ORIE 3310/5310
Lecture 28

2 May 2013

Last time, we look at an algorithm for solving

$$\begin{array}{ll} \text{Max} & c^T x \\ \text{s.t.} & Ax \leq b \\ & x \geq 0 \end{array} \iff \begin{array}{ll} \text{Max} & c^T x \\ \text{s.t.} & b - Ax \geq 0 \\ & x \geq 0 \end{array}$$

Ex:

$$\begin{array}{ll} \text{Max} & 192x_1 + 97x_2 \\ \text{s.t.} & 100 - x_1 - x_2 \geq 0 \\ & x_1 \geq 0 \\ & x_2 \geq 0. \end{array}$$

fix $\beta > 1$.

The Algorithm:

Step 1: Choose an initial $\mu^{(0)} > 0$.

Step 2: A the i^{th} iteration, we have $\mu^{(i)}$

Find the optimal solution of:

$$\max_{x \in \mathbb{R}^n} \; f_{\mu^{(i)}}(x)$$

call this opt solution $x(\mu^{(i)})$.

where $f_\mu(x) = c^T x + \mu\left[\sum_{j=1}^{m} \log(b_j - a_j^T x) + \sum_{i=1}^{n} \log(x_i)\right]$

Step 3: Let $\mu^{(i+1)} = \frac{\mu^{(i)}}{\beta}$.

Repeat Steps 2 & 3 $(i = 0, 1, 2, \ldots, N)$ (i.e N times)

Remarks: ① The set of points $x(\mu)$ for $\mu > 0$ forms a path that approaches x^*, the opt LP solution. $\{x(\mu) \mid \mu > 0\}$ is called the central path.

② Our algorithm produces points $x(\mu^{(i)})$ that traces the central path.

③ Computing $x(\mu^{(i)})$ in step 2 is normally not easy. So, we approximate $x(\mu^{(i)})$ using Newton's Method.

So: Rewriting the algorithm:
(Fix $\beta > 1$)

Step 1: Choose an initial $\mu^{(0)} > 0$
Choose an initial $x^{(-1)}$, strictly feasible for LP
(i.e. $x^{(-1)}$ is an interior point)

Step 2: At the i^{th} iteration, we have $\mu^{(i)}$, $x^{(i-1)}$
Using $x^{(i-1)}$ as an initial point,
carry out k_i iterations of Newton's Method to solve:

$$\nabla f_{\mu^{(i)}}(x) = 0$$

Let $x^{(i)}$ be the last Newton's Method iterate.

<u>Step 3</u>: Let $\mu^{(i+1)} = \frac{\mu^{(i)}}{\beta}$.

Repeat Steps 2 & 3 for $i = 1, \ldots, N$.

<u>Remarks</u>: ① The points $x^{(i)}$ $(i = 0, \ldots, N)$ approximate $x(\mu^{(i)})$. So, $x^{(i)}$ is not exactly on the central path, but (depending on how many Newton's method iterations k_i we carry out), $x^{(i)}$ <u>is close to $x(\mu^{(i)})$</u>.

Because of this, our interior-point method algorithm is often called the path-following algorithm because the iterates $x^{(i)}$ follows the central path.

② The question that many of you ask at the end of last class was:

If we know that $x(\mu)$ approaches x^* as μ approaches 0, why don't we just compute $x(0)$ directly?

This is a great question!

- The first answer: $x(\mu)$ is the opt. solution to:

$$\text{Max} \quad \underbrace{c^T x + \mu\left[\sum_{j=1}^{m} \log(b_j - a_j^T x) + \sum_{i=1}^{n} \log(x_i)\right]}_{f_\mu(x)}$$

So, for $\mu = 0$, $x(0)$ is the opt. solution to

$$\text{Max} \quad c^T x \quad , \text{ unconstrained}$$

so we lose all information about the constraints.

- The second answer:

Suppose we write down the optimality conditions:

$$\nabla f_\mu(x) = 0 \,. \quad — ⊛$$

Often, the solution to ⊛ for $\mu = 0$ exists, and it is actually the opt LP solution x^*.

However, computing x directly from ⊛ is

normally hard.
Computing x from ⊛ using Newton's method when $\mu=0$ is <u>numerically messy</u>.

<u>Recall</u> our first example from Tuesday:

$$\begin{array}{ll} \max & 3x_1 \\ \text{s.t.} & x_1 \le 2 \\ & x_1 \ge 0 \end{array} \quad \Bigg| \quad \begin{array}{ll} \max & 3x_1 \\ \text{s.t.} & 2-x_1 \ge 0 \\ & x_1 \ge 0. \end{array}$$

Then: $f_\mu(x) = 3x_1 + \mu\left(\log(2-x_1) + \log(x_1)\right)$

Set $f_\mu'(x) = 0$

$\therefore \quad 3 - \frac{\mu}{2-x_1} + \frac{\mu}{x_1} = 0.$

<u>Remark</u>
← when $x_1=0$ or $x_1=2$, bad!
← when $\mu=0$, bad!

→ So, $\boxed{\mu>0}$.

However, we did some maneuvers: multipling everything by $(2-x_1)x_1$:

$$3(2-x_1)x_1 - \mu x_1 + \mu(2-x_1) = 0.$$

$$\therefore \quad 3x_1^2 + (2\mu-6)x_1 - 2\mu = 0. \quad \text{—— (**)}$$

The solution: $x_1 = \left(1-\frac{\mu}{3}\right) + \frac{1}{3}\sqrt{\mu^2+9}$, $\boxed{\mu>0}$

But here, as $\mu \downarrow 0$, $x_1 \to 1 + \frac{1}{3}\sqrt{9} = 2$.

→ If we use NM to solve $3x_1^2+(2\mu-6)x_1-2\mu=0$:

$g_1(x_1) := 3x_1^2+(2\mu-6)x_1-2\mu$.

$g_1'(x_1) = 6x_1 + (2\mu-6)$

$\ell_1(x_1) = g_1(\bar{x}_1) + g_1'(\bar{x}_1)(x_1-\bar{x}_1) = 0.$

Solve for x_1 s.t. $\ell_1(x_1)=0$:

$$x_1 = \bar{x}_1 - (6\bar{x}_1+2\mu-6)^{-1} g_1(\bar{x}_1).$$

(3) Another question that was asked a few times:

- How do we choose $\mu^{(0)}$, the initial parameter?
- How do we choose $x^{(-1)}$, the initial starting point?
- How do we choose β, the factor by which $\mu^{(i)}$ is reduced at each iteration?
- How many Newton iterations each time ($k_i = ?$)?
- What should N = #iterations be?

"Answers":

- These are great questions, and the choice of $\mu^{(0)}$, $x^{(-1)}$, β matters a lot when we try to implement our algorithm.

- Depending on your problem, $\mu^{(0)}$ has to be a "sufficiently large" positive number.

- After $\mu^{(0)}$ is chosen, $x^{(-1)}$ should be a point that is "sufficiently close to $x(\mu^{(0)})$".

- The number of Newton iteration k_i might vary in each iteration and there is some tradeoff:
  - (*) the larger k_i is, the closer $x^{(i)}$ is to $x(\mu^{(i)})$ so, the more precise our solution will be.
  - (*) however, if k_i is too large, then we have to do a lot more computation.

  However, in general k_i is a small constant, and $k_i = 1$ can be sufficient (if β is not too large).

- If β is not too large (say $\beta = 1.1$) then $x(\mu^{(i)})$ is close to $x(\mu^{(i+1)})$

  $\therefore$ $x^{(i)}$ is close to $x(\mu^{(i+1)})$.

  So, $k_i = 1$ could be enough for obtaining a good approx to $x(\mu^{(i+1)})$

  However, if β is too close to 1, then N might have to be large in order to have $\mu^{(N)}$ close to 0.

  Note: $\mu^{(N)} = \mu^{(0)} \frac{1}{\beta^N}$

This brings us to the last part of our discussion on interior-point methods:

Given $\mu > 0$, how close to x^* is $x(\mu)$?

or: $\boxed{\text{Given } \mu > 0, \text{ how close to } \underbrace{c^T x^*}_{\text{the actual opt value}} \text{ is } \underbrace{c^T x(\mu)}_{\text{the opt value of } x(\mu)} ?}$

This question is important because:

① If in each iteration, $x^{(i)}$ is close to $x(\mu^{(i)})$,
then $x^{(i)}$ is close to x^*
if $x(\mu^{(i)})$ is close to x^*.

So, this will tell us how small $\mu^{(i)}$ would have to be in order that $x(\mu^{(i)})$ be close to x^*.

∴ How large N has to be so that

$$\mu^{(N)} = \mu^{(0)} \cdot \frac{1}{\beta^N} \text{ is small enough.}$$

② At the same time, this will give us a proof that $x^{(i)}$ does get closer and closer to x^*.

Recall: Weak Duality for linear programs:

The dual of
$$\left.\begin{array}{ll} \max & c^T x \\ \text{s.t.} & Ax \le b \\ & x \ge 0 \end{array}\right\} (P)$$

is
$$\left.\begin{array}{ll} \min & b^T y \\ \text{s.t.} & A^T y - t = c \\ & y \ge 0 \\ & t \ge 0 . \end{array}\right\} (D)$$

$$\begin{array}{ll} \min & b^T y \\ \text{s.t.} & \\ & c - A^T y + t = 0 \\ & y \ge 0 \\ & t \ge 0 . \end{array}$$

Weak duality: Suppose x is any feasible solution for (P)

(y,t) is any feasible solution for D

Then, $c^T x \leq b^T y$.

- The quantity $b^T y - c^T x$ is called the duality gap and is always ≥ 0 for all x, y feasible.

when $b^T y - c^T x = 0$, then y, x are both optimal

- Fix μ.
  Solve for $x(\mu)$.
  Then, $x(\mu)$ is (strictly) feasible for (P).

We can easily obtain (y,t) that is strictly feasible for (D) by looking at $x(\mu)$:

$x(\mu)$ satisfies:

$$\frac{\partial f_\mu(x)}{\partial x_i} = 0 \qquad \forall i.$$

$$f_\mu(x) = c^T x + \mu\left[\log(b_1 - a_1^T x) + \ldots + \log(b_m - a_m^T x) + \log(x_1) + \ldots + \log(x_n)\right].$$

$$\therefore \frac{\partial f_\mu}{\partial x_i}(x) = c_i - \frac{\mu a_{1i}}{b - a_1^T x} - \cdots - \frac{\mu a_{mi}}{b - a_m^T x} + \frac{\mu}{x_i} = 0.$$

$$\therefore \nabla f(x) = c - A^T \begin{pmatrix} \mu/b - a_1^T x \\ \vdots \\ \mu/b - a_m^T x \end{pmatrix} + \begin{pmatrix} \mu/x_1 \\ \vdots \\ \mu/x_n \end{pmatrix} = 0.$$

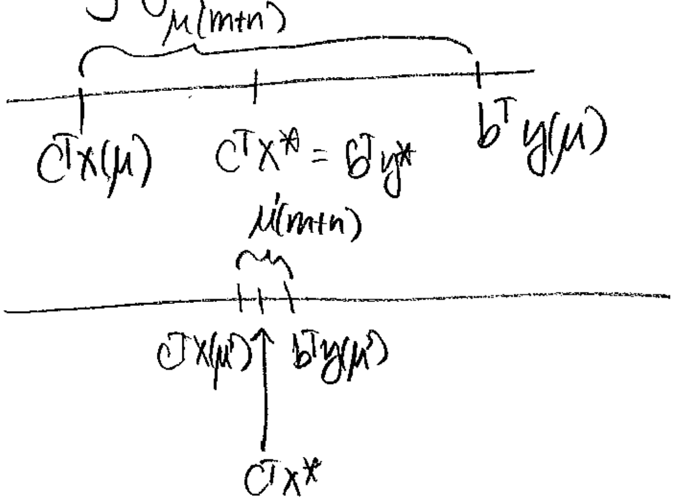
So, let $y = \begin{pmatrix} \frac{\mu}{b - a_1^T x(\mu)} \\ \vdots \\ \frac{\mu}{b - a_m^T x(\mu)} \end{pmatrix}$, $t = \begin{pmatrix} \mu/x(\mu)_1 \\ \vdots \\ \mu/x(\mu)_n \end{pmatrix}$

Then, (y,t) is feasible for (D).

• Then, choosing y, t in this way, we can show that

$b^T y - c^T x(\mu) = \mu(m+n)$ $\quad$ m = # constraints, n = # variables.

So, the duality gap decreases to zero as $\mu \downarrow 0$.



So, as $\mu \downarrow 0$, $\quad c^T x(\mu)$ is closer and closer to $c^T x^*$

$\therefore$ $x(\mu)$ is closer and closer to x^*.

→ Close with a summary of what has been covered this semester.

→ Thank the class

→ Apologies for non-reply of emails.

→ Announce teaching evaluations.