Problem Set 5 CS265, Winter 2022
Due: February 9 (Wednesday) at 11:30 (Pacific Time)

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. **(10 pt.)** The *equilateral dimension* of a metric space is the maximum number of points in the space that are all at the same distance from each other. In this problem, we will determine the equilateral dimension of the d-dimensional Euclidean space $\mathcal{E}^d = (\mathbb{R}^d, \ell_2)$, and also explore a relaxed version of the equilateral dimension where the pairwise distances are only required to be *approximately* the same.

(a) **(2 pt.)** Let X be a set of $d+1$ points in $\mathcal{E}^m$ for some arbitrary positive integer m. Show that (X, ℓ_2) can be isometrically embedded into $\mathcal{E}^d$.
[**HINT:** *Suppose $X = \{x_0, \ldots, x_d\}$. Consider the linear subspace spanned by the vectors $x_i - x_0$ for $i = 1, \ldots, d$. How large can the dimension of the subspace be?*]

(b) **(2 pt.)** For all positive integers d, use Part (a) to show that there exist $d+1$ points in $\mathcal{E}^d$ that are all at distance 1 from each other. This shows that the equilateral dimension of $\mathcal{E}^d$ is at least $d+1$.

(c) **(3 pt.)** Show that there exists a constant $c > 0$ such that for all positive integers d, one can find at least 2^{cd} points in $\mathcal{E}^d$ that are all at distances between 1 and 1.1 from each other.
[**HINT:** *Try applying the Johnson–Lindenstrauss lemma.*]

(d) **(0 pt.) [Optional: this won't be graded.]** For all positive integers d, show that one cannot find $d+2$ points in $\mathcal{E}^d$ that are all at distance 1 from each other. Together with Part (b), this shows that the equilateral dimension of $\mathcal{E}^d$ is exactly $d+1$.
[**HINT:** *Prove by contradiction. Suppose there are $d+2$ such points $x_0, x_1, \ldots, x_{d+1}$. Consider the vectors $v_i = x_i - x_0$ and the matrix $A = [v_1, \cdots, v_{d+1}]$ (with the v_i as columns) of size $d \times (d+1)$. Explicitly compute the inner products $v_i^\mathsf{T} v_j$ and the matrix $A^\mathsf{T}A$. Use the fact that* $\mathrm{rank}(A^\mathsf{T}A) = \mathrm{rank}(A)$ *to derive a contradiction.*[1]]

(e) **(3 pt.)** Show that there exists a constant $C > 0$ such that for all positive integers d, one cannot find more than 2^{Cd} points in $\mathcal{E}^d$ that are all at distances between 1 and 1.1 from each other.
[**HINT:** *You can use the fact that the volume of a d-dimensional ball of radius r scales as γr^d for some constant γ that depends on d but not r. It may be helpful if you understand the proof of Lemma 3 in lecture notes #9, but that is not required for working on this problem.*]

3. **(0 pt.) [Optional: this won't be graded.]** Prove that the Johnson-Lindenstrauss lemma is "nearly tight".[2] Specifically, for a sufficiently small positive constant c, show that for every positive integer $n > 100$ and any $\epsilon \in [1/\sqrt{n}, 1/10)$, one can find a positive integer d together with a set X of n points in $\mathbb{R}^d$ such that for any positive integer $m < \frac{c \ln n}{\epsilon^2 \ln(1/\epsilon)}$, there is no way to embed (X, ℓ_2) into $(\mathbb{R}^m, \ell_2)$ with distortion $1+\epsilon$. You can use the following theorem in linear algebra without proof:

Theorem 1 (Alon '03)**.** *There is a small positive constant c with the following property. Let $A = (a_{ij})$ be an n by n real matrix with $a_{ii} = 1$ for all i and $|a_{ij}| \le \epsilon$ for all $i \neq j$. If $n > 4$ and $\epsilon \in [1/\sqrt{n}, 1/2)$, then*

$$\mathrm{rank}(A) \ge \frac{c \ln n}{\epsilon^2 \ln(1/\epsilon)}.$$

[**HINT:** *Try using a similar proof strategy as Problem 2(d).*]

[1] This proof strategy can be used to show that the Johnson-Lindenstrauss lemma is "nearly tight". See Problem 3.

[2] For improved results, see paper by Larsen and Nelson: Optimality of the Johnson-Lindenstrauss Lemma. (Link to the FOCS 2017 version: `https://ieeexplore.ieee.org/abstract/document/8104096`).