

UEA Datos a Gran Escala

Métodos y Tecnologías para el Análisis (Modelado) de los Datos a Gran Escala: **Redes Neuronales Artificiales y *Deep Learning***



Dr. Pedro Pablo González Pérez

e-mail: pgonzalez@correo.cua.uam.mx

<http://dcni.cua.uam.mx/division/usuario?p=31#>

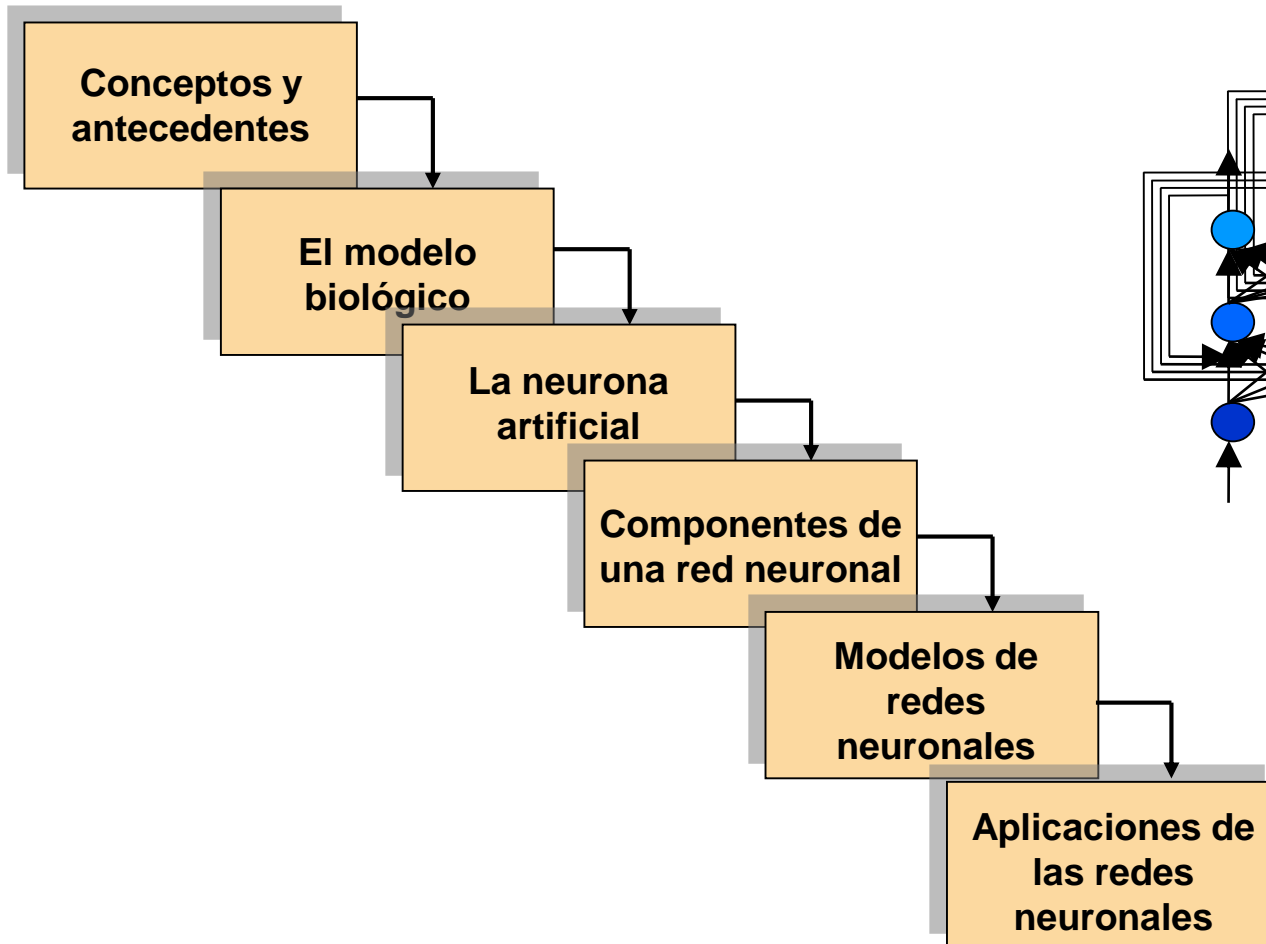
Departamento de Matemáticas Aplicadas y Sistemas



UNIVERSIDAD
AUTÓNOMA
METROPOLITANA
Unidad Cuajimalpa

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales



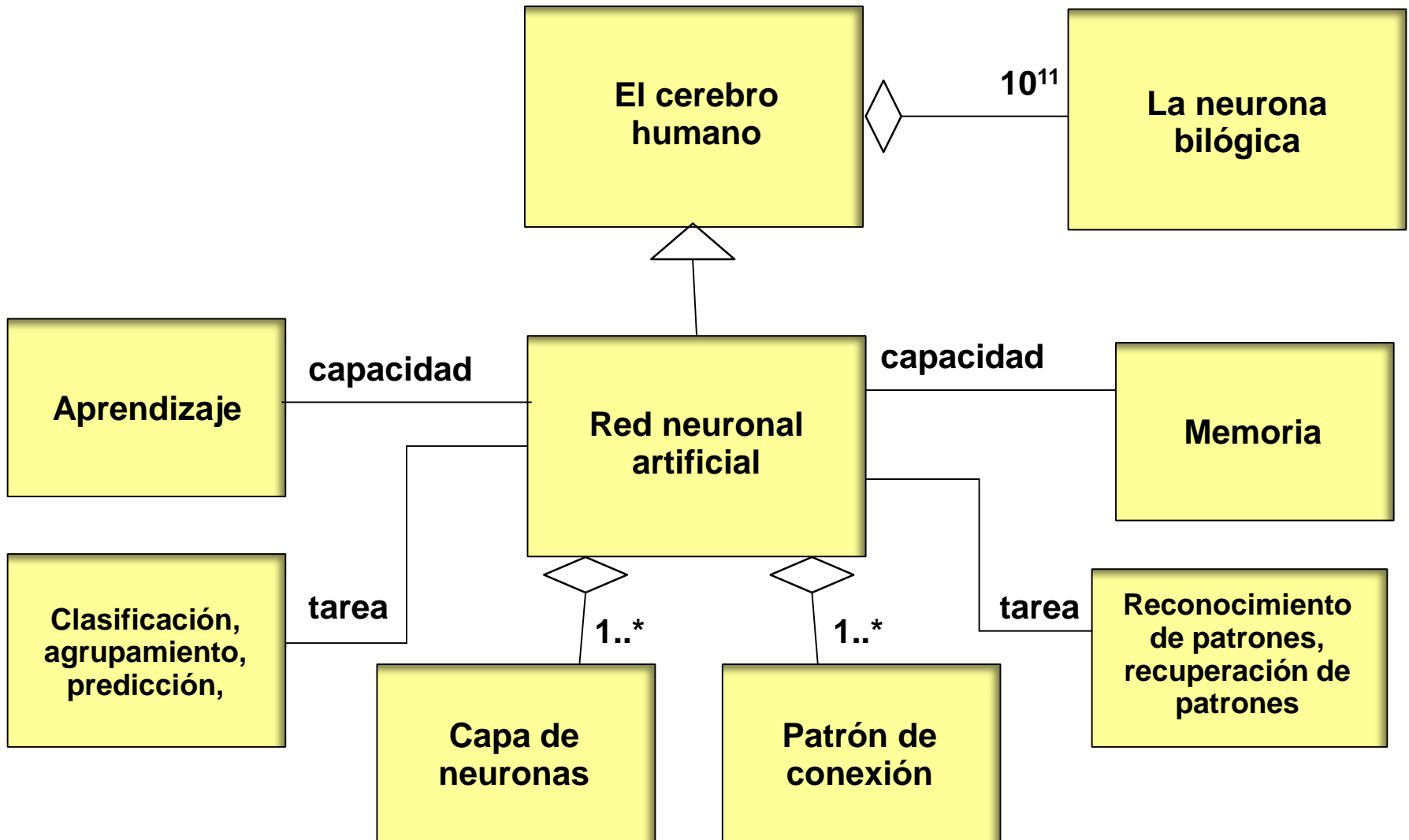
UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

- Las redes neuronales artificiales son sistemas de procesamiento de información, cuya esencia central es la analogía que éstas exhiben con la estructura y funcionamiento de las redes neuronales biológicas.
- Estos sistemas de procesamiento de información son concebidos para emular una serie de funciones llevadas a cabo por el cerebro, tales como: reconocimiento de patrones, memoria, aprendizaje y la generalización o abstracción del conocimiento, entre otras.
- Una red neuronal artificial consiste de un conjunto de unidades de procesamiento (conocidas como neuronas artificiales), las cuales se conectan entre sí siguiendo diferentes patrones de interconexión. Comúnmente, las neuronas artificiales se organizan en capas o estratos.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales



UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Año	Autores	Propuesta
1943	McCulloch and Pitts	El primer modelo artificial de neurona biológica utilizando funciones de umbral binario.
1943	Landahl, McCulloch and Pitts	Implementación de varios operadores lógicos y aritméticos utilizando el modelo de la neurona de McCulloch and Pitts neuron model.
1949	Hebb	La regla de aprendizaje de Hebb (aprendizaje por reforzamiento).
1956	Taylor	Memoria asociativa utilizando la regla de aprendizaje de Hebb.
1958	Rosenblatt	Perceptron: un método de aprendizaje para el modelo de la neurona artificial de McCulloch and Pitts.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Years	Authors	Proposal
1960	Widrow and Hoff	<i>Adaline</i> : un Sistema simple similar a un perceptron, entrenado por una regla de gradiente descendente para minimizar el error cuadrático medio.
1961	Rosenblatt	Esquema de retropropagación de errores (<i>backpropagation</i>) para el entrenamiento-aprendizaje en redes multi-estrato. Esta primera propuesta no resultó exitosa.
1969	Minsky and Papert	Demostración de los límites de los <i>perceptrons</i> simples. Sólo pueden funcionar bien para clases linealmente separables.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Years	Authors	Proposal
1986	Rumelhart et al.	El perceptron multi-estrato y el algoritmo de aprendizaje basado en la retropropagación de errores (<i>backpropagation</i>).
Década de los 90's	Varios autores	Propuestas de nuevas arquitecturas de redes neuronales, modelos y algoritmos de aprendizajes y desarrollo de muchas aplicaciones.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Years	Authors	Proposal
Finales de los 90's – inicios de los 2000's	Varios autores	<p>Desarrollo de nuevos modelos de redes neuronales artificiales:</p> <ul style="list-style-type: none"><input type="checkbox"/> Perceptron Multi-Estrato (<i>MLP</i>, del inglés)<input type="checkbox"/> Máquinas Restringidas de Boltzmann (<i>RBM</i>, del inglés)<input type="checkbox"/> Máquinas de soporte vectorial (<i>SVM</i>, del inglés).<input type="checkbox"/> Redes neuronales recurrentes (<i>RNN</i>, del inglés).<input type="checkbox"/> Red neuronal convolucional (<i>CNN</i>, del inglés).

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Years	Authors	Proposal
2000 – 2020	Varios autores	<p>Desarrollo de modelos de redes neuronales basados en el aprendizaje profundo (<i>deep learning</i>), para permitir un aprendizaje mucho más complejo.</p> <ul style="list-style-type: none"><input type="checkbox"/> Red de creencia profunda, del inglés <i>Deep belief network</i> (DBN).<input type="checkbox"/> <i>Deep neural networks</i> (DNN).<input type="checkbox"/> <i>Deep reinforcement learning</i> (DRL).<input type="checkbox"/> <i>Recurrent neural networks</i> (RNN).<input type="checkbox"/> <i>Convolutional neural networks</i> (CNN).

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Desde una perspectiva de procesamiento de información, tres niveles principales pueden ser distinguidos en el cerebro humano:

- **Nivel estructural:** sinapsis, neuronas, redes locales, capas y columnas, mapas topográficos, sistemas.
- **Nivel fisiológico:** reacciones químicas y eventos físicos, transmisión de sustancias.
- **Nivel cognitivo:** comportamiento humano.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

Niveles de organización del Sistema nervioso, según la caracterización hecha por (Shepherd, 1988)

Comportamiento

Sistemas y vías

Circuitos centrales y locales

Neurona

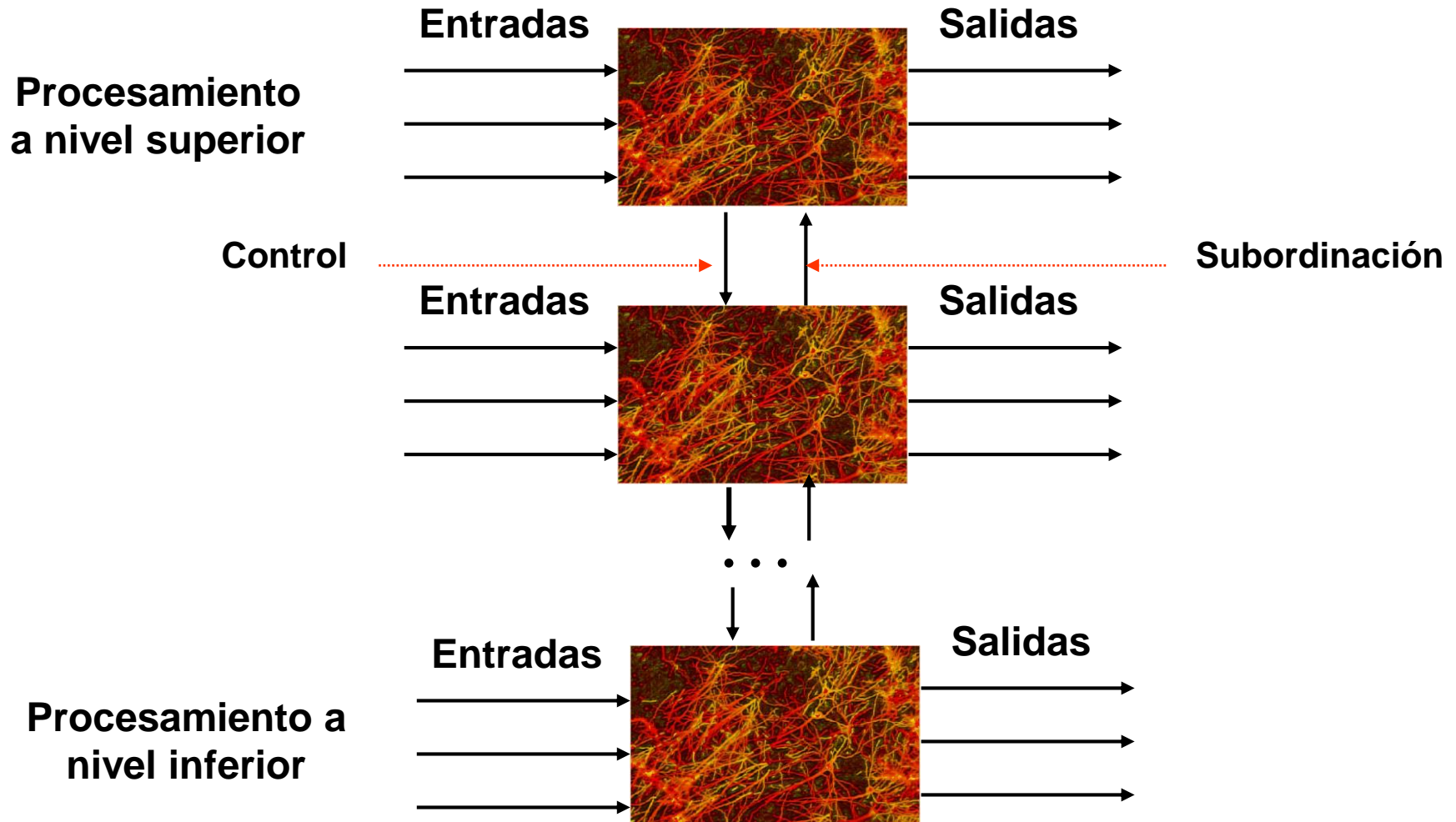
Microcircuitos

Sinapsis

Membranas, moléculas, iones

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales



UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

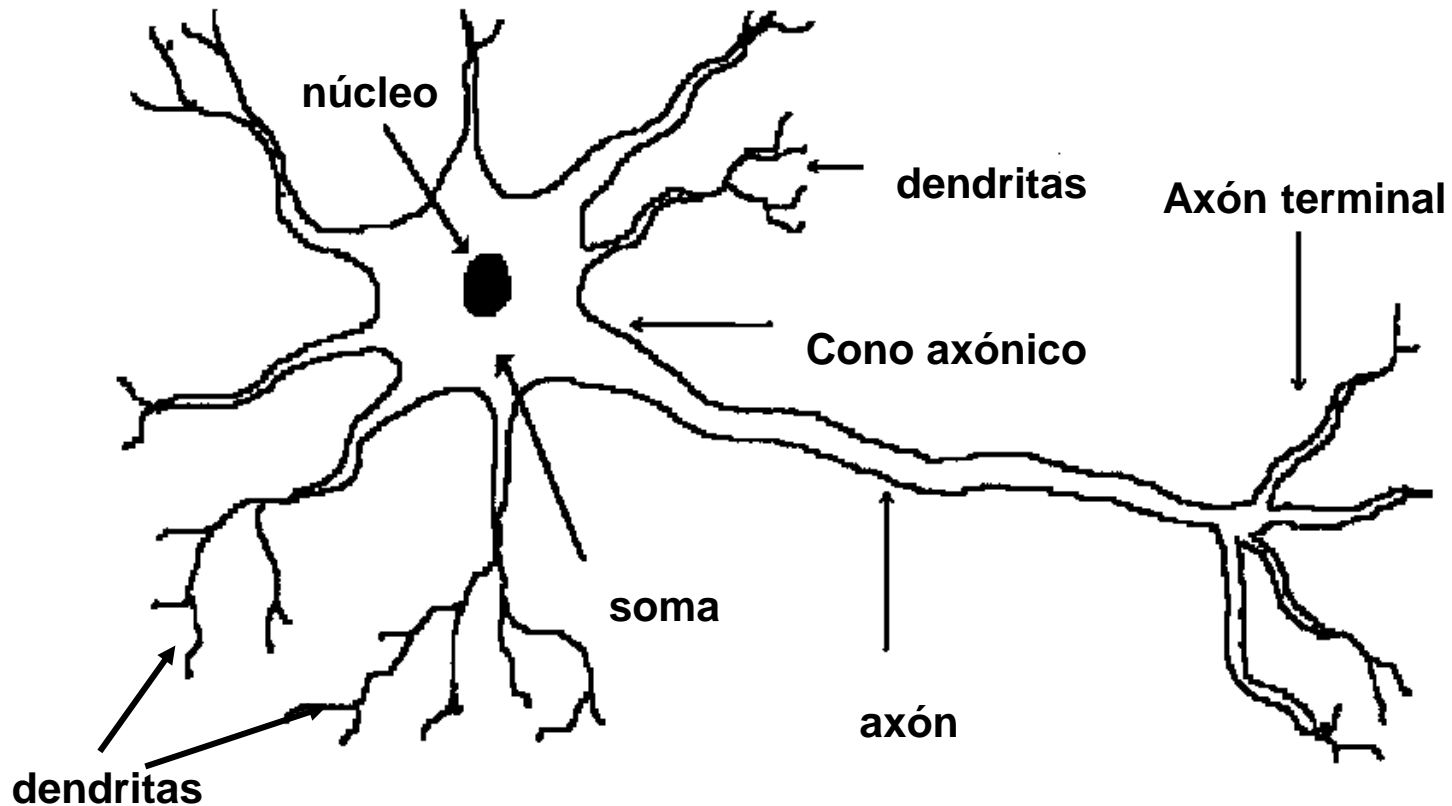
Características relevantes de la computación neuronal (Churchland and Sejnowski, 1992)

- Especialización de función
- Neuronas y sinapsis
- Conectividad
- Entradas análogas y salidas discretas
- Temporización
- Efectos célula-a-célula
- Patrones de disparo de las neuronas
- Campos receptivos
- Sistemas específicos y no específicos
- Acción a distancia.
- Arquitectura paralela.

UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

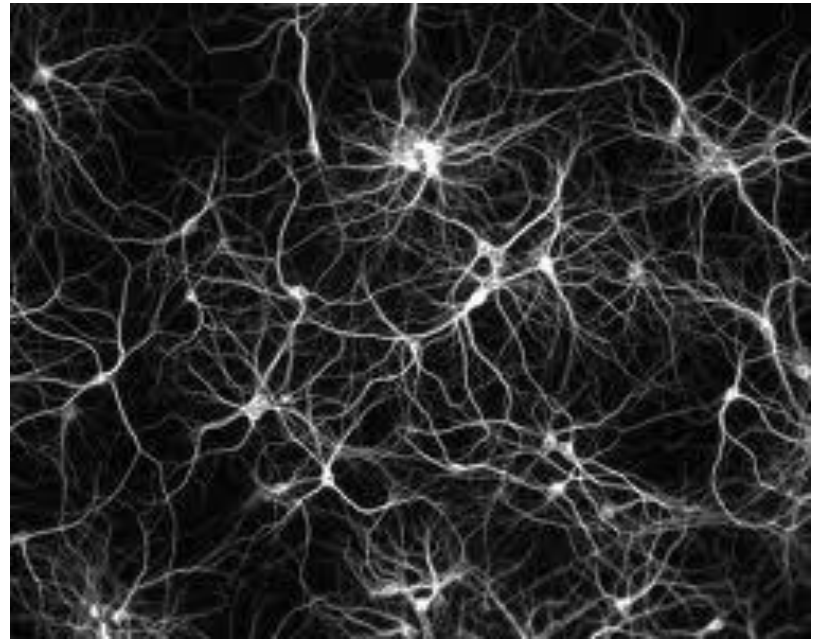
La neurona biológica



UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

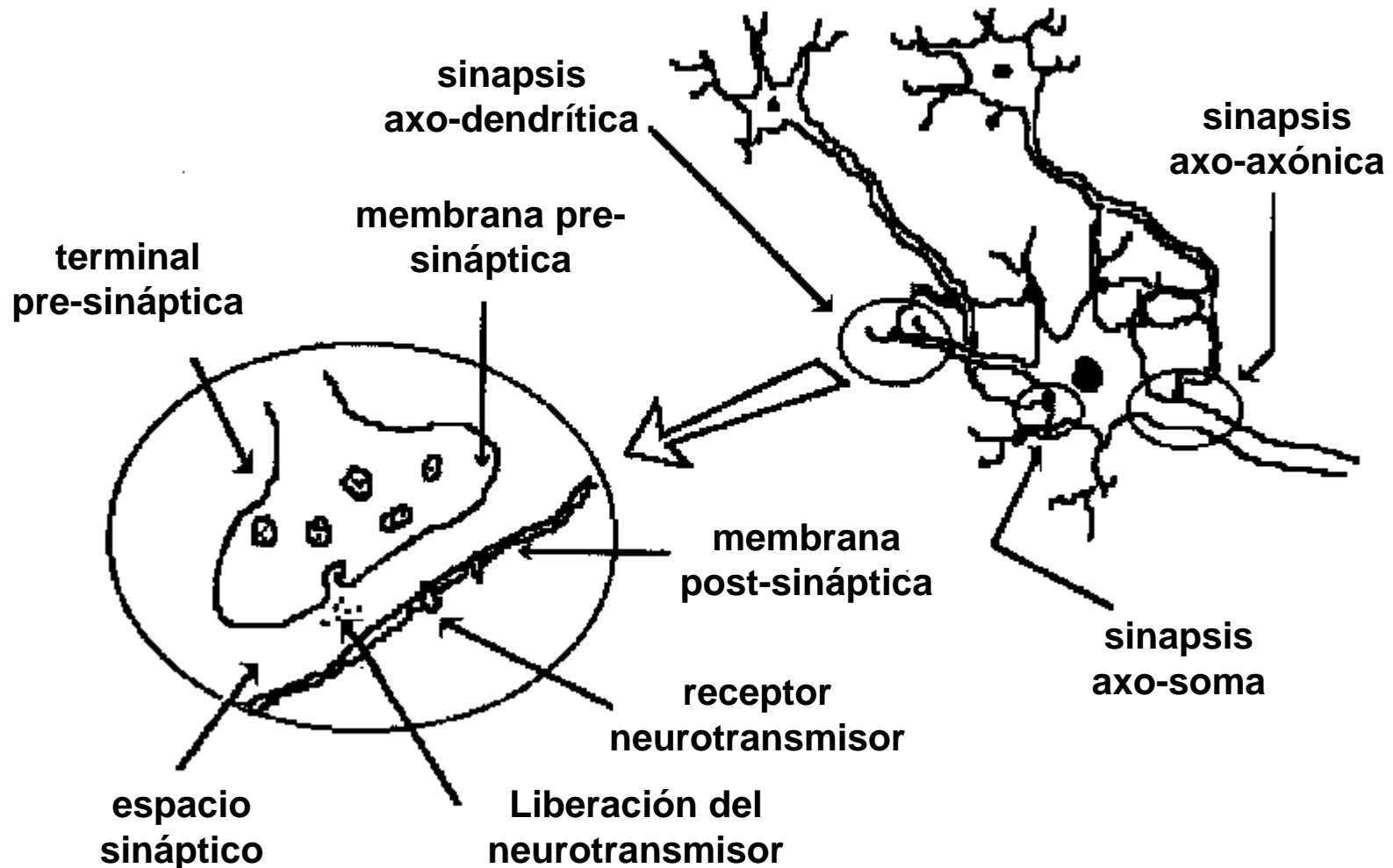
Las redes neuronales biológicas



UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

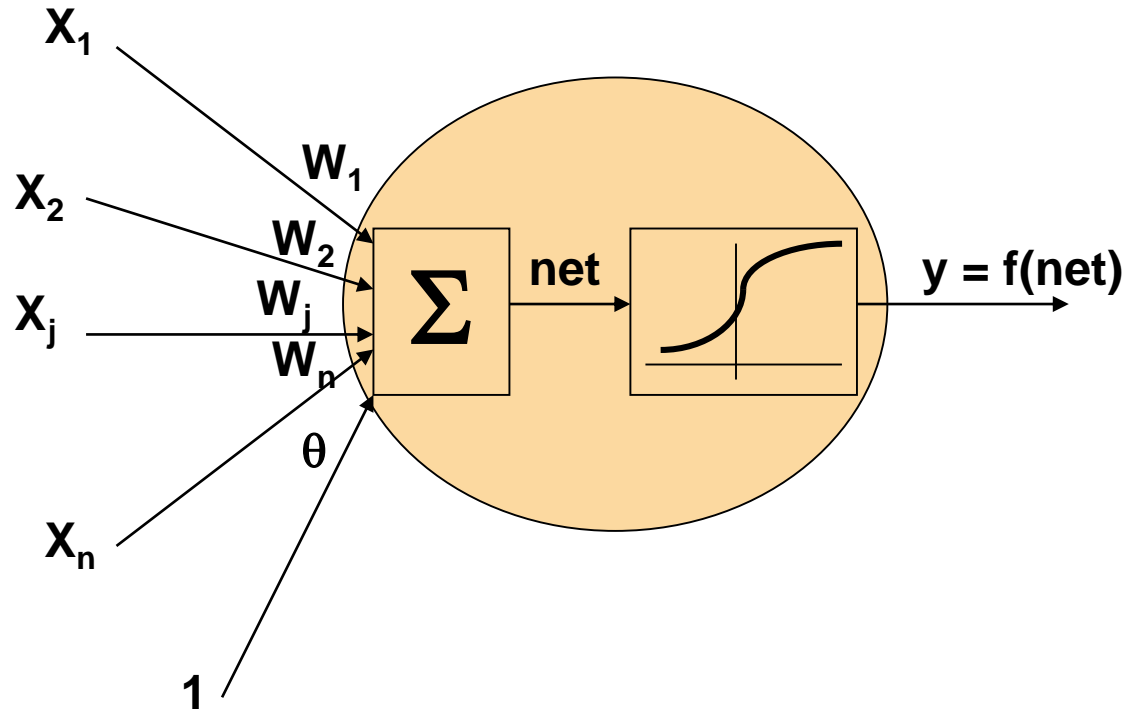
Neuronas y sinapsis



UEA Datos a Gran Escala

Fase de Modelado: Redes Neuronales Artificiales

La neurona artificial



Valores de entrada: $X_1, X_2, \dots, X_j, \dots, X_n$

$$\text{net} = \sum_j W_j X_j + \theta$$

y: Valor de salida

Pesos sinápticos: $X_1, X_2, \dots, X_j, \dots, X_n$

f(): función de activación

UEA Datos a Gran Escala

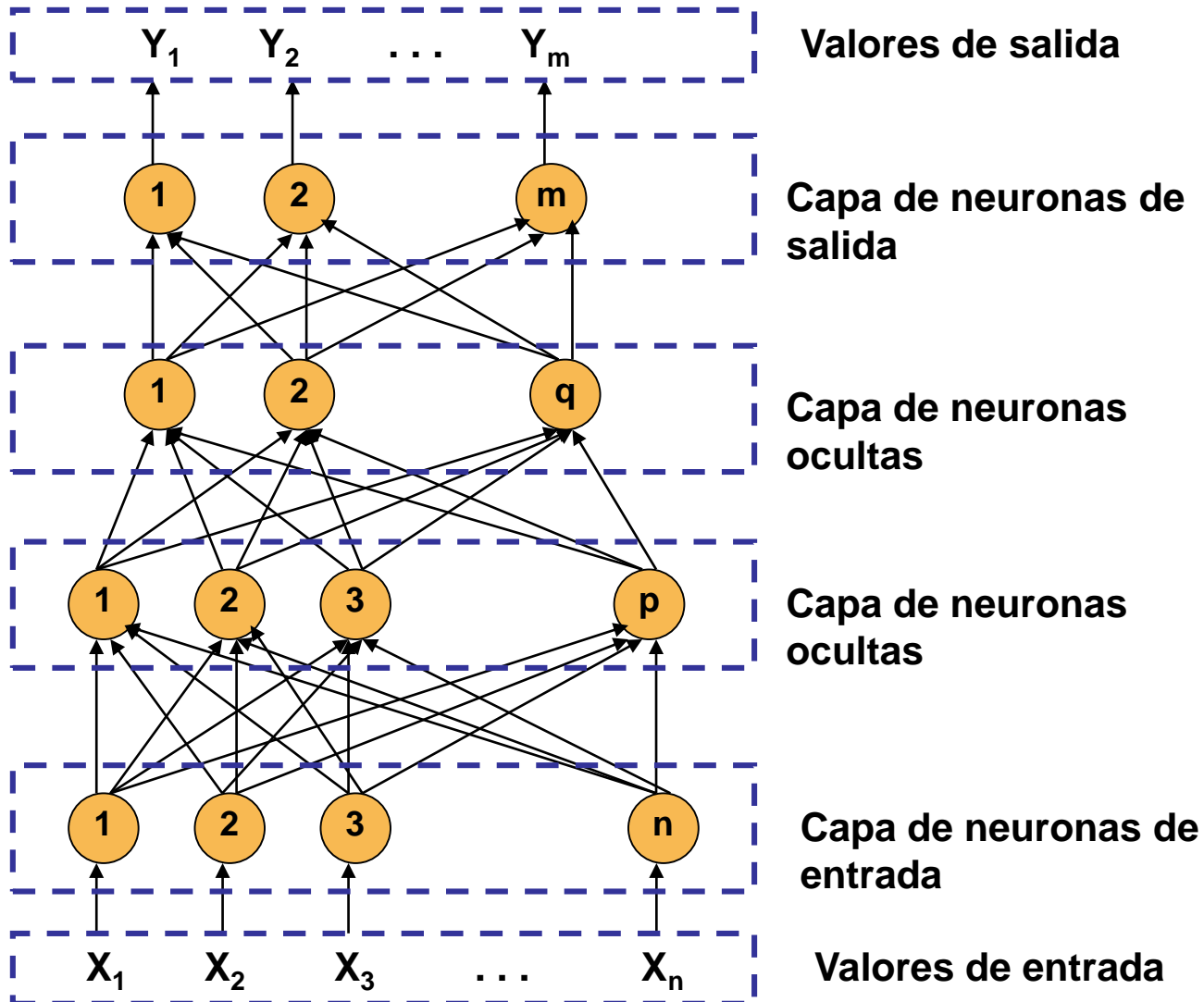
Fase de Modelado: Redes Neuronales Artificiales

Componentes y funciones de una red neuronal

- Arquitectura de la red neuronal: estratos (capas) de neuronas y patrón de interconexión.
- Función \sum , para calcular el valor de activación de cada neurona en la red neuronal.
- Función $f(\sum) = f(net)$, para calcular el valor de salida o disparo de cada neurona en la red neuronal.
- Expresiones para el algoritmo de aprendizaje: supervisado o no supervisado.

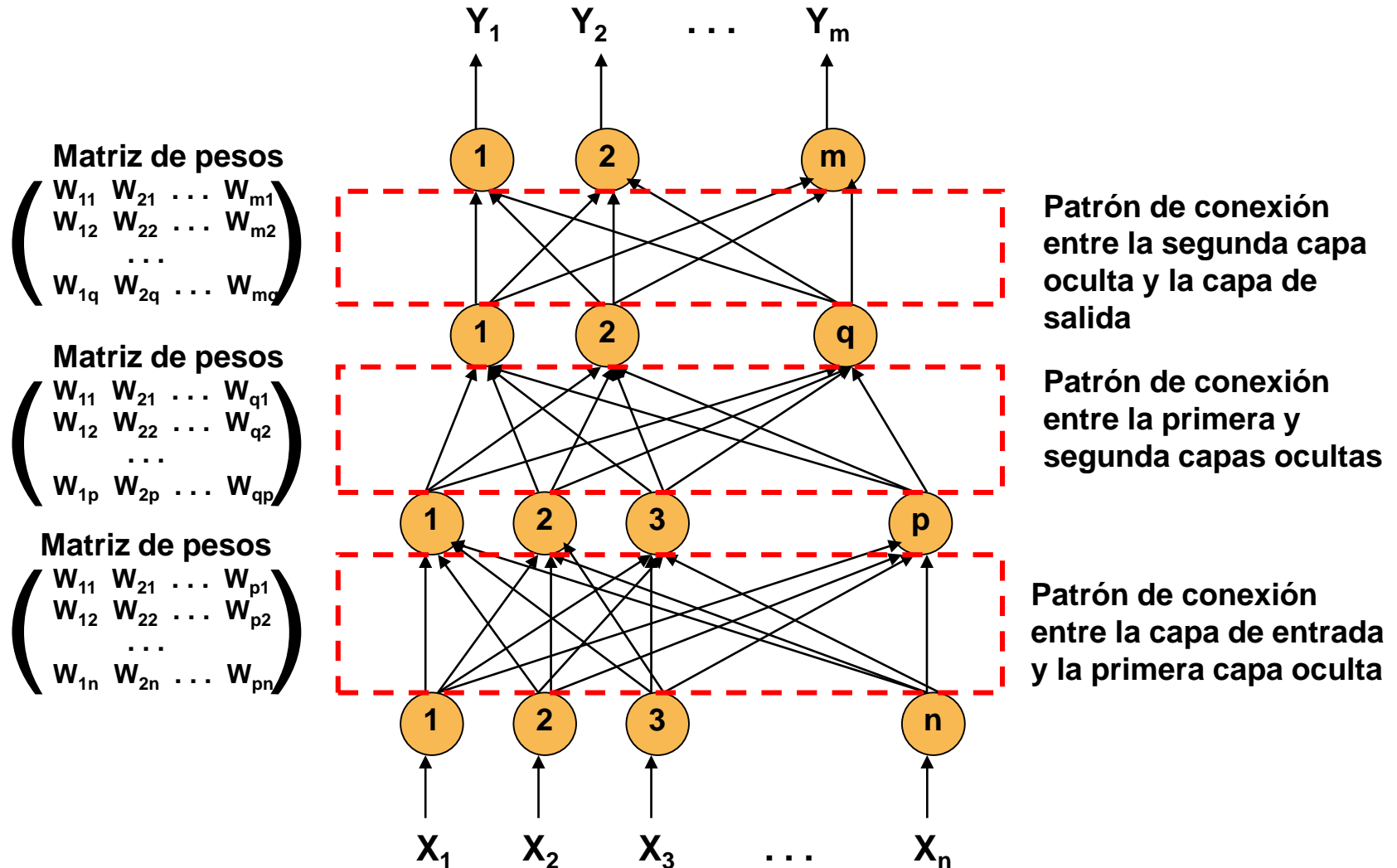
Fase de Modelado: Redes Neuronales Artificiales

Arquitecturas de redes neuronales



Fase de Modelado: Redes Neuronales Artificiales

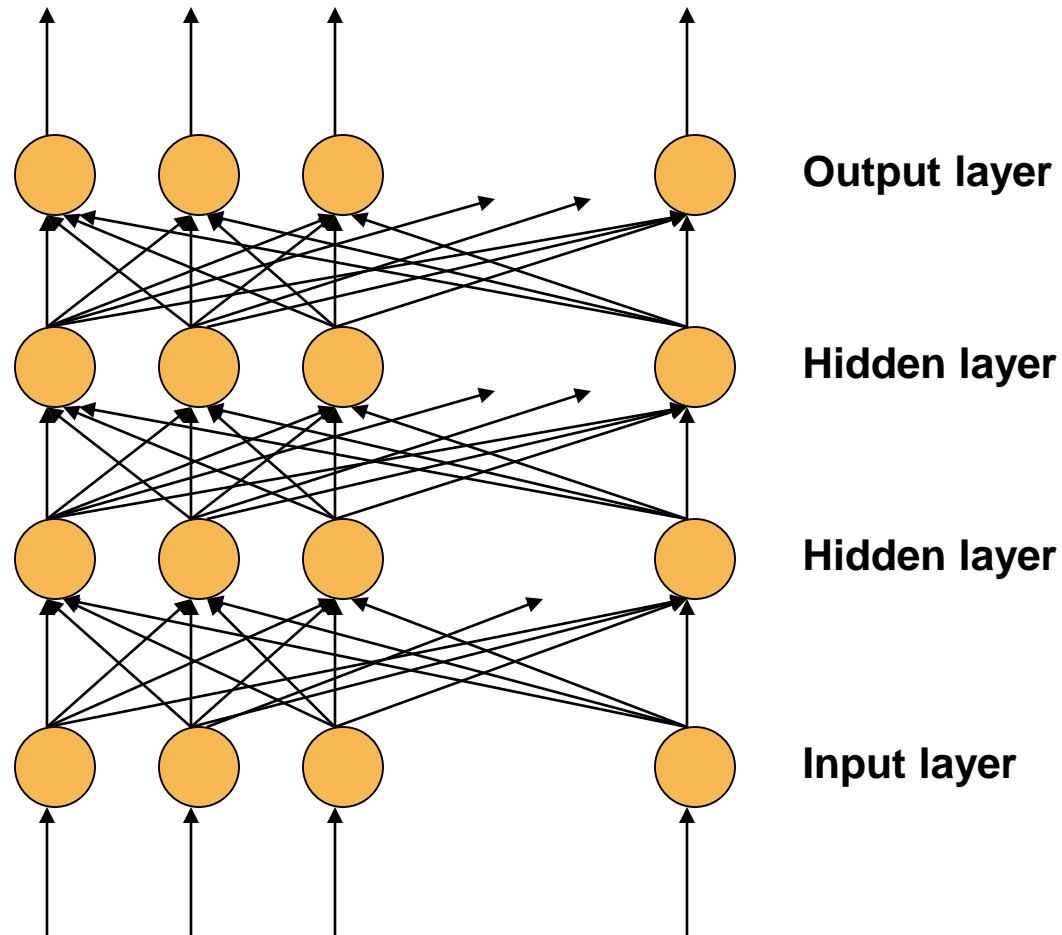
Arquitecturas de redes neuronales



Fase de Modelado: Redes Neuronales Artificiales

Arquitecturas de redes neuronales

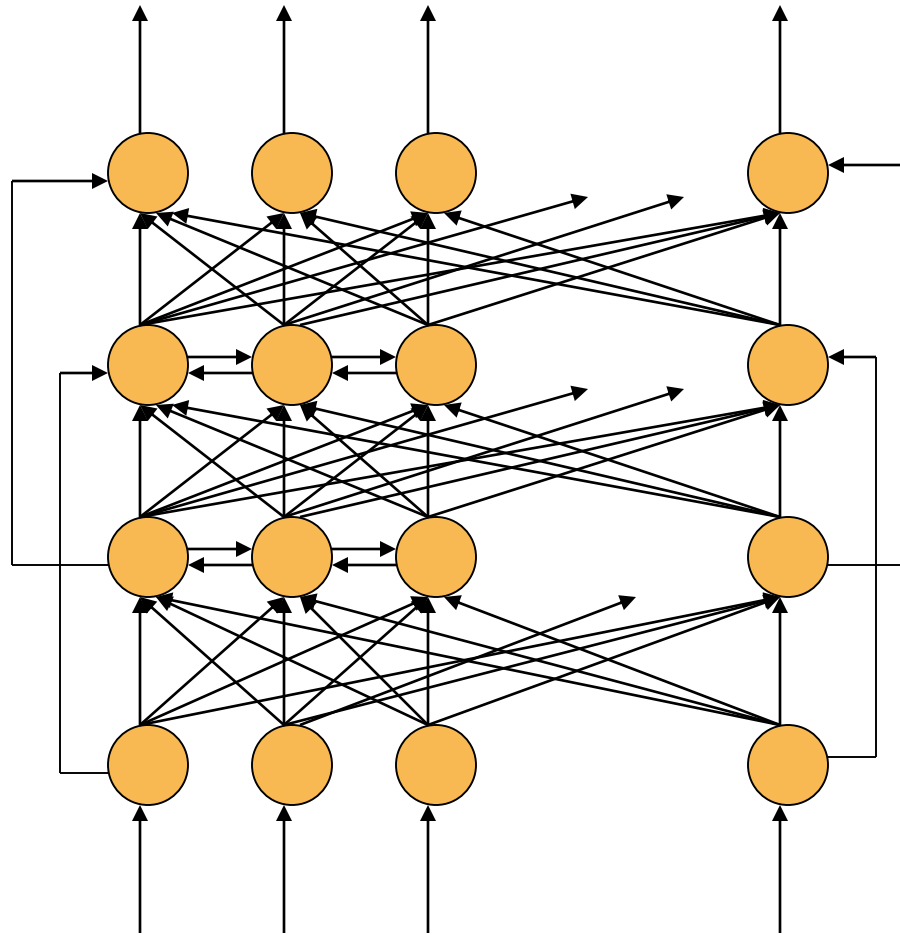
Redes *feedforward*
(conexión hacia adelante): las conexiones son permitidas sólo desde neuronas en la capa i hacia neuronas en la capa $i+1$.



Fase de Modelado: Redes Neuronales Artificiales

Arquitecturas de redes neuronales

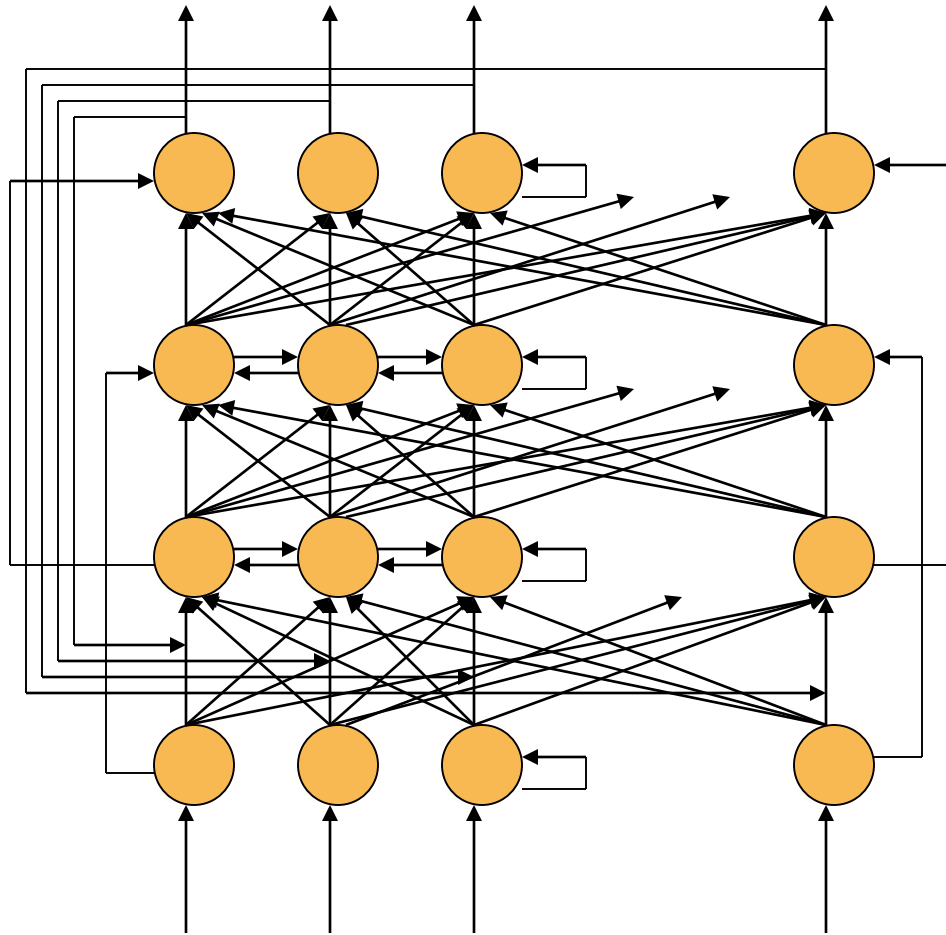
En este tipo de red neuronal no se permiten conexiones entre una capa j a una capa k , si $j > k$. Todos los otros tipos de conexiones son permitidas.



Fase de Modelado: Redes Neuronales Artificiales

Arquitecturas de redes neuronales

Redes estratificadas completamente conectadas: en este tipo de red cada neurona puede ser conectada con cualquier otra neurona, y los patrones de conexión resultantes pueden ser *feedforward*, *feedback*, intra-estrato o *feedback* (retroalimentación).



Funciones Σ o *net*

La función Σ o función *net* determina cómo las entradas a la red $\{X_j; 1 \leq j \leq N\}$ son combinadas al interior de la neurona artificial. En la gran mayoría de los modelos de neuronas artificiales, todas las entradas X_j son pesadas por la eficiencia de la transmisión sináptica $\{W_j; 1 \leq j \leq N\}$ y entonces son sumadas. En este caso el valor o magnitud que ingresa a la neurona viene dado por la siguiente expresión:

$$net = \sum_{j=1}^N W_j X_j + \theta$$

El parámetro θ es conocido como sesgo y es utilizado para modelar el umbral.

Fase de Modelado: Redes Neuronales Artificiales

Funciones de activación

Función de activación	Expresión
Función Identidad	$f(net) = net$
Función Umbral	$f(net) = \begin{cases} 1 & \text{if } net > \theta \\ 0 & \text{if } net < \theta \end{cases}$
Función Umbral Lineal	$f(net) = \begin{cases} a & \text{if } net < c \\ b & \text{if } net > d \\ a + ((net - c)(b - a)) / (d - c) & \text{if otherwise} \end{cases}$
Función Sigmoidea	$f(net) = \frac{1}{1 + e^{-net}}$
Función Tangente Hiperbólica	$f(net) = \tanh(net)$
Función Gaussiana	$f(net) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\left(\frac{net - u}{\sigma}\right)^2\right)$

Algoritmos de aprendizaje

➤ Supervised learning

- ☐ *Perceptrons*.
- ☐ *Multilayer Perceptrons* (MLP): algoritmo de aprendizaje por retropropagación de errores (backpropagation algorithm).

➤ Unsupervised learning

- ☐ Aprendizaje no competitivo.
- ☐ Aprendizaje competitivo (*winner takes all*): inspirados en la regla de *Hebb*.

Algoritmos de aprendizaje

➤ Regla de aprendizaje del Perceptron(aprendizaje supervisado)

$$W_{jk}(t+1) = W_{jk}(t) + \eta(Y_j^*(t) - Y_j(t))X_k(t)$$

➤ Regla de aprendizaje *Hebbiana* (aprendizaje no supervisado)

$$W_{jk}(t+1) = W_{jk}(t) + \varepsilon Y_j(t)X_k(t)$$

Notación:

j, k, \dots la unidad j, k, \dots

X_k la salida de la unidad k

Y_j la salida de la unidad j

Y_j^* la salida deseada de la unidad j

W_{jk} el peso de la conexión desde la unidad k a la unidad j

$0 < \eta, \varepsilon < 1$ tasas de aprendizaje

Aprendizaje supervisado

- Principales aplicaciones caracterizadas por aprendizaje supervisado:
 - ☐ Clasificación.
 - ☐ Predicción.
 - ☐ Memoria auto-asociativa (asociación de un patrón consigo mismo).
 - ☐ Memoria hetero-asociativa (asociación con uno o más patrones diferentes).

Aprendizaje no supervisado

- Cuando se utiliza el aprendizaje no supervisado, entonces la red neuronal intenta descubrir características y patrones especiales a desde los datos de entrada sin utilizar ninguna señal supervisora externa. Entre las principales aplicaciones de este tipo de aprendizaje se encuentran:
 - ❑ Agrupamiento.
 - ❑ Cuantificación vectorial (*vector quantization*).
 - ❑ Distribuciones de probabilidad.
 - ❑ Extracción de características.

Modelos clásicos de redes neuronales artificiales

- Perceptron.
- Perceptron Multi-Estrato (*Multi-Layer Perceptron*).
- Red Neuronal Recurrente con Retropropagación de Errores (*Recurrent Backpropagation Network*).
- Redes del tipo “El Ganador se lo Lleva Todo” (*Winner-Take-All Networks*).
- Red de Hopfield (*Hopfield Networks*).
- Mapas Auto-Organizativos de Kohonen (*Kohonen Self Organizing Maps*).

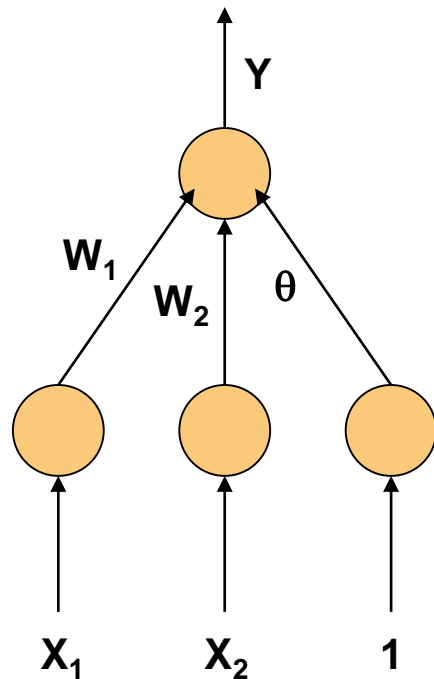
Introducción al Perceptron

- Un Perceptron puede ser representado por una única neurona con una función net del tipo $\sum_{i=1}^N W_i X_i$ y una función de activación Umbral.
- El Perceptron puede ser visto como la neurona de *McCulloch y Pitts* más un algoritmo de aprendizaje supervisado.
- El valor de salida de una única neurona sólo puede determinar a cuál de dos clases pertenece cada patrón de entrada (separabilidad lineal).
- Los valores correctos de los W_i del vector de peso W son estimados usando un algoritmo de aprendizaje secuencial, conocido como Algoritmo de Aprendizaje del Perceptron.

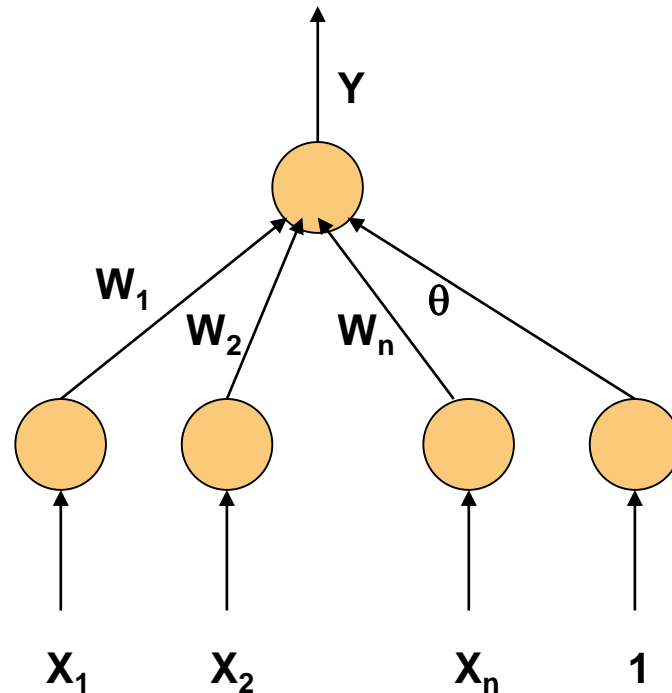
Fase de Modelado: Redes Neuronales Artificiales

Arquitectura del Perceptron

Un Perceptron simple de dos entradas y una salida



Perceptron genérico para el espacio n-dimensional



$$net = \sum_{j=1}^n W_j X_j + \theta \quad f(net) = \begin{cases} 1 & \text{si } net > 0 \\ 0 & \text{en otro caso} \end{cases}$$

Fase de Modelado: Redes Neuronales Artificiales

Características del Perceptron

Capas de neuronas	Una capa de neuronas de entrada Una capa de neuronas de salida
Patrón de conexión	<i>Feedforward</i>
Tipos de valores de entrada	Reales
Función de activación	<i>Hard limiter</i>
Método de aprendizaje y algoritmo de aprendizaje	Supervisado $W_{jk}(t+1) = W_{jk}(t) + \eta(Y_j^*(t) - Y_j(t))X_k(t)$

Fase de Modelado: Redes Neuronales Artificiales

Perceptron Multi-Estrato (*MLP*) y algoritmo de retropropagación de errores (*backpropagation*)

- Un *perceptron* multi-estrato es una red neuronal estratificada, *feedforward*, que contiene un estrato de entrada, al menos un estrato oculto, y un estrato de salida.
- Las neuronas del estrato de entrada sólo transmiten el patrón de entrada a las neuronas del estrato oculto (o primer estrato oculto, en caso de que haya más de uno), comúnmente éstas no ejecutan ningún tipo de cómputo.
- Las neuronas en los estratos ocultos y de salida tienen valores de entrada y de salida continuos, una función *net*, y una función de activación no lineal.

Fase de Modelado: Redes Neuronales Artificiales

Perceptron Multi-Estrato (*MLP*) y algoritmo de retropropagación de errores (*backpropagation*)

- El algoritmo de retropropagación de errores (*backpropagation*) es una implementación de la técnica de los mínimos cuadrados, que modifica los pesos de las conexiones de la red neuronal para minimizar el error cuadrático medio entre la salida deseada y la salida actual de la red neuronal.

Fase de Modelado: Redes Neuronales Artificiales

Perceptron Multi-Estrato (MLP) y algoritmo de retropropagación de errores (backpropagation)

Notation:

L : índice del estrato

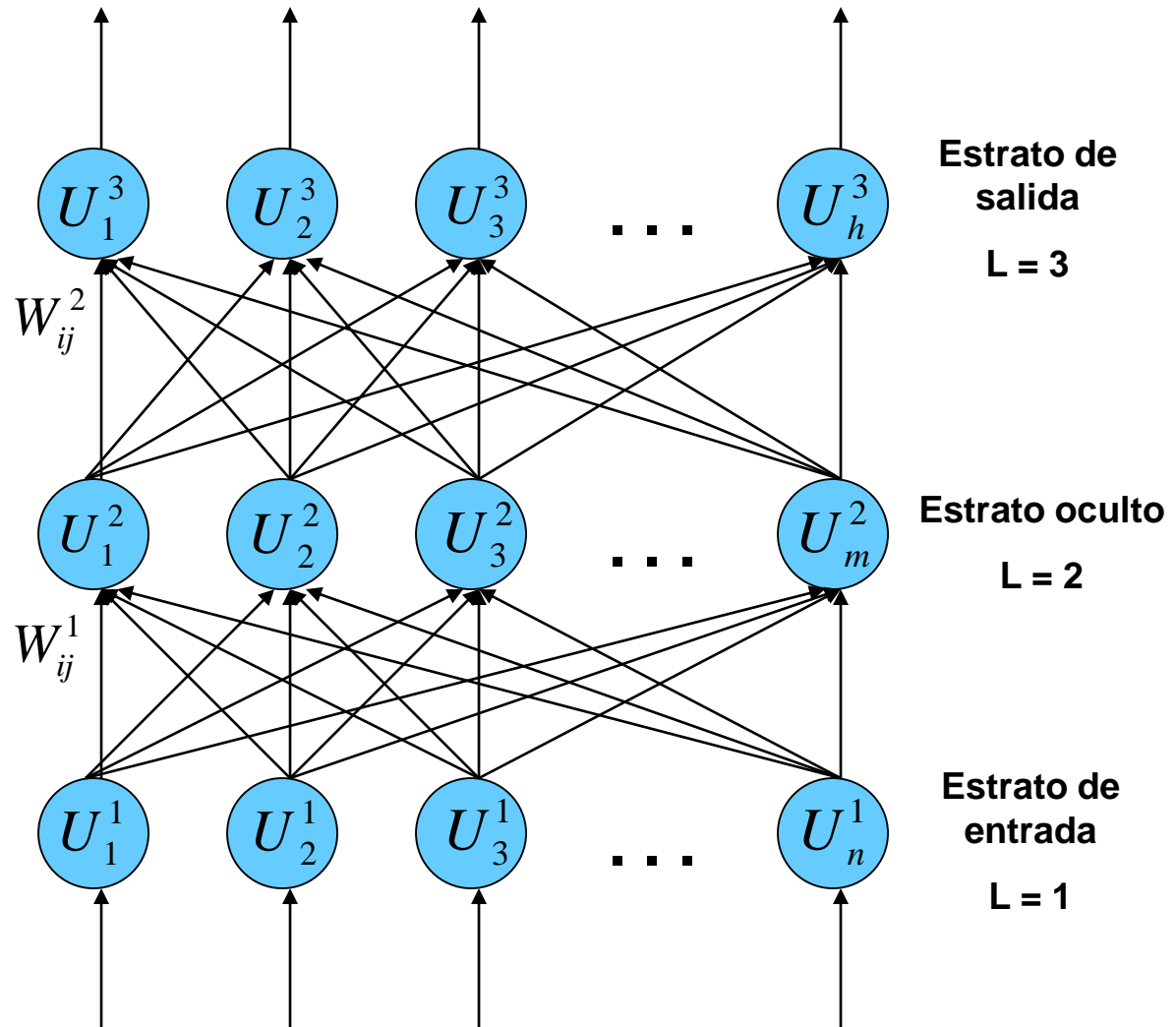
U_i^L : neurona i en el estrato L

W_{ij}^L : el peso de la conexión desde
la neurona i en el estrato L a la
neurona j en el estrato $L+1$

n : número de neuronas en el
estrato de entrada

m : número de neuronas en el
estrato oculto

h : número de neuronas en el
estrato de salida



Fase de Modelado: Redes Neuronales Artificiales

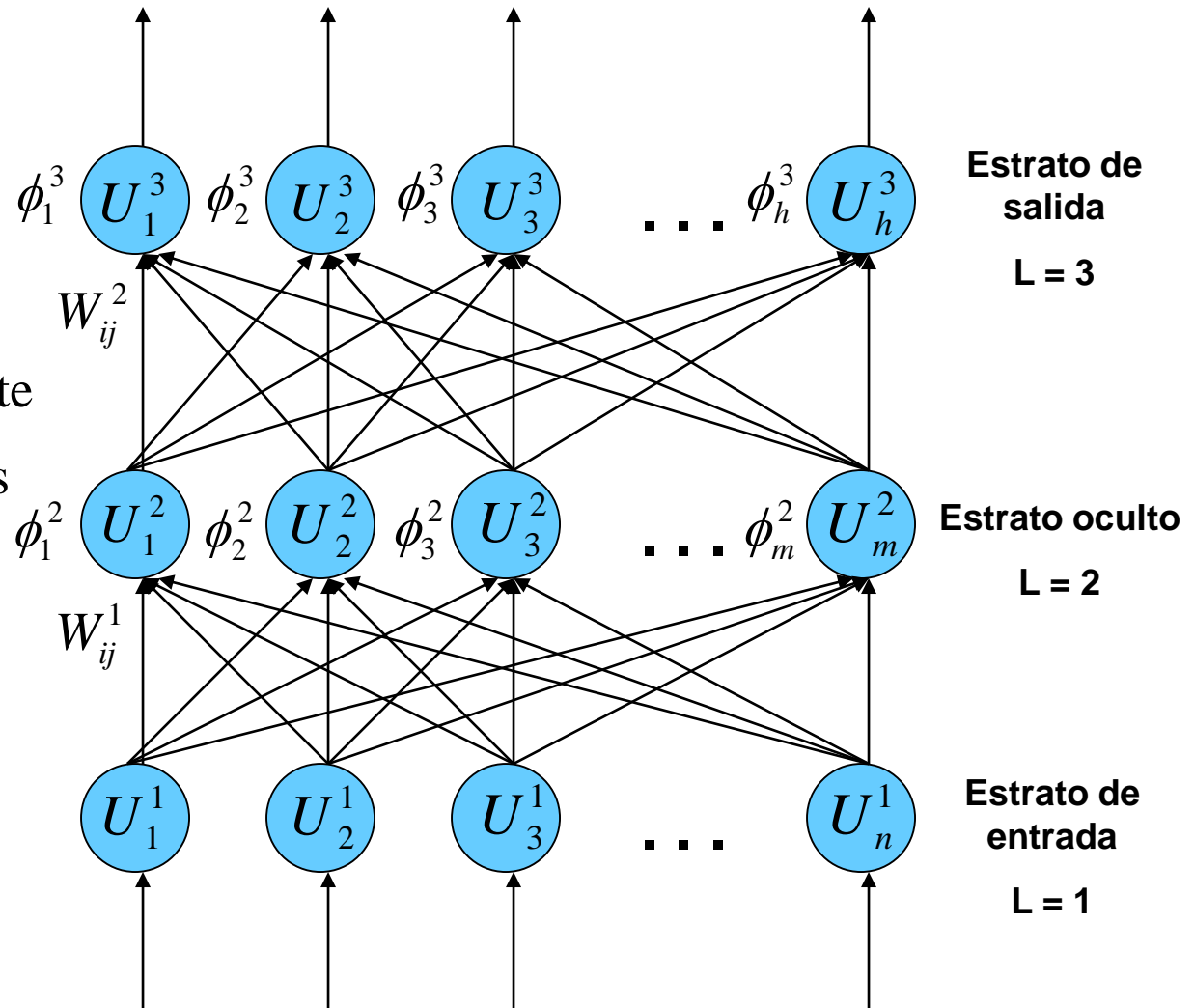
Características del Perceptron Multi-Estrato (MLP)

Estratos o capas de neuronas	Un estrato de entrada Uno o más estratos ocultos Un estrato de salida
Patrón de conexión	<i>Feedforward</i>
Tipos de valores de entrada	Reales
Función de activación	Sigmoidea
Método y algoritmo de aprendizaje	Supervisado Algoritmo de retropropagación de errores (<i>backpropagation</i>)

Fase de Modelado: Redes Neuronales Artificiales

Algoritmo de retropropagación de errores (*backpropagation*)

ϕ_i^L es calculado iterativamente
a partir de ϕ_j^{L+1} y de los pesos
en el estrato $L + 1$

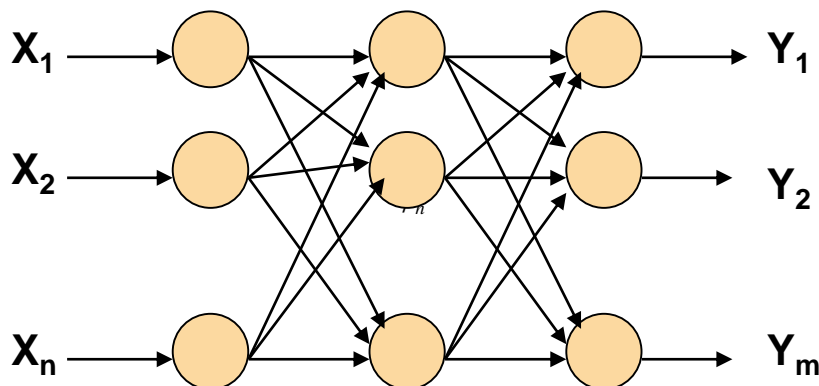


Fase de Modelado: Redes Neuronales Artificiales

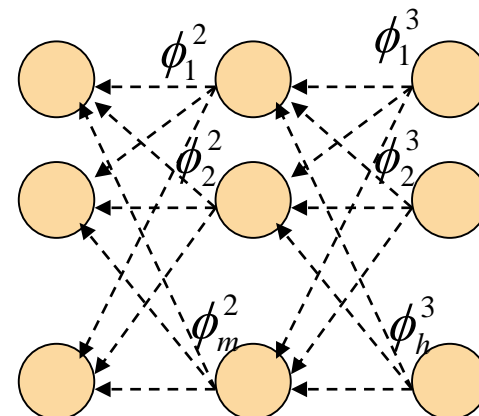
Propagación hacia adelante (*forward*) y propagación hacia atrás (*backward*)

- **Propagación hacia adelante (*forward*):** las entradas son proporcionadas desde el estrato de entrada y propagadas a través de los estratos ocultos hasta el estrato de salida.
- **Propagación hacia atrás (*backward*):** el error es inicialmente calculado en el estrato de salida y propagado hacia atrás, a través de los estratos ocultos, para calcular la modificación a efectuar en los pesos.

Propagación hacia adelante (*forward*)



Propagación hacia atrás (*backward*)



Fase de Modelado: Redes Neuronales Artificiales

La fórmula del error *backpropagation*

$$E = \sum_{k=1}^K [e(k)]^2 = \sum_{k=1}^K [d(k) - y(k)]^2 \text{ es la suma del error cuadrático}$$

$$W_{ij}^{L,t+1} = W_{ij}^{L,t} + \alpha \Delta W_{ij}^L$$

$$\Delta W_{ij}^L \text{ es proporcional a } -\frac{\delta E}{\delta W_{ij}^L}$$

$$-\frac{\delta E}{\delta W_{ij}^L} = \phi_j^{L+1} O_i^L$$

$$\text{donde : } \phi_i^L = \begin{cases} (d_i^L - y_i^L) (F_i^L(A_i^L))' & \text{si } L = \text{índice del estrato de salida} \\ (F_i^L(A_i^L))' \sum_j \phi_j^{L+1} W_{ij}^L & \text{de lo contrario} \end{cases}$$

$$W_{ij}^{L,t+1} = W_{ij}^{L,t} + \alpha \phi_j^{L+1} Y_i^L$$

F_i^L : función de activación de la neurona i en el estrato L

Fase de Modelado: Redes Neuronales Artificiales

Algoritmo de retropropagación de errores (*backpropagation*)

Establecer el valor de los parámetros:

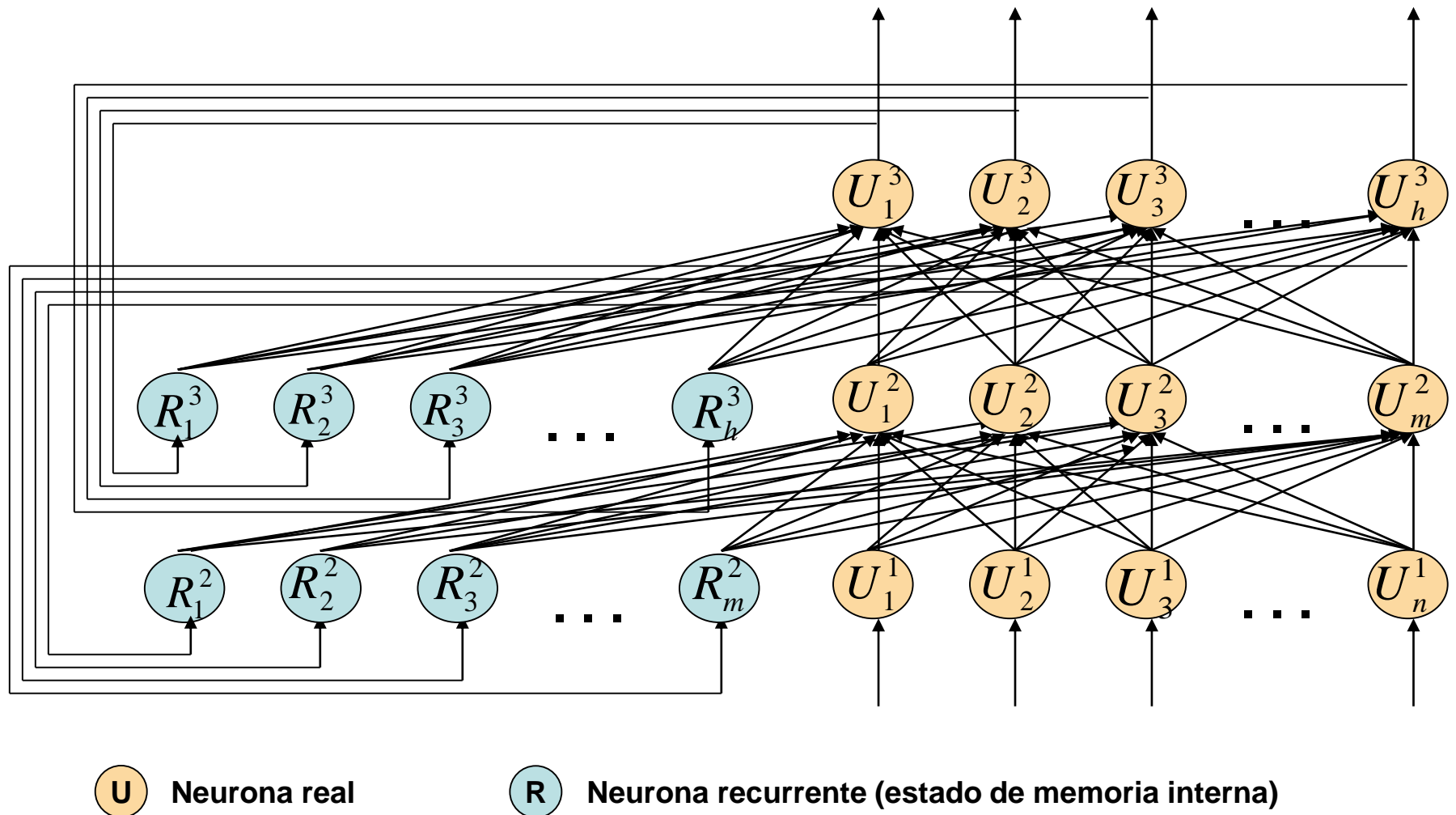
- ☐ Inicialización aleatoria de los pesos de todas las conexiones en la red.
- ☐ Actualización de pesos.
- ☐ Número de ejemplos.
- ☐ Tasa de aprendizaje.
- ☐ Máximo error aceptable durante la fase de aprendizaje.
- ☐ Máximo error aceptable durante la fase de generalización o respuesta del modelo.
- ☐ Número de estratos (capas) ocultos y número de neuronas en cada uno de éstos.

Introducción a las Redes Neuronales Recurrentes (*RNN*)

- Las redes neuronales recurrentes tienen conexiones (hacia atrás) *feedback*. Es decir, contienen conexiones desde las neuronas del estrato de salida hacia las neuronas de los estratos ocultos y de entrada.
- Este tipo de red neuronal maneja la relación temporal de los patrones de entrada, manteniendo estados de memoria interna (ver las neuronas recurrentes en la siguiente figura).
- El algoritmo de aprendizaje típicamente usado es el de retropropagación de errores a través del tiempo.
- Este tipo de redes tiene un gran uso en problemas de aproximación de funciones, en problemas tales como pronóstico y control.

Fase de Modelado: Redes Neuronales Artificiales

Introducción a las Redes Neuronales Recurrentes (RNN)



Fase de Modelado: Redes Neuronales Artificiales

Introducción a las Redes Neuronales Recurrentes (*RNN*)

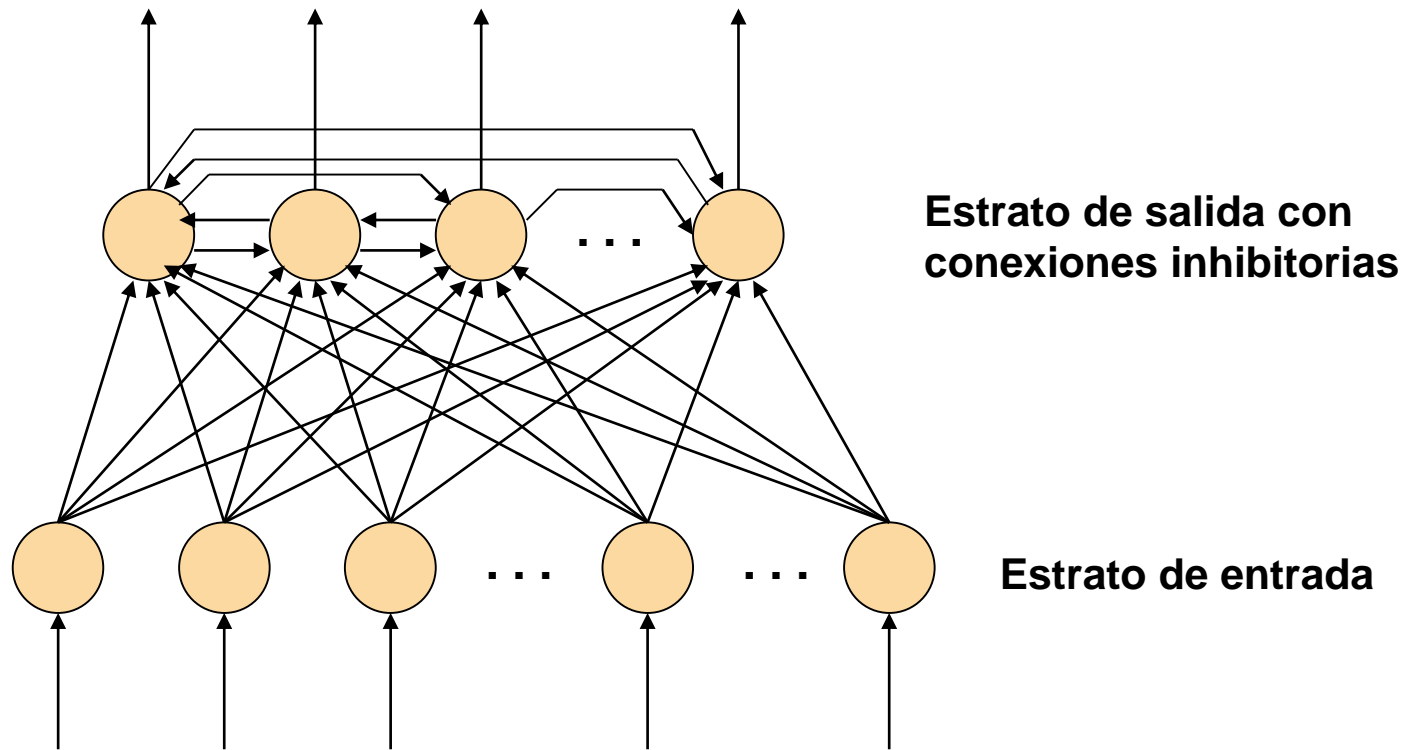
Estratos de neuronas	Un estrato de entrada Uno o más estratos ocultos Un estrato de salida
Patrón de conexión	<i>Feed forward y feed backward</i>
Tipos de valores de entrada	Reales
Función de activación	Sigmoidea
Método de aprendizaje y algoritmo de aprendizaje	Supervisado Retropropagación de errores a través del tiempo

Introducción a las Redes *Winner-Takes-All*

- Este tipo de red neuronal también se organiza en estratos o capas. Existen conexiones excitatorias entre las neuronas de diferentes estratos y conexiones inhibitorias entre las neuronas en el mismo estrato.
- Las neuronas en el estrato de salida compiten entre sí para representar el patrón de entrada actual, cada una inhibe a las restantes con su actual nivel de activación.
- Después de esta competencia la neurona ganadora es activada, y su vector de peso será ajustado para asemejar mucho más al patrón de entrada.
- Los pesos de las conexiones que llegan a la neurona ganadora (vector de pesos) son actualizados utilizando la regla de aprendizaje de Hebb.

Fase de Modelado: Redes Neuronales Artificiales

Introducción a las Redes *Winner-Takes-All*



Fase de Modelado: Redes Neuronales Artificiales

Características de las Redes *Winner-Takes-All*

Estratos de neuronas	Un estrato de entrada Un estrato de salida con conexiones inhibitorias
Patrón de conexión	<i>Feedforward</i> Conexiones inhibitorias entre las neuronas del estrato de salida
Tipos de valores de entrada	Reales Binarios
Función Net Función de activación	Métrica de distancia Función <i>Hard-limiter</i> (comúnmente usada)
Método de aprendizaje Regla de aprendizaje	No supervisado Regla de aprendizaje de Hebb generalizada

Introducción a las Redes de *Hopfield*

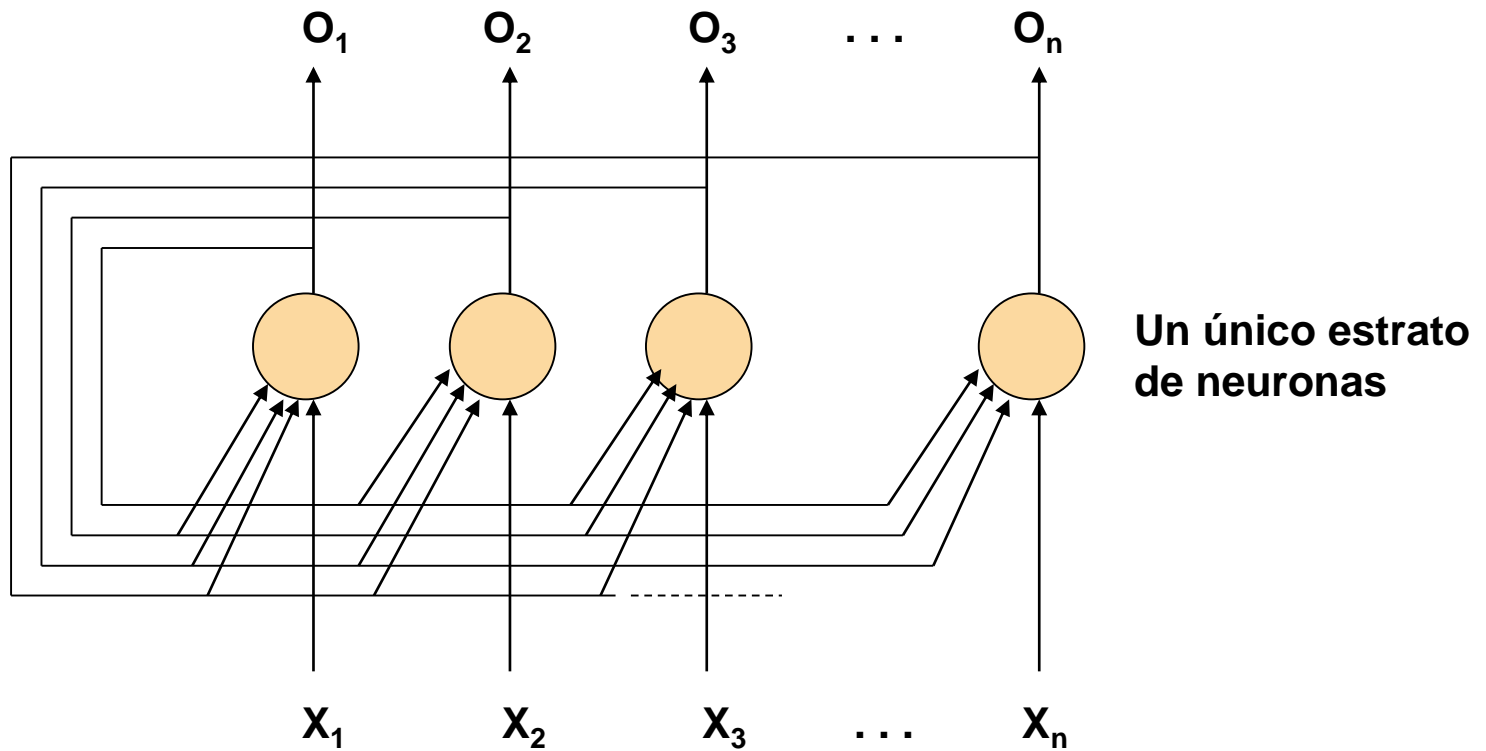
- Las redes de Hopfield son redes neuronales comúnmente usadas para tareas de auto-asociación y control.
- Es un tipo de red neuronal *feed backward* completamente conectada: cada neurona en la red tiene una conexión *backward* hacia todas las otras neuronas en la red, excepto hacia ella misma.
- Los pesos son actualizados usando la regla de *Hebb*.
- Cada neurona en la red recibe un patrón de entrada compuesto por una entrada externa y las salidas pesadas de las restantes neuronas.
- Cada neurona en la red constituye un atractor para cierto tipo de patrón específico.

Introducción a las Redes de *Hopfield*

- Cuando la red es usada para problemas de auto-asociación, el vector de salida deseado para un patrón de entrada dado consiste del patrón de atracción (atractor) almacenado más cercano.
- Cada neurona en la red ejecuta los siguientes dos pasos:
(1) computa la suma de sus correspondientes entradas externas y las salidas pesadas de las otras neuronas, y
(2) aplica una función no lineal a la suma calculada, resultando en un valor de salida de +1 o -1.
- Cuando un patrón de entrada es presentado a la red, los valores de salida de las neuronas pueden cambiar hasta alcanzar un estado en el cual la red converge hacia un atractor específico.

Fase de Modelado: Redes Neuronales Artificiales

Arquitectura de la red de *Hopfield*



Fase de Modelado: Redes Neuronales Artificiales

Los principales procesos en la red de *Hopfield*

- Almacenamiento del patrón de entrada en la red neuronal.
- El proceso de recuerdo.

Fase de Modelado: Redes Neuronales Artificiales

Características de la red de *Hopfield*

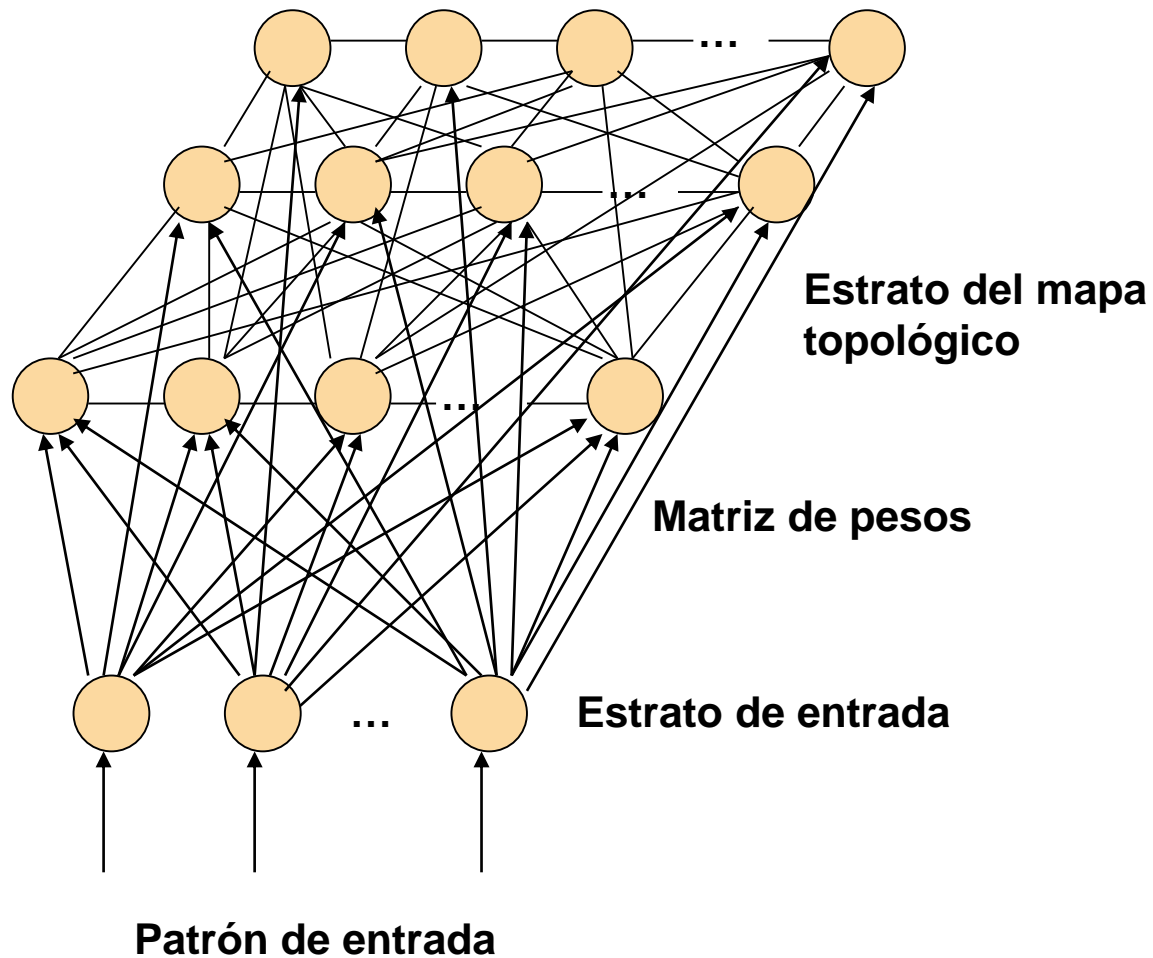
Estratos de neuronas	Un único estrato
Patrón de conexión	<i>Feedback</i> completamente conectada
Tipos de valores de entrada	Vectores discretos
Función Net Función de activación	Suma pesada Función <i>Step</i>
Método de aprendizaje Algoritmo de aprendizaje	No supervisado Algoritmo de aprendizaje Hebbiano

Introducción a los Mapas Auto-Organizativos de Kohonen

- Un Mapa Auto-Organizativo (del inglés, *Self-Organizing Map (SOM)*), desarrollado por Kohonen a inicios de los años 80's (Kohonen, 2001), es una red neuronal artificial que construye mapeos preservando la topología de los datos de entrenamiento.
- Una red SOM consiste de dos estratos: un estrato de entrada y un mapa de características (una configuración topológica que constituye el estrato de salida).
- Los nodos de la red se sintonizan específicamente con varios patrones de señales de entrada (o clases de patrones) en un modo ordenado.
- El proceso de aprendizaje de la red es competitivo y no supervisado y, como resultado de esta competencia, sólo un nodo en la red se encuentra activo a la vez, correspondiendo a cada entrada.
- Una red SOM puede ser vista como una red neuronal basada en el enfoque del agrupamiento (clustering) divisivo (Kohonen, 2001).

Fase de Modelado: Redes Neuronales Artificiales

Arquitectura de la red SOM de Kohonen



La red SOM para clasificación de patrones

- Cuando una red SOM es usada para la clasificación de patrones, los patrones de entrada son asignados a una serie de *clusters* sobre la base de la similitud existente entre el patrón de entrada y el centro de cada uno de los *clusters*, conocidos como vectores de referencia.
- Antes de que inicie la tarea de clasificación o reconocimiento de patrones, el usuario tiene que predefinir una topología de nodos (comúnmente, una rejilla bidimensional rectangular o hexagonal), donde cada nodo en la rejilla representa un *cluster*.

Fase de Modelado: Redes Neuronales Artificiales

La red SOM para la clasificación de patrones

- Inicialmente, vectores de referencia aleatorios son generados y asignados a cada nodo. Después de lo cual, inicia un proceso iterativo que abarca los siguientes pasos:
 1. Un patrón de entrada es seleccionado aleatoriamente, y el nodo (vector de referencia) más similar (que mapee más cercano) a dicho patrón de entrada es seleccionado.
 2. El vector de referencia del nodo seleccionado (en el espacio de patrones de entrada) es entonces ajustado, de forma tal que éste sea aun más similar al patrón de entrada asignado.
 3. Los vectores de referencia de los otros nodos cercanos al nodo seleccionado en la rejilla bidimensional (considerando un determinado criterio de vecindad) son también ajustados.

Fase de Modelado: Redes Neuronales Artificiales

Características de la red SOM

Estratos de neuronas	Un estrato de entrada Un estrato del tipo mapa topológico
Patrón de conexión	<i>Feedforward</i> Conexiones inhibitorias entre los nodos en el mapa de características
Tipos de valores de entrada	Reales, binarios
Función Net Función de activación	Distancia Euclidiana Sigmoidea
Método de aprendizaje Algoritmo de aprendizaje	Competitivo Algoritmo de aprendizaje auto-organizativo

Aplicaciones de las redes neuronales artificiales

- Clasificación de patrones.
- Asociación de patrones.
- Reconocimiento de patrones.
- Problemas de predicción.
- Problemas de diagnóstico.
- Procesamiento de señales.
- Problemas de agrupamiento (*clustering*).
- Problemas de control y monitoreo.

Fase de Modelado: Redes Neuronales Artificiales

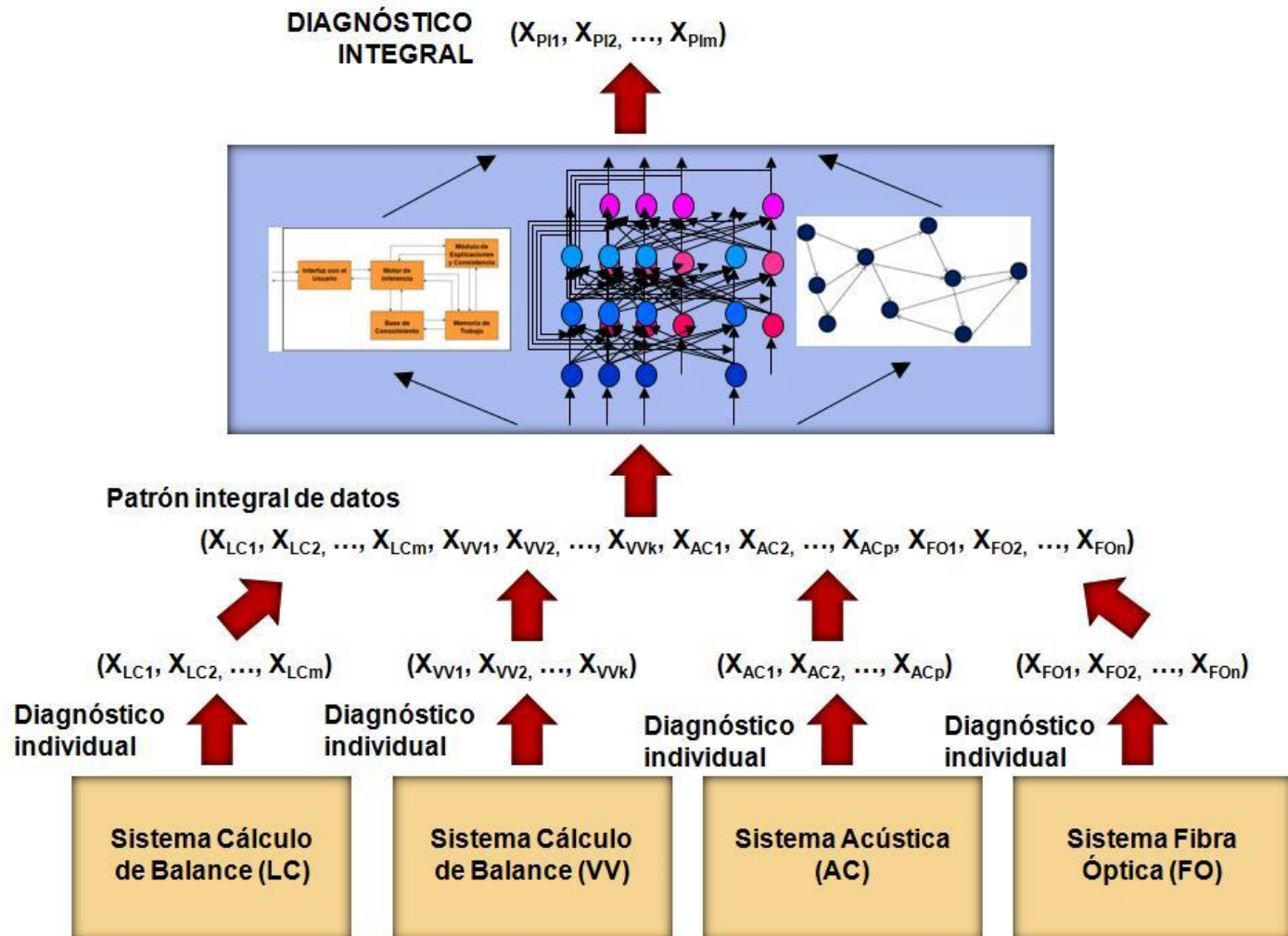
Redes neuronales para tareas de clasificación

Tarea	Modelo de red neuronal	Dominio de aplicación
Clasificación	<ul style="list-style-type: none">▪ Perceptron.▪ Perceptron multi-estrato con algoritmo de retropropagación de errores.▪ Red neuronal recurrente.	<p>En cualquier dominio donde se requiera la clasificación de patrones, como por ejemplo:</p> <ul style="list-style-type: none">❖ Diagnóstico médico❖ Ciencias e ingeniería❖ Análisis del lenguaje hablado❖ Mercado bursátil❖ Economía y finanzas❖ ...

Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de clasificación

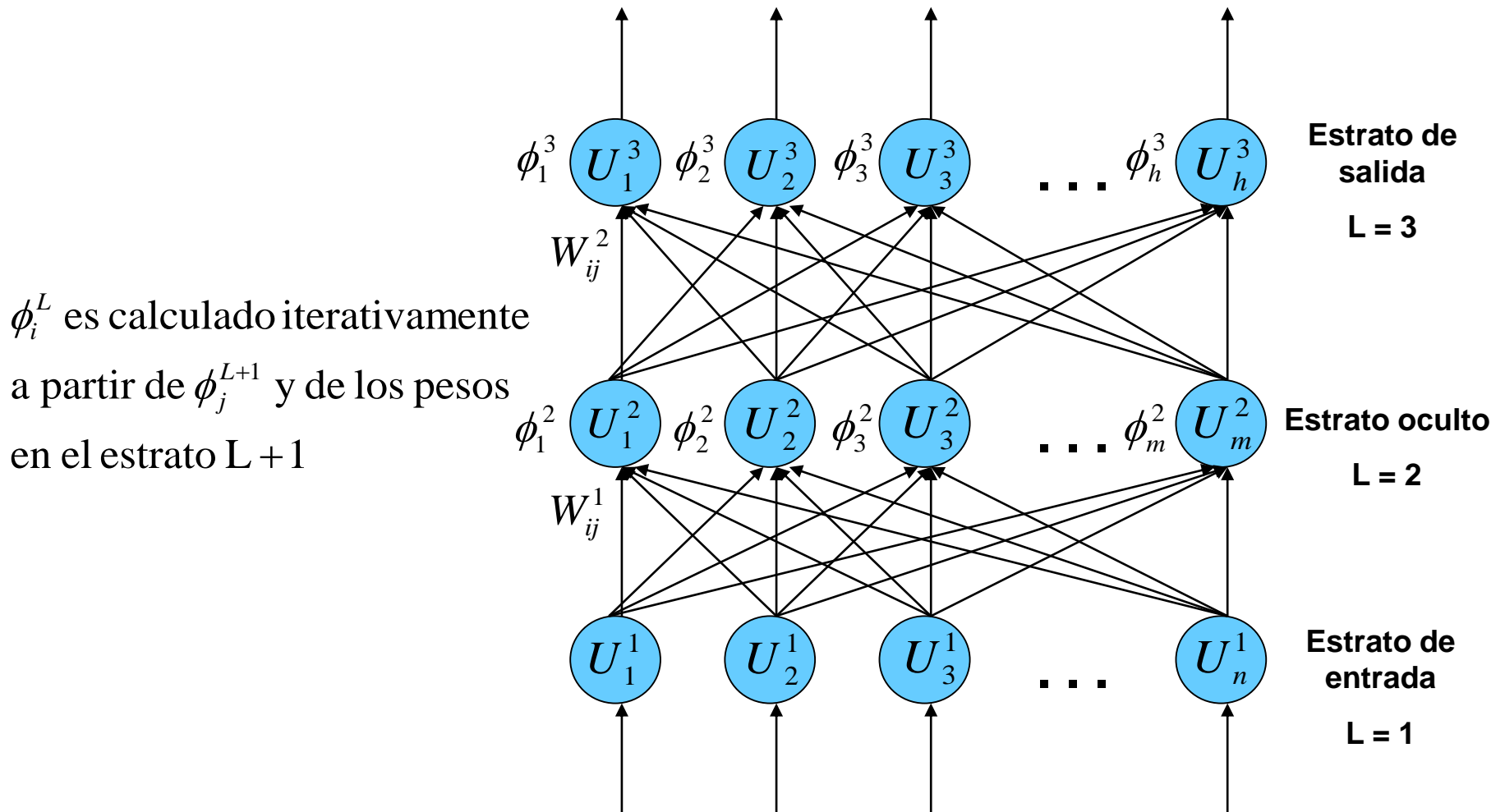
Sistema para la Detección de Fugas y Tomas Clandestinas en Ductos que transportan Hidrocarburos (González et A., 2013)



Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de clasificación

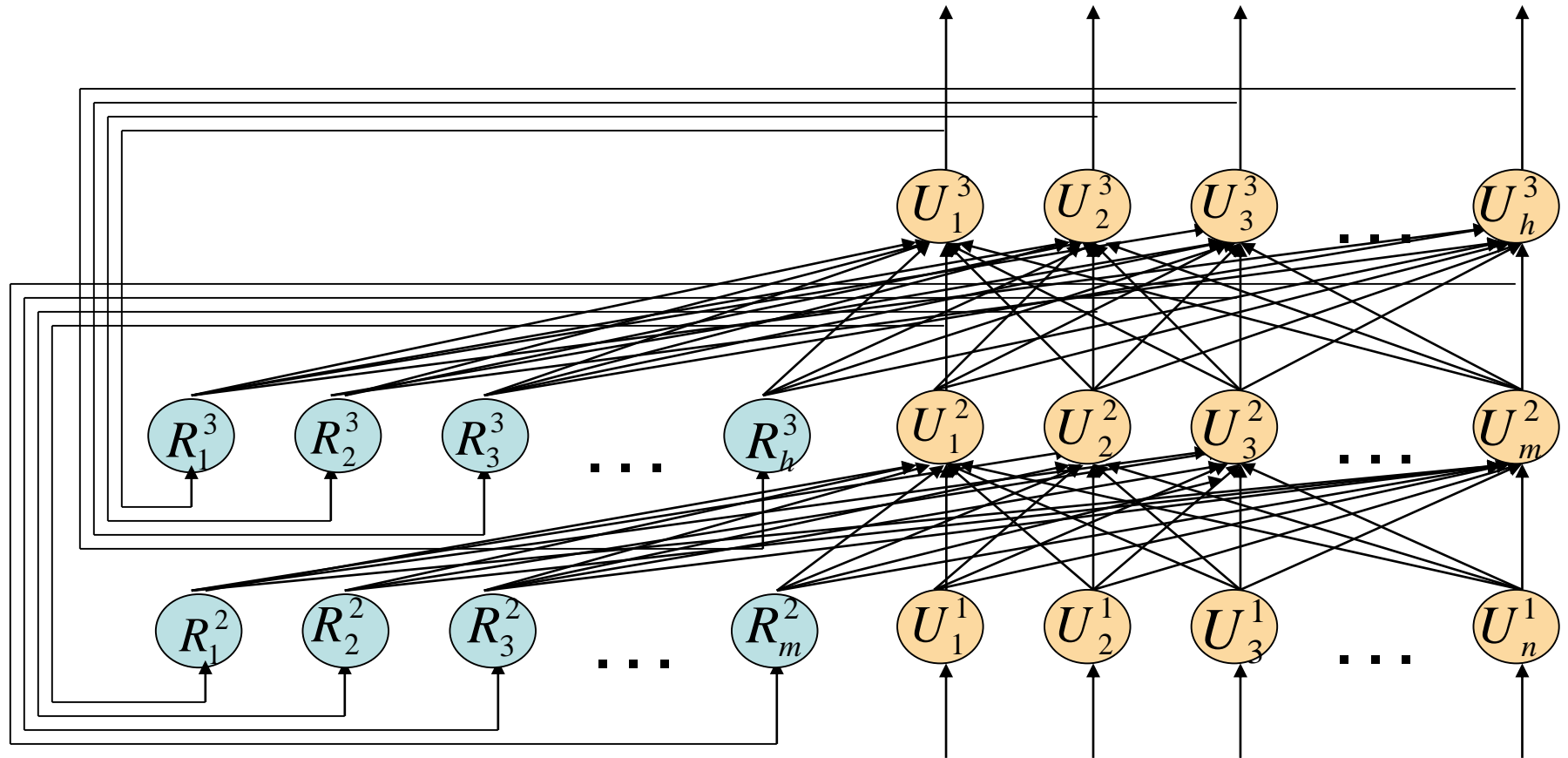
Sistema para la Detección de Fugas y Tomas Clandestinas en Ductos que transportan Hidrocarburos (González et A., 2013)



Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de clasificación

Sistema para la Detección de Fugas y Tomas Clandestinas en Ductos que transportan Hidrocarburos (González et A., 2013)



U Neurona real

R Neurona recurrente (estado de memoria interna)

Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para procesamiento de señales

Tarea	Modelo de red neuronal	Dominio de aplicación
Procesamiento de señales visto como clasificación de patrones	<ul style="list-style-type: none">▪ Perceptron multi-estrato con algoritmo de retropropagación de errores.▪ Red neuronal recurrente.	<ul style="list-style-type: none">❖ Reconocimiento de señales biomédicas❖ Reconocimiento del habla❖ Reconocimiento de caracteres ópticos❖ Reconocimiento de señales acústicas❖ Reconocimiento de rostros❖ ...

Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de asociación de patrones

Tarea	Modelo de red neuronal	Dominio de aplicación
Asociación de patrones	<ul style="list-style-type: none">▪ Red de Hopfield▪ Perceptron multi-estrato con algoritmo de retropropagación de errores	<ul style="list-style-type: none">❖ Tareas de auto-asociación o memoria asociativa. Cualquiera que conlleve a la asociación de un patrón consigo mismo. Por ejemplo, recuperación de una imagen completa a partir de la imagen corrompida, reconocimiento de firmas.❖ Tareas de hetero-asociación. Cualquiera que conlleve a la asociación de un patrón con otro patrón diferente. Por ejemplo, asociación de patrones en investigaciones biomédicas.

Fase de Modelado: Redes Neuronales Artificiales

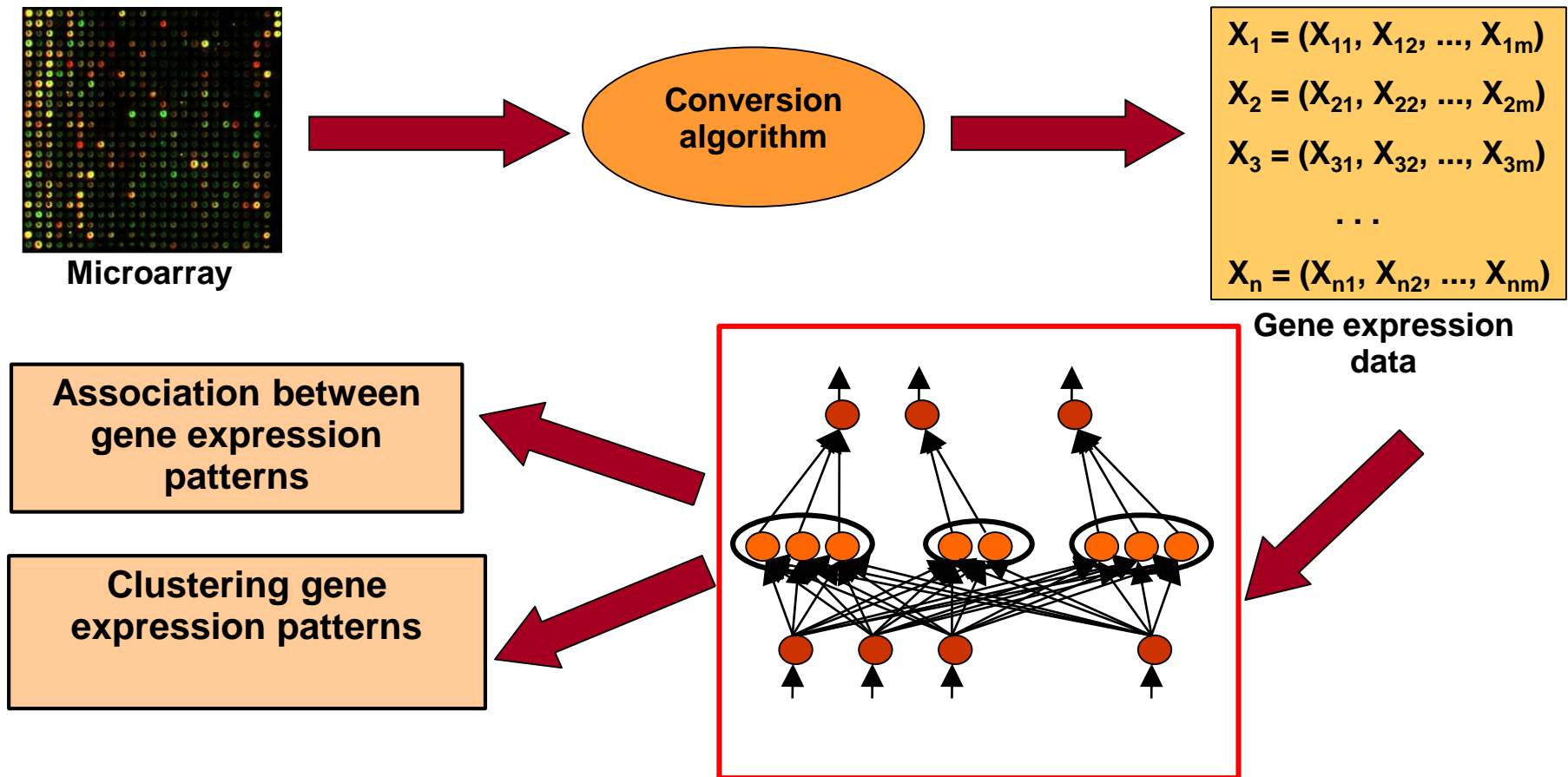
Redes neuronales para tareas de agrupamiento (*clustering*)

Tarea	Modelo de red neuronal	Dominio de aplicación
Agrupamiento (<i>clustering</i>)	<ul style="list-style-type: none">▪ Mapas auto-organizativos (SOM).▪ Redes neuronales evolutivas.▪ Modelos de redes neuronales con aprendizaje no supervisado.	❖ En cualquier tarea que requiera la clasificación de patrones, para los cuales no se conoce a priori la clase o categoría a la que pertenecen. Por ejemplo, en ciencias biomédicas, 1) en la interpretación de patrones de expresión de genes, 2) en el agrupamiento de vectores de descriptores moleculares, representando secuencias de proteínas; en el reconocimiento de emociones en el lenguaje hablado.

Redes neuronales para tareas de agrupamiento (*clustering*)

An Evolving Neural Network for the Interpretation of Gene Expression Patterns

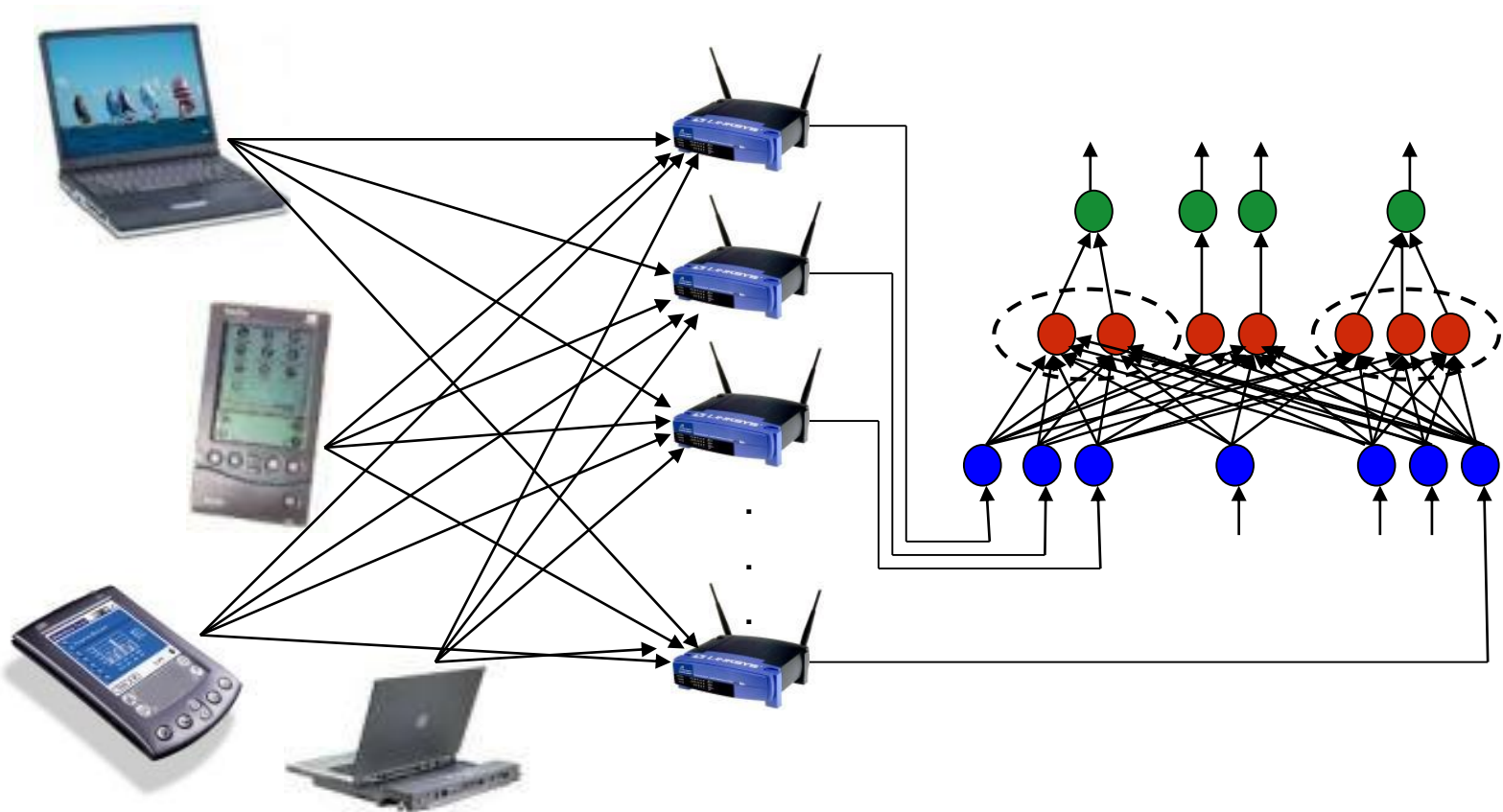
Márquez, M.C., González, P.P. and Lagúnez, J. (2005). OMICS A Journal of Integrative Biology, Volume 9, Number 2, pages 209-217



Redes neuronales para tareas de agrupamiento (*clustering*)

A Neural Network for RF-based User Location in Wireless Systems

González, P.P., Gramellini, C., Buda, C., Lucchi, M. and Dardari, D.
DEIS, Università degli Studi di Bologna, Italia, 2003



Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de procesamiento de imágenes

Tarea	Modelo de red neuronal	Dominio de aplicación
Procesamiento de imágenes	<ul style="list-style-type: none">▪ Redes feedforward con el algoritmo de retropropagación de errores (<i>backpropagation</i>), para reconocimiento de patrones.▪ Redes neuronales no supervisadas, para extracción de características.	<ul style="list-style-type: none">❖ Análisis de imágenes biomédicas: detección y caracterización de patrones de enfermedades.❖ Cuantificación, segmentación, comprensión y restauración de imágenes.

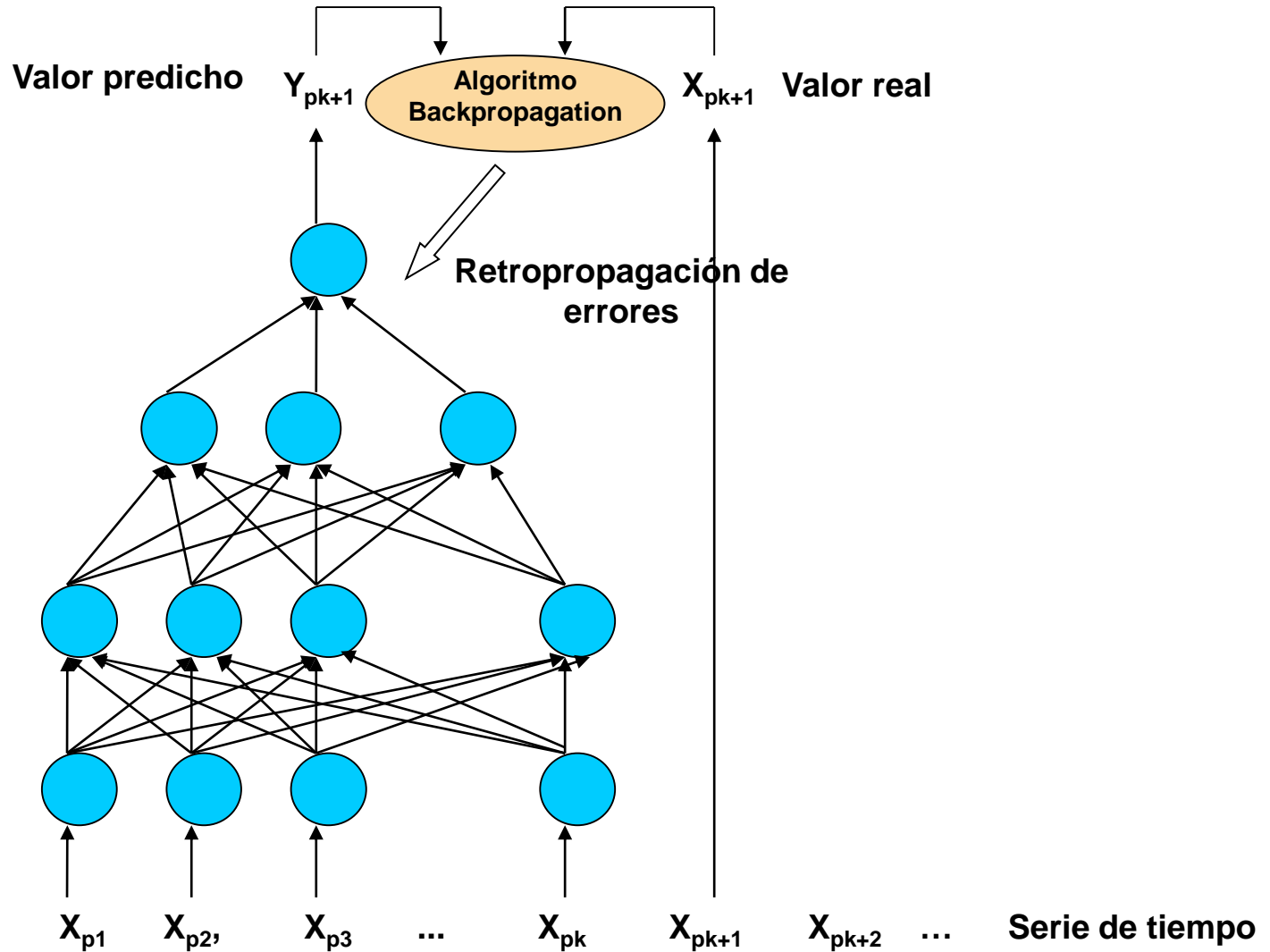
Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para predicción de series de tiempo

Tarea	Modelo de red neuronal	Dominio de aplicación
Predicción de series de tiempo	<ul style="list-style-type: none">▪ Redes feedforward con el algoritmo de retropropagación de errores.▪ Redes neuronales recurrentes.	<ul style="list-style-type: none">❖ Pronóstico económico y financiero.❖ Predicción de riesgo médico.❖ Predicción en series de tiempo caóticas.

Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para predicción de series de tiempo



Fase de Modelado: Redes Neuronales Artificiales

Redes neuronales para tareas de control

Tarea	Modelo de red neuronal	Dominio de aplicación
Aplicaciones de control	<ul style="list-style-type: none">▪ Redes feedforward con el algoritmo de retropropagación de errores.▪ Redes neuronales recurrentes.	❖ Aplicaciones de control y monitoreo de procesos automatizados en la industria: industria automotriz, industria para la producción de componentes electrónicos, etc.

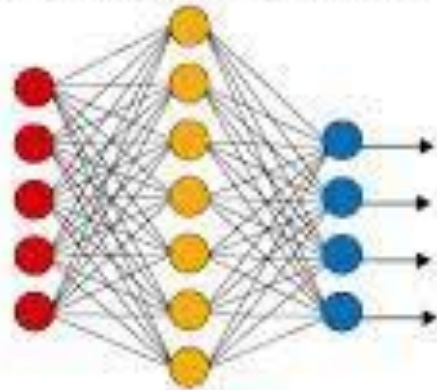
Fase de Modelado: Aprendizaje Profundo (*Deep Learning*)

Aprendizaje profundo (*Deep learning*) (Sánchez-Gutiérrez and González-Pérez, 2020; Sánchez-Gutiérrez and González-Pérez, 2022; González-Pérez and Sánchez-Gutiérrez, 2022)

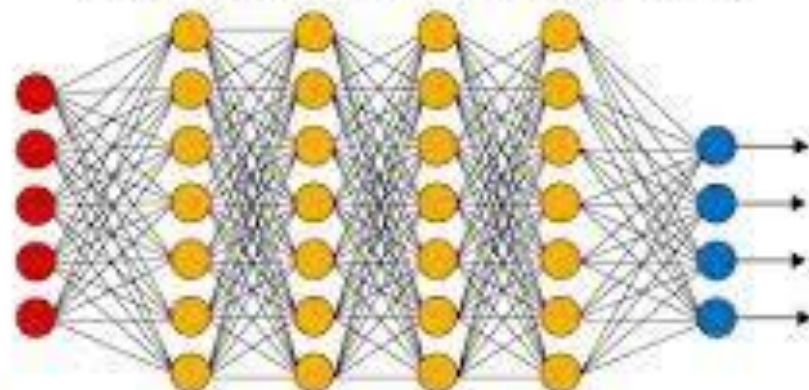
- El aprendizaje profundo (del inglés, *deep learning*) se ha convertido en uno de los paradigmas más aceptados con respecto al aprendizaje automático.
- Esta técnica de *machine learning* se enfoca en el uso de modelos de datos jerárquicos y se basa en la noción de que para aprender sobre representaciones de datos de alto nivel, se necesita una mejor comprensión de la representación en niveles intermedios de la red neuronal.
- Una red neuronal profunda utiliza una jerarquía de representaciones intermedias (capas ocultas) para aprender representaciones de datos de alto nivel. Es decir, tiene varias capas de neuronas ocultas entre las capas de entrada y salida.

Fase de Modelado: Aprendizaje Profundo (*Deep Learning*)

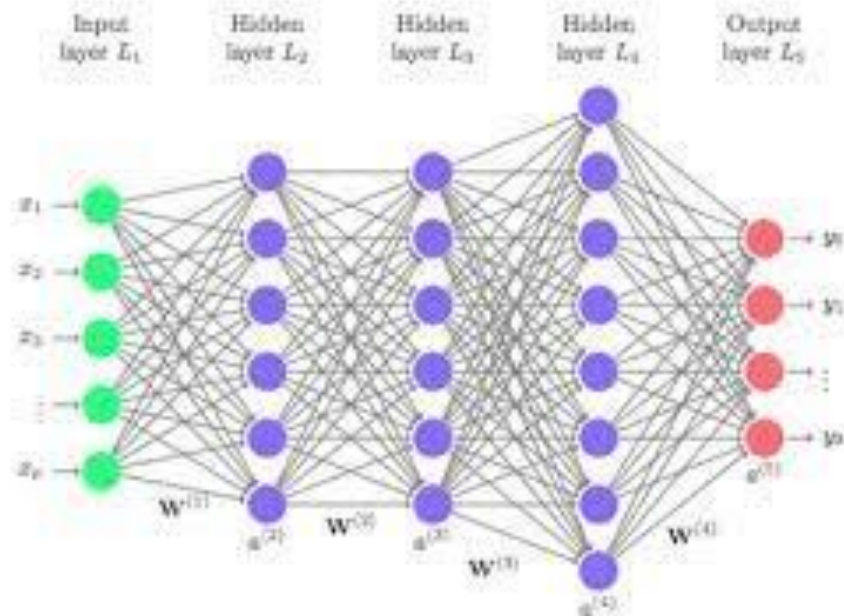
Simple Neural Network



Deep Learning Neural Network



● Input Layer ● Hidden Layer ● Output Layer



Fase de Modelado: Aprendizaje Profundo (*Deep Learning*)

Aprendizaje profundo (*Deep learning*) y Máquinas Restringidas de Boltzmann (Sánchez-Gutiérrez and González-Pérez, 2020; Sánchez-Gutiérrez and González-Pérez, 2022; González-Pérez and Sánchez-Gutiérrez, 2022)

- Las Máquinas Restringidas de Boltzmann (del inglés, RBM) y las Redes de Creencias Profundas (del inglés, *Deep Belief Networks* (DBN)) son dos tipos principales de algoritmos de aprendizaje profundo que se usan comúnmente en una amplia gama de tareas de clasificación y reconocimiento de patrones.
- Las RBM y las DBN se han aplicado con éxito con varios tipos de tareas. Por ejemplo, Sarikaya et al. propusieron un sistema para comprender el lenguaje natural usando DBN (2014), un clasificador de imágenes neurológicas basado en RBM se presentó en Hjelm et al. (2014) y un modelo para pronosticar series de tiempo fue propuesto en Kuremoto et al. (2014). Las RBM y DBN también se utilizan para la síntesis de voz paramétrica (Zen and Senior, 2014) y el modelado de redes estadísticas y probabilísticas (Atwood et al., 2014), entre otros.

Fase de Modelado: Redes Neuronales Artificiales

Aprendizaje profundo (*Deep learning*) y Máquinas Restringidas de Boltzmann (Sánchez-Gutiérrez and González-Pérez, 2020; Sánchez-Gutiérrez and González-Pérez, 2022; González-Pérez and Sánchez-Gutiérrez, 2022)

- En los últimos años, las RBM (Smolensky, 1986) se han utilizado como primera etapa en los sistemas de clasificación, ya sea como extractores de características o como una forma de inicializar redes neuronales (Hinton y Salakhutdinov, 2006; Erhan et al., 2010). ; Albornoz et al., 2014; Sánchez-Gutiérrez et al., 2014).
- Como se muestra en la siguiente figura, una RBM es una red neuronal artificial con dos capas, una capa de unidades visibles que funcionan como un receptor de entrada y la otra con unidades ocultas. Hay conexiones entre unidades de las dos capas, pero no hay conexiones entre unidades en la misma capa.
- Una RBM es una red estocástica generativa, por lo que puede aprender la distribución de probabilidad sobre los datos.

Fase de Modelado: Redes Neuronales Artificiales

Aprendizaje profundo (*Deep learning*) y Máquinas Restringidas de Boltzmann (Sánchez-Gutiérrez and González-Pérez, 2020; Sánchez-Gutiérrez and González-Pérez, 2022; González-Pérez and Sánchez-Gutiérrez, 2022)

M.E. Sánchez-Gutiérrez and P.P. González-Pérez

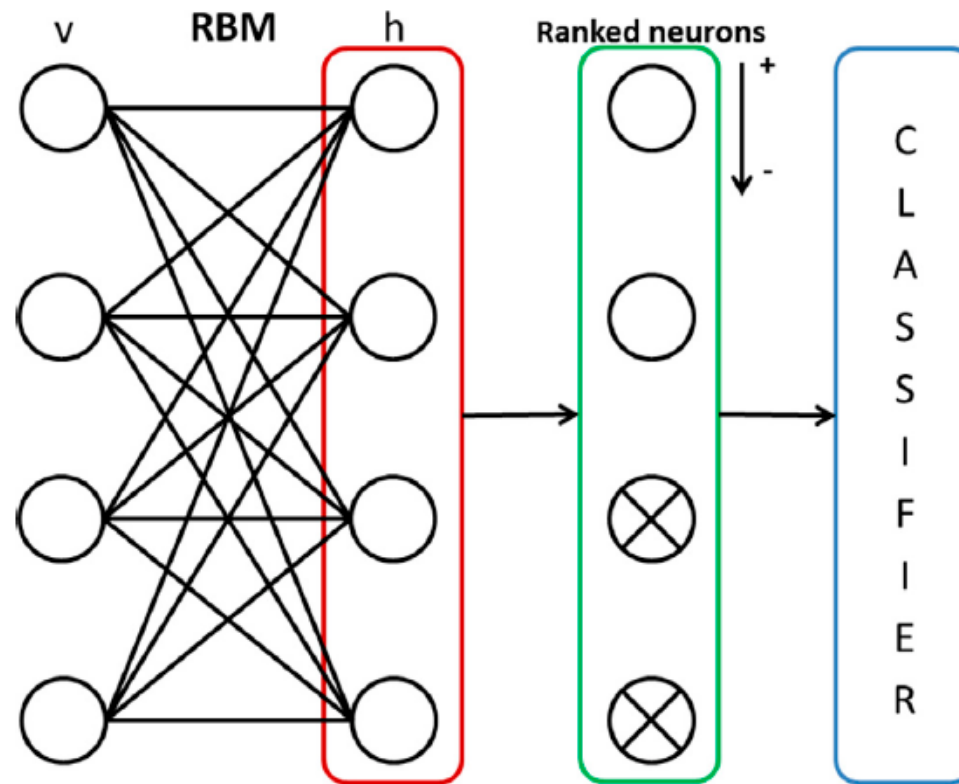


Fig. 1. Scheme for discriminative selection of neurons. It includes an unsupervised trained RBM, a method for ranking the most discriminative neurons and a final classifier.

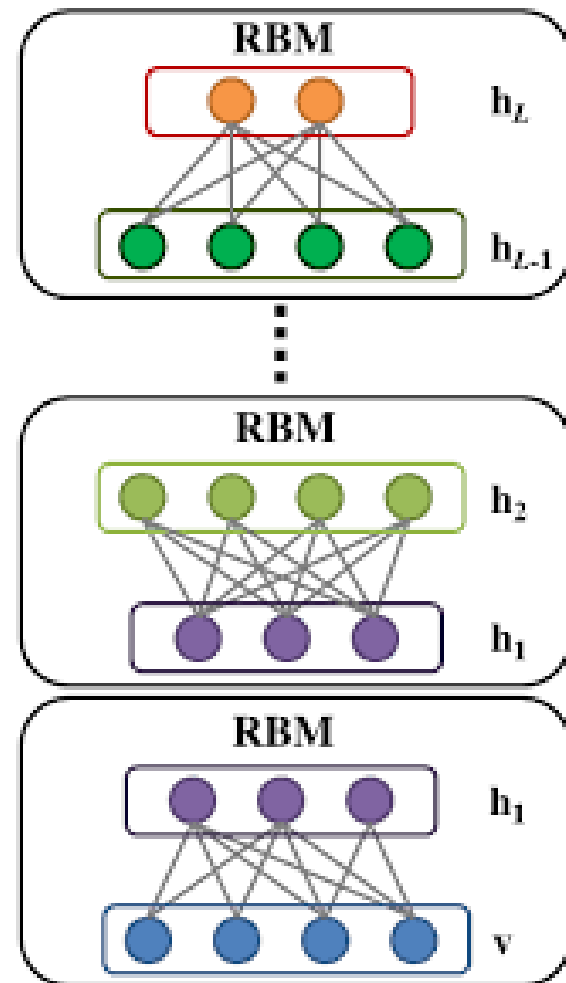
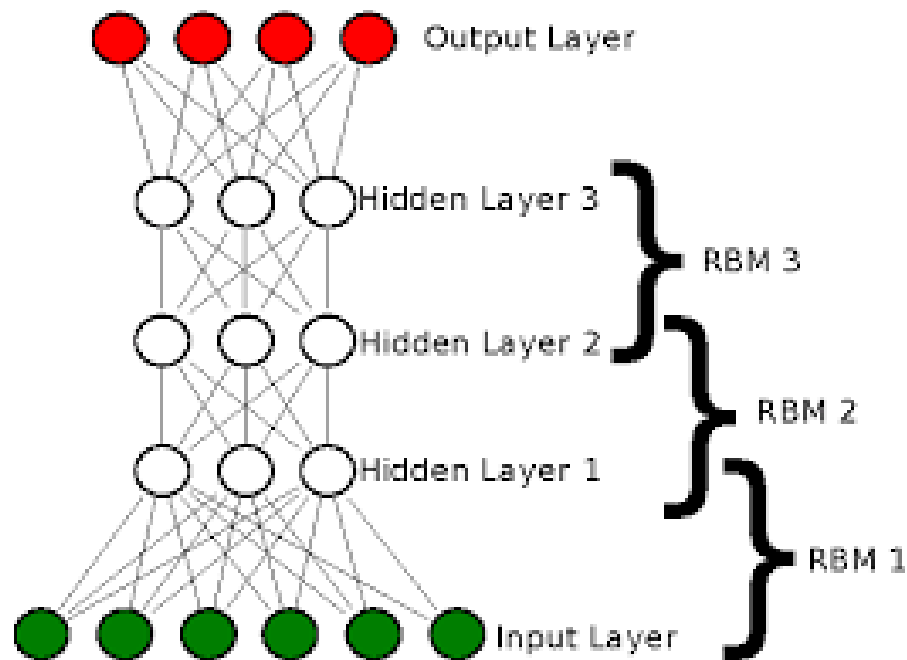
Fase de Modelado: Redes Neuronales Artificiales

Aprendizaje profundo (*Deep learning*) y Redes de Creencias Profundas (Sánchez-Gutiérrez and González-Pérez, 2020; Sánchez-Gutiérrez and González-Pérez, 2022; González-Pérez and Sánchez-Gutiérrez, 2022)

- Una Red de Creencias Profundas (del inglés, *Deep Belief Network* (DBN)) puede verse como una red neuronal profunda que comprende varias RBM. Una DBN aprende la representación interna de los datos de entrada y también puede aprender a reconstruir sus entradas.
- Una DBN es una red neuronal profunda basada en muchas capas de unidades ocultas. En una DBN, cada par de capas conectadas es una RBM. La capa de entrada establece la entrada de los datos, mientras que la capa oculta caracteriza la descripción abstracta de esta entrada.

Fase de Modelado: Redes Neuronales Artificiales

Redes de Creencias Profundas



Fase de Modelado: Redes Neuronales Artificiales

REFERENCIAS

1. Mehrotra, K., Mohan, C.K. y S. Ranka. (1997). Elements of Artificial Neural Networks. MIT Press.
2. Fausett, L. V. (1994). Fundamentals of neural networks. Prentice Hall.
3. Arbib, M. (1998). The handbook of the brain theory and neural networks. MIT Press.
4. Hertz, J., Krogh, A. y Palmer, R.G. (1991). Introduction to the theory of neural computation. Addison-Wesley.
5. Sánchez-Gutiérrez, M.E., González-Pérez, P.P. Discriminative neural network pruning in a multiclass environment: A case study in spoken emotion recognition, Speech Communication, Volume 120, 2020, Pages 20-30, <https://doi.org/10.1016/j.specom.2020.03.006>.
6. Sánchez-Gutiérrez, M.E.; González-Pérez, P.P. Multi-Class Classification of Medical Data Based on Neural Network Pruning and Information-Entropy Measures. *Entropy* **2022**, *24*, 196. <https://doi.org/10.3390/e24020196>