

Homework Out: February 12, 2017

Due Date: February 26, 2017, before class

The HW contains some exercises (fairly simple problems to check you are on board with the concepts; dont submit your solutions), and problems (for which you should submit your solutions, and which will be graded). Some problems have sub-parts that are exercises. For this problem set, its OK to work with others. (Groups of 2, maybe 3 max.) That being said, please think about the problems yourself before talking to others. Please cite all sources you use, and people you work with. The expectation is that you try and solve these problems yourself, rather than looking online explicitly for answers. Submissions due at beginning of class on the due date. Please check the Piazza for details on submitting your *LaTeXed* solutions.

Exercise 1

Simple Samplers. Suppose X is a random variable which takes on values in the interval $[0, 1]$; let $\mathbb{E}[X] = c$. Initially, you dont know anything about c , or about the probability distribution of X . However, you are given a black-box that every time you query it, it gives you an independent random sample drawn according to X . You want to estimate c . A natural scheme is: sample from the black-box N times call these samples X_1, X_2, \dots, X_n and return the empirical mean $\hat{c} := \frac{1}{N} \sum_{i=1}^N X_i$. The natural question is: how big does the number of samples N have to be so that

$$\mathbb{P}[|\hat{c} - c| \leq \epsilon] \geq 1 - \delta \tag{1}$$

I.e., you want to be within error ϵ with confidence $1 - \delta$.

- Use Chebyshev's inequality to show that $N = O\left(\frac{1}{\epsilon^2 \delta}\right)$ samples suffice to Ensure 1. Hence, to get $\delta = \frac{1}{n^k}$ for some value n , we would take $O\left(\frac{n^k}{\epsilon^2}\right)$ samples.
- Hoeffding's bound says the following (you dont have to prove this, of course):

Theorem 1. *Suppose Y_1, Y_2, \dots, Y_T are independent $[0, M]$ -bounded random variables, and define $Y := \sum_{t=1}^T Y_t$ be their sum. Let $\mu = \mathbb{E}[Y]$. Then*

$$\mathbb{P}[Y \geq \mu + \lambda] \leq \exp\left(-\frac{\lambda^2}{M(2\mu + \lambda)}\right)$$

$$\mathbb{P}[Y \leq \mu - \lambda] \leq \exp\left(\frac{-\lambda^2}{2M\mu}\right).$$

Use Hoeffding's bound to show that $N = O\left(\frac{1}{\epsilon^2} \ln \frac{1}{\delta}\right)$ 2 samples are sufficient. Hence, to get $\delta = 1/n^k$ we would take $O\left(\frac{k \log n}{\epsilon^2}\right)$ samples.

Exercise 2

Matrix Multiplication is Useful. Given an undirected simple graph $G = (V, E)$, a triangle is just a clique of size 3; i.e., 3 vertices such that all 3 edges are present. Give algorithms for the following problems:

- Find a triangle in G in time n^ω
- Find a $3k$ -clique in a graph in time $n^{k\omega}$.
- (Bonus) Find a triangle in G in time $m^{1.5}$. (This one is slightly harder not an exercise and does not use matrix multiplication.)

Exercise 3

Low-Diameter Decompositions for Simple Graphs. Recall the concept of a β -low-diameter decomposition, which given graph G and distance D , randomly breaks the graph into pieces of max-distance D , such that each pair x, y is separated with probability at most $\beta \cdot \frac{d_G(x,y)}{D}$

- Show that if each edge $(x, y) \in E(G)$ is cut with probability $\beta \cdot \frac{d_G(x,y)}{D}$, then so is any pair $x, y \in V$. Hence, if the graph only has unit-weight edges, each edge can be cut with probability at most β/D .
- Show that (i) any path graph has an LDD with $\beta = 1$, (ii) any tree with $\beta = 2$, and (iii) the standard k -dimensional $(n^{\frac{1}{k}} \times \dots \times n^{\frac{1}{k}})$ -grid with $\beta = k$.

Exercise 4

Approximation via Randomized Simplification In Lecture #5 we encountered low-stretch spanning trees. Now we use this to solve the following k -median problem on general metrics: given an n -point metric (X, d) , and a number k , find a set C of size k that minimizes $\Phi_d(C) := \sum_{v \in X} d(v, C)$. We outlined an algorithm for the case of distances d_T on a tree T . We now show how to solve this problem (approximately) on a general n -point metric (X, d) .

- Use a low-stretch spanning tree construction on metric (X, d) to sample a tree T from D . Show that if C_X is the optimal solution on metric X , and C_T is the optimal solution on the tree T , then

$$\mathbb{E}[\Phi_{d_T}(C_T)] \leq \mathbb{E}[\Phi_{d_T}(C_X)] \leq \alpha \times \Phi_d(C_X).$$

- Observe that you can solve the clustering problem optimally on T to find the centers C_T using HW1. (Nothing to show here.)
- Show that $\Phi_d(C_T) \leq \Phi_{d_T}(C_T)$.
- Deduce that the expected cost of the solution C_T is at most $O(\log n)$ times $\Phi_d(C_X)$, the cost of the optimal solution. (Hence you have output an $O(\log n)$ -approximate solution to this NP-hard problem.)
- Look at your analysis in the previous part and outline for what kinds of problems on metrics does this method apply. E.g., does it work for the TSP on a metric? How about the K -center problem? Or the K -means problem which wants to minimize $\Psi_d(C) := \sum_{v \in X} (d(v, C))^2$. Why or why not?

Exercise 5

(Approximate MSTs and Shortest Paths Show that the expected weight of a low-stretch spanning tree as above is at most $O(\alpha)$ times the MST.

Problem 1

Another Halving Trick. Suppose you have an algorithm A that, given a directed graph G with integer edge lengths we satisfying $w_e \geq -1$, returns feasible potentials $\Phi : V \rightarrow \mathbb{R}$ in time T (or shows there is a negative cycle). We will show that you can use this to find feasible potentials for graphs with integer weights $w_e \geq -M$ in time $O((T + m) \log M)$ (or shows there is a negative cycle).

- Define weights $w'_e = \lceil w_e/2 \rceil$. Show that if there are no negative cycles with respect to w , there are none w.r.t. w' .

- b. Suppose $\Phi' : V \rightarrow \mathbb{R}$ gives feasible potentials w.r.t. w' . Then show that the reduced cost $\hat{w}_{uv} := 2\Phi'_u + w_{uv} - 2\Phi'_v$ satisfies $\hat{w}_e \geq -1$ for all edges e .
- c. Now use A to find feasible potentials w.r.t. \hat{w} . How does this give feasible potentials w.r.t. w ? Show that if $w_e \geq -M$, we take $O((T + m) \log M)$ time.

Problem 2

Short on Average. Given a directed graph $G = (V, A)$ with possibly negative edge-weights $w_{aa \in A}$, define the weight-ratio of a directed cycle C to be $\rho(C) = \frac{\sum_{a \in C} w_a}{|C|}$. We want to find a cycle C with minimum weight-ratio. Denote this by $\rho^*(G) := \min_{\text{cycles } C} \rho(C)$. (You may assume that edge-weights are integers in the range $[-M, M]$.)

- a. (Nothing to submit.) Observe that you can check if a graph has a negative-weight cycle, using Bellman-Ford-Moore. (If all vertices are reachable from s this is immediate, else what will you do?)
- b. Show how to use Bellman-Ford-Moore to find a cycle of zero weight assuming there is no negative-weight cycle. (Hint: Recall how you used B-F to compute feasible potentials in Johnsons algorithm from Lecture #4.)
- c. Observe: $\rho^*(G) = \max\{\alpha \in \mathbb{Q} \mid G \text{ with arc weights } (w_a - \alpha) \text{ has no negative cycle}\}$. Use this observation to compute $\rho^*(G)$, and also find a cycle C of this weight ratio. Your algorithm should run in time $O(mn(\log M + \log n))$ time. Caveat: binary search over a range of K integers takes $O(\log K)$ time. But binary search over rationals or reals may not terminate.

Problem 3

Sparsification in Dynamic Algorithms. In this problem we develop an approach to transform an $O(m^{2/3})$ update time algorithm for dynamic connectivity into an $O(n^{2/3} \log n)$ one. For all parts, assume all edge sets are over vertex set V with $|V| = n$, and all graphs are simple. Let $K = \lceil \frac{n-1}{2} \rceil$.

- a. For two edge sets E, E' with spanning forests F, F' respectively, show that any spanning forest F'' of $F \cup F'$ is also a spanning forest of $E \cup E'$.
- b. A sparse partition of edge set E is a partition into K sets (E^1, E^2, \dots, E^K) , where some prefix E^1, \dots, E^k all have exactly n edges, E^{k+1} is potentially partially full, and E^{k+2} onwards are empty. Suppose E has sparse partition (E^1, E^2, \dots, E^K) . Suppose E' is obtained from E by either adding or removing one edge. Show how to get a sparse partition for E' such that $(E')^i = E^i$ for all but (at most) two indices i , and for these two indices i the size of the symmetric difference $|(E')^i \Delta E^i| \leq 2$.
- c. Define a complete binary tree with leaves $\{1, 2, \dots, K\}$. Each internal node corresponds to some interval $i \dots j$, define $E^{ij} := \cup_{s=i}^j E^s$, and F^{ij} be a spanning forest on E^{ij} . This defines a collection of $O(K)$ spanning forests. Given a sparse partition (E^1, E^2, \dots, E^K) for E , and an associated collection of spanning forests, suppose we add or remove a single edge from one of the E^i s. Show how to maintain the resulting collection of spanning forests with total update time $O(n^{2/3} \log n)$. Infer that we can maintain dynamic connectivity in $O(n^{2/3} \log n)$ time.

Problem 4

Sparse Spanners. Given a graph G with edge lengths ℓ_e , a subgraph H is a spanner with stretch $\gamma \geq 1$ if for every edge $(x, y) \in E(G)$, $d_H(x, y) \leq \gamma \cdot d_G(x, y)$.

- a. Use the triangle inequality to show that for all $x, y \in V$, even if (x, y) is not an edge, $d_H(x, y) \leq \gamma \cdot d_G(x, y)$. Clearly if $H = G$, we can set $\gamma = 1$. The goal is to find H with few edges, such that γ is also small. Lets give two different constructions of good spanners.
- b. *Approach #1.* Consider the following randomized process: sample $t = 4 \log n$ trees T_1, T_2, \dots, T_t from an α -stretch probabilistic embedding into subtrees. Let H be the union of all these edges.

- (a) Show that for any fixed edge $(x, y) \in E(G)$,

$$P[d_H(x, y) \geq 2\alpha d_G(x, y)] \leq 2^{-t}.$$

(Hint: for any single value of i , bound $\mathbb{P}[d_{T_i}(x, y) \geq 2\alpha d_G(x, y)]$.)

- (b) Use the results quoted in Lecture to show that with probability $1 - \frac{1}{n^2}$, the graph H is an $O(\log n \log \log n)$ -stretch spanner with $O(n \log n)$ edges.
- c. *Large-girth graphs are Sparse.* On a seemingly different note, here is a useful graphtheoretic fact. Define the girth of a graph G is the length of the shortest cycle in G . We will show that any graph G with m edges and n nodes, and girth strictly more than g must have $m \leq O(n + n^{1 + \frac{1}{\lfloor g/2 \rfloor}})$.
 - (a) G has average degree $\hat{d} := \frac{2m}{n}$. Show that there exists a subset $S \subseteq V$ such that the induced subgraph $H := G[S]$ has minimum degree at least $\hat{d}/2$. [Hint: drop some low-degree vertices.]
 - (b) . For this graph H and any vertex $v \in H$, show that the number of distinct vertices at distance at most $\lfloor g/2 \rfloor$ from v is at least $(\hat{d}/2 - 1)^{\lfloor g/2 \rfloor}$.
 - (c) Show that the number of edges in the original graph G , i.e., $m \leq O(n + n^{1 + \frac{1}{\lfloor g/2 \rfloor}})$.
- d. (d) *Approach #2.* Now consider a variant of Kruskals algorithm for $\alpha \geq 1$. Consider the edges of G in increasing order of lengths e_1, e_2, \dots, e_m . Initialize $H_0 = \emptyset$. When considering edge $e_i = (x, y) \in E(G)$, if the current distance $d_{H_{i-1}}(x, y) \leq \alpha d_G(x, y)$, then discard e (i.e., set $H_i \leftarrow H_{i-1}$, else set $H_i \leftarrow H_{i-1} \cup \{e_i\}$.
 - (a) (Do not submit.) Show that if we set $\alpha = n - 1$, then you will get Kruskals algorithm. Also, observe that by construction, the graph H at the end of the process is an $(n - 1)$ -stretch spanner. (In fact, an $(n - 1)$ -stretch spanning tree.)
 - (b) If we set $\alpha = O(\log n)$, use (c) with $g = O(\log n)$ to show the final graph H is a $O(\log n)$ -stretch spanner with $O(n)$ edges.