

UNIVERSIDAD POLITÉCNICA DE VALENCIA



ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES

TESINA

*Reducción de variables en la
identificación de modelos lineales
para MPC*

Autor: David Laurí Pla
Titulación: Máster en Automática e Informática Industrial
Dirección: Miguel A. Martínez Iranzo

17 de octubre de 2008

Valencia

Índice

Lista de acrónimos	1
Nomenclatura	3
1. Motivación y objetivos	6
2. Identificación de procesos dinámicos	9
2.1. Introducción	9
2.2. Proceso de Identificación	11
2.3. Obtención de datos	13
2.4. Pretratamiento de los datos	15
2.5. Bondad del modelo	20
2.6. Validación del modelo	24
2.6.1. Validación cruzada	25
3. Estructuras de modelos lineales E/S	28
3.1. Introducción	28
3.2. Modelo de predicción a 1 instante vista	29
3.3. Modelo de predicción a i instantes vista	36
4. Estimación de parámetros	39
4.1. Introducción	39
4.2. MLR	40
4.3. PCR	44
4.3.1. Reducción a las componentes principales	44
4.3.2. PCA	45
4.3.3. Regresión sobre los componentes principales	50
4.4. PLS	52
4.5. KPLS	59
4.6. DPLS	63
4.7. Selección de variables	67
4.8. PLSA	69

5. Ejemplos de aplicación	73
5.1. BJ SISO primer orden	78
5.2. MIMO	85
6. Conclusiones y futuros trabajos	93
Anexos	95
6.1. Relación señal ruido	95
Bibliografía	98

Lista de acrónimos

ARX	<i>Autoregressive Exogenous</i>
ARMAX	<i>Autoregressive Moving Average Exogenous</i>
ARIMAX	<i>Autoregressive Integral Moving Average Exogenous</i>
BJ	<i>Box-Jenkins</i>
DPLS	<i>Dynamic Partial Least Squares</i>
fdt	<i>Función de transferencia</i>
FIR	<i>Finite Impulse Response</i>
GA	<i>Genetic Algorithm</i>
IPLS	<i>Iterative PLS</i>
IVS-PLS	<i>Iterative variable selection for PLS</i>
LMOCV	<i>Leave-many-out cross validation</i>
LOOCV	<i>Leave-one-out cross validation</i>
LS	<i>Least Squares</i>
MPC	<i>Model Predictive Control</i>
MIMO	<i>Multiple Input Multiple Output</i>
MISO	<i>Multiple Input Single Output</i>
MLR	<i>Multiple Linear Regression</i>
MSEP	<i>Mean Square Error of Prediction</i>
NIPALS	<i>Nonlinear Iterative Partial Least Squares</i>
OE	<i>Output Error</i>
PCA	<i>Principal Components Analysis</i>
PCR	<i>Principal components Regression</i>
PLS	<i>Partial Least Squares</i>
PRBS	<i>Pseudo-random binary signal</i>
PRESS	<i>Prediction error sum of squares</i>
RMSECV	<i>Root Mean Square Error of Cross Validation</i>
RMSEP	<i>Root Mean Square Error of Prediction</i>
RSEP	<i>Root Square Error of Prediction</i>
QP	<i>Quadratic programming</i>
SEP	<i>Squared Error of Prediction</i>
SISO	<i>Single Input Single Output</i>

SNR *Signal to noise ratio*
SOS *Sum of Squares*
UVE-PLS *Uninformative variable elimination by PLS*

Nomenclatura

A_s	Polinomio en z^{-1} de los denominadores de la fila s de \mathbf{G}
$B_{e,s}$	Polinomio en z^{-1} del numerador de $G_{s,e}$ (salida y_s y entrada u_e)
B	Matriz de regresión en PCR ($ncp \times n_s$)
\mathcal{B}	Matriz de regresión en PLS ($ncp \times ncp$)
C_s	Polinomio en z^{-1} del numerador de H_s (salida y_s)
D_s	Polinomio en z^{-1} del denominador de H_s (salida y_s)
E	Matriz de residuos tras la identificación ($m \times n_s$)
$E(k)$	Fila k de la matriz E
$e(k)$	Fila k de la matriz E asumiendo que sólo tiene una columna
e_s	Error de identificación en la salida s (columna s de E)
\mathcal{E}	Residuos de la matriz X al realizar la descomposición ($m \times n_e$)
\mathcal{F}	Residuos de la matriz Y al realizar la descomposición ($m \times n_s$)
$G_{s,e}$	Función de transferencia que relaciona la salida y_s con la entrada u_e
\mathbf{G}	Matriz de funciones de transferencia que relaciona Y con U
H_s	Función de transferencia que relaciona la entrada ξ con la salida y_s
$J(\theta)$	Índice de coste evaluado para la matriz θ
$\mathbf{M}(\theta)$	Familia de modelos
m	Número de muestras en la matriz de regresión X
\mathbf{m}	Modelo dentro de la familia $\mathbf{M}(\theta)$ definido por $\hat{\theta}$
n	Número de muestras del experimento de identificación
n_a	Orden común de los polinomios A
n_b	Orden común de los polinomios B
n_c	Orden común de los polinomios C
n_d	Orden común de los polinomios D
n_e	Número de entradas al proceso
n_f	Tamaño de la ventana de predicción donde se evalúa el modelo
n_s	Número de salidas del proceso
n_x	Número de columnas de la matriz de regresión
ncp	Número de componentes principales utilizado
$n_{identif}$	Número de conjuntos de datos generados durante la validación interna
P	matriz de pesos de X ($n_e \times ncp$)

p_i	Vector columna i de P
p_{i_j}	Elemento j del vector columna p_i
\mathbf{R}	Matriz de referencias para las salidas ($n \times n_s$)
T_s	Periodo de muestreo en el experimento
T	Matriz de puntuaciones de X ($m \times ncp$)
t_i	Vector columna i de T
t_{i_j}	Elemento j del vector columna t_i
Q	matriz de pesos de Y ($n_s \times ncp$)
q_i	Vector columna i de Q
q_{i_j}	Elemento j del vector columna q_i
U	Matriz de puntuaciones de Y ($m \times ncp$)
u_i	Vector columna i de U
u_{i_j}	Elemento j del vector columna u_i
$\underline{\mathbf{U}}$	Matriz de entradas aplicadas al proceso ($n \times n_e$)
$\underline{\mathbf{u}}_e$	Columna e de $\underline{\mathbf{U}}$
$\underline{\mathbf{u}}_e(k)$	Fila k de $\underline{\mathbf{u}}_e$
\mathbf{U}	Matriz de entradas tras el pretratamiento ($n \times n_e$)
\mathbf{u}_e	Columna e de \mathbf{U}
$\mathbf{u}_e(k)$	Fila k del vector columna \mathbf{u}_e
U	Matriz de entradas reducida ($m \times n_e$)
$U(k)$	Fila k de la matriz U
u_e	Columna e de U
$u_e(k)$	Fila k de u_e
$u(k)$	Fila k de la matriz U asumiendo que sólo tiene una columna
W	matriz de pesos de X utilizada en PLS ($n_e \times ncp$)
w_i	Vector columna i de W
X	Matriz de regresión ($m \times n_x$)
$X(k)$	Fila k de la matriz X
x_i	Columna i de X
$x_i(k)$	Fila k de x_i
$\underline{\mathbf{X}}$	Matriz de regresión a partir de datos sin pretratamiento ($m \times n_x$)
$\underline{\mathbf{Y}}$	Matriz de salidas del proceso ($n \times n_s$)
\mathbf{Y}	Matriz de salidas del proceso tras el pretratamiento ($n \times n_s$)
$\mathbf{Y}(k)$	Fila k de la matriz \mathbf{Y}
\mathbf{y}_s	Columna s de \mathbf{Y}
$\mathbf{y}_s(k)$	Fila k de \mathbf{y}_s
$\mathbf{y}(k)$	Fila k de la matriz \mathbf{Y} asumiendo que sólo tiene una columna
Y	Matriz de salidas del proceso para identificación ($m \times n_s$)
$Y(k)$	Fila k de la matriz Y
y_s	Columna s de Y
$y_s(k)$	Fila k de y_s

$y(k)$	Fila k de la matriz Y asumiendo que sólo tiene una columna
\hat{Y}	Matriz de salidas estimadas ($m \times n_s$)
$\hat{Y}(k)$	Fila k de la matriz \hat{Y}
\hat{y}_s	Columna s de \hat{Y}
$\hat{y}_s(k)$	Fila k de \hat{y}_s
$\hat{y}(k)$	Fila k de la matriz \hat{Y} asumiendo que sólo tiene una columna
$\hat{Y}(k k-1)$	Estimación de Y en k con información hasta $k-1$
θ^*	Matriz de parámetros real ($n_e \times n_s$)
θ	Matriz de parámetros ($n_e \times n_s$)
$\hat{\theta}$	Matriz de parámetros estimada ($n_e \times n_s$)
$\tilde{\theta}$	Matriz de parámetros estimada que incluye sY y sX ($n_e \times n_s$)
ξ	Ruido blanco ($n \times 1$)
$\xi(k)$	Fila k del vector columna ξ

Capítulo 1

Motivación y objetivos

El modelado de procesos físicos mediante ecuaciones matemáticas es una herramienta ampliamente utilizada en ingeniería de control. La mayoría de las técnicas de control se apoyan en un modelo matemático para determinar las acciones a aplicar sobre el proceso con el fin de conseguir unos objetivos. Conforme más ambiciosos sean dichos objetivos, se requiere un modelo más preciso.

Según la estructura del modelo matemático a utilizar, se puede aproximar en mayor o menor medida el comportamiento dinámico y estático del proceso. Cuanto más elaborada es la estructura del modelo, éste puede aproximar mejor el comportamiento del proceso a coste de un diseño más complejo del controlador o un mayor coste computacional para el cálculo de las acciones de control. Todos los procesos físicos son de naturaleza no lineal, no obstante, se puede obtener un modelo lineal entorno a una zona de funcionamiento, lo cual es suficiente en muchos casos.

Existen multitud de técnicas de control, siendo el control predictivo multivariable¹ (MPC) la que cubre un 90% de las aplicaciones multivariable en el entorno industrial [Lii07], sobretodo en los sectores químico y petroquímico, encontrándose en expansión en la industria alimentaria, del papel y de la pasta del papel. En la industria minerometalúrgica la mayoría de bucles de control son PID o PID en cascada, aunque para los problemas multivariable se utilizan MPC o controladores fuzzy [Thw07]. La gran mayoría de las soluciones MPC industriales están basadas en modelos lineales [Ble02] e índices de coste cuadráticos, luego se trata de un problema de optimización convexo donde se puede asegurar la obtención del óptimo global, o el frente de pareto global si se define un índice de coste multiobjetivo. Sin embargo en algunas aplicaciones donde un modelo lineal no es capaz de aproximar suficientemente el comportamiento del proceso, se utilizan

¹El éxito del MPC está fundamentado en la simplicidad de las ideas sobre las que se desarrolla, la posibilidad de contemplar restricciones y la consideración de las interacciones entre los distintos subsistemas para el caso MIMO.

modelos no lineales convirtiendo el problema de optimización en un problema no convexo a resolver *on-line*. En este caso no existen técnicas que puedan asegurar la obtención del óptimo o frente de Pareto global y el tiempo de cómputo es considerablemente mayor que el necesario en el caso de un problema convexo resoluble con programación cuadrática (QP).

En el desarrollo de un proyecto industrial basado en MPC, la fase de modelado cubre la mayor parte del proyecto siendo un campo de investigación abierto en el que se pueden realizar muchas aportaciones [Zhu02, Mor99]. Uno de los principales problemas en los procesos con muchas variables es la falta de información en el conjunto de datos de identificación, lo que hace que muchas de las técnicas disponibles no sean utilizables.

La presente tesina se orienta en la identificación de modelos lineales para procesos MIMO que van a ser utilizados en un controlador predictivo, luego evaluados en una ventana de predicción. Los objetivos principales de la tesina son:

- Analizar el estado del arte de las técnicas de identificación de modelos lineales respecto a sus parámetros, profundizando en las técnicas basadas en la reducción de variables, necesarias cuando las matrices de datos de identificación no contienen mucha información del proceso.
- Definir un conjunto de indicadores donde se pueda comparar de forma rápida y robusta la calidad de varias técnicas de estimación simultáneamente. Se mide la calidad de una técnica como la capacidad de obtener un buen modelo para MPC.
- Hacer aportaciones en el campo de identificación de procesos MIMO utilizando técnicas basadas en la reducción de variables. Las aportaciones realizadas son las técnicas KPLS y PLSA, la primera intenta aunar las ventajas de PLS y PCR, mientras que la segunda intenta mejorar el modelo PLS en un entorno donde el modelo va a ser utilizado para realizar predicciones dentro de una ventana de predicción.

La tesina se estructura en 5 capítulos a parte de éste. En el capítulo 2 se introduce el concepto de identificación de procesos dinámicos y se describen los pasos a seguir desde la obtención de los datos experimentales hasta la validación del modelo. En el capítulo 3 se introducen los modelos lineales respecto a los parámetros así como la transformación de los modelos en función de transferencia a la forma general de los modelos lineales. En el capítulo 4 se exponen algunas de las técnicas de estimación de parámetros para modelos lineales respecto a sus parámetros, principalmente técnicas basadas en la reducción de variables. En el capítulo 5 se introducen dos ejemplos con el fin de comparar la calidad de algunas de las técnicas explicadas en el capítulo 4, así como las disponibles en el paquete de *software* MATLAB[®]. Finalmente en el capítulo 6 se resumen las conclusiones a las

que se ha llegado en el desarrollo de la tesina y se plantean posibles líneas de trabajo a desarrollar en la tesis.

Capítulo 2

Identificación de procesos dinámicos

2.1. Introducción

Desde el punto de vista de control, se entiende la identificación de procesos dinámicos como la determinación del modelo matemático del sistema a controlar a partir del conocimiento previo del proceso y de experiencias prácticas [Oll91].

En función del conocimiento previo de la estructura del proceso a identificar se utiliza una de las siguientes 3 alternativas que definen el tipo de modelo [Iko02]:

- Primeros principios (*white-box*): Se plantean las ecuaciones diferenciales que aproximan el comportamiento del proceso basándose en leyes físicas y relaciones (como balances de masa o energía).

Ventajas:

- Los parámetros tienen significado físico.
- El valor de los parámetros se puede determinar de forma aproximada sin disponer de datos experimentales, esto hace especialmente útiles este tipo de modelos en la fase de diseño del proceso donde aún no se dispone de datos experimentales.

Inconvenientes:

- Alto coste en tiempo de modelado ya que es necesario conocer completamente el proceso.
- En muchas ocasiones el conocimiento disponible del proceso a priori es insuficiente, lo cual limita la bondad máxima del modelo.
- La estimación de los coeficientes del modelo una vez se dispone de datos experimentales puede ser muy compleja dependiendo de la estructura

del modelo, esta característica toma una mayor relevancia en procesos variantes con el tiempo donde es necesario realizar una estimación adaptativa *on-line* de los coeficientes del modelo.

- La utilización de modelos complejos no generalizables dificulta el diseño metodológico de controladores.
- Parametrizado (*black-box*): Se escoge una estructura genérica definida como familia de modelos $\mathbf{M}(\theta)$, donde cada modelo se diferencia del resto por su matriz de parámetros θ . A partir de datos experimentales se determina el modelo que mejor aproxima el comportamiento del proceso a identificar.

Ventajas:

- Tiempo de diseño mucho menor que los modelos basados en primeros principios.
- Se pueden obtener modelos de estructura genérica, como los modelos lineales, para los que existen metodologías de diseño de controladores ampliamente desarrolladas.

Inconvenientes:

- Los parámetros no tienen significado físico.
- La elección de la familia de modelos $\mathbf{M}(\theta)$ se basa en la experiencia del diseñador y el conocimiento previo del proceso. Es necesaria una exploración de las distintas posibilidades utilizando indicadores de validación para comparar la bondad del mejor modelo obtenido en cada familia.
- Es imprescindible disponer de datos experimentales representativos del comportamiento del proceso para poder realizar la identificación.
- Híbrido (*grey-box*): Consiste en aunar las mejores características de los dos tipos anteriores, de modo que se define una estructura con una componente genérica y otra específica obtenida a partir del conocimiento previo disponible del proceso a identificar.

La presente tesina está enfocada a la estimación de parámetros en modelos lineales a partir de datos experimentales, no obstante el conocimiento previo del proceso es importante para el diseño del experimento y para la elección de la familia de modelos $\mathbf{M}(\theta)$. En el siguiente apartado se introducen los pasos a seguir durante la identificación, ampliando en posteriores apartados la información relativa a cada una de las fases más importantes en el proceso de identificación.

2.2. Proceso de Identificación

La identificación se puede entender como un proceso iterativo compuesto por varias etapas como se muestra en la figura 2.1, donde se representan mediante bloques rectangulares las fases en las que es necesaria una selección entre distintas alternativas por parte del diseñador.

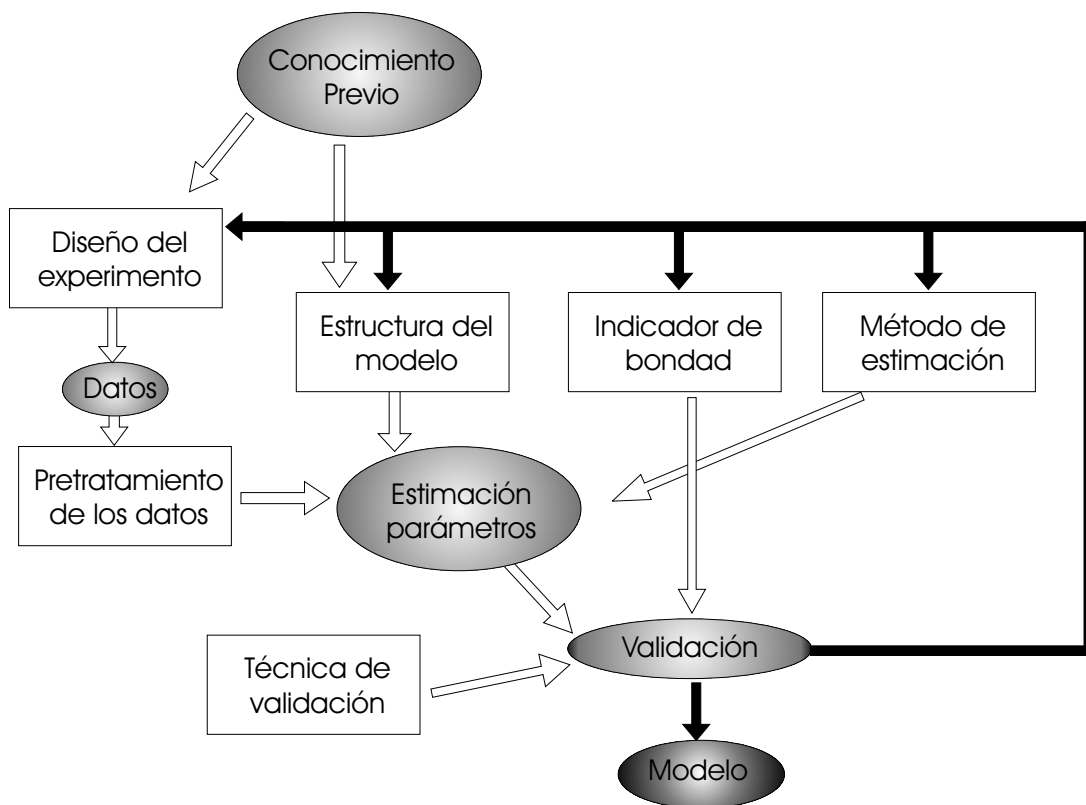


Figura 2.1: *Proceso de Identificación*

El proceso de identificación parte de un conocimiento inicial¹ del proceso para diseñar el experimento a realizar sobre el mismo con el fin de obtener los datos de identificación, el objetivo en esta fase es maximizar la información acerca del sistema contenida en los datos, al menos la información relevante del régimen en el que se va a utilizar el modelo. En el apartado 2.3 se introducen algunas nociones básicas para el diseño de experimentos.

¹Al menos se han de conocer las variables sobre las que se va a actuar \underline{U} y las que se van a registrar \underline{Y} .

Es aconsejable realizar un pretratamiento de los datos antes de seguir en el proceso de identificación, en la sección 2.4 se introducen los problemas más comunes que pueden aparecer en los datos y se explican las operaciones a realizar.

Tras la obtención de los datos de identificación se ha de decidir la estructura del modelo² apropiada para aproximar el comportamiento del proceso físico, en función de ésta se pueden aproximar comportamientos más o menos complejos. Es objetivo de la presente tesina abordar el ajuste de modelos lineales respecto a sus parámetros, dicho tipo de modelos se encuentran suficientemente explicados en el capítulo 3.

Una vez seleccionada la estructura del modelo y disponiendo de los datos de identificación tras el pretratamiento, se procede a la estimación de los parámetros con la finalidad de determinar el modelo que mejor aproxima el comportamiento del proceso físico dentro de la familia de modelos matemáticos escogida³. En esta fase es importante la selección del método de estimación, en el capítulo 4 se profundiza en algunos de los métodos disponibles, así como sus ventajas e inconvenientes.

Tras la estimación de parámetros se pueden observar las propiedades estadísticas de los residuos obtenidos, si estos presentan las propiedades del ruido blanco 3.4, se puede decir que el modelo explica los datos hasta donde se puede llegar [Box97], en caso contrario, las propiedades estadísticas del error pueden dar una idea sobre el comportamiento no modelado.

Tras obtener un modelo para el proceso, es aconsejable realizar la validación de dicho modelo con la finalidad de conocer las propiedades estadísticas de los parámetros estimados y el carácter predictivo del modelo obtenido utilizando para ello indicadores de bondad. En la sección 2.5 se introducen los indicadores de bondad más utilizados y en la sección 2.6 se introduce la filosofía empleada en las técnicas de validación.

El proceso de identificación acaba tras la fase de validación si el resultado obtenido, en términos del valor del indicador de bondad, es aceptable, en caso contrario se empieza de nuevo el proceso desde la selección de la estructura del modelo, o incluso desde el diseño del experimento si es necesario cambiar el tipo de señal para excitar más el sistema o modificar el periodo de muestreo empleado, etc.

²Por ejemplo una función de transferencia con n_a polos y n_b ceros.

³También existen técnicas de estimación no paramétricas, las cuales no emplean de forma explícita un vector de parámetros de dimensión finita en la búsqueda de la mejor descripción. Dentro de la identificación no paramétrica se distinguen dos tipos de estimaciones, las que tratan de obtener la respuesta impulsional aplicando un impulso, un escalón o una secuencia estocástica estacionaria y las que tratan de obtener la respuesta frecuencial aplicando entradas senoidales a distintas frecuencias o mediante el análisis espectral de las señales. Este tipo de técnicas requieren datos obtenidos en bucle abierto y quedan fuera del alcance del presente trabajo remitiendo al lector a las siguientes citas bibliográficas [Mor03, Lju99, Agu02].

2.3. Obtención de datos

Por lo general tanto la adquisición de datos como el posterior control se implementan sobre un soporte digital muestreando las señales continuas cada T_s ⁴. Como regla práctica se suele escoger T_s tal que la frecuencia de muestreo ($f_s = \frac{1}{T_s}$) sea del orden de diez veces el ancho de banda del proceso ($\omega_s = 10\omega_p = 2\pi f_s$) [Oll91].

Durante el diseño del experimento se ha de decidir entre un esquema de identificación en bucle abierto o en bucle cerrado, a continuación se exponen brevemente las características de ambos métodos [Zhu02]:

- **Bucle abierto:** Este método consiste en aplicar acciones de control sobre el sistema de forma que se excite la dinámica del mismo que se desea modelar. Las ventajas de este método son:
 - Se seleccionan de forma directa las propiedades de las acciones de control \underline{U} .
 - Las entradas no están correlacionadas con las perturbaciones no medibles, lo que ayuda a obtener parámetros no sesgados con técnicas de regresión como LS.

Los principales inconvenientes que presenta esta técnica son:

- No es aplicable para procesos inestables
- Es un proceso muy caro en términos de mano de obra y tiempo.
- Los datos pueden no contener información del carácter multivariable del proceso si las entradas se han de excitar de forma independiente.

Las opciones más utilizadas en la selección de \underline{U} son las siguientes [Lju99]:

- Ruido gaussiano filtrado: Se genera una señal aleatoria con espectro en frecuencia plano, luego componentes en todas las frecuencias, y se filtra para dejar sólo las componentes de frecuencia en las que se desea realizar la identificación. Este tipo de señal presenta un alto factor de cresta C_r , lo cual limita la potencia de la señal de identificación. Para la entrada e se define el factor de cresta como:

$$C_{r_e} = \sqrt{\frac{\max_k (\underline{\mathbf{u}}_e(k) - E\{\underline{\mathbf{u}}_e\})^2}{\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n (\underline{\mathbf{u}}_e(k) - E\{\underline{\mathbf{u}}_e\})^2}} \quad (2.1)$$

⁴El proceso de muestreo y digitalización convierte las señales continuas presentes en el proceso en secuencias discretas con muestras equiespaciadas en el tiempo cada T_s (asumiendo muestreo convencional).

- Señal PRBS (*pseudo-random binary signal*): Sólo puede tomar los valores $\pm \underline{u}_{max}$, luego presenta el factor de cresta ideal $C_r = 1$ maximizando la potencia de la señal de identificación, aunque no es posible definir el espectro en frecuencia de las señales, vectores columna de \underline{U} , con la misma precisión que en el caso anterior.
 - *Chirp* signal: Senoidal de frecuencia variable pero amplitud fija. Presenta un mejor factor de cresta que el ruido gaussiano y es más fácil seleccionar el rango de frecuencias presentes que en la señal PRBS.
- **Bucle cerrado:** En este esquema no se deciden las acciones de control a aplicar sino las referencias \underline{R} y es el controlador quien determina \underline{U} . Este método supera los inconvenientes presentes en el esquema en bucle abierto y añade las siguientes ventajas:
- Los datos obtenidos contienen menos no linealidades del proceso y más información en el rango de operación normal, luego se obtiene un mejor modelo para el control.
 - Más fácil de implementar que en lazo abierto.
 - El reajuste del modelo no requiere parar la planta para volver a obtener las señales de identificación.

Por contra la selección de las propiedades de \underline{U} no es tan directa como en bucle abierto y las acciones de control están correlacionadas con las perturbaciones no medibles del proceso, lo que hace que algunos de los métodos de estimación proporcionen estimaciones sesgadas de los parámetros.

A partir de datos en bucle cerrado se distinguen tres métodos de identificación:⁵

- Directo: Se utilizan \underline{U} e \underline{Y} para la obtención del modelo del proceso.
- Indirecto: Se utilizan \underline{R} e \underline{Y} para la obtención del modelo del bucle de control, posteriormente, y a partir del conocimiento detallado del controlador, se extrae el modelo del proceso.
- Unión entrada-salida: Se utilizan \underline{R} y las perturbaciones medibles como entradas y \underline{U} e \underline{Y} como salidas en la identificación, posteriormente, y a partir del conocimiento del controlador, se extrae el modelo del proceso.

Al final de esta etapa se dispone de la matriz de entradas aplicadas al proceso \underline{U} y la matriz de salidas registradas \underline{Y} . Ambas matrices tienen n filas, siendo n el número de muestras del experimento, y tantas columnas como entradas y salidas se utilicen respectivamente.

⁵Desde el punto de vista de MPC, al que está enfocado el presente trabajo, sólo el método directo es factible debido a que si se consideran restricciones, no se dispone de una expresión explícita para el controlador.

2.4. Pretratamiento de los datos

A continuación se exponen los problemas que pueden presentar los datos obtenidos de forma experimental, así como posibles soluciones para el condicionamiento del nuevo conjunto de datos superando dichos problemas:

Offset: Las señales de entrada y salida normalmente presentan *offset*, éste puede ser incluido como un parámetro más en el modelo⁶, o eliminado de las matrices de datos de identificación mediante uno de los siguientes pretratamientos:

- Filtro paso alto
- Pasar las señales a sus incrementos. Esta alternativa no es aconsejable para procesos con componentes de alta frecuencia que no se desea incluir en el modelo, ya que la transformación a incrementos equivale a filtrar las señales con un filtro paso alto con grandes ganancias para altas frecuencias como se observa en el diagrama de bode de la figura 2.2. Se define el filtro para una sola señal como:

$$out(k) = in(k) - out(k - 1)$$

$$out(1 + z^{-1}) = in$$

$$\frac{out}{in} = \frac{1}{1 + z^{-1}}$$

- El centrado de los datos consiste en quitarle el valor medio a cada señal⁷, esta operación ha de realizarse tanto en el espacio de entradas \underline{U} como en el de salidas \underline{Y} . En la figura 2.3 se observa el efecto de dicha operación, donde cada barra vertical representa el rango de una variable. De forma genérica se define la operación de centrado como:

$$OUT(i, j) = IN(i, j) - E\{IN(:, j)\} \quad (2.2)$$

Donde IN es la matriz sobre la que se desea realizar el centrado, OUT es dicha matriz tras la operación y $E\{\}$ es el operador de esperanza matemática.

Desviación estándar no homogénea: Las magnitudes físicas presentes en el proceso pueden ser captadas como señales de ordenes muy distintos, con desviaciones estándar muy distintas entre ellas. Este fenómeno es especialmente importante si se

⁶Para ello se añade una columna de unos en la matriz de regresión X , definida en la ecuación 3.1, obteniendo en el parámetro asociado a dicha columna el *offset* de la señal.

⁷Alternativamente se puede quitar el *set point* a cada columna [Wol01].

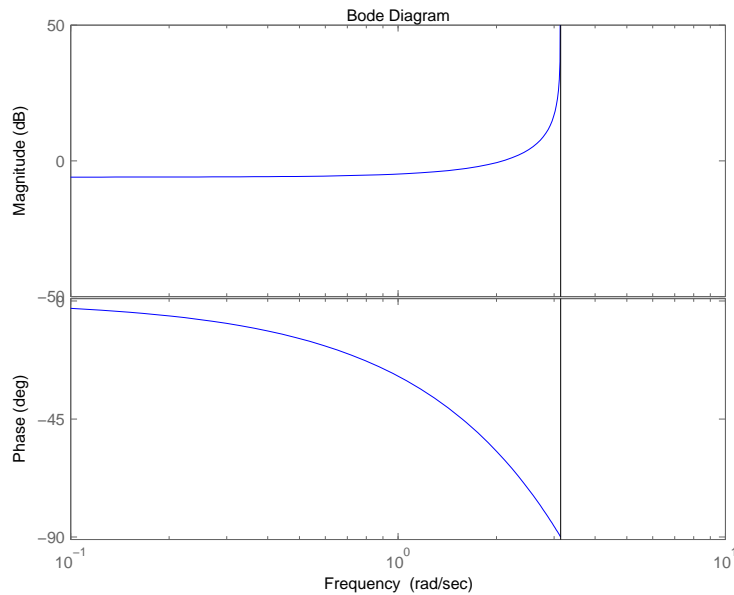


Figura 2.2: Diagrama de bode del filtro $\frac{1}{1+z^{-1}}$ con $T_s = 1$

utiliza la reducción de las matrices X e Y a sus componentes principales⁸ durante el proceso de identificación, ya que las variables con mayor desviación estándar tendrán una mayor importancia en el modelo si no se utilizan todos los componentes principales. Para evitar este efecto, es aconsejable realizar un escalado de los datos, el cual consiste en dividir cada elemento por la desviación estándar de su columna, en la figura 2.3 se observa el efecto de realizar dicha operación. Este problema no se manifiesta en técnicas que utilizan las matrices X e Y completas para realizar la estimación de parámetros, como la técnica MLR explicada en la sección 4.2. No obstante, si se trata de un proceso MIMO o SIMO, es aconsejable realizar esta operación con el fin de disponer de todas las señales en el mismo rango, evitando de este modo que las señales de mayor magnitud tengan más efecto en el índice coste. De forma genérica se define la operación de escalado como:

$$OUT(i, j) = \frac{IN(i, j)}{std(IN(:, j))} \quad (2.3)$$

Outliers: Por distintos motivos pueden aparecer datos erróneos en las matrices de identificación, dichos datos perjudican el ajuste de los parámetros del modelo luego es aconsejable eliminarlos. Existen multitud de técnicas para detectar *outliers* y poder eliminarlos, no obstante, no es el objetivo de la presente tesina la profundización en dichas técnicas.

⁸En los apartados PCR y PLS del capítulo 4, se profundiza en conceptos relacionados con los componentes principales de una matriz.

Ruido de alta frecuencia: Si se desea ajustar el modelo en un rango de frecuencias bajas y las señales tienen componentes de alta frecuencia, se pueden filtrar las señales obtenidas con un filtro paso bajo.

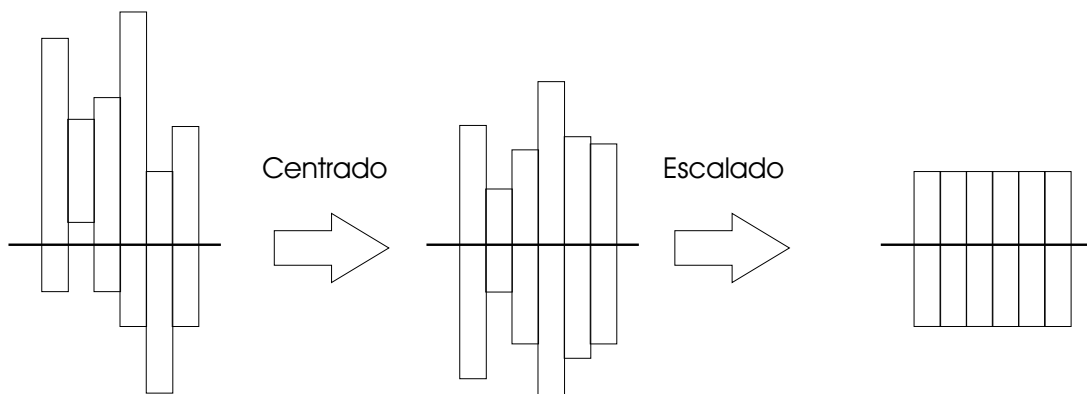


Figura 2.3: *Centrado y escalado de una matriz*

Suele ser conveniente realizar al menos el centrado y escalado de las matrices de datos [Gel86, Lea00, Cen96]. A continuación se utiliza un ejemplo simple para observar el efecto de estas dos operaciones de pretratamiento de datos sobre el resultado final de identificación, para ello se define un proceso simulado donde:

$$X = [x_1 \ x_2]$$

$$Y = X\theta + \xi = 0,2x_1 + 0,8x_2 + \xi$$

Donde:

x_1 : Secuencia aleatoria de amplitud 10, valor medio 1

x_2 : Secuencia aleatoria de amplitud 1, valor medio 5.5

ξ : Secuencia aleatoria de amplitud 0.5, valor medio 0.25

θ : Matriz de parámetros del modelo lineal

En la figura 2.4 se muestra: En la primera gráfica la matriz X con 40 muestras, en la segunda gráfica los vectores de X centrados y escalados y en la tercera gráfica la salida

Y , así como las estimaciones \hat{Y} obtenidas con PLS⁹ utilizando únicamente 1 componente principal ante distintos pretratamientos realizados sobre los datos originales, en la leyenda aparece el indicador de bondad del modelo $RSEP$ ¹⁰. Se observa que en el caso donde se han realizado tanto el centrado como el escalado de los datos, el modelo obtenido es mejor que en el resto de casos con un valor $RSEP = 1,1$.

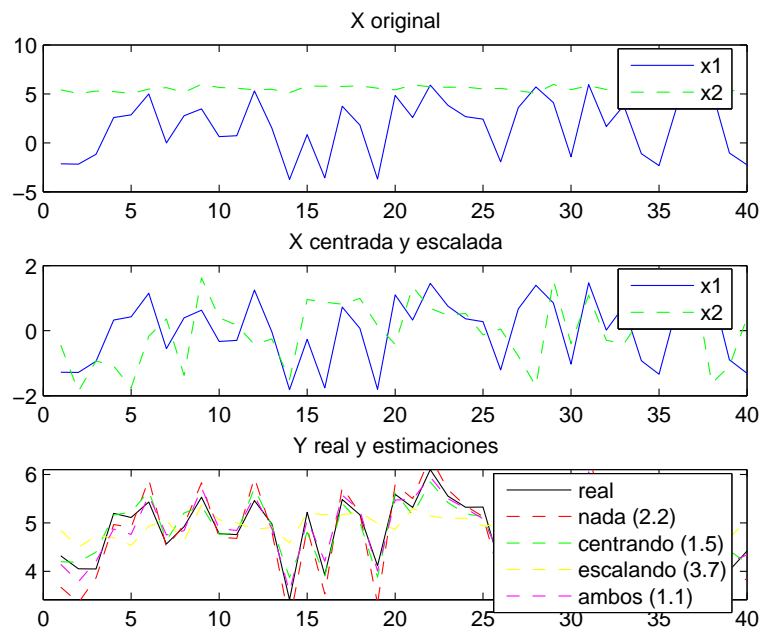


Figura 2.4: Ejemplo del efecto del pretratamiento de los datos en la estimación PLS

En los casos en los que es difícil estimar el valor medio o la desviación estándar de cada una de las variables debido a un número reducido de muestras disponibles, puede ser conveniente no realizar dichas operaciones. En la figura 2.5 se observa el resultado de un ejemplo como el anterior, pero donde se dispone de 10 muestras, en este caso sería mejor no realizar ninguna de estas operaciones de pretratamiento, ya que el menor $RSEP = 0,44$ se obtiene sin centrar ni escalar.

⁹El método de regresión lineal PLS se explica en el apartado 4.4

¹⁰Definido en el siguiente apartado, en la ecuación 2.5

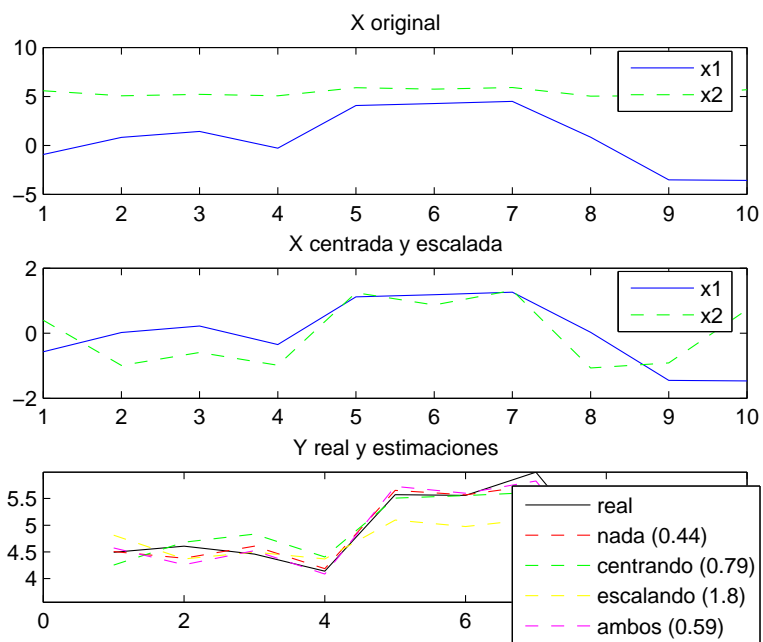


Figura 2.5: *Pretratamiento con pocas muestras*

2.5. Bondad del modelo

Se entiende la bondad de un modelo como su utilidad¹¹ ya que es una aproximación matemática de un proceso físico[Lju99]. Los indicadores de bondad se definen con el fin de disponer de una herramienta para cuantificar la calidad de la aproximación del comportamiento del proceso por un modelo concreto para una aplicación concreta. De forma general, los indicadores de bondad monodimensionales se plantean como una función del error de predicción $J(e)$, siendo e un vector columna y J una función simétrica definida positiva con un único mínimo en cero y que cumple $J : \mathbb{R}^m \rightarrow \mathbb{R}^1$, donde m es el número de filas del vector e . A continuación se nombran los indicadores más utilizados:

Error cuadrático de predicción:¹²

$$SEP = e'e \quad (2.4)$$

Raíz cuadrada del error cuadrático de predicción:

$$RSEP = \sqrt{e'e} \quad (2.5)$$

Error cuadrático medio de predicción:

$$MSEP = \frac{1}{m} e'e \quad (2.6)$$

Raíz cuadrada del error cuadrático medio de predicción:¹³

$$RMSEP = \sqrt{\frac{1}{m} e'e} \quad (2.7)$$

Coefficiente de correlación múltiple:¹⁴ Es una medida de la fuerza de la asociación lineal entre una variable dependiente y_s y dos o mas variables independientes, columnas de X .

¹¹ Se puede clasificar la utilidad de un modelo dentro de tres grupos distintos de tareas [Iko02]:

- Caracterizar o entender el comportamiento entrada-salida
- Predecir respuestas futuras
- Ajustar controladores

¹²Si las predicciones han sido obtenidas utilizando validación cruzada se denomina PRESS

¹³Si las predicciones han sido obtenidas utilizando validación cruzada se denomina RMSECV

¹⁴Si las predicciones han sido obtenidas utilizando validación cruzada se representa con Q

Su valor aumenta con el número de columnas en X , luego si se utiliza para comparar varios métodos todos han de tener el mismo número de columnas en X .

$$R = \sqrt{1 - \frac{SSerr}{SStot}} \quad (2.8)$$

Donde:

$$SSerr = e'e$$

$$SStot = (y - E\{y\})'(y - E\{y\})$$

Coefficiente de determinación múltiple: Es más significativo y robusto que R .

$$R^2 = 1 - \frac{SSerr}{SStot} \quad (2.9)$$

Se entiende como vector de error e , cada una de las columnas de la matriz de error E obtenida como la diferencia entre la salida real Y y la estimada \hat{Y} según: $E = Y - \hat{Y}$. No obstante, ante matrices \mathbf{U} e \mathbf{Y} conocidas existen tres formas de evaluar el modelo obteniendo tres posibles matrices de error distintas. A continuación se explica cada una de las formas de evaluar un modelo con la finalidad de obtener una matriz de error entre el comportamiento del proceso real y el del modelo.

Modelo de predicción En cada instante en el que se evalúa el modelo se dispone de información pasada y presente tanto de las salidas como de las entradas, el objetivo es estimar las salidas en el siguiente instante de muestreo.

Para comprender las tres formas de evaluar un modelo se plantea un ejemplo sencillo SISO con un modelo lineal $n_a = 1$ y $n_b = 0$, donde se parte de los vectores columna \mathbf{u} e \mathbf{y} obtenidos al realizar un experimento sobre el proceso real. La estructura del modelo propuesto es la siguiente:

$$\hat{y}(k) = b\mathbf{u}(k) + a\mathbf{y}(k-1) = [\mathbf{u}(k) \quad \mathbf{y}(k-1)] \begin{bmatrix} b \\ a \end{bmatrix} = X(k)\theta$$

En la parte superior de la figura 2.6 se observa el procedimiento a seguir para obtener la matriz de error, en este caso un vector columna, ante los datos experimentales dados \mathbf{u} e \mathbf{y} . En primer lugar se forma la matriz de regresión X , posteriormente se multiplica la matriz de regresión por la matriz de parámetros que define el modelo lineal, cada fila de \hat{y} se obtiene de forma directa a partir de la expresión del modelo con $u(k)$ e $y(k-1)$ conocidas. Finalmente la diferencia entre el vector y y el vector \hat{y} definen el error de predicción del modelo.

Modelo de simulación En un instante k se dispone de información pasada y presente tanto de las salidas como de las entradas, así como las entradas futuras que se van a aplicar al proceso, el objetivo es estimar las salidas en instantes posteriores a k , de esta forma los errores de predicción se van acumulando porque se utiliza la estimación en $k + i$ para realizar la estimación de las salidas en $k + i + 1$.

Retomado el ejemplo de la figura 2.6, al evaluar el modelo de simulación, se observa que no se forma la matriz de regresión X completamente desde el principio como en el caso anterior, sino que sólo se utiliza información de la salida en el instante inicial. De esta forma el vector \hat{y} no se puede obtener realizando una sola operación, sino que cada estimación se introduce en la matriz de regresión X para obtener las siguientes estimaciones, acumulando de esta forma el error cometido en cada estimación para las siguientes. En este caso la matriz de error también es un vector columna.

Modelo de predicción ventana Este es un caso intermedio entre los dos anteriores, ya que se define una ventana de predicción de tamaño n_f , y se evalúa en cada instante el modelo de simulación pero sólo para el tamaño de la ventana n_f . Las dos formas de evaluar el modelo explicadas anteriormente son casos particulares de este, el modelo de predicción se puede ver como un caso particular donde $n_f = 1$ y el modelo de simulación puede verse como un caso particular en el que $n_f = m$. Esta forma de evaluar el modelo presenta especial interés para modelos que van a ser utilizados en un controlador predictivo, donde se define una ventana de predicción.

Para el ejemplo de la figura 2.6, se observa que en lugar de obtenerse un vector columna de error, se obtiene una matriz de error donde la primera columna coincide con el vector de error obtenido en el primer caso si $n_f = 1$, y la columna i representa el error de predicción obtenido por el modelo evaluado en $k + i$, si al realizar las estimaciones sólo se disponía de información hasta k . El proceso de identificación se realiza p veces, siendo $p = m - n_f + 1$. Si $n_f = m$, la matriz E tendrá dimensiones $p \times n_f = 1 \times m$ y será el vector obtenido en el caso de modelo de simulación pero transpuesto.

Para procesos con una salida y $n_f > 1$, o procesos con más de una salida, la matriz E tiene más de un vector columna, en dichos casos los indicadores de bondad introducidos anteriormente, se evalúan para cada una de las columnas de E obteniendo indicadores en \mathbb{R}^{n_s} si se evalúa el modelo como modelo de predicción o simulación, o en $\mathbb{R}^{n_s \times n_f}$ si se evalúa como modelo de predicción ventana. En la figura 2.7 se muestra, en la parte izquierda, la evaluación de uno de los indicadores, de forma general se le llama J , para un proceso con n_s salidas y el modelo evaluado como modelo de predicción o simulación. En la parte derecha, se observa lo mismo pero con el modelo evaluado como predicción ventana. Dichos indicadores multidimensionales pueden ser transformados a

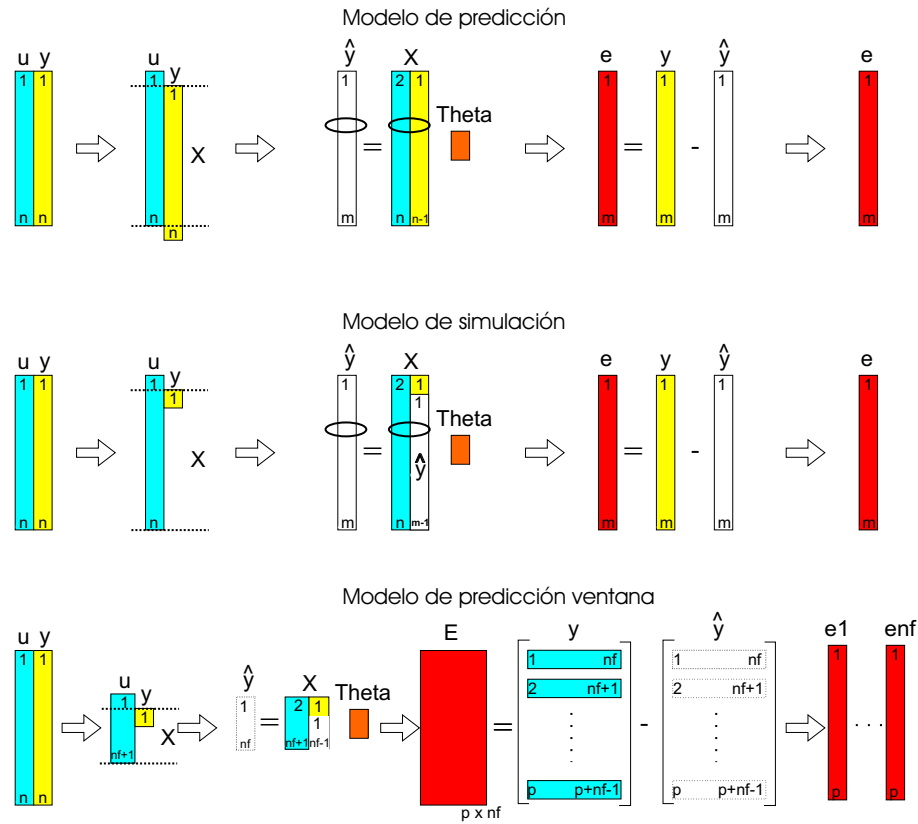


Figura 2.6: Distintas formas de evaluar el modelo

una sola dimensión utilizando una función de $\mathbb{R}^{n_s} \rightarrow \mathbb{R}^1$ ó $\mathbb{R}^{n_s \times n_f} \rightarrow \mathbb{R}^1$ con el fin de obtener un indicador monodimensional, por ejemplo se puede realizar la suma o el valor medio de los elementos, o la norma 2 del vector o matriz.

En el apartado de ejemplos del presente trabajo, se evalúan los distintos modelos obtenidos como modelos de predicción ventana y se observan los indicadores multidimensionales para comparar entre varios métodos de identificación. Al evaluar los indicadores en la ventana de predicción, se dispone de información del comportamiento del modelo no sólo en las predicciones a un instante vista, sino dentro de un horizonte de predicción, tal como se utiliza en los controladores predictivos.

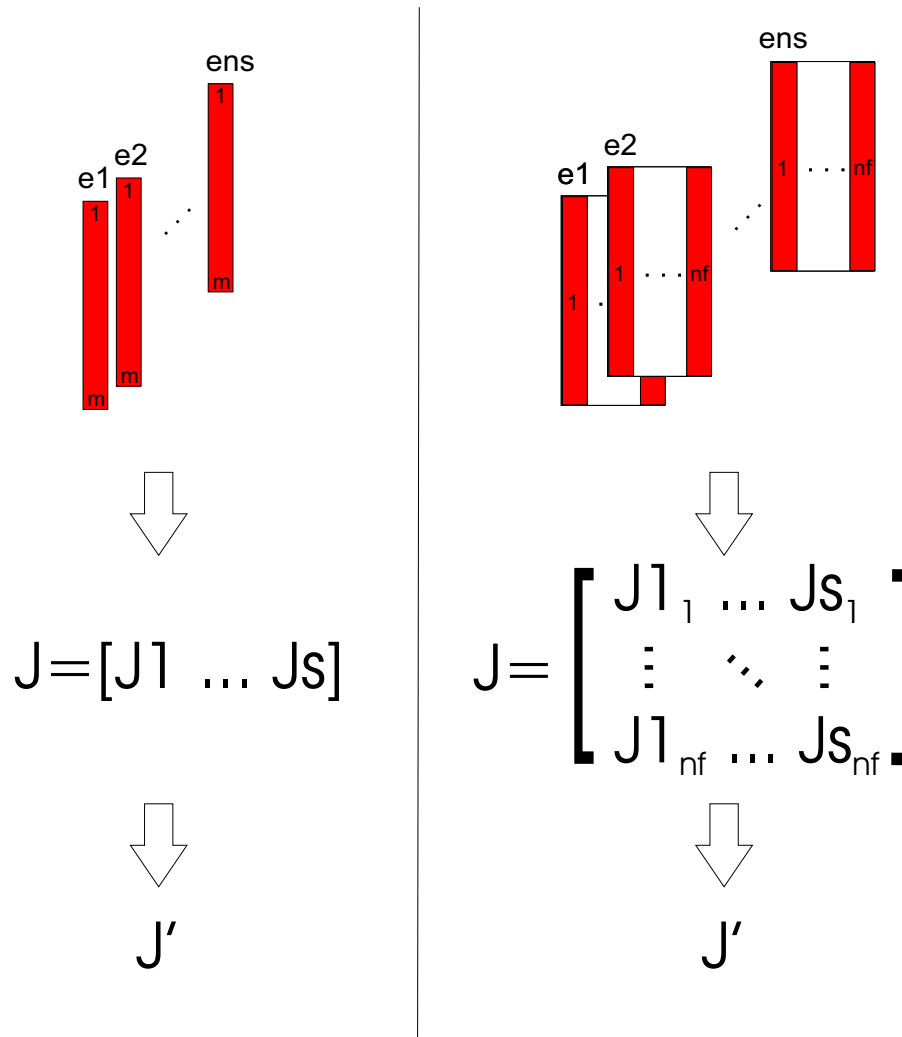


Figura 2.7: Indicadores de bondad

2.6. Validación del modelo

Se entiende la validación como el proceso de demostrar la condición de que los coeficientes de un modelo son los más apropiados para que el modelo pueda describir aceptablemente el comportamiento estructural del proceso [Zan08].

Durante el proceso de estimación de los parámetros se obtiene la matriz $\hat{\theta}$ que define el modelo $\mathbf{m} = \mathbf{M}(\hat{\theta})$ como el mejor modelo dentro de la familia de modelos dada $\mathbf{M}(\theta)$, para aproximar el comportamiento del proceso ante los datos de identificación. No obstante y debido a la presencia de perturbaciones y discrepancias entre la estructura

del modelo y la naturaleza del proceso, puede existir otra matriz de parámetros $\hat{\theta}_2$ que aproxime mejor el comportamiento del proceso para un nuevo conjunto de datos, luego es necesario disponer de herramientas para cuantificar el poder predictivo de un modelo.

La mejor solución para cuantificar el poder predictivo de un modelo es realizar un experimento de validación lo suficientemente informativo, evaluar el modelo para dicho conjunto de datos y obtener el valor de los indicadores de bondad para dicho conjunto de datos \mathbf{U} e \mathbf{Y} , este procedimiento es conocido como validación externa [Tro03].

No obstante en ocasiones no es posible realizar más experimentos sobre el proceso y sólo se dispone de un conjunto de datos, en dicho caso se puede separar el conjunto de datos en dos bloques, uno para identificar y otro para la validación externa, es especialmente importante en este caso que ambos conjuntos contengan información representativa del proceso.

Alternativamente se puede utilizar el conjunto de datos completo y realizar una validación interna. Las técnicas de validación interna generan I conjuntos de datos a partir de los datos de identificación obteniendo θ para cada uno de dichos conjuntos, a partir del conjunto de I matrices θ , se pueden extraer indicadores estadísticos de la estimación $\hat{\theta}^{15}$, basándose en la idea de que la variabilidad de $\hat{\theta}$ alrededor de θ^* ¹⁶ es reproducida por la variabilidad de θ entorno a $\hat{\theta}$. Existen multitud de técnicas de validación, dos de las más utilizadas son *bootstrap* y validación cruzada. La técnica de *bootstrap* genera los conjuntos de identificación seleccionando muestras del conjunto inicial de forma aleatoria, luego cada muestra puede aparecer más de una vez [Weh00]. A continuación se describe brevemente la técnica de validación cruzada.

2.6.1. Validación cruzada

Es una técnica de validación interna que divide los datos de entrada X e Y en 2 conjuntos, uno de identificación y otro de test de forma que toda muestra del experimento está en un conjunto o en el otro. Atendiendo al número de muestras en el conjunto de test se distingue entre LOOCV, donde el conjunto de test dispone de una sola muestra, y LMOCV, donde hay k muestras en el conjunto de test. Se estima el valor θ para el conjunto de datos de identificación, y utilizando el modelo obtenido se estima la salida para las muestras del conjunto de test, de esta forma se realizan estimaciones con muestras que no se encontraban en el conjunto utilizado para ajustar los parámetros θ .

¹⁵Estimación realizada con todos los datos de identificación.

¹⁶Aunque en los procesos físicos no existe dicha matriz θ^* por tratarse de procesos de naturaleza no lineal, la matriz θ^* representa la matriz de parámetros ideal, a la que debería converger $\hat{\theta}$ para obtener el mejor modelo dentro de la familia de modelos $\mathbf{M}(\theta)$.

Llegados a este punto se dispone de tantas estimaciones para las salidas como muestras tenía el conjunto de test. Este proceso se repite hasta que se obtenga una estimación para cada una de las muestras del conjunto inicial, de modo que en el caso LOOCV se repetirá m veces el proceso siendo m el número de muestras. En la figura 2.8 se observa el resultado de aplicar LOOCV a un conjunto de datos con 9 muestras¹⁷, en dicha figura se observa cada una de las 9 estimaciones del parámetro como la pendiente de la recta que atraviesa la nube de puntos, y la salida estimada para la muestra del conjunto de validación (símbolo + en la figura), así como el error de estimación para dicha muestra representado por un segmento discontinuo entre el símbolo + y el círculo(valor real de la salida).

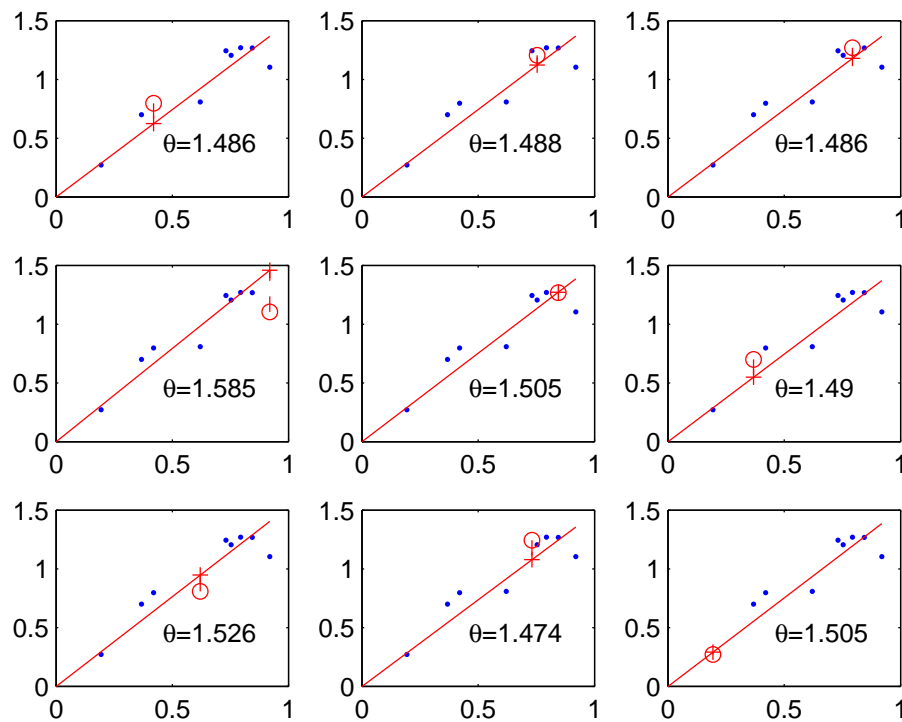


Figura 2.8: LOOCV: obtención del modelo para cada uno de los 9 conjuntos de identificación.

Al acabar el proceso se dispone de I (9 en el ejemplo abordado) matrices de parámetros θ y una matriz \hat{Y}_{CV} formada a partir de cada una de las estimaciones representada con el símbolo + en la figura 2.8. Posteriormente se evalúa el indicador de bondad del modelo deseado para la matriz de error obtenida según $E_{CV} = Y - \hat{Y}_{CV}$,

¹⁷El modelo presenta la forma $\hat{y}(k) = \theta u(k)$, en el eje de abcisas de la figura 2.8 se representa la variable u y en el eje de ordenadas la variable y .

y a partir del histograma de las I matrices θ obtenidas, se determinan las propiedades estadísticas de la estimación $\hat{\theta}$. En la figura 2.9 se muestra el resultado obtenido para el ejemplo propuesto donde θ es una matriz de dimensiones (1×1) . En la gráfica *a*) se observa la pendiente del modelo final $\hat{\theta}$ y el valor de los indicadores de bondad RMSEP, a partir de las estimaciones obtenidas por validación cruzada (J_v) y a partir de la estimación como modelo de predicción con los datos de identificación (J_i). El indicador de bondad J_i es menos representativo que J_v porque se obtiene a partir de las estimaciones realizadas con $\hat{\theta}$, la cual ha sido obtenida utilizando todas las muestras. En la gráfica *b*) se observa el histograma de las matrices θ obtenidas así como el valor de $\hat{\theta}$, de donde se puede concluir que la estimación es consistente debido a la poca variabilidad y *offset* que presenta el histograma entorno a $\hat{\theta}$.

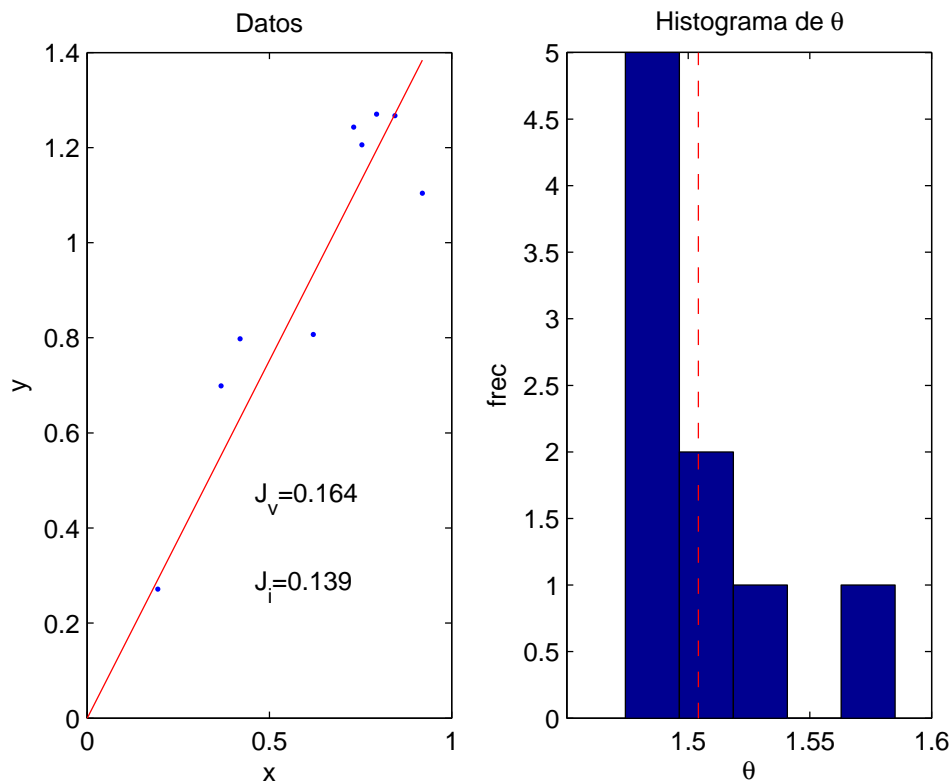


Figura 2.9: LOOCV: *a*) datos y modelo final; *b*) histograma de θ

Capítulo 3

Estructuras de modelos lineales E/S

3.1. Introducción

Se puede entender un sistema físico como la suma de un proceso determinista mas uno estocástico [Lju99]. Se define el modelo de predicción de un sistema como una aproximación del comportamiento determinista del mismo cuyo objetivo es predecir el valor de las salidas para un conjunto de entradas aplicadas conocido. No obstante, la presencia de perturbaciones no medibles afecta a la salida del sistema real, pero el valor de estas no es conocido a priori, luego todo modelo presentará una discrepancia modelo-proceso.

De forma general se puede formular un modelo de predicción como:

$$\hat{Y}(k) = f(X(k)) \quad (3.1)$$

Donde el vector fila $X(k)$ contiene tanto entradas como salidas en instantes anteriores para capturar el comportamiento dinámico del proceso, luego a partir de un vector fila $X(k)$ y la función f , es posible obtener la estimación de las salidas $\hat{Y}(k)$. El presente trabajo se centra en el ajuste de modelos lineales respecto a sus parámetros, luego la anterior ecuación se puede escribir como:

$$\hat{Y}(k) = X(k)\hat{\theta} \quad (3.2)$$

A continuación se analizan las principales ventajas e inconvenientes de utilizar estructuras lineales respecto a sus parámetros.

Ventajas:

Propiedades matemáticas: Se satisface el principio de superposición, luego es más fácil trabajar con este tipo de modelos tanto en la estimación de sus parámetros, como en el posterior diseño de controladores y análisis de estabilidad:

$$\begin{aligned}Y_1(k) &= f(X_1(k)) \\Y_2(k) &= f(X_2(k)) \\aY_1(k) + bY_2(k) &= f(aX_1(k) + bX_2(k))\end{aligned}$$

Inconvenientes:

Mundo no lineal: La naturaleza del espacio físico es no lineal, luego la linealización del proceso entorno a un punto de trabajo conlleva un error de aproximación que depende de la zona de trabajo en la que se encuentre durante su evaluación, siendo mínimo para la zona en la que han sido estimados los parámetros del modelo.

Limitados: Para aproximar relaciones entre conjuntos de datos mediante una función f , la aproximación será tanto mejor cuanto más grados de libertad presente dicha función. La regresión lineal es la aproximación más simple ya que se utiliza una recta para estimar la salida a partir de la entrada. No obstante, entorno a un punto de trabajo, la aproximación del comportamiento entrada-salida mediante una recta, es suficiente en muchos de los casos, siendo posible una estimación adaptativa *on-line* de los parámetros en casos en los que el sistema se mueva por varios puntos de funcionamiento.

En los siguientes apartados se desarrollan las expresiones de los modelos en función de transferencia más conocidas para expresarlas según la ecuación general de los modelos de predicción lineales 3.2, tanto para la predicción a 1 instante vista como para la predicción a i instantes vista.

3.2. Modelo de predicción a 1 instante vista

Una solución general para representar la dinámica de un proceso físico SISO es la estructura *Box-Jenkins* [Iko02], donde se asume que el proceso es ergódico¹:

$$y(k) = \frac{B(z^{-1})z^{-d}}{A(z^{-1})}u(k) + \frac{C(z^{-1})}{D(z^{-1})}\xi(k) = G(z^{-1})u(k) + H(z^{-1})\xi(k) \quad (3.3)$$

¹Cualquier conjunto de datos que se tome del proceso presenta las mismas características estadísticas que todo el proceso [Mor03].

Donde²:

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{n_a} z^{-n_a} \\ B(z^{-1}) &= b_1 z^{-1} + b_2 z^{-2} + \dots + b_{n_b} z^{-n_b} \\ C(z^{-1}) &= 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_{n_c} z^{-n_c} \\ D(z^{-1}) &= 1 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_{n_d} z^{-n_d} \end{aligned}$$

En las ecuaciones anteriores: $y(k)$ es el valor de la magnitud medida en el proceso físico en el instante k , la secuencia $y(k)$ se obtiene muestreando $y(t)$ a un periodo T_s . $u(k)$ es la acción de control aplicada en k . El retardo del proceso es múltiplo del periodo de muestreo según $d \cdot T_s$. Los elementos de los polinomios en z^{-1} A , B , C y D son los parámetros que definen el modelo siendo n_a , n_b , n_c y n_d sus ordenes. $\xi(k)$ es una secuencia discreta de ruido blanco tal que³:

$$\begin{aligned} \text{Media cero : } E\{\xi(k)\} &= 0 \\ \text{No correlacionada con } y : E\{\xi(k) \cdot y(k - \tau)\} &= 0 \\ \text{No correlacionada con } u : E\{\xi(k) \cdot u(k - \tau)\} &= 0 \\ \text{Correlacionada consigo misma : } E\{\xi(k) \cdot \xi(k - \tau)\} &= \sigma^2 \delta(\tau) \end{aligned} \quad (3.4)$$

Donde σ^2 es la varianza de ξ y la función delta Kronecker se define como:

$$\delta(\tau) = \begin{cases} 1, & \tau = 0 \\ 0, & \tau \neq 0 \end{cases}$$

Sin pérdida de generalidad se asume $d = 0$, luego el retardo del sistema se puede definir haciendo cero los correspondientes coeficientes de B.

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})} u(k) + \frac{C(z^{-1})}{D(z^{-1})} \xi(k) \quad (3.5)$$

La generalización al caso MIMO con sólo una señal de perturbación se puede expresar como⁴:

$$\begin{bmatrix} y_1(k) \\ \vdots \\ y_{n_s}(k) \end{bmatrix} = \begin{bmatrix} \frac{B_{1,1}}{A_1} & \dots & \frac{B_{n_e,1}}{A_1} \\ \vdots & \ddots & \vdots \\ \frac{B_{1,n_s}}{A_{n_s}} & \dots & \frac{B_{n_e,n_s}}{A_{n_s}} \end{bmatrix} \begin{bmatrix} u_1(k) \\ \vdots \\ u_{n_e}(k) \end{bmatrix} + \begin{bmatrix} \frac{C_1}{D_1} \\ \vdots \\ \frac{C_{n_s}}{D_{n_s}} \end{bmatrix} \xi(k) \quad (3.6)$$

²Notar que el término independiente del polinomio B es cero, luego $y(k)$ no depende de $u(k)$ pero sí de $[u(k-1), \dots, u(k-n_b)]$, de esta forma se considera el retardo intrínseco de un proceso muestreado.

³El operador \cdot implementa el producto *Hadamard* que puede expresarse de forma general como: $c = a \cdot b$, donde $c(i, j) = a(i, j)b(i, j)$.

⁴Por simplicidad de las expresiones, en adelante se utiliza la letra del polinomio en z^{-1} para referirse al polinomio, por ejemplo se utiliza A en lugar de $A(z^{-1})$.

El hecho de que los denominadores sean iguales entre ellos por filas no implica pérdida de generalidad, ya que se pueden asumir distintos y desarrollar según:

$$y_s(k) = \frac{\bar{B}_{1,s}}{\bar{A}_{1,s}}u_1(k) + \dots + \frac{\bar{B}_{n_e,s}}{\bar{A}_{n_e,s}}u_{n_e}(k) = \frac{\sum_{j=1}^{n_e} [\prod_{k=1}^{n_e} \bar{A}_{k,s} \bar{B}_{j,s}] u_j(k)}{\prod_{j=1}^{n_e} \bar{A}_{j,s}} \quad \forall k \neq j$$

Llegando al caso planteado donde se define un polinomio A por cada salida:

$$B_{e,s} = \prod_{k=1}^{n_e} \bar{A}_{k,s} \bar{B}_{e,s} \quad \forall k \neq e$$

$$A_s = \prod_{j=1}^{n_e} \bar{A}_{j,s}$$

Retomando la función de transferencia para el caso SISO 3.5 con el objetivo de expresarla de forma análoga a la ecuación general de los modelos lineales 3.2, se puede operar de la siguiente forma:

$$ADy(k) = BDu(k) + AC\xi(k)$$

$$(1 - 1 + AD)y(k) = BDu(k) + AC\xi(k)$$

$$y(k) = (1 - AD)y(k) + BDu(k) + AC\xi(k) \quad (3.7)$$

En la ecuación 3.7 se observa que el modelo *Box-Jenkins* relaciona la salida en k con la salida, la entrada controlable y la perturbación en instantes anteriores según:

$$y(k) = f(y(k - \alpha), u(k - \beta), \xi(k - \gamma)), \quad \forall \begin{matrix} \alpha \in [1, \dots, n_a + n_d] \\ \beta \in [1, \dots, n_b + n_d] \\ \gamma \in [0, \dots, n_c + n_a] \end{matrix}$$

Se observa que f depende del ruido ξ , el cual es una perturbación no medible que no puede aparecer en el modelo de predicción donde el objetivo es predecir la salida en k con información disponible hasta $k - 1$. Es necesario obtener un modelo que relacione la salida con valores anteriores de la salida, la acción de control y el error en anteriores estimaciones⁵:

$$\hat{y}(k|k - 1)^6 = g(y(k - \alpha), u(k - \beta), e(k - \gamma)), \quad \forall \begin{matrix} \alpha \in [1, \dots, n_a + n_d] \\ \beta \in [1, \dots, n_b + n_d] \\ \gamma \in [1, \dots, n_c + n_a] \end{matrix}$$

⁵Alternativamente se podría obtener en función de valores anteriores de la salida, la acción de control y el valor de estimaciones anteriores de la salida.

⁶Esta expresión se entiende como la salida \hat{y} en el instante k con información disponible hasta $k - 1$.

Se define el error en la estimación SISO como:

$$e(k) = y(k) - \hat{y}(k|k-1) \quad (3.8)$$

En el mejor de los casos⁷ el error de estimación del modelo en k será igual a la parte estocástica del proceso en dicho instante:

$$y(k) = g(\dots) + \xi(k) = \hat{y}(k|k-1) + e(k)$$

Asumiendo $\xi(k) = e(k)$, la ecuación 3.7 queda como:

$$y(k) = (1 - AD)y(k) + BDu(k) + ACe(k) \quad (3.9)$$

Con el objetivo de separar la información $k - j$, $\forall j > 0$ de la información en k en la ecuación 3.9, se puede descomponer el término $ACe(k)$ en dos términos:

$$ACe(k) = e(k) + (AC - 1)e(k) \quad (3.10)$$

El primer sumando contiene información del error en k . El segundo sumando depende exclusivamente de valores anteriores de e ya que tanto A como C se han definido mónicos⁸, luego el término independiente de $AC - 1$ es cero.

A partir de 3.9 y 3.10, se obtiene:

$$y(k) = (1 - AD)y(k) + BDu(k) + (AC - 1)e(k) + e(k)$$

⁷Asumiendo A y C polinomios mónicos y $\xi(k - j)$ conocido $\forall j > 0$.

⁸su término independiente vale 1.

Se deduce el modelo de predicción a un instante vista como:⁹

$$\hat{y}(k|k-1) = (1 - AD)y(k) + BDu(k) + (AC - 1)e(k) \quad (3.11)$$

Para simplificar la notación se redefine el modelo como:

$$\hat{y}(k|k-1) = \tilde{A}y(k) + \tilde{B}u(k) + \tilde{C}e(k)$$

Donde:

$$\tilde{A} = 1 - AD = -(a_1 + d_1)z^{-1} - (a_2 + d_2 + a_1d_1)z^{-2} - \dots - (a_{n_a}d_{n_d})z^{-(n_a+n_d)}$$

$$\tilde{B} = BD = b_1z^{-1} + (b_2 + b_1d_1)z^{-2} + \dots + (b_{n_b}d_{n_d})z^{-(n_b+n_d)}$$

$$\tilde{C} = AC - 1 = (a_1 + c_1)z^{-1} + (a_2 + c_2 + a_1c_1)z^{-2} + \dots + (a_{n_a}c_{n_c})z^{-(n_a+n_c)}$$

Retomando la expresión general de los modelos lineales 3.2 para un proceso MIMO 3.6 donde todos los polinomios A , B , C y D son del mismo orden¹⁰, se definen:

$$\hat{Y}(k|k-1) = [\hat{y}_1(k|k-1) \dots \hat{y}_{n_s}(k|k-1)] \quad (3.12)$$

$$X(k|k-1) = \begin{bmatrix} u_1(k-1) \dots u_1(k-n_{\tilde{b}}), \dots, u_{n_e}(k-1) \dots u_{n_e}(k-n_{\tilde{b}}), \\ y_1(k-1) \dots y_1(k-n_{\tilde{a}}), \dots, y_{n_s}(k-1) \dots y_{n_s}(k-n_{\tilde{a}}), \\ e_1(k-1) \dots e_1(k-n_{\tilde{c}}), \dots, e_{n_s}(k-1) \dots e_{n_s}(k-n_{\tilde{c}}) \end{bmatrix} \quad (3.13)$$

⁹En [Lju99] se define el modelo de predicción como:

$$\hat{y}(k|k-1) = H(z^{-1})^{-1}G(z^{-1})u(k) + (1 - H(z^{-1})^{-1})y(k)$$

A continuación se demuestra que ambas expresiones son equivalentes:

$$\hat{y} = \frac{D}{C} \frac{B}{A} u + \left(1 - \frac{D}{C}\right) y = \frac{BD}{CA} u + \frac{A(C-D)}{AC} y$$

$$AC\hat{y} = BDu + A(C-D)y$$

$$\hat{y} = (1 - AC)\hat{y} + BDu + ACy - ADy = \hat{y} + BDu + AC(y - \hat{y}) - ADy = \hat{y} + BDu + ACe - ADy$$

$$\hat{y} = \hat{y} + BDu + (AC-1)e + e + (1-AD)y - y = BDu + (AC-1)e + e + (1-AD)y - e = BDu + (AC-1)e + (1-AD)y$$

¹⁰Sin pérdida de generalidad, ya que en caso contrario los coeficientes correspondientes serán cero.

$$\hat{\theta} = \begin{bmatrix} \tilde{B}_{1,1} & \dots & \tilde{B}_{n_e,1} & \tilde{A}_1 & 0 & \dots & \dots & 0 & \tilde{C}_1 & 0 & \dots & \dots & 0 \\ & & & 0 & \ddots & & & \vdots & 0 & \ddots & & & \vdots \\ \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ & & & \vdots & & \ddots & & \vdots & \vdots & & \ddots & & \vdots \\ \tilde{B}_{1,n_s} & \dots & \tilde{B}_{n_e,n_s} & 0 & \dots & \dots & 0 & \tilde{A}_{n_s} & 0 & \dots & \dots & 0 & \tilde{C}_{n_s} \end{bmatrix}^T \quad (3.14)$$

Los elementos marcados como ceros son vectores del mismo tamaño que los polinomios de la submatriz diagonal de su misma fila \tilde{A} o \tilde{C} . Los polinomios \tilde{A} , \tilde{B} y \tilde{C} de la matriz $\hat{\theta}$ se refieren a sus coeficientes sin incluir el operador z^{-i} , ya que dicho operador ya está considerado en la matriz $X(k|k-1)$.

Finalmente se han obtenido las 3 matrices $\hat{Y}(k|k-1)$, $X(k|k-1)$ y $\hat{\theta}$ que definen el modelo lineal genérico 3.2 en función de los parámetros del modelo *Box-Jenkins* MIMO. Por simplicidad de notación, se da nombre a las submatrices de las ecuaciones 3.13, 3.14 y se reescribe la ecuación 3.2 según:

$$\hat{Y}(k|k-1) = [\mathbb{U}(k) \quad \mathbb{Y}(k|k-1) \quad \mathbb{E}(k|k-1)] \begin{bmatrix} \tilde{\mathbb{B}} \\ \tilde{\mathbb{A}} \\ \tilde{\mathbb{C}} \end{bmatrix} = X(k|k-1)\hat{\theta} \quad (3.15)$$

A continuación se definen las matrices \tilde{A} , \tilde{B} y \tilde{C} para modelos en función de transferencia más simples como casos particulares del modelo *Box-Jenkins*¹¹:

- ARX: C=1; D=A

$$y(k) = \frac{B}{A}u(k) + \frac{1}{A}\xi(k) \quad (3.16)$$

$$Ay(k) = Bu(k) + \xi(k)$$

$$y(k) = (1 - A)y(k) + Bu(k) + \xi(k)$$

$$\hat{y}(k|k-1) = (1 - A)y(k) + Bu(k) \quad (3.17)$$

$$\tilde{A} = (1 - A); \quad \tilde{B} = B; \quad \tilde{C} = 0$$

¹¹Por simplicidad de notación se utiliza un caso SISO, no obstante la extensión al caso MIMO queda contemplada utilizando los subíndices e y s como se observa en la matriz de parámetros general 3.14.

- OE: $C=1$; $D=1$

$$y(k) = \frac{B}{A}u(k) + \xi(k) \quad (3.18)$$

$$Ay(k) = Bu(k) + A\xi(k)$$

$$y(k) = (1 - A)y(k) + Bu(k) + (A - 1)e(k) + e(k)$$

$$\hat{y}(k|k-1)^{12} = (1 - A)y(k) + Bu(k) + (A - 1)e(k) \quad (3.19)$$

$$\tilde{A} = (1 - A); \quad \tilde{B} = B; \quad \tilde{C} = (A - 1)$$

- ARMAX: $D=A$

$$y(k) = \frac{B}{A}u(k) + \frac{C}{A}\xi(k) \quad (3.20)$$

$$Ay(k) = Bu(k) + C\xi(k)$$

$$y(k) = (1 - A)y(k) + Bu(k) + (C - 1)e(k) + e(k)$$

$$\hat{y}(k|k-1) = (1 - A)y(k) + Bu(k) + (C - 1)e(k) \quad (3.21)$$

$$\tilde{A} = (1 - A); \quad \tilde{B} = B; \quad \tilde{C} = C - 1$$

- ARIMAX(CARIMA): $D = \Delta A$ ($\Delta = 1 - z^{-1}$)

$$y(k) = \frac{B}{A}u(k) + \frac{C}{\Delta A}\xi(k) \quad (3.22)$$

$$\Delta Ay(k) = \Delta Bu(k) + C\xi(k)$$

$$y(k) = (1 - A\Delta)y(k) + B \Delta u(k) + (C - 1)e(k) + e(k)$$

$$\hat{y}(k|k-1) = (1 - A\Delta)y(k) + B \Delta u(k) + (C - 1)e(k) \quad (3.23)$$

$$\tilde{A} = (1 - A\Delta); \quad \tilde{B} = B\Delta; \quad \tilde{C} = C - 1$$

¹²Se puede comprobar que esta expresión es equivalente a la utilizada en la bibliografía [Iko02]:

$$\begin{aligned} \hat{y}(k|k-1) &= (1 - A)y(k) + Bu(k) + (A - 1)e(k) \\ &= (1 - A)y(k) + Bu(k) + (A - 1)(y(k) - \hat{y}(k)) \\ &= (1 - A)y(k) + Bu(k) - (1 - A)y(k) + (1 - A)\hat{y}(k) \\ &= Bu(k) + (1 - A)\hat{y}(k) \end{aligned}$$

No obstante se opta por una definición distinta a la de la bibliografía con el fin de referir todos los modelos en función de transferencia a una misma estructura de modelo lineal.

La expresión obtenida para el modelo de predicción a un instante vista es lineal respecto a sus parámetros si se evalúa en un instante k disponiendo de información hasta $k - 1$. No obstante, desde el punto de vista de identificación donde se necesita disponer a priori de las matrices X e Y , el modelo obtenido no es lineal ya que X contiene la submatriz \mathbb{E} , la cual se obtiene utilizando $\hat{\theta}$, luego desde el punto de vista de identificación, sólo se puede considerar lineal respecto a sus parámetros el modelo cuya submatriz $\tilde{\mathbb{C}}$ sea cero, lo que elimina la presencia de \mathbb{E} en X . El único modelo de predicción a 1 instante vista cuya matriz $\tilde{\mathbb{C}}$ es nula, es el obtenido a partir de la estructura ARX, luego desde el punto de vista de identificación de modelos lineales respecto a sus parámetros, se utiliza el modelo ARX para aproximar el comportamiento del proceso. La expresión 3.15 para el caso ARX se presenta como:

$$\hat{Y}(k|k-1) = [\mathbb{U}(k) \quad \mathbb{Y}(k|k-1)] \begin{bmatrix} \tilde{\mathbb{B}} \\ \tilde{\mathbb{A}} \\ \tilde{\mathbb{C}} \end{bmatrix} = X(k|k-1)\hat{\theta} \quad (3.24)$$

3.3. Modelo de predicción a i instantes vista

Se puede expresar el modelo de predicción a i instantes vista partiendo de la ecuación 3.15 de la siguiente forma:

$$\hat{Y}(k+i) = [\mathbb{U}(k+i) \quad \mathbb{Y}(k+i|k-1) \quad \mathbb{E}(k+i|k-1)] \begin{bmatrix} \tilde{\mathbb{B}} \\ \tilde{\mathbb{A}} \\ \tilde{\mathbb{C}} \end{bmatrix} = X(k+i)\hat{\theta} \quad (3.25)$$

Donde:

$$\begin{aligned} \mathbb{U}(k+i) &= [u_1(k+i-1) \dots u_1(k+i-n_{\tilde{b}}), \dots, u_{n_e}(k+i-1) \dots u_{n_e}(k+i-n_{\tilde{b}})] \\ \mathbb{Y}(k+i|k-1) &= [y_1(k+i-1) \dots y_1(k+i-n_{\tilde{a}}), \dots, y_{n_s}(k+i-1) \dots y_{n_s}(k+i-n_{\tilde{a}})] \\ \mathbb{E}(k+i|k-1) &= [e_1(k+i-1) \dots e_1(k+i-n_{\tilde{c}}), \dots, e_{n_s}(k+i-1) \dots e_{n_s}(k+i-n_{\tilde{c}})] \end{aligned}$$

Si $\alpha > k - 1$

$$y_s(\alpha) = \hat{y}_s(\alpha) \quad e_s(\alpha) = 0, \quad \forall s \in [1 \dots n_s]$$

En caso contrario

$$y_s(\alpha) = y_s(\alpha) \quad e_s(\alpha) = e_s(\alpha), \quad \forall s \in [1 \dots n_s]$$

A continuación se desarrolla un ejemplo sencillo utilizando un proceso ARMAX con todos sus polinomios de orden 2¹³ con la finalidad de llegar a la expresión general de la ecuación 3.25:

$$y(k) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} u(k) + \frac{1 + c_1 z^{-1} + c_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \xi(k)$$

Desarrollando la expresión anterior se llega a:

$$y(k) = b_1 u(k-1) + b_2 u(k-2) - a_1 y(k-1) - a_2 y(k-2) + \xi(k) + c_1 \xi(k-1) + c_2 \xi(k-2) \quad (3.26)$$

Se define la predicción como¹⁴:

$$y(k) = \hat{y}(k) + \xi(k)$$

$$\hat{y}(k) = b_1 u(k-1) + b_2 u(k-2) - a_1 y(k-1) - a_2 y(k-2) + c_1 \xi(k-1) + c_2 \xi(k-2)$$

Se puede obtener el modelo de predicción a un instante vista a partir de la ecuación del modelo expandida 3.26 de la siguiente forma:

$$y(k+1) = b_1 u(k) + b_2 u(k-1) - a_1 y(k) - a_2 y(k-1) + \xi(k+1) + c_1 \xi(k) + c_2 \xi(k-1)$$

Desarrollando para eliminar los términos no disponibles ($y(k)$):

$$y(k+1) = b_1 u(k) + b_2 u(k-1) - a_1 (\hat{y}(k) + \xi(k)) - a_2 y(k-1) + \xi(k+1) + c_1 \xi(k) + c_2 \xi(k-1)$$

$$y(k+1) = b_1 u(k) + b_2 u(k-1) - a_1 \hat{y}(k) - a_2 y(k-1) + \xi(k+1) - a_1 \xi(k) + c_1 \xi(k) + c_2 \xi(k-1)$$

$$y(k+1) = b_1 u(k) + b_2 u(k-1) - a_1 \hat{y}(k) - a_2 y(k-1) + \xi(k+1) + (c_1 - a_1) \xi(k) + c_2 \xi(k-1)$$

Se define la predicción para $i = 1$ como¹⁵:

$$y(k+1) = \hat{y}(k+1) + \xi(k+1) + (c_1 - a_1) \xi(k)$$

$$\hat{y}(k+1) = b_1 u(k) + b_2 u(k-1) - a_1 \hat{y}(k) - a_2 y(k-1) + c_2 \xi(k-1)$$

De la misma forma se puede obtener el modelo de predicción a dos instantes vista:

$$y(k+2) = b_1 u(k+1) + b_2 u(k) - a_1 y(k+1) - a_2 y(k) + \xi(k+2) + c_1 \xi(k+1) + c_2 \xi(k)$$

¹³Sin pérdida de generalidad se asume el modelo ARMAX, ya que si el proceso presenta una estructura BJ se tendría que $B_{ARMAX} = AB$, $A_{ARMAX} = AD$ y $C_{ARMAX} = CA$.

¹⁴ $\xi(k+\beta)$, $\forall \beta \geq 0$ es desconocido, luego no pueden incluirse en el modelo de predicción $\hat{y}(k)$ los términos asociados a valores presentes y futuros de la perturbación.

¹⁵Al igual que en el modelo de predicción con $i = 0$, se han de despreciar los términos asociados a valores presentes y futuros de la perturbación porque no son conocidos.

Desarrollando para eliminar los términos no disponibles ($y(k)$ y $y(k+1)$):

$$\begin{aligned}
 y(k+2) &= b_1u(k+1) + b_2u(k) - a_1(\hat{y}(k+1) + \xi(k+1) + (c_1 - a_1)\xi(k)) - a_2(\hat{y}(k) + \dots \\
 &\quad \dots \xi(k)) + \xi(k+2) + c_1\xi(k+1) + c_2\xi(k) \\
 y(k+2) &= b_1u(k+1) + b_2u(k) - a_1\hat{y}(k+1) - a_2\hat{y}(k|k-1) + \dots \\
 &\quad \dots \xi(k+2) + (c_1 - a_1)\xi(k+1) + (c_2 - a_2 - a_1(c_1 - a_1))\xi(k)
 \end{aligned}$$

Se define la predicción para $i = 2$ como:

$$\begin{aligned}
 y(k+2) &= \hat{y}(k+2) + \xi(k+2) + (c_1 - a_1)\xi(k+1) + (c_2 - a_2 - a_1(c_1 - a_1))\xi(k) \\
 \hat{y}(k+2) &= b_1u(k+1) + b_2u(k) - a_1\hat{y}(k+1) - a_2\hat{y}(k)
 \end{aligned}$$

Luego las tres expresiones obtenidas para las predicciones son:

$$\begin{aligned}
 \hat{y}(k) &= b_1u(k-1) + b_2u(k-2) - a_1y(k-1) - a_2y(k-2) + c_1\xi(k-1) + c_2\xi(k-2) \\
 \hat{y}(k+1) &= b_1u(k) + b_2u(k-1) - a_1\hat{y}(k) - a_2y(k-1) + c_2\xi(k-1) \\
 \hat{y}(k+2) &= b_1u(k+1) + b_2u(k) - a_1\hat{y}(k+1) - a_2\hat{y}(k)
 \end{aligned}$$

Se verifica la expresión de la ecuación 3.25 para el modelo de predicción a i instantes vista. De forma intuitiva se puede entender como la evaluación del modelo de predicción a un instante vista donde la matriz de regresión se forma utilizando toda la información conocida hasta k mas las estimaciones realizadas entre k y $k+i-1$, asumiendo $e(k+\beta) = 0, \forall \beta \geq 0$ y $e(k-\beta) = y(k-\beta) - \hat{y}(k-\beta) = \xi(k-\beta), \forall \beta > 0$. En el apartado 2.5 se introdujo la evaluación del modelo de predicción ventana que es un caso particular de la expresión 3.25 cuando el modelo es ARX.

Capítulo 4

Estimación de parámetros

4.1. Introducción

La estimación de parámetros puede definirse como la determinación de constantes o variables de un sistema, de forma óptima con respecto a un índice estadístico, y basadas en una serie de medidas tomadas sobre el proceso [Oll91]. Este capítulo está orientado a la estimación de los parámetros en modelos de estructura lineal respecto a sus parámetros donde se dispone de las matrices X e Y ¹ obtenidas a partir de experimentos realizados sobre el proceso. La elección de la familia de modelos $\mathbf{M}(\theta)$ ² condiciona el número de parámetros a determinar, siendo $\mathbf{m} = \mathbf{M}(\hat{\theta})$ el mejor modelo dentro de la familia $\mathbf{M}(\theta)$ para aproximar el comportamiento del proceso real. El objetivo es determinar dicha matriz $\hat{\theta}$, donde E es la matriz sobre la que se ha de evaluar el índice de coste que guía la identificación:

$$Y - \hat{Y} = E \quad ; \quad Y - X\hat{\theta} = E$$

La determinación de $\hat{\theta}$ se puede entender como un sistema de ecuaciones con $m \cdot n_s$ ecuaciones y $n_x \cdot n_s$ incógnitas. Si los datos experimentales procedieran de un proceso determinista lineal, la entrada aplicada durante el experimento fuera persistentemente excitante y $m = n_x$, se trataría de un sistema de ecuaciones compatible determinado con una única solución y $E = 0_{m \times n_s}$. No obstante, la realidad es no lineal y el número de muestras puede ser menor o mayor que el número de columnas en la matriz de regresión, luego no es posible obtener una solución exacta resolviendo un sistema de ecuaciones.

¹Como se ha comentado en el capítulo anterior, esto sólo se cumple para modelos ARX.

²Dentro de los modelos lineales, se ha de definir el orden de los numeradores y denominadores de las funciones de transferencia n_b y n_a .

Las técnicas de regresión lineal basadas en el error de predicción obtienen $\hat{\theta}$ mediante un proceso de optimización referido a un índice de coste que puede incluir alguna propiedad de la matriz E como su norma o la autocorrelación de cada una de sus columnas. Los indicadores de bondad introducidos en 2.5 son ejemplos de posibles índices de coste a minimizar con el objetivo de determinar $\hat{\theta}$. Distintos índices, así como la forma de realizar la optimización, dan lugar a las distintas técnicas de regresión explicadas en los siguientes apartados.

4.2. MLR

La regresión lineal múltiple obtiene $\hat{\theta}$ cuando $m \geq n_x$. El algoritmo de mínimos cuadrados (LS) es el más conocido para realizar regresiones lineales múltiples, éste proporciona la siguiente expresión explícita para obtener $\hat{\theta}$:

$$\hat{\theta} = (X'X)^{-1}X'Y \quad (4.1)$$

Obtenida a partir de la minimización del cuadrado del error de la estimación introducido en 2.4:

$$\min_{\theta} J = \min_{\theta} \|(Y - X\theta)\|^2 = \min_{\theta} (Y - X\theta)'(Y - X\theta)$$

Obtención del mínimo de la función cuadrática utilizando el gradiente:

$$\begin{aligned} \frac{\partial J}{\partial \theta} &= \frac{\partial (Y - X\theta)'(Y - X\theta)}{\partial \theta} \\ &= 2 \left(\frac{\partial (Y - X\theta)}{\partial \theta} \right)' (Y - X\theta) \\ &= -2X'(Y - X\theta) = 2(X'X\theta - X'Y) \end{aligned}$$

Se iguala la deriva a cero para encontrar el punto de inflexión:

$$\begin{aligned} \frac{\partial J}{\partial \theta} &= 2(X'X\hat{\theta} - X'Y) = 0 \\ \hat{\theta} &= (X'X)^{-1}X'Y \end{aligned}$$

La expresión obtenida para determinar $\hat{\theta}$, coincide con la fórmula de la matriz inversa generalizada X^* [Oll91]:

$$\begin{aligned} Y &= X\hat{\theta} \\ X^*Y &= X^*X\hat{\theta} = I\hat{\theta} \end{aligned}$$

$$\hat{\theta} = X^*Y = \text{pinv}^3(X)Y$$

Donde X^* se obtiene según:

$$\begin{aligned} \text{Si } m < n_x &\longrightarrow X^* = X'(XX')^{-1} \\ \text{Si } m > n_x &\longrightarrow X^* = (X'X)^{-1}X' \\ \text{Si } m = n_x &\longrightarrow X^* = X^{-1} \end{aligned}$$

Luego de forma análoga a la expresión obtenida por LS, se obtiene a partir de la inversa generalizada la expresión para determinar $\hat{\theta}$ cuando $m < n_x$. En dicho caso existen infinitas soluciones por tratarse de un sistema de ecuaciones compatible indeterminado, obteniendo como solución la matriz $\hat{\theta}$ de norma mínima dentro del conjunto de matrices θ que son solución al problema.

Para que la matriz $(X'X)$ sea invertible, es necesario que las señales contenidas en X sean persistentemente excitantes al menos de orden $n = \max(n_a, n_b)$ [Ise92], lo que se cumple si las matrices H_{nin} son definidas positivas, es decir, su determinante es mayor que cero para todas las entradas ($\forall nin \in [1 \dots n_e]$).

$$\begin{aligned} H_{nin} &= \{h_{ij} = \Phi_{u_{nin}u_{nin}}^N(i-j)\}, \quad i, j \in [1 \dots n] \\ \Phi_{u_{nin}u_{nin}}(\tau) &= \lim_{N \rightarrow \infty} \sum_{k=0}^{N-1} u_{nin}(k)u_{nin}(k+\tau) \end{aligned}$$

Para que $\hat{\theta}$ obtenido por mínimos cuadrados tienda al valor real θ^* conforme el número de muestras tiende a infinito, es necesario que el proceso a identificar presente una estructura ARX⁴ y $\xi(k)$ ha de cumplir las propiedades del ruido blanco 3.4.

Propiedades del algoritmo:

Insesgado: $E\{\hat{\theta}\} = \theta^*$ para un número de muestras finito m si ξ es ruido blanco:

$$Y = X\theta^* + \xi$$

Premultiplicando ambos lados de la igualdad por $(X'X)^{-1}X'$:

$$((X'X)^{-1}X')Y = ((X'X)^{-1}X')X\theta^* + (X'X)^{-1}X'\xi$$

Utilizando la ecuación 4.1:

$$\hat{\theta} = \theta^* + (X'X)^{-1}X'\xi \quad (4.2)$$

³Moore-Penrose pseudoinverse disponible en el paquete informático MATLAB[®].

⁴Introducida en 3.16.

$$\begin{aligned}
 E\{\hat{\theta}\} &= E\{\theta^* + (X'X)^{-1}X'\xi\} \\
 E\{\hat{\theta}\} &= E\{\theta^*\} + E\{(X'X)^{-1}X'\xi\} \\
 E\{\hat{\theta}\} &= \theta^* + m \cdot E\{(X'X)^{-1}X'\} \cdot E\{\xi\} \longleftrightarrow \text{corr}(\xi, X) = 0 \\
 E\{\hat{\theta}\} &= \theta^* \longleftrightarrow E\{\xi\} = 0
 \end{aligned}$$

Matriz de covarianza de los parámetros estimados:

$$\begin{aligned}
 P &= \text{cov}(\hat{\theta}) = E\{(\hat{\theta} - E\{\hat{\theta}\})(\hat{\theta} - E\{\hat{\theta}\})'\} \\
 P &= E\{(\hat{\theta} - \theta^*)(\hat{\theta} - \theta^*)'\}
 \end{aligned}$$

Retomando la ecuación 4.2:

$$\begin{aligned}
 \hat{\theta} - \theta^* &= (X'X)^{-1}X'\xi \\
 P &= E\{((X'X)^{-1}X'\xi)((X'X)^{-1}X'\xi)'\} \\
 P &= E\{(X'X)^{-1}X'\xi\xi'X(X'X)^{-1}\} \\
 P &= E\{(X'X)^{-1}X'X(X'X)^{-1}\}E\{\xi\xi'\} \longleftrightarrow \text{corr}(\xi, X) = 0 \\
 P &= E\{(X'X)^{-1}\}\sigma^2 \\
 P &= (X'X)^{-1}\sigma^2
 \end{aligned}$$

A continuación se dispone un ejemplo con la interpretación geométrica de la operación realizada por el algoritmo LS para un caso con dos variables de entrada y una de salida. Se puede interpretar $\hat{\theta}$ como la orientación⁵ del plano dibujado en la figura 4.1, y el error cuadrático de la aproximación como la suma del cuadrado de cada uno de los segmentos representados en línea discontinua entre el valor real de la salida (representado con o) y el valor estimado (representado con +). El algoritmo LS define la orientación del plano tomando el punto (0, 0) como punto de giro, de forma que el plano pase lo más cerca posible de todas las muestras.

⁵En este caso $\hat{\theta}$ es un vector columna con dos componentes. El $\text{arccotan}(\hat{\theta}(1, 1)) = 37^\circ$ representa el ángulo entre el eje x_1 y el plano, mientras que $\text{arccotan}(\hat{\theta}(2, 1)) = 56^\circ$ representa el ángulo entre el eje x_2 y el plano.

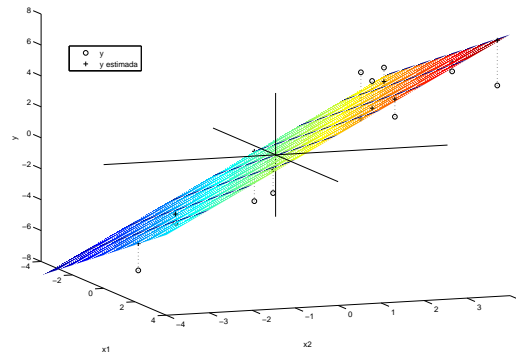


Figura 4.1: Aproximación por mínimos cuadrados

Las ventajas e inconvenientes de la técnica de regresión lineal múltiple LS son las siguientes:

Ventajas:

- Se dispone de una ecuación explícita para obtener $\hat{\theta}$.

Inconvenientes:

- Si existe colinealidad entre las columnas de la matriz X , esta no es de rango completo, y no se puede obtener la inversa de $X'X$, luego no se puede obtener $\hat{\theta}$ con la ecuación 4.1.
- Se obtiene una estimación sesgada si ξ no es ruido blanco, lo que ocurre si:
 - El proceso no presenta la estructura ARX.
 - Los datos han sido obtenidos en bucle cerrado.

A partir de LS se han desarrollado muchas otras técnicas intentado superar los problemas que esta presenta o añadir funcionalidades como poder ponderar la importancia de cada muestra en la estimación. En la tesis [Ram07], se introducen técnicas como mínimos cuadrados regularizado, mínimos cuadrados contraído o regresión ridge, que son algunas de las alternativas a LS para realizar MLR. No obstante, no se profundiza en dichas técnicas ya que la presente tesina está orientada al estudio de las técnicas basadas en la reducción de variables para eliminar el problema de la colinealidad.

4.3. PCR

El método de regresión sobre los componentes principales (PCR) consiste en dos fases. En primer lugar se reduce el número de dimensiones del espacio de entradas X utilizando la técnica PCA. Posteriormente, a partir del espacio reducido, se realiza la regresión lineal múltiple con el fin de obtener la matriz de parámetros del modelo $\hat{\theta}$. La reducción del espacio en X tiene como objetivo superar los problemas relacionados con la colinealidad de las columnas de X .

Este apartado está dividido en tres secciones, en la primera se explica la necesidad de la reducción a las componentes principales, en el segundo se introduce el uso de la técnica PCA para la reducción de la matriz X , y en el tercer apartado se obtiene la expresión del modelo PCR.

4.3.1. Reducción a las componentes principales

La matriz de regresión X en procesos con dinámica de orden elevado o para procesos MIMO dispone de varias columnas que contienen información del proceso. Muchas de las columnas utilizadas en la matriz X pueden estar correlacionadas, lo cual significa que con menos columnas se podría contener la misma información, o al menos la información más relevante. El hecho de utilizar demasiadas variables (columnas de X) genera los siguientes problemas:

LS no aplicable Como se ha comentado en el apartado 4.2, la colinealidad entre las columnas de X genera problemas a la hora de estimar los parámetros con LS, ya que $X'X$ no es invertible.

La maldición de la dimensionalidad La precisión de la estimación de los parámetros de un modelo está condicionada por la densidad de muestras en el espacio de entrada. Para espacios con muchas variables se necesita un número muy elevado de muestras cubriendo todo el espacio de entrada, en caso contrario la estimación estará basada en extrapolación en lugar de interpolación obteniendo un peor modelo [Yu,05].

Aumenta MSEP A partir de la varianza del error se obtiene la siguiente expresión para el indicador MSEP⁶:

⁶El operador $*$ implementa el producto *Hadamard* que puede expresarse de forma general como: $c = a * b$, donde $c(i, j) = a(i, j)b(i, j)$.

$$\begin{aligned}
 \text{Var}(e) &= E\{(e - E\{e\}) \cdot (e - E\{e\})\} \\
 &= E\{e \cdot e + E\{e\}^2 - 2eE\{e\}\} \\
 &= E\{e \cdot e\} + E\{E\{e\}^2\} - E\{2eE\{e\}\} \\
 &= E\{e \cdot e\} + E\{e\}^2 - 2E\{e\}E\{e\} \\
 &= E\{e \cdot e\} + E\{e\}^2 - 2E\{e\}^2 \\
 &= E\{e \cdot e\} - E\{e\}^2
 \end{aligned}$$

$$MSEP = E\{e \cdot e\} = E\{e\}^2 + \text{Var}(e) \quad (4.3)$$

Se observa en la ecuación 4.3 que el error cuadrático medio depende del cuadrado del error medio y de la varianza del error. Aumentar n_x generalmente disminuye el cuadrado del error medio, pero la varianza del error incrementa a partir de un cierto número de variables de entrada ya que la incertidumbre de cada uno de los parámetros aumenta la variabilidad de la predicción [Hös88].

La reducción de variables consiste en obtener un nuevo espacio de datos (con menos columnas) que contenga la información más importante del espacio original.

4.3.2. PCA

La descomposición en componentes principales (PCA) es un método para escribir una matriz X de rango ncp como la suma de ncp matrices de rango 1 [Gel86].

$$X = t_1 p'_1 + t_2 p'_2 + \dots + t_{ncp} p'_{ncp} + \mathcal{E} = \sum_{i=1}^{ncp} t_i p'_i + \mathcal{E}$$

$$X = TP' + \mathcal{E} \quad (4.4)$$

En el espacio original de X se dispone de m muestras, filas de X . Se define T como la matriz de puntuaciones de X , la cual contiene la posición de las m muestras en el nuevo espacio con ncp dimensiones. La matriz de pesos P es la que define la orientación de los componentes principales, siendo el vector p_i el que define la orientación del componente principal i en el espacio n_x dimensional.

Si la matrix X tiene n_x columnas y $ncp = n_x$, la matriz de residuos $\mathcal{E} = 0_{m \times n_e}$ y T es una matriz $m \times n_x$, luego no se realiza la reducción del espacio ya que tanto X como T son ncp dimensionales. Si es necesario reducir el número de dimensiones, se escoge $ncp < n_x$, luego \mathcal{E} contendrá la información de X no incluida en el nuevo espacio de entrada T .

En la figura 4.2 se observa la obtención de los componentes principales de X para un caso concreto donde $n_x = 3$. En trazado continuo se observa el sistema de coordenadas original, con el símbolo $*$ se representan las m muestras de la matriz X . En trazado no continuo los distintos componentes principales obtenidos y sobre estos, con el símbolo $+$ se representan las proyecciones de las muestras de X sobre los componentes principales, que son las puntuaciones T . Se puede entender el sistema de coordenadas definido por los componentes principales como una rotación del sistema de coordenadas original donde P determina la dirección de los nuevos vectores de coordenadas en el espacio inicial. La clave de esta rotación se observa en dicha figura, ya que las puntuaciones del primer componente principal ⁷ son las que más variabilidad de X representan siendo las proyecciones de las muestras sobre este componente principal las más alejadas del origen. Mientras que las del tercer componente principal a penas contienen información estando todas las puntuaciones muy cerca del origen.

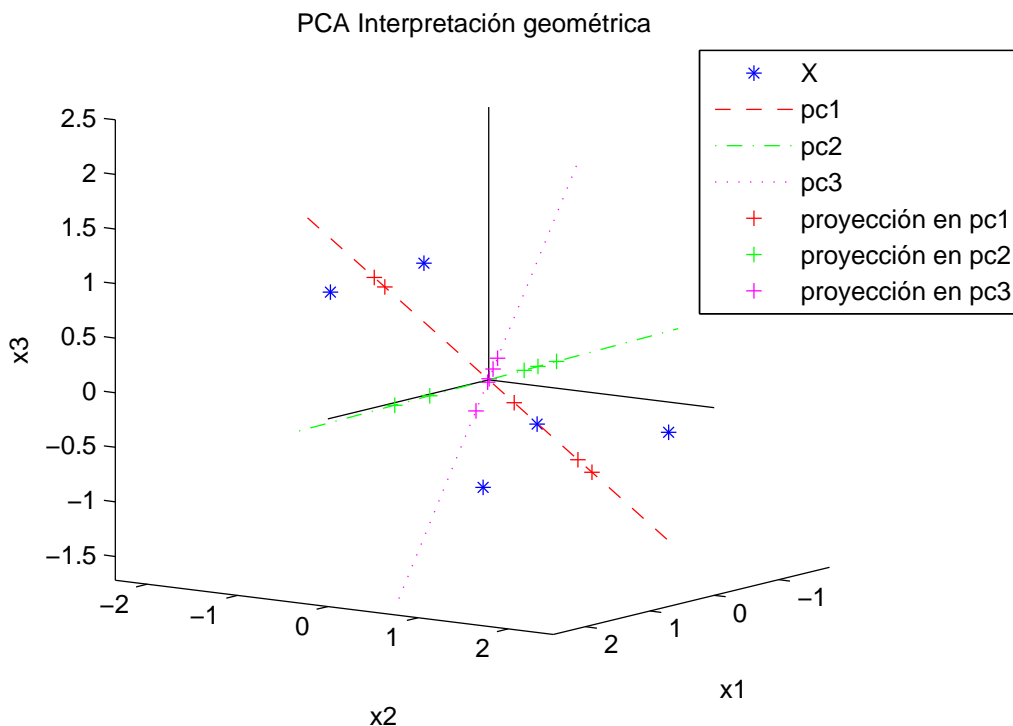


Figura 4.2: *Componentes Principales*

Si se realiza una reducción del número de dimensiones en el espacio de entrada, es decir, se prescinde del tercer componente $n_{cp} = 2$, se obtiene el resultado de la figura

⁷proyecciones de las muestras de X sobre el componente principal 1.

4.3, donde se observa la aproximación de las muestras de X dibujadas con círculos. La matriz \mathcal{E} en dicho caso contiene la diferencia entre las muestras reales (*) y las estimadas (círculos).

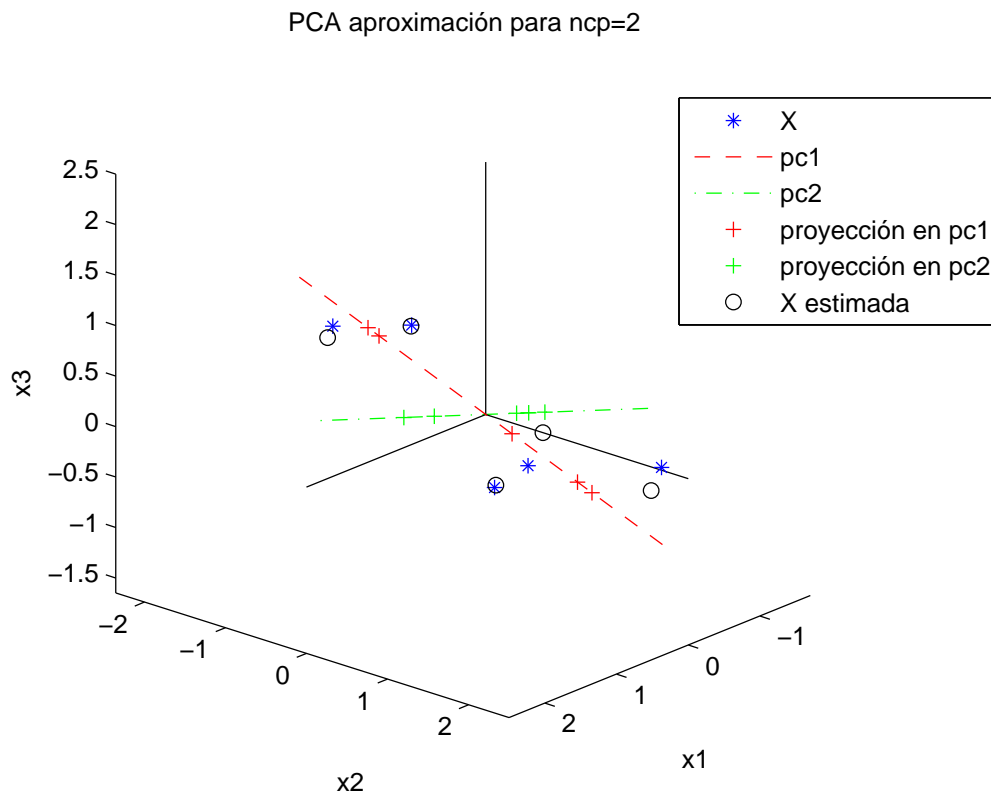


Figura 4.3: Aproximación de X con 2 componentes principales

Para la obtención de las matrices T y P se utiliza el algoritmo NIPALS [Gel86]. Dicho algoritmo obtiene los componentes principales progresivamente, orientando el primer componente principal en la dirección de máxima varianza de X , una vez obtenido se resta a la matriz X la información representada por este componente principal y se vuelve a empezar utilizando $\mathcal{E} = X - t_1 p_1'$ en lugar de X , ya que esta contiene la información restante. Se procede de esta forma sucesivamente hasta obtener el componente n_{cp} .

Algoritmo NIPALS:

1. Inicializar la matriz de residuos $\mathcal{E} = X$
2. $h = 1$

Obtención del componente principal h

3. Seleccionar el vector columna de \mathcal{E} con mayor norma y llamarlo t_h

4. Calcular $p'_h = \frac{t'_h \mathcal{E}}{t'_h t_h}$
5. Normalizar p_h : $p_h = \frac{p_h}{\|p_h\|}$
6. Calcular $t_h = \mathcal{E} p_h$
7. Comparar t_h con el utilizado en el punto 4, si son iguales se pasa al punto 8, en caso contrario se vuelve al punto 4 con el nuevo t_h .
8. Deflación de los residuos $\mathcal{E} = \mathcal{E} - t_h p'_h$
9. se almacenan t_h y p_h como las puntuaciones y pesos del componente h . $T = [T \ t_h]$ y $P = [P \ p_h]$.
10. Se incrementa h y si es mayor a n_{cp} se acaba, en caso contrario se vuelve al punto 3. Existen varios criterios para determinar n_{cp} , por ejemplo se puede definir como el n_{cp} a partir del cual $\|\mathcal{E}\|$ queda por debajo de un umbral definido, o a partir del cual los siguientes componentes principales no reducen dicha norma de forma significativa.

Finalmente:

$$X = t_1 p'_1 + t_2 p'_2 + \dots + t_{n_{cp}} p'_{n_{cp}} + \mathcal{E}$$

Para entender de forma gráfica las operaciones realizadas en el bucle 4-7 para la obtención de un componente principal, se aborda un ejemplo sencillo con la siguiente matriz X ⁸:

$$X = \begin{bmatrix} x_1 & x_2 \\ 0,11 & 1,25 \\ 1,27 & -0,5 \end{bmatrix} = t_1 p'_1 + t_2 p'_2 = \begin{bmatrix} t_{11} \\ t_{12} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \end{bmatrix} + \begin{bmatrix} t_{21} \\ t_{22} \end{bmatrix} \begin{bmatrix} p_{21} & p_{22} \end{bmatrix}$$

En la figura 4.4 se representan las 4 iteraciones necesarias para que t_1 y p_1 converjan. La línea continua representa el primer componente principal cuya orientación viene definida por p_1 , en la figura se imprime la pendiente de la recta $m = \frac{p_{12}}{p_{11}}$. Las dos muestras de X se representan mediante un círculo y un cuadrado. Los símbolos + son las puntuaciones t_1 sobre el componente principal p_1 . Los símbolos * son las proyecciones de las dos muestras de X sobre el componente principal.

El algoritmo empieza con un valor inicial para las puntuaciones t_1 que no es el real, como se puede comprobar ya que los puntos + no coinciden con los puntos * en la iteración 1. A partir de dicho t_1 inicial se obtiene, utilizando la fórmula de mínimos cuadrados 4.1 y

⁸X no ha sido centrada y escalada para una mayor claridad del ejemplo.

la fórmula de la descomposición PCA 4.4, la orientación del primer componente principal determinada por p_1 .

$$\hat{Y} = X\hat{\theta} \stackrel{LS}{\Rightarrow} \hat{\theta} = (X'X)^{-1}X'Y \quad (4.5)$$

Análogamente se puede obtener p'_1 como:

$$X = \mathcal{E} = t_1 p'_1 \stackrel{LS}{\Rightarrow} p'_1 = (t'_1 t_1)^{-1} t'_1 \mathcal{E}$$

Geoméricamente se puede entender el resultado de la fórmula LS como la orientación del primer componente principal que minimice la suma del cuadrado de los segmentos discontinuos de la figura 4.4. Una vez obtenido el primer componente principal, que para la primera iteración toma la orientación:

$$m = \frac{p_{12}}{p_{11}} = -3,7$$

Se obtienen las proyecciones de los puntos de la matriz X sobre dicho componente principal (puntuaciones t_1 representados con *) utilizando de nuevo la fórmula de mínimos cuadrados 4.5:

$$\mathcal{E} = t_1 p'_1 \quad \parallel \quad \mathcal{E}' = p_1 t'_1 \quad \stackrel{LS}{\Rightarrow} \quad t'_1 = (p'_1 p_1)^{-1} p'_1 \mathcal{E}' \quad \parallel \quad t_1 = \mathcal{E} p_1 (p'_1 p_1)^{-1}$$

Debido a que P es una matriz de rotación que cumple $P'P = I$, se puede obtener t_1 como:

$$\mathcal{E} = t_1 p'_1 \quad ; \quad \mathcal{E} p_1 = t_1 p'_1 p_1 = t_1 \quad ; \quad t_1 = \mathcal{E} p_1$$

Se puede comprobar que las dos expresiones obtenidas son equivalentes debido a que $p'_1 p_1 = 1$.

El nuevo vector t_1 se utiliza como vector origen en la segunda iteración. Se observa en la figura 4.4 que la diferencia entre el t_1 inicial (+) y final (*) es menor conforme converge el algoritmo, considerándose que converge en la iteración 4. Una vez obtenidos p_1 y q_1 , se sigue con el punto 8 del algoritmo.

La descomposición en componentes principales proporciona finalmente las proyecciones de las muestras de X (puntuaciones T) sobre un nuevo espacio definido por P . La ventaja de esta transformación es que se pueden eliminar los componentes menos representativos, y las puntuaciones obtenidas son ortogonales, luego se puede realizar una regresión lineal a partir de estas porque $T'T$ sí es invertible.

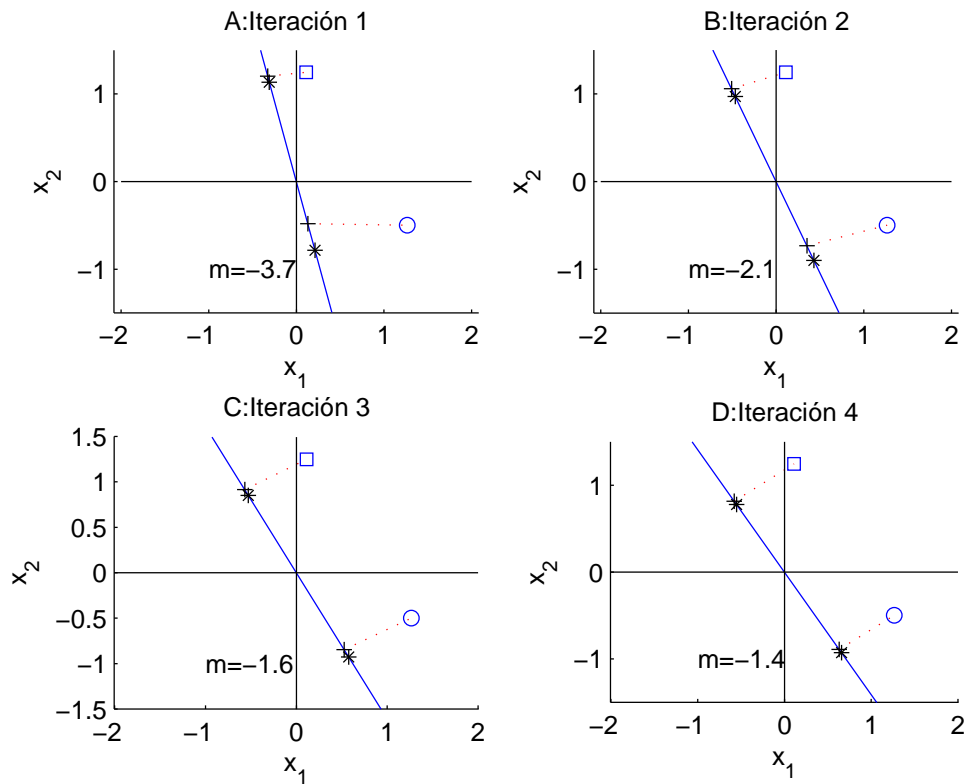


Figura 4.4: Obtención de un componente principal

4.3.3. Regresión sobre los componentes principales

Una vez obtenidos los componentes principales, se puede expresar la matriz de salidas Y en función de las puntuaciones de X en el espacio de componentes principales.

$$Y = TB + \mathcal{F} = \hat{Y} + \mathcal{F}$$

Se obtiene B como el resultado de la regresión lineal múltiple mediante LS:

$$Y = TB + \mathcal{F} \xrightarrow{LS} B = (T'T)^{-1}T'Y \quad (4.6)$$

$$\hat{Y} = TB$$

Para disponer de un modelo que relacione Y con X es necesario obtener una expresión que relacione T con X , para ello se utiliza la matriz de rotación P :

$$X = TP' + \mathcal{E} \quad ; \quad XP = TP'P + \mathcal{E}P$$

Como P es una matriz de rotación que cumple $P'P = I$ y es ortogonal a los residuos $\mathcal{E}P = 0$:

$$T = XP$$

Quedando el modelo final como:

$$\hat{Y} = XPB$$

Retomando la ecuación general de los modelos lineales 3.2, se puede expresar el modelo como:

$$\hat{Y} = X\hat{\theta}$$

Donde:

$$\hat{\theta} = PB = P(T'T)^{-1}T'Y$$

Ventajas:

- Supera el problema de matriz X no bien condicionada transformándola en la matriz T bien condicionada para obtener B , ya que T es ortogonal, luego $T'T$ es invertible y es posible utilizar la ecuación 4.1.
- Puede obtener mejores modelos que LS con matrices X con ruido si se utiliza un número de componentes principales adecuado.
- Proporciona información acerca de la importancia de las variables de entrada del proceso, proporcionando información útil para el rediseño del experimento de identificación si fuera necesario.

Inconvenientes:

- La aproximación de Y se hace tras la descomposición en componentes principales de X , luego nada asegura que información útil para el carácter predictivo del modelo quede en componentes principales de X que se desprecian [Gel86].
- El resultado final está condicionado por el número de componentes principales utilizado, luego es importante una selección óptima de n_{cp} .

4.4. PLS

El método de mínimos cuadrados parciales (PLS), al igual que PCR, realiza una reducción del espacio de entrada a sus componentes principales previa a la regresión lineal. En este caso los componentes principales no son aquellos que maximizan la varianza de X , sino los que maximizan la covarianza entre las puntuaciones de los espacios de entrada y salida. La regresión se realiza para cada componente antes de obtener los residuos a partir de los cuales se determinará el siguiente componente [Wol01, Har01, Hös88, Gel86].

La descomposición de X e Y queda expresada por:

$$\begin{array}{lcl} X = TP' & \text{si } n_{cp} < n_x & X = TP' + \mathcal{E} \\ Y = UQ' & \implies & Y = UQ' + \mathcal{F} \end{array} \quad (4.7)$$

Donde: T contiene las puntuaciones de cada uno de los componentes principales obtenidos para el espacio de entrada X . P es la matriz de pesos que define la orientación de los componentes principales en el espacio de n_x dimensiones. \mathcal{E} son los residuos de X , es decir, la información no incluida en los componentes utilizados, si se asume $n_{cp} = n_x$ dicha matriz de residuos $\mathcal{E} = 0_{m \times n_s}$. Análogamente pero en el espacio de salida Y , U son las puntuaciones, Q los pesos y \mathcal{F} los residuos⁹.

Para la obtención de los componentes principales se procede de forma progresiva obteniendo un componente cada vez al igual que en PCA, pero en este caso se intercambian los vectores de las puntuaciones entre los espacios de entrada y salida en el algoritmo NIPALS explicado en 4.3.2, maximizando de esta forma la covarianza entre las puntuaciones de ambos espacios. Para la descomposición se utiliza una matriz de pesos alternativa W con el objetivo de obtener T con columnas ortogonales¹⁰.

A continuación se listan los pasos a seguir para realizar la descomposición de X e Y según el algoritmo PLS.

Algoritmo PLS:

1. $h = 1$
2. Inicializar las matrices de residuos $\mathcal{E}_h = X$; $\mathcal{F}_h = Y$

⁹Aunque $n_{cp} = n_s$ no se puede asegurar ($\mathcal{F} = 0_{m \times n_s}$) debido a que las columnas de U no son ortogonales.

¹⁰Al intercambiar los pesos en el algoritmo NIPALS entre los espacios de entrada y salida, las columnas de T dejan de ser ortogonales, esto se puede evitar utilizando W como matriz de pesos y P para la reducción de \mathcal{E} tras la obtención de cada componente principal. Alternativamente se puede asumir $W = P$ con la salvedad de que las columnas de T no serán ortogonales [Smi04].

Obtención del componente principal h

3. Seleccionar el vector columna de \mathcal{F}_h con mayor norma y llamarlo u_h
4. Calcular $w'_h = \frac{u'_h \mathcal{E}_h}{u'_h u_h}$
5. Normalizar w_h : $w_h = \frac{w_h}{\|w_h\|}$
6. Calcular $t_h = \mathcal{E}_h w_h$
7. Calcular $q'_h = \frac{\mathcal{F}'_h t_h}{t'_h t_h}$
8. Normalizar q_h : $q_h = \frac{q_h}{\|q_h\|}$
9. Calcular $u_h = \frac{\mathcal{F}_h q_h}{q'_h q_h}$
10. Comparar u_h con el utilizado en el punto 4, si son iguales se pasa al punto 11, en caso contrario se vuelve al punto 4 con el nuevo u_h .
11. Calcular $p_h = \frac{\mathcal{E}'_h t_h}{t'_h t_h}$
12. Regresión lineal: $b_h = \frac{u'_h t_h}{t'_h t_h}$
13. Deflación de los residuos $\mathcal{E}_h = \mathcal{E}_h - t_h p'_h$; $\mathcal{F}_h = \mathcal{F}_h - b_h t_h q'_h$
14. Se almacenan las puntuaciones y pesos del componente h .
 $T = [T \ t_h]$, $P = [P \ p_h]$, $U = [U \ u_h]$, $Q = [Q \ q_h]$, $B(h, h) = b_h$.
15. Se incrementa h y si es mayor a ncp se acaba, en caso contrario se vuelve al punto 3 con $\mathcal{E}_h = \mathcal{E}_{h-1}$ y $\mathcal{F}_h = \mathcal{F}_{h-1}$. En este punto se suele implementar una técnica de validación interna para determinar el número de componentes principales que maximiza la predictividad del modelo.

Con el fin de entender el algoritmo desde el punto de vista geométrico se define un ejemplo sencillo con dos muestras y dos dimensiones tanto en el espacio de X como en el de Y utilizando las siguientes matrices:

$$X = \begin{bmatrix} x_1 & x_2 \\ -0,2 & -0,5 \\ 0,5 & -0,1 \end{bmatrix} \quad Y = \begin{bmatrix} y_1 & y_2 \\ -0,1 & -0,2 \\ 0,4 & 0,1 \end{bmatrix} \quad (4.8)$$

En la figura 4.5 se representan las 3 primeras iteraciones del algoritmo PLS para la obtención del primer componente principal de las matrices X e Y definidas en 4.8. La figura contiene 6 gráficas A-F, siendo A-B, C-D y E-F las gráficas de cada una de las 3 iteraciones. La línea continua representa el primer componente principal en el

espacio X (gráficas A, C y E) e Y (gráficas B, D y F), se muestra la pendiente de dicho componente como m en cada gráfica. Las dos muestras de X e Y se representan mediante un círculo y un cuadrado, asumiendo que el círculo de Y está asociado al círculo de X y lo mismo para el cuadrado. Los símbolos $+$ son las puntuaciones sobre los componentes principales del espacio opuesto, por ejemplo, los símbolos $+$ de la gráfica A son las proyecciones de las muestras de Y sobre q_1 . Los símbolos $*$ son las puntuaciones sobre los componentes principales del mismo espacio, por ejemplo, los símbolos $*$ de la gráfica A son las proyecciones de las muestras de X sobre w_1 .

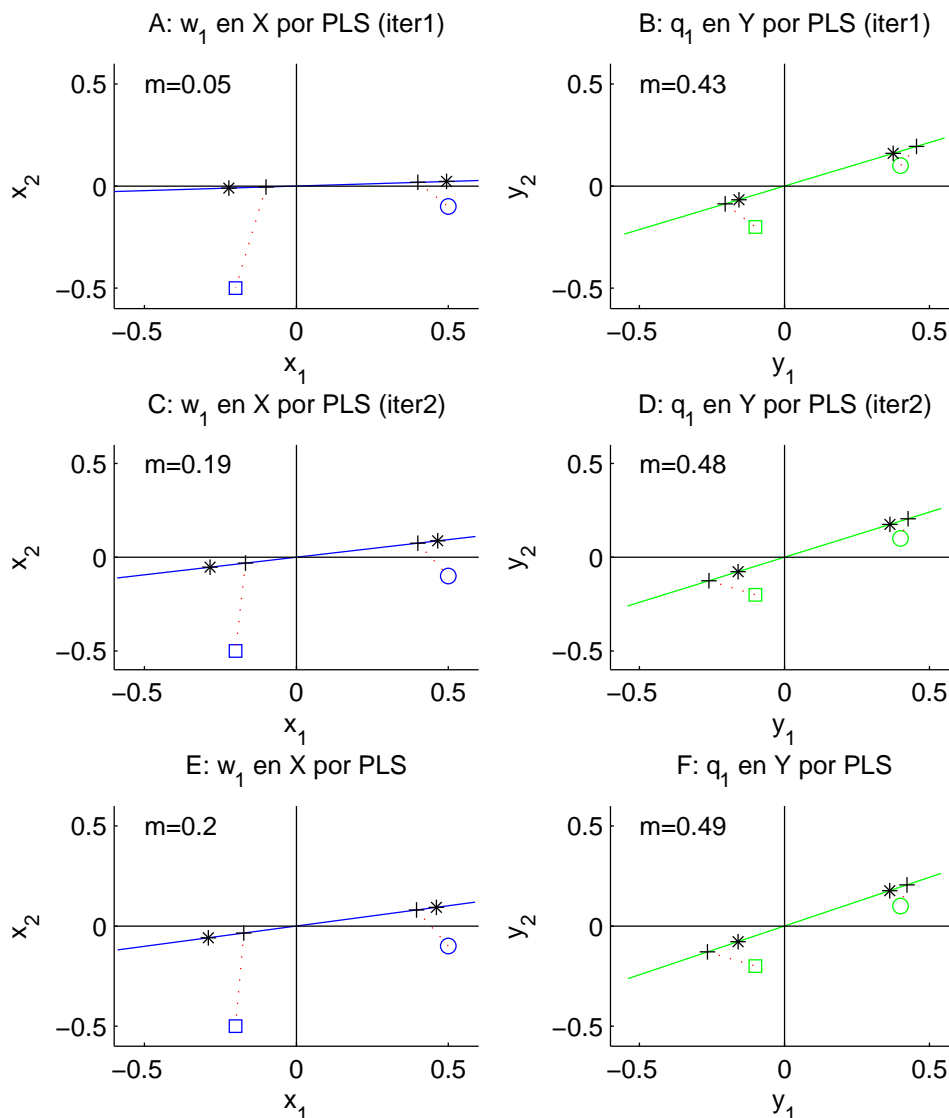


Figura 4.5: Primer componente algoritmo PLS

La forma de obtener los pesos y puntuaciones en cada iteración es análoga a la explicada en el algoritmo NIPALS 40, pero se intercambian las puntuaciones entre los espacios de entrada y salida con la finalidad de orientar w_1 y q_1 de forma que se maximice la covarianza entre las puntuaciones t_1 y u_1 . Gráficamente se puede entender como la orientación del primer componente principal de X e Y que minimiza los segmentos discontinuos de la figura 4.5, acercando las proyecciones de una misma muestra en ambos subespacios. De esta forma se puede obtener por LS el parámetro b_1 que minimiza los residuos en: $u_1 = t_1 b_1 + \text{residuos}$, como muestra la gráfica A de la figura 4.6.

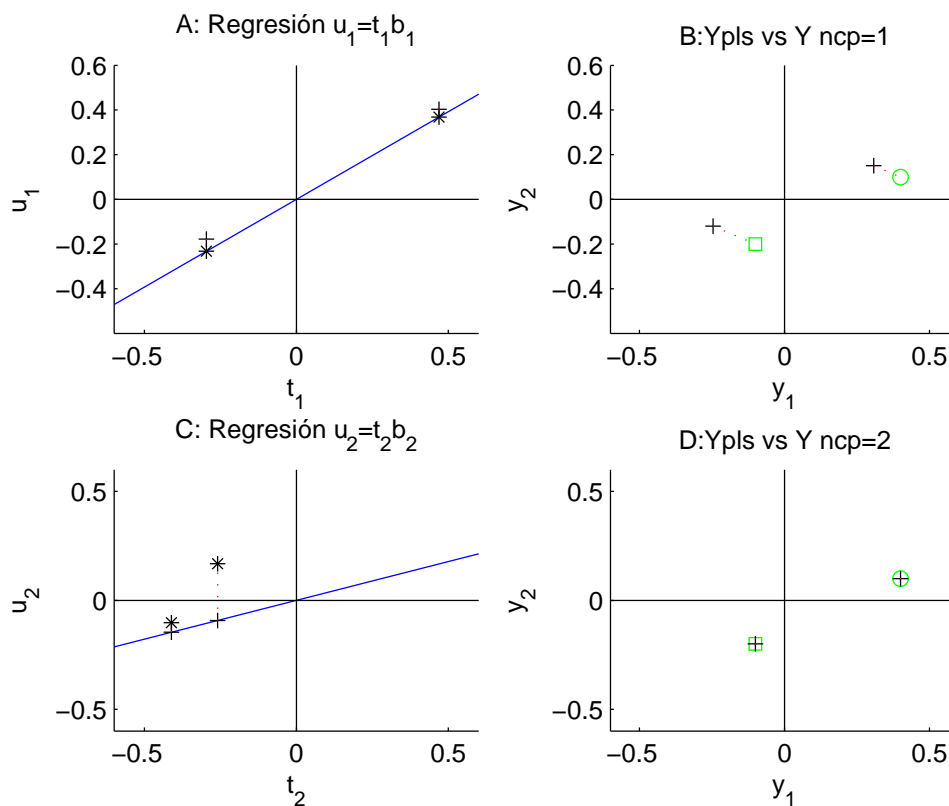


Figura 4.6: Ejemplo PLS $ncp=2$

Tras la obtención de cada componente principal se realiza la regresión lineal b_h entre las puntuaciones de los espacios de entrada t_h y salida u_h . De esta forma la relación entre los componentes principales queda definida por la matriz diagonal \mathcal{B}^{11} :

$$U = T\mathcal{B} + \text{residuos} \quad \implies \quad \hat{U} = T\mathcal{B}$$

¹¹Obtenida como $\mathcal{B}(h, h) = b_h \forall h \in [1, \dots, ncp]$

Con el objetivo de expresar el modelo conforme a la ecuación 3.2 se procede de la siguiente forma a partir de las ecuaciones anteriores:

$$\hat{Y} = \hat{U}Q' = TBQ'$$

Es necesario obtener T a partir de X como se hizo en el apartado PCR si se desea disponer de un modelo capaz de estimar \hat{Y} para un X dado. No obstante, debido a la utilización de W para la rotación y P para la reducción de los residuos en el espacio de entrada para cada componente, la expresión obtenida en este caso dependerá también de W , $T = f(X, P, W)$. Para determinar f se puede proceder de la siguiente forma:

$$X = TP' + \mathcal{E} \quad ; \quad X - \mathcal{E} = TP' \quad ; \quad (X - \mathcal{E})pinv(P') = T$$

No obstante, ante un nuevo conjunto de datos \mathcal{E} no es conocida luego es necesario definir una expresión para obtener $X - \mathcal{E}$ a partir de X sin conocer \mathcal{E} :

$$\begin{aligned} X &= TP' + \mathcal{E} \\ X - \mathcal{E} &= TP' = XF_{ncp} \end{aligned}$$

Se define la matriz F_{ncp} , como el filtro que se ha de aplicar a X para obtener TP' . Con el objetivo de determinar F_1 , asumiendo 1 componente principal, se desarrolla la descomposición de X para un componente principal:

$$\begin{aligned} X &= t_1p_1' + \mathcal{E}_1 = X_1 + \mathcal{E}_1 \\ X_1 &= t_1p_1' \\ t_1 &= X \frac{w_1}{w_1'w_1} \\ X_1 &= X \frac{w_1}{w_1'w_1} p_1' = X f_1 \\ X &= X f_1 + \mathcal{E}_1 \\ X - \mathcal{E}_1 &= X f_1 \\ F_1 &= f_1 \end{aligned}$$

En el caso de utilizar dos componentes principales:

$$\begin{aligned} X &= t_1p_1' + t_2p_2' + \mathcal{E}_2 = X_2 + \mathcal{E}_2 \\ X_1 &= XF_1 \end{aligned}$$

$$X_2 = X_1 + (X - X_1) \frac{w_2}{w_2' w_2} p_2' = X_1 + (X - X_1) f_2$$

$$X_2 = X F_1 + (X - X F_1) f_2 = X(F_1 + f_2 - F_1 f_2)$$

$$X - \mathcal{E}_2 = X(F_1 + f_2 - F_1 f_2)$$

$$F_2 = F_1 + f_2 - F_1 f_2$$

En el caso de utilizar tres componentes principales:

$$X = t_1 p_1' + t_2 p_2' + t_3 p_3' + \mathcal{E}_2 = X_3 + \mathcal{E}_3$$

$$X_1 = X F_1; \quad X_2 = X F_2$$

$$X_3 = X_2 + (X - X_2) \frac{w_3}{w_3' w_3} p_3' = X_2 + (X - X_2) f_3$$

$$X_3 = X F_2 + (X - X F_2) f_3 = X(F_2 + f_3 - F_2 f_3)$$

$$X - \mathcal{E}_3 = X(F_2 + f_3 - F_2 f_3)$$

$$F_3 = F_2 + f_3 - F_2 f_3$$

De lo anterior se puede deducir la expresión para cualquier número de componentes principales F_{ncp} :

$$F_{ncp} = F_{ncp-1} + f_{ncp} - F_{ncp-1} f_{ncp}$$

Donde:

$$f_{ncp} = \frac{w_{ncp}}{w_{ncp}' w_{ncp}} p_{ncp}'$$

$$F_1 = f_1$$

Luego la expresión $T = f(X, P, W)$ queda definida como:

$$T = X F_{ncp} p_{inv}(P')$$

Existe otra expresión equivalente pero más compacta para obtener $T = f(X, P, W)$ según [Har01]:

$$T = XW(P'W)^{-1} \tag{4.9}$$

Se puede observar que en el caso PCR o PLS con vectores T no ortogonales, $P = W$ y $P'P = I$, luego $(P'W)^{-1} = (P'P)^{-1} = I$, quedando la anterior expresión como la obtenida en el apartado PCR:

$$T = XP$$

Finalmente se puede definir el modelo como:

$$\hat{Y} = T\mathcal{B}Q' = XW(P'W)^{-1}\mathcal{B}Q'$$

$$\hat{Y} = X\hat{\theta}$$

Donde:

$$\hat{\theta} = W(P'W)^{-1}\mathcal{B}Q'$$

Aplicando PLS al ejemplo de la ecuación 4.8, se obtiene el resultado mostrado en la figura 4.6. En la gráfica B se observa con símbolos + la estimación con $\hat{\theta}$ obtenida utilizando un sólo componente principal y en la gráfica D utilizando dos componentes principales, en dicho caso la solución es exacta.

Ventajas:

- La ortogonalidad de los componentes principales sobre los que se realiza la regresión evita problemas de colinealidad, de forma que siempre es posible obtener la matriz $\hat{\theta}$ a partir de X e Y .
- Orienta los componentes principales maximizando la covarianza entre las puntuaciones de X e Y , luego se obtiene un modelo que relaciona Y con X .
- A partir de la matriz de residuos \mathcal{E} se puede determinar cuales son las variables menos relevantes para el modelo, ya que las columnas asociadas a dichas variables presentarán un alto valor SOS (*Sum of Squares*). Del mismo modo se detectan los *outliers* como las filas de \mathcal{E} con mayor norma [Gel86].

Inconvenientes:

- El resultado final está condicionado por el número de componentes principales utilizado, luego es importante una selección óptima de *n_{cp}*.

4.5. KPLS

Los dos métodos de regresión explicados en los apartados anteriores se diferencian principalmente en la orientación de los componentes principales. PCR busca maximizar la varianza de X explicada en cada componente y PLS maximizar la covarianza entre las puntuaciones de entrada y salida. No obstante, ninguna de las dos soluciones es mejor que la otra par todos los casos, aunque de forma general PLS obtiene modelos más predictivos.

Existe un técnica llamada *Continuum regression* que proporciona una solución intermedia entre LS, PLS y PCR [Ser05], donde w_h se obtiene resolviendo un problema de maximización con restricciones, que para el caso de una salida se puede expresar como:

$$w_h = \underset{a}{\operatorname{argmax}} \{ \operatorname{Cov}(Xa, y)^2 \operatorname{Var}(Xa)^{\left(\frac{\delta}{1-\delta} - 1\right)} \}$$

Sujeto a:

$$\|w_h\| = 1 \quad ; \quad \operatorname{Cov}(Xw_h, Xw_j) = 0 \quad \forall j < h$$

En función de δ se obtienen los distintos casos particulares donde $\delta = 0$ es la solución LS¹², $\delta = 0,5$ es la solución PLS y $\delta = 1$ es la solución PCR [Ser05].

La solución propuesta en este apartado KPLS, se puede ver como un caso particular de *Continuum regression* que obtiene una solución intermedia entre PLS y PCR en la orientación de w_h . Tanto en el algoritmo PLS como en la descomposición PCA se utiliza la siguiente ecuación para determinar la orientación del componente principal h :

$$w'_h = \frac{\alpha'_h \mathcal{E}}{\alpha'_h \alpha_h} \quad (4.10)$$

Donde: w_h es el vector de pesos para el componente h ¹³, α_h es el vector de puntuaciones que en cada método toma un valor distinto:

PCR	$\alpha_h = t_h$
PLS	$\alpha_h = u_h$

Luego si se utilizan las puntuaciones del espacio de entrada t_h en la obtención de w_h se maximiza la varianza de t_h , mientras que si se utilizan las puntuaciones del espacio

¹²Se maximiza la correlación entre Xa e y .

¹³En PCR solamente se utiliza P , no obstante se puede comprobar que si se añadiera P en el algoritmo de forma análoga a como se añade en PLS, resultaría que $P = W$, luego de forma general en el presente apartado se utiliza W .

de salida se maximiza $Cov(t_h, u_h)$. En el algoritmo KPLS se utilizan tanto u_h como t_h en la orientación de w_h , para ello se define una variable k que determina la importancia de cada uno de los vectores anteriores¹⁴, la ecuación 4.10 queda de la siguiente forma:

$$w'_h = \frac{\alpha'_h \underline{\mathcal{E}}}{\alpha'_h \alpha_h}$$

Donde:

$$\alpha_h = \begin{bmatrix} kt_h \\ (1-k)u_h \end{bmatrix}$$

$$\underline{\mathcal{E}} = \begin{bmatrix} k\mathcal{E} \\ (1-k)\mathcal{E} \end{bmatrix}$$

En las ecuaciones anteriores si $k = 0$ la orientación de w_h es aquella que maximiza la covarianza entre las puntuaciones de entrada y salida, si vale 1 maximiza la varianza de \mathcal{E} explicada, y para valores intermedios se obtienen las soluciones KPLS. Para determinar k de forma robusta se suelen utilizar indicadores obtenidos por validación cruzada.

Algoritmo KPLS:

1. $h = 1$
2. Inicializar las matrices de residuos $\mathcal{E}_h = X$; $\mathcal{F}_h = Y$
Obtención del componente principal h
3. Seleccionar el vector columna de \mathcal{F}_h con mayor norma y llamarlo u_h
4. Seleccionar el vector columna de \mathcal{E}_h con mayor norma y llamarlo t_h
5. Determinar $\alpha_h = [kt_h; (1-k)u_h]$ y $\underline{\mathcal{E}}_h = [k\mathcal{E}_h; (1-k)\mathcal{E}_h]$
6. Calcular $w'_h = \frac{\alpha'_h \underline{\mathcal{E}}_h}{\alpha'_h \alpha_h}$
7. Normalizar w_h : $w_h = \frac{w_h}{\|w_h\|}$
8. Calcular $t_h = \mathcal{E}_h w_h$
9. Calcular $q'_h = \frac{\mathcal{F}'_h t_h}{t'_h t_h}$
10. Normalizar q_h : $q_h = \frac{q_h}{\|q_h\|}$

¹⁴La inclusión de dicha variable k se puede interpretar como la utilización de la ecuación de mínimos cuadrados ponderado.

11. Calcular $u_h = \frac{\mathcal{F}_h q_h}{q_h' q_h}$
12. Comparar u_h con el utilizado en el punto 5, si son iguales se pasa al punto 13, en caso contrario se vuelve al punto 5 con el nuevo u_h .
13. Calcular $p_h = \frac{\mathcal{E}_h' t_h}{t_h' t_h}$
14. Regresión lineal: $b_h = \frac{u_h' t_h}{t_h' t_h}$
15. Deflación de los residuos $\mathcal{E}_h = \mathcal{E}_h - t_h p_h'$; $\mathcal{F}_h = \mathcal{F}_h - b_h t_h q_h'$
16. Se almacenan las puntuaciones y pesos del componente h .
 $T = [T \ t_h]$, $P = [P \ p_h]$, $U = [U \ u_h]$, $Q = [Q \ q_h]$, $B(h, h) = b_h$.
17. Se incrementa h y si es mayor a ncp se acaba, en caso contrario se vuelve al punto 3 con $\mathcal{E}_h = \mathcal{E}_{h-1}$ y $\mathcal{F}_h = \mathcal{F}_{h-1}$. En este punto se suele implementar una técnica de validación interna para determinar el número de componentes principales que maximiza la predictividad del modelo.

Con el fin de entender de forma intuitiva lo que ocurre en el algoritmo durante la convergencia de w_h , se utiliza el mismo ejemplo utilizado en el apartado PLS, ecuación 4.8. En la figura 4.7 se muestra la evolución de w_1 para las 3 primeras iteraciones, dicha figura es homóloga a la figura 4.5 obtenida en el caso PLS, con la diferencia de que en KPLS aparecen mas segmentos discontinuos. La orientación de w_1 se obtiene minimizando la suma del cuadrado de dichos segmentos ponderados por k ó $(1 - k)$, de forma que los segmentos discontinuos dibujados en rojo, que son los obtenidos si $\alpha_h = u_h$, son ponderados por $(1 - k)$ y los dibujados en color *cyan*, que son los obtenidos si $\alpha_h = t_h$, son ponderados por k .

Ventajas:

- Se pueden obtener modelos más predictivos que con PLS o PCR en algunos casos.

Inconvenientes:

- Mayor carga computacional al ser necesario utilizar técnicas de validación cruzada para determinar k .

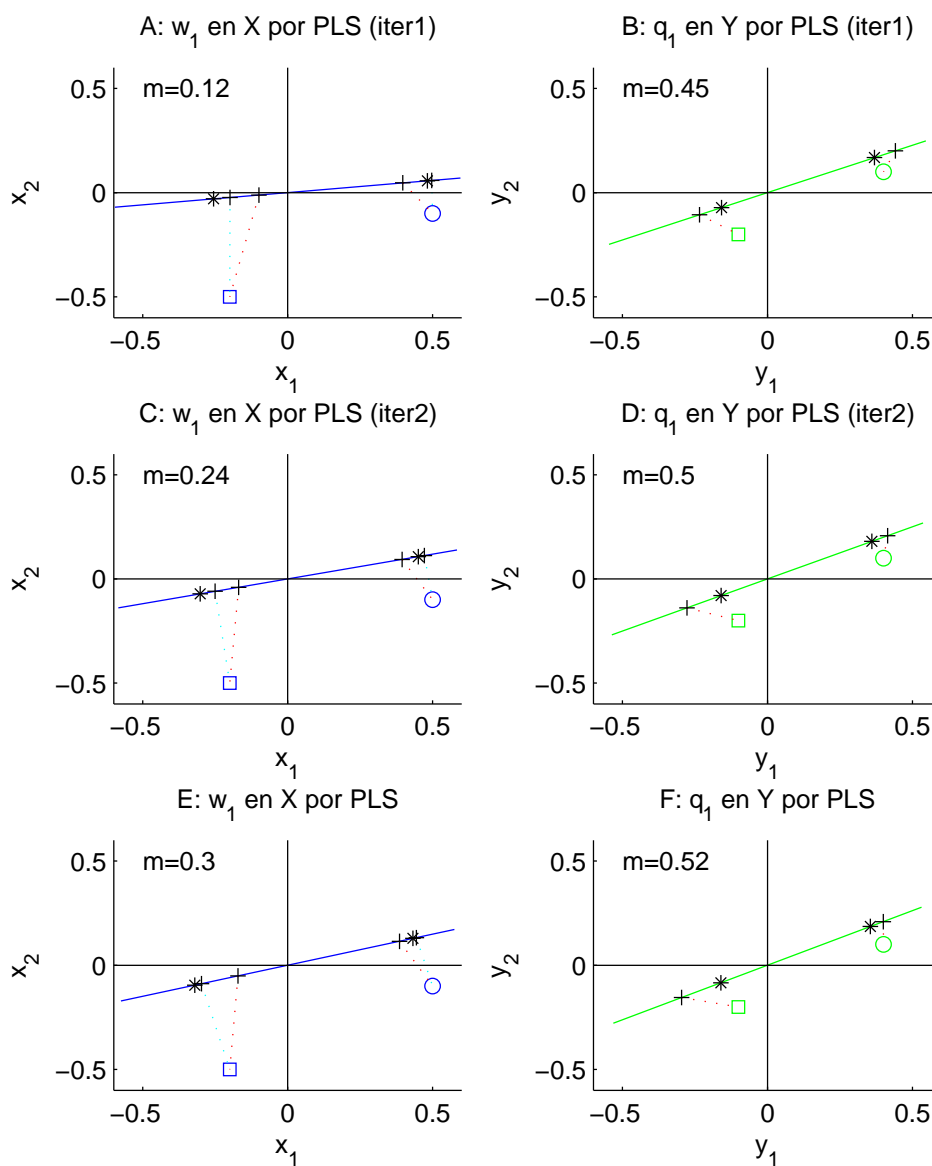


Figura 4.7: Primer componente algoritmo KPLS

4.6. DPLS

PLS dinámico es un método de regresión lineal que realiza una reducción de variables pero sólo en el espacio de las entradas aplicadas al proceso.

En el apartado 4.4 se ha introducido PLS, que parte de X la cual contiene valores anteriores de las salidas y de las entradas para capturar el carácter dinámico del proceso. Sobre dicho espacio X se realiza la reducción de variables, luego dicha reducción afecta tanto a las entradas como a las salidas y entradas anteriores. DPLS parte de una situación inicial distinta donde $X = U$, luego la reducción de variables sólo afecta a las entradas aplicadas al proceso [Pat98, Lak97].

El algoritmo DPLS, al igual que PLS, se basa en una descomposición tanto de X como de Y ¹⁵:

$$\begin{aligned} X &= TP' + \mathcal{E} \\ Y &= UQ' + \mathcal{F} \end{aligned}$$

La regresión entre los espacios de entrada y salida se realiza en el espacio de las componentes principales mediante la función f :

$$\hat{u}_h = f(t_h) \quad \forall h \in [1 \dots ncp]$$

En el algoritmo PLS original, dicha función f es una regresión lineal según:

$$\hat{u}_h = b_h t_h \quad \forall h \in [1 \dots ncp] \quad (4.11)$$

Mientras que en DPLS la función f es dinámica¹⁶:

$$\hat{u}_h(k) = G_h(z)t_h(k) = \frac{B_h(z)}{A_h(z)}t_h(k) \quad \forall h \in [1 \dots ncp] \quad (4.12)$$

¹⁵Si existe retardo d , la matriz X utilizada ha de estar desplazada d filas hacia abajo, de forma que exista relación entre $X(k)$ e $Y(k)$, luego se denomina $X(k)$ a $X(k-d)$. Esto es aconsejable porque la orientación de los componentes principales en PLS busca maximizar la covarianza entre u y t en el instante k .

¹⁶Notar que, como X ha sido desplazada d filas, existe relación entre $X(k)$ e $Y(k)$, luego $B(z^{-1})$ tiene término independiente distinto de cero : $B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_{n_b} z^{-n_b}$

En este punto se puede utilizar cualquiera de los modelos en función de transferencia introducidos en 3.2. En DPLS, según se describe en [Lak97], se utiliza la estructura ARX, luego el modelo dinámico queda definido por la siguiente expresión:

$$\begin{aligned} Y(k) &= \sum_{h=1}^{ncp} G_h(z)t_h(k)q'_h + \mathcal{F}(k) \\ \hat{Y}(k) &= \sum_{h=1}^{ncp} G_h(z)t_h(k)q'_h \end{aligned} \quad (4.13)$$

La expresión anterior define el modelo que relaciona \hat{Y} con T para $G_h(z)$ y q_h conocidas para todo h . Retomando el resultado de la ecuación 4.9, se define R como:

$$T = XW(P'W)^{-1} = XR \implies t_h = Xr_h \quad \forall h \in [1 \dots ncp]$$

Donde r_h es un vector columna de R y t_h es un vector columna de T . Se puede expresar el modelo dinámico como¹⁷:

$$\hat{Y}(k) = \sum_{h=1}^{ncp} G_h(z)X(k)r_hq'_h \quad h_h \triangleq r_hq'_h \implies \hat{Y}(k) = \sum_{h=1}^{ncp} G_h(z)X(k)h_h = \sum_{h=1}^{ncp} \frac{B_h(z)}{A_h(z)}X(k)h_h$$

Se puede despejar la expresión anterior para la salida s como:

$$\hat{y}_s(k) = \frac{\sum_{h=1}^{ncp} [(\prod_{i=1}^{ncp} A_i(z))B_h(z)X(k)h_{h_s}]}{\prod_{h=1}^{ncp} A_h(z)} \quad \forall s \in [1 \dots n_s], \quad \forall i \neq h$$

Donde h_h es una matriz de dimensiones $n_e \times n_s$ y h_{h_s} es el vector columna s de la matriz h_h . Se puede descomponer la expresión anterior como un sumatorio de la influencia de las distintas entradas sobre la salida s :

$$\hat{y}_s(k) = \sum_{e=1}^{n_e} \left[\frac{\sum_{h=1}^{ncp} [(\prod_{i=1}^{ncp} A_i(z))B_h(z)x_e(k)h_h(e, s)]}{\prod_{h=1}^{ncp} A_h(z)} \right] \quad \forall s \in [1 \dots n_s], \quad \forall i \neq h \quad (4.14)$$

¹⁷Al definir $h_h \triangleq r_hq'_h$ se pasa de evaluar $G_h(z)$ SISO a MIMO, ya que inicialmente $t_h(k) = G_h(z)X(k)r_h$ donde $G_h(z)$ es una función de transferencia SISO porque Xr_h es un vector columna. Sin embargo Xh_h es una matriz con n_s columnas, luego se evalúa $G_h(z)$ para cada una de dichas columnas, lo cual equivale a considerar $G_h(z)$ como una matriz diagonal de funciones de transferencia con todos sus elementos iguales.

A partir de la ecuación anterior se puede obtener $\hat{\theta}$ para expresar el modelo dinámico obtenido con DPLS según la ecuación:

$$\hat{Y}(k) = X(k) \hat{\theta} \quad (4.15)$$

Para ello se definen:

$$\bar{B}_{e,s}(z) = \sum_{h=1}^{ncp} [(\prod_{i=1}^{ncp} A_i(z)) B_h(z) h_h(e, s)] \quad \forall i \neq h$$

$$\bar{B}_{e,s}(z) = \bar{b}_{e,s,0} + \bar{b}_{e,s,1} z^{-1} + \dots + \bar{b}_{e,s,n_{\bar{b}}} z^{-n_{\bar{b}}}$$

$$\bar{A}(z) = \prod_{h=1}^{ncp} A_h(z)$$

$$\bar{A}(z) = 1 + \bar{a}_1 z^{-1} + \dots + \bar{a}_{n_{\bar{a}}} z^{-n_{\bar{a}}}$$

Donde $\bar{B}_{e,s}$ es un polinomio en z no mónico de orden ($n_{\bar{b}} = n_a \cdot (ncp - 1) + n_b$) y $\bar{A}(z)$ es un polinomio mónico de orden ($n_{\bar{a}} = n_a \cdot ncp$). Se puede reescribir la ecuación 4.14 como:

$$\hat{y}_s(k) = \sum_{e=1}^{n_e} \left[\frac{\bar{B}_{e,s}(z) x_e(k)}{\bar{A}(z)} \right] = \frac{1}{\bar{A}} \sum_{e=1}^{n_e} [\bar{B}_{e,s} x_e(k)] \quad \forall s \in [1 \dots n_s]$$

Si se obtiene \mathcal{F}_s ruido blanco, se comprueba que el modelo obtenido corresponde con la estructura OE de la ecuación 3.18, que para el caso MISO toma la siguiente expresión:

$$y_s(k) = \frac{1}{\bar{A}} \sum_{e=1}^{n_e} [\bar{B}_{e,s} x_e(k)] + \mathcal{F}_s(k)$$

Se puede desarrollar la expresión anterior de forma análoga a como se desarrolló en 3.18:

$$\begin{aligned} \bar{A} y_s(k) &= \sum_{e=1}^{n_e} [\bar{B}_{e,s} x_e(k)] + \bar{A} \mathcal{F}_s(k) \\ y_s(k) &= (1 - \bar{A}) y_s(k) + \sum_{e=1}^{n_e} [\bar{B}_{e,s} x_e(k)] + (\bar{A} - 1) \mathcal{F}_s(k) + \mathcal{F}_s(k) \end{aligned}$$

$$\hat{y}_s(k) = (1 - \bar{A})y_s(k) + \sum_{e=1}^{n_e} [\bar{B}_{e,s}x_e(k)] + (\bar{A} - 1)e_s(k)$$

Retomando la expresión general para definir modelos MIMO lineales 3.15, y a partir de la expresión anterior, se definen¹⁸ los siguientes polinomios de la ecuación 3.14:

$$\tilde{A}_s = (1 - \bar{A}); \quad \tilde{B}_{e,s} = \bar{B}_{e,s}; \quad \tilde{C}_s = \bar{A} - 1$$

Finalmente se analizan las ventajas e inconvenientes del algoritmo DPLS.

Ventajas:

- Se pueden diseñar n_{cp} controladores SISO y realizar el control en el espacio de las componentes principales debido a la ortogonalidad de los componentes principales del espacio de entradas (acciones de control)¹⁹.

Inconvenientes:

- Todas las matrices \tilde{A} que forman la matriz diagonal \mathbb{A} son iguales, lo cual es una limitación a la hora de aproximar procesos con polos muy distintos en sus funciones de transferencia.
- Sólo es útil en procesos cuya matriz U tenga muchas columnas, ya que la reducción a las componentes principales se realiza en dicho espacio. En el caso en que el número de entradas sea 1, $n_{cp} = 1$ y el resultado es equivalente a obtener el modelo mediante la técnica que se utilice para estimar G_h de la ecuación 4.12²⁰.

¹⁸Como la matriz X inicial había sido desplazada d filas, $x_e(k)$ es en la matriz real $x_e(k - d)$, luego $\bar{B}_{e,s}x_e(k - d) = \bar{B}_{e,s}z^{-d}x_e(k)$. Considerando el retardo intrínseco $d = 1$ se comprueba que $\bar{B}_{e,s}(z) = \bar{b}_{e,s,0}z^{-1} + \dots + \bar{b}_{e,s,n_{\bar{b}}-1}z^{-n_{\bar{b}}-1}$.

¹⁹Un problema en esta línea es la determinación de U a partir de Y , en el artículo [Sal06] se trata más en profundidad dicho problema y se proponen soluciones para la estimación de U .

²⁰En el apartado de ejemplos se utiliza PLS para estimar G_h en DPLS.

4.7. Selección de variables

Con el objetivo de obtener la matriz de parámetros $\hat{\theta}$ que proporcione un modelo más predictivo y simple, aparecen un conjunto de herramientas cuyo objetivo es realizar una selección de las variables (columnas de X) más relevantes sobre las que aplicar una técnica de regresión lineal, generalmente PLS, para obtener el modelo final [Lea00]. La característica común de este conjunto de técnicas es la utilización de indicadores de validación interna como guía para la selección de las variables relevantes, lo cual no elimina completamente la posibilidad de sobreaprendizaje porque la selección de variables depende del conjunto de identificación. A continuación se describen brevemente algunas de las técnicas de selección de variables más utilizadas [Abr03]:

- **IPLS:** En el algoritmo PLS iterativo se selecciona un conjunto de variables aleatoriamente, se aplica PLS y se evalúa el modelo utilizando validación cruzada. En cada iteración se añaden o quitan variables aleatoriamente y se obtiene de nuevo el modelo y su indicador de validación cruzada, generalmente RMSECV. En función del valor del indicador de validación cruzada obtenido, se deshace el cambio realizado en la presente iteración o no, en función de si el modelo ha mejorado. El algoritmo acaba cuando cada una de las variables se ha probado una vez sin mejorar el modelo.
- **UVE-PLS:** La idea base es añadir variables aleatorias y descartar posteriormente las aleatorias y todas las que sean menos importantes que éstas. Para ello se expande la matriz X con otra matriz de sus mismas dimensiones generada aleatoriamente con cada uno de sus elementos multiplicados por 10^{-10} ²¹. Se elimina una muestra cada vez y se obtiene un modelo para dicha matriz X ampliada ($m - 1 \times 2n_x$). Al final del proceso se habrán obtenido m modelos $\hat{\theta}$ que se pueden agrupar en la matriz $\mathcal{M}_{2n_x \times n_s \times m}$. Se obtiene la matriz $C_{2n_x \times n_s}$ a partir de \mathcal{M} según:

$$C(e, s) = \frac{E\{\hat{\theta}(e, s, :)\}}{std(\hat{\theta}(e, s, :))}$$

Se eliminan todas las columnas de X añadidas al realizar la ampliación, y las columnas i tales que todos los elementos del vector columna c_i sean menores al máximo valor de C asociado a las variables aleatorias introducidas. A partir de este momento se dispone de X reducida para realizar la regresión [Cen96].

- **IVS-PLS:** La selección de variables se realiza para cada componente principal obtenido durante la ejecución del algoritmo PLS. La forma de proceder es obtener w_1 con una iteración de PLS, y sobre dicho w_1 , se seleccionan grupos de variables consecutivas (ventanas) evaluando el indicador RMSECV, finalmente se selecciona

²¹El objetivo es hacer que estas variables estén al menos un orden por debajo de la precisión del instrumento y no afecten a los parámetros obtenidos para las variables reales.

la ventana que proporciona un mejor valor del indicador, se ponen como cero el resto de elementos del vector w_1 y se aplica una iteración del algoritmo PLS con el nuevo vector w_1 como origen. Esta es una versión de PLS modificada que en la orientación de sus componentes principales no busca sólo una máxima correlación de las puntuaciones entre los espacios de entrada y salida, sino que también selecciona las variables más relevantes utilizando la validación cruzada, lo cual influye en la selección de w_h que representa la rotación del sistema de coordenadas[Abr03].

- **GAPLS:** Se utilizan algoritmos genéticos en la selección de variables, para ello se definen cromosomas binarios de dimensiones $(1 \times n_x)$, dicho cromosoma se multiplica por X , haciendo cero las columnas que, según dicho cromosoma, no han de aparecer en el modelo. Cada individuo da lugar a una matriz X modificada sobre la que se obtiene el modelo PLS y se evalúa el indicador de validación cruzada, de modo que los cromosomas con mejor valor del indicador de validación cruzada tienen más probabilidad de sobrevivir, para explorar el espacio se realizan mutaciones y cruces entre los individuos de cada población. Cuando el algoritmo converge se obtiene el individuo dominante que da lugar a la matriz X reducida a partir de la cual se realiza la identificación mediante PLS [Has97].

Las técnicas de selección de variables presentan las siguientes ventajas e inconvenientes.

Ventajas:

- El modelo obtenido puede ser más predictivo que el obtenido utilizando la matriz X original, ya que se eliminan las variables que no aportan información antes de aplicar el algoritmo de estimación de $\hat{\theta}$.
- La matriz de parámetros tiene un tamaño menor porque se reduce el número de variables presentes en X .

Inconvenientes:

- Alto coste computacional.
- La selección de variables se realiza utilizando el conjunto de datos de identificación y técnicas de validación interna, luego depende del conjunto de datos de identificación.

4.8. PLSA

Se propone PLSA como una extensión del método PLS donde se aumenta el espacio muestral con el fin de ayudar en la estimación de los parámetros del modelo.

En PLS se estima $\hat{\theta}$ de forma que a partir de una fila de la matriz de regresión $X(k)$, se pueda obtener una estimación de las salidas en dicho instante:

$$\hat{Y}(k) = X(k)\hat{\theta}$$

No obstante, no se considera en la obtención de $\hat{\theta}$, que dicha matriz de parámetros pueda ser utilizada no sólo para realizar la estimación a un instante vista, sino dentro de una ventana de predicción utilizando el modelo de predicción a i instantes vista. Si se desea que la obtención de $\hat{\theta}$ considere el posterior uso del modelo en la ventana de predicción, se puede reformular el problema desde uno de los dos siguientes puntos de vista:

- Ampliar el espacio de salidas, con lo que aumentará el número de elementos en la matriz θ , y establecer restricciones entre los elementos de dicha matriz θ , tantas como sea necesario para que el número de grados de libertad del problema ampliado sea el mismo que el del problema original.

$$Y(k) = X(k)\hat{\theta} + E(k)$$

$$[Y(k) Y(k+1) \dots Y(k+n_f)] = Y_a(k) = X_a(k)\hat{\theta}_a + E(k)$$

Esta solución presenta un problema importante a la hora de establecer las restricciones entre los elementos de $\hat{\theta}_a$, ya que aparecen relaciones no lineales que dificultan la estimación de los parámetros. A continuación se dispone un ejemplo SISO para ilustrar el procedimiento:

$$Y(k) = X(k)\hat{\theta} + E(k)$$

$$\hat{Y}(k) = X(k)\hat{\theta} = [u(k-1) \quad y(k-1)] \begin{bmatrix} b_1 \\ -a_1 \end{bmatrix}$$

$$\hat{y}(k) = b_1 u(k-1) - a_1 y(k-1)$$

La predicción para $i = 1$ se obtiene según²²:

$$\hat{y}(k+1) = b_1 u(k) - a_1 y(k) = b_1 u(k) - a_1 (e(k) - \hat{y}(k))$$

²²Se procede de forma análoga a la utilizada en el apartado 3.3.

Como $e(k+i)$, $\forall i \geq 0$ no se conoce, se asume cero para minimizar la varianza del error en la predicción²³, luego:

$$\hat{y}(k+1) = b_1 u(k) - a_1 \hat{y}(k) = b_1 u(k) - a_1 b_1 u(k-1) + a_1^2 y(k-1)$$

$$\begin{bmatrix} \hat{y}(k) & \hat{y}(k+1) \end{bmatrix} = \begin{bmatrix} u(k) & u(k-1) & y(k-1) \end{bmatrix} \begin{bmatrix} 0 & b_1 \\ b_1 & -a_1 b_1 \\ -a_1 & a_1^2 \end{bmatrix} = X_a(k) \hat{\theta}_a$$

Se observa que en $\hat{\theta}_a$ hay dos parámetros a estimar a_1 y b_1 , al igual que en $\hat{\theta}$, pero se han establecido restricciones que ayudan en la obtención de un modelo que va a ser utilizado para realizar predicciones en una ventana de predicción.

- Ampliar el espacio muestral de la matriz de regresión X utilizando estimaciones de las salidas para reproducir la situación en la que será utilizado el modelo si se desea hacer predicciones en una ventana de predicción. De esta forma se evita tener que ampliar la matriz de parámetros y establecer restricciones no lineales entre los elementos de la matriz de parámetros ampliada, aunque se trata de un procedimiento iterativo ya que para ampliar X e Y se necesita $\hat{\theta}$, y para obtener $\hat{\theta}$ se necesitan las matrices X_a e Y_a . El ejemplo anterior toma la siguiente forma si se realiza una ampliación en el espacio muestral en lugar de en el espacio de salidas:

$$\begin{bmatrix} \hat{y}(k) \\ \hat{y}(k+1) \end{bmatrix} = \begin{bmatrix} u(k-1) & y(k-1) \\ u(k) & \hat{y}(k) \end{bmatrix} \begin{bmatrix} b_1 \\ -a_1 \end{bmatrix} = X_a(k) \hat{\theta} = \hat{Y}_a(k)$$

PLSA se basa en la ampliación del espacio muestral ya que de esta forma se puede obtener una solución general y el problema es menos costoso de resolver computacionalmente. PLSA es un método iterativo en el que se obtiene un primer modelo que es evaluado para ampliar el número de muestras en la matriz de regresión según el siguiente procedimiento:

1. Inicializar las matrices ampliadas: $X_a = X$; $Y_a = Y$
2. $i = 1$
3. Definir las matrices temporales de regresión $X_i = X$ y de salidas $Y_i = Y$
4. Obtener $\hat{\theta}$ mediante PLS
5. Determinar la predicción a un instante vista: $\hat{Y} = X_i \hat{\theta}$

²³Debido a que en realidad no es cero, el término de error asociado a $\hat{y}(k+1)$ no será sólo el ruido blanco $e(k+1)$ sino $y(k+1) = \hat{y}(k+1) + e(k+1) - a e(k)$, luego no se puede utilizar LS para estimar los parámetros ya que no se dispondrá de una ecuación como en el caso de modelo de predicción a un instante vista y modelo ARX, $Y = X\theta + E$ donde E es ruido blanco.

6. Descomponer X_i en dos submatrices $X_i = [X_u \ X_y]$, una que contiene las entradas anteriores y otra que contiene las salidas anteriores.
7. Eliminar la primera fila de X_u y formar la nueva $X_i = [X_u \ \hat{Y} \ X_y]$
8. Eliminar la última fila y las últimas n_s columnas de X_i
9. Eliminar la primera fila de Y_i
10. Ampliar las matrices: $X_a = [X_a; X_i]$; $Y_a = [Y_a; Y_i]$
11. Incrementar i y si es menor o igual a n_f volver al punto 5, en caso contrario finalizar

En la figura 4.8 se representa de forma esquemática el procedimiento para la obtención de X_a . Cada matriz X_i constituye la matriz de regresión que se tendrá al hacer la predicción para $k + i$ con información de las salidas disponible hasta k .

Una vez obtenidas las matrices X_a e Y_a se utiliza el algoritmo PLS para determinar $\hat{\theta}$ de nuevo y se vuelven a obtener las matrices ampliadas para la nueva matriz de parámetros. Cuando los parámetros obtenidos no cambien de una iteración a la siguiente, se considera que el algoritmo ha convergido.

Debido a la posible diferencia estructural entre el modelo y el proceso, la primera estimación $\hat{\theta}$ puede no aproximar adecuadamente el comportamiento del proceso, luego las estimaciones \hat{Y} utilizadas en las matrices X_i pueden ser muy distintas de los valores reales de las salidas, tanto más cuanto mayor sea i . Esto puede generar problemas de convergencia en el algoritmo ya que conforme se ha planteado, todas las matrices X_i introducidas en X_a presentan la misma importancia. Alternativamente se pueden formar las matrices ampliadas según:

$$X_a = \begin{bmatrix} X_a \\ (n_f - i)^{pesos} X_i \end{bmatrix} \quad Y_a = \begin{bmatrix} Y_a \\ (n_f - i)^{pesos} Y_i \end{bmatrix}$$

De esta forma la importancia de cada submatriz X_i en la matriz X_a será menor conforme mayor sea i , luego más lejos esté del instante k . La variable *pesos* define el orden de la ponderación, luego el resultado final dependerá de esta variable que se ha de escoger durante el proceso de identificación.

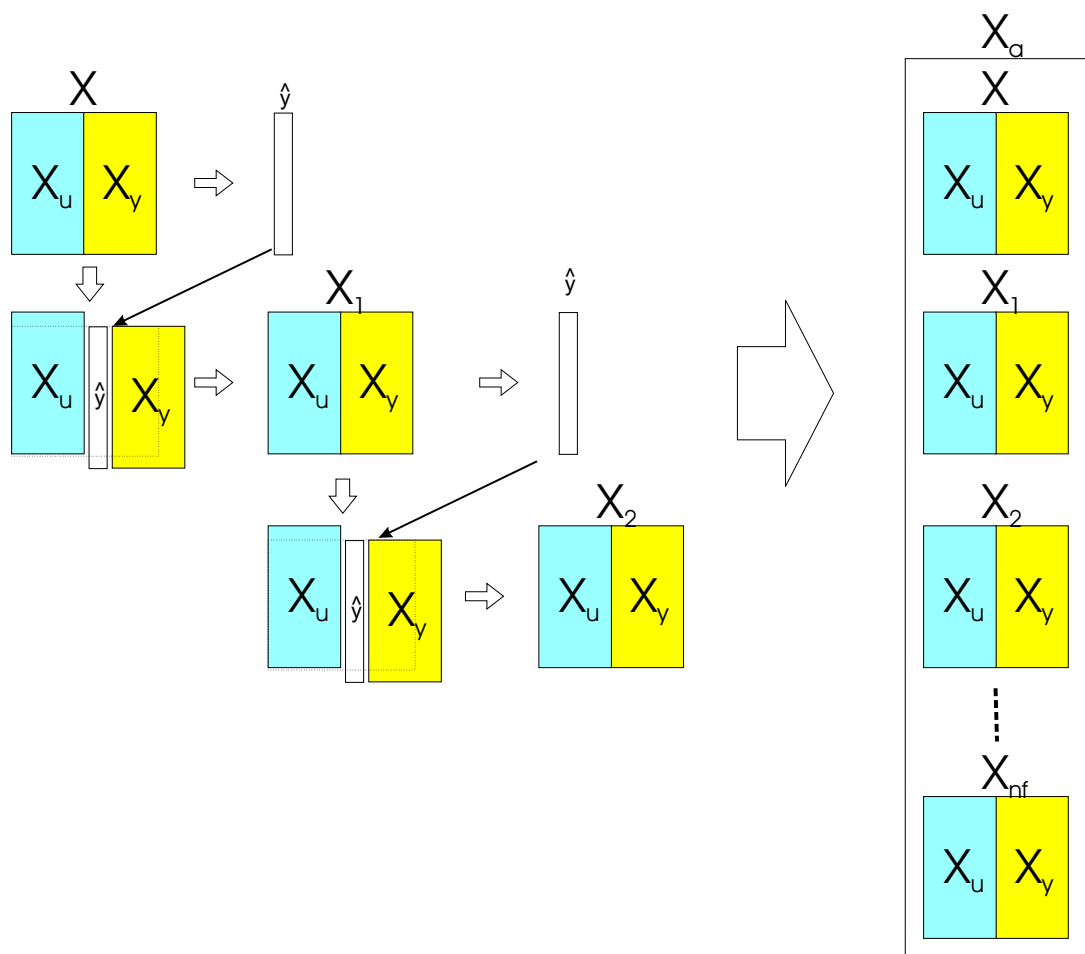


Figura 4.8: Obtención de X_a

Ventajas:

- Se puede obtener un mejor modelo que el obtenido por PLS convencional si el modelo va a ser utilizado para realizar predicciones en un horizonte de predicción n_f .

Inconvenientes:

- Para valores bajos de la variable *pesos* y grandes ventanas de predicción n_f , el algoritmo puede presentar problemas de convergencia.
- El resultado depende del valor de la variable *pesos*.

Capítulo 5

Ejemplos de aplicación

En esta capítulo se estima la matriz de parámetros $\hat{\theta}$ de la ecuación general de modelos lineales respecto a sus parámetros para dos procesos distintos. La estimación se realiza a partir de datos obtenidos de los procesos utilizando algunas de las técnicas explicadas en el capítulo 4, así como las técnicas disponibles en MATLAB[®] para la identificación de modelos en función de transferencia.

- Técnicas de regresión lineal explicadas en el capítulo 4:
 - LS
 - PLS
 - PCR
 - KPLS
 - PLSA
 - DPLS
- Técnicas basadas en modelos en fdt de MATLAB[®]:
 - ARX
 - ARMAX
 - OE
 - BJ
 - IV

Las técnicas de regresión lineal utilizan la matriz de regresión $X = [\mathbf{U} \ \mathbf{Y}]$, obteniendo un modelo lineal respecto a los parámetros. En cambio, la transformación

de los modelos en función de transferencia a la estructura $\hat{Y}(k) = X(k)\hat{\theta}$ da lugar a la matriz $X = [\underline{\mathbf{U}} \quad \underline{\mathbf{Y}} \quad \underline{\mathbf{E}}]$, como se vió en el apartado 3.2. Con el fin de comparar entre todas las técnicas en igualdad de condiciones respecto a tamaño y complejidad del modelo final, sólo se utilizan los términos de $\hat{\theta}$ asociados a $X = [\underline{\mathbf{U}} \quad \underline{\mathbf{Y}}]$, que son las submatrices $\tilde{\mathbf{A}}$ y $\tilde{\mathbf{B}}$, luego la submatriz $\tilde{\mathbf{C}}$ obtenida a partir de los modelos en función de transferencia es despreciada.

Pretratamiento de los datos

En referencia al pretratamiento de los datos se realizan las operaciones de centrado y escalado de las matrices de datos de identificación $\underline{\mathbf{U}}$ e $\underline{\mathbf{Y}}$ según las ecuaciones 2.2 y 2.3, obteniendo:

$$\mathbf{U}(i, j) = \frac{\underline{\mathbf{U}}(i, j) - E\{\underline{\mathbf{U}}(:, j)\}}{std(\underline{\mathbf{U}}(:, j))} \quad \mathbf{Y}(i, j) = \frac{\underline{\mathbf{Y}}(i, j) - E\{\underline{\mathbf{Y}}(:, j)\}}{std(\underline{\mathbf{Y}}(:, j))}$$

Expresando de forma matricial¹:

$$\mathbf{U} = (\underline{\mathbf{U}} - mU) ./ sU \quad \mathbf{Y} = (\underline{\mathbf{Y}} - mY) ./ sY$$

Donde: mU es un vector fila en el que el elemento i es el valor medio de la columna i de la matriz $\underline{\mathbf{U}}$ obtenido como $mU(1, i) = E\{\underline{\mathbf{U}}(:, i)\}$, $\forall i \in [1 \dots n_e]$. sU es un vector fila en el que el elemento i es la desviación estándar de la columna i de la matriz $\underline{\mathbf{U}}$ según $sU(1, i) = std(\underline{\mathbf{U}}(:, i))$, $\forall i \in [1 \dots n_e]$. De forma análoga se obtienen mY y sY .

Matrices X e Y

Las matrices de regresión X y salidas Y para realizar la estimación, se obtienen a partir de los datos de identificación pretratados \mathbf{U} e \mathbf{Y} utilizando las ecuaciones 3.12 y 3.13, pero sin incluir la submatriz $\underline{\mathbf{E}}$ como se ha comentado anteriormente:

$$Y = \begin{bmatrix} \mathbf{y}_1(\alpha + 1) & \dots & \mathbf{y}_{n_s}(\alpha + 1) \\ \vdots & \ddots & \vdots \\ \mathbf{y}_1(n) & \dots & \mathbf{y}_{n_s}(n) \end{bmatrix}$$

¹El operador ./ implementa la división vectorial que puede expresarse de forma general como: $a./b = \frac{a(i,j)}{b(i,j)}$, en este caso particular b es un vector fila, luego: $a./b = \frac{a(i,j)}{b(1,j)}$. El operador resta entre la matriz A y el vector b se implementa $A(j, i) - b(i)$.

$$X = \begin{bmatrix} \mathbf{u}_1(\alpha) & \dots & \mathbf{u}_1(\alpha - n_{\bar{b}}) & , \dots , & \mathbf{u}_{n_e}(\alpha) & \dots & \mathbf{u}_{n_e}(\alpha - n_{\bar{b}}) \\ \vdots & \ddots & \vdots & \dots & \vdots & \ddots & \vdots \\ \mathbf{u}_1(n-1) & \dots & \mathbf{u}_1(n-1 - n_{\bar{b}}) & , \dots , & \mathbf{u}_{n_e}(n-1) & \dots & \mathbf{u}_{n_e}(n-1 - n_{\bar{b}}) \\ \\ \mathbf{y}_1(\alpha) & \dots & \mathbf{y}_1(\alpha - n_{\bar{a}}) & , \dots , & \mathbf{y}_{n_s}(\alpha) & \dots & \mathbf{y}_{n_s}(\alpha - n_{\bar{a}}) \\ \vdots & \ddots & \vdots & \dots & \vdots & \ddots & \vdots \\ \mathbf{y}_1(n-1) & \dots & \mathbf{y}_1(n-1 - n_{\bar{a}}) & , \dots , & \mathbf{y}_{n_s}(n-1) & \dots & \mathbf{y}_{n_s}(n-1 - n_{\bar{a}}) \end{bmatrix}$$

Donde: $\alpha = \max(n_{\bar{a}}, n_{\bar{b}})$, n es el número de muestras obtenidas en el experimento y se define m como el número de muestras de la matriz de regresión $m = n - \alpha$.

Modelo Final

Para obtener la expresión matemática del modelo que relacione entradas y salidas sin pretratamiento se han de utilizar: $\hat{\theta}$, mX , mY , sX y sY .

sX y mX se pueden obtener a partir de mU , sU , mY y sY . El vector mX se forma como la concatenación de n_b veces el vector mU y n_a veces el vector mY y el vector sX se forma como la concatenación de n_b veces el vector sU y n_a veces el vector sY según:

$$\begin{aligned} mX &= [mU \dots mU \quad mY \dots mY] \\ sX &= [sU \dots sU \quad sY \dots sY] \end{aligned}$$

Se obtiene la siguiente expresión que define las operaciones de pretratamiento realizadas, pero en el espacio de la matriz de regresión X :

$$X = (\underline{\mathbf{X}} - mX) ./ sX$$

Ante una nueva muestra en un instante p se forma $\underline{\mathbf{X}}(p)$ con información disponible medida del proceso según:

$$\underline{\mathbf{X}}(p) = \begin{bmatrix} \mathbf{u}_1(p-1) & \dots & \mathbf{u}_1(p-1 - n_{\bar{b}}) & , \dots , & \mathbf{u}_{n_e}(p-1) & \dots & \mathbf{u}_{n_e}(p-1 - n_{\bar{b}}) \\ \\ \mathbf{y}_1(p-1) & \dots & \mathbf{y}_1(p-1 - n_{\bar{a}}) & , \dots , & \mathbf{y}_{n_s}(p-1) & \dots & \mathbf{y}_{n_s}(p-1 - n_{\bar{a}}) \end{bmatrix}$$

A partir de $\underline{\mathbf{X}}(p)$ se estima $\hat{\mathbf{Y}}(p|p-1)$ siguiendo los siguientes pasos:

1. Pretratamiento de la nueva muestra $\underline{\mathbf{X}}(p)$ según:

$$X(p) = (\underline{\mathbf{X}}(p) - mX) ./ sX$$

2. Estimación de la salida:

$$\hat{Y}(p|p-1) = X(p)\hat{\theta}$$

3. Operación inversa al pretratamiento de la salida²:

$$\hat{\underline{Y}}(p|p-1) = (\hat{Y}(p|p-1) .* sY) + mY$$

Alternativamente se puede transformar $\hat{\theta}$ de forma que incluya parte de las operaciones de pretratamiento, para ello se procede de la siguiente forma:

$$\begin{aligned} \hat{Y} &= X\hat{\theta} \\ (\hat{\underline{Y}} - mY) ./ sY &= (\underline{\mathbf{X}} - mX) ./ sX \hat{\theta} \\ (\hat{\underline{Y}} - mY) &= (\underline{\mathbf{X}} - mX) ./ sX \hat{\theta} .* sY \\ (\hat{\underline{Y}} - mY) &= (\underline{\mathbf{X}} - mX) \tilde{\theta} \\ \hat{\underline{Y}} &= (\underline{\mathbf{X}} - mX) \tilde{\theta} + mY \end{aligned}$$

Donde $\tilde{\theta}$ se define como la matriz $\hat{\theta}$ que incluye la operación de centrado de los datos:

$$\begin{aligned} \tilde{\theta} &= (\hat{\theta}' ./ sX)' .* sY \\ \hat{\theta} &= (\tilde{\theta}' .* sX)' ./ sY \end{aligned}$$

Luego el modelo final de predicción es:

$$\hat{\underline{Y}} = (\underline{\mathbf{X}} - mX) \tilde{\theta} + mY \quad (5.1)$$

Comparación de técnicas

El objetivo de este capítulo es comparar las propiedades estadísticas de los modelos obtenidos con los distintos métodos de estimación, así como su capacidad predictiva en un horizonte de predicción $n_f = 15$. Para esto se utilizan técnicas de validación tanto interna como externa.

Para la validación interna se utiliza LMOCV, que en su aplicación normal obtendría cada una de las matrices de datos de validación e identificación (X_i, X_v, Y_i, Y_v) de la siguiente forma³:

²el operador .* implementa el producto *Hadamard* que puede expresarse de forma general como: $a .* b = a(i, j)b(i, j)$, en este caso particular b es un vector fila, luego: $a .* b = a(i, j)b(1, j)$

³Donde cada una de las submatrices de validación contienen n_f muestras.

$$X = \begin{bmatrix} X_{anterior} \\ X_{validacion} \\ X_{posterior} \end{bmatrix} \quad Y = \begin{bmatrix} Y_{anterior} \\ Y_{validacion} \\ Y_{posterior} \end{bmatrix}$$

$$X_i = \begin{bmatrix} X_{anterior} \\ X_{posterior} \end{bmatrix} \quad X_v = [X_{validacion}]$$

$$Y_i = \begin{bmatrix} Y_{anterior} \\ Y_{posterior} \end{bmatrix} \quad Y_v = [Y_{validacion}]$$

No obstante, y debido a que algunas de las técnicas como PLSA requieren unas matrices X_i e Y_i formadas con muestras consecutivas, se forman las matrices de datos de identificación según:

Si $X_{anterior}$ tiene más muestras que $X_{posterior}$:

$$X_i = X_{anterior} \quad Y_i = Y_{anterior}$$

En caso contrario:

$$X_i = X_{posterior} \quad Y_i = Y_{posterior}$$

A continuación se resumen los indicadores utilizados para comparar las distintas técnicas de estimación.

▪ Validación Interna LMOCV:

- Desviación estándar parámetros: Se obtiene mediante LMOCV la desviación estándar de cada uno de los elementos de la matriz $\hat{\theta}$ obtenida para cada uno de los distintos métodos a comparar.
- Q^2 : Se obtiene el coeficiente de determinación múltiple para las salidas obtenidas utilizando los modelos LMOCV como modelos de predicción ventana, figura 2.6, luego se obtienen $n_f \times n_s$ valores de Q^2 .
- max(e): Se obtiene el máximo del valor absoluto de los errores obtenidos utilizando los modelos LMOCV como modelos de predicción ventana, figura 2.6, luego se obtienen $n_f \times n_s$ valores.

▪ Validación Externa:

- R^2 : Se obtiene el coeficiente de determinación múltiple para las predicciones del modelo evaluado ante el conjunto de datos de validación externa como modelo de predicción ventana, figura 2.6, luego se obtienen $n_f \times n_s$ valores de R^2 .

Los indicadores comentados se unen en un conjunto de gráficas que proporcionan una imagen comparativa de los distintos métodos.

5.1. BJ SISO primer orden

Como primer ejemplo se define un proceso discreto ($T_s = 0,01s$) de primer orden con estructura *Box-Jenkins* según:

$$y(k) = \frac{0,5z^{-1}}{1 + 0,5z^{-1}}u(k) + \frac{1}{1 + 0,8z^{-1}}\xi(k)$$

Se realizan dos experimentos en lazo abierto sobre el proceso con $u(k)$ ruido gaussiano filtrado en la ventana $[0,1 \dots 0,2]$ respecto a la frecuencia de *Nyquist* ($f_N = \frac{\pi}{T_s}$) obteniendo los datos de identificación y validación de la figura 5.1. La relación señal ruido para el experimento de identificación es de 7 dB .

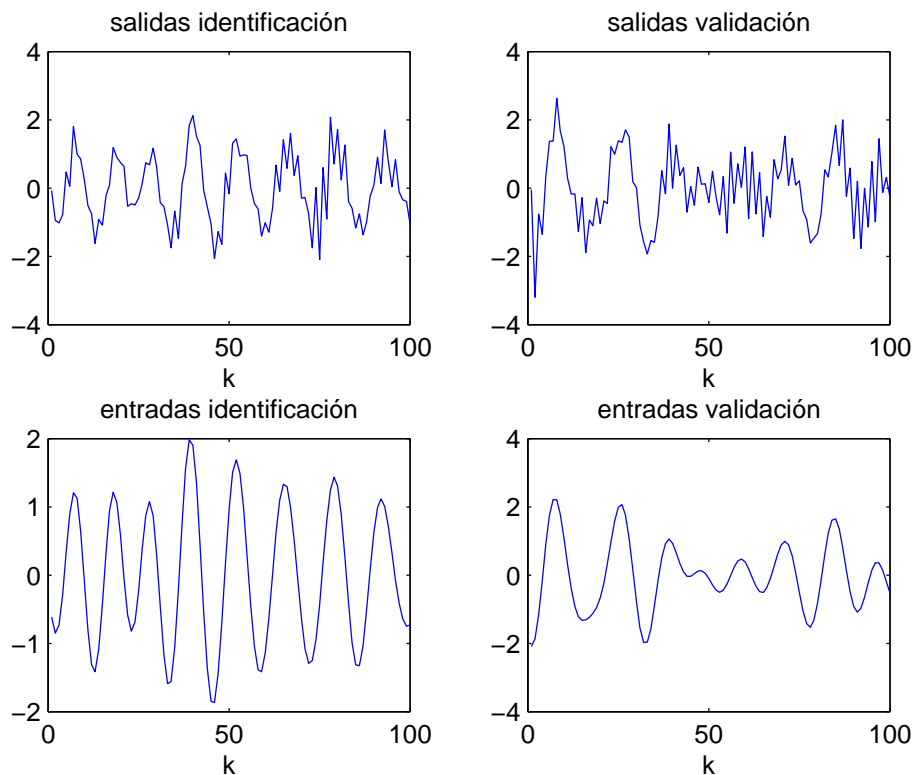


Figura 5.1: *Datos de los experimentos en lazo abierto*

Transformando el anterior proceso de función de transferencia *Box-Jenkins* a la forma general de los modelos lineales de la ecuación 3.2 se tiene:

$$\tilde{\mathbb{B}} = BD = 0,5z^{-1} + 0,4z^{-2}$$

$$\tilde{\mathbf{A}} = 1 - AD = [-1,3z^{-1} - 0,4z^{-2}]$$

$$\tilde{\mathbf{C}} = 0,5z^{-1}$$

$$n_{\tilde{b}} = n_b + n_d = 2; \quad n_{\tilde{a}} = n_a + n_d = 2; \quad n_{\tilde{c}} = n_a + n_c = 1$$

Luego:

$$y(k) = X(k)\theta^* + \xi(k)$$

$$y(k) = [u(k-1) \quad u(k-2) \quad y(k-1) \quad y(k-2) \quad \xi(k-1)] \begin{bmatrix} 0,5 \\ 0,4 \\ -1,3 \\ -0,4 \\ 0,5 \end{bmatrix} + \xi(k)$$

El anterior proceso va a ser aproximado por el siguiente modelo:

$$\hat{y}(k) = [u(k-1) \quad u(k-2) \quad y(k-1) \quad y(k-2)] \begin{bmatrix} \tilde{b}_1 \\ \tilde{b}_2 \\ \tilde{a}_1 \\ \tilde{a}_2 \end{bmatrix} = X(k)\hat{\theta}$$

Al realizar las operaciones de pretratamiento sobre los datos iniciales $\underline{\mathbf{Y}}$ y $\underline{\mathbf{U}}$ se obtiene:

$$\begin{aligned} mX &= [0 \quad 0 \quad -0,0013 \quad -0,0013] & mY &= -0,0013 \\ sX &= [1,0050 \quad 1,0050 \quad 0,3941 \quad 0,3941] & sY &= 0,3941 \end{aligned}$$

La matriz de parámetros real que considera el pretratamiento, luego es comprable con las $\hat{\theta}$ obtenidas, se obtiene:

$$\Theta = ((\theta^*)' \cdot * sX)' ./ sY$$

$$\Theta = \begin{bmatrix} 1,2752 \\ 1,0202 \\ -1,3000 \\ -0,4000 \end{bmatrix}$$

Se estima $\hat{\theta}$ mediante las distintas técnicas a comparar, utilizando las matrices X e Y para las técnicas de regresión lineal y \mathbf{U} , \mathbf{Y} en las técnicas basadas en funciones de transferencia de MATLAB[®]⁴.

- Técnicas de regresión lineal:
 - LS
 - PLS: Se utilizan 3 componentes principales de 4. La figura 5.2 muestra la evolución del índice J (RMSECV) utilizado para tomar la decisión
 - PCR: Se utilizan 3 componentes principales de 4. La figura 5.3 muestra la evolución del índice J (RMSECV) utilizado para tomar la decisión
 - KPLS⁵: Se utilizan 3 componentes principales de 4 y el valor de k para el modelo obtenido con X e Y completas es $k = [0 \ 0 \ 1]$
 - PLSA: $pesos = 3$
 - DPLS: Se utiliza 1 componente principal ya que sólo se dispone de una entrada
- Técnicas basadas en modelos en fdt de MATLAB[®]:
 - ARX: $n_a = 2$ y $n_b = 2$
 - ARMAX: $n_a = 2$, $n_b = 2$ y $n_c = 1$
 - OE: $n_a = 2$ y $n_b = 2$
 - BJ: $n_a = 1$, $n_b = 1$, $n_c = 0$ y $n_d = 1$
 - IV: $n_a = 2$ y $n_b = 2$

En la figura 5.4 se observa el resultado de validación obtenido, a continuación se comenta el contenido de cada una de las 4 gráficas, así como una interpretación de los resultados:

- Validación Interna LMOCV:
 - std: Contiene el valor de la desviación estándar de cada uno de los cuatro parámetros de $\hat{\theta}$. Se observa que OE y ARMAX son los que presentan una mayor desviación estándar en la estimación, siendo máxima para el parámetro 2 que es \tilde{b}_2 asociado a $u(k-2)$. PCR es la técnica que proporciona los parámetros con menos desviación estándar.

⁴Para más información acerca del uso de estas técnicas, se remite al lector al manual del *System Identification Toolbox* de MATLAB[®].

⁵Se selecciona para cada componente el valor de k que minimiza la norma de la matriz de errores en la ventana de predicción.

- Q^2 : Contiene el coeficiente de determinación múltiple LMOCV dentro de la ventana de predicción definida por $n_f = 15$. Se observa que OE, ARMAX y IV presentan un peor valor del índice dentro de toda la ventana de predicción. El resto de métodos presentan valores en el mismo rango convergiendo a $Q^2 = 0,8$ conforme mayor es i ($y(k+i)$, $\forall i \in [1 \dots n_f]$).
 - $\max(e)$: Todas las técnicas, a excepción de OE y ARMAX, presentan un error máximo acotado de aproximadamente 1,5, que para el rango de la señal de salida 4, supone un 40 %.
- Validación Externa:
- R^2 : Contiene el coeficiente de determinación múltiple dentro de la ventana de predicción definida por $n_f = 15$ utilizando los datos del conjunto de validación. Se observa similitud de comportamientos con los resultados de la gráfica Q^2 , lo que pone de manifiesto el grado de robustez de los indicadores obtenidos por validación interna.

Conclusiones

Se observa que a excepción de OE, IV y ARMAX, cualquiera de los métodos comparados presenta resultados aceptables para estimar $\hat{\theta}$ en este caso. Cabe remarcar que se trata de un caso simple donde el conjunto de datos contiene mucha información del sistema debido al contenido en frecuencias de la señal de excitación, luego no se aprecia ninguna ventaja de las técnicas basadas en la reducción de variables frente a LS.

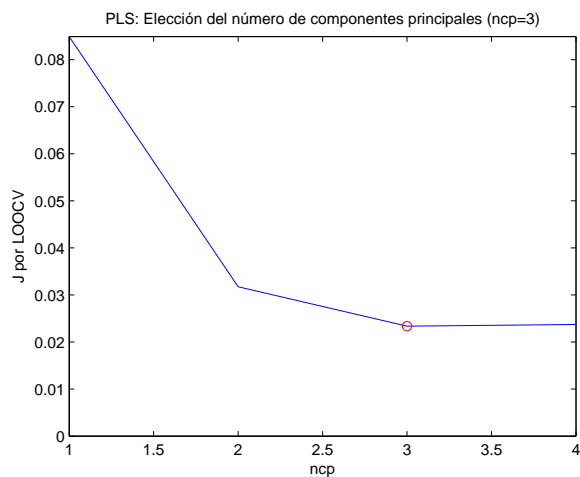


Figura 5.2: Selección del número de componentes principales PLS

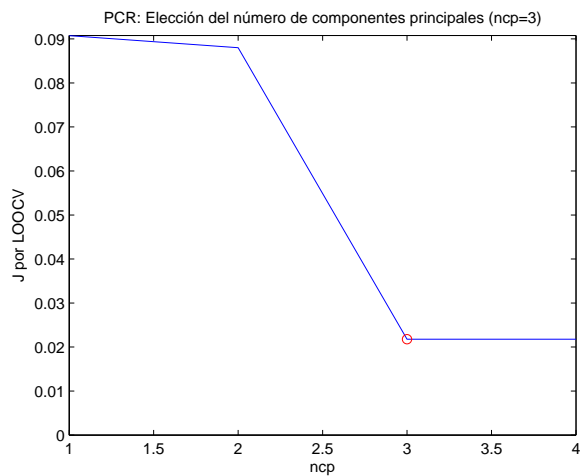


Figura 5.3: Selección del número de componentes principales PCR

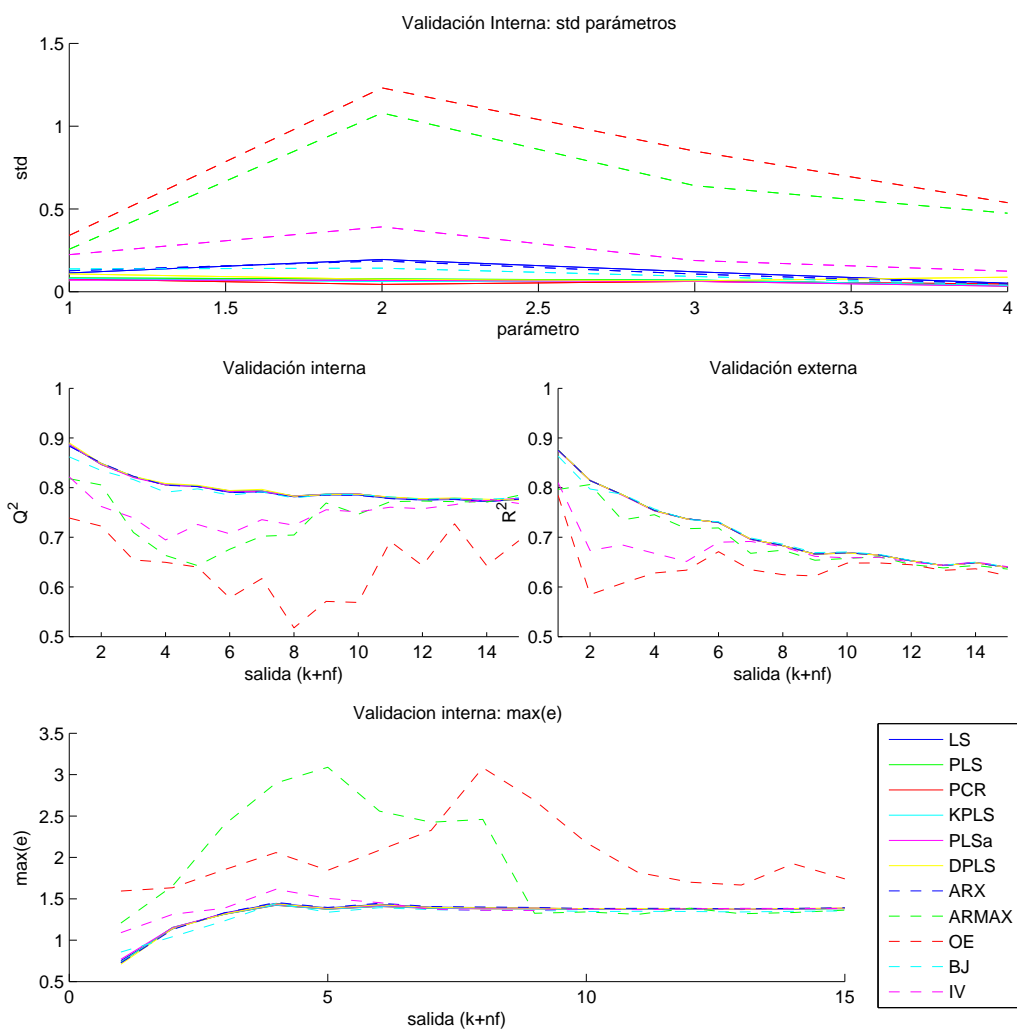


Figura 5.4: Gráficas de validación

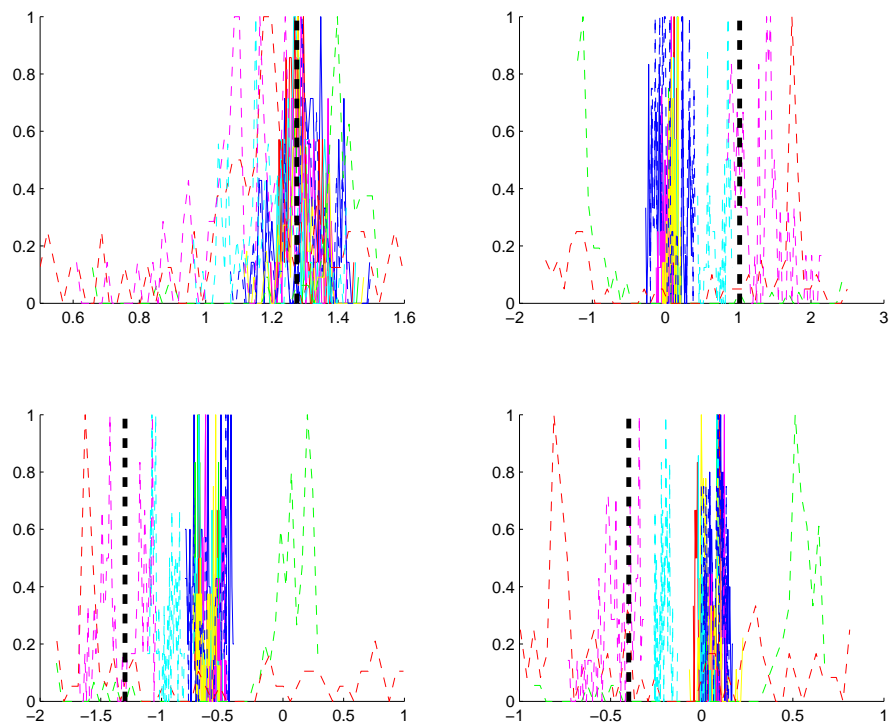


Figura 5.5: *histogramas de los parámetros*

5.2. MIMO

Se pretende modelar una columna de destilación con 2 entradas y 2 salidas, para generar los datos se utiliza el modelo *Wood and Berry*(1973). La unidad de tiempo son minutos:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{12,8e^{-s}}{16,7s+1} & \frac{-18,9e^{-3s}}{21s+1} \\ \frac{6,6e^{-7s}}{10,9s+1} & \frac{-19,4e^{-3s}}{14,4s+1} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \frac{3,8e^{-8,1s}}{14,9s+1} & \frac{0,22e^{-7,7s}}{22,8s+1} \\ \frac{4,9e^{-3,4s}}{13,2s+1} & \frac{0,14e^{-9,2s}}{12,1s+1} \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \quad (5.2)$$

Se realizan dos experimentos en lazo abierto sobre el proceso discretizado con \mathbf{U} ruido gaussiano filtrado en la ventana $[0 \dots 0,05]$ respecto a la frecuencia de *Nyquist* ($f_N = \frac{\pi}{T_s}$) obteniendo los datos de identificación y validación de la figura 5.6. ξ ha sido fijada de forma que la relación señal ruido para ambas salidas sea del orden de 15 dB.

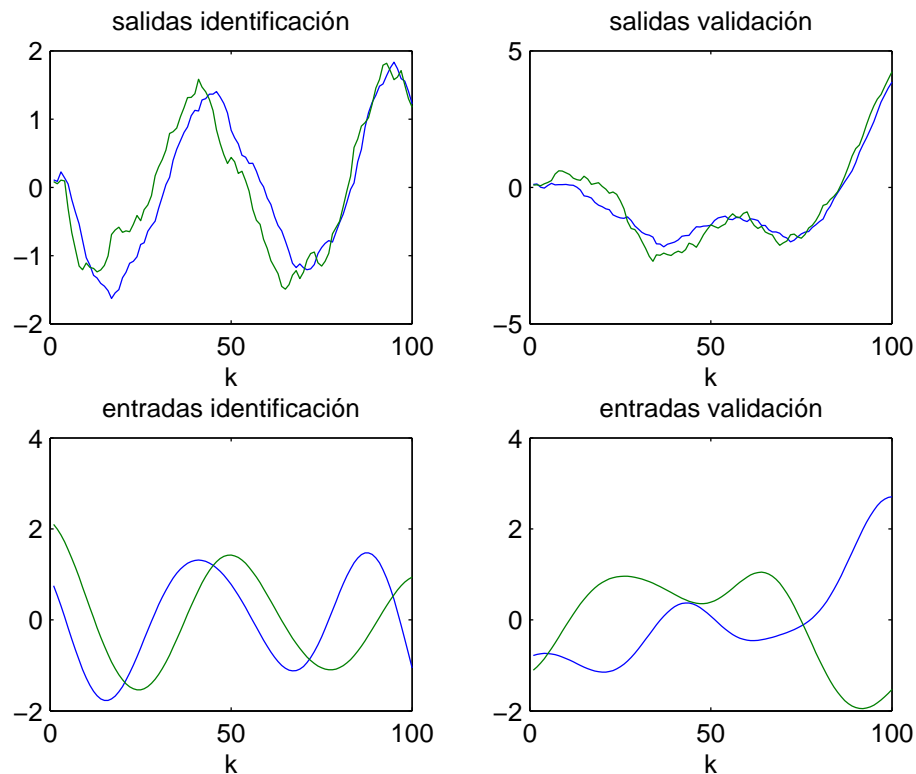


Figura 5.6: Datos de los experimentos en lazo abierto

Se asume la siguiente estructura *Box-Jenkins* para aproximar el proceso, donde todos los polinomios son de orden 2:

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{B_{1,1}}{A_1} & \frac{B_{2,1}}{A_1} \\ \frac{B_{1,2}}{A_2} & \frac{B_{2,2}}{A_2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} \frac{C_1}{D_1} \\ \frac{C_2}{D_2} \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \quad (5.3)$$

Transformando el anterior proceso de matriz de funciones de transferencia *Box-Jenkins* a la forma general de los modelos lineales se obtienen los siguientes órdenes:

$$n_{\bar{b}} = 6; \quad n_{\bar{a}} = 6; \quad n_{\bar{c}} = 6$$

Se estima $\hat{\theta}$ mediante las distintas técnicas a comparar, utilizando las matrices X e Y para las técnicas de regresión lineal y \mathbf{U} e \mathbf{Y} en las técnicas basadas en funciones de transferencia de MATLAB[®].

- Técnicas de regresión lineal:
 - LS
 - PLS: Se utilizan 3 componentes principales de 24. La figura 5.7 muestra la evolución del índice J (RMSECV) utilizado para tomar la decisión
 - PCR: Se utilizan 3 componentes principales de 24. La figura 5.8 muestra la evolución del índice J (RMSECV) utilizado para tomar la decisión
 - KPLS: Se utilizan 3 componentes principales de 24 y el valor de k para el modelo obtenido con X e Y completas es $k = [0 \ 0 \ 0]$, luego para el experimento de validación externa KPLS es exactamente igual a PLS, no obstante durante la validación interna se han obtenido vectores k no nulos, es por esto que PLS y KPLS no coinciden en la gráfica Q^2
 - PLSA: $pesos = 3$
 - DPLS: Se obtienen dos modelos; DPLS utilizando un componente principal y DPLS2 utilizando 2 componentes principales
- Técnicas basadas en modelos en fdt de MATLAB[®]:
 - ARX: $n_a = 6$ y $n_b = 6$
 - ARMAX: Se obtiene un modelo MISO para cada salida con $n_a = 6$, $n_b = 6$ y $n_c = 2$
 - OE: Se obtiene un modelo MISO para cada salida con $n_a = 2$ y $n_b = 2$
 - BJ: Se obtiene un modelo MISO para cada salida con $n_a = 2$, $n_b = 2$, $n_c = 2$ y $n_d = 2$
 - IV: Se obtiene un modelo MISO para cada salida con $n_a = 6$ y $n_b = 6$

En la figura 5.9 se observa el resultado de validación obtenido y en la figura 5.10 se observa el mismo resultado, pero dejando sólo las gráficas asociadas a PLS, PCR, KPLS y PLSA. A continuación se comenta el contenido de cada una de las 4 gráficas de ambas figuras:

■ Validación Interna LMOCV:

- std^6 : Como en este caso hay 2 salidas, se trazan dos líneas por cada método, cada una asociada a la desviación estándar de los parámetros relacionados con una salida. Se observa que las técnicas basadas en función de transferencia de MATLAB[®], presentan desviaciones estándar del orden de 100 veces mayor que las desviaciones estándar obtenidas con los métodos basados en la reducción de variables, principalmente el método BJ⁷.
- Q^2 : En este caso se dispone de 2 salidas, luego $n_f \times n_s = 15 \times 2 = 30$ valores de Q^2 que se representan en una sola línea, donde los primeros 15 elementos de cada línea en la gráfica Q^2 están referidos a la salida y_1 y del 16 al 30 a y_2 . Se observa que las técnicas basadas en la reducción de variables son las únicas que presentan resultados aceptables, siendo PCR y PLSA las que mejor modelo proporcionan según Q^2 . Respecto a DPLS se observa que cuando sólo se considera un componente principal el resultado es mejor que si se consideran los dos componentes principales, no obstante la limitación a la hora de aproximar los polos, hace que en este caso el resultado obtenido sea peor que por el resto de técnicas basadas en la reducción de variables.
- $\text{max}(e)$: De forma análoga a Q^2 se dispone $\text{max}(e)$, mostrando el error de y_1 en los primeros $n_f = 15$ elementos y el error de y_2 en los elementos del 16 al 30. Se observa que en validación interna y para predicciones lejanas del instante k , el método PLSA es el que proporciona menor error máximo. No obstante para predicciones cercanas a k , DPLS utilizando un sólo componente principal de dos, es el que proporciona menor error máximo.

⁶Es importante tener en cuenta que los términos cuya varianza asociada es cero para las técnicas basadas en función de transferencia, es debido a las submatrices cero de la matriz \tilde{A} , ecuación 3.14, no ocurre lo mismo en las técnicas basadas en la reducción de variables debido a que se estiman todos los elementos de $\hat{\theta}$, luego no hay elementos con valor fijo cero.

⁷Las técnicas BJ, OE y ARMAX se basan en optimización no lineal, luego pueden presentar problemas de convergencia o dar como resultado mínimos locales, lo que tiene un efecto directo en la desviación estándar de los parámetros obtenidos con estas técnicas.

■ Validación Externa:

- R^2 : Se observa similitud con Q^2 para los métodos basados en la reducción de variables, lo cual refuerza el uso de estos.

Los conjuntos de datos se han generado de forma que contienen muy poca información del sistema, no obstante si se repite el experimento pero se utilizan señales de entrada más excitantes, ruido gaussiano filtrado en la ventana $[0,1 \dots 0,2]$, con 500 muestras en lugar de 100, y se repite todo el proceso, seleccionando esta vez 4 componentes principales para PLS y 5 para PCR en lugar de 3, se obtiene la gráfica de validación de la figura 5.11. Se comprueba que en estas condiciones LS y ARX son capaces de obtener modelos mejores que las técnicas basadas en la reducción de variables, aunque las estimaciones siguen presentando valores muy altos de desviación estándar. De forma general y conforme aumenta el número de variables a considerar en el modelo, se tiende a una situación más similar a la primera abordada en este apartado donde la falta de información hace necesario el uso de técnicas basadas en la reducción de variables.

Conclusiones

Se observa que los métodos basados en la reducción de variables presentan un resultado aceptable ante conjuntos de datos con poca información del proceso, y la desviación estándar de los parámetros es mucho menor que con LS o el resto de técnicas disponibles en MATLAB®.

De las aportaciones realizadas en el presente trabajo, PLSA y KPLS, ambas presentan un buen comportamiento mejorando, en algunos casos, a la técnica original PLS.

Respecto a DPLS se verifican las limitaciones a la hora de aproximar la dinámica del proceso, pero cabe recordar la ventaja que aportan debido a la reducción de problemas MIMO en SISO.

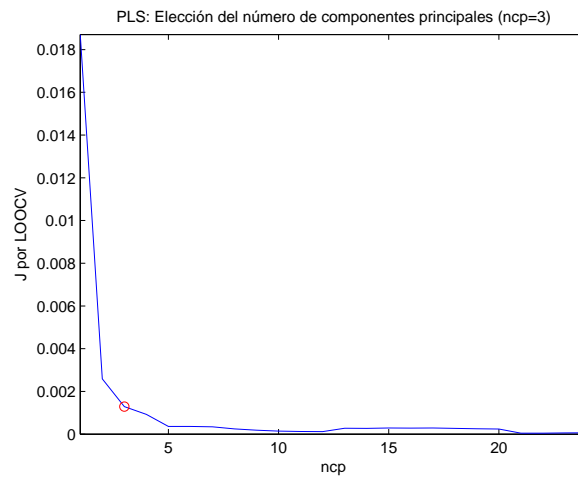


Figura 5.7: Selección del número de componentes principales PLS

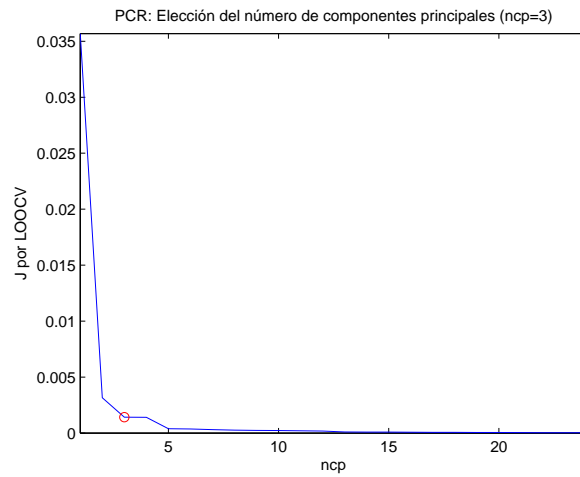


Figura 5.8: Selección del número de componentes principales PCR

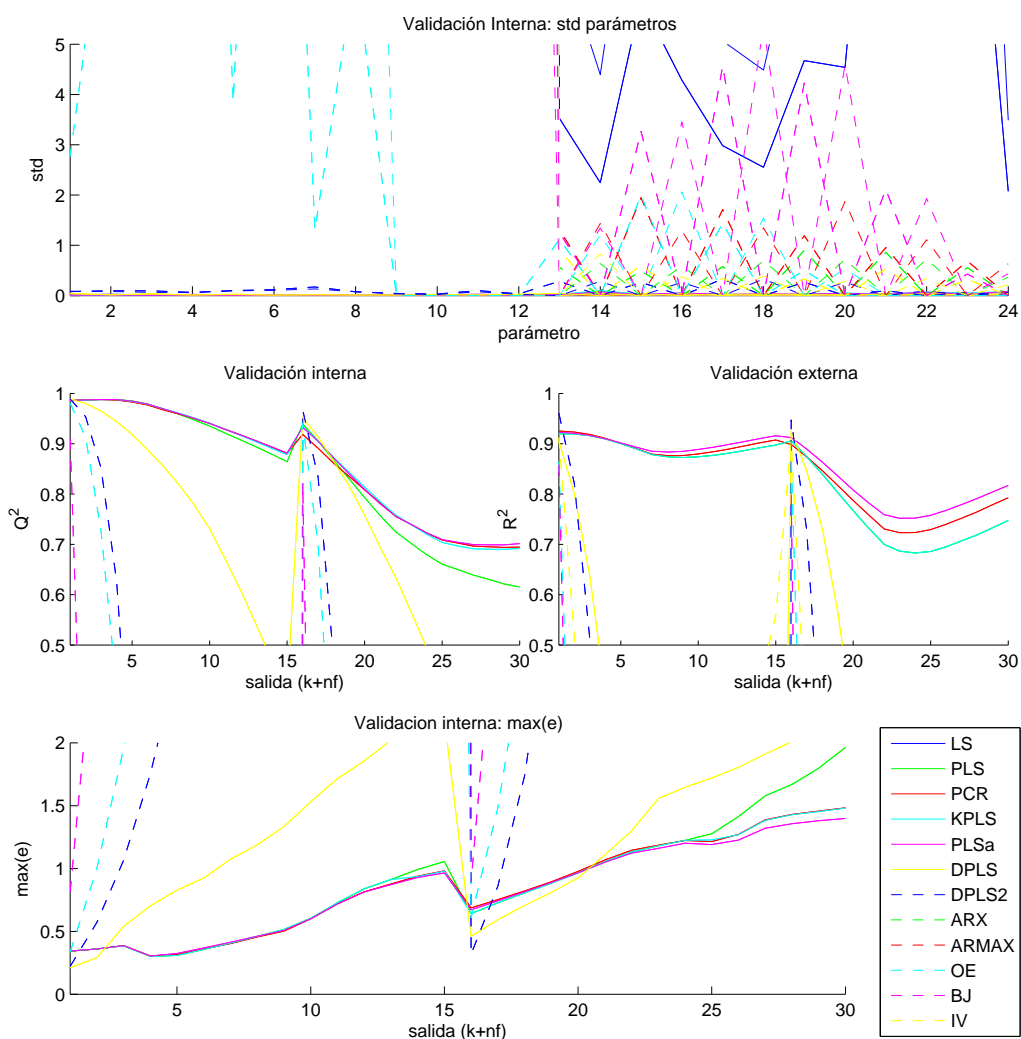


Figura 5.9: Gráficas de validación

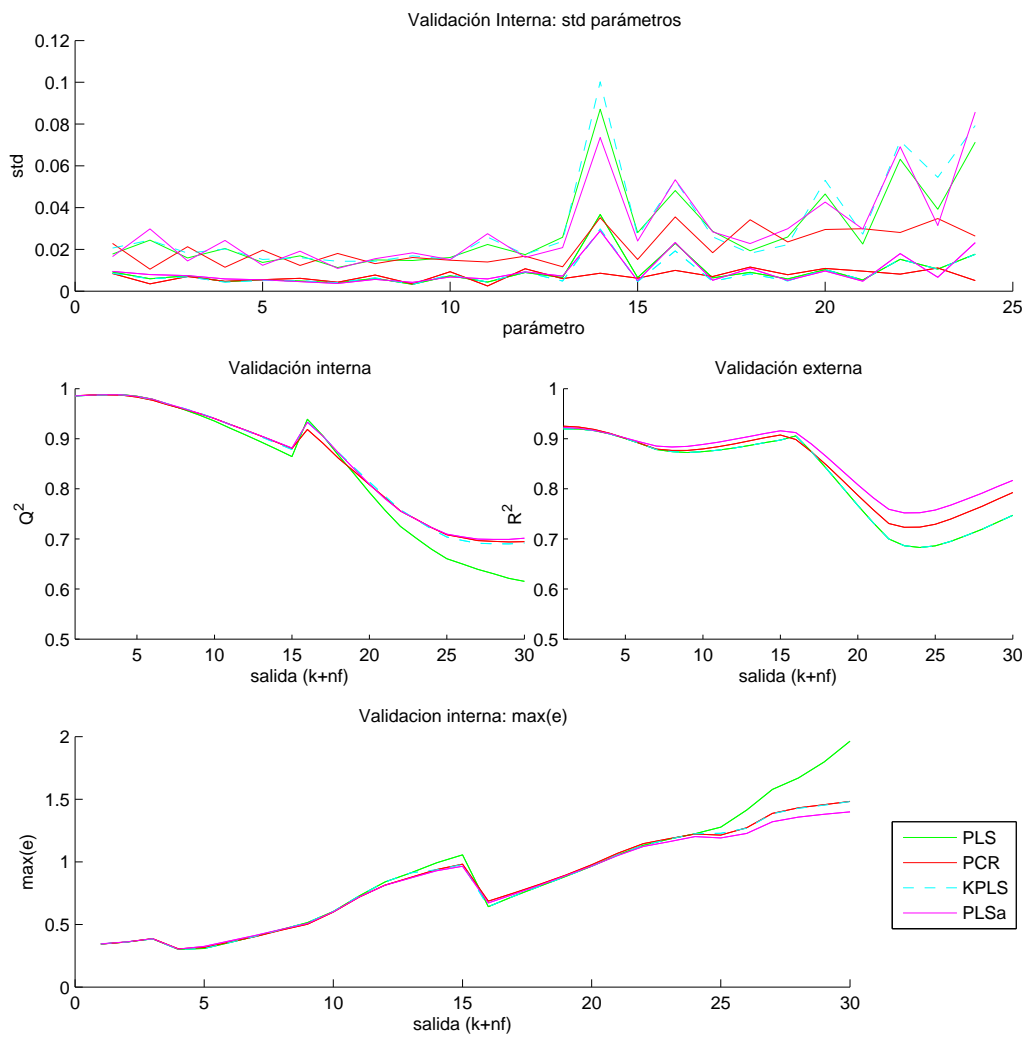


Figura 5.10: Gráficas de validación eliminando algunos métodos

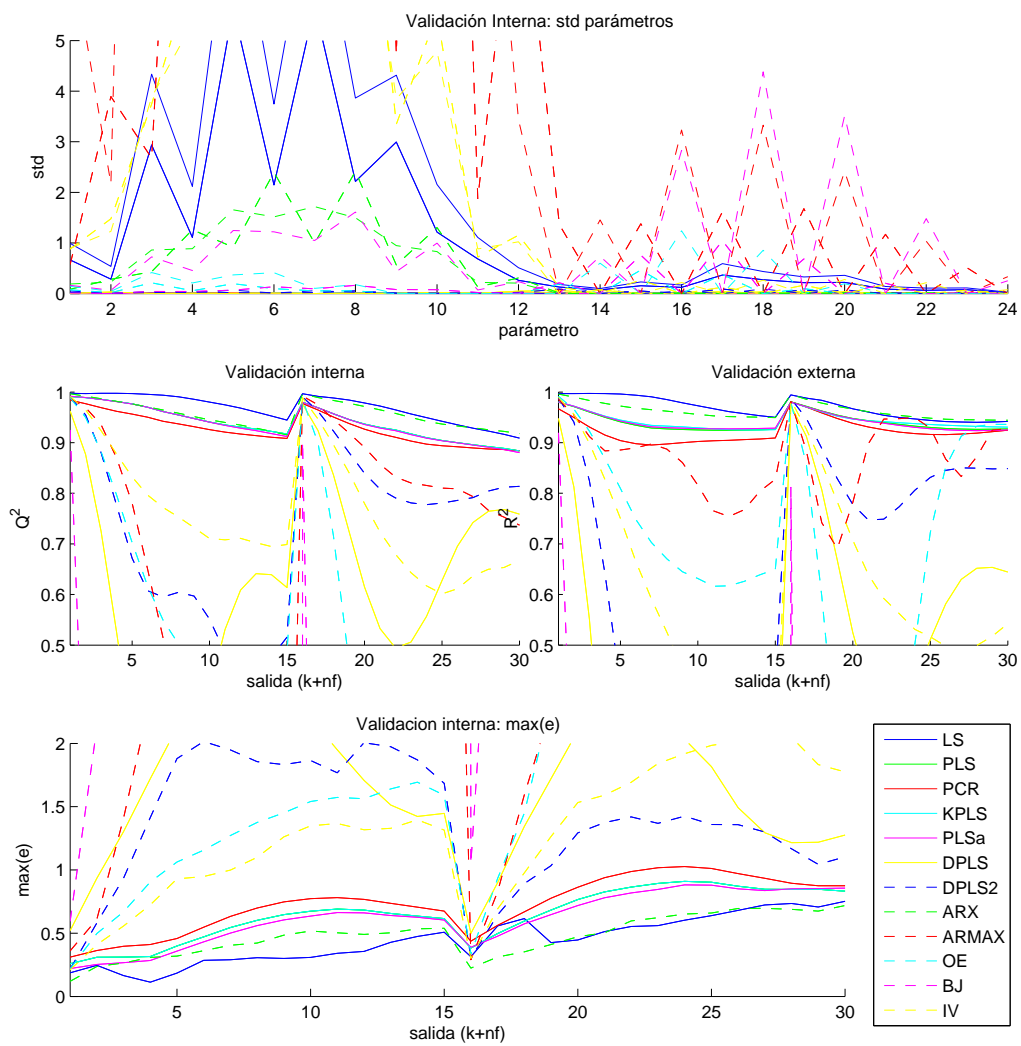


Figura 5.11: Gráficas de validación con \underline{U} más excitantes

Capítulo 6

Conclusiones y futuros trabajos

Se ha analizado el estado del arte de las técnicas de identificación de modelos lineales para procesos MIMO donde la carencia de información del sistema en los datos de identificación dificulta el proceso de estimación de los parámetros. A continuación se exponen las conclusiones más importantes así como los futuros trabajos en la línea de la presente tesina.

Conclusiones:

- Ante conjuntos de datos poco excitantes, es necesario utilizar técnicas basadas en la reducción de variables para estimar los parámetros del modelo lineal de forma consistente.
- Las herramientas disponibles en MATLAB[®] para la identificación de modelos lineales proporcionan estimaciones con una desviación estándar mucho mayor que la obtenida utilizando técnicas basadas en la reducción de variables, sobretodo ante conjuntos de datos de identificación poco excitantes.
- Se pueden realizar aportaciones en el campo de técnicas de estimación de parámetros basadas en la reducción de variables enfocando el problema desde el punto de vista de modelos útiles en MPC.
- Desde el punto de vista de control es muy interesante la utilización del modelo obtenido en DPLS ya que se puede desacoplar un proceso MIMO en procesos SISO, no obstante DPLS presenta limitaciones a la hora de aproximar el comportamiento dinámico del proceso, luego sólo será conveniente utilizarlo si los polos de las funciones de transferencia presentes en G son similares.

Futuros trabajos:

- Considerar la submatriz \tilde{C} en los modelos de predicción para comparar los distintos métodos considerando todos los parámetros del modelo.
- Profundizar en la elección de la función de ponderación utilizada en PLSA para obtener mejores resultados, así como la estimación de \tilde{C} asociada al error en previas estimaciones.
- Aplicar la idea utilizada en PLSA sobre PCR generando el método PCRA.
- Añadir el resultado obtenido con otras técnicas de identificación, como la identificación por los métodos basados en subespacios.
- Comparar KPLS con *Continuum Regression* y profundizar en las distintas técnicas para seleccionar k.
- La identificación en bucle cerrado presenta muchas ventajas en un entorno industrial [Zhu02], queda como trabajo futuro comparar los métodos con datos obtenidos en bucle cerrado.
- Realizar la comparativa de los métodos ante datos de procesos industriales reales obtenidos en bucle cerrado donde aparecen no linealidades y la estructura suele no ser conocida.
- Buscar soluciones en DPLS para superar la limitación que presenta desde el punto de vista de identificación.

Anexos

6.1. Relación señal ruido

La relación señal ruido es la relación entre la potencia de la señal y la potencia del ruido, para la salida s se obtiene mediante la siguiente fórmula:

$$SNR_s = 10 \log_{10} \left(\frac{P_{y_s}}{P_{v_s}} \right) = 20 \log_{10} \left(\frac{\sqrt{E\{y_s \cdot y_s^*\}}}{\sqrt{E\{v_s \cdot v_s^*\}}} \right) \quad \forall s \in [1 \dots n_s]$$

Con el objetivo de relacionar visualmente el orden de magnitud de las señales de ruido y la salida con el valor del ratio SNR, se dispone la figura 6.1. Se observa una señal de salida y y dos señales de ruido, v que presenta un SNR de 4.7 dB respecto de la señal de salida y $v2$ que presenta un SNR de 9.3 dB respecto de la señal y .

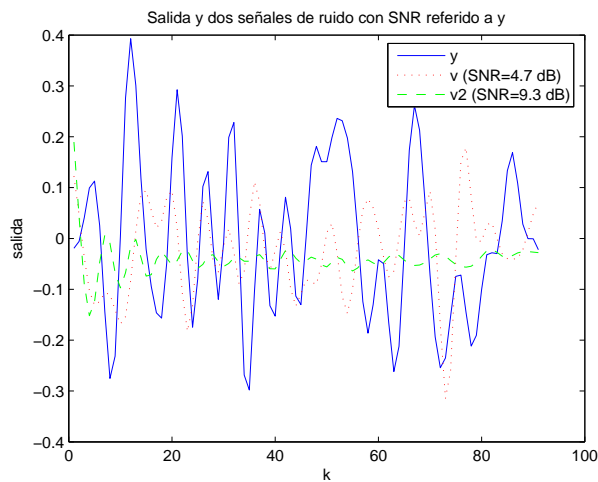


Figura 6.1: *Ratio señal-ruido (SNR)*

Bibliografía

- [Abr03] Abrahamsson, Christoffer, et al., *Comparison of different variable selection methods conducted on NIR transmission measurements on intact tablets*, Chemometrics and Intelligent Laboratory Systems **69** (2003), 3–12. 4.7, 54
- [Agu02] Aguado, Alberto y Martínez, Miguel, *Identificación y control adaptativo*, Prentice Hall, 2002. 3
- [Ble02] Blet, N., et al., *Non linear MPC versus MPC using on-line linearisation - a comparative study*, IFAC, 2002. 1
- [Box97] Box, George and Luceño, Albeto, *Statistical control by monitoring and feedback adjustment*, Wiley series in probability and statistics, 1997. 4
- [Cen96] Centner, V., et al., *Elimination of uninformative variables for multivariate calibration*, Analytical Chemistry **68** (1996), no. 21, 3851–3858. 2.4, 54
- [Gel86] Geladi, Paul and Kowalski, Bruce R., *Partial least-squares regressions: a tutorial*, Analytica Chimica Acta **185** (1986), 1–17. 2.4, 4.3.2, 40, 4.3.3, 4.4, 44
- [Har01] Harald Martens, *Reliable and relevant modelling of real world data: a personal account of the development of PLS Regression*, Chemometrics and intelligent laboratory systems **58** (2001), 85–95. 4.4, 44
- [Has97] Hasegawa, K., et al., *GA strategy for variable selection in QSAR studies: GA-based PLS analysis of calcium channel antagonists*, Journal of Chemical Information and Computer Sciences **37** (1997), no. 2, 306–310. 54
- [Hös88] Höskuldsson, Agnar, *PLS regression methods*, Journal of Chemometrics **2** (1988), no. 3, 211–228. 39, 4.4
- [Iko02] Ikonen, Enso and Najim, Kaddour, *Advanced process identification and control*, Marcel Dekker, Inc., 2002. 2.1, 11, 3.2, 12

- [Ise92] Isermann, R., et al., *Adaptive control systems*, Prentice Hall, 1992. 4.2
- [Lak97] Lakshminarayanan, S., et al., *Modeling and control of multivariable processes: Dynamic PLS approach*, *AIChE* **43** (1997), no. 9, 2307–2322. 4.6, 49
- [Lea00] Leardi, Riccardo, *Application of genetic algorithm-PLS for feature selection in spectral data sets*, *Journal of Chemometrics* **14** (2000), no. 5-6, 643–655. 2.4, 4.7
- [Lii07] Liisa, Sirkka and Jounela, Jämsä, *Future trends in process automation*, *Annual Reviews in Control* **31** (2007), 211–220. 1
- [Lju99] Ljung, Lennart, *System identification. Theory for the user*, Prentice Hall, 1999. 3, 5, 12, 3.1, 9
- [Mor99] Morari, Manfred and Lee, Jay H., *Model predictive control: past, present and future*, *Computers & chemical engineering* **23** (1999), 667–682. 1
- [Mor03] Moreno, Luis, et al., *Ingeniería de control, modelado y control de sistemas dinámicos*, Ariel ciencia, 2003. 3, 1
- [Oll91] Ollero, Anibal, *Control por computador. Descripción interna y diseño óptimo*, Boixareu editores, 1991. 2.1, 5, 4.1, 4.2
- [Pat98] Patwardhan, Rohit S., et al., *Constrained nonlinear MPC using hammerstein and wiener models: PLS framework*, *AIChE* **44** (1998), no. 7, 1611–1622. 4.6
- [Ram07] Ramos, César, *Control predictivo basado en modelos, robusto con BDU (bounded data uncertainties)*, Ph.D. thesis, UPV, 2007. 38
- [Sal06] Salvador, García, et al., *Optimization of batch operating policies. part I. handling multiple solutions*, *Industrial & Engineering Chemistry Research* **45** (2006), no. 23, 7856–7866. 19
- [Ser05] Serneels, Sven, et al., *Robust continuum regression*, *Chemometrics and intelligent laboratory systems* **76** (2005), 197–204. 4.5, 45
- [Smi04] Smilde, Age, et al., *Multi-way analysis applications in the chemical sciences*, Wiley, 2004. 10
- [Thw07] Thwaites, Philip, *Process control in metallurgical plants-from an Xstrata perspective*, *Annual Reviews in Control* **31** (2007), 211–239. 1

- [Tro03] Tropsha, Alexander, et al., *The importance of being earnest: Validation is the absolute essential for successful application and interpretation of QSPR models*, QSAR & Combinatorial Science **22** (2003), no. 1, 69–77. 2.6
- [Weh00] Wehrens, Ron, et al., *The bootstrap: a tutorial*, Chemometrics and Intelligent Laboratory Systems **54** (2000), 35–52. 17
- [Wol01] Wold Svante, et al., *PLS-regression: a basic tool of chemometrics*, Chemometrics and intelligent laboratory systems **58** (2001), 109–130. 7, 4.4
- [Yu,05] Yu, Tina, et al., *Genetic programming theory and practice III (genetic programming)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005. 4.3.1
- [Zan08] Zang, C., et al., *Model validation for structural dynamic analysis: An approach...*, Comput. Methods Appl. Mech. Engrg. **197** (2008), 2645–2659. 2.6
- [Zhu02] Zhu, Yucai and Butoyi Firmin, *Case studies on closed-loop identification for MPC*, Control engineering practice **10** (2002), 403–417. 1, 5, 6