Majority element and Heavy hitters

- Recall that x is a majority of $S = \{x_1, x_2, \dots, x_n\}$ if $|\{x_i \mid x_i = x\}| > \frac{n}{2}$
- We have seen a randomized algorithm for this with $O(n)$ expected running time

Assume now that we have a (long) datastream $\langle x_1, x_2, \dots, x_m \rangle$ where each $x_i \in \{1, 2, \dots n\}$

How do we detect a majority when there is one?

Here is a deterministic algorithm using one pass of the stream

Majority(S)

$c := 0$, $\ell := \phi$

For $i := 1$ to m

  if $(x_i = \ell)$ then $c := c+1$

  else $c := c-1$

  if $c \leq 0$ then

    $c := 1$, $\ell := a_i$

Return ℓ

We claim that the algorithm outputs the majority if there is one

Majority(S)

$c := 0$, $\ell := \emptyset$

For $i := 1$ to n do

if $(x_i = \ell)$ then $c := c+1$

else $c := c-1$

if $c \le 0$ then

$c := 1$, $\ell := a_i$

Return ℓ

Uses only $O(\log n)$ space for counter and $O(\log n)$ space for value as $x_i \in \{1, 2, \ldots n\}$

$S = \{3, 2, 1, 2, 2\}$

after loop	c	ℓ
1	1	3
2	1	2
3	1	1
4	1	2
5	2	2

2 is output

$S = \{3, 3, 2, 3, 2\}$

after loop	c	ℓ
1	1	3
2	2	3
3	1	3
4	2	3
5	1	3

3 is output

Why correct when x_j is the majority?

Suppose X_j occours more than $m/2$ times in S

Let $x = x_j$ be the value of the majority.

For each i such that $x_i = X$ we

(1) either have $\ell \neq X$ and then we decrease counter (and possibly set $\ell = X_i$)

(2) or $\ell = X$ and we increase the counter.

(1) can happen less than $m/2$ times

(2) The counter is ≥ 1 at the termination of each loop

so (2) will occur at the end

Heavy hitters in datastreams

k-frequency estimation (k-counters)

- let f_j be the number of times the value $j \in \{1, 2, \dots n\}$ occours in the stream A when

$A = \langle a_1, a_2, \dots a_m \rangle \qquad a_i \in \{1, 2, \dots n\}$

- Want to find estimates $\hat{f}_j$ such that

$$f_j - \frac{m}{k} \leq \hat{f}_j \leq f_j \quad \text{for all values } j \text{ in the stream}$$

- Suppose we are given ε with $0 < \varepsilon < 1$

Want a datastructure for ε-approximate heavy hitters so that we can return

- all j such that $f_j \geq \frac{m}{k}$ are in the list
- Every element in the list occours atleast $\frac{m}{k} - \varepsilon m$ times in A

Misra-Gries algorithm

Majority algorithm with k counters $C[1], C[2] \ldots C[k]$ instead of 1.

Let $L[1], L[2], \ldots L[k]$ be an array of k locations

Misra-Gries (A) (* A datastream of integers in $[n]$ *)

$C[i] := 0$, $L[i] := \emptyset$ forall $i \in [k]$

For $i := 1$ to m do

  If there is $j \in [k]$ s.t. $L[j] = a_i$ then $C[j] := C[j] + 1$

Else

  If $L[j] = \phi$ for some $j \in [k]$ then $C[j] := 1$, $L[j] := a_i$

  else for $j := 1$ to k $C[j] : C[j] - 1$

For $j := 1$ to k do

  If $C[j] \leq 0$ do $L[j] := \phi$

If $L[j] = \phi$ for some $j \in [k]$ then $C[j] := 1$, $L[j] := a_i$ (* try to use a counter for a_i if one is free *)

Return C, L

On query $q \in [n]$:

- If $\exists j \in [k]$ with $L[j] = q$ return $\hat{f}_q = C[j]$
- otherwise return $\hat{f}_q = 0$

Misra-Gries (A) (* A datastream of integers in [n] *)

$C[i] := 0,\ L[i] := \emptyset$ forall $i \in [k]$

For $i := 1$ to m do

  If there is $j \in [k]$ s.t. $L[j] = a_i$ then $C[j] := C[j]+1$

Else

  if $L[j] = \emptyset$ for some $j \in [k]$ then $C[j] := 1,\ L[j] := a_i$

  else for $j := 1$ to k $C[j] : C[j] - 1$

For $j := 1$ to k do

  if $C[j] \le 0$ do $L[j] := \emptyset$

if $L[j] = \emptyset$ for some $j \in [k]$ then $C[j] := 1,\ L[j] := a_i$ (* try to use a counter for a_i if one is free *)

Return C, L

Correctness

- A counter $C[j]$ with $L[j] = q$ is only incremented if $a_i = q$ so $\hat{f}_q \le f_q$ holds always
- If $C[j]$ with $L[j] = q$ is decremented then all other counters are also decremented

Happens $\le \frac{m}{k}$ times (as all $C[i] \ge 1$)

so the counter $C[j]$ represents $L[j] = q$

is decremented at most m/k times

so $f_q - m/k \le \hat{f}_q$

ε-approximate frequency estimation

with $\varepsilon = \frac{1}{2k}$ the algorithm

outputs all values with frequency count

at least $\frac{m}{k}$ and only

values with frequency count at least $\frac{m}{2k}$

our space usage: $O(k) = O\left(\frac{1}{\varepsilon}\right)$

counters each of size $\log m$

Reasonable since there may be up to

$\frac{m}{k}$ heavy hitters (occurring $\geq \frac{m}{k}$ times)

Count-min Sketch

Assume S is a (possibly very long) stream of data on which we want to estimate the frequencies of elements which occur often in S

For example to solve the approximate heavy hitters problem

- Let b, ℓ be integers to be determined below
- Let $\mathcal{H}$ be a universal family of hash functions $h \in \mathcal{H}$ hashes $\mathcal{U} \to [b]$ where $\mathcal{U}$ is the universe of all possible elements in the stream.
- Let $h_1, h_2, \dots, h_\ell$ is distinct members from $\mathcal{H}$
- When we say that $h_i \in \mathcal{H}$ is universal we mean that h_i is a random member of $\mathcal{H}$

We use $h_1, h_2, \dots h_\ell$ to build an $\ell \times b$ array M of counters as follows

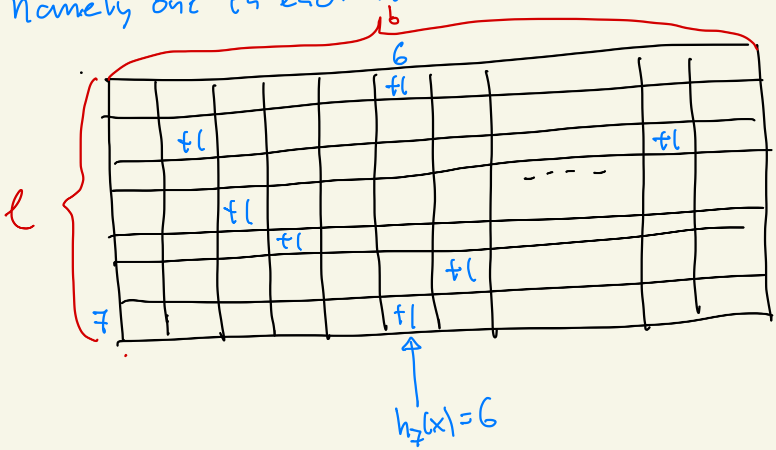
Initialy $M_{i,j} = 0$ for $i \in [\ell]$ and $j \in [b]$

For each element x in the stream we process it as follow:

For every $i \in \ell$: $M_{i,h_i(x)} \Leftarrow M_{i,h_i(x)} + 1$

So each new element x increases exactly ℓ entries of M

namely one in each row



Small example $\ell = 3$; $b = 4$

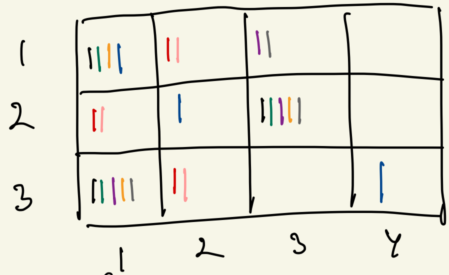
	A	B	C	D
h_1	1	2	1	3
h_2	3	1	2	3
h_3	1	2	4	1

$S = \{A, B, A, D, A, B, D, C$

||||	||	||	
||	|	|||||	
|||||	||		|

$S = \{A, B, A, D, A, B, D, C$

	A	B	C	D
h_1	1	2	1	3
h_2	3	1	2	3
h_3	1	2	4	1



What can be said about the frequencies of the elements of the stream seen so far?

A: $m_{1,1} = 4$ real frequency $= 3$

$m_{2,3} = 5$

$m_{3,1} = 5$

B: $m_{1,2} = 2$

$m_{2,1} = 2$ real frequency $= 2$

$m_{3,2} = 2$

C: $m_{1,1} = 4$

$m_{2,2} = 1$ real frequency $= 1$

$m_{3,4} = 1$

D: $m_{1,3} = 2$

$m_{2,3} = 5$ real frequency $= 2$

$m_{3,1} = 5$

We saw that $M_{i,h_i(x)}$ is always at least the frequency of x and often higher.

Why?

(a) Each occurence of x increases $M_{i,h_i(x)}$ by one

So $M_{i,h_i(x)} \geq f_x$ when f_x is real frequency of x (so far)

(b) Every occurence of a $y \neq x$ with $h_i(x) = h_i(y)$ will also increase $M_{i,h_i(x)}$

- Let S_n be the first n elements of the stream ($n = 8$ in example)
- Denote by f_y, the frequency (# of occurences) of y in S_n
- Denote $M_{i,h_i(x)}$ by $Z_{i,x}$

$Z_{i,x}$ is a random variable depending on the random choice of $h_i \in \mathcal{H}$

Define the indicator variable $I_{i,x}$ as follows

$$I_{i,x}(y) = \begin{cases} 1 & \text{if } h_i(x) = h_i(y) \\ 0 & \text{otherwise} \end{cases}$$

As h_i is universal $p(I_{i,x}(y) = 1) \leq \frac{1}{b}$

Then it follows from (a) and (b) that

$$Z_{i,x} = f_x + \sum_{\{y \in S_n \mid y \neq x\}} f_y \cdot I_{i,x}(y) \geq f_x$$

What is the expected value of $Z_{i,x}$?

We will use $\sum_{y \in S_n} f_y = n = |S_n|$

$$E(Z_{i,x}) = E\left(f_x + \sum_{\{y \in S_n | y \neq x\}} f_y \cdot I_{i,x}(y)\right)$$

$$= E(f_x) + E\left(\sum_{\{y \in S_n | y \neq x\}} f_y \cdot I_{i,x}(y)\right)$$

$$= f_x + \sum_{\{y \in S_n | y \neq x\}} f_y \cdot E(I_{i,x}(y))$$

$$\leq f_x + \sum_{\{y \in S_n | y \neq x\}} f_y \cdot \frac{1}{b}$$

$$\leq f_x + \frac{1}{b} \sum_{\{y \in S_n | y \neq x\}} f_y$$

$$\leq f_x + \frac{1}{b} \sum_{y \in S_n} f_y$$

$$= f_x + \frac{n}{b}$$

So the expected value of $Z_{i,x}$ is off by at most $\frac{n}{b}$.

As n may be huge and we have only b counters our estimate must depend on n

Let us bound the probability that our estimate for f_x is more than $\frac{2n}{b}$ away

$$p\left(Z_{i,x} - f_x \geq \frac{2n}{b}\right) \leq \frac{E(Z_{i,x} - f_x)}{\frac{2n}{b}} = \frac{\frac{n}{b}}{\frac{2n}{b}} = \frac{1}{2} \quad (\square)$$

This holds for all values $i \in [\ell]$!

Let $\hat{f}_x = \min_{i \in [\ell]} Z_{i,x}$ then $\hat{f}_x \geq f_x$

and since $h_1, h_2, \ldots, h_\ell$ are independent of each other

$(\square)$ implies that

$$p\left(\hat{f}_x - f_x \geq \frac{2n}{b}\right) \leq \frac{1}{2^\ell} \qquad (*)$$

Suppose we are given ε, δ and we want that

$$p\left(\hat{f}_x - f_x \geq \varepsilon n\right) \leq \delta$$

By $(*)$ if we take $b = \frac{2}{\varepsilon}$ and $\ell = \log_2\left(\frac{1}{\delta}\right)$ We ignore that these may not be integers

Then

$$p\left(\hat{f}_x - f_x \geq \varepsilon n\right) = p\left(\hat{f}_x - f_x \geq \frac{2n}{b}\right) \leq 2^{-\ell} = 2^{-\log\left(\frac{1}{\delta}\right)} = \frac{1}{\frac{1}{\delta}} = \delta$$

So $p\left(\hat{f}_x - f_x \geq \varepsilon n\right) \leq \delta$

We use $b \cdot \ell = \frac{2}{\varepsilon} \cdot \log\left(\frac{1}{\delta}\right)$ counters (the array M)

to implement the sketch and we achieve the desired accuracy $p(\hat{f}_x - f_x \geq \varepsilon n) \leq \delta$

Independently of n the length of the stream!!

For example: suppose we want to estimate the frequencies of those elements that have frequency at least $\frac{n}{100}$ and we want the estimate to be off by 1% with probability at most $\frac{1}{1000}$

Take $\varepsilon = 10^{-4}$ and $\delta = 10^{-3}$ then 1% of $\frac{n}{100}$ is $\frac{n}{10^4} = 10^{-4} n$
$= \varepsilon n$

so

$$p\left(\hat{f}_x - f_x \geq 10^{-4} n\right) \leq 10^{-3}$$

when we use $b = 2 \cdot 10^4$ and $\ell = \log_2(10^3) \sim 10$

So we use only $2 \cdot 10^4 \cdot 10 = 200000$ counters

to achieve the desired accuracy no matter how long the stream is