Exercise Set VII, Advanced Algorithms 2022

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

These problems are taken from various sources at EPFL and on the Internet, too numerous to cite individually.

1 Design a one-pass (streaming) algorithm that, for a stream that possesses a majority element (appearing more than $m/2$ times), terminates with this element. Prove the correctness of your algorithm.

2 *(MinHashing)* Suppose we have a universe U of elements. For $A, B \subseteq U$, the Jaccard distance of A, B is defined as

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}.$$

This definition is used in practice to calculate a notion of similarity of documents, webpages, etc. For example, suppose U is the set of English words, and any set A represents a document considered as a bag of words. Note that for any two $A, B \subseteq U$, $0 \leq J(A, B) \leq 1$. If $J(A, B)$ is close to 1, then we can say $A \approx B$.

Let $h : U \to [0, 1]$ where for each $i \in U$, $h(i)$ is chosen uniformly and independently at random. For a set $S \subseteq U$, let $h_S := \min_{i \in S} h(i)$. **Show that**

$$\Pr[h_A = h_B] = J(A, B).$$

Now, if we have sets $A_1, A_2, \ldots, A_n$, we can use the above idea to figure out which pair of sets are "close" in time essentially $O(n|U|)$. We can also obtain a good approximation of $J(A, B)$ with high probability by using several independently chosen hash functions. Note that the naive algorithm would take $O(n^2|U|)$ to calculate all pairwise similarities.

3 In this problem we are going to formally analyze the important median trick. Suppose that we have a streaming algorithm for distinct elements that outputs an estimate $\hat{d}$ of the number d of distinct elements such that

$$\Pr[\hat{d} > 3d] \leq 47\% \qquad \text{and} \qquad \Pr[\hat{d} < d/3] \leq 47\%\,,$$

where the probabilities are over the randomness of the streaming algorithm (the selection of hash functions). In other words, our algorithm overestimates the true value by a factor of 3 with a quite large probability 47% (and also underestimates with large probability). We want to do better!

An important and useful technique for doing better is the median trick: run t independent copies in parallel and output the median of the t estimates (it is important that it is the median and *not* the mean as a single horrible estimate can badly affect the mean). Prove that if we select $t = C\ln(1/\delta)$ for some large (but reasonable) constant C, then the estimate $\hat{d}$ given by the median trick satisfies

$$d/3 \leq \hat{d} \leq 3d \qquad \text{with probability at least } 1-\delta.$$

Hint: an important tool in this exercise are the Chernoff Bounds, which basically say that sums of independent variables are highly concentrated. Two such bounds can be stated as follows. Suppose $X_1, X_2, \ldots, X_n$ are independent random variables taking values in $\{0,1\}$. Let X denote their sum and let $\mu = \mathbb{E}[X]$ denote the sum's expected value. Then for any $\delta \in (0,1)$,

$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\frac{\delta^2\mu}{2}} \qquad \text{and} \qquad \Pr[X \geq (1+\delta)\mu] \leq e^{-\frac{\delta^2\mu}{3}}\,.$$

4 *(*, Pairwise independent random variables)* Let $y_1, y_2, \ldots, y_n$ be uniform random bits. For each non-empty subset $S \subseteq \{1,2,\ldots,n\}$, define $X_S = \oplus_{i\in S}\, y_i$. Show that the bits $\{X_S : \emptyset \neq S \subseteq \{1,2,\ldots,n\}\}$ are pairwise independent.

This shows how to stretch n truly random bits to $2^n - 1$ pairwise independent bits.

Hint: Observe that it is sufficient to prove $\mathbb{E}[X_S] = 1/2$ *and* $\mathbb{E}[X_S X_T] = 1/4$ *to show that they are pairwise independent. Also use the identity* $\oplus_{i\in A}\, y_i = \frac{1}{2}\left(1 - \prod_{i\in A}(-1)^{y_i}\right)$.