University of Edinburgh
INFR11156: Algorithmic Foundations of Data Science (2019)
Solution 6

Problem 1: We discussed in class an efficient construction of a family of k-wise independent hash functions h such that it holds for any x that $\mathbf{E}[h^i(x)] = 1$ if $i \geq 1$ is an even number, and $\mathbf{E}[h^i(x)] = 0$ otherwise. In this question, you need to construct a family of k-wise independent hash functions g such that it holds for any x that $\mathbf{E}[h^i(x)] = 1$ if i is divisible by 3, and 0 otherwise.

***Solution*:** Let ω be a primitive third root of unity, and we define $\mathcal{R} = \{\omega, \omega^2, \omega^3\}$. We construct a family of k-wise independent hash functions in the usual way, but set the range of each hash function h to be $\mathcal{R}$, i.e., for every x, $h(x)$ is uniformed distributed in $\mathcal{R}$. Therefore, we have that

$$\begin{aligned}
\mathbf{E}[h(i)] &= \frac{1}{3} \cdot \left(\omega + \omega^2 + \omega^3\right) = \frac{\omega}{3} \cdot \left(1 + \omega + \omega^2\right) = \omega \cdot \frac{\omega^3 - 1}{\omega - 1} = 0,\\
\mathbf{E}[h^2(i)] &= \frac{1}{3} \cdot \left(\omega^2 + \omega^4 + \omega^6\right) = \frac{\omega^2}{3} \cdot \left(1 + \omega^2 + \omega^4\right) = \omega^2 \cdot \frac{\omega^6 - 1}{\omega^2 - 1} = 0,\\
\mathbf{E}[h^3(i)] &= \frac{1}{3} \cdot \left(\omega^3 + \omega^6 + \omega^9\right) = \frac{1}{3} \cdot (1 + 1 + 1) = 1.
\end{aligned}$$

Generalising this analysis, it is easy to see that $\mathbf{E}[h^i(x)] = 1$ if i is divisible by 3, and 0 otherwise.

Problem 2: For any undirected graph $G = (V, E)$ with n vertices, we say three vertices u, v, w form a triangle if there are three edges connecting u, v, w respectively. This problem is to analyse a streaming algorithm for approximately computing the number of triangles in an undirected graph. To describe the proposed algorithm, let $\mathcal{H}$ be a family of 12-wise independent hash functions, where every $h \in \mathcal{H}$ is of the form $h : V \to \{-1, 1\}$. Let Z be our estimator, which is set to be 0 initially. The algorithm is described in Algorithm 1 below. Prove that the returned value $Z^3/6$ is an unbiased estimator of the number of triangles in G, i.e.,

$$\mathbf{E}\left(\frac{Z^3}{6}\right) = \text{the number of triangles in } G.$$

Algorithm 1 Approximate number of triangles

```
Pick a function $h$ uniformly at random from $\mathcal{H}$;
$Z \leftarrow 0$;
while an edge $\{u, v\}$ arrives do
    $Z \leftarrow Z + h(u) \cdot h(v)$;
end while
Return $Z^3/6$.
```

***Solution*:** We have that

$$\begin{aligned}
\mathbf{E}\left[Z^3\right] &= \mathbf{E}\left[\left(\sum_{e=\{u,v\}} h(u)h(v)\right)^3\right]\\
&= \mathbf{E}\left[\sum_{e_1=\{u_1,v_1\}} \sum_{e_2=\{u_2,v_2\}} \sum_{e_3=\{u_3,v_3\}} \prod_{i=1}^{3} h(u_i)h(v_i)\right]\\
&= \sum_{e_1=\{u_1,v_1\}} \sum_{e_2=\{u_2,v_2\}} \sum_{e_3=\{u_3,v_3\}} \mathbf{E}\left[\prod_{i=1}^{3} h(u_i)h(v_i)\right],
\end{aligned}$$

where the last equality comes from the linearity of the expectation. We will now argue that the last formulation is exactly 6 times the number of triangles in G. Under expectation, only the terms with products of even powers of $h(u_i)$ and $h(v_i)$ survive. In a combination of three edges $e_i = \{u_i, v_i\}$, every vertex u_i or v_i is connected to at most three other vertices. Moreover, the power of each $h(x_i)$ is the number of times x_i appears in the combination. We see that only the terms where each vertex appears exactly twice survive, which can only happen if the three edges form a cycle. Since every triangle is counted 6 times (once for every permutation of its edges) wee see that $\mathbf{E}\left[\, Z^3 \,\right]$ equals to 6 times the number of triangles in G.

Problem 3: We are given two independent streams of elements from $\{1, \ldots, n\}$, and we only consider the cash register model. Let $A[1, \ldots, n]$ and $B[1, \ldots, n]$ be the number of occurrences of item i in two streams, respectively. Design a streaming algorithm to estimate $X = \sum_{i=1}^{n} A[i]B[i]$ with additive error $\varepsilon \cdot \|A\|_1 \cdot \|B\|_1$. You need to analyse the space complexity of your proposed algorithm, and analyse the correctness of your algorithm.

Solution: The algorithm follows the framework of the Count-Min sketch. We will make use of two tables C and D, each of size $d \times w$, where $d = \lceil \log(1/\delta) \rceil$ and $w = \mathrm{e}/\varepsilon$. The i-th row of each table corresponds to a hash function $h_i : [n] \to [w]$ chosen from a family of unievrsal hash functions. The two tables support two operations *Insert(x)* and *Query* as follows:

Algorithm 2 *Insert(x)*

```
Result: Inserts a new element $x$ from the stream
for $i = 1, d$ do
    Compute $h_i(x)$
    if $x$ is from the first stream then
        $C[i, h_i(x)] \leftarrow C[i, h_i(x)] + 1$
    else
        $D[i, h_i(x)] \leftarrow D[i, h_i(x)] + 1$
    end if
end for
```

Algorithm 3 *Querry*

```
Result: Provides the answer to the querry $X = \sum_{i=1}^{n} A[i]B[i]$
Return $X' := \min_{1 \le i \le d} C[i]D[i]$, where $C[i]D[i] = \sum_{j=1}^{w} C[i,j]D[i,j]$
```

By construction, for any $x \in [n]$ and any row i, x will be mapped to the same column $h_i(x)$ in the two tables. Thus, when computing the dot product $C[i]D[i]$, we are guaranteed to have the sum $A[x]B[x]$. By taking the minimum over all i's and taking into account the values in the two vectors are nonnegative, it follows that $X' \geq X$.

For the other direction, we will prove that with constant probability $1 - \delta$ we have that $X' \leq X + \varepsilon \|A\|_1 \|B\|_1$. Fix a row i and suppose $C[i]D[i] = \sum_{i=1}^{n} A[i]B[i] + Z_i$, where Z_i is the excess obtained from the dot product. Such an excess can occur if and only if we encounter collisions of the hash function. Namely, whenever two distinct $x, y \in [n]$ are such that $h_i(x) = h_i(y) = z$, computing $C[i,z]D[i,z]$ yields an excess of $A[x]B[y] + A[y]B[x]$. Hence, we conclude that

$$Z_i = \sum_{\substack{x \neq y \\ h_i(x) = h_i(y)}} A[x]B[y].$$

Since we used universal hash functions, it follows that $\forall x \neq y$,

$$\mathbf{P}[h_i(x) = h_i(y)] \leq \frac{1}{w} = \frac{\varepsilon}{\mathrm{e}}.$$

This, in turn, implies that

$$\mathbf{E}\left[\, Z_i \,\right] = \sum_{x \neq y} \mathbf{P}[h_i(x) = h_i(y)] A[x] B[y] \leq \frac{\varepsilon}{\mathrm{e}} \left\| A \right\|_1 \left\| B \right\|_1 .$$

To complete the proof, observe that

$$\mathbf{P}\left[\, X' > X + \varepsilon \left\| A \right\|_1 \left\| B \right\|_1 \right] = \mathbf{P}[\forall i : Z_i > \varepsilon \left\| A \right\|_1 \left\| B \right\|_1] \leq \mathbf{P}\left[\forall i : Z_i > \mathrm{e}\mathbf{E}\left[\, Z_i \,\right]\right] \leq \mathrm{e}^{-d} \leq \delta,$$

where the last inequality is obtained by applying Markov's inequality. The space used by the algorithm is essentially dominated by the two tables used to store the number of appearances of the elements in the two streams, which is $O(wd) = O\left(\frac{1}{\varepsilon}\log(1/\delta)\right)$.