Optimization Theory and Algorithms

Solving LP Barrier Systems by Newton's Method

Problem

For given data (A, b, c), where $A \in \mathbb{R}^{m \times n}$ $(m < n)$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$, consider solving the following system of equations, for a parameter $\mu > 0$, by Newton's method,

$$F_\mu(x, y, z) = \begin{pmatrix} A^T y + z - c \\ Ax - b \\ x_1 z_1 - \mu \\ x_2 z_2 - \mu \\ \vdots \\ x_n z_n - \mu \end{pmatrix} = 0, \tag{1}$$

where the variables are $x, z \in \mathbb{R}^n$, which should be both positive (note $\mu > 0$), and $y \in \mathbb{R}^m$. This system is called the barrier system for a particular form of linear program (LP).

At any fixed (x, y, z), Newton method solves the linear system of equations for the step (dx, dy, dz):

$$F'_\mu(x, y, z) \begin{pmatrix} dx \\ dy \\ dz \end{pmatrix} = \begin{pmatrix} r_d \\ r_p \\ r_c \end{pmatrix}, \tag{2}$$

where $F'_\mu(x, y, z)$ is the Jacobian matrix of F_μ evaluated at (x, y, z), and the right-hand side is just $-F_\mu(x, y, z)$ devided into 3 sub-vectors with $r_d, r_c \in \mathbb{R}^n$ and $r_p \in \mathbb{R}^m$ (see (1)).

Derivations

- Derive a concise expression for the Jacobian matrix $F'_\mu(x, y, z)$. The matrix will be very sparse, with a specific block structure.
- To solve the linear system in (2) efficiently, derive a block Gaussian elimination scheme in which the variables dz and dx are eliminated, leading to a smaller linear system for dy only. After solving the small system for dy, then dx and dz are recovered by back substitutions.

Please typeset your derived formulas in LaTeX.

Matlab

- Write a Matlab function to solve the linear system in (2), at any given positive x, z, and given y:

  ```
  [dx,dy,dz] = mylinsolve(A,rd,rp,rc,x,z);
  ```

  in which you should implement the block Gaussian elimination scheme derived above.
- Download the file `handout_barrier.zip` and run `test_barrier.m` (with or without your code).
- Submit your code and the outputs for 2 runs: `p = 1` and `p = 4` (or 3 if your code cannot handle 4).