Análisis Matemático para Inteligencia Artificial

Verónica Pastor (vpastor@fi.uba.ar),
Martín Errázquin (merrazquin@fi.uba.ar)

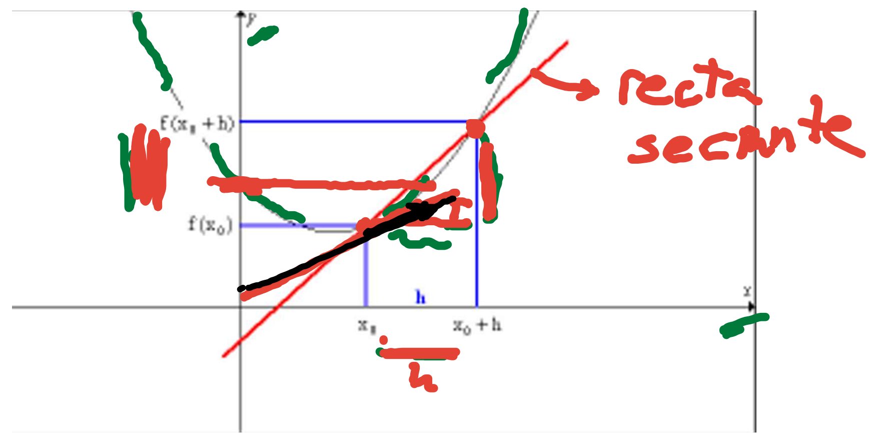
Especialización en Inteligencia Artificial

27/5/2022

Repaso

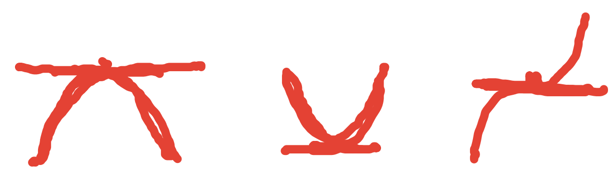
1. En los videos de repaso definimos funciones de cuyo dominio y codominio eran los reales, la gráfica de la función se representa en $\mathbb{R}^2$.
2. Toda función f describe el cambio de una magnitud (v. dependiente) en términos de otra (v. independiente), cuando esta variable se mueve en cierto intervalo $[x_0, x_0 + h]$ la variación total se mide como $f(x_0 + h) - f(x_0)$.
3. Mientras que la variación media es $\frac{f(x_0+h)-f(x_0)}{(x_0+h)-x_0}$. Geométricamente, podemos ver la variación media como la pendiente de la recta secante.
4. Cuando hacemos que $h \to 0$, ...

$$f : \mathbb{R} \to \mathbb{R}$$
$$x \mapsto y = f(x)$$



... esto nos conduce a la definición de derivada de f en x_0:

$$lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h} = f'(x_0)$$



Dada $f : D \subset \mathbb{R}^n \to \mathbb{R}^m$.

- Si $m = 1$ diremos que es una función $f(x) = x^2$
  - escalar, si $n = 1$,
  - campo escalar, $n > 1$. $n=2$ $\quad f(x,y) = x^2 + y^2$ $\quad z = x^2 + y^2$
- Si $m > 1$ diremos que es una función
  - vectorial, si $n = 1$, $\quad f(t) = (t, 2t^2)$
  - campo vectorial, $n > 1$. $\quad f(x,y) = (\underbrace{x^2}_{f_1}, \underbrace{0}_{f_2}, \underbrace{y^2 - 1}_{f_3})$

$k = 1$
$1 = x^2 + y^2$
$k = 4$
$4 = x^2 + y^2$

Conjuntos de Nivel Dada $f : D \subset \mathbb{R}^n \to \mathbb{R}^m$ el conjunto de nivel k de f, $L_k \subset \mathbb{R}^n$, definido por:

$$L_k = \{x \in \mathbb{R}^n \ / \ x \in D \wedge f(x) = k\}$$

La representación geométrica de L_k se obtiene identificando gráficamente los puntos del dominio de la función para los cuales el valor de f es igual a k, para graficar no es necesario agregar un eje.

Derivando campos ...

- escalares: Sea $f : D \subset \mathbb{R}^n \to \mathbb{R}$, $(x_1, ..., x_n)^T \mapsto f((x_1, ..., x_n)^T)$, se definen las derivadas parciales como:

$$\frac{\partial f}{\partial x_1} = lim_{h \to 0} \frac{f(x_1 + h, \overbrace{x_2, ..., x_n}^{cte}) - f(x_1, \overbrace{x_2, ..., x_n}^{cte})}{h}$$

$$\vdots$$

$$\frac{\partial f}{\partial x_n} = lim_{h \to 0} \frac{f(\overbrace{x_1, x_2, ...,}^{cte} x_n + h) - f(\overbrace{x_1, x_2, ...,}^{cte} x_n)}{h}$$

Se define el gradiente como: $\nabla f = \left(\frac{\partial f}{\partial x_1} \cdots \frac{\partial f}{\partial x_n} \right)$.

Importante: El gradiente apunta en la dirección de máximo crecimiento.

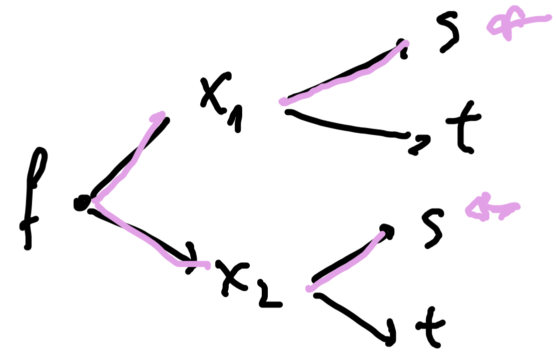
- vectoriales: Sea $f : D \subset \mathbb{R}^n \to \mathbb{R}^m$, $(x_1, ..., x_n)^T \mapsto (f_1((x_1, ..., x_n)^T), ..., f_m((x_1, ..., x_n)^T))$, se define el jacobiano como:

$$J_f = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ & \vdots & \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \begin{array}{l} \to \nabla f_1 \\ \\ \to \nabla f_m \end{array} \qquad J\{m \times n\}$$

Regla de la Cadena en forma matricial

Sea $f(\underbrace{x_1(s,t)}_{f_1}, \underbrace{x_2}_{f_2}(s,t))$

$$\frac{\partial f}{\partial s} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial s} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial s}$$



$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial t}$$

Y luego

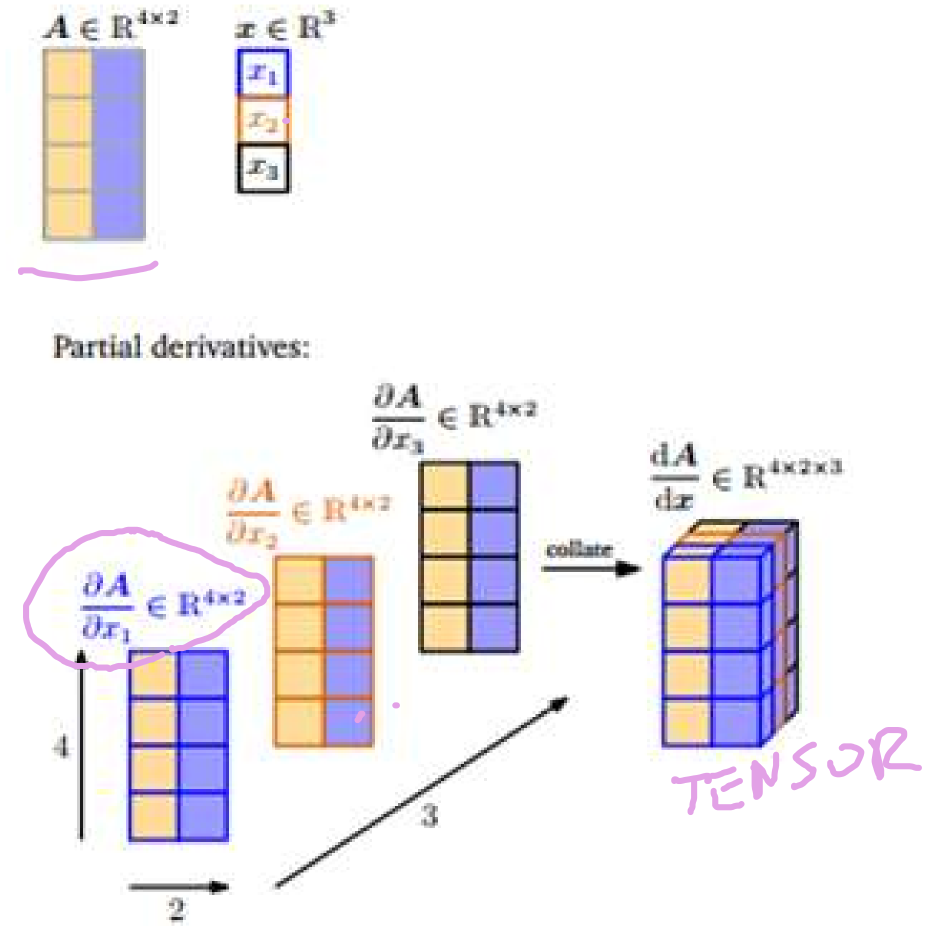
$$\frac{df}{d(s,t)} = \frac{df}{dx}\frac{dx}{d(s,t)} = \underbrace{\begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix}}_{\nabla f(x_1,x_2)} \underbrace{\begin{bmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial s} & \frac{\partial x_2}{\partial t} \end{bmatrix}}_{J}$$

Recordemos reglas de derivación:

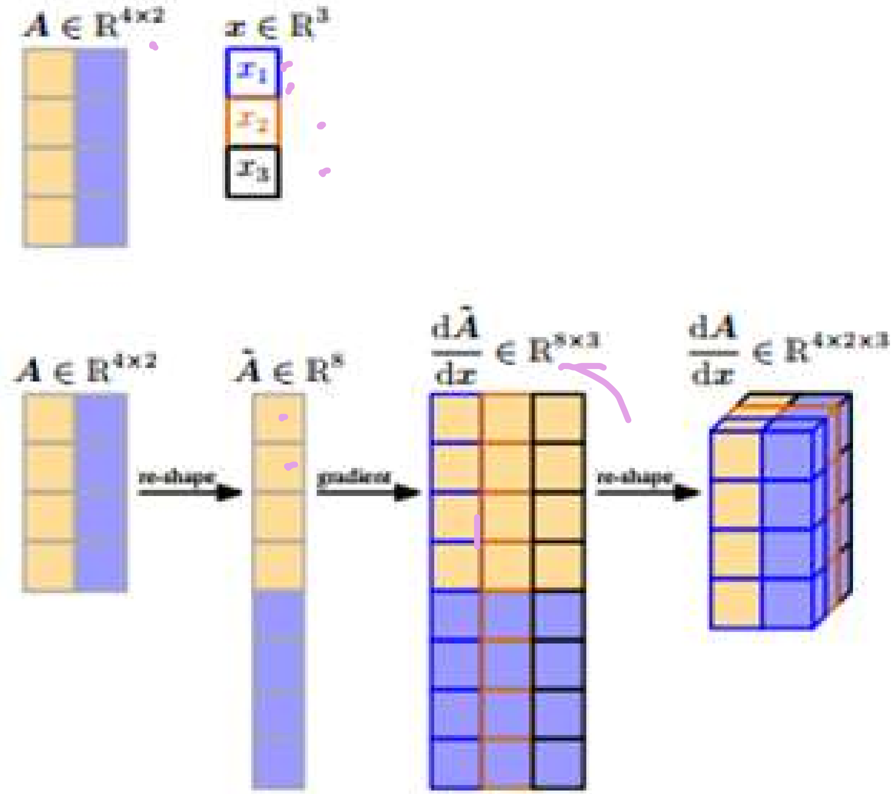
- $\frac{\partial (f+g)(s)}{\partial s} = \frac{\partial f}{\partial s} + \frac{\partial g}{\partial s}$
- $\frac{\partial (fg)(s)}{\partial s} = \frac{\partial f}{\partial s} g(s) + f(s)\frac{\partial g}{\partial s}$

Derivada de matrices

A x



(a) Approach 1: We compute the partial derivative $\frac{\partial A}{\partial x_1}, \frac{\partial A}{\partial x_2}, \frac{\partial A}{\partial x_3}$, each of which is a 4×2 matrix, and collate them in a $4 \times 2 \times 3$ tensor.



(b) Approach 2: We re-shape (flatten) $A \in \mathbb{R}^{4\times2}$ into a vector $\tilde{A} \in \mathbb{R}^8$. Then, we compute the gradient $\frac{d\tilde{A}}{dx} \in \mathbb{R}^{8\times3}$. We obtain the gradient tensor by re-shaping this gradient as illustrated above.

La matriz Hessiana es aquella cuyas derivadas de orden 2 de f respecto a $x \in \mathbb{R}^n$ se ubican:

derivada segunda

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

deriva resp a x_1

respecto a x_1 2 veces

deriva resp. a x_n

$$\frac{\partial^2 f}{\partial x_1^2} = \frac{\partial}{\partial x_1}\left(\frac{\partial f}{\partial x_1}\right)$$

$$* \quad \frac{\partial^2 f}{\partial x_1 \partial x_n} = \frac{\partial}{\partial x_1}\left(\frac{\partial f}{\partial x_n}\right)$$

Ejemplo $f : \mathbb{R}^2 \to \mathbb{R}$

$$f(x,y) = 2xy^2$$

• Calculo

$$\frac{\partial f}{\partial x}(x,y) = 2y^2 \qquad \frac{\partial^2 f}{\partial x^2}(x,y) = 0 \qquad \frac{\partial^2 f}{\partial y \partial x}(x,y) = 4y$$

$$\frac{\partial f}{\partial y}(x,y) = 4xy \qquad \frac{\partial^2 f}{\partial x \partial y}(x,y) = 4y \qquad \frac{\partial^2 f}{\partial y^2}(x,y) = 4x$$

$$H = \begin{bmatrix} 0 & 4y \\ 4y & 4x \end{bmatrix}$$

$H = H^T$ simétrica

$$\exists S : S^T H S = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

Relacionado a F. Cuadráticas

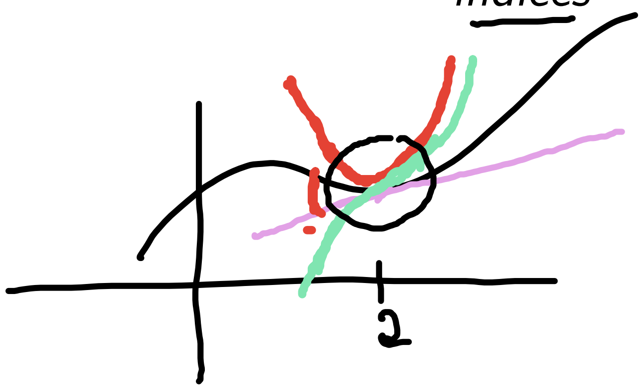
Sea f un campo escalar $f : \mathbb{R}^n \to \mathbb{R}$, asumiendo que posee derivadas parciales de todo orden en un entorno de un punto $a \in \mathbb{R}^n$, se define el polinomio de Taylor de grado k:

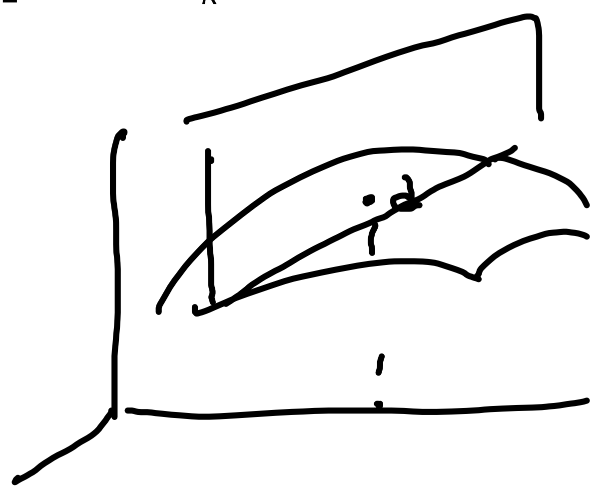
$a = (a_1, .., a_n)$

$$P_k(x) = f(a) + \underbrace{\sum_{i=1}^{n} \frac{\partial f}{\partial x_i}(a)(x_i - a_i)}_{\text{aprox. lineal}} + \frac{1}{2!} \sum_{i,j=1}^{n} \frac{\partial^2 f}{\partial x_i \partial x_j}(a)(x_i - a_i)(x_j - a_j) + \ldots +$$

$\frac{1}{3!} \sum \partial^3$

$$\ldots + \frac{1}{k!} \sum_{\text{indices}} \frac{\partial^k f}{\partial x_{i_1} \ldots \partial x_{i_k}}(a)(x_{i_1} - a_{i_1}) \ldots (x_{i_k} - a_{i_k})$$





Diferenciación Automática

```
class Cube:
    def forward(x):
        self.last_x = x
        return x**3
    def backwards():
        return(self.last_x^2 · 3
```

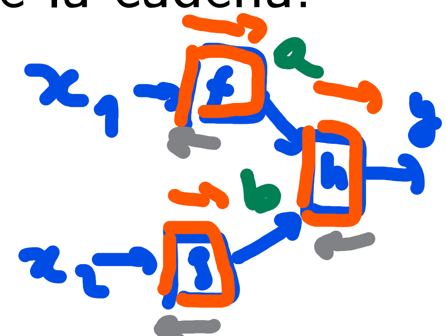
Sean, para una función f:

- $x_1, \ldots, x_d$ las variables de entrada
- $x_{d+1}, \ldots, x_{D-1}$ las variables intermedias
- x_D la variable de salida
- g_i funciones elementales
- $Hij(x_i)$ el conjunto de nodos hijos de cada x_i

Así queda definido un grafo de cómputo. Recordando que $f = D$, tenemos que $\frac{\partial f}{\partial x_D} = 1$. Para las otras variables x_i aplicamos la regla de la cadena:

$$\frac{\partial y}{\partial x_1} = \underbrace{\frac{\partial y}{\partial a}}_{h_1} \cdot \underbrace{\frac{\partial a}{\partial x_1}}_{f}$$

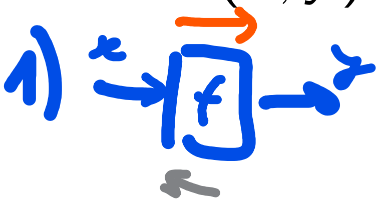
$$\frac{\partial f}{\partial x_i} = \sum_{x_j \in Hij(x_i)} \frac{\partial f}{\partial x_j}\frac{\partial x_j}{\partial x_i} = \sum_{x_j \in Hij(x_i)} \frac{\partial f}{\partial g_j}\frac{\partial x_j}{\partial x_i}$$



- La diferenciación automática se puede utilizar siempre que la función pueda representarse como un grafo de cómputo.
- La gran ganancia de este mecanismo está en que cada función sólo precisa saber cómo derivarse a sí misma, permitiendo OOP.

Diferenciación Automática: ejemplo

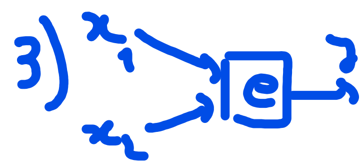
Sean $e(x, y) = \cancel{xy}\ x^y$, $f(x) = 3x$, $g(x) = x^2$, $h(x) = sen(x)$

1) 

$\frac{dy}{dx} = f'(x)$

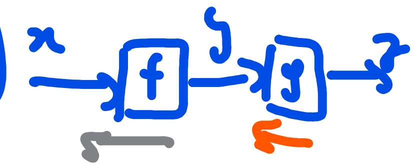
2) 

$\frac{dy}{dx} = g'(x)$

3) 

$\nabla e(x_1, x_2) = \left(\frac{\partial e}{\partial x_1}, \frac{\partial e}{\partial x_2}\right)$

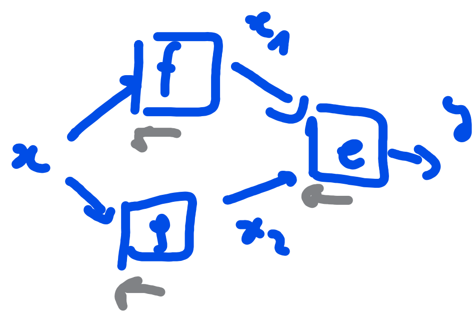
↳ $\frac{\partial e}{\partial x_1} = \nabla e_1(x_1, x_2)$

4) 

$\frac{dz}{dx} = \underbrace{\frac{dz}{dy}} \cdot \underbrace{\frac{dy}{dx}} = g'(y) \cdot f'(x)$

requiere forward de f

$y = g \circ f(x) = (3x)^2$

5) 

$\frac{dy}{dx} = \frac{dy}{dx_1} \cdot \frac{dx_1}{dx} + \frac{dy}{dx_2} \cdot \frac{dx_2}{dx}$

$= \nabla e_1(x_1, x_2) \cdot f'(x) + \nabla e_2(x_1, x_2) \cdot g'(x)$

$y = (3x) . (x^2)$

$\frac{dy}{dx} = \frac{dy}{df} \cdot \frac{df}{dx} + \frac{dy}{dg} \cdot \frac{dg}{dx}$

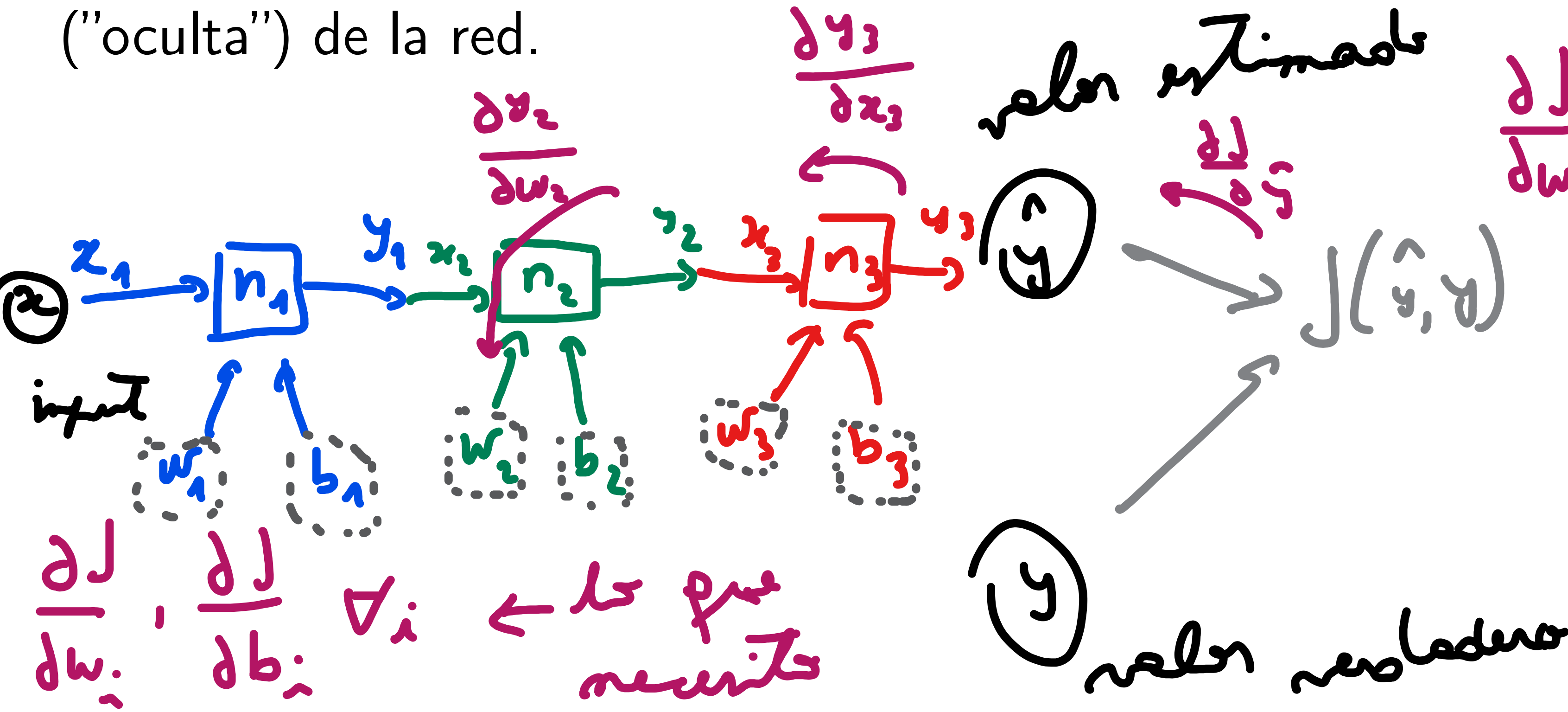
Backpropagation

¿Dónde se aplica la diferenciación automática? En Backpropagation (o simplemente Backprop), el algoritmo utilizado para entrenar redes neuronales.

$y = f(x, w, b)$

¿Qué función cumple? La de computar las derivadas de la función de error/costo respecto de *cada* parámetro de la red neuronal.

En este caso, las variables intermedias son cada salida de cada capa interna ("oculta") de la red.



$$\frac{\partial J}{\partial w_2} = \underbrace{\frac{\partial J}{\partial \hat{y}}}_{J} \cdot \underbrace{\frac{\partial y_3}{\partial x_3}}_{n_3} \cdot \underbrace{\frac{\partial y_2}{\partial w_2}}_{n_2}$$

$$\frac{\partial J}{\partial y_2} \cdot \frac{\partial y_2}{\partial w_2}$$