Constrained optimization – Non-differentiable problems

Optimization with Applications to Image Processing

19/04/2012

1 Constrained optimization

1.1 Projected gradient method

Let \( A \) be an invertible matrix in \( \mathbb{R}^{N \times N} \) and \( b \in \mathbb{R}^N \), we want to find the minimum of

\[
J(x) = \|Ax - b\|_2^2, \quad \text{subject to} \quad x_i \geq 0, \quad i = 1 \leq i \leq N.
\]

1. Compute \( \nabla J \).
2. Show that \( J \) is strongly convex, and that \( \nabla J \) is Lipschitz (give the associated constants).
3. Specify the projected gradient algorithm for the minimization problem considered here. For what values of the step-size \( \tau \) is convergence ensured?
4. Write down a function \( x = \text{minimize}(A, b, \tau, r) \) that computes the output of the previous algorithm at iteration \( r \).
5. Apply the previous parts to solve the minimization problem with

\[
A = \begin{pmatrix} 1 & 2 \\ -1 & 3 \end{pmatrix}, \quad b = \begin{pmatrix} -3 \\ 4 \end{pmatrix}.
\]
6. Check that the obtained solution is correct, by computing the solution by hand using Kuhn-Tucker relations.

1.2 Projected gradient method (bis)

Consider the set \( C \in \mathbb{R}^3 \),

\[
C = \{ (x_1, x_2, x_3) \in \mathbb{R}^3, x_1^2 + x_2^2 \leq 1 \}.
\]

1. (a) What is the geometric form of \( C \)? Show that \( C \) is convex.
   (b) Write an explicit expression for \( P_C \), the orthogonal projection on \( C \).
   (c) Write down a function \( z = \text{project}(x) \) that takes as input a vector \( x \in \mathbb{R}^3 \) and compute its projection on \( C \in \mathbb{R}^3 \).
2. Let \( A \in \mathbb{R}^{3 \times 3} \) invertible, and \( b \in \mathbb{R}^N \). We consider the function \( J : \mathbb{R}^3 \to \mathbb{R}^3 \) defined by \( J(x) = \|Ax - b\|_2^2 \), and the problem (P): minimize \( J(x) \) under the constraint \( x \in C \).
   (a) Show that (P) admits a unique solution.
   (b) Specify the projected gradient algorithm for the minimization problem considered here. For what values of the step-size \( \tau \) is convergence ensured?
   (c) Write down a function \( x = \text{minimize}(A, b, \tau, r) \) that computes the output of the previous algorithm at iteration \( r \).

*Exercises by J.-F. Aujol
(d) Solve (P) numerically for
\[
A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 1 \\ 1 & 0 & 2 \end{pmatrix}, \quad b = \begin{pmatrix} 3 \\ 1 \\ 3 \end{pmatrix}.
\]

(e) For this particular case, and for a precision of $10^{-4}$, find empirically the largest step-size $\tau_c$ for which convergence is observed.

### 1.3 Active constraint and Uzawa’s method

In this exercise we consider the problem (P):
\[
\begin{align*}
\text{maximize } & f(x_1, x_2) = x_2^2 + x_2^2 - 14x_1 - 6x_2 - 7 \\
\text{s.t. } & x_1 + x_2 \leq 2, \\
& x_1 + 2x_2 \leq 3.
\end{align*}
\]

1. Compute the gradient and the Hessian matrix of $f$.
2. Solve first the unconstrained minimization of $f$, by a method of your choice.
3. Implement a penalization method to solve the constrained minimization problem (P). Find the minimizer. Which is the active constraint?
4. Implement the Newton-Lagrange method, which consists in applying Newton’s method to $f$ restricted to the active constraint.
5. Write (P) in the form: $\min < Au, u > - < b, u > \text{ s.t. } Cu \leq d$, and solve it using Uzawa’s method. Check consistency between this numerical solution and the previous one.

### 2 A non-differentiable problem: Total Variation minimization

In the following exercises we explore different approaches to minimize the energy associated to the image restoration model proposed by Rudin, Osher and Fatemi (ROF):
\[
E(u) = \frac{1}{2\mu} \| f - u \|^2 + \int_\Omega |Du|.
\]

#### 2.1 Differentiable approximation of the Total Variation

Consider the following approximation of ROF’s model:
\[
E_\epsilon(u) = \int_\Omega \sqrt{|\nabla u|^2 + \epsilon^2} + \frac{1}{2\mu} \| f - u \|^2.
\]

##### 2.1.1 Gradient descent

Compute the Euler-Lagrange equation associated to $E_\epsilon$. Find the minimizer using a gradient descent method. Choose an image and perturb it with white Gaussian noise. Restore it and estimate empirically the convergence speed of the method.

##### 2.1.2 Quasi-Newton method

Minimize $E_\epsilon$ using a quasi-Newton method. Determine empirically the convergence speed. Remark: it can be proved that this kind of method has linear convergence rate, however quadratic convergence rates are usually observed in practice.

#### 2.2 Projection Algorithms

We recall that the solution of ROF model is given by
\[
u = f - \mu P_G(f/\mu).
\]
Hence, the computation of $u$ is straightforward once we know how to compute the projection on $G$. 

2
2.2.1 Chambolle’s algorithm

To compute the projection, the discrete problem to be solved is

$$\min \left\{ \|\text{div}(p) - f/\mu\|^2 : p \in (\mathbb{R}^{N\times N})^2, \|p(i, j)\|^2 \leq 1, \ 1 \leq i, j \leq N \right\}.$$ 

One possibility is to use a fixed point method to solve the Kuhn-Tucker relations. This gives:

$$p_0 = 0,$$

$$p_{i,j}^{n+1} = p_{i,j}^n + \tau (\nabla \text{div}(p) - f/\mu)_{i,j}$$

Chambolle shows that for $\tau < 1/8$, then $\text{div}(p)$ converges to $P_{\mathcal{G}}(f/\mu)$ as $n \to +\infty$. In practice, convergence is observed for $\tau < 1/4$.

Implement the previous iteration, test it on images perturbed by noise, and check empirically the previous assertions on $\tau$. Perform an empirical analysis of the convergence speed.

2.2.2 Projected gradient algorithm

The projection can also be computed using a projected gradient method:

$$v^n = f/\mu + \text{div}(p^n)$$

$$p_{i,j}^{n+1} = \frac{p_{i,j}^n + \tau (\nabla v^n)_{i,j}}{\max \{1, |p_{i,j}^n + \tau (\nabla v^n)_{i,j}|\}}$$

It can be shown that for $\tau < 1/4$, the iteration converges to the minimizer or $E(u)$.

Implement the previous iteration, test it on images perturbed by noise, and check empirically the previous assertions on $\tau$. Perform an empirical analysis of the convergence speed.

2.2.3 Extension to deconvolution

We now consider the deconvolution problem, with the following model:

$$E(u) = \frac{1}{2\mu} \|A_{\text{a}} - f\|^2 + \int_{\Omega} |Du|,$$

where $A$ is a blur operator (consider here a Gaussian kernel). It can be proved that the following scheme converges to $u$ when the step-size $\nu$ satisfies $\nu\|A^*A\| \leq 1$:

$$\nu^n = u^n + \nu A^*(f - Au^n)$$

$$\nu^{n+1} = \arg \min_u \left\{ \frac{1}{2\mu} \|\nu^n - u\|^2 + \int |Du| \right\}$$

Implement this minimization procedure using a projected gradient method. Compare its convergence speed to the one obtained by directly minimization of the approximated differentiable total variation using a quasi-Newton algorithm.

2.3 Nesterov’s algorithm

Nesterov’s algorithm is particularly efficient. For the minimization of $E(u)$ it takes the form:

Initialization: $k = 0, \nu^0 = 0, x^0 = 0, L = 8\mu$.

Repeat until stopping criteria:

1. Set $k = k + 1$, and compute $\eta^k = -\nabla(f - \mu \text{div}(x^k))$. 3
2. Set $y^k = P_K(x^k - \eta^k/L)$, with $K = \{x \in (L^2)^2 | \|x\| \leq 1\}$.

3. Set $v^k = v^{k-1} + \frac{k+1}{k+2} \eta^k$.

4. Set $z^k = P_K(-v^k/L)$.

5. Set $x^{k+1} = \frac{2}{k+1} z^k + \frac{k+1}{k+2} y^k$.

Output: $u = f - \mu \text{div}(y^{\text{lim}})$.

Implement this method. Compare its convergence speed with the ones of the other methods explored in this problem set.