MATH 126 Winter 18: Homework 3

posted on Jan. 26; due by Feb. 2

1. [10 points] Derive a steepest descent direction with respect to the ℓ_∞-norm, and give a simple interpretation of such steepest descent direction.

2. [10 points] Show that the following convex function is smooth:

$$f(\boldsymbol{x}) = \log\left(\sum_{l=1}^{d} e^{x_l}\right), \tag{0.1}$$

   where $\boldsymbol{x} = (x_1, \ldots, x_d)^\top$.

3. [10 points each] Consider the following smooth convex problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^d} \{f(\boldsymbol{x}) \equiv \phi(\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b})\} \tag{P}$$

   where a pseudo-Huber function (a smooth approximation of ℓ_1-norm) is defined as

$$\phi(\boldsymbol{y}) = \sum_{i=1}^{p} \delta^2 \left(\sqrt{1 + \frac{y_i^2}{\delta^2}} - 1\right).$$

   Let $\boldsymbol{A}$ be the 100×100 Hilbert matrix, $\boldsymbol{b} = -(1, 1/2, \ldots, 1/100)^\top$, and $\delta = 1$. Implement and run the following four gradient methods with the initial vector $\boldsymbol{x}_0 = (1, 2, \ldots, 100)^\top$ for total 50 number of iterations:
   - A gradient method with a constant step (learned in the class)
   - A gradient method with a backtracking line search (see below that is slightly different from the version we learned in the class. note that L_k is nondecreasing.)
   - Nesterov's fast gradient method with a constant step (learned in the class)
   - Nesterov's fast gradient method with a backtracking line search (see below)

   Use $L_0 = 2$ and $\eta = 1.2$ for the backtracking line search.

   (1) Show that $f(\boldsymbol{x})$ is L-smooth and convex. Report the numerical value of your constant step size $\frac{1}{L}$.

   (2) Plot $f(\boldsymbol{x}_k)$ vs. iteration (k) for four iterative algorithms. [1]

[1] The backtracking line search versions will likely be faster than the constant step versions in this plot, but be aware that the backtracking line search versions has longer per-iteration run time compared to the constant step versions. On the other hand, the constant step versions require additionally computing the step size $\frac{1}{L}$ before iterating.

Algorithm 1 Gradient method with a backtracking line search

Input: $f \in \mathcal{F}_L^{1,1}$, $\boldsymbol{x}_0 \in \mathbb{R}^d$, $L_0 > 0$, $\eta > 1$.
for $k \geq 0$ **do**
  Find the smallest nonnegative integers i_k such that $f\left(\boldsymbol{x}_k - \frac{1}{\bar{L}}\nabla f(\boldsymbol{x}_k)\right) \leq f(\boldsymbol{x}_k) - \frac{1}{2\bar{L}}||\nabla f(\boldsymbol{x}_k)||_2^2$ with $\bar{L} = \eta^{i_k} L_k$
  $L_{k+1} = \eta^{i_k} L_k$
  $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \frac{1}{L_{k+1}}\nabla f(\boldsymbol{x}_k)$

Algorithm 2 Nesterov's fast gradient method with a backtracking line search

Input: $f \in \mathcal{F}_L^{1,1}$, $\boldsymbol{x}_0 = \boldsymbol{y}_0 \in \mathbb{R}^d$, $t_0 = 1$, $L_0 > 0$, $\eta > 1$.
for $k \geq 0$ **do**
  Find the smallest nonnegative integers i_k such that $f\left(\boldsymbol{y}_k - \frac{1}{\bar{L}}\nabla f(\boldsymbol{y}_k)\right) \leq f(\boldsymbol{y}_k) - \frac{1}{2\bar{L}}||\nabla f(\boldsymbol{y}_k)||_2^2$ with $\bar{L} = \eta^{i_k} L_k$
  $L_{k+1} = \eta^{i_k} L_k$
  $\boldsymbol{x}_{k+1} = \boldsymbol{y}_k - \frac{1}{L_{k+1}}\nabla f(\boldsymbol{y}_k)$
  $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
  $\boldsymbol{y}_{k+1} = \boldsymbol{x}_{k+1} + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{x}_{k+1} - \boldsymbol{x}_k)$