Contents

1 Hash functions for sampling

1.1 Exercise 1

1.1.1 (a)

We are asked to prove $p \leq Pr[h_m(x)/m < p] \leq 1.01p$. We will use various facts to show this. Firstly that $h(x) = h_m(x)/m$ is a Strong Universal Hash Function, secondly as we are give $p \geq 100/m$ this implies that $p/100 \geq 1/m$, thirdly $h_m(x)/m \leq p$ implies $h_m(x) \leq mp$ and finally we will use that for any y, $y \leq \lceil y \rceil < y + 1$.

We observe that

$$\begin{aligned}
&Pr[h_m(x)/m < p] \\
&= \sum_{0 \leq k < mp} Pr[h_m(x) = k] \\
&= \sum_{0 \leq k < mp} \frac{1}{m} \\
&= \frac{1}{m} \, |[0, mp)| \\
&= \frac{1}{m} \cdot \lceil mp \rceil \\
&= \frac{\lceil mp \rceil}{m}
\end{aligned}$$

Thus we conclude

$$p = \frac{pm}{m} \leq Pr[h_m(x)/m < p] = \frac{\lceil pm \rceil}{m} \leq \frac{pm + 1}{m} \leq p + \frac{p}{100} = 1.01p$$

1.1.2 (b)

We are asked to bound the probability that two keys share the same hash value $\dfrac{h_m(x)}{m} = \dfrac{h_m(y)}{m}$, given that $A \subset U, m \geq 100|A|^2$.

To prove this we use that $\dfrac{\binom{n}{2}}{2} = \dfrac{\frac{n(n-1)}{2}}{m} = \dfrac{n(n-1)}{2m}$ The probability can be written as

$$\begin{aligned}
Pr[\exists\{x,y\} \in A : \frac{h_m(x)}{m} = \frac{h_m(x)}{m}] \\
&\leq \sum_{\{x,y\}\in A} Pr\left[\frac{h_m(x)}{m} = \frac{h_m(x)}{m}\right] \\
&= \frac{\binom{|A|}{2}}{m} \\
&\leq \frac{|A|(|A|-1)}{2m} \\
&\leq \frac{|A|(|A|-1)}{2 \cdot 100|A|^2} \\
&\leq \frac{|A|(|A|-1)}{200|A|^2}
\end{aligned}$$

Thus the bound for two keys sharing the same hash value is

$$\leq \frac{1}{200}$$

2 Bottom-*k* sampling

2.1 Frequency Estimation

2.1.1 Exercise 2

We are asked to show that $E\left[|C \cap S_h^k(A)|/k\right] = |C|/|A|$.

We are told that $S_h^k(A)$ is a uniformly random subset of A and C is a subset of A which is independent from $S_h^k(A)$, knowing these we can say

$$Pr\left[x \in S_h^k(A)\right] = p = \frac{k}{|A|} \tag{1}$$

$$Pr\left[x \in C\right] = Pr(C) = \frac{|C|}{|A|} \tag{2}$$

$$\begin{aligned}
&E\left[|C \cap S_h^k(A)|/k\right] \\
&= \frac{1}{k} \cdot E\left[|C \cap S_h^k(A)|\right] \\
&= \frac{1}{k} \sum_{a \in A} E\left[a \in C \wedge a \in S_h^k(A)\right] \\
&= \frac{1}{k} \sum_{a \in A} Pr\left[a \in C \wedge a \in S_h^k(A)\right] \\
&= \frac{1}{k} \sum_{a \in A} \left(Pr\left[a \in C\right] \cdot Pr\left[a \in S_h^k(A)\right]\right) && \text{Independence} \\
&= \frac{1}{k} \cdot \sum_{a \in A} \left(\frac{|C|}{|A|} \cdot \frac{k}{|A|}\right) && \text{From 1 and 2} \\
&= \frac{1}{k} \cdot \frac{|C|}{|A|} \cdot \frac{k}{|A|} \cdot \sum_{a \in A} 1 && \text{Elements in summation independent of } a \in A \\
&= \frac{1}{k} \cdot \frac{|C|}{|A|} \cdot k \\
&= \frac{|C|}{|A|}
\end{aligned}$$

2.1.2 Exercise 3 (a)

For this we would implement a maximum heap structure, where we only store the k smallest keys. This would allow for insertion time $O(\log_2 k)$ as checking if the key is smaller than the largest key, would only take $O(1)$.

2.1.3 Exercise 3 (b)

As mentioned above, the insertion would take $O(\log_2 k)$.

2.2 Similarity Estimation

2.2.1 Exercise 4 (a)

$$S_h^k(A \cup B) = S_h^k\big(S_h^k(A) \cup S_h^k(B)\big)$$

We will use the definition of S_h^k to prove this equality analytically. $S_h^k(A) = \{$the k keys $x \in A$ with the smallest hash values$\}$. Similarly for $S_h^k(B)$. The LHS of the expression is the smallest k keys in all keys in A and B together. The RHS of the expression is the smallest k keys in the union of the smallest k keys in each of A and B. Imagine some possible scenarios:

1. All the smallest keys came from only one set, say A, then both sides of the equations will return just these k smallest keys.
2. The k smallest keys in all of A and B come from both sets, the on the LHS we are are finding all the smallest keys from both these sets conjoined, and on the RHS we are first combining the smallest keys from both sets, resulting in $2k$ keys and then picking the k smallest of these.

As these operations are taking only the smallest k keys from the sets we have equality.

2.3 Exercise 4 (b)

$$A \cap B \cap S_h^k(A \cup B) = S_h^k(A) \cap S_h^k(B) \cap S_h^k(A \cup B)$$

Again, we will prove this equality analytically. Firstly the LHS of the expression. An element can only satisfy this expression if there is some element in the set of smallest keys in of the union of A and B which belongs to A, also belongs to B. An element can only satisfy this expression if there is some element in the set of smallest keys in of the union of A and B which belongs to the set of smallest keys in A, also belongs to the set of smallest keys in B. Both the LHS and RHS are limited by the elements in $S_h^k(A \cup B)$, therefore we have equality.

2.3.1 Exercise 4 (c)

The running time for $\frac{|S_h^k(A) \cap S_h^k(B) \cap S_h^k(S_h^k(A) \cup S_h^k(B))|}{k}$ when using a max-heap can be split into the following running times:

- Union: $S_h^k(A), S_h^k$run in $O(k \log_2 n)$ and the union in $\Theta(2k)$, thus: $O(k \log_2 n) + \Theta(k)$
- Intersections: Given ordered heaps, intersect only requires linear time $O(k)$ and extractions can be performed in $O(k \log_2 n)$ giving: $O(k \log_2 n) + O(k)$ for each interesection. Since the intersect with $S_h^k(S_h^k(A) \cup S_h^k(B))$ takes linear time on the largest heap and both have maximum size k we find that the time complexity is: $O(k \log_2 n) + O(k) + O(k)$
- $S_h^k(union)$: As we have $2k$ elements we can extract in $O(k \log_2 2k)$ time.

- Finally we need to account for the fact that the number of entries can be accounted in $O(1)$ time.

$$\begin{aligned} Union :& O(k\log_2 n) + \Theta(k). \\ Intersections :& O(k\log_2 n) + O(k) + O(k) \\ S_h^k(union) :& O(k\log_2 2k) \\ \text{Yielding} :& O(1) + O(k\log_2 n)) + O(k) + O(k) + O(k\log_2 2k) + O(k\log_2 k) + \Theta(k) = O(k\log_2 n) \end{aligned}$$

3 Bottom-*k* sampling with strong universality

3.1 A union bound

3.1.1 Exercise 5

3.2 Upper bound with 2-independence

3.2.1 Exercise 6

3.2.2 Exercise 7

References