Your Name	September 10, 2024
6.5240 Problem Set 1	
Collaborators: *Firstname Lastname*	Due date: September 18, 2024

Homework guidelines: You may work with other students, as long as (1) they have not yet solved the problem, (2) you write down the names of all other students with which you discussed the problem, and (3) you write up the solution on your own. No points will be deducted, no matter how many people you talk to, as long as you are honest. If you already knew the answer to one of the problems (call these "famous" problems), then let us know that in your solution writeup – it will not affect your score, but will help us in the future. It's ok to look up famous sums and inequalities that help you to solve the problem, but don't look up an entire solution.

1. **Estimating average degrees via empirical sampling:** In this exercise, you will compute the expectation and bound the variance of the following algorithm for estimating the average degree $\overline{d}(G)$ of a graph G, and you will show that these quantities alone are not enough to show a sublinear bound on the query complexity.

   Empirical Average via Vertex Sampling

   Require: Query access to unknown graph G, number of vertices n, parameter k
   1 Take k uniformly random vertices $v_1, \ldots, v_k$
   2 Make k degree queries to get their degrees $d_1, \ldots, d_k$
   3 Return the empirical average $\hat{d} := \frac{1}{k}\sum_{i=1}^{k} d_i$

   a) Show that $\mathrm{Var}\left[\hat{d}\right] \leq \frac{1}{nk}\sum_{v\in V} d_v^2$.

   b) Deduce that, to get a 2-approximation with probability $\geq 9/10$, it would suffice to have
   $$k \geq 40n \cdot \frac{\sum_{v\in V} d_v^2}{(\sum_{v\in V} d_v)^2}$$

   c) Show that, for every connected graph G,
   $$\frac{\sum_{v\in V} d_v^2}{(\sum_{v\in V} d_v)^2} \leq \frac{1}{2}$$
   Conclude that taking $k \geq 20n$ is enough.

   d) Unfortunately, this is a number of samples that is *linear in* n, while we are hoping for much smaller ($O(\sqrt{n})$). One way to achieve this would be to show that:
   $$\frac{\sum_{v\in V} d_v^2}{(\sum_{v\in V} d_v)^2} \leq O(1/\sqrt{n}).$$

Unfortunately, this is not true in general: show that for every $n \geq 2$, there exists a connected graph G on n vertices such that, for this graph, $\sum_{v \in V} d_v^2 = \Omega(n^2)$ and $\sum_{v \in V} d_v = O(n)$, and so

$$\frac{\sum_{v \in V} d_v^2}{(\sum_{v \in V} d_v)^2} = \Omega(1)$$

(Hint: we have discussed this graph in class.)

2. **A simple algorithm for 2-approximation:** This problem shows how to analyse our algorithm in Problem 1 better, going beyond a naive expectation/variance argument. Let $\beta \in (0, 1)$ be a constant. Given a connected graph G with $n \geq 2$ vertices and m edges, we say a vertex $v \in V$ is *light* if $d_v < \sqrt{m/\beta}$, and *heavy* otherwise. Similarly, an edge is heavy if both its endpoints are heavy, light if both are light, and medium otherwise.

   a) Show that there are at most $2\sqrt{\beta m}$ heavy vertices.

   b) Deduce that there are at most $2\beta m$ heavy edges.

   c) Letting m_L be the number of light edges and m_M the number of medium edges, show that

   $$\sum_{v \text{ light}} d_v = 2m_L + m_M$$

   d) Deduce from the above that $\sum_{v \text{ light}} d_v \geq (1 - 2\beta)m$, and that

   $$\frac{1 - 2\beta}{2} \cdot \overline{d}(G) \leq \frac{1}{n} \sum_{v \text{ light}} d_v \leq \overline{d}(G).$$

   e) This suggests that it would be sufficient, to get a 2-estimate, to only consider the light vertices. Of course, one issue is that given a vertex v, we cannot tell whether it is a light vertex or not (since this would require checking if $d_v < \sqrt{m/\beta}$, and we don't know m). But we can still use this for the analysis of our algorithm to reduce the number of samples k needed.

   Consider the following thought experiment: instead of defining

   $$\hat{d} := \frac{1}{k} \sum_{i=1}^{k} d_i$$

   we instead define

   $$\hat{d}_L := \frac{1}{k} \sum_{i=1}^{k} d_i \mathbf{1}[v_i \text{ is light}].$$

   Show that $\frac{1-2\beta}{2} \cdot \overline{d}(G) \leq \mathrm{E}\left[\hat{d}_L\right] \leq \overline{d}(G)$.

f) Show that $\text{Var}\left[\hat{d}_L\right] \leq \frac{\sqrt{m/\beta}}{k} \cdot \text{E}\left[\hat{d}_L\right]$.

g) Argue that $\hat{d} \geq \hat{d}_L$ (always), and that $\Pr[\hat{d} < t] \leq \Pr[\hat{d}_L < t]$ for all t.

h) Deduce that, for $0 \leq \beta < 1/4$,

$$\Pr\left[\hat{d} < \frac{(1-2\beta)^2\overline{d}(G)}{2}\right] \leq \frac{\sqrt{m/\beta}}{k\beta^2\text{E}\left[\hat{d}_L\right]} \leq \frac{1}{\beta^{5/2}(1-2\beta)} \cdot \frac{n}{k\sqrt{m}} \leq \frac{2\sqrt{2}}{\beta^{5/2}} \cdot \frac{\sqrt{n}}{k}$$

(Prove all 3 inequalities. The constants don't matter much; don't try to optimize them.)

i) Show that $\Pr\left[\hat{d} > \frac{2}{1-2\beta}\overline{d}(G)\right] \leq \frac{1-2\beta}{2}$.

j) Conclude by showing that, for every $\varepsilon \in (0,1)$, there exists $C = C(\varepsilon)$ and a constant $c > 0$ such that the algorithm, run with parameter $k \geq C\sqrt{n}$, returns a $(2+\varepsilon)$-estimate of $\overline{d}(G)$ with probability at least $1/2 + c\varepsilon$.

k) Briefly explain how to amplify the probability of success to 9/10.

3. **Estimating the number of triangles:** In this problem we will design and analyze an algorithm for estimating the number of triangles in a graph, assuming access to *uniform edge samples* in addition to the degree, pair, and uniform neighbor queries of the general query model. Also assume that the number of edges m is known, as well as a lower bound $\hat{T}$ such that $\hat{T} < T(G)$, where $T(G)$ is the number of triangles in G. Our final running time will be $\mathsf{poly}(1/\epsilon, 1/\hat{T}) \cdot o(m^3)$, thus a better lower bound $\hat{T}$ on the number of triangles will yield an improved running time.

a) Consider sampling an edge uniformly, then choosing one of its endpoints uniformly. For a vertex v, what is the probability that v is the chosen vertex, in terms of d_v?

b) Let heavy vertices be defined as those with degree at least $\sqrt{2m}$. Give a procedure that uses $O(1)$ queries and outputs a vertex u (or $\perp$) as follows: if u is heavy, then it is returned with probability $1/\sqrt{2m}$. If u is light then it is returned with 0 probability.

c) Recall from lecture that every vertex has at most $\sqrt{2m}$ *outgoing* neighbors (neighbors with higher degree, with ties broken by vertex ID). Design a procedure for sampling outgoing neighbors such that each outgoing neighbor is sampled with probability $1/\sqrt{2m}$. As in (b), the procedure may output $\perp$. (Hint: consider separately nodes of degree $> \sqrt{2m}$ and nodes of degree $\leq \sqrt{2m}$.

d) Prove that we can assign each triangle to one of its edges such that each edge has at most $\sqrt{2m}$ assigned triangles. (Hint: consider ordering the vertices according to degree).

e) Design an algorithm that samples each triangle with probability $\frac{1}{m\sqrt{2m}}$. As in (b) and (c), the algorithm may output $\perp$.

f) Design an algorithm that outputs an estimate t such that, with probability at lesat 2/3, $(1-\varepsilon)T(G) \leq t \leq (1+\varepsilon)T(G)$.

4. **A query lower bound for deterministic algorithms.** Prove that any deterministic algorithm for estimating the number of connected components to within an $\epsilon \cdot n$ additive factor in the adjacency list model requires $\Omega(n)$ time for some constant $\epsilon \in (0, 1)$.

   More specifically, for any deterministic algorithm A that does not require $\Omega(n)$ time, show that for infinitely many n, there are pairs of graphs (G, G') (each over n vertices) such that the number of connected components are very different (off by at least $2\epsilon n$), but A outputs the same answer on them.