Homework Assignment 3
Discussed Friday February 28, 2014

Optional Reading. Read selected sections in Luenberger and Ye’s Linear and Nonlinear Programming Third Edition Chapters 8, 9 and 10.

Solve the following problems:

1. 8.6 of LY.

2. 8.24 of LY.

3. Prove (1) of slide 2 of Lecture Note 12.

4. In Logistic Regression, we like to determine $x_0$ and $x$ to maximize

$$
\left( \prod_{i,c_i=1} \frac{1}{1 + \exp(-a_i^T x - x_0)} \right) \left( \prod_{i,c_i=-1} \frac{1}{1 + \exp(a_i^T x + x_0)} \right),
$$

which is equivalent to maximize the log-likelihood probability

$$
-\sum_{i,c_i=1} \log (1 + \exp(-a_i^T x - x_0)) - \sum_{i,c_i=-1} \log (1 + \exp(a_i^T x + x_0)).
$$

Or to minimize the log-logistic-loss

$$
\sum_{i,c_i=1} \log (1 + \exp(-a_i^T x - x_0)) + \sum_{i,c_i=-1} \log (1 + \exp(a_i^T x + x_0)).
$$

a) Write down the gradient vector function of the log-logistic-loss function.

b) Consider the specific problem

$$
f(x) = \log (1 + \exp(-x_1 - 2x_2 - x_0)) + \log (1 + \exp(-2x_1 - x_2 - x_0)) + \log (1 + \exp(x_0)) .
$$

Use the accelerated steepest descent and the BB methods to solve the problem in Matlab.
5. Consider the following bounded polytope in $\mathbb{R}^m$ represented by $n > m$ linear inequalities:

$$\Omega = \{ y \in \mathbb{R}^m : c - A^T y \geq 0 \}$$

where $A \in \mathbb{R}^{m \times n}$ and $c \in \mathbb{R}^n$ are given and $A$ has rank $m$. Denote the interior of $\Omega$ by:

$$\Omega^o = \{ y \in \mathbb{R}^m : c - A^T y > 0 \}$$

The logarithmic barrier function for $\Omega$ is given by:

$$B(y) = -\sum_{j=1}^n \log(c_j - a_j^T y)$$

where $a_j$ is the $j$-th column of $A$.

a) Derive the gradient and Hessian of $B(y)$.

Let $\eta_d(y)^2 = \nabla B(y)^T (\nabla^2 B(y))^{-1} \nabla B(y)$, and let $S$ be the diagonal matrix of the slack vector $s = c - A^T y$. Given some $y \in \Omega^o$, we call it an $\eta$-approximate analytic center if $\eta_d(y) \leq \eta < 1$. The Newton procedure would start from some $y \in \Omega^o$ and compute the Newton step via:

$$d_y = -(AS^{-2}A^T)^{-1}AS^{-1}e$$

It then updates the iterate via:

$$y^+ := y + d_y$$

b) Show that if the starting $y$ has $\eta_d(y) < 1$, then we have:

$$s^+ = c - A^T y^+ > 0 \quad \text{and} \quad \eta_d(y^+) \leq \eta_d(y)^2$$

c) Suppose that one applies the above Newton procedure with the update rule:

$$y^+ = y + \frac{\alpha}{\eta_d(y)} d_y$$

where $\alpha \in (0, 1)$ is some constant. Show that if $\eta_d(y) \geq 3/4$, then we have:

$$B(y^+) - B(y) \leq -\delta$$

where:

$$\delta = \frac{3\alpha}{4} - \frac{\alpha^2}{2(1 - \alpha)} > 0.$$
d) Implement this algorithm in Matlab or any other framework and run some simulations for randomly generated $A$ and $c$. For example

```matlab
A=rand(m,n);
x=ones(n,1);
y=0*ones(m,1);
c=2*ones(n+1,1)+rand(n+1,1);
b=A*x;
A=[A -b];
```

Then, the polytope defined by $A$ and $c$ will be bounded and $y = 0$ is an interior point close to the analytic center.