Weekplan: Streaming I

Philip Bille Inge Li Gørtz Eva Rotenberg

References and Reading

[1] Amit Chakrabarti: *Data Stream Algorithms* 2011 (updated July 2020) chapter 0 except 0.3 and chapter 1.

[2] R. Morris: Counting Large Numbers of Events in Small Registers.

We recommend reading the specified chapters and sections of [1] and [2] in detail.

Probability theory cheat-sheet

Variance: Recall, the variance is:

$$\mathrm{Var}[X] = \mathbb{E}[(X - \mathbb{E}(X))^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$$

Assume X_i are *uncorrelated*, then:

$$\mathrm{Var}\left[\sum_i X_i\right] = \sum_i \mathrm{Var}[X_i]$$

Markov's inequality: For Y being a positive-valued random variable,

$$P[Y \geq t] \leq \frac{\mathbb{E}[Y]}{t}$$

Chebyshev's inequality: For a random variable X with mean $\mu_X = \mathbb{E}(X)$ and standard deviation $\sigma_X = \sqrt{\mathrm{Var}[X]}$,

$$P\Big[|X - \mu_X| \geq t\sigma_X\Big] \leq \frac{1}{t^2}$$

Chernoff bound: $X_1, \ldots, X_n$ independent random $\in \{0, 1\}$ with $P[X_i = 1] = p$ and $X = \sum_i X_i$:

$$P\left[X > (1+\delta)\mathbb{E}[X]\right] < \left[\frac{e^\delta}{(1+\delta)^{1+\delta}}\right]$$

Exercises

The following exercise relates to the streaming model. Remember that we use the number of bits when we calculate space in the streaming model.

1 Missing numbers

1.1 Assume you get $n-1$ different integers from the set $\{1, \ldots, n\}$ in a stream. Can you deduce the missing number using only $O(\log n)$ space?

1.2 Assume now you only get $n-2$ different integers from the set. Can you find the two missing numbers in $O(\log n)$ space?

2 Largest numbers Given n numbers, suppose we want to find the n/k largest.

2.1 In the RAM-model, how would you solve this task? What is your total running time?

2.2 In the streaming model, how little space is necessary to solve this task? What is your running time? Can you get a competitive running time?

3 Reservoir sampling[1] Reservoir sampling is a method for choosing an item item uniformly at random from an arbitrarily long stream of data; for example, the sequence of packets that pass through a router, or the sequence of IP addresses that access a given web page. Like all data stream algorithms, this algorithm must process each item in the stream quickly, using very little memory.

Algorithm 1: GETONESAMPLE(stream S)

$\ell \leftarrow 0$
while S *is not done* **do**
  $x \leftarrow$ next item in S
  $\ell \leftarrow \ell + 1$
  if RANDOM(ℓ) $= 1$ **then**
    sample $\leftarrow x$ $(\star)$
  return *sample*
end

Here RANDOM(a) is a random number generator that uniformly at random returns an integer between 1 and a (both included). At the end of the algorithm, the variable ℓ stores the length of the input stream S; this number is not known to the algorithm in advance. If S is empty, the output of the algorithm is (correctly!) undefined. In the following, consider an arbitrary non-empty input stream S, and let n denote the (unknown) length of S.

3.1 Prove that the item returned by GETONESAMPLE(S) is is chosen uniformly at random from S.

3.2 What is the *exact* expected number of times that GETONESAMPLE(S) executes line $(\star)$?

3.3 What is the *exact* expected value of ℓ when GETONESAMPLE(S) executes line $(\star)$ for the *last* time?

3.4 What is the *exact* expected value of ℓ when either GETONESAMPLE(S) executes line $(\star)$ for the *second* time or the algorithm ends (whichever happens first)?

3.5 Describe and analyze an algorithm that returns a subset of k distinct items chosen uniformly at random from a data stream of length at least k. The integer k is given as part of the input to your algorithm. Prove that your algorithm is correct.

For example, if $k = 2$ and the stream contains the sequence $\langle \spadesuit, \heartsuit, \diamondsuit, \clubsuit \rangle$, the algorithm should return the subset $\{\diamondsuit, \spadesuit\}$ with probability $1/6$.

The following exercises relate to chapter 1 in [1].

4 Frequency $[w]$ Consider the trivial solution to the frequency problem: Keeping as many counters as there are colours. What is the space-consumption?

5 Misra-Gries $[w]$ Run Misra-Gries' algorithm on the following stream with $k = 3$. What do you output? How large was your largest counter?
b a b b a m b a m b a n a n a n a n a

6 Tightness of Misra-Gries Given k and n, design a stream of length n that contains some character $n/(k+1)$ times yet this character is not output by Misra-Gries' algorithm.

7 Exercises from [1] Solve exercises 1-1 and 1-3 from [1].

[1] This exercise is from Jeff Erickson's notes on streaming