

Neurocomputación

Teoría de Autómatas y Lenguajes Formales II. Curso 2008-09.
3º curso de Ingeniería Técnica en Informática de Sistemas
Escuela Técnica Superior de Ingeniería Informática
Universidad de Valladolid

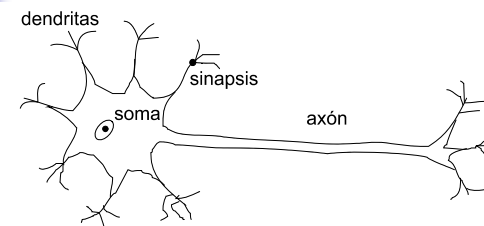
Introducción (I)

- Neurocomputación:
 - Modelo computacional
 - Basado en redes neuronales
 - Gran apogeo en los 80's y 90's.
- Objetivo de crear sistemas inteligentes:
"tareas inteligentes -> reservadas al cerebro"
- Idea inicial:
"un modelo computacional más cercano al cerebro, que a los clásicos, como el de Von Neumann."

Introducción (II)

- Modelo alternativo porque:
 - Masivamente paralelos (miles de millones de neuronas)
 - Computación y representación distribuida
 - Aprendizaje a base de ejemplos frente a los basados en conocimiento explícito: reglas de los sistemas expertos
 - Generalización / adaptabilidad
 - Procesamiento de la información inherente al contexto: ruido
 - Tolerante a fallos: reorganización de las neuronas cuando muere alguna de ellas
 - Bajo consumo de energía (inspiración biológica)

Modelo biológico (I)



- Entradas: dendritas
- Salida: axón
- Soma: procesador de información

Modelo biológico (II)

- **Neuronas sensitivas:**
 - Reciben las entradas directamente. Ej: las situadas en las yemas de los dedos (pinchazos, temperatura, etc)
- **Neuronas actuadoras:**
 - Materializan la respuesta. Ej: las situadas en el seno del tejido estriado de los músculos, contrayéndolo ante un pinchazo en el dedo.
- **Velocidad de procesamiento:**
 - 10^{11} neuronas en el cortex
 - $10^3 - 10^4$ conexiones
 - Lentitud de las señales electroquímicas (milisegundos)
 - Reconocimiento de una cara en milisegundos
 - Por tanto, es un sistema conexionista: alto grado de paralelismo

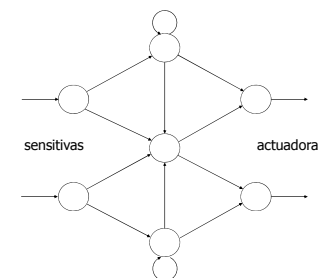
Breve reseña histórica (I)

- **(1943) El neurobiólogo McCulloch y el estadístico Pitts:**
 - Modelo de neurona artificial (Suma ponderada de las entradas)
- **(1962) Rosenblatt:**
 - Teorema de convergencia del perceptrón
 - Extendió la euforia: "RNA, solución universal"
- **(1969) Minsky y Papert:**
 - Limitaciones del perceptrón: problema del XOR

Breve reseña histórica (II)

- **(1982) Hopfield:**
 - Red más plausible con el modelo biológico
- **(1986) Rumelhart y McClelland:**
 - Aprendizaje del perceptrón multicapa: regla delta basada en el algoritmo del gradiente.
 - Original de Werbos (1974): Tesis Doctoral

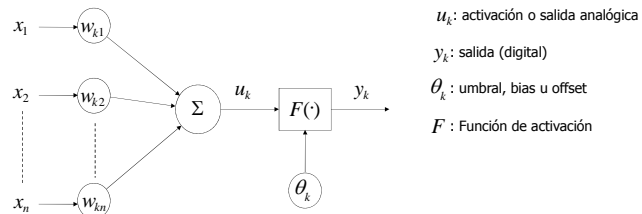
Modelo de McCulloch y Pitts (I)



RNA (sistema computacional):

- Conjunto de autómatas celulares (neuronas)
- Flujo de información según una topología de interconexiones (sinapsis)
- El paralelismo está marcado por la disponibilidad de las entradas.
 - Una neurona no obtiene su salida hasta no disponer de todas sus entradas

Modelo de McCulloch y Pitts (II)



u_k : activación o salida analógica
 y_k : salida (digital)
 θ_k : umbral, bias u offset
 F : Función de activación

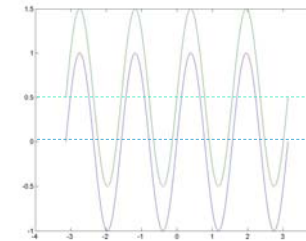
$$u_k = \sum_{j=1}^n w_{kj} x_j \in (-\infty, \infty) \quad y_k = F(u_k - \theta_k)$$

Práctica: $u_k = \sum_{j=0}^{n-1} w_{kj} x_j + w_{kn} \quad y_k = F(u_k)$

w_{kn} es como si fuera un peso de entrada constante e igual a uno

Modelo de McCulloch y Pitts (III)

- Término bias, offset o umbral:
 - Eliminación del nivel de continua, que no aporta información:



Modelo de McCulloch y Pitts (IV)

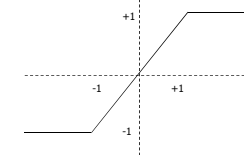
- Si el peso es positivo/negativo:
 - Sinapsis excitadora/inhibidora
- Función de activación F(x):
 - Filtrar la salida para acotarla en sintonía con las neuronales biológicas:
 - Los potenciales electroquímicos están acotados.
 - Si la intensidad de una señal (sentido) supera un valor, deja de percibirse como tal.
 - Ejemplo: altas y bajas frecuencias del sonido.

Modelo de McCulloch y Pitts (IV)

- F(x) debe ser acotada:

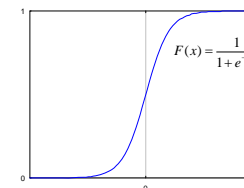
No diferenciables {

- Heaviside: $sgn(x)$
- Semilineal



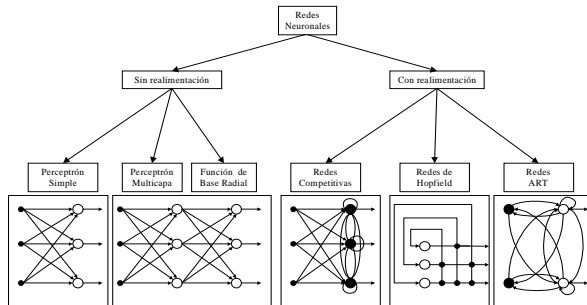
N - diferenciables {

- Arcotangente
- Sigmoide



Arquitectura

- Grafo dirigido y ponderado
- Clasificación: lazos de realimentación



Aprendizaje y recuperación (I)

- Sistema neuronal (hardware):
 - Altamente no lineal
 - Gran robustez
 - Tolerante a fallos
- Adicionalmente (software):
 - Aprendizaje: adaptación de sus pesos a cambios en el entorno
 - Manejo de información imprecisa, difusa, con ruido y basada en probabilidades
 - Generalización: respuesta satisfactoria ante casos desconocidos

Aprendizaje y recuperación (II)

- El modelo computacional equivalente al de Von Neumann
- Principal inconveniente:
 - No se conoce algoritmo capaz de entrenar una RNA arbitraria
- Actualmente:
 - Se conocen tipos particulares de redes con, al menos, un algoritmo de aprendizaje

Aprendizaje y recuperación (III)

- Aprendizaje:
 - Definir una función objetivo explícita o implícitamente para representar el estado de la red.
 - Buscar un óptimo de esta función modificando los pesos.
- Tipos:
 - Supervisado: maneja (entrada, salida deseada)
 - Ajuste de pesos para minimizar la diferencia entre salidas deseadas y las reales.
 - Ejemplo: perceptrón multicapa (MLP)
 - No supervisado:
 - Evolución del sistema hasta un estado estable
 - Ejemplo: mapa autoorganizado (SOM)

Aprendizaje y recuperación (IV)

- Redes heteroasociativas: aprendizaje => más cómputo

The diagram illustrates supervised learning for class "a". An oval containing five handwritten 'a's is connected by an arrow labeled "Supervisor" to a box labeled "Clase 'a'". Below this, a section labeled "Sin heteroasociación:" shows a network where each of the five 'a's is connected to both "Clase 'a'" and "Clase 'a'", indicating a lack of specific learning for the class.

Aprendizaje y recuperación (IV)

- Redes autoasociativas: recuperación => más cómputo

The diagram shows four 10x10 grids representing handwritten digits. A bracket groups the first three grids, and an arrow points from this group to a fourth grid below, illustrating the concept of pattern recovery in auto-associative networks.

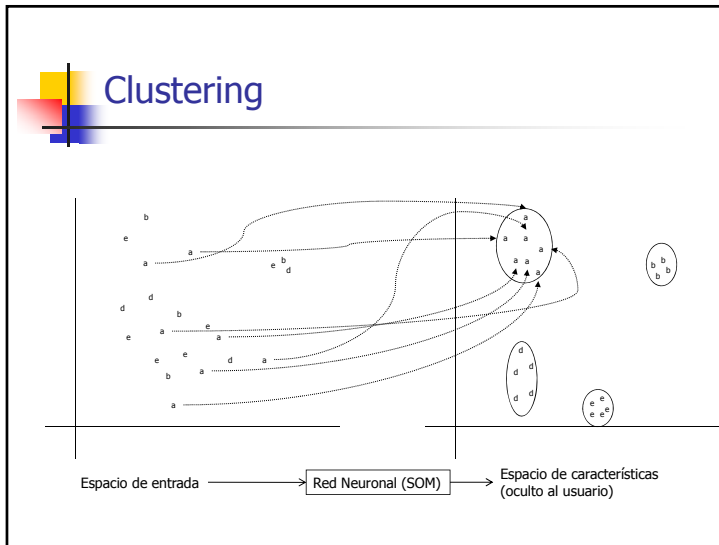
Reconocimiento de patrones estáticos (no aparece el tiempo)

The flowchart shows the process of static pattern recognition. It starts with "Manuscrito" (handwritten digits 1, 1, 1), followed by "Digitalización" (10x10 grids), "Vectorización" (10x10 binary vectors), and finally "Clasificación" leading to "Patrón '1'".

Manuscrito	Digitalización	Vectorización
		(0,0,1,0,0, 0,1,1,0,0, 1,0,1,0,0, 0,0,1,0,0, 0,0,1,0,0, 0,0,1,0,0, 0,0,1,0,0, 1,1,1,1,1)
		(0,0,1,0,0, 0,0,1,0,0, 0,1,1,0,0, 1,0,1,0,0, 0,0,1,0,0, 0,0,1,0,0, 1,1,1,0,0, 0,0,1,1,1)
		(0,1,0,0,0, 0,1,0,0,0, 1,1,0,0,0, 0,1,0,0,0, 0,0,1,0,0, 0,0,1,0,0, 0,0,1,0,0, 1,1,1,1,1)

Reconocimiento de patrones dinámicos (aparece el tiempo)

The diagram shows a dynamic pattern recognition process. It features a waveform visualization of speech sounds for the numbers "uno", "dos", "tres", "cuatro", "cinco", "seis", "siete", "ocho", "nueve", and "cero".



- ## Aplicaciones (II)
- Aproximación funcional:
 - Correspondencia (x, y) contaminados con ruido
 - Predicción:
 - Adivinar el siguiente término de una serie:
 - Meteorología
 - Bolsa
 - Demanda de consumo

- ## Aplicaciones (III)
- Optimización
 - Definir función objetivo
 - Encontrar óptimo bajo restricciones
 - Ejemplo: problema del viajante
 - Memorias asociativas:
 - Direccionables por contenido
 - Recuperar la información original a partir de una parcialmente distorsionada

- ## Aplicaciones (IV)
- Generación de señales de control
 - Calcular señales de control para que la "consigna" siga a la "referencia"
 - Especialmente aplicables en sistemas altamente no lineales y complejos
 - Se consigue la respuesta a partir del aprendizaje con ejemplos