Asmt 4: Frequent Items

Turn in through Canvas by 2:45pm:
Wednesday, March 14
50 points

Overview

In this assignment you will explore finding frequent items in data sets, with emphasis on streaming techniques designed to work at enormous scale. For simplicity you will work on more manageably sized data sets, and simulate the stream by just processing with a for loop.

You will use two data sets for this assignment:

- `http://www.cs.utah.edu/~jeffp/teaching/cs5140/A4/S1.txt`
- `http://www.cs.utah.edu/~jeffp/teaching/cs5140/A4/S2.txt`

These data sets each describe a set of $m = 1{,}000{,}000$ characters. The order of the file represents the order of the stream.

As usual, it is highly recommended that you use LaTeX for this assignment. If you do not, you may lose points if your assignment is difficult to read or hard to follow. Find a sample form in this directory: *`http://www.cs.utah.edu/~jeffp/teaching/latex/`*

1 Streaming Algorithms

A (20 points): Run the Misra-Gries Algorithm (see **L11.3.1**) with $(k-1) = 9$ counters on streams S1 and S2. Report the output of the counters at the end of the stream.

In each stream, use just the counters to report how many objects *might* occur more than 20% of the time, and which must occur more than 20% of the time.

B (20 points): Build a Count-Min Sketch (see **L12.1.1**) with $k = 10$ counters using $t = 5$ hash functions. Run it on streams S1 and S2.

For both streams, report the estimated counts for objects a, b, and c. Just from the output of the sketch, which of these objects, with probably $1 - \delta = 31/32$ (that is assuming the randomness in the algorithm does not do something bad), *might* occur more than 20% of the time?

C (5 points): How would your implementation of these algorithms need to change (to answer the same questions) if each object of the stream was a "word" seen on Twitter, and the stream contained all tweets concatenated together?

D (5 points): Describe one advantage of the Count-Min Sketch over the Misra-Gries Algorithm.

2 BONUS

The exact heavy-hitter problem is as follows: return *all* objects that occur more than 10% of the time. It cannot return any false positives or any false negatives. In the streaming setting, this requires $\Omega(\min\{m, n\})$ space if there are n objects that can occur and the stream is of length m.

A: (1 point) A 2-Pass streaming algorithm is one that is able to read all of the data in-order exactly twice, but still only has limited memory. Describe a small space algorithm to solve the exact heavy hitter problem, i.e., with $\varepsilon = 0$, (say for $\phi = 10\%$ threshold).