A software package for system identification in the behavioral setting

Ivan Markovsky

Vrije Universiteit Brussel

Outline

Introduction: system identification in the behavioral setting

Solution approach: structured low-rank approximation

Examples

Work plan

1. define a problem
2. develop methods/algorithms to solve the problem
3. implement the algorithm in software

an identification problem involves:

- data collection $\rightsquigarrow$ set of time-series (trajectories)
- choice of model class $\rightsquigarrow$ bounded complexity LTI
- choice of identification criteria $\rightsquigarrow$ weighted 2-norm

Representation free problem formulation

- model $\mathscr{B}$ — set of trajectories $w : \mathbb{Z} \to \mathbb{R}^q$
- representation — equations, which solution set $= \mathscr{B}$
- parameters — specify the representation
- behavioral setting — problem definitions involve the model, not its representations
- solution methods do involve representations, however they are implementation details of the methods

LTI model representations

- kernel representation

$$\mathscr{B} = \{ w \mid R_0 w + R_1 \sigma w + \cdots + R_\ell \sigma^\ell w = 0 \} \quad \text{(KER)}$$

where σ is the shift operator $(\sigma w)(t) = w(t+1)$

- input/state/output representation

$$\mathscr{B} = \{ w = \Pi(u,y) \mid \exists\, x, \text{ such that } \sigma x = Ax + Bu,\; y = Cx + Du \} \quad \text{(I/S/O)}$$

Π is a permutation matrix, defining the I/O partitioning

Model class $\mathscr{L}_{\mathtt{m},\ell}$

- the smallest ℓ, for which (KER) exists, is the lag of $\mathscr{B}$
- the smallest $\mathtt{n} = \dim(x)$, for which (I/S/O) exists, is the order of $\mathscr{B}$
- the number of inputs $\mathtt{m}$ is invariant of the repr. (I/S/O)
- $(\mathtt{m}, \ell)$ and $(\mathtt{m}, \mathtt{n})$ are measures of model's complexity
- $\mathscr{L}_{\mathtt{m},\ell}$ — LTI models with $\leq \mathtt{m}$ inputs and lag $\leq \ell$

Approximation criterion

- orthogonal distance between data and model

$$M(w, \mathscr{B}) := \min_{\widehat{w} \in \mathscr{B}} \|w - \widehat{w}\|_2$$

- $M(w, \mathscr{B})$ shows how much $\mathscr{B}$ fails to "explain" w
- called misfit (lack of fit) between w and $\mathscr{B}$

system identification problem

$$\text{minimize} \quad \text{over } \widehat{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell} \quad M(w, \widehat{\mathscr{B}}) \qquad \text{(SYSID)}$$

Generalizations

- multiple time-series $w = \{ w^1, \dots, w^N \}$

$$M(w, \mathscr{B}) := \min_{\{ \widehat{w}^1, \dots, \widehat{w}^N \} \subset \mathscr{B}} \sqrt{\textstyle\sum_{i=1}^N \| w^i - \widehat{w}^i \|_2^2}$$

- fixed initial conditions w_{ini}

$$M(w, \mathscr{B}) := \min_{(w_{\text{ini}}, \widehat{w}) \in \mathscr{B}} \| w - \widehat{w} \|_2$$

- fixed variables $\mathscr{I} \subset \{ 1, \dots, q \}$

$$M(w, \mathscr{B}) := \min_{\widehat{w} \in \mathscr{B}, \ \widehat{w}_{\mathscr{I}} = w_{\mathscr{I}}} \| w - \widehat{w} \|_2$$

- missing data: $w_j^i(t) = \mathtt{NaN} \implies w_j^i(t)$ is missing

Mosaic-Hankel matrix

$1 \times N$ block matrix

$$\mathscr{H}_{\ell+1}(w) := \begin{bmatrix} \mathscr{H}_{\ell+1}(w^1) & \cdots & \mathscr{H}_{\ell+1}(w^N) \end{bmatrix}$$

with block-Hankel blocks

$$\mathscr{H}_{\ell+1}(w^i) := \begin{bmatrix} w^i(1) & w^i(2) & \cdots & w^i(T-\ell) \\ w^i(2) & w^i(3) & \cdots & w^i(T-\ell+1) \\ \vdots & \vdots & & \vdots \\ w^i(\ell+1) & w^i(\ell+2) & \cdots & w^i(T) \end{bmatrix}$$

Mosaic-Hankel low-rank approximation

$$\big(w^i(1),\dots,w^i(T_i-\ell)\big) \in \mathscr{B} \in \mathscr{L}_{\mathrm{m},\ell}, \quad \text{for } i=1,\dots,N$$

$$\Updownarrow$$

$$\operatorname{rank}\big(\mathscr{H}_{\ell+1}(w)\big) \leq \ell q + \mathrm{m}$$

(SYSID) is mosaic-Hankel low-rank approx. problem:

$$\text{minimize} \quad \text{over } \widehat{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell} \quad M(w,\widehat{\mathscr{B}})$$

$$\Updownarrow$$

$$\begin{aligned} &\text{minimize} \quad \text{over } \widehat{w} \quad \underbrace{(w-\widehat{w})^\top \operatorname{diag}(v)(w-\widehat{w})}_{\|w-\widehat{w}\|_v} \\ &\text{subject to} \quad \operatorname{rank}\big(\mathscr{H}_{\ell+1}(w)\big) \leq \ell q + \mathrm{m} \end{aligned}$$

$v_i = 0$ if $w_i = \mathtt{NaN}$, $v_i = \infty$ if w_i is exact, $w_i = 1$ otherwise

Solution method

- kernel representation of the rank constraint

$$\operatorname{rank}\left(\mathscr{H}_{\ell+1}(\widehat{w})\right) \leq r \quad \iff \quad \begin{array}{l} R\mathscr{H}_{\ell+1}(\widehat{w}) = 0 \\ RR^\top = I_{(\ell+1)q-r} \end{array}$$

- variable projection: elimination of the variable $\widehat{w}$

$$M(R) := \min_{\widehat{w}} \|w - \widehat{w}\|_v \quad \text{subject to} \quad R\mathscr{H}_{\ell+1}(\widehat{w}) = 0$$

is a least-norm problem with analytic solution

$$M(R) = \operatorname{vec}^\top(w)\Gamma^{-1}(R)\operatorname{vec}(w)$$

where Γ is a positive definite banded Toeplitz matrix

SLRA software package

- the identification problem is then

$$\text{minimize} \quad \text{over } R \quad M(R) \quad \text{subject to} \quad RR^\top = I$$

- nonconvex optimization problem on a manifold
- efficient evaluation of $M(R)$ exploiting the structure
- software implementation is available

Usage and implementation

- `[sysh,info,wh] = ident(w, m, ell, opt)`
  - `sysh` — (I/S/O) repr. of the identified model
  - `opt.sys0` — (I/S/O) repr. of initial approximation
  - `opt.wini` — initial conditions
  - `opt.exct` — exact variables
  - `info.Rh` — parameter R of (KER)
  - `info.M` — misfit
- `[M, wh, xini] = misfit(w, sysh, opt)`
- implemented as a literate program
  - the code "lives" in research papers
  - the papers document the code
  - the code reveals the full implementation details

The simulation setup

- simulation parameters

```
ell = 2; m = 1; p = 1; T = 30; s = 0.02;
```

- define constants

```
q = m + p; n = ell * p;
```

- the "true" system and "true" data

```
sys0 = drss(n, p, m);
u0 = rand(T, 1); y0 = lsim(sys0, u0);
```

The simulation setup

- add noise

```
w0 = [u0 y0]; w = w0 + s * randn(T, q);
```

- generate noisy data for the examples

```
clear all
ell = 2; m = 1; p = 1; T = 30; s = 0.02;

q = m + p; n = ell * p;

sys0 = drss(n, p, m);
u0 = rand(T, 1); y0 = lsim(sys0, u0);
w0 = [u0 y0]; w = w0 + s * randn(T, q);
```

Invariance to variables permutation

- identify a model with permuted variables

```
io = fliplr(1:q);
[sysh, info] = ident(w(:, io), m, ell);
disp(info.M)
    0.0086
```

- identify a model with the original variables order

```
io = 1:q;
[sysh, info] = ident(w(:, io), m, ell);
disp(info.M)
    0.0086
```

Zero initial conditions

- identify the system with option `opt.wini = 0`

```
opt.wini = 0;
[sysh0, info0, wh] = ident(w, m, ell, opt);
disp(info0.M)
    0.0090
```

- verify that $(0,\ldots,0,\widehat{w}) \in \widehat{\mathscr{B}}$

```
whext = [zeros(ell, q); wh];
disp(misfit(whext, sysh0))
   1.8227e-32
```

- compare the misfit with
  - free initial conditions $8.61 \cdot 10^{-3}$
  - zero initial conditions $8.98 \cdot 10^{-3}$

Identification from multiple trajectories

- split w into two parts and apply `ident` on them

```
wm{1} = w(1:10, :); wm{2} = w(11:T, :);
[sysh, infom, wmh] = ident(wm, m, ell);
disp(infom.M)
    0.0072
```

- compare the misfit when fitting
  - one trajectory $8.61 \cdot 10^{-3}$
  - the same trajectory split into two parts $7.22 \cdot 10^{-3}$

System identification with missing data

- w^1 is the noisy trajectory of the system
- $w^2 = (\delta, \texttt{NaN})$ — exact input, missing output, and zero initial conditions
- $\widehat{w}^2 = (\delta, \widehat{h})$ — estimate of the system's impulse resp.
- data-driven simulation problem
- $\widehat{y}^2$ is compared with the true impulse response and the impulse response of the model identified from w^1

System identification with missing data

```
wm{1} = w;
wm{2} = [1 NaN; 0 NaN; 0 NaN; 0 NaN];
opt.wini = {[] 0}; opt.exct = {[] 1};
[~, info, wh] = ident(wm, m, ell, opt);
disp(wh{2}(:, 2)')
   -0.0633    0.1407    0.0563    0.0287
disp(impulse(sys0, 3)')
   -0.0713    0.1353    0.0487    0.0360
```