Advanced Algorithm 2023 Fall Take-Home Final

Due date: noon January 9th, 2024 UTC +8

Max-Cut in random graph

In this question, we analyze the Greedy algorithm for Max-Cut on $G(n, 1/2)$. Recall that in $G(n, 1/2)$, each edge is present i.i.d with probability $1/2$.

(1) Prove that with high probability, a graph drawn from $G(n, 1/2)$ has a maximum cut at most $\frac{n^2}{8} + O(n^{1.5})$.

(2) Recall that at iteration i of the greedy algorithm, the vertex v_i joins either S or T to maximize $E(S, T)$. Let a_i, b_i be the number of edges v_i is connected to S and T respectively. Show that $\mathbb{E}\left[|a_i - b_i|\right] \geq \Omega(\sqrt{i})$.

(3) Show that $\mathbf{Var}\left[|a_i - b_i|\right] = O(i)$.

(4) Conclude that with probability at least 0.99, the greedy algorithm will find a cut of value at least $\frac{n^2}{8} + \Omega(n^{1.5})$.

Almost k-wise independence

We show that there exists a small sample space that guarantees "almost k-wise independence".

Fix any $\varepsilon > 0$. A random n-bit string $\vec{x}$ is said to be "k-wise ε-independent" if $\forall \vec{b} \in \{0,1\}^n, S \in \binom{[n]}{k}$,

$$\left| \mathbf{Pr}\left[\forall j \in S, x_j = b_j\right] - \frac{1}{2^k} \right| < \varepsilon.$$

Fix any integers n, k with $n > k$ and $\varepsilon > 0$. Prove that for $m = O(\frac{2^k k \ln n}{\varepsilon^2})$, there are bit strings $y^{(1)}, \ldots, y^{(m)}$, each of length n, such that if we pick i uniformly at random from $[m]$, then $y^{(i)}$ is "k-wise ε-independent".

HINT: you only need a proof of existence, not an explicit construction. Partial credit will be awarded if you can show existence for $m \ll 2^n$ whenever $k \ll n$.

List Coloring and Lovász local lemma

Let $G = (V, E)$ be an undirected graph and suppose each $v \in V$ is associated with a set $S(v)$ of $32r$ colors, where $r \geq 1$. Suppose, in addition, that for each $v \in V$ and $c \in S(v)$ there are at most r neighbors u of v such that c lies in $S(u)$. Use local lemma to prove that there exists a proper coloring of G assigning to each vertex v a color from its class $S(v)$ such that, for any edge $(u, v) \in E$, the colors assigned to u and v are different. Furthermore, give a polynomial time randomized algorithm to find such a coloring.

Spectrum and Cuts

Given G and H on the same vertex set V.

(1) If $L_G \preceq L_H$, prove that $|E_G(S, \bar{S})| \leq |E_H(S, \bar{S})|$ for every $S \subseteq V$.

(2) Prove that the converse is not true, by constructing G, H such that $|E_G(S, \bar{S})| \leq |E_H(S, \bar{S})|$ for every $S \subseteq V$, but $L_G \not\preceq L_H$.

Effective resistance and Connectedness

We consider unweighted graphs in this question.

(1) A corollary of Menger's theorem in graph theory is that: a graph is k-edge connected if between any pairs of vertices, there are at least k edge disjoint paths connecting them. Let G be a connected graph with maximum effective resistance over edges R. In other words, $R := \max_{uv \in E(G)} R_{\text{eff}}(u, v)$. Show that G must be $1/R$-edge connected.

(2) Show that the converse to the above is not true.

(3) Show that for any simple unweighted graph G and any edge $(u, v) \in E(G)$, the effective resistance between u, v satisfies $R_{\text{eff}}(u, v) \geq \frac{1}{\deg(u)+1} + \frac{1}{\deg(v)+1}$.

(4) Let G be a d-regular graph. The second largest eigenvalue of its adjacency matrix is $\alpha_2 = \varepsilon d$ for some constant $\varepsilon \in (0, 1)$. Prove that for any pair of vertices u, v, the effective resistance between them satisfies $R_{\text{eff}}(u, v) \geq \frac{1}{(1-\varepsilon)d}$.

Universal Routing

Consider a game between a (galactic) taxi driver and a passenger. The passenger wants to visit n cities in space, each at least once, in as little time as possible. The taxi driver wants to delay the tour for as long as possible, so that they can charge a higher fare. In each city, there are exactly d routes leading to d different cities. The passenger must choose one from these d routes to decide where to go next. However, the passenger does not know the map, nor the name of the cities: they have no idea where these routes are leading to. Each route has a unique label from 1 to d. The taxi driver will follow the label consistently, that is, the next time they come back to the same city, these d routes have the same unique labels. It takes 1 unit of time to move between cities.

(1) Prove that there is a randomized strategy for the passenger to visit every city at least once, and it takes at most $O(n^2 d)$ time in expectation.

(2) Prove that there is a deterministic strategy for the passenger to visit every city at least once, and it takes at most $O(n^3 d^2 \ln(nd))$ time. In other words, such a deterministic strategy is universal and applicable to every possible labelling and layouts of n cities.

HINT: You only need to show the existence of such a strategy. You might find it helpful to think about the probability that your randomized strategy keeps failing after repeating $nd \ln(nd)$ times.

(3) Suppose that $d = n$, that is, the underlying map is a labeled complete graph with self-loops, but with unknown labels. Find a deterministic strategy for the passenger to visit every city at least once within $O(n^3 \ln(n))$ time.

Greedy and Local Search

(1) Given two matroids $\mathcal{M}_1 = (U_1, \mathcal{I}_1)$ and $\mathcal{M}_2 = (U_2, \mathcal{I}_2)$ with $U_1 \cap U_2 = \varnothing$, the direct sum of $\mathcal{M}_1$ and $\mathcal{M}_2$, denoted as $\mathcal{M}_1 \oplus \mathcal{M}_2$, is defined as $(U_1 \cup U_2, \{A_1 \cup A_2 : A_1 \in \mathcal{I}_1, A_2 \in \mathcal{I}_2\})$. Prove that $\mathcal{M}_1 \oplus \mathcal{M}_2$ is a matroid.

(2) Recall that in the max-coverage problem, we are given a ground set U with $|U| = n$, m subsets $S_1, S_2, \cdots, S_m$ of U, and an integer $k \in [m]$. Our goal is to find a set $C \subseteq [m]$ of size k so as to maximize $|\bigcup_{i \in C} S_i|$.

For two constants $\alpha \geq 1, \beta \geq 1$, an algorithm is an (α, β)-bicriteria approximation algorithm if it outputs a set $C \subseteq [m]$ of size at most $\lceil \alpha k \rceil$, such that $|\bigcup_{i \in C} S_i| \geq \beta \cdot \mathsf{opt}$, where opt is the maximum number of elements that can be covered using k subsets, i.e., the value of the given instance.

Given a constant $\alpha \geq 1$, design an efficient $(\alpha, 1 - e^{-\alpha})$-bicriteria approximation algorithm for the max-coverage problem; prove that it achieves this goal.

Linear Programming and Linear Programming Rounding

(1) A doctor wants to combine three food kinds such that the mixture's vitamin content includes a minimum of 8 elements of vitamin A, 10 elements of vitamin B and 8 elements of vitamin C. There are three food kinds, 'I', 'II' and 'III'. The following table gives the units of vitamin A, B

and C contained in per kilogram of food kind 'I', 'II' and 'III', and the price (in US dollars) for per kilogram of each food kind.

Food kind	Vitamin A	Vitamin B	Vitamin C	Price
I	2	3	1	5
II	1	4	2	7
III	4	2	3	10
Requirement	8	10	8	

You need to minimize the price of a combination satisfying the requirement. Write down the linear program that solves the problem. (You do not need to solve the LP).

(2) Given a graph $G = (V, E)$ and an integer $k \geq 1$, the graph G is said to be k-orientable if we can obtain a directed graph $\vec{G} = (V, \vec{E})$ by choosing a direction for every edge in G, so that the in-degree of every vertex $v \in V$ is at most k in $\vec{G}$.

Design a polynomial time algorithm that, given a graph $G = (V, E)$ and an integer $k \geq 1$, decides if G is k-orientable or not. Prove its correctness.

Primal Dual

(1) Write down the dual LP for your LP to problem (1) in the "Linear Programming and Linear Programming Rounding" section.

(2) Recall that in the maximum flow problem, we are given a directed graph $G = (V, E)$, with a source $s \in V$ and a sink $t \in E$. We are given a capacity $c_e \in \mathbb{Z}_{>0}$ for every edge $e \in E$. Consider the natural linear program for the maximum flow problem.

$$\max \quad \sum_{e \in \delta^{\text{in}}(t)} x_e$$

$$\begin{aligned} x_e &\leq c_e && \forall e \in E \\ \sum_{e \in \delta^{\text{out}}(v)} x_e - \sum_{e \in \delta^{\text{in}}(v)} x_e &= 0 && \forall v \in V \setminus \{s, t\} \\ x_e &\geq 0 && \forall e \in E \end{aligned}$$

Write down the dual LP for the above LP. Explain why the LP is solving the following problem: give every edge $e \in E$ a length $y_e \geq 0$ so that the shortest $s \to t$ path in G with respect to lengths $(y_e)_{e \in E}$ is at least 1, so as to minimize $\sum_{e \in E} c_e y_e$. To get a full score for the problem, you need to prove the statement directly.