

Tema 6: Identificación de sistemas

MODELOS COMPUTACIONALES Y SIMULACIÓN DE SISTEMAS
Curso 2025-2026



UNIVERSITAT D'ALACANT
UNIVERSIDAD DE ALICANTE
Escola Politècnica Superior
Escuela Politécnica Superior



Temario

B1 - Sistemas Dinámicos

T1: Fundamentos del modelado de sistemas dinámicos

T2: Estabilidad, controlabilidad y observabilidad

T3: Paradigmas de simulación

B2 - Sistemas Complejos

T4: Lenguajes formales para modelos conceptuales

T5: Redes complejas y modelado estructural

B3 - Modelado con IA

T6: Identificación de sistemas

T7: Inteligencia artificial aplicada al modelado de sistemas

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

Índice

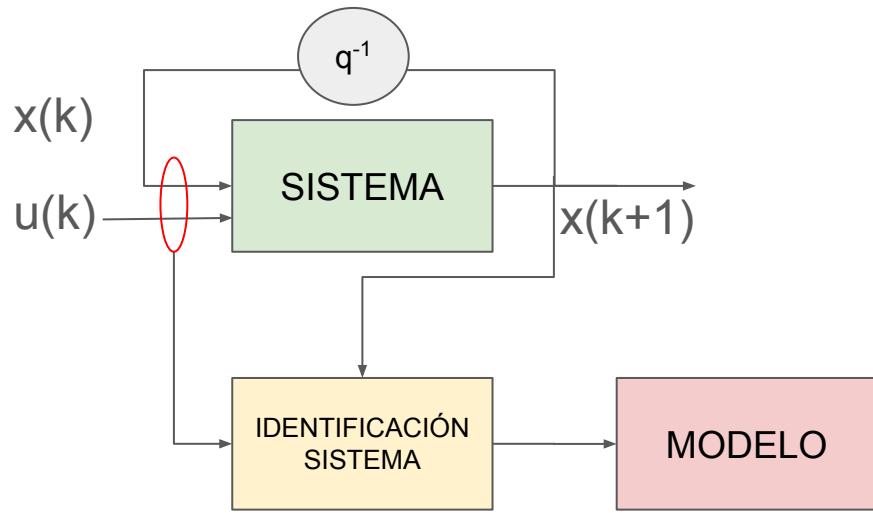
- 1. Introducción**
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

1. Introducción

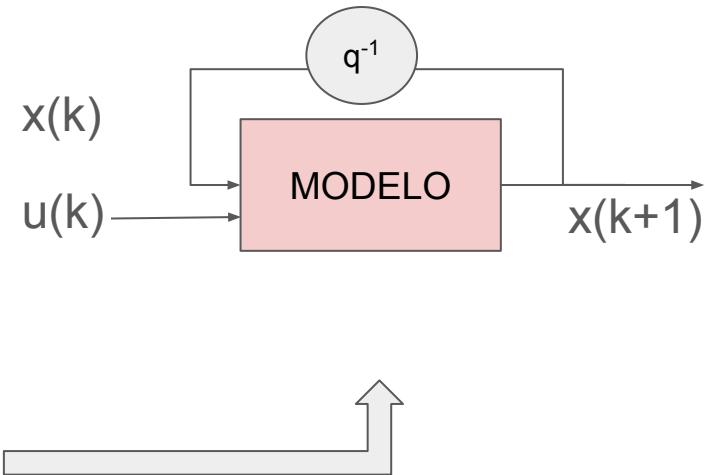
- La **identificación de sistemas** es la rama del modelado y la simulación que se ocupa de construir modelos matemáticos de sistemas dinámicos **a partir de datos experimentales**.
- A diferencia del modelado puramente **físico** o de **primeros principios**, en identificación partimos de mediciones de entradas y salidas, y tratamos de ajustar un modelo que reproduzca, con cierta precisión, el comportamiento observado.

1. Introducción

Entrenamiento



Inferencia



Índice

1. Introducción
2. **Fundamentos de la identificación de sistemas a partir de datos**
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

2. Fundamentos de la identificación

Consideremos un sistema dinámico (desconocido) que recibe una señal de entrada $u(t)$ y produce una salida $x(t)$. Según el contexto, trabajaremos en tiempo continuo o discreto. Para identificación por ordenador suele ser más práctico el tiempo discreto, donde tomamos muestras:

$$u(k) = u(kT_s), \quad x(k) = x(kT_s),$$

siendo T_s el periodo de muestreo y $k \in \mathbb{Z}$ el índice de tiempo discreto.

2. Fundamentos de la identificación

- **Objetivo básico**

El objetivo de la identificación es encontrar un modelo M en alguna clase de modelos (lineales, no lineales, estocásticos, etc.) tal que, dado $u(k)$, el modelo genere una salida $\hat{x}(k)$ que aproxime bien a la salida real $x(k)$:

$$\hat{x}(k) \approx x(k), \quad \forall k \text{ en el conjunto de interés.}$$

2. Fundamentos de la identificación

- **Objetivo básico**

Típicamente, un modelo se describe mediante una estructura con parámetros desconocidos θ :

$$M : \quad \hat{x}(k) = f(u(k), u(k-1), \dots, x(k-1), \dots; \theta),$$

y el problema de identificación se convierte en estimar θ a partir de los datos medidos:

$$\{u(k), x(k)\}_{k=1}^N.$$

2. Fundamentos de la identificación

Según el grado de conocimiento físico previo del sistema, se habla de:

- **Modelo de caja blanca (white-box)**: basado casi exclusivamente en leyes físicas conocidas; pocos parámetros desconocidos.
- **Modelo de caja gris (grey-box)**: se conoce parcialmente la estructura física, pero se dejan algunos términos o parámetros a estimar a partir de datos.
- **Modelo de caja negra (black-box)**: se desconoce la estructura interna del sistema; se usa una clase de modelos muy flexible (ej.\ series de Volterra, redes neuronales, etc.) y se determinan los parámetros sólo a partir de datos.

2. Fundamentos de la identificación

El procedimiento típico de identificación comprende las siguientes fases:

1. **Diseño del experimento y adquisición de datos.** Elegir cómo excitar al sistema (tipo de entrada), duración del experimento, muestreo, etc.
2. **Selección de la estructura del modelo.** Elegir una familia de modelos: ARX, ARMAX, modelos de espacio de estados, redes neuronales, etc.
3. **Estimación de parámetros.** Ajustar los parámetros theta minimizando algún criterio de error, típicamente basado en mínimos cuadrados o máxima verosimilitud.
4. **Validación del modelo.** Verificar si el modelo reproduce adecuadamente datos diferentes a los de entrenamiento, analizando residuales y criterios de calidad.
5. **Refinamiento.** Si el modelo no es satisfactorio, se revisan la estructura, los órdenes, la calidad de los datos o el tipo de excitación, y se repite el proceso.

2. Fundamentos de la identificación

Supongamos que el sistema real es discretizado y descrito por:

$$x(k+1) = a x(k) + b u(k) + e(k),$$

donde $e(k)$ es un ruido de medida (no conocido), a y b son parámetros constantes desconocidos. Dado un conjunto de datos $\{u(k), x(k)\}$, el problema de identificación consiste en estimar \hat{a} y \hat{b} tales que el modelo

$$\hat{x}(k+1) = \hat{a} x(k) + \hat{b} u(k)$$

reproduzca lo mejor posible las observaciones $x(k+1)$.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
- 3. Modelos paramétricos vs. no paramétricos**
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

3. Modelos paramétricos vs. no paramétricos

- **Modelos paramétricos**

En un modelo paramétrico se asume de antemano una *estructura funcional* para la relación entrada-salida, que depende de un número finito de parámetros. Ejemplos típicos:

- Modelos lineales invariantes en el tiempo (LTI) en forma de función de transferencia:

$$G(q^{-1}; \theta) = \frac{B(q^{-1}; \theta)}{A(q^{-1}; \theta)} = \frac{b_0 + b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}}{1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}}.$$

3. Modelos paramétricos vs. no paramétricos

- **Modelos paramétricos**

- Modelos ARX/ARMAX, donde la dinámica del sistema y del ruido se describen mediante polinomios en q^{-1} .
- Modelos de espacio de estados discretos:

$$x(k+1) = Ax(k) + Bu(k), \quad y(k) = Cx(k) + Du(k).$$

3. Modelos paramétricos vs. no paramétricos

- **Modelos NO paramétricos**

En los modelos no paramétricos no se impone una estructura de modelo de orden finito a priori, sino que se trata de estimar directamente ciertas características del sistema a partir de los datos:

- **Respuesta al impulso:** estimar la secuencia $\{g(k)\}$ que relaciona entrada y salida mediante

$$x(k) = \sum_{i=0}^{\infty} g(i) u(k - i) + e(k).$$

3. Modelos paramétricos vs. no paramétricos

- **Modelos NO paramétricos**

- **Función de transferencia en frecuencia:** estimar $G(e^{j\omega})$ mediante técnicas de análisis espectral.
- **Espectros de potencia:** estimación de la densidad espectral $S_{yy}(\omega)$ y la coherencia entre u y x .

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
- 4. Métodos clásicos**
 - a. **Método de mínimos cuadrados (LS)**
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

4.a. Métodos clásicos: Mínimos Cuadrados

El **método de mínimos cuadrados** es una técnica fundamental para estimar parámetros en modelos lineales respecto a los parámetros.

Consideremos el modelo lineal en parámetros:

$$x(k) = \varphi^\top(k) \theta + e(k),$$

donde:

- $\theta \in \mathbb{R}^p$ es el vector de parámetros desconocidos,
- $\varphi(k) \in \mathbb{R}^p$ es el vector de regresores (compuesto de entradas y salidas pasadas),
- $e(k)$ es el error de modelado/ruido.

4.a. Métodos clásicos: Mínimos Cuadrados

Acumulando N muestras:

$$X = \Phi\theta + E,$$

donde

$$X = \begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(N) \end{bmatrix}, \quad \Phi = \begin{bmatrix} \varphi^\top(1) \\ \varphi^\top(2) \\ \vdots \\ \varphi^\top(N) \end{bmatrix}, \quad E = \begin{bmatrix} e(1) \\ e(2) \\ \vdots \\ e(N) \end{bmatrix}.$$

4.a. Métodos clásicos: Mínimos Cuadrados

El método de mínimos cuadrados busca $\hat{\theta}$ que minimice la suma de cuadrados del error:

$$J(\theta) = \sum_{k=1}^N (x(k) - \varphi^\top(k)\theta)^2 = \|X - \Phi\theta\|^2.$$

Derivando e igualando a cero se obtiene la solución LS (si $\Phi^\top\Phi$ es invertible):

$$\hat{\theta}_{\text{LS}} = (\Phi^\top\Phi)^{-1}\Phi^\top X.$$

4.a. Métodos clásicos: Mínimos Cuadrados

Ejemplo: estimación de un sistema de primer orden Retomemos el ejemplo:

$$x(k+1) = ax(k) + bu(k) + e(k).$$

Podemos escribir:

$$x(k+1) = \underbrace{\begin{bmatrix} x(k) & u(k) \end{bmatrix}}_{\varphi^\top(k)} \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{\theta} + e(k).$$

Con datos de $k = 0$ a $N - 1$, se construye Φ con filas $\varphi^\top(k)$ y X con elementos $x(k+1)$, y se calcula $\hat{\theta}$ usando la fórmula LS.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
- 4. Métodos clásicos**
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX**
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

4.b. Métodos clásicos: Modelos ARX

Un modelo ARX (*AutoRegresivo con entrada eXógena*) describe la salida de un sistema como combinación lineal de salidas pasadas y entradas pasadas:

$$A(q^{-1})x(k) = B(q^{-1})u(k) + e(k),$$

donde

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \cdots + a_{n_a}q^{-n_a},$$

$$B(q^{-1}) = b_1q^{-1} + b_2q^{-2} + \cdots + b_{n_b}q^{-n_b},$$

y $e(k)$ es un término de error (ruido blanco idealmente).

Desarrollando la ecuación:

$$x(k) + a_1x(k-1) + \cdots + a_{n_a}x(k-n_a) = b_1u(k-1) + \cdots + b_{n_b}u(k-n_b) + e(k).$$

4.b. Métodos clásicos: Modelos ARX

Esto puede reescribirse en forma lineal en los parámetros:

$$x(k) = -a_1x(k-1) - \cdots - a_{n_a}x(k-n_a) + b_1u(k-1) + \cdots + b_{n_b}u(k-n_b) + e(k).$$

Definiendo el vector de regresores:

$$\varphi^\top(k) = \begin{bmatrix} -x(k-1) & \dots & -x(k-n_a) & u(k-1) & \dots & u(k-n_b) \end{bmatrix},$$

4.b. Métodos clásicos: Modelos ARX

y el vector de parámetros:

$$\theta^\top = \begin{bmatrix} a_1 & \dots & a_{n_a} & b_1 & \dots & b_{n_b} \end{bmatrix},$$

obtenemos:

$$x(k) = \varphi^\top(k) \theta + e(k),$$

y podemos estimar θ por mínimos cuadrados.

Los modelos ARX son sencillos de identificar y muy usados como punto de partida aunque su tratamiento del ruido puede ser limitado.

4.b. Métodos clásicos: Modelos ARX

- **Inferencia:**

Supongamos que hemos identificado un modelo ARX de la forma:

$$x(k) = a_1x(k-1) + a_2x(k-2) + \dots + b_1u(k-1) + b_2u(k-2).$$

Una vez determinados los parámetros \hat{a}_i, \hat{b}_j , la **predicción a un paso** es:

$$\hat{x}(k+1) = \hat{a}_1x(k) + \hat{a}_2x(k-1) + \dots + \hat{b}_1u(k) + \hat{b}_2u(k-1).$$

4.b. Métodos clásicos: Modelos ARX

- **Inferencia:**

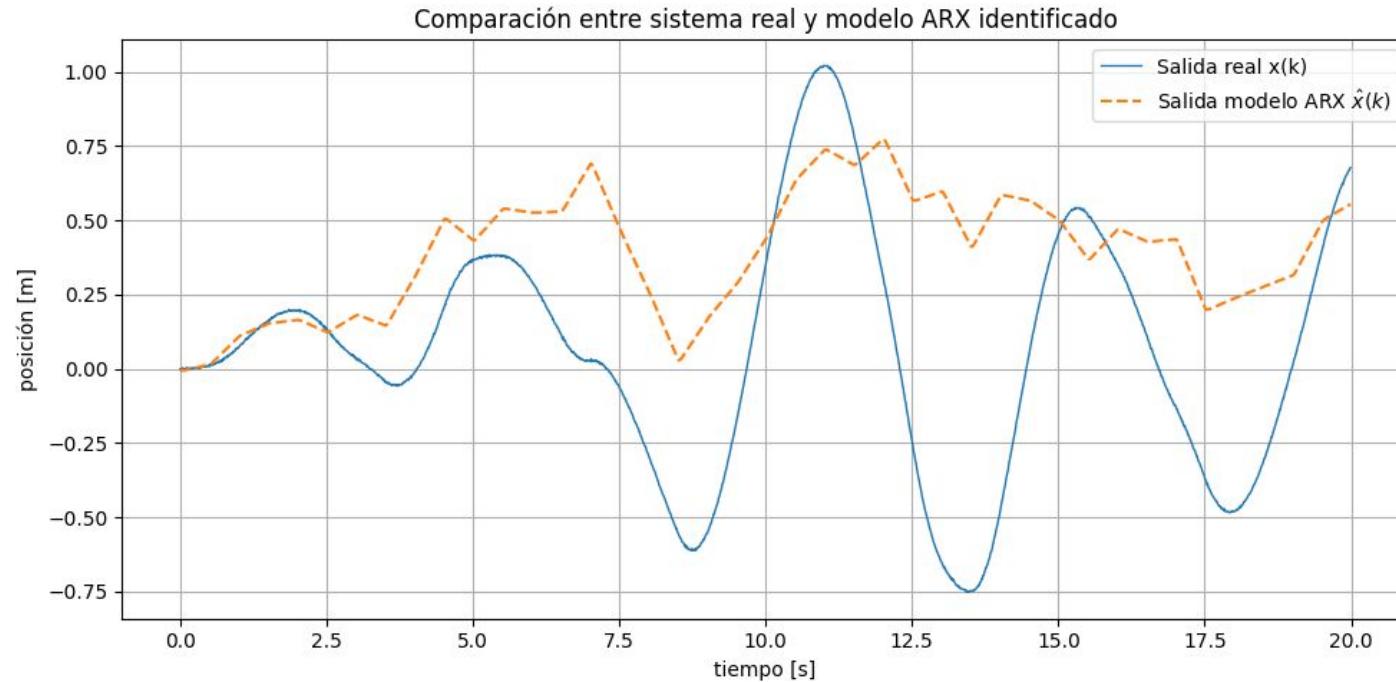
Si se desea predecir más lejos en el tiempo, se utiliza recursivamente la propia predicción como si fuera el dato real:

$$\hat{x}(k+2) = \hat{a}_1\hat{x}(k+1) + \hat{a}_2x(k) + \dots + \hat{b}_1u(k+1) + \hat{b}_2u(k).$$

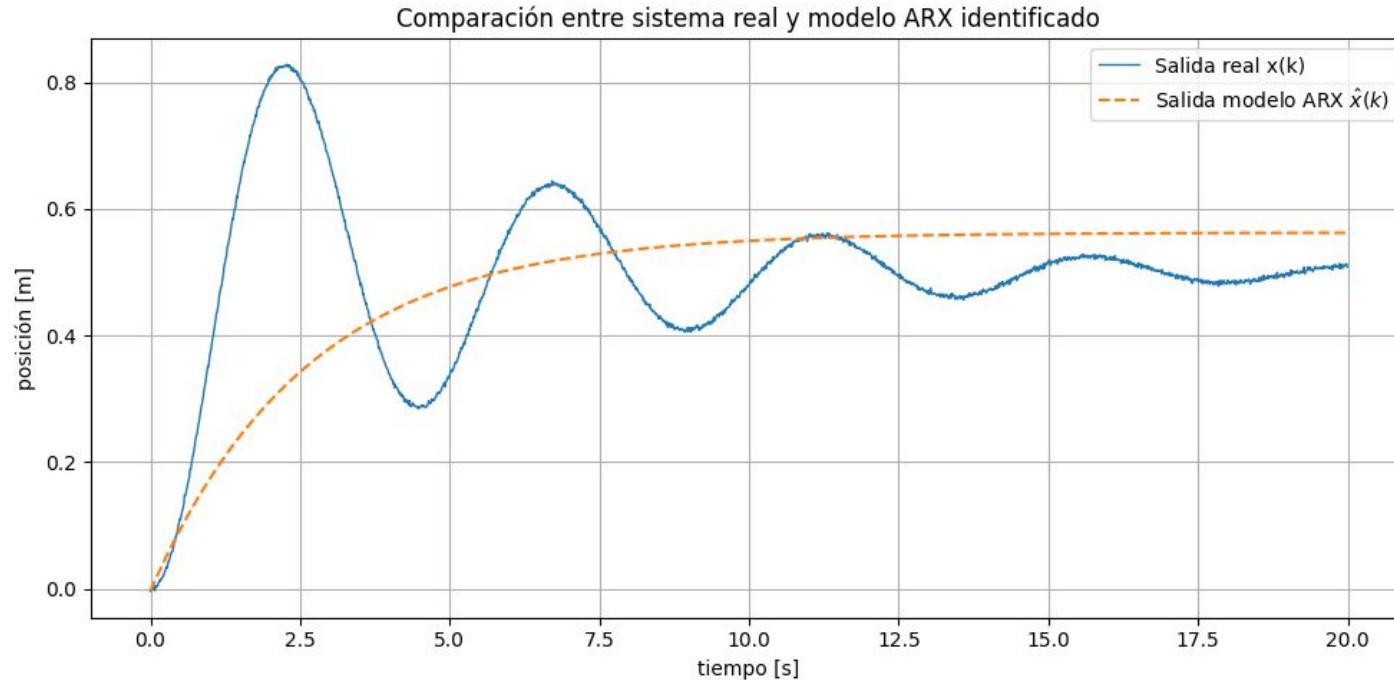
$$\hat{x}(k+3) = \hat{a}_1\hat{x}(k+2) + \hat{a}_2\hat{x}(k+1) + \dots$$

Esto permite **simular** el sistema identificado a futuro.

4.b. Métodos clásicos: Modelos ARX



4.b. Métodos clásicos: Modelos ARX



Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
- 4. Métodos clásicos**
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX**
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

4.c. Métodos clásicos: Modelos ARMAX

Los modelos ARMAX (*AutoRegresivo–Media Móvil con entrada eXógena*) incorporan un modelo más rico del ruido:

$$A(q^{-1})x(k) = B(q^{-1})u(k) + C(q^{-1})e(k),$$

donde, además de A y B , se introduce:

$$C(q^{-1}) = 1 + c_1q^{-1} + \cdots + c_{n_c}q^{-n_c}.$$

Aquí el ruido $e(k)$ entra filtrado por $C(q^{-1})$, lo que permite modelar correlaciones temporales en el ruido (es decir, ruido con estructura de media móvil).

4.c. Métodos clásicos: Modelos ARMAX

A diferencia del ARX, la presencia de $e(k)$ filtrado hace que el modelo ya no sea lineal en los parámetros de forma directa, por lo que la identificación de ARMAX suele requerir métodos de optimización iterativos (p. ej. máxima verosimilitud, gradiente, Gauss-Newton, etc.).

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
- 4. Métodos clásicos**
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados**
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

4.d. Recursividad y mínimos cuadrados

En aplicaciones en tiempo real, donde llegan datos continuamente, puede interesar actualizar la estimación de parámetros en línea. El **método de mínimos cuadrados recursivos** (RLS) permite actualizar $\hat{\theta}(k)$ al llegar cada nueva muestra $(\varphi(k), x(k))$.

La actualización típica es:

$$K(k) = \frac{P(k-1)\varphi(k)}{\lambda + \varphi^\top(k)P(k-1)\varphi(k)},$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[x(k) - \varphi^\top(k)\hat{\theta}(k-1)],$$

$$P(k) = \frac{1}{\lambda} \left[P(k-1) - K(k)\varphi^\top(k)P(k-1) \right],$$

4.d. Recursividad y mínimos cuadrados

donde:

- $P(k)$ es la matriz de covarianza asociada a la incertidumbre en los parámetros,
- $K(k)$ es la ganancia de adaptación,
- $\lambda \in (0, 1]$ es un factor de olvido que permite dar más peso a datos recientes.

4.d. Recursividad y mínimos cuadrados

En aplicaciones en tiempo real, donde llegan datos continuamente, puede interesar actualizar la estimación de parámetros en línea. El **método de mínimos cuadrados recursivos** (RLS) permite actualizar $\hat{\theta}(k)$ al llegar cada nueva muestra $(\varphi(k), x(k))$.

La actualización típica es:

$$\begin{aligned}
 K(k) &= \frac{P(k-1)\varphi(k)}{\lambda + \varphi^\top(k)P(k-1)\varphi(k)}, \\
 \hat{\theta}(k) &= \hat{\theta}(k-1) + K(k)[x(k) - \varphi^\top(k)\hat{\theta}(k-1)], \\
 P(k) &= \frac{1}{\lambda} \left[P(k-1) - K(k)\varphi^\top(k)P(k-1) \right],
 \end{aligned}$$

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5. Introducción a redes neuronales

- Cuando los sistemas son fuertemente no lineales o presentan fenómenos difíciles de capturar con modelos lineales tradicionales (ARX/ARMAX), se recurre a técnicas de **aprendizaje automático**, especialmente redes neuronales.
- Las **redes neuronales artificiales (RNA)** pueden aproximar funciones no lineales complejas. Para identificación, se suelen usar **arquitecturas que modelan la dependencia temporal**.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. **Redes neuronales para identificación**
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5.a. Redes neuronales para identificación

- Modelos **NARX** (Nonlinear AutoRegressive with eXogenous input):

$$x(k) = F(x(k-1), \dots, x(k-n_y), u(k-1), \dots, u(k-n_u)) + e(k),$$

donde $F(\cdot)$ se implementa mediante una red neuronal (por ejemplo, un perceptrón multicapa).

5.a. Redes neuronales para identificación

- **Redes recurrentes (RNN, LSTM):** incorporan memoria interna para capturar dependencias de largo plazo.

En el caso de un perceptrón multicapa, se tiene típicamente:

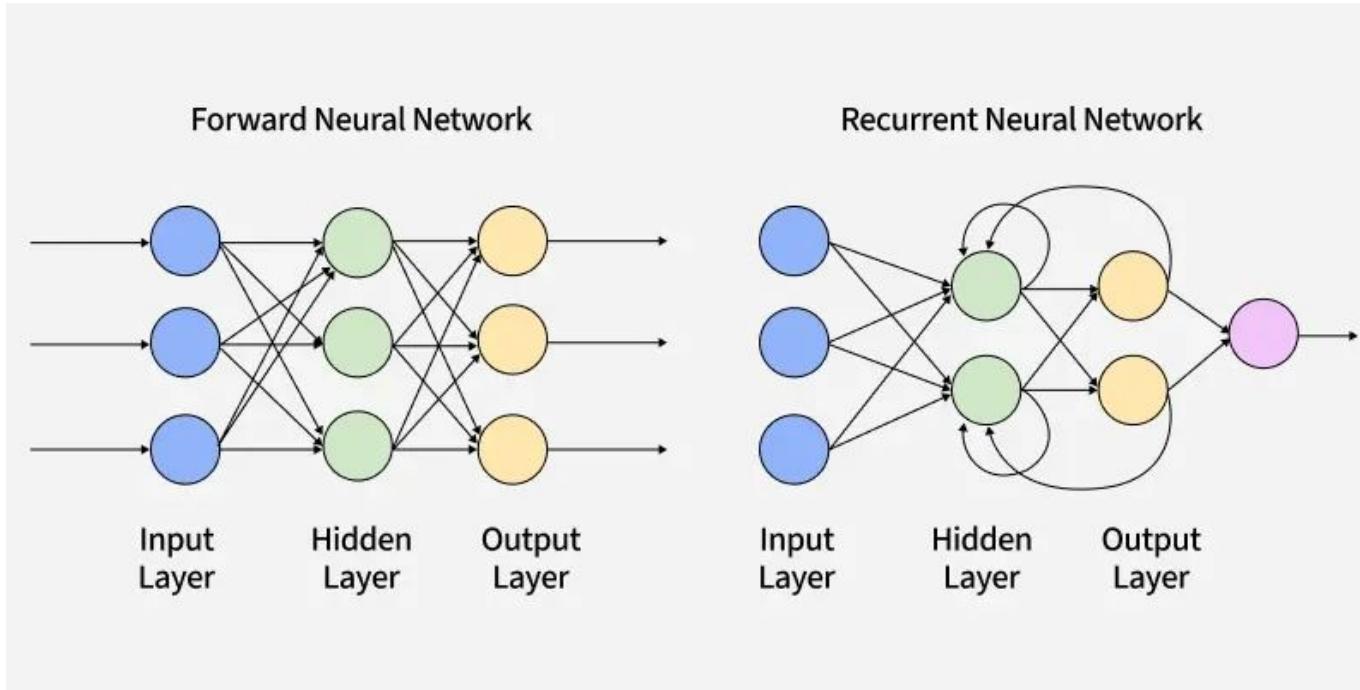
$$\hat{x}(k) = f_{\text{NN}}(\varphi(k); \theta),$$

donde $\varphi(k)$ incluye salidas y entradas pasadas, y θ agrupa todos los pesos y sesgos de la red. La estimación de θ se realiza minimizando un coste (p. ej. suma de cuadrados) mediante algoritmos como *backpropagation*.

Índice

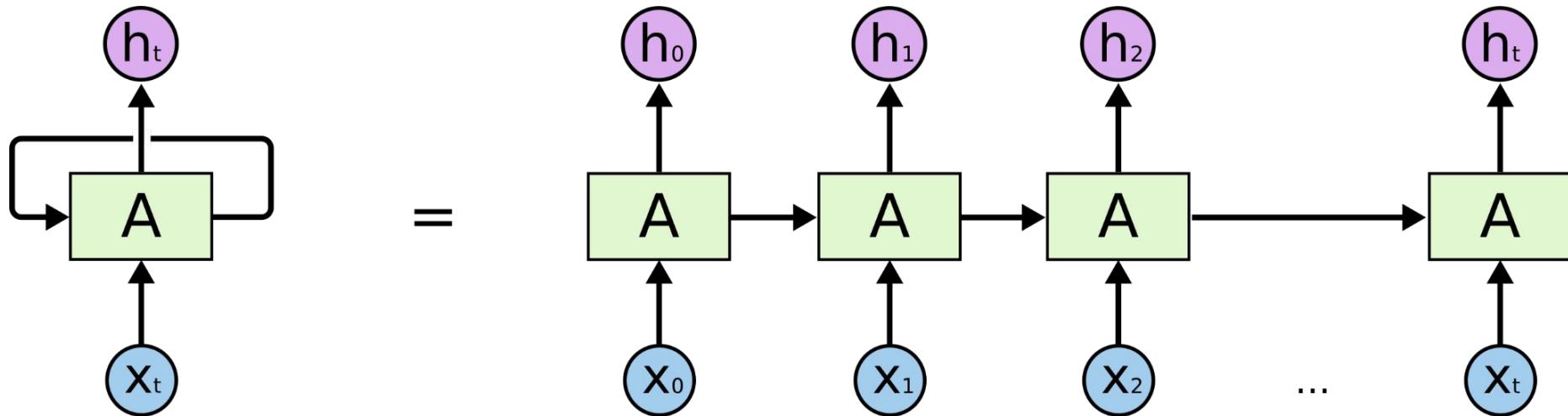
1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. **Redes neuronales para identificación (RNN)**
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5.a. Recurrent Neural Networks (RNN)



5.a. Recurrent Neural Networks (RNN)

- Es como si la red se “desenrollara” en el tiempo



5.a. Recurrent Neural Networks (RNN)

- Las **Redes Neuronales Recurrentes (RNN)** constituyen una extensión natural de los modelos de identificación basados en series temporales.
- A diferencia de los modelos ARX, donde la memoria se introduce explícitamente mediante retardos, una RNN posee un **estado interno recurrente** que se actualiza en cada instante temporal.
- Esto las hace especialmente adecuadas para modelar dinámicas no lineales complejas.

5.a. Recurrent Neural Networks (RNN)

$$h(k) = \sigma(W_h h(k-1) + W_u u(k) + W_x x(k)),$$

$$\hat{x}(k+1) = W_o h(k),$$

donde:

- $h(k)$ es el estado interno recurrente,
- $\sigma(\cdot)$ es una función de activación no lineal (típicamente tanh o ReLU),
- W_h, W_u, W_x, W_o son parámetros aprendidos mediante optimización.

5.a. Recurrent Neural Networks (RNN)

En una RNN, la memoria del sistema está codificada en el vector $h(k)$, que actúa como una aproximación al estado dinámico real del sistema. Esto permite aproximar funciones dinámicas no lineales del tipo:

$$x(k+1) = F(x(k), x(k-1), \dots, u(k), u(k-1), \dots),$$

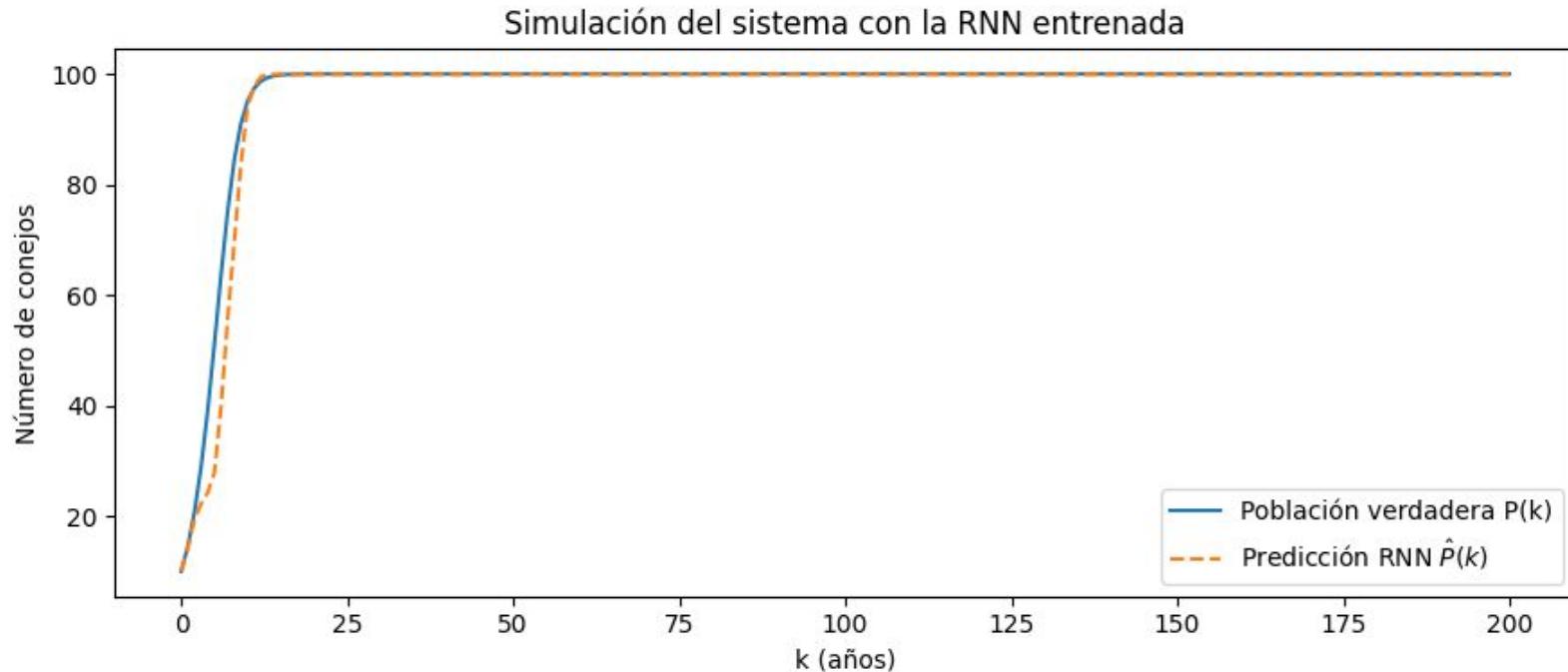
sin necesidad de elegir manualmente cuántos retardos incluir, como ocurre en los modelos ARX o NARX clásicos.

5.a. Recurrent Neural Networks (RNN)

Las RNN suelen funcionar bien para:

- Sistemas suaves con dependencias temporales cortas.
- Dinámica no lineal moderada.
- Señales de entrada ricas.
- Predicción a uno o varios pasos vista como un problema de modelado secuencial.

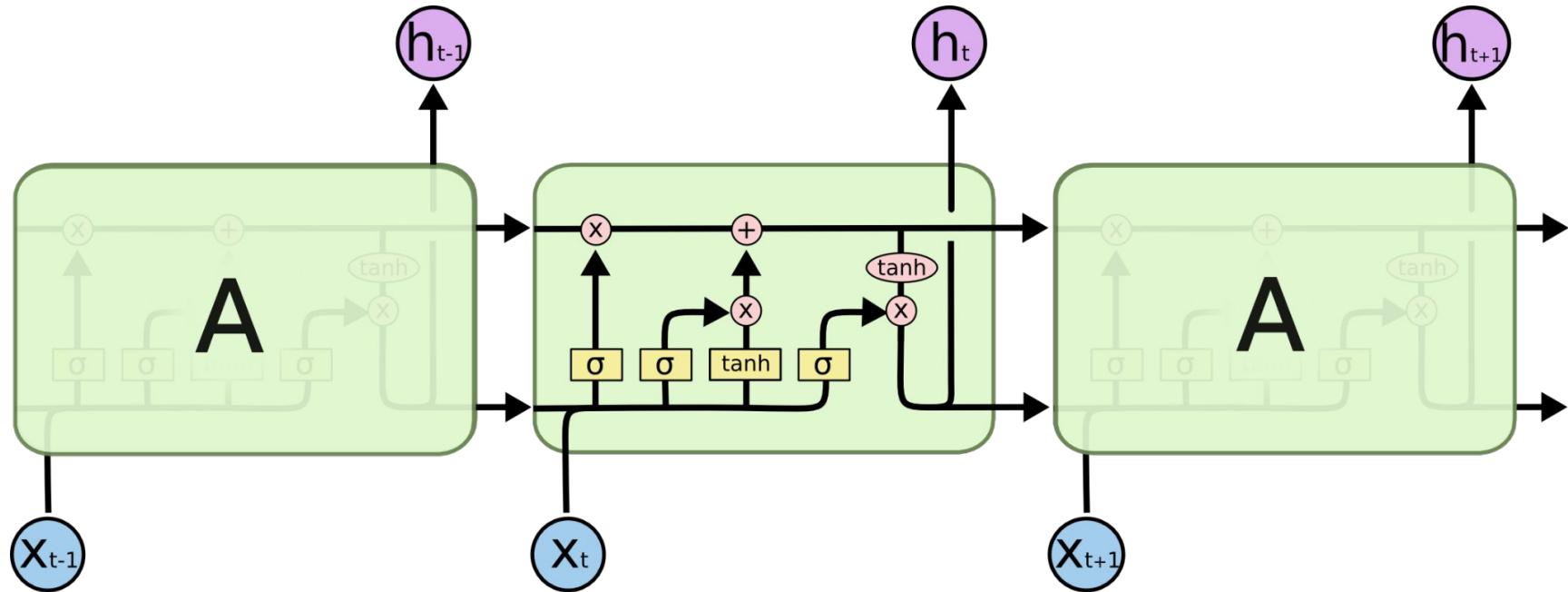
5.a. Recurrent Neural Networks (RNN)



Índice

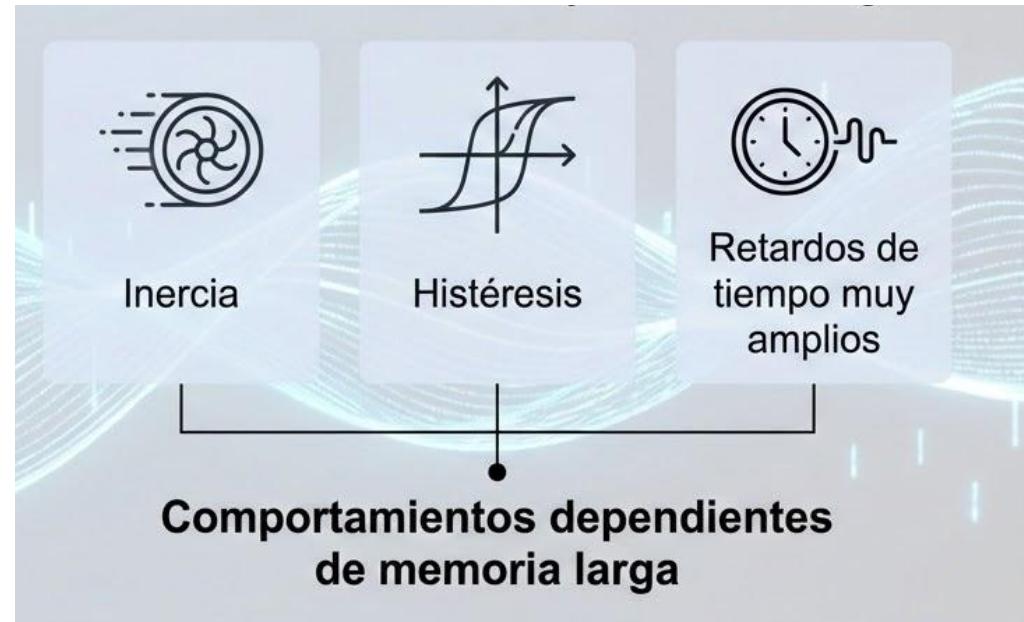
1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. **Redes neuronales para identificación (LSTM)**
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5.a. Long Short-Term Memory (LSTM)



5.a. Long Short-Term Memory (LSTM)

- Las redes **LSTM (Long Short-Term Memory)** se diseñaron específicamente para resolver el problema de los gradientes vanishing/exploding de las RNN estándar.
- En dinámica de sistemas, las LSTM permiten aprender comportamientos dependientes de **memoria larga**, como inercia, histéresis o retardos de tiempo muy amplios.



5.a. Long Short-Term Memory (LSTM)

Las LSTM son modelos extremadamente potentes para:

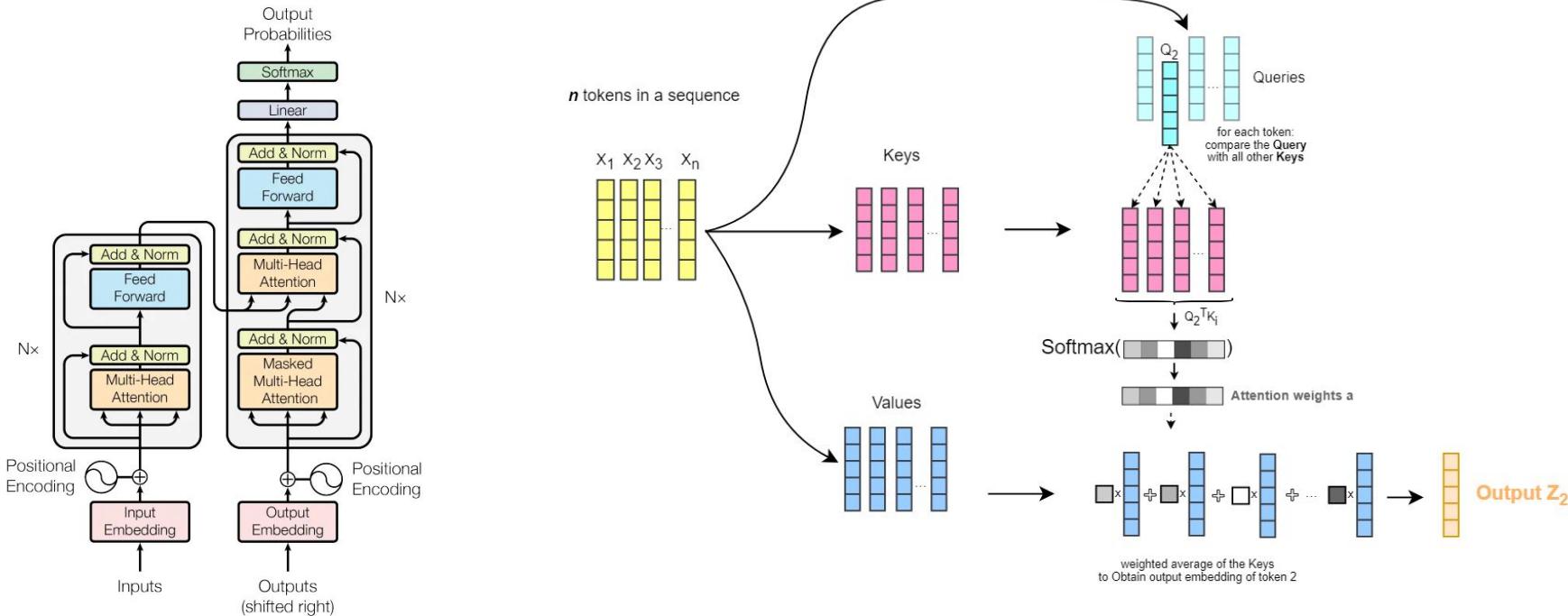
- Sistemas mecánicos con histéresis,
- Sistemas con retardos largos,
- Sistemas biológicos o ecológicos con dinámica lenta,
- Series temporales ruidosas donde la estructura se acumula lentamente.

Además, las LSTM son muy utilizadas como modelo base en estructuras **NARX no lineales**, permitiendo capturar dinámicas que serían imposibles de representar con modelos ARX clásicos de orden razonable.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. **Redes neuronales para identificación (Transformers)**
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5.a. Transformers (Attention / Self Attention)



5.a. Transformers (Attention / Self Attention)

Los **Transformers** han revolucionado el modelado secuencial en los últimos años, desplazando en muchos ámbitos a las RNN y LSTM. Su característica distintiva es el mecanismo de **self-attention**, que permite relacionar directamente todos los instantes temporales entre sí sin recurrencia.

- Aunque nacieron para lenguaje natural, hoy en día se aplican cada vez más a:
 - **Identificación de sistemas.**
 - Simulación de dinámica compleja.
 - Predicción multihorizonte.
 - Sistemas multiagente.
 - Predicción de series muy largas.

5.a. Transformers (Attention / Self Attention)

Los Transformers son especialmente potentes para:

- Sistemas complejos con alta no linealidad.
- Series temporales largas donde la memoria de largo plazo es crucial.
- Dinámica con múltiples escalas temporales.
- Estimación conjunta de entrada y estado.
- Reconstrucción de series incompletas.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. **Redes neuronales para identificación**
 - b. Modelos híbridos: físicos + datos
6. Evaluación de desempeño del modelo

5.a. Redes neuronales para identificación

Ventajas y desventajas:

- **Ventajas:** Capacidad para modelar no linealidades complejas, flexibilidad, buen desempeño cuando se dispone de muchos datos.
- **Desventajas:** Necesitan grandes conjuntos de datos, riesgo de sobreajuste, interpretabilidad limitada, selección delicada de arquitecturas (número de capas, neuronas, etc.).

5.a. Redes neuronales para identificación

Implementación (librerías):



PyTorch

Utilizada en la práctica 6 (parte 2)



TensorFlow



Keras

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
- 5. Introducción a redes neuronales y modelos híbridos**
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos**
6. Evaluación de desempeño del modelo

5.b. Modelos híbridos

Los **modelos híbridos** combinan:

- un modelo de *primeros principios* (físico),
- con un componente *aprendido a partir de datos* (por ejemplo, una red neuronal).

Una forma común de modelo híbrido es:

$$x(k) = x_{\text{físico}}(k; \theta_f) + x_{\text{NN}}(k; \theta_n) + e(k),$$

5.b. Modelos híbridos

donde:

- $x_{\text{físico}}(k; \theta_f)$ refleja la parte del sistema bien entendida (por ejemplo, ecuaciones de balance de masas, leyes de Newton, etc.),
- $x_{\text{NN}}(k; \theta_n)$ modela efectos desconocidos (no linealidades, fricciones complejas, retardos, etc.).

Otra variante es usar la red neuronal para modelar solo el *término de error* o incertidumbre de un modelo físico:

$$e_{\text{modelo}}(k) = x(k) - x_{\text{físico}}(k; \theta_f), \quad e_{\text{modelo}}(k) \approx f_{\text{NN}}(\cdot; \theta_n).$$

5.b. Modelos híbridos

Ventajas de los modelos híbridos:

- Aprovechan conocimiento físico previo, lo que reduce la cantidad de datos necesarios.
- Mejoran la interpretabilidad frente a un modelo puramente caja negra.
- Suelen generalizar mejor fuera de la región exacta donde se tomaron los datos.

Índice

1. Introducción
2. Fundamentos de la identificación de sistemas a partir de datos
3. Modelos paramétricos vs. no paramétricos
4. Métodos clásicos
 - a. Método de mínimos cuadrados (LS)
 - b. Modelos ARX
 - c. Modelos ARMAX
 - d. Recursividad y mínimos cuadrados
5. Introducción a redes neuronales y modelos híbridos
 - a. Redes neuronales para identificación
 - b. Modelos híbridos: físicos + datos
- 6. Evaluación de desempeño del modelo**

6. Evaluación de desempeño del modelo

Una vez identificado un modelo, es esencial evaluar su desempeño y verificar si es adecuado para el propósito (simulación, predicción, control, etc.).

Es habitual dividir los datos disponibles en (**IMPORTANTE**):

- **Datos de entrenamiento (training)**: usados para ajustar los parámetros del modelo.
- **Datos de validación (validation)**: usados para seleccionar hiperparámetros (orden del modelo, estructura, regularización) y evitar sobreajuste.
- **Datos de prueba (test)**: usados para estimar el desempeño final del modelo sobre datos no vistos.

Esta separación es especialmente importante cuando se usan modelos flexibles como redes neuronales.

6. Evaluación de desempeño del modelo

- **Índices de error**

Dado un conjunto de datos de evaluación $\{x(k), \hat{x}(k)\}$, se pueden usar distintos índices de error:

- **Error cuadrático medio (MSE):**

$$\text{MSE} = \frac{1}{N} \sum_{k=1}^N (x(k) - \hat{x}(k))^2.$$

- **Raíz del error cuadrático medio (RMSE):**

$$\text{RMSE} = \sqrt{\text{MSE}}.$$

6. Evaluación de desempeño del modelo

- Índices de error

- Error absoluto medio (MAE):

$$\text{MAE} = \frac{1}{N} \sum_{k=1}^N |x(k) - \hat{x}(k)|.$$

- Porcentaje de ajuste (fit):

$$\text{fit} = 100 \left(1 - \frac{\|x - \hat{x}\|}{\|x - \bar{x}\|} \right) \%,$$

donde \bar{x} es la media de x . Un valor de fit cercano al 100% indica un muy buen ajuste.

6. Evaluación de desempeño del modelo

Para comparar modelos con diferentes números de parámetros, no basta con ver el error: un modelo con más parámetros tiende a ajustar mejor los datos de entrenamiento, pero puede sobreajustar. Para penalizar la complejidad se usan criterios como:

- **Criterio de Información de Akaike (AIC):**

$$AIC = 2p + N \ln(\hat{\sigma}^2),$$

donde p es el número de parámetros del modelo y $\hat{\sigma}^2$ es una estimación de la varianza del error.

- **Criterio de Información Bayesiano (BIC):**

$$BIC = p \ln(N) + N \ln(\hat{\sigma}^2).$$

6. Evaluación de desempeño del modelo

- **Análisis de residuos**

Los **residuos** del modelo se definen como:

$$\varepsilon(k) = x(k) - \hat{x}(k).$$

6. Evaluación de desempeño del modelo

● Análisis de residuos

Para que el modelo sea adecuado, se espera que los residuos tengan las siguientes propiedades:

- Media aproximadamente cero: $\mathbb{E}[\varepsilon(k)] \approx 0$.
- No correlación temporal: la autocorrelación de $\varepsilon(k)$ debe ser cercana a cero para retardos distintos de cero.
- No correlación con la entrada $u(k)$: la correlación cruzada entre $u(k)$ y $\varepsilon(k)$ debe ser aproximadamente cero.

Si se observa estructura en la autocorrelación o correlación cruzada (picos significativos), indica que el modelo no ha capturado toda la dinámica o que hay errores de estructura.

6. Evaluación de desempeño del modelo

Validez frente al propósito:

La **validez** del modelo debe juzgarse en función del propósito para el cual se va a utilizar:

- Un modelo para **control** puede requerir precisión en un rango de frecuencias o en torno a ciertos puntos de operación.
- Un modelo para **simulación** puede necesitar buena precisión a largo plazo.
- Un modelo para **detección de fallos** puede centrarse en reproducir ciertos patrones anómalos.

Por ello, la evaluación no es solo numérica sino también contextual: un modelo muy complejo puede no ser necesario si uno más simple cumple el objetivo.

Temario

B1 - Sistemas Dinámicos

T1: Fundamentos del modelado de sistemas dinámicos

T2: Estabilidad, controlabilidad y observabilidad

T3: Paradigmas de simulación

B2 - Sistemas Complejos

T4: Lenguajes formales para modelos conceptuales

T5: Redes complejas y modelado estructural

B3 - Modelado con IA

T6: Identificación de sistemas

T7: Inteligencia artificial aplicada al modelado de sistemas

Tema 6: Identificación de sistemas

MODELOS COMPUTACIONALES Y SIMULACIÓN DE SISTEMAS
Curso 2025-2026



UNIVERSITAT D'ALACANT
UNIVERSIDAD DE ALICANTE
Escola Politècnica Superior
Escuela Politécnica Superior

