

Discrete ill-posed problems

Gérard MEURANT

October, 2008

- 1 Introduction to ill-posed problems
- 2 Tikhonov regularization
- 3 The L-curve criterion
- 4 Generalized cross-validation
- 5 Comparisons of methods

Introduction to ill-posed problems

We speak of a discrete ill-posed problem (DIP) when the solution is sensitive to perturbations of the data

Example:

$$A = \begin{pmatrix} 0.15 & 0.1 \\ 0.16 & 0.1 \\ 2.02 & 1.3 \end{pmatrix}, \quad c + \Delta c = A \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 0.01 \\ -0.032 \\ 0.01 \end{pmatrix}$$

The solution of the perturbed least squares problem (rounded to 4 decimals) using the QR factorization of A is

$$x_{QR} = \begin{pmatrix} -2.9977 \\ 7.2179 \end{pmatrix}$$

Why is it so?

The SVD of A is

$$U = \begin{pmatrix} -0.0746 & 0.7588 & -0.6470 \\ -0.0781 & -0.6513 & -0.7548 \\ -0.9942 & -0.0058 & 0.1078 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 2.4163 & 0 \\ 0 & 0.0038 \\ 0 & 0 \end{pmatrix}$$

$$V = \begin{pmatrix} -0.8409 & -0.5412 \\ -0.5412 & 0.8409 \end{pmatrix}$$

The component $(u^2)^T \Delta c / \sigma_2$ (u^2 being the second column of U) corresponding to the smallest nonzero singular value is large being 6.2161

This gives the large change in the solution

$$Ax \approx c = \bar{c} - e$$

where A is a matrix of dimension $m \times n$, $m \geq n$ and the right hand side \bar{c} is contaminated by a (generally) unknown noise vector e

- ▶ The standard solution of the least squares problem $\min \|c - Ax\|$ (even using backward stable methods like QR) may give a vector x severely contaminated by noise
- ▶ This may seem hopeless
- ▶ The solution is to modify the problem by regularization
- ▶ We have to find a balance between obtaining a problem that we can solve reliably and obtaining a solution which is not too far from the solution without noise

Tikhonov regularization

Replace the LS problem by

$$\min_x \{ \|c - Ax\|^2 + \mu \|x\|^2 \}$$

where $\mu \geq 0$ is a regularization parameter to be chosen
For some problems (particularly in image restoration) it is better to consider

$$\min_x \{ \|c - Ax\|^2 + \mu \|Lx\|^2 \}$$

where L is typically the discretization of a derivative operator of first or second order

The solution x_μ of the problem solves the linear system

$$(A^T A + \mu I)x = A^T c$$

The main problem is to choose μ

- ▶ If μ is too small the solution is contaminated by the noise in the right hand side
- ▶ if μ is too large the solution is a poor approximation of the original problem
- ▶ Many methods have been devised for choosing μ
- ▶ Most of these methods lead to the evaluation of bilinear forms with different matrices

Some methods for choosing μ

- ▶ **Morozov's** discrepancy principle

Ask for the norm of the residual to be equal to the norm of the noise vector

$$\|c - A(A^T A + \mu I)^{-1} A^T c\| = \|e\|$$

- ▶ The **Gfrerer/Raus** method

$$\mu^3 c^T (A A^T + \mu I)^{-3} c = \|e\|^2$$

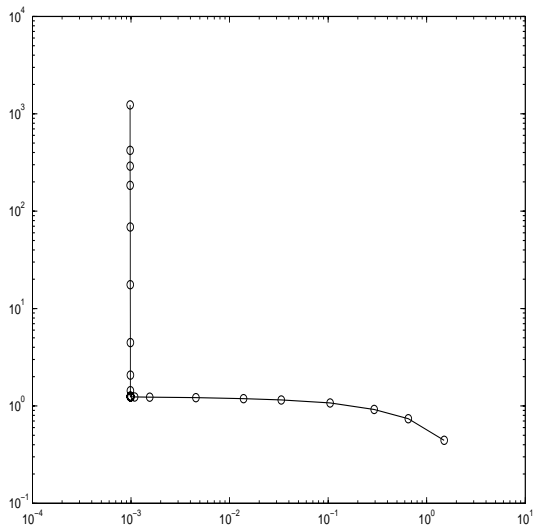
- ▶ The quasi-optimality criterion

$$\min[\mu^2 c^T A (A^T A + \mu I)^{-4} A^T c]$$

The L-curve criterion

- ▶ plot the curve $(\|x_\mu\|, \|b - Ax_\mu\|)$ obtained by varying the value of $\mu \in [0, \infty)$ in log-log scale
- ▶ In most cases this curve is shaped as an “L”
- ▶ **Lawson and Hanson** proposed to choose the value μ_L corresponding to the “corner” of the L-curve (the point of maximal curvature (see also **Hansen; Hansen and O’Leary**))
- ▶ This is done to have a balance between μ being too small and the solution contaminated by the noise, and μ being too large giving a poor approximation of the solution. The “vertex” of the L-curve gives an average value between these two extremes

An example of L-curve



The L-curve for the Baart problem, $m = n = 100$, $noise = 10^{-3}$

How to locate the corner of the L-curve?

see [Hansen](#) and al.

- ▶ Easy if we know the SVD of A
- ▶ Otherwise compute points on the L-curve and use interpolation
- ▶ However, computing a point on the L-curve is expensive
- ▶ Alternative, L-ribbon approximation ([Calvetti](#), [Golub](#) and [Reichel](#))

The L-ribbon

$$\|x_\mu\|^2 = c^T A(A^T A + \mu I)^{-2} A^T c$$

and

$$\|c - Ax_\mu\|^2 = c^T c + c^T A(A^T A + \mu I)^{-1} A^T A(A^T A + \mu I)^{-1} A^T c - 2c^T A(A^T A + \mu I)^{-1} A^T c$$

By denoting $K = A^T A$ and $d = A^T c$

$$\|c - Ax_\mu\|^2 = c^T c + d^T K(K + \mu I)^{-2} d - 2d^T (K + \mu I)^{-1} d$$

Define

$$\phi_1(t) = (t + \mu)^{-2}$$

$$\phi_2(t) = t(t + \mu)^{-2} - 2(t + \mu)^{-1}$$

we are interested in $s_i = d^T \phi_i(K)d$, $i = 1, 2$

We can obtain bounds using the **Golub–Kahan** bidiagonalization algorithm

At iteration k , the algorithm computes a **Jacobi** matrix $J_k = B_k^T B_k$ and the Gauss rule gives

$$I_k^G(\phi_i) = \|d\|^2 (e^1)^T \phi_i(J_k) e^1$$

We can also use the Gauss–Radau rule with a prescribed node $a = 0$

$$I_k^{GR}(\phi_i) = \|d\|^2 (e^1)^T \phi_i(\hat{J}_k) e^1$$

$\hat{J}_k = \hat{B}_k^T \hat{B}_k$ where \hat{B}_k is obtained from B_k by setting the last diagonal element $\delta_k = 0$

Theorem

$$I_k^G(\phi_1) \leq s_1 \leq I_k^{GR}(\phi_1)$$

where

$$\begin{aligned} I_k^G(\phi_1) &= \|d\|^2 (e^1)^T (B_k^T B_k + \mu I)^{-2} e^1 \\ I_k^{GR}(\phi_1) &= \|d\|^2 (e^1)^T (\hat{B}_k^T \hat{B}_k + \mu I)^{-2} e^1 \end{aligned}$$

$$I_k^{GR}(\phi_2) \leq s_2 \leq I_k^G(\phi_2)$$

where

$$\begin{aligned} I_k^G(\phi_2) &= \|d\|^2 [(e^1)^T B_k^T B_k (B_k^T B_k + \mu I)^{-2} e^1 - 2(e^1)^T (B_k^T B_k + \mu I)^{-1} e^1] \\ I_k^{GR}(\phi_2) &= \|d\|^2 [(e^1)^T \hat{B}_k^T \hat{B}_k (\hat{B}_k^T \hat{B}_k + \mu I)^{-2} e^1 - 2(e^1)^T (\hat{B}_k^T \hat{B}_k + \mu I)^{-1} e^1] \end{aligned}$$

$$x^-(\mu) = \sqrt{I_k^G(\phi_1)}, \quad x^+(\mu) = \sqrt{I_k^{GR}(\phi_1)}$$

$$y^-(\mu) = \sqrt{c^T c + I_k^{GR}(\phi_2)}, \quad y^+(\mu) = \sqrt{c^T c + I_k^G(\phi_2)}$$

For a given value of $\mu > 0$ the bounds are

$$x^-(\mu) \leq \|x_\mu\| \leq x^+(\mu), \quad y^-(\mu) \leq \|c - Ax_\mu\| \leq y^+(\mu)$$

Calvetti, Golub and Reichel defined the L-ribbon as the union of rectangles for all $\mu > 0$

$$\bigcup_{\mu > 0} \{ \{x(\mu), y(\mu)\} : x^-(\mu) \leq x(\mu) \leq x^+(\mu), y^-(\mu) \leq y(\mu) \leq y^+(\mu) \}$$

Then, we have to select a point (a value of μ) inside the L-ribbon

The L-curvature

Another possibility is to obtain bounds of the curvature (in log-log scale) and to look for the maximum

$$C_\mu = 2 \frac{\rho''\eta' - \rho'\eta''}{((\rho')^2 + (\eta')^2)^{3/2}}$$

where $'$ denotes differentiation with respect to μ and

$$\rho(\mu) = \frac{1}{2} \log \|c - Ax_\mu\| = \log \mu^2 c^T \phi(AA^T)c$$

$$\eta(\mu) = \frac{1}{2} \log \|x_\mu\| = \log c^T A \phi(A^T A) A^T c$$

where $\phi(t) = (t + \mu)^{-2}$

The first derivatives can be computed as

$$\begin{aligned}\rho'(\mu) &= \frac{c^T A(A^T A + \mu I)^{-3} A^T c}{\mu c^T (A A^T + \mu I)^{-2} c} \\ \eta'(\mu) &= -\frac{c^T A(A^T A + \mu I)^{-3} A^T c}{c^T A(A^T A + \mu I)^{-2} A^T c}\end{aligned}$$

The numerator is more complicated

$$\begin{aligned}\rho' \eta'' - \rho'' \eta' &= \left(\frac{c^T A(A^T A + \mu I)^{-3} A^T c}{\mu c^T (A A^T + \mu I)^{-2} c \cdot c^T A(A^T A + \mu I)^{-2} A^T c} \right)^2 \\ &\quad (c^T (A A^T + \mu I)^{-2} c \cdot c^T A(A^T A + \mu I)^{-2} A^T c \\ &\quad + 2\mu c^T (A A^T + \mu I)^{-3} c \cdot c^T A(A^T A + \mu I)^{-2} A^T c \\ &\quad - 2\mu c^T (A A^T + \mu I)^{-2} c \cdot c^T A(A^T A + \mu I)^{-3} A^T c)\end{aligned}$$

Locating the corner of the L-curve

- ▶ Using the SVD (Hansen): 1c
- ▶ Pruning algorithm (Hansen, Jensen and Rodriguez): 1p
- ▶ Rotating the L-curve (GM): 1c1
- ▶ Finding an interval where $\log \|x_\mu\|$ and $\log \|c - Ax_\mu\|$ are almost constant (GM): 1c2

L-curve algorithms, Baart problem, $n = 100$

noise	meth	μ	$\ c - Ax\ $	$\ x - x_0\ $
10^{-3}	opt	$2.4990 \cdot 10^{-8}$	$9.8720 \cdot 10^{-4}$	$1.5080 \cdot 10^{-1}$
	lc	$4.5414 \cdot 10^{-9}$	$9.8524 \cdot 10^{-4}$	$1.6030 \cdot 10^{-1}$
	lp	$8.2364 \cdot 10^{-9}$	$9.8545 \cdot 10^{-4}$	$1.5454 \cdot 10^{-1}$
	lc1	$6.3232 \cdot 10^{-9}$	$9.8534 \cdot 10^{-4}$	$1.5669 \cdot 10^{-1}$
	lc2	$5.8203 \cdot 10^{-12}$	$9.8463 \cdot 10^{-4}$	$4.1492 \cdot 10^{-1}$
			$4.1297 \cdot 10^{-8}$	$9.8996 \cdot 10^{-4}$

L-curve algorithms, Phillips problem, $n = 200$

noise	meth	μ	$\ c - Ax\ $	$\ x - x_0\ $
10^{-3}	opt	$8.5392 \cdot 10^{-7}$	$9.9864 \cdot 10^{-4}$	$7.3711 \cdot 10^{-3}$
	lc	$7.1966 \cdot 10^{-10}$	$8.5111 \cdot 10^{-4}$	$5.3762 \cdot 10^{-1}$
	lp	$4.5729 \cdot 10^{-10}$	$8.3869 \cdot 10^{-4}$	$6.8849 \cdot 10^{-1}$
	lc1	$3.6084 \cdot 10^{-10}$	$8.3172 \cdot 10^{-4}$	$7.8603 \cdot 10^{-1}$
	lc2	$1.0250 \cdot 10^{-9}$	$8.6013 \cdot 10^{-4}$	$4.4563 \cdot 10^{-1}$
			$2.9147 \cdot 10^{-7}$	$9.7098 \cdot 10^{-4}$

L-ribbon

Ex	noise	nb it	μ	nb it no reorth.
Baart	10^{-7}	11	$6.0889 \cdot 10^{-17}$	40
	10^{-5}	9	$6.1717 \cdot 10^{-13}$	19
	10^{-3}	8	$6.3232 \cdot 10^{-9}$	10
	10^{-1}	6	$7.2928 \cdot 10^{-5}$	6
	10	5	$3.260 \cdot 10^{-2}$	5

Generalized cross-validation

GCV comes from statistics (Golub, Heath and Wahba)

The regularized problem is written as

$$\min\{\|c - Ax\|^2 + m\mu\|x\|^2\}$$

where $\mu \geq 0$ is the regularization parameter and the matrix A is m by n

The GCV estimate of the parameter μ is the minimizer of

$$G(\mu) = \frac{\frac{1}{m}\|(I - A(A^T A + m\mu I)^{-1}A^T)c\|^2}{\left(\frac{1}{m}\text{tr}(I - A(A^T A + m\mu I)^{-1}A^T)\right)^2}$$

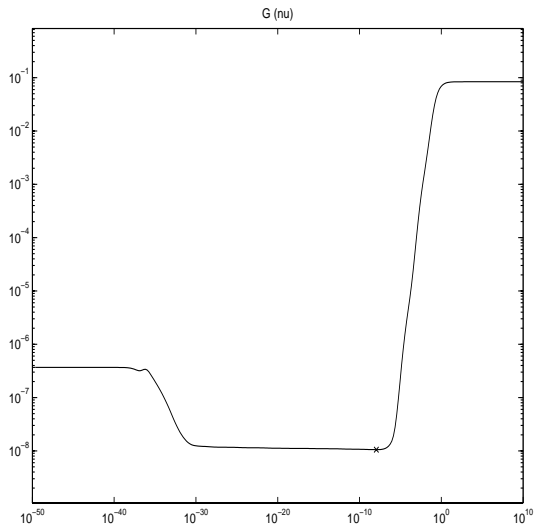
If we know the SVD of A and $m \geq n$ this can be computed as

$$G(\nu) = \frac{m \left\{ \sum_{i=1}^r d_i^2 \left(\frac{\nu}{\sigma_i^2 + \nu} \right)^2 + \sum_{i=r+1}^m d_i^2 \right\}}{\left[m - n + \sum_{i=1}^r \frac{\nu}{\sigma_i^2 + \nu} \right]^2}$$

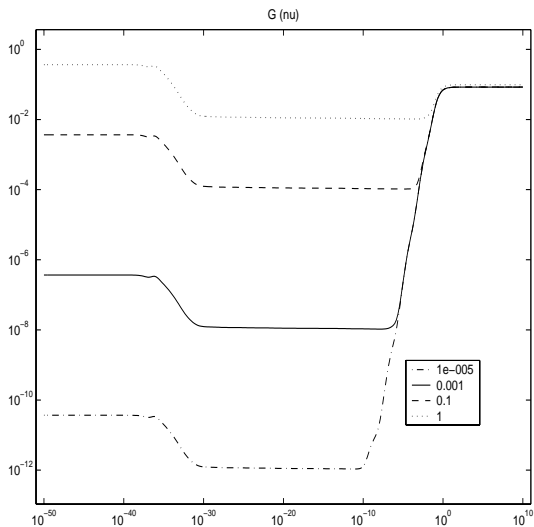
where $\nu = m\mu$

- ▶ G is almost constant when ν is very small or large, at least in log-log scale
- ▶ When $\nu \rightarrow \infty$, $G(\nu) \rightarrow \|c\|^2/m$
- ▶ When $\nu \rightarrow 0$ the situation is different whether $m = n$ or not

An example of GCV function



GCV function for the Baart problem, $m = n = 100$, $\text{noise} = 10^{-3}$



GCV functions for the Baart problem, $m = n = 100$ for different noise levels

The main problem is that the GCV function is usually quite flat near the minimum

For large problems we cannot use the SVD

- ▶ First we approximate the trace in the denominator $\rightarrow \tilde{G}$
- ▶ Then using the **Golub–Kahan** bidiagonalization algorithms we can obtain bounds of all the terms in \tilde{G}
- ▶ Finally we have to locate the minimum of the lower and/or upper bounds

Proposition

Let B be a symmetric matrix of order n with $\text{tr}(B) \neq 0$

Let \mathcal{Z} be a discrete random variable with values 1 and -1 with equal probability 0.5 and let \mathbf{z} be a vector of n independent samples from \mathcal{Z}

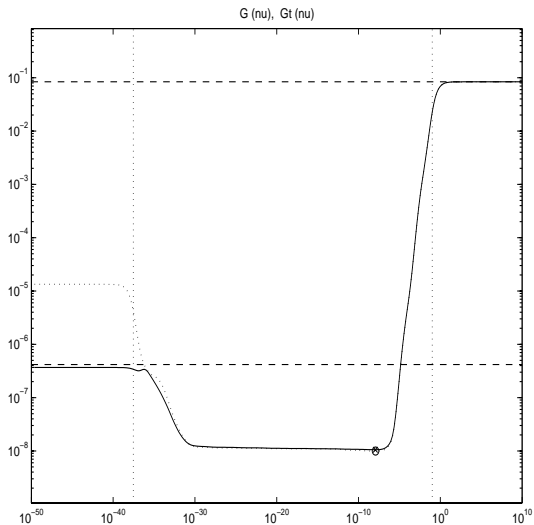
Then $\mathbf{z}^T B \mathbf{z}$ is an unbiased estimator of $\text{tr}(B)$

$$E(\mathbf{z}^T B \mathbf{z}) = \text{tr}(B)$$

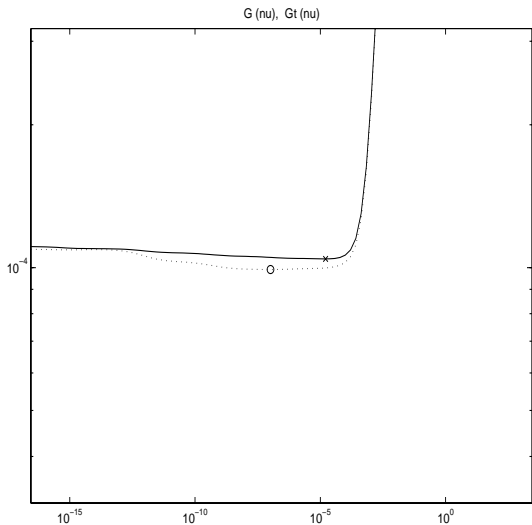
$$\text{var}(\mathbf{z}^T B \mathbf{z}) = 2 \sum_{i \neq j} b_{i,j}^2$$

where $E(\cdot)$ denotes the expected value and var denotes the variance

For GCV we just use one vector \mathbf{z}



G (plain) and \tilde{G} (dotted) functions for the Baart problem,
 $m = n = 100$, $\text{noise} = 10^{-3}$



G (plain) and \tilde{G} (dotted) functions for the Baart problem,
 $m = n = 100, \text{noise} = 10^{-1}$

The Golub and Von Matt algorithm

Let $s_z(\nu) = z^T (A^T A + \nu I)^{-1} z$, where z is a random vector
Using Gauss and Gauss–Radau we can obtain

$$g_z(\nu) \leq s_z(\nu) \leq r_z(\nu)$$

We can also bound $s_c^{(p)}(\nu) = c^T A (A^T A + \nu I)^{-p} A^T c$, $p = 1, 2$
satisfying

$$g_c^{(p)}(\nu) \leq s_c^{(p)}(\nu) \leq r_c^{(p)}(\nu)$$

We want to compute approximations of

$$\tilde{G}(\mu) = m \frac{c^T c - s_c^{(-1)}(\nu) - \nu s_c^{(-2)}(\nu)}{(m - n + \nu s_z(\nu))^2}$$

We define

$$L_0(\nu) = m \frac{c^T c - r_c^{(-1)}(\nu) - \nu r_c^{(-2)}(\nu)}{(m - n + \nu r_z(\nu))^2}$$

$$U_0(\nu) = m \frac{c^T c - g_c^{(-1)}(\nu) - \nu g_c^{(-2)}(\nu)}{(m - n + \nu g_z(\nu))^2}$$

These quantities L_0 and U_0 are lower and upper bounds for the estimate of $G(\mu)$

We can also compute estimates of the derivatives of L_0 and U_0

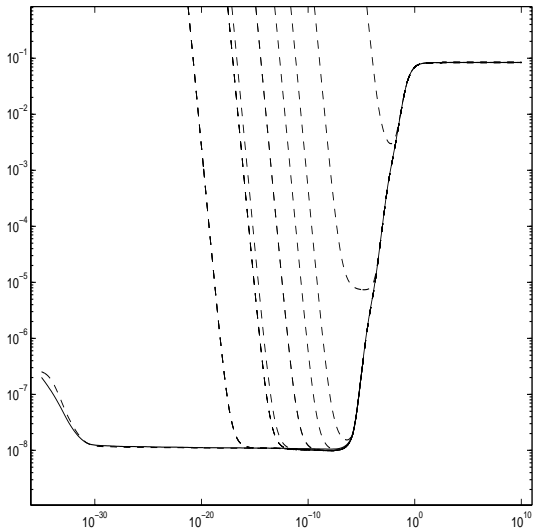
These bounds improve with the number of Lanczos iterations

- ▶ They first do $k_{min} = \lceil 3 \log \min(m, n) \rceil$ Lanczos iterations
- ▶ Then the global minimizer $\hat{\nu}$ of $U_0(\nu)$ is computed
- ▶ If one can find a ν such that $0 < \nu < \hat{\nu}$ and $L_0(\nu) > L_0(\hat{\nu})$, the algorithm stops and return $\hat{\nu}$
- ▶ Otherwise, the algorithm executes one more Lanczos iteration and repeats the convergence test

Von Matt computed the minimum of the upper bound:

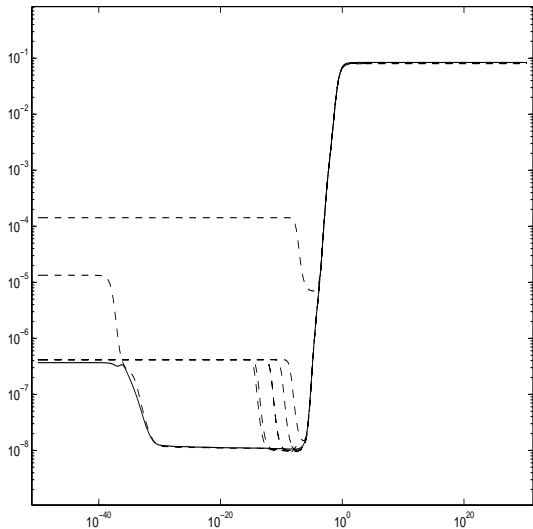
- ▶ By sampling the function on 100 points with an exponential distribution
- ▶ If the neighbors of the minimum do not have the same values, he looked at the derivative and sought for a local minimum in either the left or right interval depending on the sign of the derivative
- ▶ The local minimum is found by using bisection

The upper bound does not have the right asymptotic behavior when $m = n$ and $\nu \rightarrow 0$



G (plain) and \tilde{G} (dashed) functions and upper bounds for the Baart problem, $m = n = 100$, noise = 10^{-3}

To obtain a better behavior we add a term $\|c\|^2$ to the denominator



G (plain) and \tilde{G} (dashed) functions and upper bounds for the Baart problem, $m = n = 100$, $\text{noise} = 10^{-3}$

Optimization of the algorithm

- ▶ We choose a (small) value of ν (denoted as ν_0)
- ▶ When

$$\left| \frac{U_k^0(\nu_0) - U_{k-1}^0(\nu_0)}{U_{k-1}^0(\nu_0)} \right| \leq \epsilon_0$$

we start computing the minimum of the upper bound

The algorithm for finding the minimum is modified as follows

- ▶ We work in log–log scale and compute only a minimizer of the upper bound
- ▶ We evaluate the numerator of the approximation by computing the SVD of B_k once per iteration
- ▶ We compute 50 samples of the function on a regular mesh
- ▶ We locate the minimum, say the point k , we then compute again 50 samples in the interval $[k - 1, k + 1]$

- ▶ We use the Von Matt algorithm for computing a local minimum in this interval
- ▶ After locating a minimum ν_k with a value of the upper bound U_k^0 at iteration k , the stopping criteria is

$$\left| \frac{\nu_k - \nu_{k-1}}{\nu_{k-1}} \right| + \left| \frac{U_k^0 - U_{k-1}^0}{U_{k-1}^0} \right| \leq \epsilon$$

GCV algorithms, Baart problem

	noise	μ	$\ c - Ax\ $	$\ x - x_0\ $	t (s)
vm	10^{-7}	$9.6482 \cdot 10^{-15}$	$9.8049 \cdot 10^{-8}$	$5.9424 \cdot 10^{-2}$	0.38
	10^{-5}	$9.7587 \cdot 10^{-12}$	$9.8566 \cdot 10^{-6}$	$6.5951 \cdot 10^{-2}$	0.18
	10^{-3}	$1.2018 \cdot 10^{-8}$	$9.8573 \cdot 10^{-4}$	$1.5239 \cdot 10^{-1}$	0.16
	10^{-1}	$1.0336 \cdot 10^{-7}$	$9.8730 \cdot 10^{-2}$	1.6614	—
gm-opt	10^{-7}	$1.0706 \cdot 10^{-14}$	$9.8058 \cdot 10^{-8}$	$5.9519 \cdot 10^{-2}$	0.18
	10^{-5}	$1.0581 \cdot 10^{-11}$	$9.8588 \cdot 10^{-6}$	$6.5957 \cdot 10^{-2}$	0.27
	10^{-3}	$1.3077 \cdot 10^{-8}$	$9.8582 \cdot 10^{-4}$	$1.5205 \cdot 10^{-1}$	0.14
	10^{-1}	$1.1104 \cdot 10^{-7}$	$9.8736 \cdot 10^{-2}$	1.6227	—

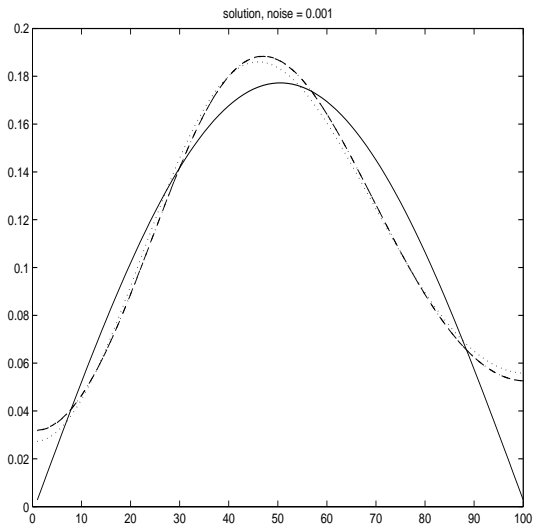
GCV algorithms, Phillips problem

	noise	μ	$\ c - Ax\ $	$\ x - x_0\ $	t (s)
vm	10^{-7}	$8.7929 \cdot 10^{-11}$	$9.0162 \cdot 10^{-8}$	$2.2391 \cdot 10^{-4}$	29.50
	10^{-5}	$4.5432 \cdot 10^{-9}$	$9.0825 \cdot 10^{-6}$	$2.2620 \cdot 10^{-3}$	6.09
	10^{-3}	$4.3674 \cdot 10^{-7}$	$9.7826 \cdot 10^{-4}$	$1.0057 \cdot 10^{-2}$	1.14
	10^{-1}	$3.8320 \cdot 10^{-5}$	$9.8962 \cdot 10^{-2}$	$9.3139 \cdot 10^{-2}$	0.16
gm-opt	10^{-7}	$1.6343 \cdot 10^{-10}$	$1.1260 \cdot 10^{-7}$	$2.2163 \cdot 10^{-4}$	15.30
	10^{-5}	$5.3835 \cdot 10^{-9}$	$9.1722 \cdot 10^{-6}$	$2.1174 \cdot 10^{-3}$	6.09
	10^{-3}	$4.1814 \cdot 10^{-7}$	$9.7737 \cdot 10^{-4}$	$1.0375 \cdot 10^{-2}$	0.66
	10^{-1}	$4.1875 \cdot 10^{-5}$	$9.9016 \cdot 10^{-2}$	$9.0659 \cdot 10^{-2}$	0.22

Comparisons of methods

Baart problem, $n = 100$






noise	meth	μ	$\ c - Ax\ $	$\ x - x_0\ $
10^{-3}	μ opt	$2.7826 \cdot 10^{-8}$	$2.3501 \cdot 10^{-3}$	$1.5084 \cdot 10^{-1}$
	vm	$1.2018 \cdot 10^{-8}$	$9.8573 \cdot 10^{-4}$	$1.5239 \cdot 10^{-1}$
	gm-opt	$1.3077 \cdot 10^{-8}$	$9.8582 \cdot 10^{-4}$	$1.5205 \cdot 10^{-1}$
	gcv	$9.4870 \cdot 10^{-9}$	$9.8554 \cdot 10^{-4}$	$1.5362 \cdot 10^{-1}$
	disc	$8.4260 \cdot 10^{-8}$	$1.0000 \cdot 10^{-3}$	$1.5556 \cdot 10^{-1}$
	gr	$1.7047 \cdot 10^{-7}$	$1.0235 \cdot 10^{-3}$	$1.6373 \cdot 10^{-1}$
	lc	$4.5414 \cdot 10^{-9}$	$9.8524 \cdot 10^{-4}$	$1.6028 \cdot 10^{-1}$
	qo	$1.2586 \cdot 10^{-8}$	$9.8450 \cdot 10^{-4}$	$6.6072 \cdot 10^{-1}$
	L-rib	$6.3232 \cdot 10^{-9}$	$9.8534 \cdot 10^{-4}$	$1.5669 \cdot 10^{-1}$
	L-cur	$5.8220 \cdot 10^{-9}$	$9.8531 \cdot 10^{-4}$	$1.5749 \cdot 10^{-1}$













Solutions for the Baart problem, $m = n = 100$, $noise = 10^{-3}$,
solid=unperturbed solution, dashed=vm, dot-dashed=gm-opt

Phillips problem, $n = 200$

noise	meth	μ	$\ c - Ax\ $	$\ x - x_0\ $
10^{-5}	μ opt	$1.3725 \cdot 10^{-7}$	$2.9505 \cdot 10^{-14}$	$1.6641 \cdot 10^{-3}$
	vm	$4.5432 \cdot 10^{-9}$	$9.0825 \cdot 10^{-6}$	$2.2620 \cdot 10^{-3}$
	gm-opt	$5.3835 \cdot 10^{-9}$	$9.1722 \cdot 10^{-6}$	$2.1174 \cdot 10^{-3}$
	gcv	$3.1203 \cdot 10^{-9}$	$8.9283 \cdot 10^{-6}$	$2.6499 \cdot 10^{-3}$
	disc	$1.2107 \cdot 10^{-8}$	$1.0000 \cdot 10^{-5}$	$1.6873 \cdot 10^{-3}$
	gr	$4.1876 \cdot 10^{-8}$	$1.5784 \cdot 10^{-5}$	$1.9344 \cdot 10^{-3}$
	lc	$3.6731 \cdot 10^{-14}$	$2.4301 \cdot 10^{-6}$	$7.9811 \cdot 10^{-1}$
	qo	$1.5710 \cdot 10^{-8}$	$1.0542 \cdot 10^{-5}$	$1.6463 \cdot 10^{-3}$
	L-rib	$2.6269 \cdot 10^{-14}$	$2.2118 \cdot 10^{-6}$	$8.9457 \cdot 10^{-1}$
	L-cur	$4.7952 \cdot 10^{-14}$	$2.6093 \cdot 10^{-6}$	$7.2750 \cdot 10^{-1}$

-  D. CALVETTI, G.H. GOLUB AND L. REICHEL, *Estimation of the L-curve via Lanczos bidiagonalization*, BIT, v 39 n 4, (1999), pp 603–619
-  D. CALVETTI, P.C. HANSEN AND L. REICHEL, *L-curve curvature bounds via Lanczos bidiagonalization*, Elec. Trans. Numer. Anal., v 14, (2002), pp 20–35
-  H. GFRENERER, *An a posteriori parameter choice for ordinary and iterated Tikhonov regularization of ill-posed problems leading to optimal convergence rates*, Math. Comp., v 49, (1987), pp 507–522
-  G.H. GOLUB, M. HEATH AND G. WAHBA, *Generalized cross-validation as a method to choosing a good ridge parameter*, Technometrics, v 21 n 2, (1979), pp 215–223
-  G.H. GOLUB AND U. VON MATT, *Tikhonov regularization for large scale problems*, in Scientific Computing, G.H. Golub, S.H. Lui, F. Luk and R. Plemmons Eds., Springer, (1997), pp 3–26

-  G.H. GOLUB AND U. VON MATT, *Generalized cross-validation for large scale problems*, in Recent advances in total least squares techniques and errors in variable modeling, S. van Huffel ed., SIAM, (1997), pp 139–148
-  M. HANKE AND T. RAUS, *A general heuristic for choosing the regularization parameter in ill-posed problems*, SIAM J. Sci. Comput., v 17, (1996), pp 956–972
-  P.C. HANSEN, *Regularization tools: a Matlab package for analysis and solution of discrete ill-posed problems*, Numer. Algo., v 6, (1994), pp 1–35
-  P.C. HANSEN AND D.P. O'LEARY, *The use of the L-curve in the regularization of discrete ill-posed problems*, SIAM J. Sci. Comput., v 14, (1993), pp 1487–1503
-  P.C. HANSEN, T.K. JENSEN AND G. RODRIGUEZ, *An adaptive pruning algorithm for the discrete L-curve criterion*, J. Comp. Appl. Math., v 198 n 2, (2007), pp 483–492

-  C.L. LAWSON AND R.J. HANSON, *Solving least squares problems*, SIAM, (1995)
-  A.S. LEONOV, *On the choice of regularization parameters by means of the quasi-optimality and ratio criteria*, Soviet Math. Dokl., v 19, (1978), pp 537–540
-  V.A. MOROZOV, *Methods for solving incorrectly posed problems*, Springer, (1984)
-  A.N. TIKHONOV, *Ill-posed problems in natural sciences*, Proceedings of the international conference Moscow August 1991, (1992), TVP Science publishers.
-  A.N. TIKHONOV AND V.Y. ARSENIN, *Solutions of ill-posed problems*, (1977), Wiley