🧠 REDES NEURONALES

🧪 Facultad de Ciencias

🏫 Universidad Nacional Autónoma de México

📄 Tarea No. 02

Realizado por:

Iván Alejandro Ramos Herrera

💜 @arhcoder

[01]

🧮 Demostrar que bajo un conjunto de pesos $\theta^* = \{w^*, b^*\}$ el perceptrón y la regresión logística son clasificadores binarios equivalentes:

Definiciones

Definición del Perceptrón:

$$f_p(x) = sign(w^*x + b^*)$$

Es decir...

$$f_p(x) = \begin{cases} 1 & \text{si } w^*x + b^* > 0 \\ 0 & \text{si } w^*x + b^* \leq 0 \end{cases}$$

Donde:

- x vector de entradas en $\mathbb{R}^d$
- w^* es un vector de pesos en $\mathbb{R}^d$
- b^* es el valor de sesgo.

Definición de la Regresión Logística:

$$f_r(x) = \sigma(w^*x + b^*)$$

Es decir..

$$f_r(x) = P(y = 1|x)$$

$$f_r(x) = \frac{1}{1+e^{-(w^*x+b^*)}}$$

En donde para una clasificación binaria **(0, 1)**:

$$f_r(x) = \begin{cases} 1 & \text{si } P(y=1|x) \geq 0.5 \\ 0 & \text{si } P(y=1|x) < 0.5 \end{cases}$$

Donde:

- x vector de entradas en $\mathbb{R}^d$
- w^* es un vector de pesos en $\mathbb{R}^d$
- b^* es el valor de sesgo.

Demostración

¿Bajo un mismo conjunto de pesos θ, ambos clasificadores tomarán las mismas decisiones?

$$w^*x + b^* = P(y=1|x) - 0.5$$

Simplificando las definiciones:

$$f_p(x) = \begin{cases} 1 & \text{si } w^*x + b^* > 0 \\ 0 & \text{en cualquier otro caso} \end{cases}$$

$$f_r(x) = \begin{cases} 1 & \text{si } P(y=1|x) \geq 0.5 \\ 0 & \text{en cualquier otro caso} \end{cases}$$

Si se demuestra que se clasifica como 1 en los dos primeros casos de cada función, se deduce que también se clasificará 0 (en cualquier otro caso), por lo que sólo basta con demostrar la equivalencia con las clasificaciones de 1.

Demostrando equivalencia con los casos:

- $w^*x + b^* > 0$
- $P(y=1|x) \geq 0.5$

$$P(y=1|x) \geq 0.5 = w^*x + b^* > 0$$

$$\therefore$$

$$\frac{1}{1+e^{-(w^*x+b^*)}} \geq 0.5$$

Resolviendo la desigualdad:

Pasar multiplicando el denominador:

$$1 \geq 0.5(1 + e^{-(w^*x+b^*)})$$

$$1 \geq 0.5 + 0.5e^{-(w^*x+b^*)}$$

Pasando el 0.5 al lado izquierdo:

$1 - 0.5 \geq 0.5e^{-(w^*x+b^*)}$

$0.5 \geq 0.5e^{-(w^*x+b^*)}$

Eliminando el 0.5:

$1 \geq e^{-(w^*x+b^*)}$

Eliminando e con ln:

$ln(1) \geq -(w^*x + b^*)$

$0 \geq -(w^*x + b^*)$

Eliminando el signo invirtiendo la desigualdad:

$0 \leq w^*x + b^*$

Esto es igual al caso en el perceptrón:

$w^*x + b^* > 0$

$\therefore$

$f_p = f_r$ con $\theta^* = \{w^*, b^*\}$ bajo las definiciones descritas.

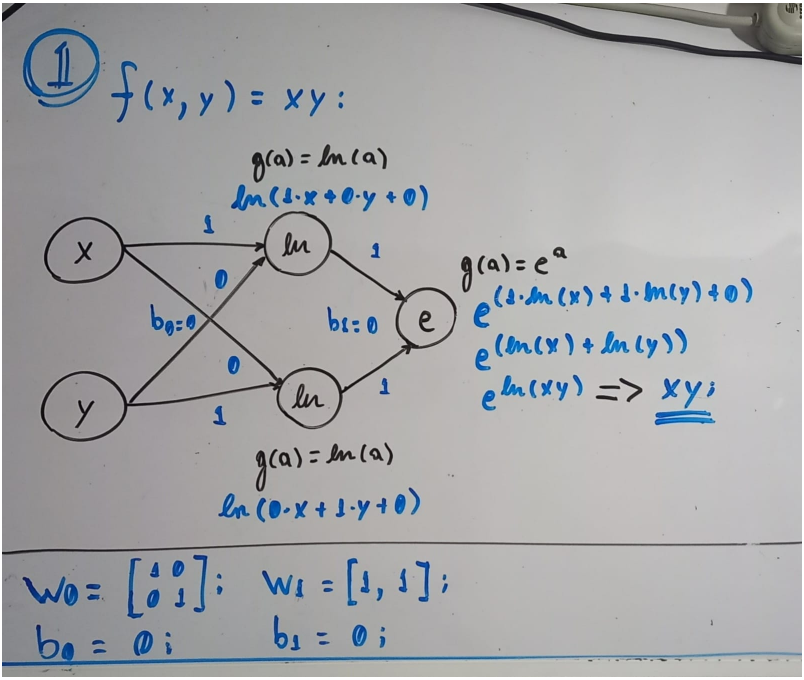
El Modelo de Perceptrón con Función Escalonada y la Regresión Logística son equivalentes bajo la definición de clasificiación binaria.

[02]

🧠 Definir una Red Neuronal que obtenga cada una de las siguientes funciones de $\mathbb{R}^2$ a $\mathbb{R}$:

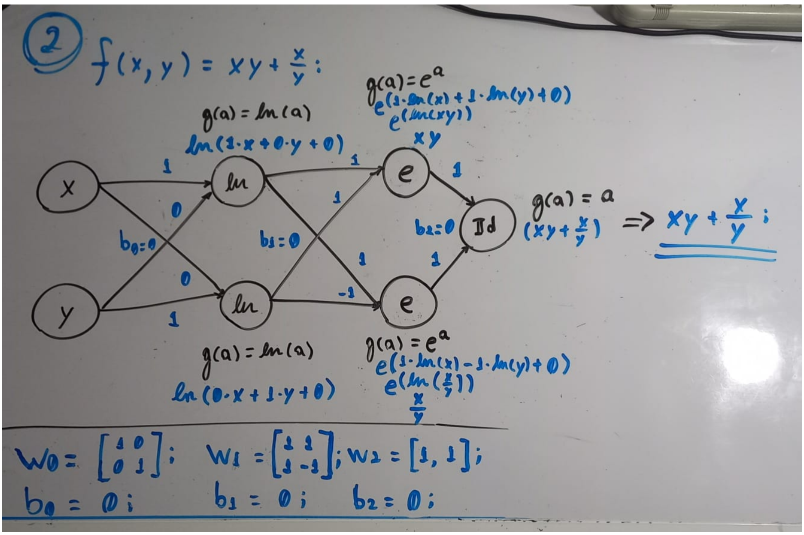
1. $f(x, y) = xy$
2. $f(x, y) = xy + \frac{x}{y}$
3. $f(x, y) = 2x^2 + 2xy$
4. $f(x, y) = x^2 + 2xy + y^2$

xy:



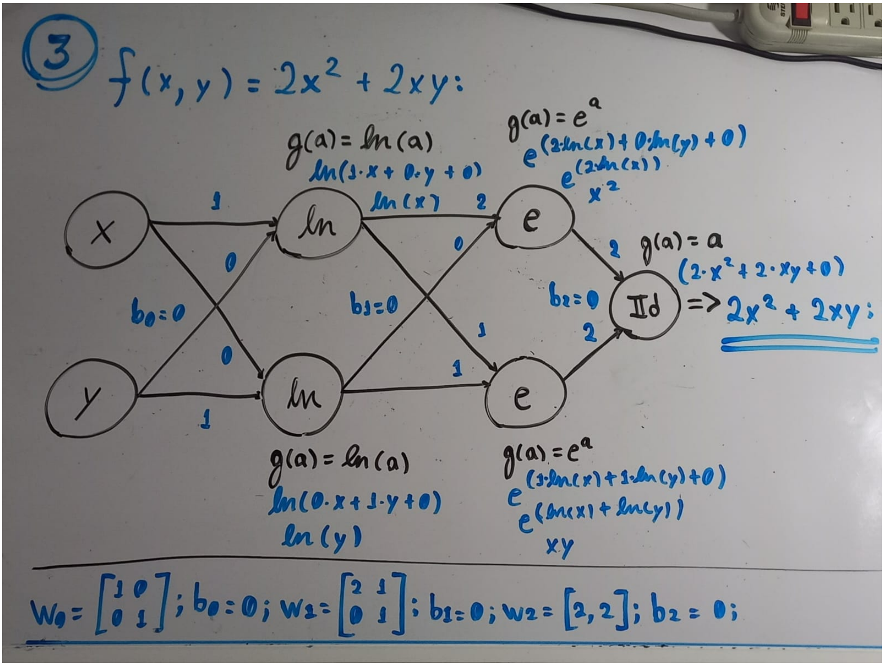
① f(x, y) = xy :
g(a) = ln(a)
ln(1·x + 0·y + 0)
g(a) = e^a
e^(1·ln(x) + 1·ln(y) + 0)
e^(ln(x) + ln(y))
e^ln(xy) => xy;
b0=0
b1=0
g(a) = ln(a)
ln(0·x + 1·y + 0)
W0 = [1 0; 0 1]; W1 = [1, 1];
b0 = 0; b1 = 0;

$xy + \frac{x}{y}$:

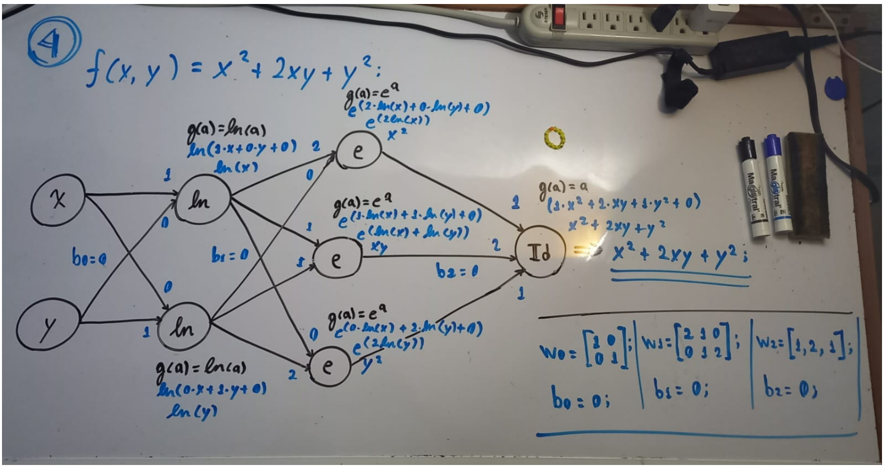


② f(x, y) = xy + x/y :
g(a) = ln(a)
ln(1·x + 0·y + 0)
g(a) = e^a
e^(1·ln(x) + 1·ln(y) + 0)
e^(ln(xy))
xy
g(a) = a
(xy + x/y) => xy + x/y ;
b0=0
b1=0
b2=0
Id
g(a) = ln(a)
ln(0·x + 1·y + 0)
g(a) = e^a
e^(1·ln(x) − 1·ln(y) + 0)
e^(ln(x/y))
x/y
W0 = [1 0; 0 1]; W1 = [1 1; 1 −1]; W2 = [1, 1];
b0 = 0; b1 = 0; b2 = 0;

- $2x^2 + 2xy$:



- $x^2 + 2xy + y^2$:

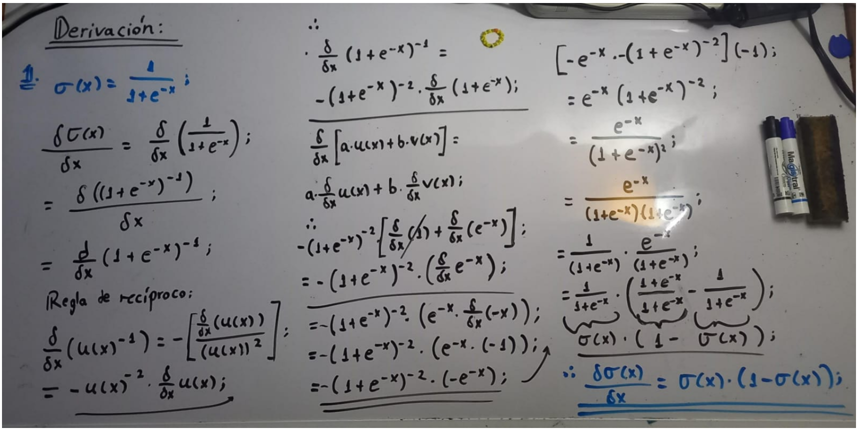


[03]

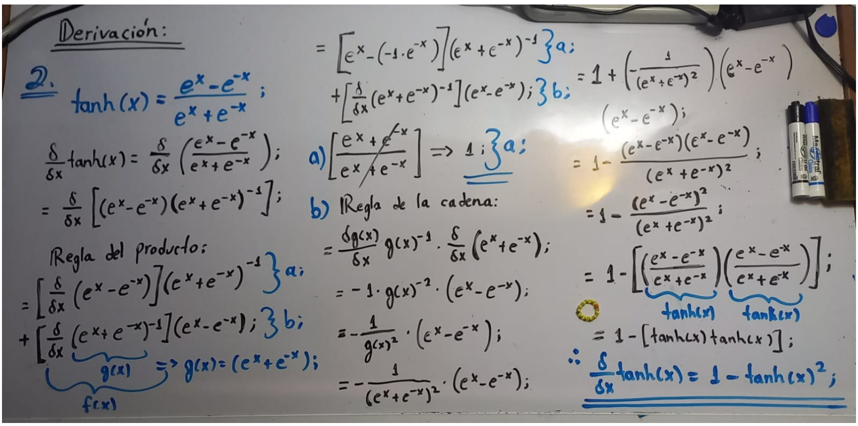
✨ Realizar el proceso de derivación para las funciones;

- Sigmoide
- Tangente Hiperberbólica
- Softplus

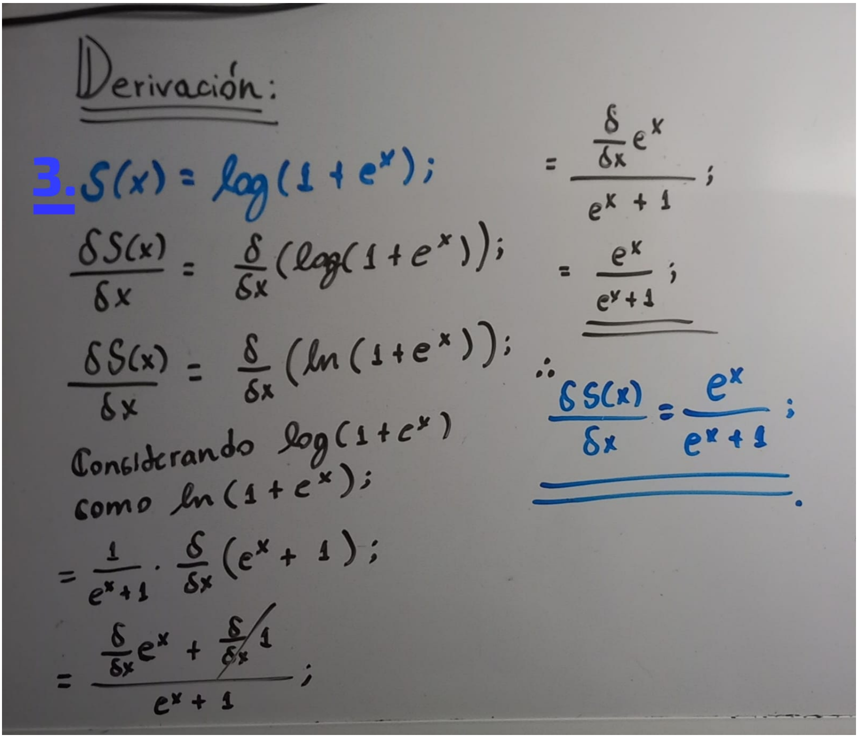
Sigmoide derivada



Tangente Hiperbólica derivada



Softplus derivada



[04]

📓 Demostrar la identidad:

$$\frac{e^{a_i+c}}{\sum_j e^{a_j+c}} = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

para cualquier constante arbitraria c $\in \mathbb{R}$:

Demostración

$$\frac{e^{a_i+c}}{\sum_j e^{a_j+c}} = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

Utilizando la propiedad:

$$e^{a+b} = e^a e^b$$

Factorizando:

$$\frac{e^{a_i}e^c}{\sum_j e^{a_j}e^c} = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

$\frac{e^c}{e^c}$ se elimina...

$$\frac{e^{a_i}\cancel{e^c}}{\sum_j e^{a_j}\cancel{e^c}} = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

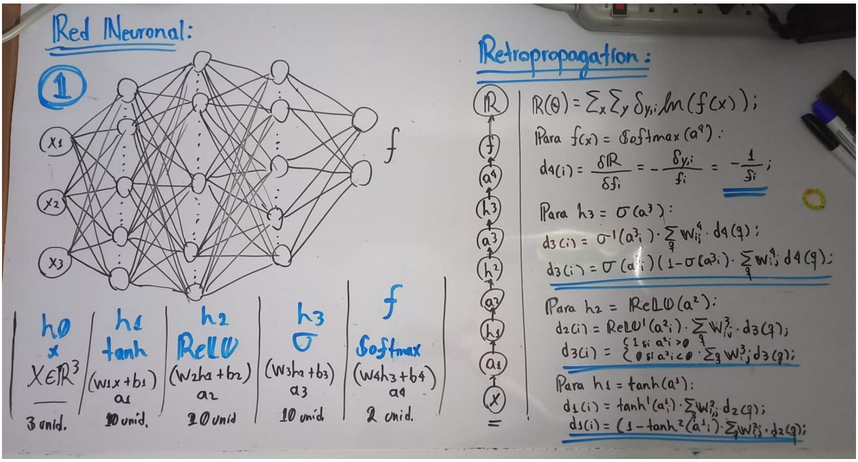
$\therefore$

$$\frac{e^{a_i}}{\sum_j e^{a_j}} = \frac{e^{a_i}}{\sum_j e^{a_j}}$$

[05]

🔁 Utilizar el algoritmo de backpropagation para obtener las derivadas de la siguiente Red Neuronal:

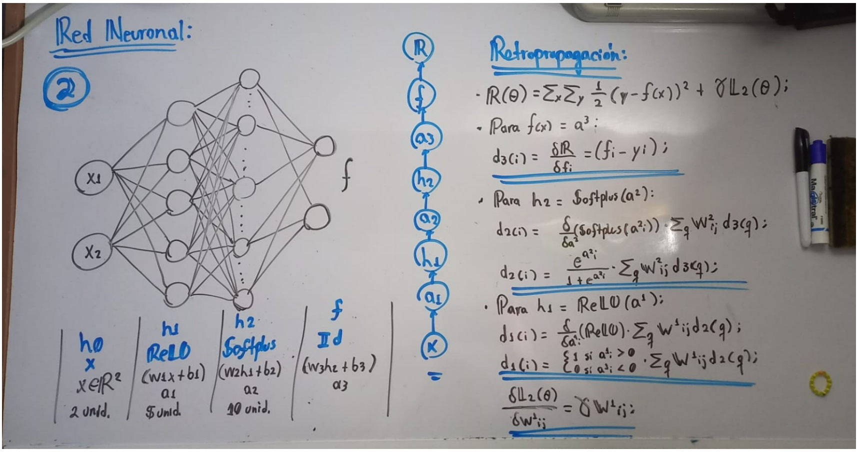
- Capa de entrada: $x \in \mathbb{R}^3$
- Primera capa: $h_1 = tanh(W_1x + b_1)$ con 10 unidades
- Segunda capa: $h_2 = ReLU(W_2h_1 + b_2)$ con 20 unidades
- Tercera capa: $h_3 = \sigma(W_3h_2 + b_3)$ con 10 unidades
- Capa de salida: $f(x) = Softmax(W_4h_3 + b_4)$ con 2 unidades
- Función de riesgo: $R(\theta) = \sum_x \sum_y \delta_{y,i} \ln(f(x))$



[06]

🔁 Usar el metodo de backpropagation para obtener las derivadas de la siguiente Red de regresión que utiliza regularización L_2:

- Capa de entrada: $x \in \mathbb{R}^2$
- Primera capa: $h_1 = ReLU(W_1 x + b_1)$ con 5 unidades
- Segunda capa: $h_2 = Softplus(W_2 h_1 + b_2)$ con 10 unidades
- Capa de salida: $f(x) = W_3 h_2 + b_3$ con 2 unidades
- Función de riesgo: $R(\theta) = \sum_x \sum_y \frac{1}{2}(y - f(x))^2 + \gamma L_2(\theta)$



[08]

Realizado por:

Iván Alejandro Ramos Herrera

💜 [@arhcoder]()