Jan Sokol - kgj360 || Denis Trebula

1. Hash functions for sampling

(a) Prove that $p \leq Pr[hm(x)/m < p] \leq 1.01p$. For this we will use few things.

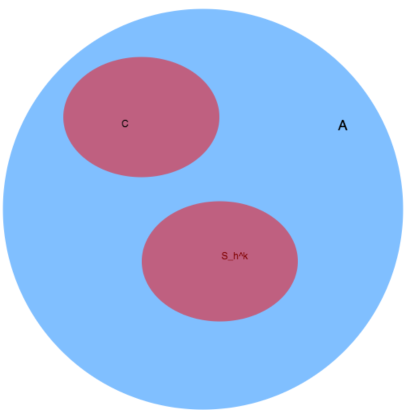
$h(x) = h_m(x)/m$ is Strong Independent Hash Function;
$p \geq 100/m \Rightarrow p/100 \geq 1/m$;
$hm(x)/m \leq p \Rightarrow hm(x) \leq mp$;

(b)

2. Bottom-k sampling

Prove that $E[|C \cap S_h^k(A)|/k] = |C|/|A|$, assuming that $S_h^k(A)$is a uniformly random size-k $\subset A$.

We assume that $C \subset A$ and $S_h^k(A) \subset A$ are independent, as shown on Venn diagram bellow:



Seeing this we can say:

$Pr[x \in S_h^k(A)] = \frac{k}{|A|}$
$Pr[x \in C] = \frac{C}{|A|}$

$E[|C \cap S_h^k(A)|/k]$
$= \frac{1}{k}E[|C \cap S_h^k(A)|]$
$= \frac{1}{k}\sum_{\forall a \in A} E[a \in C \wedge a \in S_h^k(A)]$
$= \frac{1}{k}\sum_{\forall a \in A} Pr[a \in C \wedge a \in S_h^k(A)]$
$= \frac{1}{k}\sum_{\forall a \in A} Pr[a \in C] \cdot Pr[a \in S_h^k(A)]$ (From independence)
$= \frac{1}{k}\sum_{\forall a \in A} (\frac{C}{|A|} \cdot \frac{k}{|A|})]$ (Substitution from above)
$= \frac{1}{k}\sum_{\forall a \in A} (\frac{C}{|A|} \cdot \frac{k}{|A|})$ (Substitution from above)
$= \frac{1}{k}\frac{C}{|A|} \cdot \frac{k}{|A|}\sum_{\forall a \in A} 1$ (Elements in sum are not dependant on $a \in A$)
$= \frac{1}{k}\frac{C}{|A|} \cdot \frac{k}{|A|}|A|$
$= \frac{C}{|A|}$

2.1 Frequency estimation

Exercise 2

(a) As a data structure we would use binary heap (max heap). Binary heaps are a one of the ways of implementing priority queues. In the heap we would only store k smallest keys.

(b) To process another key from the stream it would take $O(log_2 k)$.

2.2 Similarity estimation

(a)

(b)

(c)

3 Bottom-k sampling with strong universality

3.1 A union bound

exercise 5

3.2 Upper bound with 2-independence

exercise 6

exercise 7