Approximation Algorithms

EO249

(Problem Set 2) Due: Feb 20th, 2015

Solutions need to be submitted by email to eo249iisc@gmail.com. We prefer latexed solution. Before giving the solutions, you should write how many problems you have attempted and how many you think you have solved. Starred problems are optional and (possibly) more fun.

Exercise 1. Give an example of a set cover instance where the greedy solution has cost at least $c \cdot H_K \cdot OPT$ for some constant c, where OPT is the optimal greedy set cover cost.

Exercise 2. In the **vertex cover** problem, we are given a graph and we need to find the minimum cardinality subset of vertices such that each edge has at least one end point in that subset. A greedy algorithm for the problem picks the largest degree vertex, deletes it, and proceeds recursively. How good is the algorithm? Give both upper bounds and lower bounds.

Exercise 3. Recall the submodular set cover problem done in class as well as in the notes – given a universe U with cost c_i for each $i \in U$, a **monotone** submodular function f defined over the subsets of the universe, and a number R, find the minimum cost set A such that $f(A) \geq R$. Analyse the greedy algorithm for the problem. Hint: Go through both the analyses done in class – they may give you different answers.

Exercise 4. Consider the source location problem done in class: given a directed graph G where capacity of all edges is 1 with a single sink t and a set of candidate sources $U = \{s_1, \ldots, s_k\}$ where c_i is the cost of source i, pick the minimum cost set of sources $S \subseteq U$ which can together send R units of flow to t. Show that this problem is as hard as the set cover problem. Does your analysis go through if the graph is allowed to be undirected.

Exercise 5. Consider the following for unconstrained submodular maximization problem (for general, not monotone, submodular functions): pick an element i which has the largest marginal value stopping if marginals of all remaining elements is ≤ 0. More formally, we always maintain a set S which is initialized to the empty set and we pick the element i which maximizes $f(S \cup i) - f(S)$ till all such marginals are negative.

What is the approximation factor of this algorithm?

Starred Problems

Exercise 6. (*) Let us change the greedy algorithm for the vertex cover described in exercise 2. Instead of picking the vertex with the maximum degree, pick a vertex v *randomly* with probability of picking vertex v proportional to degree of the vertex v. That is, if X is the set of vertices remaining, pick $v \in X$ with probability $d(v)/\sum_{v \in X} d(v)$ where $d(v)$ is the degree of v in the remaining graph.

Obviously, the final set S that is picked is a random set. What can you say about the $\mathbf{Exp}[|S|]$ compared to OPT, the minimum cardinality vertex cover? What can you say for the "bad" examples you have created in exercise 2.

Exercise 7. (*) Consider the following problem: we are given a connected, undirected graph G whose edge set is E. Suppose f is a *monotone, submodular* function defined over subsets of E. The goal is to find a spanning tree T of G which maximizes $f(E(T))$. Propose a greedy algorithm for this problem. Can you analyze this?