UNIVERSITÀ DEGLI STUDI DI PADOVA
FACOLTÀ DI INGEGNERIA

—

LAUREA MAGISTRALE IN INGEGNERIA DELL'AUTOMAZIONE
Progettazione di sistemi di controllo

IDENTIFICAZIONE DI MODELLI TERMODINAMICI

Prof. Schenato Luca
Prof. Chiuso Alessando

ALBERTON RICCARDO, AUSSERER MARKUS, BARAZZUOL ANDREA

ANNO ACCADEMICO 2010-2011

Indice

Introduzione

0.1 Pem: Prediction Error Methods

In questo capitolo viene spiegato brevemente il principio di funzionamento dei metodi a minimizzazione dell'errore di predizione.
Dato un modello $M(\theta)$ appartenente a una certa classe parametrica $\mathcal{M} \equiv \{M(\theta); \theta \in \Theta\}$ e dato una sequenza di N misure dei segnali d'ingresso e d'uscita

$$u^N := \{u(t); t = 1, \dots, N\} \qquad y^N := \{y(t); t = 1, \dots, N\}$$

si procede nel modo seguente:

1. Per un qualche valore di θ fissato si costruisce il miglior (secondo qualche criterio) predittore all'istante $t-1$ dell'uscita $y(t)$. Questo predittore è una funzione dei dati passati che viene denotata col simbolo $\hat{M}(\theta)$:

$$\hat{M}(\theta) : (y^{t-1}, u^{t-1}) \to \hat{y}_\theta(t|t-1).$$

   La predizione $\hat{y}_\theta(t|t-1)$ si può pensare come funzione dei dati passati e quindi come una funzione aleatoria che viene indicata usando il simbolo in grassetto $\hat{\mathbf{y}}_\theta(t|t-1)$.

2. Si formano gli errori di predizione

$$\varepsilon_\theta(t) := y(t) - \hat{y}_\theta(t) \qquad \text{per } t = 1, \dots, N.$$

   Analogamente a quanto detto nel caso del predittore anche gli errori di predizione possono essere visti come quantità aleatorie e vengono quindi indicate col simbolo in grassetto $\varepsilon_\theta(t)$.

3. Si minimizza rispetto a θ una cifra di merito che descrive quanto bene il modello predice il dato successivo. Ad esempio si minimizza l'errore quadratico medio di predizione

$$V_N(\theta) := \frac{1}{N} \sum_{t=1}^{N} \varepsilon_\theta(t)^2$$

   o più in generale una media degli errori quadratici di predizione pesati con una cifra di merito non negativa $\beta(N, t) > 0$,

$$V_N(\theta) := \frac{1}{N} \sum_{t=1}^{N} \beta(N, t)\varepsilon_\theta(t)^2$$

che per N piccoli dà peso minore agli errori di predizione compiuti nella fase iniziale dell'algoritmo quando l'influenza di condizioni iniziali stimate in modo approssimativo è più deleteria. Per N grande $\beta(N, t)$ tende a essere inutile e quindi deve tendere a uno.
In ogni caso si ricava la stima si θ dalla minimizzazione della cifra di merito

$$\hat{\theta}_N := Arg \min_{\theta} V_N(\theta)$$

che è la stima PEM del parametro del modello. Lo stimatore $\hat{\theta}_N$ viene chiamato stimatore PEM del parametro θ.

4. Infine si prende come stima della varianza dell'innovazione $\lambda^2 = var\{\mathbf{e}(t)\}$ l'errore quadratico residuo

$$\hat{\lambda}^2 := V_N(\hat{\theta}_N).$$

Per quanto questa procedura sembra intuitivamente sensata, l'unica giustificazione per la sua adozione nei procedimenti di identificazione sta nelle sue proprietà statistiche. Per avere una trattazione completa delle proprietà statistiche dello stimatore PEM si consulti [1].

Finora si è visto come ricavare lo stimatore PEM nel caso si ha un parametro θ fissato. Poiché in generale però non si ha alcuna informazioni sulla struttura del sistema vero serve uno strumento per scegliere la classe parametrica $\mathcal{M}$ alla quale appartiene il modello identificato $M(\theta)$. Dato un set di misure e diverse classi di modelli con ordini p plausibili si calcolano degli indici che danno la possibilità di scegliere la classe che promette di ottenere un modello identificato con buone prestazioni. Questi indici sono sostanzialmente una generalizzazione del criterio FPE (Final Prediction Error, vedere [1] per la trattazione completa). Nelle simulazioni svolte per questo elaborato sono stati usati l'AICC (Akaike Information Criterion with a second-order Correction) e il BIC (Bayesian Information Criterion) definito dalle seguenti formule:

$$AICC = N \cdot \ln \frac{rss}{N} + N \cdot \frac{N+p}{N-p-2}$$
$$BIC = N \cdot \ln \frac{rss}{N} + p \cdot \ln(N)$$

dove rss (Residual Sum of Squares) è definito da $rss = \sum_{t=1}^{N} \varepsilon_\theta(t)^2$, mentre N è il numero di misure e p è l'ordine del parametro θ del modello.
Nel progetto in esame è stata implementata una routine nella quale viene considerato la classe dei modelli ARMAX. In un ciclo ad ogni passo viene aumentato l'ordine del parametro θ e vengono calcolati i valori dei due indici. Terminato il ciclo si controlla in relazione a quale ordine gli indici hanno raggiunto i valori minimi. Viene quindi rifatta l'identificazione PEM usando i rispettivi ordini ritenuti ottimi ottenendo due modelli che spesso si verificano aver ordine differente. Si noti però che per i due criteri considerati all'aumentare di N la probabilità che la scelta dell'ordine sia sbagliato tende a zero (e quindi si dice che la selezione dell'ordine è consistente).

0.2 Definizione del problema

Sia $\{z_t\}_{t\in\mathbb{Z}}\, z_t \in \mathbb{R}^m$ un processo stocastico, stazionario che descrive l'evoluzione temporale di qualche variabile di interesse. Con un piccolo abuso di notazione in questo contesto z_t indica sia la variabile aleatoria che la sua realizzazione. Come mostrato in figure 1 ogni componente del processo $\{z_t\}_{t\in\mathbb{Z}}$ può essere pensato come un nodo di una grafo. L'obiettivo di questi algoritmi di identificazione è di trovare modelli lineari dinamici che descrivono tutti i componenti di $\{z_t\}$ come funzione degli altri componenti. A questo scopo si definisce $y_t := z_t^{[i]}$ (la componente i-esima di z_t) come uscita e tutte le altre componenti $u_t := z_t^{[-i]} \in \mathbb{R}^{m-1}$ come ingressi. Ovviamente il procedimento può essere ripetuto per ogni $i = 1, \ldots, m$ ottenendo in questo modo una descrizione completa di tutte le variabili di z_t come funzione degli altri. In questa sezione però per questioni di chiarezza si sceglie un valore specifica per i, per esempio $i = 1$. In questo caso z_t può essere rappresentato come

$$z_t = \begin{bmatrix} y_t \\ u_t \end{bmatrix}$$

Si definiscono gli insiemi delle variabili passate all'istante t

$$Y^t = [y_{t-1}, yt-2, \ldots] \quad U^t = [u_{t-1}, ut-2, \ldots]$$

Dalla stazionarietà del processo $\{z_t\}_{t\in\mathbb{Z}}$ segue che anche i processi $\{y_t\}_{t\in\mathbb{Z}}$ e $\{u_t\}_{t\in\mathbb{Z}}$ sono processi stocastici, congiuntamente stazionari che possono quindi essere pensati rispettivamente come ingressi e uscita di un sistema dinamico tempo-invariante

$$y_t = \sum_{k=1}^{\infty} f_k u_{t-k} + \sum_{k=0}^{\infty} g_k e_{t-k} \tag{1}$$

dove $f_k \in \mathbb{R}^{1\times m}$ (avendo assunto assenza di retroazione diretta, cioè $f_0 = 0$) e $g_k \in \mathbb{R}$ sono coefficienti incogniti della risposta impulsiva e e_t rappresenta un processo di innovazione che è equivalente all'errore di predizione di un passo

$$e_t := y_t - \hat{y}_{t|t-1} = y_t - E[y_t|Y^t, U^t]$$

$$E[y_t|Y^t, U^t] := \sum_{j=1}^{m-1}\left[\sum_{k=1}^{\infty} h_k^j u_{t-k}^{[j]}\right] + \sum_{k=1}^{\infty} h_k^{[m]} y_{t-k}.$$

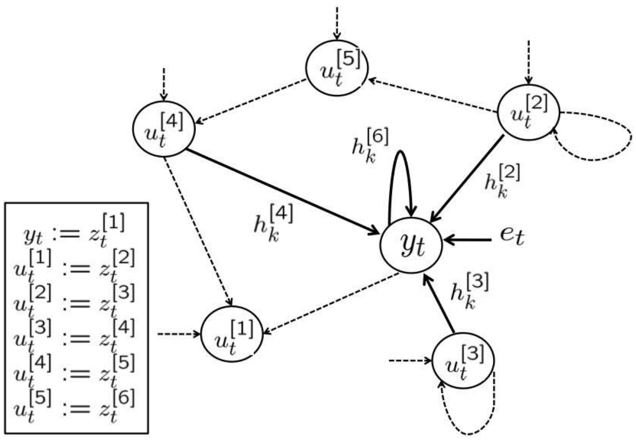

Figura 1: Grafo che rappresenta l'interconnessione di $m = 6$ variabili. Gli archi solidi rappresentano le connessioni nel modello dinamico con il nodo $y_t := z_t^{[1]}$ da tutti gli altri nodi. L'assenza degli archi da $u - t^{[i]},\ i = 1, 5$ a y_t indica che $h_k^{[i]} = 0,\ i = 1, 5,\ \forall k \in \mathbb{Z}^+$. Il nodo y_t ha l'arco entrante indicando la dipendenza dall'errore di predizione a un passo e_t. Gli archi punteggiati indicano dipendenze tra gli ingressi u_t.

Le sequenze $h_k := [h_k^{[1]}, h_k^{[2]}, \ldots, h_k^{[m]}] \in \mathbb{R}^{1\times m}, k \in \mathbb{Z}^+$ rappresentano i coefficienti della risposta impulsiva del predittore e servono a descrivere sistemi (BIBO) stabili e si ha quindi $h^{[m]} \in l_1(\mathbb{Z}^+)$.

Nell'ambito della PEM la predizione di sistemi dinamici può essere vista come predizione dei coefficienti della risposta impulsiva h_k del predittore. Si vuole prestare particolare attenzione a situazioni in cui non tutte le variabili sono necessari per predire y_t, il che in termini matematici equivale ad avere $h_k^{[i]} = 0$ per ogni $k \in \mathbb{Z}^+$. In una rappresentazione grafica c'è una arco diretto $u_k^{[i]}$ al nodo y_k se e solo se esiste $k \in \mathbb{Z}^+$ tale che $h_k^{[i]} \neq 0$ per $i = 1, \ldots, m-1$; inoltre non c'è un capo se e solo se esiste $k \in \mathbb{Z}^+$ tale che $h_k^{[m]} \neq 0$. Per esempio per il grafo rappresentato in figura 1 si ha $h_k^{[5]} = h_k^{[1]} = 0$, $\forall k \in \mathbb{Z}+$, mentre $h_k^{[2]} = h_k^{[3]} = h_k^{[4]}$ e $h_k^{[6]}$ non sono identicamente nulle. Questo significa che per predire y_t serve il passato di $u^{[2]},\ u^{[3]},\ u^{[4]}$ e di y stesso.

In pratica non si sa a priori quali variabili saranno significativi per predire la uscita y_t. I metodi PEM e N4SID non effettuano alcuna selezione di ingressi e stimano un modello "completo" che tiene conto di tutti gli ingressi. Questo è un motivo per cui questi due algoritmi a volte producono stime scarse.

Qui di seguito si presentano degli algoritmi che favoriscono la sparsità e che sono quindi capaci di individuare una struttura del sistema dinamico simile a quella riportata in figura 1. Allo stesso momento riescono a stimare le risposte impulsive $h_k^{[i]}$ nulle.

0.3 Predizione in RKHS e approssimazione finito-dimensionale

Si assume che le risposte impulsive $h^{[i]}$ siano (le versioni campionate) di funzioni in $\mathcal{H}_K$, lo spazio di Hilbert riproduttivo associato al nucleo campionato K. Questo corrisponde a dire che le risposte impulsive $h^{[i]}$ sono processi Gaussiani con funzione di covarianza K.
Il problema di stimare le risposte impulsive $h^{[i]}$ da dati misurati può essere formulato come il seguente problema di regolarizzazione di tipo Tikhonov:

$$\{\hat{h}^{[i]}\} = \arg \min_{h^{[i]} \in \mathcal{H}_K} \sum_{t=t_0}^{N} (y_t - \hat{y}_{t|t-1})^2 + \gamma_2^2 \left(\sum_{i=1}^{m} \|h^{[i]}\|_{\mathcal{H}_K}^2 \right) \tag{2}$$

soggetto a

$$\hat{y}_{t|t-1} = \sum_{i=1}^{m-1} \left[\sum_{k=1}^{\infty} h_k^{[i]} u_{t-k}^{[i]} \right] + \sum_{k=1}^{\infty} h_k^{[m]} y_{t-k}. \tag{3}$$

Il parametro γ_2 è il cosiddetto parametro di regolarizzazione che deve regolare il trade-off tra fit e regolarità di $h^{[i]}$.

Per portare questo problema in dimensione finita si modifica il nucleo K in modo che la realizzazione del nucleo modificato faccia soddisfare la risposta impulsiva alla condizione $h_k^{[i]} = 0, \ \forall k > J$. È facile vedere che il nucleo (la covarianza, indicata con K_J) che soddisfa la condizione è $K_J(h,k) = 0, \ \forall (h,k) : \ k > J$ oppure $h > J$.

Raccogliendo i coefficienti della risposta impulsiva in un vettore colonna si ha

$$\underline{h}^{[i]} := [h_1^{[i]}, h_2^{[i]}, \ldots, h_{t_0}^{[i]}]^T.$$

Il numero J non ha influenza sul trade-off tra bias e varianza, ma serve solo per ragioni di complessità computazionale. Con un piccolo abuso di notazione si indica con $K_J \in \mathbb{R}^{j \times j}$ la matrice simmetrica, definita positiva formato dai nuclei campionati fino al ritardo J. Sotto queste condizione la norma $\|h^{[i]}\|_{\mathcal{H}_{K_J}}^2$ può essere scritta come

$$\|h^{[i]}\|_{\mathcal{H}_{K_J}}^2 = (\underline{h}^{[i]})^T K_J^{-1} \underline{h}^{[i]}.$$

Il predittore a un passo $\hat{y}_{t|t-1}$ diventa

$$\hat{y}_{t|t-1} = \sum_{i=1}^{m-1} \left[\sum_{k=1}^{J} h_k^{[i]} u_{t-k}^{[i]} \right] + \sum_{k=1}^{J} h_k^{[m]} y_{t-k}. \tag{4}$$

Risulta che sotto la restrizione $h^{[i]} \in \mathcal{H}_{K_J}$ la soluzione del problema 2 è

$$\arg \min_{\underline{h}^{[i]} \in \mathbb{R}^{t_0}} \sum_{t=t_0}^{N} (y_t - \hat{y}_{t|t-1})^2 + \gamma_2^2 \sum_{i=1}^{m} \left[(\underline{h}^{[i]})^T K_J^{-1} \underline{h}^{[i]} \right] \tag{5}$$

soggetto a 4. Si vede quindi che il problema può essere riformulato come quello di stimare un modello ARX con un peso di regolarizzazione incluso nel problema della stima.

0.4 Stabil Spline Group LAR (SSGLAR) algoritm

Viene ora discusso come si può risolvere il problema 5 invocando sparsità nei gruppi $h_{t_0}^{[i]}$ usando l'algoritmo GLAR. Per fare questo si assume che i parametri θ siano stati fissati; senza alcun'informazione a priori questo parametro viene posto uguale a zero.

Il problema 5 è un problema di regressione lineare con una funzione peso in l_2 che può essere riscritto nella forma

$$\overline{y}_{t_0}^{+} = \sum_{i=1}^{m} \overline{A}_{Ji} h_{t_0}^{[i]} + W \tag{6}$$

avendo definito

$$\overline{y}_{t_0}^{+} := \begin{bmatrix} y_{t_0}^{+} \\ 0_{1\times(t_0 \cdot m)} \end{bmatrix}$$

$$\overline{A}_{Ji} := [A_{Ji} \quad v_i \otimes \Lambda] \quad \Lambda := \gamma_2 K_J^{-1/2}$$

$$v_i := [\underbrace{0\ldots0}_{i-1}\, 1\, \underbrace{0\ldots0}_{m-i}]^T$$

La soluzione degli ingressi come descritto prima può essere eseguito dall'algoritmo GLAR applicato al problema di regressione. Si definisce SS-Glar il seguente algotimo:

0.4.1 Algoritmo SS-Glar

1. stabilire il parametro β del nucleo;

2. stabilire il parametro γ_2 di 5 e formare $\overline{A}_{Ji}$ in 6;

3. stima di $h^{[i]}$ applicando l'algoritmo GLAR a 6;

0.4.2 Stima dei hyperparametri

Prima di poter eseguire l'algoritmo appena visto bisogna fissare i seguenti parametri:

1. la funzione peso γ_2 in l_2 di equazione 6 (regolarità di $[h_1^{[i]}, h_2^{[i]}, \ldots, h_{t_0}^{[i]}]^T$ nello spazio $\mathcal{H}_{K_{t_0}}$);

2. il parametro β del nucleo (velocità di decrescita del nucleo);

3. Il numero di blocchi diversi da zero stimati dall'algoritmo GLAR;

Questi parametri possono essere stimati usando un approccio di validazione come segue: Disponendo delle misure $\{y_t, u_t\}_{t=1,\ldots,N}$ si suddivide l'insieme in due parti. Il primo $\{y_t, u_t\}_{t=1,\ldots,2N/3}$ viene chiamato l'insieme di identificazione, i dati rimanenti l'insieme di validazione. Si esegue l'algoritmo per hyperparametri fissati usando l'insieme di identificazione e successivamente si valuta il modello ottenuto sull'insieme di validazione. Dall'insieme degli hyperparametri ($\beta \in \mathbb{R}^+, \gamma \in \mathbb{R}^+$) si sceglie un insieme con un numero finito (possibilmente piccolo) di alternative, su quale poi deve essere testato l'algoritmo. Inoltre vengono anche testati diversi livelli di sparsità. Alla fine viene scelto il modello identificato che ha dato le miglior prestazioni sul set di validazione in termini di RMS_1 dell'errore di predizione a un passo:

$$RMS_k := \sqrt{\frac{1}{500}\sum_{i=1}^{500}(y_t - \hat{y}_{t|t-k})^2}.$$

Vengono quindi fissati gli hyperparametri e il livello di sparsità con i quali viene rifatto la stima del modello usando l'intero insieme di dati $\{y_t, u_t\}_{t=1,\ldots,N}$.

0.5 Stable Spline with Exponential Hyperparametri (SSEH) Algoritm

In questo sezione il predittore di equazione 3 viene formulato in un contesto Bayesiano assumendo che le risposte impulsive $h^{[i]}$ siano processi Gaussiani con covarianza $\lambda_i K(t,s)$. Serve specificare i hyperparametri β, θ e λ_i e la varianza di rumore σ^2 che descrivono interamente le densità di probabilità delle misure di y e le risposte impulsive dei predittori $h^{[i]}$.

0.5.1 Hyperprobabilità a priori per i hyperparametri

La varianza di rumore σ^2 viene stimata in un passo preliminare usando un modello ARX. Quindi questo parametro viene supposto noto nella stima Bayesiana. I hyperparametri β, θ e λ_i vengono invece modellati come variabili aleatorie indipendenti. β ha una densità di probabilità poco informativa su $\mathbb{R}^+$, mentre θ è distribuita uniformemente. Ogni λ_i è una variabile aleatoria esponenziale con l'inversa della media $\xi \in \mathbb{R}^+$, cioè

$$p(\lambda_i) = \xi exp(-\xi\lambda_i)\chi(\lambda_i \geq 0), \qquad i = 1, \dots, m$$

con χ la funzione indicatrice. ξ può essere interpretato come una variabile aleatoria poco informativa su $\mathbb{R}^+$. Tutti i hyperparametri possono essere raccolti in vettore aleatorio indicato con $\zeta := [\lambda_1, \dots, \lambda_m, \theta_1, \theta_2, \beta, \xi]$.
Si noti che i parametri λ_i, che hanno lo stesso ruolo di σ^2/γ^2 dove γ^2 è definito in 2, possono ora essere tutti differenti, il che aumenta la flessibilità del modello.

0.5.2 Il modello Bayesiano intero

Sia $A_{ti} \in \mathbb{R}^{N\times\infty}$ in modo che per le uscite future $y^+ := y_1^+$ risulta

$$y^+ = \left[\sum_{i=1}^{m} A_{1i} h^{[i]}\right] + e^+,$$

dove $e^+ := e_1^+$.
In pratica $y^- := [y_0, y_{-1}, y-2, \ldots]$ non è mai interamente noto e una soluzione consiste nel porre a zero i componenti non noti. Viene quindi imposto la seguente approssimazione:

$$\begin{aligned} p(y^+, \{h^{[i]}\}, y^-, u|\zeta) &\propto \left[\prod_{t=1}^{N} p(y_t|\{h^{[i]}\}, y_t^-, \zeta, u_t^-)\right] p(y^-, \{h^{[i]}\}, u^-|\zeta) \\ &\approx \left[\prod_{t=1}^{N} p(y_t|\{h^{[i]}\}, y_t^-, \zeta, u_t^-)\right] p(\{h^{[i]}\}|\zeta) p(y^-, u^-) \end{aligned}$$

Il $\propto$ deriva dal assunzione che il predittore di u_t dato il passato di u_t^- e y_{t+1}^- non dipende da ζ, il $\approx$ invece dall'assunzione che il passato u_t^- e y_t^- porta informazione sulla risposta impulsiva del predittore e sui hyperparametri. Questo modello statistico è descritto dalla rete Bayesiana riportata in figura 2 (sinistra).
Come visto sopra, per questioni di complessità computazionale bisogna inoltre troncare la risposta impulsiva $h^{[i]}$ a una certa lunghezza t_0.

0.5.3 Stima degli hyperparametri

Il vettore degli hyperparametri ζ viene stimato minimizzando la probabilità marginale, cioè la densità di y^+, ζ e $\{h^{[i]}\}$. Può essere dimostrato (si guardi [2]) che la stima si ottiene da

$$\hat{\zeta} = arg \min_{\zeta} J(y^+; \zeta) \tag{7}$$

con $\xi, \beta > 0$, $\lambda_i > 0$, $i = 1, \ldots, m$ e J è dato da

$$J(y^+; \zeta) = \frac{1}{2} log(det[2\pi V[y^+]]) + \frac{1}{2}(y^+)^T(V[y^+])y^+ + \xi \sum_{i=1}^{m} \lambda_i - log(\xi) \tag{8}$$

$$V[y^+] = \sigma^2 I_N + \sum_{k=1}^{m} \lambda_i A_{1i} K A_{1i}^T. \tag{9}$$

Una questione molto importante per l'uso pratica dello schema matematico è l'utilizzo di un buon valore iniziale. Sotto viene descritto un schema con il quale si ottiene una soluzione sub ottimale risolvendo il problema di ottimizzazione in $\mathbb{R}^4$ in relazione al modello Bayesiano in figura 2 (destra).

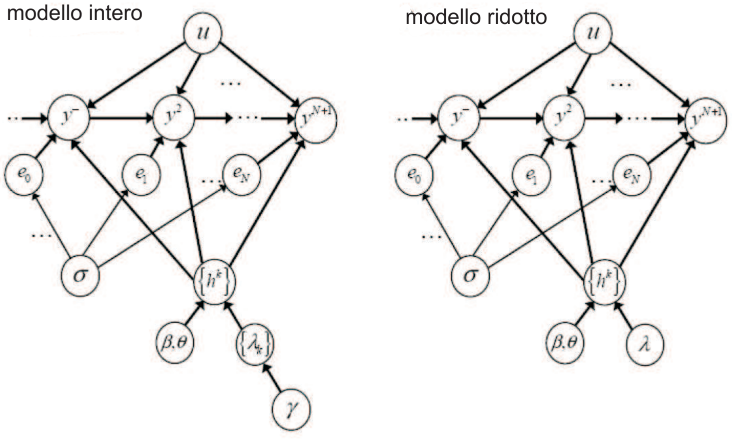

Figura 2: Rete Bayesiana che descrive il nuovo modello non parametrico per identificazione di sistemi dinamici sparsi, dove $y^- := [y_{l-1}, y_{l-2}, \ldots]$ e nel modello ridotto $\lambda := \lambda_1, \ldots, \lambda_m$.

1. Si ottengono $\{\hat{\lambda}_i\}$, $\hat{\theta}$ e $\hat{\beta}$ risolvendo il problema 7 nella versione modificato seguente
$$arg \min_{\zeta}[J(y^+;\zeta) - \xi \sum_{i=1}^{m} \lambda_i + log(\xi)]$$
con $\beta > 0,\ \lambda_1 = \ldots = \lambda_m \geq 0$

2. Si fissano $\hat{\xi} = 1/\hat{\lambda}_1$ e $\hat{\zeta} := [\hat{\lambda}_1, \ldots, \hat{\lambda}_m, \hat{\theta}_1, \hat{\theta}_2, \hat{\beta}, \hat{\xi}]$. Poi, per $i = 1, \ldots, m$ si fissa $\bar{\zeta} = \hat{\zeta}$ tranne che la i-esima componente di $\bar{\zeta}$ che viene messa a zero e se $J(y^+;\bar{\zeta}) \leq J(y^+;\hat{\zeta})$ si fissa $\hat{\zeta} = \bar{\zeta}$.

La procedura descritta al passo 2 potrebbe non essere adatta nel caso in cui gli ingressi sono altamente correlati, perché potrebbe risultare sensitiva dal ordine con cui le componenti di $\hat{\zeta}$ convergono a zero. Per aggirare questo problema si propone di usare l'algoritmo SS-GLAR per effettuare questa selezione. In particolare si può sostituire il passo 2 con il passo seguente:

1. Si fissa $\gamma_2^2 = \sigma^2/\hat{\lambda}_1$ e si aggiusta il nucleo visto in sezione 0.4 con $K(:,:,\hat{\theta},\hat{\beta})$. Si esegue poi l'algoritmo SS-GLAR selezionando il numero di componenti non nulli usando l'approccio di validazione. A questo punto i componenti $\{j_1, \ldots, j_k\} \subseteq \{1, \ldots, m\}$ di $\hat{\zeta}$ vengono annullati, il che equivale a annullare $\hat{h}^{[j]} = 0,\ j \in \{j_1, \ldots, j_k\}$.

Il valore $\hat{\zeta}$ finale può essere usato come punto iniziale in problema 9.

0.5.4 Stima delle risposte impulsive del predittore per ζ noto

Sotto certi condizioni (vedi [2]) si può dimostrare:

$$\{\hat{h^{[i]}}\}_{i=1}^{m} = arg \min_{\{f^i \in \mathcal{H}\}_{i=1}^{m}} \|y^+ - \sum_{i=1}^{m} A_{1if^i}\|^2 + \sigma^2 \sum_{i=1}^{m} \frac{\|f^i\|_{\mathcal{H}_K}^2}{\lambda_i^2}$$

Inoltre per $k = 1, \dots, m+1$ si ha

$$\hat{h^{[i]}} = \lambda_i^2 K A_{1i}^T c, \quad c = \left(\sigma^2 I_n + \sum_{k=1}^{m+1} \lambda_i A_{1i} K A_{1i}^T \right)^{-1} y^+$$

Dopo aver ottenute le stime di $\{h^{[i]}\}$ semplice formule possono essere usate per ottenere le risposte impulsive f e g in 1 e quindi anche il predittore a k passi (per dettagli vedere [**?**]).

0.6 Validazione

Una parte molto importante dell'identificazione è la fase di validazione del modello ottenuto. Si cerca di capire la bontà del modello identificato nel descrivere il processo. L'analisi di prestazione deve essere fatta su dati diversi da quelli usati per l'identificazione. I dati disponibili vengono quindi suddivisi in due insiemi:

- dati per l'identificazione,
- dati per la validazione.

La validazione consiste nel confrontare l'uscita reale con l'uscita del modello identificato avendo come ingressi i dati reali e di calcolare successivamente diversi indici che danno informazione sulla bontà del modello.
Di seguito vengono spiegati i diversi indici significativi alla fine di fare validazione.

Covarianza e correlazione: La covarianza (campionaria) e la correlazione (campionaria) si possono facilmente calcolare in Matlab `cov`(x_1, x_2) e `corr`(x_1, x_2) rispettivamente.
La covarianza indica se esiste una dipendenza tra le due variabili. Valori elevati indicano una forte legame tra le due variabili, mentre per un legame debole la covarianza tende a zero. La covarianza dipende però della base scelta per rappresentare il modello (per esempio dipende dalla unità di misura utilizzata) e non è dunque un indice oggettivo per confrontare la bontà di diversi modelli Per migliori chiarimenti vedere [1]. La correlazione è invece un parametro adimensionale compreso tra -1 e 1 che indica la relazione lineare tra due variabili. Valori negativi della correlazione indicano che le due variabili sono inversamente correlati, cioè se una variabile cresce l'altra tende a decrescere. Due variabili indipendenti hanno correlazione nulla mentre non è vero il viceversa perché la correlazione può essere nulla anche se esiste una qualsiasi relazioni non lineari tra le due variabili.

Test di bianchezza: I due indici appena visti spiegano l'importanza del test di bianchezza. Serve per verificare se l'errore di predizione (residuo) tra i dati reali e l'uscita dal modello identificato è effettivamente un processo bianco. In base alla bianchezza di una realizzazione del processo vero si decide se il modello identificato è una buona descrizione del sistema dinamico reale.

Per effettuare il test si bianchezza è utile il commando `resid` di Matlab che plotta un grafico per la auto-correlazione del errore di predizione e un grafico con la cross-correlazione tra l'errore di predizione e ogni ingresso reale che poi serve per il test di indipendenza. Inoltre nei grafici viene visualizzato l'intervallo di confidenza al 99%, che è un intervallo di valori plausibili ricavato dalla specifica distribuzione di probabilitá dei residui. Per avere bianchezza tutte le correlazioni (tranne la prima che in realtá è la varianza del residuo) dovrebbero trovarsi all'interno dell'intervallo di confidenza, o almeno non eccessivamente fuori. Nel caso contrario si ha che i campioni dell'errore di predizione sono correlati, il che si può avere per esempio se l'ordine del modello usato per l'identificazione non è adeguato e quindi il modello non è adeguato per descrivere il sistema dinamico vero.

Assumendo che l'errore di predizione $\varepsilon(t)$ sia bianco si ha che la covarianza $r_\varepsilon(\tau) = 0$ per ogni $\tau \neq 0$ e quindi per la covarianza campionaria

$$\hat{r}_\varepsilon(\tau) = \frac{1}{N} \sum_{t=1}^{N-\tau} \varepsilon(t+\tau)\varepsilon(t)$$

si ha per $N \to \infty$

$$\begin{aligned} \hat{r}_\varepsilon(\tau) &\to 0 \qquad \forall\, \tau \neq 0, \\ \hat{r}_\varepsilon(0) &\to E\varepsilon^2(t). \end{aligned}$$

Per il test di bianchezza infine viene considerato la quantità normalizzata

$$x_\tau = \frac{\hat{r}_\varepsilon(\tau)}{\hat{r}_\varepsilon(0)}.$$

Test di indipendenza: Il test di indipendenza è relativo alla cross-correlazione tra l'errore di predizione e gli ingressi reali. Si verifica se i residui sono effettivamente indipendenti dalle osservazioni passate e quindi se è stato estratto tutta la informazione possibile dai dati. Come accennato prima per concludere che il modello identificato è adeguato si controlla nei grafici della cross-correlazione tra

il residuo e gli ingressi reali ottenuto mediante il commando `resid` che non ci sono picchi significativamente fuori della regione di confidenza. Se per esempio si ha un picco al ritardo k-esimo che sta fuori della regione di confidenza questo significa che l'uscita $y(t)$ non ha ancora estratto tutta l'informazione possibile dall'ingresso $u(t-k)$ e un altro modello potrebbe spiegare meglio l'uscita all'istante t.
Il commando `resid` assume che il residuo $\varepsilon(t)$ sia indipendente dagli ingressi passati e sia a media nulla e che si ha quindi $r_{\varepsilon u}(\tau) = E\varepsilon(t+\tau)u(t) = 0$.
Per τ si considerano sia valori positivi che negativi. Mentre se il modello non è adeguato per descrivere il sistema per valori di τ positivi si può aspettare che ci sono picche di cross-correlazione che sono fuori dell'intervallo di confidenza questo non è detto per valori di τ negativi. Per esempio nel caso che $u(t)$ sia un processo bianco questi picchi sono comunque nulle.
Calcolando la correlazione campionaria come

$$\hat{r}_{\varepsilon u}(\tau) = \frac{1}{N} \sum_{t=1-min(0,\tau)}^{N-max(0,\tau)} \varepsilon(t+\tau)u(t)$$

per il test di indipendenza si considera la quantità normalizzata

$$x_\tau = \frac{\hat{r}_{\varepsilon u}(\tau)}{[\hat{r}_\varepsilon(0)\hat{r}_\varepsilon(0)]^{1/2}}.$$

Mappa zeri-poli: Tracciando la mappa zeri-poli di ogni funzione di trasferimento del modello identificato si può dare una prima indicazione sull'esattezza dell'ordine del modello. Se ci sono delle quasi-cancellazioni si può aspettare che è possibile ridurre l'ordine del modello identificato.

fit: Il fit da la percentuale della varianza spiegata ed è calcolato mediante la seguente formula:

$$fit = 100 * \left[1 - \frac{norm(y_h - y)}{norm(y - mean(y))}\right]$$

dove y_h sono le uscite dal modello identificato, mentre y è l'uscita reale. Dalla formula si vede che più l'uscita del modello è vicina a quella reale, più vicino è il fit al 100% e piú sarrà quindi adeguato il modello identificato per spiegare il sistema reale.

Errore quadratico medio: Un indice che valuta la prestazione del modello identificato che viene spesso usato è l'MSE (mean square error). Esso è definito da

$$\begin{aligned} MSE &= \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \\ &= Var(\hat{y}) + mean^2(\hat{y} - y) \end{aligned}$$

Questo indice da quindi informazioni sulla bontà dello stimatore in termini di varianza e dispersione della stima. Spesso si calcola anche la radice di questo indice, che viene chiamata con radice dell'errore di predizione (in inglese root mean square error, RMSE) che è quindi proporzionale alla deviazione standard della stima.

Sparsità La sparsità da la percentuale di funzioni di trasferimento nulle nel modello identificato. Avere una funzione di trasferimento nulla significa che la rispettiva misura (ingresso) non è stata utilizzata alla fine di calcolare l'uscita del modello identificato e quindi che il rispettivo sensore può essere eliminato nel calcolo della rispettiva uscita.

Conclusioni

Bibliografia

[1] Giorgio Picci. *Metodi statistici per l'identificazione di sistemi lineari*. 2011.

[2] Alessandro Chiuso and Gianluigi Pillonetto. Identification of large-scale sparse linear dynamic systems: a regularization based approach. 2010.

[3] Lennart Ljung. *System identification - theory for the user*. 1999.

[4] Alessandro Chiuso and Gianluigi Pillonetto. Nonparametric sparse estimators for identification of large scale linear systems.

[5] Alessandro Chiuso and Gianluigi Pillonetto. Learning sparse dynamic linear systems using stable spline kernels and exponential hyperpriors.

[6] Federico Bonollo, Daniele Cattai, and Marco Verga. Identificazione termodinamica di un edificio con l'utilizzo di pca e pls. 2007-2008.

[7] Brandly Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *Journal of the Royal Statistical Society*, 2004.

[8] Elena Toffoli, Giancarlo Baldan, and Guido Albertin. Identificazione termodinamica di un edificio. 2006-2007.

[9] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society*, 2006.

[10] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 1996.

[11] Sensirion AG. *Datasheet SHT1x - Humidity and temperature sensor*, 2010.

[12] Moteiv Corporation. *Tmote sky - humidity, light and temperature sensor with USB*, 2006.