

Grupo 3

Participantes:

David Arias Calderón 20181020149

Luis Miguel Polo 20182020158

Taller 2 Ejercicio 2

Enunciado

Considerando la representación de una imagen digital como la mostrada en la figura 2 (se tiene como ejemplo una imagen digital de 8 x 8 píxeles), empleando una red neuronal tipo perceptrón diseñar un sistema para la identificación de imágenes.

Figura 2

0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	0
0	0	0	1	1	0	0	0
0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0

Configuraciones:

- Imagen digital: 6x6
- Número de imágenes: 6

Requerimientos de diseño:

- Promedio de identificación superior al 75%

Solución

Script de matlab

```
%Imágenes básicas de prueba

close all
clear all
warning off

%Imágenes de forma matricial
aj_z=[
    0 0 1 1 0 0;
    0 0 1 1 0 0;
    1 1 0 0 1 1;
    1 1 0 0 1 1;
    0 0 1 1 0 0;
    0 0 1 1 0 0
];
figure
imshow(aj_z,'InitialMagnification','fit')

aj=[
    1 0 1 0 1 0;
    0 1 0 1 0 1;
    1 0 1 0 1 0;
    0 1 0 1 0 1;
    1 0 1 0 1 0;
    0 1 0 1 0 1
];
figure
imshow(aj,'InitialMagnification','fit')

mo=[
    0 1 1 1 1 0;
    1 0 1 1 0 1;
    0 0 0 0 0 0;
    0 1 0 0 1 0;
    0 0 0 0 0 0;
    0 1 0 0 1 0
];
figure
imshow(mo,'InitialMagnification','fit')

hu=[
    1 1 0 0 1 1;
    1 1 0 0 1 1;
    0 0 0 0 0 0;
    1 0 0 0 0 1;
    1 0 1 1 0 1;
    1 0 0 1 0 0
];
figure
imshow(hu,'InitialMagnification','fit')
```

```

mo2=[
    1 0 1 1 0 1;
    1 1 0 0 1 1;
    1 1 0 0 1 1;
    1 0 1 1 0 1;
    0 1 0 0 1 0;
    1 1 0 0 1 1
];
figure
imshow(mo2,'InitialMagnification','fit')

ca=[
    0 0 0 0 0 0;
    1 0 0 0 0 1;
    0 1 1 1 1 0;
    0 1 1 1 1 0;
    1 0 0 0 0 1;
    0 0 0 0 0 0
];
figure
imshow(ca,'InitialMagnification','fit')

%generación de las cadenas de entrenamiento
AJ= [aj(:,1); aj(:,2); aj(:,3); aj(:,4); aj(:,5);aj(:,6)];
AJ_Z= [aj_z(:,1); aj_z(:,2); aj_z(:,3); aj_z(:,4); aj_z(:,5);aj_z(:,6)];
CA= [ca(:,1); ca(:,2); ca(:,3); ca(:,4); ca(:,5);ca(:,6)];
HU= [hu(:,1); hu(:,2); hu(:,3); hu(:,4); hu(:,5);hu(:,6)];
MO = [mo(:,1); mo(:,2); mo(:,3); mo(:,4); mo(:,5);mo(:,6)];
MO2 = [mo2(:,1); mo2(:,2); mo2(:,3); mo2(:,4); mo2(:,5);mo2(:,6)];

%vectorizando todas
P=[AJ, AJ_Z, CA, HU, MO, MO2];

%Función de salida
T=[
    1 0 0 0 0 0;
    0 1 0 0 0 0;
    0 0 1 0 0 0;
    0 0 0 1 0 0;
    0 0 0 0 1 0;
    0 0 0 0 0 1];

%Rangos de entrada
R=[zeros(36,1) ones(36,1)];

%Identificación de imágenes mediante redes neuronales

```

```

%Red perceptrón codificando seis salidas
net = newp(R,6);

%Simulación sin entrenamiento
Y = sim(net,P)

%Entrenamiento
net.trainParam.epochs = 160;
net = train(net,P,T);

%Simulación por separado
Y = sim(net,AJ)
Y = sim(net,AJ_Z)
Y = sim(net,CA)
Y = sim(net,HU)
Y = sim(net,MO)
Y = sim(net,MO2)

%Simulación de todos los datos
Y = sim(net,P)

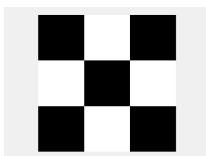
%imagen de prueba
prueba=[
    0 0 0 0 0 0;
    1 1 0 0 1 1;
    0 0 1 1 0 0;
    0 0 1 1 0 0;
    1 1 0 0 1 1;
    0 0 0 0 0 0;
    ];

figure
imshow(prueba,'InitialMagnification','fit')
PRUEBA= [prueba(:,1); prueba(:,2); prueba(:,3); prueba(:,4); prueba(:,5); prueba(:,6)];
%Prueba de la red neuronal con una imagen modificada
Y = sim(net,PRUEBA)

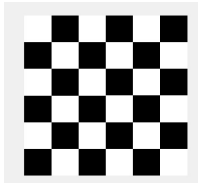
```

Imágenes de entrenamiento

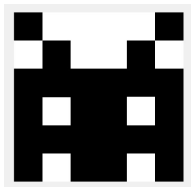
1. aj_z



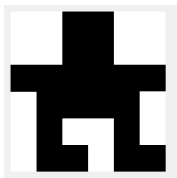
2. aj



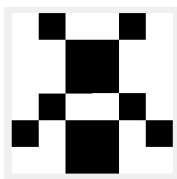
3. mo



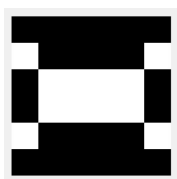
4. hu



5. mo2




6. ca



Entrenamiento de red neuronal

Training Results

Training finished: Met performance criterion 

Training Progress

Unit	Initial Value	Stopped Value	Target Value	
Epoch	0	2	160	▲
Elapsed Time	-	00:00:06	-	
Performance	0.833	0	0	▼

Training Algorithms

Data Division: Cyclical Weight/Bias Rule `trainc`

Performance: Mean Absolute Error `mae`

Calculations: MATLAB

Training Plots

Simulación sin entrenamiento

Y =

```

1      1      1      1      1      1
1      1      1      1      1      1
1      1      1      1      1      1
1      1      1      1      1      1
1      1      1      1      1      1
1      1      1      1      1      1

```

Simulación de imágenes por separado

Imagen 1

Y =

```

1
0
0
0
0
0

```

Imagen 2

Y =

0
1
0
0
0
0

Imagen 3

Y =

0
0
1
0
0
0

Imagen 4

Y =

0
0
0
1
0
0

Imagen 5

Y =

0
0
0
0
1
0

Imagen 6

Y =

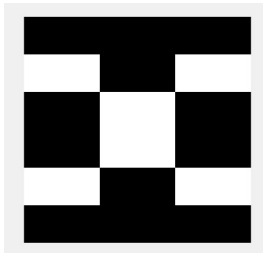
0
0
0
0
0
1

Simulación de imágenes completa

Y =

1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

Imagen de prueba



Simulación con imagen de prueba

Y =

0
0
1
1
0
1