EE 381V: Large Scale Optimization **Fall 2012**

Lecture 5 — September 13

Lecturer: Caramanis & Sanghavi *Scribe: Debarati Kundu & Tejaswini Ganapathi*

5.1 Topics covered

- Recap of definitions and theorems taught in the previous lecture
- Coordinate Descent Method
- Steepest Descent Method

In the last lecture, the gradient descent algorithm was elaborated, along with the introduction of the concept of strong convexity and its implications. Moreover, the convergence rate was analyzed for exact line search and backtracking line search methods. In this lecture, after a brief recap, two new descent methods were introduced, namely, Coordinate Descent, and the method of Steepest Descent.

5.2 Recap of previous lecture

Definition: $f \in C_L^{1,1}$, if $\|\nabla f(x) - \nabla f(y)\|_2 \leq \|x - y\|_2$.

Theorem 5.1. *For any $f \in C_L^{1,1}$ (not necessarily convex), such that $f^* = \min_x f(x) > -\infty$, the gradient descent algorithm with $\eta < \frac{2}{L}$ will converge to a stationary point.*

Definition of Strong Convexity: The objective function f is said to be strongly convex with $m > 0$, $M > 0$ if $mI \preceq \nabla^2 f \preceq MI$, $\forall x$.

Lemma 5.2. *For such an f,*

$$\begin{aligned} f(y) &\geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{m}{2}\|y - x\|^2 \\ f(y) &\leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{M}{2}\|y - x\|^2 \end{aligned} \tag{5.1}$$

Theorem 5.3. *The gradient descent algorithm for a strongly convex function f with step size $\eta = \frac{1}{M}$ will converge as*

$$f(x^{(k)}) - f^* \leq c^k (f(x^{(0)}) - f^*) \tag{5.2}$$

where $c = 1 - \frac{m}{M}$. This rate of convergence is known as linear convergence.

In general, the value of M is not known. Hence, line search is done in order to determine where the next iterate would be. For exact line search, $c = 1 - \frac{m}{M}$. For backtracking line search, $c = 1 - \min\{2m\alpha, \frac{2\beta\alpha m}{M}\} < 1$.
Aside: For a descent method, convexity of the function and continuity of the second derivative guarantees the existence of M if m exists.

5.3 Coordinate Descent Method

Coordinate descent belongs to the class of several nonderivative methods used for minimizing differentiable functions. Here, cost is minimized in one coordinate direction in each iteration. The order in which coordinates are chosen may vary in the course of the algorithm.

Let the minimization be carried out over n variables. In the case where the order is cyclical, given $x^{(k)}$, the i-th coordinate of $x^{(k+1)}$ can be determined by:

$$\begin{aligned} {x_j}^{(k+1)} &= {x_j}^{(k)}, j \neq i \\ {x_i}^{(k+1)} &= \arg\min_{\xi \in \Re} f(x_{\setminus i}^{(k)}, \xi) \end{aligned} \tag{5.3}$$

The minimization over the variable i can be done using gradient descent method with a fixed stepsize η:

$${x_i}^{(k+1)} = {x_i}^{(k)} - \eta \frac{\partial f}{\partial x_i}(x^{(k)}) \tag{5.4}$$

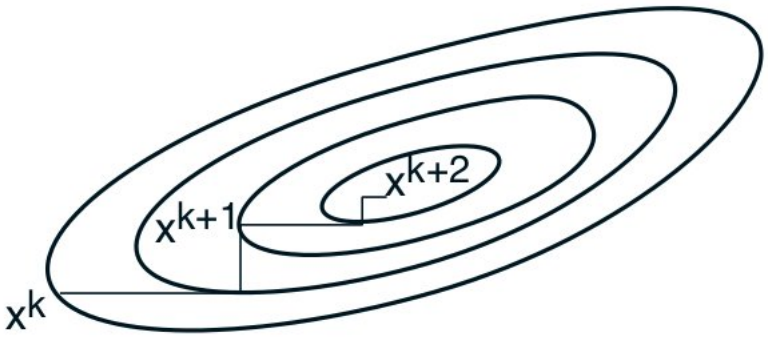

Figure 5.1. Illustration of coordinate descent method

Figure 5.1 illustrates the algorithm. The method can also be used for the minimization of f, subject to upper and lower bounds on the variables x_i, $x \in \{1, \cdots, n\}$. The minimization

over $\xi \in \Re$ in the previous equation is replaced by minimization over the appropriate interval in the previous equation.

5.3.1 Advantages of coordinate descent

An important advantage of coordinate descent is that it is well suited for *parallel computation*. In particular, suppose that there is a subset of coordinates $x_{i_1}, x_{i_2}, \cdots, x_{i_m}$, which are not coupled through the cost function. That is $f(x)$ can be expressed as $\sum_{r=1}^{m} f_{i_r}(x)$, where for each r, $f_{i_r}(x)$ does not depend on the coordinates $x_{i_s}, \forall s \neq r$. Then m coordinate descent iterations

$$x_{i_r}^{(k+1)} = \arg\min_{\xi \in \Re} f(x^{(k)} + \xi e_{i_r}), r = 1, \cdots, m \tag{5.5}$$

independently and in parallel. Thus is problems with special structure where the set of coordinates can be partitioned into p subsets with the above mentioned independence property, one can perform a full cycle of coordinate descent iterations in p parallel steps (as opposed to n), assuming the availability of sufficient number of parallel processors.

A second advantage of the coordinate descent method lies in the fact that it can be very useful in cases where the actual gradient of the function is not known.

5.3.2 Disadvantage of coordinate descent

The coordinate descent method may not reach the local minimum even for a convex function, as shown in Figure 5.2. The algorithm may get stuck at a non-stationary point (labelled by 'X' in the figure) if the level curves of a function are not smooth. An example of this type of function is $f(x_1, x_2) = \max(x_1, x_2)$.

5.3.3 Convergence of Coordinate Descent

The coordinate descent method generally has similar convergence properties to steepest descent. For continuously differentiable cost functions, it can be shown to generate sequences whose limit points are stationary.

Lemma 5.4. *Suppose $\nabla f(x)$ is continuous and for every x and i, $f(x_{\setminus i}, \xi)$ has a unique minimum ξ^*, and is monotonic between x_i and ξ. Then cyclic coordinate descent with exact line search will reach stationary point. (Proposition 2.7.1, Bertsekas).*

Proof: Let

$$z_i^{(k)} = (x_1^{(k+1)}, \cdots, x_i^{(k+1)}, x_{(i+1)}^{(k)}, \cdots, x_n^{(k)}) \tag{5.6}$$

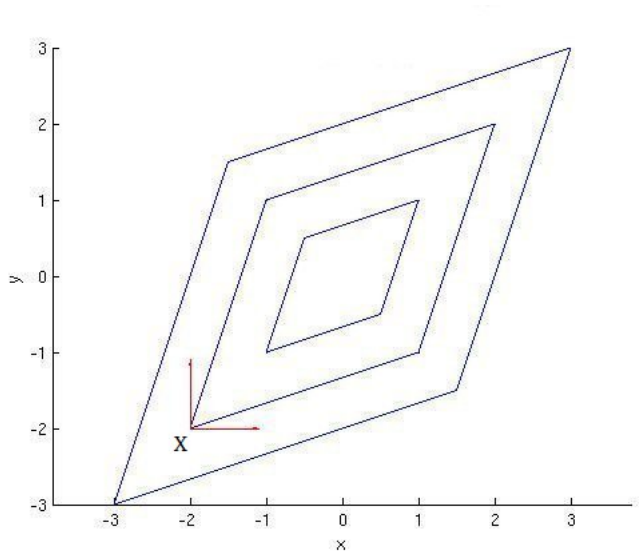

Figure 5.2. Disadvantage of coordinate descent method

By Equation 5.3, we can write

$$f(x^{(k)}) \geq f(z_1^{(k)}) \geq f(z_2^{(k)}) \geq \cdots \geq f(z_{n-1}^{(k)}) \geq f(x^{(k+1)}), \forall k \tag{5.7}$$

Let $\bar{x} = (\bar{x}_1, \cdots, \bar{x}_n)$ be a limit point of the sequence $x^{(k)}$. Let $x \in X$, where X is a closed set. Hence, $\bar{x} \in X$. Equation 5.7 indicates that the sequence $f(x^{(k)})$ converges to $f(\bar{x})$. Now, it is to be shown that $\bar{x}$ minimizes f over X.

Let $\{x^{(k_j)} | j = 0, 1, \cdots\}$ be a subsequence of $\{x^{(k)}\}$ that converges to $\bar{x}$. We first show that $\{x_1^{(k_j+1)} - x_1^{(k_j)}\}$ converges to zero as $j \to \infty$. Assume the contrary, or equivalently, that $\{z_1^{(k_j)} - x^{(k_j)}\}$ does not converge to zero. Let $\gamma^{(k_j)} = \|z_1^{(k_j)} - x^{(k_j)}\|$. By possibly restricting to a subsequence of $\{k_j\}$, we may assume that there exists some $\bar{\gamma} > 0$ such that $\gamma^{(k_j)} \geq \bar{\gamma}$ for all j. Let $s_1^{(k_j)} = \frac{z_1^{(k_j)} - x^{(k_j)}}{\gamma^{(k_j)}}$. Thus $z_1^{(k_j)} = x^{(k_j)} + \gamma^{(k_j)} s_1^{(k_j)}$, $\|s_1^{(k_j)}\| = 1$, and $s_1^{(k_j)}$ differs from zero only along the first coordinate direction. $s_1^{(k_j)}$ belongs to a compact set and therefore has a limit point $\bar{s}_1$. By restricting to a further subsequence of $\{k_j\}$, we can assume that $s_1^{(k_j)}$ converges to $\bar{s}_1$.

Let us fix some $\epsilon \in [0,1]$. Now, $0 \leq \epsilon\bar{\gamma} \leq \gamma^{(k_j)}$. Therefore, $x^{(k_j)} + \epsilon\bar{\gamma} s_1^{(k_j)}$ lies on the segment of the line joining $x^{(k_j)}$ and $x^{(k_j)} + \gamma^{(k_j)} s_1^{(k_j)} = z_1^{(k_j)}$, and belongs to X, because X is convex. Using the fact that $z_1^{(k_j)}$ minimizes f over all x that differ from $x^{(k_j)}$ along the first coordinate direction, we obtain,

$$f(z_1^{(k_j)}) \quad = \quad f(x^{(k_j)} + \gamma^{(k_j)} s_1^{(k_j)}) \leq f(x^{(k_j)} + \epsilon\bar{\gamma} s_1^{(k_j)}) \leq f(x^{(k_j)}) \tag{5.8}$$

Since $f(x^{(k)})$ converges to $f(\bar{x})$, Equation 5.7 shows that $f(z_1^{(k)})$ also converges to $f(\bar{x})$. Now we can take the limit as $j \to \infty$, to obrain $f(\bar{x}) \leq f(\bar{x} + \epsilon\bar{\gamma}\bar{s}_1) \leq f(\bar{x})$. We conclude that

$f(\bar{x}) = f(\bar{x} + \epsilon\bar{\gamma}\bar{s}_1)$, for every $\epsilon \in [0,1]$. Since $\bar{\gamma}\bar{s}_1 \neq 0$, this contradicts the hypothesis that f is uniquely minimized when viewed as a function of the first coordinate direction. This contradiction establishes that $x_1^{(k_j+1)} - x_1^{(k_j)}$ converges to zero. In particular, $z_1^{(k_j)}$ converges to $\bar{x}$.

From Equation 5.3, we have

$$f(z_1^{(k_j)}) \quad \leq \quad f(x_1, x_2^{(k_j)}, \cdots, x_n^{(k_j)}), \forall x_1 \tag{5.9}$$

Taking the limit as $j \to \infty$, we obtain

$$f(\bar{x}) \quad \leq \quad f(x_1, \bar{x}_2, \cdots, \bar{x}_n), \forall x_1 \tag{5.10}$$

Using the conditions for optimality over a convex set, we conclude that

$$\nabla_1 f(\bar{x})'(x_1 - \bar{x}_1) \quad \geq \quad 0, \forall x_1 \tag{5.11}$$

where $\nabla_i f$ denotes the gradient of f with respect to the component x_i.

Let us now consider the sequence $\{z_1^{(k_j)}\}$. It has been already shown that $z_1^{(k_j)}$ converges to $\bar{x}$. By similar arguments, it can be shown that $x_2^{(k_j+1)} - x_2^{(k_j)} \to 0$ and $\nabla_2 f(\bar{x})'(x_2 - \bar{x}_2) \geq 0, \forall x_2$. Continuing inductively, we obtain $\nabla_i f(\bar{x})'(x_i - \bar{x}_i) \geq 0, \forall x_i$, $i \in \{1, \cdots, n\}$. Adding this inequalities, we conclude that $\nabla f(\bar{x})'(x - \bar{x}) \geq 0$ for every $x \in X$. Hence proved. □

5.3.4 Method of selecting the coordinate for next iteration

Different methods have been proposed for the selection of the coordinate, on which the next descent iteration would be performed. Some of the well known methods are:

- **Cyclic Coordinate Descent**: This method has been described already, where at each iteration, line search is done along one coordinate direction at the current point. The different coordinate directions are used cyclically in course of the algorithm.

- **Greedy Coordinate Descent**: In this method, at each iteration, the coordinate direction i, $i \in 1, \cdots, n$ along which line search is to be done is chosen in a greedy manner by the following maximization problem:

$$i^* \quad = \quad \arg\max_i \left| \frac{\partial f}{\partial x_i}(x) \right| \tag{5.12}$$

  But this method is intensive computationally because at each iteration, finding the coordinate direction that maximizes the gradient of the function at the point is done in linear time.

- **(Uniform) Random Coordinate Descent**: This algorithm belongs to the broader class of stochastic gradient descent algorithms. In this case, the coordinate direction i, along which the line search is to be done is chosen uniformly randomly over $i, \cdots, n$. For a constant step size η, the rule for updating the position in the next iteration is given by:

$$x^+ = x - \eta(\frac{\partial f}{\partial x_i})e_i \tag{5.13}$$

  Now, computing the expectation of the position in the next iteration,

$$\mathbb{E}[x^+] = x - \frac{\eta}{n}\nabla f(x) \tag{5.14}$$

We will study the performance of stochastic gradient descent later in the class.

5.4 Steepest Descent Method

The gradient descent method takes many iterations to converge for certain starting points, when the function has elongated level sets and the descent direction is slowly varying. The steepest descent method aims at choosing the best descent direction at each iteration.

Given a norm $\|\cdot\|$, a *normalized steepest descent direction* is defined as follows:

$$\Delta x_{nsd} = \arg\min_v \{< \nabla f(x), v >, s.t. \, ||v|| \leq 1\} \tag{5.15}$$

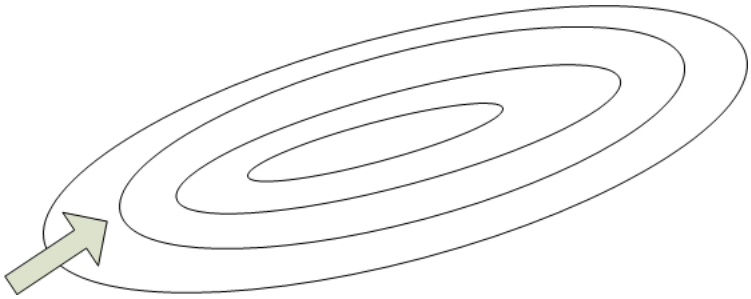

Figure 5.3. Illustration of a function having elongated level sets with slowly varying descent direction

Iteratively, the algorithm follows the following steps:

- Calculate direction of descent, Δx_{nsd}
- Calculate step size, t
- $x_+ = x + t\Delta x_{nsd}$

5.4.1 Steepest Descent for l_2, l_1 and l_∞ norms

- $||\cdot||_2$

  If we impose the constraint $||v||_2 \leq 1$ in Equation 5.15, then the steepest descent direction coincides with the direction of $-\nabla f(x)$, and the algorithm is the same as gradient descent.

$$\Delta x_{nsd} = \frac{-\nabla f(x)}{||\nabla f(x)||_2}$$

- $||\cdot||_1$

  For $||x||_1 = \sum_i |x_i|$, a descent direction is as follows,

$$\Delta x_{nsd} = -sign\left(\frac{\partial f(x)}{\partial x_{i^*}}\right) e_{i^*}$$
$$i^* = \arg\max_i \left|\frac{\partial f}{\partial x_i}\right|$$

  In the above set of equations, e_i is the standard basis corresponding to index i. Figure 5.4 geometrically illustrates this concept.

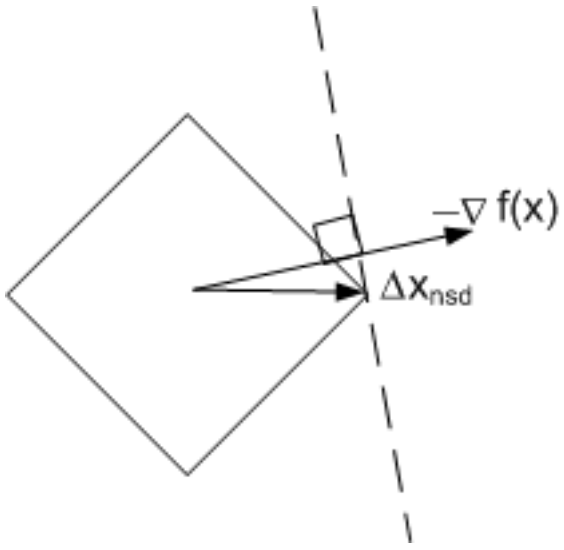

Figure 5.4. Geometric illustration of the normalized steepest descent direction for l_1 norm

- $||.||_\infty$

  For $||x||_\infty = \arg\max_i |x_i|$, a descent direction is as follows,

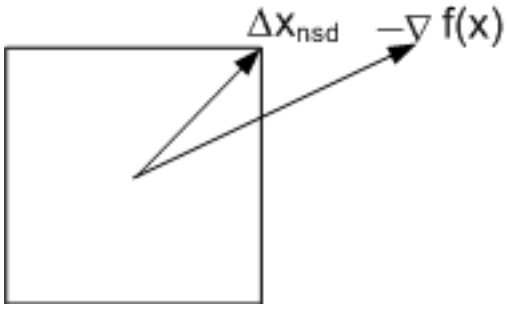

Figure 5.5. Geometric illustration of the normalized steepest descent direction for l_∞ norm

$$\Delta x_{nsd} = sign(-\nabla f(x))$$

Figure 5.5 geometrically illustrates this solution.

Aside: Dual Norm
Dual norm of $||.||$ is defined as,

$$||z||_* = sup\{< z, x >, s.t., ||x|| = 1\}$$

Therefore,

$$||.||_2 \iff ||.||_2$$
$$||.||_1 \iff ||.||_\infty$$

and,

$$< \nabla f(x), \Delta x_{nsd} >= ||\nabla f(x)||_*$$

5.4.2 Rate of Convergence under strong convexity

Fact: Any norm can be bounded by $||.||_2$, i.e., $\exists \gamma \& \tilde{\gamma} \in (0, 1]$ such that, $||x|| \geq \gamma ||x||_2$ and $||x||_* \geq \gamma ||x||_2$

Theorem 5.5. *If f is strongly convex with respect to m and M, and $||.||$ has γ, $\tilde{\gamma}$ as above then steepest descent with backtracking line search has linear convergence with rate $c = 1 - 2m\alpha\tilde{\gamma}^2 \min\{1, \frac{\beta\gamma}{M}\}$*

Proof: Will be proved in the next lecture □

5.5 References

1. *Nonlinear Programming*, Dimitri P. Bertsekas, MIT.