IIIS 2014 Spring: ATCS - Selected Topics in Optimization
Lecture date: Apr. 16th, 2014
Instructor: Jian Li **Scribe: Song Zuo**

1 Preliminaries

In this section, we review some important concepts related to *convex optimization*.

1.1 Convex Program

Before stating the definition of convex program, we need the following definitions.

Definition 1 (Convex Set) *A set S is convex, if*

$$\forall x, y \in S, \theta \in [0,1],\ \theta x + (1-\theta)y \in S$$

Definition 2 (Convex Function) *A function $f : \mathcal{D} \to \mathbb{R}$ is convex (where $\mathcal{D} \subseteq \mathbb{R}^n$ is the domain of this function), if $\mathcal{D}$ is convex and*

$$\forall x, y \in \mathcal{D}, \theta \in [0,1],\ f\big(\theta x + (1-\theta)y\big) \le \theta f(x) + (1-\theta) f(y)$$

Moreover, if $-f$ is convex, f is concave.

Definition 3 (Convex Program) *An optimization problem on the form*

$$\begin{aligned} \inf \quad & f(x) \\ \text{subj.t.} \quad & g_i(x) \le 0,\ i = 1, \dots, m \end{aligned}$$

is convex if the functions $f, g_1, \dots, g_m$ are convex.

Alternatively, the following optimization problem is convex, if $f_0, \dots, f_m$ are convex and $h_1, \dots, h_k$ are affine.

$$\begin{aligned} \inf \quad & f_0(x) \\ \text{subj.t.} \quad & f_i(x) \le 0,\ i = 1, \dots, m \\ & h_j(x) = 0,\ j = 1, \dots, k \end{aligned} \tag{1}$$

To introduce two important examples, we need the following notion.

Definition 4 (Positive Semidefinite) [1] *An n by n matrix P is positive semidefinite, denoted by $P \succeq 0$, if it is symmetric ($P \in S^n$) and*

$$\forall x \in \mathbb{R}^n,\ x^{\mathrm{T}} P x \ge 0$$

Notice that $P \succeq P'$ is equivalent to $P - P' \succeq 0$.

[1] There are many important equivalent definitions for this notion. `http://en.wikipedia.org/wiki/Positive-definite_matrix#Characterizations`

Example 5 (Quadratic Program) *Given $P \succeq 0$.*

$$\begin{aligned} \min \quad & \frac{1}{2}x^{\mathrm{T}}Px + q^{\mathrm{T}}x + r \\ \text{subj.t.} \quad & Gx \le h \\ & Ax = b \end{aligned}$$

Example 6 (Semidefinite Program(SDP)) [2] *Given $G, F_1, \ldots, F_n \in S^k$.*

$$\begin{aligned} \min \quad & c^{\mathrm{T}}x \\ \text{subj.t.} \quad & x_1F_1 + \cdots + x_nF_n + G \preceq 0 \\ & Ax = b \end{aligned}$$

1.2 Duality

Definition 7 (Lagrangian) *The Lagrangian according to convex program (1) is*

$$L(x, \lambda, \nu) = f_0(x) + \sum_i \lambda_i f_i(x) + \sum_j \nu_j h_j(x)$$

Definition 8 (Lagrange Dual) *The Lagrange dual problem of the primal (1) is*

$$\begin{aligned} \max \quad & g(\lambda, \nu) \\ \text{subjt.t.} \quad & \lambda \succeq 0 \end{aligned} \tag{2}$$

where $g(\lambda, \nu)$ is the Lagrange dual function defined as follows,

$$g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu), \;\; \mathcal{D} = \bigcap \mathbf{dom} f_i \bigcap \mathbf{dom} h_j$$

Suppose the OPTs of the primal and the dual are p^* and d^* respectively, the following property called *weak duality* always holds.

$$d^* \le p^*$$

Meanwhile, the following *strong duality* does not hold for arbitrary convex programs.

$$d^* = p^*$$

An important necessary condition of strong duality is provided as follows.

Definition 9 (Slater's Condition[2]) *Suppose that f_{i_1}'s are affine functions and f_{i_2}'s are convex functions, then Slater's condition is*

$$\exists x \in \mathbf{relint}\mathcal{D}, \;\; s.t. \;\; f_{i_1}(x) \le 0, \;\; f_{i_2}(x) < 0, \;\; Ax = b$$

Theorem 10 *[2] Slater's condition implies strong duality.*

[2] For Goemans-Williamson MAX-CUT approximation algorithm, the famous application of SDP, please see [1].

1.3 Unconstraint Convex Programs

For unconstraint convex programs, we have the following observation, which actually applies to all nonlinear programs.

Lemma 11 (Optimality Condition) *The following two statements apply to all nonlinear programs.*

1. *x_0 is a minimum point $\Longrightarrow \nabla f(x_0) = 0$.*
2. *If $f \in C^2$, then*

$$\nabla f(x_0) = 0,\ \nabla^2 f(x_0) \succ 0 \Longrightarrow x_0 \textit{ is a minimum point}$$

Now we give a brief proof to the second statement.

Proof: Since $f \in C^2$ and $\nabla^2 f(x_0) \succ 0$, there exists $r > 0$ such that $\forall x \in B(x_0, r)$, $\nabla^2 f(x) \succ 0$.
Using Taylor expansion with Lagrange remainder at any $x \in B(x_0, r)$,

$$f(x) = f(x_0) + (x - x_0)^{\mathrm{T}} \nabla f(x_0) + \frac{1}{2}(x - x_0)^{\mathrm{T}} \nabla^2 f(\xi_L)(x - x_0) \geq f(x_0)$$

where ξ_L is some point between x and x_0. □

1.4 Strongly Convex Function

Finally, we introduce the last notion in this section, which is very important for the upcoming sections.

Definition 12 (Strongly Convex Function) [3] *A function f is strongly convex with parameter $m > 0$, if for all x, y in its domain, and $\theta \in [0, 1]$.*

$$f\big(\theta x + (1-\theta)y\big) \leq \theta f(x) + (1-\theta) f(y) - \frac{1}{2} m\theta(1-\theta)\|x - y\|_2^2$$

Specially, for twice continuously differentiable function f, it is strongly convex with parameter m, if and only if for all x in its domain, $\nabla^2 f(x) \succeq mI$.

2 Gradient Descent

In this section, we briefly introduce the gradient descent method which is widely used to find the nearest local minimum of a differentiable function. This method basically starts at a given point x_0, and repeats the following iteration until some terminal condition is satisfied.

$$x_{i+1} = x_i + t\Delta x = x_i - t\nabla f(x_i)$$

where t is the step size.
Two typical ways to decide the step size are listed here.

[3] See `http://en.wikipedia.org/wiki/Convex_function#Strongly_convex_functions`.

1. Exact line search. Choose t to be the optimal value that minimizes $f(x_{i+1})$, i.e.,

$$t = \arg\min_{s>0} f(x_i + s\Delta x)$$

2. Backtrack line search, with parameters $\alpha, \beta \in (0, 1)$.

   This method aims to find a proper t such that the point $\big(x_{i+1}, f(x_{i+1})\big)$ is below the line $f(x_i) + \alpha t \nabla f(x_i)$. It works by first guessing the value of t, and if the t does not work, shrink it by factor β each time until the proper value is found.

2.1 Condition Number

Condition number, denoted by κ, is an important notion required for further discussion on convergence rate of gradient descent. The condition number of a matrix A is

$$\kappa(A) = \frac{\lambda_{\text{max}}(A)}{\lambda_{\text{min}}(A)}$$

Similarly, the condition number of a set C is

$$\kappa(C) = \left(\frac{\text{max width}}{\text{min width}}\right)^2 = \frac{\sup_{\|q\|_2=1}\left(\sup_{z\in C} q^{\mathrm{T}}z - \inf_{z\in C} q^{\mathrm{T}}z\right)^2}{\inf_{\|q\|_2=1}\left(\sup_{z\in C} q^{\mathrm{T}}z - \inf_{z\in C} q^{\mathrm{T}}z\right)^2}$$

Consider the following example.

Example 13 (Conditional Number of an Ellipsoid) *Suppose we have the following ellipsoid defined by a matrix $A \succ 0$.*

$$\mathcal{E} = \left\{x | (x - x_0)^{\mathrm{T}} A^{-1} (x - x_0) \le 1\right\}$$

Then

$$\kappa(\mathcal{E}) = \frac{\sup_{\|q\|_2=1} \|A^{1/2}q\|^2}{\inf_{\|q\|_2=1} \|A^{1/2}q\|^2} = \frac{\lambda_{\text{max}}(A)}{\lambda_{\text{min}}(A)} = \kappa(A)$$

Since conditional on $\|q\| = 1$,

$$\begin{aligned}
\left(\sup_{z\in\mathcal{E}} q^{\mathrm{T}}z - \inf_{z\in\mathcal{E}} q^{\mathrm{T}}z\right)^2 &= 4\sup_{z\in\mathcal{E}} \left(q^{\mathrm{T}}(z - x_0)\right)^2 \\
&= 4\sup_{z\in\mathcal{E}} \|z - x_0\|_2^2 \\
&= 4\sup\left\{\|y\|_2^2 | y^{\mathrm{T}} A^{-1} y \le 1\right\} \\
&= \frac{4}{\lambda_{\text{min}}(A^{-1})} \\
&= 4\lambda_{\text{max}}(A)
\end{aligned}$$

2.2 Convergence Rate

Theorem 14 (Convergence Rate) *Gradient descent method with exact line search returns x_k such that $f(x_k) - p^* \le \epsilon$ after k iterations. The convergence rate k is bounded as*

$$k = O\Bigg(\frac{\log\big(f(x_0) - p^*\big)/\epsilon}{m/M}\Bigg),$$

where x_0 is the start point, p^* is the OPT*of the unconstraint convex program, and m/M is the condition number.*

Moreover, the objective function f is strongly convex in its domain with parameter m, and $M > 0$ is some constant such that $\nabla^2 f(x) \preceq MI$ for all x in the sublevel set $C_{f(x_0)}$.

Proof: Firstly, by applying Taylor expansion with Lagrange remainder at x and the strongly convexity of f, we get

$$\begin{aligned} f(y) &= f(x) + (y-x)^{\mathrm{T}}\nabla f(x) + \frac{1}{2}(y-x)^{\mathrm{T}}\nabla^2 f(\xi)(y-x) \\ &\ge f(x) + (y-x)^{\mathrm{T}}\nabla f(x) + \frac{m}{2}\|y-x\|_2^2 \end{aligned} \tag{3}$$

Let x_0 be the point such that $\nabla f(x_0) = 0$, and we get

$$f(y) \ge f(x_0) + \frac{m}{2}\|y - x_0\|_2^2,$$

which implies that when $\forall y \in C_{f(x_0)}$, $\|y - x_0\|$ is upper bounded by a finite value. In other words, the sublevel set $C_{f(x_0)}$ is bounded and hence $M > 0$ is also guaranteed to be finite.

By choosing y^* to be the minimizer of (3), i.e., $y^* = x - \frac{1}{m}\nabla f(x)$, we have

$$f(y) \ge f(x) + (y^* - x)^{\mathrm{T}}\nabla f(x) + \frac{m}{2}\|y^* - x\|_2^2 = f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2 \tag{4}$$

Letting $y = x_0$, the inequality above implies that the smaller $\|\nabla f(x)\|_2$ is, the closer to optimal $f(x)$ is.

Now we come back to the iteration of the method. By the definition of exact line search,

$$f(x_{i+1}) = f\big(x_i + t_i\nabla f(x_i)\big) \le f\big(x_i - \nabla f(x_i)/M\big) \le f(x_i) - \frac{1}{2M}\|\nabla f(x_i)\|_2^2$$

The last inequality is based on the following, which can be proved similarly with (3).

$$f(y) \le f(x) + (y-x)^{\mathrm{T}}\nabla f(x) + \frac{M}{2}\|y-x\|_2^2$$

Combining with (4),

$$\|\nabla f(x_i)\|_2^2 \ge 2m\big(f(x_i) - p^*\big) \Longrightarrow f(x_{i+1}) - p^* \le \Big(1 - \frac{m}{M}\Big)\big(f(x_i) - p^*\big)$$

Therefore

$$f(x_k) - p^* \le \left(1 - \frac{m}{M}\right)^k \left(f(x_0) - p^*\right)$$

To guarantee that $f(x_k) - p^* \le \epsilon$, we need the number of iterations to be

$$k = O\left(\frac{\log \frac{\epsilon}{f(x_0)-p^*}}{\log\left(1 - \frac{m}{M}\right)}\right) = O\left(\frac{\log\left(f(x_0) - p^*\right)/\epsilon}{m/M}\right),$$

Notice that we use the approximation that $\log(1-z) \approx -z$ when $|z|$ is small. □

2.3 Steepest Descent

Steepest descent is a more general descent method. In stead of simply choosing Δx to be $-\nabla f(x)$, steepest descent chooses Δx w.r.t. some norm $\|\cdot\|$, i.e.,

- for normalized case,

$$\Delta x_{nsd} = \arg\min_{\|v\|=1} v^{\mathrm{T}} \nabla f(x),$$

- and for unnormalized case.

$$\Delta x_{sd} = \|\nabla f(x)\|_* \cdot \Delta_{nsd} x.$$

Recall the $\|\cdot\|_*$ is the dual norm of $\|\cdot\|$,

$$\|z\|_* = \sup_{\|w\| \le 1} z^{\mathrm{T}} w$$

Example 15 (Quadratic Norm) *Consider quadratic norm defined by a positive define matrix P.*

$$\|z\|_P = \left(z^{\mathrm{T}} P z\right)^{1/2} = \left\|P^{1/2} z\right\|_2,$$

and

$$\|z\|_* = \left\|P^{-1/2} z\right\|_2.$$

Hence

$$\begin{aligned}
\Delta x_{sd} &= \|\nabla f(x)\|_{P^{-1}} \cdot \arg\min_{\|v\|_P=1} v^{\mathrm{T}} \nabla f(x) \\
&= -\left(\nabla f(x)^{\mathrm{T}} P^{-1} \nabla f(x)\right)^{1/2} \cdot \arg\max_{\|v\|_P=1} v^{\mathrm{T}} \nabla f(x) \\
&= -\left(\nabla f(x)^{\mathrm{T}} P^{-1} \nabla f(x)\right)^{1/2} \cdot \frac{P^{-1} \nabla f(x)}{\left(\nabla f(x)^{\mathrm{T}} P^{-1} \nabla f(x)\right)^{1/2}} \\
&= -P^{-1} \nabla f(x)
\end{aligned}$$

Notice that $v^{\mathrm{T}} \nabla f(x) = \|\nabla f(x)\|_{P^{-1}}$ and $\|v\|_P = 1$.

References

[1] Goemans, Michel X., and David P. Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming." *Journal of the ACM (JACM)* 42.6 (1995): 1115-1145.

[2] Boyd, Stephen P., and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.