Parameter estimation in regularization models for Poisson data

L. Zanni

Department of Physics, Computer Science and Mathematics,
University of Modena and Reggio Emilia, Italy

First French-German Mathematical Image Analysis Conference

Paris, 13 - 15 January 2014

Joint work with:

M. Bertero, University of Genova, Italy
V. Ruggiero, University of Ferrara, Italy

Outline

Poisson data

- Consider imaging processes where image intensity is measured via the counting of incident particles
- Fluctuations in the emission-counting process can be described by modeling the data as realizations of Poisson random variables
- The probability of receiving n particles is given by

$$p(n) = \frac{e^{-\lambda}\lambda^n}{n!}, \qquad n = 0, 1, 2, \dots$$

  where λ is the expected value of the counts
- A statistical model appropriate for describing data in different imaging applications
  (emission tomography, fluorescence microscopy, optical/infrared astronomy, etc.)

Poisson noisy image restoration: problem setting

- $\boldsymbol{x}^* \in \mathbb{R}^n$ $\longrightarrow$ the unknown true image; $x_i^* \geq 0$
- $A \in \mathbb{R}^{n \times n}$ $\longrightarrow$ the imaging matrix representing the blurring phenomenon ($A = I$ in denoising)
- $b \in \mathbb{R}$ $\longrightarrow$ the nonnegative background radiation
- $(A\boldsymbol{x}^* + b)$ $\longrightarrow$ the image that would be recorded in absence of noise
- $\boldsymbol{y} \in \mathbb{R}^n$ $\longrightarrow$ the observed blurred noisy image; $y_i \geq 0$

$\Downarrow$

Given $A, b, \boldsymbol{y}$, determine an estimate of the true image $\boldsymbol{x}^*$

Poisson noisy image restoration: the optimization problem

Following the maximum a posteriori (MAP) approach, an estimate $\boldsymbol{x}^*_\beta$ of the unknown image can be obtained by solving

The MAP constrained optimization problem

$$\min_{\boldsymbol{x}\geq \boldsymbol{0}} \; f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x}), \qquad \beta > 0$$

- $f_0(\boldsymbol{x})$ data-fidelity function (generalized Kullback-Leibler divergence)

$$f_0(\boldsymbol{x}) = D_{KL}(\boldsymbol{y}, \boldsymbol{x}) = \sum_{i=1}^{n} \left\{ y_i \log \frac{y_i}{(A\boldsymbol{x}+b)_i} + (A\boldsymbol{x}+b)_i - y_i \right\}$$

- $f_1(\boldsymbol{x}) = f(L\boldsymbol{x})$ regularization term (L linear operator)

$$f_1(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{x}\|_2^2, \qquad f_1(\boldsymbol{x}) = \|\boldsymbol{x}\|_1, \qquad f_1(\boldsymbol{x}) = \||\nabla \boldsymbol{x}|\|_1$$

- β is the regularization parameter

Regularization parameter estimation

- Look for estimations of a regularization parameter β suitable for balancing the data fidelity with the regularity of the solution
  [Engl-Hanke-Neubauer,1996], [Bertero-Boccacci,1998]

- Focus on ideas based on the discrepancy principle: a suitable β is such that a measure of the discrepancy $D_{\boldsymbol{y}}(\beta)$ between the corrupted data $\boldsymbol{y}$ and the reconstruction $\boldsymbol{x}_\beta^*$ equals some known error τ:

$$D_{\boldsymbol{y}}(\beta) = \tau$$

- We consider

$$D_{\boldsymbol{y}}(\beta) = D_{KL}(\boldsymbol{y}, \boldsymbol{x}_\beta^*) = \sum_{i=1}^{n} \left\{ y_i \log \frac{y_i}{(A\boldsymbol{x}_\beta^* + b)_i} + (A\boldsymbol{x}_\beta^* + b)_i - y_i \right\}$$

$$A_{i,j} \geq 0, \quad \textstyle\sum_i A_{i,j} = 1, \quad \sum_j A_{i,j} > 0, \quad \forall i, j, \qquad b > 0$$

[Bardsley-Goldes,2009], [Bertero et al., 2010], [Carlavan, Blanc-Féraud, 2011,2012], [Teuber-Steidl-Chan, 2013]

Following the discrepancy principle: what is necessary?

$$D_{\boldsymbol{y}}(\beta) = \sum_{i=1}^{n} \left\{ y_i \log \frac{y_i}{(A\boldsymbol{x}^*_\beta + b)_i} + (A\boldsymbol{x}^*_\beta + b)_i - y_i \right\} = \tau$$

- Find a suitable value for the constant τ, given the Poisson data assumption and the above discrepancy function.
- Exploit effective algorithms for finding β such that $D_{\boldsymbol{y}}(\beta) = \tau$. Alternative approaches:
  - Determine β by solving directly the nonlinear equation
    [Zanella-Boccacci-Z.-Bertero 2009], [Bertero et al., 2010]
  - Determine β by solving the constrained minimization problem

$$\min_{\boldsymbol{x} \geq \boldsymbol{0}} \left\{ f_1(\boldsymbol{x}) \quad \text{sub. to} \quad D_{\boldsymbol{y}}(\beta) \leq \tau \right\}$$

[Carlavan, Blanc-Féraud, 2011,2012], [Teuber-Steidl-Chan, 2013]

A possible setting for the constant τ

Lemma ([Zanella et al., Inverse Problems 2009, 2013])

Let Y_λ be a Poisson r.v. with expected value λ and consider

$$F(Y_\lambda) = 2\left\{Y_\lambda \ln\left(\frac{Y_\lambda}{\lambda}\right) + \lambda - Y_\lambda\right\}.$$

Then the expected value of $F(Y_\lambda)$ satisfies

$$E\left\{F(Y_\lambda)\right\} = 1 + O\left(\frac{1}{\lambda}\right), \qquad \lambda \to +\infty.$$

The asymptotic estimate of the expected value of $D_{\boldsymbol{y}}(\beta)$ is $\frac{n}{2}$. Then

$$\boxed{\tau = \frac{n}{2}}$$

In [Carlavan, Blanc-Féraud, IEEE T. Image Proc. 2011] $\tau = \frac{m}{2},\; m = \#\{y_i,\, y_i > 0\}$

Find $\bar{\beta}$ such that $D_{\boldsymbol{y}}(\bar{\beta}) = \tau$

Consider the nonlinear equation

$$D_{\boldsymbol{y}}(\beta) - \tau = \sum_{i=1}^{n} \left\{ y_i \log \frac{y_i}{(A\boldsymbol{x}_\beta^* + b)_i} + (A\boldsymbol{x}_\beta^* + b)_i - y_i \right\} - \tau = 0$$

where

$$\boldsymbol{x}_\beta^* = \operatorname*{argmin}_{\boldsymbol{x} \geq 0} f_\beta(\boldsymbol{x}) = f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x})$$

and $f_0(\boldsymbol{x})$ is nonnegative, convex and coercive on $\mathbb{R}^n_{\geq 0}$. Assume

$f_1(\boldsymbol{x})$ differentiable, nonnegative, convex and such that

$$\mathcal{N}\left[\nabla^2 f_0(\boldsymbol{x})\right] \cap \mathcal{N}\left[\nabla^2 f_1(\boldsymbol{x})\right] = \{0\}$$

$\Downarrow$

- $f_\beta(\boldsymbol{x})$ is coercive and strictly convex $\Rightarrow$ $D_{\boldsymbol{y}}(\beta)$ is well-defined
- $D_{\boldsymbol{y}}(\beta)$ is an increasing function of β $\Rightarrow$ if $\bar{\beta}$ exists, it is unique

Look at an example: edge preserving regularization

$$f_1(\boldsymbol{x}) = \sum_{k=1}^{n} \sqrt{\Delta_k}$$

$$\Delta_k = (x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2 + \delta^2 = \|L_k \boldsymbol{x}\|^2 + \delta^2$$

- $(L_k)_{2\times n}$, $\quad L = [L_1^T, \dots, L_n^T]^T, \qquad E(\boldsymbol{x}) = diag\left(\Delta_k^{\frac{1}{2}} I_2\right)_{k=1,\dots,n}$
- $F(\boldsymbol{x}) = diag\left(I_2 - \frac{1}{\Delta_k} A_k \boldsymbol{x}\boldsymbol{x}^T A_k^T\right)_{k=1,\dots,n}$

$$\nabla f_1(\boldsymbol{x}) = L^T E(\boldsymbol{x}) L\boldsymbol{x}\,, \qquad \nabla^2 f_1(\boldsymbol{x}) = L^T E(\boldsymbol{x})^{-1} F(\boldsymbol{x}) L\,,$$

$$\Downarrow$$

$$\mathcal{N}\left[\nabla^2 f_1(\boldsymbol{x})\right] = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{x} = c\mathbf{1}_n,\ c \in \mathbb{R}^n\}$$

Look at an example: edge preserving regularization

Recall that

$$f_0(\boldsymbol{x}) = \sum_{i=1}^{n} \left\{ y_i \, \log \frac{y_i}{(A\boldsymbol{x}+b)_i} + (A\boldsymbol{x}+b)_i - y_i \right\}$$

- $A_{i,j} \geq 0, \quad \sum_i A_{i,j} = 1, \quad \sum_j A_{i,j} > 0, \quad \forall i,j, \qquad b > 0, \quad \boldsymbol{x} \geq 0$
- $\nabla f_0(\boldsymbol{x}) = \mathbf{1}_n - A^T \dfrac{y}{(A\boldsymbol{x}+b)}, \qquad \nabla^2 f_0(\boldsymbol{x}) = A^T \dfrac{y}{(A\boldsymbol{x}+b)^2} A$

$$\mathcal{N}\left[\nabla^2 f_0(\boldsymbol{x})\right] = \{\boldsymbol{x} \in \mathbb{R}^n \mid (A\boldsymbol{x})_i = 0, \; i \in \mathcal{I}_1\}, \qquad \mathcal{I}_1 = \{i \mid y_i > 0\}$$

⇩

$$\mathcal{N}\left[\nabla^2 f_0(\boldsymbol{x})\right] \cap \mathcal{N}\left[\nabla^2 f_1(\boldsymbol{x})\right] = \{0\}$$

➤ $f_\beta(\boldsymbol{x})$ is strictly convex and $D_{\boldsymbol{y}}(\beta)$ is an increasing function of β

Let $\quad f_1(\boldsymbol{x}) = \sum_{k=1}^{n} \sqrt{\Delta_k}$

(i) $\lim_{\beta \to 0} \boldsymbol{x}_\beta^* = \boldsymbol{x}^* , \qquad \boldsymbol{x}^* = \underset{\boldsymbol{x}^* \geq 0}{\operatorname{argmin}} f_0(\boldsymbol{x})$

(ii) If $\frac{1}{n} \sum_{j=1}^{n} (A^T \boldsymbol{y})_j > b$ then

$$\lim_{\beta \to \infty} \boldsymbol{x}_\beta^* = \bar{c}\mathbf{1}_n , \qquad \bar{c} \ : \ \sum_{i \in \mathcal{I}_1} \frac{\mathcal{A}_i y_i}{\mathcal{A}_i \bar{c} + b} = n , \qquad \mathcal{A}_i = \sum_{j=1}^{n} A_{i,j}$$

⇩

If $\frac{1}{n} \sum_{j=1}^{n} (A^T \boldsymbol{y})_j > b , \qquad f_0(\boldsymbol{x}^*) < \frac{n}{2} , \qquad f_0(\bar{c}\mathbf{1}_n) > \frac{n}{2} ,$

then $\bar{\beta}$ such that $D_{\boldsymbol{y}}(\bar{\beta}) = \tau$ exists and is unique

(iii) If $\mathcal{A}_i = 1$ then $\bar{c} = \bar{y} - b , \quad \bar{y} = \frac{1}{n} \sum_{i \in \mathcal{I}_1} y_i$

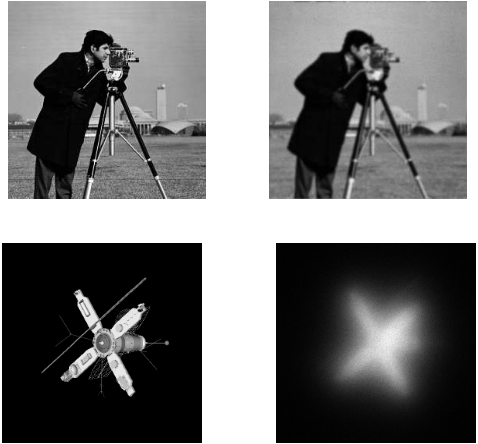
$$f_0(\bar{c}\mathbf{1}_n) > \frac{n}{2} \qquad \Leftrightarrow \qquad \frac{1}{n} \sum_{i \in \mathcal{I}_1} y_i \ln y_i > \frac{1}{2} + \bar{y} \ln \bar{y}$$

Blurred images corrupted by Poisson noise: two test problems

Test environment: Matlab 7.14.0 on a processor Intel Core i7 CPU Q720 1.60 GHz, 4GB RAM

Test problems: Cameraman (256×256), Spacecraft (256×256)

Original Image | Observed Image



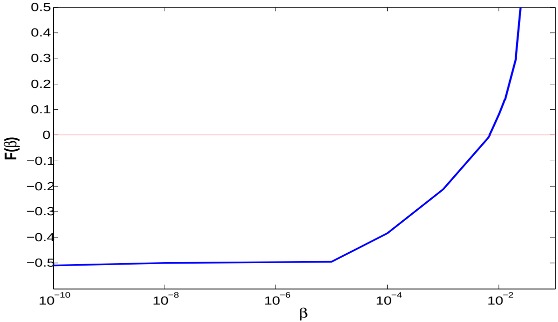
Edge preserving regularization: cameraman test problem

$$n = 256^2, \qquad \bar{c} = 1407.84$$

$$f_0(\boldsymbol{x}^*) = 16403.5 < 32768 = \frac{n}{2} < 14879957.3 = f_0(\bar{c}\mathbf{1}_n)$$

Behaviour of

$$F(\beta) = \frac{2}{n} D_{\boldsymbol{y}}(\beta) - 1$$



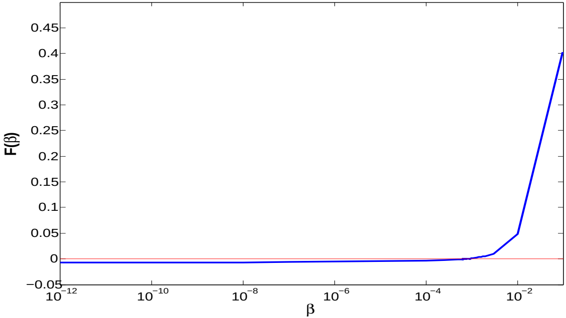
Edge preserving regularization: spacecraft test problem

$$n = 256^2, \qquad \bar{c} = 154.22$$

$$f_0(\boldsymbol{x}^*) = 32399.6 < 32768 = \frac{n}{2} < 8022676.7 = f_0(\bar{c}\mathbf{1}_n)$$

Behaviour of

$$F(\beta) = \frac{2}{n} D_{\boldsymbol{y}}(\beta) - 1$$



Solving $D_{\boldsymbol{y}}(\bar{\beta}) = \tau$: the optimization solver

- effective constrained optimization solvers for

$$\boldsymbol{x}^*_\beta = \underset{\boldsymbol{x}\geq 0}{\operatorname{argmin}}\, f_\beta(\boldsymbol{x}) = f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x})$$

  - suitable for nonnegative constraints
  - efficient for progressive stopping tolerance and warm starting
  - robust for different values of β
  - limited memory requirements

- Assuming $f_1(\boldsymbol{x})$ differentiable, recent accelerated gradient projection methods can be exploited

Scaled Gradient Projection (SGP) methods

$$\min_{\boldsymbol{x} \geq \boldsymbol{0}} \quad f(\boldsymbol{x})$$

$$\boldsymbol{x}^{(0)} \geq \boldsymbol{0}, \qquad \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \lambda_k \boldsymbol{d}^{(k)}, \qquad k = 0, 1, \ldots$$

- $\boldsymbol{d}^{(k)}$ feasible descent direction

$$\boldsymbol{d}^{(k)} = P_+ \left(\boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \nabla f(\boldsymbol{x}^{(k)}) \right) - \boldsymbol{x}^{(k)}$$

  - $\mathcal{D}_k = diag(d_1, \ldots, d_n)$, $\frac{1}{\rho} \leq d_i \leq \rho$, diagonal scaling matrix
  - $\alpha_k \in [\alpha_{min}, \alpha_{max}]$ step-length parameter

- $\lambda_k \in (0, 1]$ line-search parameter to ensure (via backtracking)

$$f(\boldsymbol{x}^{(k)} + \lambda_k \boldsymbol{d}^{(k)}) \leq f_{ref}^{(k)} + \gamma \lambda_k \nabla f(\boldsymbol{x}^{(k)})^T \boldsymbol{d}^{(k)}, \qquad \gamma \in (0, 1)$$

A basic convergence property

Assume that $\Omega_0 = \{\boldsymbol{x} \geq \mathbf{0} : f(\boldsymbol{x}) \leq f(\boldsymbol{x}^{(0)})\}$ is bounded. Every accumulation point $\boldsymbol{x}^*$ of the sequence $\{\boldsymbol{x}^{(k)}\}$ generated by SGP is a constrained stationary point:

$$\nabla f(\boldsymbol{x}^*)^T(\boldsymbol{x} - \boldsymbol{x}^*) \geq 0 \quad \forall \boldsymbol{x} \geq \mathbf{0}.$$

- E. G. Birgin, J. M. Martínez, and M. Raydan, *Nonmonotone spectral projected gradient methods on convex sets*, SIAM J. Optim. **10:4** (2000)
- E. G. Birgin, J. M. Martínez, and M. Raydan, *Inexact spectral projected gradient methods on convex sets*, IMA J. Numer. Anal. **23** (2003)
- S. Bonettini, R. Zanella, and L. Zanni, *A scaled gradient projection method for constrained image deblurring*, Inverse Problems 25 (2009), 015002

- Liu-Dai, JOTA 2001 $\longrightarrow$ R-linear convergence (unconstrained case)
- Hager-Mair-Zhang, *Math. Program.* (2009) $\longrightarrow$ R-linear convergence (constrained case)

The step-length selections: different rules but similar derivation

Suppose to have defined the diagonal scaling matrix $\mathcal{D}_k$.

Look for effective selection rules for the step-length α_k:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \lambda_k \left(P_+(\boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \nabla f(\boldsymbol{x}^{(k)})) - \boldsymbol{x}^{(k)} \right)$$

Barzilai-Borwein (BB) like selection rules [Barzilai-Borwein 1988]

Widely studied and successfully used in many applications in the last years.
Essentially based on the information from the last two iterations

$$\boldsymbol{x}^{(k)}, \;\; \boldsymbol{x}^{(k-1)}, \;\; \nabla f(\boldsymbol{x}^{(k)}), \;\; \nabla f(\boldsymbol{x}^{(k-1)})$$

Selection rules based on the Ritz values [Fletcher 2012]

Recently proposed for limited memory steepest descent methods.
The gradients of the last m it. are exploited (m small, $m = 3, 4, 5$):

$$\nabla f(\boldsymbol{x}^{(k)}), \; \ldots, \; \nabla f(\boldsymbol{x}^{(k-m+1)})$$

Derivation of the step-length selection strategies

- Consider the gradient method for the unconst. problem $\min f(\boldsymbol{x})$:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}, \qquad \boldsymbol{g}^{(k)} = \nabla f(\boldsymbol{x}^{(k)}), \qquad k = 0, 1, \dots$$

- $f(x) = \frac{1}{2} x^T A x - b^T x \quad A = diag(\lambda_1, \dots, \lambda_N), \quad 0 < \lambda_1 < \dots < \lambda_n$

$$\Downarrow$$

$$g_i^{(k+1)} = (1 - \alpha_k \lambda_i) g_i^{(k)} \qquad i = 1, \dots, n$$

- $\alpha_k = \frac{1}{\lambda_i} \quad \Rightarrow \quad g_i^{(k+1)} = 0 \quad \Rightarrow \quad g_i^{(k+j)} = 0, \quad j = 2, 3 \dots$
- $\alpha_{k+i-1} = \frac{1}{\lambda_i}, \quad i = 1, \dots, N \quad \Rightarrow \quad \boldsymbol{g}^{(k+N)} = 0$ (Finite Termination)

α_k must aim at approximating the inverse of the eigenvalues of A

Step-length selection: exploiting the BB rules

$$\color{red}{\alpha_k}^{\mathrm{BB1}} = \frac{{\boldsymbol{s}^{(k-1)}}^T \boldsymbol{s}^{(k-1)}}{{\boldsymbol{s}^{(k-1)}}^T \boldsymbol{z}^{(k-1)}}, \qquad \color{red}{\alpha_k}^{\mathrm{BB2}} = \frac{{\boldsymbol{s}^{(k-1)}}^T \boldsymbol{z}^{(k-1)}}{{\boldsymbol{z}^{(k-1)}}^T \boldsymbol{z}^{(k-1)}} \qquad \begin{array}{l} \boldsymbol{s}^{(k-1)} = \boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)} \\ \boldsymbol{z}^{(k-1)} = \boldsymbol{g}^{(k)} - \boldsymbol{g}^{(k-1)} \end{array}$$

Alternate Barzilai-Borwein selection rule [Zhou-Gao-Dai (2006)]

$$\alpha_k^{ABB} = \begin{cases} \alpha_k^{BB2} & \text{if } \frac{\alpha_k^{BB2}}{\alpha_k^{BB1}} < \tau, \qquad \tau \in (0,1) \\ \alpha_k^{BB1} & \text{otherwise} \end{cases}$$

$\mathrm{ABB_{min}}$ rule [Frassoldati-Zanghirati-Zanni (2008)]

$$\alpha_k^{ABB_{min}} = \begin{cases} \min\left\{\alpha_j^{BB2} \,|\, j = \max\{1, k - M_\alpha\}, ..., k\right\} & \text{if } \alpha_k^{BB2} / \alpha_k^{BB1} < \tau \\ \alpha_k^{BB1} & \text{otherwise} \end{cases}$$

where $M_\alpha > 0$ is a parameter.

Behaviour of the BB adaptive alternation

Example

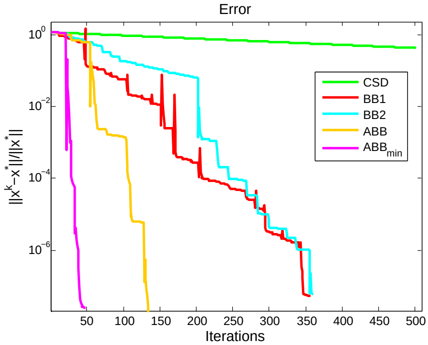
$$f(x) = \frac{1}{2}x^T Ax - b^T x$$

- $A = diag(\lambda_1, \ldots, \lambda_{10}), \quad \lambda_i = 111i - 110$
- b random vector; $b_i \in [-10, 10]$.

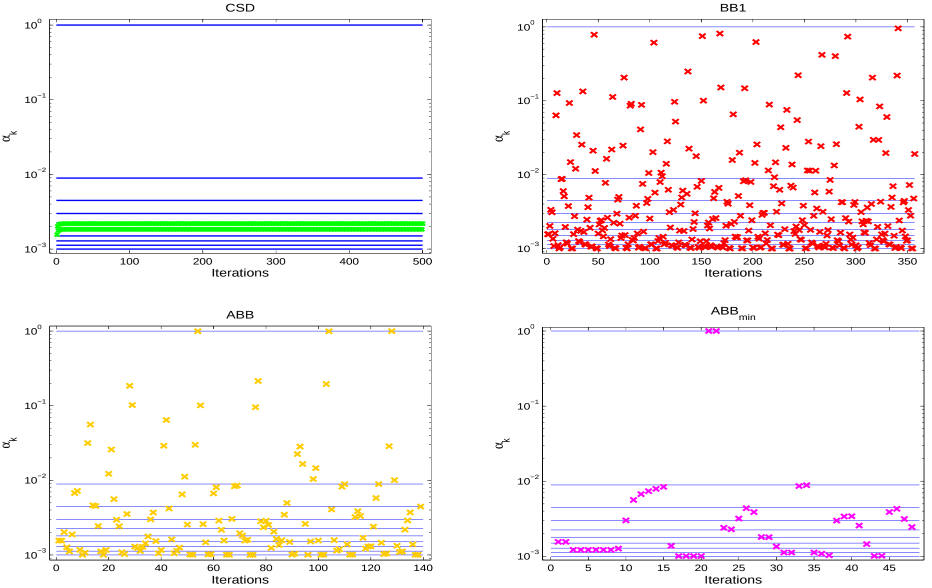
$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}$$

Error $= \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\| / \|\boldsymbol{x}^*\|$

- Cauchy Steepest Descent (CSD)
  $\alpha_k = \text{argmin}_{\alpha>0} f(\boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)})$
- BB1 $\rightarrow$ $\alpha_k = \alpha_k^{BB1}$
- BB2 $\rightarrow$ $\alpha_k = \alpha_k^{BB2}$
- ABB $\rightarrow$ alternation
- ABB_{min} $\rightarrow$ modified alternation



The distribution of the steplengths w.r.t. $\frac{1}{\lambda_i}, \quad i = 1, \dots, n$



Step-length selection: exploiting the Ritz Values

Unconstr. problem: $\min f(\boldsymbol{x})$, $f(x)=\frac{1}{2}x^T Ax - b^T x$, $\boldsymbol{g}(\boldsymbol{x})=\nabla f(\boldsymbol{x})$

Basic properties

Consider the Krylov sequence: $\{\boldsymbol{g}^{(k-m)},\ A\boldsymbol{g}^{(k-m)},\ \dots,\ A^{m-1}\boldsymbol{g}^{(k-m)}\}$

- Lanczos iterative process, starting from $\boldsymbol{q}_1 = \dfrac{\boldsymbol{g}^{(k-m)}}{\|\boldsymbol{g}^{(k-m)}\|}$, generates orthonormal basis vectors for the Krylov sequence:

$$Q_{n\times m} = [\boldsymbol{q}_1, \dots, \boldsymbol{q}_m]$$

- The eigenvalues (Ritz Values) of the tridiagonal matrix

$$T_{m\times m} = Q^T A Q$$

are estimates of the eigenvalues λ_i of A

Step-length selection: exploiting the Ritz Values

Goal [Fletcher, Math. Program. 2012]

- Define T starting from

$$G = \left[\boldsymbol{g}^{(k-m)}, \ldots, \boldsymbol{g}^{(k-1)}\right] \qquad (m \text{ small; } m = 3,4,5)$$

without explicit use of Q and A

- Compute the Ritz Values θ_i, $i = 1, \ldots, m$
(eigenvalues of the $m \times m$ tridiagonal matrix T)

- Exploit $\alpha_{k-1+i} = \frac{1}{\theta_i}$, $i = 1, \ldots, m$ for m iterations of the gradient methods

$$\boldsymbol{x}^{(k+i)} = \boldsymbol{x}^{(k-1+i)} - \alpha_{k-1+i}\, \boldsymbol{g}^{(k-1+i)}, \qquad i = 1, \ldots, m$$

Behaviour of the Ritz values

Example

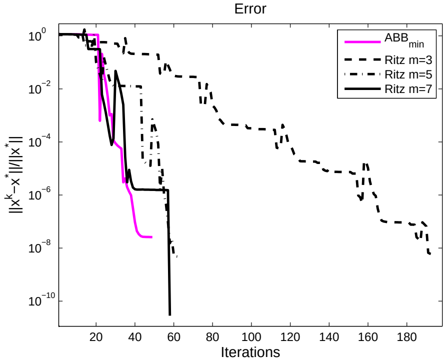
$$f(x) = \frac{1}{2} x^T A x - b^T x$$

- $A = diag(\lambda_1, \ldots, \lambda_{10}), \quad \lambda_i = 111i - 110$
- b random vector; $b_i \in [-10, 10]$.

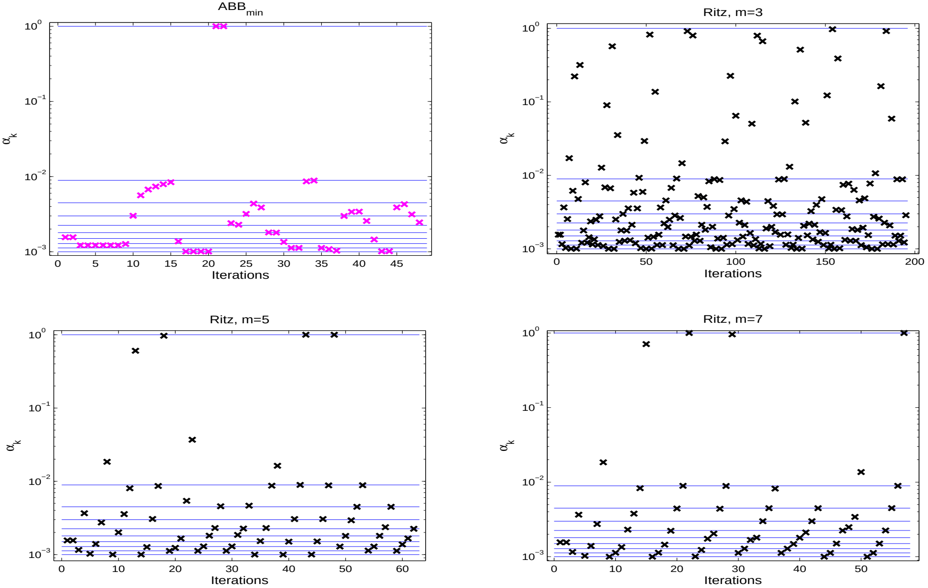
$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \boldsymbol{g}^{(k)}$$

Error $= \|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\| / \|\boldsymbol{x}^*\|$

- —— ABB_{min} $\rightarrow$ BB adaptive alternation
- – – – Ritz with $m = 3$
- – · –· Ritz with $m = 5$
- —— Ritz with $m = 7$



The distribution of the steplengths w.r.t. $\frac{1}{\lambda_i}, \quad i = 1, \ldots, N$



The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Recall the derivation of the BB rules without scaling ($\mathcal{D}_k = I$):

Regard $B(\alpha_k) = (\alpha_k I)^{-1}$ as an approximation of the Hessian $\nabla^2 f(\boldsymbol{x}^{(k)})$

The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Recall the derivation of the BB rules without scaling ($\mathcal{D}_k = I$):

Regard $B(\alpha_k) = (\alpha_k I)^{-1}$ as an approximation of the Hessian $\nabla^2 f(\boldsymbol{x}^{(k)})$

Determine α_k by forcing a quasi-Newton property on $B(\alpha_k)$:

$$\alpha_k^{\text{BB1}} = \underset{\alpha \in \mathbb{R}}{\operatorname{argmin}} \|B(\alpha)\boldsymbol{s}^{(k-1)} - \boldsymbol{z}^{(k-1)}\| = \frac{\boldsymbol{s}^{(k-1)^T}\boldsymbol{s}^{(k-1)}}{\boldsymbol{s}^{(k-1)^T}\boldsymbol{z}^{(k-1)}}$$

or

$$\alpha_k^{\text{BB2}} = \underset{\alpha \in \mathbb{R}}{\operatorname{argmin}} \|\boldsymbol{s}^{(k-1)} - B(\alpha)^{-1}\boldsymbol{z}^{(k-1)}\| = \frac{\boldsymbol{s}^{(k-1)^T}\boldsymbol{z}^{(k-1)}}{\boldsymbol{z}^{(k-1)^T}\boldsymbol{z}^{(k-1)}}$$

where $\boldsymbol{s}^{(k-1)} = \left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right)$ and $\boldsymbol{z}^{(k-1)} = (\boldsymbol{g}^{(k)} - \boldsymbol{g}^{(k-1)})$.

The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Derivation of the BB rules with scaling:

Regard $B(\alpha_k) = (\alpha_k \mathcal{D}_k)^{-1}$ as an approximation of $\nabla^2 f(\boldsymbol{x}^{(k)})$

The step-lengths in Scaled Gradient Methods: the BB case

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Derivation of the BB rules with scaling:

Regard $B(\alpha_k) = (\alpha_k \mathcal{D}_k)^{-1}$ as an approximation of $\nabla^2 f(\boldsymbol{x}^{(k)})$

Determine α_k by forcing a quasi-Newton property on $B(\alpha_k)$:

$$\alpha_k^{\text{BB1}} = \frac{{\boldsymbol{s}^{(k-1)}}^T \mathcal{D}_k^{-1} \mathcal{D}_k^{-1} \boldsymbol{s}^{(k-1)}}{{\boldsymbol{s}^{(k-1)}}^T \mathcal{D}_k^{-1} \boldsymbol{z}^{(k-1)}}$$

or

$$\alpha_k^{\text{BB2}} = \frac{{\boldsymbol{s}^{(k-1)}}^T \mathcal{D}_k \boldsymbol{z}^{(k-1)}}{{\boldsymbol{z}^{(k-1)}}^T \mathcal{D}_k \mathcal{D}_k \boldsymbol{z}^{(k-1)}},$$

where $\boldsymbol{s}^{(k-1)} = \left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right)$ and $\boldsymbol{z}^{(k-1)} = (\boldsymbol{g}^{(k)} - \boldsymbol{g}^{(k-1)})$.

The Ritz values in Scaled Gradient Methods

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

The Ritz values in Scaled Gradient Methods

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Recall the quadratic case: $\min f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A \boldsymbol{x} - b^T \boldsymbol{x}$

- consider the problem $\tilde{f}(\boldsymbol{y}) = \frac{1}{2}\boldsymbol{y}^T \mathcal{D}^{\frac{1}{2}} A \mathcal{D}^{\frac{1}{2}} \boldsymbol{y} - b^T \mathcal{D}^{\frac{1}{2}} \boldsymbol{y}$ and

$$\boldsymbol{y}^{(k+1)} = \boldsymbol{y}^{(k)} - \alpha_k \tilde{\boldsymbol{g}}^{(k)}, \qquad \tilde{\boldsymbol{g}}^{(k)} = \nabla \tilde{f}(\boldsymbol{y}^{(k)})$$

- Let $\boldsymbol{y}^{(k)} = \mathcal{D}^{-\frac{1}{2}} \boldsymbol{x}^{(k)}$; we have $\tilde{\boldsymbol{g}}^{(k)} = \mathcal{D}^{\frac{1}{2}} \boldsymbol{g}^{(k)}$ and

$$\boldsymbol{y}^{(k+1)} = \mathcal{D}^{-\frac{1}{2}} (\boldsymbol{x}^{(k)} - \alpha_k \mathcal{D} \boldsymbol{g}^{(k)}) = \mathcal{D}^{-\frac{1}{2}} \boldsymbol{x}^{(k+1)}$$

- gradient step on $\boldsymbol{y}^{(k)}$ $\leftrightarrow$ scaled gradient step on $\boldsymbol{x}^{(k)}$

The Ritz values in Scaled Gradient Methods

Consider the scaled gradient method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \boldsymbol{g}^{(k)}$

Recall the quadratic case: $\min f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A \boldsymbol{x} - b^T \boldsymbol{x}$

- consider the problem $\tilde{f}(\boldsymbol{y}) = \frac{1}{2}\boldsymbol{y}^T \mathcal{D}^{\frac{1}{2}} A \mathcal{D}^{\frac{1}{2}} \boldsymbol{y} - b^T \mathcal{D}^{\frac{1}{2}} \boldsymbol{y}$ and

$$\boldsymbol{y}^{(k+1)} = \boldsymbol{y}^{(k)} - \alpha_k \tilde{\boldsymbol{g}}^{(k)}, \qquad \tilde{\boldsymbol{g}}^{(k)} = \nabla \tilde{f}(\boldsymbol{y}^{(k)})$$

- Let $\boldsymbol{y}^{(k)} = \mathcal{D}^{-\frac{1}{2}} \boldsymbol{x}^{(k)}$; we have $\tilde{\boldsymbol{g}}^{(k)} = \mathcal{D}^{\frac{1}{2}} \boldsymbol{g}^{(k)}$ and

$$\boldsymbol{y}^{(k+1)} = \mathcal{D}^{-\frac{1}{2}}(\boldsymbol{x}^{(k)} - \alpha_k \mathcal{D} \boldsymbol{g}^{(k)}) = \mathcal{D}^{-\frac{1}{2}} \boldsymbol{x}^{(k+1)}$$

- gradient step on $\boldsymbol{y}^{(k)}$ $\leftrightarrow$ scaled gradient step on $\boldsymbol{x}^{(k)}$

$$G = \left[\mathcal{D}_{k-m}^{\frac{1}{2}} \boldsymbol{g}^{(k-m)}, \ldots, \mathcal{D}_{k-1}^{\frac{1}{2}} \boldsymbol{g}^{(k-1)}\right]$$

$$G^T G = R^T R \qquad R^T \boldsymbol{r} = G^T \mathcal{D}_k^{\frac{1}{2}} \boldsymbol{g}^{(k)} \qquad T = [R \;\; \boldsymbol{r}]\, J\, R^{-1}$$

Diagonal scaling matrix in SGP: the updating rule

- A standard choice: $\mathcal{D}_k = \text{diag}\left(\mathcal{D}_1^{(k)}, \mathcal{D}_2^{(k)}, \dots, \mathcal{D}_n^{(k)}\right)$

$$\mathcal{D}_i^{(k)} = \min\left\{\rho, \max\left\{\frac{1}{\rho}, \left(\frac{\partial^2 f(\boldsymbol{x}^{(k)})}{(\partial x_i)^2}\right)^{-1}\right\}\right\}, \quad i = 1, \dots, n,$$

- Exploit first-order optimality condition (KKT condition)
  To simplify the exposition, consider

$$\min_{\boldsymbol{x} \geq \boldsymbol{0}} \quad f(\boldsymbol{x})$$

KKT condition:

$$\nabla f(\boldsymbol{x}) - \boldsymbol{\lambda} = 0, \quad \boldsymbol{x} \geq \boldsymbol{0}, \quad \boldsymbol{\lambda} \geq \boldsymbol{0}, \quad x_i \lambda_i = 0, \quad i = 1, \dots, n$$

$$\Downarrow$$

$$\boldsymbol{x} \cdot \nabla f(\boldsymbol{x}) = \boldsymbol{0}, \quad \boldsymbol{x} \geq \boldsymbol{0}, \quad \nabla f(\boldsymbol{x}) \geq \boldsymbol{0}$$

" $\cdot$ " denotes the component-wise product

Diagonal scaling matrix in SGP: the updating rule

Split the gradient [Lantéri-Roche-Aime, *Inv. Prob.* (2002)]:

$$\nabla f(\boldsymbol{x}) = V(\boldsymbol{x}) - U(\boldsymbol{x}), \quad V(\boldsymbol{x}) > 0, \quad U(\boldsymbol{x}) \geq 0$$

and use the splitting in the nonlinear equation $\boldsymbol{x} \cdot \nabla f(\boldsymbol{x}) = \mathbf{0}$:

$$\boldsymbol{x} \cdot V(\boldsymbol{x}) = \boldsymbol{x} \cdot U(\boldsymbol{x}) = \boldsymbol{x} \cdot (-\nabla f(\boldsymbol{x}) + V(\boldsymbol{x})),$$

⇩

$$\boldsymbol{x} = \boldsymbol{x} - \frac{\boldsymbol{x}}{V(\boldsymbol{x})} \cdot \nabla f(\boldsymbol{x}) = \boldsymbol{x} - \mathcal{D}\nabla f(\boldsymbol{x}), \quad \mathcal{D} = \text{diag}\left(\frac{x_1}{V_1(\boldsymbol{x})}, \dots, \frac{x_n}{V_n(\boldsymbol{x})}\right)$$

Iterative methods for $\boldsymbol{x} \cdot \nabla f(\boldsymbol{x}) = \mathbf{0}$ based on scaled gradient direction:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \mathcal{D}_k \nabla f(\boldsymbol{x}^{(k)}), \quad \mathcal{D}_k = \text{diag}\left(\frac{x_1^{(k)}}{V_1(\boldsymbol{x}^{(k)})}, \dots, \frac{x_n^{(k)}}{V_n(\boldsymbol{x}^{(k)})}\right)$$

Diagonal scaling matrix in SGP: the updating rule

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \lambda_k \left(P_+(\boldsymbol{x}^{(k)} - \alpha_k \mathcal{D}_k \nabla f(\boldsymbol{x}^{(k)})) - \boldsymbol{x}^{(k)} \right)$$

- use the split gradient idea to define the SGP scaling matrix:

$$\mathcal{D}_i^{(k)} = \min \left\{ \rho, \max \left\{ \frac{1}{\rho}, \frac{x_i^{(k)}}{V_i(\boldsymbol{x}^{(k)})} \right\} \right\}, \quad V_i(\boldsymbol{x}^{(k)}) > 0, \quad i = 1, \dots, n,$$

  - In some applications the splitting $\nabla f(\boldsymbol{x}) = V(\boldsymbol{x}) - U(\boldsymbol{x})$ is suggested by the form of the gradient (problem dependent scaling matrix) e.g.: algorithms in imaging (EM, ISRA) exploit this approach

  - similar idea used in [Hager-Mair-Zhang, *Math. Program.* (2009)] in case of special constraints (e.g. $\boldsymbol{x} \geq 0$)

$$\mathcal{D}_i^{(k)} = \frac{\alpha_k x_i^{(k)}}{x_i^{(k)} + \alpha_k \left(\nabla f(\boldsymbol{x}^{(k)}) \right)_i^+}, \quad i = 1, \dots, n, \quad (t)^+ = \max\{0, t\}$$

SGP(ABB$_{min}$) vs SGP(Ritz)

Benchmark test problems in image deblurring

SGP as solver for the regularized problem

$$\min_{\boldsymbol{x}\geq \mathbf{0}} \; f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x}), \qquad \beta > 0$$

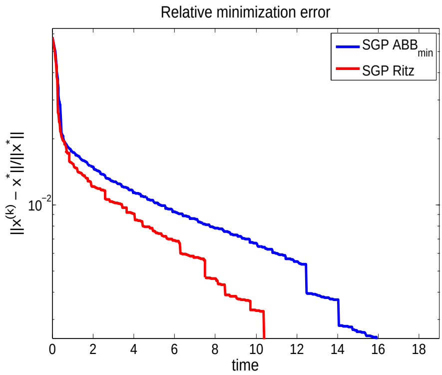
$$f_0(\boldsymbol{x}) = \sum_{i=1}^{n} \left\{ y_i \, \log \frac{y_i}{(A\boldsymbol{x}+b)_i} + (A\boldsymbol{x}+b)_i - y_i \right\}$$

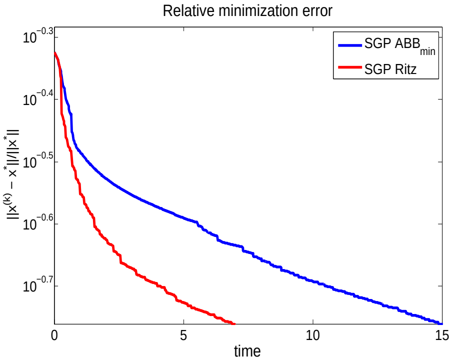
$$f_1(\boldsymbol{x}) = \sum_{k=1}^{n} \sqrt{\Delta_k}, \qquad \Delta_k = (x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2 + \delta^2$$

SGP as solver for a regularized problem

$$x^* = \operatorname*{argmin}_{x \geq 0} f_\beta(x) \ ; \qquad \text{stopping rule: } |f_\beta(x^{(k)}) - f_\beta(x^{(k-1)})|/|f_\beta(x^{(k)})| \leq t_f$$

	Cameraman $(t_f = 10^{-8})$			**Spacecraft** $(t_f = 10^{-6})$		
	it.	Sec.	err.	it.	Sec.	err.
SGP ABB$_{min}$	974	18.0	0.0011	1063	20.0	0.16
SGP Ritz	504	11.3	0.0023	400	8.7	0.16





Solving $D_{\boldsymbol{y}}(\bar{\beta}) = \tau$ by using SGP as inner solver

Given β, the computation of $D_{\boldsymbol{y}}(\beta)$ requires

$$\boldsymbol{x}^*_\beta = \underset{\boldsymbol{x}\geq 0}{\operatorname{argmin}}\, f_\beta(\boldsymbol{x}) = f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x})$$

We obtain $\boldsymbol{x}^*_\beta$ by SGP(Ritz) with stopping rule on $f_\beta(\boldsymbol{x}^{(k)})$

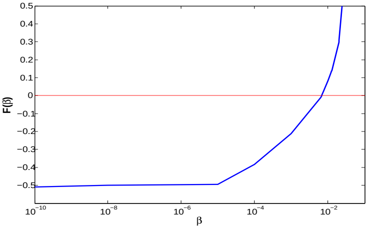
$$er_k < t_f \qquad er_k = |f_\beta(\boldsymbol{x}^{(k)}) - f_\beta(\boldsymbol{x}^{(k-1)})| \;/\; |f_\beta(\boldsymbol{x}^{(k)})|$$

➤ Design a root finding solver for $D_{\boldsymbol{y}}(\bar{\beta}) = \tau$

Two phases secant-based algorithm

1) Bracketing Phase:
$\beta_l < \beta_u \;\Rightarrow\; D_{\boldsymbol{y}}(\beta_l) < D_{\boldsymbol{y}}(\beta_u)$

2) Secant Phase in $[\beta_l\,,\;\beta_u]$



Two phases secant-based algorithm: the Bracketing phase

Input: a tentative β, an initial step $d\beta$, $\gamma \in (0,1)$, and $F(\beta) = \frac{2}{n} D_{\boldsymbol{y}}(\beta) - 1$

```
if    F(β) < 0
    β_l = β,    β = β + dβ
    while    F(β) < 0
        ds = secant step,  dβ = dβ + ds      ⟵  secant-like steps to increase β
        β_l = β,    β = β + dβ
    end
    β_u = β
else
    β_u = β,    β = β_u γ
    while    F(β) > 0
        β_u = β,    β = β_u γ                 ⟵  progressive reduction of β
    end                                           for slowly approaching β = 0
    β_l = β                                       (β ≈ 0 ⇒ difficult opt. problems)
end
```

Output: β_l, β_u such that $\bar{\beta} \in [\beta_l\,,\ \beta_u]$

Two phases secant-based algorithm: the Bracketing phase

SGP (Ritz): warm start not useful; progressive stopping tol. very useful.

		Cameraman ($\beta_0 = 0.1$, SGP reference tol.$= 10^{-9}$, $\boldsymbol{x}^{(0)} = \boldsymbol{y}$)					
		severe SGP tol.			progressive SGP tol.		
it.	β	t_f	inner it.	$F(\beta)$	t_f	inner it.	$F(\beta)$
1	0.1	10^{-8}	504	1.9900	10^{-6}	197	1.8099
2	0.01	10^{-9}	440	0.0810	$2 \cdot 10^{-7}$	198	0.0821
3	0.001	10^{-9}	644	-0.2113	$4 \cdot 10^{-8}$	428	-0.2111
			1588			**823**	

		Spacecraft ($\beta_0 = 0.1$, SGP reference tol.$= 10^{-9}$, $\boldsymbol{x}^{(0)} = \boldsymbol{y}$)					
		severe SGP tol.			progressive SGP tol.		
it.	β	t_f	inner it.	$F(\beta)$	t_f	inner it.	$F(\beta)$
1	0.1	10^{-8}	3046	0.3658	10^{-6}	400	0.5427
2	0.01	10^{-9}	1966	0.0403	$2 \cdot 10^{-7}$	576	0.0417
3	0.001	10^{-9}	1546	0.0014	$4 \cdot 10^{-8}$	1022	0.0017
4	0.0001	10^{-9}	2370	-0.0046	$8 \cdot 10^{-9}$	1589	-0.0044
			8928			**3587**	

Two phases secant-based algorithm: the Secant phase

Input: β_l, β_u, $flag(t_f) \in \{0,1\}$ (SGP progressive tol.), $t_{min} > 0$, $stop(\beta) = 0$

if $flag(t_f) = 1$
reduce t_f
refine $F(\beta_l)$ by improving $\boldsymbol{x}^*_{\beta_l}$ (warm start in SGP($t_f, \boldsymbol{x}^*_{\beta_l}$))
refine $F(\beta_u)$ by improving $\boldsymbol{x}^*_{\beta_u}$ (warm start in SGP($t_f, \boldsymbol{x}^*_{\beta_u}$))
end

update β by a secant step and $F(\beta)$ by SGP($t_f, \boldsymbol{y}$)

$t_f = t_{min}$

while $(\sim stop(\beta))$
update β by a secant-like step and $F(\beta)$ (warm start in SGP($t_f, \boldsymbol{x}^*_\beta$))
end

Output: $\bar{\beta}$ such that $F(\bar{\beta}) \approx 0$

Two phases secant-based algorithm: the Secant phase

SGP (Ritz): warm start very useful;

	Cameraman (SGP reference tol.$= 10^{-9}$)			
	β	t_f	inner it.	$F(\beta)$
1	1.00e-3	6.3e-9	37	-2.1e-1
2	1.00e-2	6.3e-9	173	8.1e-2
3	7.50e-3	6.3e-9	357	2.0e-2
4	6.67e-3	3.3e-10	271	-1.0e-3
5	6.71e-3	3.3e-10	38	-9.8e-5
			876	

	Spacecraft (SGP reference tol.$= 10^{-9}$)			
	β	t_f	inner it.	$F(\beta)$
1	1.00e-4	2.8e-9	374	-4.5e-3
2	1.00e-3	2.8e-9	919	1.4e-3
3	7.88e-4	2.8e-9	1756	3.6e-4
4	7.14e-4	3.3e-10	323	1.1e-4
			3372	

A constrained model for the regularization parameter

An alternative approach for computing $\bar{\beta}$ such that

$$D_{\boldsymbol{y}}(\bar{\beta}) = D_{KL}(\boldsymbol{y}, A\boldsymbol{x}^*_{\bar{\beta}}) = \tau \qquad (1)$$

can be derived by exploiting the relation between the problems

$$\operatorname*{argmin}_{\boldsymbol{x}\geq 0} f_1(L\boldsymbol{x}) \qquad \text{subject to } D_{KL}(\boldsymbol{y}, A\boldsymbol{x}) \leq \tau \qquad (2)$$

and

$$\operatorname*{argmin}_{\boldsymbol{x}\geq 0} f_1(L\boldsymbol{x}) + \lambda D_{KL}(\boldsymbol{y}, A\boldsymbol{x}) \qquad (3)$$

⇩

By solving (2) by a primal-dual algorithm, a sequence $\{\lambda^{(k)}\}_k$ is generated converging to a parameter $\hat{\lambda}$ such that $\bar{\beta} = \frac{1}{\hat{\lambda}}$ satisfies the discrepancy equation (1).

[Carlavan, Blanc-Feraud, 2011,2012], [Teuber, Steidl, Chan, 2013]

A constrained model for the regularization parameter

$$\tau_0 = \min_{\boldsymbol{x} \geq 0} D_{KL}(\boldsymbol{y}, A\boldsymbol{x}), \quad \tau_L = \min_{\boldsymbol{x} \geq \boldsymbol{0}, \boldsymbol{x} \in \mathcal{N}(L)} D_{KL}(\boldsymbol{y}, A\boldsymbol{x}), \quad \mathcal{K} = \{\boldsymbol{x} \geq 0 : A\boldsymbol{x} > 0\}$$

Let $\boldsymbol{y} > 0$, $\mathcal{K} \neq \emptyset$, $\mathcal{N}(L) \cap \mathcal{N}(A) = \{\boldsymbol{0}\}$, $\mathrm{argmin}_{\boldsymbol{x} \geq \boldsymbol{0}} D_{KL}(\boldsymbol{y}, A\boldsymbol{x}) \cap \mathcal{N}(L) = \emptyset$.
If $\hat{\boldsymbol{x}}$ is a solution of (2) with $\tau_0 < \tau < \tau_L$, then there exists a unique $\lambda > 0$ such that $\hat{\boldsymbol{x}}$ is a solution of (3).
Moreover λ does not depend on the chosen solution of (2).

Under the above assumptions, if

$$\tau_0 < \frac{n}{2} < \tau_L$$

the solution $\bar{\beta}$ of the discrepancy equation exists and is unique ($\bar{\beta} = \frac{1}{\lambda}$).

⇩

Compute $\bar{\beta}$ by solving the divergence constrained problem (2)

ADMM for the constrained problem

We set $\boldsymbol{q}_i = \gamma \boldsymbol{p}_i$, $i = 1, 2, 3$, $\gamma > 0$;

- $\boldsymbol{q}_i^{(0)} = 0$, $\boldsymbol{w}_1^{(0)} = A\boldsymbol{y}$, $\boldsymbol{w}_2^{(0)} = L\boldsymbol{y}$, $\boldsymbol{w}_3^{(0)} = \boldsymbol{y}$; $\lambda_0 = \lambda_{-1} = 0$;
- For $k = 0, 1, ...$ repeat until a suitable stopping criterion is fulfilled

1. $\boldsymbol{x}^{(k+1)} = \text{argmin}_{\boldsymbol{x}} ||\boldsymbol{q}_1^{(k)} + A\boldsymbol{x} - \boldsymbol{w}_1^{(k)}\|^2 + ||\boldsymbol{q}_2^{(k)} + L\boldsymbol{x} - \boldsymbol{w}_2^{(k)}\|^2 + ||\boldsymbol{q}_3^{(k)} + \boldsymbol{x} - \boldsymbol{w}_3^{(k)}\|^2$
2. $\boldsymbol{w}_1^{(k+1)} = \text{argmin}_{\boldsymbol{w}_1 \in \text{lev}_\tau D_{KL}(\boldsymbol{y}, \boldsymbol{w}_1)} \frac{1}{2\gamma} \|\boldsymbol{q}_1^{(k)} + A\boldsymbol{x}^{(k+1)} - \boldsymbol{w}_1\|^2$;
   computation of λ_{k+1}, Lagrange multiplier of the inequality constraint;
3. $\boldsymbol{w}_2^{(k+1)} = \text{argmin}_{\boldsymbol{w}_2} f_1(\boldsymbol{w}_2) + \frac{1}{2\gamma} \|\boldsymbol{q}_2^{(k)} + Lx^{(k+1)} - \boldsymbol{w}_2\|^2$
4. $\boldsymbol{w}_3^{(k+1)} = \text{argmin}_{\boldsymbol{w}_3 \geq 0} \frac{1}{2\gamma} \|\boldsymbol{q}_3^{(k)} + x^{(k+1)} - \boldsymbol{w}_3\|^2 = \max(\boldsymbol{q}_3^{(k)} + \boldsymbol{x}^{(k+1)}, 0)$
5. $\boldsymbol{q}_1^{(k+1)} = \boldsymbol{q}_1^{(k)} + A\boldsymbol{x}^{(k+1)} - \boldsymbol{w}_1^{(k+1)}$
6. $\boldsymbol{q}_2^{(k+1)} = \boldsymbol{q}_2^{(k)} + L\boldsymbol{x}^{(k+1)} - \boldsymbol{w}_2^{(k+1)}$
7. $\boldsymbol{q}_3^{(k+1)} = \boldsymbol{q}_3^{(k)} + \boldsymbol{x}^{(k+1)} - \boldsymbol{w}_3^{(k+1)}$

Crucial steps

- The first step requires the solution of a linear system:

$$\boldsymbol{x}^{(k+1)} = (A^T A + L^T L + I)^{-1} (A^T (\boldsymbol{w}_1^{(k)} - \boldsymbol{q}_1^{(k)}) + L^T (\boldsymbol{w}_2^{(k)} - \boldsymbol{q}_2^{(k)}) + (\boldsymbol{w}_3^{(k)} - \boldsymbol{q}_3^{(k)}))$$

- The computation of $\boldsymbol{w}_2^{(k+1)}$ depends on the regularization term
- If $\boldsymbol{y} > 0\,,\;\; \tau > 0\,,\;\; \boldsymbol{z} = \boldsymbol{q}_1^{(k)} + A\boldsymbol{x}^{(k+1)}\,,$ from the solution of

$$\min_{\boldsymbol{w}} \frac{1}{2} \|\boldsymbol{z} - \boldsymbol{w}\|^2 \text{ sub. to } \quad D_{KL}(\boldsymbol{y}, \boldsymbol{w}) \le \tau$$

  we compute $\boldsymbol{w}_1^{(k+1)}$ and λ_{k+1} [Carlavan, Blanc-Feraud, 2011,2012].

  By few Newton's steps we compute the solution $\hat{\lambda}$ of $D_{KL}(\boldsymbol{y}, \boldsymbol{w}(\boldsymbol{z}, \lambda)) = \tau$ where $\boldsymbol{w}(\boldsymbol{z}, \lambda)$ is the solution of $\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{w}\|^2 + \lambda D_{KL}(\boldsymbol{y}, \boldsymbol{w})$.
  Set $\lambda_{k+1} = \frac{\hat{\lambda}}{\gamma}$ and $\boldsymbol{w}_1^{(k+1)} = \boldsymbol{w}(\boldsymbol{z}, \hat{\lambda})$.

The sequence $\{(\boldsymbol{x}^{(k)}, \boldsymbol{w}^{(k)}, \boldsymbol{q}^{(k)}, \lambda_k)\}$ converges to $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{w}}, \tilde{\boldsymbol{q}}, \frac{1}{\beta})$, where $\tilde{\boldsymbol{x}}$ is a solution of the constrained problem and the related penalized problem with $\beta > 0$ and $\tilde{\boldsymbol{p}} = \frac{\tilde{\boldsymbol{q}}}{\gamma}$ is a solution of the dual problems [Teuber, Steidl, Chan 2013].

Cameraman test problem: $\min_{\boldsymbol{x}\geq\mathbf{0}} \; f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x})$

$$f_1(\boldsymbol{x}) = \sum_{k=1}^{n} \sqrt{\Delta_k}, \qquad \Delta_k = (x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2 + \delta^2$$

Solving the discrepancy equation

Method	It.	SGP it.	Time	β	$D_{\boldsymbol{y}}(\beta)$
Secant-based	8	1699	37.0	$6.710\ 10^{-3}$	32765

Solving the constrained model

Method	It.	Time	β	$D_{\boldsymbol{y}}(\beta)$
ADMM ($\gamma = 10$)	540	19.3	$6.706\ 10^{-3}$	32766
ADMM ($\gamma = 50$)	943	33.2	$6.717\ 10^{-3}$	32771
ADMM ($\gamma = 100$)	2020	71.4	$6.717\ 10^{-3}$	32771
ADMM ($\gamma = 200$)	4093	146.0	$6.717\ 10^{-3}$	32771

Spacecraft test problem: $\min_{\boldsymbol{x}\geq \boldsymbol{0}} \; f_0(\boldsymbol{x}) + \beta f_1(\boldsymbol{x})$

$$f_1(\boldsymbol{x}) = \sum_{k=1}^{n} \sqrt{\Delta_k}, \qquad \Delta_k = (x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2 + \delta^2$$

Solving the discrepancy equation

Method	It.	SGP it.	Time	β	$D_{\boldsymbol{y}}(\beta)$
Secant-based	8	6959	153.5	$7.14\ 10^{-4}$	32772

Solving the constrained model

Method	It.	Time	β	$D_{\boldsymbol{y}}(\beta)$
ADMM ($\gamma = 0.9$)	4891	173.1	$6.800\ 10^{-4}$	32771
ADMM ($\gamma = 0.95$)	6373	225.4	$7.062\ 10^{-4}$	32771
ADMM ($\gamma = 1$)	9093	317.8	$7.273\ 10^{-4}$	32771

Estimating β by the discrepancy principle: the reconstructions

Original Image

Observed Image

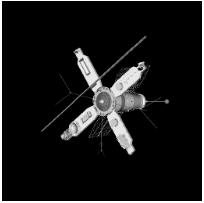
reconstruction

Secant-based rec.: $\beta = 6.710\ 10^{-3}$, reconstruction err.$=0.08600$

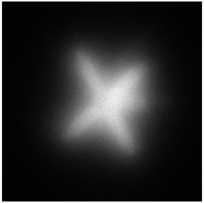
ADMM rec.: $\gamma = 50$, $\beta = 6.717\ 10^{-3}$, err.$=0.08559$

Estimating β by the discrepancy principle: the reconstructions

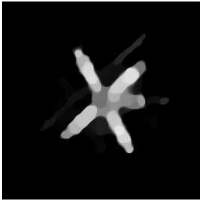
Original Image



Observed Image



reconstruction



Secant-based rec.: $\beta = 7.14\ 10^{-4}$, reconstruction err.$=0.3071$

ADMM rec.: $\gamma = 0.95$, $\beta = 7.06\ 10^{-4}$, err.$=0.3074$

Conclusions

- The regularization parameter estimation provided by the discrepancy principle can be computed

  - by solving directly the non linear equation
    it works well if the root finding solver is equipped with an effective inner optimization solver (for differentiable regularization terms this is the case of the SGP solver)

  - by solving the constrained problem with ADMM
    suitable approach also in case of nondifferentiable regularization terms

- Work in progress for improving the estimation provided by the discrepancy principle.