CME 307 / MS&E 311 Winter 2016-2017
Optimization Feb 10, 2017
Prof. Yinyu Ye Due Thursday Feb 23 5pm

Homework Assignment 3
Discuss Session Friday Feb 24 in Class

Reading. Read selected sections in Luenberger and Ye's *Linear and Nonlinear Programming Fourth Edition* Chapters 3, 8, 9 and 10.

Solve the following problems:

1. 8.6 in text book L&Y.

   Consider
   $$x_{k+1} = \frac{1}{2}[x_k + \frac{a}{x_k}]$$
   Assuming this sequence converges, what does it converge two and what is the order of convergence.

Since the sequence converges we have that $\lim_{k\to\infty} |x_{k+1} - x_k| = 0$. Thus we have

$$\lim_{k\to\infty} |x_{k+1} - x_k| = 0$$
$$\lim_{k\to\infty} |\frac{1}{2}[\frac{a - x_k^2}{x_k}]| = 0$$
$$\lim_{k\to\infty} |[\frac{a - x_k^2}{x_k}]| = 0$$

So the $\lim_{x_k\to\infty} = \pm\sqrt{a}$, where $a > 0$.
Let $g(x) = \frac{1}{2}x - \frac{a}{2x}$. Then $g(x^*) = x^*$. Note that $g'(x) = \frac{1}{2} - \frac{a}{2x^2}, g''(x) = \frac{1}{x^3}$. Also note that $g'(x^*) = 0$, and $g\prime(x^*) = \frac{1}{\pm\sqrt{a}}$, since $x^* = \pm\sqrt{a}$ Taylor expanding $g(x)$

$$g(x_k) = g(x^*) - g'(x^*)(x_k - x^*) + \frac{(x_k - x^*)^2}{2} f''(\psi)$$

where ψ is in the region $[x_k, x^*]$. Now since $g'(x^*) = 0$. we have

$$g(x_k) = g(x^*) + \frac{(x_k - x^*)^2}{2} f''(\psi)$$

Letting the successive error $e_k = x_k - x^*$. we have

$$e_{k+1} = g(x_{k+1}) - x^* = \frac{(e_k^2)^2}{2} f''(\psi)$$

Each successive error term is proportional to the square of the previous error, thus the method has quadratic convergence.

---

2. Prove Lemma 1 of Lecture Note 11. (Hint: using the Taylor Expansion Theorem.)

I am going to prove this slightly differently than the typical taylor expansion proof. First note that if $f(x)$ is β-Lipschitz, we have

$$||\nabla f(x) - \nabla f(y)|| \leq \beta ||x - y||$$

For $\beta > 0$. Now let $z(t) = f(y + t(x - y))$ so that $z(0) = f(y)$ and $z(1) = f(x)$. Furthermore note that

$$g'(t)|_{t=0} = \nabla f(y + t(x - y))^T (x - y)$$

and that $\int_0^1 g'(t)dt = g(1) - g(0)$. Then it follows that

$$\begin{aligned}
|f(x) - f(y) - \nabla f(y)^T (x - y)| &= |\int_0^1 g'(t)dt - \nabla f(y)^T (x - y)| \\
&= |\int_0^1 \nabla f(y + t(x - y))^T (x - y)dt - \nabla f(y)^T (x - y)| \\
&\leq \int_0^1 |\nabla f(y + t(x - y))^T (x - y) - \nabla f(y)^T (x - y)|dt \\
&\leq \int_0^1 ||\nabla f(y + t(x - y)) - \nabla f(y)||||x - y||dt \text{ by Cauchy Schwartz} \\
&\leq \int_0^1 \beta t||x - y||^2 dt \text{ by } \beta\text{- Lipschitz} \\
&\leq \beta \frac{||x - y||^2}{2} dt
\end{aligned}$$

and thus the proof is complete.

---

3. Using the simplex method to solve 3.10 in text book L&Y, assuming that you start from corner origin (there is two typos in the text book where: "minimize" should be :maximize" and "$x_1 \geq 0$" should be "$x_i \geq 0$".

We have the problem

$$\begin{aligned}
&\text{maximize } 2x_1 + 4x_2 + x_3 + x_4 \\
&\text{such that: } x_1 + 3x_2 + x_4 \leq 4 \\
&\qquad\qquad 2x_1 + x_2 \leq 3 \\
&\qquad\qquad x_2 + 4x_3 + x_4 \leq 3 \\
&\qquad\qquad x_i \geq 0
\end{aligned}$$

Introducing the slack variables w_1, w_2, w_3 we have

$$\begin{aligned}
&\text{maximize } 2x_1 + 4x_2 + x_3 + x_4 \\
&\text{such that:} \\
&\qquad\qquad w_1 = 4 - x_1 - 3x_2 - x_4 \\
&\qquad\qquad w_2 = 3 - 2x_1 - x_2 \\
&\qquad\qquad w_3 = 3 - x_2 - 4x_3 - x_4 \\
&x_i, w_i \geq 0
\end{aligned}$$

Now not our basic set $B = \{w_1, w_2, w_3\}$ is feasible. Noticing that the objective increases most rapidly with x_2, we pivot x_2 into the basis and w_1 out of the basis, yielding $B = \{x_2, w_2, w_3\}, NB = \{w_1, x_1, x_3, x_4\}$.

$$\text{maximize } \frac{16}{3} + \frac{1}{3}x_1 - \frac{4}{3}w_1 + x_3 - \frac{1}{3}x_4$$

such that:

$$x_2 = \frac{1}{3}[4 - x_1 - w_1 - x_4]$$
$$w_2 = \frac{1}{3}[5 - 5x_1 + w_1 + x_4]$$
$$w_3 = \frac{1}{3}[5 + x_1 + w_1 - 12x_3 - 2w_4]$$
$$x_i, w_i \geq 0$$

Now pivoting x_3 into the basis and w_3 out of the basis $B = \{x_2, w_2, w_3\}, NB = \{x_1, x_3, x_4, w_1\}$.

$$\text{maximize } 5.75 + .75x_1 - \frac{5}{4}w_1 - .25w_3 - .5x_4$$

such that:

$$x_2 = \frac{1}{3}[4 - x_1 - w_1 - x_4]$$
$$w_2 = \frac{1}{3}[5 - 5x_1 + w_1 + x_4]$$
$$x_3 = \frac{1}{12}[5 + x_1 + w_1 - .25w_3 - 2w_4]$$
$$x_i, w_i \geq 0$$

Now, pivoting x_1 into the basis and w_2 out of the basis so $B = \{x_1, x_2, x_3\}, NB = \{w_1, w_2, w_3, x_4\}$

$$\text{maximize } 6.5 - 4.5w_1 - 1.1w_1 - .25w_3 - .35x_4$$

such that:

$$x_2 = \frac{1}{10}[10 + 2w_2 - 4w_1 - x_4]$$
$$x_1 = \frac{1}{10}[10 - 6w_2 + 2w_1 + 2x_4]$$
$$x_3 = \frac{1}{20}[10 - w_2 + 2w_1 - 5w_3 - 3w_4]$$
$$x_i, w_i \geq 0$$

This is optimal, so the optimal value is 6.5, with $x_1 = 1, x_2 = 1, x_3 = .5, x_4 = 0$.

---

4. In Logistic Regression, we like to determine x_0 and $\mathbf{x}$ to maximize

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

which is equivalent to maximize the log-likelihood probability

$$-\sum_{i,c_i=1} \log\left(1+\exp(-\mathbf{a}_i^T\mathbf{x} - x_0)\right) - \sum_{i,c_i=-1} \log\left(1+\exp(\mathbf{a}_i^T\mathbf{x} + x_0)\right).$$

Or to minimize the log-logistic-loss

$$\sum_{i,c_i=1} \log\left(1+\exp(-\mathbf{a}_i^T\mathbf{x}-x_0)\right) + \sum_{i,c_i=-1} \log\left(1+\exp(\mathbf{a}_i^T\mathbf{x}+x_0)\right).$$

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

Let $f(x, x_0)$ be the log-logistic loss function.

$$\nabla f(\mathbf{x}, x_0)_{x_j} = \sum_{i,c_i=1} \frac{-a_{ij}\exp[-\mathbf{a}_i^T\mathbf{x}-x_0]}{1+\exp[-\mathbf{a}_i^T\mathbf{x}-x_0]} + \sum_{i,c_i=-1} \frac{a_{ij}\exp[\mathbf{a}_i^T\mathbf{x}+x_0]}{1+\exp[\mathbf{a}_i^T\mathbf{x}+x_0]} \;\forall j$$

$$\nabla f(\mathbf{x}, x_0)_{x_0} = \sum_{i,c_i=1} \frac{-\exp[-\mathbf{a}_i^T\mathbf{x}-x_0]}{1+\exp[-\mathbf{a}_i^T\mathbf{x}-x_0]} + \sum_{i,c_i=-1} \frac{\exp[\mathbf{a}_i^T\mathbf{x}+x_0]}{1+\exp[\mathbf{a}_i^T\mathbf{x}+x_0]}$$

(b) Consider the specific problem described in question (b) of Problem 6 in HW2, and find the solution by Logistic regression using the KKT conditions.

5 (Computation Team Work) Use the steepest descent, the accelerated steepest descent, the conjugate direction, and the BB methods to numerically solve the problem in (4.b) and compare their convergence speeds.

6. Questions (a) to (c) of 8.24 in the text book L&Y. (Bonus Question: In this method, we fist take the standard $1/\beta$ step-size. If some coordinate in the new iterate become negative, then we take a smaller step-size such that the iterate remain non-negative. What is the convergence speed of the method?)

Consider a problem of the form

$$\text{minimize } f(x)$$
$$\text{such that } x \geq 0$$

At a give point $x = (x_1, \ldots, x_n)$. the direction $d = (d_1, \ldots, d_n)$ is determined from the gradient vector $\nabla f(x)^T = g = (g_1, \ldots, g_n)$ by

$$d_i = \begin{cases} -g_i \text{ if } x_i > 0 \text{ or} g_i < 0 \\ 0 \text{if } x_i = 0 \text{ and } g_i \geq 0 \end{cases}$$

i) What are the first order conditions for a minimum of the problem.

The lagrangian of the problem is

$$L(x, \lambda) = f(x) - \lambda^T x$$

Thus the KKT conditions are

$$(1)\nabla f(x) - \lambda = 0$$
$$(2)\lambda_i \geq 0 \;\forall i$$
$$(3)\lambda_i x_i = 0 \;\forall i$$

---

ii) Show that d, as determined by the algorithm, is zero only at a point that satisfies the first order conditions.

We have that $d_i = 0$ when $x_i = 0$ and $g_i \geq 0$. From the KKT conditions we know that if $x_i = 0$ for all i, we have condition (3) satisfied no matter λ. Furthermore, since $g_i \geq 0$, we have that $\nabla f(x) = \lambda \geq 0, \forall i$. Thus condition (1) and (2), are satisfied. Therefore if $d_i = 0$, the KKT conditions are satisfied. Now $d_i \neq 0$, then $x_i \neq 0$ and $g_i < 0$ it is obvious the KKT conditions do not hold. Assume $d_i < 0$ and $x_i \neq 0$, then we have that $\nabla f(x)_i < 0$, since $\nabla f(x)_i = \lambda_i$, we must have that $\lambda_i < 0$, which violates the second KKT condition. Futhermore since $\lambda_i x_i = 0$, for condition (3), it follows that λ_i must equal zero, which is a contradiction.

---

iii) Show that if $d \neq 0$, it is possible to decrease the value of f by movement along d.

One thing to notice is that this is projected gradient descent. Let $\Pi_C(x)$ be the projection operator onto a set C. This problem reduces to projected gradient descent onto the non-negative orthant. To show you this, lets start the algorithm at some x_0. We have that

$$x_{k+1} = \Pi_{\mathbb{R}^+_n}(x_k - \alpha g)$$

Now since the nonnegative orthant is closed and convex, the projection exists and is unique. In particular, for some x_i the projection onto the non-negative orthant is $\Pi_{\mathbb{R}^+_n}(x_i) = \max(x_i, 0)$. Lets see the iteration procedure in this light.

$$\begin{aligned} x_{k+1} &= \Pi_{\mathbb{R}^+_n}(x_k - \alpha_k g) \\ &= \max(x_k - \alpha g, 0) \end{aligned}$$

Now fix an i, we see that $x^i_{k+1} = \max(x^i_k - \alpha g_i, 0)$. It is obvious that if both $x^i_k = 0$ and $g_i \geq 0$, the update is just zero. Now for the case $x^i_{k+1} > 0 || g_i < 0$. There are three cases, either $x^i_k > 0$, $g_i < 0$ or $x^i_{k+1} \& g_i < 0$.Notice if g_i in the kth iteration is larger than x^i_k, we can scale α, so that $\alpha g_i < x^i_k \forall i$. Assume only that $x^i_k > 0$. If $g_i > 0$, the function increases in x, thus any step we take will be in a direction of steepest decent, i.e.

$$x_{k+1} = \max(x^i_k - \alpha g_i, 0) \quad = x^i_k - \alpha g_i \text{ for an appropriately scaled alpha}$$

Now since $g_i > 0$, then $f(x_{k+1}) < f(x_k)$. If $g_i = 0$, then $d_i = 0$, and $f(x^i_{k+1}) = f(x_k)$. Now if $g_i < 0$, we have that the gradient is increasing with x_i. Furthermore $x^i_k - \alpha g_i$ will always be greater than zero, so this is standard steepest descent. With this update, it follows that $f(x_{k+1}) > f(x_k)$, since $g_i < 0$. Now fix the $g_i < 0$. For all x^i_k we have that $x^i_{k+1} = x^i_k - \alpha g_i > x^i_k$. Since $g_i < 0$, it follows that $f(x_{k+1})^i > f(x^i_k)$. Therefore, if $d \neq 0$, it is possible to find an x in the direction of d that decreases the value f.

---

7. Consider the problem 8.24 in the text L&Y, but solve the Barrier problem

$$\min_{x>0} \ \phi(\mathbf{x}) := f(\mathbf{x}) - \mu \sum_{j=1}^{n} \ln(x_j)$$

where μ is a small positive constant.

(a) What are the first-order necessary conditions for the minimizer of $\phi(\mathbf{x})$?

First notice that the log penalty constraint prevents the minimization from ever going negative, thus there is no need for a lagrangian constraint term. The first order necessary conditions are thus

$$\nabla_x \phi(x) = 0$$
$$\nabla_x f(x) - \mu \text{diag}(x)^{-1}\mathbf{e} = 0$$

(b) Let $f(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|^2$. Then develop the standard steepest gradient iterative process to find a KKT solution.

The gradient of $\phi(x)$ becomes

$$\nabla_x \phi(\mathbf{x}) = A^T(A\mathbf{x} - b) - \mu \text{diag}(\mathbf{x})^{-1}\mathbf{e}$$

Thus the steepest descent algorithm is as follows

1: Steepest descent

1: initialize x_0, α_0, μ
2: $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k(A^T(A\mathbf{x} - b) - \mu \text{diag}(\mathbf{x})^{-1}\mathbf{e})$

where $\alpha_k, \mathbf{x}_k > 0$.

(c) Again let $f(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|^2$, and at the k step consider the (affine) scaled gradient vector

$$(X^k)^2 \nabla \phi(\mathbf{x}^k),$$

where X^k is the diagonal matrix of kth iterate $\mathbf{x}^k (> 0)$. Let us generate the next iterate by

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha^k (X^k)^2 \nabla \phi(\mathbf{x}^k),$$

for some stepsize α^k. How is this process different from (b)

This is SDM with affine scaling, an interior point method that ensures that the SDM alogrithm can continue to take large steps even when the x_k is close to the feasible regions boundary. It does this by doing gradient descent in a re-scaled version of the problem and then rescaling back to the original problem. In the case above, not only does it eliminate the need to take the inverse of the matrix (which could lead to instability if the values in the diagonal are small), it also keeps the algorithm from staying strictly within the feasible region.

(d) (Computation Team Work) Generate some random data and solve them with $\mu = 0$ and $\mu = 10^{-4}$, and observe any possible difference between methods (b) and (c).

8. Now consider the problem:

$$\min \ f(X) \quad \text{s.t.} \quad X \succeq 0,$$

and its Barrier problem

$$\min_{X \succ 0} \ \phi(X) := f(X) - \mu \ln \det(X)$$

where μ is a small positive constant.

(a) What are the (symmetric) gradient matrix and the first-order necessary conditions for the minimizer of $\phi(X)$?

Since the log of the determinent functions as a barrier ensuring that that the matrix X is always positive definite (if $det(X) <= 0$, the log blows up to negative infinity). Taking the gradient with respect to $\phi(\mathbf{x})$

$$\nabla_x \phi(\mathbf{x}) = \nabla_x f(X) - \mu(X^{-1})$$

Thus the first order conditions are

$$\nabla_x f(X) = \mu X^{-1}$$

$X \succeq 0$

(b) Let $f(X) = \frac{1}{2}\|\mathcal{A}X - \mathbf{b}\|^2$, where

$$\mathcal{A}X = \begin{pmatrix} A_1 \bullet X \\ ... \\ A_m \bullet X \end{pmatrix}.$$

for some symmetric data matrices A_1,...,A_m, and data vector $\mathbf{b}$. Then develop the standard steepest gradient iterative process to find a KKT solution.

We have that $\nabla f(X) = \mathcal{A}^{\mathcal{T}}(\mathcal{A}X - b)$. Thus

$$\nabla \phi(\mathbf{x}) = \mathcal{A}^T(\mathcal{A}X - b) - \mu X^{-1}$$

The first order conditions are then

$$\mathcal{A}^T(\mathcal{A}X - b) = \mu X^{-1}$$

$X \succeq 0$.

So the SDM becomes

2: Steepest descent SDP

1: initialize x_0, α_0, μ

2: $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k(\mathcal{A}^{\mathcal{T}}(\mathcal{A}X - b) - \mu(X^{-1}))$

(c) Again let $f(X) = \frac{1}{2}\|\mathcal{A}X - \mathbf{b}\|^2$, and at the k step consider the scaled gradient matrix

$$X^k \nabla \phi(X^k) X^k,$$

where $X^k \succ 0$ is the kth iterate solution matrix. Let us generate the next iterate by

$$X^{k+1} = X^k - \alpha^k X^k \nabla \phi(X^k) X^k,$$

for some stepsize α^k. What is the difference between the two iterative processes in (b) and (c).

Again, this is a form of affine scaling for SDP cone, where in each step Di n

---

(d) (Computation Team Work) Apply this method for Sensor Network Localization problems in HW1 and HW2 varying with $\mu = 0$ to $\mu = 10^{-4}$.

9. (20pts Computation Team Work) All pieces of Computation Team Work stated above, and one more decribed below.

   There is a simple nonlinear least squares approach for Sensor Network Localization:

$$\min \quad \sum_{(ij)\in N_x} \left(\|\mathbf{x}_i - \mathbf{x}_j\|^2 - d_{ij}^2\right)^2 + \sum_{(kj)\in N_a} \left(\|\mathbf{a}_k - \mathbf{x}_j\|^2 - d_{kj}^2\right)^2 \tag{1}$$

   which is an unconstrained nonlinear minimization problem.

   Use the SDP solution (which may not accurate) as the initial solution for (1) and apply the Steepest Descent Method for a few number of steps for the two SNL problems in HW1 and HW2, where you can add some noises to the distance measurements. How is the final solution come out after steepest descent? How does the final solution compare with the SDM starting from a random initial solution?