IIC 2433 Minería de Datos

https://github.com/marcelomendoza/IIC2433

- GRADIENTE DESCENDENTE -

Regresión logística

Si el ajuste es adecuado:

$$P(y \mid \mathbf{x}) = \theta(y \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x})$$

Luego, podemos expresar la función de verosimilitud:

$$P(y_1, \ldots, y_N \mid \mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n).$$

Regresión logística

Maximizamos la función de verosimilitud:

$$\begin{aligned} & \max && \textstyle\prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n) \\ \Leftrightarrow\ & \max && \ln\left(\textstyle\prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n)\right) \\ \equiv\ & \max && \textstyle\sum_{n=1}^{N} \ln P(y_n \mid \mathbf{x}_n) \end{aligned}$$

Regresión logística

Maximizamos la función de verosimilitud:

$$
\begin{aligned}
& \max && \prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n) \\
\Leftrightarrow\ & \max && \ln\left(\prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n)\right) \\
\equiv\ & \max && \sum_{n=1}^{N} \ln P(y_n \mid \mathbf{x}_n) \\
\Leftrightarrow\ & \min && -\frac{1}{N}\sum_{n=1}^{N} \ln P(y_n \mid \mathbf{x}_n) \\
\equiv\ & \min && \frac{1}{N}\sum_{n=1}^{N} \ln \frac{1}{P(y_n \mid \mathbf{x}_n)} \\
\equiv\ & \min && \frac{1}{N}\sum_{n=1}^{N} \ln \frac{1}{\theta(y_n \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x}_n)}
\end{aligned}
$$

Regresión logística

Maximizamos la función de verosimilitud:

$$
\begin{aligned}
& \max && \prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n) \\
\Leftrightarrow\ & \max && \ln\left(\prod_{n=1}^{N} P(y_n \mid \mathbf{x}_n)\right) \\
\equiv\ & \max && \sum_{n=1}^{N} \ln P(y_n \mid \mathbf{x}_n) \\
\Leftrightarrow\ & \min && -\frac{1}{N}\sum_{n=1}^{N} \ln P(y_n \mid \mathbf{x}_n) \\
\equiv\ & \min && \frac{1}{N}\sum_{n=1}^{N} \ln \frac{1}{P(y_n|\mathbf{x}_n)} \\
\equiv\ & \min && \frac{1}{N}\sum_{n=1}^{N} \ln \frac{1}{\theta(y_n \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x}_n)} \\
\equiv\ & \min && \frac{1}{N}\sum_{n=1}^{N} \ln(1 + e^{-y_n \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x}_n})
\end{aligned}
$$

$$\theta(s) = \frac{1}{1+e^{-s}}.$$

Regresión logística

Tenemos una expresión para:

Parámetros del modelo

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + e^{-y_n \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x}_n})$$

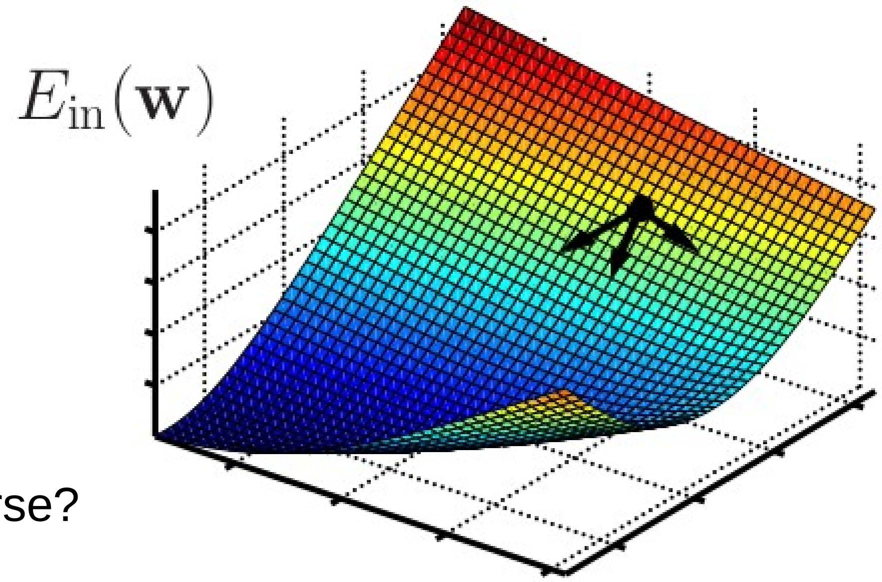
Cross-entropy

La función es convexa, por lo que podemos optimizarla de forma iterativa:

Idea del gradiente descendente:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \eta\hat{\mathbf{v}}$$

¿En cuál dirección conviene moverse?



Gradiente descendente

$$\begin{aligned}\Delta E_{\text{in}} &= E_{\text{in}}(\mathbf{w}(t+1)) - E_{\text{in}}(\mathbf{w}(t)) \\ &= E_{\text{in}}(\mathbf{w}(t) + \eta\hat{\mathbf{v}}) - E_{\text{in}}(\mathbf{w}(t)) \\ &= \eta\, \nabla E_{\text{in}}(\mathbf{w}(t))^{\mathrm{T}}\hat{\mathbf{v}} + O(\eta^2)\end{aligned}$$

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(k+1)}(c)}{(k+1)!}(x - x_0)^{k+1}.$$

$$\Delta E_{\text{in}} = E_{\text{in}}(\mathbf{w}(t+1)) - E_{\text{in}}(\mathbf{w}(t))$$

$$\underbrace{f(x_0 + \eta \cdot x)}_{f(x_0) + \nabla f(x_0) \cdot (x_0 + \eta \cdot x - x_0) + \ldots} - f(x_0)$$

(expansión de Taylor de 1er orden)

Gradiente descendente

$$
\begin{aligned}
\Delta E_{\text{in}} &= E_{\text{in}}(\mathbf{w}(t+1)) - E_{\text{in}}(\mathbf{w}(t)) \\
&= E_{\text{in}}(\mathbf{w}(t) + \eta\hat{\mathbf{v}}) - E_{\text{in}}(\mathbf{w}(t)) \\
&= \eta\, \nabla E_{\text{in}}(\mathbf{w}(t))^{\mathrm{T}}\hat{\mathbf{v}} + O(\eta^2)
\end{aligned}
$$

$$
f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(k+1)}(c)}{(k+1)!}(x - x_0)^{k+1}.
$$

$$
\Delta E_{\text{in}} = E_{\text{in}}(\mathbf{w}(t+1)) - E_{\text{in}}(\mathbf{w}(t))
$$

$$
\underbrace{f(x_0 + \eta \cdot x)}_{\cancel{f(x_0)} + \nabla f(x_0) \cdot (\cancel{x_0} + \eta \cdot x - \cancel{x_0}) + \dots} - \cancel{f(x_0)}
$$

(expansión de Taylor de 1er orden)

Gradiente descendente

$$-\nabla E_{\text{in}}(\mathbf{w}(t))$$

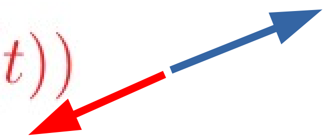
$$= \eta \underbrace{\nabla E_{\text{in}}(\mathbf{w}(t))^{\mathrm{T}} \hat{\mathbf{v}}} + O(\eta^2) \qquad \nabla E_{\text{in}}(\mathbf{w}(t))$$

Minimizado en $\hat{\mathbf{v}} = -\dfrac{\nabla E_{\text{in}}(\mathbf{w}(t))}{\| \nabla E_{\text{in}}(\mathbf{w}(t)) \|}$

Gradiente descendente

$$-\nabla E_{\text{in}}(\mathbf{w}(t))$$

$$= \eta \nabla E_{\text{in}}(\mathbf{w}(t))^{\mathrm{T}}\hat{\mathbf{v}} + O(\eta^2) \qquad \nabla E_{\text{in}}(\mathbf{w}(t))$$



Minimizado en $\hat{\mathbf{v}} = -\dfrac{\nabla E_{\text{in}}(\mathbf{w}(t))}{\|\nabla E_{\text{in}}(\mathbf{w}(t))\|}$

Gradiente descendente

1: Initialize at step $t = 0$ to $\mathbf{w}(0)$.
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: Compute the gradient

$$\mathbf{g}_t = \nabla E_{\text{in}}(\mathbf{w}(t)).$$

4: Move in the direction $\mathbf{v}_t = -\mathbf{g}_t$.
5: Update the weights:

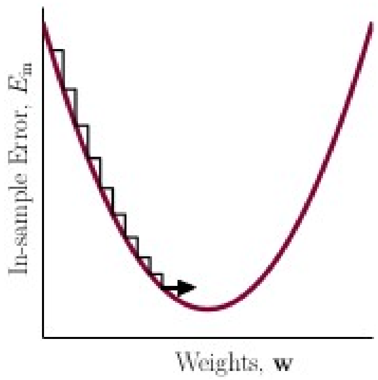
$$\mathbf{w}(t+1) = \mathbf{w}(t) + \eta \mathbf{v}_t.$$

6: Iterate 'until it is time to stop'.
7: **end for**
8: Return the final weights.

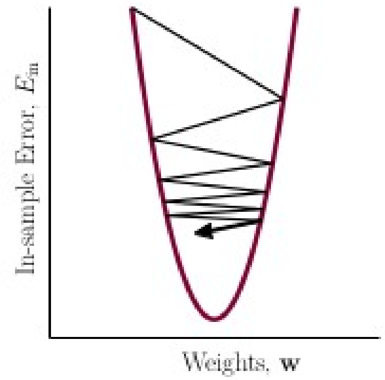
Gradiente descendente

El efecto del **learning rate**:

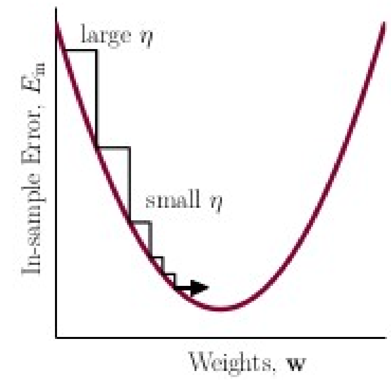
η muy pequeño



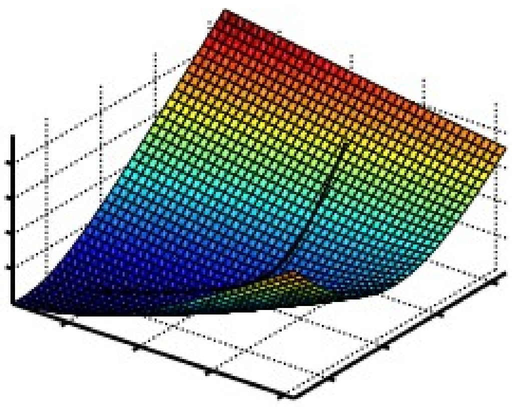
η muy grande



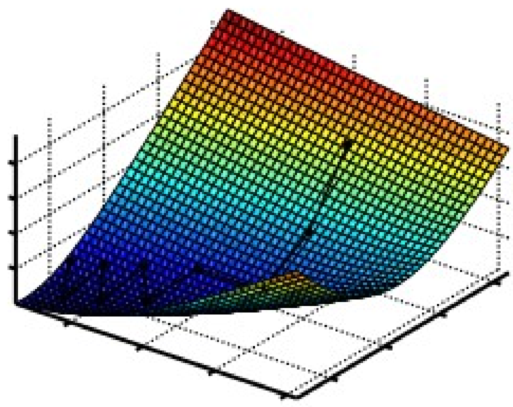
η_t



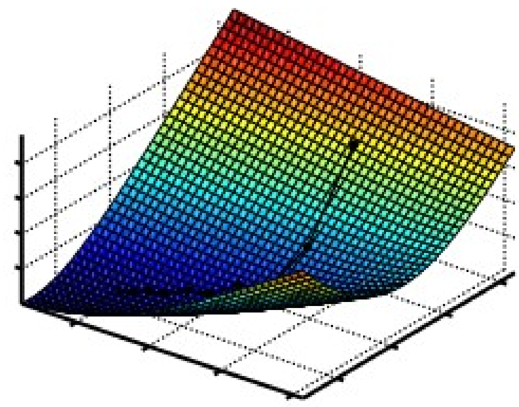
Adam



$\eta = 0.1$; 75 steps



$\eta = 2$; 10 steps



variable η_t; 10 steps

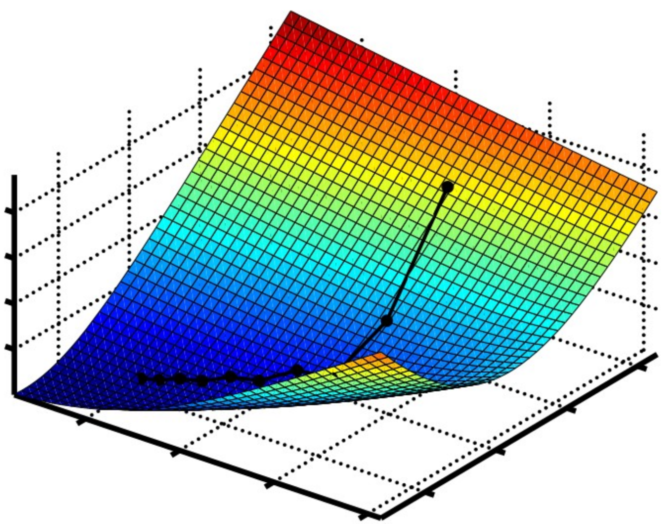
Gradiente descendente

Una variación de gradiente descendente considera la evaluación del gradiente en una muestra del training set.

$$\mathbf{w}(t+1) \leftarrow \mathbf{w}(t) - \eta \nabla_{\mathbf{w}} e(\mathbf{w}, \mathbf{x}_*, y_*)$$

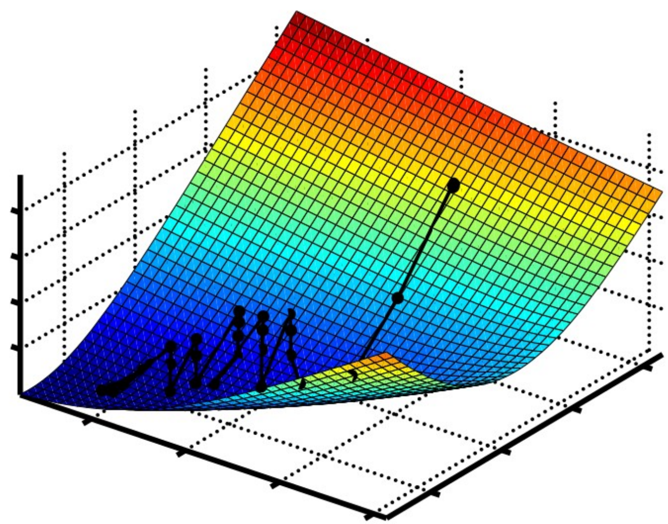
Dado que la muestra se toma al azar, se le denomina gradiente descendente estocástico.

GD



$\eta = 6$
10 steps
$N = 10$

SGD



$\eta = 2$
30 steps

Puede ayudar a escapar de óptimos locales

Regresión logística + gradiente descendente

Para regresión logística encontramos que:

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + e^{-y_n \cdot \mathbf{w}^{\mathrm{T}}\mathbf{x}_n})$$

Muestre que:

$$\begin{aligned}\nabla E_{\text{in}}(\mathbf{w}) &= -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n \mathbf{x}_n}{1 + e^{y_n \mathbf{w}^{\mathrm{T}}\mathbf{x}_n}} \\ &= \frac{1}{N}\sum_{n=1}^{N} -y_n \mathbf{x}_n \theta(-y_n \mathbf{w}^{\mathrm{T}}\mathbf{x}_n).\end{aligned}$$

Regresión logística + gradiente descendente

Logistic regression algorithm:

1: Initialize the weights at time step $t = 0$ to $\mathbf{w}(0)$.

2: **for** $t = 0, 1, 2, \ldots$ **do**

3: Compute the gradient

$$\mathbf{g}_t = -\frac{1}{N}\sum_{n=1}^{N}\frac{y_n \mathbf{x}_n}{1 + e^{y_n \mathbf{w}^{\mathrm{T}}(t)\mathbf{x}_n}}.$$

4: Set the direction to move, $\mathbf{v}_t = -\mathbf{g}_t$.

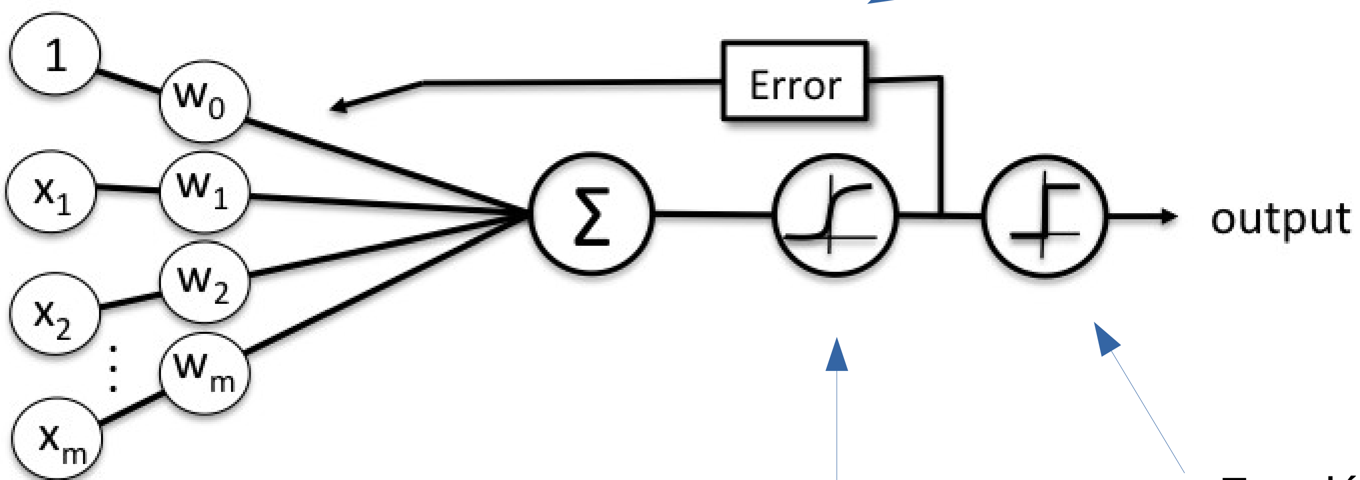
5: Update the weights: $\mathbf{w}(t+1) = \mathbf{w}(t) + \eta\mathbf{v}_t$.

6: Iterate to the next step until it is time to stop.

7: Return the final weights $\mathbf{w}$.

Regresión logística + gradiente descendente

$$\approx \quad -\eta \| \nabla E_{\text{in}}(\mathbf{w}(t)) \|$$



Función umbral

Función logística

- BACKPROPAGATION -

El perceptrón multicapa (Multi-Layer Perceptron)

Forward propagation

$$\mathbf{x} = \mathbf{x}^{(0)} \xrightarrow{\mathrm{W}^{(1)}} \mathbf{s}^{(1)} \xrightarrow{\theta} \mathbf{x}^{(1)} \xrightarrow{\mathrm{W}^{(2)}} \mathbf{s}^{(2)} \xrightarrow{\theta} \mathbf{x}^{(2)} \cdots \xrightarrow{\mathrm{W}^{(L)}} \mathbf{s}^{(L)} \xrightarrow{\theta} \mathbf{x}^{(L)} = h(\mathbf{x}).$$

Forward propagation

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{x}$
2: **for** $\ell = 1$ to L **do**
3: $\quad \mathbf{s}^{(\ell)} \leftarrow (\mathrm{W}^{(\ell)})^{\mathrm{T}}\mathbf{x}^{(\ell-1)}$
4: $\quad \mathbf{x}^{(\ell)} \leftarrow \begin{bmatrix} 1 \\ \theta(\mathbf{s}^{(\ell)}) \end{bmatrix}$
5: **end for**
6: $h(\mathbf{x}) = \mathbf{x}^{(L)}$

Objetivo supervisado:

$$E_{\text{in}}(h) = E_{\text{in}}(\mathrm{W}) = \frac{1}{N}\sum_{n=1}^{N}(h(\mathbf{x}_n) - y_n)^2$$

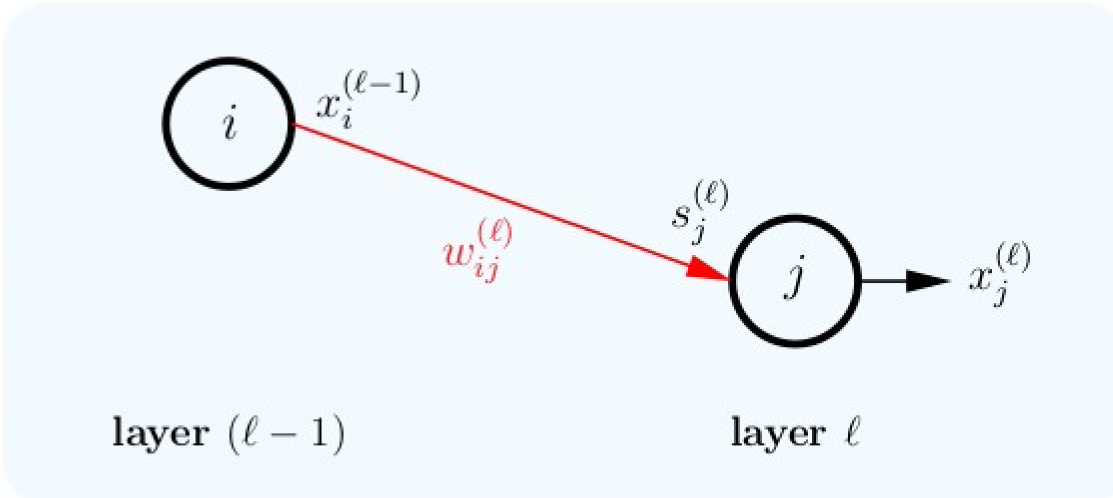
Dado que $\theta = \tanh$, E_{in} es diferenciable usando GD sobre

$$\underbrace{\mathrm{W} = \{\mathrm{W}^{(1)}, \mathrm{W}^{(2)}, \ldots, \mathrm{W}^{(L)}\}}_{\text{Parámetros del modelo}}$$

Feed-Forward

$$\mathbf{x} = \mathbf{x}^{(0)} \xrightarrow{\mathrm{W}^{(1)}} \mathbf{s}^{(1)} \xrightarrow{\theta} \mathbf{x}^{(1)} \xrightarrow{\mathrm{W}^{(2)}} \mathbf{s}^{(2)} \xrightarrow{\theta} \mathbf{x}^{(2)} \cdots \xrightarrow{\mathrm{W}^{(L)}} \mathbf{s}^{(L)} \xrightarrow{\theta} \mathbf{x}^{(L)} = h(\mathbf{x}).$$

Nodo a nodo:

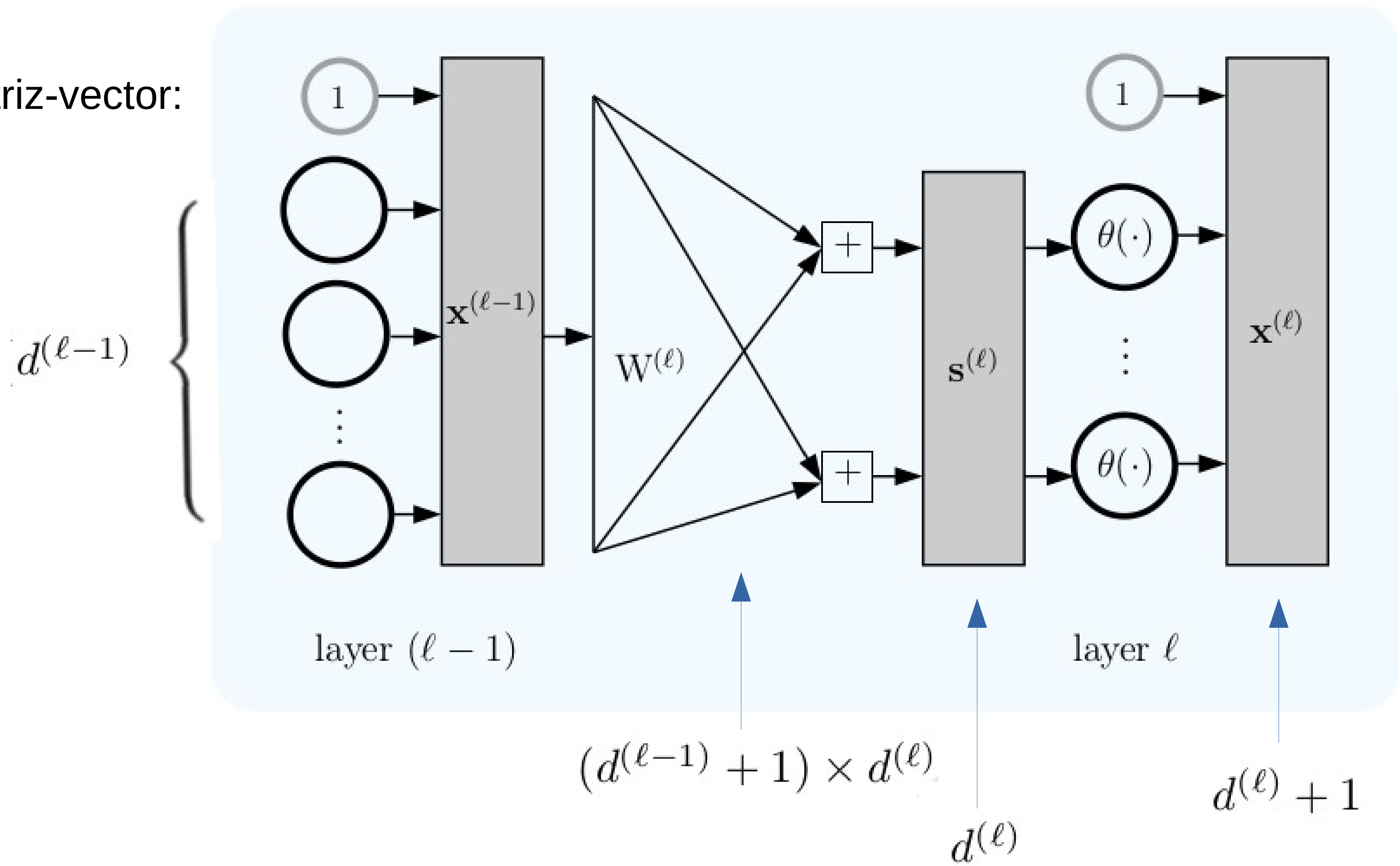


Feed-Forward

$$\mathbf{x} = \mathbf{x}^{(0)} \xrightarrow{\mathrm{W}^{(1)}} \mathbf{s}^{(1)} \xrightarrow{\theta} \mathbf{x}^{(1)} \xrightarrow{\mathrm{W}^{(2)}} \mathbf{s}^{(2)} \xrightarrow{\theta} \mathbf{x}^{(2)} \cdots \xrightarrow{\mathrm{W}^{(L)}} \mathbf{s}^{(L)} \xrightarrow{\theta} \mathbf{x}^{(L)} = h(\mathbf{x}).$$

$$\mathbf{s}^{(\ell)} \leftarrow (\mathrm{W}^{(\ell)})^{\mathrm{T}} \mathbf{x}^{(\ell-1)}$$

Matriz-vector:



Backpropagation

Minimizar:

$$E_{\text{in}}(h) = E_{\text{in}}(\mathrm{W}) = \frac{1}{N}\sum_{n=1}^{N}(h(\mathbf{x}_n) - y_n)^2$$

Podemos usar la idea de gradiente descendente:

$$\mathrm{W}(t+1) = \mathrm{W}(t) - \eta \nabla E_{\text{in}}(\mathrm{W}(t))$$

Dado que:

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}\overbrace{\mathrm{e}(h(\mathbf{x}_n), y_n)}^{\mathrm{e}_n}$$

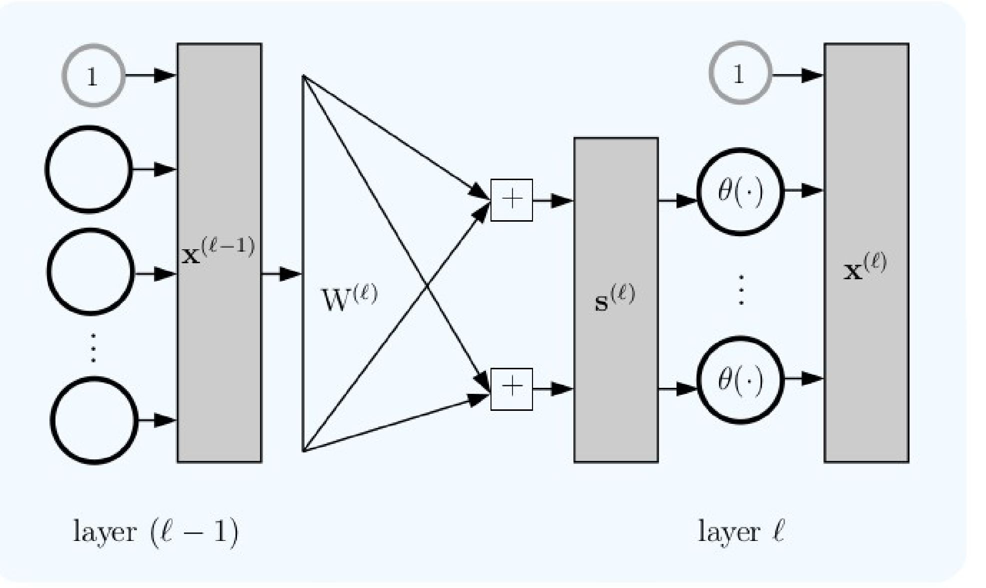
$$\frac{\partial E_{\text{in}}(\mathbf{w})}{\partial \mathrm{W}^{(\ell)}} = \frac{1}{N}\sum_{n=1}^{N}\frac{\partial \mathrm{e}_n}{\partial \mathrm{W}^{(\ell)}}$$

Necesitamos:

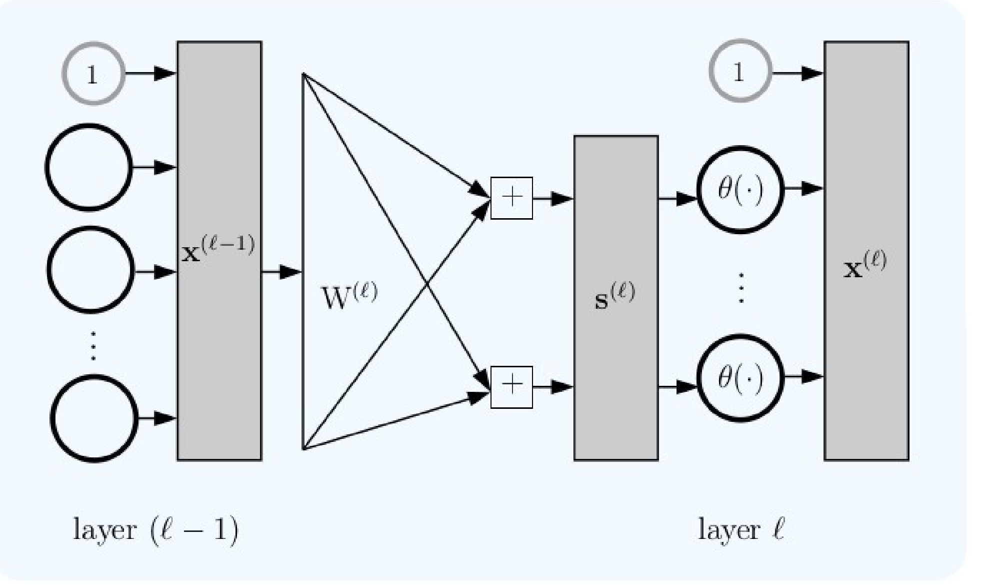
$$\frac{\partial \mathrm{e}(\mathbf{x})}{\partial \mathrm{W}^{(\ell)}}$$

Backpropagation

Vamos a usar la **regla de la cadena** para expresar las derivadas parciales de la capa $(\ell-1)$ en función de las derivadas parciales de la capa ℓ.



Backpropagation



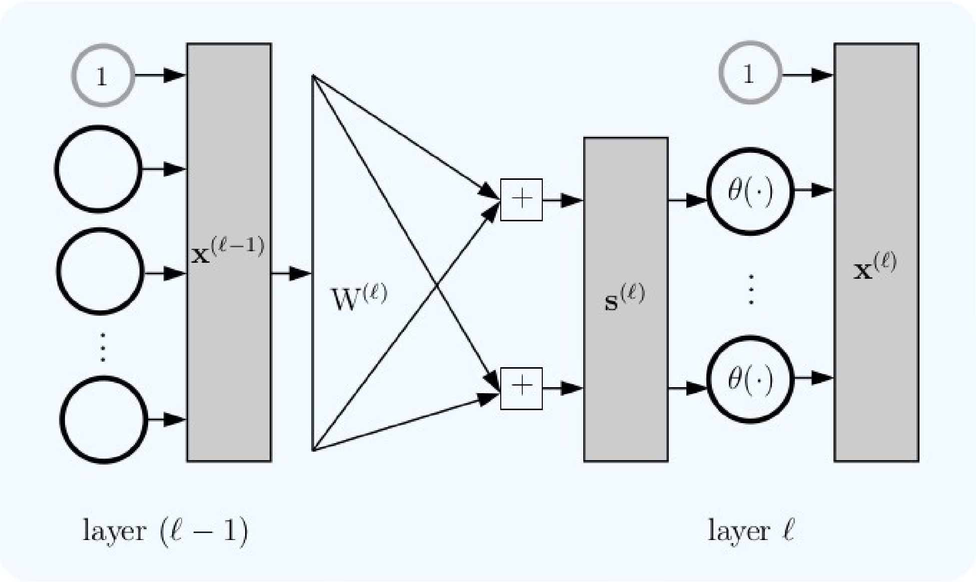
Vamos a usar la **regla de la cadena** para expresar las derivadas parciales de la capa $(\ell-1)$ en función de las derivadas parciales de la capa ℓ .

Tenemos: $\mathbf{s}^{(\ell)} = (\mathrm{W}^{(\ell)})^{\mathrm{T}}\mathbf{x}^{(\ell-1)}$

Definimos la sensibilidad de la capa ℓ :

$$\boldsymbol{\delta}^{(\ell)} = \frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}$$

Backpropagation



Vamos a usar la **regla de la cadena** para expresar las derivadas parciales de la capa $(\ell-1)$ en función de las derivadas parciales de la capa ℓ .

Tenemos: $\mathbf{s}^{(\ell)} = (\mathrm{W}^{(\ell)})^{\mathrm{T}}\mathbf{x}^{(\ell-1)}$

Definimos la sensibilidad de la capa ℓ :

$$\boldsymbol{\delta}^{(\ell)} = \frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}$$

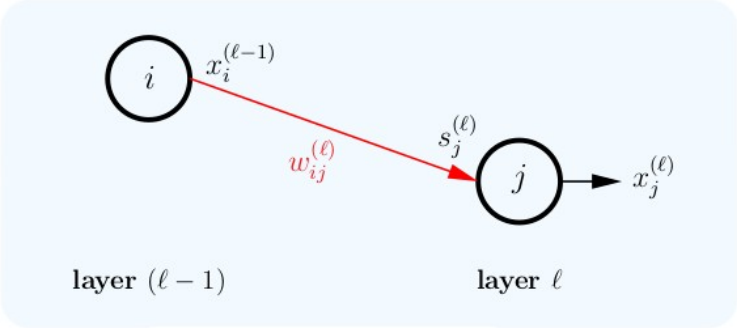
Aplicando la regla de la cadena:

$$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

?

Backpropagation



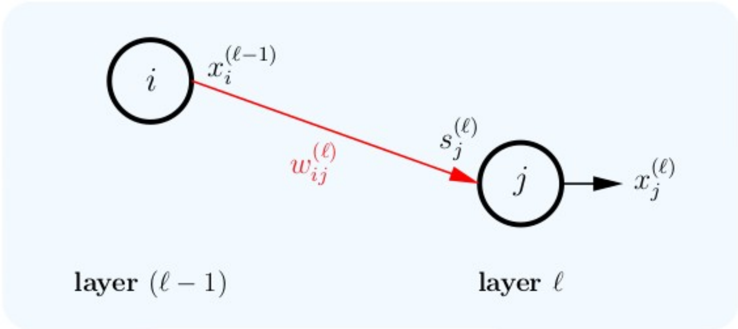
Miremos esto:

$$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$

a nivel de un enlace.

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

Backpropagation



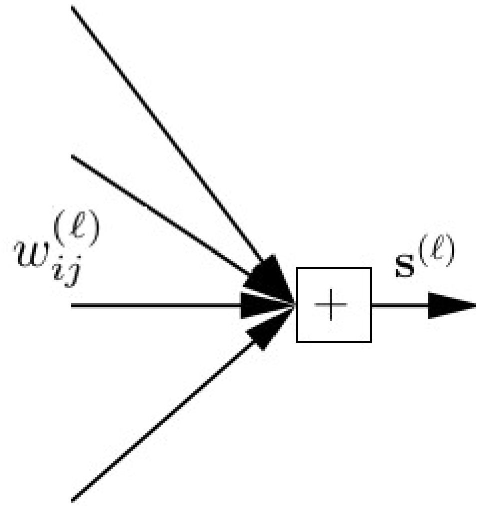
Miremos esto: $$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$ a nivel de un enlace.

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

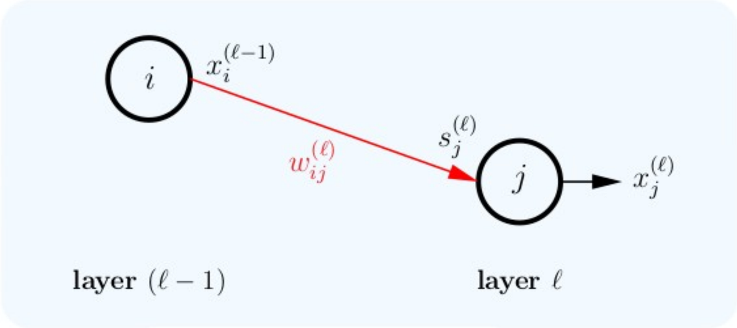
Tenemos:

$$\frac{\partial \mathrm{e}}{\partial\, w_{ij}^{(\ell)}} = \frac{\partial \mathbf{s}_j^{(\ell)}}{\partial w_{ij}^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$$

y sabemos que: $\mathbf{s}_j^{(\ell)} = \sum_{\alpha=0}^{d^{(\ell-1)}} w_{\alpha j}^{(\ell)} \mathbf{x}_\alpha^{(\ell-1)}$

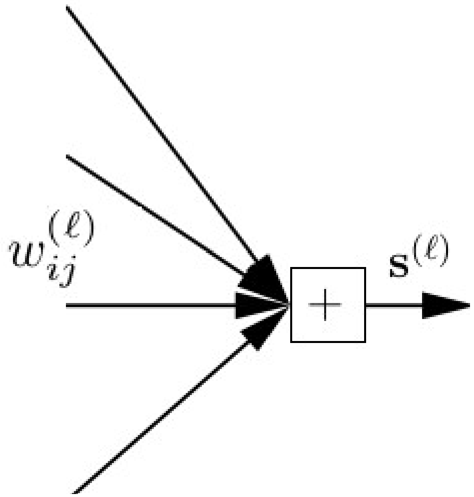


Backpropagation



Miremos esto: $$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$ a nivel de un enlace.

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

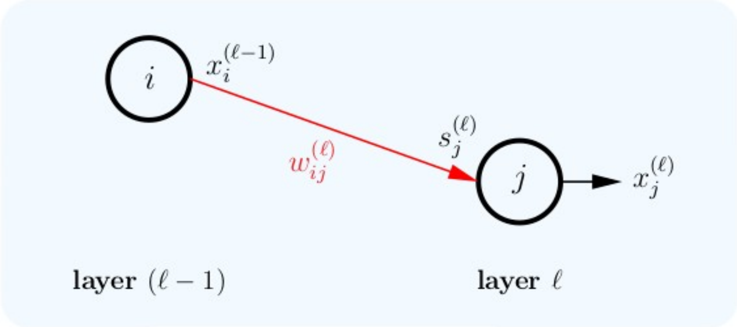


Tenemos: $$\frac{\partial \mathrm{e}}{\partial\, w_{ij}^{(\ell)}} = \frac{\partial \mathbf{s}_j^{(\ell)}}{\partial w_{ij}^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$$

y sabemos que: $\mathbf{s}_j^{(\ell)} = \sum_{\alpha=0}^{d^{(\ell-1)}} w_{\alpha j}^{(\ell)} \mathbf{x}_\alpha^{(\ell-1)}$

por lo que al derivar con respecto a $w_{ij}^{(\ell)}$, queda: $\mathbf{x}_i^{(\ell-1)} \cdot \boldsymbol{\delta}_j^{(\ell)}$

Backpropagation

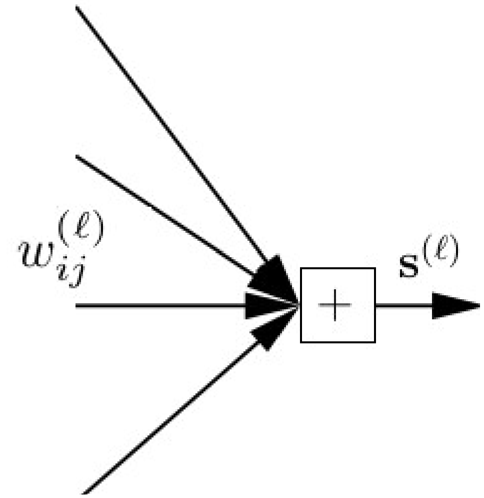


Miremos esto: $$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$ a nivel de un enlace.

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

Tenemos:

$$\frac{\partial \mathrm{e}}{\partial\, w_{ij}^{(\ell)}} = \frac{\partial \mathbf{s}_j^{(\ell)}}{\partial w_{ij}^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$$



y sabemos que: $\mathbf{s}_j^{(\ell)} = \sum_{\alpha=0}^{d^{(\ell-1)}} w_{\alpha j}^{(\ell)} \mathbf{x}_\alpha^{(\ell-1)}$

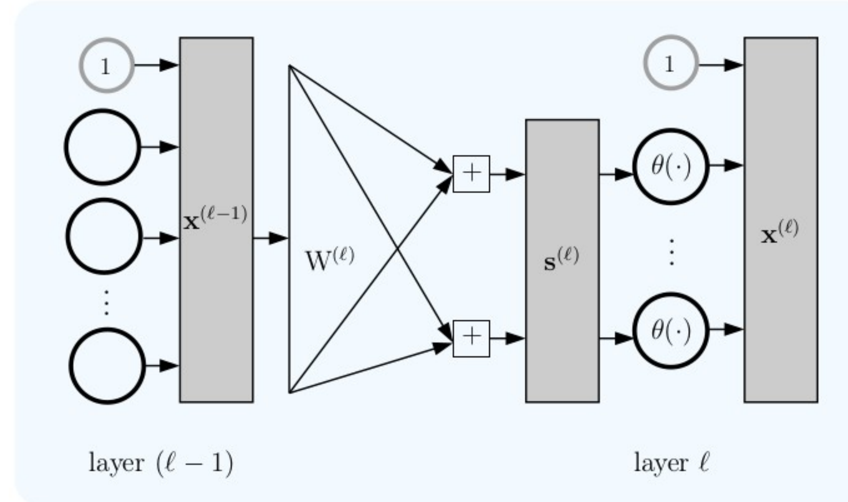
por lo que al derivar con respecto a $w_{ij}^{(\ell)}$, queda: $\mathbf{x}_i^{(\ell-1)} \cdot \boldsymbol{\delta}_j^{(\ell)}$

Luego, haciendo lo mismo para cada parámetro, encontramos que:

$$\frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} = \mathbf{x}^{(\ell-1)} \quad \checkmark$$

Backpropagation

Ahora trabajaremos con: $$\delta_j^{(\ell)} = \frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$$



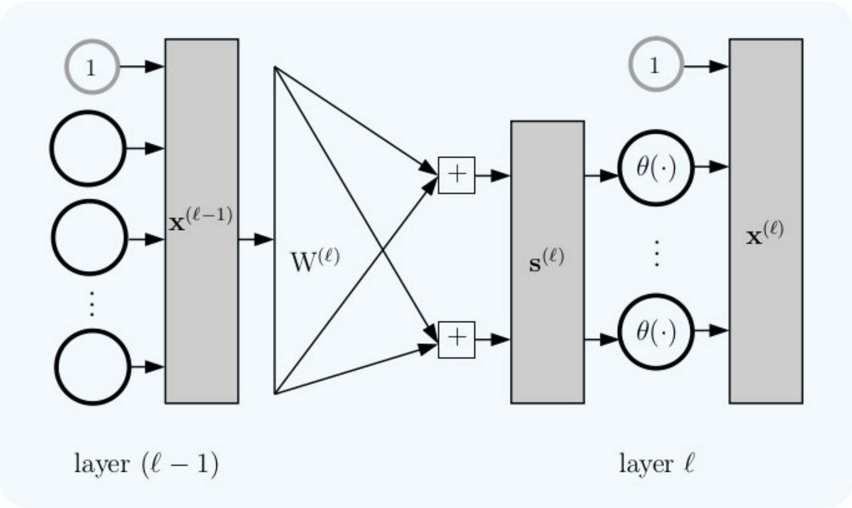
$$\frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} = \mathbf{x}^{(\ell-1)} \quad \checkmark$$

Backpropagation

Ahora trabajaremos con: $$\boldsymbol{\delta}_j^{(\ell)} = \frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$$

Aplicamos regla de la cadena:

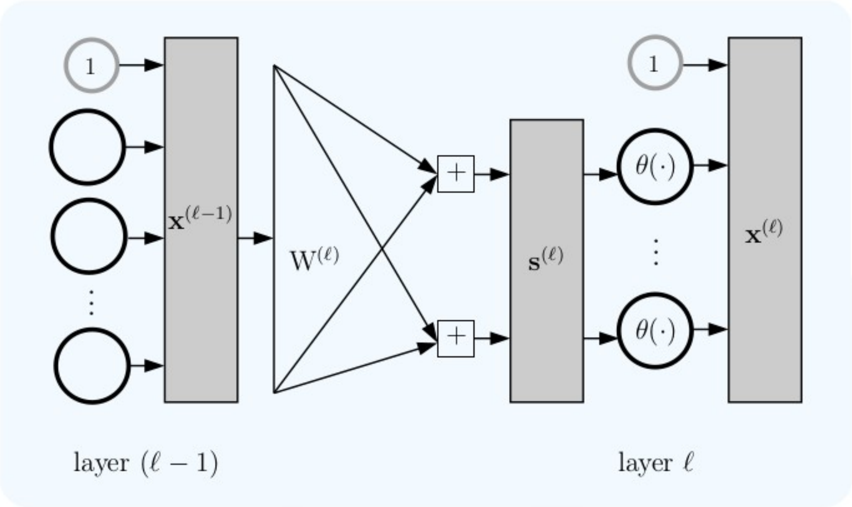
$$\frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}} = \frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathbf{x}_j^{(\ell)}}{\partial \mathbf{s}_j^{(\ell)}}$$



$$\frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} = \mathbf{x}^{(\ell-1)} \quad \checkmark$$

Backpropagation

Ahora trabajaremos con: $\boldsymbol{\delta}_j^{(\ell)} = \dfrac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}}$



$$\frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} = \mathbf{x}^{(\ell-1)} \quad \checkmark$$

Aplicamos regla de la cadena:

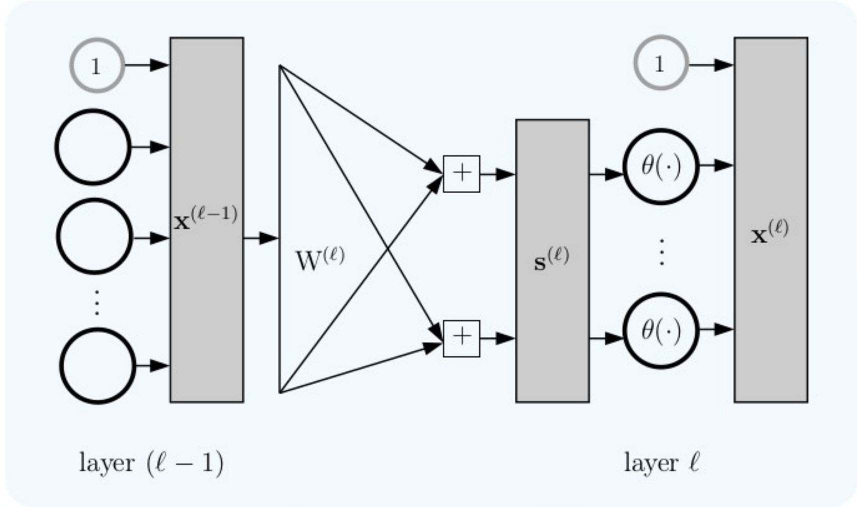
$$\frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}} = \frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathbf{x}_j^{(\ell)}}{\partial \mathbf{s}_j^{(\ell)}}$$

$$= \frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \theta'\left(\mathbf{s}_j^{(\ell)}\right)$$

Derivada de la función de activación

Backpropagation

Ahora trabajaremos con: $$\delta_j^{(\ell)} = \frac{\partial e}{\partial \mathbf{s}_j^{(\ell)}}$$



$$\frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} = \mathbf{x}^{(\ell-1)} \checkmark$$

Aplicamos regla de la cadena:

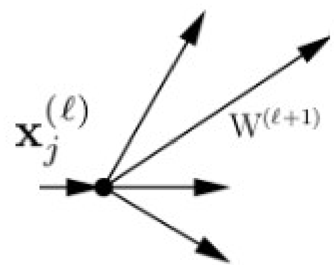
$$\frac{\partial e}{\partial \mathbf{s}_j^{(\ell)}} = \frac{\partial e}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathbf{x}_j^{(\ell)}}{\partial \mathbf{s}_j^{(\ell)}}$$

$$= \frac{\partial e}{\partial \mathbf{x}_j^{(\ell)}} \cdot \theta'\left(\mathbf{s}_j^{(\ell)}\right)$$

Derivada de la función de activación

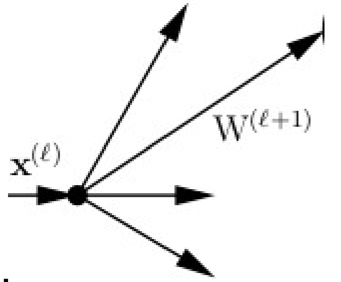
Ahora veamos que ocurre en:

$$\frac{\partial e}{\partial \mathbf{x}_j^{(\ell)}}$$



Hay una multiplexión

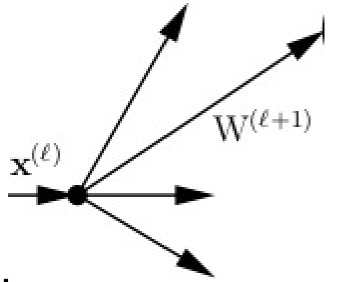
Backpropagation



Dado que una componente de $\mathbf{x}^{(\ell)}$ afecta a todas las componentes de $\mathbf{s}^{(\ell+1)}$, necesitamos sumar estas dependencias:

$$\frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} = \sum_{k=1}^{d^{(\ell+1)}} \frac{\partial \mathbf{s}_k^{(\ell+1)}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_k^{(\ell+1)}}$$

Backpropagation



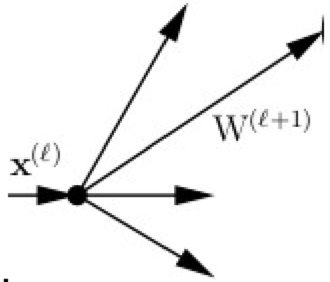
Dado que una componente de $\mathbf{x}^{(\ell)}$ afecta a todas las componentes de $\mathbf{s}^{(\ell+1)}$, necesitamos sumar estas dependencias:

$$\frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} = \sum_{k=1}^{d^{(\ell+1)}} \frac{\partial \mathbf{s}_k^{(\ell+1)}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_k^{(\ell+1)}}$$

$$\frac{\partial\ \cancel{\mathbf{x}_j^{(\ell)}} \cdot w_{jk}^{(\ell+1)}}{\partial \mathbf{x}_j^{(\ell)}}$$

$$= \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}.$$

Backpropagation



Dado que una componente de $\mathbf{x}^{(\ell)}$ afecta a todas las componentes de $\mathbf{s}^{(\ell+1)}$, necesitamos sumar estas dependencias:

$$\frac{\partial\, \mathbf{x}_j^{(\ell)} \cdot w_{jk}^{(\ell+1)}}{\partial \mathbf{x}_j^{(\ell)}}$$

$$\frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} = \sum_{k=1}^{d^{(\ell+1)}} \frac{\partial \mathbf{s}_k^{(\ell+1)}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathrm{e}}{\partial \mathbf{s}_k^{(\ell+1)}}$$

$$= \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}.$$

Luego:

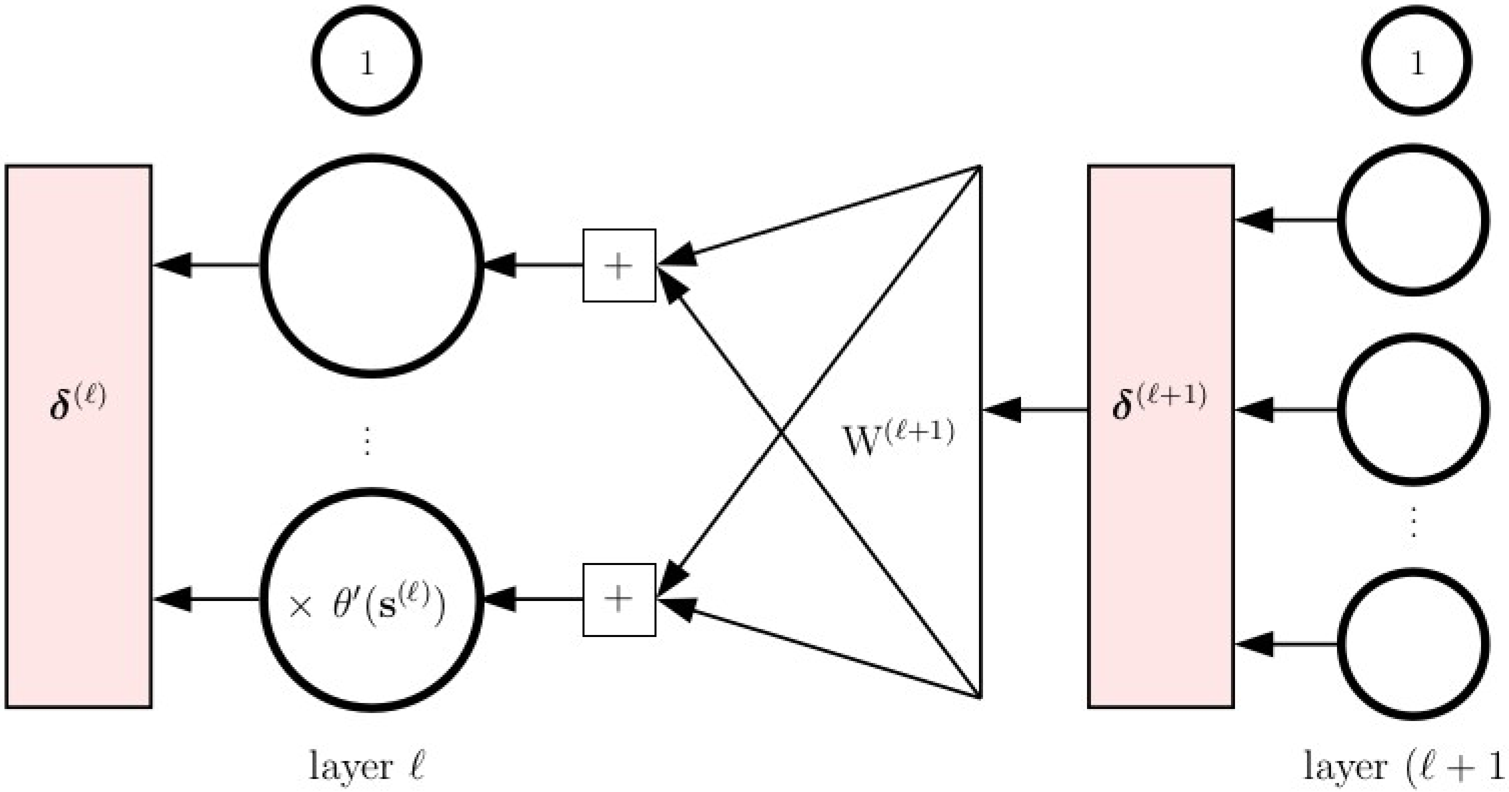
$$\frac{\partial \mathrm{e}}{\partial \mathbf{s}_j^{(\ell)}} = \frac{\partial \mathrm{e}}{\partial \mathbf{x}_j^{(\ell)}} \cdot \frac{\partial \mathbf{x}_j^{(\ell)}}{\partial \mathbf{s}_j^{(\ell)}}$$

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$

Backpropagation

$$\tanh'(\mathbf{s}^{(\ell)}) = \mathbf{1} - \tanh^2(\mathbf{s}^{(\ell)})$$

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$



$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \cdots \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Backpropagation

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$

Backpropagation de sensibilidad:

$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \cdots \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Nos falta calcular $\boldsymbol{\delta}^{(L)}$.

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$

Backpropagation

Backpropagation de sensibilidad:

$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \cdots \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Nos falta calcular $\boldsymbol{\delta}^{(L)}$.

Sabemos que: $\mathrm{e} = (\mathbf{x}^{(L)} - y)^2 = (\theta(\mathbf{s}^{(L)}) - y)^2$

Backpropagation

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$

Backpropagation de sensibilidad:

$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \cdots \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Nos falta calcular $\boldsymbol{\delta}^{(L)}$.

Sabemos que: $\mathrm{e} = \underline{(\mathbf{x}^{(L)} - y)^2} = (\theta(\mathbf{s}^{(L)}) - y)^2$

Luego

$$\begin{aligned} \boldsymbol{\delta}^{(L)} &= \frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(L)}} \\ &= \frac{\partial}{\partial \mathbf{s}^{(L)}} (\mathbf{x}^{(L)} - y)^2 \end{aligned}$$

Backpropagation

$$\boldsymbol{\delta}_j^{(\ell)} = \theta'(\mathbf{s}_j^{(\ell)}) \sum_{k=1}^{d^{(\ell+1)}} w_{jk}^{(\ell+1)} \boldsymbol{\delta}_k^{(\ell+1)}$$

Backpropagation de sensibilidad:

$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \cdots \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Nos falta calcular $\boldsymbol{\delta}^{(L)}$.

Sabemos que: $\mathrm{e} = \underline{(\mathbf{x}^{(L)} - y)}^2 = (\theta(\underline{\mathbf{s}^{(L)})} - y)^2$

Luego

$$\begin{aligned}\boldsymbol{\delta}^{(L)} &= \frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(L)}} \\ &= \frac{\partial}{\partial \mathbf{s}^{(L)}}(\mathbf{x}^{(L)} - y)^2 \\ &= 2(\mathbf{x}^{(L)} - y)\frac{\partial \mathbf{x}^{(L)}}{\partial \mathbf{s}^{(L)}} \\ &= 2(\mathbf{x}^{(L)} - y)\underline{\theta'(\mathbf{s}^{(L)})}.\end{aligned}$$

$$\begin{aligned}\tanh'(\mathbf{s}^{(\ell)}) &= \mathbf{1} - \tanh^2(\mathbf{s}^{(\ell)}) \\ &= 1 - (x^{(L)})^2\end{aligned}$$

Backpropagation

Backpropagation

$\delta^{(L)} \leftarrow 2(x^{(L)} - y) \cdot \theta'(s^{(L)})$

for $\ell = L - 1$ to 1 **do**

  Compute $\theta'(\mathbf{s}^{(\ell)}) = \left[1 - \mathbf{x}^{(\ell)} \otimes \mathbf{x}^{(\ell)}\right]_1^{d^{(\ell)}}$

  $\boldsymbol{\delta}^{(\ell)} \leftarrow \theta'(\mathbf{s}^{(\ell)}) \otimes \left[\mathrm{W}^{(\ell+1)}\boldsymbol{\delta}^{(\ell+1)}\right]_1^{d^{(\ell)}}$

end for

Backpropagation nos permite obtener la cadena de sensibilidades:

$$\boldsymbol{\delta}^{(1)} \longleftarrow \boldsymbol{\delta}^{(2)} \ \ldots \ \longleftarrow \boldsymbol{\delta}^{(L-1)} \longleftarrow \boldsymbol{\delta}^{(L)}$$

Backpropagation

Recordar que: $$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$

Luego, podemos calcular los gradientes para aplicar GD:

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

Algorithm to Compute $E_{\text{in}}(\mathbf{w})$ and $\mathbf{g} = \nabla E_{\text{in}}(\mathbf{w})$:
Input: weights $\mathbf{w} = \{\mathrm{W}^{(1)}, \ldots, \mathrm{W}^{(L)}\}$; data $\mathcal{D}$.
Output: error $E_{\text{in}}(\mathbf{w})$ and gradient $\mathbf{g} = \{\mathrm{G}^{(1)}, \ldots, \mathrm{G}^{(L)}\}$.

Initialize: $E_{\text{in}} = 0$; for $\ell = 1, \ldots, L$, $\mathrm{G}^{(\ell)} = 0 \cdot \mathrm{W}^{(\ell)}$.
for Each data point $\mathbf{x}_n$ $(n = 1, \ldots, N)$ **do**
    Compute $\mathbf{x}^{(\ell)}$ for $\ell = 0, \ldots, L$. [forward propagation]
    Compute $\boldsymbol{\delta}^{(\ell)}$ for $\ell = 1, \ldots, L$. [backpropagation]
    for $\ell = 1, \ldots, L$ **do**
        $\mathrm{G}^{(\ell)}(\mathbf{x}_n) = [\mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}]$
        $\mathrm{G}^{(\ell)} \leftarrow \mathrm{G}^{(\ell)} + \frac{1}{N}\mathrm{G}^{(\ell)}(\mathbf{x}_n)$.
    end for
end for

Backpropagation

Recordar que: $$\frac{\partial \mathrm{e}}{\partial \mathrm{W}^{(\ell)}} = \frac{\partial \mathbf{s}^{(\ell)}}{\partial \mathrm{W}^{(\ell)}} \cdot \left(\frac{\partial \mathrm{e}}{\partial \mathbf{s}^{(\ell)}}\right)^{\mathrm{T}}$$

$$= \mathbf{x}^{(\ell-1)}(\boldsymbol{\delta}^{(\ell)})^{\mathrm{T}}$$

Luego, podemos calcular los gradientes para aplicar GD:

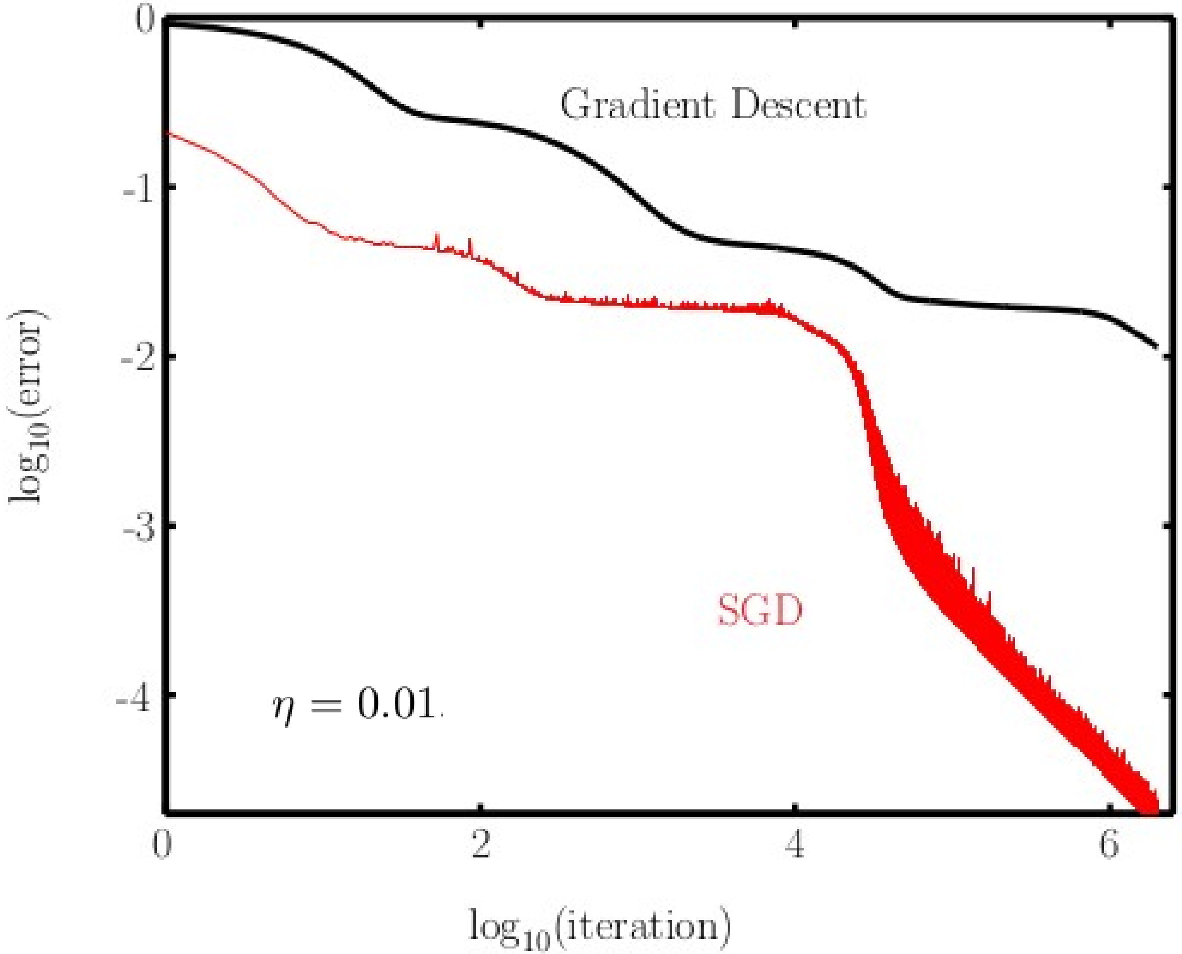
Algorithm to Compute $E_{\text{in}}(\mathbf{w})$ and $\mathbf{g} = \nabla E_{\text{in}}(\mathbf{w})$:
Input: weights $\mathbf{w} = \{\mathrm{W}^{(1)}, \ldots, \mathrm{W}^{(L)}\}$; data $\mathcal{D}$.
Output: error $E_{\text{in}}(\mathbf{w})$ and gradient $\mathbf{g} = \{\mathrm{G}^{(1)}, \ldots, \mathrm{G}^{(L)}\}$.

```
Initialize: E_in = 0; for ℓ = 1, ..., L, G^(ℓ) = 0 · W^(ℓ) .
for Each data point x_n (n = 1, ..., N) do
    Compute x^(ℓ) for ℓ = 0, ..., L.    [forward propagation]
    Compute δ^(ℓ) for ℓ = 1, ..., L.    [backpropagation]
    for ℓ = 1, ..., L do
        G^(ℓ)(x_n) = [x^(ℓ-1)(δ^(ℓ))^T]
        G^(ℓ) ← G^(ℓ) + (1/N) G^(ℓ)(x_n).
    end for
end for
```

$$E_{\text{in}} \leftarrow E_{\text{in}} + \tfrac{1}{N}(\mathbf{x}_n^{(L)} - y_n)^2.$$

Backpropagation

GD para redes feed-forward: $\mathrm{W}^{(\ell)} = \mathrm{W}^{(\ell)} - \eta \mathrm{G}^{(\ell)}(\mathbf{x}_n).$



Digits