New York University Tandon School of Engineering
Computer Science and Engineering

CS-GY 6763: Homework 2.
Due Monday, March 7th, 2022, 11:59pm ET.

Collaboration is allowed on this problem set, but solutions must be written-up individually. Please list collaborators for each problem separately, or write "No Collaborators" if you worked alone.

Problem 1: Johnson-Lindenstrauss Approximates Inner Products.

(10 pts)

1. Suppose that Π is a Johnson-Lindenstrauss matrix with $k = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$ rows. Prove that for any x, y:

$$|\langle x, y\rangle - \langle \Pi x, \Pi y\rangle| \leq \epsilon(\|x\|_2^2 + \|y\|_2^2)$$

   with probability $\geq 1 - \delta$. **Hint:** Try exploiting connections between norms and inner products.

2. Show that a better bound can be obtained by augmenting our sketches with the norms of each vector. I.e., instead of just Πx and Πy, store as the sketches for x and y both $(\Pi x, \|x\|_2)$ and $(\Pi y, \|y\|_2)$ and show that using these sketches we can compute an estimate z such that

$$|\langle x, y\rangle - z| \leq \epsilon(\|x\|_2\|y\|_2)$$

   with $1 - \delta$ probability when $k = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$. By the AM-GM inequality, we always have that $\|x\|_2\|y\|_2 \leq \|x\|_2^2 + \|y\|_2^2$.

Problem 2: Estimating Quantiles in a Stream

(20 points) Consider the standard turnstile model of streaming discussed in class, where there is an underlying frequency vector $f \in \mathbb{R}^n$, initialized to $\vec{0}$, and which receives a sequence of updates to its coordinates $(i, \Delta) \in [n] \times \mathbb{Z}$, causing the change $f_i \leftarrow f_i + \Delta$.

- **(10 points)** The sample variance of a vector $f \in \mathbb{R}^n$ is defined to be

$$v = \frac{1}{n}\sum_{i=1}^{n}(f_i - \mu)^2$$

  where $\mu = \sum_i \frac{f_i}{n}$. Show how to output a number $\tilde{v}$ such that, with probability at least $9/10$, we have $(1-\epsilon)\tilde{v} \leq v \leq (1+\epsilon)\tilde{v}$. Your algorithm should use at most $O(1/\epsilon^2)$ words of space, and you may assume the algorithm knows the number n in advance.

- **(10 points)** For exactly one of the functions 1) : $g_1(f) = \sum_{i=1}^{n}(f_i^2 - 10f_i + 16)$, and 2) : $g_2(f) = \sum_{i=1}^{n}(f_i^2 - 8f_i + 16)$, it is possible, with probability at least $2/3$, to output a number $\tilde{g}$ such that $g(f)/2 \leq \tilde{g} \leq 3g/2$, and for which the algorithm uses $O(1)$ words of space. For the other function, any algorithm requires at least $\Omega(n)$ bits of space to output such an $\tilde{g}$ with probability at least $2/3$. Show which function is which, and prove why in both cases.

  For your lower bound argument, you should use the fact that any randomized algorithm which, with probability at least $2/3$, distinguishes at the end of the stream between the case that either all coordinates f_i are in the set $\{0, 1\}$, or there is exactly one $i \in [n]$ for which $f_i \notin \{0, 1\}$, requires $\Omega(n)$ bits of space. Let us refer to this problem as problem $\mathcal{P}$.

  Hint: Think about being given an input stream to problem $\mathcal{P}$, modifying the stream in a certain way, and using the output $\tilde{g}$ of an algorithm for one of the functions above, run on a modified stream, to solve problem $\mathcal{P}$.

Problem 3: Compressed classification.

(10 pts) In machine learning, the goal of many classification methods (like support vector machines) is to separate data into classes using a *separating hyperplane*.

Recall that a hyperplane in $\mathbb{R}^d$ is defined by a unit vector $a \in \mathbb{R}^d$ ($\|a\|_2 = 1$) and scalar $c \in \mathbb{R}$. It contains all $h \in \mathbb{R}^d$ such that $\langle a, h\rangle = c$.

Suppose our dataset consists of n unit vectors in $\mathbb{R}^d$ (i.e., each data point is normalized to have norm 1). These points can be separated into two sets X, Y, with the guarantee that there exists a hyperplane such that every point in X is on one side and every point in Y is on the other. In other words, for all $x \in X, \langle a, x\rangle > c$ and for all $y \in Y, \langle a, y\rangle < c$.

Furthermore, suppose that the ℓ_2 distance of each point in X and Y to this separating hyperplane is at least ϵ. When this is the case, the hyperplane is said to have "margin" ϵ.

1. Show that this margin assumption equivalently implies that for all $x \in X, \langle a, x\rangle > c + \epsilon$ and for all $y \in Y, \langle a, y\rangle < c - \epsilon$.

2. Show that if we use a Johnson-Lindenstrauss map Π to reduce our data points to $O(\log n/\epsilon^2)$ dimensions, then the dimension reduced data can still be separated by a hyperplane with margin $\epsilon/4$, with high probability (say $> 9/10$).

Problem 4: Join Size Estimation

(15 pts) One powerful application of sketching is in database applications. For example, a common goal is to estimate the *inner join size* of two tables without performing an actual inner join (which is expensive, as it requires enumerating the keys of the tables). Formally, consider two sets of keys $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_n\}$ which are subsets of $1, 2, \ldots, U$. Our goal is to estimate $|X \cap Y|$ based on small space compressions of X and Y. We consider two approaches below.

1. Using your result from Problem 1, describe a method based on inner product estimation that constructs independent sketches of X and Y of size $k = O\left(\frac{1}{\epsilon^2}\right)$ and from these sketches can return an estimate Z for $|X \cap Y|$ satisfying

$$|Z - |X \cap Y|| \leq \epsilon\sqrt{|X||Y|}$$

   with probability 9/10.

2. Alternatively, consider compressing the sets as follows:

   - Choose k uniform random hash functions $h_1, \ldots, h_k : \{1, \ldots, U\} \to [0, 1]$.
   - Let $C^X = [C_1^X, \ldots, C_k^X]$ where $C_i^X = \min_{j=1,\ldots,m} h_i(x_j)$.
   - Let $C^Y = [C_1^Y, \ldots, C_k^Y]$ where $C_i^Y = \min_{j=1,\ldots,n} h_i(y_j)$.

   Given the sketches C^X and C^Y., which each contain k numbers, we estimate join size as $Z = \frac{k'}{k} \cdot (\frac{1}{S} - 1)$ where $k' \leq k$ equals $k' = \sum_{i=1}^{k} \mathbb{1}[C_i^X = C_i^Y]$ and

$$S = \frac{1}{k}\sum_{i=1}^{k} \min(C_i^X, C_i^Y).$$

   Show that if we set $k = O(\frac{1}{\epsilon^2})$ then with probability 9/10,

$$|Z - |X \cap Y|| \leq \epsilon\sqrt{|X \cap Y||X \cup Y|}.$$

   In your proof, you may use the following fact: given a uniform hash function h and a set $A = \{a_1, \ldots, a_T\}$, define the random variable $M = \min_{i=1,\ldots,T} h(a_i)$. Then it holds that $\mathbb{E}[M] = \frac{1}{T+1}$, and $\mathrm{Var}(M) \leq (\frac{1}{T+1})^2$.

   Hint: Think about k'/k and $(\frac{1}{S} - 1)$ separately. What quanties do we expect these random variables to be close to?

3. Which method give better accuracy? The JL based method or the hashing based method?