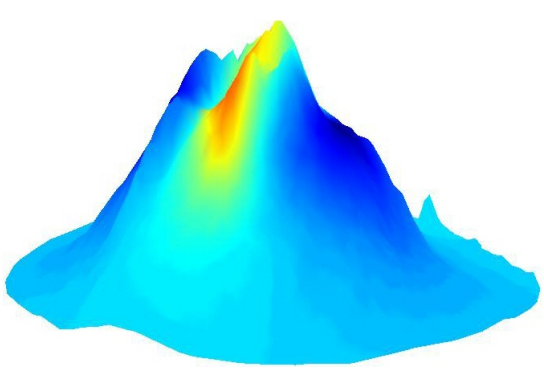
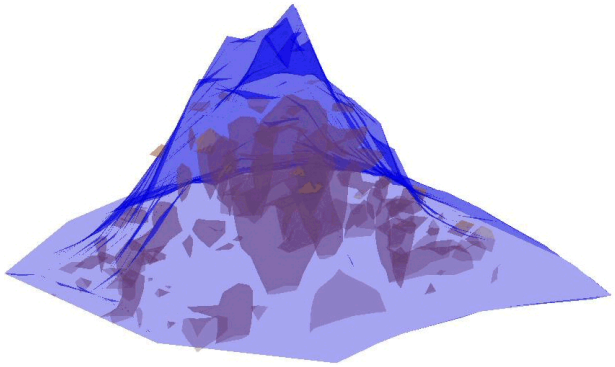
Motivation: Why Inverse Problems?

Example: from measurements of the magnetic field on the surface, we determine the activity magnetization the volcano.



Measurements
on the surface

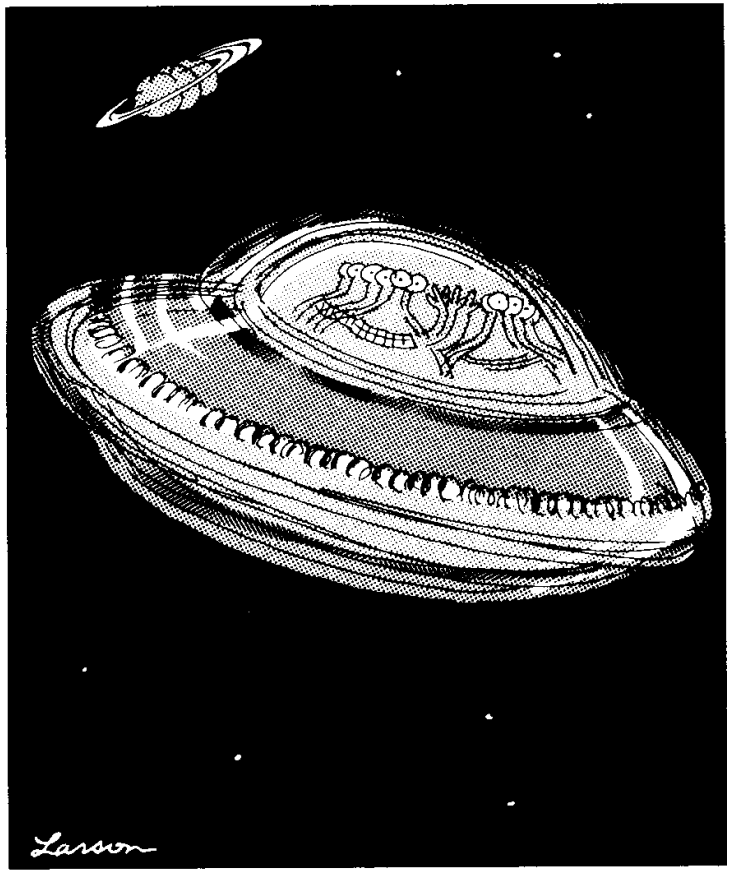
$\Rightarrow$



Reconstruction
inside the volcano

Another Example: the Hubble Space Telescope

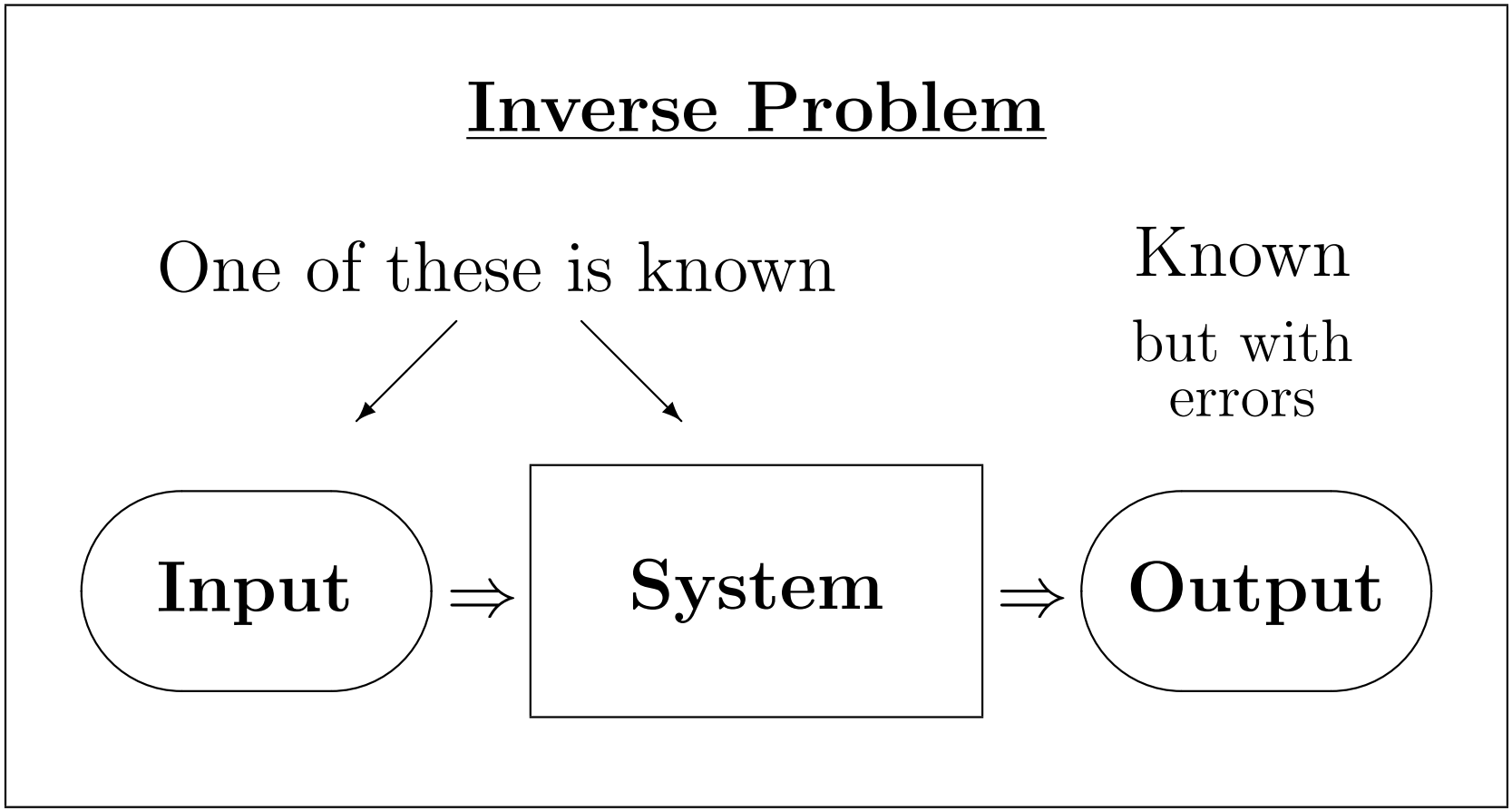
For several years, the HST produced blurred images.



Another photograph from the Hubble telescope

Inverse Problems

. . . typically arise when one wants to compute information about some "interior" properties using "exterior" measurements.



Inverse Problems: Examples

A quite generic formulation:

$$\boxed{\int_{\Omega} \text{input} \times \text{system}\, d\Omega = \text{output}}$$

Image restoration

$$\text{scenery} \rightarrow \text{lens} \rightarrow \text{image}$$

Tomography

$$\text{X-ray source} \rightarrow \text{object} \rightarrow \text{damping}$$

Seismology

$$\text{seismic wave} \rightarrow \text{layers} \rightarrow \text{reflections}$$

Discrete Ill-Posed Problems

Our generic ill-posed problem:

A Fredholm integral equation of the first kind

$$\int_0^1 K(s,t)\, f(t)\, dt = g(s)\ , \qquad 0 \le s \le 1\ .$$

Definition of a discrete ill-posed problem (DIP):

1. a square or over-determined system of linear algebraic equations
$$A\, x = b \qquad \text{or} \qquad \min_x \|A\, x - b\|_2$$
2. whose coefficient matrix A has a huge condition number, and
3. comes from the discretization of an inverse/ill-posed problem.

The Role of the Condition Number

Discrete ill-posed problems are characterized by having coefficient matrices with a *very large condition number*.

The naive solution is very sensitive to any perturbation of the right-hand side, representing the errors in the data.

Specifically, assume that the exact and perturbed solutions x^{exact} and x satisfy

$$A\,x^{\text{exact}} = b^{\text{exact}}, \qquad A\,x = b = b^{\text{exact}} + e,$$

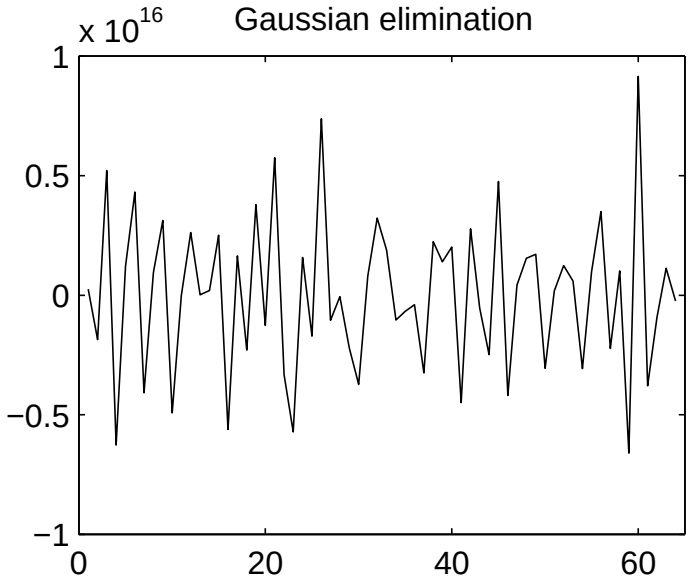
where e denotes the perturbation. Then classical perturbation theory leads to the bound

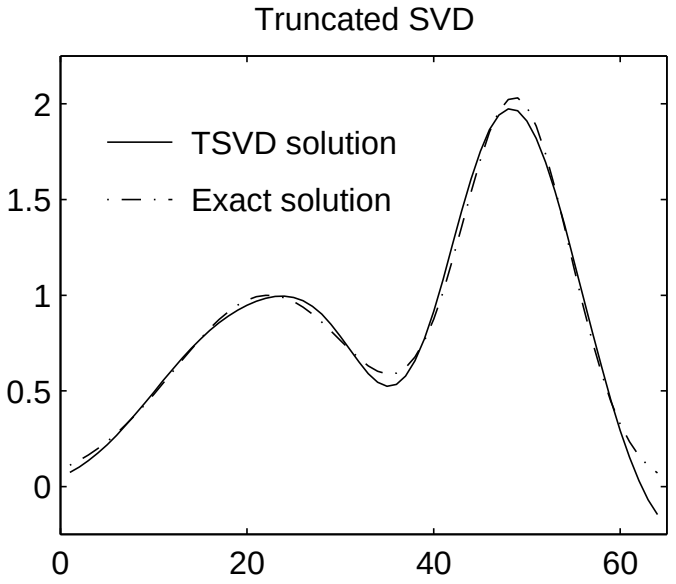
$$\frac{\|x^{\text{exact}} - x\|_2}{\|x^{\text{exact}}\|_2} \leq \text{cond}(A)\,\frac{\|e\|_2}{\|b^{\text{exact}}\|_2}.$$

Since $\text{cond}(A) = \sigma_1/\sigma_n$ is large, this implies that x can be very far from x^{exact}.

Computational Issues

The plots below show solutions x to the 64×64 DIP $A\,x = b$.





- Standard numerical methods (`x = A\b`) produce useless results.
- Specialized methods (this course) produce "reasonable" results.

The Mechanisms of Ill-Conditioned Problems

Consider a linear system with coefficient matrix and right-hand side

$$A = \begin{pmatrix} 0.16 & 0.10 \\ 0.17 & 0.11 \\ 2.02 & 1.29 \end{pmatrix}, \qquad b = \begin{pmatrix} 0.27 \\ 0.25 \\ 3.33 \end{pmatrix} = A \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0.01 \\ -0.03 \\ 0.02 \end{pmatrix}.$$

There is no vector x such that $A\,x = b$.

The least squares solution, which solves the problem

$$\min_x \|A\,x - b\|_2,$$

is given by

$$x_{\mathrm{LS}} = \begin{pmatrix} 7.01 \\ -8.40 \end{pmatrix} \qquad \Rightarrow \qquad \|A\,x_{\mathrm{LS}} - b\|_2 = 0.022\ .$$

Far from exact solution $(\,1\,,\,1\,)^T$ yet the residual is small.

Other Solutions with Small Residual

Two other "solutions" with a small residual are

$$x' = \begin{pmatrix} 1.65 \\ 0 \end{pmatrix} \qquad \Rightarrow \qquad \|A\,x' - b\|_2 = 0.031$$

$$x'' = \begin{pmatrix} 0 \\ 2.58 \end{pmatrix} \qquad \Rightarrow \qquad \|A\,x'' - b\|_2 = 0.036\ .$$

All the "solutions" x_{LS}, x' and x'' have small residuals, yet they are far from the exact solution!

- The matrix A is ill conditioned.
- Small perturbations of the data (here: b) can lead to large perturbations of the solution.
- A small residual does not imply a good solution.

(All this is well known stuff from matrix computations.)

Stabilization!

It turns out that we can modify the problem such that the solution is more stable, i.e., less sensitive to perturbations.

Example: enforce an upper bound on the solution norm $\|x\|_2$:

$$\min_x \|A\,x - b\|_2 \qquad \text{subject to} \qquad \|x\|_2 \leq \delta \ .$$

The solution x_δ depends in a nonlinear way on δ:

$$x_{0.1} = \begin{pmatrix} 0.08 \\ 0.05 \end{pmatrix} \ , \qquad x_1 = \begin{pmatrix} 0.84 \\ 0.54 \end{pmatrix}$$

$$x_{1.385} = \begin{pmatrix} 1.17 \\ 0.74 \end{pmatrix} \ , \qquad x_{10} = \begin{pmatrix} 6.51 \\ -7.60 \end{pmatrix} \ .$$

By supplying the correct additional information we can compute a good approximate solution.

Inverse Problems → Ill-Conditioned Problems

Whenever we solve an inverse problem on a computer, we face difficulties because the computational problems are ill conditioned.

The purpose of my lectures are:

1. To explain why ill-conditioned computations always arise when solving inverse problems.
2. To explain the fundamental "mechanisms" underlying the ill conditioning.
3. To explain how we can modify the problem in order to stabilize the solution.
4. To show how this can be done efficiently on a computer.

Regularization methods is at the heart of all this.

Inverse Problems are Ill-Posed Problems

Hadamard's definition of a *well-posed problem* (early 20th century):

1. the problem must have a solution,
2. the solution must be unique, and
3. it must depend continuously on data and parameters.

If the problem violates any of these requirements, it is *ill posed*.

Condition 1 can be fixed by reformulating/redefining the solution.

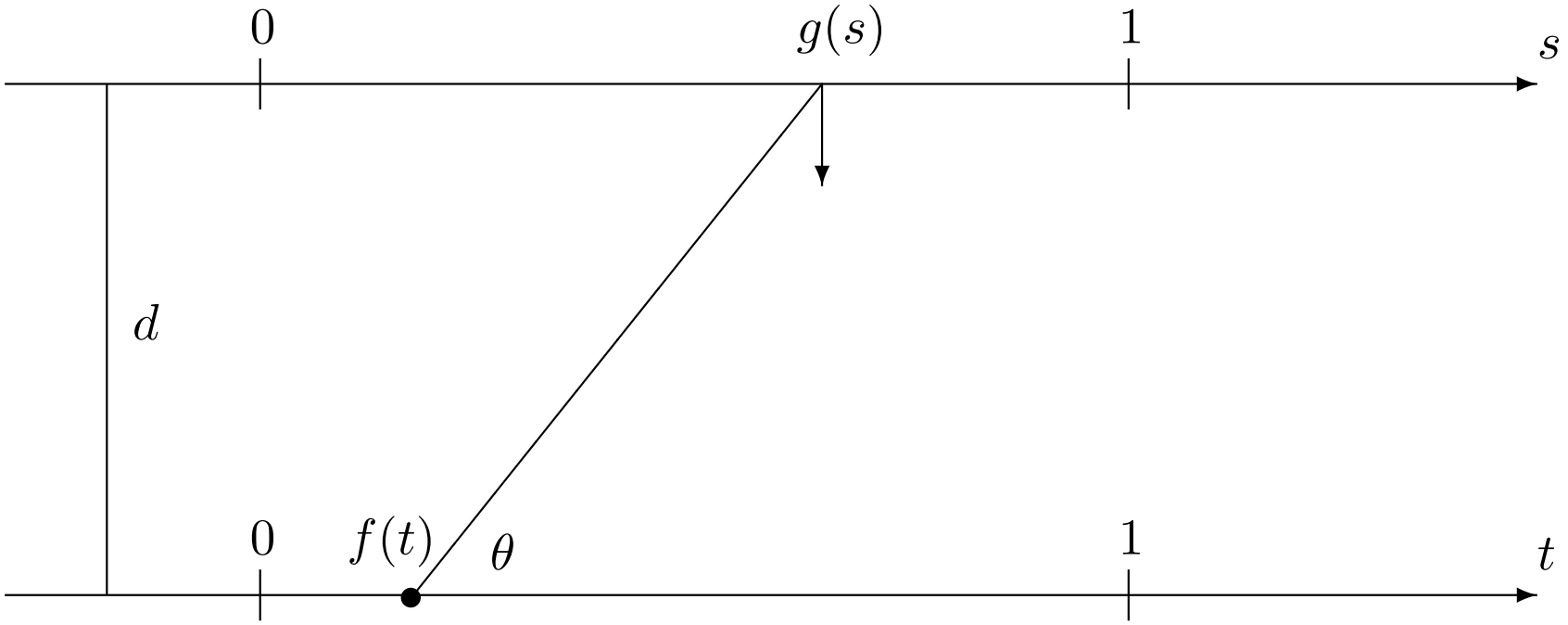
Condition 2 can be "fixed" by additional requirements to the solution, e.g., that of minimum norm.

Condition 3 is harder to "fix" because it implies that

- arbitrarily small perturbations of data and parameters can produce arbitrarily large perturbations of the solution.

Model Problem: Gravity Surveying

- Unknown mass density distribution $f(t)$ at depth d below surface, from 0 to 1 on t axis.
- Measurements of vertical component of gravitational field $g(s)$ at surface, from 0 to 1 on the s axis.



Setting Up the Integral Equation

The value of $g(s)$ due to the part dt on the t axis

$$dg = \frac{\sin\theta}{r^2}\, f(t)\, dt \ ,$$

where $r = \sqrt{d^2 + (s-t)^2}$. Using that $\sin\theta = d/r$, we get

$$\frac{\sin\theta}{r^2}\, f(t)\, dt = \frac{d}{(d^2 + (s-t)^2)^{3/2}}\, f(t)\, dt \ .$$

The total value of $g(s)$ for $0 \leq s \leq 1$ is therefore

$$g(s) = \int_0^1 \frac{d}{(d^2 + (s-t)^2)^{3/2}}\, f(t)\, dt \ .$$

This is the forward problem.

Our Integral Equation

Fredholm integral equation of the first kind:

$$\int_0^1 \frac{d}{(d^2 + (s-t)^2)^{3/2}} \, f(t) \, dt = g(s) \ , \qquad 0 \le s \le 1 \ .$$

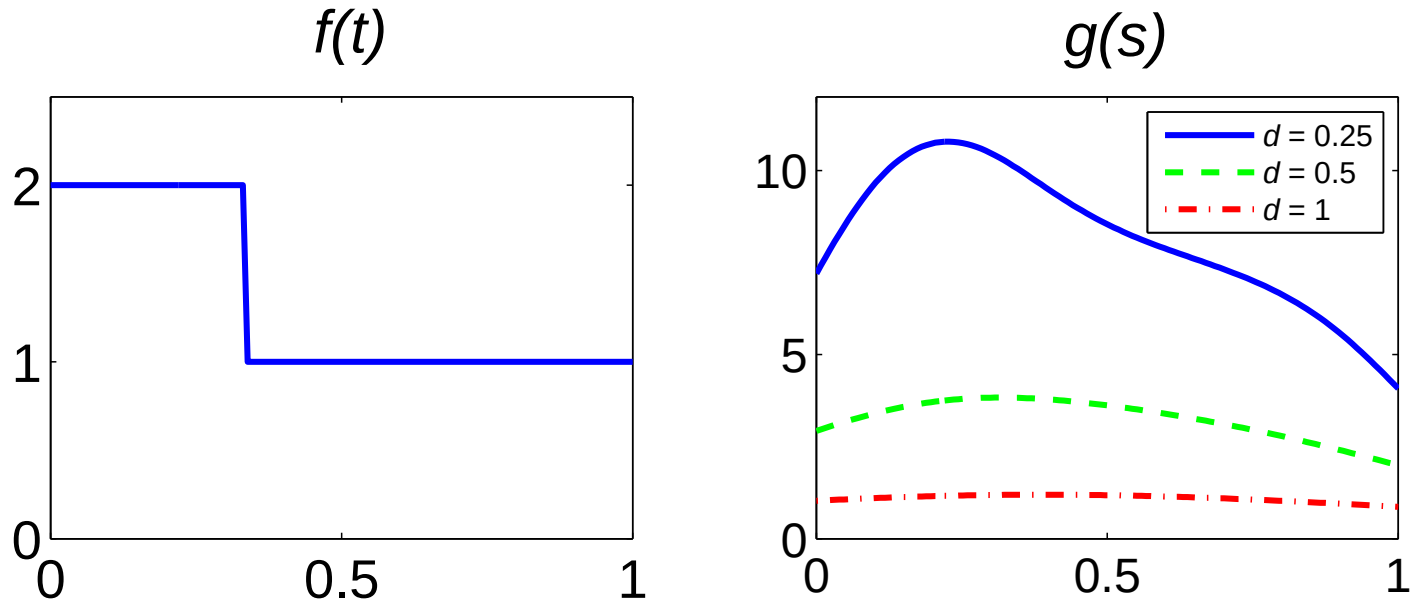
The kernel K, which represents the model, is

$$K(s,t) = h(s-t) = \frac{d}{(d^2 + (s-t)^2)^{3/2}} \ ,$$

and the right-hand side g is what we are able to measure.

From K and g we want to compute f, i.e., an inverse problem.

Numerical Examples



Observations:

- The signal/"data" $g(s)$ is a smoothed version of the source $f(t)$.
- The deeper the source, the weaker the signal.
- The discontinuity in $f(t)$ is not visible in $g(s)$.

Fredholm Integral Equations of the First Kind

Our generic inverse problem:

$$\int_0^1 K(s,t)\, f(t)\, dt = g(s), \qquad 0 \le s \le 1 \ .$$

Here, the *kernel* $K(s,t)$ and the *right-hand side* $g(s)$ are known functions, while $f(t)$ is the unknown function.

In multiple dimensions, this equation takes the form

$$\int_{\Omega_{\mathbf{t}}} K(\mathbf{s},\mathbf{t})\, f(\mathbf{t})\, d\mathbf{t} = g(\mathbf{s}), \qquad \mathbf{s} \in \Omega_{\mathbf{s}} \ .$$

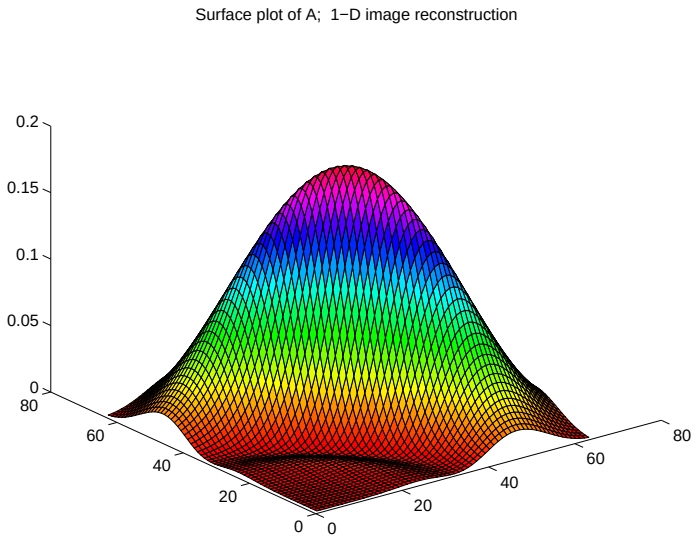
An important special case: $K(s,t) = h(s-t) \rightarrow$ *deconvolution:*

$$\int_0^1 h(s-t)\, f(t)\, dt = g(s), \qquad 0 \le s \le 1$$

(and similarly in more dimensions).

Another Example: 1-D Image Restoration

Kernel K: point spread function for an infinitely long slit of width one wavelength.
Independent variables t and s are the angles of the incoming and scattered light.
Regularization Tools: `shaw`.



$$K(s,t) = (\cos(s) + \cos(t))^2 \left(\frac{\sin(u)}{u} \right)^2$$

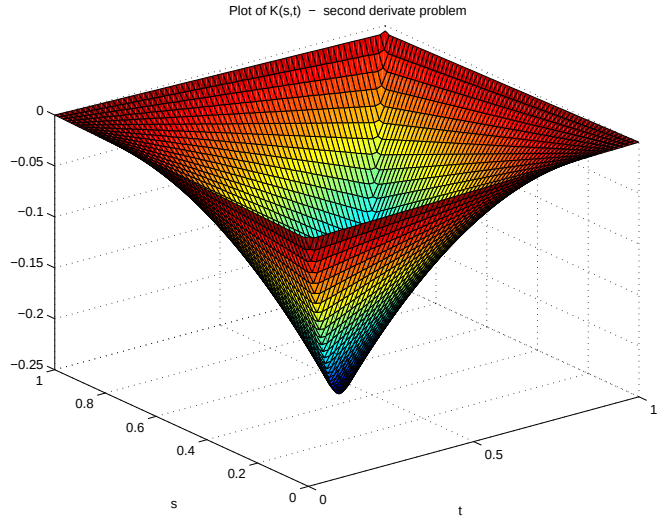
$$u = \pi \, (\sin(s) + \sin(t))$$

$$\int_{-\pi/2}^{\pi/2} K(s,t) \, f(t) \, dt = g(s) \; , \quad -\pi/2 \leq s \leq \pi/2$$

Yet Another Example: Second Derivative

Kernel K: Green's function for the second derivative

$$K(s,t) = \begin{cases} s(t-1) \;, & s < t \\ t(s-1) \;, & s \geq t \end{cases}$$



Regularization Tools: `deriv2`.

Not differentiable across the line $t = s$.

$$\int_0^1 K(s,t)\, f(t)\, dt = g(s) \;, \quad 0 \leq s \leq 1$$

Solution:

$$f(t) = g''(t) \;, \qquad 0 \leq t \leq 1 \;.$$

The Riemann-Lebesgue Lemma

Consider the function

$$f(t) = \sin(2\pi p\, t) \ , \qquad p = 1, 2, \ldots$$

then for $p \to \infty$ and "arbitrary" K we have

$$g(s) = \int_0^1 K(s,t)\, f(t)\, dt \to 0 \ .$$

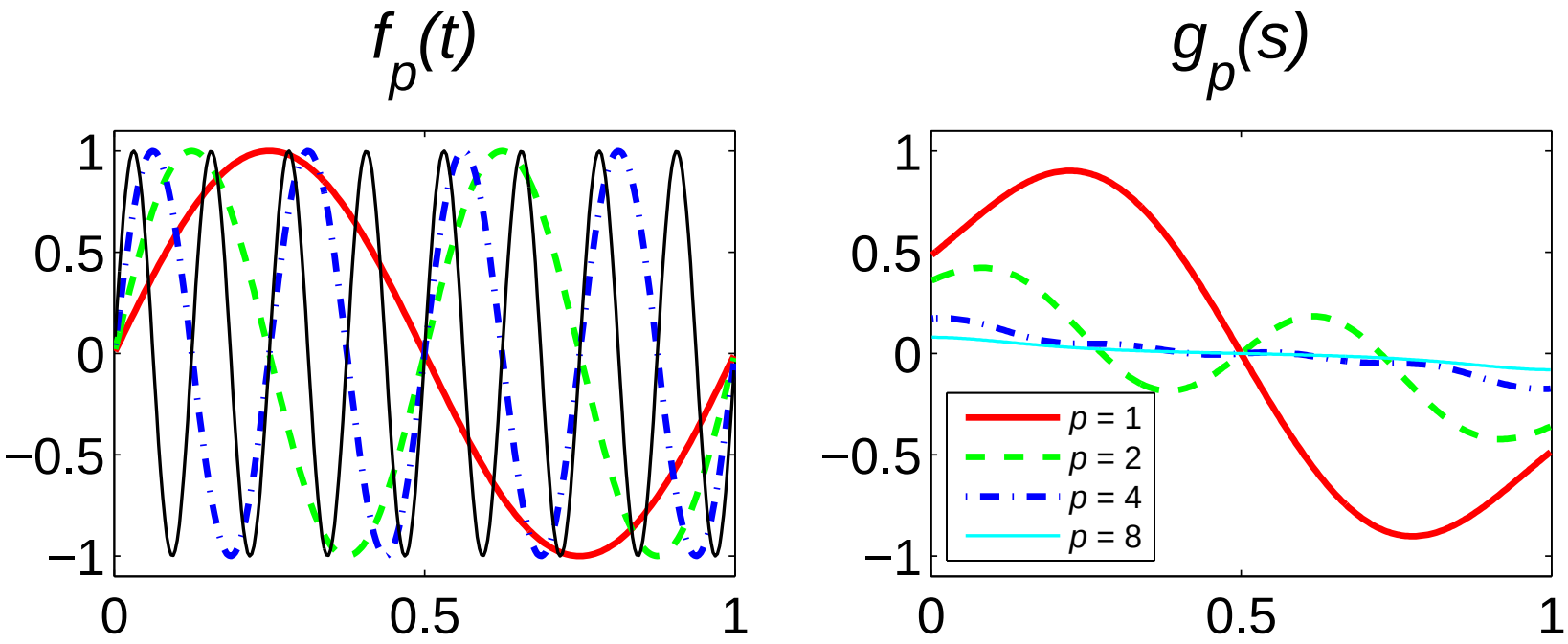
Smoothing: high frequencies are damped in the mapping $f \mapsto g$.

Hence, the mapping from g to f must amplify the high frequencies.

Therefore we can expect difficulties when trying to reconstruct f from noisy data g.

Illustration of the Riemann-Lebesgue Lemma

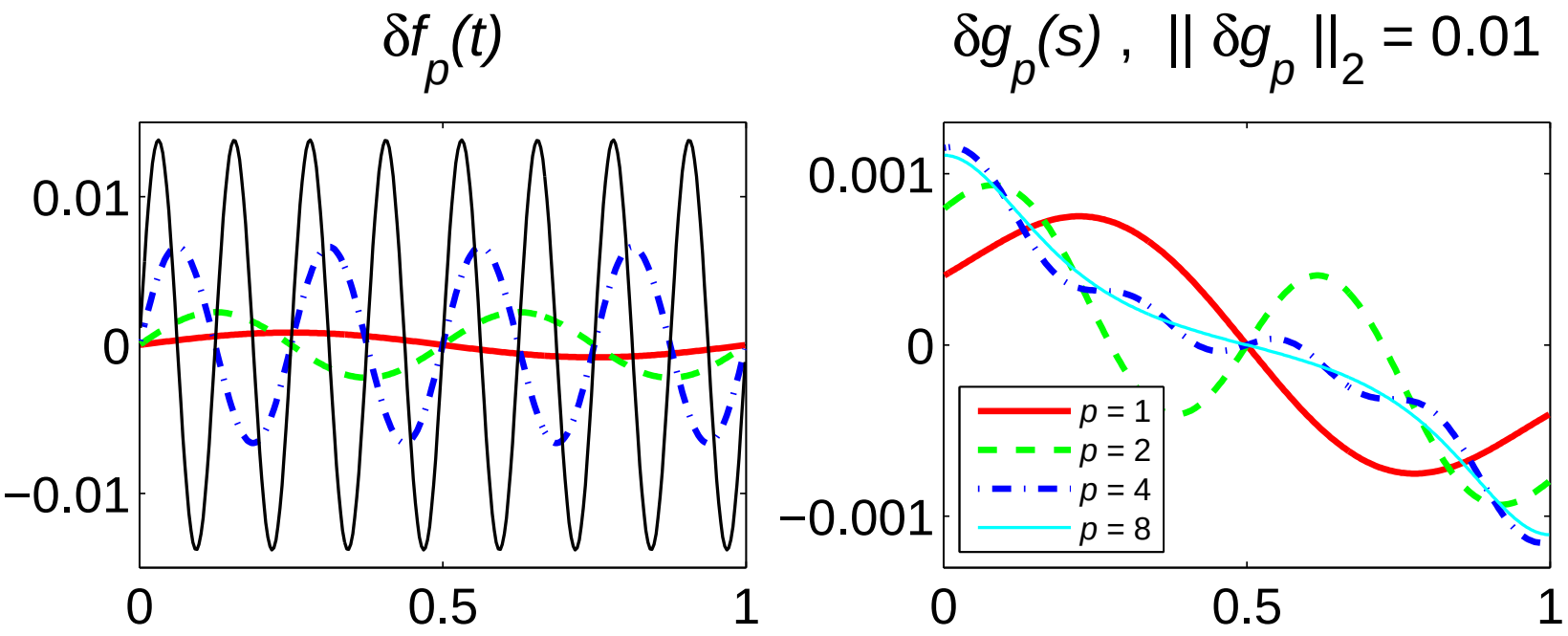
Gravity problem with $f(t) = \sin(2\pi p\, t)$, $p = 1$, 2, 4, and 8.



Higher frequencies are dampen more than low frequencies.

Difficulties with High Frequencies

In this example $\delta g(s) = \int_0^1 K(s,t)\,\delta f(t)\,dt$ and $\|\delta g\|_2 = 0.01$.



Higher frequencies are amplified more in the reconstruction process.

Why do We Care?

Why bother about these (strange) issues?

- Ill-posed problems model a variety of real applications:
  - Medical imaging (brain scanning, etc.)
  - Geophysical prospecting (search for oil, land-mines, etc.)
  - Image deblurring (astronomy, CSI[a], etc.)
  - Deconvolution of instrument's response.
- We can only hope to compute useful solutions to these problems if we fully understand their *inherent* difficulties ...
- and how these difficulties carry over to the discretized problems involved in a computer solution,
- and how to deal with them in a satisfactory way.

[a] Crime Scene Investigation.

Quadrature Discretization

Choose points s_i and t_j to obtain

$$\begin{pmatrix} w_1K(s_1,t_1) & w_2K(s_1,t_2) & \cdots & w_nK(s_1,t_n) \\ w_1K(s_2,t_1) & w_2K(s_2,t_2) & \cdots & w_nK(s_2,t_n) \\ \vdots & \vdots & & \vdots \\ w_1K(s_n,t_1) & w_2K(s_n,t_2) & \cdots & w_nK(s_n,t_n) \end{pmatrix} \begin{pmatrix} \tilde{f}_1 \\ \tilde{f}_2 \\ \vdots \\ \tilde{f}_n \end{pmatrix} = \begin{pmatrix} g(s_1) \\ g(s_2) \\ \vdots \\ g(s_n) \end{pmatrix}$$

or simply

$$\boxed{A\,x = b}$$

where A is $n \times n$ with

$$\left.\begin{aligned} a_{ij} &= w_j\, K(s_i,t_j) \\ x_j &= \tilde{f}(t_j) \\ b_i &= g(s_i) \end{aligned}\right\} \qquad i,j = 1,\ldots,n\ .$$

The midpoint rule $t_j = \frac{j-0.5}{n}$ gives $a_{ij} = n^{-1}K(s_i,t_j)$.

The Singular Value Decomposition

Assume that A is $m \times n$ and, for simplicity, also that $m \geq n$:

$$A = U\,\Sigma\,V^T = \sum_{i=1}^{n} u_i\,\sigma_i\,v_i^T .$$

Here, Σ is a diagonal matrix with the *singular values*, satisfying

$$\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)\ , \qquad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0\ .$$

The matrices U and V consist of *singular vectors*

$$U = (u_1, \ldots, u_n)\ , \qquad V = (v_1, \ldots, v_n)$$

and both matrices have orthonormal columns: $U^TU = V^TV = I_n$.

Then $\|A\|_2 = \sigma_1$, $\|A^{-1}\|_2 = \|V\,\Sigma^{-1}U^T\|_2 = \sigma_n^{-1}$, and

$$\mathrm{cond}(A) = \|A\|_2\,\|A^{-1}\|_2 = \sigma_1/\sigma_n.$$

SVD Software for Dense Matrices

Software package	Subroutine
ACM TOMS	`HYBSVD`
EISPACK	`SVD`
IMSL	`LSVRR`
LAPACK	`_GESVD`
LINPACK	`_SVDC`
NAG	`F02WEF`
Numerical Recipes	`SVDCMP`
Matlab	`svd`, `ssvd`
Reg. Tools	`csvd`

Complexity of SVD algorithms: $\mathcal{O}(m\,n^2)$.

Important SVD Relations

Relations similar to the SVE

$$A\,v_i = \sigma_i\,u_i, \qquad \|A\,v_i\|_2 = \sigma_i, \qquad i = 1, \ldots, n.$$

Also, if A is nonsingular, then

$$A^{-1}u_i = \sigma_i^{-1}\,u_i, \qquad \|A^{-1}v_i\|_2 = \sigma_i^{-1}, \qquad i = 1, \ldots, n.$$

These equations are related to the (least squares) solution:

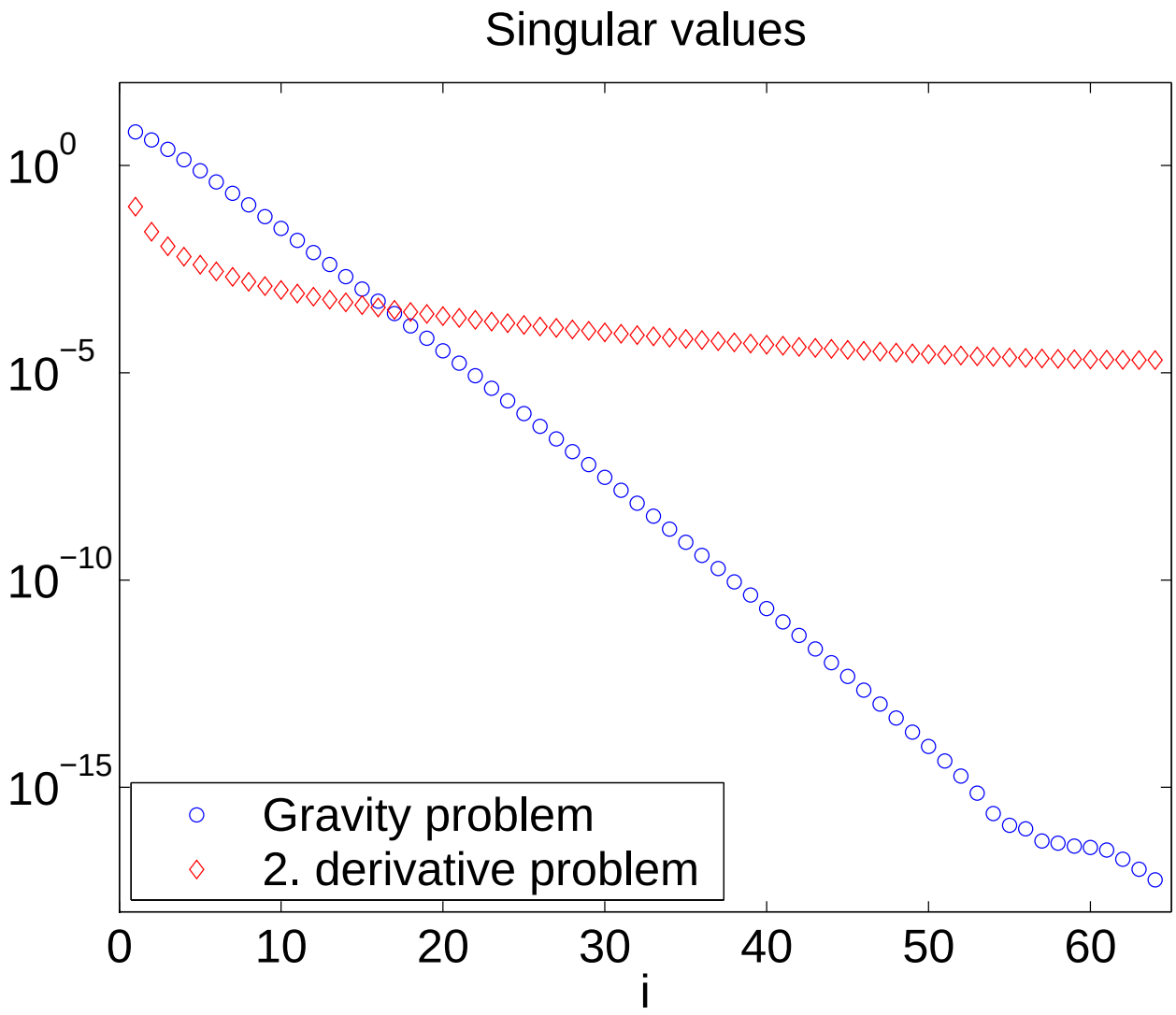
$$\begin{aligned} x &= \sum_{i=1}^{n} (v_i^T x)\,v_i \\ A\,x &= \sum_{i=1}^{n} \sigma_i\,(v_i^T x)\,u_i\ , \quad b = \sum_{i=1}^{n} (u_i^T b)\,u_i \\ A^{-1}b &= \sum_{i=1}^{n} \frac{u_i^T b}{\sigma_i}\,v_i\ . \end{aligned}$$

What the SVD Looks Like

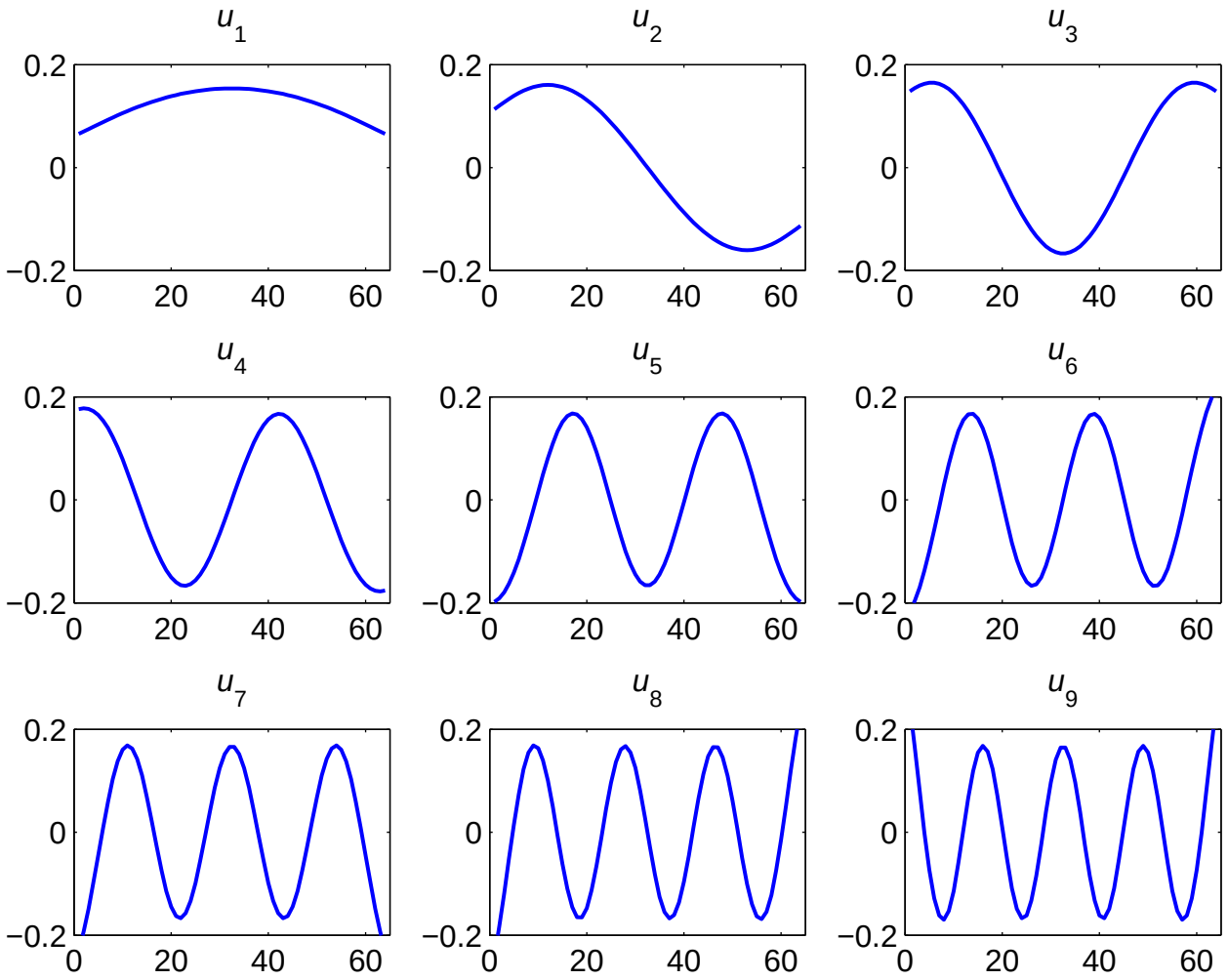
The following figures show the SVD of the 64×64 matrix A, computed by means of `csvd` from Regularization Tools:

```
>> help csvd
 CSVD Compact singular value decomposition.

  s = csvd(A)
  [U,s,V] = csvd(A)
  [U,s,V] = csvd(A,'full')

  Computes the compact form of the SVD of A:
     A = U*diag(s)*V',
  where
     U  is  m-by-min(m,n)
     s  is  min(m,n)-by-1
     V  is  n-by-min(m,n).

  If a second argument is present, the full U and V are returned.
```

The Singular Values



The Left and Right Singular Vectors

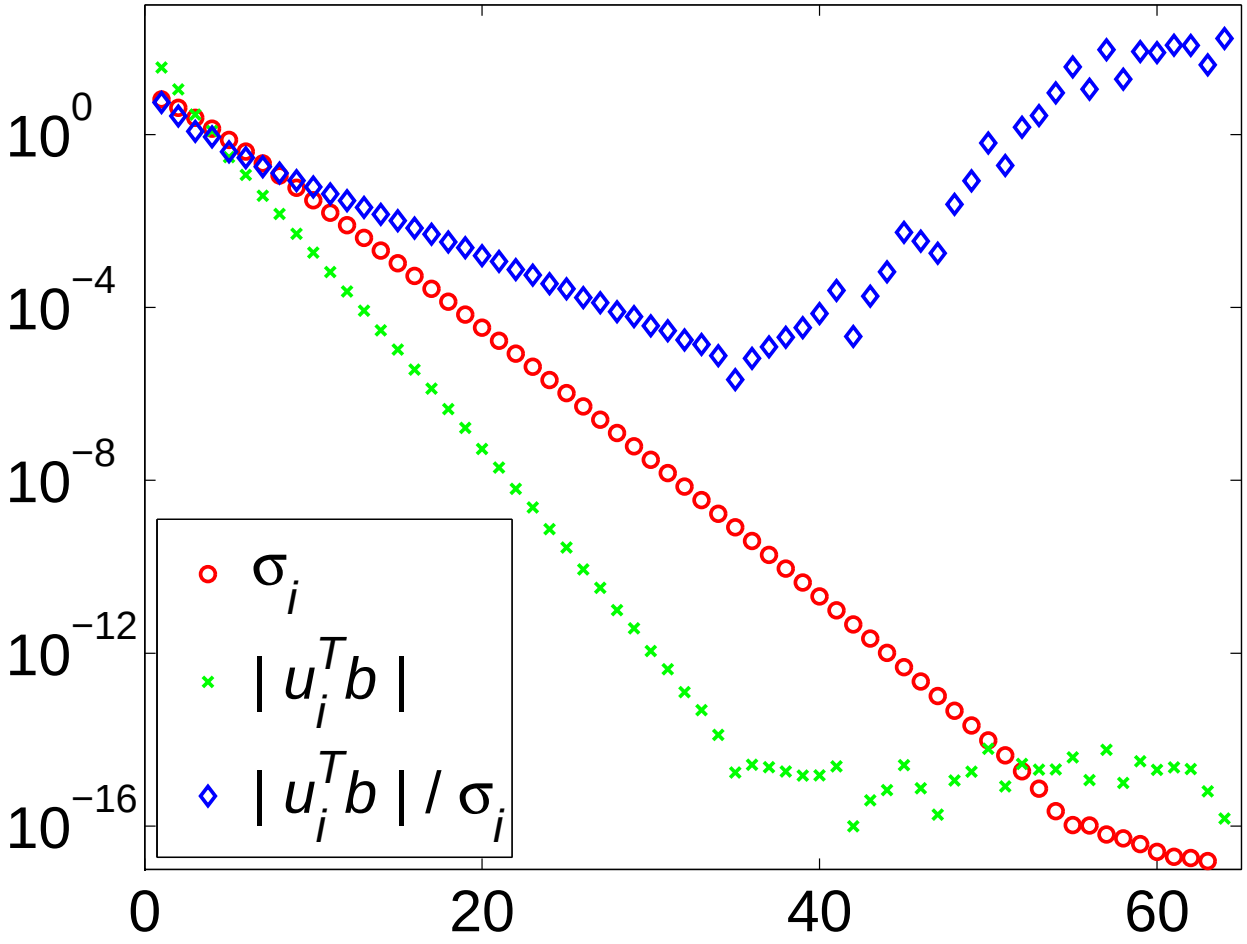


Some Observations

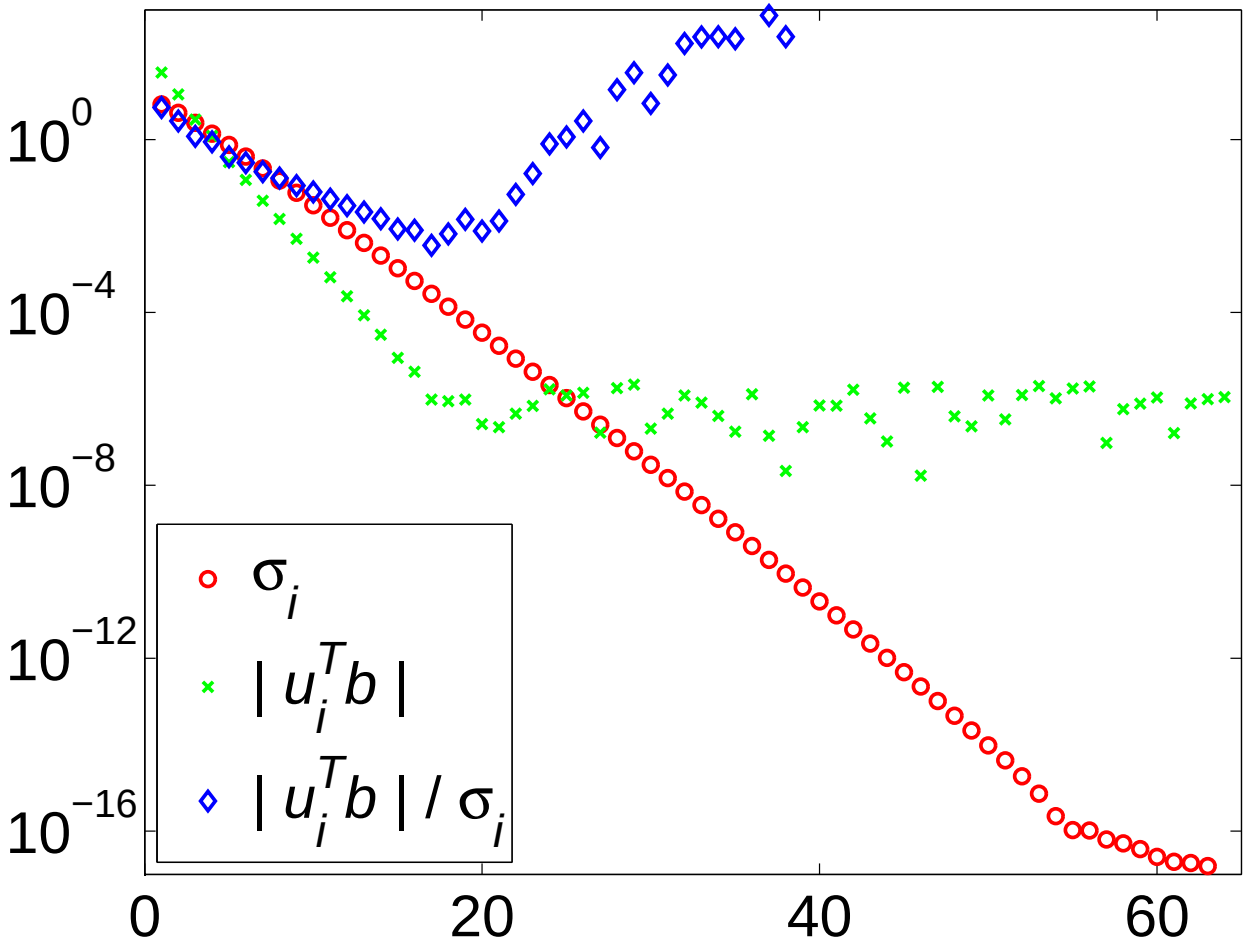
- The singular values decay gradually to zero.
- No gap in the singular value spectrum.
- Condition number $\text{cond}(A) =$ "∞."
- Singular vectors have more oscillations as i increases.
- In this problem, # sign changes $= i - 1$.

The following pages: Picard plots with increasing noise.

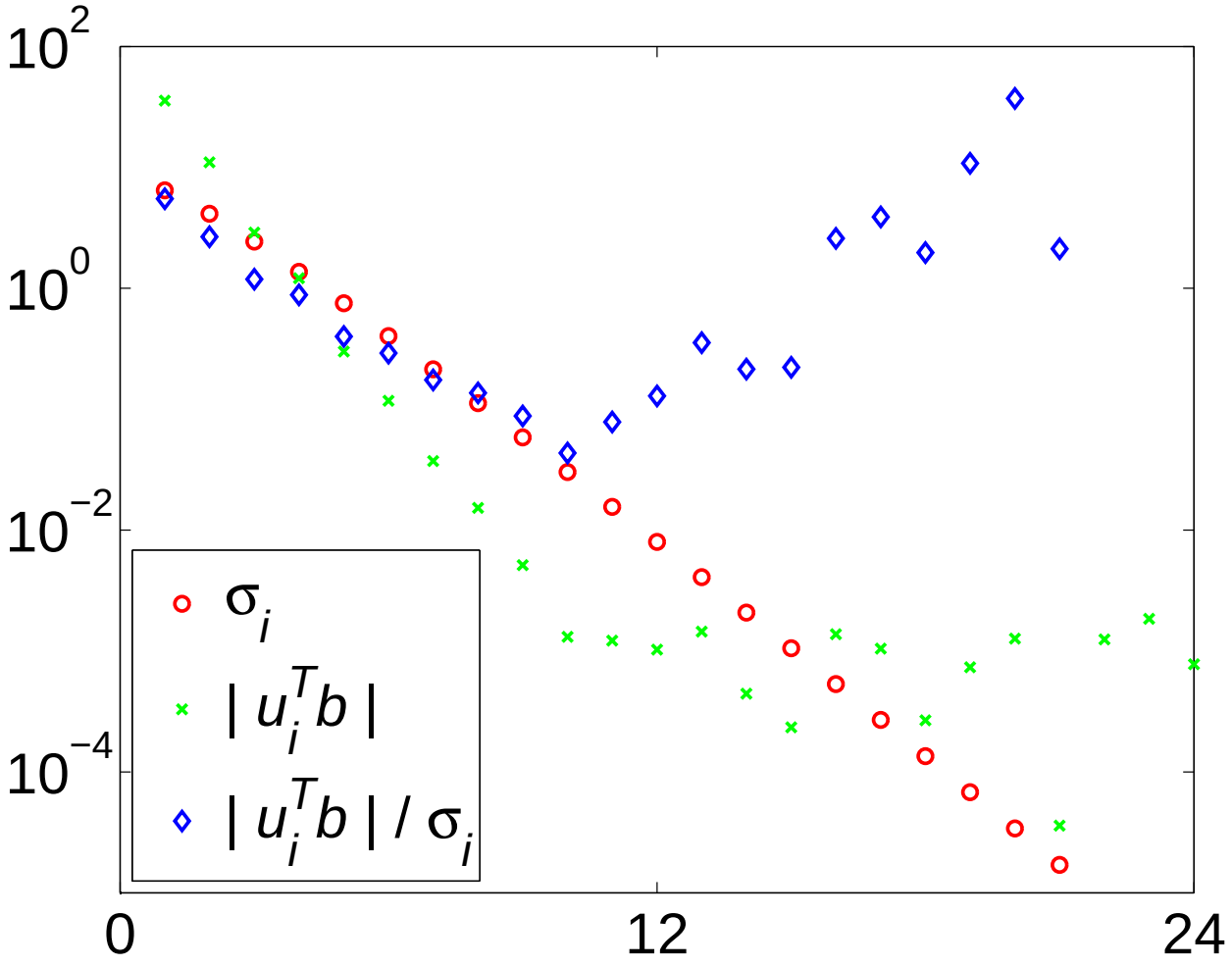
The Discrete Picard Plot



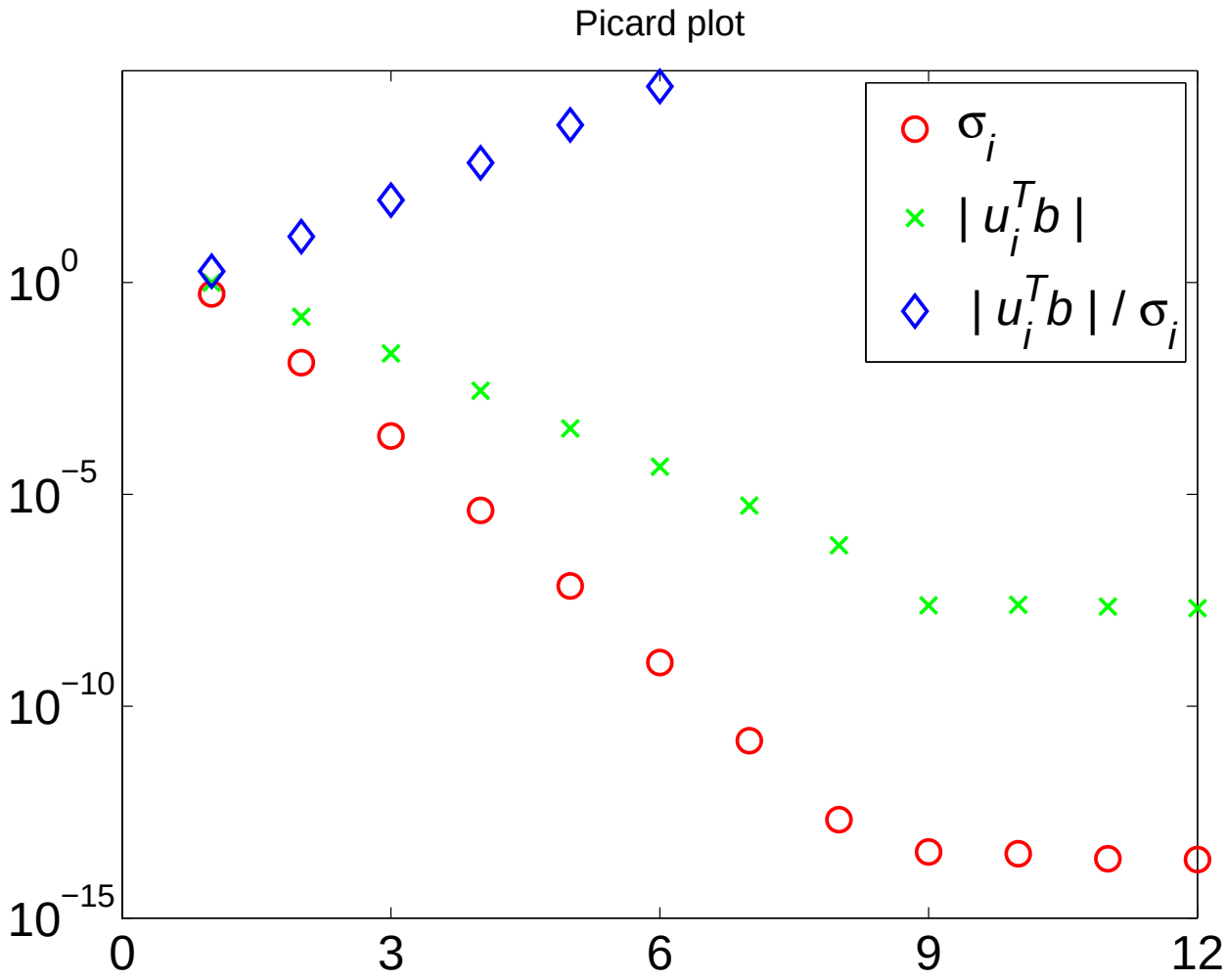
Discrete Picard Plot with Noise



Discrete Picard Plot – More Noise



The Ursell Problem



The Discrete Picard Condition

The relative decay of the singular values σ_i and the right-hand side's SVD coefficients $u_i^T b$ plays a major role!

The Discrete Picard Condition is satisfied if the coefficients $|u_i^T b^{\text{exact}}|$, on the average, *decay* to zero faster than the corresponding singular values σ_i.

Noisy Problems

Real problems have noisy data! Recall that we consider problems

$$\boxed{A\,x = b} \qquad \text{or} \qquad \boxed{\min_x \|A\,x - b\|_2}$$

with a very ill-conditioned coefficient matrix A,

$$\text{cond}(A) \gg 1.$$

Noise model:

$$b = b^{\text{exact}} + e, \qquad \text{where} \qquad b^{\text{exact}} = A\,x^{\text{exact}}\ .$$

The ingredients:

- x^{exact} is the exact (and unknown) solution,
- b^{exact} is the exact data, and
- the vector e represents the noise in the data.

Statistical Issues

Let $\mathrm{Cov}(b)$ be the covariance for the right-hand side.

Then the covariance matrix for the (least squares) solution is

$$\mathrm{Cov}(x) = A^{-1}\,\mathrm{Cov}(b)\,A^{-T}.$$

$$\mathrm{Cov}(x_{\mathrm{LS}}) = (A^T A)^{-1} A^T\,\mathrm{Cov}(b)\,A\,(A^T A)^{-1}.$$

Unless otherwise stated, we assume for simplicity that b^{exact} and e are uncorrelated, and that

$$\mathrm{Cov}(b) = \mathrm{Cov}(e) = \eta^2 I,$$

then

$$\mathrm{Cov}(x) = \mathrm{Cov}(x_{\mathrm{LS}}) = \eta^2 (A^T A)^{-1}.$$

$\mathrm{cond}(A) \gg 1 \Rightarrow$

$\mathrm{Cov}(x)$ and $\mathrm{Cov}(x_{\mathrm{LS}})$ are likely to have very large elements.

Need for Stabilization = Noise Reduction

Recall that the (least squares) solution is given by

$$x = \sum_{i=1}^{n} \frac{u_i^T b}{\sigma_i} v_i.$$

Must get rid of the "noisy" SVD components. Note that

$$u_i^T b = u_i^T b^{\text{exact}} + u_i^T e \approx \begin{cases} u_i^T b^{\text{exact}}, & |u_i^T b^{\text{exact}}| > |u_i^T e| \\ u_i^T e, & |u_i^T b^{\text{exact}}| < |u_i^T e|. \end{cases}$$

Hence, due to the DPC:

- "noisy" SVD components are those for which $|u_i^T b^{\text{exact}}|$ is small,
- and therefore they correspond to the smaller singular values σ_i.

The Story So Far

- Inverse problems are ill posed: they are very sensitive to perturbations of the data.
- Discretization $\rightarrow$ a matrix problem $A\,x = b$.
- The singular value decomposition, SVD, is a powerful tool to analyze discrete inverse problems.
- The discrete Picard condition gives information about the existence of a meaningful solution.
- The troublemakers: the large condition number $\mathrm{cond}(A)$ and the noise in the right-hand side.

Matrix Problems

From now on, the coefficient matrix A is allowed to have more rows than columns, i.e.,

$$A \in \mathbb{R}^{m \times n} \qquad \text{with} \qquad m \geq n.$$

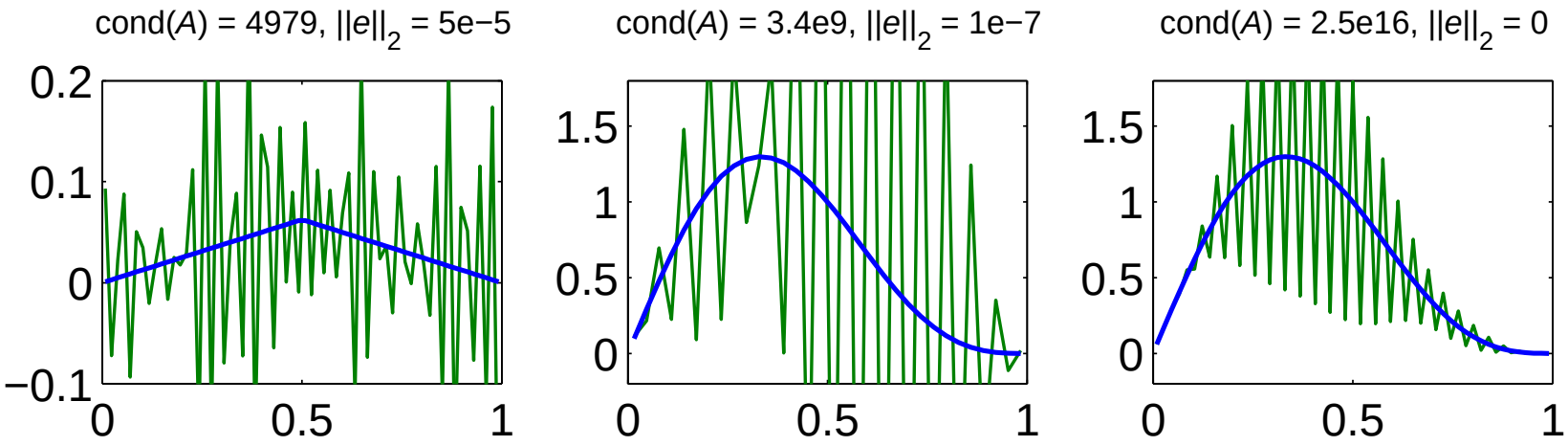
For $m > n$ it is natural to consider the least squares problem $\min_x \|A\,x - b\|_2$.

When we say "naive solution" we either mean the solution $A^{-1}b$ (when $m = n$) or the least squares solution (when $m > n$).

We emphasize the convenient fact that the naive solution has precisely the same SVD expansion in both cases:

$$x^{\text{naive}} = \sum_{i=1}^{n} \frac{u_i^T b}{\sigma_i}\, v_i.$$

Naive Solutions are Useless



Exact solutions (blue smooth lines) together with the naive solutions (jagged green lines) to two test problems.

Left: `deriv2` with $n = 64$.

Middle and right: `gravity` with $n = 32$ and $n = 53$.

Need For Regularization

Discrete ill-posed problems are characterized by having coefficient matrices with a very large condition number.

The naive solution is very sensitive to any perturbation of the right-hand side, representing the errors in the data.

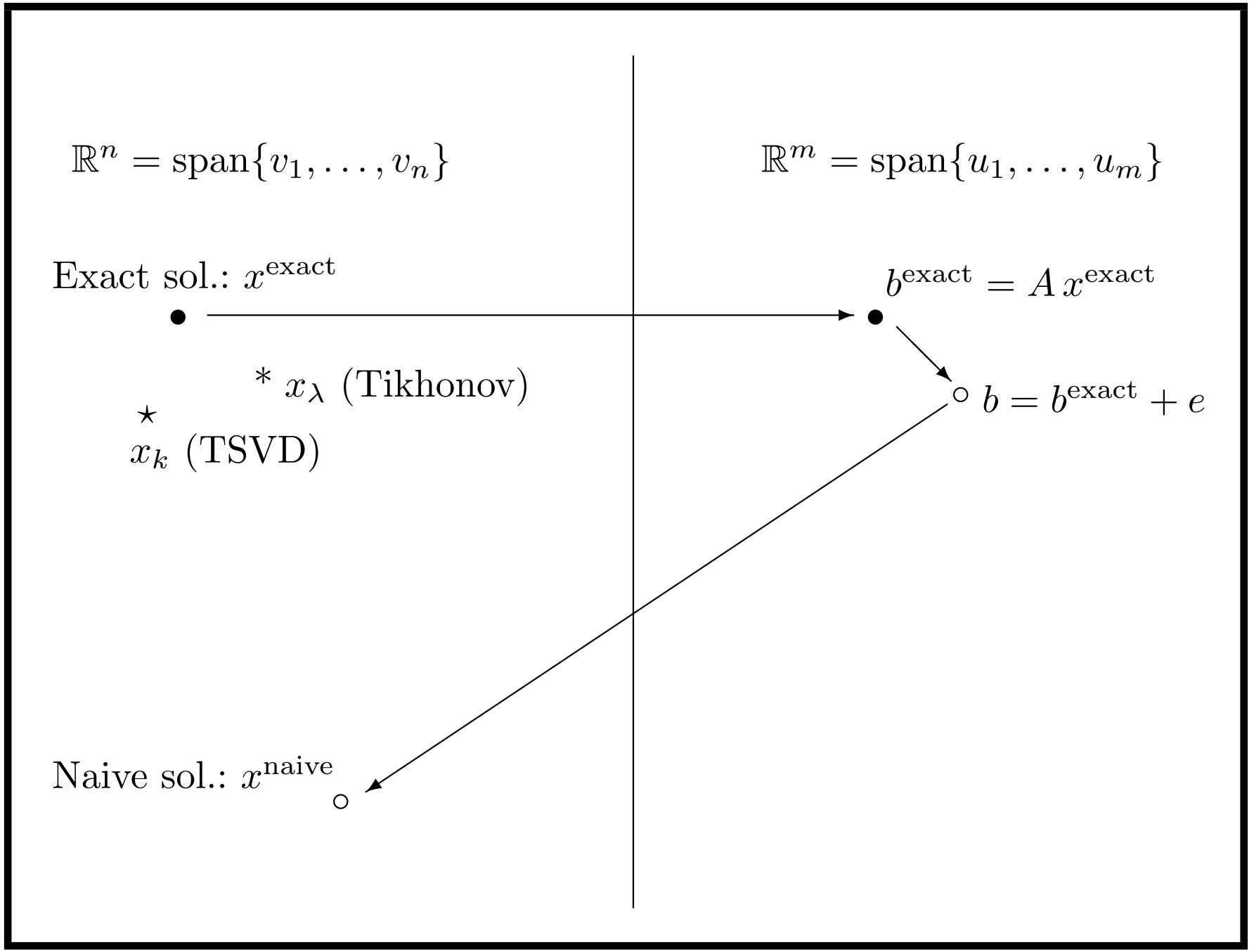
Specifically, assume that the exact and perturbed solutions x^{exact} and x satisfy

$$A\,x^{\text{exact}} = b^{\text{exact}}, \qquad A\,x = b = b^{\text{exact}} + e,$$

where e denotes the perturbation. Then classical perturbation theory leads to the bound

$$\frac{\|x^{\text{exact}} - x\|_2}{\|x^{\text{exact}}\|_2} \leq \text{cond}(A)\,\frac{\|e\|_2}{\|b^{\text{exact}}\|_2}.$$

Since $\text{cond}(A) = \sigma_1/\sigma_n$ is large, this implies that x can be very far from x^{exact}.


$\mathbb{R}^n = \text{span}\{v_1, \ldots, v_n\}$
$\mathbb{R}^m = \text{span}\{u_1, \ldots, u_m\}$
Exact sol.: x^{exact}
$b^{\text{exact}} = A\,x^{\text{exact}}$
$* \; x_\lambda$ (Tikhonov)
$\circ \; b = b^{\text{exact}} + e$
$\star$
x_k (TSVD)
Naive sol.: x^{naive}
$\circ$

Regularization Methods → Spectral Filtering

Almost all the regularization methods treated in this course produce solutions which can be expressed as a filtered SVD expansion of the form

$$x_{\text{reg}} = \sum_{i=1}^{n} \varphi_i \, \frac{u_i^T b}{\sigma_i} \, v_i ,$$

where φ_i are the *filter factors* associated with the method.

These methods are called *spectral filtering methods* because the SVD basis can be considered as a spectral basis.

Truncated SVD

A simple way to reduce the influence of the noise is to discard the SVD coefficients corresponding to the smallest singular values.

Define truncated SVD (TSVD) solution as

$$x_k = \sum_{i=1}^{k} \frac{u_i^T b}{\sigma_i} v_i, \qquad k < n.$$

Reg. Tools: `tsvd`. Can show that if $\mathrm{Cov}(b) = \eta^2 I$ then

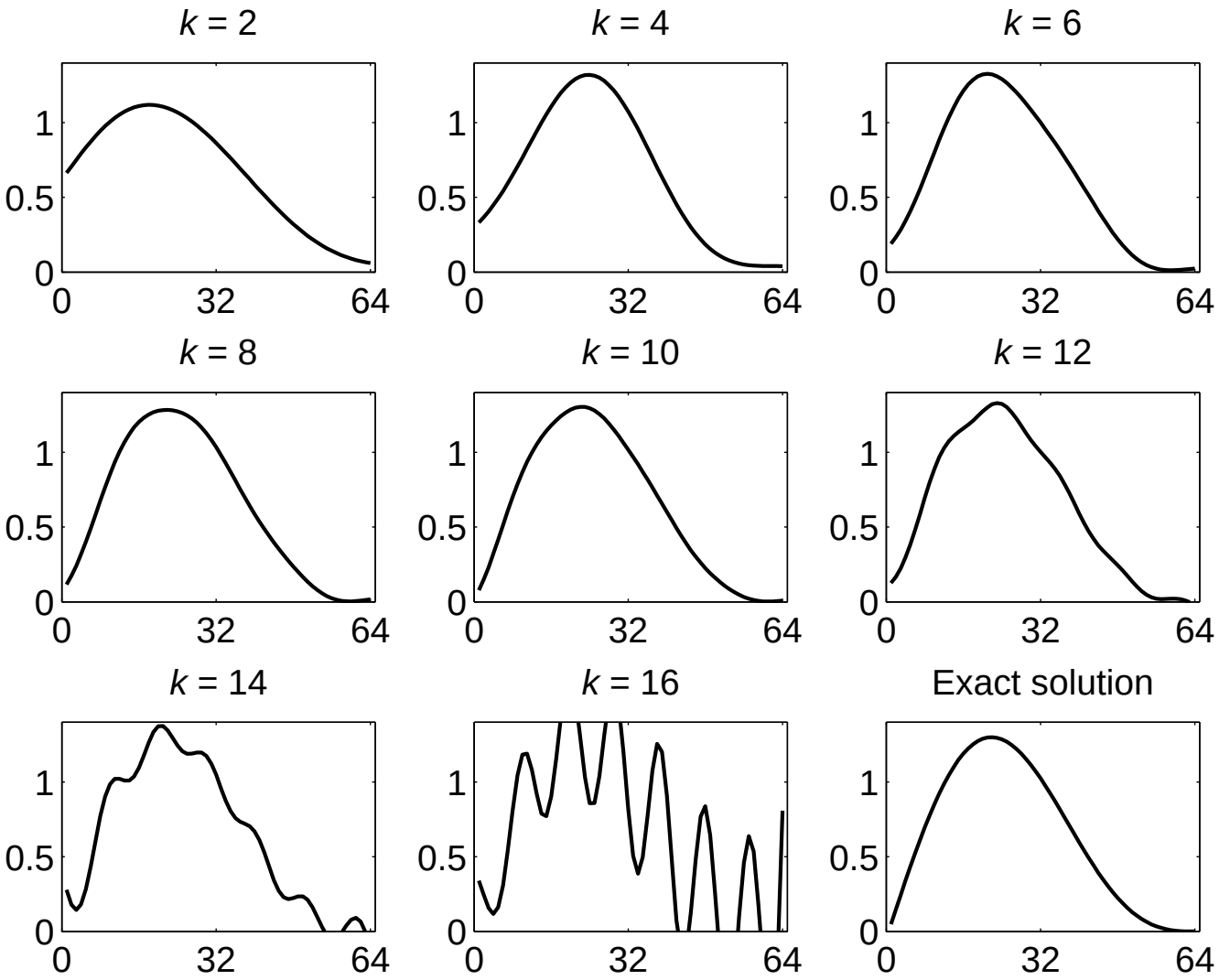
$$\mathrm{Cov}(x_k) = \eta^2 \sum_{i=1}^{k} \frac{1}{\sigma_i^2} v_i \, v_i^T$$

and thus we can expect that

$$\|x_k\|_2 \ll \|x^{\mathrm{naive}}\|_2 \qquad \text{and} \qquad \|\mathrm{Cov}(x_k)\|_2 \ll \|\mathrm{Cov}(x^{\mathrm{naive}})\|_2.$$

The prize we pay for smaller covariance is *bias:* $\mathcal{E}(x_k) \neq \mathcal{E}(x^{\mathrm{naive}})$.

Truncated SVD Solutions



The Truncation Parameter

Note: the truncation parameter k in

$$x_k = \sum_{i=1}^{k} \frac{u_i^T b}{\sigma_i} v_i$$

is dictated by the coefficients $u_i^T b$, not the singular values!

Basically we should choose k as the index i where $|u_i^T b|$ start to "level off" due to the noise.

Discrete Tikhonov Regularization

Minimization of the residual takes the form

$$\min_x \|A\,x - b\|_2 \ , \qquad A \in \mathbb{R}^{m\times n} \ ,$$

where A and b are obtained by discretization of the integral eq.

We also introduce a smoothing norm

$$\Omega(x) = \|x\|_2$$

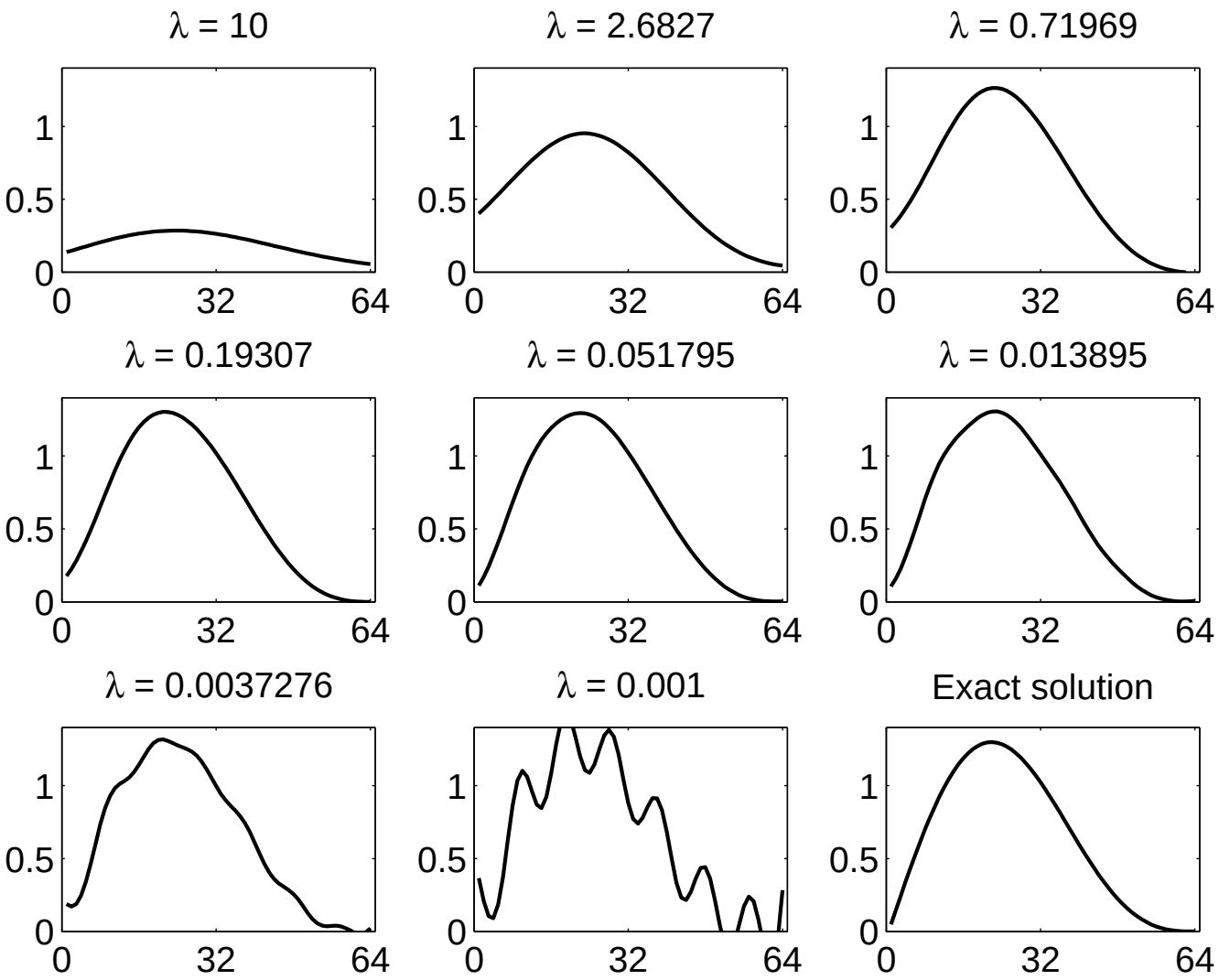
that penalizes a large solution norm.

The resulting discrete Tikhonov problem is thus

$$\boxed{\min_x \ \left\{ \|A\,x - b\|_2^2 + \lambda^2 \, \|x\|_2^2 \right\}.}$$

Regularization Tools: `tikhonov`.

Tikhonov Solutions



Efficient Implementation

The original formulation

$$\min \left\{ \|A\,x - b\|_2^2 + \lambda^2 \, \|x\|_2^2 \right\} .$$

Two alternative formulations

$$(A^T A + \lambda^2 I)\, x = A^T b$$

$$\min \left\| \begin{pmatrix} A \\ \lambda\, I \end{pmatrix} x - \begin{pmatrix} b \\ 0 \end{pmatrix} \right\|_2$$

The first shows that we have a linear problem. The second shows how to solve it stably:

- treat it as a least squares problem,
- utilize any sparsity or structure.

SVD and Tikhonov Regularization

We can write the discrete Tikhonov solution x_λ in terms of the SVD of A as

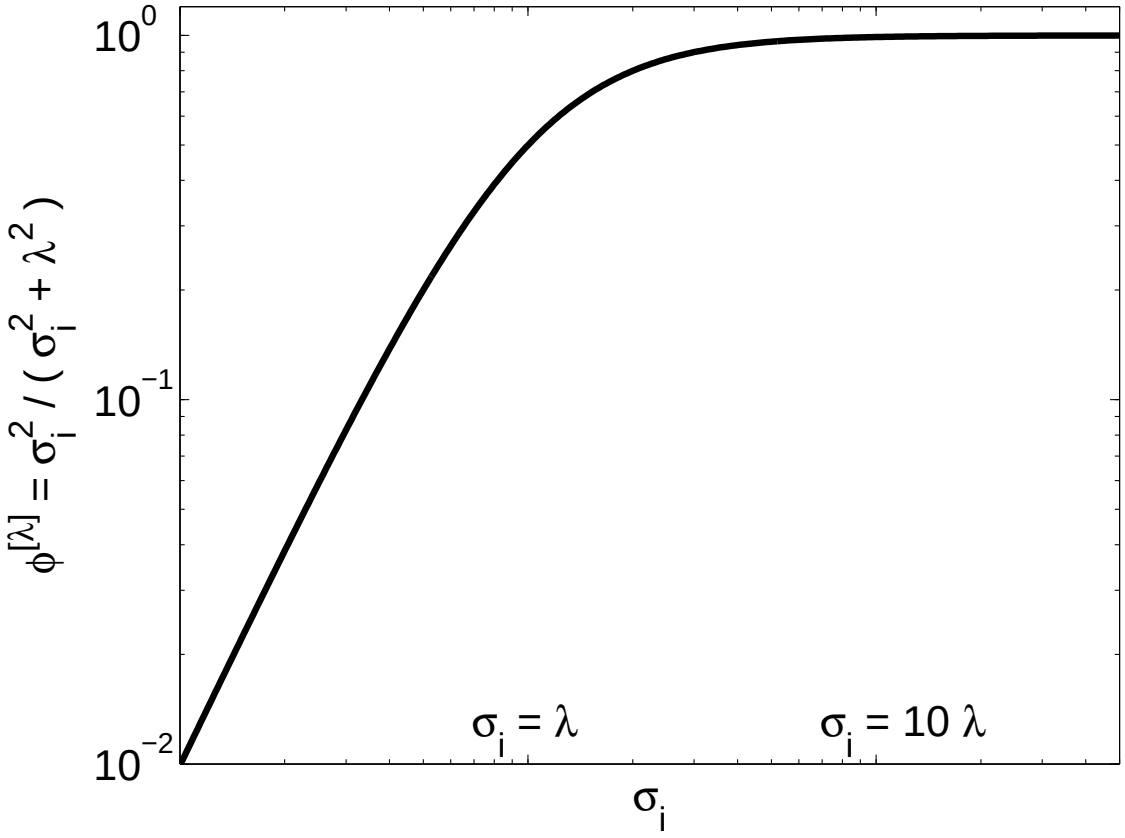
$$x_\lambda = \sum_{i=1}^{n} \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \, \frac{u_i^T b}{\sigma_i} \, v_i = \sum_{i=1}^{n} \phi_i^{[\lambda]} \, \frac{u_i^T b}{\sigma_i} .$$

The *filter factors* are given by

$$\phi_i^{[\lambda]} = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \ ,$$

and their purpose is to dampen the components in the solution corresponding to small σ_i.

Tikhonov Filter Factors

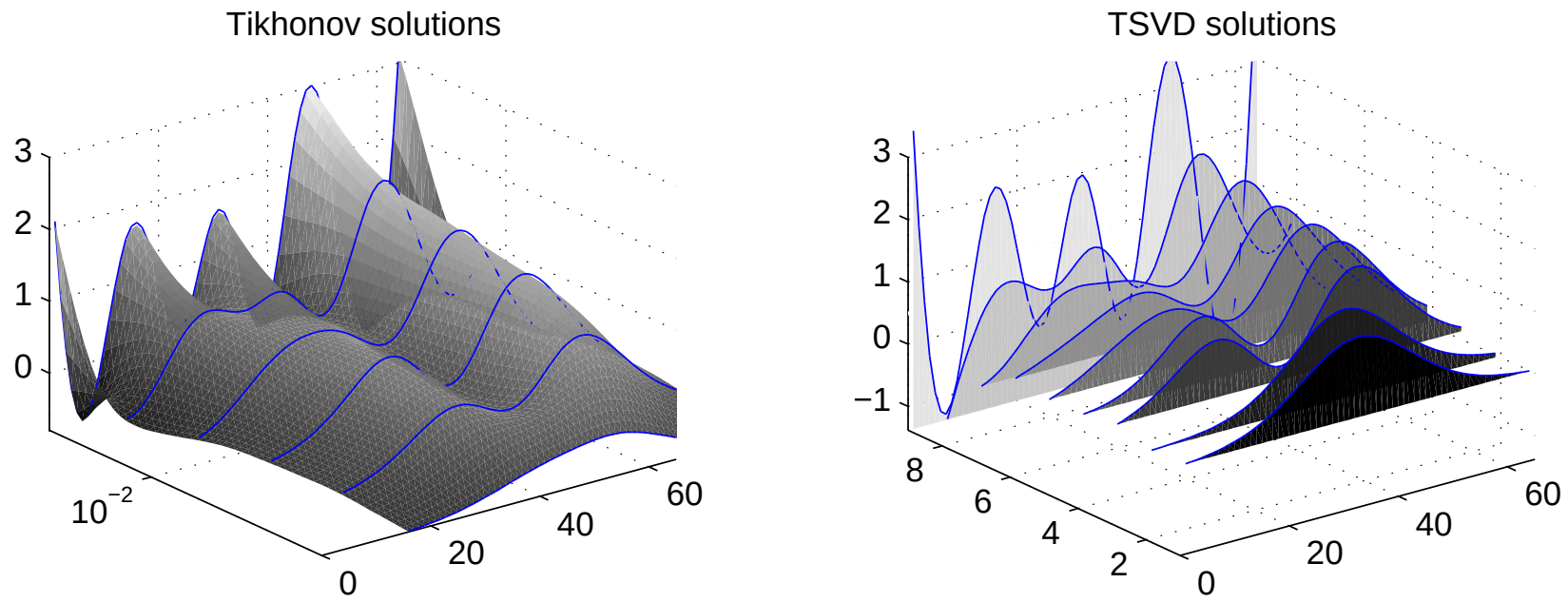


$$\phi_i^{[\lambda]} = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \approx \begin{cases} 1\ , & \sigma_i \gg \lambda \\ \sigma_i^2/\lambda^2\ , & \sigma_i \ll \lambda. \end{cases}$$

TSVD and Tikhonov Regularization

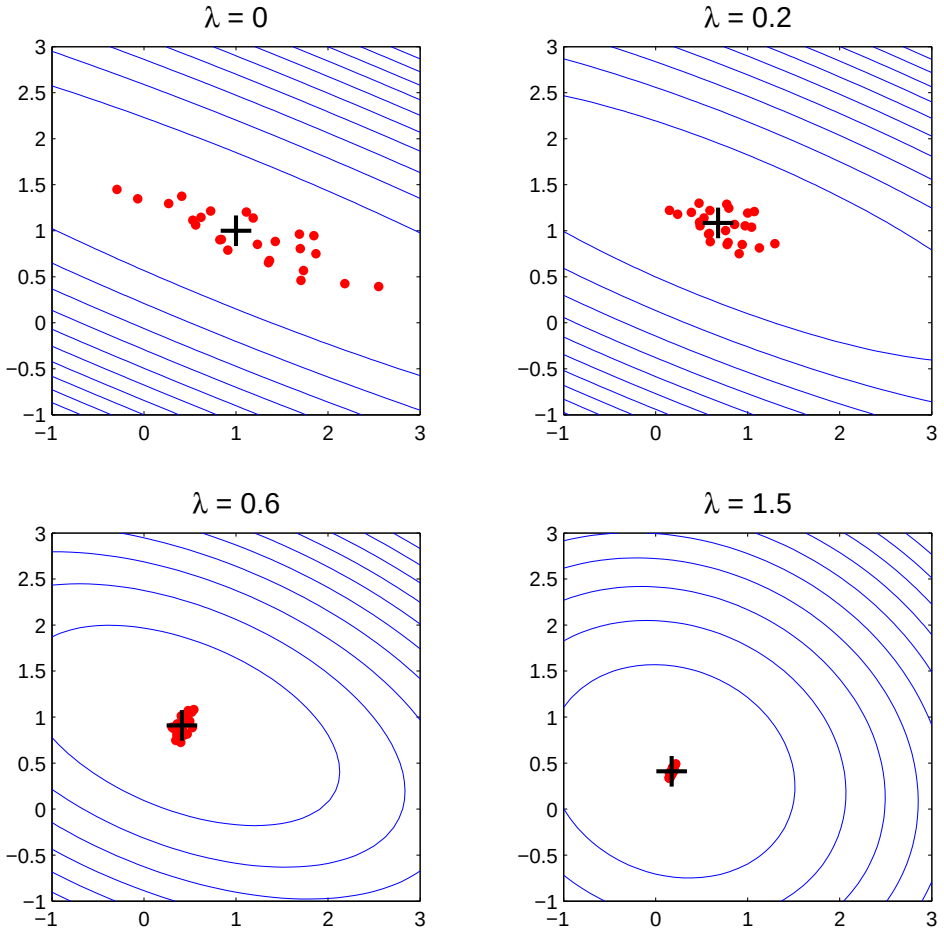
TSVD and Tikhonov solutions are both filtered SVD expansions.

The regularization parameter is either k or λ.



For each k, there exists a λ such that $x_\lambda \approx x_k$.

Illustration of Sensitivity

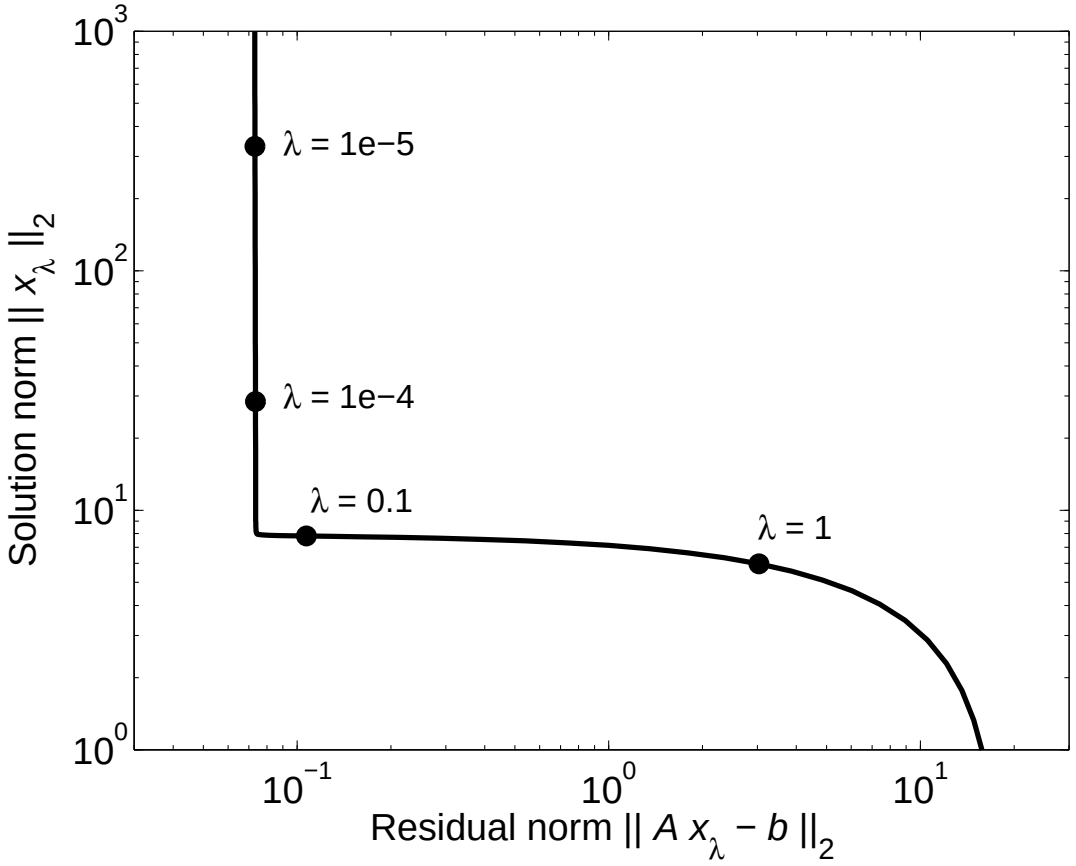


Red dots: x_λ for 25 random perturbations of b.

Black crosses: unperturbed x_λ – note the bias.

The L-Curve for Tikhonov Regularization

Plot of $\|x_\lambda\|_2$ versus $\|A\,x_\lambda - b\|_2$ in *log-log scale.*



The Story So Far

- The purpose of regularization is to suppress the influence of the noise, while still achieving an *approximation* to the exact solution.
- This is done by filtering the SVD components, e.g., by
  - a sharp filter $\rightarrow$ truncated SVD
  - a smooth filter $\rightarrow$ Tikhonov.

  This works because it is mainly the "high-frequency" SVD components that are affected by the noise.
- The discrete Picard condition ensures that the "low-frequency" SVD components are approximated well.
- The L-curve provides a means for displaying the tradeoff between solution norm and residual norm (over- versus under-smoothing).

Choosing the Regularization Parameter

At our disposal: several regularization methods, based on filtering of the SVD components.

Often fairly straightforward to "eyeball" a good TSVD truncation parameter from the Picard plot.

Need: a reliable and automated technique for choosing the regularization parameter, such as k (for TSVD) or λ (for Tikhonov).

1. Perspectives on regularization
2. The discrepancy principle
3. Generalized cross validation (GCV)
4. The L-curve criterion
5. The NCP method

Once Again: Tikhonov Regularization

Focus on Tikhonov regularization; the ideas carry over to many other methods.

Recall that the Tikhonov solution x_λ solves the problem

$$\min_x \left\{ \|A\,x - b\|_2^2 + \lambda^2 \|x\|_2^2 \right\},$$

and that it is formally given by

$$x_\lambda = (A^T A + \lambda^2 I)^{-1} A^T b = A_\lambda^{\#} b,$$

where $A_\lambda^{\#} = (A^T A + \lambda^2 I)^{-1} A^T$ is a "regularized inverse."

Our noise model

$$b = b^{\text{exact}} + e$$

where $b^{\text{exact}} = A\,x^{\text{exact}}$ and e is the error.

Classical and Pragmatic Parameter-Choice

Assume we are given the problem $A\,x = b$ with $b = b^{\text{exact}} + e$, and that we have a strategy for choosing the regularization parameter λ as a function of the "noise level" $\|e\|_2$.

Then *classical* parameter-choice analysis is concerned with the convergence rates of

$$x_\lambda \to x^{\text{exact}} \qquad \text{as} \qquad \|e\|_2 \to 0 \quad \text{and} \quad \lambda \to 0 \ .$$

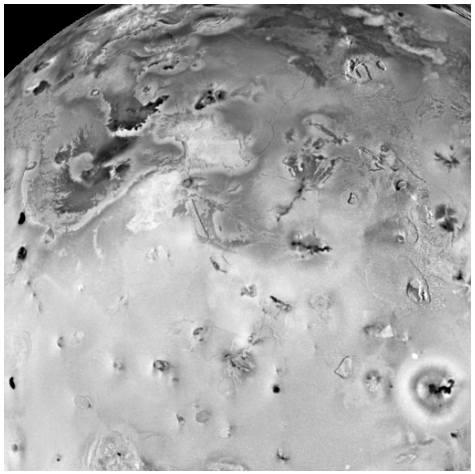
The typical situation in practice is different:

- The norm $\|e\|_2$ is not known, and
- the errors are fixed (not practical to repeat the measurements).

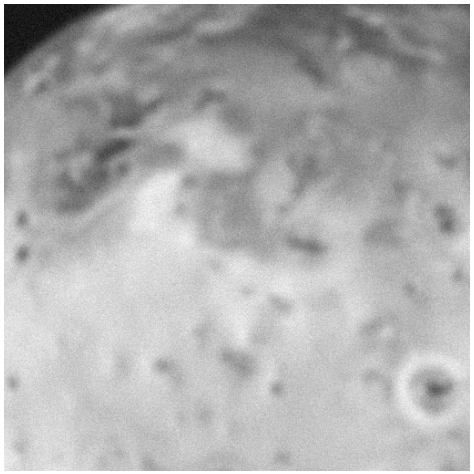
The *pragmatic* approach to choosing the regularization parameter is based on the forward/prediction error, or the backward error.

An Example (Image of Io, a Moon of Saturn)

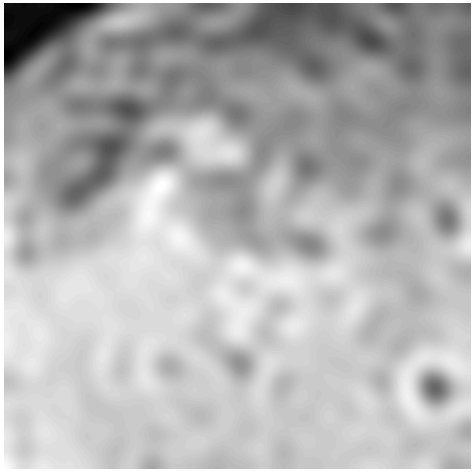
Exact



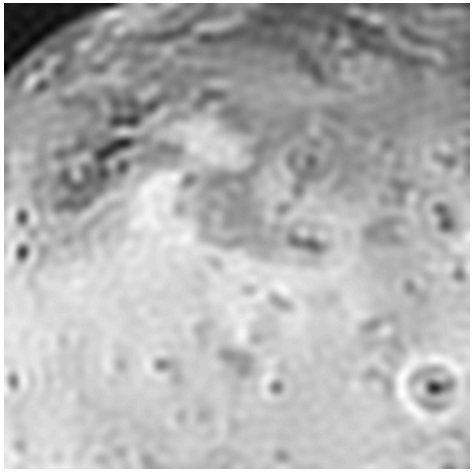
Blurred



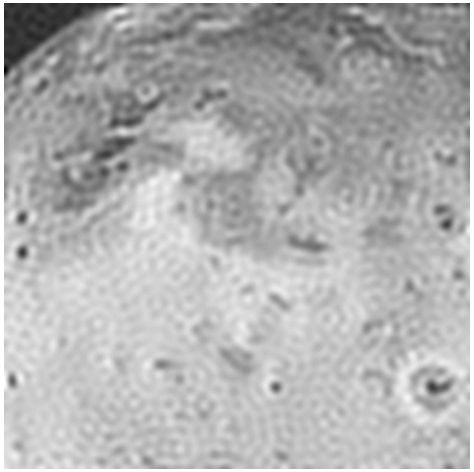
λ too large



$\lambda \approx$ ok



λ too small



Perspectives on Regularization

Problem formulation: balance fit (residual) and size of solution.

$$x_\lambda = \arg\min \left\{ \|A\,x - b\|_2^2 + \lambda^2 \|x\|_2^2 \right\}$$

Cannot be used for choosing λ.

Forward error: balance regularization and perturbation errors.

$$\begin{aligned} x^{\text{exact}} - x_\lambda &= x^{\text{exact}} - A_\lambda^{\#}(b^{\text{exact}} + e) \\ &= \left(I - A_\lambda^{\#} A\right) x^{\text{exact}} - A_\lambda^{\#} e \ . \end{aligned}$$

Backward/prediction error: balance residual and perturbation.

$$\begin{aligned} b^{\text{exact}} - A\,x_\lambda &= b^{\text{exact}} - A\,A_\lambda^{\#}(b^{\text{exact}} + e) \\ &= \left(I - A\,A_\lambda^{\#}\right) b^{\text{exact}} - A\,A_\lambda^{\#} e \ . \end{aligned}$$

More About the Forward Error

The forward error in the SVD basis:

$$\begin{aligned} x^{\text{exact}} - x_\lambda &= x^{\text{exact}} - V\,\Phi^{[\lambda]}\,\Sigma^{-1}\,U^T b \\ &= x^{\text{exact}} - V\,\Phi^{[\lambda]}\,\Sigma^{-1}\,U^T A\,x^{\text{exact}} - V\,\Phi^{[\lambda]}\,\Sigma^{-1}\,U^T e \\ &= V\left(I - \Phi^{[\lambda]}\right) V^T x^{\text{exact}} - V\,\Phi^{[\lambda]}\,\Sigma^{-1}\,U^T e. \end{aligned}$$

The first term is the *regularization error:*

$$\Delta x_{\text{bias}} = V\left(I - \Phi^{[\lambda]}\right) V^T x^{\text{exact}} = \sum_{i=1}^{n} \left(1 - \varphi_i^{[\lambda]}\right) \left(v_i^T x^{\text{exact}}\right) v_i,$$

and we recognize this as (minus) the bias term.

The second error term is the *perturbation error:*

$$\Delta x_{\text{pert}} = V\,\Phi^{[\lambda]}\,\Sigma^{-1}\,U^T e.$$

Regularization and Perturbation Errors – TSVD

For TSVD solutions, the regularization and perturbation errors take the form

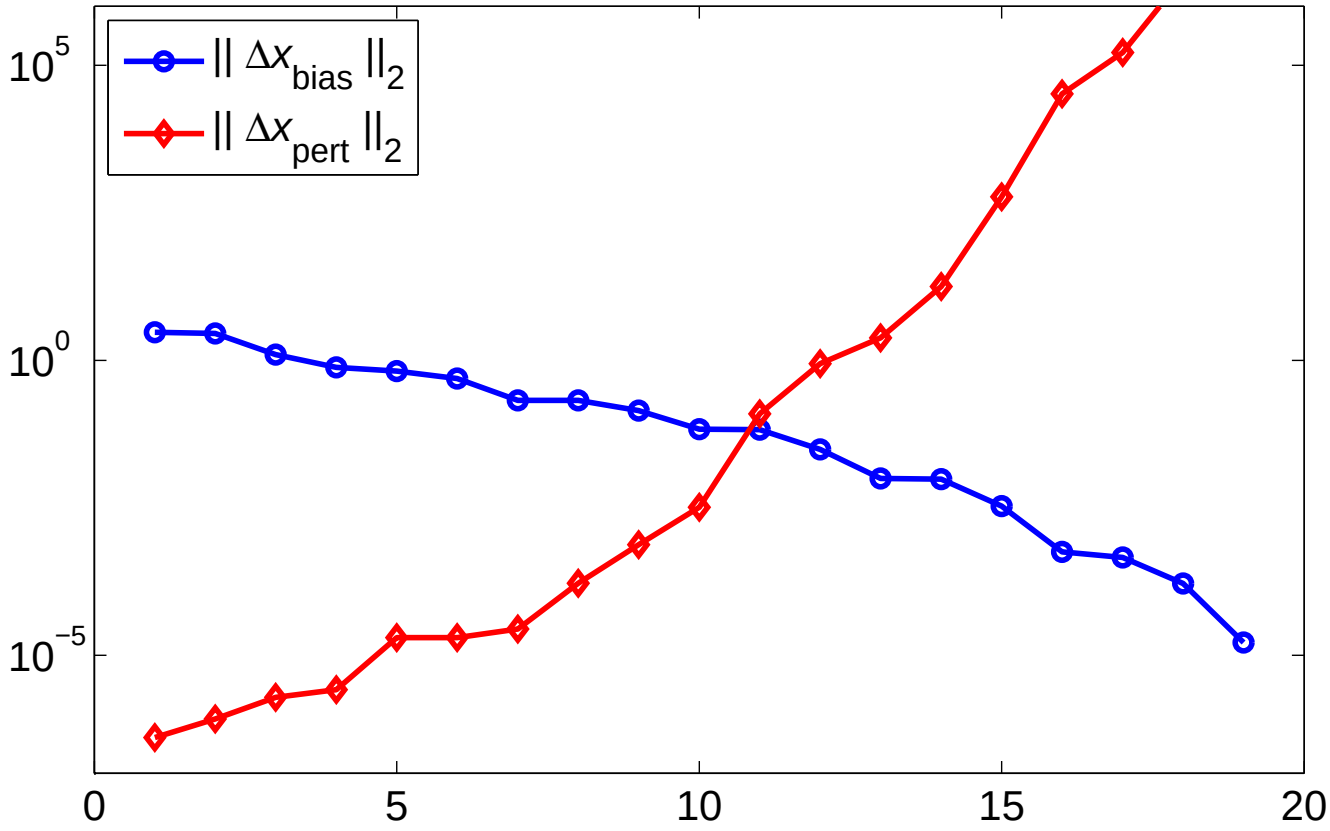
$$\Delta x_{\text{bias}} = \sum_{i=k+1}^{n} (v_i^T x^{\text{exact}})\, v_i, \qquad \Delta x_{\text{pert}} = \sum_{i=1}^{k} \frac{u_i^T e}{\sigma_i}\, v_i.$$

We use the truncation parameter k to prevent the perturbation error from blowing up (due to the division by the small singular values), at the cost of introducing bias in the regularized solution.

A "good" choice of the truncation parameter k should balance these two components of the forward error (see next slide).

The behavior of $\|x_k\|_2$ and $\|A\,x_k - b\|_2$ is closely related to these errors – see the analysis in §5.1.

The Regularization and Perturbation Errors



The norm of the regularization and perturbation error for TSVD as a function of the truncation parameter k. The two different errors approximately balance each other for $k = 11$.

The TSVD Residual

Let k_η denote the index that marks the transition between decaying and flat coefficients $|u_i^T b|$.

Due to the discrete Picard condition, the coefficients $|u_i^T b|/\sigma_i$ will also decay, on the average, for all $i < k_\eta$.

$$k < k_\eta : \ \|A\,x_k - b\|_2^2 \approx \sum_{i=k+1}^{k_\eta} (u_i^T b)^2 + (n - k_\eta)\eta^2 \approx \sum_{i=k+1}^{k_\eta} (u_i^T b^{\text{exact}})^2$$

$$k > k_\eta : \ \|A\,x_k - b\|_2^2 \approx (n - k)\,\eta^2.$$

For $k < k_\eta$ the residual norm decreases steadily with k.

For $k > k_\eta$ it decreases much more slowly.

The transition between the two types of behavior occurs at $k = k_\eta$ when the regularization and perturbation errors are balanced.

The Discrepancy Principle

Recall that $\mathcal{E}(\|e\|_2) \approx n^{1/2}\eta$.

We should ideally choose k such that $\|A\,x_k - b\|_2 \approx (n-k)^{1/2}\,\eta$.

The *discrepancy principle* (DP) seeks to combine this:

Assume we have an upper bound δ_e for the noise level, then solve

$$\|A\,x_\lambda - b\|_2 = \tau\,\delta_e\ , \quad \text{where} \qquad \|e\|_2 \le \delta_e$$

and τ is some parameter $\tau = O(1)$. See next slide.

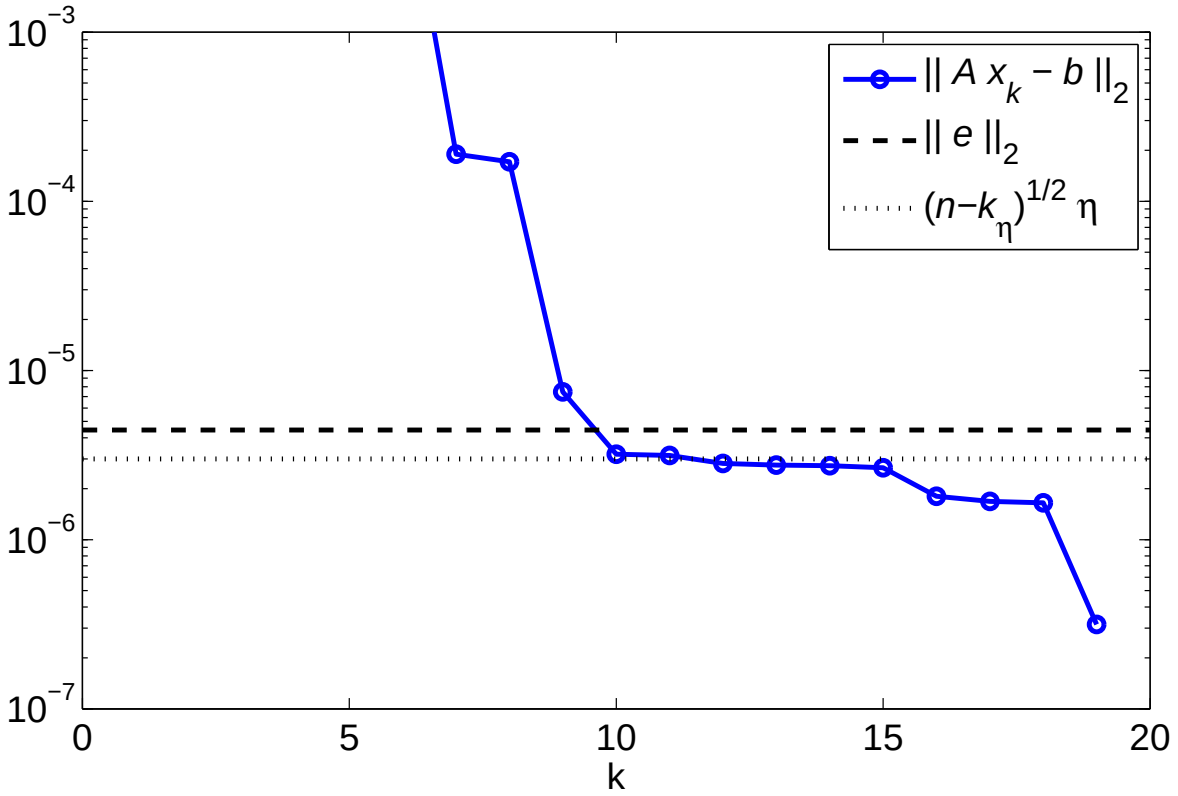
A statistician's point of view. Write $x_\lambda = A_\lambda^{\#} b$ and assume $\mathrm{Cov}(b) = \eta^2 I$; choose the λ that solves

$$\|A\,x_\lambda - b\|_2 = \left(\|e\|_2^2 - \eta^2\,\mathrm{trace}(A\,A_\lambda^{\#})\right)^{1/2}\ .$$

Note that the right-hand side now depends on λ.

Both versions of the DP are very sensitive to the estimate δ_e.

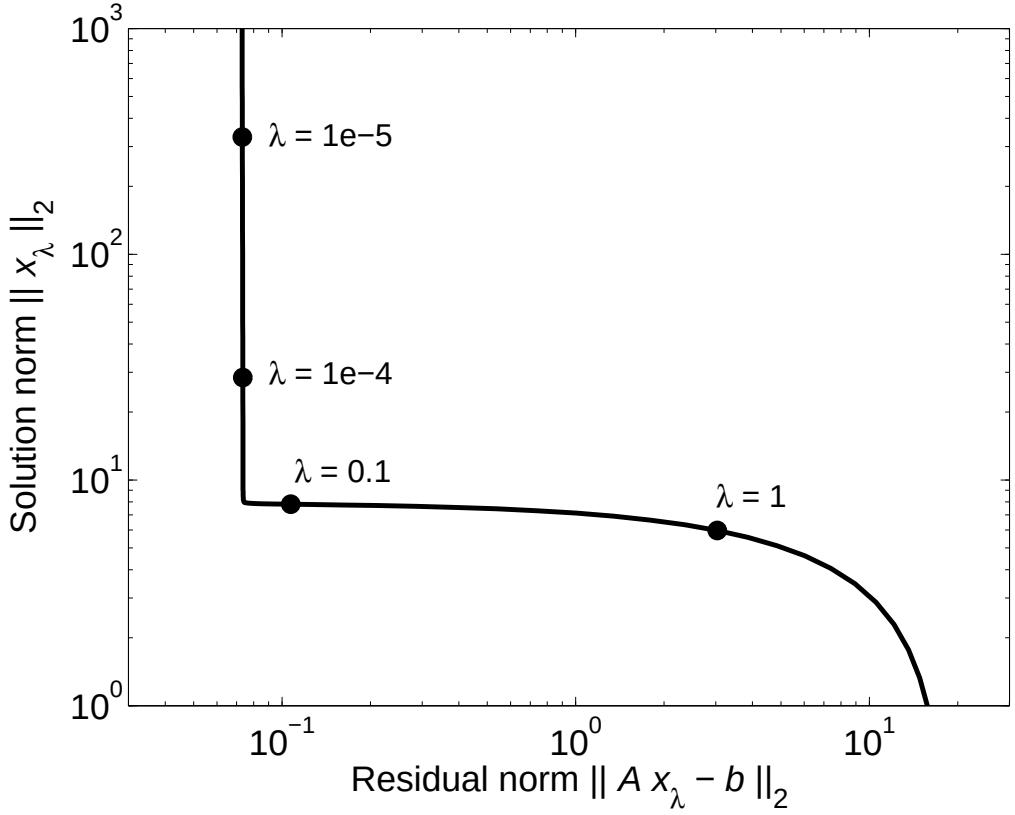
Illustration of the Discrepancy Principle



The choice $\|A\,x_k - b\|_2 \approx (n - k_\eta)^{1/2}\eta$ leads to a too large value of the truncation parameter k, while the more conservative choice $\|A\,x_k - b\|_2 \approx \|e\|_2$ leads to a better value of k.

The L-Curve for Tikhonov Regularization

Recall that the L-curve is a log-log-plot of the solution norm versus the residual norm, with λ as the parameter.



Parameter-Choice and the L-Curve

Recall that the L-curve basically consists of two parts.

- A "flat" part where the regularization errors dominates.
- A "steep" part where the perturbation error dominates.

The optimal regularization parameter (in the pragmatic sense) must lie somewhere near the L-curve's corner.

The component b^{exact} dominates when λ is large:

$$\|x_\lambda\|_2 \approx \|x^{\text{exact}}\|_2 \text{ (constant)}$$

$$\|b - A\,x_\lambda\|_2 \text{ increases with } \lambda.$$

The error e dominates when λ is small:

$$\|x_\lambda\|_2 \text{ increases with } \lambda^{-1}$$

$$\|b - A\,x_\lambda\|_2 \approx \|e\|_2 \text{ (constant.)}$$

The L-Curve Criterion

The flat and the steep parts of the L-curve represent solutions that are dominated by regularization errors and perturbation errors.

- The balance between these two errors must occur near the L-curve's corner.
- The two parts – and the corner – are emphasized in log-log scale.
- Log-log scale is insensitive to scalings of A and b.

An *operational* definition of the corner is required.

Write the L-curve as

$$(\log \|A\,x_\lambda - b\|_2\,,\, \log \|x_\lambda\|_2)$$

and seek the point with maximum curvature.

The Curvature of the L-Curve

We want to derive an analytical expression for the L-curve's curvature ζ in log-log scale. Define

$$\xi = \|x_\lambda\|_2^2 \ , \qquad \rho = \|A\,x_\lambda - b\|_2^2$$

and

$$\hat{\xi} = \log \eta \ , \qquad \hat{\rho} = \log \rho \ .$$

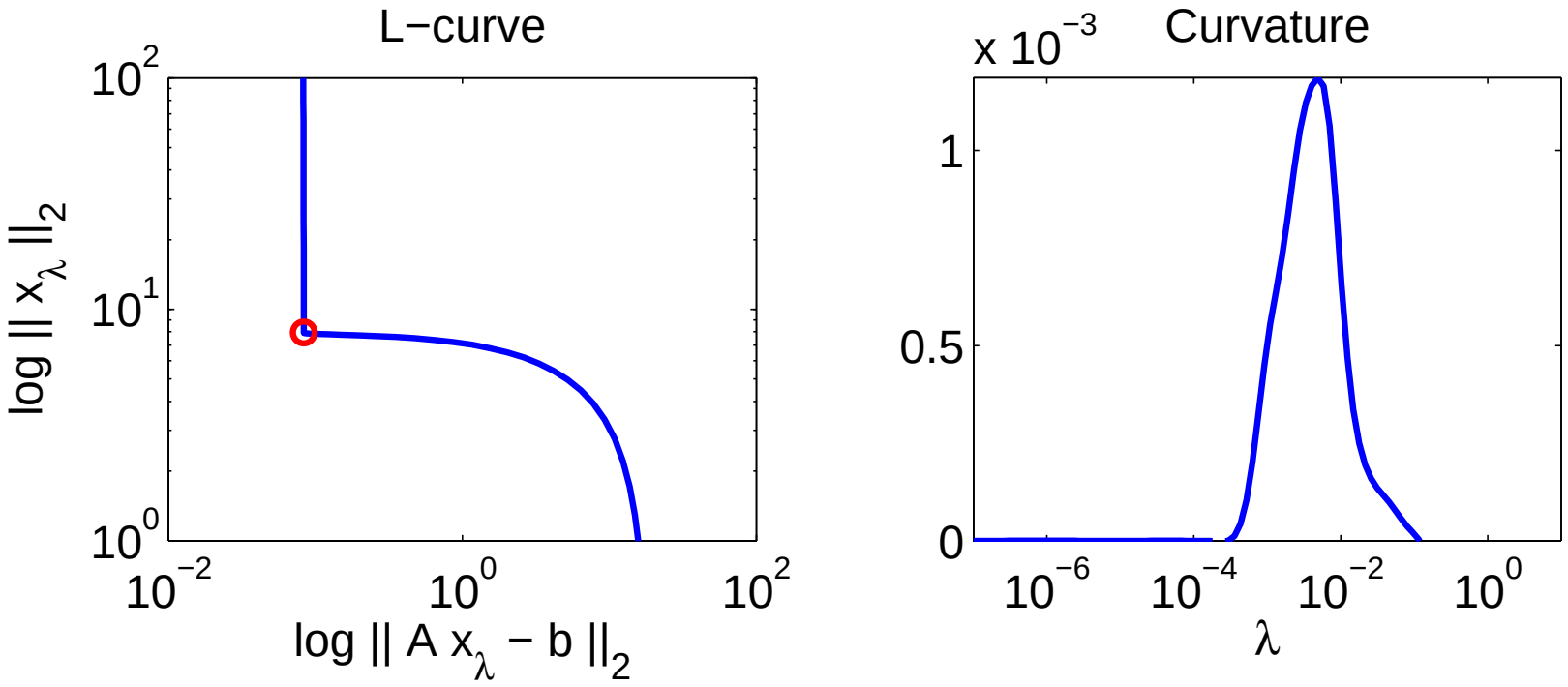
Then the curvature is given by

$$\hat{c}_\lambda = 2\,\frac{\hat{\rho}'\hat{\xi}'' - \hat{\rho}''\hat{\xi}'}{((\hat{\rho}')^2 + (\hat{\xi}')^2)^{3/2}} \ ,$$

where a prime denotes differentiation with respect to λ.

This can be used to define the "corner" of the L-curve as the point with maximum curvature.

Illustration



An L-curve and the corresponding curvature $\hat{c}_\lambda$ as a function of λ. The corner, which corresponds to the point with maximum curvature, is marked by the red circle; it occurs for $\lambda_{\mathrm{L}} = 4.86 \cdot 10^{-3}$.

The Prediction Error

A different kind of goal: find the value of λ or k such that $A\,x_\lambda$ or $A\,x_k$ predicts the *exact* data $b^{\text{exact}} = A\,x^{\text{exact}}$ as well as possible.

We split the analysis in two cases, depending on k:

$$k < k_\eta : \qquad \|A\,x_k - b^{\text{exact}}\|_2^2 \approx k\,\eta^2 + \sum_{i=k+1}^{k_\eta} (u_i^T b^{\text{exact}})^2$$

$$k > k_\eta : \qquad \|A\,x_k - b^{\text{exact}}\|_2^2 \approx k\,\eta^2.$$

For $k < k_\eta$ the norm of the prediction error decreases with k.

For $k > k_\eta$ the norm increases with k.

The minimum arises near the transition, i.e., for $k \approx k_\eta$. Hence it makes good sense to search for the regularization parameter that minimizes the prediction error. But b^{exact} is unknown ...

(Ordinary) Cross-Validation

Leave-one-out approach:
skip ith element b_i and predict this element.

$$\begin{aligned} A^{(i)} &= A([1\colon i-1, i+1\colon m], :) \\ b^{(i)} &= b([1\colon i-1, i+1\colon m]) \\ x_\lambda^{(i)} &= \left(A^{(i)}\right)_\lambda^{\#} b^{(i)} \quad \text{(Tikh. sol. to reduced problem)} \\ b_i^{\text{predict}} &= A(i, :)\, x_\lambda^{(i)} \quad \text{(prediction of “missing” element.)} \end{aligned}$$

The optimal λ minimizes the quantity

$$\mathcal{C}(\lambda) = \sum_{i=1}^{m} \left(b_i - b_i^{\text{predict}}\right)^2 .$$

But λ is hard to compute, and depends on the ordering of the data.

Generalized Cross-Validation

Want a scheme for which λ is independent of any orthogonal transformation of b (incl. a permutation of the elements).

Minimize the GCV function

$$G(\lambda) = \frac{\|A\,x_\lambda - b\|_2^2}{\mathrm{trace}(I_m - A\,A_\lambda^{\#})^2}$$

where

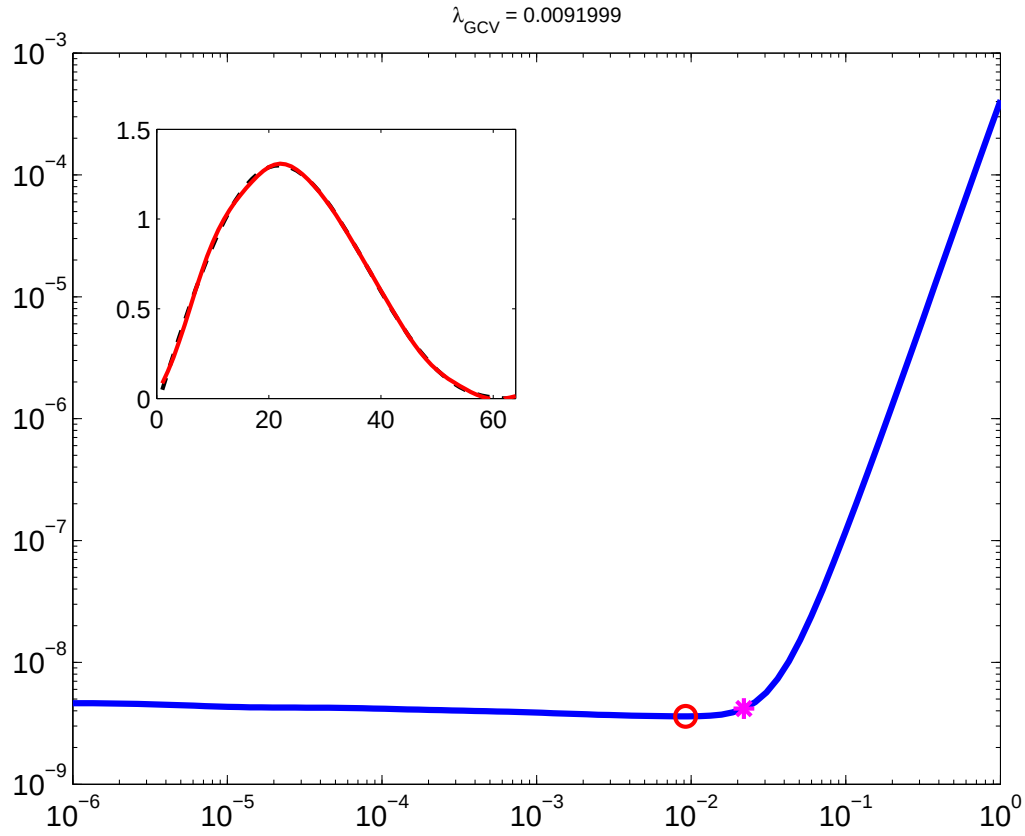
$$\mathrm{trace}(I_m - A\,A_\lambda^{\#}) = m - \sum_{i=1}^{n} \varphi_i^{[\lambda]} \ .$$

Easy to compute the trace term when the SVD is available.

For TSVD the trace term is particularly simple:

$$m - \sum_{i=1}^{n} \varphi_i^{[\lambda]} = m - k \ .$$

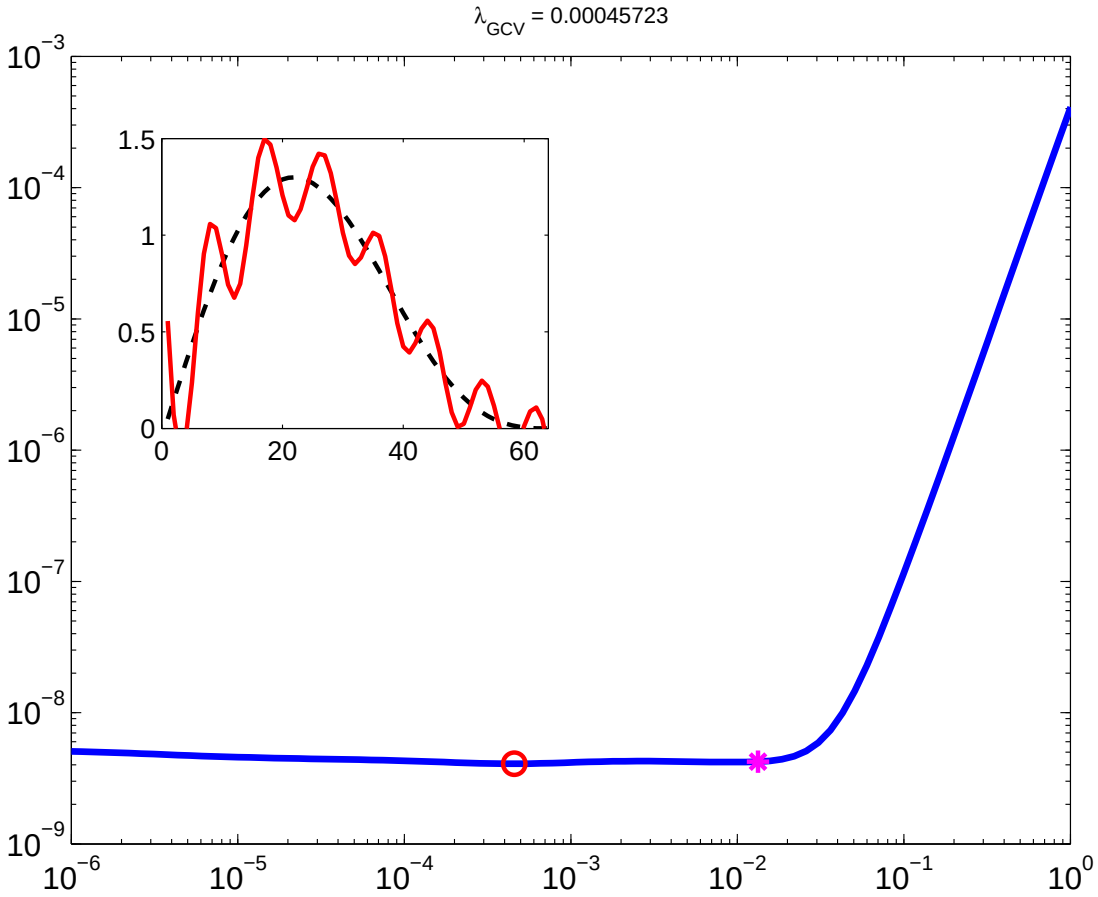
The GCV Function



The GCV function $G(\lambda)$ for Tikhonov regularization; the red circle shows the parameter λ_{GCV} as the minimum of the GCV function, while the cross indicates the location of the optimal parameter.

Occasional Failure

Occasional failure leading to a too small λ; more pronounced for correlated noise.



Extracting Signal in Noise

An observation about the residual vector.

- If λ is too large, not all information in b has not been extracted.
- If λ is too small, only noise is left in the residual.

Choose the λ for which the residual vector changes character from "signal" to "noise."

Our tool: the **normalized cumulative periodogram** (NCP).
Let $p_\lambda \in \mathbb{R}^{n/2}$ be the residual's power spectrum, with elements

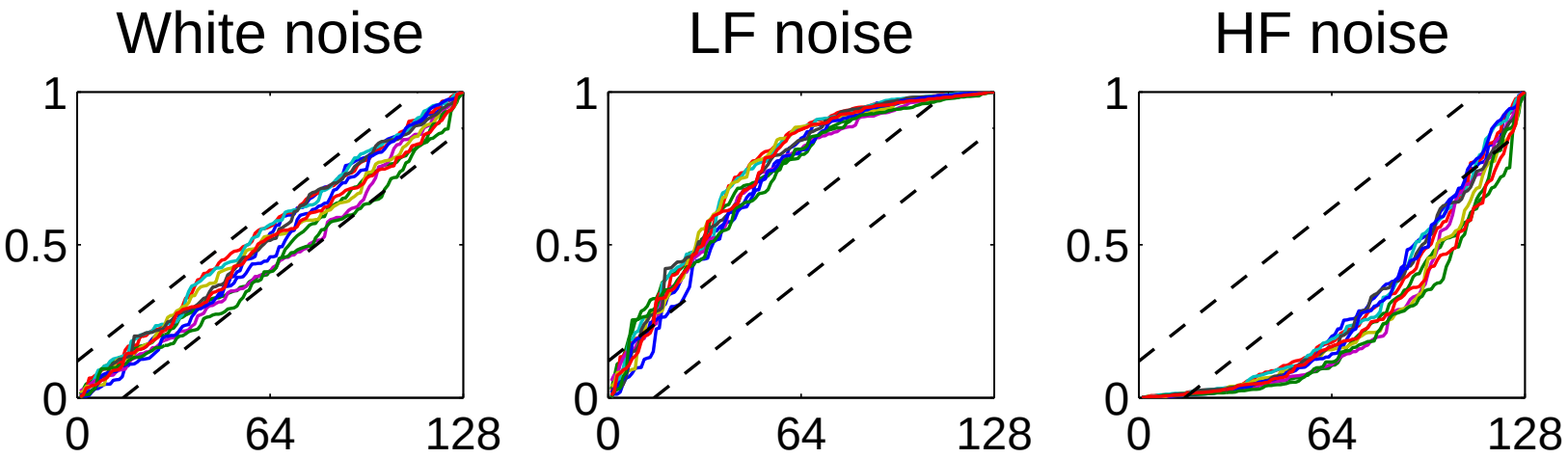
$$(p_\lambda)_k = |\mathrm{dft}(A\,x_\lambda - b)_k|^2, \qquad k = 1, 2, \ldots, n/2\ .$$

Then the vector $c(r_\lambda) \in \mathbb{R}^{n/2-1}$ with elements

$$c(r_\lambda) = \frac{\|p_\lambda(2\colon k{+}1)\|_1}{\|p_\lambda(2\colon n)\|_1}, \qquad k = 1, \ldots, n/2 - 1$$

is the NCP for the residual vector.

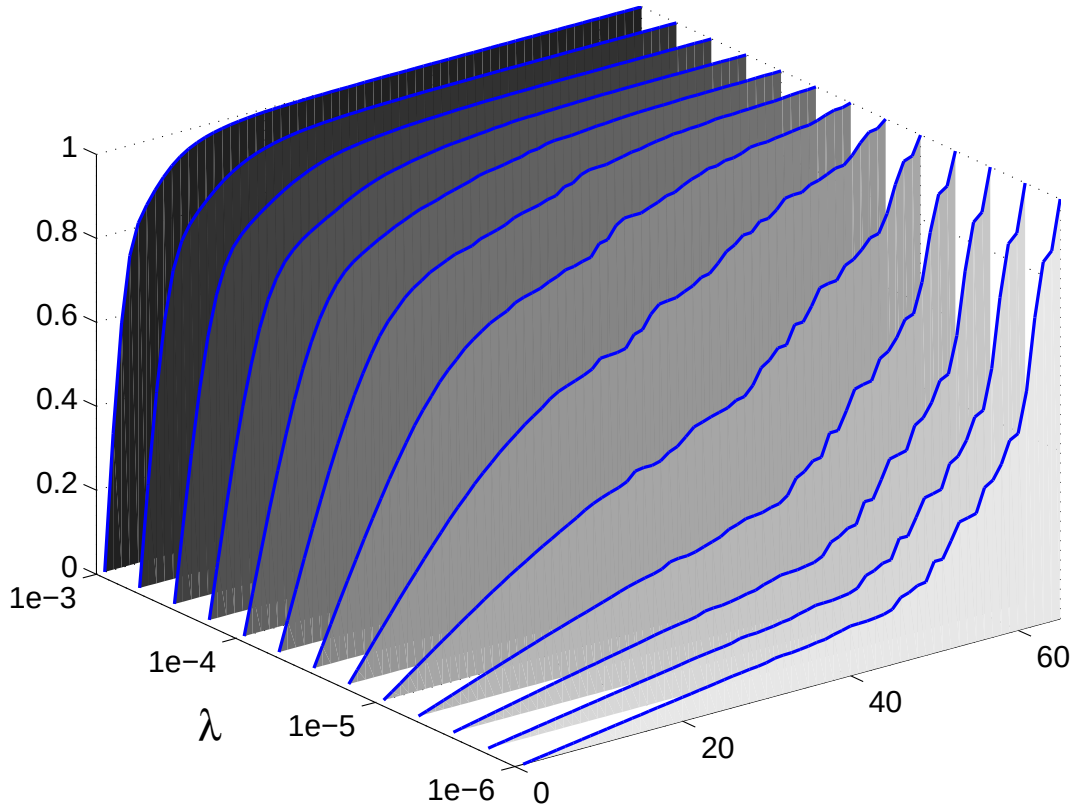
NCP Analysis



Left to right: 10 instances of white-noise residuals, 10 instances of residuals dominated by low-frequency components, and 10 instances of residuals dominated by high-frequency components.

The dashed lines show the Kolmogorov-Smirnoff limits $\pm 1.35\, q^{-1/2} \approx \pm 0.12$ for a 5% significance level, with $q = n/2 - 1$.

The Transition of the NCPs



Plots of NCPs for various regularization parameters λ, for the test problem `deriv2(128,2)` with rel. noise level $\|e\|_2/\|b^{\text{exact}}\|_2 = 10^{-5}$.

Implementation of NCP Criterion

Two ways to implement a pragmatic NCP criterion.

- Adjust the regularization parameter until the NCP lies solely within the K-S limits.
- Choose the regularization parameter for which the NCP is closest to a straight line $c_{\text{white}} = (1/q, 2/q, \ldots, 1)^T$.

The latter is implemented in Regularization Tools.

Summary of Methods (Tikhonov)

Discrepancy principle (`discrep`):

$$\text{Choose } \lambda = \lambda_{\mathrm{DP}} \text{ such that } \|A\,x_\lambda - b\|_2 = \nu_{\mathrm{dp}}\|e\|_2.$$

L-curve criterion (`l_curve`):

$$\text{Choose } \lambda = \lambda_{\mathrm{L}} \text{ such that the curvature } \hat{c}_\lambda \text{ is maximum.}$$

GCV criterion (`gcv`):

$$\text{Choose } \lambda = \lambda_{\mathrm{GCV}} \text{ as the minimizer of } G(\lambda) = \frac{\|A\,x_\lambda - b\|_2^2}{\left(m - \sum_{i=1}^{n} \varphi_i^{[\lambda]}\right)^2}.$$

NCP criterion (`ncp`):

$$\text{Choose } \lambda = \lambda_{\mathrm{NCP}} \text{ as the minimizer of } d(\lambda) = \|c(r_\lambda) - c_{\mathrm{white}}\|_2.$$

Comparison of Methods

To evaluate the performance of the four methods, we need the optimal regularization parameter λ_{opt}:

$$\lambda_{\text{opt}} = \text{argmin}_\lambda \|x^{\text{exact}} - x_\lambda\|_2.$$

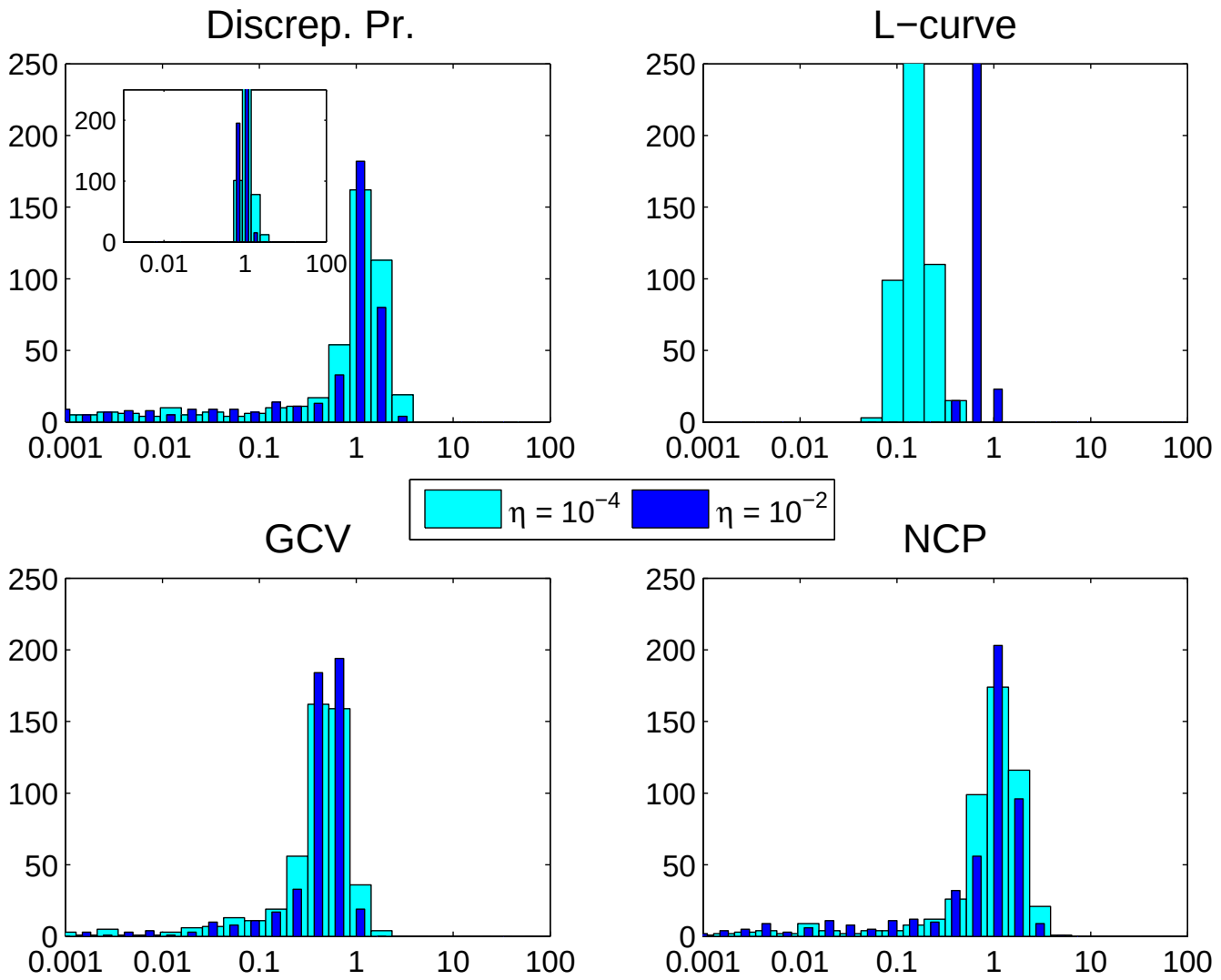
This allows us to compute the four ratios

$$R_{\text{DP}} = \frac{\lambda_{\text{DP}}}{\lambda_{\text{opt}}}, \qquad R_{\text{L}} = \frac{\lambda_{\text{L}}}{\lambda_{\text{opt}}}, \qquad R_{\text{GCV}} = \frac{\lambda_{\text{GCV}}}{\lambda_{\text{opt}}}, \qquad R_{\text{NCP}} = \frac{\lambda_{\text{NCP}}}{\lambda_{\text{opt}}},$$

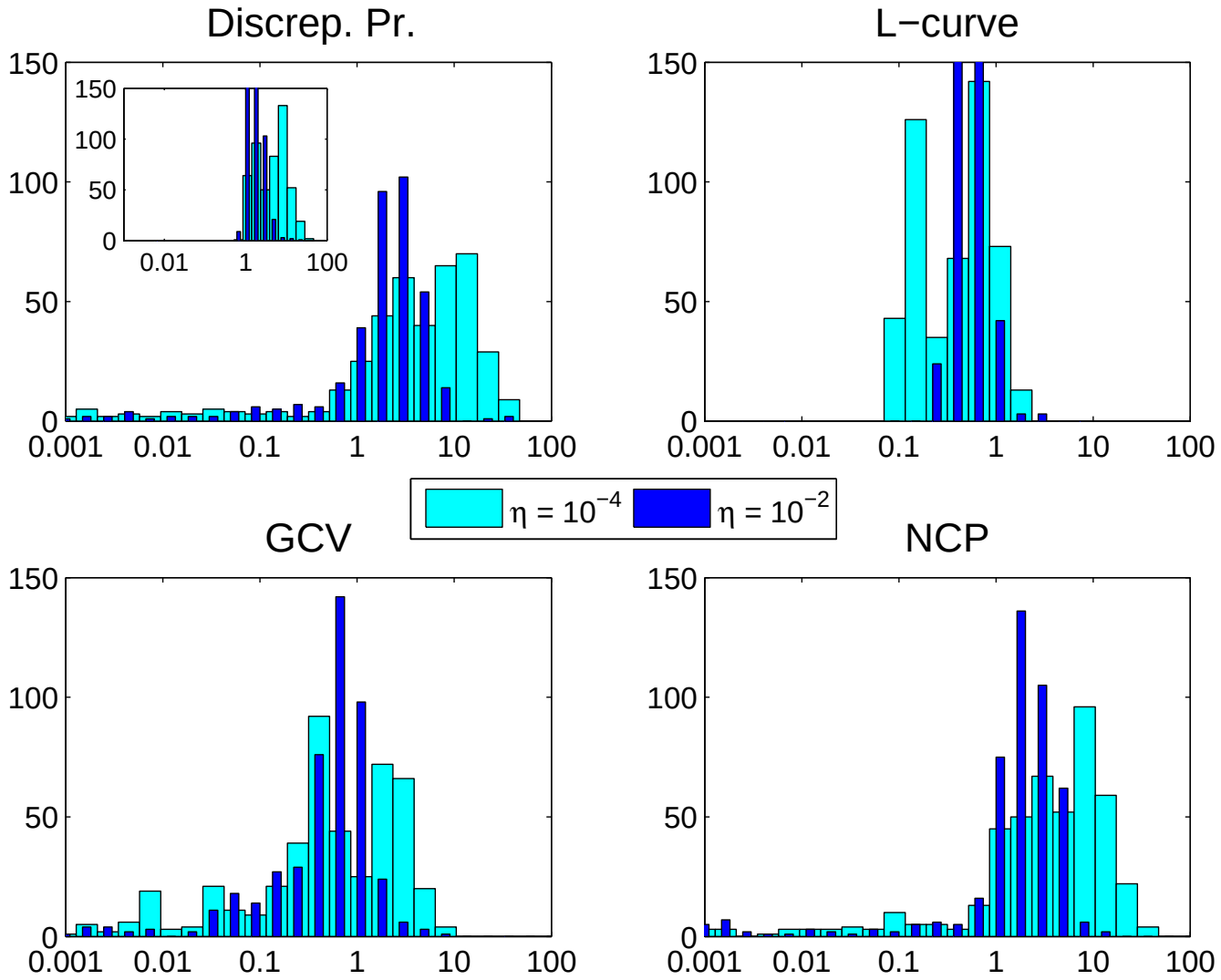
one for each parameter-choice method, and study their distributions via plots of their histograms (in log scale).

The closer these ratios are to one, the better, so a spiked histogram located at one is preferable.

First Example: `gravity`



Second Example: `shaw`



Summary of Parameter-Choice Methods

- The *discrepancy principle* is a simple method that seeks to reveal when the residual vector is noise-only. It relies on a good estimate of $\|e\|_2$ which may be difficult to obtain in practise.
- The *L-curve criterion* is based on an intuitive heuristic and seeks to balance the two error components via inspection (manually or automated) of the L-curve. This method fails when the solution is very smooth.
- The *GCV criterion* seeks to minimize the prediction error, and it is often a very robust method – with occasional failure, often leading to ridiculous under-smoothing that reveals itself.
- The NCP criterion is a statistically-based method for revealing when the residual vector is noise-only, based on the power spectrum. It can mistake LF noise for signal and thus lead to under-smoothing.