CAS CS 365
Lec 16

1. Overview
    a. Streaming
        i. Big Data/ Massive data
    b. The simplest problems become challenging (Morris Algorithm for estimating the length of the stream)
        i. How to sample
        ii. Estimate F_0
        iii. Estimate F_1
        iv. Estimate frequencies
        v. Distinct elements
2. Count-Min Sketch for Heavy Hitters
    a. Let $[f_1, f_2, \ldots, f_n]$ be the vector of frequencies of the set of elements [n] after seeing a stream of length $m = f_1 + f_2 + \ldots + f_n$
    b. Given $0 < \varphi < 1$, a heavy hitter (φ - HH) is an element i such that $f_i \geq \phi m$
    c. The goal is to return all the approximate heavy hitters {i: such that $f_i \geq (\phi - \epsilon)m$ }
    d. An elegant solution: count min (CM) sketch
3. Data structure
    a. Struct CMSketch
        i. r::Int64
        ii. b::Int64
        iii. cm::Matrix{Unit64}//2d array with r rows and b buckets/columns

       End
    b. Each row j is associated with a hash function h_j and the functions are pairwise independent
    c. Basic API
        i. init(CMSketch)
            1. Set all counters in CMSketch.cm to 0
        ii. insert(CMSketch, i)
            1. Update the data structure when element i arrives in the stream
        iii. query(CMSketch, i)
            1. Obtain an estimate of the true frequency of element i
4. Insert(CM, i)
    a. To update the data structure, we hash element i r times with each h_j hash function for j = 1..r. Specifically, we update the data structure as follows:
        b. $cm[j, \mathbf{h_j}(i)] = cm[j, \mathbf{h_j}(i)]+1$ for j=1..r

c. Suppose 100 is the first element to arrive and $h_1(100) = 2$ and $h_2(100) = 4$

i.

	1	2	3	4
1		+1		
2				+1

5. Query(CM, i)
   a. To obtain an estimate of the frequency fi of element i we query the data structure. Specifically we hash element i with each h_j hash function for j = 1…r and we return the smallest count among the entries of the cells CM[j, h_j[i] over all j].
   b. For example, the estimated frequency of element 100 is min(CM[1, $h_1(100)$], CM[2, $h_2(100)$]) = 110
   c.

	1	2	3	4
1	50	110	20	5
2				130

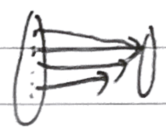
6. Theoretical guarantees
   a. Suppose m is the length of the stream. Let the number of buckets B (#cols) be equal to [e / epsilon] and the number of repetitions (rows) set to log(1/delta). Then the estimated frequency of the true frequency satisfies the following guarantees:
      i. True frequency <= estimated frequency
      ii. Estimated frequency <= true frequency + epsilon * m

Data Structure (CM sketch) associated with everything is hashed functions

$h_1 : [n] \to [b]$

→ assume every row is associated with perfect hash function

$f[n] \to [B]$



B^n

$f(1) \to B_0$

$f(2) \to B_1$ → need $\log_2(B)$ bits for each n

need $n\log_2(B)$ bits for the function, which is too big

hash functions

ex) $(ax+b) \bmod B$

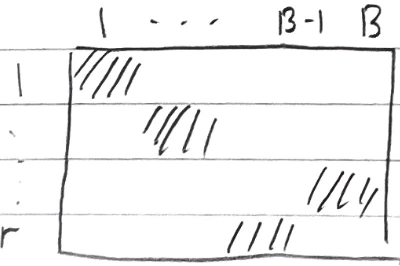
Choose only a and b → now need to store only a & b
(but the question on how to choose good a and b is questionable)

- Back to data structure

Update (CM, i)

$h_1(i), h_2(i), \ldots, h_i(i)$

↳ column where i hashes for the first 1st row



$CM[\text{row}, h_{row}(i)] += 1$

	1	2	3	4
1		+1		
2				+1

ex) $h_1(100) = 2$

$h_2(100) = 4$

$CM[row, h_{row}(i)] \geq f_i$

Query $\langle$ $V \leftarrow \infty$

for each row J:

compute $h_j(x)$

if $CM[J, h_j(x)] < V$

then $V \leftarrow CM[j, h_j(x)]$

return V

- Choose # rows and # columns
  - ↳ small rows:

- Suppose querying an element that never appeared in stream
  - ↳ one row & small # of buckets: if everything goes to a bucket, the frequency can be m, which is incorrect

- $f_x \leq \hat{f}_x$ with probability $\underline{1}$

$\hat{f}_x = \min(\hat{f}_{x,1}, \dots \hat{f}_{x,r})$ where $\hat{f}_{x,j} = CM[j, h_j(x)]$

- $\hat{f}_{x,j} \geq f_x$ for all y in $(1, \dots, r)$

$\hat{f}_{x,j} = f_x + \sum_{y \neq x} f_y$

$h_j(y) = h_j(x)$

$$E\Big[\sum_{\substack{y \neq x \\ h_j(y)=h_j(x)}} f_y\Big] = \frac{1}{B}\sum_{y \neq x} f_y \leq \frac{1}{B}\cdot m$$

Let $Z_j = \sum_{\substack{y \neq x \\ h_j(y)=h_j(x)}} f_y$.

$$E[Z_j] \leq m/B$$

$$Pr(Z_j \geq \varepsilon m) \leq \frac{E(Z_j)}{\varepsilon \cdot m} = \frac{m/B}{\varepsilon m} = \frac{1}{B\cdot\varepsilon}$$

By setting $B = 3/\varepsilon$, $Pr(Z_j \geq \varepsilon m) \leq \frac{1}{3}$

↳ the more rows you have, you multiply 1/3 → closer to 0

$$Pr(\min(Z_1,\ldots,Z_r) \geq \varepsilon m) = \underbrace{Pr(Z_1 \geq \varepsilon m, \ldots, Z_r \geq \varepsilon m)}_{\text{hashes are independent}} \leq \left(\tfrac{1}{3}\right)^r = \delta$$

$$r = \log_3\left(\tfrac{1}{\delta}\right)$$

$$\Downarrow$$

$$B = O\left(\tfrac{1}{\varepsilon}\right)$$

$$r = O\left(\log\left(\tfrac{1}{\delta}\right)\right)$$

$f_x \leq \hat{f_x} \leq f_x + \varepsilon m$ with probability $\geq 1-\delta$