Problem Set 2: COSC-548 (Streaming Algorithms)

Due October 25th, 2016 by end of class

1. Recall that a hash family $\mathcal{H}$ of hash functions mapping $[n] \to [r]$ is k-wise independent if for every k distinct values $x_1, \ldots, x_k \in [n]$, and any $y_1, \ldots, y_k \in [r]$,

$$\Pr_{h \leftarrow \mathcal{H}}[h(x_1) = y_1 \text{ and } h(x_2) = y_2 \text{ and } \ldots \text{ and } h(x_k) = y_k] = 1/r^k.$$

   In class, we saw that for any prime p, the following hash family of hash functions mapping $[p] \to [p]$ is pairwise independent:

$$\mathcal{H} := \{h_{a,b} \colon a, b \in [p]\},$$

   where

$$h_{a,b}(x) := (ax + b) \bmod p.$$

   Any hash function in this family can be represented with $O(\log p)$ bits and can be evaluated at any input with one modular multiplication and one modular addition.

   Generalize the above construction as follows. For any $k \geq 2$, identify a k-wise independent hash family $\mathcal{H}$ of hash functions mapping $[p] \to [p]$ such that any $h \in \mathcal{H}$ can be represented with $O(k \log p)$ bits, and evaluated at any input with $O(k)$ modular multiplications and additions.

   You may use the fact that for any k distinct values $x_1, \ldots, x_k \in [p]$, and any values $y_1 \ldots, y_k \in [p]$ there exists a unique polynomial g of degree at most $k-1$ such that $g(x_i) = y_i$ for all i.

2. Suppose we toss n balls at random into m bins.

   2a) What is the probability that the first and second ball land in the same bin?

   2b) Prove that there is some sufficiently small constant $c > 0$ such that if $n < c \cdot m^{1/2}$, the probability that *no* two balls land in the same bin is at least $99/100$.

   2c) Prove that there is some sufficiently large constant $c > 0$ such that if $n > c \cdot m^{1/2}$, the probability that there exists *some* pair of balls that land in the same bin is at least $99/100$.

   Hint: Let E be the event that there is no pair of distinct balls landing in the same bin. The question asks you to prove that $\Pr[E] \leq 0.01$. For E to occur, the first $n/2$ balls must land in distinct bins. Conditioned on this occurring, give an expression for the probability that none of the last $n/2$ balls land in the same bin as any of the first $n/2$ balls. Then prove that this expression is at most 0.01 (for an appropriate choice of constant c in the statement of the problem).

   For the last step, you may find it helpful to use the fact that for any $x \geq 1$, the following holds: $(1 - 1/x)^x \leq 1/e < 1/2$.

3. (More Practice with Chernoff Bounds) Let $\phi, \varepsilon \in (0,1)$, and suppose $\phi \geq \varepsilon$. Given a data stream $\sigma = \langle a_1, \ldots, a_m \rangle$ in the (unit-weight update) insert-only model, with frequency vector $f = (f_1, \ldots, f_n)$, suppose we wish to output a list of items such that:

   - For every item i such that $f_i \geq \phi \cdot m$, i appears in the list.
   - Every item j that appears in the list satisfies $f_j \geq (\phi - \varepsilon) \cdot m$.

   3a) Prove that one can achieve the above using $O(\varepsilon^{-1} \cdot (\log n + \log m))$ bits of space by running Misra-Gries on σ and outputting any item that is assigned a counter with value larger than $(\phi - \varepsilon) \cdot m$.

   3b) Let c be a sufficiently large constant. Suppose we sample $\ell = c \cdot \varepsilon^{-2} \cdot \log n$ stream updates at random. Show that there is some threshold t such that if we output all items that appear in the sample t or more times, then with probability at least 99% over the random sample, the list output by the algorithm satisfies the desired two properties.

   You may use the following "additive Chernoff Bound": Let $X_1, \ldots, X_m$ be independent Poisson trials with expectation p (i.e., X_i takes value 1 with probability p, and 0 with probability $1 - p$). Let $X = \sum_{i=1}^{m} X_i$ and $\mu = \mathbb{E}[X] = m \cdot p$. Then the following holds:

$$\text{For } 0 < \lambda \leq 1, \Pr(|X - \mu| > \lambda \cdot m) \leq 2 \cdot e^{-2\lambda^2 \cdot m}.$$

   Remark. The space usage of this sampling algorithm is worse than Misra-Gries by a factor of $\Theta\left(\varepsilon^{-1} \cdot \log n\right)$.

   3c) Suppose our goal was instead to output a list of items such that

   - For every item i such that $f_i \geq \varepsilon \cdot m$, i appears in the list.
   - Every item j that appears in the list satisfies $f_j \geq \frac{1}{2} \cdot \varepsilon \cdot m$.

   Let c be a sufficiently large constant. Suppose we sample $\ell = c \cdot \varepsilon^{-1} \cdot \log n$ stream updates at random. Show that there is some threshold t such that if we output all items that appear in the sample t or more times, then with probability at least 99% over the random sample, the list output by the algorithm satisfies the desired two properties.

   You may use the "multiplicative Chernoff Bound" stated in the previous problem set, and repeated here for your convenience. Let $X_1, \ldots, X_m$ be independent Poisson trials with expectation p (i.e., X_i takes value 1 with probability p, and 0 with probability $1 - p$). Let $X = \sum_{i=1}^{m} X_i$ and $\mu = \mathbb{E}[X] = m \cdot p$. Then the following holds:

$$\text{For } 0 < \delta \leq 1, \Pr(|X - \mu| > \delta\mu) \leq 2 \cdot e^{-\mu\delta^2/3}.$$

4. Recall that the Count-Min Sketch for the strict turnstile update model works as follows.

- Let $k = \lceil 2/\varepsilon \rceil$. Consider $t = \log(1/\delta)$ arrays, $C_1, \ldots, C_t$, each containing k counters initialized to 0.
- Choose t hash functions $h_1, \ldots, h_t \colon [n] \to [k]$ independently and uniformly at random from a pairwise independent hash family.
- When processing stream update (a_i, δ_i):
  For each $\ell = 1, \ldots, t$:
  $C_\ell[h_\ell(a_i)] \leftarrow C_\ell[h_\ell(a_i)] + \delta_i$.
- When asked to return an estimate for the frequency of item j, return $\min_{\ell=1,\ldots,t} C_\ell[h_\ell(j)]$.

4a) Suppose we replace min in the final line of the algorithm with median. Show that the returned estimate for any query j is never more accurate with median than with min.

4b) Nevertheless, let us show that the resulting *Count-Median* algorithm satisfies error guarantees that are similar to Count-Min. Specifically, suppose we increase k and t by constant factors, say to $k = 6/\varepsilon$ and $t = 10 \cdot \log(1/\delta)$. Show that for each fixed $i \in [n]$, with probability at least $1 - \delta$ the *Count-Median* estimate $\hat{f}_i$ satisfies $|\hat{f}_i - f_i| \leq \varepsilon \cdot M$, where $M = \sum_{i=1}^{n} f_i$.