Tutorial session 1 - Convex optimization

1 Context

Optimization methods are widely employed for the recovery of digital images that have undergone deterioration.
Recently, in the attic of our family mansion, I stumbled upon an old photograph of my grandmother while she was young. Regrettably, the picture has suffered degradation over time. Having digitized it, I am now poised to apply my expertise in optimization to restore the image.

2 Data

1. Examine the Matlab script `test_restor` and execute the section of the code that downloads the degraded image `lena_degraded.mat` from the class webpage. The script utilizes the `imshow` function to display the image. What is the size of the image ? What two types of degradation can be observed ?
2. How can we estimate the indices of pixels corresponding to the tearing in the picture and the indices of the pixels in the complementary area ? In the following, the set of the latter indices is denoted by $\mathbb{I}$.

3 Variational approach

A digital gray-scale image of size $K \times L$ can be viewed as a real-valued matrix $x = (x_{k,\ell})_{1 \le k \le K, 1 \le k \le L}$. The underlying Hilbert space is thus $\mathcal{H} = \mathbb{R}^{K \times L}$ endowed with the Frobenius norm.
We propose to generate the restored image by solving the following optimization problem:

$$\underset{x \in C}{\text{minimize}} \;\; g(Dx) \tag{1}$$

where $C \subset \mathcal{H}$ is some suitable nonempty closed convex constraint set. Hereabove, D is a linear operator from $\mathcal{H}$ to some finite dimensional Hilbert space $\mathcal{G}$, and g is a function in $\Gamma_0(\mathcal{G})$.

a Quadratic smoothness criterion

The cost function is chosen so as to promote the smoothness of the restored image. In this context, D is the gradient operator

$$\begin{aligned} D \colon \mathcal{H} &\to \mathcal{G} = (\mathbb{R}^2)^{(K-1)\times(L-1)} \\ x &\mapsto \delta = (v_{k,\ell}, h_{k,\ell})_{1 \le k \le K-1, 1 \le \ell \le L-1} \end{aligned}$$

where, for every $k \in \{1, \ldots, K-1\}$ and $\ell \in \{1, \ldots, L-1\}$,

$$\begin{aligned} v_{k,\ell} &= x_{k+1,\ell} - x_{k,\ell} \\ h_{k,\ell} &= x_{k,\ell+1} - x_{k,\ell} \end{aligned}$$

are the discrete spatial gradients at pixel (k, ℓ) in the vertical and horizontal directions. It can shown, by spectral arguments, that the norm of this operator is $\sqrt{8}$. The operator D is implemented in the provided `grad` function and its adjoint (the minus divergence) can be computed with the provided

`div` function.
For simplicity, we propose first to consider the simple squared norm:

$$\big(\forall \delta = (h_{k,\ell}, v_{k,\ell})_{1\le k\le K-1, 1\le \ell\le L-1} \in \mathcal{G}\big) \qquad g(\delta) = \frac{1}{2}\|\delta\|^2 = \frac{1}{2}\sum_{k=1}^{K-1}\sum_{\ell=1}^{L-1}(v_{k,\ell}^2 + h_{k,\ell}^2).$$

3. Show that $g \circ D$ is differentiable. What is the gradient of this function ? What is the Lipschitz constant of this gradient ?

4. If no constraint is imposed ($C = \mathcal{H}$), propose a gradient algorithm to solve numerically Problem (1) by completing the code in `test_restor`. Comment on the results.

b Constraint

In order to perform the restoration task, we introduce the following constraint set:

$$C = \Big\{x = (x_{k,\ell})_{1\le k\le K, 1\le \ell\le L} \in \mathcal{H} \mid \sum_{(k,\ell)\in \mathbb{I}} (x_{k,\ell} - z_{k,\ell})^2 \le \rho^2\Big\}$$

where $z = (z_{k,\ell})_{1\le k\le K, 1\le \ell\le L}$ is the degraded image and $\rho \in \,]0, +\infty[$ is a given bound.

5. What is the projection onto C? Complete the function `projball2` for computing this projection.

6. Modify the algorithm developed in Question 4 to obtain a projected gradient algorithm solving the constrained optimization problem.

7. Implement a stopping rule based on the following condition:

$$|g(Dx_n) - g(Dx_{n+1})| < \epsilon g(Dx_n)$$

   where n is the current iteration and ϵ is a tolerance parameter set to 10^{-6}.

8. Test the algorithm. (A good value for ρ is $0.2 \times \sqrt{KL}$.) Observe the decrease of the cost value along the iterations.

c Second cost function

A popular smoothness function in image processing is the smoothed total variation defined as

$$(\forall \delta = (h_{k,\ell}, v_{k,\ell})_{1\le k\le K-1, 1\le \ell\le L-1} \in \mathcal{G}) \qquad g(\delta) = \sum_{k=1}^{K-1}\sum_{\ell=1}^{L-1}\tau(v_{k,\ell}, h_{k,\ell}),$$

where

$$(\forall (\upsilon, \theta) \in \mathbb{R}^2) \quad \tau(\upsilon, \theta) = \sqrt{\upsilon^2 + \theta^2 + \eta^2}$$

with $\eta \in \,]0, +\infty[$ (typically, $\eta = 5 \times 10^{-3}$).

9. Show that τ is differentiable and has a $1/\eta$-Lipschitzian gradient. Provide the expression of its gradient. Same questions for g.

10. Write the Matlab code of a projected gradient algorithm for solving Problem (1) when the cost function is the smoothed total variation. How should the step-size be chosen in this case ? What about the results in terms of image quality and convergence speed ?

d A faster algorithm

It is possible to make the convergence of the projected gradient faster by using a Nesterov acceleration. The modified algorithm for minimizing a Lipschitz-differentiable function f onto a nonempty closed convex set C reads

$$(\forall n \in \mathbb{N}) \qquad \begin{cases} x_{n+1} = P_C\big(y_n - \gamma \nabla f(y_n)\big) \\ \lambda_n = \dfrac{n}{n+1+\zeta} \\ y_{n+1} = x_{n+1} + \lambda_n (x_{n+1} - x_n) \end{cases}$$

where $x_0 = y_0$, $\zeta \in]2, +\infty[$ (e.g., $\zeta = 2.1$), and $\gamma \in]0, 1/\beta]$ where β is the Lipschitz constant of ∇f. In this case, both $(x_n)_{n\in\mathbb{N}}$ and $(y_n)_{n\in\mathbb{N}}$ converge to a minimizer of the constrained problem.

11. Implement this algorithm for the smoothed total variation cost function. Which conclusions can be drawn ?