

Redes neuronales (Parte I: Introducción)

Inteligencia artificial

Patricio García Báez

Grado en Ingeniería Informática

Índice

- Definiciones
- Inspiración biológica
- Áreas y Aplicaciones
- Propiedades y Capacidades
- Características
- Software

“Entender el cerebro y emular su potencia”

- **Red Neuronal**

Una red neuronal (RN) es un procesador distribuido masivamente paralelo que tiene una propensión natural para almacenar el conocimiento a partir de su experiencia y hacerlo disponible para su uso. Se parece al cerebro en dos sentidos:

- El conocimiento es adquirido por la red mediante un proceso de aprendizaje
- La fuerza de las conexiones entre neuronas conocidas como pesos sinápticos se utilizan para almacenar el conocimiento

(Alejsabder y Morton, 1990)

- Red Neuronal

Una red neuronal artificial (RNA) es un sistema de procesamiento de información que tiene ciertas aptitudes en común con las redes neuronales biológicas:

- El procesamiento de información ocurre en muchos elementos simples llamados neuronas
- Las señales son transferidas entre neuronas a través de enlaces de conexión
- Cada conexión tiene un peso asociado, el cual, típicamente, multiplica a la señal transmitida
- Cada neurona aplica una función de activación (usualmente no lineal) a su entrada de red (suma de entradas pesadas) para determinar su salida

(Laurene Fausett)

- Red Neuronal

Una RN es una estructura de proceso de información distribuida y paralela, que consta de elementos de proceso (los cuales pueden poseer memoria local y pueden llevar a cabo operaciones localizadas de procesamiento de información) interconectados mediante canales de señales unidireccionales denominados conexiones. Cada elemento de proceso tiene una única conexión de salida que se ramifica (*fan out*) en tantas conexiones como se deseen, cada una llevando la misma señal, la señal de salida del elemento de proceso. Dicha salida puede ser de cualquier tipo matemático deseado. El procesamiento de información que acompaña a cada elemento de proceso puede ser definido arbitrariamente con las restricciones de que debe ser completamente local, esto es, solo puede depender de los valores actuales de las señales de entrada y de los valores almacenados en la memoria local de dicho elemento de proceso

Definiciones

- Algoritmo de aprendizaje
 - Función que modifica los pesos sinápticos de un modo ordenado de manera que alcanza el objetivo deseado
- Sinónimos de RN
 - neurocomputador, neurocomputación, red conexionista, procesos distribuidos en paralelo (pdp's), redes neuronales (artificiales)

Inspiración biológica

- Cerebro
 - Complejo, no lineal, paralelo
 - Gran velocidad de proceso
 - Tratamiento de grandes cantidades de información procedentes de:
 - Los sentidos
 - Memoria almacenada
 - Capacidad de tratar situaciones nuevas
 - Capacidad de aprendizaje

Inspiración biológica

- **Neurona**

- Número: de 10^{10} a 10^{11}
- Tamaño: 5 ó 6 órdenes de magnitud menor que puertas lógicas en silicio
- Árbol dendrítico de entradas
- Un axón de salida
- Sobre 10^4 sinapsis
- Comunicación mediante Potenciales de Acción (PA)
- Periodo refractario de 10^{-3} segundos entre PAs
- Tiempo eventos: milisegundos (10^{-3} s), en silicio es de nanosegundos (10^{-9} s)
- Plasticidad Neuronal
 - Creación de nuevas neuronas
 - Creación de nuevas sinapsis
 - Modificación de sinapsis

Inspiración biológica

- Transmisión neuronal
 - Impulso eléctrico que viaja por el axón
 - Liberación de neurotransmisores
 - Apertura/cierre de canales iónicos
 - Variación potencial en dendrita
 - Integración de entradas en soma
 - Si se supera umbral de disparo se genera un PA

Inspiración biológica

- Sistema Nervioso Central
 - Inclínación a adquirir conocimiento desde la experiencia
 - Conocimiento almacenado en conexiones sinápticas
 - Gran plasticidad neuronal
 - Comportamiento altamente no-lineal
 - Alta tolerancia a fallos (muerte neuronal)
 - Apto para reconocimiento, percepción y control

Inspiración biológica

- Comparación con ordenadores
 - Muerte neuronal
 - Optimizados para distintos problemas
 - Reconocimiento de patrones
 - Percepción
 - Control
 - Secuencialidad vs. Paralelismo

Inspiración biológica

- Organización estructural en niveles
 - Sistema Nervioso Central
 - Circuitos entre regiones
 - Circuitos locales
 - Neuronas
 - Árboles dendríticos
 - Microcircuitos neuronales
 - Sinapsis
 - Canales iónicos
 - Moléculas

Áreas y Aplicaciones

Procesamiento de Señales
Control
Análisis de Datos
Reconocimiento de Patrones
Generación

Aplicaciones

Desarrollo de aplicaciones basadas en técnicas conexionistas

Implementaciones

Construcción de modelos neuronales

Arquitectura y Teoría de RN

Proponer y validar modelos de funcionamiento de arquitecturas neuronales

Informática

Psicología

Matemáticas

Física

Neurofisiología

Propiedades y Capacidades

- Generalización
- Estructura altamente paralela
- No linealidad
- Mapeo de Entrada-Salida
- Adaptabilidad
- Respuesta graduada
- Información Contextual
- Tolerancia a fallos
- Implementación VLSI
- Uniformidad en el Análisis y Diseño
- Analogía Neurobiológica

Características

- Arquitectura o topología:
 - Estructura de capas y patrón de conexionado
- Modelo de neurona artificial
 - Función suma y función de activación
- Algoritmo de aprendizaje:
 - Método de definición de pesos de conexión

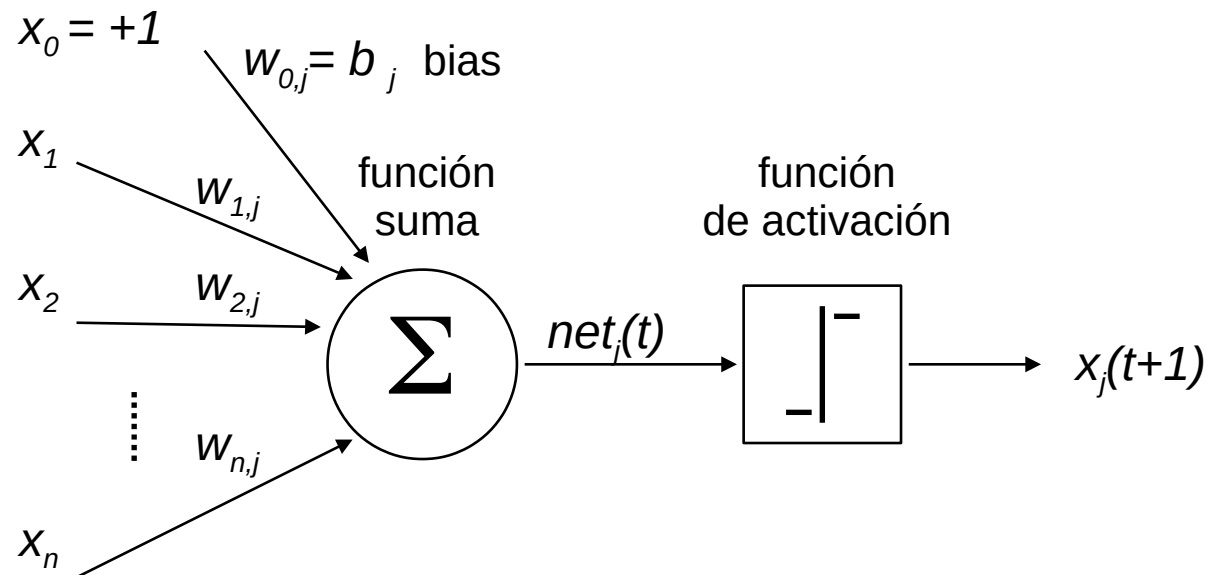
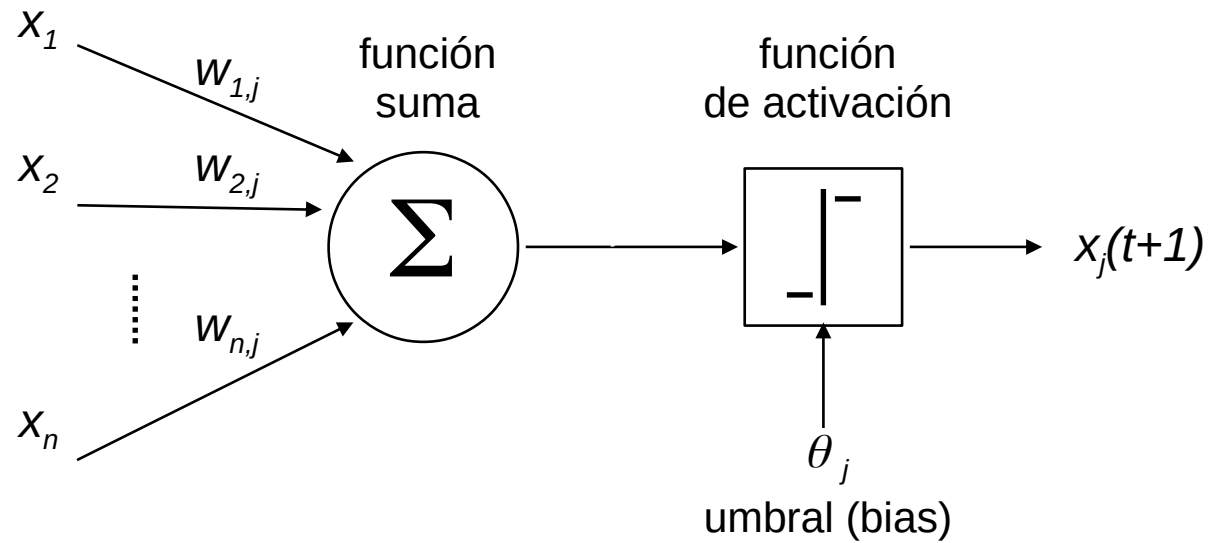
Características: Modelo de neurona

- Neurona artificial
 - Grupo de entradas (x)
 - Pesos sinápticos (w)
 - Función suma (net)
 - Función de activación (act)
 - Una única salida (y)
 - Funcionamiento en modo aprendizaje o ejecución

Características: Modelo de neurona

- Neurona natural vs neurona artificial
 - Neurona = Unidad de proceso
 - Conexiones sinápticas = Conexiones Pesadas
 - Efectividad sináptica = Peso sináptico
 - Excitatorio/Inhibitorio = Pesos + ó -
 - Efecto combinado de sinapsis = Función suma
 - Activación-> Ratio disparo = Función activación -
> salida

Características: Modelo de neurona



Características: Modelo de neurona

- Ejemplo de funciones suma y **activación**

$$net_j(t) = \sum_i w_{ij} x_i$$

función suma

$$x_j(t+1) = f_{act}(net_j(t), x_j(t), \theta_j)$$

función de activación

$$f_{act}(net_j(t), x_j(t), \theta_j) = \begin{cases} 1 \Rightarrow net_j(t) - \theta_j \geq 0 \\ 0 \Rightarrow net_j(t) - \theta_j < 0 \end{cases}$$

función
escalón

$$f_{act}(net_j(t), x_j(t), \theta_j) = \begin{cases} 1 \Rightarrow net_j(t) - \theta_j \geq \frac{1}{2} \\ net_j(t) - \theta_j + \frac{1}{2} \\ 0 \Rightarrow net_j(t) - \theta_j \leq -\frac{1}{2} \end{cases}$$

función
lineal a
trozos

$$f_{act}(net_j(t), x_j(t), \theta_j) = \frac{1}{1 + \exp(-a (net_j(t) - \theta_j))}$$

función
sigmoide

$$f_{act}(net_j(t), x_j(t), \theta_j) = \tanh\left(\frac{net_j(t) - \theta_j}{2}\right)$$

f. tangente
hiperbólica

$$f_{act}(net_j(t), x_j(t), \theta_j) = \max(0, net_j(t) - \theta_j)$$

f. ReLU

$$f_{act}(net_j(t), x_j(t), \theta_j) = net_j(t) / \sum_k net_k(t)$$

f. SoftMax

Características: Arquitecturas de red

- Según e/o/s
 - Número y tipo de entradas
 - Elementos ocultos
 - Elementos de salida
- Según conectividad entre capas
 - Feedforward (hacia adelante)
 - Monocapa Feedforward
 - Multicapa Feedforward
 - Redes Recurrentes
 - Sin neuronas ocultas
 - Con neuronas ocultas
 - Estructuras Enrejadas (Lattice)

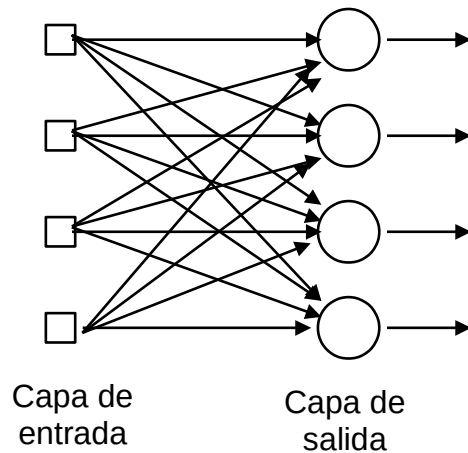
Características: Arquitecturas de red

- Según conexión entre capas
 - Totalmente conectados (fully connected)
 - Parcialmente conectados
 - Conexión uno a uno
- Sincronía (actualización de valores)
 - Simultánea
 - Aleatoria
 - Según orden topológico

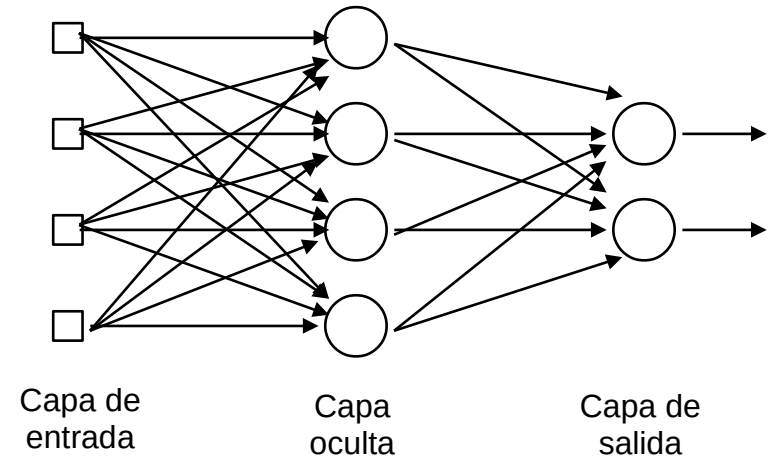
Características: Arquitecturas de red

– Ejemplos feedforward

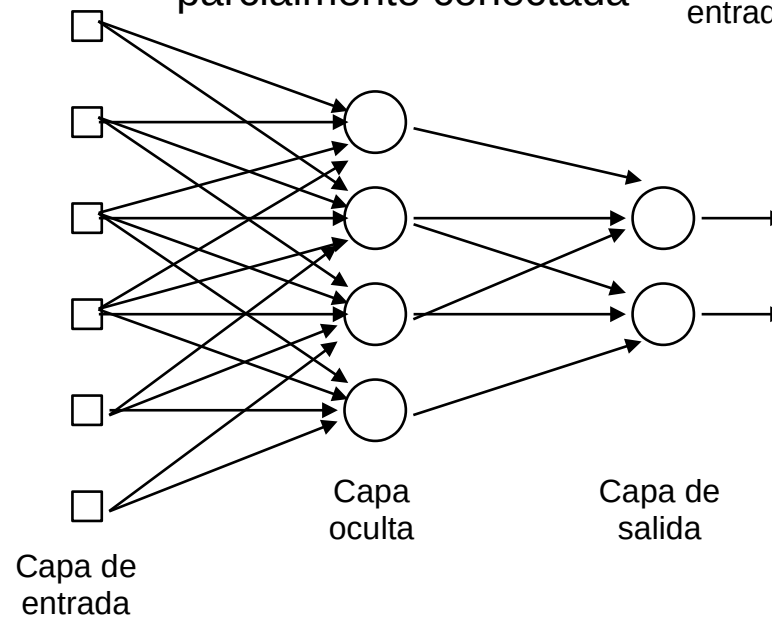
Monocapa feedforward



Multicapa feedforward



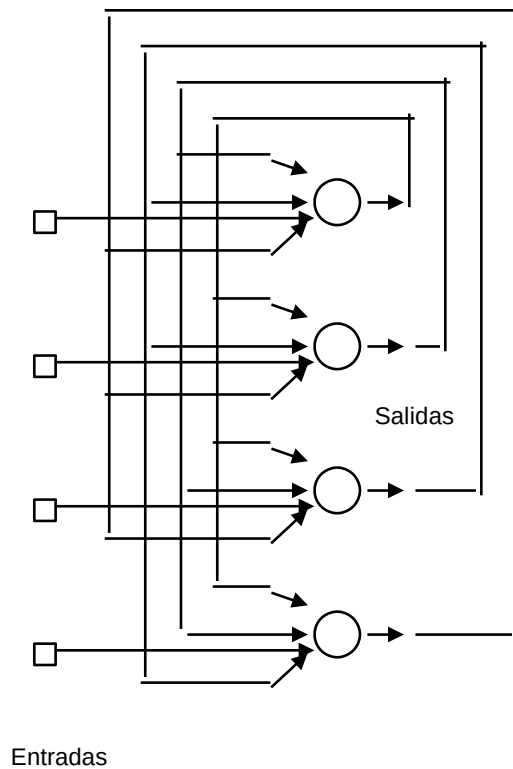
Multicapa feedforward
parcialmente conectada



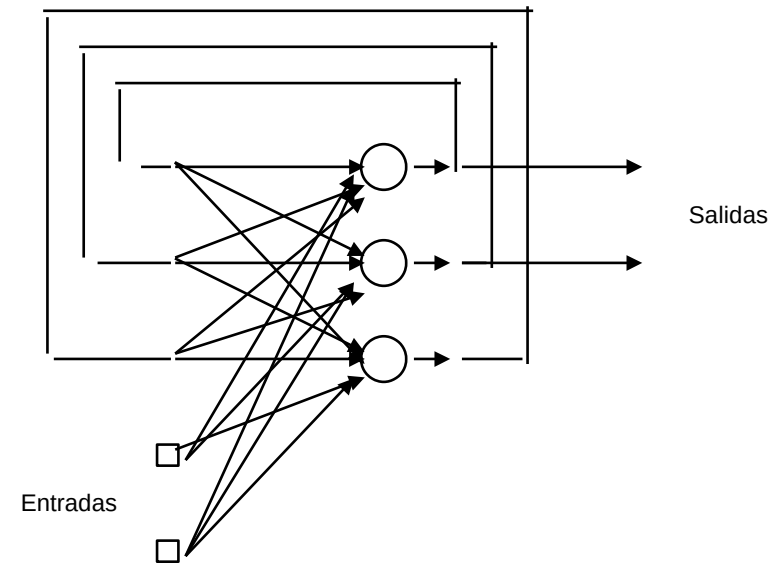
Características: Arquitecturas de red

– Ejemplos recurrentes

Recurrente con
Interconexiones
laterales



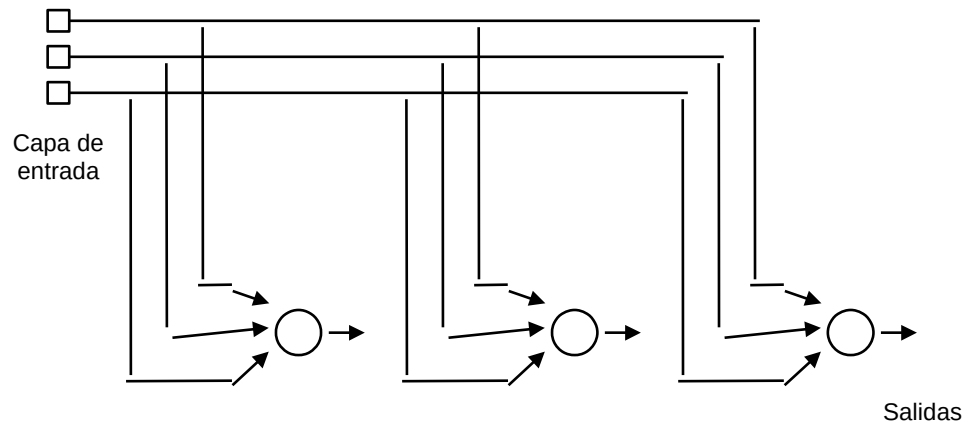
Recurrente con
neuronas
ocultas



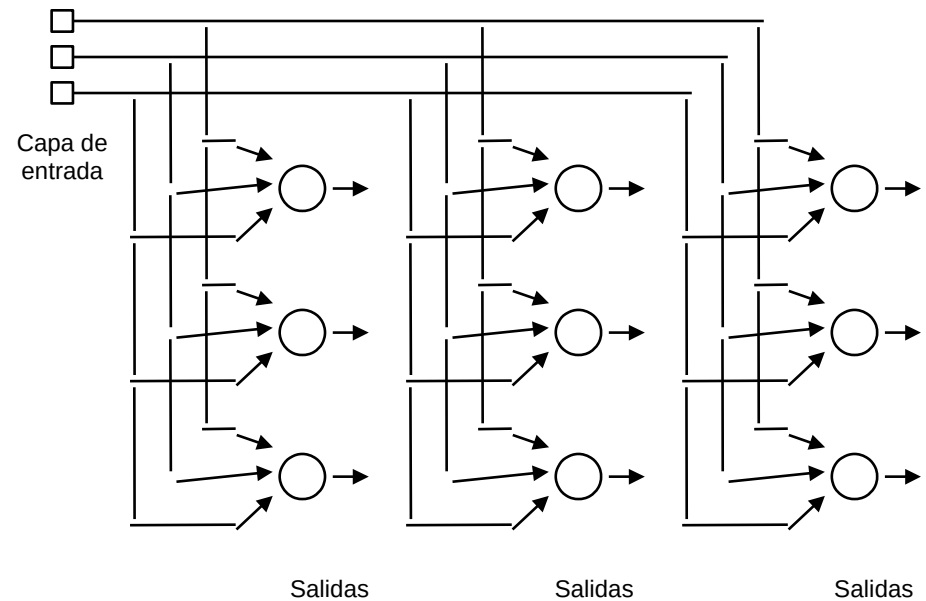
Características: Arquitecturas de red

– Ejemplos enrejados

Enrejada
de una
dimensión



Enrejada
de dos
dimensiones



Características: Aprendizaje

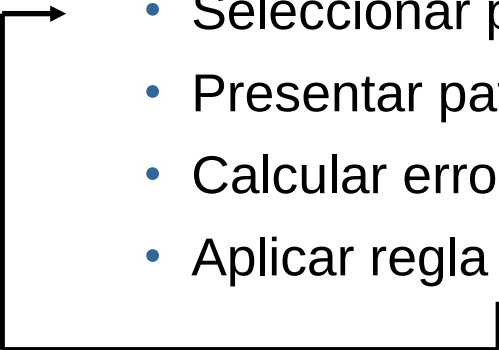
- Eventos:

- - Estimulación de la RN por el entorno
 - Cambios en la RN debido a estimulación
 - Nueva forma de responder debido a cambios de la estructura interna de la RN

Características: Aprendizaje

- Taxonomía:
 - Paradigmas de aprendizaje:
 - Aprendizaje Supervisado
 - Aprendizaje por Reforzamiento
 - Aprendizaje Auto-organizado (No Supervisado)
 - Precalculado o prefijado
 - Algoritmos de aprendizajes (reglas):
 - Corrección del error
 - Optimización de la energía (Apr. de Boltzmann)
 - Hebbiano
 - Competitivo
 - Otros:
 - local vs. global, por dato vs. por épocas (*batch*), valores iniciales de pesos, ...

Características: Aprendizaje

- Aprendizaje supervisado (Clasificación)
 - Se presentan pares de patrones de entrada y salida deseada
 - Pasos:
 - Fijar pesos aleatorios las conexiones
 - Seleccionar par de entrenamiento
 - Presentar patrón de entrada y calcular salida
 - Calcular error o discrepancia con la salida deseada
 - Aplicar regla de aprendizaje
 - Dudosa plausibilidad biológica:
 - Requiere propagar información hacia atrás
 - Requiere de instructor
- 

Características: Aprendizaje

- Aprendizaje no supervisado (Agrupación)
 - Se presentan sólo patrones de entrada
 - Basado en la redundancia en las entradas
 - Aprendizaje extrae de los patrones:
 - Familiaridad con patrones típicos o promedios del pasado
 - Análisis de las Componentes Principales
 - Clustering
 - Prototipos, correspondientes a las categorías existentes
 - Codificación
 - Mapa de Características
 - Grandes plausibilidades biológicas

Características: Tipos de problemas

- Aproximación
- Asociación
- Clasificación de Patrones
- Predicción
- Control
- En general se usa si:
 - Difícil describir conocimiento/forma de resolverlos
 - Se dispone de una gran cantidad de datos

Características: Codificación e/s

- Tipo de señales
 - Global vs. local
 - Discretas vs. continuas
- Relación e/s
 - Leer salida para cada entrada
 - Evolución de la salida ante la entrada

Características: Codificación e/s

- Preparación de datos
 - Diseño de conjuntos de entrenamiento y testeo
 - Longitud de la distribución (Grandes conjuntos)
 - Eliminación de información redundante e irrelevante
 - Manualmente
 - Preproceso de reducción de datos (extracción de características)
 - PCA, Momentos de Zernike, ...
 - Centrado, escalado o normalización
 - Eliminación de no-linealidades conocidas

Software

- scikit-learn

- Librería Python de aprendizaje automático

- Weka

- Entorno de aprendizaje automático y análisis de datos

- TensorFlow

- Librería principalmente para entrenamiento y ejecución de RNAs desarrollada por Google Brain

- PyTorch

- Librería para aprendizaje automático basada en Torch y desarrollada por Meta AI

- Keras

- Librería de interface Python para RNAs. Puede trabajar sobre JAX, TensorFlow o PyTorch









Créditos

Esta presentación está bajo una licencia

Creative Commons Attribution-ShareAlike 4.0 International License



Referencias

-  *Introduction to the theory of neural computation.* Hertz, Krogh, Palmer. Addison-Wesley. 1991
-  *Fundamentals of Neural Networks. Architectures, algorithms, and applications.* Fausett. Prentice-Hall. 1994
-  *Neural Networks. A comprehensive foundation.* Haykin. Macmillan. 1994
-  *Neurocomputing.* Hecht-Nielsen. Addison-Wesley. 1989
-  *Neural Computing. Theory and Practice.* Wasserman. Van Nostrand. 1989
-  *Descripción Formal de Modelos de Redes Neuronales.* J.R. Álvarez. IX Cursos de Verano de la UNED. 1998
-  *Neural Netowrks for Pattern Recognition.* C.M. Bishop. 1995
-  *Deep Learning.* I. Goodfellow, Y. Bengio, A. Courville. MIT Press. 2016