DYSCO course on low-rank approximation and its applications

Exercises

Ivan Markovsky

Vrije Universiteit Brussel

Outline

Introduction

Computational tools

Behavioral approach

Approximate identification

Exact identification

Line fitting

problem: give a condition on the data

$$\mathscr{D} = \{ d_1, \ldots, d_N \} \subset \mathbb{R}^2$$

that is equivalent to the condition that

the points $d_1, \ldots, d_N$ are on a line in $\mathbb{R}^2$

Solution

the points $d_i = (a_i, b_i)$, $i = 1, \ldots, N$ lie on a line

$$\Updownarrow$$

there is $(R_1, R_2, R_3) \neq 0$, such that
$R_1 a_i + R_2 b_i + R_3 = 0$, for $i = 1, \ldots, N$

$$\Updownarrow$$

there is $(R_1, R_2, R_3) \neq 0$, such that

$$\begin{bmatrix} R_1 & R_2 & R_3 \end{bmatrix} \begin{bmatrix} a_1 & \cdots & a_N \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} = 0$$

$$\Updownarrow$$

$$\text{rank}\left(\begin{bmatrix} a_1 & \cdots & a_N \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} \right) \leq 2$$

Note

- $\mathscr{B} = \{\, d \mid Rd = 0 \,\}$ — linear static model
- $\mathscr{B} = \{\, d \mid R \left[\begin{smallmatrix} d \\ 1 \end{smallmatrix}\right] = 0 \,\}$ — affine static model
- in exact modeling

$$\begin{array}{c}\text{affine fitting} \\ \Updownarrow \\ \text{data centering + linear modeling}\end{array}$$

- HW: is the same true in approximate modeling?

Conic section fitting

- conic section (with parameters $S = S^\top$, u, v)

$$\mathscr{B}(S,u,v) = \{\, d \in \mathbb{R}^2 \mid d^\top S d + u^\top d + v = 0 \,\}$$

1. give a condition on the data

$$\mathscr{D} = \{\, d_1, \dots, d_N \,\} \subset \mathbb{R}^2$$

   that is equivalent to the condition that

   $d_1, \dots, d_N$ are lying on a conic section

2. find a conic section fitting the points

$$d_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad d_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad d_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad d_4 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Solution

the points $d_i = (a_i, b_i)$, $i = 1, \dots, N$ lie on a conic section

$$\Updownarrow$$

$\exists\ S = S^\top$, u, v, at least one of them nonzero, such that
$d_i^\top S d_i + u^\top d_i + v = 0$, for $i = 1, \dots, N$

$$\Updownarrow$$

there is $(s_{11}, s_{12}, s_{22}, u_1, u_2, v) \neq 0$, such that

$$\begin{bmatrix} s_{11} & 2s_{12} & u_1 & s_{22} & u_2 & v \end{bmatrix} \begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ a_1 & \cdots & a_N \\ b_1^2 & \cdots & b_N^2 \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix} = 0$$

Solution (continued)

the points $d_i = (a_i, b_i)$, $i = 1, \ldots, N$ lie on a conic section

$$\Updownarrow$$

$$\text{rank}\left(\begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ a_1 & \cdots & a_N \\ b_1^2 & \cdots & b_N^2 \\ b_1 & \cdots & b_N \\ 1 & \cdots & 1 \end{bmatrix}\right) \leq 5$$

```
f = @(a, b) [a .^ 2; a .* b; a; b .^ 2; b;
             ones(size(a))];
```

Solution (continued)

- finding exact models

```
R = null(f(d(1, :), d(2, :))')';
```

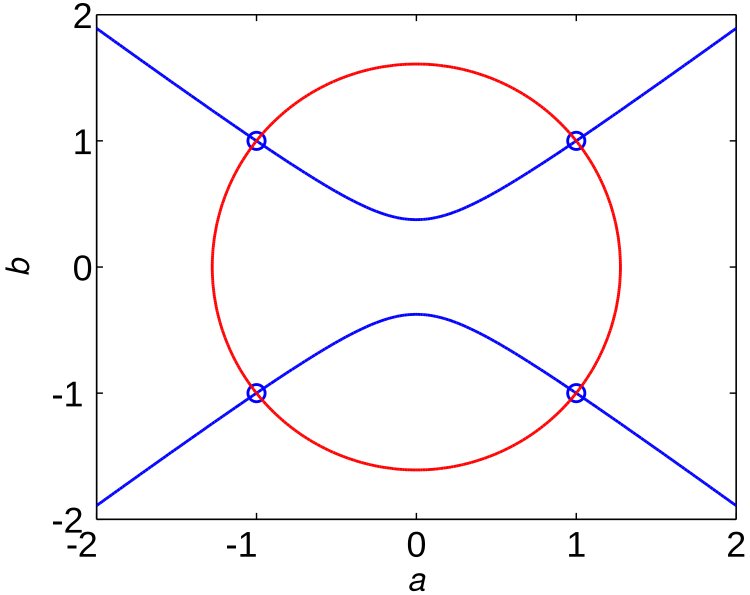
- plotting a model

```
function H = plot_model(th, f, ax, c)
H = ezplot(@(a, b) th * f(a, b), ax);
for h = H', set(h, 'color', c, 'linewidth',
```

- show results

```
plot(d(1, :), d(2, :), 'o', 'markersize', 12
ax = 2 * axis;
for i = 1:size(R, 1)
  hold on, plot_model(R(i, :), f, ax, c(i))
end
```

Solution (continued)



Subspace clustering

- union of two lines (with parameters $R^1, R^2 \in \mathbb{R}^{1\times 2}$)

$$\mathscr{B}(R^1, R^2) = \{\, d \in \mathbb{R}^2 \mid (R^1 d)(R^2 d) = 0 \,\}$$

1. give a condition on the data

$$\mathscr{D} = \{\, d_1, \dots, d_N \,\} \subset \mathbb{R}^2$$

   that is equivalent to the condition that

   $d_1, \dots, d_N$ are lying on a union of two lines

2. find a union of two lines model fitting the points

$$d_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}, \quad d_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad d_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad d_4 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Solution

the points $d_i \in \mathbb{R}^2$, $i = 1, \ldots, N$ lie on a union of two lines

$$\Updownarrow$$

there are $R^1 \neq 0$ and $R^2 \neq 0$, v, such that
$(R^1 d_i)(R^2 d_i) = 0$, for $i = 1, \ldots, N$

$$\Updownarrow$$

there are $\begin{bmatrix} R_1^1 & R_2^1 \end{bmatrix} \neq 0$ and $\begin{bmatrix} R_1^2 & R_2^2 \end{bmatrix} \neq 0$, such that

$$\begin{bmatrix} R_1^1 R_1^2 & R_1^1 R_2^2 + R_2^1 R_1^2 & R_2^1 R_2^2 \end{bmatrix} \begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ b_1^2 & \cdots & b_N^2 \end{bmatrix} = 0$$

Solution (continued)

- if $d_i \in \mathbb{R}^2$, $i = 1, \ldots, N$ lie on a union of two lines, then

$$\text{rank}\left(\begin{bmatrix} a_1^2 & \cdots & a_N^2 \\ a_1 b_1 & \cdots & a_N b_N \\ b_1^2 & \cdots & b_N^2 \end{bmatrix}\right) \leq 2$$

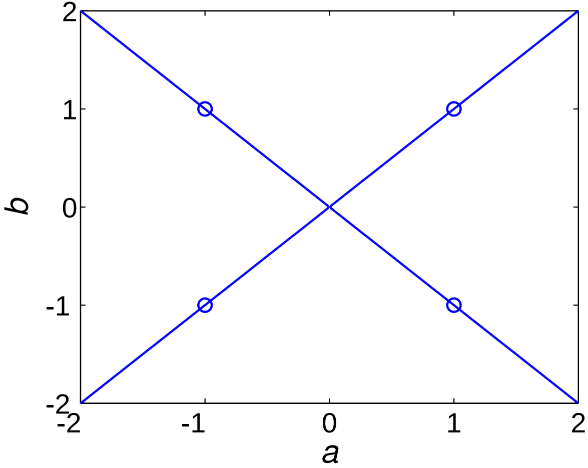
- in this case, the rank condition is only necessary

- in additional, a basis for the left kernel is

$$\begin{bmatrix} 1 & \alpha+\beta & \alpha\beta \end{bmatrix}, \quad \text{for some } \alpha \text{ and } \beta$$

- union of two lines fitting is a special case of the Generalized principal component analysis

Solution (continued)



- HW: how to "extract" R^1 and R^2 from ker(D)

Recursive sequences

- $w = \big(w(1), \dots, w(T)\big)$ is recursive of order ℓ if

$$R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0,$$
$$\text{for } t = 1, \dots, T-\ell \text{ and some } R_0, R_1, \dots, R_\ell \in \mathbb{R}$$

1. give a condition on w that is equivalent to

   w is a recursive of order ℓ

2. find the minimal recursive order of

$$(1, 2, 4, 7, 13, 24, 44, 81)$$

Solution

$$w \begin{array}{c}\text{is recursive}\\ \text{of order } \ell\end{array} \iff \exists\, R \in \mathbb{R}^{1\times(\ell+1)} \text{ s.t. } R\mathscr{H}_{\ell+1}(w) = 0$$

$$\implies \operatorname{rank}\left(\mathscr{H}_{\ell+1}(w)\right) \leq \ell$$

where

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

- for $\ell = 1, 2, \ldots$, if $\operatorname{rank}\left(\mathscr{H}_{\ell+1}(w)\right) = \ell$, stop
- $\ell_{\min} = 3$ and $R = \begin{bmatrix} 1 & 1 & 1 & -1 \end{bmatrix}$

Polynomial common divisor

- the polynomials

$$p(z) = p_0 + p_1 z + \cdots + p_{\ell_p} z^{\ell_p}$$
$$q(z) = q_0 + q_1 z + \cdots + q_{\ell_q} z^{\ell_q}$$

  have a common divisor

$$c(z) = c_0 + c_1 z + \cdots + c_{\ell_c} z^{\ell_c}$$

  iff $p = ca$ and $q = cb$ for some polynomials a and b

- give a condition on p, q that is equivalent to

  p, q have a common divisor of degree ℓ_c

Solution

$$c(z) = a(z)b(z) \iff \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{\ell_c} \end{bmatrix} = \begin{bmatrix} a_0 & & & \\ a_1 & a_0 & & \\ \vdots & a_1 & \ddots & \\ a_{\ell_a} & \vdots & \ddots & a_0 \\ & a_{\ell_a} & & a_1 \\ & & \ddots & \vdots \\ & & & a_{\ell_a} \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{\ell_b} \end{bmatrix}$$

$$\iff : \ c = S_{\ell_b}(a)b \iff c = S_{\ell_a}(b)a$$

Solution (continued)

$$\begin{array}{c} p \in \mathbb{R}[z] \text{ and } q \in \mathbb{R}[z] \\ \text{have common divisor} \\ c \in \mathbb{R}[z],\ \deg(c) = \ell_c \end{array} \iff \begin{array}{l} \exists\ a \in \mathbb{R}[z],\ \deg(a) = \ell_p - \ell_c \\ \exists\ b \in \mathbb{R}[z],\ \deg(b) = \ell_q - \ell_c \\ \text{such that } p = ca \text{ and } q = cb \end{array}$$

$$\iff \quad qa - pb = 0$$

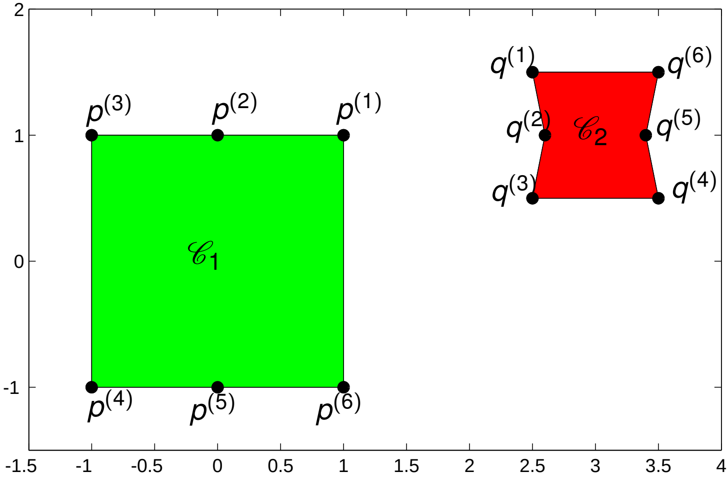
$$\iff \quad \begin{bmatrix} S_{\ell_a}(q) & S_{\ell_b}(p) \end{bmatrix} \begin{bmatrix} a \\ -b \end{bmatrix} = 0$$

$$\iff \quad \begin{bmatrix} S_{\ell_a}(q) & S_{\ell_b}(p) \end{bmatrix} \text{ is rank deficient}$$

Outline

Least squares contour alignment

given contours $\mathscr{C}_1, \mathscr{C}_2$, specified by matching points $p^{(i)} \leftrightarrow q^{(i)}$



Problem

- find a transformation (rotation + scaling + translation)

$$\mathscr{A}_{a,\theta,s}(p) = s \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} p + a$$

  that minimizes the LS distance between

$$\min_{a\in\mathbb{R}^2,\theta\in[0,2\pi),s\in\mathbb{R}_+} \sum_{i=1}^{N} \|p^{(i)} - \mathscr{A}_{a,\theta,s}(q^{(i)})\|_2^2$$

  $\mathscr{C}_1$ and $\mathscr{A}_{a,\theta,s}(\mathscr{C}_2)$

- apply the solution on the data in the example

Data in the example

```
p = [1      0      -1      -1      0      1
     1      1       1      -1     -1     -1];

q = [2.5    2.6    2.5    3.5    3.4    3.5
     1.5    1.0    0.5    0.5    1.0    1.5];
```

Hint

- use the change of variables

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = s \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \theta \\ s \end{bmatrix} = \begin{bmatrix} \sin^{-1}(b_2/\sqrt{b_1^2 + b_2^2}) \\ \sqrt{b_1^2 + b_2^2} \end{bmatrix}$$

- to obtain an equivalent problem

$$\min_{(a_1,a_2,b_1,b_2)\in\mathbb{R}^4} \left\| \begin{bmatrix} p_1^{(1)} \\ p_2^{(1)} \\ \vdots \\ p_1^{(N)} \\ p_2^{(N)} \end{bmatrix} - \begin{bmatrix} 1 & 0 & q_1^{(1)} & -q_2^{(1)} \\ 0 & 1 & q_2^{(1)} & q_1^{(1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & q_1^{(N)} & -q_2^{(N)} \\ 0 & 1 & q_2^{(N)} & q_1^{(N)} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix} \right\|_2$$

Orthogonal Procrustes problem

- HW: alignment by reflection + scaling + translation
- rigid transformation =
  rotation + reflection + scaling + translation
- contour alignment by rigid transformation is related to the orthogonal Procrustes problem:
  - given $m \times n$ real matrices C_1 and C_2

    minimize over Q $\quad \|C_1 - QC_2\|_F \quad$ subject to $\quad Q^\top Q = I$
  - solution: $Q = UV^\top$, where $U\Sigma V^\top$ is the SVD of $C_1^\top C_2$

Outline

Check whether $w \overset{?}{\in} \mathscr{B}$

- $w = (u_d, y_d) = \big((0,1),(0,1),(0,1),(0,1)\big)$

```
w = [0 0 0 0; 1 1 1 1];
```

- $\mathscr{B} = \ker\big(R(\sigma)\big)$, where $R(z) = \begin{bmatrix}1 & -1\end{bmatrix} + \begin{bmatrix}-1 & 1\end{bmatrix} z$

```
R = [1 -1 -1 1]; ell = 1;
```

$w \overset{?}{\in} \ker\left(R(\sigma)\right)$

$$
\begin{aligned}
&\iff R(\sigma)w = 0 \\
&\iff R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0 \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{for } t = 1, \ldots, T-\ell \\
&\iff \underbrace{\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell & & & \\ & R_0 & R_1 & \cdots & R_\ell & & \\ & & \ddots & \ddots & & \ddots & \\ & & & R_0 & R_1 & \cdots & R_\ell \end{bmatrix}}_{\mathscr{M}_T(R) \in \mathbb{R}^{p(T-\ell)\times qT}} \underbrace{\begin{bmatrix} w(1) \\ w(2) \\ \vdots \\ w(T) \end{bmatrix}}_{\operatorname{vec}(w)} = 0
\end{aligned}
$$

$w \overset{?}{\in} \ker\left(R(\sigma)\right)$

$$\begin{aligned} &\iff \quad \mathscr{M}_T(R)\,\mathrm{vec}(w) = 0 \\ &\iff \quad R\mathscr{H}_{\ell+1}(w) = 0 \end{aligned}$$

where

$$\underbrace{\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix}}_{R} \underbrace{\begin{bmatrix} w(1) & w(2) & \cdots & w(T-\ell) \\ w(2) & w(3) & \cdots & \\ \vdots & \vdots & & \vdots \\ w(\ell+1) & w(\ell+2) & \cdots & w(T) \end{bmatrix}}_{\mathscr{H}_{\ell+1}(w) \in \mathbb{R}^{q(\ell+1)\times(T-\ell)}} = 0$$

- compute $e = \|R\mathscr{H}_{\ell+1}(w)\|$ and check if $e < \varepsilon$

```
w = [0 0 0 0; 1 1 1 1];
R = [1 -1 -1 1]; ell = 1;
norm(R * blkhank(w, ell + 1))
```

- `blkhank` constructs a block-Hankel matrix $\mathscr{H}_L(w)$

```
function H = blkhank(w, i, j)
[q, T] = size(w);
if T < q, w = w'; [q, T] = size(w); end
if nargin < 3, j = T - i + 1; end
H = zeros(i * q, j);
for ii = 1:i
  H(((ii - 1) * q + 1):(ii * q), :) ...
                   = w(:, ii:(ii + j - 1));
end
```

Homework

- use image representation to check

$$w \overset{?}{\in} \text{image}\big(P(\sigma)\big)$$

- use state space representation to check

$$w \overset{?}{\in} \mathscr{B}(A, B, C, D)$$

Affine time-invariant system

- an LTI system $\mathscr{B} \in \mathscr{L}_{\mathrm{m},\ell}$ admits a kernel repr.

$$\mathscr{B} = \ker\big(R(\sigma)\big) := \{w \mid R(\sigma)w = 0\}$$

  for some $R(z) = R_0 z^0 + R_1 z^1 + \cdots + R_\ell z^\ell$

- show that

$$\mathscr{B}_c := \{\, w \mid R(\sigma)w = c \,\}$$

  is an affine time-invariant system, *i.e.*, $\mathscr{B}_c = \mathscr{B} + w_\mathrm{p}$
  for LTI model $\mathscr{B} \in \mathscr{L}_{\mathrm{m},\ell}$ and trajectory w_p

- find $\mathscr{B}$ and w_p, s.t. $\mathscr{B} + w_\mathrm{p} = \{\, w \mid (0.5 + \sigma)w = 1 \,\}$

- using the matrix representation of $R(\sigma)$

$$
\begin{aligned}
w \in \mathscr{B}_c \quad &\iff \quad \mathscr{M}_T(R)w = \mathbf{1}_{T-\ell} \otimes c =: \mathbf{c} \\
&\iff \quad \mathscr{M}_T(R)(w - w_\mathrm{p}) = 0 \\
&\iff \quad w - w_\mathrm{p} \in \ker\big(R(\sigma)\big) = \mathscr{B}
\end{aligned}
$$

- therefore, $\mathscr{B}_c = \mathscr{B} + w_\mathrm{p}$, where $\mathscr{B} \in \mathscr{L}_{\mathrm{m},\ell}$ and

$$
\mathscr{M}_T(R)w_\mathrm{p} = \mathbf{c}
$$

- *e.g.*, the least-norm solution

$$
w_\mathrm{p} = \mathscr{M}_T^\top(R)\big(\mathscr{M}_T(R)\mathscr{M}_T^\top(R)\big)^{-1}\mathbf{c}
$$

- HW: find input/state/output representation of $\mathscr{B}_c$

- in the case of $\{\, w \mid (0.5+\sigma)w = 1 \,\}$
- `sylv(R, T)` constructs the matrix $\mathscr{M}_T(R)$

```
function S = sylv(R, T)
nR = length(R); q = 2;
n = (nR / q) - 1;
S = zeros(T - n, q * T);
for i = 1:T - n
  S(i, (1:nR) + (i - 1) * q) = R;
end
```

Transfer function $\mapsto$ kernel representation

- what model $\mathscr{B}_{\text{tf}}(H)$ is specified by a transfer function

$$H(z) = \frac{q(z)}{p(z)} = \frac{q_0 + q_1 z^1 + \cdots + q_\ell z^\ell}{p_0 + p_1 z^1 + \cdots + p_\ell z^\ell}$$

- find R, such that

$$\mathscr{B}_{\text{tf}}(H) = \ker(R)$$

- write a function `tf2ker` converting H (`tf` object) to R

- $H(z) = q(z)/p(z) \quad \overset{?}{\leftrightarrow} \quad R(z)$
- $y(z) = H(z)u(z) \quad \leftrightarrow \quad p(\sigma)y = q(\sigma)u$

$$\underbrace{\begin{bmatrix} q(\sigma) & -p(\sigma) \end{bmatrix}}_{R(\sigma)} \begin{bmatrix} u \\ y \end{bmatrix} = 0$$

- note: z may correspond to σ^{-1} as well as σ
- does $\mathscr{B}_{\text{tf}}(H)$ assume zero initial conditions?
  - if so,

$$\mathscr{B}_{\text{tf}}(H) = \{w \mid R(\sigma)(0 \wedge w) = 0\big)$$

  - otherwise,

$$\mathscr{B}_{\text{tf}}(H) = \ker\big(R(\sigma)\big)$$

- note: MATLAB uses descending order of coefficients

```
function R = tf2ker(H)
[Q P] = tfdata(tf(H), 'v');
R = vec(fliplr([Q; -P]))';
```

Specification of initial conditions

- initial conditions are explicitly specified in I/S/O repr.

- in MATLAB

```
LSIM(SYS,U,T,X0) specifies the initial
state vector X0 at time T(1)
  (for state-space models only).
```

- in transfer function representation initial conditions are often set to 0

- explain how to specify initial conditions in a representation free manner

- what is the link to $x_{\text{ini}} = x(1)$ in I/S/O repr?

- assuming that $\mathscr{B}$ is controllable

- initial conditions can be specified by prefix trajectory

$$w_{\text{ini}} = \big(w_{\text{ini}}(1), \ldots, w_{\text{ini}}(T_{\text{ini}})\big)$$

  i.e., by $w_{\text{ini}} \wedge w \in \mathscr{B}$

- the link between w_{ini} and x_{ini} is given by

$$y_{\text{ini}} = \mathscr{O}_\ell(A, C) A^{-\ell} x_{\text{ini}} + \mathscr{T}_\ell(A, B, C, D) u_{\text{ini}}$$

```
function x0 = inistate(w, sys)
l = size(sys, 'order');
x0 = obsv(sys) \ (w(1:l, 2) ...
                  - lsim(sys, w(1:l, 1)));
```

Output matching

- given y_f and $\mathscr{B}$
- find u_f, such that $(u_f, y_f) \in \mathscr{B}$

Setup

- random SISO unstable system $\mathscr{B}$

```
clear all, n = 3;
Br = drss(n); [Qr, Pr] = tfdata(Br, 'v');
B = ss(tf(fliplr(Qr), fliplr(Pr), -1));
```

- reference output

```
T = 100; yf = ones(T, 1);
```

- $\mathscr{M}_T(R)w = 0 \quad \Longrightarrow \quad \mathscr{M}_T(P)u = \mathscr{M}_T(P)y$

  ```
  R = tf2ker(B); M = sylv(R, T);
  Mu = M(:, 1:2:end); My = - M(:, 2:2:end);
  ```

- many solutions (why?); compute a particular one

  ```
  uf = pinv(Mu) * My * yf;
  ```

- $(u_\mathrm{f}, y_\mathrm{f}) \overset{?}{\in} \ker\big(R(\sigma)\big)$

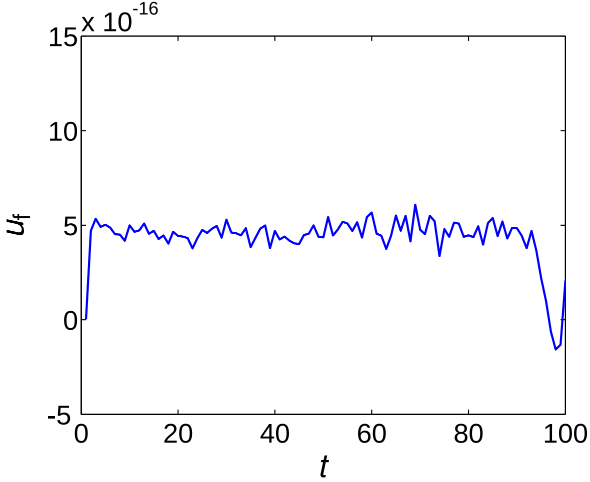
- $(u_\mathrm{f}, y_\mathrm{f}) \overset{?}{\in} \operatorname{image}\big(P(\sigma)\big)$

- $(u_\mathrm{f}, y_\mathrm{f}) \overset{?}{\in} \mathscr{B}(A, B, C, D)$

- where is the problem?

- the system is anti-stable
- the test $w \in \mathscr{B}(A, B, C, D)$ is ill-conditioned
- do backwards in time simulation

- particular (least squares) input



- HW: find all inputs

Outline

Misfit computation using image repr.

- given
  - data $w = \big(w(1), \dots, w(T)\big)$ and
  - LTI system $\mathscr{B} = \text{image}\,\big(P(\sigma)\big)$
- derive a method for computing

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

- *i.e.*, find the orthogonal projection of w on $\mathscr{B}$

$w \overset{?}{\in} \text{image}\,(P(\sigma))$

$\iff$ there is v, such that $w = P(\sigma)v$

$\iff$ there is v, such that for $t = 1, \dots, T$

$$w(t) = P_0 v(t) + P_1 v(t+1) + \cdots + P_\ell v(t+\ell)$$

$\iff$ there is solution v of the system

$$\begin{bmatrix} w(1) \\ w(2) \\ \vdots \\ w(T) \end{bmatrix} = \underbrace{\begin{bmatrix} P_0 & P_1 & \cdots & P_\ell & & & \\ & P_0 & P_1 & \cdots & P_\ell & & \\ & & \ddots & \ddots & & \ddots & \\ & & & P_0 & P_1 & \cdots & P_\ell \end{bmatrix}}_{\mathscr{M}_{T+\ell}(P)} \begin{bmatrix} v(1) \\ v(2) \\ \vdots \\ v(T+\ell) \end{bmatrix}$$

Solution

- we showed that

$$\widehat{w} \in \ker\big(R(\sigma)\big) \quad \iff \quad \widehat{w} = \mathscr{M}_T(P)v, \text{ for some } v$$

- then the misfit computation problem

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

  becomes

$$\text{minimize} \quad \text{over } v \quad \|w - \mathscr{M}_T(P)v\|$$

- this is a standard least-norm problem
- projector on $\mathscr{B} = \text{image}(P)$

$$\Pi_{\text{image}(P)} := \mathscr{M}_T(P)\big(\mathscr{M}_T^\top(P)\mathscr{M}_T(P)\big)^{-1}\mathscr{M}_T^\top(P)$$

- misfit

$$\text{misfit}(w, \mathscr{B}) := \sqrt{w^\top \big(I - \Pi_{\text{image}(P)}\big) w}$$

and optimal approximation

$$\widehat{w} = \Pi_{\text{image}(P)} w$$

- HW: misfit computation with $\mathscr{B} = \ker\big(R(\sigma)\big)$

Misfit computation using I/S/O repr.

- given
  - data $w = \big(w(1), \dots, w(T)\big)$ and
  - LTI system $\mathscr{B} = \mathscr{B}(A, B, C, D)$

- derive a method for computing

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

- *i.e.*, find the orthogonal projection of w on $\mathscr{B}$

$w \overset{?}{\in} \mathscr{B}(A,B,C,D)$

$$\mathscr{B}(A,B,C,D) = \{\, (u,y) \mid \sigma x = Ax + Bu,\ y = Cx + Du \,\}$$

$$(u_d, y_d) \in \mathscr{B}(A,B,C,D) \quad \iff \quad \exists\, x_{\text{ini}} \in \mathbb{R}^n, \text{ such that}$$

$$y = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{T-1} \end{bmatrix}}_{\mathscr{O}_T(A,C)} x_{\text{ini}} + \begin{bmatrix} D & & & & \\ CB & D & & & \\ CAB & CB & D & & \\ \vdots & \ddots & \ddots & \ddots & \\ CA^{T-1}B & \cdots & CAB & CB & D \end{bmatrix} u$$

Solution

- we showed that

$$\widehat{w} \in \mathscr{B}(A,B,C,D) \iff \widehat{y} = \mathscr{O}_T(A,C)\widehat{x}_{\text{ini}} + \mathscr{T}_T(H)\widehat{u}$$

- then the misfit computation problem

$$\min_{\widehat{x}_{\text{ini}},\widehat{u}} \left\| \begin{bmatrix} u_d \\ y_d \end{bmatrix} - \begin{bmatrix} 0 & I \\ \mathscr{O}_T(A,C) & \mathscr{T}_T(H) \end{bmatrix} \begin{bmatrix} \widehat{x}_{\text{ini}} \\ \widehat{u} \end{bmatrix} \right\|$$

- exploiting the structure in the problem
  $\rightsquigarrow$ EIV Kalman filter

Latency computation

- given
  - data w and
  - LTI system $\mathscr{B}_{\text{ext}} = \ker\big(R(\sigma)\big)$ $\qquad (w_{\text{ext}} := \left[\begin{smallmatrix} \widehat{e} \\ w \end{smallmatrix}\right])$
- find an algorithm for computing

$$\text{minimize} \quad \text{over } e \quad \|\widehat{e}\| \quad \text{subject to} \quad (\widehat{e}, w) \in \mathscr{B}_{\text{ext}}$$

- HW: latency computation with $\mathscr{B}_{\text{ext}} = \mathscr{B}(A, B, C, D)$
  (this is the ordinary Kalman filter)

Solution

- partition $R = \begin{bmatrix} R_e & R_w \end{bmatrix}$ conformably with $w_{\text{ext}} = \left[\begin{smallmatrix} e \\ w \end{smallmatrix}\right]$
- by analogy with the derivation on page 47, we have

$$\begin{bmatrix} e \\ w \end{bmatrix} \in \ker\big(R(\sigma)\big) \iff \begin{bmatrix} \mathscr{M}_T(R_e) & \mathscr{M}_T(R_w) \end{bmatrix} \begin{bmatrix} e \\ w \end{bmatrix} = 0$$

- the latency computation problem is

$$\min_{e} \quad \|e\|_2 \quad \text{subject to} \quad \mathscr{M}_T(R_e)e = -\mathscr{M}_T(R_w)w$$

- the solution is given by

$$\widehat{e} = -\underbrace{\left(\mathscr{M}_T(R_e)^\top \mathscr{M}_T(R_e)\right)^{-1} \mathscr{M}_T(R_e)^\top}_{\mathscr{M}_T(R_e)^+} \mathscr{M}_T(R_w)w$$

Software

- mosaic-Hankel low-rank approximation

  http://slra.github.io/software.html

- `[sysh,info,wh] = ident(w, m, ell, opt)`
  - `sysh` — I/S/O representation 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 kernel repr.
  - `info.M` — misfit

- `[M, wh, xini] = misfit(w, sysh, opt)`

- demo file

Variable permutation

- verify that permutation of the variables doesn't change the optimal misfit

```
T = 100; n = 2; B0 = drss(n);
u = randn(T, 1); y = lsim(B0, u) + 0.001 * ranc
[B1, info1] = ident([u y], 1, n); disp(info1.M)
   2.9736e-05
[B2, info2] = ident([y u], 1, n); disp(info2.M)
   2.9736e-05
disp(norm(B1 - inv(B2)))
   5.8438e-12
```

Output error identification

- verify that the results of `oe` and `ident` coincide

```
T = 100; n = 2; B0 = drss(n);
u = randn(T, 1); y = lsim(B0, u) + 0.001 * ranc
opt = oeOptions('InitialCondition', 'estimate')
B1 = oe(iddata(y, u), [n + 1 n 0], opt);
B2 = ident([u y], 1, n, struct('exct', 1));
norm(B1 - B2) / norm(B1)

ans =

   1.4760e-07
```

Outline

Identification without PE input

- given exact data $w = (u_d, y_d) \in \mathscr{B} \in \mathscr{L}^2_{1,\ell}$
- assuming controllability and PE of u_d of order $2\ell + 1$
- $\text{left ker}\left(\mathscr{H}_{\ell+1}(w)\right)$ completely specifies $\mathscr{B}$
  (the model is identifiable from the data)
- what "goes wrong" when u_d is not PE of order $\ell + 1$?
- verify it numerically

Solution

- u_d not PE $\implies \exists R_u \in \mathbb{R}^{1\times(\ell+1)} \neq 0$, $R_u \mathscr{H}_{\ell+1}(u_d) = 0$
- then, left ker$\big(\mathscr{H}_{\ell+1}(w)\big)$ contains R, $\ker(R) = \mathscr{B}$ and input annihilator $\begin{bmatrix} R_u & 0 \end{bmatrix}$
- HW: how to distinguish R from $\begin{bmatrix} R_u & 0 \end{bmatrix}$?

Spurious poles

- given exact data $w \in \mathscr{B} \in \mathscr{L}^{\mathrm{p}}_{0,\ell}$ (autonomous system)
- assume that w is PE of maximal order, *i.e.*, $\operatorname{rank}\big(\mathscr{H}_{\ell+1}(w)\big) = \ell$
- the roots of $P \neq 0$, $P\mathscr{H}_{\ell+1}(w) = 0$ are the poles of $\mathscr{B}$
- consider now $\operatorname{left\,ker}\big(\mathscr{H}_{\ell+2}(w)\big)$ (ℓ is over-specified)
- what are the roots of $P \neq 0$, $P\mathscr{H}_{\ell+2}(w) = 0$?
- how to recover $\mathscr{B}$ from $\operatorname{left\,ker}\big(\mathscr{H}_{\ell+2}(w)\big)$?

Solution

- the roots of $P \in \mathbb{R}^{1\times(\ell+2)}$ are the poles of $\mathscr{B}$
  $+$ an additional pole (called spurious)

- $\dim\left(\operatorname{left}\ker\left(\mathscr{H}_{\ell+2}(w)\right)\right) = 2 \;\rightsquigarrow\;$ two independent annihilators

- their common divisor is an annihilator of $\mathscr{B}$

Identification from short record(s)

- what is the minimum number $T_{\min}$ of sequential samples needed for identification of a model in $\mathscr{L}^{q}_{\mathtt{m},\ell}$ from $N = 1$ trajectory?

- what is the minimum number $T'_{\min}$ of sequential samples for identification of a model in $\mathscr{L}^{q}_{\mathtt{m},\ell}$ if $N' > 1$ trajectories with $T'_{\min}$ samples can be used?

MPUM for noisy data

- consider "noisy data"

$$w = \overline{w} + \widetilde{w}, \qquad \text{where} \quad \overline{w} \in \overline{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell}$$

  and $\widetilde{w}$ is white noise

- under the usual assumptions $\mathscr{B}_{\text{mpum}}(\overline{w}) = \overline{\mathscr{B}}$
- what is $\mathscr{B}_{\text{mpum}}(w)$ for the noisy data?
- suggest modifications of exact identification methods that make them suitable for approximation

Solution

- a.s., there is no exact model of bounded complexity

$$\mathscr{B}_{\text{mpum}}(w)|_T \overset{\text{a.s.}}{=} (\mathbb{R}^q)^T \qquad \text{(a trivial model)}$$

- the requirement that $\widehat{\mathscr{B}}$ is unfalsified is too restrictive
- subspace identification methods: replace
  - "kernel" by "approx. kernel" obtained from the SVD
  - "rank revealing factorization" by "low-rank approx."
  - "solution (of a system)" by "LS approximation"

Impulse response estimation

- implement the method $w \to H$
- compare it with `impulseest`