Convex Optimization Homework #3

Posted Friday May 27, 2022 (ver 1), Revised May 29 (ver 2) and June 8 (ver 3)
Due: Monday June 13, 2022, 11am.

1. (100%) Consider the semidefinite optimization problem

$$\begin{aligned} \underset{x \in \mathbf{R}^n}{\text{minimize}} \quad & f_0(x) = c^T x \\ \text{subject to} \quad & f_1(x) = G + \sum_{i=1}^{n} x_i F_i \preceq O, \end{aligned} \tag{1}$$

where $c \in \mathbf{R}^n$, $G, F_1, ..., F_n \in \mathbf{S}^n$, $f_0 : \mathbf{R}^n \to \mathbf{R}$, and $f_1 : \mathbf{R}^n \to \mathbf{S}^n$.

(a) (5%) Determine if Problem (1) is a convex problem. Justify your answer.

(b) (5%) Write the Lagrangian of the problem $L(x, Z)$ where $Z \in \mathbf{S}^n$ is the Langrange multiplier associated with the generalized inequality constraint.

(c) (5%) Find the Lagrange dual function $g(Z)$ and its domain **dom** g.

(d) (5%) Formulate the Lagrange dual problem. Determine if it is a convex problem.

(e) (7%) If Problem (1) is convex, please take the parameters defined in Table 1 and use `CVX` to solve the problem. Write down the **optimal value** and the **optimal point** you obtained from `CVX`. Denote them as $p^*_{\texttt{CVX}}$ and $x^*_{\texttt{CVX}}$, respectively.

(f) (5%) Find a generalized logarithm $\psi()$ that is concave, close, twice continuously differentiable, with $\textbf{dom}\ \psi = \textbf{int}\ \mathbf{S}^n_+$, and for all $s > 0$, $\psi(sy) = \psi(y) + \theta \log s$. Find the *degree* θ of the ψ function you chose.

(g) (5%) Formulate the centrality problem with parameter t, where the objective function is written as $f(x) = tf_0(x) - \psi(-f_1(x))$. Find the domain of f.

(h) (8%) Find the gradient $\nabla f(x)$ and Hessian $\nabla^2 f(x)$, of f, respectively.
(Hint: It might be useful to note that if A is an invertible matrix whose entries are functions of a, then $\frac{\partial A^{-1}}{\partial a} = -A^{-1} \frac{\partial A}{\partial a} A^{-1}$.)

(i) (5%) Write a matlab m-file as a function which takes inputs t, c, F_1, F_2, F_3, G, and x, and evaluates the objective function at the point x, as well as the gradient and the Hessian.
Hint: the function can have a header that looks like the following, where `F` is a three-way array.

```
function [f, g, H] = my_objective(x, t, c, F, G)
```

(j) (5%) From this step on, write another m-file as the main function for your program to solve the centrality problem. Set c, F_i's, and G as in Table 1 and set the initial point $x^{(0)} = 0$. *Hint: the function may be declared something simliar to the following. You don't have to use exactly the same names of the function parameters, though.* [1]

```
function [x_central, lambda_2_2] = centrality_problem(x_init, t, c, F, G, alpha, beta)
```

(k) (5%) Use the Newton step's formula: $\Delta x_{\text{nt}} = -(\nabla^2 f(x))^{-1} \nabla f(x)$ and calculate the Newton step $\Delta x^{(k)}_{\text{nt}}$ for the kth inner iteration.

(l) (5%) Calculate the Newton decrement $\lambda^{(k)}(x) = (-\nabla f(x)^T \Delta x^{(k)}_{\text{nt}})^{1/2}$. Store the values of ${\lambda^{(k)}}^2/2$ of each iteration for future use.

(m) (5%) Perform backtracking line search along search direction using $\beta = 0.7$ starting from $s = 1$ until $f(x + s\Delta x^{(k)}_{\text{nt}}) \leq f(x) - \alpha s \lambda^{(k)}(x)^2$, where $\alpha = 0.1$. [2]

(n) (5%) Perform the update $x^{(k+1)} = x^{(k)} + s\Delta x^{(k)}_{\text{nt}}$. Determine whether $\lambda^2/2 \leq \epsilon$ where $\epsilon_{inner} = 10^{-8}$. Terminate the iteration loop if it is true.

[1] The m-file for solving the centrality problem shall return the central point as will as a list of $\lambda^2/2$ for each iteration.

[2] We use s to denote the backtracking parameter instead of t because t has been used as the centrality problem's parameter.

(o) (5%) In your main script m-file, set $\mu = 2$ and $t^{(0)} = 1$. Write a loop to solve the centrality problem for t, starting from $t = t^{(0)}$, by calling the function you wrote from (j) to (n). Terminate this outer loop if $n/t < \epsilon_{outer}$ where $\epsilon_{outer} = 10^{-8}$. When an outer loop iteration is done, do not reset the inner loop index k, letting it represent the total number of Newton steps being run at any given moment. Use l to denote the outer loop index and denote the centrality point of the lth outer loop as $x^*(t^{(l)})$ where $t^{(l)} = \mu^t \cdot t^{(0)}$ is the centrailty problem's parameter in the lth outer iteration, $l \geq 0$.

(p) (5%) Compare the optimal value p^* and optimal point x^* you obtained with those obtained from `CVX` as in (e). Specifically, calculate $|p^* - p^*_{\texttt{CVX}}|$ and $||x^* - x^*_{\texttt{CVX}}||_2$, respectively.

(q) (5%) Plot $f_0(x^*(t^{(l)})) - p^*$ versus l in log-scale.

(r) (5%) Plot ${\lambda^{(k)}}^2/2$ versus k in log-scale.

(s) (5%) Submit your source code as `hw3.m` or `hw3.py`.

Parameters	n	F_1	F_2		
	3	$\begin{bmatrix} 0.9 & -0.4 & 0.6 \\ -0.4 & 3.9 & -1.0 \\ 0.6 & -1.0 & 2.7 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0.8 & -1.0 \\ 0.8 & 1.7 & -2.6 \\ -1.0 & -2.6 & 5.4 \end{bmatrix}$		
Parameters	c	F_3	G		
	$\begin{bmatrix} -4 \\ -2 \\ -4 \end{bmatrix}$	$\begin{bmatrix} 0.2 & 0.1 & 0.6 \\ 0.1 & 0.6 & -0.1 \\ 0.6 & -0.1 & 3.3 \end{bmatrix}$	$\begin{bmatrix} -1.2 & 0.5 & -1.2 \\ 0.5 & -1.4 & -1.0 \\ -1.2 & -1.0 & -3.3 \end{bmatrix}$		

Table 1: Parameters for the SDP Problem

Guidelines of Homework Submission:

- You are allowed to discuss with other students, ask for hints from the TAs. But you have to write your answers and argument solely on your own, without looking at any part of anyone else's answers. Sharing your written (or typed) answers with others is *strongly prohibited.* Both parties will get a zero-score penalty for this mis-conduct.

- Submit your answer online as a document file (in *.pdf only) that contains all answers in this problem set.

- Submit your files online onto the NTU Cool system. No paper shall be handed in. If needed, you can choose to write (sketch) your answers on a sheet first and convert the image(s) to a single pdf file.

- Late submissions will be processed according to the following rules.

  (1) Homework received by 11am, June 13 (t_1) will be counted in full.

  (2) Homework received after 2pm, June 13 (t_2) will not be counted.

  (3) Homework received between t_1 and t_2 will be counted with a discount rate

  $$\frac{t_2 - t}{t_2 - t_1}$$

  where t is the submission time. Note that $t_2 - t_1$ is three hours.