Homework 2
Due Fri Oct 30, 4:30pm

Prof. Amit Chakrabarti
Computer Science Department
Dartmouth College

General Instructions: Please write concisely, but rigorously, and show your calculations explicitly, as we do in class. Each problem is worth 5 points, and only "nearly flawless" solutions will earn full credit.

Honor Principle: You are allowed to discuss the problems and exchange solution ideas with your classmates. But when you write up any solutions for submission, you must work alone. You may refer to any textbook you like, including online ones. However, you may not refer to published or online solutions to the specific problems on the homework, if you intend to turn it in for credit. *If in doubt, ask the professor for clarification!*

---

1. The first AMS algorithm for estimating the frequency moments F_k (not the tug-of-war sketch) was described in class for vanilla streams only. Explain how it can be generalized to the turnstile model with positive updates; i.e., each token is of the form (j, c) — to be interpreted as the instruction "$f_j \leftarrow f_j + c$" — with c being a positive integer. Give careful pseudocode. You don't need to prove the correctness of your algorithm from scratch; instead, rely on the correctness proof we gave in class.

2. We have studied two algorithms for estimating the number of distinct elements in a stream, but neither of them produces a "sketch," as we have defined the term. Fix this.

   In greater detail: Consider a stream σ, in the turnstile model, that implicitly defines a frequency vector $\mathbf{f} = (f_1, \ldots, f_n)$. Suppose that, at all times during the processing of the stream and for all $j \in [n]$, we have $|f_j| \leq m$. Design an algorithm that computes a sketch of σ so that, based on the sketch, one can quickly estimate the quantity $L_0(\sigma) = |\{j : f_j \neq 0\}|$. Your algorithm should use space $\mathrm{poly}(\log m, \log n, 1/\varepsilon)$, work *without* the assumption that $\mathbf{f} \geq 0$, and return an estimate that is off by no more than $(1 \pm \varepsilon)$, with probability at least 99% (say).

   Hint: Do *not* try to generalize either of the DISTINCT-ELEMENTS algorithms. Instead, observe that one of the sketches we have seen in class goes a long way towards solving this.

3. We are reading two streams σ and σ' (both in the strict turnstile model). The streams are not synchronized in any way and may, in fact, be of different lengths. However, both streams are over the same universe $[n]$. Let $\mathbf{f}$ and $\mathbf{f}'$ be the frequency vectors defined by these streams, and let $\mathbf{p} = \mathbf{f}/\|\mathbf{f}\|_1$ and $\mathbf{p}' = \mathbf{f}'/\|\mathbf{f}'\|_1$ be the corresponding empirical probability distributions. Let m be an upper bound on both $|f_j|$ and $|f'_j|$ at all times and over all $j \in [n]$.

   Give an algorithm for the VARIATION-DISTANCE problem, which is to estimate the *total variation distance*

   $$\mathrm{V}(\mathbf{p}, \mathbf{q}) \;=\; \frac{1}{2}\|\mathbf{p} - \mathbf{q}\|_1 \;=\; \frac{1}{2}\sum_{j=1}^{n} |p_j - q_j|$$

   between $\mathbf{p}$ and $\mathbf{q}$. Your algorithm should use space $\mathrm{poly}(\log m, \log n, 1/\varepsilon)$ and return an estimate that is off by no more than $(1 \pm \varepsilon)$, with probability at least 99%.

4. Consider a stream σ in the turnstile model, defining a frequency vector $\mathbf{f}$. The count-min sketch solves the problem of estimating f_j, given j, but does not directly give us a *quick* way to identify, e.g., the set of elements with frequency greater than some threshold. Fix this.

   In greater detail: Let α be a constant with $0 < \alpha < 1$. We would like to maintain a suitable summary of the stream (some enhanced version of the count-min sketch, say) so that we can, on demand, quickly produce a set $S \subseteq [n]$ satisfying the following properties w.h.p.: (1) S contains every j such that $f_j \geq \alpha F_1$; (2) S does not contain any j such that $f_j < (\alpha - \varepsilon)F_1$. Here, $F_1 = F_1(\sigma) = \|\mathbf{f}\|_1$. Design a data stream algorithm that achieves this. Your space usage, as well as the time taken to process each token and to produce the set S, should be polynomial in the usual parameters, $\log m$, $\log n$, and $1/\varepsilon$, and may depend arbitrarily on α.