

Otras funciones de activación (Clase 7) 05/11/2023

1) $\text{ReLU}(x) = \max(x, 0)$ (Nair and Hinton 2010)

Obs : la derivada es por partes

$$\text{Si } x \leq 0 \Rightarrow \text{ReLU}' \rightarrow 0$$

$$\text{Si } x > 0 \Rightarrow \text{ReLU}' \text{ es constante}$$

2) $\text{Tanh}(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$ rango: $[-1, 1]$

Obs : funciones definidas en relación a
una parábola \rightarrow no un círculo (hipérbola
unitaria). \longrightarrow Teoría de Relatividad
Especial.

La clase anterior entre nosotros redes neuronales
utilizando de manera directa algunas implementaciones
en SKlearn o PyTorch. Para poder generar una
última capa de salida más fácil de interpretar
se suele utilizar una función llamada Softmax.

- La función softmax se utiliza para obtener la probabilidad de un objeto de entrada $x \in \mathcal{C}$ donde \mathcal{C} es una clase. ($C \in \mathcal{C}$ es una clase)
- En cambio, como vimos en Clases pasadas, en un problema de Regresión generamos como salida valores continuos.

La función Softmax, a veces considerada como una capa de salida, es dada por:

$$\hat{y}_i = \text{Softmax}(\text{output}) = \frac{e^{(\text{output}_i)}}{\sum_j e^{(\text{output}_j)}}$$

C : número de clases

Debido a que la función es

monoacusa y creciente, preservan el orden de la salida.

Recordatorio : $f : A \rightarrow B$ es monótona
 \Leftrightarrow Dados $x, y \in A \subseteq \mathbb{R}$
 con $x \leq y \Rightarrow f(x) \leq f(y)$

O bien si $x \leq y \Rightarrow f(x) \geq f(y)$.

Es decir crece o decrece en todo su dominio.

Utilizaremos Softmax al final de nuestras transformaciones, en problemas de clasificación:

x : input

$$a_1 = W_1 x + b_1$$

$$z_1 = \sigma(w_1)$$

$$a_2 = W_2 z_1 + b_2$$

$$\hat{y} = \text{Softmax}(a_2)$$

La función softmax convierte la salida a_2 en probabilidades.

Notebooks: Neural Network from scratch
+ Softmax.

Retornaremos brevemente al concepto de Cross-entropy (Entropía cruzada).



Entropía y Sorpresa

Técnicamente, queremos medir que tan probable las clases "reales" de los ejemplos están acordes a nuestro modelo, dadas las vectores de características :

$$P(y|x) = \prod_{i=1}^m P(y^{(i)}|x^{(i)})$$

\uparrow
i=1 m -eventos independientes

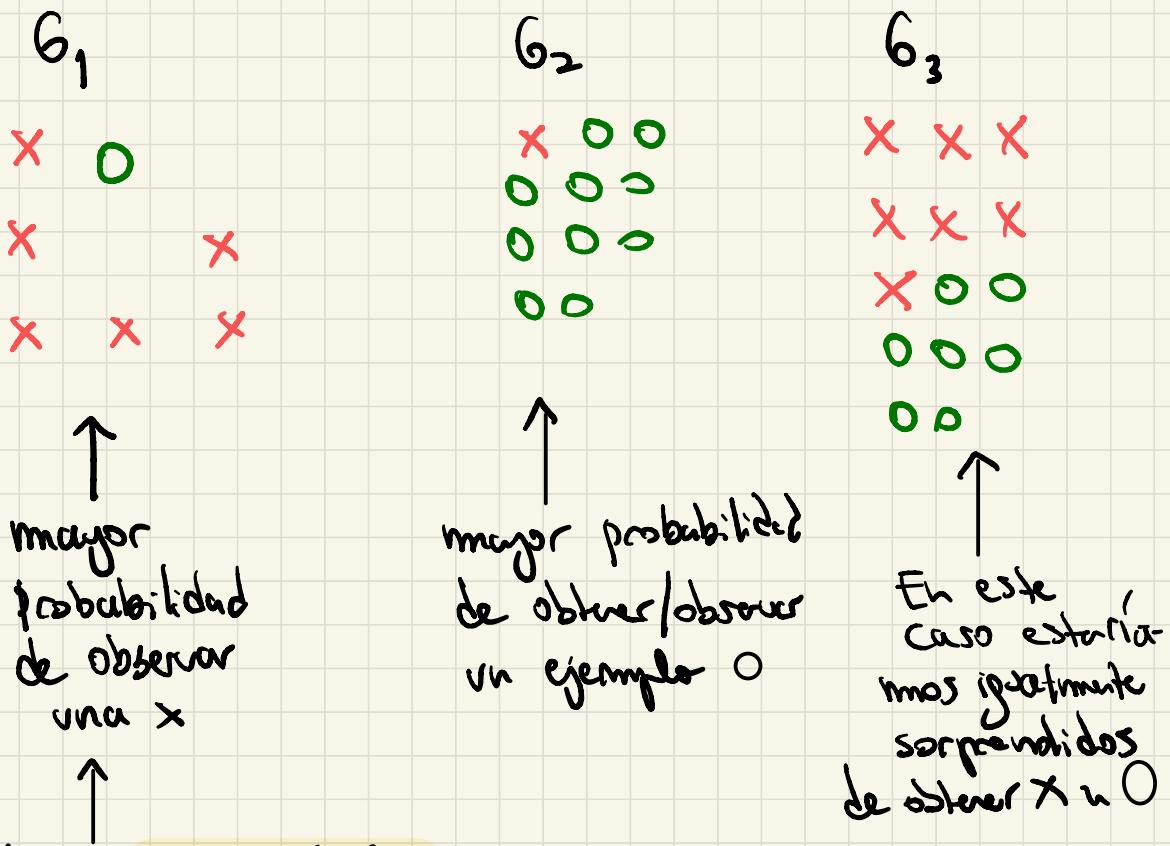
¿Qué es entropía?

- Es una medida del "desorden" o como se distribuye la información para cierto objeto que estudiamos.
Información puede ser : Energía , Reactantes
- El concepto se puede adaptar a diferentes escenarios : Clasificación , Información Mutua , Entropía relativa , Entropía Cruzada .

¿Cómo la entropía mide similitud o diferencias?

Sorpresa : (Motivación)

Supongamos que tenemos un grupo de bolas de color y las separamos en tres grupos aleatoriamente.



No nos sorprendería que al seleccionar un objeto de ese grupo obtuviéramos X

El concepto de probabilidad está ligado al concepto de "sorpresa":

- Cuando la probabilidad de observar algo es baja



El valor de "sorpresa" es alto

- Cuando la probabilidad de observar algo es alta

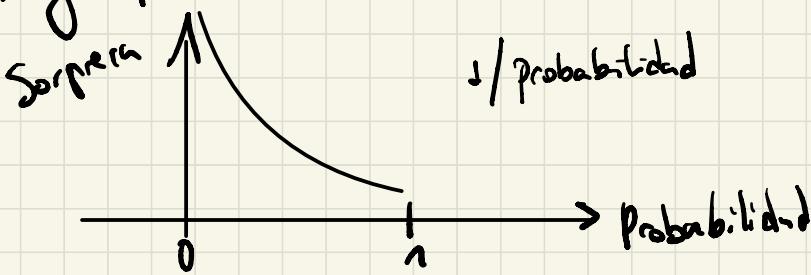


El valor de "sorpresa" es bajo

(Inversamente Proporcionales)

Es decir, gráficamente la relación es dada

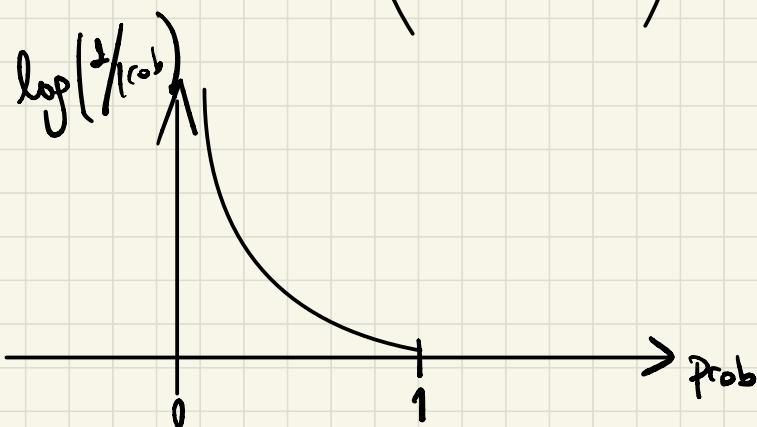
por:



- Sin embargo, la definición $\text{Sorpresa} = \frac{1}{\text{Probabilidad}}$ tiene algunos problemas. (Ya se imaginará que es la división por cero).
- Nos gustaría que para tal caso en que la probabilidad de que algo suceda sea "0" la sorpresa ante tal evento sea " ∞ ". Y en caso de algo siempre suceda probabilidad = 1 la sorpresa sea igual a 0.

• Dástería tomar el logaritmo:

$$\text{Sorpresa} = \log\left(\frac{1}{\text{probabilidad}}\right)$$



• Es costumbre usar logaritmo base 2 cuando medimos la probabilidad de eventos que poseen 2 estados (Cara/Sello ; Negro/Blanco etc.)

Por ejemplo :

Supongamos lanzamos una moneda 3 veces y obtener ,

C C S

La moneda tenía 0,9 de probabilidad de obtener Cara y 0,1 de obtener sello.

⇒ La probabilidad de obtener C-C-S es dada por :

$$0,9 \times 0,9 \times 0,1$$

Eventos Independientes

¿Cuál es el nivel de sorpresa ante tal situación?

$$\Rightarrow \log \left(\frac{1}{0,9 \times 0,9 \times 0,1} \right) = \dots$$

$$= \log(1) - \log(0,9 \times 0,9 \times 0,1)$$

$$= 0 - [\log(0,9) + \log(0,9) + \log(0,1)]$$

$$= - [\Sigma \log(p_i)]$$

Calculemos:

$$-\log_2(0,9) = +0,15$$

$$-\log_2(0,1) = +3,32$$

	Cara	Sello
Probabilidad	0,9	0,1
Sorpresa	0,15	3,32

	Cara	Sello
Probabilidad	0,9	0,1
Sorpresa	0,15	3,32

Si lanzáramos la moneda 100 veces
 > Quisiéramos estimar la "sorpresa" de obtener cara o sello en tal caso :

$$\underbrace{(0,9 \times 100)}_{\text{Prob. } > 100 \text{ repeticiones}} \times \underbrace{0,15}_{\substack{\text{Sorpresa} \\ 1 vez}}$$

$$(0,1 \times 100) \times \underbrace{3,32}_{\substack{\text{Prob. } > 100 \text{ repeticiones} \\ \text{sorpresa} \\ 1 vez}}$$

La sorpresa "total" sería dada por :

$$g = (0,9 \times 100) \times 0,15 + (0,1 \times 100) \times 3,32$$

$$\bar{g} = S/100 = \text{"Media de la sorpresa por lanzamiento"}$$

La medida de la sorpresa = Entropía

Técnicamente : La entropía es el valor esperado de la sorpresa.

$$\mathbb{E}[S] = \sum_{\text{Sorpresa}} \underbrace{s_x}_{\substack{\text{Probabilidad} \\ \text{de "observar" \\ esa sorpresa}}} \times p(x)$$

Entonces, volviendo a la definición de sorpresa:

$$\mathbb{E}[S] = \sum \log\left(\frac{1}{p(x)}\right) p(x)$$

$\langle = \rangle$

$$H_{\text{entropía}} = - \sum p(x) \log(p(x)) \quad (1948) \quad \text{Shannon}$$

Calcularemos la entropía para el ejemplo:

$$G_1 \quad \begin{matrix} X & 0 \\ X & X \\ X & X \end{matrix} \quad \rightarrow E = \underbrace{\frac{1}{7} \log_2 \left(\frac{1}{6/7} \right) + \frac{2}{7} \log_2 \left(\frac{1}{2/7} \right)}_{E = 0,59}$$
$$\begin{matrix} 0,86 \\ 0,22 \\ 0,14 \\ 2,81 \end{matrix}$$

Este más cerca de la entropía de X que de O pues X es más probable!

$$G_2 \quad \begin{matrix} X & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 \end{matrix} \quad \rightarrow E = \underbrace{\frac{1}{11} \log_2 \left(\frac{1}{1/11} \right)}_{0,09} + \underbrace{\frac{10}{11} \log_2 \left(\frac{1}{10/11} \right)}_{0,91} \approx 0,44$$
$$\begin{matrix} 3,46 \\ 0,11 \end{matrix}$$

$$E = 0,44$$

Está más cerca de la entropía de O pues es más probable.

Además $E(G_2) < E(G_1)$ esto tiene sentido porque la probabilidad de obtener menor sorpresa es mayor en G_2 .

Finalmente :

$$\begin{array}{c} 6_3 \\ \times \times \times \\ \times \times \times \\ \times 0 0 \\ 0 0 0 \\ 0 0 \end{array} \rightarrow E = \frac{1}{14} \log_2 \left(\frac{1}{\frac{1}{14}} \right) + \frac{1}{14} \log_2 \left(\frac{1}{\frac{1}{14}} \right)$$
$$E = 1$$

$E(6_3)$ es la más grande.

- Entonces, en 6_3 la sorpresa de ambos eventos es igual asumiendo un valor no tan alto como en 6_1 y 6_2 pero compenso con la probabilidad.
- Podemos pensar la entropía como una medida de similitud o diferencia en como se distribuyen los estados de un evento. Es decir 6_1 y 6_2 son similares en términos de distribución no así 6_3 .

- Observe que hay una relación de forma!
- El "categorical cross entropy" es una generalización de la entropía cruzada que vimos.

$$CE = - \sum_{i=1}^{\# \text{clases}} p_i \log(q_i)$$

↑ Proba estimada

Si tenemos solo dos clases $p \in \{0,1\}$

$$BCE = -p \log(q_i) - (1-p) \log(1-q_i)$$

*↓ Clase 1 ↓ Complemento
 o Clase 2*

- Quisiéramos que los vectores q_i tengan componentes entre 0 > 1.
- Para eso utilizamos Softmax!

Ejemplo : Caso de baja probabilidad

$$\hat{y} = \begin{bmatrix} 0,147 \\ 0,540 \\ 0,133 \\ 0,180 \end{bmatrix}$$

Predicho

$$y = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Real

$$a^{[L]} = \begin{bmatrix} 0,1 \\ 0,8 \\ 0,9 \\ 0,1 \end{bmatrix}$$

L : Última capa
[L]

$$\text{Softmax} \left(\underbrace{\text{Última capa lineal}}_{a^{[L]}} \right) = \sigma(\cdot)_j = \frac{e^{a_j}}{\sum_{k=1}^N e^{a_k}}$$

$$J = - \sum y_i \log(\hat{y}_i)$$

$$\begin{aligned} &= -0 \cdot \log(0,147) - 0 \cdot \log(0,540) \\ &\quad - 1 \cdot \log(0,133) - 0 \cdot \log(0,180) \end{aligned}$$

$$\approx 2,9$$

Ejemplo : Caso de alta probabilidad

$$z^L = \begin{bmatrix} 0,1 \\ 0,4 \\ 2,2 \\ 0 \end{bmatrix} \longrightarrow \underbrace{\text{Softmax}(a^L)}_{\hat{y}} = \begin{bmatrix} 0,088 \\ 0,118 \\ 0,715 \\ 0,079 \end{bmatrix}$$

$$y = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{aligned} \mathcal{L} &= -0 \cdot \log(0.188) - 0 \cdot \log(0.118) \cdot \\ &\quad - 1 \cdot \log(0.715) - 0 \cdot \log(0.079) \\ &\approx 0,3 \end{aligned}$$

Otro ejemplo :

Rojo - Azul - Rojo - Azul - Azul

$$P_R = 2/5$$

Estimación mala de $q_R = 4/5$

$$P_A = 3/5$$

"

$$q_A = 1/5$$

Estimación buena :

$$q_R = 2.5/5$$

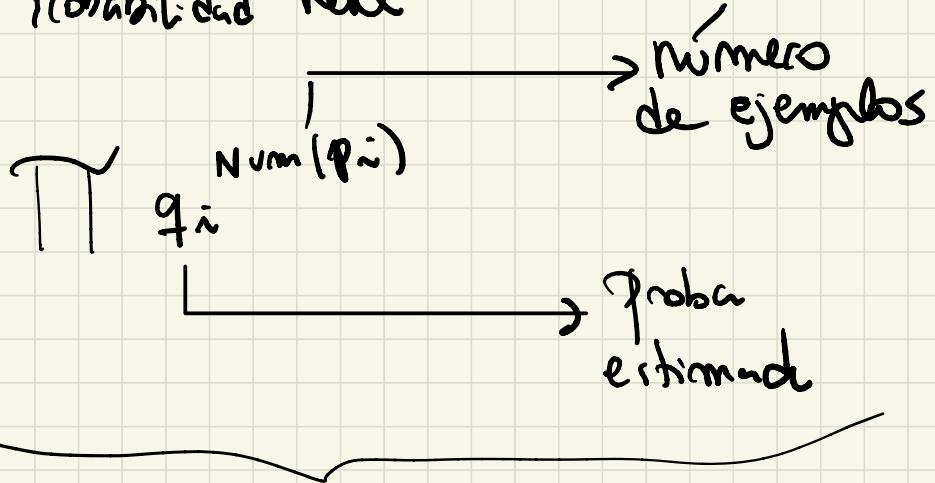
$$q_B = 2.5/5$$

$$P_{\text{mal}}(q^{\text{mala est}}) = \left(\frac{4}{5}\right)^2 \cdot \left(\frac{1}{5}\right)^3 = \frac{16}{25} \cdot \frac{1}{125}$$

$$P_{\text{mal}}(q^{\text{buena est}}) = \left(\frac{2.5}{5}\right)^2 \cdot \left(\frac{2.5}{5}\right)^3 = \frac{1}{32} \Downarrow$$

q_i ≡ Probabilidad estimada

p_i ≡ Probabilidad real



Lo que hicimos en el ejemplo anterior.

$$\frac{1}{N} \log \left(\prod_i q_i^{\text{Num}(p_i)} \right) = \sum p_i \log q_i = -H(q)$$

↓
Aquí lo pensamos
de forma continua

~ Comprime el rango

~ Es siempre
creciente

(Mantine el orden)

~ Permite sumar! (Mult & float points)

Finalmente, ¿Cómo es nuestro Loss?

Llamemos $s_i = \frac{e^{x_i}}{\sum_{j=1}^c e^{x_j}}$ (Softmax)

En el modelo el loss estará dado por:

→ Aquí lo pensamos de forma discreta!

$$l(s_i, y) = -\sum_{i=1}^c y_i \log(s_i)$$

$$l(s_i, y) = -\sum_{i=1}^c y_i \log \left(\frac{e^{a_i}}{\sum_{j=1}^c e^{a_j}} \right)$$

a : Última activación linear.

y_i : Vector en la forma de one hot encoder

c : número de clases

Anexo : Nuestros nuevos gradientes

$$\nabla \text{Softmax} = \begin{bmatrix} \frac{\partial S_1}{\partial a_1} & \frac{\partial S_1}{\partial a_2} & \dots & \frac{\partial S_1}{\partial a_m} \\ \frac{\partial S_2}{\partial a_1} & \frac{\partial S_2}{\partial a_2} & \dots & \frac{\partial S_2}{\partial a_m} \\ \vdots & \ddots & \ddots & \vdots \\ \frac{\partial S_m}{\partial a_1} & \frac{\partial S_m}{\partial a_2} & \dots & \frac{\partial S_m}{\partial a_m} \end{bmatrix}$$



- Jacobiano que representa la matriz de derivadas para $x \in \mathbb{R}^m \rightarrow s \in \mathbb{R}^m$.
- La salida de cada elemento depende de la suma de todos : No habrá valores cero en la diagonal.
- Para simplificar la notación $\alpha \equiv \alpha^{[i]}$

$$\log s_i = \log \left(\frac{e^{a_i}}{\sum_{j=1}^c e^{a_j}} \right)$$

Por propiedad de \log :

$$\log s_i = a_i - \log \left(\sum_{j=1}^c e^{a_j} \right)$$

Derivemos:

$$\frac{\partial \log s_i}{\partial a_k} = \underbrace{\frac{\partial a_i}{\partial a_k}}_{d_1} - \underbrace{\frac{\partial \log \left(\sum_{j=1}^c e^{a_j} \right)}{\partial a_k}}_{d_2}$$

$$d_1 = \begin{cases} 1 & \text{si } i=k \\ 0 & \text{en otro caso} \end{cases} \rightarrow \begin{array}{l} \text{Se llama} \\ \text{delta de} \\ \text{kroncker} \\ \text{a } k \end{array}$$

$$d_2 = \frac{1}{\sum e^{a_j}} \cdot d_1 = \frac{e^{a_k}}{\sum e^{a_j}} = s_k$$

Tomando $d_i \equiv \delta_{ik} / s_j$ entonces :

$$\frac{\partial \log s_i}{\partial a_k} = \delta_{ik} - s_k$$

Derivando a la izquierda implícitamente:

$$\frac{1}{s_i} \frac{\partial s_i}{\partial a_k} = \delta_{ik} - s_k$$

$$\therefore \frac{\partial s_i}{\partial a_k} = s_i (\delta_{ik} - s_k)$$

Gradiente Encontrado

y es una Matriz!

$$\nabla \text{Softmax} = \begin{pmatrix} s_1(1-s_1) & -s_1s_2 & \cdots & -s_1s_N \\ -s_2s_1 & s_2(1-s_2) & \cdots & -s_2s_N \\ \vdots & \ddots & \ddots & \vdots \\ -s_Ns_1 & \cdots & -s_Ns_{N-1} & s_N(1-s_N) \end{pmatrix}$$

Derivadas del cross entropy loss :

$$f(s, y) = - \sum_{i=1}^c y_i \log(s_i)$$

$$\frac{\partial f(s, y)}{\partial a_k} = - \sum_{i=1}^c y_i \frac{\partial \log(s_i)}{\partial a_k}$$

$$= - \sum_{i=1}^c y_i \frac{y_i}{s_i} \cdot \underbrace{\frac{\partial s_i}{\partial a_k}}_{\text{ya lo conocemos}}$$

$$= - \sum_{i=1}^c y_i \frac{y_i}{s_i} \underbrace{s_i(\delta_{ik} - s_k)}_{s_i \neq 0}$$

$$= - \sum_{i=1}^c y_i \delta_{ik} - y_k s_k$$

cuando $i = k$:

$$= - y_k + \sum_{i=1}^c y_k s_k$$

Es decir :

$$\frac{\partial f(s, y)}{\partial a_k} = -y_k + \sum_{i=1}^c y_i s_i$$

Como y_k es "one hot encoder" :

$$\sum_{k=1}^c y_k = 1$$

$$\begin{aligned}\therefore \frac{\partial f(s, y)}{\partial a_k} &= -y_k + s_k \\ &= s_k - y_k \quad \square\end{aligned}$$

Es decir, a pesar de haber agregado complejidad las derivadas son "simples" en su forma reducida. ^{Solo} /
notebooks : neural-networks-solver (Clase 6)
digit - softmax
Proyectos : Primera nota!