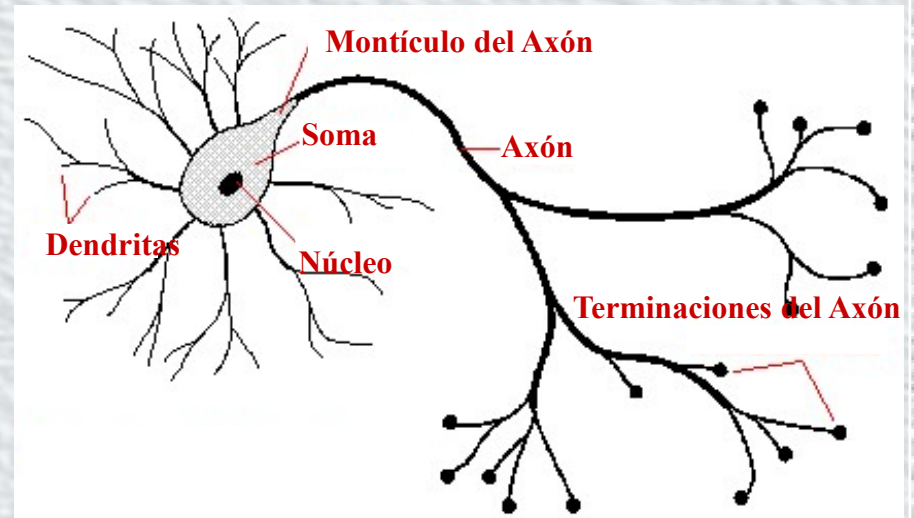


Introducción a las Redes Neuronales

● NEURONA BIOLÓGICA:

- Una neurona es una célula formada por un área engrosada que contiene el **núcleo**, una prolongación larga llamada **axón**, y unas prolongaciones más cortas llamadas **dendritas**.



- El procesamiento de las señales nerviosas incluye dos tipos de fenómenos: **eléctricos y químicos**.

El proceso eléctrico propaga una señal en el interior de la neurona, y el proceso químico transmite la señal desde una neurona a otra.

● RED NEURONAL ARTIFICIAL:

● Concepto:

Las redes neuronales son estructuras de procesamiento paralelo, que están inspiradas en el funcionamiento neurobiológico humano, de esta forma pretenden emular sus capacidades de **aprendizaje, clasificación, reconocimiento**, entre otras. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en múltiples áreas.

● Su unidad básica de procesamiento es el **nodo o neurona**:

Esta consiste fundamentalmente en una función matemática arbitraria, cuya función es recopilar la información que recibe, procesar esta información y comunicarla a otros nodos.

● La red se organiza a través de **capas de neuronas**:

Las cuales son conjuntos de neuronas cuyas entradas provienen de la misma fuente.

HISTORIA

1936: Alan Turing desarrolla teoría de inteligencia y cerebro

1943: W McCulloch y W Pitts desarrollan circuito eléctrico análogo a RN

1949: D Hebb ; concepto de activación matemática y aprendizaje

1957: F Roseblatt: Desarrolla el perceptron (RN Lineal)

1960: B Widroff y M Hoff: Addaline aplicado a reducción de eco

1969 Minsky - Papert: Perceptron no es capaz de resolver problemas simples

1974: P Werbos: Teoría de retropropagación

1980: J Hopfield: Libro de redes neuronales y aplicaciones exitosas

1984: Rumelhart- Hinton -Williams: Algoritmo de retropropagación codificado

Computacionalmente (Inicio de aplicaciones en ingeniería y problemas prácticos)

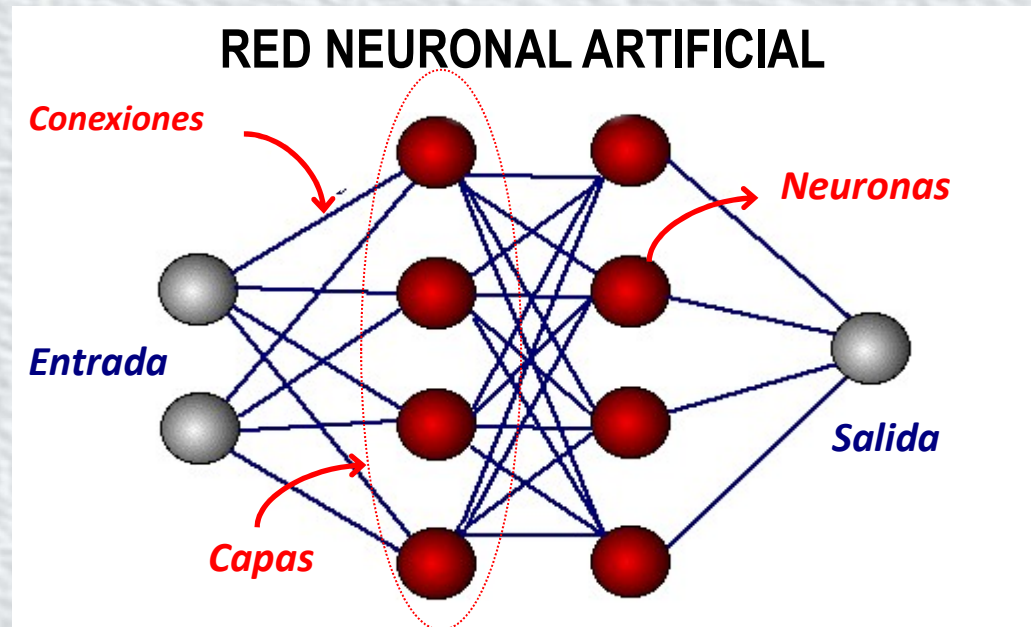
1989: Hornik: RNFF son aproximadores universales.

Estructura de una Red Neuronal

Para diseñar una red neuronal es necesario definir su estructura externa y los parámetros internos que la caracterizan:

ASPECTOS EXTERNOS

- N° de Capas
- N° de Neuronas por Capa
- Tipo de Conexión entre Neuronas



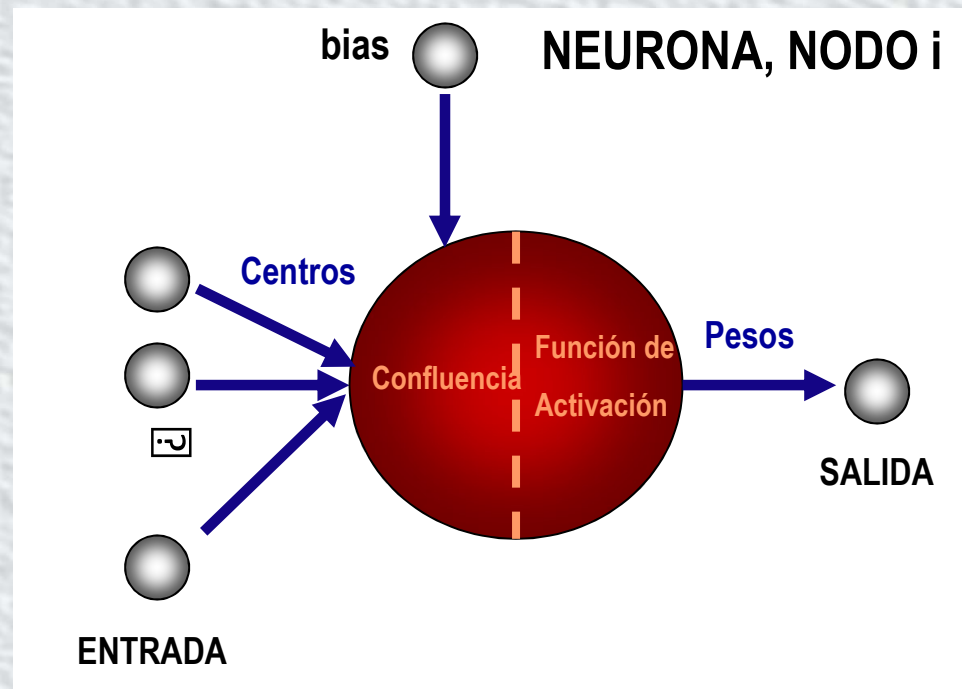
Las señales neuronales pueden seguir una propagación hacia delante (**feedforward**) a través de las capas de la red, hacia atrás (**feedback**) o una combinación de ambas.

ASPECTOS INTERNOS

- **Función de Confluencia:** Entrega una señal globalizada de todas las entradas, para esto pondera cada entrada a través de un factor denominado **centro** de la red, el que valoriza la importancia de la conexión.

- **Función de Activación**

Es la función generadora de la salida de cada neurona, la cual se aplica a la confluencia



- Tipos de Confluencia:

Producto interno

$$v_i = \sum_r^n (x_r \cdot w_{ir} + b)$$

Distancia Euclidiana

$$v_i = \|x_r - w_{ir}\| \cdot b \equiv \sqrt{\sum_r^n (x_r - w_{ir})^2} \cdot b$$

- Tipos de Función de Activación:

Exponenciales:

$$a = \phi(v) = \exp\left(-\frac{v^2}{\beta}\right)$$

Multicuadráticas:

$$\phi(v) = \frac{1}{\sqrt{(v^2 + \beta)}}$$

$$\phi(v) = \sqrt{(v^2 + \beta)}$$

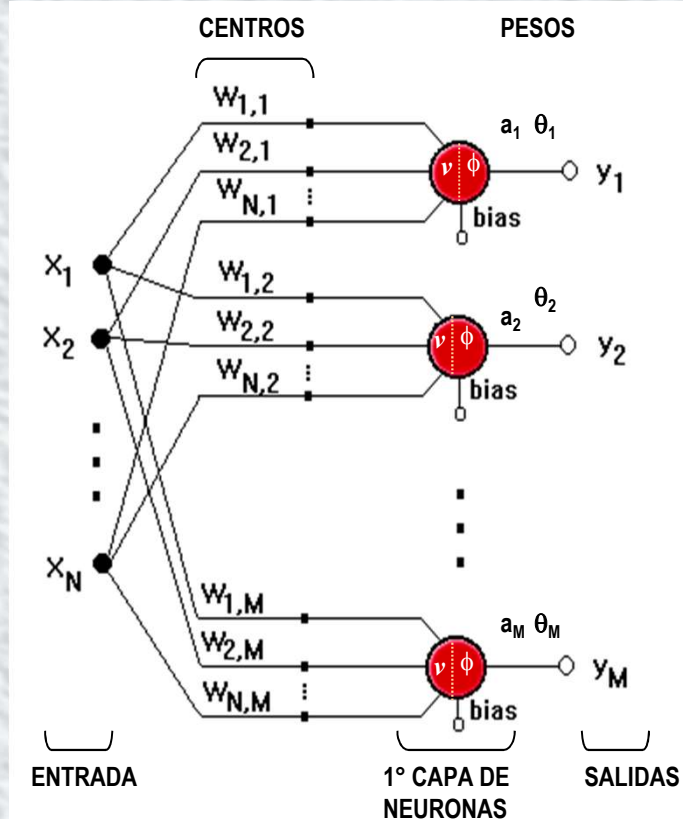
Logarítmicas:

$$\phi(v) = v^2 \log(v)$$

Sigmoidales:

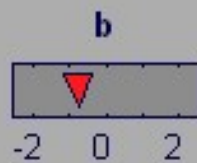
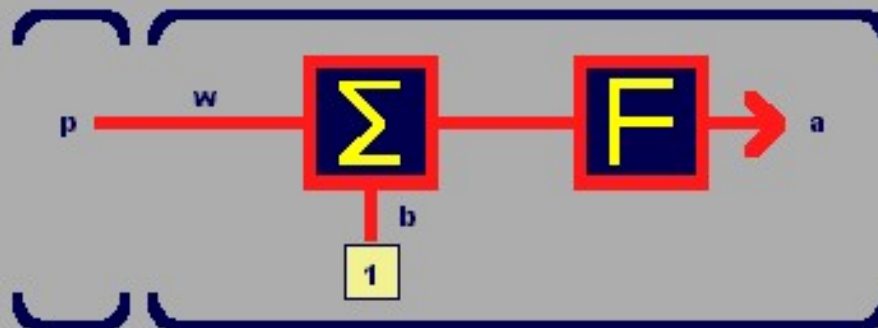
$$\phi(v) = \frac{1}{1 + e^{-x}}$$

$$\phi(v) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

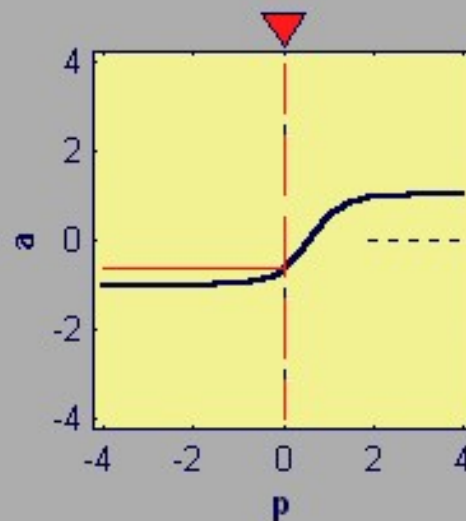




Input Tan Sigmoid Neuron: $a = \text{tansig}(w \cdot p + b)$



F:



Alter the weight, bias and input by dragging the triangular shaped indicators.

Pick the transfer function with the F menu.

Watch the change to the neuron function and its output.

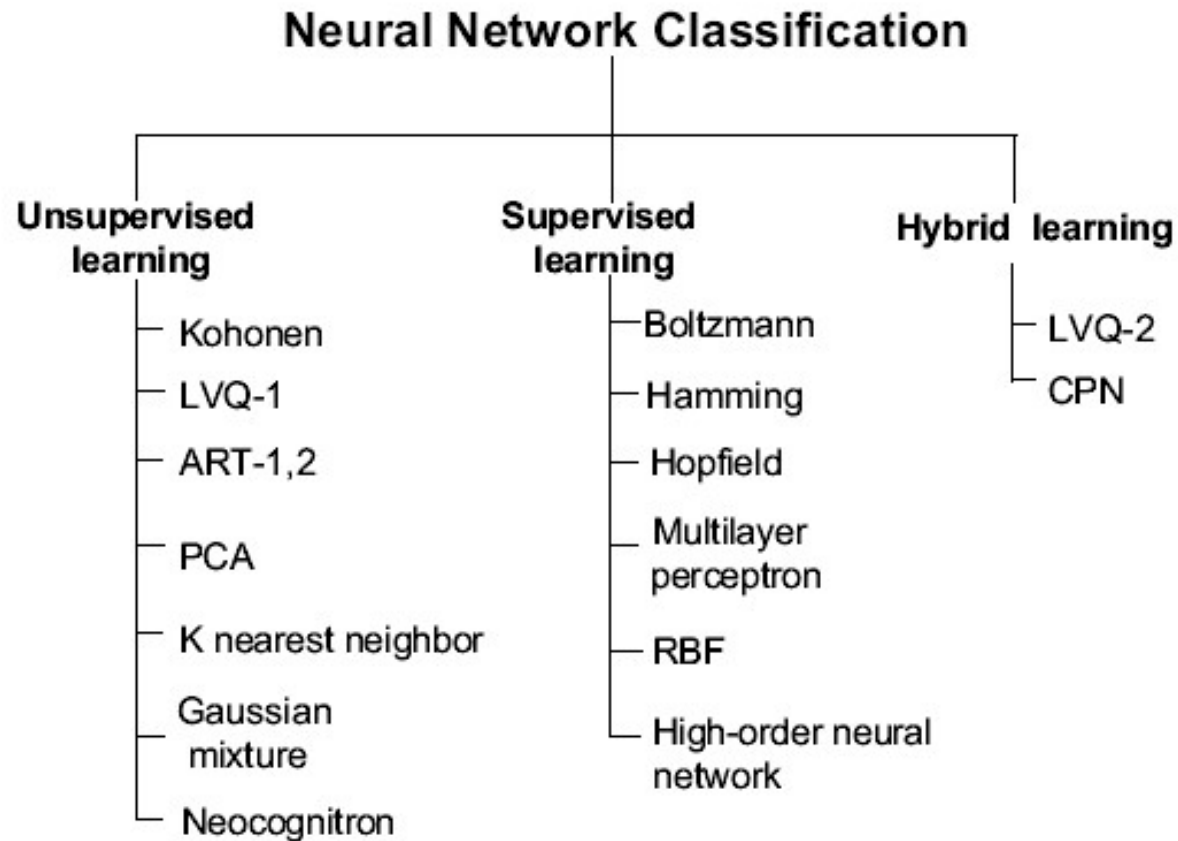
Contents

Close

Chapter 2

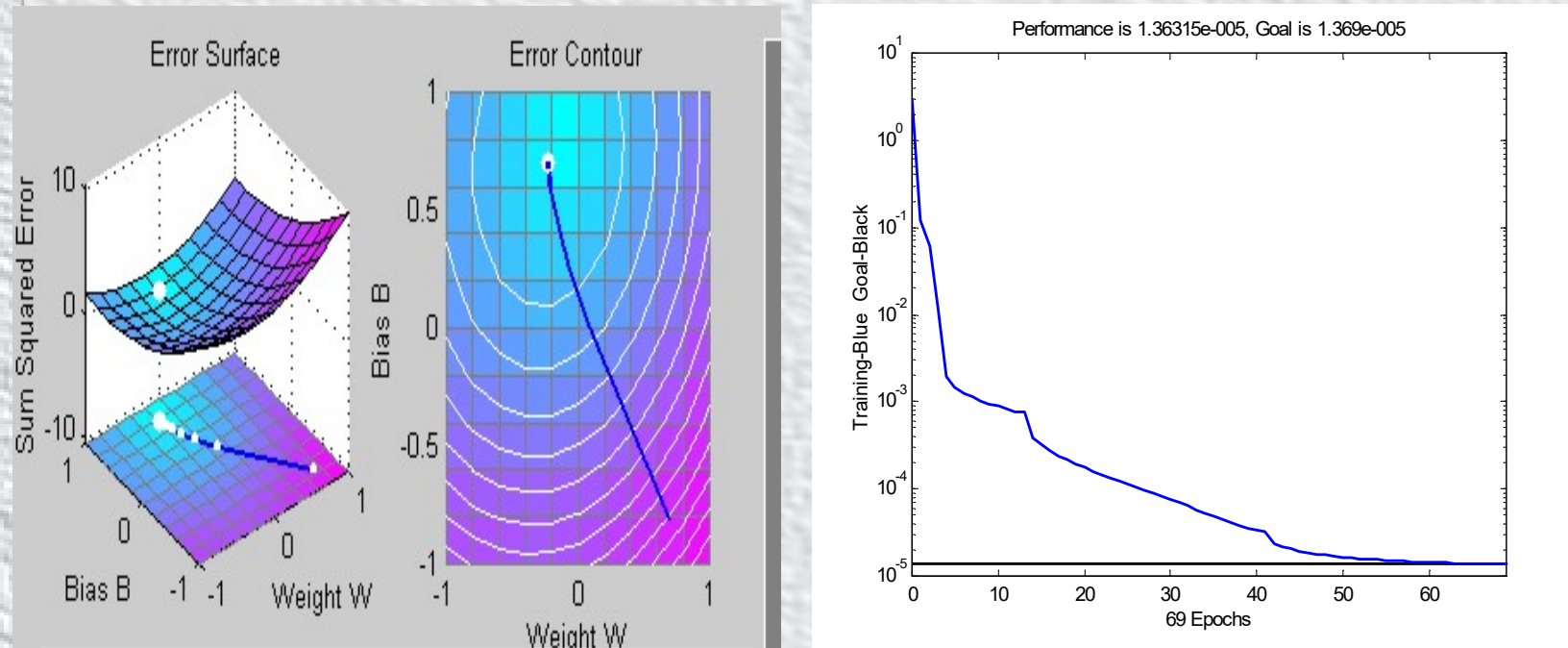
- **Regla de Aprendizaje:**

Proceso por el cual una red modifica sus pesos en respuesta a la información de entrada.



- Entrenamiento: Consiste en encontrar el mejor conjunto de parámetros internos de la red, **centros**, **pesos** y **bias**, que mejor representan un conjunto de datos.

Problema de Optimización

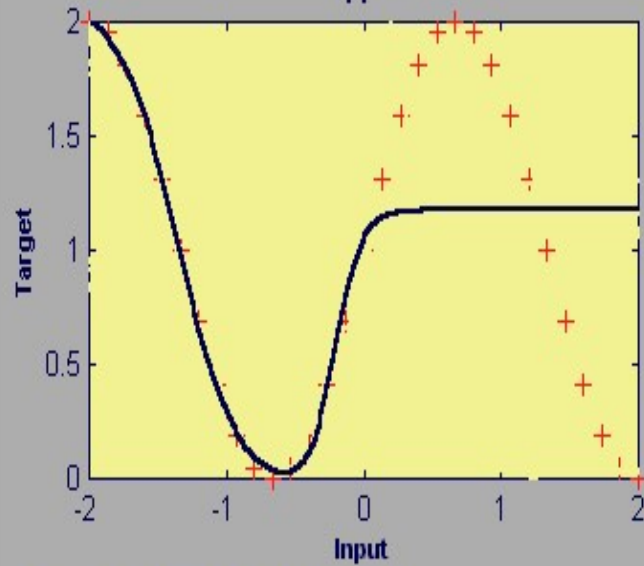


$$\min_{W, b} \sum_{Np} (Y_{exp} - Y_{red})^2$$

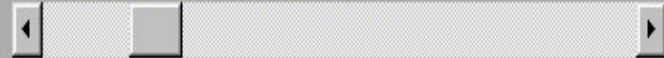
Neural Network DESIGN

General

Function Approximation



Number of Hidden Neurons S1: 2



1 9

Difficulty Index: 3

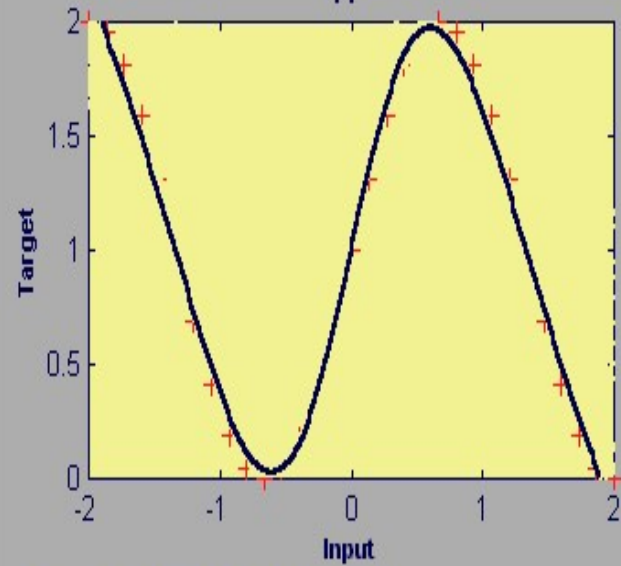


1 9

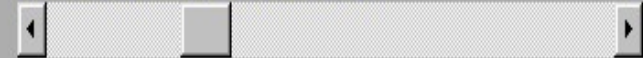
Neural Network DESIGN

General

Function Approximation



Number of Hidden Neurons S1: 3



1 9

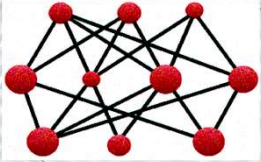
Difficulty Index: 3



1 9

- Conjunto de Entrenamiento y Validación:

Cabe destacar que no existe un método para determinar a priori la topología de una red, es decir la cantidad de nodos o capas requeridas para una determinada aplicación.

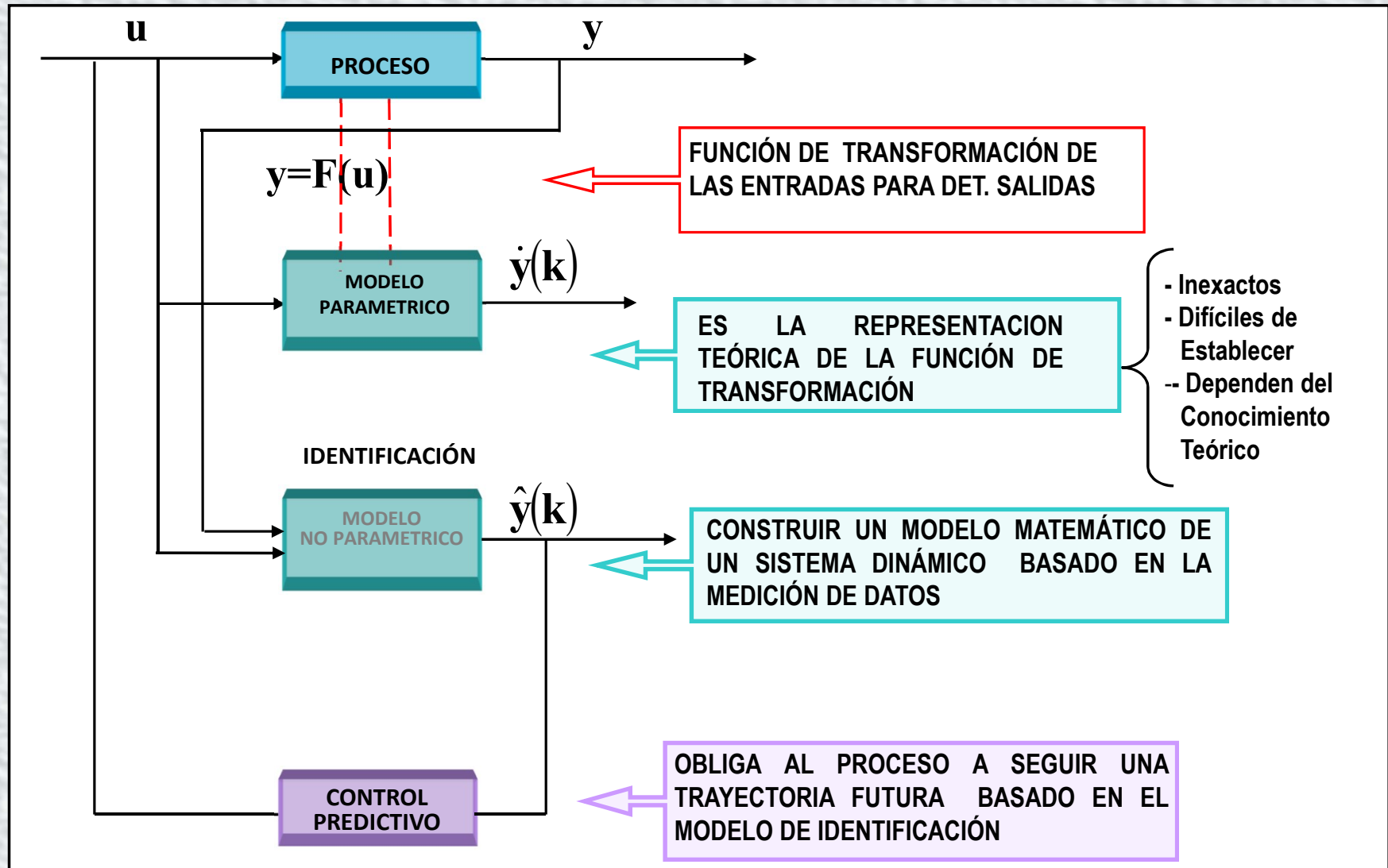


Identificación de Procesos con Redes Neuronales

● INTRODUCCIÓN:

- Desde un punto de vista matemático, **un proceso es una función** aplicada a las entradas para obtener el conjunto de salidas.
- Una forma de representar esta función es **crear un modelo** basado en los principios teóricos de las operaciones que ocurren en el proceso (**MODELO PARAMETRICO**). Pero éste tiene sus inconvenientes, ya que depende del conocimiento teórico, es inexacto y en algunos casos es difícil de establecer.
- Otra forma de obtener una representación del proceso es **a través de la identificación**. Esta forma posee evidentes ventajas sobre la modelación ya que no depende del conocimiento teórico de este (**MODELO NO PARAMETRICO**).

Esquema General:



Concepto de Identificación:

- Consiste en construir un modelo matemático no paramétrico de un sistema dinámico basado en la medición de datos de entradas y salidas.
- Esto se realiza a través de la determinación de la *estructura* y el ajuste de ciertos **parámetros** de la función mediante una optimización.
- En general, la función para la identificación de un proceso depende de datos de entrada y salida del mismo registrados en una secuencia de tiempos, generalmente discreta.

$$\hat{y}(k) = F [y(k - 1), y(k - 2), \dots , y(k - n), u(k - 1), u(k - 2), \dots , u(k - m)]$$

Pasos básicos en una Identificación

El procedimiento para determinar un modelo de un sistema dinámico desde la observación de los datos de entrada y salida involucra tres pasos:

- 1.- Seleccionar un conjunto de datos de entrada y salida del sistema dinámico. (**ACONDICIONAMIENTO DE DATOS**)

Es fundamental obtener un conjunto representativo de datos experimentales de entrada y salida del proceso a ser identificado.

Para esto es necesario examinar la dinámica del sistema, se recomienda analizar los cambios ocurridos en el sistema en respuesta a cambios en la entrada, se pueden realizar diferentes experiencias de perturbación al sistema (cambio escalón, etc.).

En base a esto, se pueden suprimir los datos que no aportan información del sistema y seleccionar una porción útil de los datos originales.

2. Seleccionar y definir la estructura de identificación a aplicar al sistema dinámico (**IDENTIFICACION ESTRUCTURAL**)

En este punto es necesario destacar que en la identificación estructural no existe un método que permita determinar la mejor estructura a priori de un sistema, para esto hay que realizar varias pruebas con modelos diferentes. Aunque siempre hay que considerar el comportamiento del sistema dinámico en estudio, debido a que existen modelos de identificación con mejor ajuste para cierto tipo de sistemas, por ejemplo lineales.

- En el conjunto de modelos candidatos varían:

El orden del modelo, que consiste fundamentalmente en definir las condiciones de entrada y salida del sistema, junto con los correspondientes tiempos de retraso a utilizar (**contexto dinámico**).

El método de predicción, se define el tipo de algoritmo elegido para la estimación de los parámetros (**IDENTIFICACION PARAMETRICA**).

3.- Análisis del Modelo de Identificación seleccionado (VALIDACION):

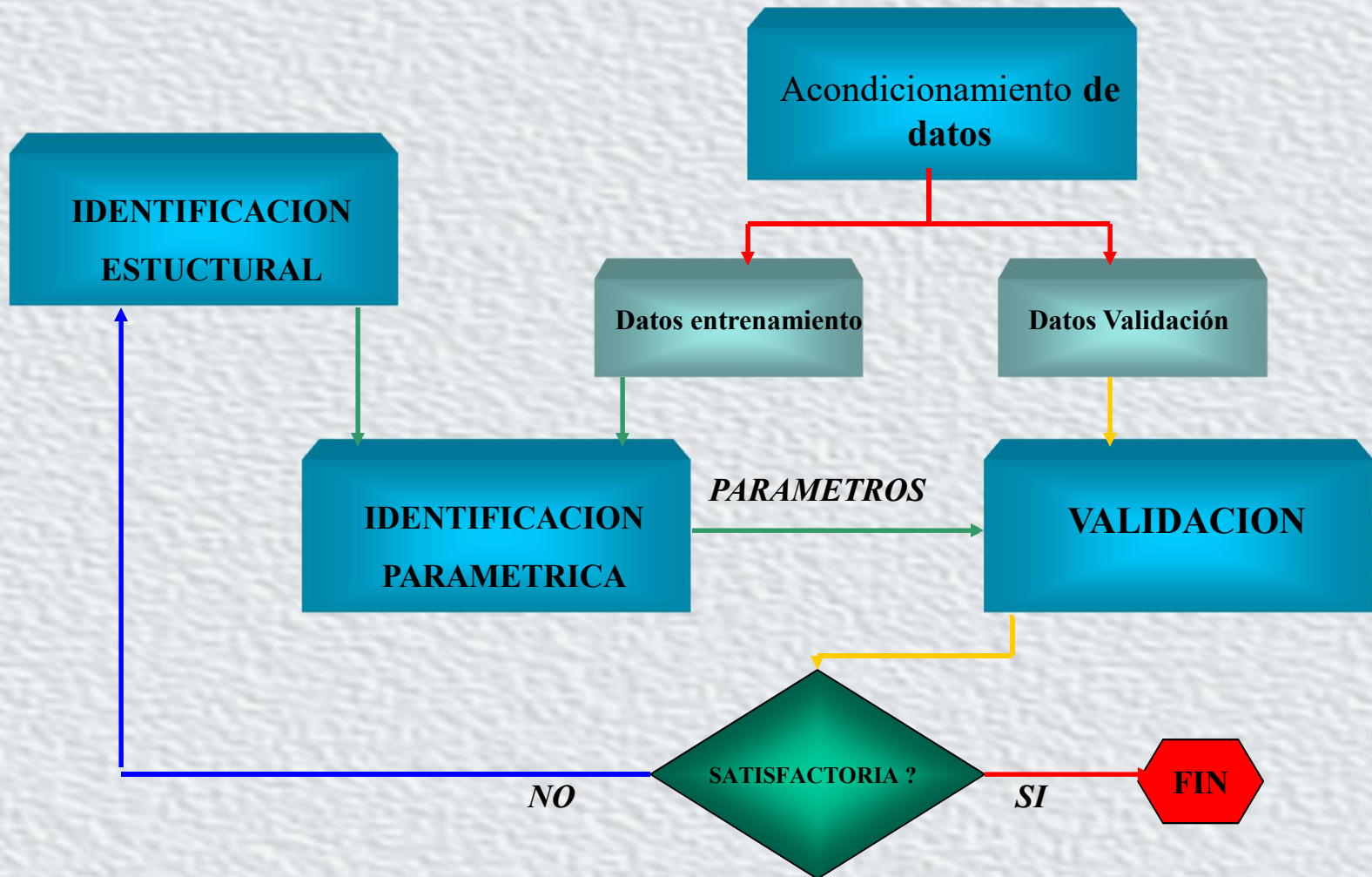
Una buena forma de obtener una evaluación de la calidad del modelo elegido es simular el sistema dinámico con un conjunto de nuevos datos de entrada al sistema pertenecientes a el denominado **Conjunto de Validación**, y posteriormente se compara cuantitativamente la salida predicha por el modelo con la salida del conjunto de validación.

Este conjunto de datos de entrada y salida debe ser representativo del sistema y además sólo se utiliza para la evaluación de los modelos.

Para seleccionar el modelo que mejor representa al sistema dinámico se debe elegir un criterio de selección.

Generalmente se utiliza la minimización del error cuadrático entre las salidas predicha y medida del sistema.

ESQUEMA DE IDENTIFICACION



Formas de Identificación de un Proceso

● IDENTIFICACIÓN FUERA DE LÍNEA:

- La Identificación Fuera de Línea, es el tipo de identificación más usual, ésta opera con una base de datos de entrada y salida del proceso adquirida previamente a la identificación, mediante un simulador o directamente del proceso. La determinación de los parámetros se basa en este conjunto de datos fijos.

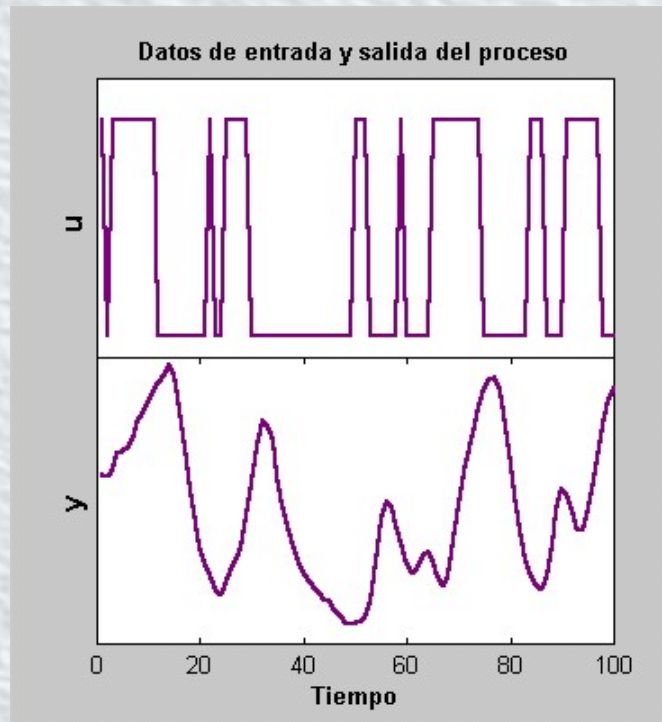
● IDENTIFICACIÓN EN LÍNEA:

- La Identificación en Línea se trabaja en un sistema dinámico, continuamente se están adquiriendo muestras discretas de los datos de entrada y salida del proceso, los cuales serán los que utilizan continuamente para la determinación de los parámetros, con esto se persigue actualizar la función de transformación del proceso.

Modelo Lineal ARX (red una capa lineal)

- Un ejemplo de modelo no paramétrico utilizado ampliamente en identificación es el modelo ARX (**AutoRegressive eXogenous**), cuya función de transformación no es más que una combinación lineal de entradas y salidas pasadas.
- La técnica de identificación de este modelo consiste en encontrar el mejor conjunto de parámetros a_i y b_i para predecir la salida $y(k)$. Este modelo utiliza el algoritmo de mínimos cuadrados para la estimación de los parámetros.
- Este modelo no es muy exacto en sistemas no lineales, lo que implica desarrollar nuevas estrategias de solución para este tipo de procesos.

- Para aplicar el modelo ARX a un proceso se registran datos para construir el conjunto de entrenamiento.



Tiempo	u	y
0	u(0)	y(0)
1	u(1)	y(1)
:	:	:
k-1	u(k-1)	y(k-1)
k	u(k)	y(k)
k+1	u(k+1)	y(k+1)
:	:	:
r	u(k+r)	y(k+r)

- Ecuación del modelo lineal ARX:

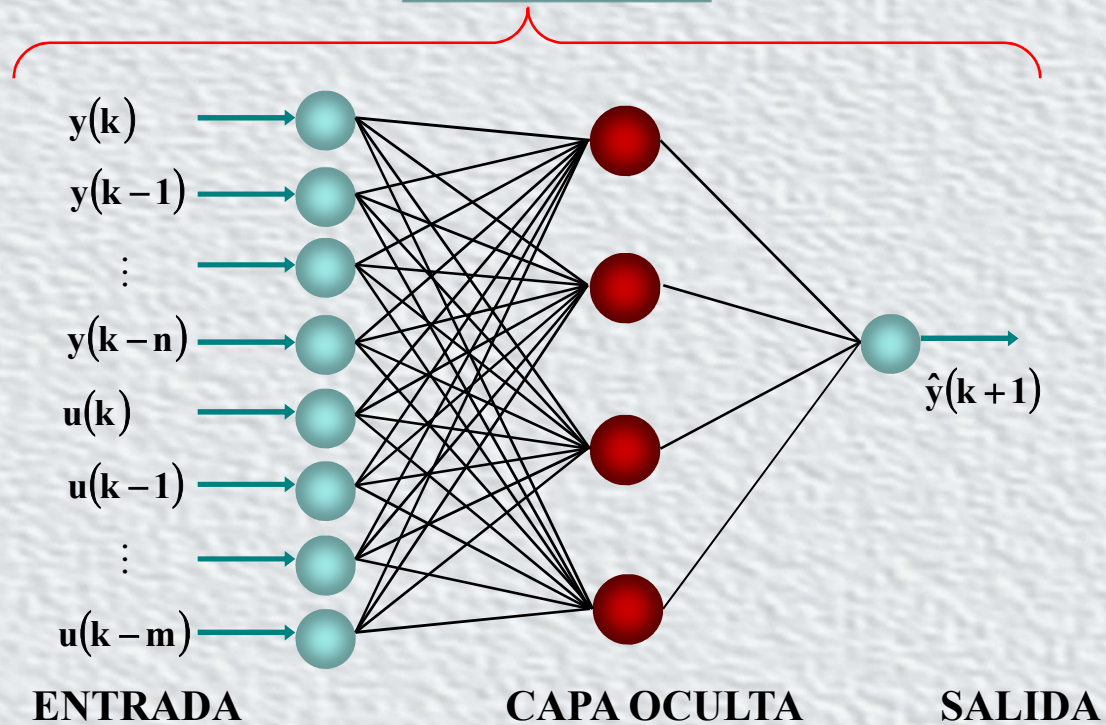
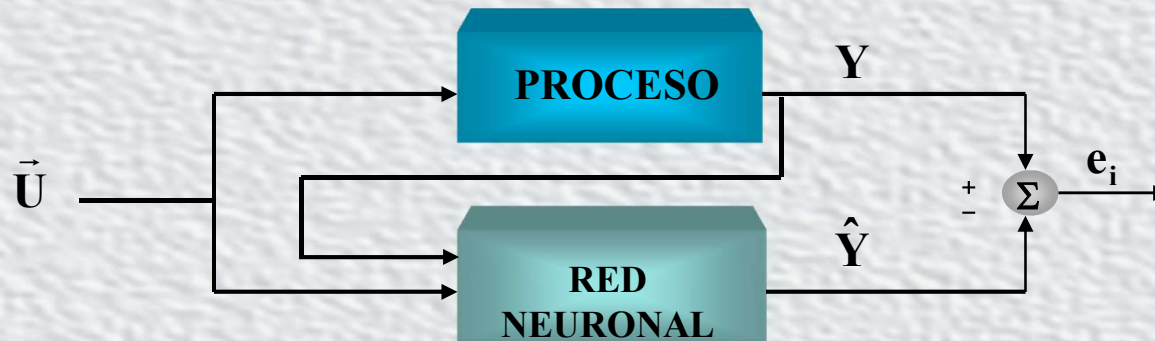
$$\hat{y}(k) = a_0 y(k-1) + a_1 y(k-2) + \dots + a_n y(k-n) +$$

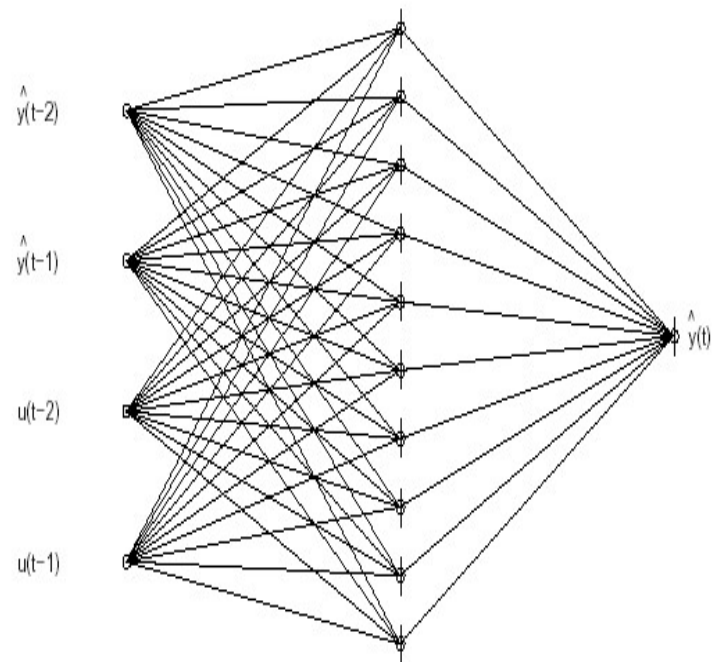
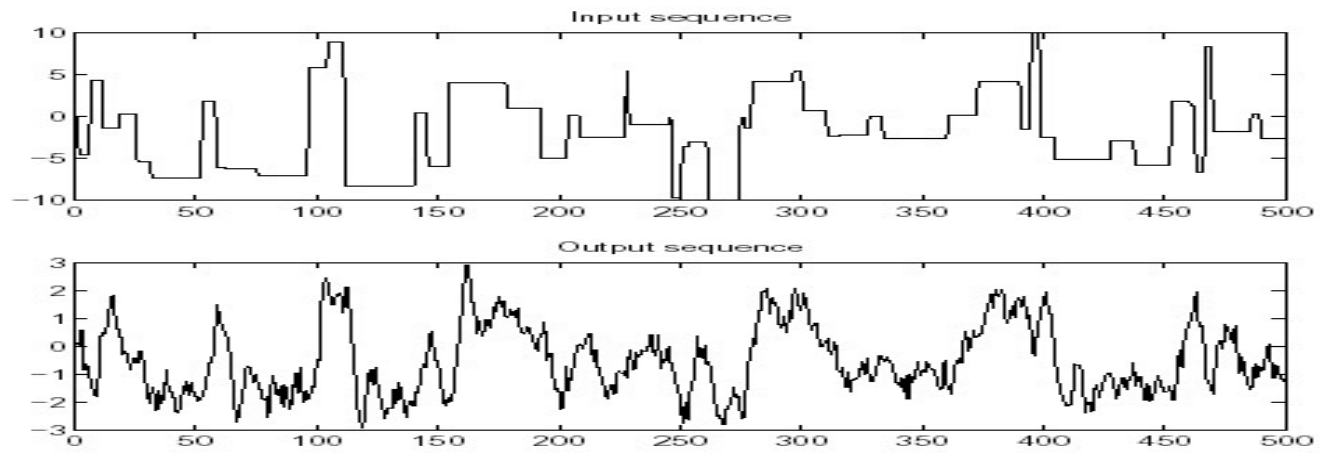
$$b_0 u(k-1) + b_1 u(k-2) + \dots + b_m u(k-m)$$

Identificación de Procesos con Redes Neuronales

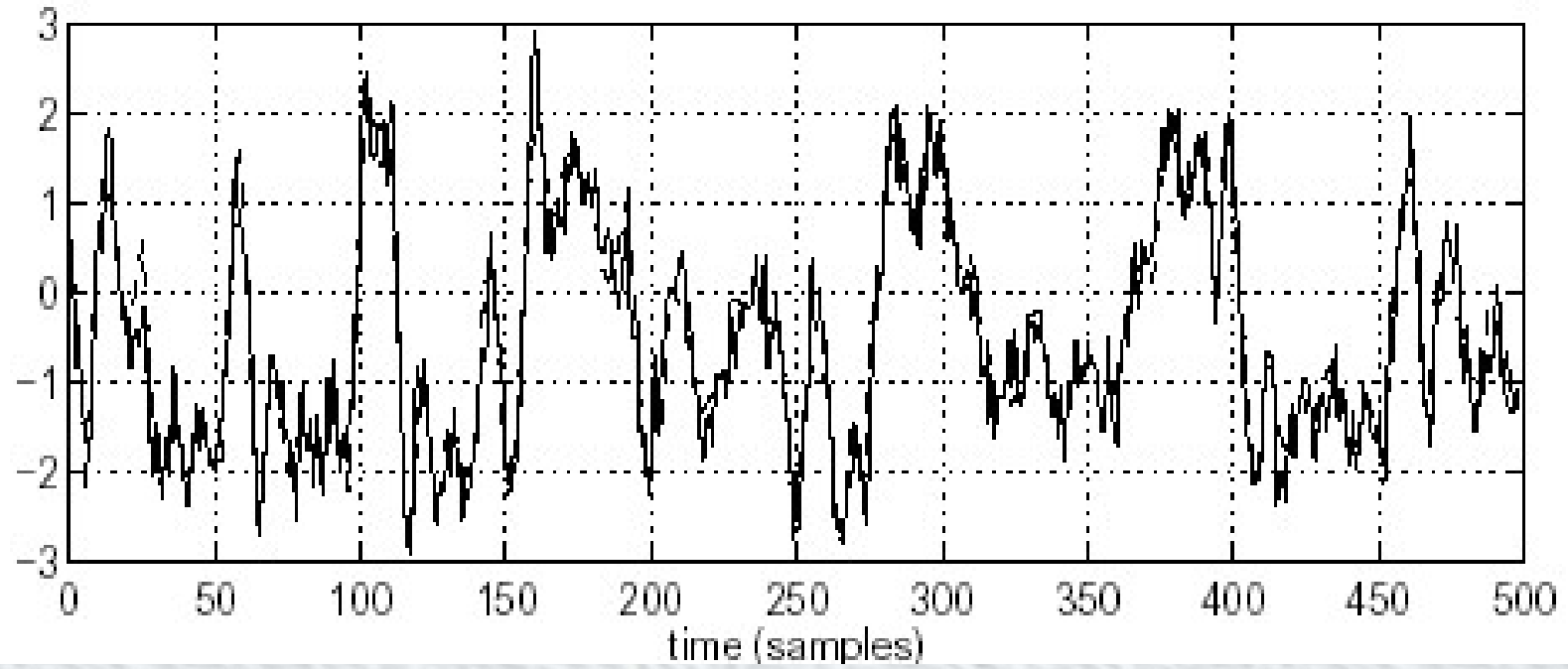
- Para la identificación de un sistema no lineal, se usan satisfactoriamente las redes neuronales, aprovechando su estructura para diseñar modelos matemáticos que representen a cualquier proceso.
- La identificación de procesos con redes neuronales, se puede realizar usando un esquema en paralelo con la planta en malla abierta, o bien trabajando con un archivo de datos históricos.
- Para la identificación de procesos químicos, el tipo de red más utilizada es una **red neuronal feedforward**, (conexiones hacia delante), con una capa oculta y funciones de activación sigmoidales.

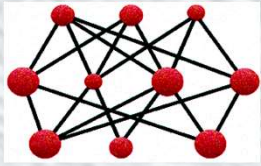
Esquema de Identificación con una red neuronal





Output (solid) and one-step ahead prediction (dashed)





Control de Procesos con Redes Neuronales

INTRODUCCIÓN

Sin duda, la capacidad de las redes neuronales para identificar sistemas complejos, con **no linealidades**, y **dinámica variable**, ha producido un gran interés en incorporarlas a los métodos de control avanzado, principalmente de dos formas :

- Como clasificador de padrones
- Como modelos de procesos no lineales.

● CLASIFICADOR DE PATRONES:

- La red es capaz de reconocer un cierto evento en el sistema:
 - Una perturbación
 - Cambio de propiedades
 - Cambio en el objetivo de control

y generar una acción correctiva, cambiando los parámetros o modificando la respuesta de un controlador convencional.

● MODELOS DE PROCESOS :

- Consiste en usar las redes neuronales como modelo de procesos, e integrarlas a algún esquema de control no lineal tipo **MBC** (Model Based Control).

Los métodos **MBC** con redes neuronales son los más ampliamente estudiados (Morris,1994), debido a que es un área bien establecida, con un abundante desarrollo teórico para sistemas lineales, aparte de tener una amplia aceptación en el ámbito industrial.

- Los esquemas de control **MBC** pueden ser agrupados en dos grandes grupos:

- **Esquemas de control Directos:**

Donde la red neuronal corresponde al controlador y la acción de control es computada en forma directa.

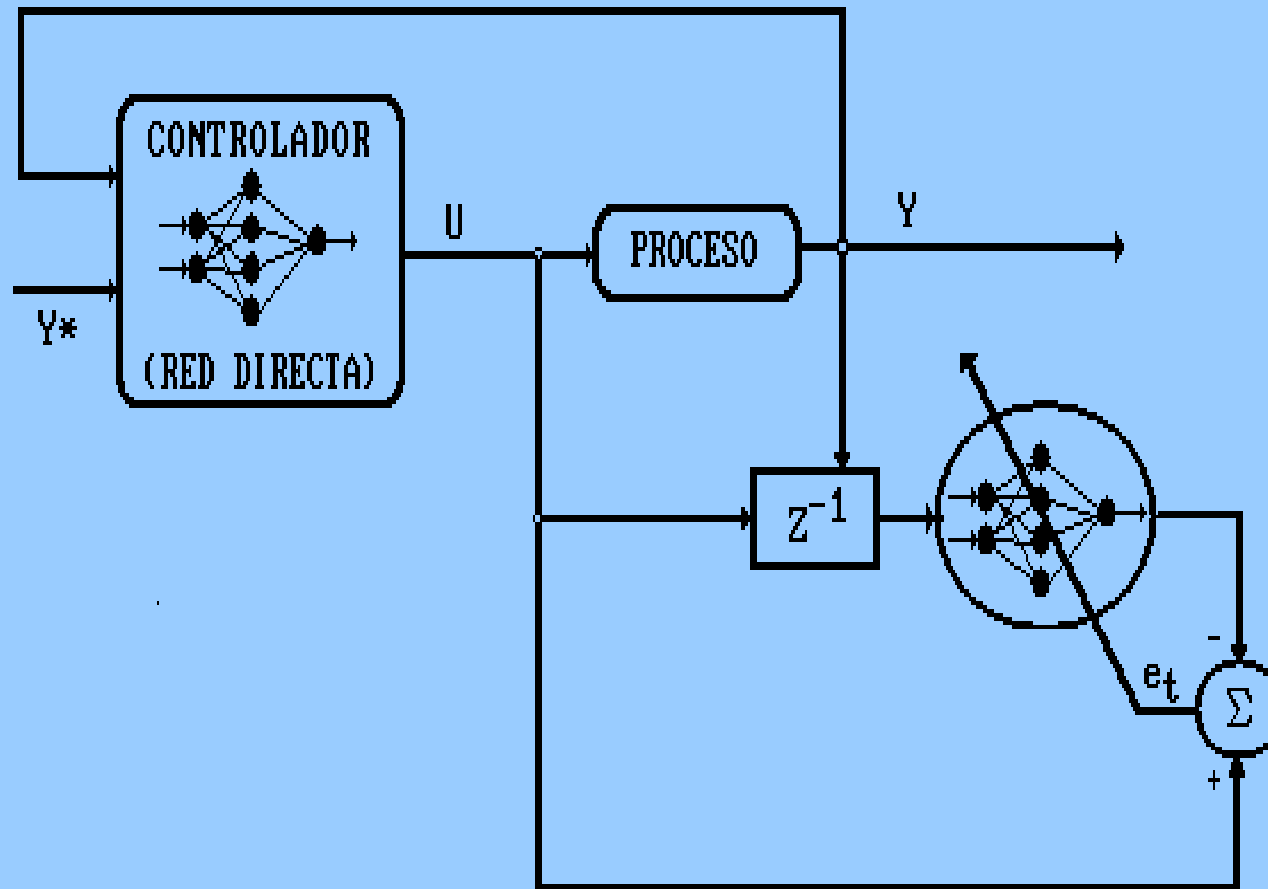
- **Esquemas de Control Indirectos:**

Donde es preciso identificar primero la dinámica directa de la planta con la red neuronal, y después calcular la acción de control mediante algún procedimiento de diseño, que incluya al modelo identificado previamente.

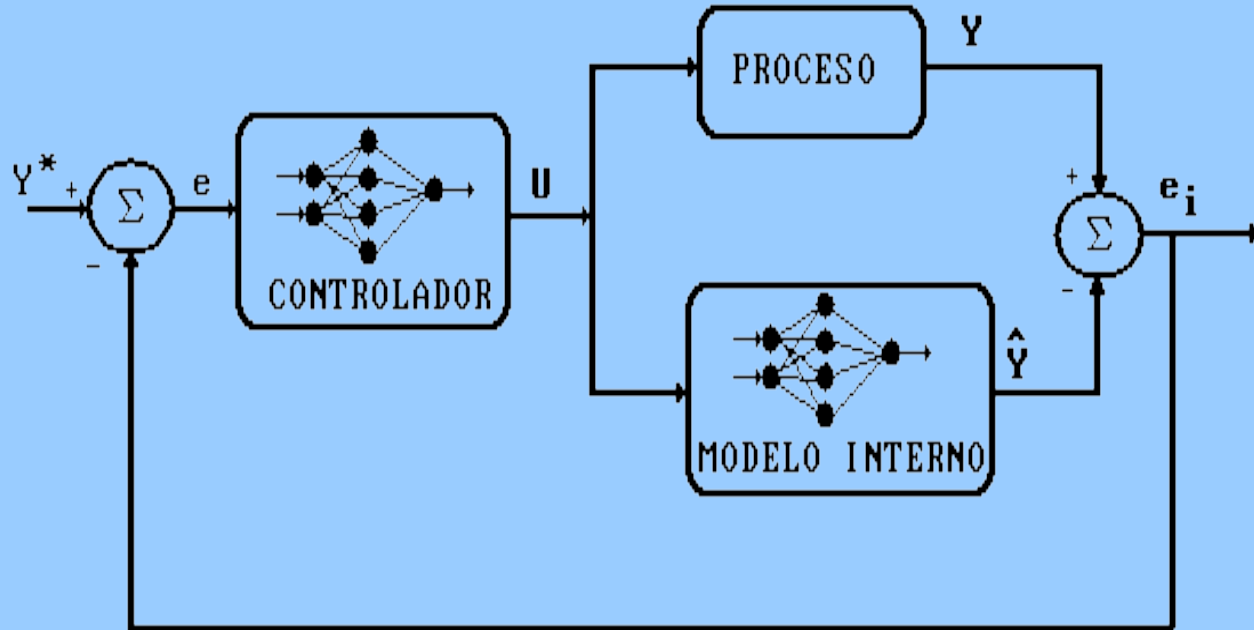


CONTROLADORES DIRECTOS:

CONTROL INPUT-MATCHING



CONTROLADOR NEURONAL IMC (Internal Model Control)



● CONTROLADORES INDIRECTOS:

- Como fue establecido anteriormente, los controladores indirectos requieren de un modelo del proceso y de un algoritmo de diseño, que tome sus decisiones en base a las predicciones del modelo. De esta forma la red neuronal es usada como un modelo no lineal del proceso, que puede ser entrenada fuera de línea, usando un archivo de entrenamiento, o entrenada *on-line*, mediante un algoritmo recursivo de adaptación.

CONTROLADORES PREDICTIVOS

- Los controladores predictivos o **MPC** (*Model Predictive Control*) son actualmente los controladores avanzados con mayor éxito en el campo industrial. En ellos, las acciones de control son determinadas, tal de minimizar una función objetivo que es calculada usando las predicciones de un modelo de proceso, a lo largo de un horizonte de predicción.
- Puesto como un problema de optimización, esta formulación permite la inclusión explícita de las restricciones del proceso. En la práctica, este controlador es capaz de regular procesos inestables y de dinámicas complejas (Hernández y Arkum,1992).

CONTROLADOR MPC NEURONAL (NMPC)

- La formulación de NMPC esta dada por

$$\min_{\Delta \mathbf{u}(\text{FUTURO})} \sum_{i=1}^N \Gamma_i \cdot (\vec{\mathbf{y}}_s - \hat{\mathbf{y}}(\mathbf{k} + \mathbf{i}))^2 + \sum_{j=1}^M \lambda_j \Delta \mathbf{u}(\text{FUTURO})$$

Sujeto a:

i) Movimiento de v. Manipuladas $\Delta \mathbf{u}_{\min} < \Delta \mathbf{u}(\mathbf{k} + \mathbf{i}) < \Delta \mathbf{u}_{\max}$

ii) Límite de v. Manipuladas $\mathbf{u}_{\min} \leq \mathbf{u}(\mathbf{k} + \mathbf{i}) \leq \mathbf{u}_{\max}$

iii) Movimiento de Predicción $\Delta \hat{\mathbf{y}}_{\min} \leq \Delta \hat{\mathbf{y}}(\mathbf{k} + \mathbf{i}) \leq \Delta \hat{\mathbf{y}}_{\max}$

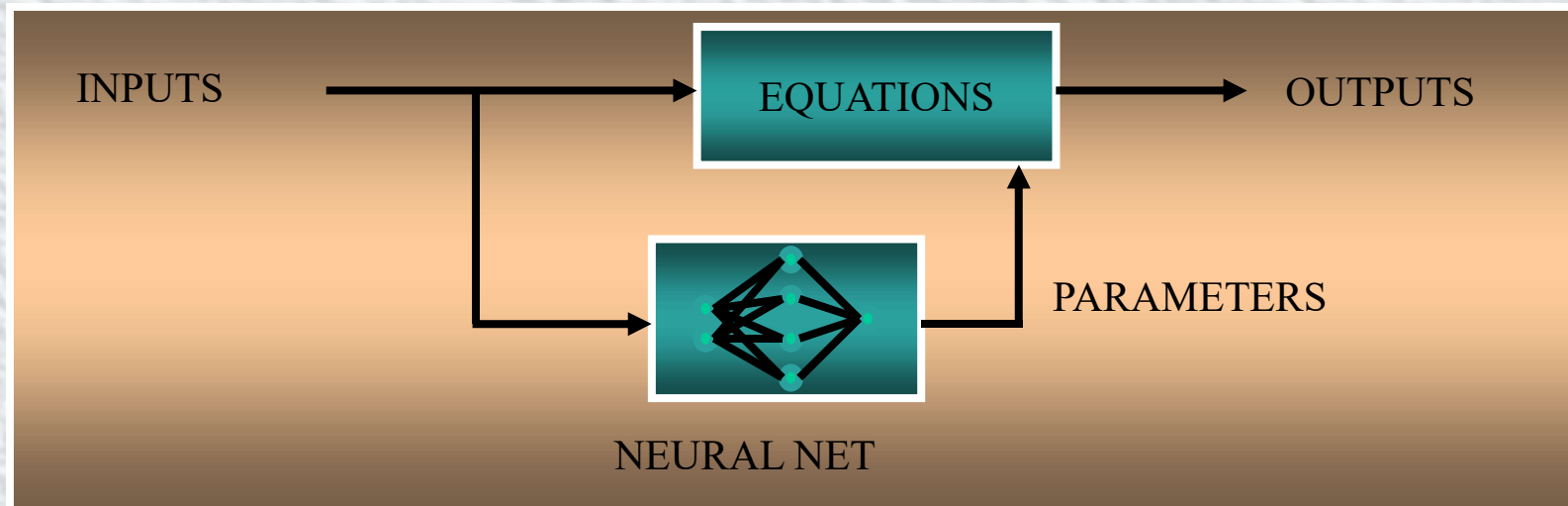
iv) Límite de Predicción $\hat{\mathbf{y}}_{\min} \leq \hat{\mathbf{y}}(\mathbf{k} + \mathbf{i}) \leq \hat{\mathbf{y}}_{\max}$

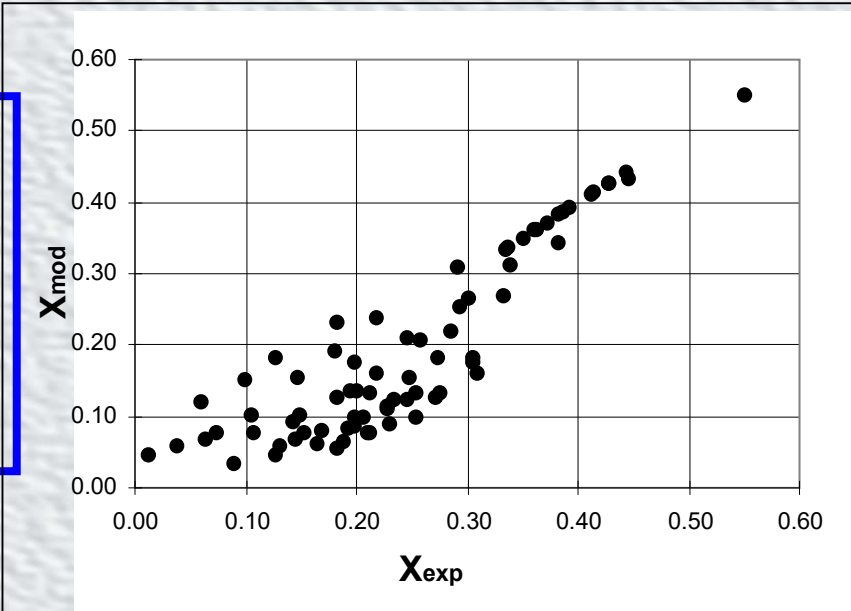
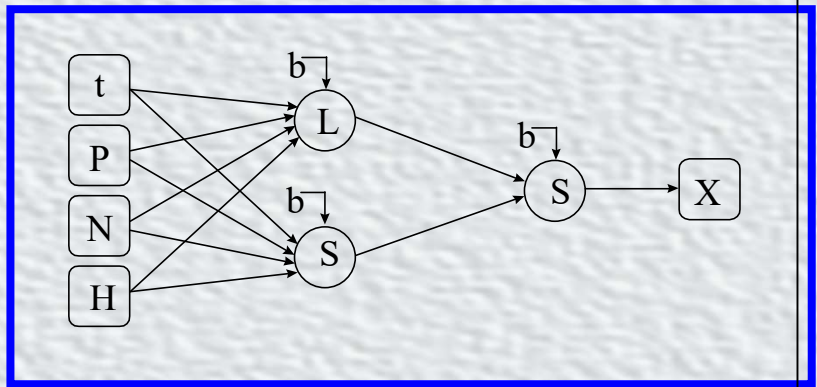
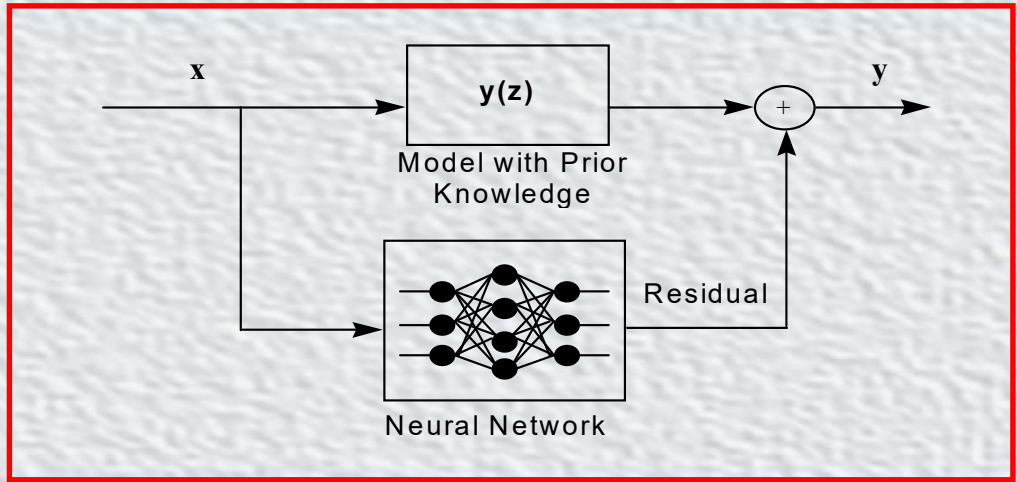
- En esta esta formulación, las redes neuronales son usadas como modelos de predicción para ser usadas en control de procesos no lineales. La red puede ser diseñada para predicción de un paso o para predicción múltiple.
- La selección de un conjunto particular de parámetros de la formulación **MPC**, genera una serie de sub-algoritmos, tales como, *Multi Step Predictive Control* , Horizonte Extendido (Ydstie et al,1985) o **IMC** de tipo indirecto.
- En todas las implementaciones es aplicado el principio de "**horizonte móvil**", donde sólo el primer movimiento de la secuencia de acciones de control es aplicado al proceso, luego el problema es nuevamente resuelto en el próximo instante de muestreo.

Modelación Híbrida o de Caja Gris

El modelo híbrido es una combinación de ecuaciones fenomenológicas y redes neuronales

- Permite estimar parámetros inciertos y desviaciones permanentes
- Modelo mas consistente, fácil de resolver y con sentido físico





SOFTWARES

- Neural Networks Toolbox de Matlab

<http://www.mathworks.nl/access/helpdesk/help/toolbox/nnet/nnet.shtml>

Libro electrónico de Matlab “neural networks design ” nnd

- Neuro solutions

<http://www.neurosolutions.com/>

<http://www.neurosolutions.com/products/ns/nnandnsvideo.html>

