DYSCO course on low-rank approximation and its applications

System identification

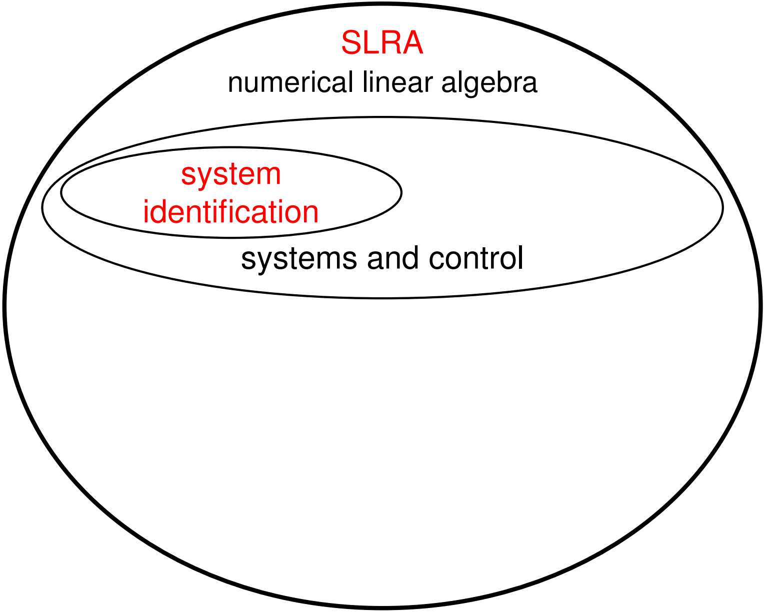
Ivan Markovsky

Vrije Universiteit Brussel

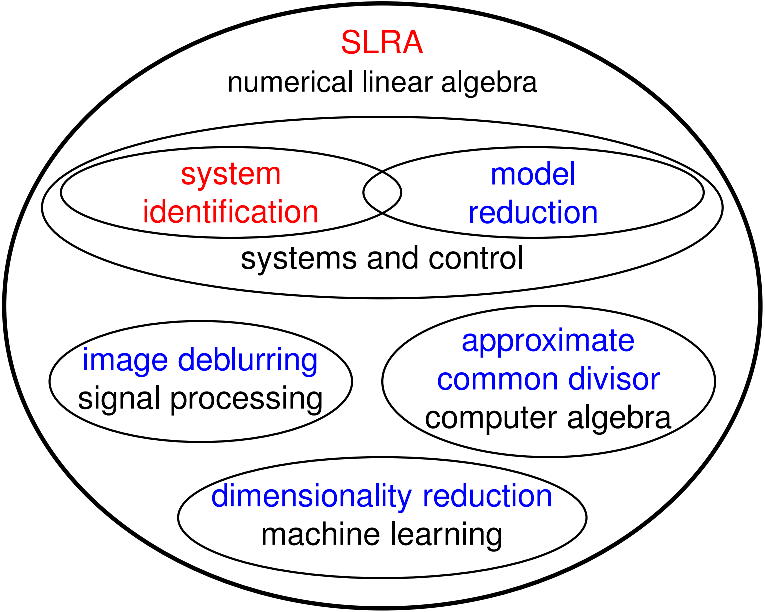
Plan

1. Introduction
2. Computational tools
3. Behavioral approach
4. System identification
5. Subspace methods
6. Generalizations

Scope


SLRA
numerical linear algebra
system
identification
systems and control

Scope


SLRA
numerical linear algebra
system
identification
model
reduction
systems and control
image deblurring
signal processing
approximate
common divisor
computer algebra
dimensionality reduction
machine learning

Outline

Introduction: data, model class, approximation

Approximation error–model complexity trade-off

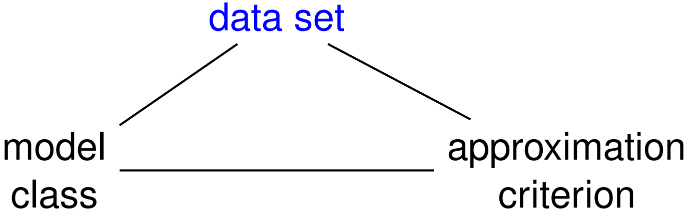
System identification $\leftrightarrow$ low-rank approximation

Solution methods: variable projection

Exercises

Outline

First is the data . . .

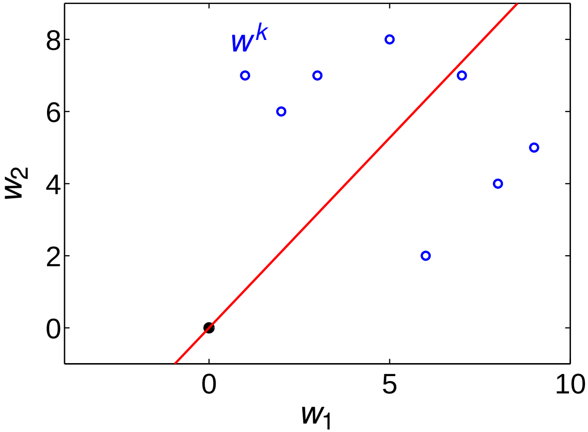


Line fitting (linear static model)

data

model —— approx.

$w^1, \ldots, w^N$ — data points (the order is not important)

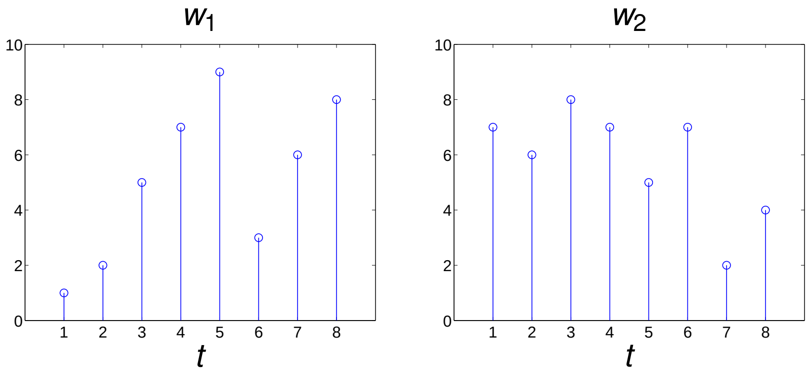


Time series data (dynamic model)

data

model —— approx.

$w(1), \dots, w(T)$ — samples in time (the order is important)



Summary: data

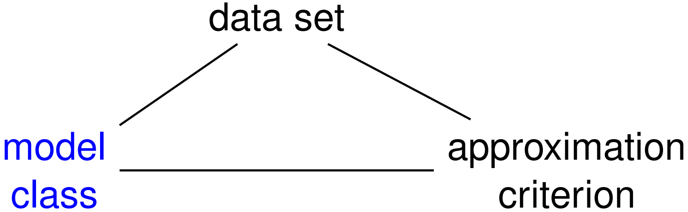
data

model ——— approx.

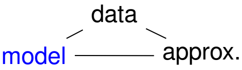
- the data is a set $w = \{ w^1, \dots, w^N \}$
- of vector valued $w^k = \begin{bmatrix} w_1^k \\ \vdots \\ w_q^k \end{bmatrix}$
- time series $w_i^k = \big(w_i^k(1), \dots, w_i^k(T_k) \big)$

  N — # of repeated experiments
  q — # of variables
  T_k — # of time samples in the kth exp.
- in static problems, $T_1 = \cdots = T_N = 1$

Next is the model class . . .

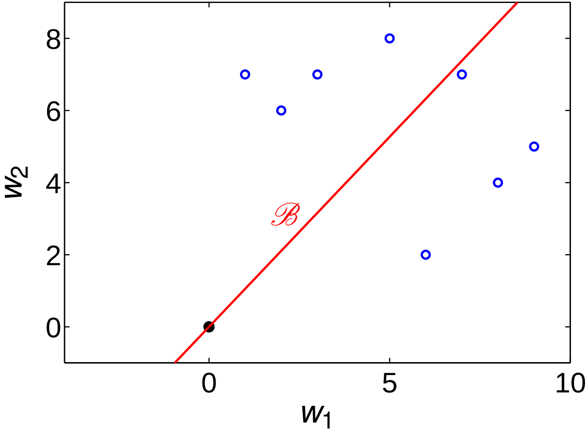


Line fitting (linear static model)

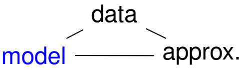


$\mathscr{B}$ — model: line through the origin

$\mathscr{M}$ — model class: all lines through the origin

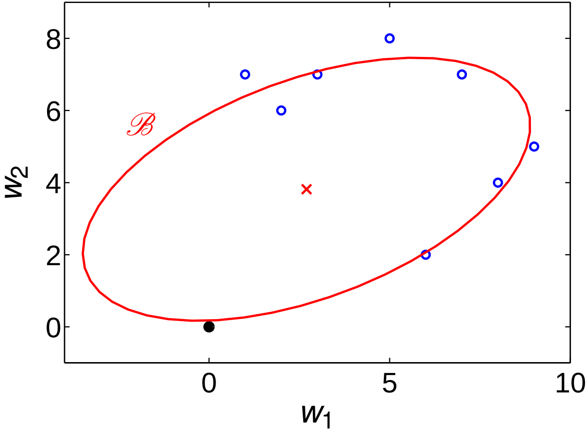


Conic section fitting (quadratic static model)



$\mathscr{B}$ — model: conic section

$\mathscr{M}$ — model class: all conic sections



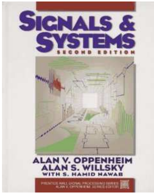
Classical definition of dynamical model

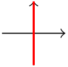
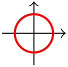
data

model ——— approx.

- dynamical model is signal processor

$$\widehat{u} \longrightarrow \boxed{\text{model}} \longrightarrow \widehat{y}$$

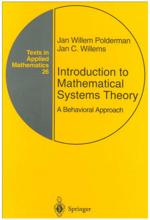


- specified by a map $\widehat{y} = f(\widehat{u})$
- "state space model", "transfer function model", ...
- however, lines and conic sections may not be graphs
- *e.g.*,  ,  can't be represented by $f : \widehat{u} \mapsto \widehat{y}$

Behavioral definition of model

data

model — approx.



- a model is a subset

$$\mathscr{B} = \{\, \widehat{w} \mid g(\widehat{w}) = 0 \text{ holds} \,\}$$

- represented by an implicit function g

- in the static case, $g(\widehat{w}) = 0$ is algebraic equation

- in the dynamic case, $g(\widehat{w}) = 0$ is difference equation

- $\widehat{w} = \begin{bmatrix} \widehat{u} \\ \widehat{y} \end{bmatrix}$, $\widehat{y} = f(\widehat{u})$ is a special case of $g(\widehat{w}) = 0$

$$(g(\widehat{u}, \widehat{y}) = \widehat{y} - f(\widehat{u}))$$

Summary: model

data
model ——— approx.

- three data modeling examples:

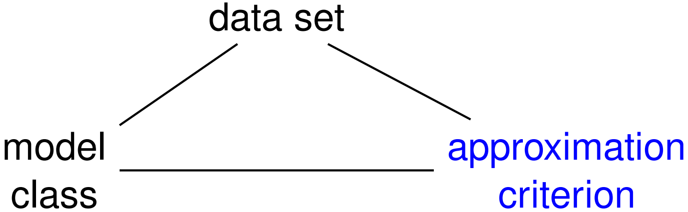
problem	model
line fitting	static linear
conic section fitting	static nonlinear
system identification	dynamic

- two definitions of a model:

classical	behavioral
map $\widehat{y} = f(\widehat{u})$	set $\{\, \widehat{w} \mid g(\widehat{w}) = 0 \,\}$
f — function	g — relation

- the classical one can not deal with all examples

Finally, the approximation criterion . . .

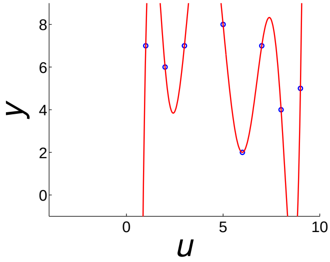


Exact model

$$w \subset \mathscr{B} \quad \iff \quad w^1, \dots, w^N \in \mathscr{B}$$
$$\iff: \quad \text{"}w \text{ is exact data of } \mathscr{B}\text{"}$$

- two well known exact modeling problems
  - realization: LTI model class, impulse resp. data
  - interpolation: static nonlinear model class

polynomial interpolation



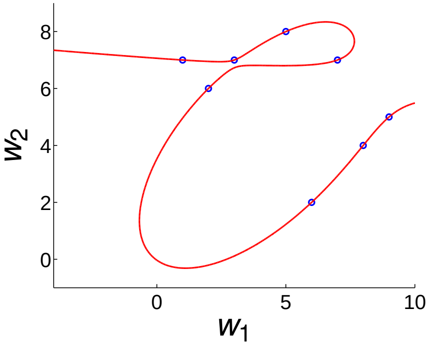
$$\mathscr{B} = \left\{ \begin{bmatrix} \widehat{u} \\ \widehat{y} \end{bmatrix} \mid \widehat{y} = f(\widehat{u}) \right\}$$

f is 8th order polynomial

Exact 3rd order nonlinear static models

$$\mathscr{B} = \left\{ \begin{bmatrix} \widehat{w}_1 \\ \widehat{w}_2 \end{bmatrix} \;\middle|\; g(\widehat{w}_1, \widehat{w}_2) = 0 \right\}$$

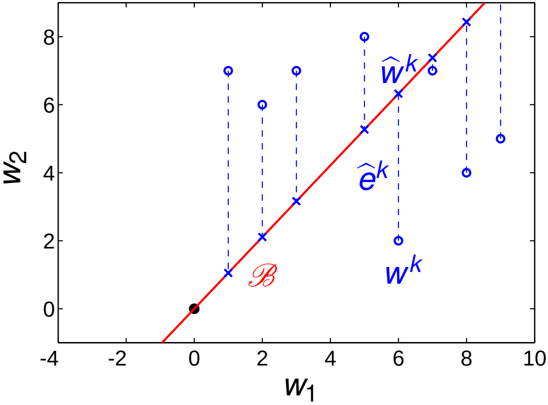
g is 3rd order polynomial in $\widehat{w}_1$, $\widehat{w}_2$



Ordinary least squares

data

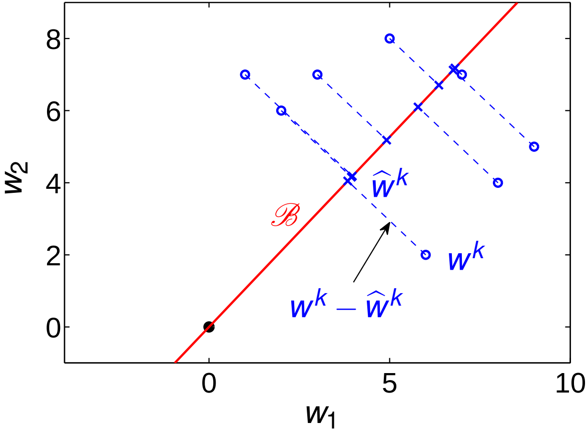
model — approx.

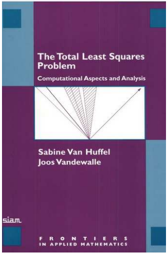


Total least squares

data

model ——— approx.

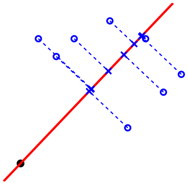




Linear static case

data

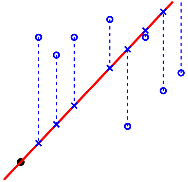
model —— approx.



- total least squares

$$\min_{\widehat{u},\widehat{y},\theta} \left\| \begin{bmatrix} u-\widehat{u} & y-\widehat{y} \end{bmatrix} \right\|_{\mathrm{F}} \text{ s.t. } \underbrace{\widehat{u}\theta = \widehat{y}}_{(\widehat{u},\widehat{y})\subset\mathscr{B}(\theta)}$$

$\widehat{w} = (\widehat{u},\widehat{y})$ approximates $w = (u,y)$



- ordinary least squares

$$\min_{\widehat{e},\theta} \|\widehat{e}\|_2 \quad \text{s.t.} \quad \underbrace{u\theta = y + \widehat{e}}_{(\widehat{e},u,y)\subset\mathscr{B}_{\text{ext}}(\theta)}$$

$\widehat{e}$ is unobserved (latent) input

Approximation criteria

- Misfit approach:

  modify w as little as possible,
  so that $\widehat{w}$ is exact

  $\|w - \widehat{w}\|$ is the misfit criterion

- Latency approach:

  augment $\mathscr{B}$ by as small as possible e,
  so that (e, w) is exact

  $\|e\|$ is the latency criterion

Deterministic vs stochastic setting

data

model —— approx.

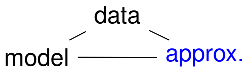
- stochastic estimation $\leftrightarrow$ deterministic approx.

$$\text{noise model} \xleftrightarrow{\text{maximum likelihood}} \text{cost function}$$

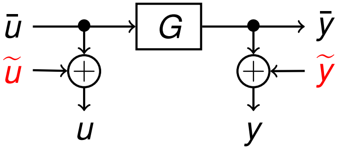
- also in control

LQG control $\leftrightarrow$ H_2 optimal control

Misfit and latency in the stochastic setting



EIV $\leftrightarrow$ misfit

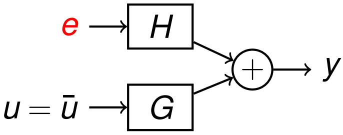


$\tilde{u}$, $\tilde{y}$ — measurement errors

$$\min_{\hat{w} \subset \mathscr{B}} \; \| w - \hat{w} \|$$

$$\mathscr{B} := \left\{ \begin{bmatrix} \hat{u} \\ \hat{y} \end{bmatrix} \mid \hat{y} = \hat{G}\hat{u} \right\}$$

ARMAX $\leftrightarrow$ latency



e — disturbance

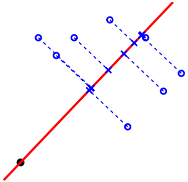
$$\min_{(\hat{e}, w) \subset \mathscr{B}_{\text{ext}}} \; \| \hat{e} \|$$

$$\mathscr{B}_{\text{ext}} := \left\{ \begin{bmatrix} \hat{e} \\ u \\ y \end{bmatrix} \mid y = \begin{bmatrix} \hat{H} & \hat{G} \end{bmatrix} \begin{bmatrix} \hat{e} \\ u \end{bmatrix} \right\}$$

Summary: approximation criterion

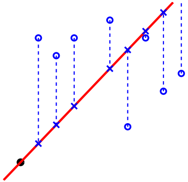
data

model —— approx.



- TLS $\leftrightarrow$ misfit $\leftrightarrow$ errors-in-variables

$$\min_{\widehat{w}\subset\mathscr{B}} \|w-\widehat{w}\| \qquad \begin{pmatrix}\text{projection}\\ \text{of } w \text{ on } \mathscr{B}\end{pmatrix}$$



- OLS $\leftrightarrow$ latency $\leftrightarrow$ ARMAX

$$\min_{(\widehat{e},w)\in\mathscr{B}_{\text{ext}}} \|\widehat{e}\|$$

Outline

Introduction: data, model class, approximation

Approximation error–model complexity trade-off

System identification $\leftrightarrow$ low-rank approximation

Solution methods: variable projection

Exercises

A general problem

$$\begin{array}{ccc} \text{data} & & \text{model} \\ w & \xrightarrow{\text{identification}} & \mathscr{B} \in \mathscr{M} \end{array}$$

the aim is to obtain "simple" and "accurate" model:

"accurate"	$\rightarrow$	min. error$(w, \widehat{\mathscr{B}})$ = misfit/latency
"simple"	$\rightarrow$	Occam's razor principle: among equally accurate models, choose the simplest

Model complexity

- simple models are small models

$$\mathscr{B}_1 \subset \mathscr{B}_2 \quad \Longrightarrow \quad \mathscr{B}_1 \text{ is simpler than } \mathscr{B}_2$$

- nonlinear model complexity is an open problem
- in the linear time-invariant case, $\mathscr{B}$ is a subspace

$$\text{size of the model} = \text{dimension of } \mathscr{B}$$

- however, models with inputs are infinite dimensional

Linear time-invariant model's complexity

- restriction of $\mathscr{B}$ on an interval $[1, T]$

$$\mathscr{B}|_T = \{\, w = \big(w(1), \dots, w(T)\big) \mid \exists\, w_{\mathrm{p}}, w_{\mathrm{f}}, \text{ such that } (w_{\mathrm{p}}, w, w_{\mathrm{f}}) \in \mathscr{B} \,\}$$

- for sufficiently large T

$$\dim(\mathscr{B}|_T) = (\#\text{ of inputs}) \cdot T + (\text{order})$$

$$\text{complexity}(\mathscr{B}) = \begin{bmatrix} \mathtt{m} \\ \ell \end{bmatrix} \begin{matrix} \to & \#\text{ of inputs} \\ \to & \text{order or lag} \end{matrix}$$

- $\mathscr{L}_{\mathtt{m},\ell}$ — set of LTI systems of bounded complexity

Complexity selection

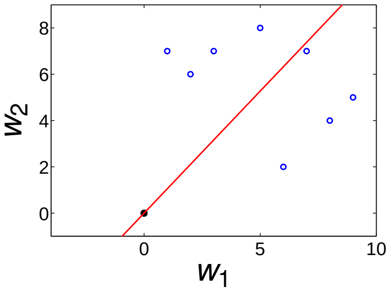
- if $\mathtt{m}$ is given and fixed, choosing the complexity is an

  order selection problem

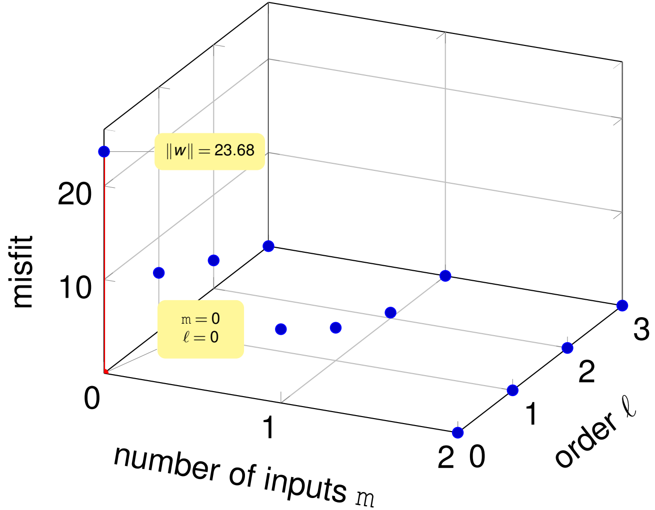
- in general, choosing the complexity involves

  order selection and input selection

  illustrated next on the example from the introduction

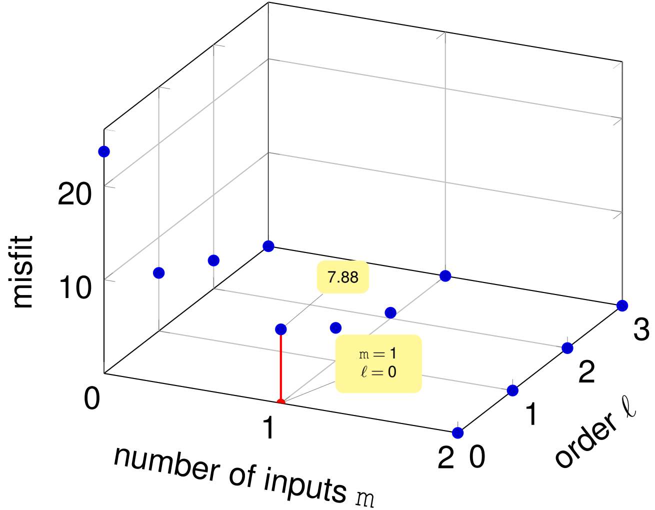


Example: misfit-complexity trade-off



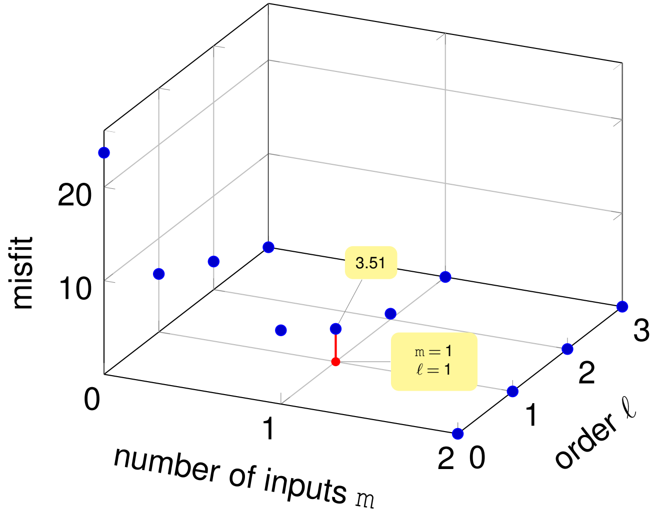
$\mathrm{m} = 0$, $\ell = 0 \implies \mathscr{B} = \{0\}$ is the only model

Example: misfit-complexity trade-off



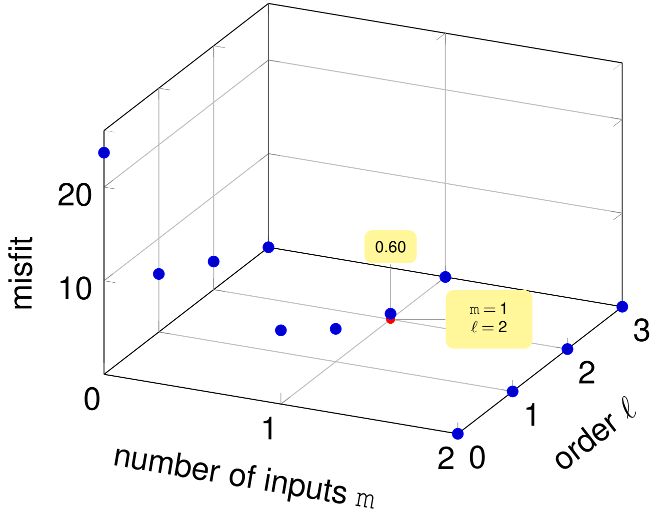
$\mathtt{m} = 1$, $\ell = 0 \implies \mathscr{B}$ is a line through 0

Example: misfit-complexity trade-off



$\mathtt{m} = 1$, $\ell = 1$ $\implies$ $\mathscr{B}$ is 1st order SISO

Example: misfit-complexity trade-off



$\mathtt{m} = 1$, $\ell = 2 \implies \mathscr{B}$ is 2nd order SISO

Approximation error-complexity trade-off

$$\text{minimize} \quad \text{over } \widehat{\mathscr{B}} \in \mathscr{L} \quad \begin{bmatrix} \text{error}(w, \widehat{\mathscr{B}}) \\ \text{complexity}(\widehat{\mathscr{B}}) \end{bmatrix}$$

three ways to "scalarize" the problem:

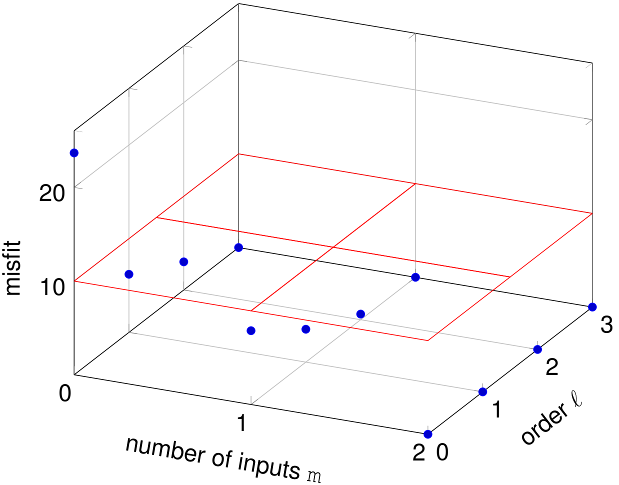
1. minimize over $\widehat{\mathscr{B}} \in \mathscr{L}$ $\text{error}(w, \widehat{\mathscr{B}}) + \lambda \, \text{complexity}(\widehat{\mathscr{B}})$

2. $$\begin{array}{ll} \text{minimize} & \text{over } \widehat{\mathscr{B}} \in \mathscr{L} \quad \text{complexity}(\widehat{\mathscr{B}}) \\ \text{subject to} & \text{error}(w, \widehat{\mathscr{B}}) \leq \mu \end{array}$$

3. $$\begin{array}{ll} \text{minimize} & \text{over } \widehat{\mathscr{B}} \quad \text{error}(w, \widehat{\mathscr{B}}) \\ \text{subject to} & \widehat{\mathscr{B}} \in \mathscr{L}_{\mathtt{m},\ell} \end{array}$$

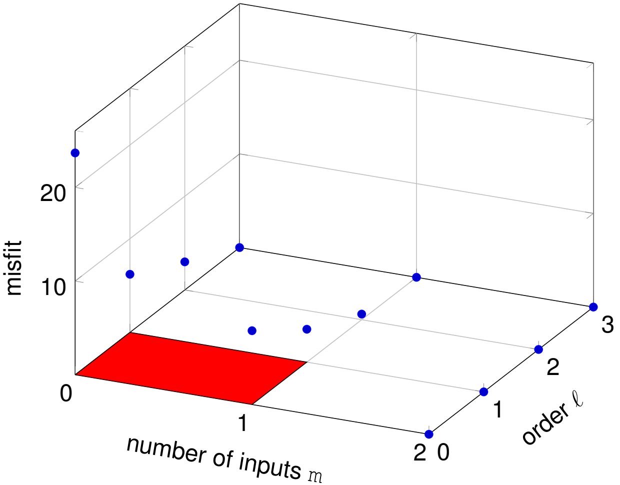
Complexity minimization with error bound

$$\begin{array}{ll} \text{minimize} & \text{over } \widehat{\mathscr{B}} \in \mathscr{L} \quad \text{complexity}(\widehat{\mathscr{B}}) \\ \text{subject to} & \text{error}(w, \widehat{\mathscr{B}}) \leq \mu \end{array}$$



Error minimization with complexity bound

$$\begin{aligned} &\text{minimize} \quad \text{over } \widehat{\mathscr{B}} \quad \text{error}(w, \widehat{\mathscr{B}}) \\ &\text{subject to} \quad \widehat{\mathscr{B}} \in \mathscr{L}_{\mathtt{m},\ell} \end{aligned}$$



Summary: error-complexity trade-off

- LTI model complexity

$$\text{complexity}(\mathscr{B}) = \begin{bmatrix} \mathtt{m} \\ \ell \end{bmatrix} \begin{array}{ll} \rightarrow & \text{\# of inputs} \\ \rightarrow & \text{order or lag} \end{array}$$

- error-complexity trade-off

$$\text{minimize} \quad \text{over } \widehat{\mathscr{B}} \in \mathscr{L} \quad \begin{bmatrix} \text{error}(w, \widehat{\mathscr{B}}) \\ \text{complexity}(\widehat{\mathscr{B}}) \end{bmatrix}$$

- tracing all optimal solutions requires hyper parameter
  1. λ — no physical meaning
  2. μ — bound on the error
  3. $(\mathtt{m}, \ell)$ — bound on the complexity

Outline

Approximate identification problem

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

- in the case error $=$ misfit

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

- the problem is

$$\text{minimize} \quad \text{over } \widehat{\mathscr{B}},\ \widehat{w} \quad \|w - \widehat{w}\|$$
$$\text{subject to} \quad \widehat{w} \in \widehat{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell}$$

Exact, noisy, and missing data

- $v_i^k(t)$ — variance of the measurement noise on $w_i^k(t)$

$$\|w - \widehat{w}\|_\alpha^2 = \sum_{k=1}^{N} \sum_{i=1}^{q} \sum_{t=1}^{T} \alpha_i^k(t) \big(w_i^k(t) - \widehat{w}_i^k(t)\big)^2$$

noisy data

$$\alpha_i^k(t) := \frac{1}{v_i^k(t)}$$

exact data

$$v_i^k(t) = 0, \; \alpha_i^k(t) = \infty$$

missing data

$$v_i^k(t) = \infty, \; \alpha_i^k(t) = 0$$

- $v_i^k(t) = \infty$ imposes equality constraint $\widehat{w}_i^k(t) = w_i^k(t)$
- $v_i^k(t) = 0$ makes $\|w - \widehat{w}\|_\alpha^2$ independent of $w_i^k(t)$

Summary: identification problem

- approximate identification in the misfit setting

$$\begin{array}{lll} \text{minimize} & \text{over } \widehat{\mathscr{B}},\ \widehat{w} & \|w - \widehat{w}\|_\alpha \\ \text{subject to} & \widehat{w} \in \widehat{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell} & \end{array} \qquad \text{(SYSID)}$$

- element-wise weighted error criterion $\|\cdot\|_\alpha$

$$\begin{array}{llll} \text{exact} & w_i^k(t) & \leftrightarrow & \alpha_i^k(t) = \infty \\ \text{missing} & w_i^k(t) & \leftrightarrow & \alpha_i^k(t) = 0 \end{array}$$

Next: SYSID $\leftrightarrow$ Hankel structured LRA

exact trajectory $w \in \mathscr{B} \in \mathscr{L}_{\mathtt{m},\ell}$

$$\Updownarrow$$

$$R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0$$

$$\Updownarrow$$

rank deficient

$$\mathscr{H}_{\ell+1}(w) := \begin{bmatrix} w(1) & w(2) & \cdots & w(T-\ell) \\ w(2) & w(3) & \cdots & w(T-\ell+1) \\ w(3) & w(4) & \cdots & w(T-\ell+2) \\ \vdots & \vdots & & \vdots \\ w(\ell+1) & w(\ell+2) & \cdots & w(T) \end{bmatrix}$$

$w \in \mathscr{B} \iff \mathscr{H}_{\ell+1}(w)$ rank deficient

- relation at time $t = 1$

$$R_0 w(1) + R_1 w(2) + \cdots + R_\ell w(\ell+1) = 0$$

- in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(1) \\ w(2) \\ \vdots \\ w(\ell+1) \end{bmatrix} = 0$$

$w \in \mathscr{B} \iff \mathscr{H}_{\ell+1}(w)$ rank deficient

- relation at time $t = 2$

$$R_0 w(2) + R_1 w(3) + \cdots + R_\ell w(\ell+2) = 0$$

- in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(2) \\ w(3) \\ \vdots \\ w(\ell+2) \end{bmatrix} = 0$$

$w \in \mathscr{B} \iff \mathscr{H}_{\ell+1}(w)$ rank deficient

- relation at time $t = T - \ell$

$$R_0 w(T-\ell) + R_1 w(T-\ell+1) + \cdots + R_\ell w(T) = 0$$

- in matrix form:

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix} \begin{bmatrix} w(T-\ell) \\ w(T-\ell+1) \\ w(T-\ell+2) \\ \vdots \\ w(T) \end{bmatrix} = 0$$

Putting it all together

- relation for $t = 1, \ldots, T - \ell$

$$R_0 w(t) + R_1 w(t+1) + \cdots + R_\ell w(t+\ell) = 0$$

- in matrix form:

$$\underbrace{\begin{bmatrix} R_0 & R_1 & \cdots & R_\ell \end{bmatrix}}_{R} \underbrace{\begin{bmatrix} w(1) & w(2) & \cdots & w(T-\ell) \\ w(2) & w(3) & \cdots & w(T-\ell+1) \\ w(3) & w(4) & \cdots & w(T-\ell+2) \\ \vdots & \vdots & & \vdots \\ w(\ell+1) & w(\ell+2) & \cdots & w(T) \end{bmatrix}}_{\mathscr{H}_{\ell+1}(w)} = 0$$

$w \in \mathscr{B} \iff \mathscr{H}_{\ell+1}(w)$ rank deficient

- with $R \in \mathbb{R}^{(q-\mathtt{m})\times q(\ell+1)}$ full row rank,

$$\operatorname{rank}\left(\mathscr{H}_{\ell+1}(w) = 0\right) \leq q\ell + \mathtt{m} \qquad (q \text{ — \# of variables})$$

$$w \in \mathscr{B} \in \mathscr{L}_{\mathtt{m},\ell} \iff \operatorname{rank}\left(\mathscr{H}_{\ell+1}(w)\right) \leq q\ell + \mathtt{m}$$

- multiple time-series $\quad\leftrightarrow\quad$ mosaic-Hankel matrix

$$\{ w^1, \dots, w^N \} \subset \mathscr{B} \in \mathscr{L}_{\mathtt{m},\ell}$$
$$\iff \operatorname{rank}\Big(\underbrace{\begin{bmatrix} \mathscr{H}_{\ell+1}(w^1) & \cdots & \mathscr{H}_{\ell+1}(w^N) \end{bmatrix}}_{\mathscr{H}_{\ell+1}(w)} \Big) \leq q\ell + \mathtt{m}$$

Structured weighted low-rank approximation

$$\begin{array}{lll} \text{minimize} & \text{over } \widehat{\mathscr{B}} \text{ and } \widehat{w} & \|w - \widehat{w}\|_\alpha \\ \text{subject to} & \widehat{w} \subset \widehat{\mathscr{B}} \in \mathscr{L}_{\mathrm{m},\ell} & \end{array} \qquad \text{(SYSID)}$$

$$\Updownarrow$$

$$\begin{array}{lll} \text{minimize} & \text{over } \widehat{w} & \|w - \widehat{w}\|_\alpha \\ \text{subject to} & \operatorname{rank}\big(\mathscr{H}_{\ell+1}(\widehat{w})\big) \le q\ell + \mathrm{m} & \end{array} \qquad \text{(SLRA)}$$

Summary: structured low-rank approximation

- (SYSID) $\iff$ (SLRA)
- LTI model class $\iff$ Hankel structure
- repeated experiments $\iff$ mosaic-Hankel structure

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

- bounded complexity $\iff$ rank constraint

$$(\mathtt{m}, \ell) \quad \leftrightarrow \quad r = q\ell + \mathtt{m}$$

Outline

Solution methods

- given: data w and complexity bound $(\mathtt{m}, \ell)$
- find: $\widehat{\mathscr{B}}$ that solves (SYSID) or, equivalently, (SLRA)

1. choice of model representation
   - transfer function
   - input/state/output
   - ...
2. choice of optimization method
   - local optimization
   - global optimization
   - convex relaxations

Model vs model representation

- 1st order SISO model $\mathscr{B} \in \mathscr{L}_{1,1}$

$$\mathscr{B}_{\text{de}}(\theta) = \left\{ \widehat{w} \;\middle|\; \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix} \begin{bmatrix} \widehat{w}_1(t) \\ \widehat{w}_2(t) \\ \widehat{w}_1(t+1) \\ \widehat{w}_2(t+1) \end{bmatrix} = 0, \ \forall t \right\}$$

- transfer functions

$$G_{w_1 \mapsto w_2}(z) = -\frac{\theta_1 + \theta_3 z}{\theta_2 + \theta_4 z} \quad , \quad G_{w_2 \mapsto w_1}(z) = -\frac{\theta_2 + \theta_4 z}{\theta_1 + \theta_3 z}$$

- state space, convolution, ..., representations

Problem formulation vs solution method

- in the classical setting, model $=$ representation
- $\implies$ problems are mixed with solution methods
- *e.g.*, "total least-squares" is both problem and method
- the behavioral setting distinguishes

	used for	involves
abstract	problem formulation	$\mathscr{B}$, $\mathscr{L}_{\mathrm{m},\ell}$
concrete	solution methods	$\mathscr{B}(\theta)$, $\theta \in \Theta$

- low-rank approx. is abstract problem formulation

Parameter optimization problem

- model representation

$$\mathscr{B}(\theta) = \{\, \widehat{w} \mid g_\theta(\widehat{w}) = 0 \,\}$$

- parameterized model class

$$\mathscr{M} = \{\, \mathscr{B}(\theta) \mid \theta \in \Theta \,\}$$

- optimization problem

$$\begin{array}{lll} \text{minimize} & \text{over } \theta \in \Theta,\ \widehat{w} & \|w - \widehat{w}\|_\alpha \\ \text{subject to} & g_\theta(\widehat{w}) = 0 & \end{array} \qquad (\mathrm{SYSID}_\theta)$$

Bilinear structure of the problem

- (SYSID$_\theta$) — constrained nonlinear least-squares
- $\mathscr{B}$ linear

$$\implies \quad g_\theta(\widehat{w}) \text{ bilinear (in } \theta \text{ and } \widehat{w})$$

$$\implies \quad (\text{SYSID}_\theta) \text{ can be solved globally for given } \theta$$

- variable projection (VARPRO)
  for separable nonlinear least-squares problems
- if $T \gg \ell$, elimination of $\widehat{w}$ leads to big reduction

System theoretic view of VARPRO

solving (SYSID_θ) for given θ

$$\Updownarrow$$

misfit evaluation: $\text{error}\big(w, \mathscr{B}(\theta)\big)$

$$\Updownarrow$$

likelihood evaluation

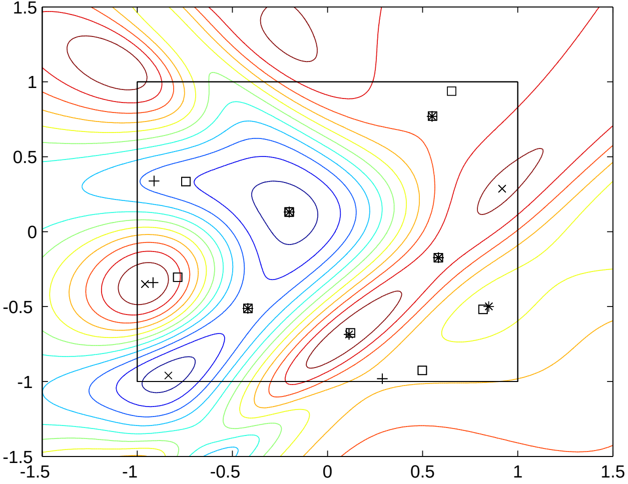
$$\Updownarrow$$

least-squares smoothing of w by $\mathscr{B}(\theta)$

$$\Updownarrow$$

fast algorithms:
Kalman smoothing
Cholesky factorization
...

Non-convexity of error$(w, \mathscr{B}(\theta))$



Computational details

- $O(T)$ evaluation of $\text{error}\big(w, \mathscr{B}(\theta)\big)$ and its derivatives
  - using the Kalman smoother
  - Cholesky factorization of banded Toeplitz matrix
  - ...
- $\mathscr{B}(\theta) = \mathscr{B}(\alpha\theta)$, for all $\alpha \neq 0$
- $\Theta = \{\, \theta \mid \|\theta\|_2 = 1 \,\} \implies$ optimization on a manifold
  - generic methods (optimization theory)
  - custom methods (system identification)
    - data driven local coordinates (McKelvey)
    - ...

Summary: solution methods

- solution methods involve two choices:
  1. model representation
  2. optimization method
- in the linear case, bilinear structure $\rightsquigarrow$ VARPRO
- constraint nonlinear least-squares problem

$$\min_{\theta \in \Theta} \; \text{error}\big(w, \mathscr{B}(\theta)\big)$$

- Θ is a manifold $\rightsquigarrow$ optimization on a manifold

Outline

Introduction: data, model class, approximation

Approximation error–model complexity trade-off

System identification $\leftrightarrow$ low-rank approximation

Solution methods: variable projection

Exercises