CSE 521: Design and Analysis of Algorithms **Fall 2023**

Problem Set 3

Deadline: Oct 24th (at 11:59PM) in gradescope

The goal of this problem set is to learn the idea of *minhash*. Minhash is a hash function which is commonly used in practice to estimate the *Jaccard similarity* of two sets.

1) 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 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

$$\mathbb{P}\left[h_A = h_B\right] = J(A, B).$$

2) **Optional 0 points:** Let $X_1, \ldots, X_n$ be independent random variables uniformly distributed in $[0, 1]$ and let $Y = \min\{X_1, \ldots, X_n\}$. Show that $\mathbb{E}\left[Y\right] = \frac{1}{n+1}$ and $\mathrm{Var}(Y) \leq \frac{1}{(n+1)^2}$.

3) Consider the following algorithm for estimating F_0, the number of unique elements in a sequence $x_1, \ldots, x_m$ in the set $\{0, 1, \ldots, n-1\}$. Let $h : \{0, 1, \ldots, n-1\} \to [0, 1]$ s.t., $h(i)$ is chosen uniformly and independently at random in $[0, 1]$ for each i. We start with $Y = 1$. After reading each element x_i in the sequence we let $Y = \min\{Y, h(x_i)\}$.

   a) Show that by the end of the stream $\frac{1}{\mathbb{E}[Y]} - 1$ is equal to F_0.

   b) Use the above idea to design a streaming algorithm to estimate the number of distinct elements in the sequence with multiplicative error $1 \pm \epsilon$. For the analysis you can assume that you have access to k independent hash functions as described above. Show that $k \leq O(1/\epsilon^2)$ many such hash functions is enough to estimate the number of distinct elements within $1 + \epsilon$ factor with probability at least 9/10.