



C261-69 Tópicos Avanzados: Redes Neuronales Artificiales

Conceptos básicos

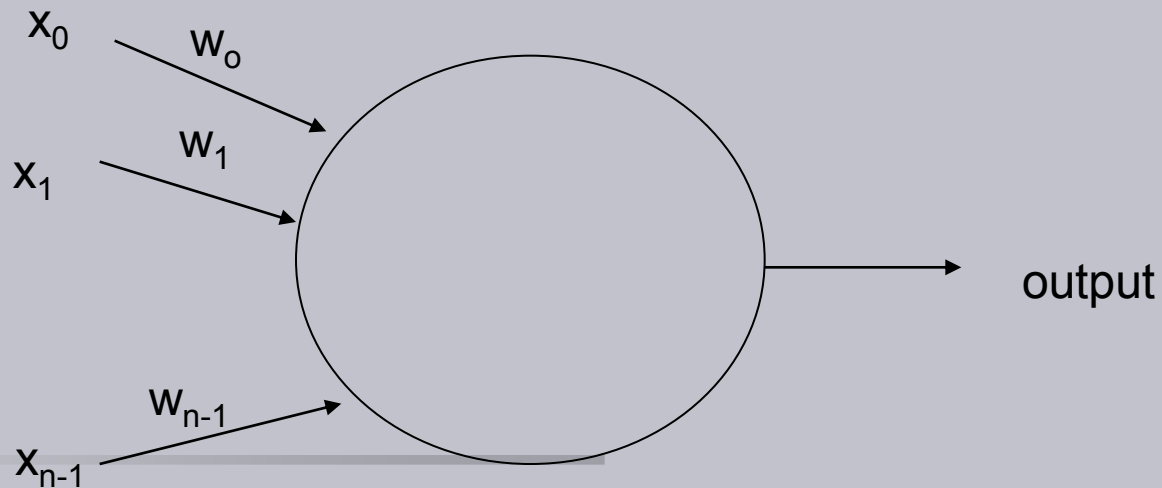
*Dra. Ma. del Pilar Gómez Gil
Primavera 2009*

pgomez@acm.org

V:13-01-09

(c) P. Gómez-Gil, INAOE 2009

Modelo Básico y Abstracto de un Neurón Artificial



$$output = F\left(\sum_{i=0}^{n-1} x_i w_i\right)$$

x 's son valores de entrada
 w 's son pesos
 F es la función de activación

El Neurón Artificial

- Es un elemento procesador al que se aplica un conjunto de entradas, cada una representando la salida de otro neurón.
- Estas entradas, que podemos llamar \mathbf{x} , se multiplican por un peso (número real) asociado a ellas, que podemos llamar \mathbf{w} ,
- Las multiplicaciones se suman y al resultado podemos llamarle *net*

$$net = \sum_{i=0}^{N-1} x_i w_i$$

Funciones de Activación

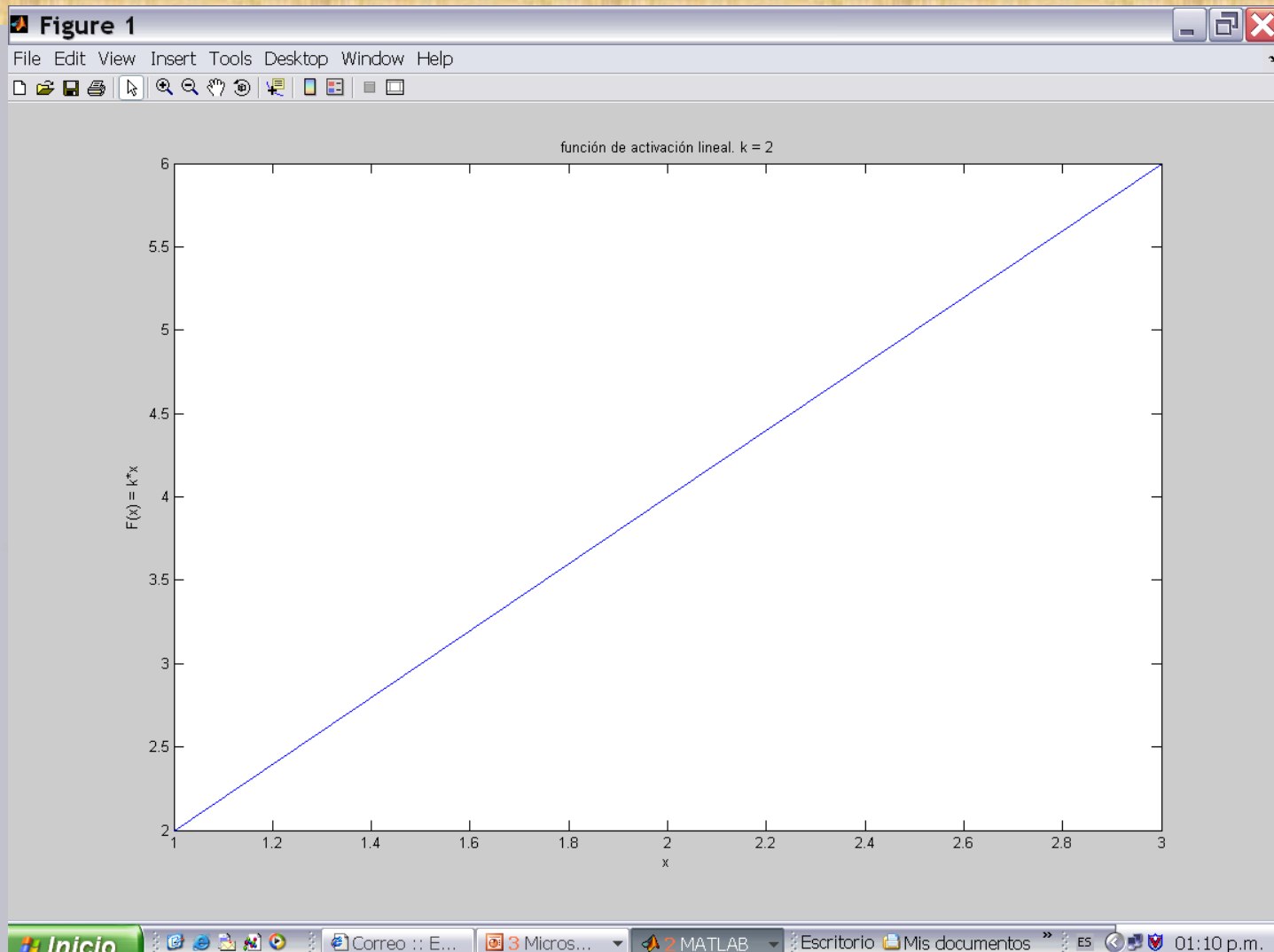
- La suma ponderada se "filtra" por medio de una función, llamada **función de activación (f)**.
- La función de activación tiene como objetivo el acotar los valores de salida del neurón para mantenerlos en ciertos rangos.
- La función de activación depende del tipo de red neuronal que se esté manejando. Hay funciones de activación lineales y no lineales

$$output = f(net)$$

Funciones de Activación Clásicas

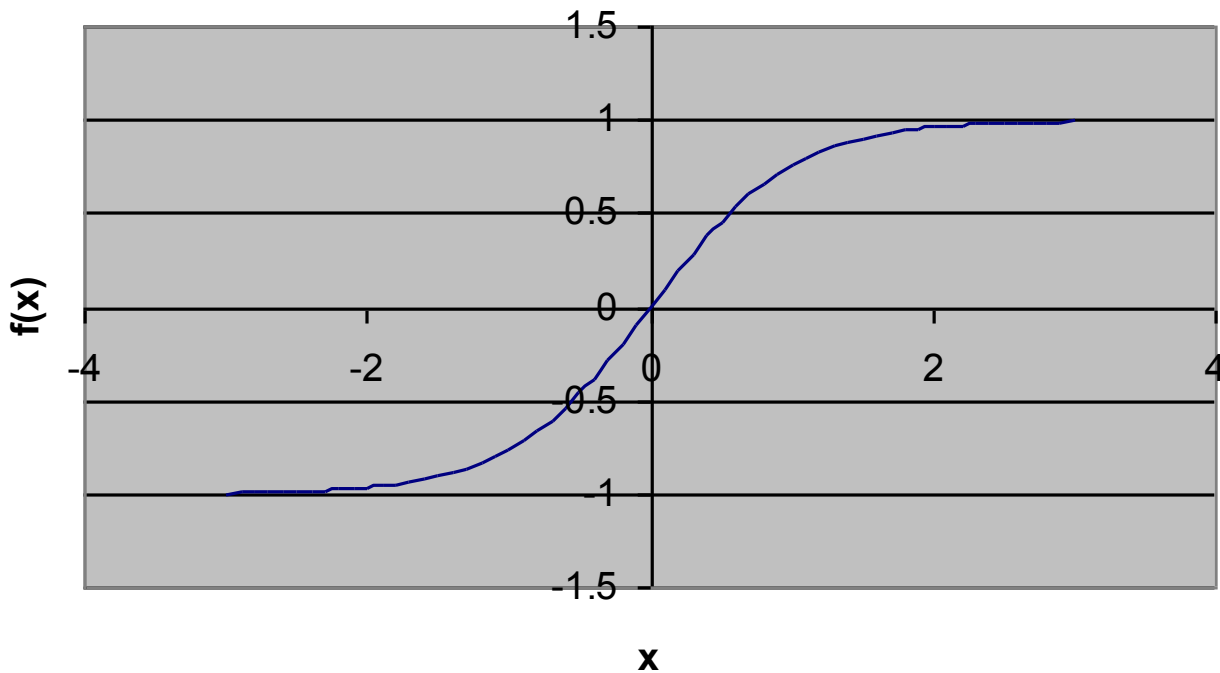
Función lineal (k es una constante)	$f(x) = k * x$
Función umbral o escalón (t es una constante)	$f(x) = \begin{cases} 1 & \text{si } x > t \\ 0 & \text{si } x \leq t \end{cases}$
Función sigmoide	$f(x) = \frac{1}{1 + \exp(-x)}$
Función tangente hiperbólica	$f(x) = \tanh(x)$

Ejemplo de una Función de Activación Lineal



Ejemplo de Función de Activación no Lineal

Función de Activación tangente hiperbólica



- Grafica en Matlab la función sigmoide



Tipos principales de redes neuronales

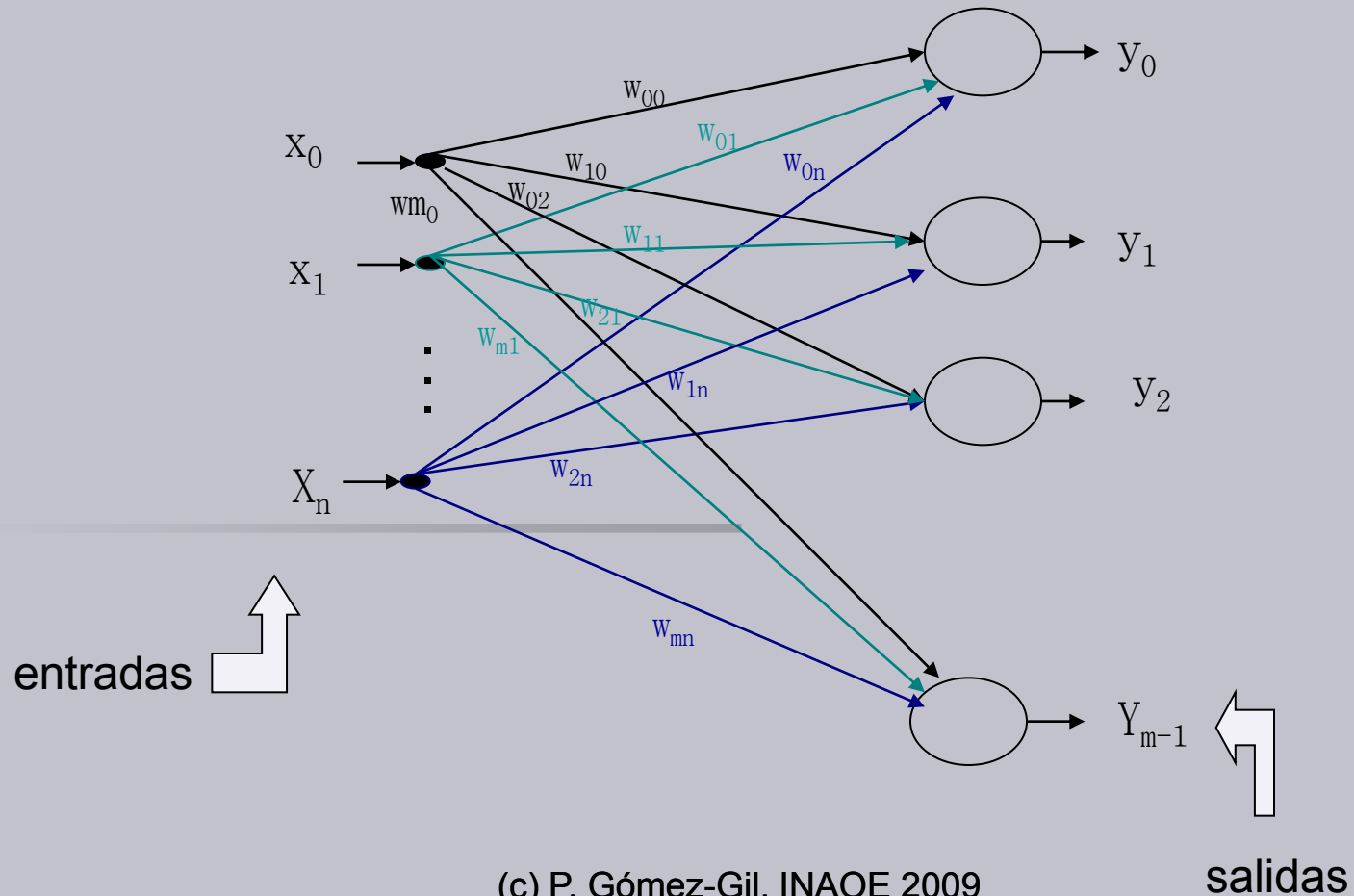
- Dependiendo de la manera en que se organizan los elementos procesadores, las RNA se pueden clasificar como:
 - De un nivel
 - De varios niveles
 - Recurrentes



Redes Neuronales de un Nivel

Son aquellas donde los elementos procesadores están organizados en una línea, recibiendo su entrada directamente del exterior y produciendo ellos directamente la salida o resultado de la RNA

Topología de Redes Neuronales de un Nivel



Representación de RNA de un Nivel

- Sea **W** una matriz de $(n+1) \times (m+1)$ que representa los pesos de una red neuronal de un nivel que tiene $(n+1)$ entradas y $(m+1)$ salidas.
- Sea **X** un vector de $(n+1)$ renglones que representa los valores de entrada a la red, y **Y** un vector de $(m+1)$ renglones que representa los valores de salida de la red
- Si **N** es el vector de los valores *net* asociados a cada neurón, tenemos que:

$$\mathbf{N} = \mathbf{X} * \mathbf{W}$$

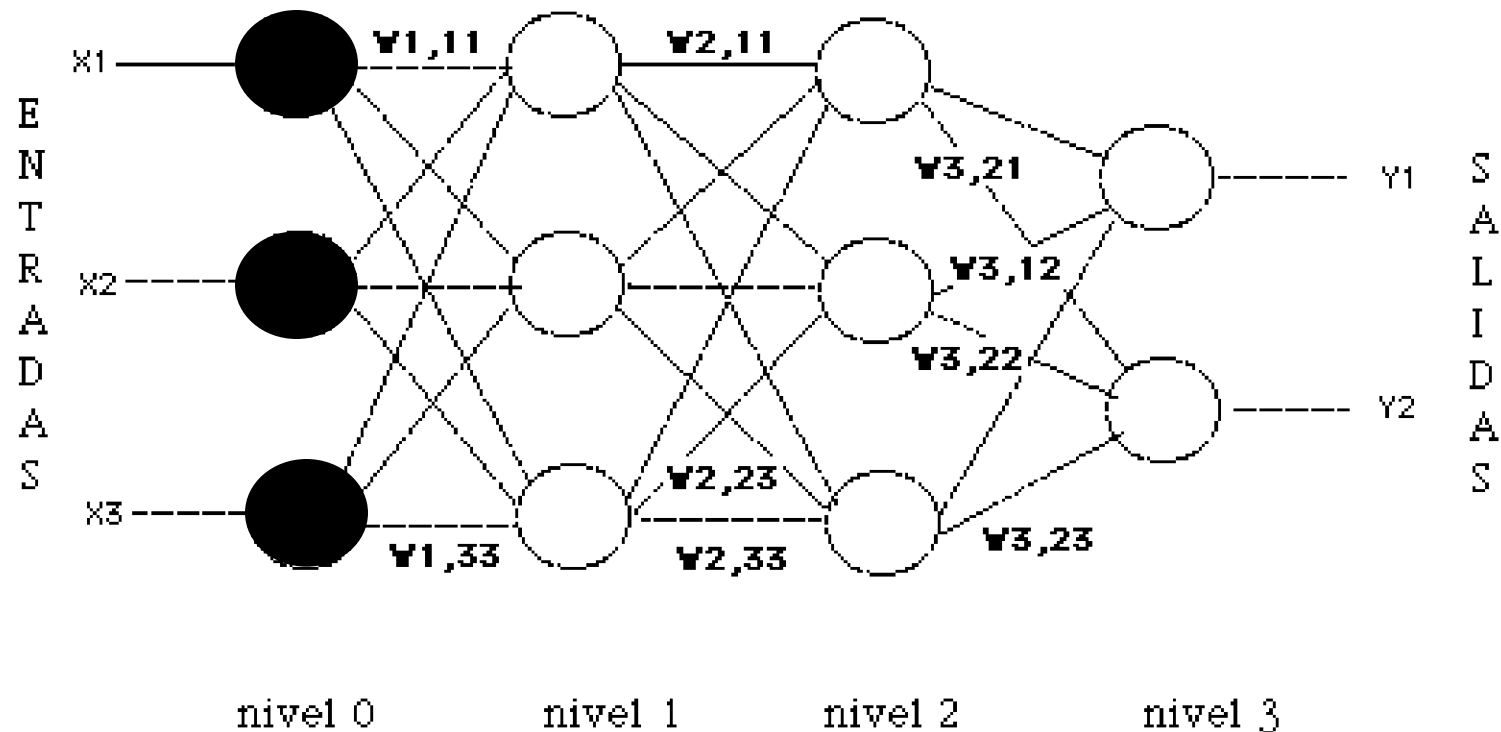
$$\mathbf{Y} = \mathbf{F}(\mathbf{N}) \text{ es la salida de la red}$$

Evaluación de una RNA de un Nivel

Por ejemplo, si la red tiene 3 entradas y 3 salidas, esta operación equivale a:

$$\mathbf{N} = \begin{matrix} & \begin{matrix} w_{00} & w_{01} & w_{02} \end{matrix} \\ \begin{matrix} x_0 & x_1 & x_2 \end{matrix} & \begin{vmatrix} w_{10} & w_{11} & w_{12} \\ w_{20} & w_{21} & w_{22} \end{vmatrix} \end{matrix}$$
$$= \begin{matrix} x_0 w_{00} + x_1 w_{10} + x_2 w_{20} & x_0 w_{01} + x_1 w_{11} + x_2 w_{21} & x_0 w_{02} + x_1 w_{12} + x_2 w_{22} \end{matrix}$$
$$y_0 = f(x_0 w_{00} + x_1 w_{10} + x_2 w_{20})$$
$$y_1 = f(x_0 w_{01} + x_1 w_{11} + x_2 w_{21})$$
$$y_2 = f(x_0 w_{02} + x_1 w_{12} + x_2 w_{22})$$

Redes Neuronales con Varios Niveles



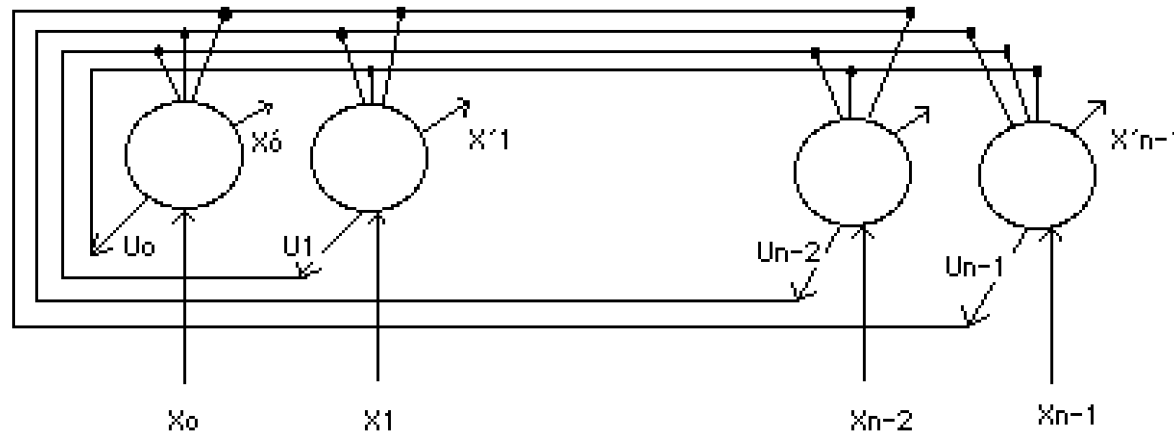
Características de RNA's con Varios Niveles

- En estas redes existen varias capas o niveles de neuronas, donde las salidas de los neuronas en el nivel i se conectan a los neuronas en el nivel $i+1$.
- Se dice entonces que este tipo de redes están "alimentadas hacia adelante" (*feed-forward networks*)
- El primer nivel de neuronas se conoce como *nivel de entrada*.
- El último nivel se conoce como *nivel de salida*.
- Los demás niveles se conocen como *niveles escondidos*.

Características de RNA's con Varios Niveles (cont.)

- La salida de este tipo de redes es la salida de los neurones que se encuentran en la última capa.
- Para determinar dicha salida, se calcula la salida de cada neurón, empezando por el primer nivel, y continuando hasta llegar al último.
- En una red con varios niveles, la función de activación debe ser no-lineal, a fin de que pueda existir representación del conocimiento en los niveles escondidos.

Redes Recurrentes

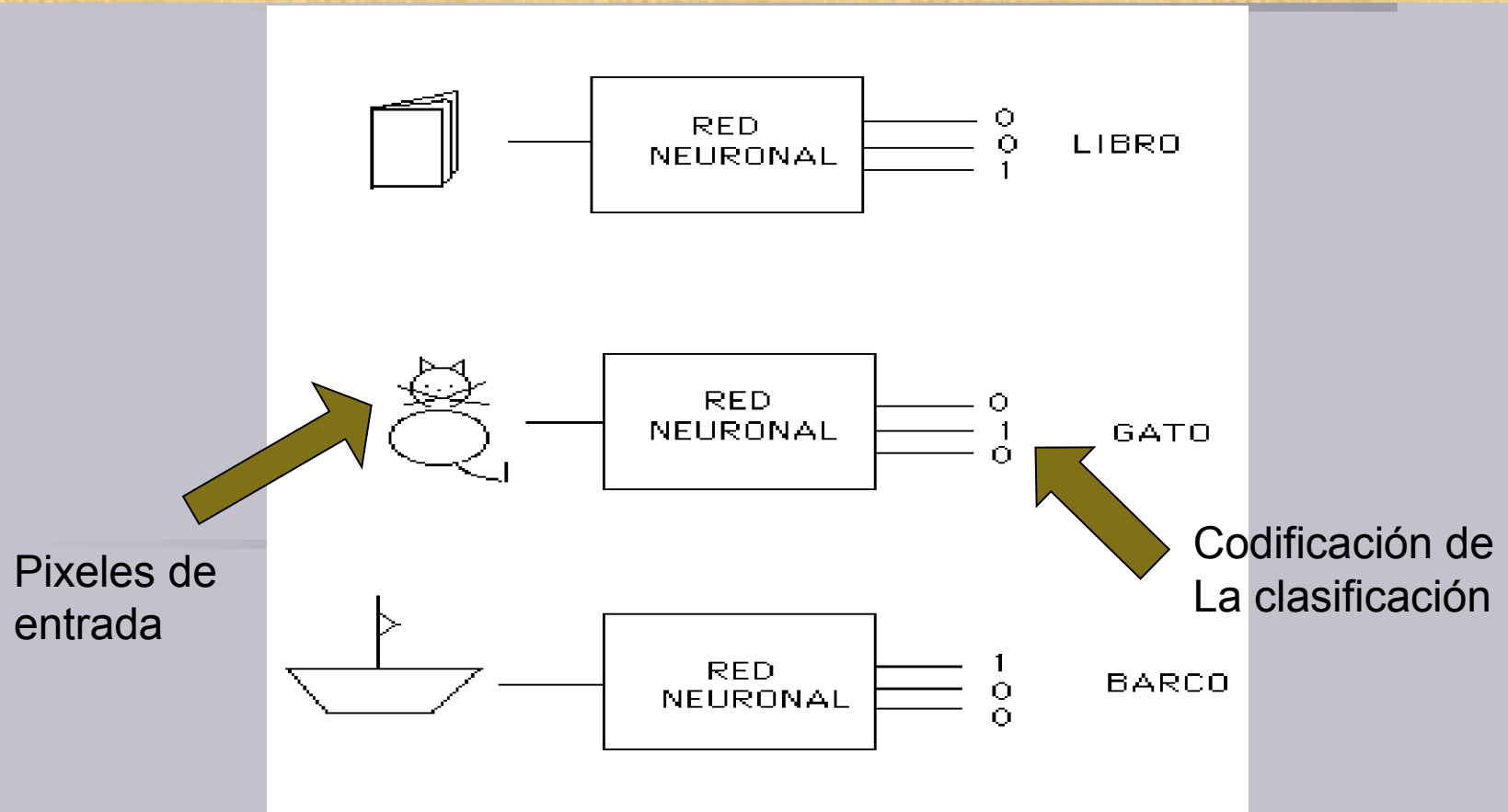


- Las redes recurrentes son una salida de un neurón en una capa en una entrada y capa.
- El valor de la salida de un neurón en una capa es la entrada y el valor de la salida de pasos anteriores.
- Las redes recurrentes presentan algunas características similares a la memoria de los humanos.
- Los algoritmos de aprendizaje en estas redes se pueden representar por medio de ecuaciones diferenciales ó ecuaciones en diferencia finita.

Entrenamiento de RNA's

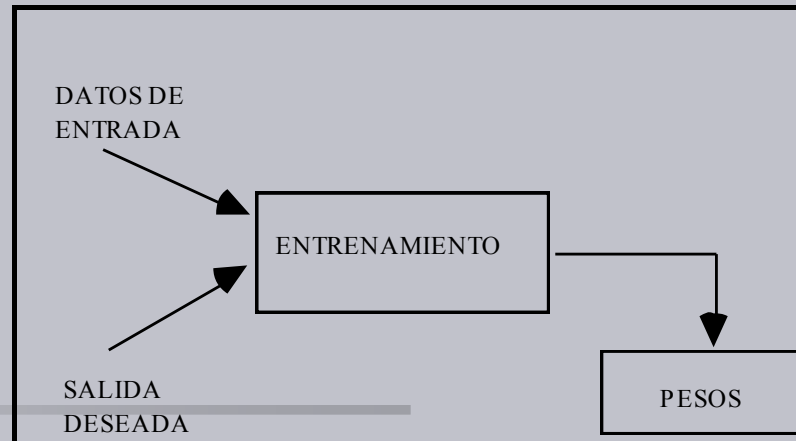
- Las R.N.A. aprenden de ejemplos.
- Una red es entrenada de manera que un conjunto de valores de entrada, aplicado en los neurones de entrada, produzca una salida deseada en los neurones de salida de la red.
- Normalmente los valores de entrada y de salida se representan de manera numérica, ya sea por dígitos binarios o números reales.
- La codificación de dichos valores depende de la arquitectura y características de la red neuronal utilizada.

Ejemplo simple de un sistema de RNA para Clasificar



Entrenamiento Supervisado

El aprendizaje se lleva a cabo a través de ejemplos que muestran explícitamente lo que se desea aprender.



Se requiere que cada vector de entrada (datos de entrada) esté relacionado a un valor de salida deseado.

A estos dos vectores se les conoce como "par de entrenamiento".

Ejemplo de Entrenamiento Supervisado

Supongamos que se desea que una red aprenda la función booleana XOR (OR exclusivo). Sabemos que dicha función tiene la siguiente tabla de verdad:

A	B	C = A XOR B
0	0	0
0	1	1
1	0	1
1	1	0



Ejemplo de Entrenamiento Supervisado (cont.)

Podemos definir como conjunto de entrenamiento lo siguiente:

ENTRADAS:

$$X_0 = [0,0]$$

$$X_1 = [0,1]$$

$$X_2 = [1,0]$$

$$X_3 = [1,1]$$

SALIDAS DESEADAS:

$$Y_0 = [0]$$

$$Y_1 = [1]$$

$$Y_2 = [1]$$

$$Y_3 = [0]$$

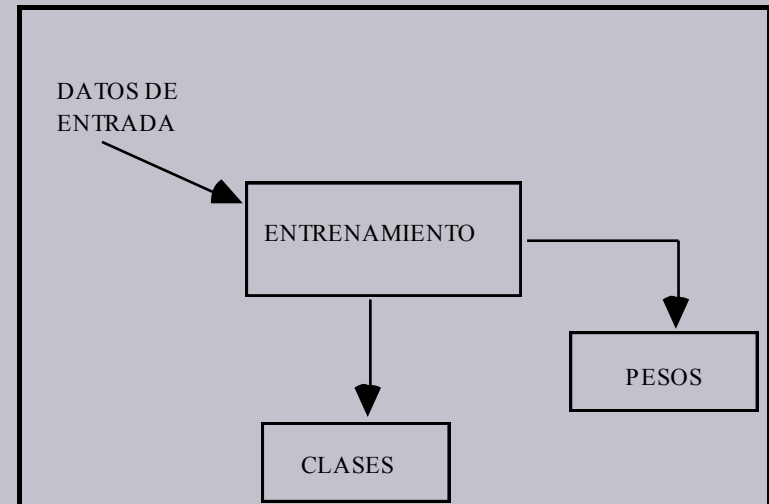
Entrenar significa que, cada par $[X_i, Y_i]$, se aplicará a la red repetidas veces siguiendo un algoritmo determinado, hasta que lo aprenda.

Entrenamiento no Supervisado

- En la naturaleza no es muy frecuente encontrar entrenamiento supervisado, sino mas bien entrenamiento no supervisado.
- Por ejemplo, el cerebro de un bebé no recibe un "vector de salida" con las respuestas a como reaccionar a los miles de estímulos que recibe cada día.

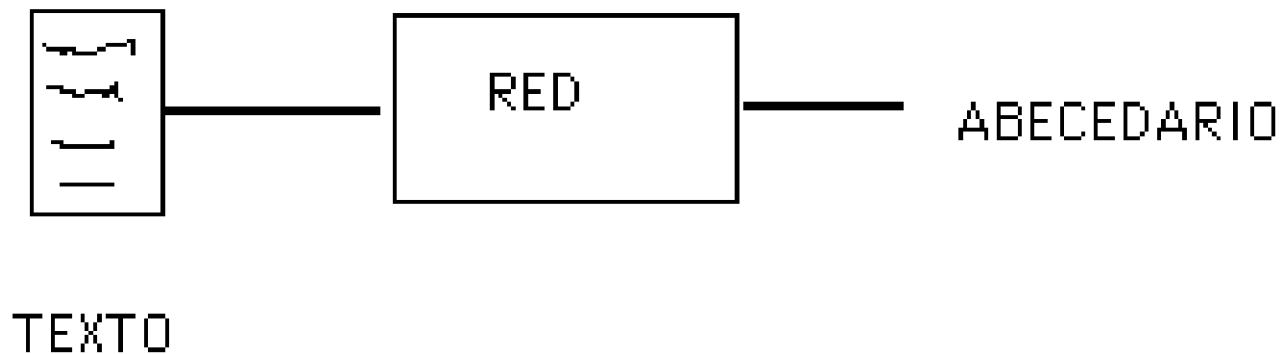
Entrenamiento no Supervisado (cont)

- Este tipo de entrenamiento no requiere de conocer la salida deseada para cada entrada, sino que los pesos se van modificando de manera que producen salidas que sean consistentes.
- En otras palabras, entradas similares producen las mismas salidas.



Ejemplo de Entrenamiento no Supervisado

Se desea que una red aprenda un conjunto de símbolos, digamos, las letras del alfabeto. Se le muestran a la red repetidas veces los símbolos, de manera que esta los clasificará, y creará grupos de los mas parecidos, generando una representación "ideal" para cada grupo.



Revisión Capítulo 2 libro de Haykin

Algoritmo de aprendizaje

- Es una serie de reglas dadas, bien definidas para la solución de un problema de aprendizaje.
- Reglas de aprendizaje básicas:
 - Aprendizaje por corrección de error
 - Aprendizaje basado en memoria
 - Aprendizaje Hebbiano
 - Aprendizaje competitivo
 - Aprendizaje Boltzman

Paradigmas de aprendizaje

- Son modelos del medio ambiente en el cual la red neuronal opera
- Tipos:
 - Aprendizaje con instructor
 - Aprendizaje sin instructor

El problema de asignación de créditos

- En algoritmos de aprendizaje para sistemas distribuidos se enfrenta el problema de decidir como asignar crédito o “culpabilidad” de las salidas globales, a cada una de las decisiones internas de la máquina, y decidir cual contribuyó a esas decisiones.

Problemas de asignación del crédito

- *Problema de asignación de crédito temporal.*
En un sistema con varias acciones realizadas, cual de estas acciones es la responsable de la salida?
- *Problema de asignación de crédito estructural.*
En un sistema con varios componentes, a cual se le asigna el crédito de la salida, y a cual se le debe modificar su comportamiento para mejorar la eficiencia del sistema?