Funciones de activación

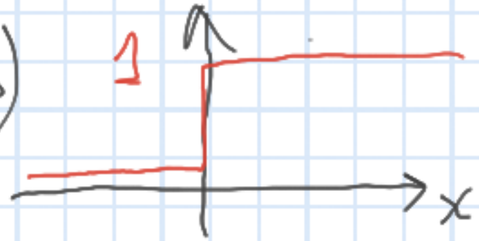
(1) No usar func. act ?

¿Qué problema tiene?

Esto es un modelo lineal

(2) Agregar func. act / $\hat{y} = f(\bar{x})$ puede ser cualquier f aprendida por la red

(3)



$$f(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases} \longrightarrow f'(x) = \begin{cases} 0 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

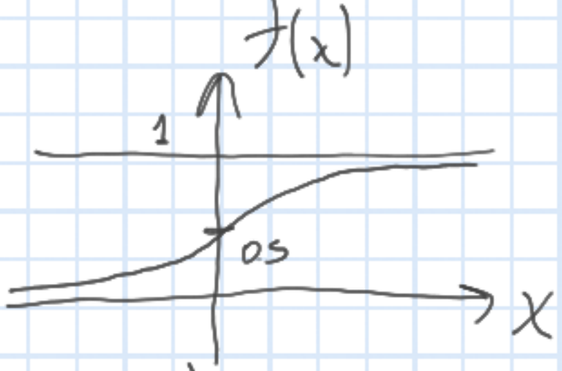
¿Qué problema tiene esta $f(x)$?

$$w_i \leftarrow w_i - \alpha \frac{\partial L}{\partial w_i}$$

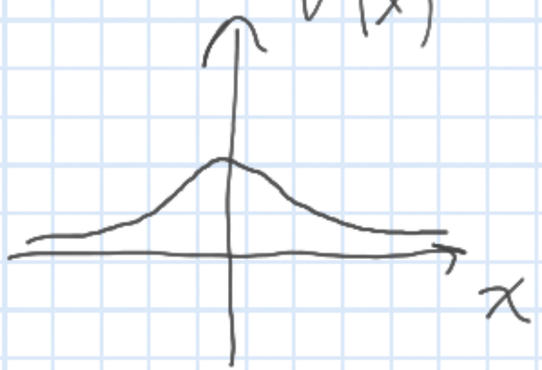
vale cero !!

(3) Sigmoid

$\sigma(x) = f(x) = \frac{1}{1+e^{-x}}$



$\sigma'(x) = \sigma(x)\left(1-\sigma(x)\right)$

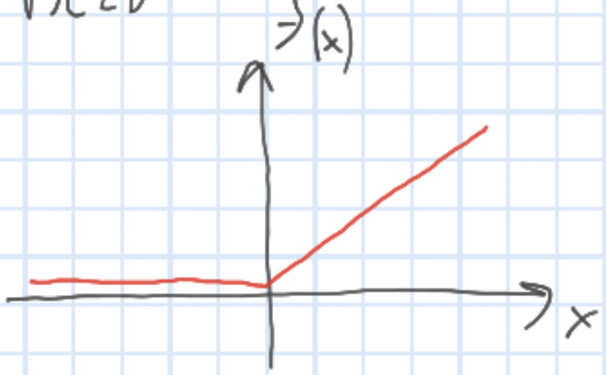


$\lim_{x \to +\infty} \sigma'(x) = 0$

$\lim_{x \to -\infty} \sigma'(x) = 0$

Problems: Vanishing gradients

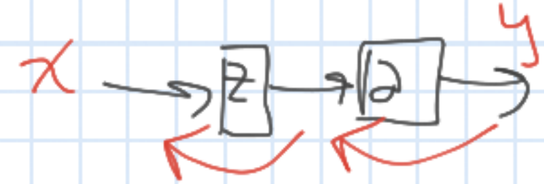
(4) ReLU



$$f(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases} = \max(x, 0)$$

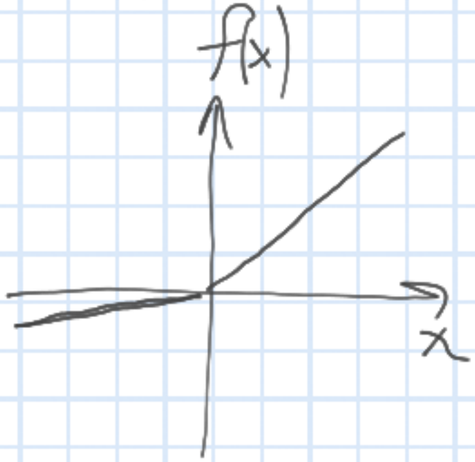
$$f'(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases}$$

$z = w_1 x_1 + \dots + w_m x_m + b$



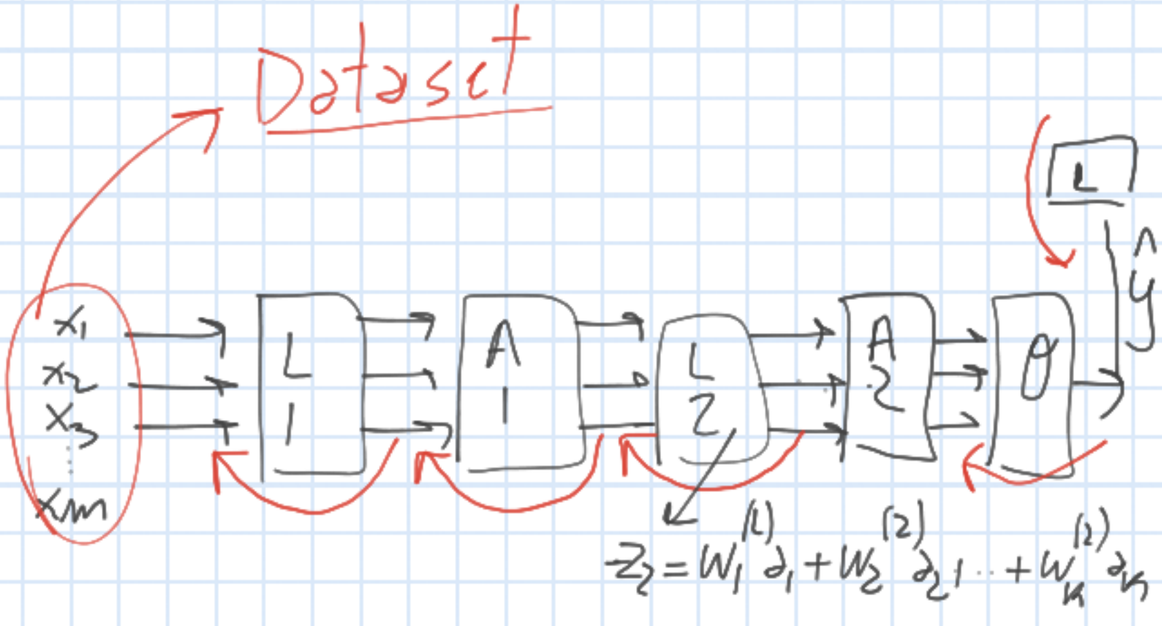
$$\frac{dy}{dx} = \frac{dy}{da}\frac{da}{dz}\frac{dz}{dx}$$

(5) Leaky ReLU



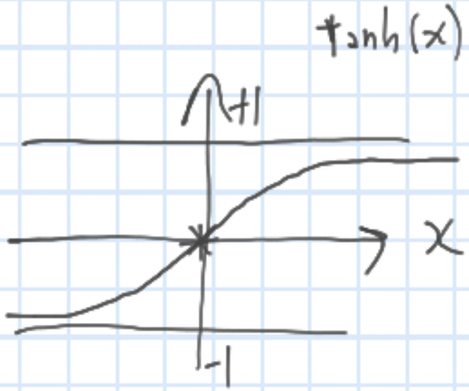
$$f(x) = \begin{cases} x & x > 0 \\ \underbrace{0.01}_{\alpha} x & x \le 0 \end{cases}$$

$$f'(x) = \begin{cases} 1 & x > 0 \\ 0.01 & x \le 0 \end{cases}$$

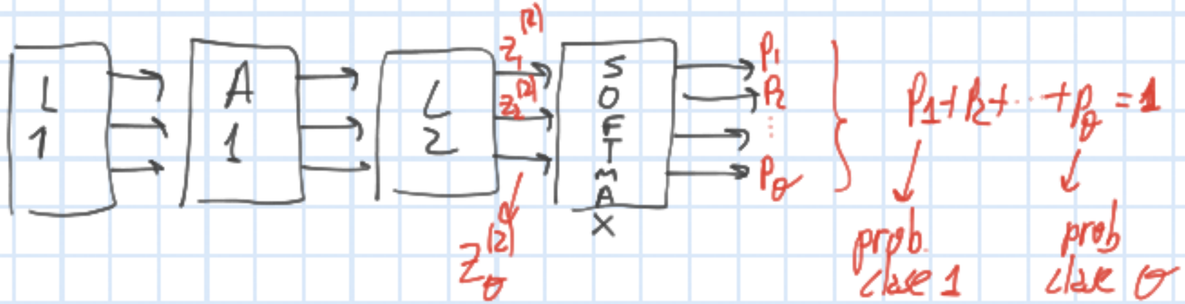


$$\frac{\partial L}{\partial w_1^{(1)}} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial a_2} \frac{\partial a_2}{\partial z_2} \frac{\partial z_2}{\partial w_1^{(1)}}$$

(3 con mejoras) $\tanh(x) = 2\sigma(x) - 1$



Funciones de salida → Regresión $\emptyset$
Clasif Bin σ
Classif multi clase → SOFTMAX



$$p_1 = \frac{e^{z_1}}{\sum_{i=1}^{\theta} e^{z_i}}$$

$$p_\theta = \frac{e^{z_\theta}}{\sum_{i=1}^{\theta} e^{z_i}}$$

Loss Functions

(1) Binary Cross Entropy

n rows, m features: $x_{1,1} \dots x_{1,m} \mid y_1$; $x_{n,1} \dots x_{n,m} \mid y_2$

m

→ SGD

→ GD

→ Mini-Batch → $b << n$

	GD	Mini	SGD
Memoria	+++	++	+
Velocidad	+	++	++

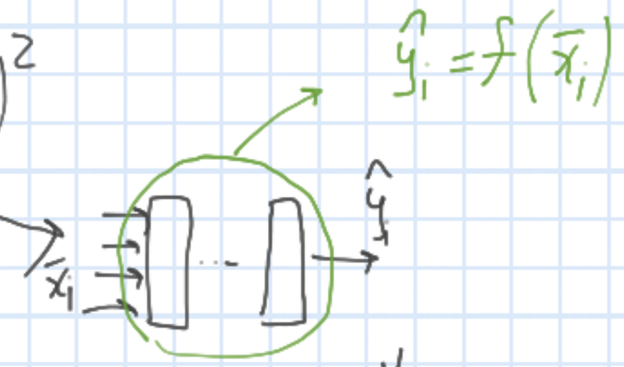
→ n_epochs * n

$$L = \frac{1}{b} \sum_{i=1}^{b} (y_i - \hat{y}_i)^2$$

* $Loss(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2 = (y_i - f(\bar{x}_i))^2$ ← ejemplo

Representa la red neuronal

$\hat{y}_i = f(\bar{x}_i)$



↳ Computo gradiente

↳ Actualizo pesos

* $Loss(y_1, y_2, \dots, y_2, \hat{y}_1 \dots \hat{y}_n) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \rightarrow \begin{bmatrix} (y_1 - \hat{y}_1)^2 \\ \vdots \\ (y_n - \hat{y}_n)^2 \end{bmatrix}$

↳ Computo gradi

↳ Actualizo pesos

$$\nabla(L) = \nabla_w \left(\frac{1}{n} \sum_{i=1}^{n} (y - \hat{y})^2 \right)$$

$$= \frac{1}{n} \sum_{i=1}^{n} \nabla_w \left[(y - \hat{y})^2 \right]$$

(1) BCE

Modelar la salida en función de la entrada

$$\mathbb{P}(\hat{Y}_i = 1 \mid \bar{X}_i = \bar{x}_i) = \rho \quad , \quad \rho = \nabla(f(\bar{x}))$$

$\nabla(f(\bar{x}))$ → red neuronal

$f(\bar{x})$ → parámetros desconocidos

$$Y_i \mid \bar{X}_i = \bar{x}_i \sim \text{Bernoulli}(\rho) \quad \begin{cases} 1 \\ 0 \end{cases}$$

$$\mathbb{P}\left(Y_1 \mid \bar{X}_1 = \bar{x}_1 \cap Y_2 \mid \bar{X}_2 = \bar{x}_2 \cap \cdots \cap Y_n \mid \bar{X}_n = \bar{x}_n\right) = \prod_{i=1}^{n} \mathbb{P}(Y_i \mid X_i \mid \bar{x}_i)$$

encontrar los $\bar{w}$ que maximizan lo observado

$$\arg\min_{\bar{w}} - \prod_{i=1}^{n} \mathbb{P}(Y_i \mid X_i = x_i) \qquad Y_i = y_i \mid \bar{X}_i = \bar{x}_i$$

$$\arg\min_{\bar{w}} - \frac{1}{n} \log\left(\prod_{i=1}^{n} \mathbb{P}(Y_i \mid X_i = x_i)\right)$$

$$\arg\min_{\bar{w}} - \frac{1}{n} \sum_{i=1}^{n} \log\left(\mathbb{P}(Y_i = y_i \mid X_i = x_i)\right)$$

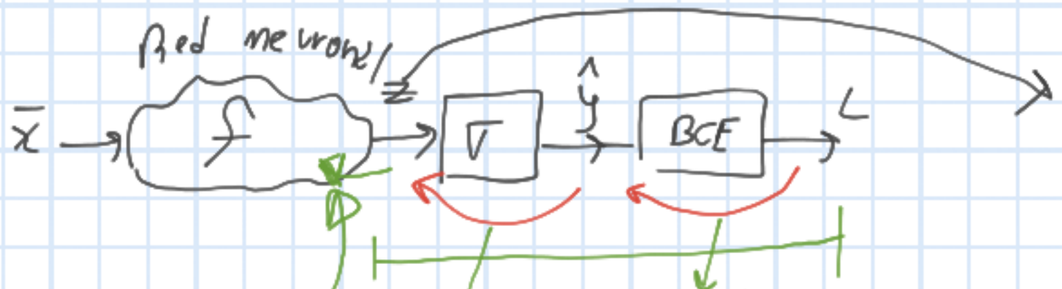
$$\arg\min_{\bar{w}} - \frac{1}{n} \sum_{i=1}^{n} \log\left(\rho_i^{\,y_i} (1-\rho_i)^{1-y_i}\right)$$

$$\arg\min_{\bar{w}} - \frac{1}{n} \sum_{i=1}^{n} \left[\log\left(\rho_i^{\,y_i}\right) + \log\left((1-\rho_i)^{1-y_i}\right)\right]$$

$$\arg\min_{\bar{w}} - \frac{1}{n} \sum_{i=1}^{n} \left[y_i \log(\rho_i) + (1-y_i)\log(1-\rho_i)\right]$$

BCE

$$\arg\min_{\bar{w}} - \frac{1}{n} \sum_{i=1}^{n} \left[y_i \log\left(\hat{y}_i = \nabla(f(\bar{x}_i)) + \ast\ast\ast\right)\right]$$



$$z = f(\bar{x}_i)$$

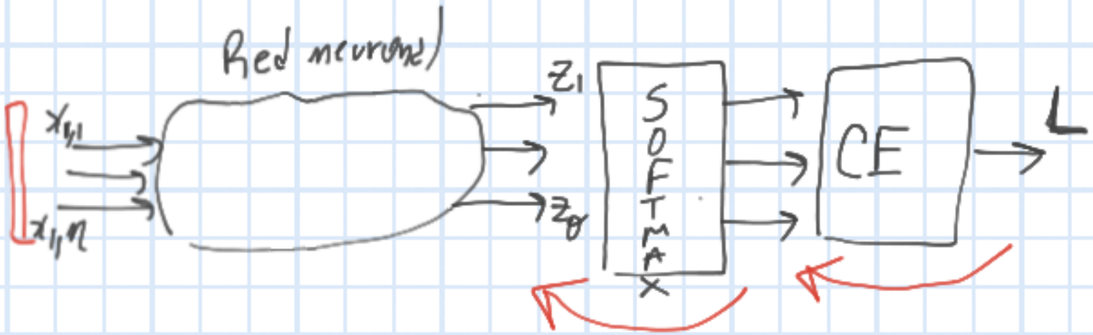
$$\frac{d\hat{y}}{dz} \quad (2) \qquad * \qquad \frac{dL}{d\hat{y}} \quad (1)$$

$$\frac{d\hat{y}}{dz} = \sigma(z)(1-\sigma(z))$$

$$\frac{dL}{d\hat{y}} = -\left[y \log(\hat{y}) + (1-y)\log(1-\hat{y})\right]$$

$$= -\frac{y}{\hat{y}} + \frac{(1-y)}{(1-\hat{y})}$$

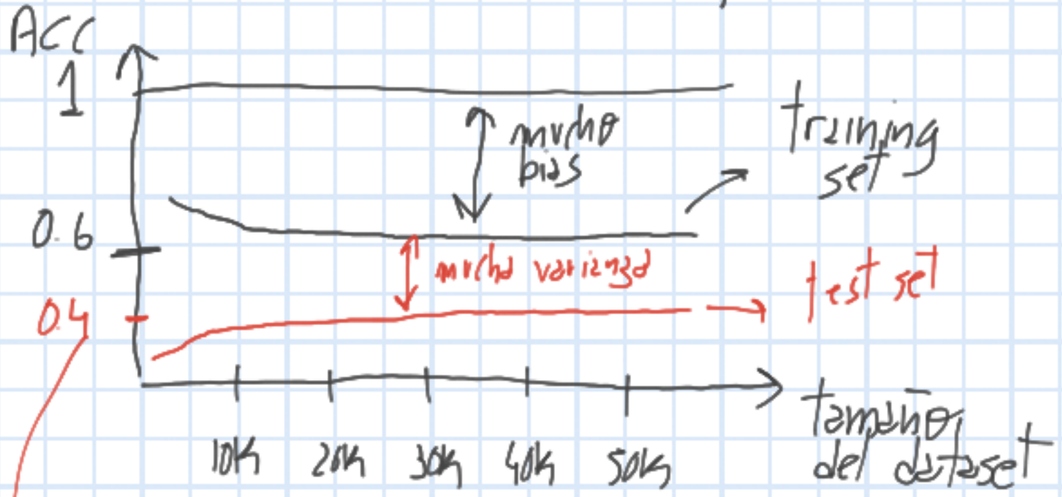
(2) Cross Entropy → Class multiclass



TODO

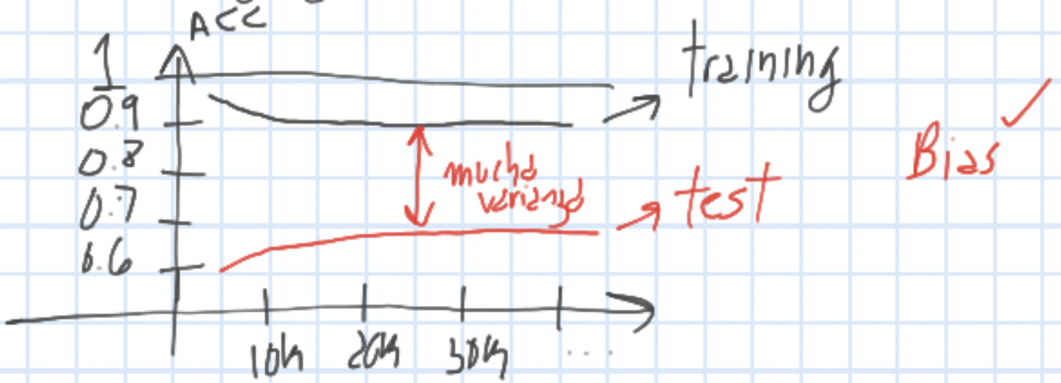
Regularización

Bias vs Varianza
Underfit vs Overfit



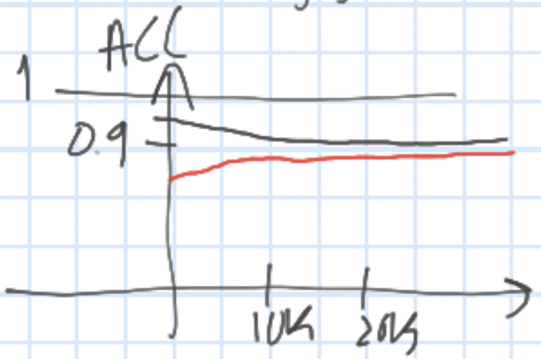
Learning curve

¿Cómo disminuir el error de bias?
- (1) Agregar complejidad al model
- (2) Agregar más features (+columnas)



Bias ✓

¿Como bajo el error por varianza?
- (1) Regularización
- (2) Agrego más filas (data)



* Regularización → L2
→ L1
→ Dropout

L2

$J(\bar{w}, y, \bar{x})$ → es un escalar → sin regularizar

$\tilde{J}(\bar{w}, y, \bar{x}, \lambda) = J(\bar{w}, y, \bar{x}) + \frac{\lambda}{2} \bar{w}^T \bar{w}$

todos los pesos de mi red neuronal

$\bar{w}^T \bar{w} = [w_1 \cdots w_k] \begin{bmatrix} w_1 \\ w_k \end{bmatrix} = w_1^2 + w_2^2 + \cdots + w_k^2 = \|\bar{w}\|_2^2$

(1) Efecto sobre update a los pesos

$\nabla_{\bar{w}}(\tilde{J}) = \nabla_{\bar{w}}\left(J + \frac{\lambda}{2}\bar{w}^T\bar{w}\right) = \nabla_{\bar{w}}(J) + \nabla_{\bar{w}}\left(\frac{\lambda}{2}\bar{w}^T\bar{w}\right) = \nabla_{\bar{w}}(J) + \lambda\bar{w}$

Actualización del param desconocido

$$\bar{w} \leftarrow \bar{w} - \alpha D(\tilde{J})$$

$$\leftarrow \bar{w} - \alpha \left(D(J) + \lambda \bar{w} \right)$$

$$\leftarrow \bar{w} - \alpha\lambda\bar{w} - D(J)\alpha$$

$$\leftarrow (1 - \lambda\alpha)\bar{w} - \alpha D(J)$$

(2) Efecto sobre el modelo

$$\hat{Y} = \bar{w}^T \bar{x} \rightsquigarrow \bar{w} = (X^T X)^{-1} X^T y$$

MSE

$$\text{MSE+L2} \longrightarrow \bar{w} = \left(X^T X + \lambda \bar{I} \right) X^T y$$

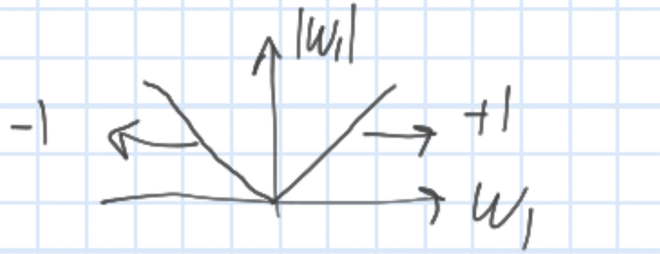
$X^T X \in \mathbb{R}^{m \times n}$ $\quad \lambda \bar{I} \in \mathbb{R}^{n \times m}$

$$\in \mathbb{R}^{m \times m}$$

L1

$$\tilde{J} = J + \lambda \|\bar{w}\|_1 \qquad \|\bar{w}\|_1 = |w_1| + |w_2| + \dots + |w_5|$$

$$\nabla(\tilde{J}) = \nabla(J) + \lambda \, \text{signo}(\bar{w})$$



$$\bar{w} \leftarrow \bar{w} - \alpha \nabla(\tilde{J})$$

$$\bar{w} - \alpha\left(\nabla[J] + \lambda \, \text{signo}(\bar{w})\right)$$

$$\bar{w} - \alpha\lambda \, \text{signo}(\bar{w}) - \alpha \nabla[J]$$

→ bajar complejidad modelo