

Aulas 2/3 – NN Tool box (Perceptrão e Rede Neuronal)

- **perceptron: cria uma rede neuronal tipo *perceptrão***

nome_rede = perceptron

- Por defeito, a função de ativação é a *hardlim* e a função de treino é a *learnp* (podem ser indicadas alternativas utilizando os argumentos opcionais da função *perceptron*)

- **feedforwardnet: cria uma rede neuronal tipo *feedforward***

nome_rede = feedforwardnet(num.hidden layers, training algorithm)

Por defeito, cria uma rede neuronal com uma camada escondida com 10 nós (a arquitetura por defeito pode ser alterada utilizando os argumentos opcionais da função);

- Os inputs e outputs não são indicados neste ponto. A sua dimensão será automaticamente configurada mais tarde durante o processo de treino (também podem ser configurados explicitamente através da função **configure**);
- Funções de ativação por defeito: camadas escondidas (*tansig*) e saída (*purelin*);
- Algoritmo de treino: *trainln*.

- **Número máximo de ciclos de treino**

net.trainParam.epochs = 100

- **Função de activação da camada**

net.layers{k}.transferFcn = 'tansig'

- **Função de Treino (traingdx = gradient descent)**

net.trainFcn = 'traingdx'

- **Percentagem de exemplos de treino e teste**

- neste caso, 100% dos exemplos são usados no treino:

- net.divideFcn = ''

- neste caso, é possível indicar a % dos exemplos que são usados no treino/teste:

- net.divideFcn = 'dividerand'

- net.divideParam.trainRatio = 0.70

- net.divideParam.valRatio = 0.15

- net.divideParam.testRatio = 0.15

- **train: treina a rede neuronal**

nome_rede = train(nome_rede, input, target)

- **sim: testa/simula a rede neuronal**

out = sim(nome_rede, input)

- **view: visualizar a rede neuronal**

view(nome_rede)

MATLAB HELP: Usar “HELP nome da função” e/ou “HELP nnnetwork”

net.IW (doc IW)

This property defines the weight matrices of weights going to layers from network inputs. It is always an NI x Ni cell array, where NI is the number of network layers (net.numLayers), and Ni is the number of network inputs (net.numInputs).

net.b (doc b)

This property defines the bias vectors for each layer with a bias. It is always an NI x 1 cell array, where NI is the number of network layers

Plotpv

Plot perceptron input/target vectors

plotpv(P,T) P=input vectos; T=target vector

Plots column vectors in P with markers based on T

Plotpc

Plot classification line on perceptron vector plot

plotpc(W,B) takes these inputs (weights, bias) WS x R weight matrix (R must be 3 or less)BS x 1 bias vector and returns a handle to a plotted classification line.
