Mathematical and Numerical Optimization, 3341.454
E. Ryu
Fall 2020

Additional Problems for Homework 4

Problem 1: *Heuristic suboptimal solution for Boolean LP.* This exercise builds on exercises 4.15 and 5.13 in *Convex Optimization*, which involve the Boolean LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & x_i \in \{0,1\}, \quad i = 1, \dots, n, \end{array}$$

with optimal value $p^\star$. Let x^{rlx} be a solution of the LP relaxation

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \\ & 0 \preceq x \preceq \mathbf{1}, \end{array}$$

so $L = c^T x^{\text{rlx}}$ is a lower bound on $p^\star$. The relaxed solution x^{rlx} can also be used to guess a Boolean point $\hat{x}$, by rounding its entries, based on a threshold $t \in [0,1]$:

$$\hat{x}_i = \begin{cases} 1 & x_i^{\text{rlx}} \geq t \\ 0 & \text{otherwise}, \end{cases}$$

for $i = 1, \dots, n$. Evidently $\hat{x}$ is Boolean (i.e., , has entries in $\{0,1\}$). If it is feasible for the Boolean LP, i.e., , if $A\hat{x} \preceq b$, then it can be considered a guess at a good, if not optimal, point for the Boolean LP. Its objective value, $U = c^T \hat{x}$, is an upper bound on $p^\star$. If U and L are close, then $\hat{x}$ is nearly optimal; specifically, $\hat{x}$ cannot be more than $(U - L)$-suboptimal for the Boolean LP.

This rounding need not work; indeed, it can happen that for all threshold values, $\hat{x}$ is infeasible. But for some problem instances, it can work well.

Of course, there are many variations on this simple scheme for (possibly) constructing a feasible, good point from x^{rlx}.

Finally, we get to the problem. Generate problem data using one of the following.

```
import numpy as np
np.random.seed(0)
(m, n) = (300, 100)
A = np.random.rand(m, n)
b = A@np.ones((n, 1))/2
c = -np.random.rand(n, 1)
```

You can think of x_i as a job we either accept or decline, and $-c_i$ as the (positive) revenue we generate if we accept job i. We can think of $Ax \preceq b$ as a set of limits on m resources. A_{ij}, which is positive, is the amount of resource i consumed if we accept job j; b_i, which is positive, is the amount of resource i available.

Find a solution of the relaxed LP and examine its entries. Note the associated lower bound L. Carry out threshold rounding for (say) 100 values of t, uniformly spaced over $[0,1]$. For each value of t, note the objective value $c^T\hat{x}$ and the maximum constraint violation $\max_i(A\hat{x}-b)_i$. Plot the objective value and the maximum violation versus t. Be sure to indicate on the plot the values of t for which $\hat{x}$ is feasible, and those for which it is not.

Find a value of t for which $\hat{x}$ is feasible, and gives minimum objective value, and note the associated upper bound U. Give the gap $U-L$ between the upper bound on $p^\star$ and the lower bound on $p^\star$.

Problem 2: Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & -\log\det X + \mathrm{trace}(SX) \\ \text{subject to} & X \text{ is tridiagonal} \end{array}$$

with domain $\mathbf{S}^n_{++}$ and variable $X \in \mathbf{S}^n$. The matrix $S \in \mathbf{S}^n$ is given. Assume a solution X_{opt} exists. Show that it satisfies

$$(X_{\mathrm{opt}}^{-1})_{ij} = S_{ij}, \quad |i-j| \le 1.$$

Problem 3: Assume $\mathbb{T}_1 \colon \mathbb{R}^n \to \mathbb{R}^m$ is L_1-Lipschitz and $\mathbb{T}_2 \colon \mathbb{R}^n \to \mathbb{R}^m$ is L_2-Lipschitz. Show that $\alpha_1\mathbb{T}_1 + \alpha_2\mathbb{T}_2$ is $(|\alpha_1|L_1 + |\alpha_2|L_2)$-Lipschitz.

Problem 4: Show that if f is convex, then $\mathrm{dom}\, f$ is convex.

Problem 5: Let f be a CCP function on $\mathbb{R}^n$. Show that $\partial f(x)$ is a closed convex set for all $x \in \mathbb{R}^n$.

Hint. Write $\partial f(x)$ as an intersection of closed half-spaces.

Remark. Remember that $\partial f(x)$ can be empty, but the empty set is a closed convex set.

Problem 6: Show that if f is a CCP function on $\mathbb{R}^n$, $A \in \mathbb{R}^{m\times n}$, and $g(x) = f(Ax)$ then

$$\partial g(x) \supseteq A^\intercal \partial f(Ax)$$

for all $x \in \mathbb{R}^n$. Also show that if f and g be CCP functions on $\mathbb{R}^n$, then

$$\partial(f+g)(x) \supseteq \partial f(x) + \partial g(x)$$

for all $x \in \mathbb{R}^n$.

Problem 7: Consider the function $f \colon \mathbb{R}^2 \to \mathbb{R} \cup \{\pm\infty\}$ defined as

$$f(x,y) = \begin{cases} x^2/y & \text{for } y > 0, \\ 0 & \text{for } x = y = 0, \\ \infty & \text{otherwise.} \end{cases}$$

Clearly f is proper, and it is possible to show that f is convex. Show that

(a) f is closed, and

(b) $f|_{\mathrm{dom}\, f} : \mathrm{dom}\, f \to \mathbb{R}$ is not continuous at $(0,0)$, i.e., show that f restricted to where it is finite is not continuous at $(0,0)$.

Remark. This example demonstrates that a CCP function need not be continuous on its domain. In convex optimization, lower semi-continuity, not continuity, is the regularity condition of interest. (A convex function is continuous on the relative interior of its domain.)

Problem 8: *Linear programming duality.* Consider the convex-concave saddle function

$$\mathbf{L}(x, \nu, \mu) = \langle c, x\rangle + \langle Ax + b, \nu\rangle - \langle x, \mu\rangle - \delta_{\mathbb{R}^m_+}(\nu) - \delta_{\mathbb{R}^n_+}(\mu),$$

convex in $x \in \mathbb{R}^n$ and concave in $(\nu, \mu) \in \mathbb{R}^m \times \mathbb{R}^n$. Here, $\mathbb{R}^m_+$ and $\mathbb{R}^n_+$ denote the m and n-dimensional nonnegative orthants. Remember that δ_C denotes the indicator function with respect to the set C.

Show that the saddle function $\mathbf{L}$ generates the primal problem

$$\begin{array}{ll} \underset{x \in \mathbb{R}^n}{\text{minimize}} & c^\mathsf{T} x \\ \text{subject to} & Ax + b \leq 0 \\ & x \geq 0. \end{array}$$

Here, the inequalities denote element-wise nonnegativity. Show that $\mathbf{L}$ generates a dual problem that is equivalent to

$$\begin{array}{ll} \underset{\nu \in \mathbb{R}^m}{\text{maximize}} & b^\mathsf{T} \nu \\ \text{subject to} & c + A^\mathsf{T} \nu \geq 0 \\ & \nu \geq 0. \end{array}$$