Problem Set 4
Convex Optimization
Department of Electrical Engineering
Sharif University of Technology

Instructor: M. Babazadeh

Spring Semester 1397–98
Deadline: Saturday–25 Khordad

Problems from textbook

Solve the following problems from [1]. 11.1, 11.4, 11.6, 11.10.

References

[1] Boyd, Stephen, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.

Matlab Assignment

As discussed in class, a large number of primal-dual interior point methods (IPM) are available whose practical efficiency strongly depends on the techniques used to solve their Newton equation system. In this assignment, you will implement a very simple version of interior point method for QP,

$$\text{Minimize} \quad \frac{1}{2}x^T Q x + c^T x \tag{1}$$
$$\text{Subject to} \quad Ax = b, \quad x \geq 0 \quad A \in \mathbb{R}^{m \times n}, \quad Q \succeq 0.$$

(a) Derive the KKT conditions of (1) with lagrange multipliers s (for inequality constraint) and y (for equality constraint).

(b) Use logarithmic barrier to eliminate inequality constraints, and rewrite the KKT conditions for the new problem. Hint: Your answer should be in the form given below, where X and S are diagonal matrices with elements of vectors x and s spread across the diagonal, respectively and $e \in \mathbb{R}^n$ is the vector of ones. You see that the only difference with part (a) is a perturbation of the complementarity slackness.

$$Ax = b,$$
$$A^T y + s - Qx = c,$$
$$XSe = \mu e,$$
$$(x, s) \geqslant 0.$$

(c) Use the procedure presented in class to show that if (x, y, s) is primal and dual feasible then the duality gap in the barrier subproblem is equal to $x^T s = n\mu$, i.e., the term μ controls the distance to optimality.

(d) Primal-dual IPM applies the Newton method to solve the KKT conditions in part (b). To be more precise, it compute the Newton direction and make one step in this direction before reducing the barrier parameter μ. The Newton direction $(\Delta x, \Delta y, \Delta s)$ is computed by solving the following system of linear equations:

$$
\begin{bmatrix} A & 0 & 0 \\ -Q & A^T & I_n \\ S & 0 & X \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} \xi_p \\ \xi_d \\ \xi_\mu \end{bmatrix} = \begin{bmatrix} b - Ax \\ c + Qx - A^T y - s \\ \sigma \mu e - XSe \end{bmatrix},
$$

Accordingly, the following simple primal-dual IPM for convex quadratic programming is presented. Implement the given algorithm and test your code for a number of random quadratic programs.

(e) The algorithm must guarantee that the error in equation $XSe = \mu e$ is small. Plot $\|XSe - \mu e\|$ versus iteration number.

Parameters

$\alpha_0 = 0.99$ a fraction-to-the-boundary stepsize factor;
$\sigma \in (0,1)$ barrier reduction parameter;
ε_p, ε_d, ε_o primal feasibility, dual feasibility and optimality tolerances:

IPM stops when $\frac{\|\xi_p^k\|}{1+\|b\|} \leqslant \varepsilon_p$, $\frac{\|\xi_d^k\|}{1+\|c\|} \leqslant \varepsilon_d$ and

$\frac{(x^k)^T s^k /n}{1+|c^T x^k + 1/2(x^k)^T Q x^k|} \leqslant \varepsilon_o$.

Initialize IPM

iteration counter $k = 0$;
primal–dual point $x^0 > 0$, $y^0 = 0$, $s^0 > 0$;
barrier parameter $\mu^0 = (x^0)^T s^0 / n$;
primal and dual infeasibilities $\xi_p^0 = b - Ax^0$ and
$\xi_d^0 = c - A^T y^0 - s^0 + Q x^0$.

Interior Point Method

while $\left(\frac{\|\xi_p^k\|}{1+\|b\|} > \varepsilon_p \text{ or } \frac{\|\xi_d^k\|}{1+\|c\|} > \varepsilon_d \text{ or } \frac{(x^k)^T s^k /n}{1+|c^T x^k + 1/2(x^k)^T Q x^k|} > \varepsilon_o \right)$ **do**

    Update (reduce) the barrier $\mu^{k+1} = \sigma \mu^k$;
    Solve the KKT system (d): find the primal–dual Newton direction $(\Delta x, \Delta y, \Delta s)$.
    Find $\alpha_P = \max\{\alpha : x^k + \alpha \Delta x \geqslant 0\}$ and
$\alpha_D = \max\{\alpha : s^k + \alpha \Delta s \geqslant 0\}$;
    Set $\alpha_P := \alpha_0 \alpha_P$ and $\alpha_D := \alpha_0 \alpha_D$;
    Make step
      $x^{k+1} = x^k + \alpha_P \Delta x$;
      $y^{k+1} = y^k + \alpha_D \Delta y$;
      $s^{k+1} = s^k + \alpha_D \Delta s$.
    Compute the infeasibilities: $\xi_p^{k+1} = b - Ax^{k+1}$ and
$\xi_d^{k+1} = c - A^T y^{k+1} - s^{k+1} + Q x^{k+1}$;
    Update the iteration counter: $k := k + 1$.
end-while