



índice

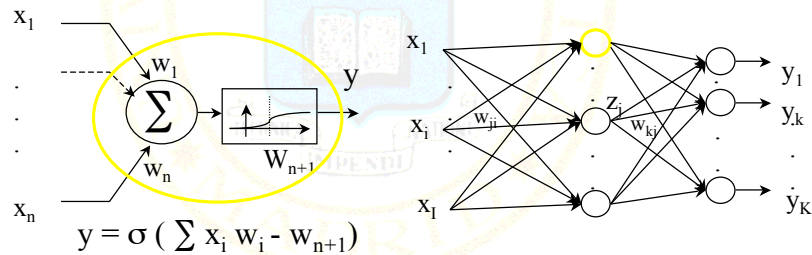
▪ *RN Supervisadas*

- Introducción
- El Perceptrón y la estructura multicapa MLP
- El aprendizaje retropropagado: BP
- Aplicaciones y ejemplos
- Características y limitaciones



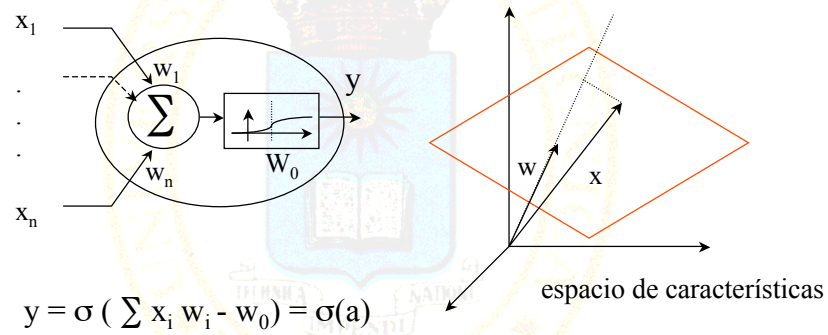
Red Neuronal Artificial

- *“Red de unidades **simples** , **adaptativas** e **interconectadas** entre sí, con capacidad de procesamiento en paralelo, cuyo objetivo es interactuar con su entorno de forma similar a las redes neuronales naturales”*



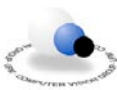


Funcionamiento del Perceptrón

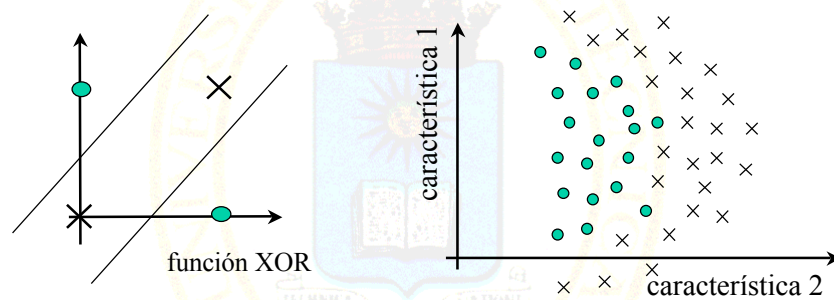


$$y = \sigma \left(\sum x_i w_i - w_0 \right) = \sigma(a)$$

espacio de características

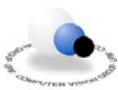
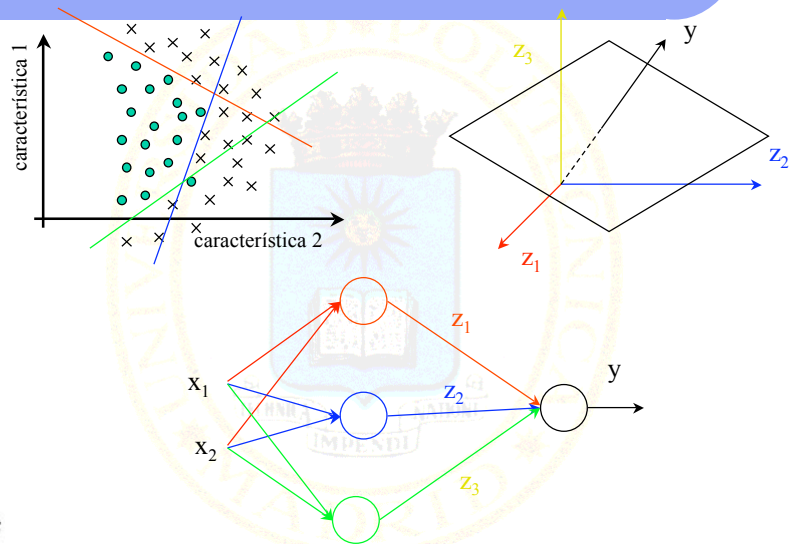


El perceptrón ante fronteras interclase no lineales





Perceptrón Multicapa (MLP)



Consideraciones matemáticas sobre el MLP

- Un MLP de dos capas puede representar cualquier función lógica con frontera convexa.
- Un MLP de tres capas puede representar cualquier función lógica con frontera arbitraria.

Un MLP de dos capas puede aproximar cualquier función continua con una precisión arbitraria.





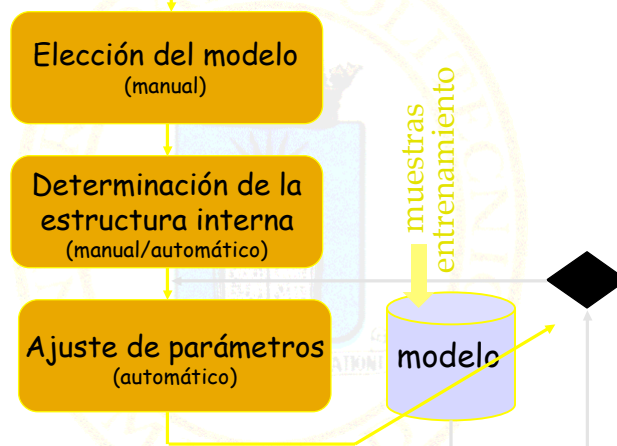
índice

▪ *RN Supervisadas*

- Introducción
- El Perceptrón y la estructura multicapa MLP
- El aprendizaje retropropagado: algoritmo BP
- Aplicaciones y ejemplos
- Características y limitaciones



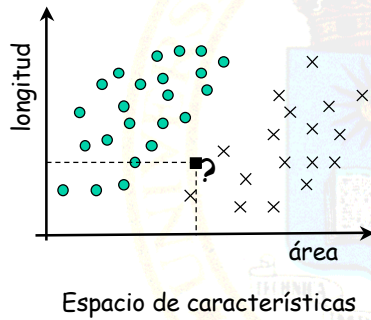
Metodología: Niveles de aprendizaje



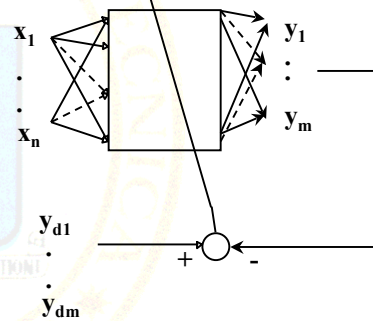


Esquema de funcionamiento: aprendizaje supervisado

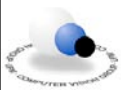
Concepto aprendizaje supervisado



Estructura de funcionamiento supervisado



Generalización de funciones de $R^n \Rightarrow R^m$



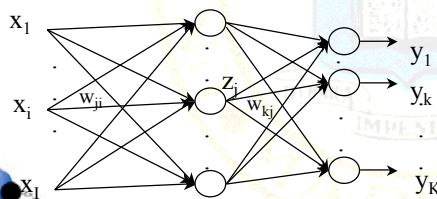
Aprendizaje: Algoritmo de Retro-propagación

$$E = 1/2 \sum_k (y_k^n - y_{dk}^n)^2 = 1/2 \sum_k (y(w_{kj}, w_{ji}, x_i)^n - y_{dk}^n)^2$$

$$\frac{\partial E}{\partial w_{kj}} = (y_k - y_{dk}) z_j$$

$$\frac{\partial E}{\partial w_{ji}} = \sum_k (y_k - y_{dk}) \frac{\partial y_k}{\partial z_j} \frac{\partial z_j}{\partial w_{ji}} = \sum_k (y_k - y_{dk}) w_{kj} \frac{\partial z_j}{\partial a_j} x_i$$

$$\frac{\partial E}{\partial w_{ji}} = \sum_k (y_k - y_{dk}) w_{kj} z_j (1 - z_j) x_i$$



$$y_j = \frac{1}{1 + e^{-a_j}} \Rightarrow \frac{\partial y_j}{\partial a_j} = \frac{e^{-a_j}}{(1 + e^{-a_j})^2} = (1 - y_j) y_j$$

$$y_j = \tanh(a_j) \Rightarrow \frac{\partial y_j}{\partial a_j} = 1 - y_j^2$$





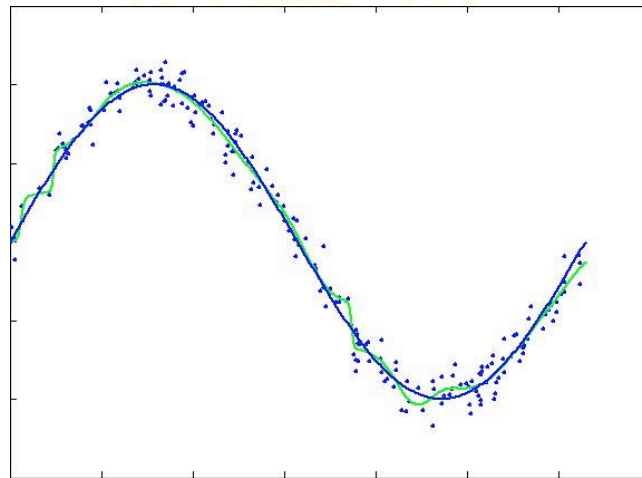
índice

▪ *RN Supervisadas*

- Introducción
- El Perceptrón y la estructura multicapa MLP
- El aprendizaje retropropagado: algoritmo BP
- Aplicaciones y ejemplos
- Características y Limitaciones



Ejemplos MLP: Generalizador de Funciones

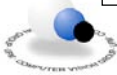




Ejemplos MLP: Generalizador de Funciones

```
% datos de entrenamiento
xe=linspace(0,2*pi,50);
for i=1:numap
yd(i)=sin(xe(i))+normrnd(0,0.1);
end
% datos de test
numtest=500;
xt=linspace(0,2*pi,numtest);
yt=sin(xt);

%creacion MLP
net = newff(minmax(xe),[10 1],{'tansig' 'purelin'},'trainlm');
% entrenamiento MLP
[net,tr]=train(net,xe,yd);
%respuesta
anst=sim(net,xt); errortest=mse(yt-anst)
```



Ejercicio de MLP como Generalizador de Funciones

Teniendo como datos de aprendizaje:

$$yd(i)=\sin(xe(i))+\text{normrnd}(0,0.1);$$

Estudiar la influencia sobre el resultado (gráfico y cuantificación de los errores de test y de aprendizaje) de los siguientes factores:

1. número de muestras de aprendizaje
2. orden de las muestras de aprendizaje
3. número de neuronas utilizadas
4. número de épocas de aprendizaje





Otros ejemplos MLP:

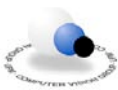
- **Generalizador de funciones:**

- Ej.:

$$y_1 = 2x_1x_2 + x_3^2$$

$$y_2 = x_2^2x_3$$

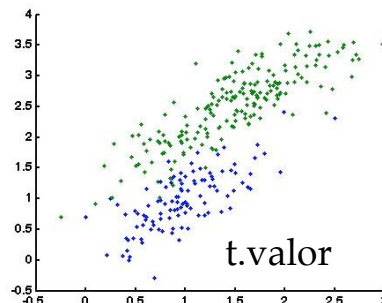
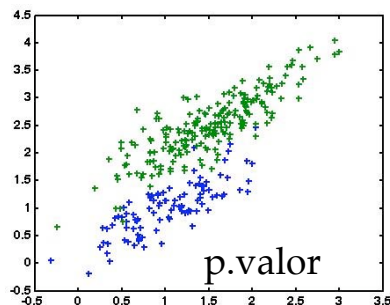
Estudiar esta generalización en función del número de neuronas y del número de muestras



MLP como clasificador

- La función de salida es un discriminante binario:

load `datos2D_clasificadosen2.mat`



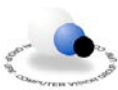


Ejercicio de MLP como clasificador

Teniendo como datos de aprendizaje del ejemplo anterior

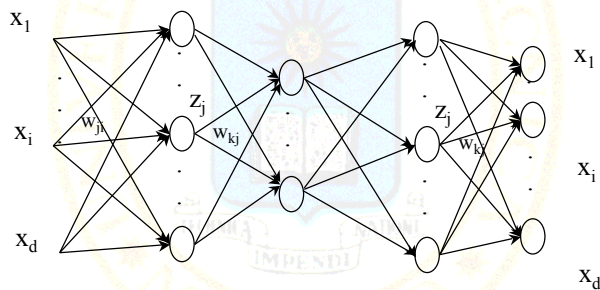
Estudiar la influencia sobre el resultado (gráfico y cuantificación de los errores de test y de aprendizaje) de los siguientes factores:

1. número de muestras de aprendizaje
2. orden de las muestras de aprendizaje
3. número de neuronas utilizadas
4. número de épocas de aprendizaje



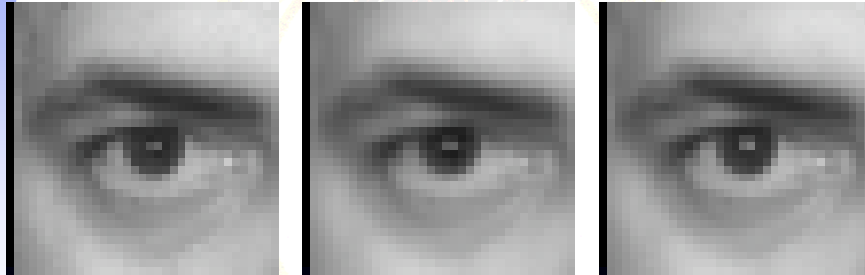
MLP para reducción dimensional no lineal: auto-encoder

- **La función de salida es la propia entrada y existe una capa intermedia con menos neuronas que la $\dim(x)$**





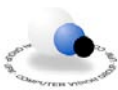
Ejemplo MLP para compresión



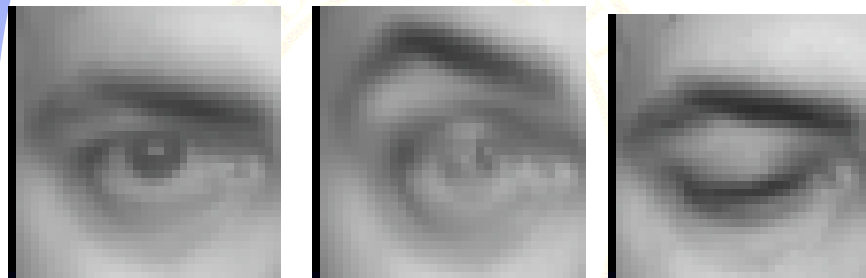
original

PCA 5

PCA 25 - MLP 5



Ejemplo MLP para síntesis



1 D (prueba 1)

1 D (prueba 2)

1 D (prueba 3)
escalonado

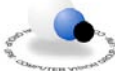


codigo Matlab del autoencoder

```
% Procesamiento con una MLP para compresión (salida=entrada)
net=newff(minmax(p_entr),[floor((Dim+ndimred)/2),ndimred,floor((Dim+ndimred)/2),Dim],{'tansig' 'purelin' 'tansig' 'purelin'},
'trainlm');
[net,tr]=train(net,p_entr,p_entr);

% Creación de una red mitad de la anterior que comprime los datos
netcompr=newff(minmax(p_entr),[floor((Dim+ndimred)/2),
ndimred],{'tansig' 'purelin'},'trainlm');
netcompr.IW{1}=net.IW{1}; netcompr.LW{2,1}=net.LW{2,1};
netcompr.b{1}=net.b{1}; netcompr.b{2}=net.b{2};

%creación de una red que descomprime los datos
netdescompr=newff(minmax(p_compr),[floor((Dim+ndimred)/2),Dim],{'tansig' 'purelin'},'trainlm');
netdescompr.IW{1}=net.LW{3,2}; netdescompr.LW{2,1}=net.LW{4,3};
netdescompr.b{1}=net.b{3}; netdescompr.b{2}=net.b{4};
```



Otros ejemplos MLP:

- **Reductor de dimensionalidad:**

- Ej.:

$$x_4 = 2x_1x_2 + x_3^2$$

$$x_5 = x_2^2x_3$$





índice

▪ *RN Supervisadas*

- Introducción
- El Perceptrón y la estructura multicapa MLP
- El aprendizaje retropropagado: BP
- Aplicaciones
- Características y limitaciones



Limitaciones del Perceptrón Multicapa

- Aprendizaje mediante minimización de funciones altamente no lineales.
 - mínimos locales
 - convergencia lenta
- Sobreaprendizaje
- Extrapolación a zonas no aprendidas

