Massachusetts Institute of Technology
6.854J/18.415J: Advanced Algorithms
Ankur Moitra

Wednesday, April 6, 2016

Problem Set 7

Due: Wednesday, April 13, 2016 – 7 pm
Dropbox Outside Stata G5

Collaboration policy: collaboration is *strongly encouraged*. However, remember that

1. You must write up your own solutions, independently.
2. You must record the name of every collaborator.
3. You must actually participate in solving all the problems. This is difficult in very large groups, so you should keep your collaboration groups limited to 3 or 4 people in a given week.
4. Write each problem in a separate sheet and write down your name on top of every sheet.
5. **No bibles. This includes solutions posted to problems in previous years.**

1 Self-concordant barriers

As we discussed in class, a function $f : (a, b) \to \mathbb{R}$ is self-concordant if it is convex and $|f'''(t)| \leq 2|f''(t)|^{3/2}$ for all $t \in (a, b)$. Moreover, a function $f : \mathbb{R}^n \to \mathbb{R}$ is self-concordant if its restriction to any line, that is, $f_{\mathbf{x},\mathbf{u}}(t) = f(\mathbf{x} + t\mathbf{u})$ is self-concordant.

Furthermore, we say that f is a c-self-concordant barrier for a convex set D if:

1. The domain of f is D and $f(\mathbf{x}) \to \infty$ as $\mathbf{x}$ approaches the boundary of D (f is a "barrier" for D).
2. f is self-concordant.
3. The restriction of f to any line satisfies $|f'(t)| \leq (c|f''(t)|)^{1/2}$.

(a) Prove that the sum of an a-self-concordant barrier for the set X and a b-self-concordant barrier for the set Y is an $(a + b)$-self-concordant barrier for $X \cap Y$.

(b) Show that $-\ln x_i$ is 1-self-concordant for set $x_i > 0$.

(c) Conclude that the logarithmic barrier $-\sum_i \ln x_i$ is n-self concordant for $\mathbf{x} > 0$ (where n is the length of $\mathbf{x}$).

2 Planar Vertex Cover

To recall, in Vertex Cover problem we are given a weighted graph $G = (V, E)$ with a weight function w defined over V and the goal is to select the minimum weight subset of vertices S in V such that for each edge $e = (u, v)$ at least one of u, v are in S.

(a) Prove that any *extreme point* solution of the standard LP relaxation of Vertex Cover problem

$$\begin{aligned}\min_x \quad & \sum_{v \in V} x_v \\ \text{s.t.} \quad & x_u + x_v \geq 1 \quad \forall (u, v) \in E \\ & 0 \leq x_v \leq 1 \quad \forall v \in V\end{aligned}$$

has property that $x_v \in \left\{0, \frac{1}{2}, 1\right\}$ for all $v \in V$. Extreme point x is a feasible solution that cannot be written as $\lambda x^1 + (1 - \lambda)x^2$ for $0 < \lambda < 1$ and feasible solution x^1 and x^2 distinct from x.

One of the advantages of LP-based approaches is that they provide a general framework so that in many cases lead to better approximation guarantee over a restricted but yet interesting subset of input by exploiting some properties of those instances.

(b) Give a $\frac{3}{2}$-approximation algorithm for Vertex Cover problem on planar graphs. You may use the fact that there are polynomial time LP solvers that return extreme point *optimal* solution and that there is polynomial time algorithms to 4-color any planar graph (i.e. each vertex is assigned to one of the colors such that the end points of any edge have different colors).

(b) (**Optional**) A graph G is a k-degenerate graph if in every subgraph of G there is a vertex of degree k. Design a $(2 - \frac{2}{k+1})$-approximation algorithm for Vertex Cover on k-degenerate graphs.

3 MAX-SAT

In this problem we aim to design randomized approximation algorithm for MAX-SAT problem. First let us define the problem formally. In MAX-SAT problem, the input consists of n boolean variables $x_1, \cdots x_n$ (each may be either `true` or `false`), m clauses $C_1, \cdots, C_m$ (each of which consists of disjunction ("or") of some number variables and their negations) and a non-negative weight w_i for each clause. The objective is to find an assignment of `true/false` to x_is that maximize the weight of *satisfied* clauses. A clause is satisfied if one of its non-negated variable is set to `true`, or one of the negated variable is set to `false`.

Moreover, we assume that no literal is repeated in a clause and at most one of x_i and $\bar{x}_i$ appears in a clause.

(a) Write down an LP-relaxation of MAX-SAT problem. **Hint:** Come up with an integer program with 0-1 variables y_i for each boolean variable x_i such that $y_i = 1$ corresponds to x_i set to `true`).

(b) Show the expected performance of the "standard" randomized rounding algorithm (on the LP-relaxation you designed in part (a)) is $(1 - 1/e)$. Note that clauses can be of any length.

A naive algorithm for MAX-SAT problem is to set each variable to `true` with probability 1/2. It is easy to see that this *unbiased randomized* algorithm of MAX-SAT is a (1/2)-approximation algorithm (in expectation).

(c) Show the algorithm that returns the best of two solutions given by the randomized rounding algorithm and the simple unbiased randomized algorithm is a (3/4)-approximation algorithm of MAX-SAT.

4 Richardson Iteration: An Alternative View

In lecture, we briefly discussed the following least squares problem:

$$\min_{\mathbf{x}} f(\mathbf{x}) = \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$$

$\mathbf{A}$ is a fixed data matrix (not necessarily square or symmetric) and $\mathbf{b}$ is a fixed vector containing the dependent variable. This problem can be solved directly by computing $\mathbf{x}^* = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$.

However, it is often much faster to apply iterative methods. Here you will give a complete analysis for applying gradient descent to the problem. Specifically, consider the following iteration, which is often called "Richardson Iteration":

$$\begin{aligned} \mathbf{x}_0 &\Leftarrow \mathbf{0} \\ \mathbf{x}_{t+1} &\Leftarrow \mathbf{x}_t + \eta \cdot 2(\mathbf{A}^T\mathbf{b} - \mathbf{A}^T\mathbf{A}\mathbf{x}_t) \end{aligned}$$

For this problem you may assume that $\mathbf{A}$ is full rank and thus $\mathbf{A}^T\mathbf{A}$ is invertible.

(a) Show that $f(\mathbf{x})$ is convex.

(b) Show that if we set $\eta = O(1/\|\mathbf{A}^T\mathbf{A}\|_2)$, after

$$t = O\left(\|\mathbf{A}^T\mathbf{A}\|_2\|(\mathbf{A}^T\mathbf{A})^{-1}\|_2 \log(\|\mathbf{A}^T\mathbf{A}\|_2\|(\mathbf{A}^T\mathbf{A})^{-1}\|_2/\epsilon)\right)$$

iterations this method will converge, specifically guaranteeing that:

$$f(\mathbf{x}_t) - f(\mathbf{x}^*) \le \epsilon\|\mathbf{b}\|_2^2.$$

For simple problems like linear system solving, there are often alternative ways of analyzing and viewing methods like Richardson-Iteration. Here we briefly consider one such approach.

(c) Show that for $\mathbf{M} = (\mathbf{I} - 2\eta\mathbf{A}^T\mathbf{A})$, our iteration is equivalent to:

$$\mathbf{x}_{t+1} \Leftarrow \mathbf{M}(\mathbf{x}_t - \mathbf{x}^*) + \mathbf{x}^*$$

(d) What property of $\mathbf{M}$ is required for the iterate $\mathbf{x}_t$ to converge to $\mathbf{x}^*$? How should we set η to guarantee this convergence? With η set as suggested, how many iterations are required to guarantee that:

$$\|\mathbf{x}_t - \mathbf{x}^*\|_2 \leq \epsilon\|\mathbf{x}^*\|_2$$