Tutorial
An Introduction to the Use of Artificial Neural Networks.
Part 4: Examples using Matlab

Dra. Ma. del Pílar Gómez Gíl
INAOE
pgomez@inaoep.mx
pgomez@acm.org

This version: October 13, 2015

Outline

Duration	Topic	Sub-topics
1 hour	1. Artificial Neural Networks. What is that?	1.1 Some definitions. 1.2 Advantages and drawbacks of ANN. 1.3 Characteristics of solutions using ANN. 1.4 The fundamental neuron. 1.5 The concept of “learning” by examples
1 hour	2. Basic architectures	2.1 Types of ANN 2.2 Single layer perceptron network 2.3 Multi-layer Perceptrons
1 hour	3. Solutions using ANN	3.1 ANN as classifiers 3.2 ANN as a function approximator. 3.3 ANN as predictors
1 hour	4. Examples using Matlab ANN toolbox	4.1 A very simple classifier. 4.2 A very simple function approximator.

Basic Examples

- In this last part of the tutorial, we present a basic way to use a multi-layer perceptron as a classifier and as a function aproximator.
- Examples are shown using the “ANN Matlab toolbox”

Matlab and ANN

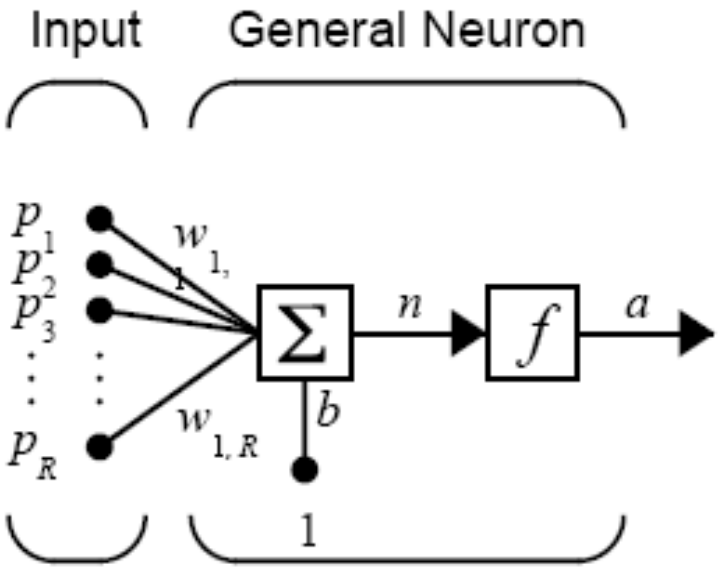
- Matlab® is a language of technical computing, developed by Mathworks™
- The “Neural Network toolbox” is part of the products offered for Statistics and Data Analysis (see figure next), used to design and simulate neural networks, based on Matlab®
- This tool is very popular. It offers support for several architectures of NN and a GUI to design and maintain a neural net.
- A detailed description of this toolbox may be found at http://www.mathworks.com/products/neuralnet/index.html?ref=pfo

Source: http://www.mathworks.com/products/pfo/

Basic Concepts to use the toolbox

- There are 4 main steps to do:
  1. Get the data to be used
  2. Define the network
  3. Train the network
  4. Using and assessing the performance of the network
- After that, the network is ready to be used.

Matlab representation of a perceptron

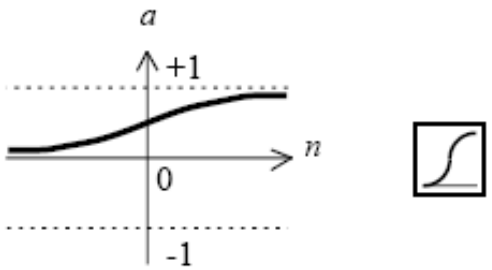


Where...

R = Number of elements in input vector

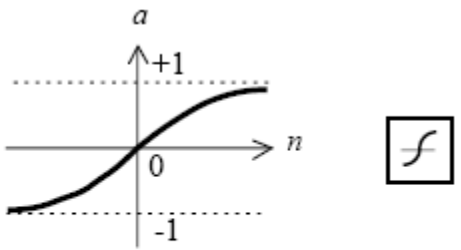
[Demuth &Mark Beale 2001]

Popular transfer function used at toolbox



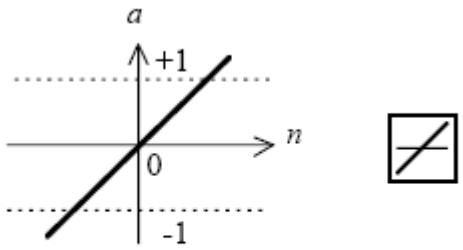
$a = logsig(n)$

Log-Sigmoid Transfer Function



$a = tansig(n)$

Tan-Sigmoid Transfer Function



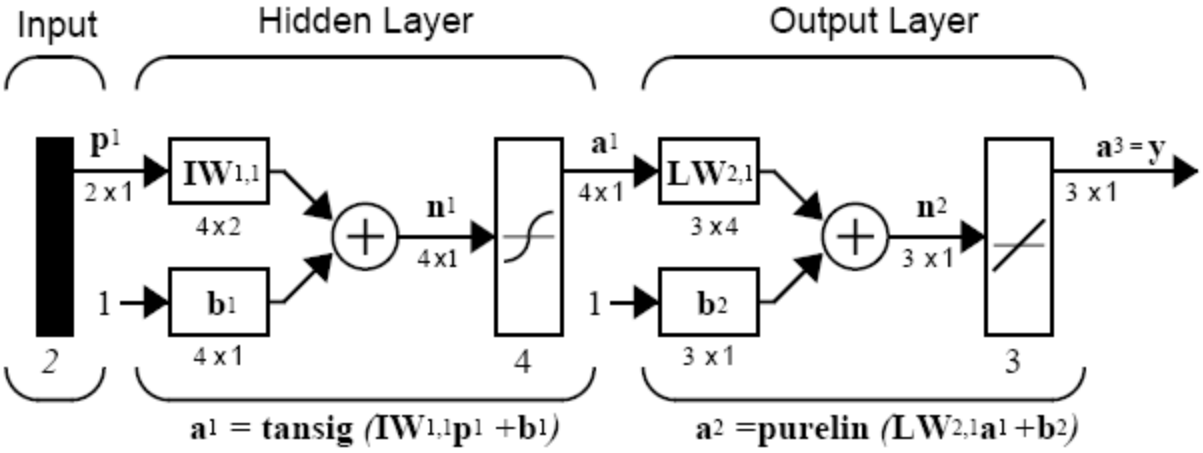
$a = purelin(n)$

Linear Transfer Function

[Demuth &Mark Beale 2001]

NN Toolbox representation of a MLP

- There is one hidden and one output layer



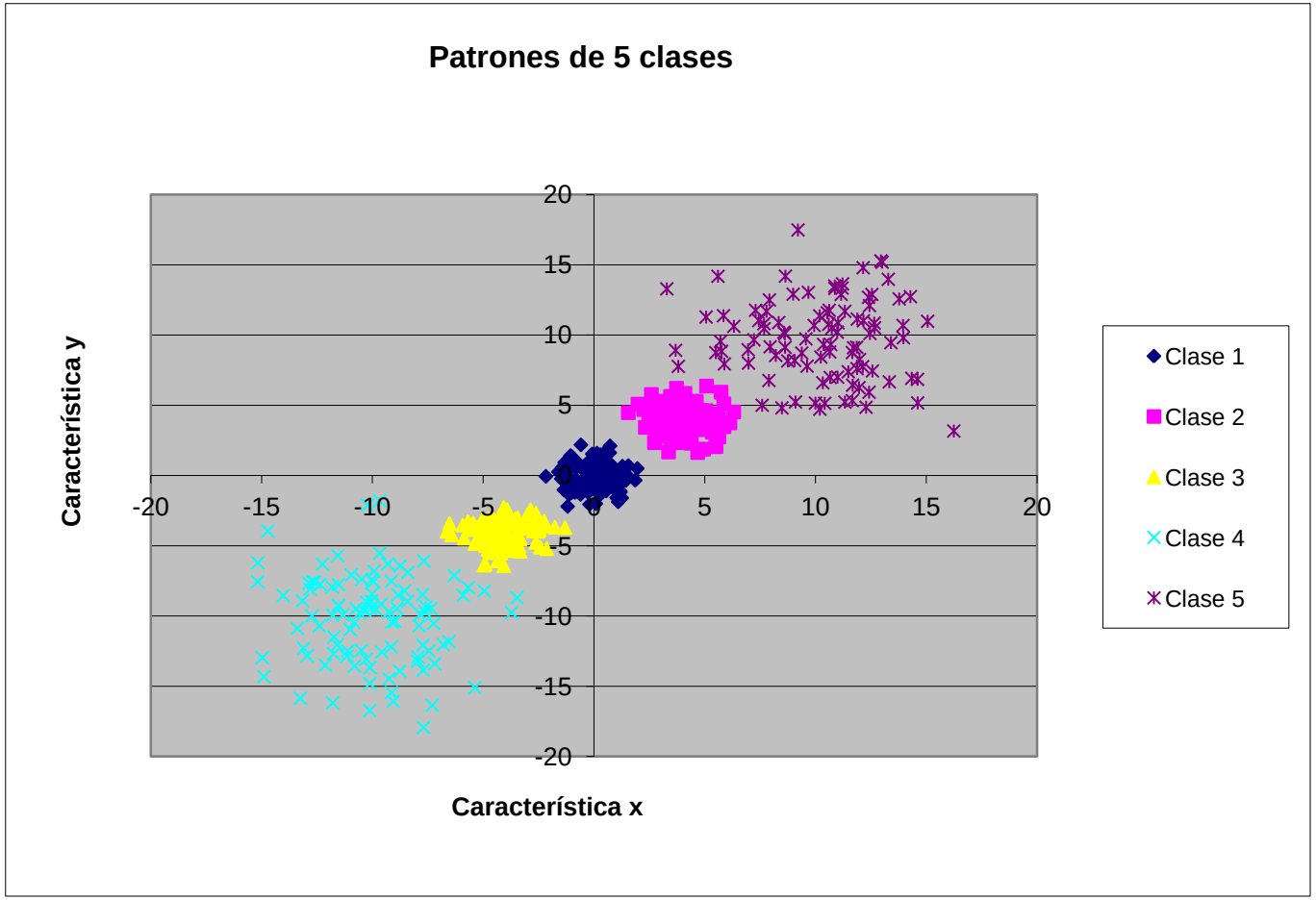
- See code classifierv4.m

A very simple classifier

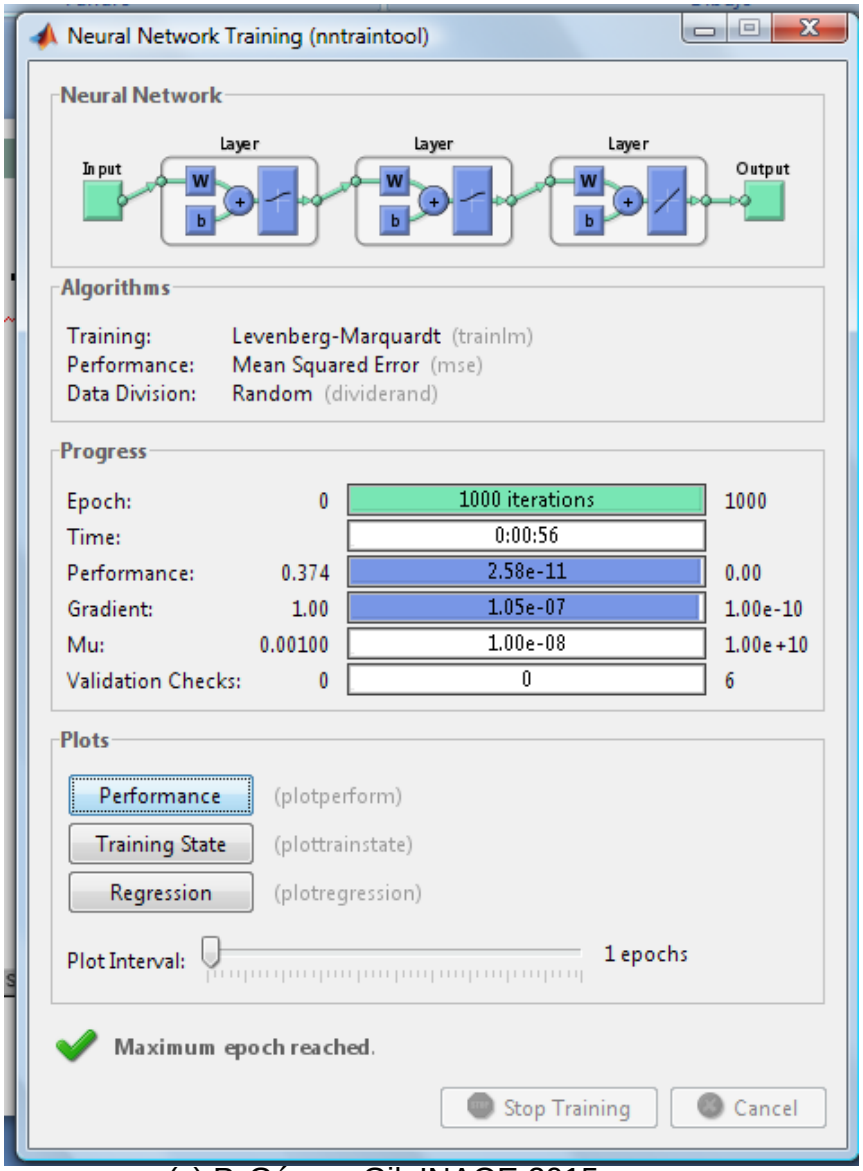
Classifying 5 types of 2-D points

CLASE	MU	SIGMA
1	0	1
2	4	1
3	-4	1
4	-10	3
5	10	3

Classifying 5 types of 2-D points (cont.)



Classifying 5 types of 2-D points (cont.)



Classifying 5 types of 2-D points (cont.)

Confusion matrix using **training data** (expected,real)

confusion =

20	0	0	0	0
0	20	0	0	0
0	0	20	0	0
0	0	0	20	0
0	0	0	0	20

Performance: 100.00 %
Error: 0.00 %

Confusion matrix using **testing data** (expected,real)

confusion =

6	0	0	0	0
0	6	0	0	0
0	0	6	0	0
0	0	1	5	0
0	0	0	0	6

Generalization performance: 96.67 %
Error: 3.33 %

A very simple function approximator

Approximating a function
[Mendoza & Gómez-Gil 2009]

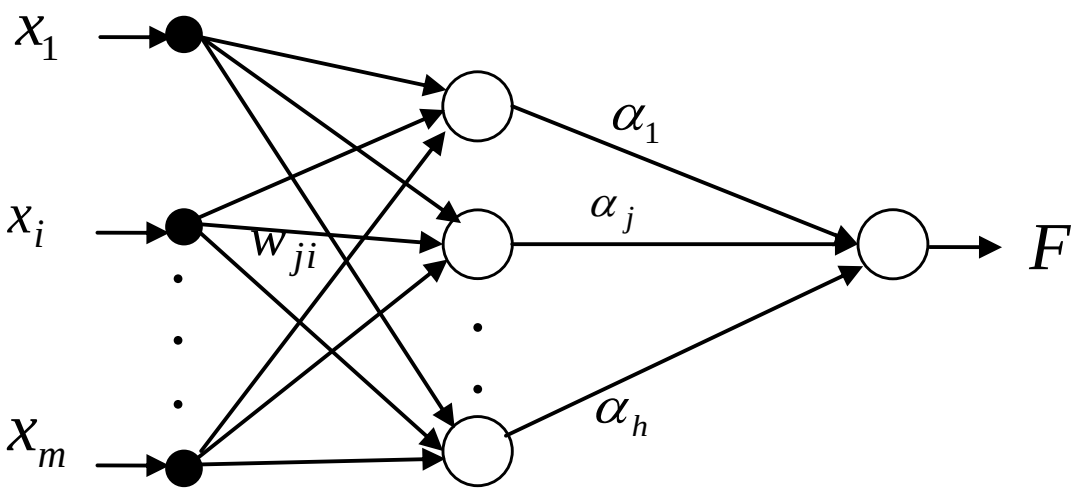
- A MLP with *m* inputs, *one* hidden layer with *h* neurons and *one* output layer with *one* neuron is able to approximate a function $f(x_1,...,x_m)$

$$F(x_1, x_2 ... x_m) = \sum_{j=1}^{h} \alpha_j \tau(\sum_{i=1}^{m} w_{ji} x_i + b_j)$$

Approximating a function (cont.)

Activation function for hidden nodes:

$$\tau(u) = \frac{1}{1+e^{-\lambda u}}$$

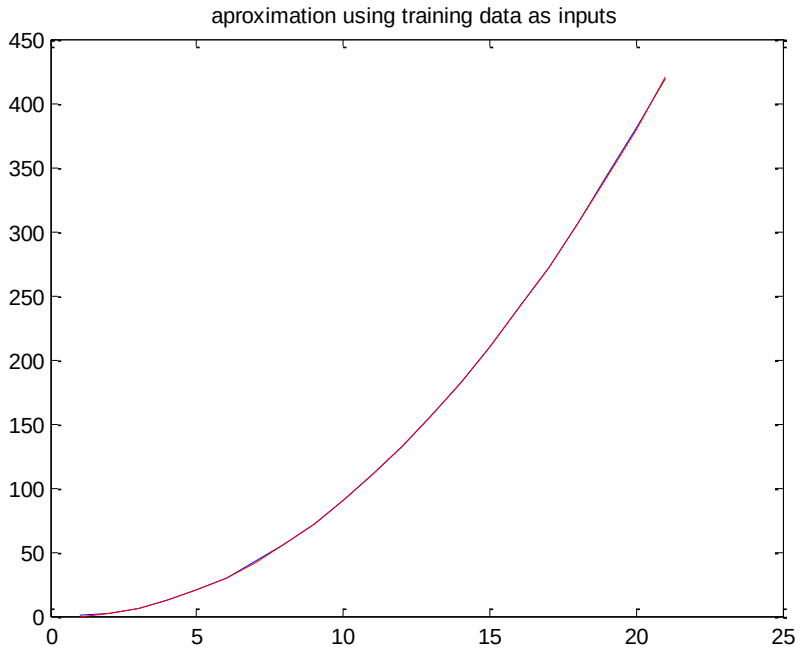


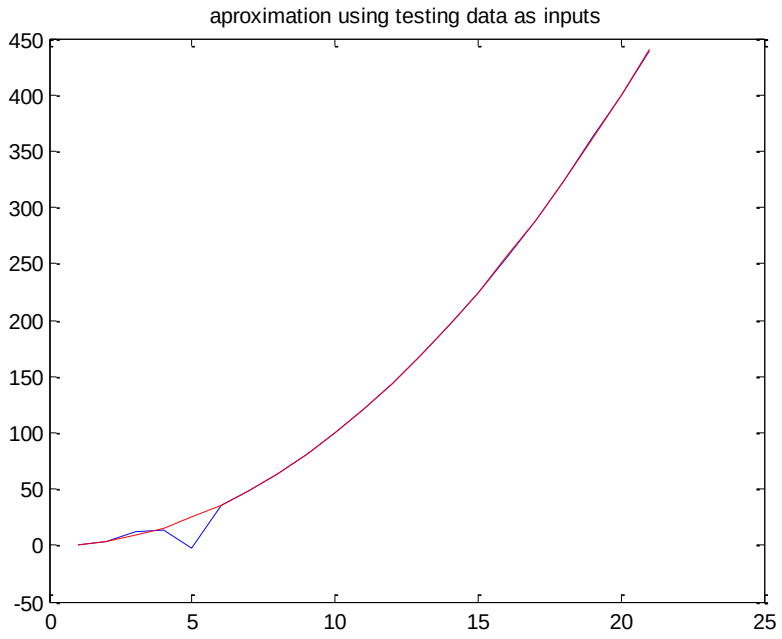
Activation function for node in output layer:

$$\mu(x) = x$$

Example: learning a parabolic function

See program `funaprox.m`





References

- *Howard Demuth &Mark Beale.* Neural Networks toolbox to use with Matlab®. User Guide, Version 4. Mathworks, 2001
- Mendoza Velázquez A, Gómez-Gil P. **“**Herramientas para el Pronóstico de la Calificación Crediticia de las Finanzas Públicas Estatales en México: Redes Neuronales Artificiales, Modelo Probit Ordenado y Análisis Discriminante**”** *Premio Nacional Bolsa Mexicana de Valores 2009.* Disponible en:

http://www.mexder.com/inter/info/mexder/avisos/Herramientas%20para%20el%20pronostico%20de%20la%20Calificacion%20Crediticia.pdf

Thank you!

Dra. Ma. del Pilar Gómez Gil
pgomez@inaoep.mx
pgomez@acm.org
ccc.inaoep.mx/~pgomez