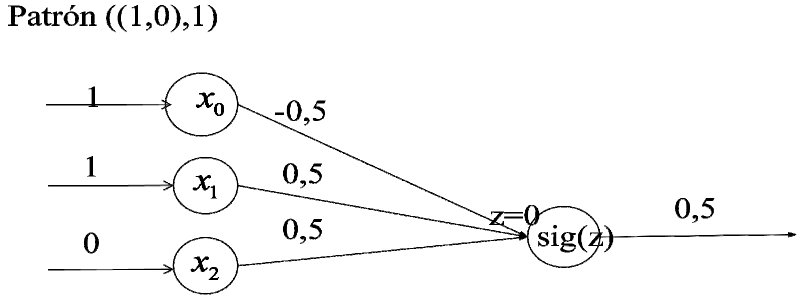
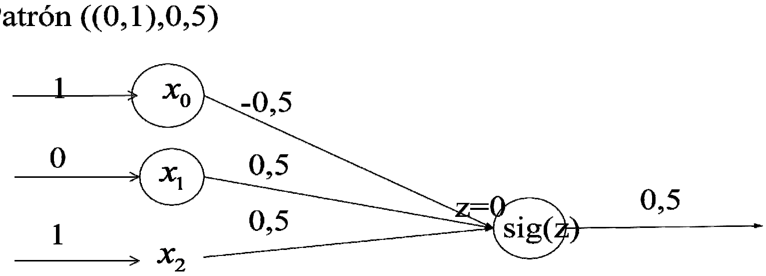
Soluciones a los ejercicios propuestos del algoritmo de retropropagación del error en redes neuronales

Ejercicio 1.

Disponemos de un perceptron simple con pesos w_1=w_2=0,5, w_0=-0,5 y un conjunto de entrenamiento D = {<(1,0),1>, <(0,1),0,5>}, donde utilizamos la función sigmoide como activación. Obtener la variación que se produce en el error cuadrático medio cometido sobre el conjunto de entrenamiento D tras un paso del algoritmo de descenso por gradiente (factor de aprendizaje η =0.8, versión off-line).

Solución.-El modelo de red es el siguiente





Para los dos patrones del conjunto de entrenamiento el error cuadrático medio inicial es:

$$\frac{1}{2}\sum_{i=1}^{2}(y_i - out_i)^2 = \frac{1}{2}[(1-0,5)^2 + (0,5-0,5)^2] = 0,125$$

Puesto que la activación hacia adelante para el **patrón 1, es**

net=-0,5+0,5*1+0,5*0=0; out=σ(0)=0,5

y la retropropagación del error, si tenemos en cuenta que queremos minimizar la función de error

$$E = \frac{1}{2}\sum_{i=1}^{2}(\mathrm{d}_i - o_i)^2 = \frac{1}{2}(\mathrm{d}_1 - o_1)^2 + \frac{1}{2}(\mathrm{d}_2 - o_2)^2$$

, a la hora de derivar con respecto a o_i, tenemos

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial o_1}\frac{\partial o_1}{\partial w_{ij}^2} = -(d_1 - o_1)\frac{\partial o_1}{\partial w_{ij}^2}$$

luego $\delta_1 = -(d_1 - o_1)\, \mathrm{o}_1(1-\mathrm{o}_1)$, d_1=1 y o_1=0,5, luego

δ_1=-(1-0,5)*0,5(1-0,5)=-0,125 ;
$\Delta w_1^{(1)}= \delta_1 * x_1$=-0,125*1=-0,125;
$\Delta w_2^{(1)}= \delta_1 * x_2$=-0,125*0=0

Para el patrón 2
net=-0,5+0,5*0+0,5*1=0; out=σ(0)=0,5
δ_2=(0,5-0,5)*0,5(1-0,5)=0
$\Delta w_1^{(2)}= \delta_2 * x_1$=0*0=0;
$\Delta w_2^{(1)}= \delta_2 * x_2$=0*1=0

Incremento total
$\Delta w_1= \Delta w_1^{(1)}+ \Delta w_1^{(2)}$=-0,125; $w_1^{*}= w_{1-}\ \eta\ \Delta w_1$=0,5-0,8*(-0,125)=0,6

$\Delta w_2= \Delta w_2^{(1)}+ \Delta w_2^{(2)}$=0; $w_2^{*}= w_{2-}\ \eta\ \Delta w_2$=0,5-0,8*(0)=0,5

$\Delta w_0^{(1)}= \delta_1$*1=-0,125, $\Delta w_0^{(2)}= \delta_2$*1=0, $\Delta w_0= \Delta w_0^{(1)}+ \Delta w_0^{(2)}$=-0,125

w_0*= $w_{0-}\ \eta\ \Delta w_0$=-0,5-0,8*(-0,125)=-0,4

Los pesos finales son w_1^{*}= 0,6 ; w_2^{*}=0,5 y w_0*=-0,4
y la nuevas salidas de red son para el patrón 1
net=-0,4+0,6*1+0,5*0=0,2; out=σ(0,2)=0,550

y para el patrón 2
net=-0,4+0,6*0+0,5*1=0,1; out=σ(0,1)=0,525

el error cuadrático medio final es:

$$\frac{1}{2}\sum_{i=1}^{2}(y_i - out_i)^2 = \frac{1}{2}[(1-0,550)^2 + (0,5-0,525)^2] = 0,1016$$

La variación es de 0,125-0,1016=0,0234 de reducción del error

Ejercicio 2.
Diseñar un perceptron simple con 3 valores de entrada y función de activación de salto que sirva para calcular la mayoría simple, es decir, que reciba 3 entradas binarias y devuelva un 1 si hay mas unos que ceros y un -1 en caso contrario (empate, o más ceros que unos).
Solución.-

Para el problema de la mayoría simple tenemos los patrones y las salidas,

x_1	x_2	x_3	y
1	1	1	1
1	1	0	1
1	0	1	1
0	1	1	1
1	0	0	-1
0	1	0	-1
0	0	1	-1
0	0	0	-1

la función sign es de la forma

$$sign(z) = \begin{cases} 1 & \text{si } z > 0 \\ -1 & \text{si } z < 0 \end{cases}$$

$z=w_{10}+w_{11}x_1+w_{12}x_2+w_{13}x_3$

Definimos un modelo de red donde todos los pesos son iguales a 1 y el sesgo toma el valor -1,5. De esta forma los siguientes cuatro patrones

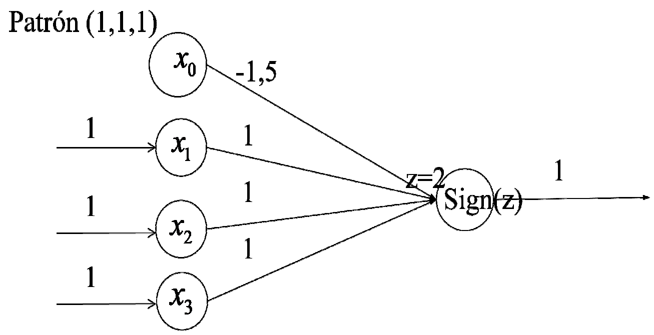
z_1=-1,5+1+1+1=1,5; z_2=-1,5+1+1+0=0,5; z_3=-1,5+1+0+1=0,5; z_4=-1,5+0+1+1=0,5

al tener salidas positivas la función *sign* les asigna un 1, mientras que los siguientes cuatro patrones

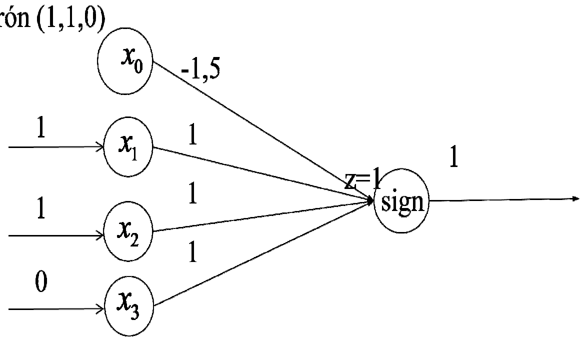
z_5=-1,5+1+0+0=-0,5; z_6=-1,5+0+1+0=-0,5, z_7=-1,5+0+0+1=-0,5; z_8=-1,5+0+0+0=-1,5

al tener salidas negativas la función *sign* les asigna un -1

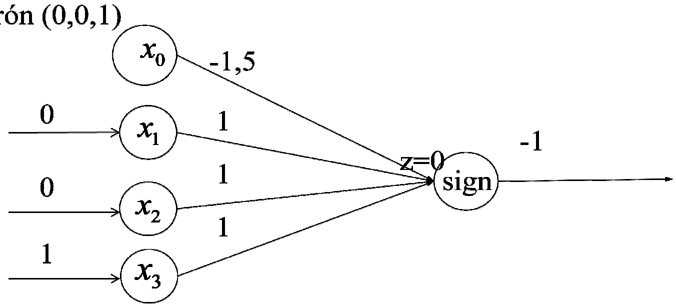
Un ejemplo para cuatro patrones es de la forma



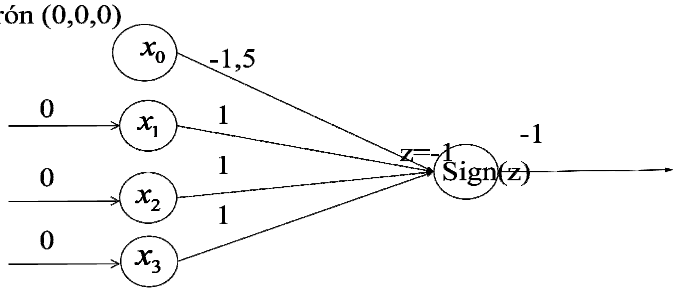
Patrón (1,1,0)



Patrón (0,0,1)

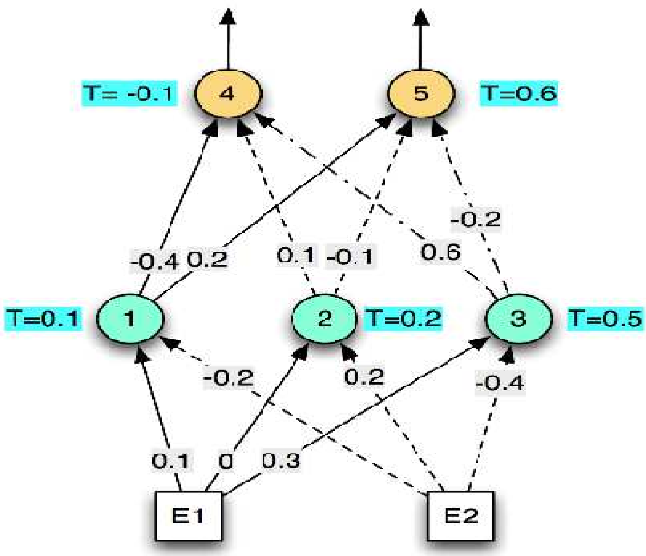


Patrón (0,0,0)



Ejercicio 3.

Disponemos de una red neuronal como la que aparece en la figura, donde T es el umbral de cada neurona (es decir, el valor a partir del cual empieza a activarse) y todas las neuronas son de tipo sigmoide.



Supongamos que alimentamos la red utilizando el patrón:

(**x**,**y**) = (**x**=[E1=0,6, E2=0,1], **y**=[S4=0, S5=1])

Aplique la propagación hacia atrás del error, indicando como cambiarían los pesos, utilizando para ello una tasa de aprendizaje de 1.

Solución.-

Si los umbrales T de cada neurona los consideramos como sesgos estos se aplican cambiándoles el signo

Si propagamos las entradas hacia adelante tenemos las entradas en la capa intermedia

net1=0,1*E1-0,2*E2-T1=0,1*0,6-0,2*0,1-0,1=-0,06

net2=0*E1+0,2*E2-T2=0*0,6+0,2*0,1-0,2=-0,18

net3=0,3*E1-0,4*E2-T3=0,3*0,6-0,4*0,1-0,5=-0,36

Las salidas de los tres nodos de la capa oculta son

$$out_1 = \sigma(net_1) = \frac{1}{1+e^{0,06}} = 0,4850$$

$$out_2 = \sigma(net_2) = \frac{1}{1+e^{0,18}} = 0,4551$$

$$out_3 = \sigma(net_3) = \frac{1}{1+e^{0,36}} = 0,4110$$

Los valores de salida de la red son

net_4=-0,4*0,4850+0,1*0,4551+0,6*0,4110+0,1=0,1981

net_5=0,2*0,4850-0,1*0,4551-0,2*0,4110-0,6=-0,6307

Las salidas de los dos nodos de la capa de salida son

$$out_4 = \sigma(net_4) = \frac{1}{1+e^{-0,1981}} = 0,5494$$

$$out_5 = \sigma(net_5) = \frac{1}{1+e^{00,6307}} = 0,3474$$

La retropropagación es

δ_5=(y_2-out_5)*out_5*(1-out_5)=(1-0,3474)*0,3474*(1-0,3474)=0,1480

δ_4=(y_1-out_4)*out_4*(1-out_4)=(0-0,5494)*0,5494*(1-0,5494)=-0,1360

δ_1=[w_{41}* δ_4+w_{51}* δ_5]*out_1*(1-out_1)=

= [(-0,4)*(-0,1360)+0,2*0,1480](0,4850)(1-0,4850)=0,0210

δ_2=[w_{42}* δ_4+w_{52}* δ_5]*out_2*(1-out_2)= -0,0071

δ_3=[w_{43}* δ_4+w_{53}* δ_5]*out3*(1-out_3)= -0,0269

Ajuste de incremento de pesos

E1=0,6; E2=0,1

Δw_{40}= δ_4*1=-0,1360, Δw_{41}= δ_4*out_1=-0,1360*0,4850=-0,0659

Δw_{42}= δ_4*out_2=-0,0619, Δw_{43}= δ_4*out_3=-0,0559

Δw_{50}= δ_5*1=0,1479, Δw_{51}= δ_5*out_1=0,0718, Δw_{52}= δ_5*out_2=0,0673,

Δw_{53}= δ_5*out_3=0,0608

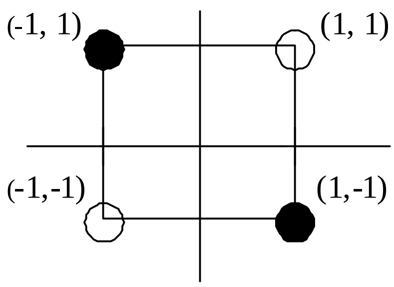
$\Delta w_{10}= \delta_1*1=0,0210$, $\Delta w_{20}= \delta_2*1=-0,0071$, $\Delta w_{30}= \delta_3*1=-0,0269$
$\Delta w_{11}= \delta_1*E1=0,0126$, $\Delta w_{12}= \delta_1*E2=0,0021$
$\Delta w_{21}= \delta_2*E1=-0,0042$, $\Delta w_{22}= \delta_2*E2=-0,0007$
$\Delta w_{31}= \delta_3*E1=-0,0161$, $\Delta w_{32}= \delta_3*E2=-0,0027$

Ajuste de pesos
$w_{10}=w_{10}-\Delta w_{10}=-0,1-0,0210=-0,1210$

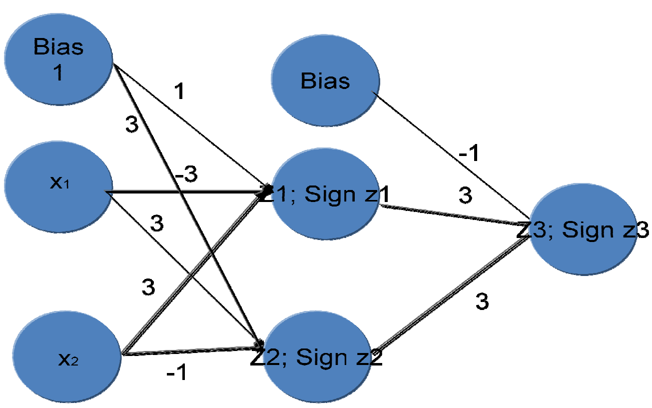
Ejercicio 4.
Demostrar geométricamente que un perceptron simple no puede calcular la función lógica XOR. Construir una red neuronal (usando la función de activación de salto) que sea capaz de obtenerla.

Solución.- La función XOR es un ejemplo de función Booleana que no es linealmente separable. Es imposible separar los “1” de los “-1” por cualquier recta. Si consideramos la recta que pasa por los puntos (1,1) y (-1,-1) tenemos la ecuación x-y= 0 y los patrones (-1,1) y (1,-1) se encuentra a ambos lados de la recta, igual ocurre con cualquier recta paralela a la anterior.

De igual forma la recta que pasa por los puntos (-1,1) y (1,-1) es de la forma x+y= 0 y los patrones (1,1) y (-1,-1) se encuentran a ambos lados de la recta, igual ocurre con cualquier recta paralela a la anterior.



Una posible solución es la que se muestra a continuación



#	Inputs		Neurona 1		Neurona 2		Neurona 3		XOR=
			$\tilde{W}=(1,-3,3)$		$\tilde{W}=(3,3,-1)$		$\tilde{W}=(-1,3,3)$		$=x_1 \oplus x_2$
	x_1	x_2	Z1	$\text{sign}(z1)$ output	Z2	$\text{sign}(z2)$ output	Z3	$\text{sign}(z3)$ output	
1)	1	1	1	1	5	1	5	1	1
2)	1	-1	-5	-1	7	1	-1	-1	-1
3)	-1	1	7	1	-1	-1	-1	-1	-1
4)	-1	-1	1	1	1	1	5	1	1

para el patrón (1,1) tenemos para la primera neurona oculta

$$z_1 = w_{01} + w_{11}x_1 + w_{21}x_2 = 1 + (-3)\times 1 + 3\times 1 = 1,\ \text{neurona 1}$$

luego sign(z_1)= 1

para la segunda neurona oculta

$$z_2 = w_{02} + w_{12}x_1 + w_{22}x_2 = 3 + 3\times 1 + (-1)\times 1 = 5,\ \text{neurona 2}$$

luego sign(z_2)= 1

Para la neurona de salida tenemos

$$z_3 = \beta_0 + \beta_1 \times signz_1 + \beta_2 \times signz_2 = -1 + 3\times 1 + 3\times 1 = 5,\ \text{neurona 3}$$

luego sign(z_3)= 1

para el patrón (1,-1) tenemos para la primera neurona oculta

$$z_1 = w_{01} + w_{11}x_1 + w_{21}x_2 = 1 + (-3)\times 1 + 3\times(-1) = -5,\ \text{neurona 1}$$

luego sign(z_1)= -1

para la segunda neurona oculta

$$z_2 = w_{02} + w_{12}x_1 + w_{22}x_2 = 3 + 3\times 1 + (-1)\times(-1) = 7,\ \text{neurona 2}$$

luego sign(z_2)= 1

Para la neurona de salida tenemos

$$z_3 = \beta_0 + \beta_1 \times signz_1 + \beta_2 \times signz_2 = -1 + 3\times(-1) + 3\times 1 = -1,\ \text{neurona 3}$$

luego sign(z_3)= -1

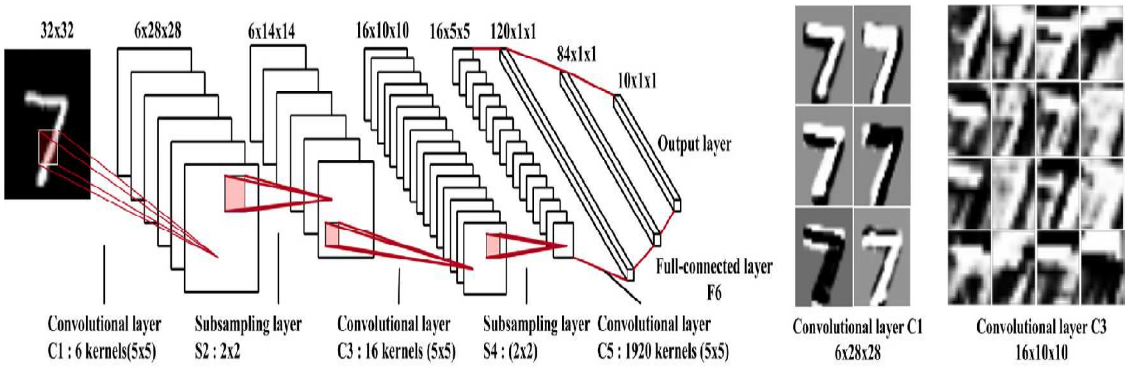
y así sucesivamente

Ejercicio 5.

Imaginemos que queremos diseñar una red neuronal para reconocer dígitos manuscritos. Proponga un posible diseño para dicha red neuronal, incluyendo su arquitectura y funciones de activación y detallando como organizaría el proceso de recolección y preprocesamiento de los datos ¿Que algoritmo podríamos utilizar para entrenar la red?.

Solución.-

El diseño más apropiado es el de una red neuronal convolucional que estudiaremos más adelante y que mostramos como un adelanto.



(a) LeNet-5 network (b) Learned features

Se recibe como entrada una imagen de 32x32 pixeles. Se le aplica una capa convolucional de 6 filtros de 5x5 cada uno con un stride=1pasando a ser imágenes 6 imágenes de 28x28

Se aplica una capa de pooling (subsample) con un filtro de 2x2 con un stride= 2, pasando a ser 6 imágenes de 14x14

Se aplica una nueva capa convolucional de 16 filtros de tamaño 5x5 y stride=1, pasando a ser 16 imágenes de 10x10. Se aplica una capa de pooling (subsample) con un filtro de 2x2 y stride=2 y tenemos 16 imágenes de 5x5

Se aplica una capa convolucional de 120 filtros de 5x5 pasando a ser 120 valores (1x1)

A continuación se aplican dos capas completamente conectadas, una con 84 nodos y la otra con 10 y una capa de salida de tipo softmax donde hay 10 nodos asociados a las salidas de los dígitos 0,1,...,9

Ejercicio 6.

Aplique el algoritmo de retropropagacion del error a un perceptron simple hasta obtener la primera actualización del peso w_2 con η= 0.1, usando activación sigmoide y el siguiente conjunto de entrenamiento y considerando como función de pérdida el error cuadrático medio :

Patrón	x_1	x_2	x_3	y
p_1	0,7	0,2	0,1	0,3
p_2	0,3	0,5	0,2	0,8
p_3	0,1	0,1	0,8	0,6

Use como pesos iniciales w_0= -0,1 y w_1=w_2=w_3=0,1,

Solución.-

Primer patrón

x_1	x_2	x_3	y

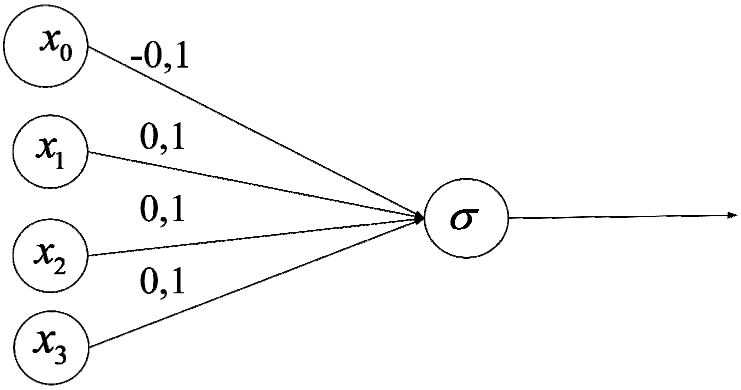
0,7	0,2	0,1	0,3

Puesto que la activación hacia adelante para el **patrón 1, es**
net=-0,1+0,1*0,7+0,1*0,2+0,1*0,1=0; out=σ(0)=0,5
y la retropropagación del error, si tenemos en cuenta que queremos minimizar la función de error cuadrático medio.

$$E = \frac{1}{3}\sum_{i=1}^{3}(\mathrm{d}_i - o_i)^2 = \frac{1}{3}(\mathrm{d}_1 - o_1)^2 + \frac{1}{3}(\mathrm{d}_2 - o_2)^2 + \frac{1}{3}(\mathrm{d}_3 - o_3)^2$$

Entonces $\delta = -\frac{2}{3}(d_1 - o_1)o_1(1 - o_1)$, d_1=0,3 y o_1=0,5, luego

$\delta^{(1)}$ =(2/3)(0,5-0,3)*0,5(1-0,5)=0,03 ; $\Delta w_2^{(1)} = \delta^{(1)} * x_2$=0,03*0,1=0,003



patrón 2
net=-0,1+0,1*0,3+0,1*0,5+0,1*0,2=0; out=σ(0)=0,5
$\delta^{(2)}$ =(2/3)(0,5-0,8)*0,5(1-0,5)=-0,075 ; $\Delta w_2^{(2)} = \delta^{(2)} * x_2$=-0,05*0,1=-0,005

patrón 3
net=-0,1+0,1*0,1+0,1*0,1+0,1*0,8=0; out=σ(0)=0,5
$\delta^{(3)}$ =(2/3)(0,5-0,6)*0,5(1-0,5)=-0,017 ; $\Delta w_2^{(3)} = \delta^{(3)} * x_2$=-0,017*0,1=-0,0017

Total
$\Delta w_2 = \Delta w_2^{(1)} + \Delta w_2^{(2)} + \Delta w_2^{(3)}$=0,003-0,005-0,0017=-0,0037
$w_2^{*} = w_2 - \eta \, \Delta w_2$=0,1-0,1*(-0,0037)=0,10037

Examen final 18 FEBRERO de 2021

Ejercicio 7
Diseñar un perceptron simple con 2 valores de entrada y función de activación de salto que sirva para calcular la mayoría simple, es decir, que reciba 2 entradas binarias y

devuelva un 1 si hay mas unos que ceros y un -1 en caso contrario (empate, o más ceros que unos).

Solución.-

Para el problema de la mayoría simple tenemos los patrones y las salidas,

x1	x2	y
1	1	1
1	0	-1
0	1	-1
0	0	-1

la función sign es de la forma

$$sign(z) = \begin{cases} 1 & \text{si } z > 0 \\ -1 & \text{si } z < 0 \end{cases}$$

z=w10+w11x1+w12x2

Definimos un modelo de red donde todos los pesos son iguales a 1 y el sesgo toma el valor -1,5. De esta forma los siguientes cuatro patrones

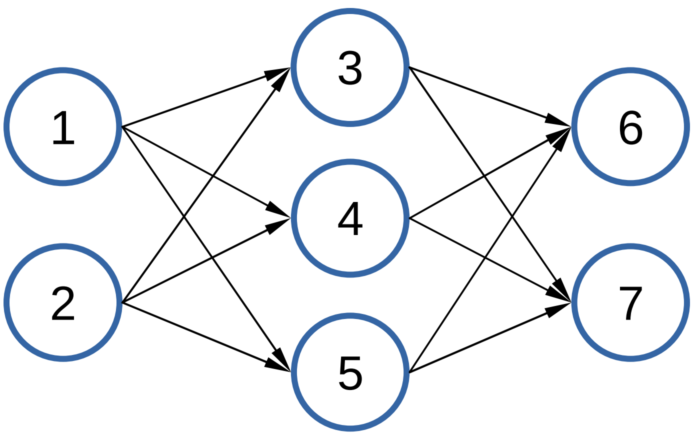
z1=-1,5+1+1=0,5; z2=-1,5+1+0=-0,5; z3=-1,5+0+1=-0,5; z4=-1,5+0+0=-1,5

Al primero, al tener salida positiva la función sign le asigna un 1, mientras que los tres siguientes al tener salidas negativas la función sign les asigna un -1

Soluciones examen parcial

Ejercicio 8.

Supongamos la siguiente red neuronal:



Todas las neuronas tienen sesgo y función de activación sigmoide. El valor inicial de todos los pesos (incluidos los sesgos) es 1. Las neuronas 1 y 2 son las neuronas de entradas, mientras que las neuronas 6 y 7 son las neuronas de salida.

a) Se pide aplicar el algoritmo de retropropagación del error para actualizar los tres pesos de la neurona 4, considerando los siguientes dos patrones: $\mathbf{p}_1$=<(1,0),(0,1)> y $\mathbf{p}_2$=<(0,1),(1,0)>, donde <(a,b),(c,d)> quiere decir que el valor de entrada para la neurona 1 es a, el valor de entrada para la neurona 2 es b, el valor deseado para la neurona 6 es *c* y el valor deseado para la neurona 7 es *d*. Considerar un algoritmo *offline* y un factor de aprendizaje de 0,8.

b) ¿Qué cálculos podría aprovechar si se pidiese lo mismo para la neurona 7?

Solución.- Propagación hacia adelante

Patrón 1. $\mathbf{p}_1$=<(1,0),(0,1)> , las variables de entrada x1=1 y x2=0, luego

n3=w31*x1+w32*x2+1=1*1+1*0+1=2,

h3(n3)=1/(1+exp(-n3))=0.8807

n4=1*1+1*0+1=2, h4(n4)=0.8807

n5=2, h5(n5)= 0.8805

n6=w63*h3+w64*h4+w65*h5+w60=1*0.8807+1*0.8807+1*0.8807+1=3.6424

o6= h6(n6)= 1/(1+exp(-n6))=0.9745,

n7=n6;

o7= h7(n7)= 1/(1+exp(-n7))=0.9745

δ6=(0.9745-0)*o6*(1-o6)=0.0242,

δ7=(0.9745-1)*o7*(1-o7)=-0.0006

$$\delta_4 = (\sum_{i=6}^{7} \delta_i w_{4i}) h_4 (1-h_4) = (0.0242 - 0.0006) 0.8807 (1-0.8807) = 0.0025$$

$$\Delta w_{14} = \delta_4 x_1 = 0.0025 * 1 = 0.0025$$;

$$\Delta w_{24} = \delta_4 x_2 = 0.0025 * 0 = 0$$;

$$\Delta w_{04} = \delta_4 x_0 = 0.0025 * 1 = 0.0025$$

Patrón 2. $\mathbf{p}_1$=<(1,0),(0,1)> , las variables de entrada x1=0 y x2=1, luego la activación hacia adelante es la misma que con el patrón 1

δ6=(0.9745-1)*o6*(1-o6)=-0.0006, y δ7=(0.9745-0)*o7*(1-o7)=0.0242, luego

δ4=0.0025 y

$$\Delta w_{14} = \delta_4 x_1 = 0.0025 * 0 = 0$$,

$$\Delta w_{24} = \delta_4 x_2 = 0.0025 * 1 = 0.0025$$ y

$$\Delta w_{04} = \delta_4 x_0 = 0.0025 * 1 = 0.0025$$

$$\Delta w_{14} = 0.0025 + 0 = 0.0025$$;

$$\Delta w_{24} = 0 + 0.0025 = 0.0025$$;

$$\Delta w_{04} = 0.0025 + 0.0025 = 0.0050$$

$$w_{14}' = w_{14} - \eta \Delta w_{14} = 1 - 0.8 * 0.0025 = 0.998$$;

$$w_{24}' = w_{24} - \eta \Delta w_{24} = 1 - 0.8 * 0.0025 = 0.998$$

$$w_{04}' = w_{04} - \eta \Delta w_{04} = 1 - 0.8 * 0.005 = 0.996$$