L. Vandenberghe
T.L. Jensen

Ph.D. Course
Algorithms for Large-Scale Convex Optimization
DTU, August 23-27, 2010

Exercise session 5

Interior-point algorithm for ℓ_1-norm approximation. In this exercise we develop a Matlab program for the ℓ_1-norm approximation problem

$$\text{minimize} \quad \|Pu + q\|_1, \tag{1}$$

where $P = \mathbf{R}^{r\times t}$, $q \in \mathbf{R}^r$, and the variable is an t-vector u. On exit, the code must guarantee a relative accuracy of 10^{-6} or an absolute accuracy of 10^{-8}, *i.e.*, the code can terminate if

$$\|Pu + q\|_1 - p^\star \leq 10^{-6} \cdot p^\star$$

or

$$\|Pu + q\|_1 - p^\star \leq 10^{-8},$$

where $p^\star$ is the optimal value of (1). You can assume that P has full rank ($\mathbf{rank}\, P = t$).

We solve the problem using the primal-dual path-following method described in lecture 5, page 23, applied to the LP

$$\begin{array}{ll} \text{minimize} & \mathbf{1}^T v \\ \text{subject to} & \begin{bmatrix} P & -I \\ -P & -I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \preceq \begin{bmatrix} -q \\ q \end{bmatrix}. \end{array} \tag{2}$$

We will take advantage of the structure in the coefficient matrix to improve the efficiency.

1. *Initialization.* The algorithm can be started at infeasible primal and dual points. However good feasible starting points for the LP (2) are readily available from the solution u_{ls} of the least-squares problem

   $$\text{minimize} \quad \|Pu + q\|_2$$

   As primal starting point we can use $u = u_{\text{ls}}$, and choose v so that we have strict feasibility in (2). To find a strictly feasible point for the dual of (2), we note that $P^T P u_{\text{ls}} = -P^T q$ and therefore the least-squares residual $r_{\text{ls}} = Pu_{\text{ls}} + q$ satisfies

   $$P^T r_{\text{ls}} = 0.$$

   This property can be used to construct a strictly dual feasible point for (2), with a positive dual objective value (if $r_{\text{ls}} \neq 0$). Since the starting points are strictly feasible, all iterates in the algorithm will remain strictly feasible.

2. The most expensive in each iteration of the algorithm is the solution of two sets of equations of the form

$$G^T W^{-2} G \Delta x = r_1 \tag{3}$$

where (for an LP) W is a positive diagonal matrix (see page 5-28). In our application, (3) has $r+t$ equations in $r+t$ variables, since

$$G = \begin{bmatrix} P & -I \\ -P & -I \end{bmatrix}, \qquad \Delta x = \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix}.$$

Exploit the structure of G to show that you can solve systems of the form (3) by solving a smaller system of the form

$$P^T D P \Delta u = r_2, \tag{4}$$

followed by a number of inexpensive operations. In (4) D is an appropriately chosen positive diagonal matrix.

3. Test your code on randomly generated P and q.