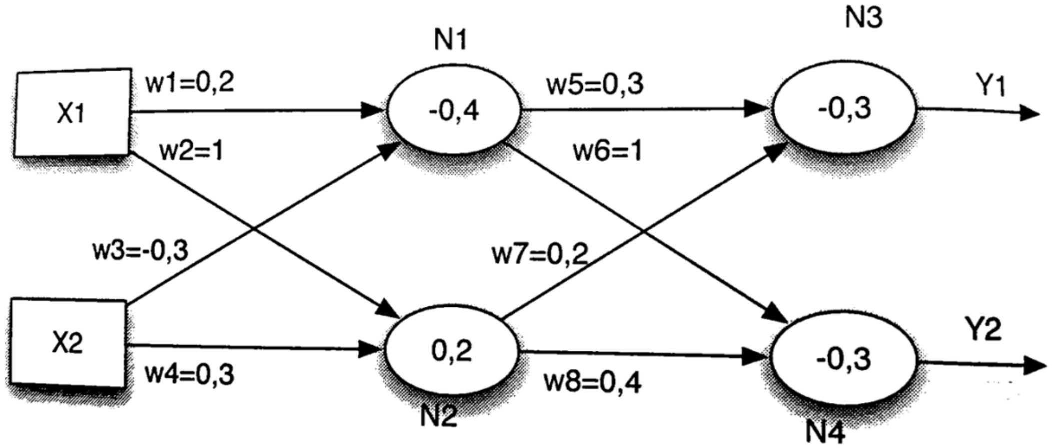
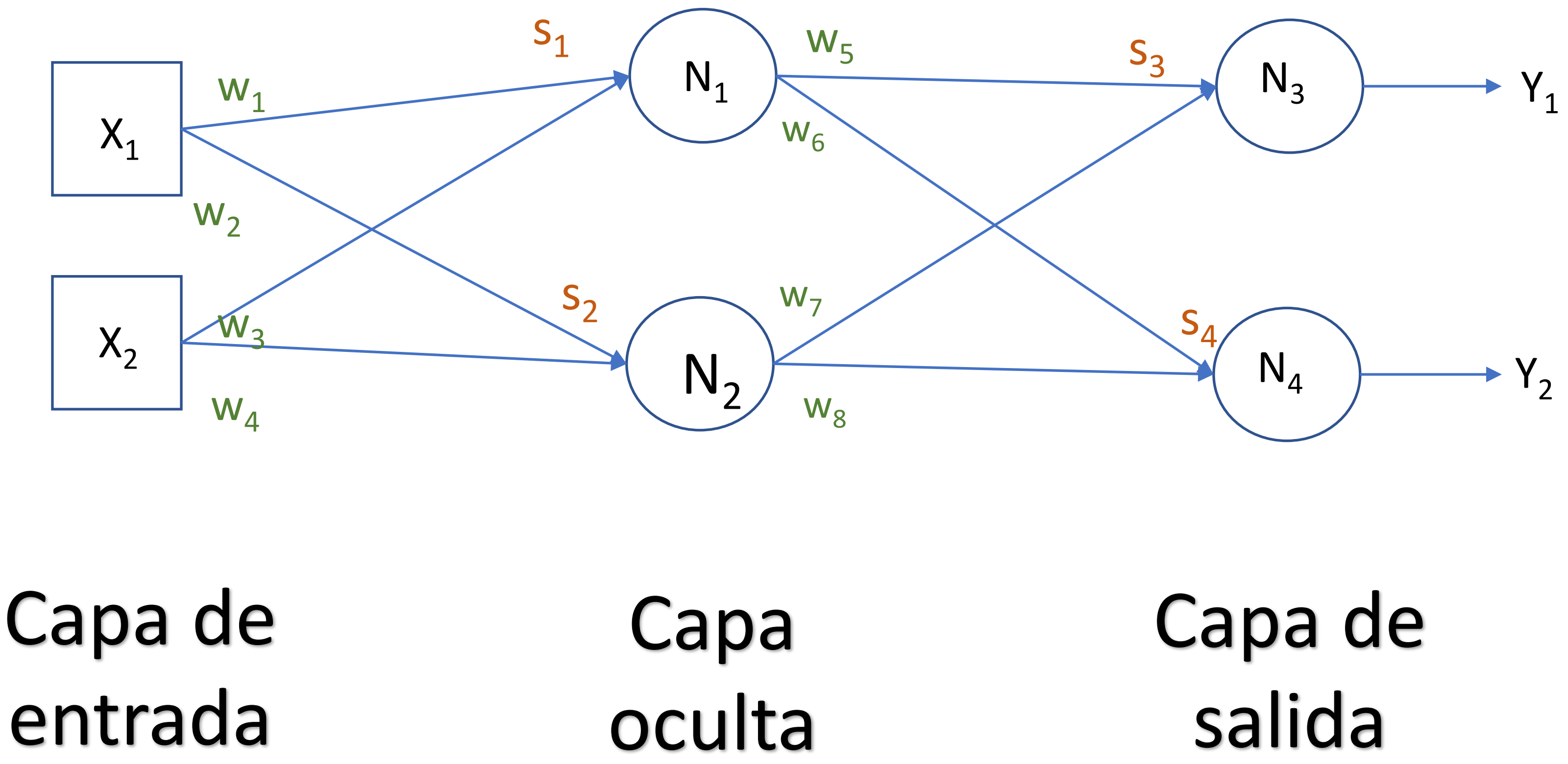
Ejercicio redes neuronales artificiales



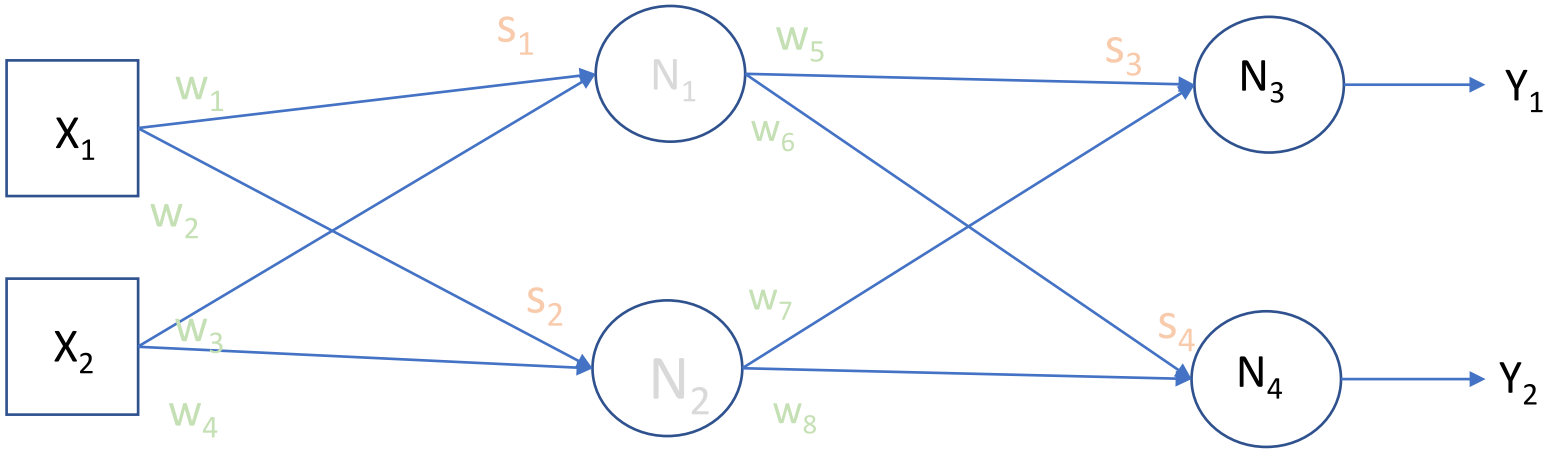
Considere la red neuronal aquí mostrada. Las neuronas N1 a N4 son neuronas del tipo Perceptrón, con entradas y salidas binarias y una función de activación umbral.
Los pesos de las conexiones son los que se observan en el dibujo. Los sesgos de las neuronas corresponden a los valores anotados en cada neurona.

Suponiendo el siguiente ejemplo de entrenamiento <(x1=0, x2=1), (y1=0, y2=0)>, describa cómo deben ajustarse los pesos de la red en el proceso de aprendizaje, suponiendo que α=0,2.

Red neuronal

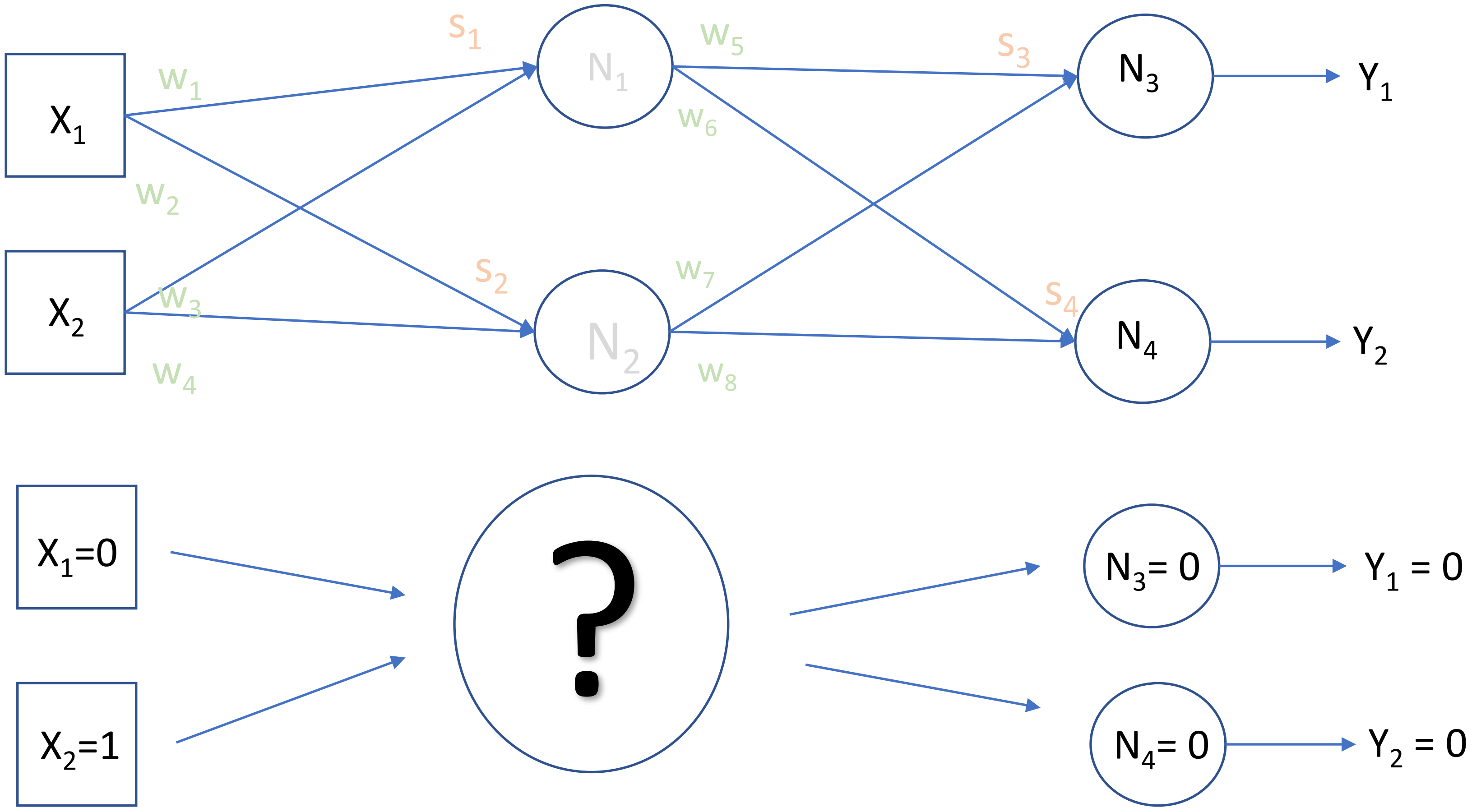


¿Qué tenemos?



Las salidas que se obtienen con unas entradas concretas.

¿Qué buscamos?

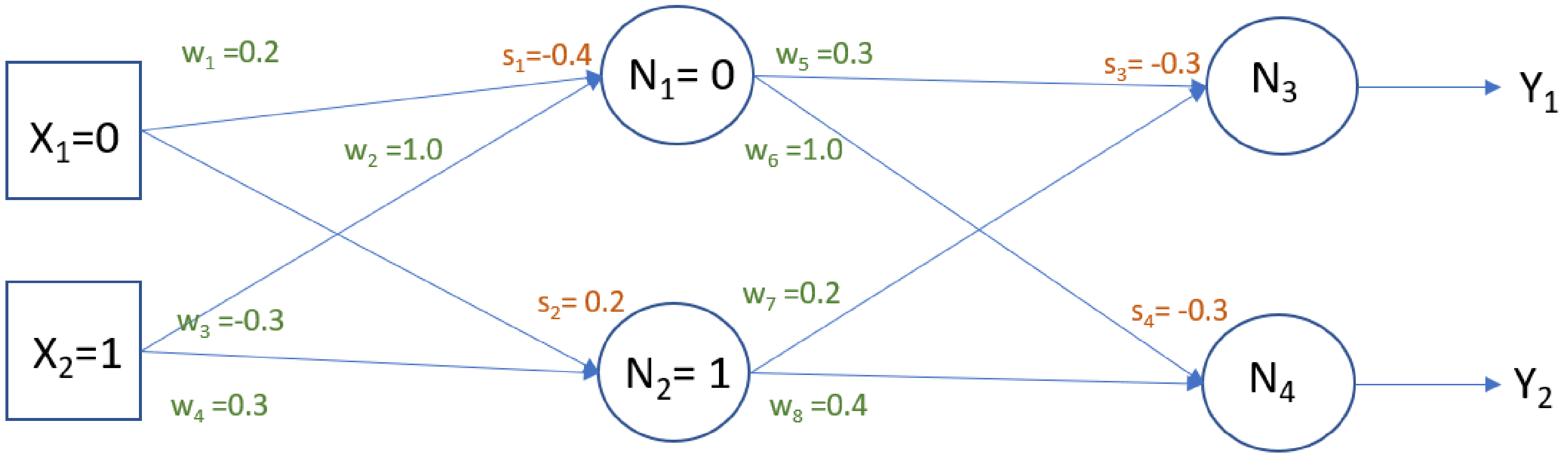


Obtener una configuración de la red (valores de pesos y sesgos), de forma que unas entradas dadas generen unas salidas esperadas.

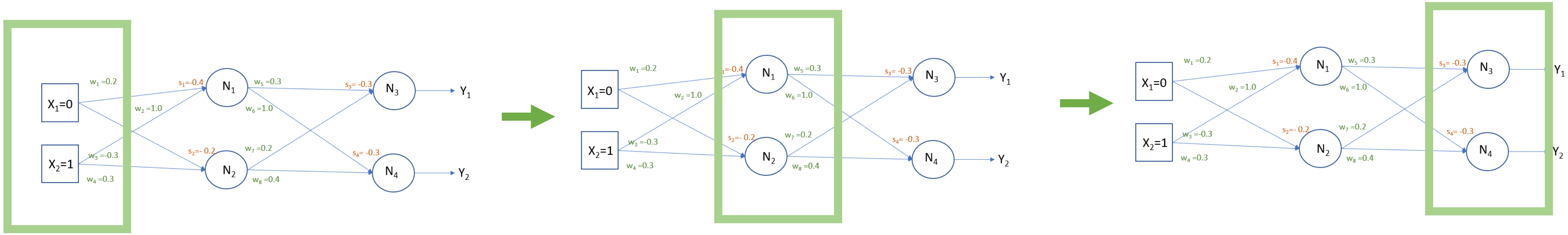
¿Cómo lo conseguimos? (parte1)

Pasos:

1- Inicialización: Asignar configuración inicial de la red, es decir, dar valores aleatorios a los pesos (w) y sesgos (s).



2- Calcular las salidas con las entradas que tenemos:

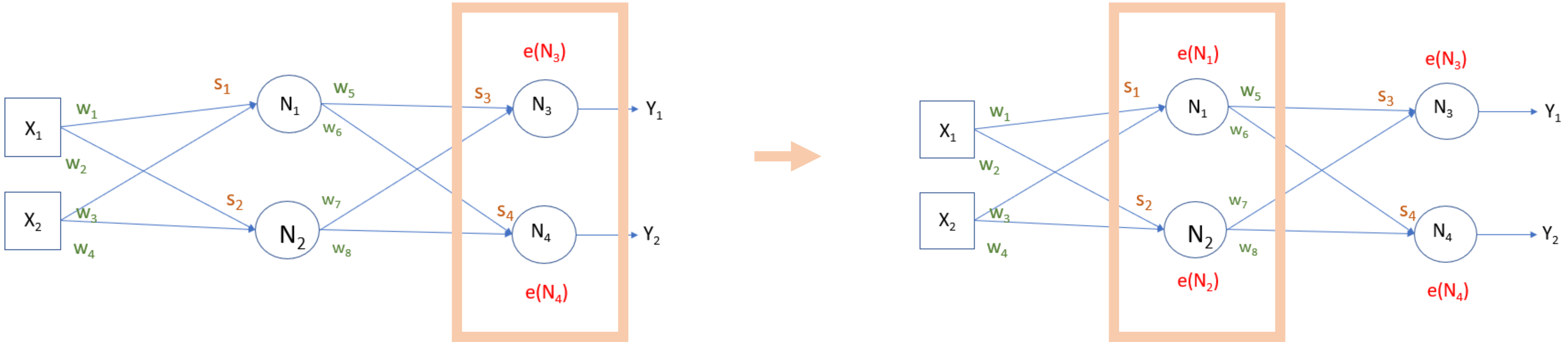


¿Cómo lo conseguimos? (parte2)

3- Backpropagation:

- Calcular el error: comparar los valores de salida obtenidos con los esperados. ERROR = Y esperado – Y obtenido

- Propagar hacia atrás el error desde el final: ¿cuánto contribuye cada neurona a producir el error final?



4- Reconfigurar la red: actualizar los valores de los pesos (w) y sesgos (s) en base a los errores obtenidos

$$w_i = w_i + \Delta w_i = w_i + \alpha \cdot e(N) \cdot x$$

$$s_i = s_i + \Delta s_i = s_i + \alpha \cdot e(N)$$

α = constante de aprendizaje

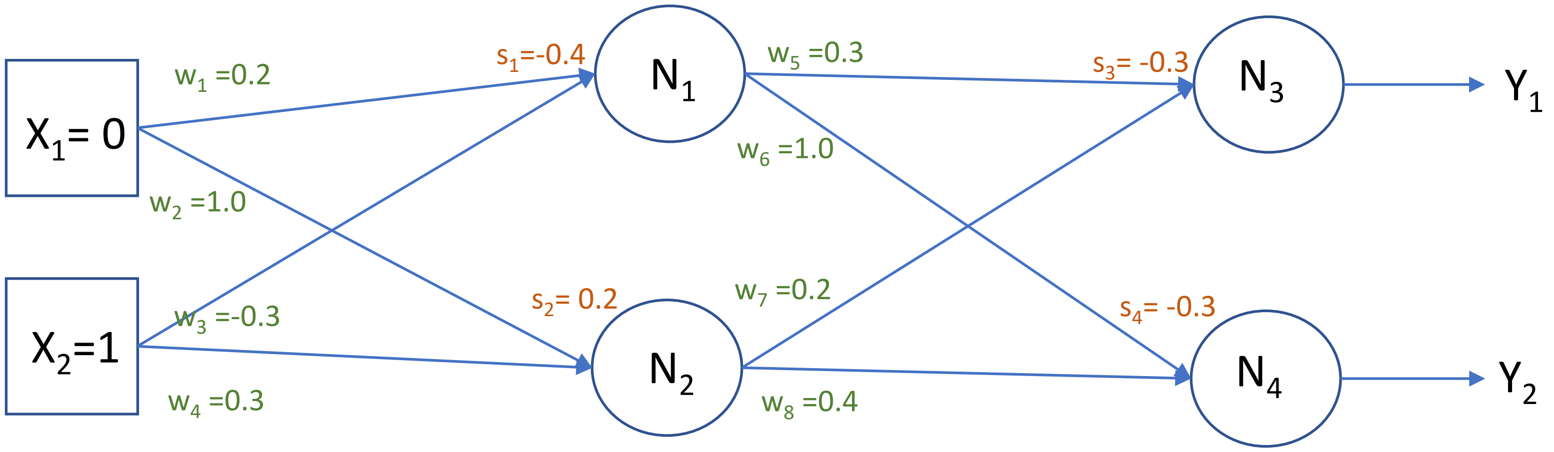
5- Repetir desde paso 2 hasta conseguir un objetivo:

- Realizar un número determinado de iteraciones
- Conseguir un error final deseado
- Lo primero que ocurra de lo anterior

Paso 1: Inicialización de la red

$$N_1=f(\ x_1\ w_1 + x_2\ w_3 + s_1\ )$$

$$N_3=f(\ N_1\ w_5 + N_2\ w_7 + s_3\ )$$



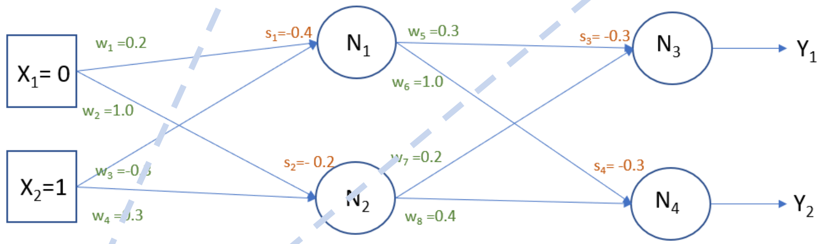
$$N_2=f(\ x_1\ w_2 + x_2\ w_4 + s_2\ )$$

$$N_4=f(\ N_1\ w_6 + N_2\ w_8 + s_4\ )$$

Paso 2: Calcular salidas

$N_1=f(\,x_1\,w_1+x_2\,w_3+s_1\,)$

$N_3=f(\,N_1\,w_5+N_2\,w_7+s_3\,)$

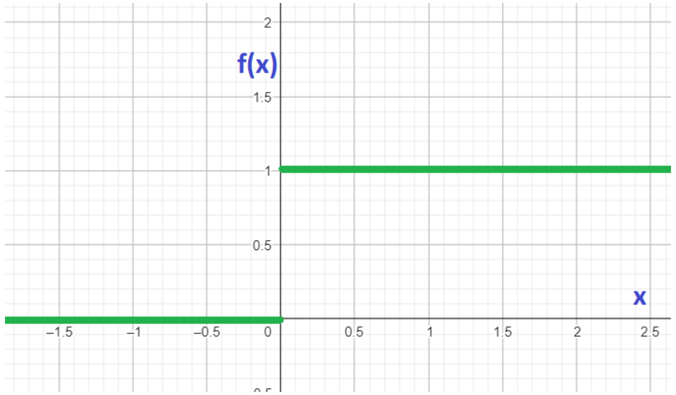


$N_2=f(\,x_1\,w_2+x_2\,w_4+s_2\,)$

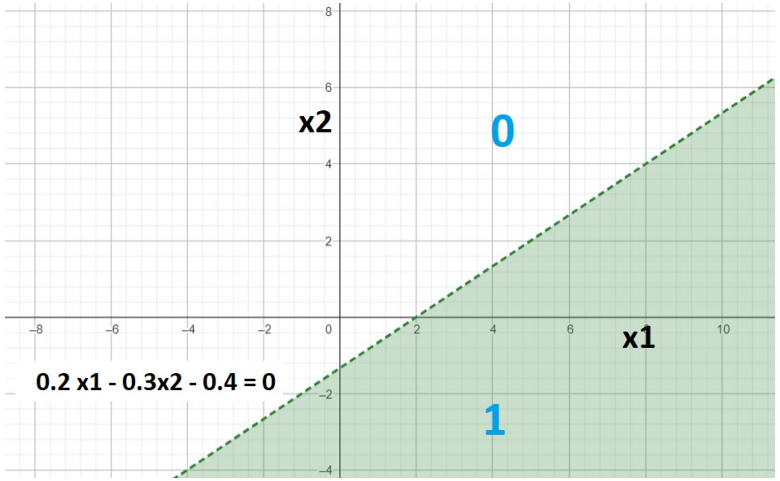
$N_4=f(\,N_1\,w_6+N_2\,w_8+s_4\,)$

Función de activación umbral

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



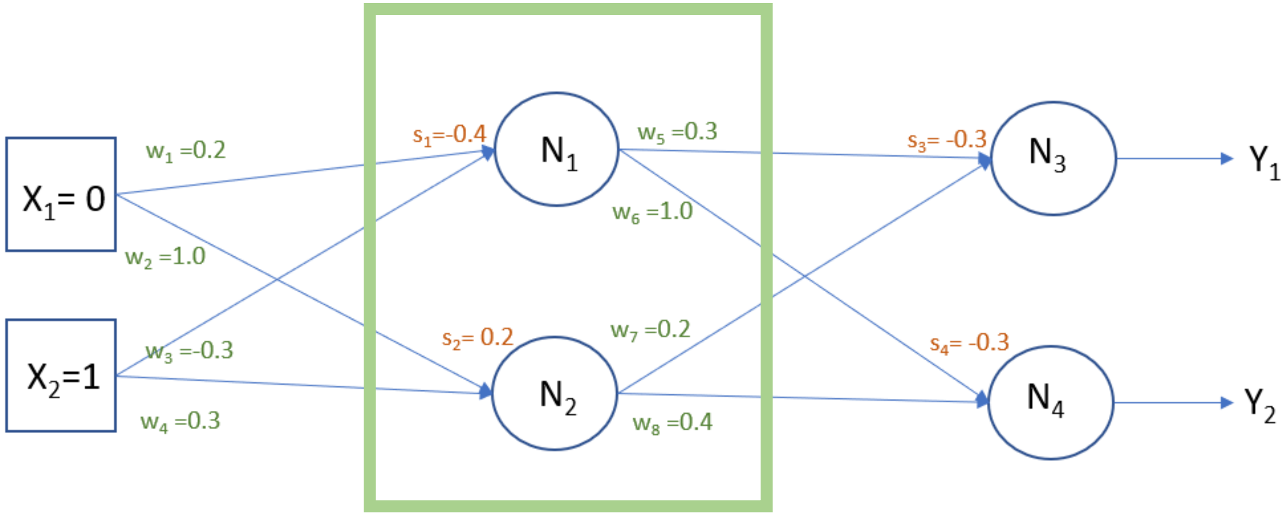
$$N_1=f(w_1\,x_1+w_3\,x_2+s_1)=\begin{cases}0 & \text{si } w_1\,x_1+w_3\,x_2+s_1\le 0\\ 1 & \text{si } w_1\,x_1+w_3\,x_2+s_1>0\end{cases}$$



Paso 2: Calcular salidas

$N_1=f(\, x_1\, w_1 + x_2\, w_3 + s_1\,)$

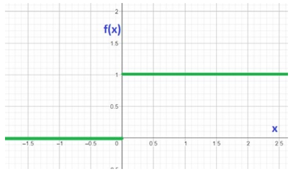
$N_3=f(\, N_1\, w_5 + N_2\, w_7 + s_3\,)$



$N_2=f(\, x_1\, w_2 + x_2\, w_4 + s_2\,)$

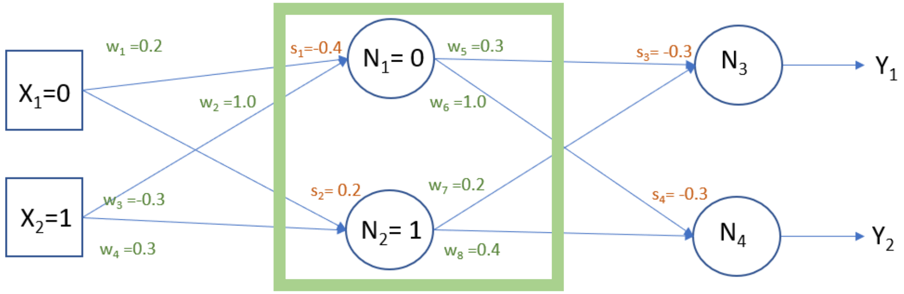
$N_4=f(\, N_1\, w_6 + N_2\, w_8 + s_4\,)$

$$N_1=f(w_1\, x_1 + w_3\, x_2 + s_1)= \begin{cases} 0 \quad \text{si} \quad w_1\, x_1 + w_3\, x_2 + s_1 \leq 0 \\ 1 \quad \text{si} \quad w_1\, x_1 + w_3\, x_2 + s_1 > 0 \end{cases}$$



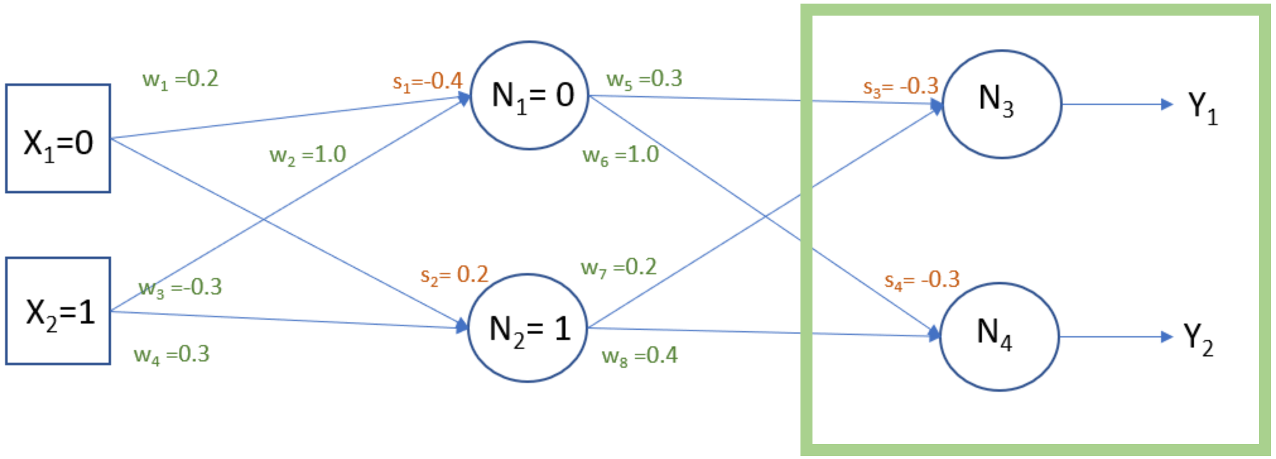
$$N_1=f(0.2\, x_1 - 0.3\, x_2 - 0.4) = f(0.2 \cdot 0 - 0.3 \cdot 1 - 0.4)= f(-0.7)= 0$$

$$N_2=f(1.0\, x_1 + 0.3\, x_2 - 0.2) = f(1.0 \cdot 0 + 0.3 \cdot 1 - 0.2)= f\,(0.1) = 1$$

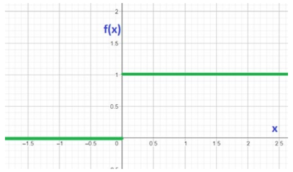


Paso 2: Calcular salidas

Capa de salida

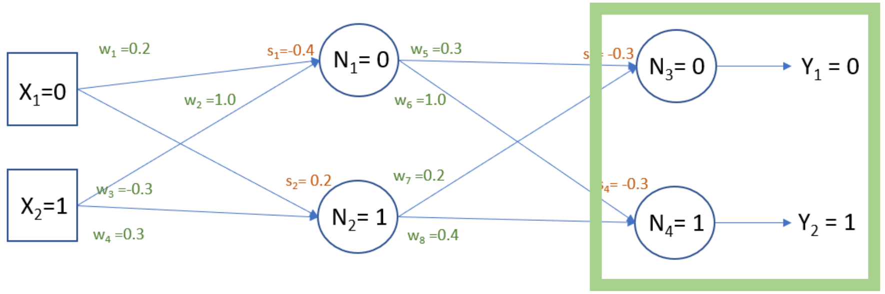


$$N_1=f(w_1\,x_1 + w_3\,x_2 + s_1)=\begin{cases} 0 \quad \text{si} \quad w_1\,x_1 + w_3\,x_2 + s_1 \leq 0 \\ 1 \quad \text{si} \quad w_1\,x_1 + w_3\,x_2 + s_1 > 0 \end{cases}$$



$$Y_1= N_3 =f(0.3\ N_1 + 0.2\ N_2 - 0.3) = f(0.3 \cdot 0 + 0.2 \cdot 1 - 0.3)= f(-0.1) = 0$$

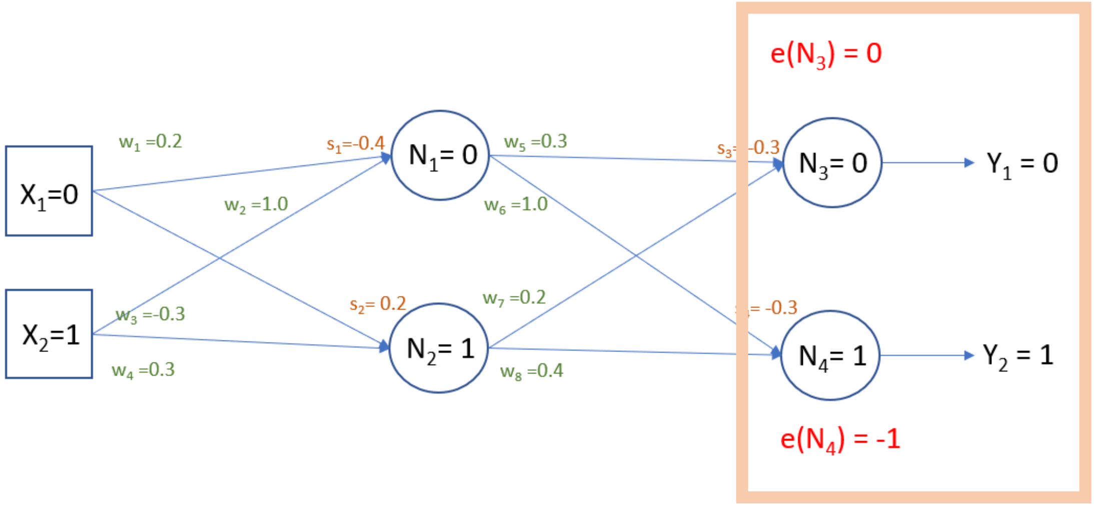
$$Y_2= N_4 =f(1.0\ N_1 + 0.4\ N_2 - 0.3) = f(1.0 \cdot 0 + 0.4 \cdot 1 - 0.3) = f\ (0.1) = 1$$



Paso 3: Cálculo de los errores en cada neurona (Backpropagation)

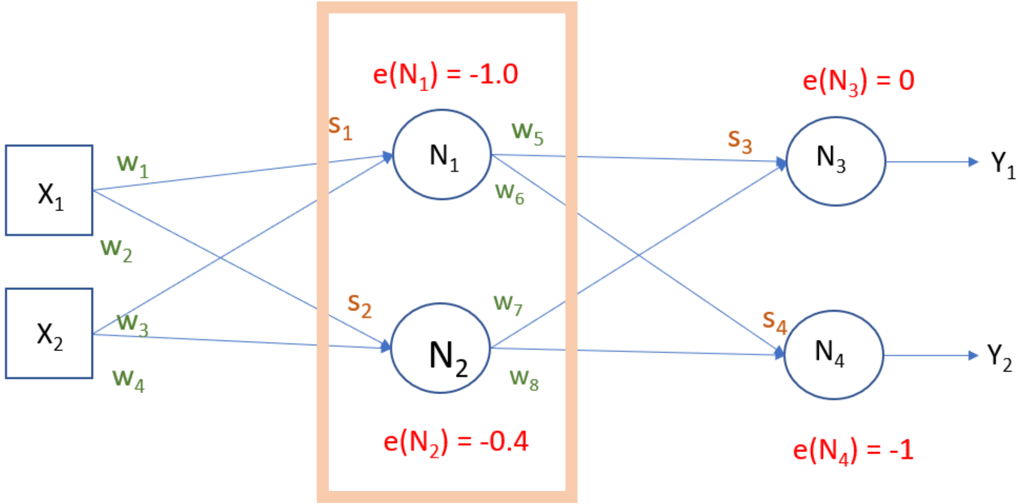
$e(N_3) = Y_1$ esperado – Y_1 obtenido $= 0 - 0 = 0$

$e(N_4) = Y_2$ esperado – Y_2 obtenido $= 0 - 1 = -1$



$$e(N_1) = \sum_i w_k \cdot e(N_i) = w_5 \cdot e(N_3) + w_6 \cdot e(N_4) = 0.3 \cdot 0 + 1.0\,(-1.0) = -1.0$$

$$e(N_2) = \sum_i w_k \cdot e(N_i) = w_7 \cdot e(N_3) + w_8 \cdot e(N_4) = 0.2 \cdot 0 + 0.4(-1.0) = -0.4$$



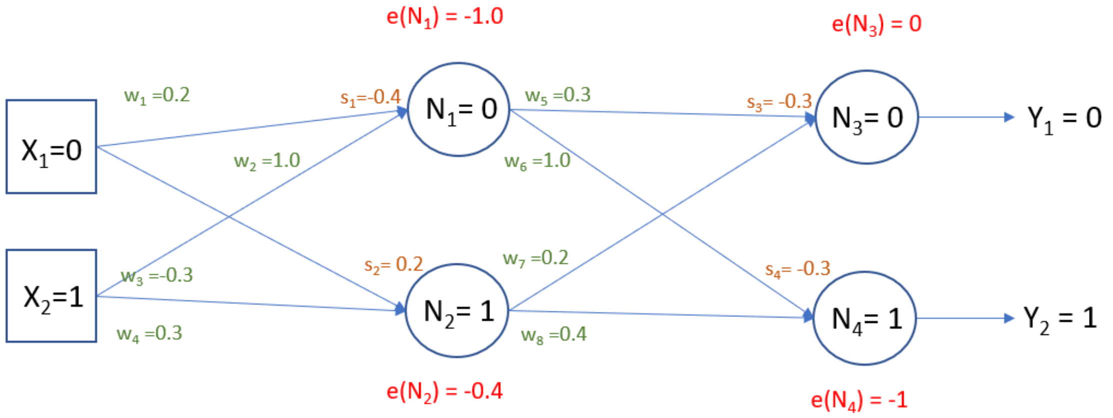
Paso 4: Actualizar los valores de los pesos (w) y sesgos (s)

$w_i = w_i + \Delta w_i = w_i + \alpha \cdot e(N) \cdot x$

$s_i = s_i + \Delta s_i = s_i + \alpha \cdot e(N)$

α = constante de aprendizaje = 0.2

x = entrada de la capa



$w_5 = w_5 + \alpha \cdot e(N_3) \cdot N_1 = 0.3 + 0.2 \cdot (0.0) \cdot 0.0 = 0.3$

$w_6 = w_6 + \alpha \cdot e(N_4) \cdot N_1 = 1.0 + 0.2 \cdot (-1.0) \cdot 0.0 = 1.0$

$w_7 = w_7 + \alpha \cdot e(N_3) \cdot N_2 = 0.2 + 0.2 \cdot (0.0) \cdot 1.0 = 0.2$

$w_8 = w_8 + \alpha \cdot e(N_4) \cdot N_2 = 0.4 + 0.2 \cdot (-1.0) \cdot 1.0 = 0.2$

$s_3 = s_3 + \alpha \cdot e(N_3) = -0.3 + 0.2 \cdot (0.0) = -0.3$

$s_4 = s_4 + \alpha \cdot e(N_4) = -0.3 + 0.2 \cdot (-1.0) = -0.5$

$w_1 = w_1 + \alpha \cdot e(N_1) \cdot x_1 = 0.2 + 0.2 \cdot (-1.0) \cdot 0.0 = 0.2$

$w_2 = w_2 + \alpha \cdot e(N_2) \cdot x_1 = 1.0 + 0.2 \cdot (-0.4) \cdot 0.0 = 1.0$

$w_3 = w_3 + \alpha \cdot e(N_1) \cdot x_2 = -0.3 + 0.2 \cdot (-1.0) \cdot 1.0 = -0.5$

$w_4 = w_4 + \alpha \cdot e(N_2) \cdot x_2 = 0.3 + 0.2 \cdot (-0.4) \cdot 1.0 = 0.22$

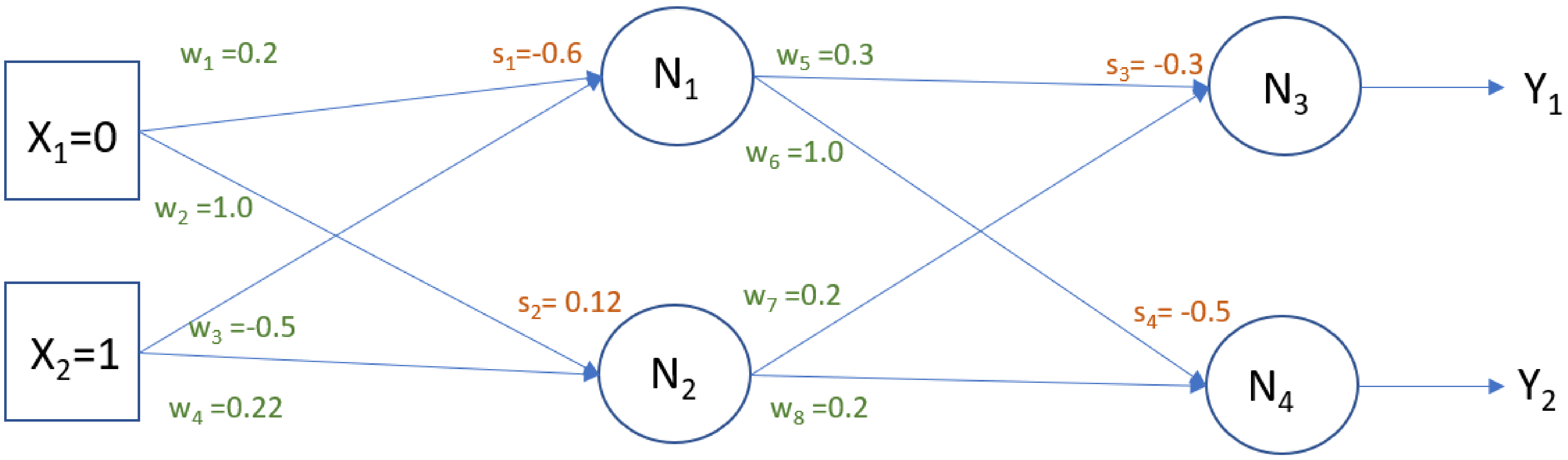
$s_1 = s_1 + \alpha \cdot e(N_1) = -0.4 + 0.2 \cdot (-1.0) = -0.6$

$s_2 = s_2 + \alpha \cdot e(N_2) = 0.2 + 0.2 \cdot (-0.4) = 0.12$

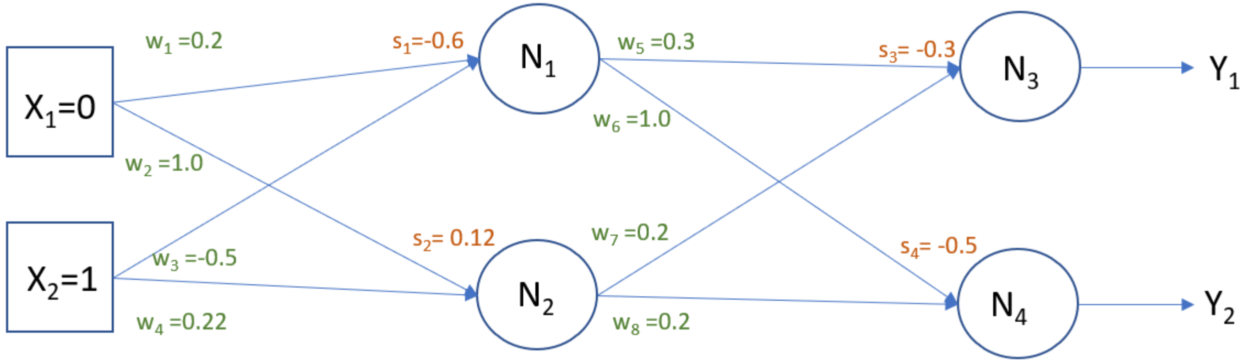
Paso 5: Repetir desde paso 2

- Calculamos el error "total": Error cuadrático medio = $\frac{1}{2}\sum_{i=1}^{2}(Y_{esperado} - Y_{obtenido})^2 = \frac{1}{2}\left((0-0)^2 + (0-1)^2\right) = 0.5$

- ¿Es acceptable este error?
  - Sí → Hemos terminado
  - No → Repetir desde paso 2 con los nuevos pesos (w) y sesgos (s)



2ª iteración:



Paso 2:

N_1=f(0.2 x_1 – 0.5 x_2 - 0.6) = f(0.2 · 0 – 0.5 · 1 - 0.6)= f(-1.1)= 0

N_2=f(1.0 x_1 + 0.22 x_2 + 0.12) = f(1.0 · 0 + 0.22 · 1 + 0.12)= f (0.32) = 1

Y_1= N_3 =f(0.3 N_1 + 0.2 N_2 - 0.3) = f(0.3 · 0 + 0.2 · 1 - 0.3)= f(-0.1) = 0

Y_2= N_4 =f(1.0 N_1 + 0.2 N_2 - 0.5) = f(1.0 · 0 + 0.2 · 1 - 0.5) = f (-0.3) = 0

Paso 3: e(N_3) = Y_1 esperado – Y_1 obtenido = 0 - 0 = 0

e(N_4) = Y_2 esperado – Y_2 obtenido = 0 - 0 = 0

Paso 4: e(N_3) y e(N_4) son 0, por lo tanto, no se generarán valores distintos de los pesos y sesgos.

Paso 5: el error total será 0: Error cuadrático medio = $\frac{1}{2}\sum_{i=1}^{2}(Y_{esperado} - Y_{obtenido})^2 = \frac{1}{2}\left((0-0)^2 + (0-0)^2\right) = 0.0$

Hemos encontrado los valores que minimizan el error. No tiene sentido seguir. La red ya está entrenada, se puede usar para predecir resultados con nuevos valores de entrada.