Solving ill-posed estimation problems through regularization: a brief introduction with examples

Damiano Varagnolo

Feb. 8, 2017

Mathematics and its Applications @ LTU

LULEÅ UNIVERSITY OF TECHNOLOGY

aim: show usefulness of regularization when doing statistical estimation

Structure

- the Stein phenomenon
- ill-conditioning
- example: the Hunt problem
- Phillips-Tikhonov nonparametric regularization
- regularization for system identification

Structure

- the Stein phenomenon
- ill-conditioning
- example: the Hunt problem
- Phillips-Tikhonov nonparametric regularization
- regularization for system identification

- some more mathematical details

the Stein phenomenon

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

aim: find estimator of $\boldsymbol{\theta}$ that minimizes $\quad \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|^2\right]$

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

aim: find estimator of $\boldsymbol{\theta}$ that minimizes $\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|^2\right]$

idea: use $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \boldsymbol{y} \implies \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} - \boldsymbol{\theta}\right\|^2\right] = N\sigma^2$?

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

aim: find estimator of $\boldsymbol{\theta}$ that minimizes $\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|^2\right]$

idea: use $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \boldsymbol{y} \quad \Longrightarrow \quad \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} - \boldsymbol{\theta}\right\|^2\right] = N\sigma^2$?

requirement: to be a good estimator of $\boldsymbol{\theta}$, $\widehat{\boldsymbol{\theta}}$ should be s.t. $\widehat{\boldsymbol{\theta}}^T \widehat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}^T \boldsymbol{\theta}$

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

aim: find estimator of $\boldsymbol{\theta}$ that minimizes $\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|^2\right]$

idea: use $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \boldsymbol{y} \implies \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} - \boldsymbol{\theta}\right\|^2\right] = N\sigma^2$?

requirement: to be a good estimator of $\boldsymbol{\theta}$, $\widehat{\boldsymbol{\theta}}$ should be s.t. $\widehat{\boldsymbol{\theta}}^T\widehat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}^T\boldsymbol{\theta}$

$$\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}^T\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\right] = \mathbb{E}\left[\boldsymbol{y}^T\boldsymbol{y}\right] = \mathbb{E}\left[\boldsymbol{\theta}^T\boldsymbol{\theta}\right] + N\sigma^2$$

Quiz time!

$$y_t = \theta_t + e_t \qquad e_t \sim \mathcal{N}\left(0, \sigma^2\right) \text{ i.i.d.} \qquad \theta_t \in \mathbb{R} \qquad \boldsymbol{y} := \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \qquad \boldsymbol{\theta} := \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_N \end{bmatrix}$$

aim: find estimator of $\boldsymbol{\theta}$ that minimizes $\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right\|^2\right]$

idea: use $\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \boldsymbol{y} \implies \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} - \boldsymbol{\theta}\right\|^2\right] = N\sigma^2$?

requirement: to be a good estimator of $\boldsymbol{\theta}$, $\widehat{\boldsymbol{\theta}}$ should be s.t. $\widehat{\boldsymbol{\theta}}^T \widehat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}^T \boldsymbol{\theta}$

$$\mathbb{E}\left[\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}^T \widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\right] = \mathbb{E}\left[\boldsymbol{y}^T \boldsymbol{y}\right] = \mathbb{E}\left[\boldsymbol{\theta}^T \boldsymbol{\theta}\right] + N\sigma^2$$

the ML solution overestimates the norm of $\boldsymbol{\theta}$!

The James-Stein estimator

$$\widehat{\boldsymbol{\theta}}_{\mathrm{JS}} := \left(1 - \frac{N-2}{\boldsymbol{y}^T\boldsymbol{y}}\sigma^2\right)\boldsymbol{y}$$

The James-Stein estimator

$$\widehat{\boldsymbol{\theta}}_{\mathrm{JS}} := \left(1 - \frac{N-2}{\boldsymbol{y}^T\boldsymbol{y}}\sigma^2\right)\boldsymbol{y}$$

Theorem 1

For $N \geq 3$ then

$$\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{JS} - \boldsymbol{\theta}\right\|^2\right] < N\sigma^2 = \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{ML} - \boldsymbol{\theta}\right\|^2\right] \qquad \forall \boldsymbol{\theta} \in \mathbb{R}^N$$

The James-Stein estimator

$$\widehat{\boldsymbol{\theta}}_{\text{JS}} := \left(1 - \frac{N-2}{\boldsymbol{y}^T \boldsymbol{y}} \sigma^2\right) \boldsymbol{y}$$

Theorem 1

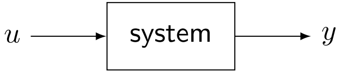
For $N \geq 3$ then

$$\mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{JS} - \boldsymbol{\theta}\right\|^2\right] < N\sigma^2 = \mathbb{E}\left[\left\|\widehat{\boldsymbol{\theta}}_{ML} - \boldsymbol{\theta}\right\|^2\right] \qquad \forall \boldsymbol{\theta} \in \mathbb{R}^N$$

Stein's phenomenon: when estimating at least 3 parameters simultaneously then $\exists$ combined estimators with lower MSE than any estimator handling the parameters separately

ill-conditioning

Some practical estimation problems

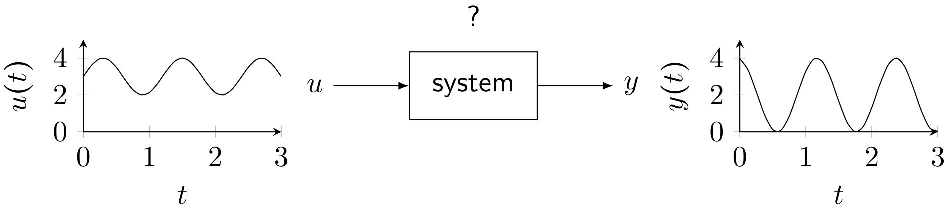


Some practical estimation problems

? $u \longrightarrow \boxed{\text{system}} \longrightarrow y$

1. inverse problems *(e.g., de-blurring)*

Some practical estimation problems



1. inverse problems *(e.g., de-blurring)*
2. direct problems *(e.g., system identification, machine learning)*

Ill-posedness and ill-conditioning

$$y_t = f\left(u_t\right) + v_t$$

Ill-posedness and ill-conditioning

$$y_t = f(u_t) + v_t$$

ill-posed problem *(in the Hadamard sense)*: solution is either not unique or does not depend continuously on the data

Ill-posedness and ill-conditioning

$$y_t = f(u_t) + v_t$$

ill-posed problem *(in the Hadamard sense)*: solution is either not unique or does not depend continuously on the data

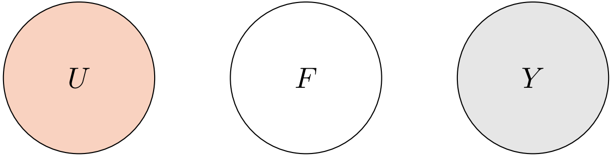
ill-conditioned problem: solution is very sensitive to the data

Ill-posedness and ill-conditioning

$$y_t = f\left(u_t\right) + v_t$$

ill-posed problem *(in the Hadamard sense)*: solution is either not unique or does not depend continuously on the data

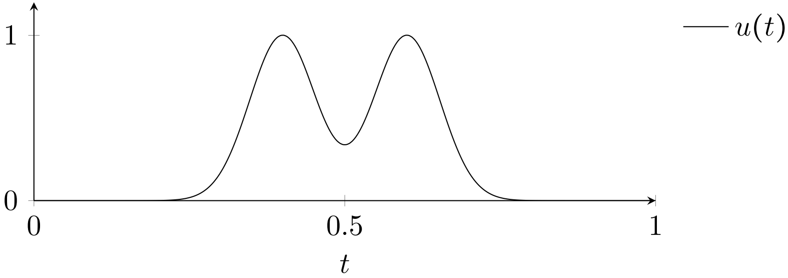
ill-conditioned problem: solution is very sensitive to the data



Example: the Hunt reconstruction problem

continuous-time system with sampled output

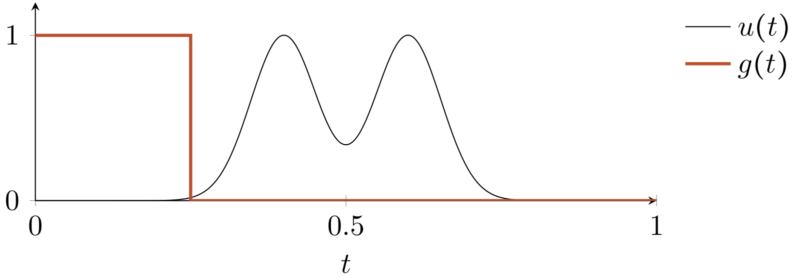
$$u(t) = \exp\left(-\left(\frac{t-0.4}{0.075}\right)^2\right) + \exp\left(-\left(\frac{t-0.6}{0.075}\right)^2\right)$$



Example: the Hunt reconstruction problem

continuous-time system with sampled output

$$u(t) = \exp\left(-\left(\frac{t-0.4}{0.075}\right)^2\right) + \exp\left(-\left(\frac{t-0.6}{0.075}\right)^2\right) \qquad g(t) = \begin{cases} 1 & \text{if } 0 \le t \le 0.25 \\ 0 & \text{otherwise,} \end{cases}$$

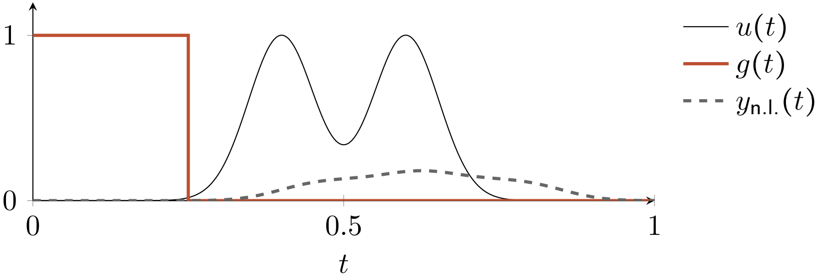


Example: the Hunt reconstruction problem

continuous-time system with sampled output

$$u(t) = \exp\left(-\left(\frac{t-0.4}{0.075}\right)^2\right) + \exp\left(-\left(\frac{t-0.6}{0.075}\right)^2\right) \qquad g(t) = \begin{cases} 1 & \text{if } 0 \leq t \leq 0.25 \\ 0 & \text{otherwise,} \end{cases}$$

$$y_{\text{n.l.}}(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau$$

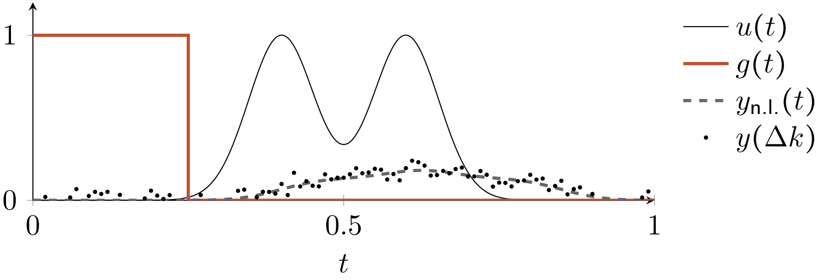


Example: the Hunt reconstruction problem

continuous-time system with sampled output

$$u(t) = \exp\left(-\left(\frac{t-0.4}{0.075}\right)^2\right) + \exp\left(-\left(\frac{t-0.6}{0.075}\right)^2\right) \qquad g(t) = \begin{cases} 1 & \text{if } 0 \le t \le 0.25 \\ 0 & \text{otherwise,} \end{cases}$$

$$y_{\text{n.l.}}(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau \qquad y(\Delta k) = y_{\text{n.l.}}(\Delta k) + v(k)$$



Example: the Hunt reconstruction problem

assumption: $u(\Delta k)$ piecewise constant $\Longrightarrow$

$$y(\Delta k) = \sum_{\tau=1}^{N} g(\Delta\tau)u(\Delta k - \Delta\tau) + v(k)$$

dataset: $\{g(\Delta k), y(\Delta k)\}_{k=1,\dots,N}$

Example: the Hunt reconstruction problem

assumption: $u(\Delta k)$ piecewise constant $\Longrightarrow$

$$y(\Delta k) = \sum_{\tau=1}^{N} g(\Delta\tau)u(\Delta k - \Delta\tau) + v(k)$$

dataset: $\{g(\Delta k), y(\Delta k)\}_{k=1,\dots,N}$

$$\begin{bmatrix} y(\Delta) \\ y(\Delta 2) \\ y(\Delta 3) \\ \vdots \end{bmatrix} = \begin{bmatrix} g(\Delta) & & & \\ g(\Delta 2) & g(\Delta) & & \\ g(\Delta 3) & g(\Delta 2) & g(\Delta) & \\ \vdots & & & \ddots \end{bmatrix} \begin{bmatrix} u(0) \\ u(\Delta) \\ u(\Delta 2) \\ \vdots \end{bmatrix}$$

Example: the Hunt reconstruction problem

assumption: $u(\Delta k)$ piecewise constant $\Longrightarrow$

$$y(\Delta k) = \sum_{\tau=1}^{N} g(\Delta\tau)u(\Delta k - \Delta\tau) + v(k)$$

dataset: $\{g(\Delta k), y(\Delta k)\}_{k=1,\dots,N}$

$$\begin{bmatrix} y(\Delta) \\ y(\Delta 2) \\ y(\Delta 3) \\ \vdots \end{bmatrix} = \begin{bmatrix} g(\Delta) & & & \\ g(\Delta 2) & g(\Delta) & & \\ g(\Delta 3) & g(\Delta 2) & g(\Delta) & \\ \vdots & & & \ddots \end{bmatrix} \begin{bmatrix} u(0) \\ u(\Delta) \\ u(\Delta 2) \\ \vdots \end{bmatrix}$$

$$\boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v}$$

Example: the Hunt reconstruction problem

assumption: $u(\Delta k)$ piecewise constant $\Longrightarrow$

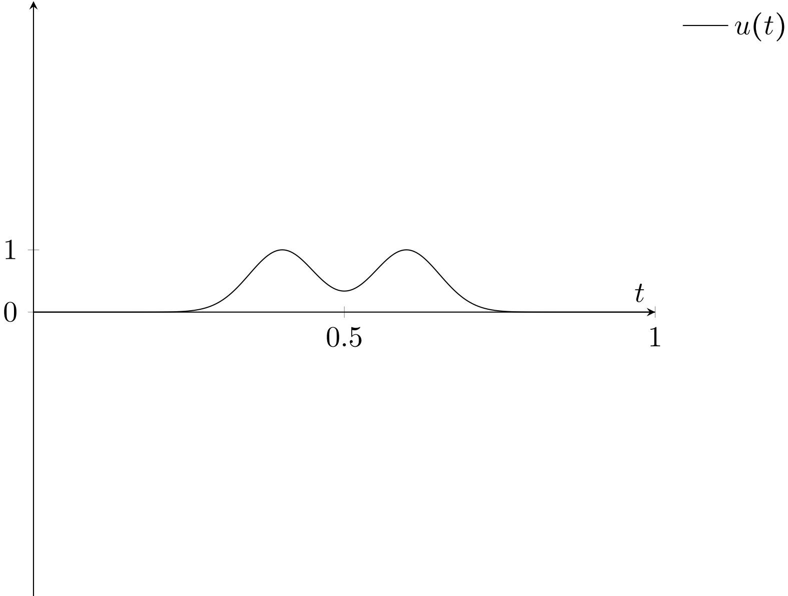
$$y(\Delta k) = \sum_{\tau=1}^{N} g(\Delta\tau)u(\Delta k - \Delta\tau) + v(k)$$

dataset: $\{g(\Delta k), y(\Delta k)\}_{k=1,\dots,N}$

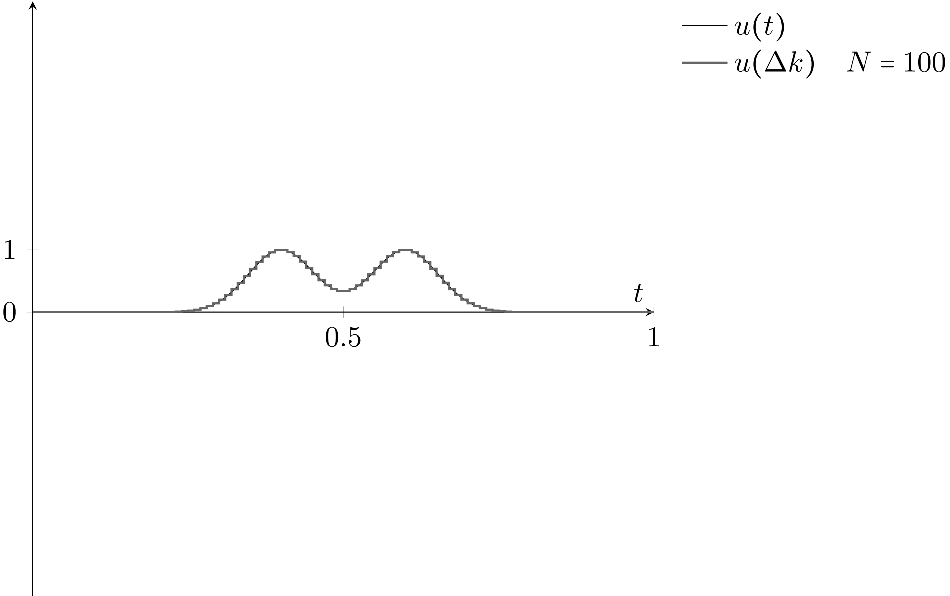
$$\begin{bmatrix} y(\Delta) \\ y(\Delta 2) \\ y(\Delta 3) \\ \vdots \end{bmatrix} = \begin{bmatrix} g(\Delta) & & & \\ g(\Delta 2) & g(\Delta) & & \\ g(\Delta 3) & g(\Delta 2) & g(\Delta) & \\ \vdots & & & \ddots \end{bmatrix} \begin{bmatrix} u(0) \\ u(\Delta) \\ u(\Delta 2) \\ \vdots \end{bmatrix}$$

$$\boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v} \qquad \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{y}$$

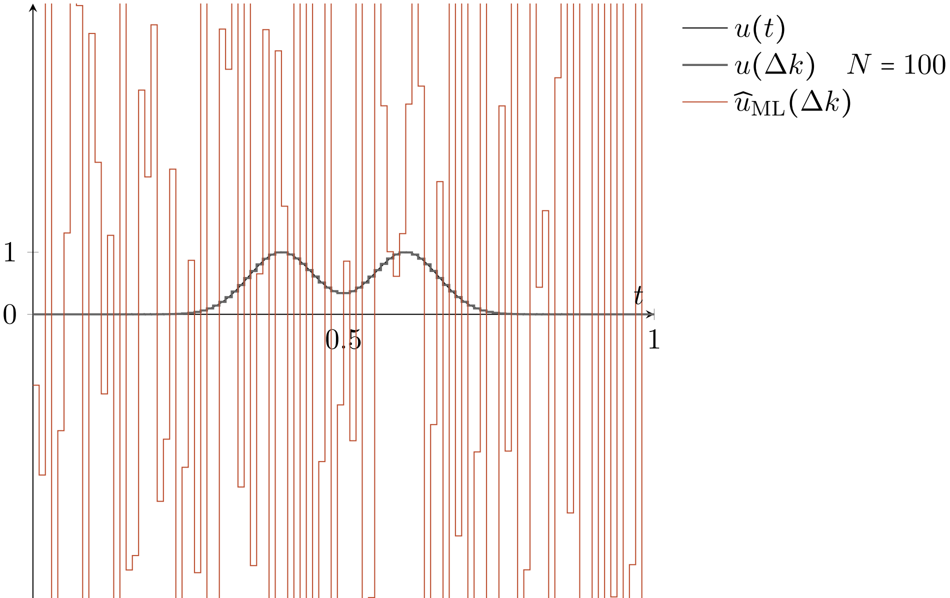
Is the Hunt reconstruction problem well defined?



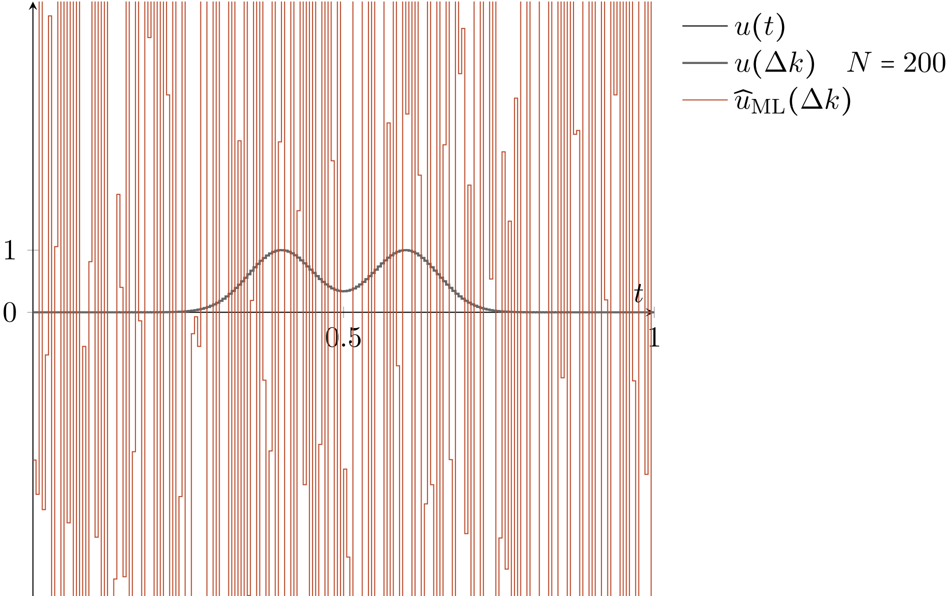
Is the Hunt reconstruction problem well defined?



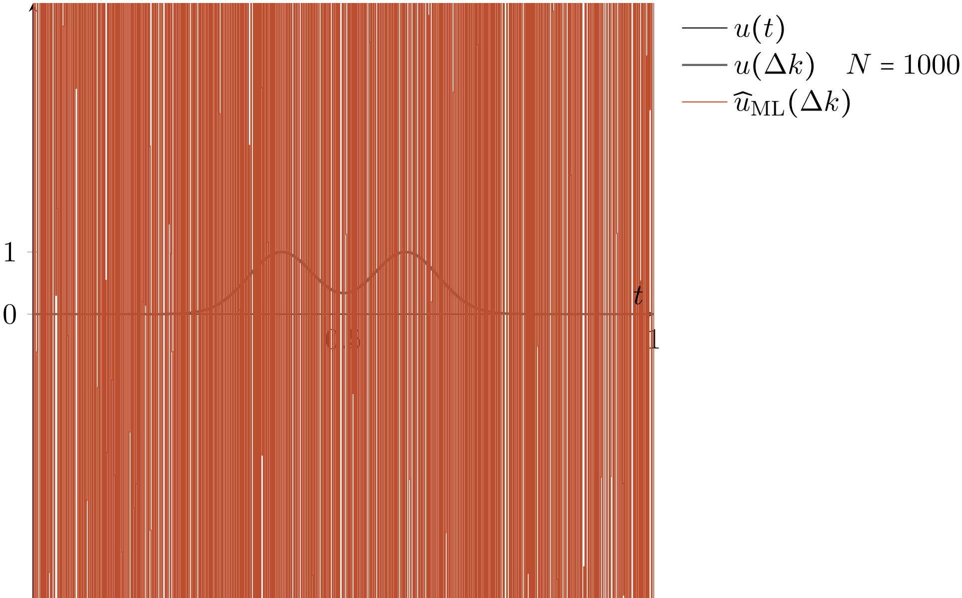
Is the Hunt reconstruction problem well defined?



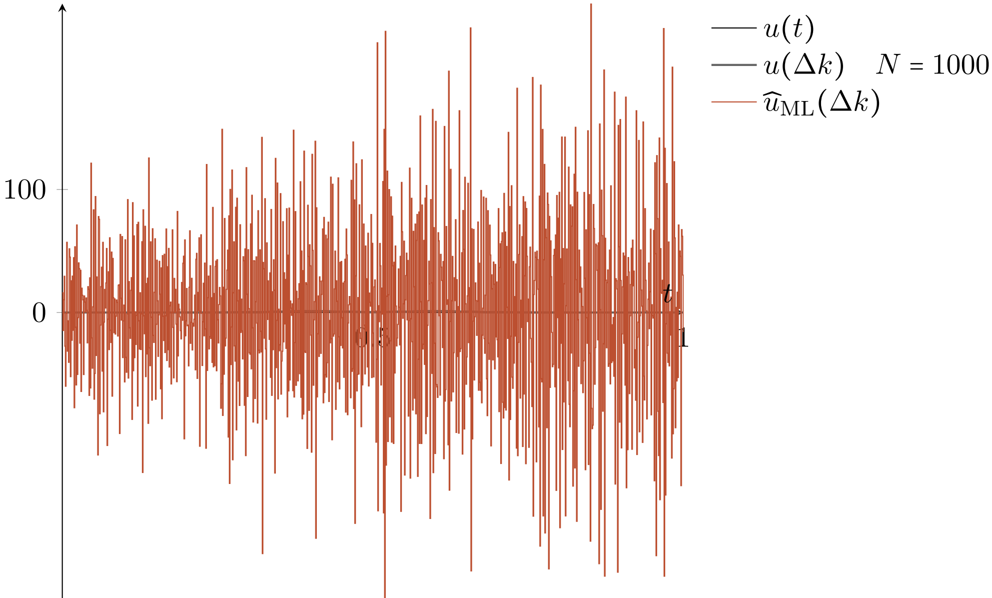
Is the Hunt reconstruction problem well defined?



Is the Hunt reconstruction problem well defined?



Is the Hunt reconstruction problem well defined?



Pitfalls of the ML estimator for the Hunt reconstruction problem

$$\begin{cases} \boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v} \\ \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{y} \end{cases} \quad \Longrightarrow \quad \boldsymbol{e} = \boldsymbol{u} - \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{v}$$

Pitfalls of the ML estimator for the Hunt reconstruction problem

$$\begin{cases} \boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v} \\ \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{y} \end{cases} \Longrightarrow \quad \boldsymbol{e} = \boldsymbol{u} - \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{v}$$

usually G *low pass, and thus usually* G^{-1} *high pass!*

Pitfalls of the ML estimator for the Hunt reconstruction problem

$$\begin{cases} \boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v} \\ \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{y} \end{cases} \implies \boldsymbol{e} = \boldsymbol{u} - \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{v}$$

usually G low pass, and thus usually G^{-1} high pass!

Analysing the problem through condition numbers = *maximum amplification of the relative error on the output measurements*:

$$\frac{\|\boldsymbol{e}\|}{\|\boldsymbol{u}\|} \leq \frac{\sigma_{\max}(G)}{\sigma_{\min}(G)} \frac{\|\boldsymbol{v}\|}{\|G\boldsymbol{u}\|}$$

Pitfalls of the ML estimator for the Hunt reconstruction problem

$$\begin{cases} \boldsymbol{y} = G\boldsymbol{u} + \boldsymbol{v} \\ \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{y} \end{cases} \quad \Longrightarrow \quad \boldsymbol{e} = \boldsymbol{u} - \widehat{\boldsymbol{u}}_{\mathrm{ML}} = G^{-1}\boldsymbol{v}$$

usually G low pass, and thus usually G^{-1} high pass!

Analysing the problem through condition numbers = *maximum amplification of the relative error on the output measurements*:

$$\frac{\|\boldsymbol{e}\|}{\|\boldsymbol{u}\|} \leq \frac{\sigma_{\max}(G)}{\sigma_{\min}(G)} \frac{\|\boldsymbol{v}\|}{\|G\boldsymbol{u}\|}$$

problems:

- the slower g the higher $\dfrac{\sigma_{\max}(G)}{\sigma_{\min}(G)}$
- the faster Δ the higher $\dfrac{\sigma_{\max}(G)}{\sigma_{\min}(G)}$

how can we improve our estimates?

Phillips-Tikhonov nonparametric regularization

The main ingredients of the nonparametric approach - in words

1. do not fix the structure of the solution a-priori
2. search for approximated solutions and not for perfect data fits
3. include information on the regularity of the estimand

The main ingredients of the nonparametric approach - in math

The main ingredients of the nonparametric approach - in math

inputs functional: *(i.e., input-output transformation)*

$$L_k[u] \qquad \text{example: } L_k[u] = \int_0^{+\infty} g(\tau)u(\Delta k - \tau)d\tau$$

The main ingredients of the nonparametric approach - in math

inputs functional: *(i.e., input-output transformation)*

$$L_k[u] \qquad \text{example: } L_k[u] = \int_0^{+\infty} g(\tau)u(\Delta k - \tau)d\tau$$

loss function: *(i.e., adherence to the experimental data)*

$$V\Big(y(\Delta k) - L_k[u]\Big) \qquad \text{example: } V = \frac{\Big(y(\Delta k) - L_k[u]\Big)^2}{\sigma_k^2}$$

The main ingredients of the nonparametric approach - in math

inputs functional: *(i.e., input-output transformation)*

$$L_k[u] \qquad \text{example: } L_k[u] = \int_0^{+\infty} g(\tau)u(\Delta k - \tau)d\tau$$

loss function: *(i.e., adherence to the experimental data)*

$$V\Big(y(\Delta k) - L_k[u]\Big) \qquad \text{example: } V = \frac{\Big(y(\Delta k) - L_k[u]\Big)^2}{\sigma_k^2}$$

regularizer: *(i.e., evaluation of the regularity of u)*

$$\|u\|_H^2 \qquad \text{example: } \int_0^T \Big(u^{(m)}(t)\Big)^2 dt$$

The main ingredients of the nonparametric approach - in math

inputs functional: *(i.e., input-output transformation)*

$$L_k[u] \qquad \text{example: } L_k[u] = \int_0^{+\infty} g(\tau)u(\Delta k - \tau)d\tau$$

loss function: *(i.e., adherence to the experimental data)*

$$V\Big(y(\Delta k) - L_k[u]\Big) \qquad \text{example: } V = \frac{\Big(y(\Delta k) - L_k[u]\Big)^2}{\sigma_k^2}$$

regularizer: *(i.e., evaluation of the regularity of u)*

$$\|u\|_H^2 \qquad \text{example: } \int_0^T \Big(u^{(m)}(t)\Big)^2 dt$$

regularization parameter: *(i.e., trade-off between loss function and regolarizer)*

$$\gamma \in \mathbb{R}_+$$

The recipe

$$\widehat{u} = \arg\min_{u \in H} \sum_{k=1}^{N} V\Big(y(\Delta k) - L_k\left[u\right]\Big) + \gamma \left\|u\right\|_H^2$$

The recipe

$$\widehat{u} = \arg\min_{u \in H} \sum_{k=1}^{N} V\Big(y(\Delta k) - L_k\left[u\right]\Big) + \gamma \left\|u\right\|_H^2$$

Example:

$$\widehat{u} = \arg\min_{u \in H} \sum_{k=1}^{N} \frac{\Big(y(\Delta k) - L_k\left[u\right]\Big)^2}{\sigma_k^2} + \gamma \int_0^T \left(u^{(m)}(t)\right)^2 dt$$

The recipe

$$\widehat{u} = \arg\min_{u \in H} \sum_{k=1}^{N} V\Big(y(\Delta k) - L_k[u]\Big) + \gamma \|u\|_H^2$$

Example:

$$\widehat{u} = \arg\min_{u \in H} \sum_{k=1}^{N} \frac{\Big(y(\Delta k) - L_k[u]\Big)^2}{\sigma_k^2} + \gamma \int_0^T \Big(u^{(m)}(t)\Big)^2 dt$$

Important results:

- for $\gamma > 0$ the solution $\exists!$
- increasing γ means increasing the bias and diminishing the variance

The recipe for some common practical cases

The recipe for some common practical cases

loss function: depends on the log-likelihood!

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \quad \Longrightarrow \quad \Big(y(\Delta k) - L_k\left[u\right]\Big)^2$

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \quad \Longrightarrow \quad \left(y(\Delta k) - L_k\left[u\right]\right)^2$
- $v_k \sim \mathcal{L}\left(0, b\right) \quad \Longrightarrow \quad \left|y(\Delta k) - L_k\left[u\right]\right|$

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \implies \left(y(\Delta k) - L_k\left[u\right]\right)^2$
- $v_k \sim \mathcal{L}\left(0, b\right) \implies \left|y(\Delta k) - L_k\left[u\right]\right|$
- $v_k \sim$ exponential family $\implies V =$ piece-wise linear quadratic

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \implies \left(y(\Delta k) - L_k\left[u\right]\right)^2$
- $v_k \sim \mathcal{L}\left(0, b\right) \implies \left|y(\Delta k) - L_k\left[u\right]\right|$
- $v_k \sim$ exponential family $\implies V =$ piece-wise linear quadratic

regularizer: corresponds to an opportune prior!

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \implies \left(y(\Delta k) - L_k\left[u\right]\right)^2$
- $v_k \sim \mathcal{L}\left(0, b\right) \implies \left|y(\Delta k) - L_k\left[u\right]\right|$
- $v_k \sim$ exponential family $\implies V =$ piece-wise linear quadratic

regularizer: corresponds to an opportune prior!

- splines $\implies$ Sobolev spaces

The recipe for some common practical cases

loss function: depends on the log-likelihood!

- $v_k \sim \mathcal{N}\left(0, \sigma^2\right) \implies \left(y(\Delta k) - L_k\left[u\right]\right)^2$
- $v_k \sim \mathcal{L}\left(0, b\right) \implies \left|y(\Delta k) - L_k\left[u\right]\right|$
- $v_k \sim$ exponential family $\implies V =$ piece-wise linear quadratic

regularizer: corresponds to an opportune prior!

- splines $\implies$ Sobolev spaces
- other RKHSs *(e.g., stable-splines Kernels)*

Reconstructing the Hunt input using a Tikhonov regularization approach

loss function = quadratic:

$$\|\boldsymbol{y} - G\boldsymbol{u}\|^2$$

regularizer = energy of 1-st discrete derivative:

$$\boldsymbol{u}^T F^T F \boldsymbol{u} \qquad F := \begin{bmatrix} 1 & 0 & & & \\ -1 & 1 & 0 & & \\ 0 & -1 & 1 & 0 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

Reconstructing the Hunt input using a Tikhonov regularization approach

loss function $=$ quadratic:

$$\|\boldsymbol{y} - G\boldsymbol{u}\|^2$$

regularizer $=$ energy of 1-st discrete derivative:

$$\boldsymbol{u}^T F^T F \boldsymbol{u} \qquad\qquad F := \begin{bmatrix} 1 & 0 & & & \\ -1 & 1 & 0 & & \\ 0 & -1 & 1 & 0 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

(energy of 2-nd discrete derivative $= \boldsymbol{u}^T F^T F^T F F \boldsymbol{u}$*, and so on...)*

Reconstructing the Hunt input using a Tikhonov regularization approach

loss function $=$ quadratic:

$$\|\boldsymbol{y} - G\boldsymbol{u}\|^2$$

regularizer $=$ energy of 1-st discrete derivative:

$$\boldsymbol{u}^T F^T F \boldsymbol{u} \qquad F := \begin{bmatrix} 1 & 0 & & & \\ -1 & 1 & 0 & & \\ 0 & -1 & 1 & 0 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

(energy of 2-nd discrete derivative $= \boldsymbol{u}^T F^T F^T F F \boldsymbol{u}$*, and so on. . .)*

formulation:

$$\widehat{\boldsymbol{u}} = \arg\min_{\boldsymbol{u}\in\mathbb{R}^N} \|\boldsymbol{y} - G\boldsymbol{u}\|^2 + \gamma \boldsymbol{u}^T F^T F \boldsymbol{u}$$

Reconstructing the Hunt input using a Tikhonov regularization approach

loss function = quadratic:

$$\|\boldsymbol{y} - G\boldsymbol{u}\|^2$$

regularizer = energy of 1-st discrete derivative:

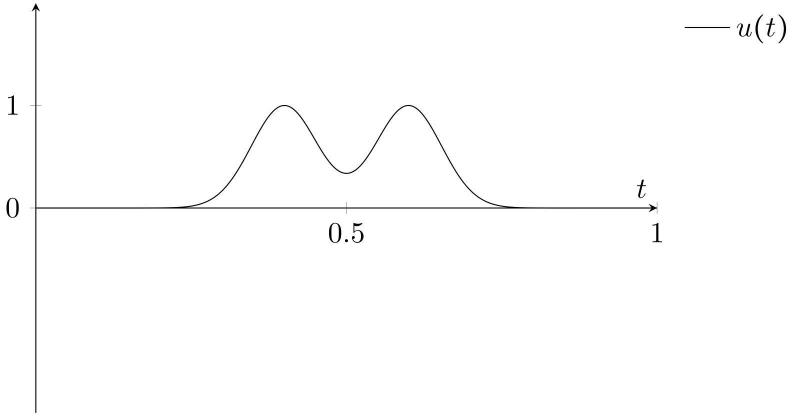
$$\boldsymbol{u}^T F^T F \boldsymbol{u} \qquad F := \begin{bmatrix} 1 & 0 & & & \\ -1 & 1 & 0 & & \\ 0 & -1 & 1 & 0 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$$

(energy of 2-nd discrete derivative = $\boldsymbol{u}^T F^T F^T F F \boldsymbol{u}$, and so on...)

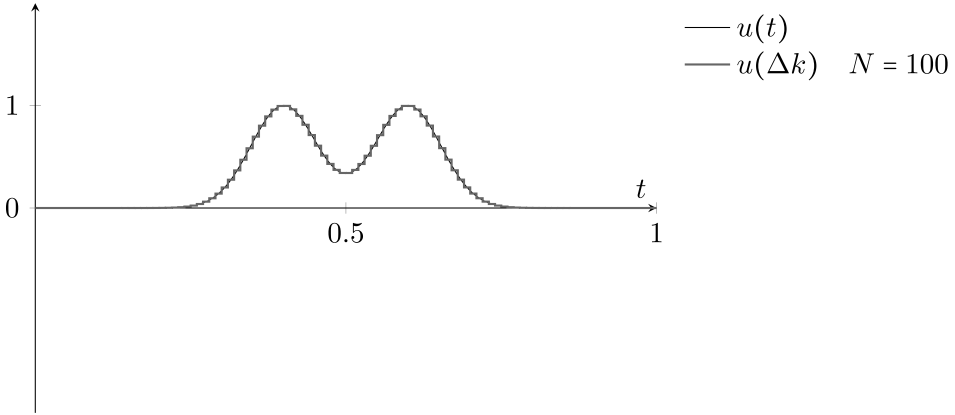
formulation:

$$\widehat{\boldsymbol{u}} = \arg\min_{\boldsymbol{u}\in\mathbb{R}^N} \|\boldsymbol{y} - G\boldsymbol{u}\|^2 + \gamma \boldsymbol{u}^T F^T F \boldsymbol{u} = \left(G^T G + \gamma F^T F\right)^{-1} G^T \boldsymbol{y}$$

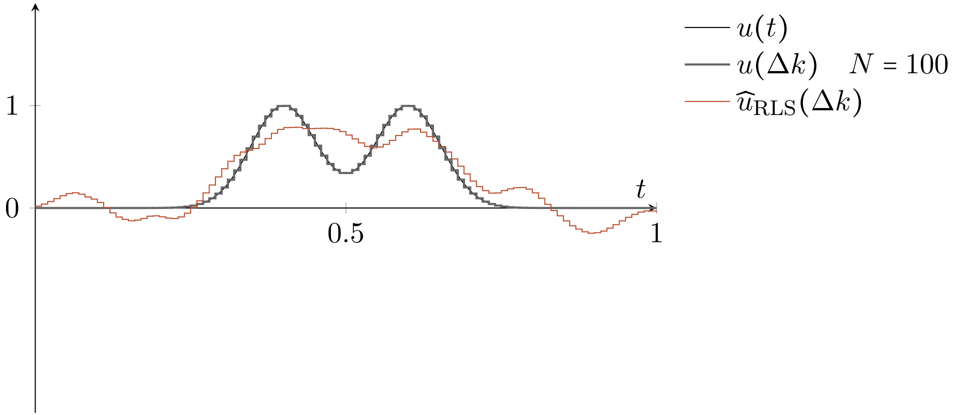
Example



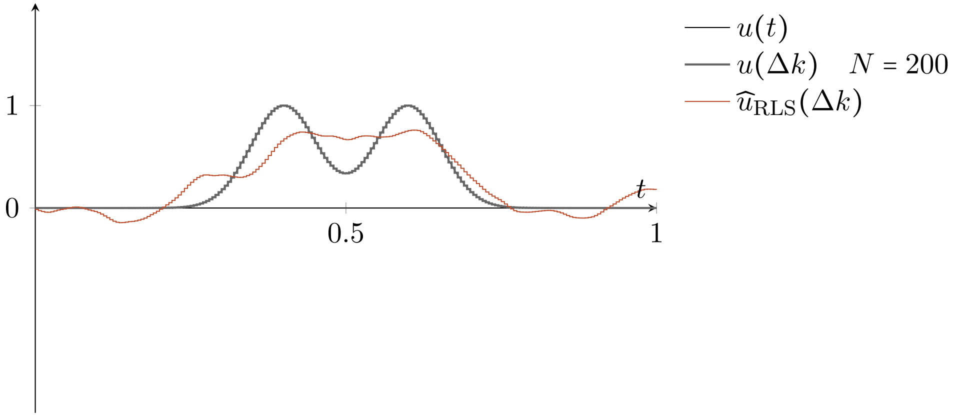
Example



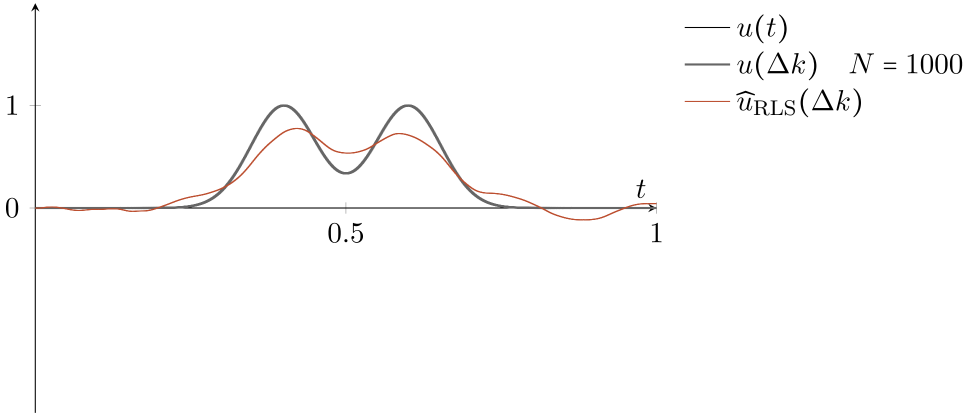
Example



Example



Example



How shall we tune γ?

$$\widehat{\boldsymbol{u}} = \left(G^T G + \gamma F^T F\right)^{-1} G^T \boldsymbol{y}$$

How shall we tune γ?

$$\widehat{\boldsymbol{u}} = \left(G^T G + \gamma F^T F\right)^{-1} G^T \boldsymbol{y}$$

- PRESS *(predicted residual error sum of squares)*
- GCV *(generalized cross-validation)*
- SURE *(Stein unbiased risk estimator)*

regularization for system identification

Direct problem $\neq$ inverse problem

$$y(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau + v(t)$$

Intuitions:

- exponentially stable system $\implies$ impulse response coefficients should decay exponentially
- impulse response is smooth $\implies$ neighboring coefficients should have a positive correlation

Direct problem $\neq$ inverse problem

$$y(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau + v(t)$$

Intuitions:

- exponentially stable system $\implies$ impulse response coefficients should decay exponentially
- impulse response is smooth $\implies$ neighboring coefficients should have a positive correlation

$\implies$ $\boldsymbol{g}^T F^T F \boldsymbol{g}$ with $F := \begin{bmatrix} 1 & 0 & & & \\ -1 & 1 & 0 & & \\ 0 & -1 & 1 & 0 & \\ & \ddots & \ddots & \ddots & \ddots \end{bmatrix}$ not the optimal regularization choice!

Direct problem $\neq$ inverse problem

$$y(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau + v(t)$$

Intuitions:

- exponentially stable system $\implies$ impulse response coefficients should decay exponentially
- impulse response is smooth $\implies$ neighboring coefficients should have a positive correlation

Direct problem $\neq$ inverse problem

$$y(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau + v(t)$$

Intuitions:

- exponentially stable system $\Longrightarrow$ impulse response coefficients should decay exponentially
- impulse response is smooth $\Longrightarrow$ neighboring coefficients should have a positive correlation

meaningful choice:* $P(\alpha) = \left[\quad \alpha^{\max i,j} \quad \right]$ α = typical exponential decay

Direct problem $\neq$ inverse problem

$$y(t) = \int_0^{+\infty} g(\tau)u(t-\tau)d\tau + v(t)$$

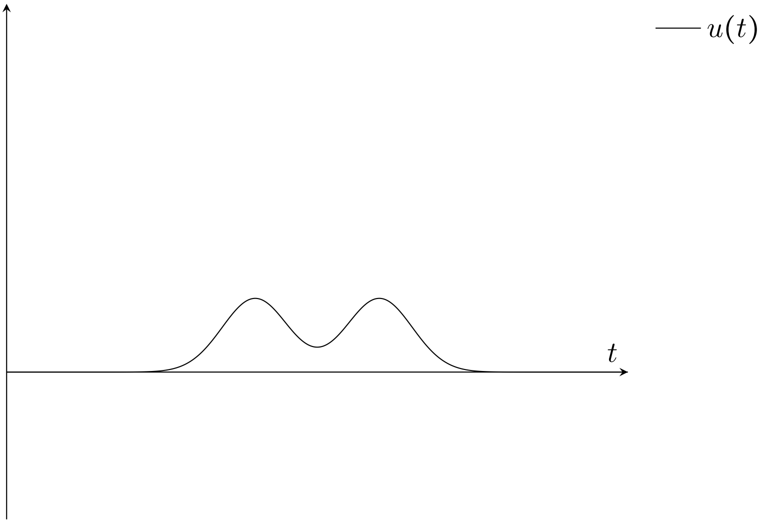
Intuitions:

- exponentially stable system $\implies$ impulse response coefficients should decay exponentially
- impulse response is smooth $\implies$ neighboring coefficients should have a positive correlation

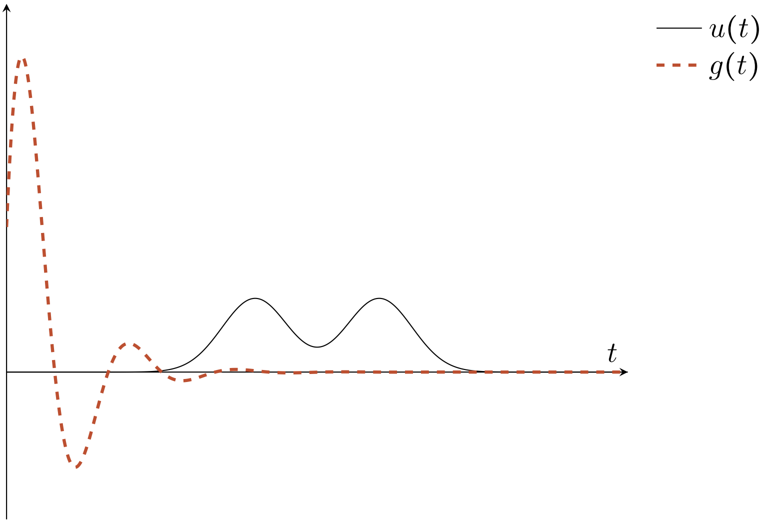
meaningful choice:* $P(\alpha) = \begin{bmatrix} & \alpha^{\max i,j} & \end{bmatrix}$ α = typical exponential decay

solution: $\widehat{\boldsymbol{g}} = \left(U^T U + \gamma P(\alpha)\right)^{-1} U^T \boldsymbol{y}$

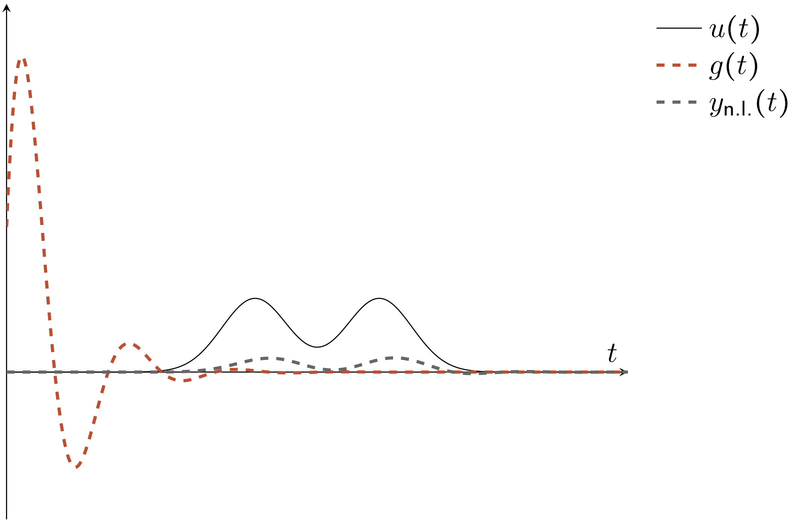
Example - system identification - definition



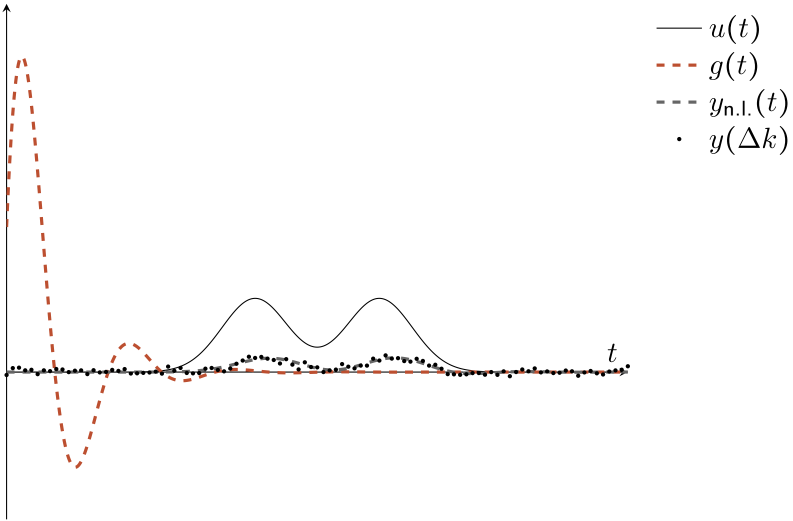
Example - system identification - definition



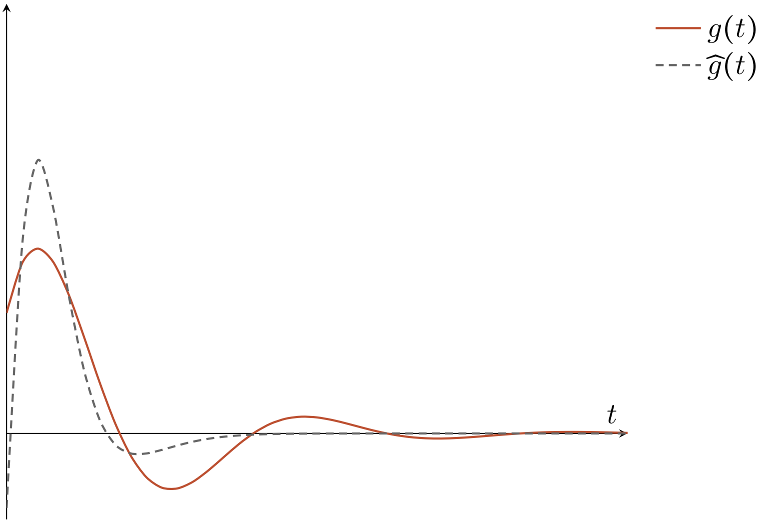
Example - system identification - definition



Example - system identification - definition



Example - system identification - solution



Summarizing. . .

- Stein $\implies$ ML is not always the best

Summarizing. . .

- Stein $\implies$ ML is not always the best
- Hunt $\implies$ ML may actually be very bad

Summarizing. . .

- Stein $\implies$ ML is not always the best
- Hunt $\implies$ ML may actually be very bad
- one potential strategy: *regularize*

Summarizing. . .

- Stein $\implies$ ML is not always the best
- Hunt $\implies$ ML may actually be very bad
- one potential strategy: *regularize*
- getting good performances requires though having a prior . . .

Summarizing. . .

- Stein $\Longrightarrow$ ML is not always the best
- Hunt $\Longrightarrow$ ML may actually be very bad
- one potential strategy: *regularize*
- getting good performances requires though having a prior . . .
- . . . but even if you don't have it you can always improve ML (cf. Stein)

Bibliography

Pillonetto, Dinuzzo, Chen, De Nicolao, Ljung

Kernel methods in system identification, machine learning and function estimation: A survey

Automatica 2014

part II: some more mathematical details

RKHS-based interpretations of regularization as a function estimation problem

Definition 1 (reproducing kernel Hilbert space)

$$\mathcal{H} \subset C^0(\mathcal{X}) = \textit{RKHS if Hilbert and if}$$
$$\forall x \in \mathcal{X} \quad \exists C_x < +\infty \textit{ s.t. } \forall f \in \mathcal{H} \quad |f(x)| \leq C_x \, \| f \|^2_{\mathcal{H}}$$

RKHS-based interpretations of regularization as a function estimation problem

Definition 1 (reproducing kernel Hilbert space)

$$\mathcal{H} \subset C^0(\mathcal{X}) = \textit{RKHS if Hilbert and if}$$
$$\forall x \in \mathcal{X} \quad \exists C_x < +\infty \textit{ s.t. } \forall f \in \mathcal{H} \quad |f(x)| \leq C_x \|f\|_{\mathcal{H}}^2$$

Practical advantage: RKHSs allow rigorous analyses

Connections between RKHSs and Mercer kernels

Definition 2 (Mercer kernel)

$K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ *that is continuous, symmetric and (semi) positive definite*

Connections between RKHSs and Mercer kernels

Definition 2 (Mercer kernel)

$K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ *that is continuous, symmetric and (semi) positive definite*

Theorem 2 (Moore-Aronszajn)

- *if* $\mathcal{H}$ *is RKHS then* $\exists!$ *Mercer* K *s.t.*
  - $K(x, \cdot) \in \mathcal{H} \ \forall x \in \mathcal{X}$
  - $\langle K(x, \cdot), f(\cdot)\rangle_{\mathcal{H}} = f(x)$ *(reproducing property)*

Connections between RKHSs and Mercer kernels

Definition 2 (Mercer kernel)

$K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ *that is continuous, symmetric and (semi) positive definite*

Theorem 2 (Moore-Aronszajn)

- *if* $\mathcal{H}$ *is RKHS then* $\exists!$ *Mercer* K *s.t.*
    - $K(x, \cdot) \in \mathcal{H} \ \forall x \in \mathcal{X}$
    - $\langle K(x, \cdot), f(\cdot) \rangle_{\mathcal{H}} = f(x)$ *(reproducing property)*
- *if* K *Mercer then* $\exists!$ $\mathcal{H}$ *RKHS*

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$
2. define the inner product of $g_1(\cdot), g_2(\cdot)$ as above as $\sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_i K(x_i, x_j)$

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$
2. define the inner product of $g_1(\cdot), g_2(\cdot)$ as above as $\sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_i K(x_i, x_j)$
3. complete $\mathcal{H}$ by adding to it all the Cauchy sequences
   (norm defined through the previous inner product)

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$
2. define the inner product of $g_1(\cdot), g_2(\cdot)$ as above as $\sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_i K(x_i, x_j)$
3. complete $\mathcal{H}$ by adding to it all the Cauchy sequences
   (norm defined through the previous inner product)

Implications

$f(\cdot) \in \mathcal{H} \implies f(\cdot)$ linear combination of a countable number of kernel sections

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$
2. define the inner product of $g_1(\cdot), g_2(\cdot)$ as above as $\sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_i K(x_i, x_j)$
3. complete $\mathcal{H}$ by adding to it all the Cauchy sequences
   (norm defined through the previous inner product)

Implications

$f(\cdot) \in \mathcal{H} \implies f(\cdot)$ linear combination of a countable number of kernel sections

$\implies$ hypothesis space $=$ countable combinations of slices of K

From K to $\mathcal{H}$

"Algorithm"

1. take all finite linear combinations $g(\cdot) = \sum_{i=1}^{p} \alpha_i K(x_i, \cdot)$
2. define the inner product of $g_1(\cdot), g_2(\cdot)$ as above as $\sum_{i=1}^{p} \sum_{j=1}^{p} \alpha_i \alpha_i K(x_i, x_j)$
3. complete $\mathcal{H}$ by adding to it all the Cauchy sequences
   (norm defined through the previous inner product)

Implications

$f(\cdot) \in \mathcal{H} \implies f(\cdot)$ linear combination of a countable number of kernel sections

$\implies$ hypothesis space $=$ countable combinations of slices of K

$\implies$ selecting K $=$ selecting properties of the final estimates

(smoothness and integrability of K reflects on smoothness and integrability of the final estimate)

Representer theorem

$$\arg\min_{f\in\mathcal{H}} \sum_{k=1}^{N} \left(y_t - f(u_t)\right)^2 + \gamma \|f\|_{\mathcal{H}}^2 = \sum_{t=1}^{N} \alpha_t K(u_t, \cdot)$$

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} = \left(\begin{bmatrix} K(u_1, u_1) & \cdots & K(u_1, u_N) \\ \vdots & & \vdots \\ K(u_N, u_1) & \cdots & K(u_N, u_N) \end{bmatrix} + \gamma I \right)^{-1} \boldsymbol{y}$$

(a.k.a. regularization network)

Representer theorem

$$\arg\min_{f\in\mathcal{H}} \sum_{k=1}^{N} \left(y_t - f(u_t)\right)^2 + \gamma \|f\|_{\mathcal{H}}^2 = \sum_{t=1}^{N} \alpha_t K(u_t, \cdot)$$

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} = \left(\begin{bmatrix} K(u_1, u_1) & \cdots & K(u_1, u_N) \\ \vdots & & \vdots \\ K(u_N, u_1) & \cdots & K(u_N, u_N) \end{bmatrix} + \gamma I \right)^{-1} \boldsymbol{y}$$

(a.k.a. regularization network)

Non-parametric approach: a priori ∞-dimensional, a posteriori N-dimensional!

Representer theorem for other types of losses

$$\arg\min_{f \in \mathcal{H}} \sum_{k=1}^{N} V\Big(y_t - L_t[f]\Big) + \gamma \, \|f\|_{\mathcal{H}}^2 = \sum_{t=1}^{N} \alpha_t K(u_t, \cdot)$$

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{bmatrix} = \text{non-trivial solutions}$$

(may require using numerical optimization tools)

Bayesian interpretations

$$f \sim \mathcal{GP}\left(0, K\right)$$

Solving ill-posed estimation problems through regularization: a brief introduction with examples

damiano.varagnolo@ltu.se

staff.www.ltu.se/~damvar

entirely written in LaTeX2ε using Beamer and Tik Z appendix