

Weekly Exercises 6

Room 01.09.014

Friday, 3.6.2016, 09:00-11:00

Submission deadline: Wednesday, 1.6.2016, 14:00, Room: 02.09.023

Theory: Line Search

(6 Points)

Exercise 1 (6 Points). In the lecture, we have seen that for $E : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$

$$E(u) = G(u) + F(u),$$

with closed, proper, convex $G : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ and L -smooth $F : \mathbb{R}^n \rightarrow \mathbb{R}$ the proximal gradient algorithm given by the iteration

$$u^{k+1} = \text{prox}_{\tau G}(u^k - \tau \nabla F(u^k)),$$

converges with rate $\mathcal{O}(1/k)$, i.e.

$$E(u^k) - E(u^*) \in \mathcal{O}(1/k).$$

In this exercise, we select τ_k using a line search (In practice this is helpful if L is not known): The line search works as follows: Start at some initial $\tau := \hat{\tau} > 0$ and iteratively compute $\tau := \beta\tau$ for $0 < \beta < 1$ until the following inequality, that we have seen in the lecture holds:

$$F(u - \tau\varphi_\tau(u)) \leq F(u) - \tau\langle \nabla F(u), \varphi_\tau(u) \rangle + \frac{\tau}{2} \|\varphi_\tau(u)\|^2.$$

Prove that proximal gradient with line search converges with rate $\mathcal{O}(1/k)$.

Hint: Show that the step size τ selected by the line search satisfies

$$\tau \geq \tau_{\min} = \min\{\hat{\tau}, \beta/L\}.$$

Programming: TV Deblurring

(12 Points)

Exercise 2 (12 Points). Given a blurry and noisy input image f , reconstruct a sharper image u^* by solving the following optimization problem

$$u^* = \arg \min_u \frac{1}{2} \|k * u - f\|^2 + \alpha \|Du\|_{2,1} \quad (1)$$

with proximal gradient descent (for the definition of the convolution $k * u$, see the lecture slides). To do so, perform the following steps:

- First construct a convolution kernel k of your choice, for example by using the MATLAB command `fspecial`.
- Then build a sparse matrix representing the convolution with an image u . You can use any boundary condition, and feel free to use `convmtx2`.
- Given the image `flowers.png`, construct a blurred and noisy version by applying your sparse matrix to it and add some Gaussian noise using `randn`.
- Restore the original image by solving (1) using proximal gradient descent. Solve the inner TV denoising problem using projected gradient descent on the dual problem.
- Experiment with different amounts of inner projected gradient descent iterations (use 15 as a starting point) and use the solution of the previous outer iteration to warm-start the algorithm.