My interest in dynamic measurement started from a textbook problem

"A thermometer reading $21^\circ C$*, which has been inside a house for a long time, is taken outside. After one minute the thermometer reads* $15^\circ C$*; after two minutes it reads* $11^\circ C$*. What is the outside temperature?"*

According to Newton's law of cooling, an object of higher temperature than its environment cools at a rate that is proportional to the difference in temperature.

Main idea: predict the steady-state value from the first few samples of the transient

textbook problem:

- 1st order dynamics
- 3 noise-free samples
- batch solution

generalizations:

- $n \geq 1$ order dynamics
- $T \geq 3$ noisy (vector) samples
- recursive computation

implementation and practical validation

Thermometer: first order dynamical system

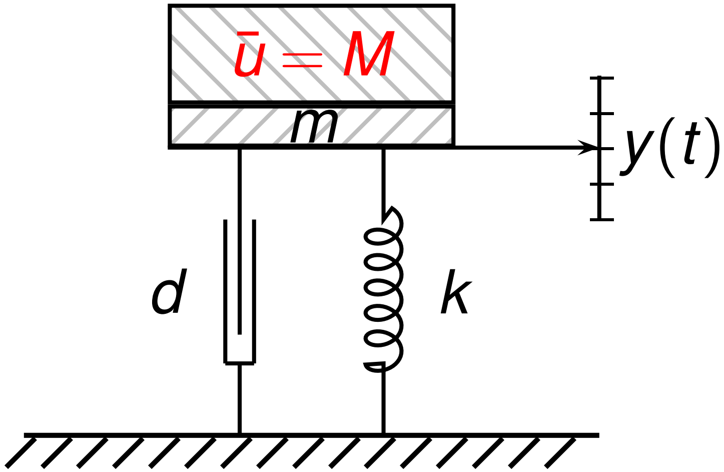
environmental temperature $\bar{u}$ $\xrightarrow{\text{heat transfer}}$ thermometer's reading y

measurement process: Newton's law of cooling

$$\dot{y} = a(\bar{u} - y)$$

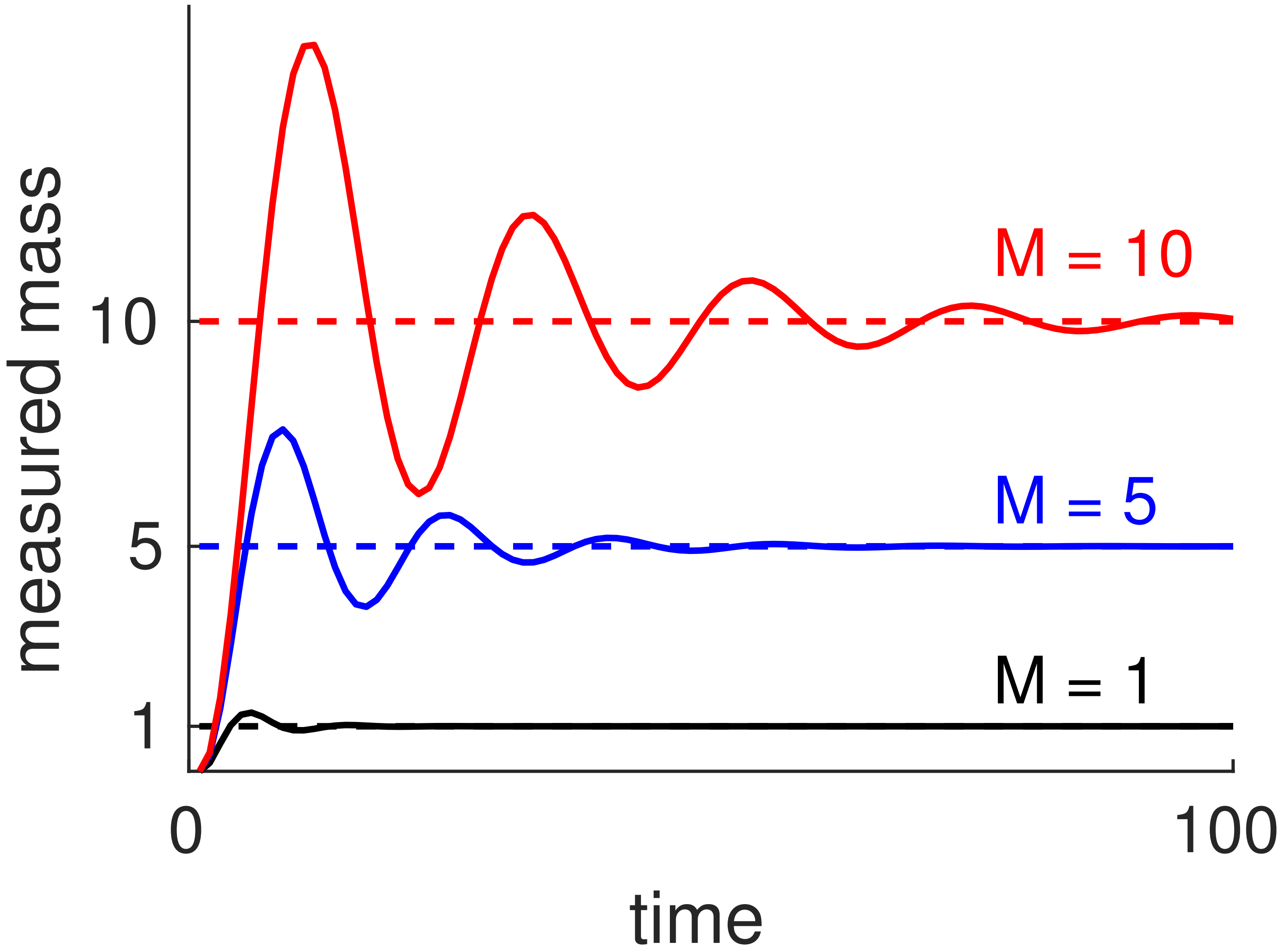
heat transfer coefficient $a > 0$

Scale: second order dynamical system



$$(M + m)\ddot{y} + d\dot{y} + ky = g\bar{u}$$

The measurement process dynamics depends on the to-be-measured mass



Dynamic measurement: take into account the dynamical properties of the sensor

to-be-measured variable u $\xrightarrow{\text{measurement process}}$ measured variable y

assumption 1: measured variable is constant $u(t) = \bar{u}$

assumption 2: the sensor is stable LTI system

assumption 3: sensor's DC-gain $= 1$ (calibrated sensor)

Outline

Structured low-rank approximation

Case study: Dynamic measurement

Model-based vs data-driven methods

The data is generated from LTI system with output noise and constant input

$$\underbrace{y_\text{d}}_{\substack{\text{measured}\\\text{data}}} = \underbrace{y}_{\substack{\text{true}\\\text{value}}} + \underbrace{e}_{\substack{\text{measurement}\\\text{noise}}}$$

$$\underbrace{y}_{\substack{\text{true}\\\text{value}}} = \underbrace{\bar{u}}_{\substack{\text{steady-state}\\\text{value}}} + \underbrace{y_0}_{\substack{\text{transient}\\\text{response}}}$$

assumption 4: e is a zero mean, white, Gaussian noise using a state space representation of the sensor

$$\begin{aligned} x(t+1) &= Ax(t), \qquad x(0) = x_0 \\ y_0(t) &= cx(t) \end{aligned}$$

we obtain

$$\underbrace{\begin{bmatrix} y_d(1) \\ y_d(2) \\ \vdots \\ y_d(T) \end{bmatrix}}_{y_d} = \underbrace{\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}}_{\mathbf{1}_T} \bar{u} + \underbrace{\begin{bmatrix} c \\ cA \\ \vdots \\ cA^{T-1} \end{bmatrix}}_{\mathscr{O}_T} x_0 + \underbrace{\begin{bmatrix} e(1) \\ e(2) \\ \vdots \\ e(T) \end{bmatrix}}_{e}$$

Maximum-likelihood model-based estimator

solve approximately

$$\begin{bmatrix} \mathbf{1}_T & \mathscr{O}_T \end{bmatrix} \begin{bmatrix} \widehat{u} \\ \widehat{x}_0 \end{bmatrix} \approx y_\mathrm{d}$$

standard least-squares problem

$$\text{minimize} \quad \text{over } \widehat{y}, \widehat{u}, \widehat{x}_0 \quad \|y_\mathrm{d} - \widehat{y}\|$$
$$\text{subject to} \quad \begin{bmatrix} \mathbf{1}_T & \mathscr{O}_T \end{bmatrix} \begin{bmatrix} \widehat{u} \\ \widehat{x}_0 \end{bmatrix} = \widehat{y}$$

recursive implementation $\rightsquigarrow$ Kalman filter

Subspace model-free method

goal: avoid using the model parameters (A, C, $\mathscr{O}_T$)

in the noise-free case, due to the LTI assumption,

$$\Delta y(t) := y(t) - y(t-1) = y_0(t) - y_0(t-1)$$

satisfies the same dynamics as y_0, *i.e.*,

$$\begin{aligned} x(t+1) &= Ax(t), \qquad x(0) = \Delta x \\ \Delta y(t) &= cx(t) \end{aligned}$$

Hankel matrix—construction of multiple "short" trajectories from one "long" trajectory

$$\mathscr{H}(\Delta y) := \begin{bmatrix} \Delta y(1) & \Delta y(2) & \cdots & \Delta y(n) \\ \Delta y(2) & \Delta y(3) & \cdots & \Delta y(n+1) \\ \Delta y(3) & \Delta y(4) & \cdots & \Delta y(n+2) \\ \vdots & \vdots & & \vdots \\ \Delta y(T-n) & \Delta y(T-n) & \cdots & \Delta y(T-1) \end{bmatrix}$$

fact: if rank $\mathscr{H}(\Delta y) = n$, then

$$\text{image}\, \mathscr{O}_{T-n} = \text{image}\, \mathscr{H}(\Delta y)$$

model-based equation

$$\begin{bmatrix} \mathbf{1}_T & \mathscr{O}_T \end{bmatrix} \begin{bmatrix} \bar{u} \\ \widehat{x}_0 \end{bmatrix} = y$$

data-driven equation

$$\begin{bmatrix} \mathbf{1}_{T-n} & \mathscr{H}(\Delta y) \end{bmatrix} \begin{bmatrix} \bar{u} \\ \ell \end{bmatrix} = y|_{T-n} \qquad (*)$$

subspace method

solve $(*)$ by (recursive) least squares

Generalizations

vector measurement (data fusion)

the method applies as it is

derivation of confidence bounds

errors-in-variables regression problem

estimation of time-varying parameters

use exponentially weighted least-squares

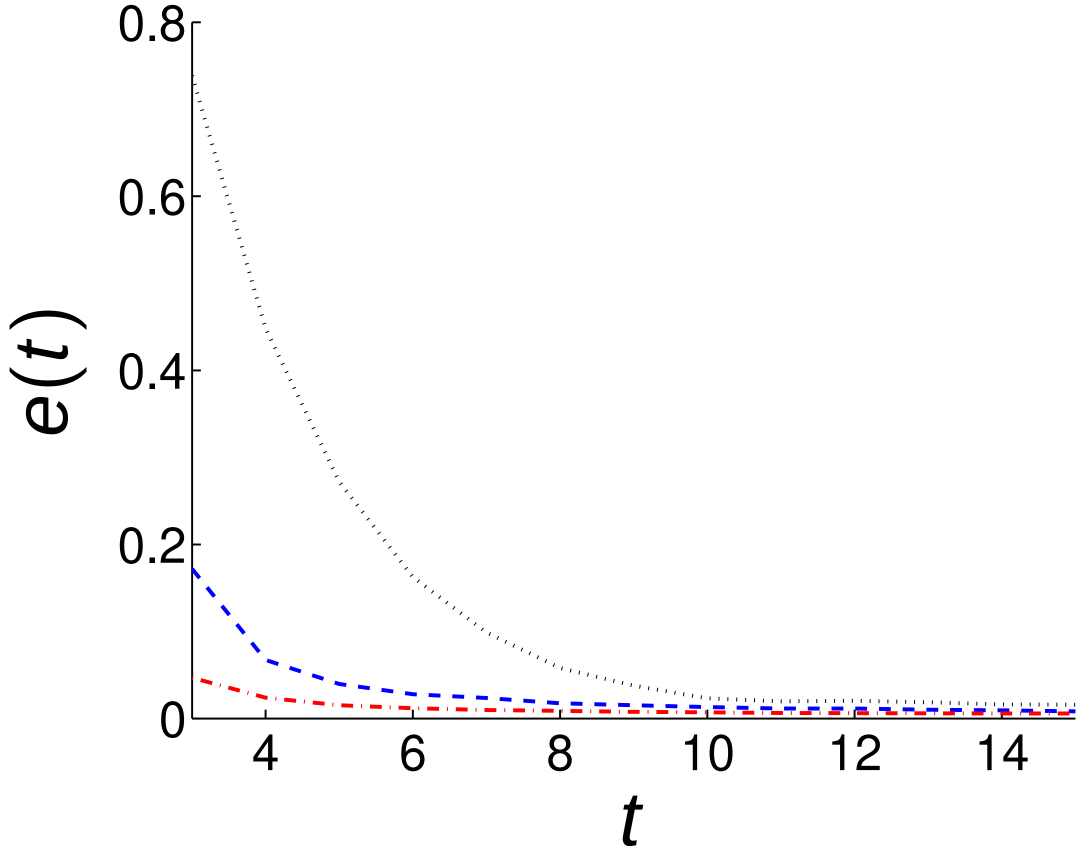
Empirical validation

dashed	—	true parameter value $\bar{u}$
solid	—	true output trajectory y_0
dotted	—	naive estimate $\widehat{u} = G^{+} y$
dashed	—	model-based Kalman filter
bashed-dotted	—	data-driven method

estimation error: $e := \frac{1}{N} \sum_{i=1}^{N} \| \bar{u} - \widehat{u}^{(i)} \|$

(for $N = 100$ Monte-Carlo repetitions)

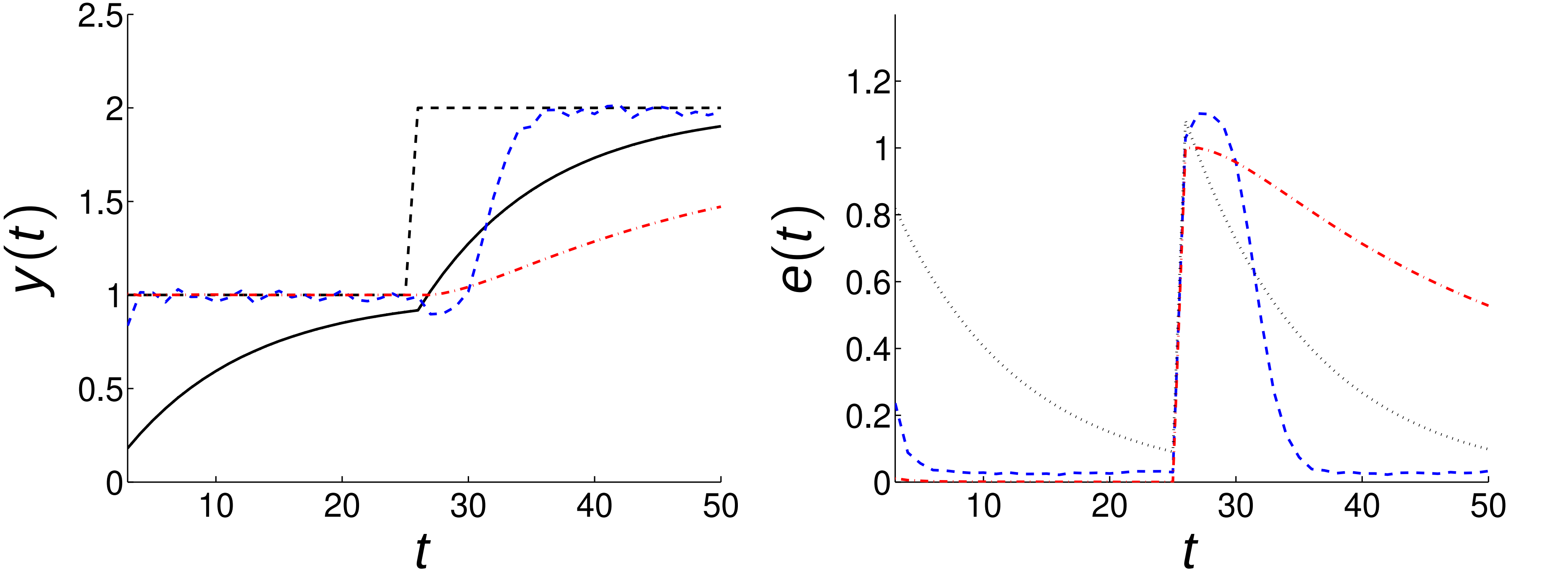
Simulated data of dynamic cooling process



$e(t) \to 0$ as $t \to \infty$ at different rates

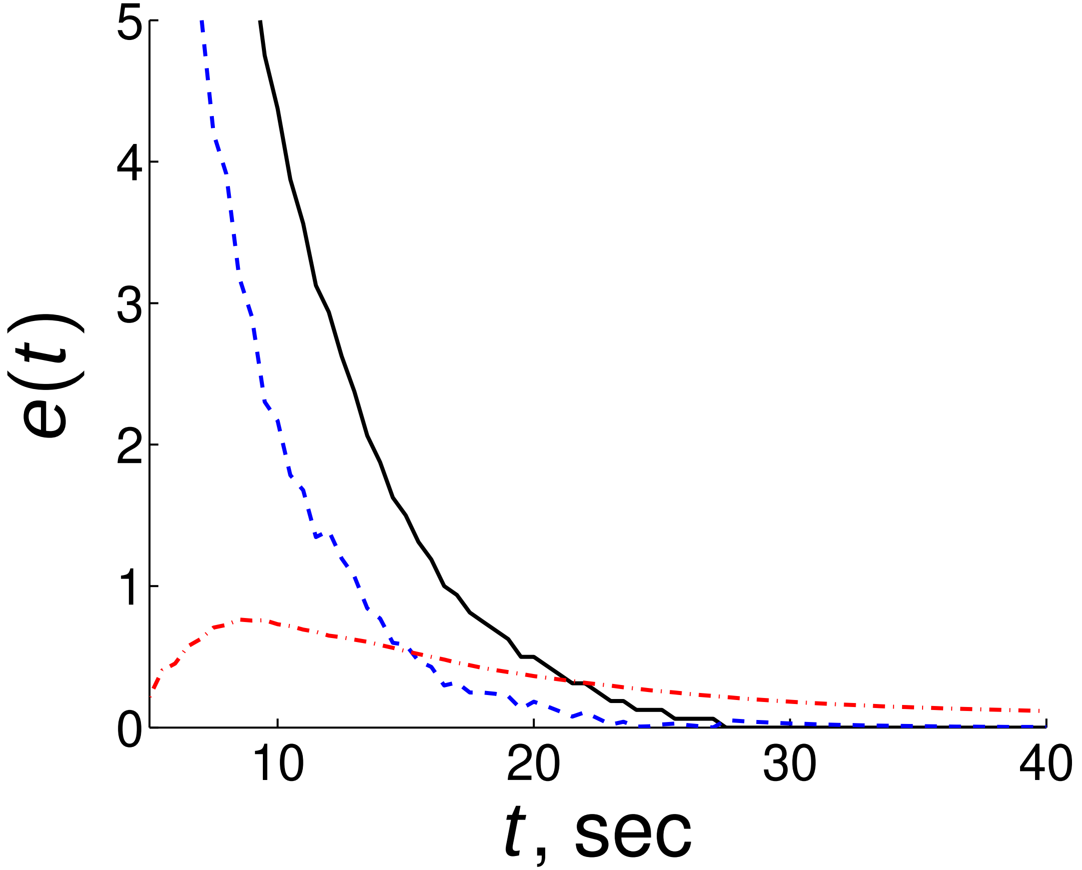
best is the Kalman filter (maximum likelihood estimator)

Simulation with time-varying parameter



Proof of concept prototype

Results in real-life experiment



Valorization