

ASIGNATURA:

**INTELIGENCIA ARTIFICIAL Y ROBOTICA**

TEMA:

**“PERCEPTRON CHALLENGE”**

**ALUMNO S :**

Cajas Monier, Erick Diego

Carlín Mendoza, Daniela del Carmen

Díaz Cortegana, Victor Hugo

Garcia Mori, Brenda

**PROFESOR :**

Jiménez Motte, Fernando

LIMA, PERÚ

2017

**1. Grafique la Arquitectura de red del Perceptrón.**

b

W1

SUMADOR

a

n

W2

F = hardlims

W3

Step 0: Initialize all weights and bias in 0. (**w1=0, w2=0, w3=0, b=0**)

Set learning rate alpha (0 < alpha < 1)

Alpha set in 1.

Step 1: Do steps 2-6

Step 2: For each training pair s:t, do steps 3-5

Step3. Set activations for input units:

Xi = Si i = 1 to 3

**APPLE = [1 1 -1]**

**X1= 1 x2=1 x3=-1**

**ORANGE= [1 -1 -1]**

**X1=1 X2=-1 X3=-1**

Step 4: compute response of output unit

**APPLE=> y\_int = b+ x1w1+ x2w2+ x3w3**

**= 0 + 0 + 0 + 0**

**= 0**

**ORANGE=> y\_int = b+ x1w1+ x2w2+ x3w3**

**= 0 + 0 + 0 + 0**

**= 0**

Step 5:

IF y diferente de t

APPLE **=> 0=1**

**ORANGE=> 0=-1**

**SE OBSERVA QUE EL OUTPUT ES DIFERENTE A EL TARGET, POR O QUE SE PROCEDE A CALCULAR EL ERROR DE APPLE Y ORANGE.**

Wi(new) = wi(old) + xiy (i = 1 to 3)

**APPLE=> w1(new) = w1(old) + alpha(t)(x1)**

**W1(new) = 0 + (1)(1)(1)**

**W1(new) = 1**

**W2(new) = w2(old) + alpha(t)(x2)**

**W2(new) = 0 + (1)(1)(1)**

**W2(new) = 1**

**W3(new) = w3(old) + alpha(t)(x3)**

**W3(new) = 0 + (1)(1)(-1)**

**W3(new) = -1**

**B(new)=b(old)+ alpha(t)**

**B(new)=0+(1)(1)**

**B(new)=1**

**ORANGE=> w1(new) = w1(old) + alpha(t)(x1)**

**W1(new) = 0 + (1)(-1)(1)**

**W1(new) = -1**

**W2(new) = w2(old) + alpha(t)(x2)**

**W2(new) = 0 + (1)(-1)(-1)**

**W2(new) = 1**

**W3(new) = w3(old) + alpha(t)(x3)**

**W3(new) = 0 + (1)(-1)(-1)**

**W3(new) = 1**

**B(new)=b(old)+ alpha(t)**

**B(new)=0+(1)(-1)**

**B(new)=-1**

2DA EPOCA

Step 2: For each training pair s:t, do steps 3-5

Step3. Set activations for input units:

Xi = Si i = 1 to 3

**APPLE = [1 1 -1]**

**X1= 1 x2=1 x3=-1**

**ORANGE= [1 -1 -1]**

**X1=1 X2=-1 X3=-1**

Step 4: compute response of output unit

**APPLE=> y\_int = b+ x1w1+ x2w2+ x3w3**

**=1 +(1)(1) +(1)(1) + (-1)(-1)**

**= 4**

**ORANGE=> y\_int = b+ x1w1+ x2w2+ x3w3**

**= -1 + (1)(-1) + (-1)(1) + (-1)(1)**

**= -4**

Step 5:

IF y diferente de t

APPLE **=> 1=1**

**ORANGE=> -1=-1**

**EN ESTE PASO SE OBSERVA QUE EL OUTPUT Y EL TARGET SON IGUALES, POR LO QUE SE PUEDE DECIR QUE YA APRENDIO.**

Wi(new) = wi(old) (i = 1 to 3)

**APPLE=> w1(new) = w1(old)**

**W1(new) = 1**

**W2(new) = w2(old)**

**W2(new) = 1**

**W3(new) = w3(old)**

**W3(new) = -1**

**B(new)=b(old)**

**B(new)=1**

**ORANGE=> w1(new) = w1(old)**

**W1(new) = -1**

**W2(new) = w2(old)**

**W2(new) = 1**

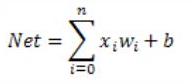
**W3(new) = w3(old)**

**W3(new) = -1**

**B(new)=b(old)**

**B(new)=-1**

**2. Describa la ecuación matemática del Perceptrón para clasificar los vectores de entrada (“input vectors”) tridimensionales**



**APPLE = [1 1 -1]**

Y\_IN= X1W1+X2W2+X3W3+B

1(W1)+1(W2)+ -1(W3)+1

W1+W2-W3+1

**ORANGE = [1 -1 -1]**

Y\_IN= X1W1+X2W2+X3W3+B

1(W1)+ -1(W2)+ -1(W3)+1

W1-W2-W3+1

**3. Calcule a mano los elementos de la matriz de peso W y elija el sesgo b para que el perceptrón distinga entre manzanas y naranjas.**

**APPLE**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | b |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| 1 | 1 | -1 | 1 | 0 | 0 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | 1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | b |
|  |  |  |  |  |  |  |  |  |  |  | 1 | 1 | -1 | 1 |
| 1 | 1 | -1 | 1 | 4 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | -1 | 1 |

**ORANGE**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | b |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| 1 | -1 | -1 | 1 | 0 | 0 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | -1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | -1 | 1 | 1 | -1 |
| 1 | -1 | -1 | 1 | -4 | -1 | -1 | 0 | 0 | 0 | 0 | -1 | 1 | 1 | -1 |

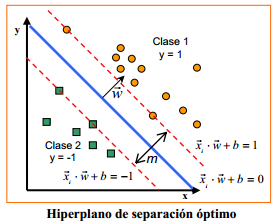
**AL HALLAR LOS PESOS Y EL BIAS DE FORMA TABULADA SE OBSERVA QUE APRENDE A EN LA ASEGUNDA EPOCA Y QUE LOS RESULTADOS SON IDENTICOS AL LOS PESOS HALLADOS CON EL ALGORITMO.**

**4. Describa la ecuación de separabilidad lineal o separatrix (hiperplano de separación).**

W1X1+W2X2+….+.WnXn+B=0

W1X1+W2X2+W3X3+B<Theta

W1X1+W2X2+W3X3+B>Theta



**HIPERPLANO POR COMPUTADORA**

net = perceptron;

p1 = [1;1;-1];

p2= [1;-1;-1];

t=[1 0];

\*\* ”**0” POR QUE LAS LIBRERIAS DE MATLAB NO TRABAJAN CON BIPOLARES EN LOS TARGET NI EN LOS OUTPUT, POR LO TANTO PARA DECLARAR UN “-1” EN TARGET Y OUTPUT, SE TIENE QUE DIGITAR “0” \*\***

net.trainParam.epochs = 2;

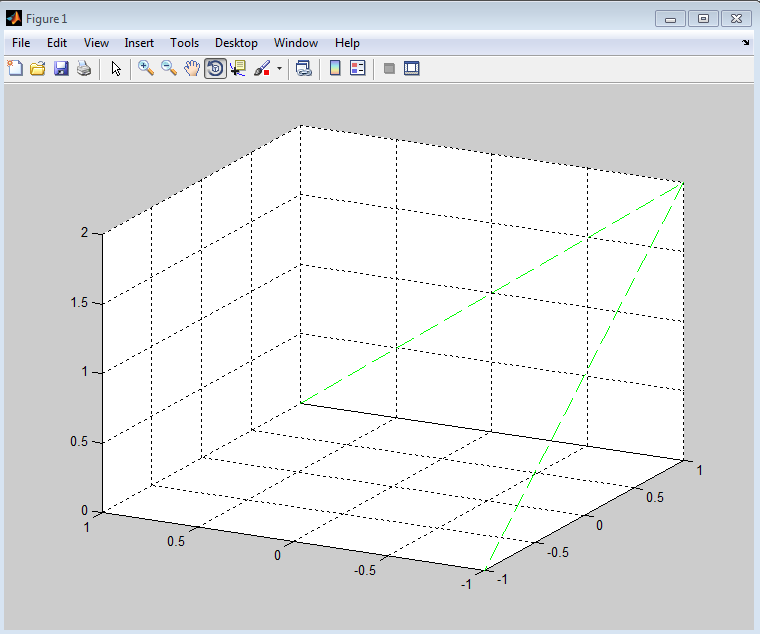
net = train(net,p,t);

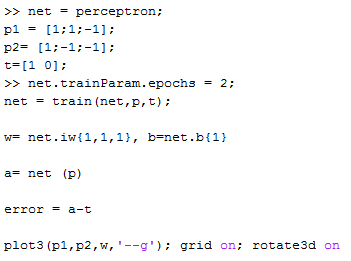
w= net.iw{1,1,1}, b=net.b{1}

a= net (p)

error = a-t

plot3(p1,p2,w,'--g'); grid on; rotate3d on





**5. Escriba una rutina en MATLAB para simular el clasificador (perceptrón) que seleccionará la fruta en la máquina. Realice un Test de la operación del “perceptron pattern classifier”**

**CODIGO ABIERTO**

function [ output\_args ] = IA1( input\_args )

function [o] = neuron(W,P,b,f)

fprintf('Input Points:\n');

display(P);

fprintf('Input Weights:\n');

display(W);

fprintf('Input Bias:\n');

display(b);

n = W \* P + b;

fprintf('Function: ');

o = f(n);

fprintf('\n');

end

function [o] = hardlim(x)

fprintf('hardlim\n');

fprintf('Input:\t');

disp(x);

fprintf('Output:\t');

if x >= 0

o = 1;

disp(o);

else

o = -1;

disp(o);

end

end

W = [0 1 0];

b = 0;

Orange = [1;-1;-1];

Apple = [1; 1; -1];

disp('Point matrix is represented as [shape; texture; weight]');

fprintf('Apple Input:\n');

display('--------------------------------------------------');

neuron(W, Apple, b, @hardlim);

fprintf('Orange Input:\n');

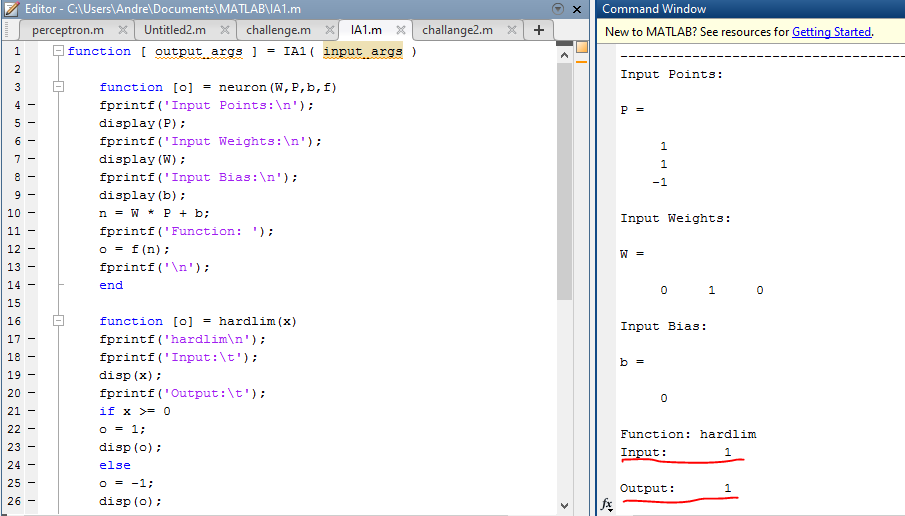
display('--------------------------------------------------');

neuron(W, Orange, b, @hardlim);

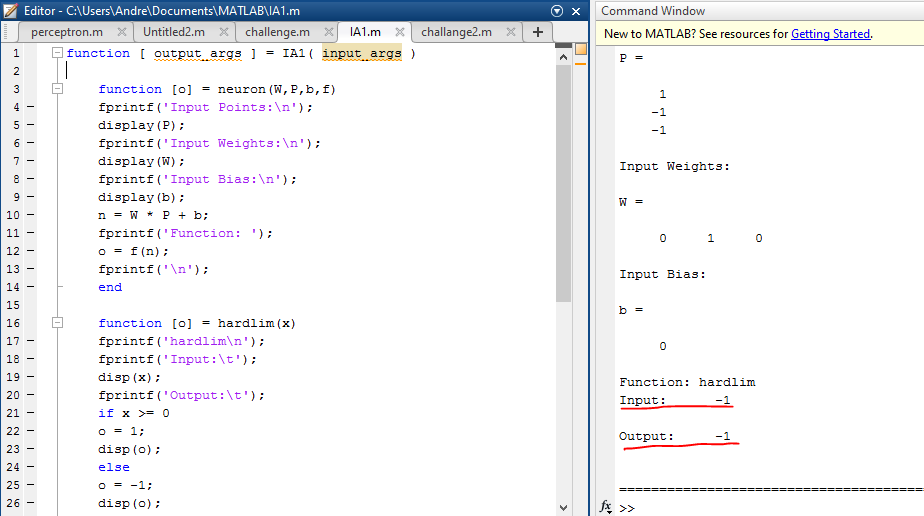
display('==================================================');

end

**APPLES**



**ORANGES**



**CODIGO CERRADO**

net = perceptron;

p = [[1;1;-1] [1;-1;-1]];

t=[1 0]; \*\* ”**0” POR QUE LAS LIBRERIAS DE MATLAB NO TRABAJAN CON BIPOLARES EN LOS TARGET NI EN LOS OUTPUT, POR LO TANTO PARA DECLARAR UN “-1” EN TARGET Y OUTPUT, SE TIENE QUE DIGITAR “0” \*\***

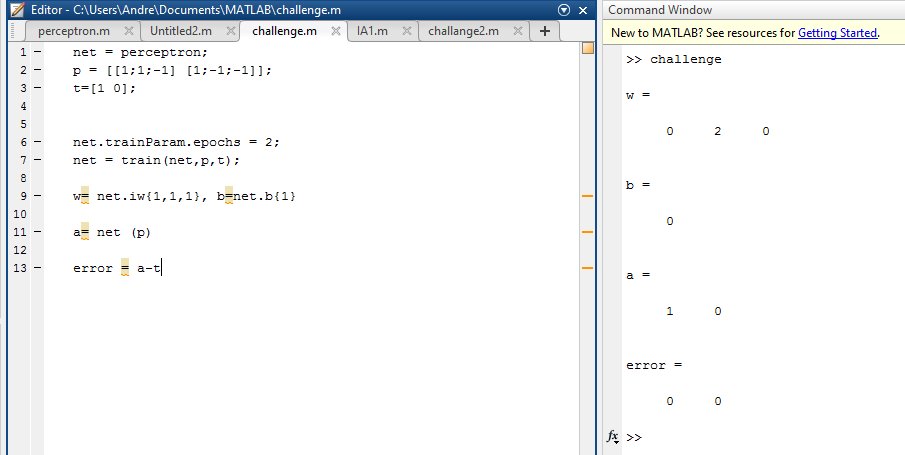
net.trainParam.epochs = 2;

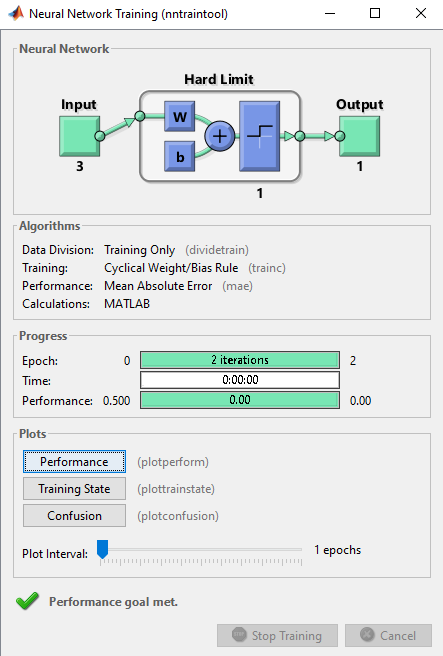
net = train(net,p,t);

w= net.iw{1,1,1}, b=net.b{1}

a= net (p)

error = a-t





**6. What If Condition ?**

**APPLE = [ELLIPTICAL, SMOOTH, POUND] = [-1 1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| -1 | 1 | -1 | 1 | 0 | 0 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | -1 | 1 | -1 | 1 |
| -1 | 1 | -1 | 1 | 4 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | -1 | 1 |

**AL INGRESAR UNA MANAZANA CON UNA FORMA ELLIPTICAL POR EL CLASIFICADOR, EL RESULTADO ES EL MISMO, SIGUE SIENDO CLASIFICADA COMO UNA MANZANA.**

**APPLE = [ROUND, ROUGH, POUND] = [1 -1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| 1 | -1 | -1 | 1 | 0 | 0 | 1 | 1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 1 | -1 | -1 | 1 |
| 1 | -1 | -1 | 1 | 4 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | -1 | -1 | 1 |

**AL INGRESAR UNA MANAZANA CON UNA TEXTURA ROUGH POR EL CLASIFICADOR, EL RESULTADO ES EL MISMO, SIGUE SIENDO CLASIFICADA COMO UNA MANZANA.**

**APPLE = [ELLIPTICAL, ROUGH, POUND] = [-1 -1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| -1 | -1 | -1 | 1 | 0 | 0 | 1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | 1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | -1 | -1 | -1 | 1 |
| -1 | -1 | -1 | 1 | 4 | 1 | 1 | 0 | 0 | 0 | 0 | -1 | -1 | -1 | 1 |

**AL INGRESAR UNA MANAZANA CON UNA FORMA ELLIPTICAL Y TEXTURA ROUGH POR EL CLASIFICADOR, EL RESULTADO ES EL MISMO, SIGUE SIENDO CLASIFICADA COMO UNA MANZANA.**

**ORANGE = [ELLIPTICAL, SMOOTH, POUND] = [-1 1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| -1 | 1 | -1 | 1 | 0 | 0 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 1 | -1 | 1 | -1 |
| -1 | 1 | -1 | 1 | -4 | -1 | -1 | 0 | 0 | 0 | 0 | 1 | -1 | 1 | -1 |

**AL INGRESAR UNA NARANJA CON UNA FORMA ELLIPTICAL Y TEXTURA SMOOTH POR EL CLASIFICADOR, EL RESULTADO ES EL MISMO, SIGUE SIENDO CLASIFICADA COMO UNA NARANJA.**

**ORANGE = [ROUND, ROUGH, POUND] = [1 -1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| 1 | -1 | -1 | 1 | 0 | 0 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | -1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | -1 | 1 | 1 | -1 |
| 1 | -1 | -1 | 1 | -4 | -1 | -1 | 0 | 0 | 0 | 0 | -1 | 1 | 1 | -1 |

**ORANGE = [ELLIPTICAL, ROUGH, POUND] = [-1 -1 -1]**

1ERA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 0 | 0 | 0 | 0 |
| -1 | -1 | -1 | 1 | 0 | 0 | -1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 |

2DA EPOCA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INPUT | | | | NET | OUT | T | WEIGHT CHANGES | | | | WEIGHTS | | | |
| X1 | X2 | X3 | B | W1 | W2 | W3 | B | W1 | W2 | W3 | B |
|  |  |  |  |  |  |  |  |  |  |  | 1 | 1 | 1 | -1 |
| -1 | -1 | -1 | 1 | -4 | -1 | -1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | -1 |

**AL INGRESAR UNA NARANJA CON UNA FORMA ELLIPTICAL POR EL CLASIFICADOR, EL RESULTADO ES EL MISMO, SIGUE SIENDO CLASIFICADA COMO UNA NARANJA.**

COMPROBACIÓN EN MATLAB SI APRENDE LA NARANJA EN FORMA ELÍPTICA

net = perceptron;

p = [-1;-1;-1];

t=[0];

\*\* ”**0” POR QUE LAS LIBRERIAS DE MATLAB NO TRABAJAN CON BIPOLARES EN LOS TARGET NI EN LOS OUTPUT, POR LO TANTO PARA DECLARAR UN “-1” EN TARGET Y OUTPUT, SE TIENE QUE DIGITAR “0” \*\***

net.trainParam.epochs = 2;

net = train(net,p,t);

w= net.iw{1,1,1}, b=net.b{1}

a= net (p)

error = a-t

