	0

$n = 4$

$m = 2$

labels

Problema de regresión

Matriz $X = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ ← b

$x_1 \rightarrow$ M, $x_2 \rightarrow$ M, M $\rightarrow \hat{y} \rightarrow$ pred

* Modelo: $\hat{y}_i = x_{i,1} w_1 + x_{i,2} w_2 + b$ (modelo lineal) $\rightarrow$ 3 parámetros: w_1, w_2, b
* Optimizador: $\bar{\nabla}_{\bar{w}} (J) = \bar{0} \rightarrow \begin{bmatrix} \partial J / \partial w_1 = 0 \\ \partial J / \partial w_2 = 0 \\ \partial J / \partial b = 0 \end{bmatrix}$
* Costo: $J_{(\bar{w})} = \frac{1}{4} \sum_{i=1}^{4} (y - \hat{y})^2$

MSE

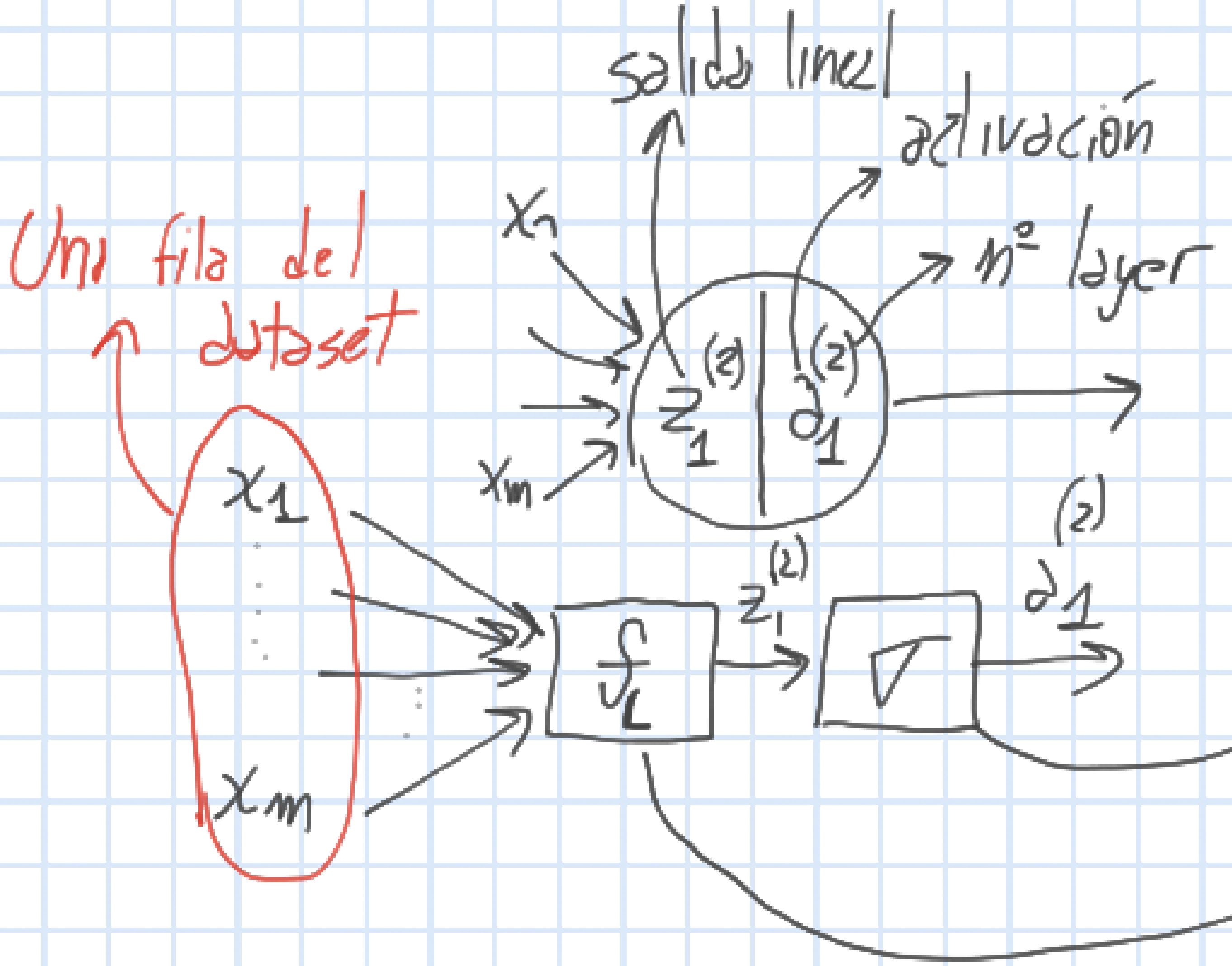
$\bar{w} = \begin{bmatrix} w_1 \\ w_2 \\ b \end{bmatrix}$

$$\bar{w} = \begin{bmatrix} w_1 \\ w_2 \\ b \end{bmatrix} = (X^T X)^{-1} X^T Y$$

$$\bar{w} = \begin{bmatrix} 0 \\ 0 \\ 0{,}5 \end{bmatrix}$$

Modelos no lineales basados en redes neuronales

Definición de neurona



Es una función no lineal

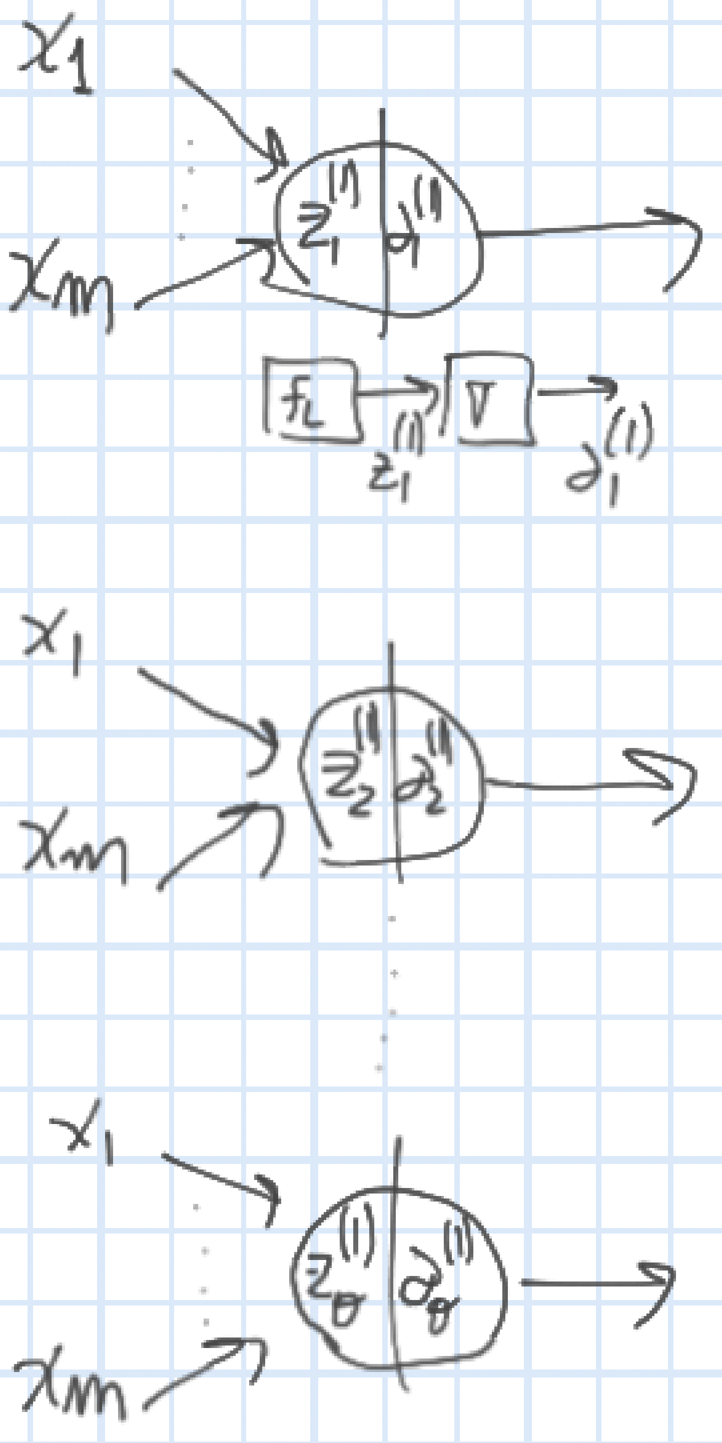
Sigmoid, ReLU, ...

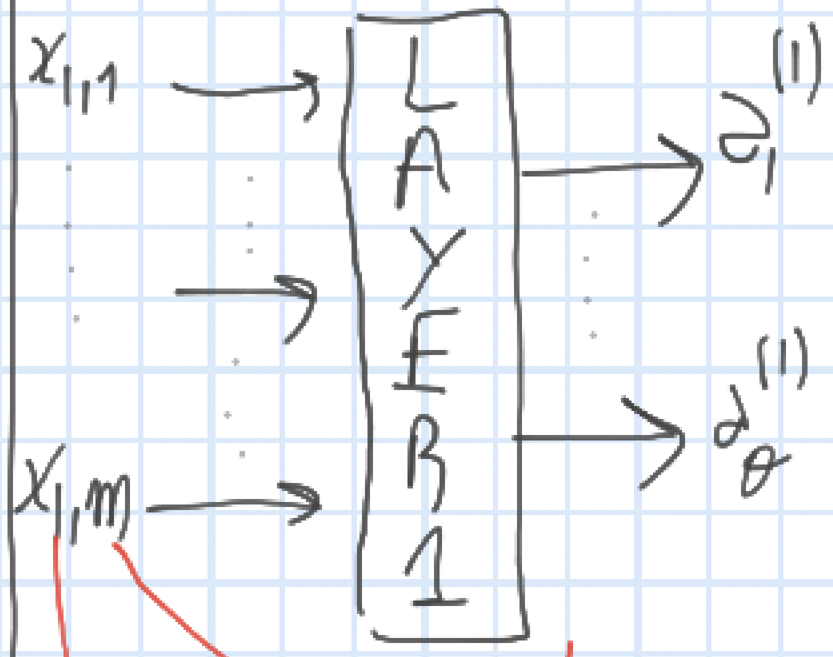
$$a_1^{(2)} = \sigma\left(z_1^{(2)}\right) \qquad \mathbb{R}^1 \to \mathbb{R}^1$$

$$z_1^{(2)} = w_1^{(2)} x_1 + \dots + w_m^{(2)} + b^{(2)}$$

$$= \overline{w}^{(2)T} \overline{x} + b \longrightarrow \mathbb{R}^m \to \mathbb{R}^1$$

Definición de Layer





m es la cant de features

el número de fila de mi dataset

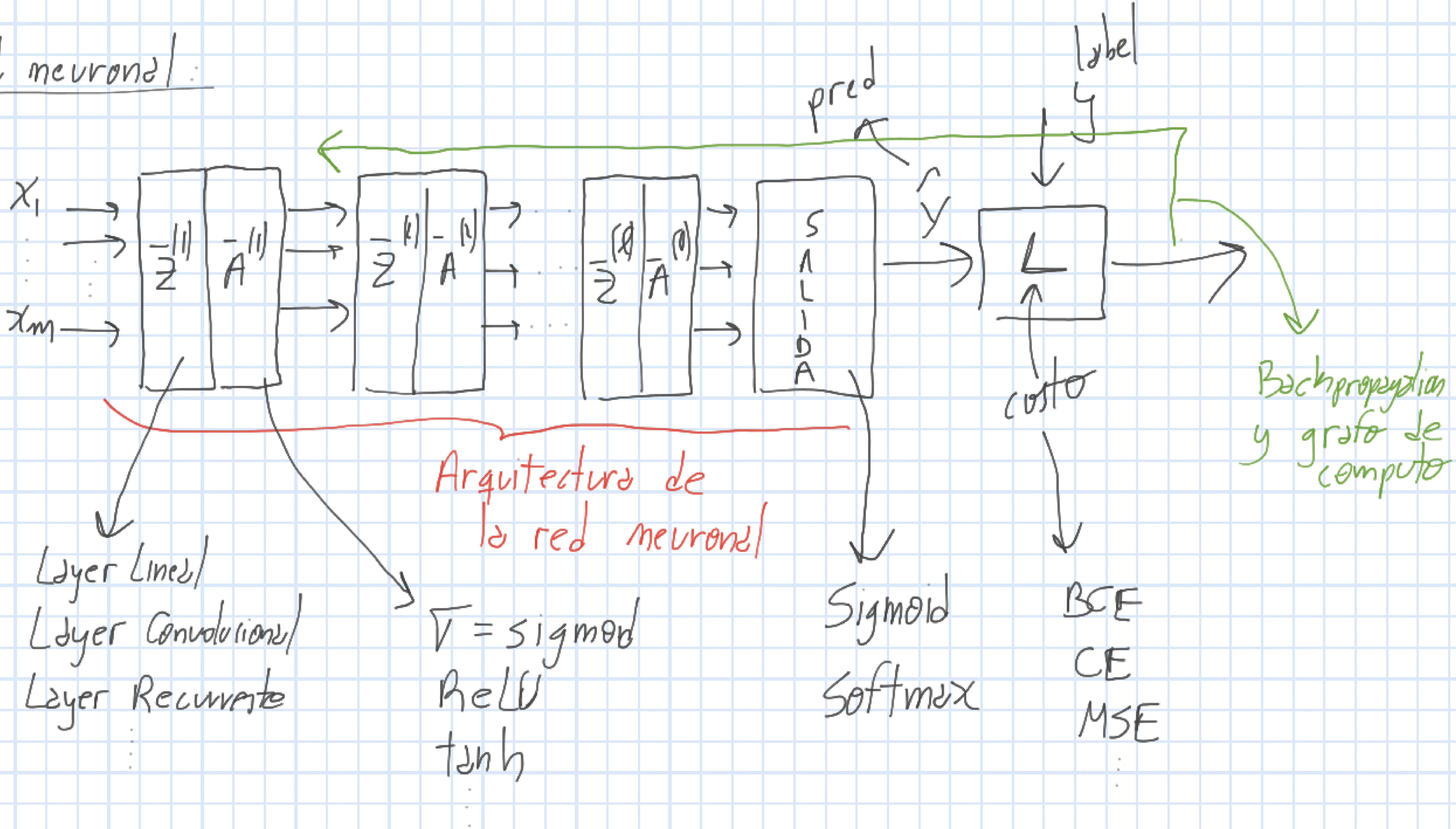
$$a_1^{(1)} = \sigma\left(z_1^{(1)}\right) = \sigma\left(x_{1,1} w_{1,1}^{(1)} + \dots + x_{1,m} w_{1,m}^{(1)} + b_1^{(1)}\right)$$

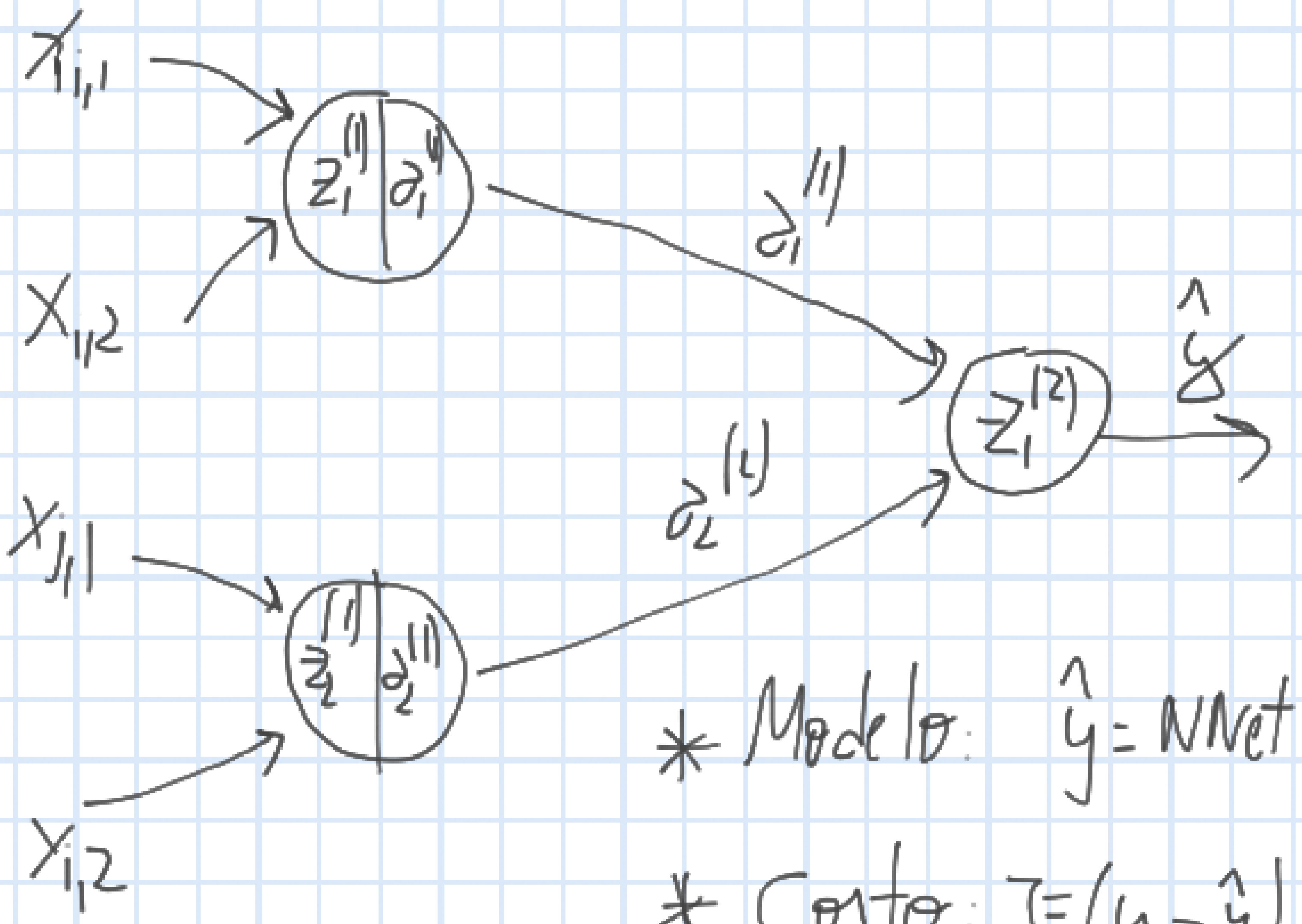
$$\underbrace{a_\theta^{(1)} = \sigma\left(z_\theta^{(1)}\right) = \sigma\left(x_{1,1} w_{\theta,1}^{(1)} + x_{1,2} w_{\theta,2}^{(2)} + \dots + x_{1,m} w_{\theta,m}^{(2)} + b_\theta\right)}_{\text{En formato vectorial}}$$

$$\bar{A}^{(1)} = \sigma\left(\bar{Z}^{(1)}\right)$$

$$\underbrace{\bar{Z}}_{\in\mathbb{R}^{\theta\times 1}} = \underbrace{W^{(1)}}_{\in\mathbb{R}^{\theta\times m}} \underbrace{\bar{X}_1}_{\in\mathbb{R}^{m\times 1}} + \underbrace{b^{(1)}}_{\in\mathbb{R}^{\theta\times 1}}$$

Red neuronal :





* Modelo: $\hat{y} = NNet(x_1, x_2)$

* Costo: $J = (y - \hat{y})^2$

* Optimizador: SGD

x_1	x_2	
0	0	
0	1	1
1	0	
1	1	

XOR con NNet

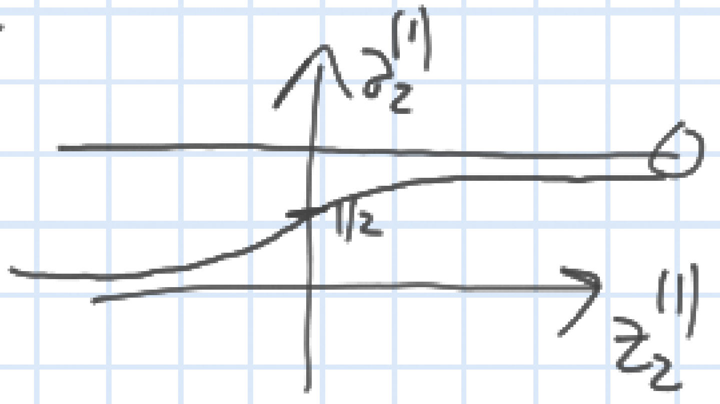
* Forward (1)

$$z_{i,1}^{(1)} = x_{i,1} w_{1,1}^{(1)} + x_{i,2} w_{1,2}^{(1)} + b_1^{(1)}$$

$$a_{i,1}^{(1)} = \sigma(z_1^{(1)}) = \frac{1}{1+e^{-z_1^{(1)}}}$$

$$z_{i,2}^{(1)} = x_{i,1} w_{2,1}^{(1)} + x_{i,2} w_{2,2}^{(1)} + b_2^{(1)}$$

$$a_{i,2}^{(1)} = \sigma(z_2^{(1)}) = \frac{1}{1+e^{-z_2^{(1)}}}$$

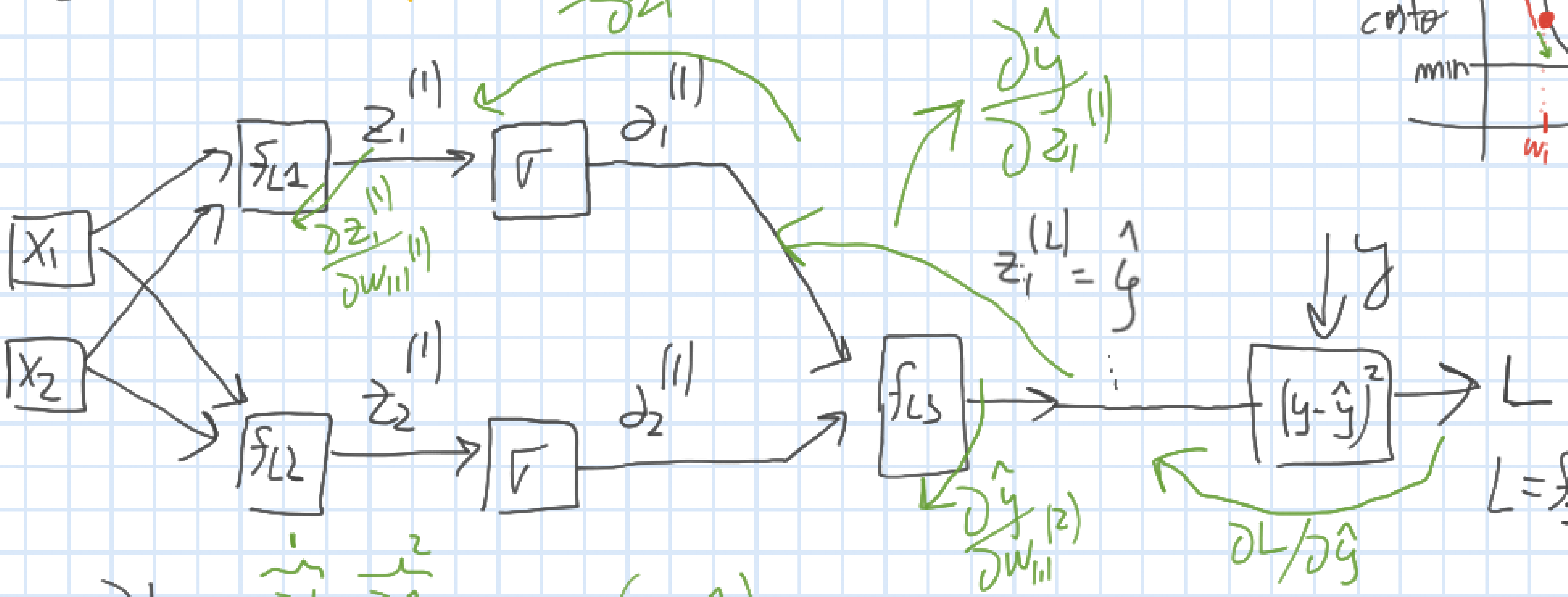


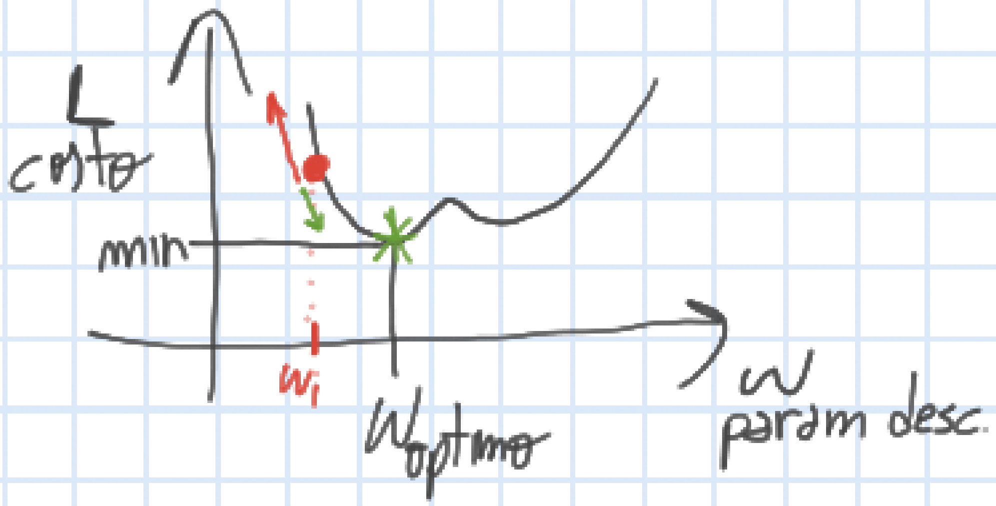
$$\hat{y}_i = z_1^{(2)} = a_{i,1}^{(1)} w_{1,1}^{(2)} + a_{i,2}^{(1)} w_{1,2}^{(2)} + b_1^{(2)}$$

¿Param desconocidos?

$3+3+3 = \underline{9}$

* Backward: (L)





$$L = f_1(y, \hat{y}) = f_1\left(y, f_2\left(w_{1,1}^{(2)}, w_{1,2}^{(2)}, b_2, a_1^{(1)}, a_2^{(1)}\right)\right)$$

$$\frac{\partial L}{\partial w_{1,1}^{(2)}} = \underbrace{\frac{\partial L}{\partial \hat{y}}}_{1} \underbrace{\frac{\partial \hat{y}}{\partial w_{1,1}^{(2)}}}_{2} = -2(y_1 - \hat{y}_1) \; a_{1,1}$$

$$\frac{\partial L}{\partial w_{12}^{(2)}} = -2(y_1 - \hat{y}_1) \; a_{1,2}$$

$$\frac{\partial L}{\partial b_1^{(2)}} = -2(y_1 - \hat{y}_1) = -2 \, err_1$$

$$\frac{\partial L}{\partial w_{1,1}^{(1)}} = \underbrace{\frac{\partial L}{\partial \hat{y}}}_{= -2 err_1} \underbrace{\frac{\partial \hat{y}}{\partial a_{1,1}^{(1)}}}_{w_{1,1}^{(2)}} \underbrace{\frac{\partial a_{1,1}^{(1)}}{\partial z_{1,1}^{(1)}}}_{\sigma(z_{1,1}^{(1)})(1-\sigma(z_{1,1}^{(1)}))} \underbrace{\frac{\partial z_{1,1}^{(1)}}{\partial w_{1,1}^{(1)}}}_{x_{1,1}}$$

Tarea: $\frac{\partial L}{\partial w_{1,2}^{(1)}}, \frac{\partial L}{\partial w_{2,1}^{(1)}}, \frac{\partial L}{\partial w_{2,2}^{(1)}}, \frac{\partial L}{\partial b_1^{(1)}}, \frac{\partial L}{\partial b_2^{(1)}}$

* Actualizacion de los pesos (B)

SGD

$$\hookrightarrow \bar{w} \leftarrow \bar{w} - \alpha \bar{\nabla}(L_i)$$

$$\begin{pmatrix} w_{1,1}^{(1)} \\ w_{12}^{(1)} \\ w_{2,1}^{(1)} \\ w_{2,2}^{(1)} \\ b_1^{(1)} \\ b_2^{(1)} \\ w_{12}^{(2)} \\ w_{1,1}^{(2)} \\ b_1^{(2)} \end{pmatrix} \leftarrow \left(\bar{w} \right) - \begin{pmatrix} \alpha\, \partial L_i / \partial w_{1,1}^{(1)} \\ \vdots \\ \alpha\, \partial L_i / \partial b_1^{(2)} \end{pmatrix}$$

Algoritmo de optimización

SGD (TAREA)

(0) Inicializar random los pesos → Unif $(0,1)$
→ Normal $(0,1)$

for epoch in range(100):

HP: nº epochs
1, 10, 100, 1000

for i in range(4):

$x_1 = X[i,1]$, $x_2 = X[i,2]$, $y = Y[i]$

(1) Forward → $a_1^{(1)}, a_2^{(1)}, z_1^{(1)}, z_2^{(2)}, \dots \to \hat{y}$

err $= y - \hat{y}$

(2) Backward (9 eq) → $\nabla(J)$

HP = 0,1 ··· 0,01 ··· 0,001

(3) Actualizar pesos → $\bar{w} = \bar{w} - \alpha \nabla(J)$

(4) Calcular MSE $= \frac{1}{4} \sum (y - \hat{y})^2$

	GD	Mini-Batch	SGD
∇?	$D\left(\frac{1}{m}\sum_{i=1}^{m} L(y_i, \hat{y}_i)\right)$	$D\left(\frac{1}{b}\sum_{i=1}^{b} L(y_i, \hat{y}_i)\right)$	$D\left(L(y_i, \hat{y}_i)\right)$
Velocidad	+++	++	+
RAM GPU	+++	++	+

	pocas fet	muchas fet
pocas filas	estadisticas	estadis generali ~~regulariz~~
muchas filas	ML tradicional	mnet