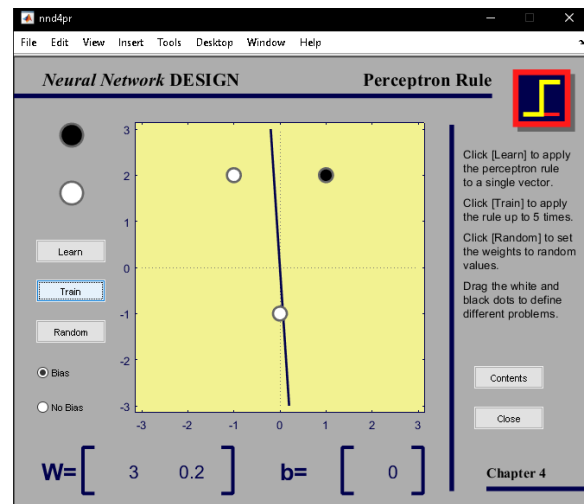
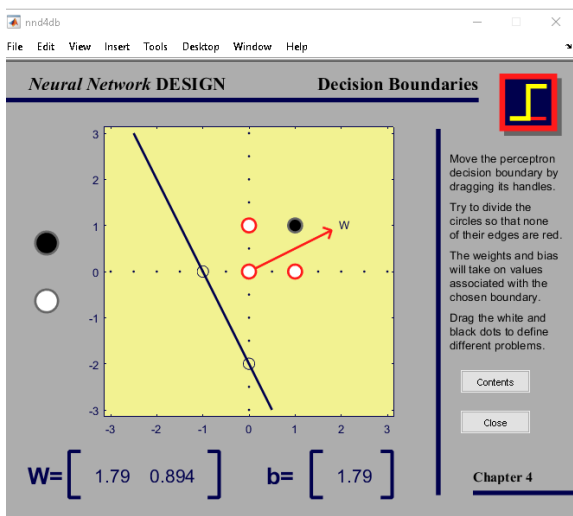
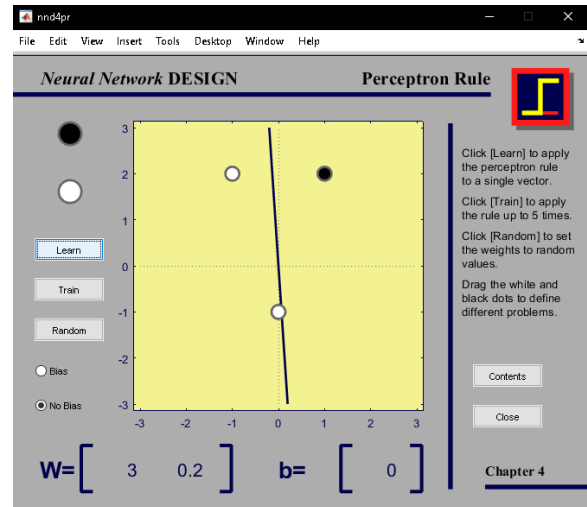
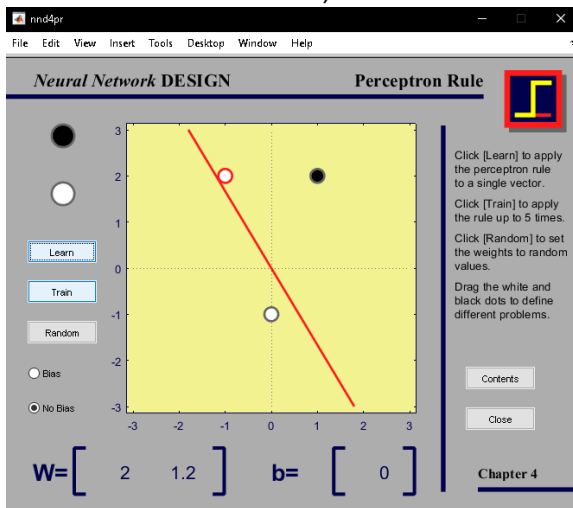
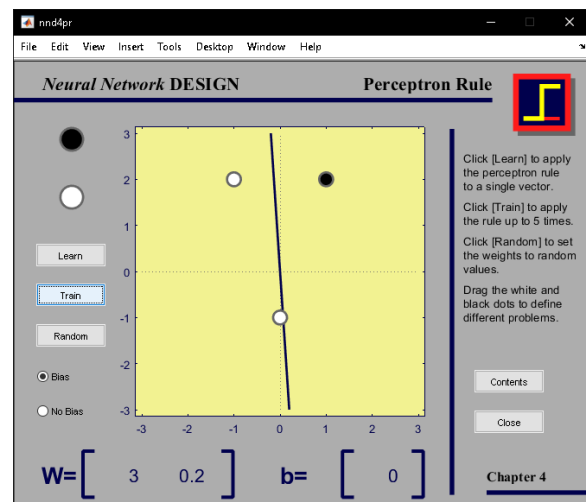


Alumno: Padilla Herrera Carlos Ignacio
Sistemas Inteligentes 2020-1

Inciso a)

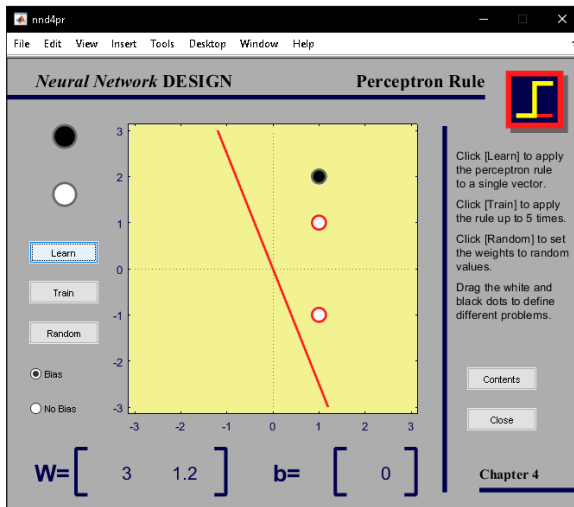


Inciso b)

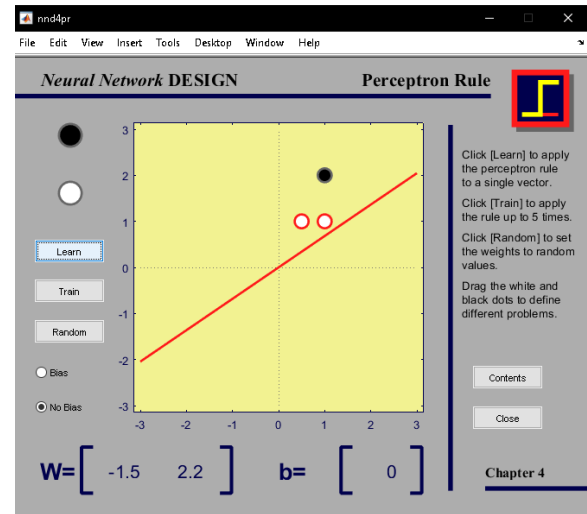


Segunda época

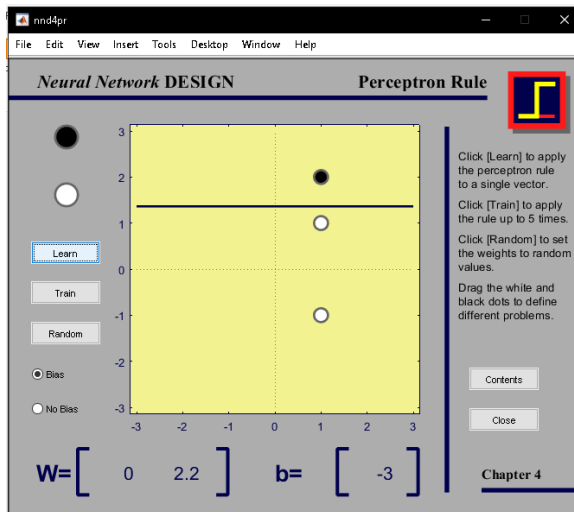
Inciso c)
Ejercicio sin bias



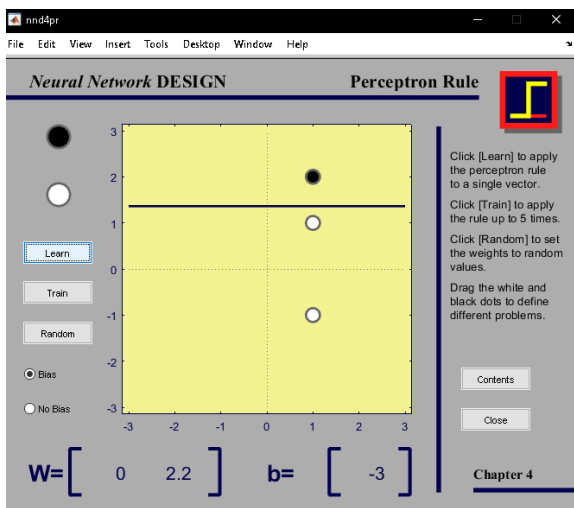
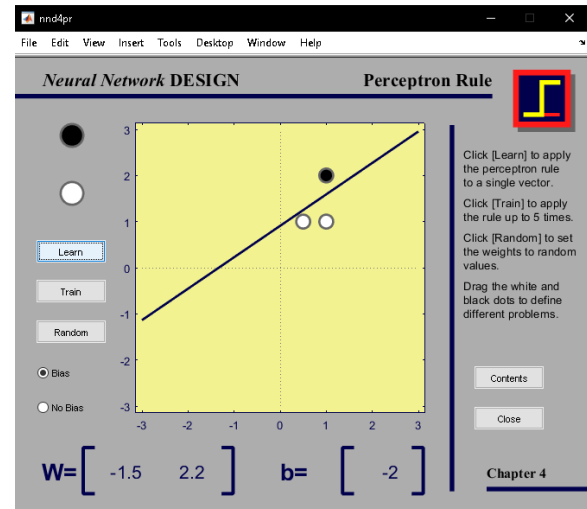
Después de 10 épocas

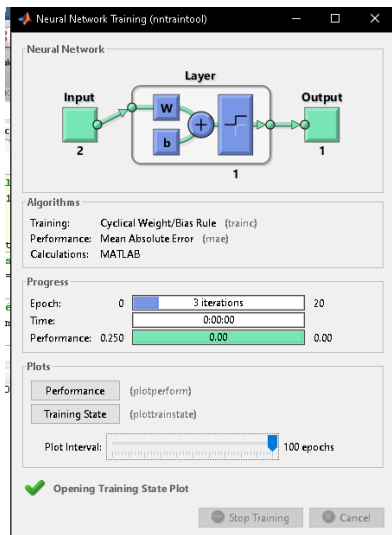


Después de 4 épocas con bias, inciso c



Después de 8 épocas,





Código en Matlab

```
% classification with a 2 input perceptron
n
clear
clc

P = [-0.5 -0.5 +0.3 -0.1; ...
    -0.5 +0.5 -0.5 +1.0];
T = [1 1 0 0];

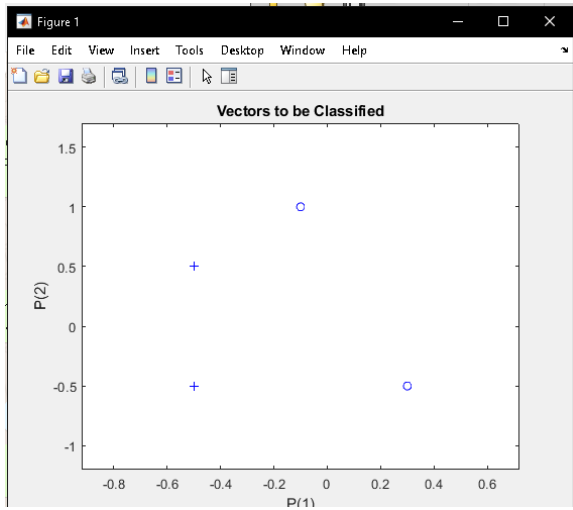
plotpv (P,T);

net = newp([-1 1; -1 1],1);
plotpv( P,T);
plotpc(net.IW{1},net.b{1});
net.adaptParam.passes = 3;
net = adapt (net,P,T);
plotpc(net.IW{1},net.b{1});

p =[0.7; 1.2];
a = sim(net,p);
plotpv(p,a);
point = findobj (gca,'type','line');
set(point, 'Color', 'red');

hold on;
plotpv(P,T);
plotpc(net,IW{1},net.b{1});
hold off;
```

Ejercicio 2



```
%% Ejercicio 2
clear all;
clc

P = [ -0.5 -0.5 +0.3 -0.1;
      -0.5 +0.5 -0.5 +1.0];
T = [ 1 1 0 0];

plotpv(P,T);

net = newp([-1 1;-1 1],1);
plotpv(P,T);
plotpc(net.IW(1),net.b(1));

net.adaptParam.passes = 3;
net = adapt(net,P,T);
%plotpc(net.IW(1),net.b(1));

p = [0.7;1.2];
a = sim(net,p);
plotpv(p,a);
point = findobj(gca,'type','line');
set(point,'Color','red');

hold on;
plotpv(P,T);
plotpc(net.IW(1),net.b(1));
hold off;
```

Conclusiones

Las redes neuronales son adecuadas para predecir cuando solamente se requiere aprender de los ejemplos, sin la

necesidad de información adicional que puede traer más confusión que efecto de predicción. Las redes neuronales son capaces de generalizar y son resistentes al ruido. No es posible generalmente determinar exactamente que ha aprendido una red neuronal y también es difícil estimar su error de predicción.