Problem Set 4 CS265/CME309, Winter 2025
Due: Friday 2/14, 11:59pm on Gradescope

Please follow the homework policies on the course website.

---

1. **(14 pt.) [Another way to sketch sparse vectors.]** Suppose that A is an list of length n, containing elements from a large universe $\mathcal{U}$. Our goal is to estimate the frequencies of each element in $\mathcal{U}$: that is, for $x \in \mathcal{U}$, how often does x appear in A?

   The catch is that A is too big to look at all at once. Instead, we see the elements of A one at a time: $A[0], A[1], A[2], \ldots$. Unfortunately, $\mathcal{U}$ is also really big, so we can't just keep a count of how often we see each element.

   In this problem, we'll see a construction of a randomized data structure that will keep a "sketch" of the list A, use small space, and will be able to efficiently answer queries of the form "approximately how often did x occur in A"?

   Specifically, our goal is the following: we would like a (small-space) data structure, which supports operations `update`(x) and `count`(x). The `update` function inserts an item $x \in \mathcal{U}$ into the data structure. The `count` function should have the following guarantee, for some $\delta, \epsilon > 0$. After calling `update` n times, `count`(x) should satisfy

   $$C_x \leq \texttt{count}(x) \leq C_x + \epsilon n \tag{1}$$

   with probability at least $1 - \delta$, where C_x is the true count of x in A.

   (a) **(3 pt.)** Your friend suggests the following strategy (this will not be our final strategy). We start with an array R of length b initialized to 0, and a random hash function $h : \mathcal{U} \to \{0, 1, \ldots, b-1\}$. You can assume that h is drawn from some universal hash family, i.e $P(h(x) = h(y)) = 1/b$ for any $x \neq y$. Then the operations are:

   - `update`(x): Increment $R[h(x)]$ by 1.
   - `count`(x): Return $R[h(x)]$.

   For every entry $A[i]$ in the list it encounters, the scheme calls `update`$(A[i])$.

   After sequentially processing all n items in the list, what is the expected value of `count`(x)?

   (b) **(2 pt.)** Show that there is a choice of b that is $O(1/\epsilon)$ so that, for any fixed $x \in \mathcal{U}$, we have

   $$\Pr[\texttt{count}(x) < C_x] = 0$$

   and

   $$\Pr[\texttt{count}(x) \geq C_x + \epsilon n] \leq \frac{1}{e}.$$

   [**HINT:** *The first of the requirements is true no matter what b is.*]

   (c) **(2 pt.)** Explain how you would use T copies of the construction in part (a) to define a data structure that, for any fixed $x \in \mathcal{U}$, satisfies (1) with high probability. How big do you need to take T so that the (1) is satisfied with probability at least $1 - \delta$? How much space does your modified construction use? (It should be sublinear in $|\mathcal{U}|$ and n).

Give a complete description and analysis of the data structure, and explain how much space it uses. You may assume that it takes $O(\log |\mathcal{U}|)$ bits to store the hash function h and $O(\log n)$ to store each element in the array R.

(d) Explain how to use your algorithm to solve the following problem:

i. **(4 pt.)** Given a k-sparse vector $a \in \mathbb{Z}^N_{\geq 0}$ ($\mathbb{Z}_{\geq 0}$ is the set of non-negative integers), design a randomized matrix $\Phi \in \mathbb{R}^{m \times N}$ for $m = O(\frac{k \log N}{\epsilon})$ so that the following happens. With probability at least 0.99 over the choice of Φ, you can recover $\tilde{a}$ given Φa, so that simultaneously for all $i \in 1, ..., N$, we have

$$|\tilde{a}[i] - a[i]| \leq \frac{\epsilon \|a\|_1}{2k}.$$

[**HINT:** *Think of the k-sparse vector a as being the histogram of the items in the list A from the previous parts.*]
[**HINT:** *How can you represent a hash function as a matrix multiplication?*]
[**HINT:** *Note that we want a tighter bound, and we want the bound to hold simultaneously for all i. How can we change b and T to achieve this?*]

ii. **(3 pt.)** Now, assuming the above holds for all i, use the k-sparseness of a to construct $\hat{a}$ from $\tilde{a}$ such that

$$\|\hat{a} - a\|_1 \leq \epsilon \|a\|_1.$$

iii. **(0 pt.) [This question is zero points, but worth thinking about.]** How does the guarantee in the previous part compare to the RIP matrices (and the compressed sensing guarantee that we can get from them, Theorem 1 in the Lecture 9 lecture notes) that we saw in class? (i.e., is this guarantee weaker? Stronger? Incomparable? The same?)

2. **(7 pt.) [Dominating set.]** Let $G = (V, E)$ be an undirected graph with n vertices. A *dominating set* is a subset U of vertices such that every vertex $v \in V \setminus U$ is adjacent to at least one vertex in U.

Suppose that G has minimum degree δ (that is, every vertex is adjacent to at least δ distinct vertices). In this problem, we will prove that G has a dominating set of size at most $n \cdot \frac{1+\ln(\delta+1)}{\delta+1}$.

(a) **(1 pt.)** Consider any set of vertices $S \subseteq V$. Let

$$T = \{v \in V \setminus S \mid v \text{ is not adjacent to any vertex in } S\}.$$

explain why $S \cup T$ is a dominating set.

(b) **(6 pt.)** Prove that G has a dominating set of size at most $n \cdot \frac{1+\ln(\delta+1)}{\delta+1}$. You may use without proof the fact that $1 - p \leq e^{-p}$ for any nonnegative p.
[**HINT:** *Use part (a)...*]

3. **(9 pt.)** Suppose we are investigating the social habits in a group of n chimpanzees, and after months of observations, for every pair of chimpanzees A and B, we know whether A has spent more time grooming B or whether B has spent more time grooming A. All of these pairwise relationships together are called the *grooming habit* of the n chimpanzees. (We assume that no pair spent equal time grooming each other). We wish to aggregate these pairwise comparisons into a single ranking of chimpanzees by altruism. Given a ranking, we say that a pair A, B of chimps is a *violated pair* if $A \geq B$ in the ranking, but in real life, A spent less time grooming B than B spent grooming A.

   (a) (2 points) Prove that there exists a ranking that violates at most half of the pairwise relationships.

   (b) (1 point) Prove that there exists a ranking that violates strictly less than half of the pairwise relationships.

   (c) (6 points) Define a "good" ranking as one that violates at most 49% of the pairwise relationships. Prove that for sufficiently large n, there exist grooming habits with no good rankings.

   [**HINT:** *Use the probabilistic method. Suppose towards a contradiction that every grooming habit has a good ranking; for a fixed grooming habit, what's the probability that a random ranking is good for it? What does that say about the probability that a random ranking is good for a* random *grooming habit? Can you find a contradiction here? (Perhaps by studying a fixed ranking and a random grooming habit?)*]