Exercises: First-Order Algorithms for Convex Optimization

Lecture 1.

- Give two different proofs for the Cauchy-Schwartz inequality

$$|\langle x, y\rangle| \leq \|x\| \cdot \|y\|.$$

是否为二范数

- Show that if $f(x) = \frac{1}{2}x^{\mathrm{T}}Ax + b^{\mathrm{T}}x + c$ then

A是否为正定矩阵

$$\nabla f(x) = Ax + b \text{ and } \nabla^2 f(x) = A.$$

Lecture 2.

- Compute $(\mathbf{R}^n_+)^*$.
- If $f(x) = \frac{1}{2}x^{\mathrm{T}}Ax + b^{\mathrm{T}}x + c$ where $A \succ 0$, compute its conjugate function

$$f^*(s) = \sup\{s^{\mathrm{T}}x - f(x) \mid x \in \mathrm{dom}\,(f)\},$$

and verify that the Fenchel-Moreau theorem holds: $(f^*)^* = f$.

Lecture 3.

- For conic optimization

$$\begin{array}{ll} \min & c^\top x \\ \text{s.t.} & Ax = b \\ & x \in \mathcal{K}, \end{array}$$

where $\mathcal{K}$ is a closed convex cone. Show that its Lagrangian dual problem is

$$\begin{array}{ll} \max & b^\top y \\ \text{s.t.} & A^\top y + s = c \\ & s \in \mathcal{K}^*. \end{array}$$

- Let $\Delta = \{x \mid x \in \mathbf{R}^n_+, e^\top x = 1\}$ where e is an n-dimensional all-one vector. Suppose that $v \notin \Delta$. Let us consider the projection problem:

$$\begin{array}{ll} \min & \frac{1}{2}\|x - v\|^2 \\ \text{s.t.} & x \in \Delta. \end{array}$$

Derive the dual of the above projection problem.

Lecture 4.

- The convexified compressive sensing model is

$$\begin{array}{ll} \min & \|x\|_1 \\ \text{s.t.} & Ax = b. \end{array}$$

Show that the above model can also be solved by linear programming.

- On page 15 of Lecture 4, the foreground and background decomposition problem is formulated as

$$\begin{array}{ll} \min & \text{rank}(X) + \mu\|Y\|_0 + \nu\|Z\|^2 \\ \text{s.t.} & D = X + Y + Z. \end{array}$$

Can you speculate how its convexified version should look like?

Lecture 5.

- Suppose that f is strongly convex. Show that the R-linear (respectively sublinear) convergence of $\|x^k - x^*\|$ and $f(x^k) - f(x^*)$ are equivalent.
- Use Matlab to do the following test.

Randomly generate $A \in \mathbf{R}^{100\times 500}$ and $b = Ae$. Write Matlab code for a simple (sub)-gradient algorithm to solve the lasso problem

$$\min \quad \|x\|_1 + \frac{\gamma}{2}\|Ax - b\|^2,$$

with diminishing steps $\alpha_k = 1/k$ for 1,000 steps. (Set $\gamma = 1$). Report your solution.

Lecture 6.

- Use Matlab to write a code the ISTA procedure to solve the previous lasso problem

$$\min \quad \|x\|_1 + \frac{\gamma}{2}\|Ax - b\|^2$$

for 1,000 steps. Report your solution.

- Consider now an over-determined linear system $Gx = h$. Let $f(x) = \frac{1}{2}\|Gx - h\|^2$. How to solve the least squares problem by the proximal point method?

Lecture 7.

- Use Matlab to code the FISTA procedure to solve the previous lasso problem

$$\min \quad \|x\|_1 + \frac{\gamma}{2}\|Ax - b\|^2$$

for 1,000 steps. Report your solution and compare the result with ISTA.

- The lasso problem can be equivalently formulated as

$$\begin{array}{ll} \min & \frac{\gamma}{2}\|Ax - b\|^2 + \|y\|_1 \\ \text{s.t.} & x - y = 0. \end{array}$$

Illustrate how to use ADMM to solve the lasso problem.