Mathematical Programming II	**Lecture 22**
ORIE 6310 Spring 2014	**April 17, 2014**
Scribe: Nanjing Jian	

Let $x_0 \in \mathbb{R}^n$, and $f \in \mathscr{F}_{LR} := \{f : \mathbb{R}^n \to \mathbb{R} : f$ convex, $C^{1,1}$, with Lipschitz constant L, and with a minimizer $x_* \in X_* = \{x : f(x) = \min f(\mathbb{R}^n)\}$ and $\|x_0 - x_*\| \leq R\}$.
Recall the algorithm (gradient method):
Start at x_0.
At each iteration k, set $x_{k+1} = x_k - \frac{1}{L}\nabla f((x_k))$.
Recall:

$$f(x_{k+1}) \leq f(x_k) - \frac{1}{2L}\left\|\nabla f((x_k))^2\right\| = f(x_k) - \frac{L}{2}\left\|x_{k+1} - x_k\right\|^2. \tag{0}$$

Theorem 1 *For $f \in \mathscr{F}_{LR}$, this algorithm produces x_k within ϵ of the minimum within $\frac{LR^2}{\epsilon}$ iterations.*

Proof: By the proposition from last lecture,

$$f(x_0) \leq f(x_*) + \nabla f((x_*))^T(x_0 - x_*) + \tfrac{1}{2}L\left\|x_0 - x_*\right\|^2.$$

By optimality of x_*, $\nabla f((x_*)) = 0$, so

$$f(x_0) - f(x_*) \leq \tfrac{1}{2}LR^2. \tag{1}$$

So for any l,

$$f(x_0) - f(x_{l+1}) \leq f(x_0) - f(x_*) \leq \tfrac{1}{2}LR^2.$$

Also summing up $f(x_k) - f(x_{k+1}) \geq \frac{1}{2L}\left\|\nabla f(x_k)\right\|^2$ from (0),

$$\frac{1}{2L}\sum_{k=0}^{l}\left\|\nabla f(x_k)\right\|^2 \leq f(x_0) - f(x_{l+1}).$$

Hence

$$\sum_{k=0}^{\infty}\left\|\nabla f(x_k)\right\|^2 \leq L^2R^2. \tag{2}$$

Finally,

$$\begin{aligned} &\left\|x_{k+1} - x_*\right\|^2 - \left\|x_k - x_*\right\|^2 \\ =&\left\|x_{k+1} - x_k\right\|^2 + 2(x_{k+1} - x_k)^T(x_k - x_*) \\ =&\frac{1}{L^2}\left\|\nabla f((x_k))\right\|^2 - \frac{2}{L}\nabla f(x_k)^T(x_k - x_*) \\ \leq&\frac{1}{L^2}\left\|\nabla f(x_k)\right\|^2 - \frac{2}{L}(f(x_k) - f(x_*)). \end{aligned} \tag{3}$$

Hence from (3), $f(x_k) - f(x_*) \leq \frac{L}{2}(\|x_k - x_*\|^2 - \|x_{k+1} - x_*\|^2) + \frac{1}{2L}\|\nabla f(x_k)\|^2$.
So

$$\begin{aligned}\sum_{j=0}^{k}(f(x_j) - f(x_*)) &\leq \frac{L}{2}\|x_0 - x_*\|^2 + \frac{1}{2L}\sum_{j=0}^{k}\|\nabla f(x_j)\|^2 \\ &\leq \tfrac{1}{2}LR^2 + \tfrac{1}{2}LR^2 \\ &\leq LR^2.\end{aligned}$$

Hence $f(x_k) - f(x_*) \leq \frac{LR^2}{k+1}$ since we have a descent method, and thus a solution within ϵ of the minimum will be reached within $k \leq \frac{LR^2}{\epsilon}$ iterations. □

Now let's get a lower bound on the complexity of minimizing $f \in \mathscr{F}_{LR}$. We will establish a bound assuming that $x_0 = 0$ and $x_{k+1} \in \operatorname{span}\{x_0, \nabla f(x_0), \ldots, \nabla f(x_k)\}$. The following function(s) is/are universally bad for such algorithms:

$$f_k(x) := \frac{L}{4}\left(\frac{1}{2}\left((e_1^T x)^2 + \sum_{j=1}^{k-1}(e_{j+1}^T x - e_j^T x)^2 + (e_k^T x)^2\right) - e_1^T x\right)$$

for $1 \leq k \leq n$. Then

$\nabla^2 f_k(x)$ is $\frac{L}{4}\begin{pmatrix} 2 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & -1 & 2 \end{pmatrix}$ in the upper-left k-by-k submatrix and 0's everywhere else. Denote $A_k = \frac{4}{L}\nabla^2 f_k(x)$. Also,

$$\nabla f_k(x) := \tfrac{L}{4}(A_k x + \begin{pmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}).$$

By the Gershgorin circle theorem, all eigenvalues of A_k are between 0 and 4, so f is convex and $C^{1,1}$ with Lipschitz constant L.

$$f_k \text{ is minimized by } x_k^* := \begin{pmatrix} \frac{k}{k+1} \\ \frac{k-1}{k+1} \\ \vdots \\ \frac{1}{k+1} \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ with value } f_k^* = \tfrac{L}{4}(-\tfrac{k}{k+1} \cdot \tfrac{1}{2}) = -\tfrac{Lk}{8(k+1)}.$$

Also $\|x_k\|^2 = \frac{\sum_{j=1}^{k} j^2}{(k+1)^2} \leq \frac{\int_0^{k+1} j^2 dj}{(k+1)^2} = \frac{1}{3}\frac{(k+1)^3}{(k+1)^2} = \frac{1}{3}(k+1) =: R^2$.
Thus all the requirements for the algorithm are met.
From our assumptions on the algorithm, $x_l \in \mathbb{R}^{n,l} := \{x \in \mathbb{R}^n : e_j^T x = 0, j > l\}$.

So the algorithm can't distinguish f_k from f_p, $p > k$, until after k^{th} iteration.
Let's apply algorithm to f_{2k+1}. Within k iterations the algorithm behaves as it does for f_k, so it generates x_k with $f_k(x_k) \geq f_k(x_k^*)$, so $f_{2k+1}(x_k) \geq f_k(x_k^*)$.
But

$$\begin{aligned} f_k^* - f_{2k+1}^* &= -\frac{Lk}{8(k+1)} + \frac{L(2k+1)}{8(2k+2)} \\ &= -\frac{L}{8}\left(\frac{k}{k+1} - \frac{2k+1}{2k+2}\right) \\ &= \frac{L}{16} \cdot \frac{1}{k+1} \\ &= \frac{1}{2} \cdot \frac{3}{16} \cdot \frac{1}{(k+1)^2} \cdot L \left\| x_{2k+1}^* - x_0 \right\|^2 \\ &= \frac{3}{32} \frac{1}{(k+1)^2} LR^2. \end{aligned}$$

So we need at least $(\frac{3}{32}\frac{LR^2}{\epsilon})^{1/2}$ steps to get within ϵ of the minimum. In fact, this lower bound holds for all first-order oracle algorithms.

We have a gap between this lower bound and the upper bound given by the gradient method. In fact, there is a better algorithm, due to Nesterov, that achieves $\mathcal{O}\left(\frac{1}{\epsilon^{1/2}}\right)$ iteration complexity.

In summary,

	Lower Bound	Upper Bound
Non-smooth convex min	$\frac{1}{4\epsilon^2}$	$\frac{1}{\epsilon^2}$ (short-step subgradient method)
Smooth convex min	$(\frac{3}{32}\frac{LR^2}{\epsilon})^{1/2}$	$\frac{LR^2}{\epsilon}$ (gradient method) $(\frac{2LR^2}{\epsilon})^{1/2}$ (Nesterov's optimal algorithm, conjugate gradient-like)
Structured non-smooth convex min		$\mathcal{O}(\frac{1}{\epsilon})$

We haven't seen the result for the structured non-smooth convex minimization problems yet. Here are two approaches.

One is in the homework: consider the problem

$$\text{minimize } \hat{f}(x) + g(x)$$

where $\hat{f}(x)$ is $C^{1,1}$ with Lipschitz constant L, and $g(x)$ is convex but non-smooth.
At each iteration, move to

$$\arg\min\{\hat{f}(x_k) + \nabla \hat{f}(x_k)^T(x - x_k) + \tfrac{1}{2}L\|x - x_k\|^2 + g(x)\}.$$

But for some non-smooth g this may be not easy.

The other is Nesterov's smoothing method.
Suppose we want to solve

$$\min_{x \in X} f(x)$$

where $f(x) := \hat{f}(x) + \max_{y \in Y}\{x^T Ay - \hat{g}(y)\}$, where $\hat{f}(x)$ is $C^{1,1}$ with Lipschitz constant L, and $\hat{g}(y)$ is convex and smooth, and $Y \subseteq B(0,1)$.
There is a dual problem:

$$\max_{y \in Y}\{-\hat{g}(y) + \min_{x \in X}\{x^T Ay + \hat{f}(x)\}\}.$$

But the inner minimization problem may be hard to solve, so a primal-dual algorithm may not work.
Instead, perturb the inner maximand in the original problem. Consider

$$f_\epsilon(x) := \hat{f}(x) + \max_{y \in Y}\{x^T Ay - \hat{g}(y) - \tfrac{1}{2}\epsilon \left\|y\right\|^2\}$$

so that the term inside the inner max is strictly concave.
Then,

(i) The inner maximization problem has a unique solution.

(ii) f_ϵ is continuously differentiable with Lipschitz continuous gradient with constant $L + \frac{\|A\|^2}{\epsilon}$.

(iii) f_ϵ is close to f: for any x, $f_\epsilon(x) \leq f(x) \leq f_\epsilon(x) + \frac{1}{2}\epsilon$.

So minimizing f_ϵ within $\frac{1}{2}\epsilon$ of its minimum also minimizes f within ϵ of its minimum.

The number of steps required is $\mathcal{O}((\frac{2(L+\frac{\|A\|^2}{\epsilon})R^2}{\epsilon})^{1/2}) \sim \mathcal{O}(\frac{L^{1/2}R}{\epsilon})$ by Nesterov's optimal algorithm.
Finally, why is the Lipschitz constant as stated in (ii)?
Assume $\hat{g} = 0$ for simplicity; then the inner problem is

$$\max_{y \in Y}\{(A^T x)^T y - \tfrac{1}{2}\epsilon \left\|y\right\|^2\}.$$

The unconstrained max is at $y(x) = \frac{A^T x}{\epsilon}$, and the derivative of this max with respect to x is $Ay(x) = \frac{AA^T x}{\epsilon}$, which is Lipschitz with constant $\frac{\|A\|^2}{\epsilon}$.