HOMEWORK 1

Due: Wednesday, Sep 22, 11:59pm EST on Gradescope

Exercises (do not submit)

1. Setting the Stage, Covering some Basics.

(a) Show that the HITTING SET problem on m sets and n ground elements is equivalent to the SET COVER problem on n sets and m ground elements.

(b) Observe that the VERTEX COVER problem on a graph with n vertices and m edges is a special case of SET COVER on a set system with m ground elements and n sets, where each element belongs to exactly two sets.

(c) Given a SET COVER instance where each element belongs to at most f sets. Give an LP-rounding algorithm for such instances that is an f-approximation. (Hint: consider the case $f = 2$, what do the LP constraints look like?)

(d) Show that the greedy algorithm for SET COVER achieves an $O(\log(n/OPT))$-approximation. (Hint: first show it for the unweighted case.)

(e) We saw an LP rounding algoritm we saw for SET COVER in Lecture #2. Give an algorithmic gap instance for it: namely give the instance, give an optimal LP solution for it, such that we would need $\Omega(\log n)$ rounds of independently picking each set S with probability x_S in order to guarantee a good set cover.

2. Bounded Degree Instances are Often Easier. Suppose each set $S \in \mathcal{S}$ has size at most $B \leq n$, and sets have costs. The charging argument in lecture already showed that the greedy algorithm incurs a cost of at most $H_B\, OPT = O(\log B)\, OPT$. Let's give an LP-rounding algorithm that achieves this result.

(a) For each element $e \in U$, let $S(e) \in \mathcal{S}$ be the least-cost set that contains e. Show that $\frac{1}{B} \sum_e c_{S(e)} \leq LP \leq OPT$.

(b) Give an algorithm that solves the LP and then picks some sets (randomly) based on the optimal LP solution (and has a clean-up phase at the end), so that the expected cost at most $O(\log B)$ times the LP value.

3. Submodularity. Given a set X, a function $f : 2^X \to \mathbb{R}$ maps subsets of X to real values. It is *monotone* if $f(A) \leq f(B)$ for all $A \subseteq B$. For a set $A \subseteq X$ and element $e \in X$, define the marginal value of e with respect to X to be

$$f_A(e) := f(A + e) - f(A). \tag{1}$$

(Henceforth we use $A + e$ to mean $A \cup \{e\}$, and $A - e$ to mean $A \setminus \{e\}$.) The function f is called *submodular* if for every $A \subseteq B$ and $e \notin B$,

$$f_A(e) \geq f_B(e). \tag{2}$$

In words, the marginal value of e with respect to supersets is smaller, we have diminishing marginal returns.

(a) Often submodularity of f is defined thus: for every C, D subsets of X,

$$f(C \cup D) + f(C \cap D) \leq f(C) + f(D). \tag{3}$$

Show that (2) and (3) are equivalent to each other.

(b) Given a set system $(U, \mathcal{S} = \{S_1, S_2, \ldots, S_m\})$ with m sets, define $X := \{1, 2, \ldots, m\}$. Define the *set coverage* function $f(A) := |\cup_{i \in A} S_i|$ for each $A \subseteq X$. Show that f is submodular and monotone.

(c) Given an undirected graph $G = (X, E)$, and a subset $A \subseteq X$, define ∂A to be the set of edges with exactly one endpoint in A, and the other outside A. Define $f(A) := |\partial A|$ be the *cut function*. Show that f is submodular but not monotone.

(d) There are many other examples of submodular functions. E.g., consider a collection of (discrete) random variables $Y_1, Y_2, \ldots, Y_n$ with some joint probability distribution. For a set $A \subseteq [n]$, define the (Shannon) entropy of r.v.s indexed by the set A as

$$H(A) := -\sum_{y_{i_1} \cdots y_{i_{|A|}}} \Pr[\wedge_{i \in A}(Y_i = y_i)] \, \log_2 \Pr[\wedge_{i \in A}(Y_i = y_i)].$$

The function H (where $H(A)$ can be be thought of as "the information content" of the r.v.s indexed by A) is a monotone submodular function.

4. Facility Location via Set Cover. Problem 1.4 from [WS10].

5. Deterministic Rounding for Max-Coverage. (This exercise steps through the deterministic rounding algorithm we discussed at the end of Lecture #2.) Given an instance of MAX-COVERAGE and a fractional solution (x, z), consider the function $f_e(x) = 1 - \prod_{S: e \in S}(1 - x_S)$, and the expected coverage function $f(x) = \sum_e f_e(x)$. Finally, define

$$g(\varepsilon) := f(x + \varepsilon(\mathbf{e}_i - \mathbf{e}_j))$$

where $\mathbf{e}_i, \mathbf{e}_j$ are the standard basis vectors in the i^{th}, j^{th} directions.

(a) Argue that $g(\varepsilon)$ is convex in the variable ε.

(b) Suppose x is a feasible solution where both x_i, x_j are fractional in $(0, 1)$. Use convexity of g to argue that there exist $\alpha, \beta > 0$ such that (i) the two solutions $x + \alpha(\mathbf{e}_i - \mathbf{e}_j)$ or $x - \beta(\mathbf{e}_i - \mathbf{e}_j)$ have at least one integer coordinate, and (ii) the f-value of at least one of these two solutions is higher than $f(x)$.

(c) If x is an integral solution feasible for the LP, then $f(x)$ is the coverage given by the feasible solution $\{S_i \mid x_i = 1\}$.

Repeatedly moving from some x^t to the new solution x^{t+1} allows us to end up with an integer solution after at most n steps. Each move increases the f value. Hence the final coverage is $f(x_{final}) \geq f(x_{init}) \geq (1 - 1/e)\, LP$; the last inequality uses $1 + y \leq e^y$ and was argued in lecture.

Problems

[WS10] is the Williamson and Shmoys textbook, linked off the course page too.

1. Fun with Vertex-Cover. For each of the following approximation algorithms for Min-Vertex-Cover with positive vertex costs: (a) prove the best approximation ratio guarantee that you can, and (b) give matching algorithmic gaps if possible. (I.e., if you show a ρ-approximation, give instances showing that the algorithm cannot do much better than ρ.) Some of these algorithms do better for the special case when all costs are 1: if that is the case, please point it out.

a) *Super Naive:* Consider all the edges in some order. If the edge $\{u, v\}$ being considered is not covered yet, pick whichever of u or v has less cost.

b) *Naive:* Consider all the edges in some order. If the edge $\{u, v\}$ being considered is not covered yet, pick *both* the vertices u and v.

c) *Randomized:* Consider all the edges in some order. If the edge $\{u, v\}$ being considered is not covered yet, with probability $\frac{c_v}{c_u+c_v}$ pick the vertex u, and with the remaining probability, pick v.

d) *LP rounding:* The standard Vertex-Cover LP is the following: minimize $\sum c_v x_v$ subject to $x_u + x_v \geq 1$ for all edges $\{u, v\} \in E$, and $x \geq 0$. Given a fractional solution for this LP, define $V_\alpha = \{v \in V \mid x_v \geq \alpha\}$. What value of α ensures that V_α is a vertex cover? What approximation guaratee can you get?

e) *Local search:* Define two solutions $S \subseteq V$ and $S' \subseteq V$ to be neighbors if S can be obtained from S' by adding, deleting, or swapping a vertex. (Swapping means simultaneously adding a vertex and dropping another.) The local search moves are simple: Start with any solution $S \subseteq V$; if you are at some solution S, move to any neighboring solution S' that has less cost. If you are at a *local optimum* — where all the neighbors have at least as much cost — output this local optimum. (Don't worry about the running time for this algorithm.)

f) *Greedy:* Repeatedly pick a vertex v that maximizes $\frac{\text{number of edges newly covered}}{c_v}$, until all the edges are covered.

2. George and Leslie. Problem 1.5 from [WS10].

3. Frame thy Fearful (A)Symmetry. Problem 1.3 from [WS10].

4. Submodular Goes Only So Far. (Please try the exercise on submodularity before you start this problem.) Given a *monotone submodular* function $f : 2^U \to \mathbb{R}$ with $f(\varnothing) = 0$, you want to pick a set A with k elements that maximizes $f(A)$.

(a) Show that this problem is NP-hard to approximate better than $1 - 1/e$. (You may use any theorems from lecture, without proof.)

(b) Show that for any set A, the marginal value function $f_A(\cdot) := f(A \cup \cdot) - f(A)$ is also monotone submodular. Moreover, show that any non-negative submodular function g is *subadditive*, i.e., $g(A \cup B) \leq g(A) + g(B)$ for disjoint sets A, B.

(c) Consider the following greedy algorithm: start with $A_0 = \varnothing$, and let $e_t \leftarrow \arg\max_{e \in U} f_{A_{t-1}}(e)$ and then $A_t \leftarrow A_{t-1} + e_t$. If A^* is an optimal set, show that

$$f(A_k) \geq (1 - 1/e) \cdot f(A^*).$$

Now consider the following variant of this submodular maximization problem: you are given a partition $U_1, U_2, \ldots, U_k$ of U into k parts. You want to pick exactly one element e_i from each part U_i to maximize $f(\{e_1, \ldots, e_k\})$. Consider the greedy algorithm as above, where now $e_t \leftarrow \arg\max_{e \in U_t} f_{A_{t-1}}(e)$.

(d) Show this modified greedy algorithm is a $\frac{1}{2}$-approximation.

(e) Give a reduction from instances $\mathcal{G}$ of MAX-LABEL-COVER to instances $\mathcal{G}$ of this problem that has (a) perfect completeness, and (b) the following soundness: if $value(\mathcal{G}) < \eta$ then $value(\mathcal{G}) < \frac{3}{4} + O(\eta)$. Give a couple sentences arguing this completeness and soundness. (Hint: this is an easy reduction, given the Lecture.)