CSCE 658: Randomized Algorithms – Spring 2024
Problem Set 3

Due: Thursday, March 7, 2024, 5:00 pm CT

Problem 1. (30 points total) CountSketch tail bounds.

For any vector $x \in \mathbb{R}^n$ and any integer $k \geq 0$, we define $\textsc{Tail}_k(x)$ to be the vector x, but with the k entries of largest magnitude to be set to 0, breaking ties arbitrarily. For example if $x = (-100, 40, 40, 1)$, then $\textsc{Tail}_2(x)$ can be either $(0, 0, 40, 1)$ or $(0, 40, 0, 1)$.

1. (5 points) Show that for any parameter $\alpha \geq 1$ and $k \leq n-1$, there exists $x \in \mathbb{R}^n$ such that

$$\alpha \cdot \|\textsc{Tail}_k(x)\|_2 < \|x\|_2.$$

   That is, the length of a tail vector of x can be arbitrarily smaller than the length of the vector x.

2. (20 points) Show that CountSketch actually provides an L_2 tail guarantee. More specifically, for $\varepsilon \in (0,1)$, suppose we use CountSketch with $\mathcal{O}\left(\frac{1}{\varepsilon^2} \cdot \log n\right)$ buckets to extract estimates $\widehat{x_i}$ for the value of each coordinate x_i. Show that with probability $1 - \frac{1}{n^2}$, we simultaneously have that for all $i \in [n]$,

$$|\widehat{x_i} - x_i| \leq \varepsilon \cdot \|\textsc{Tail}_k(x)\|_2,$$

   where $k = \frac{1}{\varepsilon^2}$.

   HINT: The analysis in class demonstrated an error of $\varepsilon \cdot \|x\|_2$. For each $i \in [n]$, what event needs to occur for the top k coordinates to not affect the estimate $\widehat{x_i}$ of x_i?

3. (5 points) Conclude that at the end of an insertion-deletion stream, CountSketch with $\mathcal{O}(k \log n)$ buckets can with high probability, recover the exact coordinates of a vector that is k-sparse, even if at intermediate times in the stream, the underlying frequency is not k-sparse.

Problem 2. (30 points total) F_p moment estimation.

Let $p \geq 1$ be a fixed constant. Suppose $f \in \mathbb{R}^n$ is defined by an insertion-only stream of length m, where each update increments a coordinate of f. Suppose we sample an update $t \in [m]$ in the stream, uniformly at random, and set a counter c to be the number of times the item appears in the stream after time t (including time t). After the stream ends, we set $Z = c^p - (c-1)^p$.

For example, suppose the stream consists of the updates $1, 2, 2, 1, 4, 1, 2, 1$, which induces the frequency vector $f = (4, 3, 0, 1)$ and suppose we sample the fourth update of the stream, corresponding to a 1. Then we see a total of three instances of 1, after that time (inclusive), so that $c = 3$ and $Z = 3^p - 2^p$. For $p = 3$ then, we would have $Z = 27 - 8 = 19$.

1. (5 points) Show that $\mathbb{E}[Z] = f_j^{p-1}$, *conditioned* on sampling $j \in [n]$.

2. (5 points) Let $F = m \cdot Z$. Show that $\mathbb{E}[F] = \|f\|_p^p$.

3. (10 points) Show that $\text{Var}[F] \le p \cdot \|f\|_1 \cdot \|f\|_{2p-1}^{2p-1}$.

   HINT: You may use the fact that for all $x \ge 1$ and $p \ge 1$, we have $x^p - (x-1)^p \le px^{p-1}$.

4. (10 points) Give an algorithm that uses $\mathcal{O}\left(\frac{1}{\varepsilon^2} n^{1-1/p}\right) \cdot \log(nm)$ bits of space and with probability at least $\frac{2}{3}$, outputs an estimate $\widehat{F}$ such that

   $$(1-\varepsilon)\|f\|_p^p \le \widehat{F} \le (1+\varepsilon)\|f\|_p^p.$$

   Justify both its correctness-of-approximation and space complexity.

   HINT: You may use the fact that for all $\|f\|_1 \cdot \|f\|_{2p-1}^{2p-1} \le n^{1-1/p}\|f\|_p^{2p}$.

Problem 3. (30 points total) Easy as 123 (approximate counting).

1. (3 points) Suppose we want to count the number of updates, i.e., the length of a data stream. Describe a naïve streaming algorithm that uses $\mathcal{O}(\log m)$ bits of space if the stream has length m, where m is not known in advance.

Consider the following algorithm:

Algorithm 1 Approximate counting

```
C ← 0
for each stream update do
    Flip a coin that is HEADS with probability 1/2^Z
    if the coin is HEADS then
        C ← C + 1
return Z = 2^X − 1
```

2. (9 points) Compute, with proof, $\mathbb{E}[Z]$.

   HINT: Use induction on the length m of the stream.

3. (9 points) Compute, with proof, $\text{Var}[Z]$.

   HINT: Use induction on the length m of the stream.

4. (9 points) Give an algorithm that uses $\mathcal{O}(\log\log m)$ bits of space and with probability at least $\frac{2}{3}$, outputs an estimate $\widehat{M}$ such that

   $$\frac{m}{2} \le \widehat{M} \le 2m,$$

   where m is the length of the stream, but is not known in advance. Justify both its correctness-of-approximation and space complexity.

Problem 4. (30 points total) Communication complexity.

In the index problem, Alice has a vector $x \in \{0,1\}^n$ and Bob has a position $i \in [n]$ and their goal is for Bob to determine whether $x_i = 0$ or $x_i = 1$ after receiving a message from Alice. It is known that any protocol for indexing that succeeds with probability at least $\frac{2}{3}$ requires $\Omega(n)$ communication from Alice and Bob.

1. (10 points) Suppose a frequency vector $x \in \mathbb{R}^n$ is implicitly defined through a insertion-only data stream requires $\Omega(n)$ space. Let $\mathcal{A}$ be a streaming algorithm that processes x, receives a query $i \in [n]$ *after the data stream*, and outputs x_i with probability at least $\frac{2}{3}$. Show by a reduction from indexing that $\mathcal{A}$ must use $\Omega(n)$ bits of space.

In the set-disjointness communication, Alice has a vector $x \in \{0,1\}^n$ and Bob has a vector $y \in \{0,1\}^n$ and their goal is to determine whether there exists an index $i \in [n]$ such that $x_i = y_i = 1$. It is known that any protocol for set-disjointness that succeeds with probability at least $\frac{2}{3}$ requires $\Omega(n)$ communication between Alice and Bob.

2. (10 points) Show that any streaming algorithm that with probability at least $\frac{2}{3}$, outputs the largest coordinate $i \in [n]$ of a frequency vector $x \in \mathbb{R}^n$ that is implicitly defined through a insertion-only data stream requires $\Omega(n)$ space.

3. (10 points) Consider an insertion-only data stream consisting of edges of a graph G with n vertices. Show that any streaming algorithm that with probability at least $\frac{2}{3}$, detects whether a graph contains a triangle requires $\Omega(n^2)$ space.