Machine Learning

Perceptrons practical

Computer & Health Informatics Department

Neuron output

PROBLEM DESCRIPTION:

Calculate the output of a simple neuron

Outline

- Define neuron parameters
- Define input vector
- Calculate neuron output
- Plot neuron output over the range of inputs

Define neuron parameters

```
close all, clear all, clc
% Neuron weights
w = [4 -2]
% Neuron bias
b = -3
% Activation function
 func = 'hardlim'
```

Define neuron parameters

```
w =
        4 -2
b =
       -3
func =
          hardlim
```

Define input vector

```
p = [2 3]
p =
     2     3
```

Calculate neuron output

```
activation_potential = p*w'+b


neuron_output = feval(func,
activation_potential)
```

Output

```
activation_potential =
-1
neuron_output =
0
```

Classification of linearly separable data with a perceptron

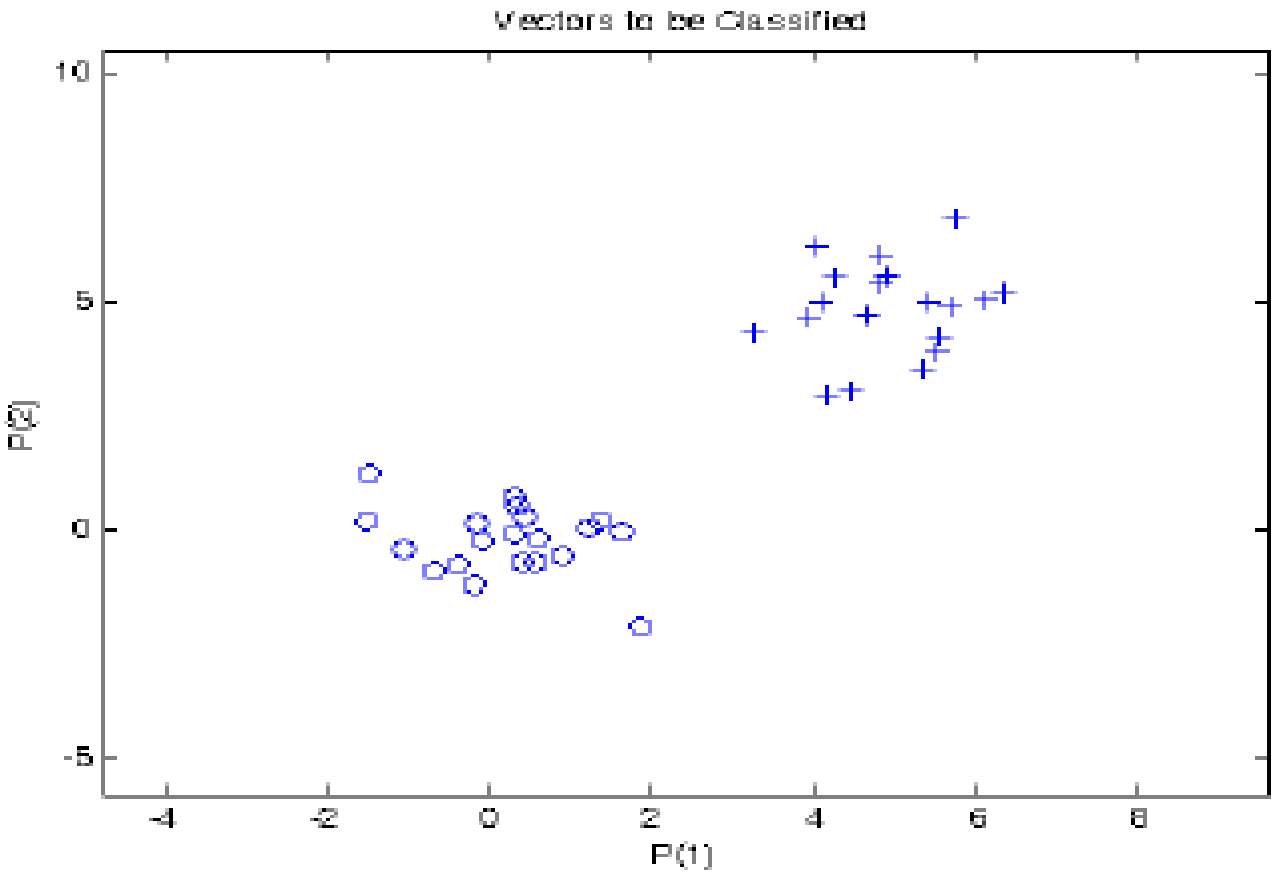
PROBLEM DESCRIPTION:

Two clusters of data, belonging to two classes, are defined in a 2-dimensional input space. Classes are linearly separable. The task is to construct a Perceptron for the classification of data.

Define input and output data

```
close all, clear all, clc
% number of samples of each class
N = 20;
% define inputs and outputs
offset = 5; % offset for second class
x = [randn(2,N) randn(2,N)+offset]; % inputs
y = [zeros(1,N) ones(1,N)]; % outputs
% Plot input samples with PLOTPV (Plot
perceptron input/target vectors)
figure(1)
plotpv(x,y);
```

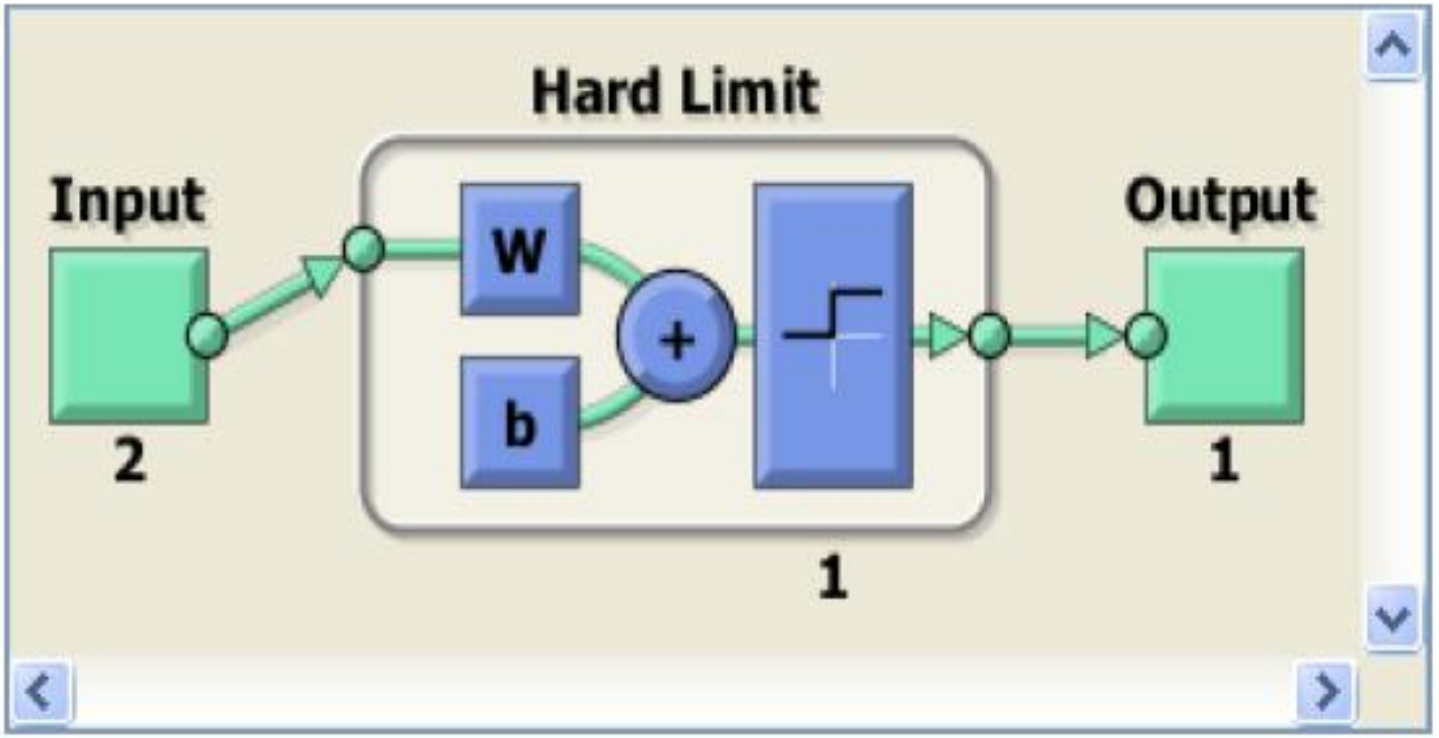
Define input and output data



Create and train perceptron

```
net = perceptron;
net = train(net,x,y);
view(net);
```

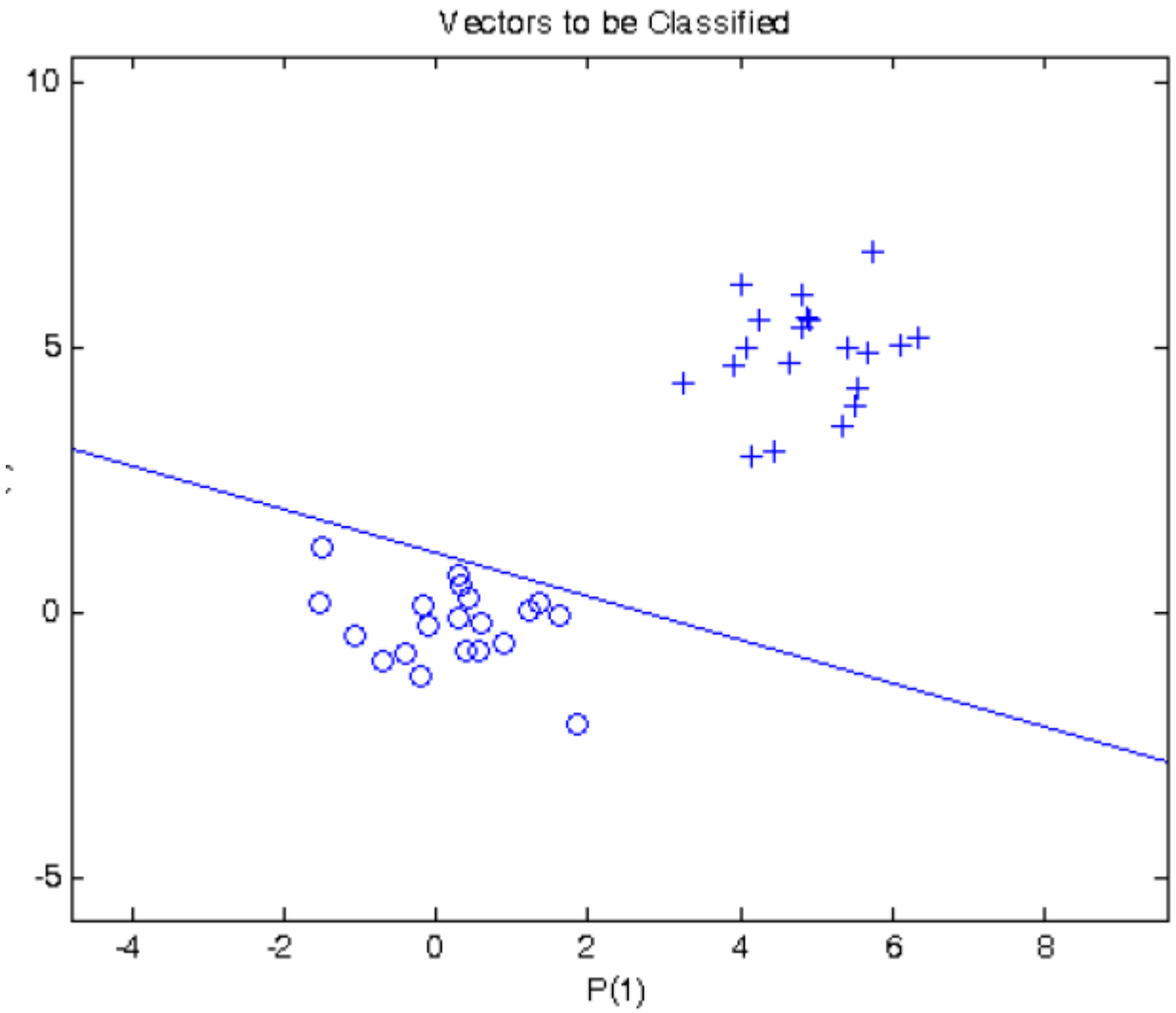
Create and train perceptron



Plot decision boundary

```
figure(1)
plotpc(net.IW{1},net.b{1});
```

Plot decision boundary



The End

?