Logistics:

- HW 1 posted (due 2/5)
- Need help? Come to office hours:
  - Tue 1-2 pm
  - Wed 4-5 pm

  } CDS 1443
  - If need be we can schedule extra office hours

Today/this week:

- Continue Count-Min Sketch
- Frequency moments & AMS sketch for F_2
- Constructing good hash functions

Last time:

Goal: - tracking elements from X coming and leaving
- providing frequency estimates on request

First attempt:

k buckets

- h is a hash function from X to $[k] = \{1, 2, \ldots, k\}$

  "fully random" ≡ each $h(x)$ uniformly distributed and independent of other values of h

- maintain array $A[1 \ldots k]$ of counters
- initially, $A[i] = 0$ for all $i \in [k]$
- operations:

  Insert(x): $A[h(x)] \leftarrow A[h(x)] + 1$

  Delete(x): $A[h(x)] \leftarrow A[h(x)] - 1$

  Query(x): return $A[h(x)]$

total number of items in the same bucket

Analysis:

Definitions:

- $f(x) \stackrel{\text{def}}{=}$ number of copies of x

  ↖ What Query(x) "should" return

- $g(x) \stackrel{\text{def}}{=}$ estimate provided by our algorithm, i.e., $A[h(x)]$

- collision $C_{x,y} = \begin{cases} 1 & \text{if } h(x)=h(y) \\ 0 & \text{if } h(x)\neq h(y) \end{cases}$

x, y: elements of X

- Note that

what we would love to obtain

$$g(x) = \sum_{\substack{y\in X \\ \text{s.t. } h(y)=h(x)}} f(y) = \boxed{f(x)} + \underbrace{\sum_{y\in X\setminus\{x\}} C_{x,y} f(y)}_{\text{error} \geq 0}$$

linearity of expectation

non-negative random variable

$$\mathbb{E}[\text{error}] = \sum_{y\in X\setminus\{x\}} f(y) \cdot \underbrace{\mathbb{E}[C_{x,y}]}_{= \Pr[h(x)=h(y)] = \frac{1}{k}}$$

$x \neq y$

+ random properties of our hash function

$$= \frac{1}{k} \sum_{y\in X\setminus\{x\}} f(y) \leq \frac{1}{k} \sum_{y\in X} f(y) = \frac{1}{k} \|f\|_1$$

(We write $\|f\|_1$ to denote $\sum_{y \in X} |f(y)| = \sum_{y \in X} f(y)$ above. This notation corresponds to treating f as a long vector of counts, where each coordinate is $f(y)$ for some $y \in X$.)

By Markov's inequality,

$$\Pr\left[\text{error} \geq \frac{2}{k}\|f\|_1\right] \leq \frac{\mathbb{E}[\text{error}]}{\frac{2}{k}\|f\|_1} \leq \frac{1}{2},$$

and by setting $k = \lceil 2/\varepsilon \rceil$ for any $\varepsilon \in (0,1)$, we obtain

(k: number of buckets)

$$\Pr\left[\text{error} \geq \varepsilon\|f\|_1\right] \leq \Pr\left[\text{error} \geq \frac{2}{k}\|f\|_1\right] \leq \frac{1}{2}.$$

Hence if $k = \lceil 2/\varepsilon \rceil$, for any fixed x,

$$f(x) \leq g(x) \leq f(x) + \varepsilon\|f\|_1,$$

($f(x) \leq g(x)$: always true; $g(x) \leq f(x) + \varepsilon\|f\|_1$: with probability $\geq \frac{1}{2}$)

Amplifying the probability of a "good" estimate

Solution:

- run t independent copies of our "first attempt"
- to answer a frequency query about x, query all t copies and return the minimum of their estimates

Analysis:

- because all estimates are $\geq f(x)$, so will be their minimum
- if at least one estimate is "good", i.e., $\leq f(x) + \varepsilon \|f\|_1$, then the returned value is in $[f(x); f(x) + \varepsilon \|f\|_1]$
- $\Pr[\text{at least one estimate "good"}]$
  $= 1 - \underbrace{\Pr[\text{no estimate is "good"}]}_{\leq \left(\frac{1}{2}\right)^t \text{ due to the independence of all copies}}$

$\geq 1 - 2^{-t}$

By setting $t = \lceil \log_2(1/\delta) \rceil$ for any $\delta \in (0, \frac{1}{2})$, we get

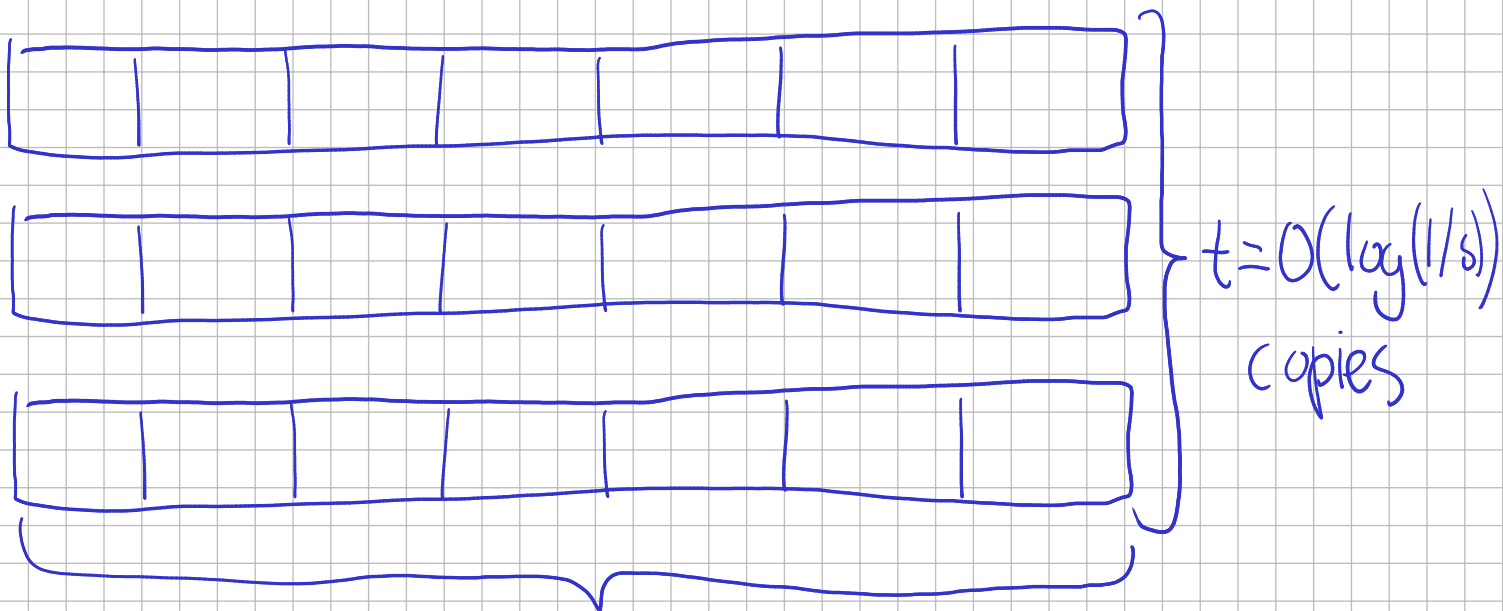
$$f(x) \leq \text{returned estimate} \leq f(x) + \varepsilon \|f\|_1$$

returned estimate = minimum of t independent copies

$f(x) \leq$ returned estimate: always true

returned estimate $\leq f(x) + \varepsilon \|f\|_1$: with probability $\geq 1 - \delta$

This is Count-Min Sketch:



Total space (ignoring hash functions): $O(\frac{1}{\varepsilon} \log(1/\delta))$

Warning: we don't have "fully random" hash functions

But what properties did we use?

Only that $\mathbb{E}[C_{x,y}] = \frac{1}{k}$ for $x \neq y$.

For this it suffices that for any pair $x \neq y$, $h(x)$ and $h(y)$ are independent and uniformly distributed.

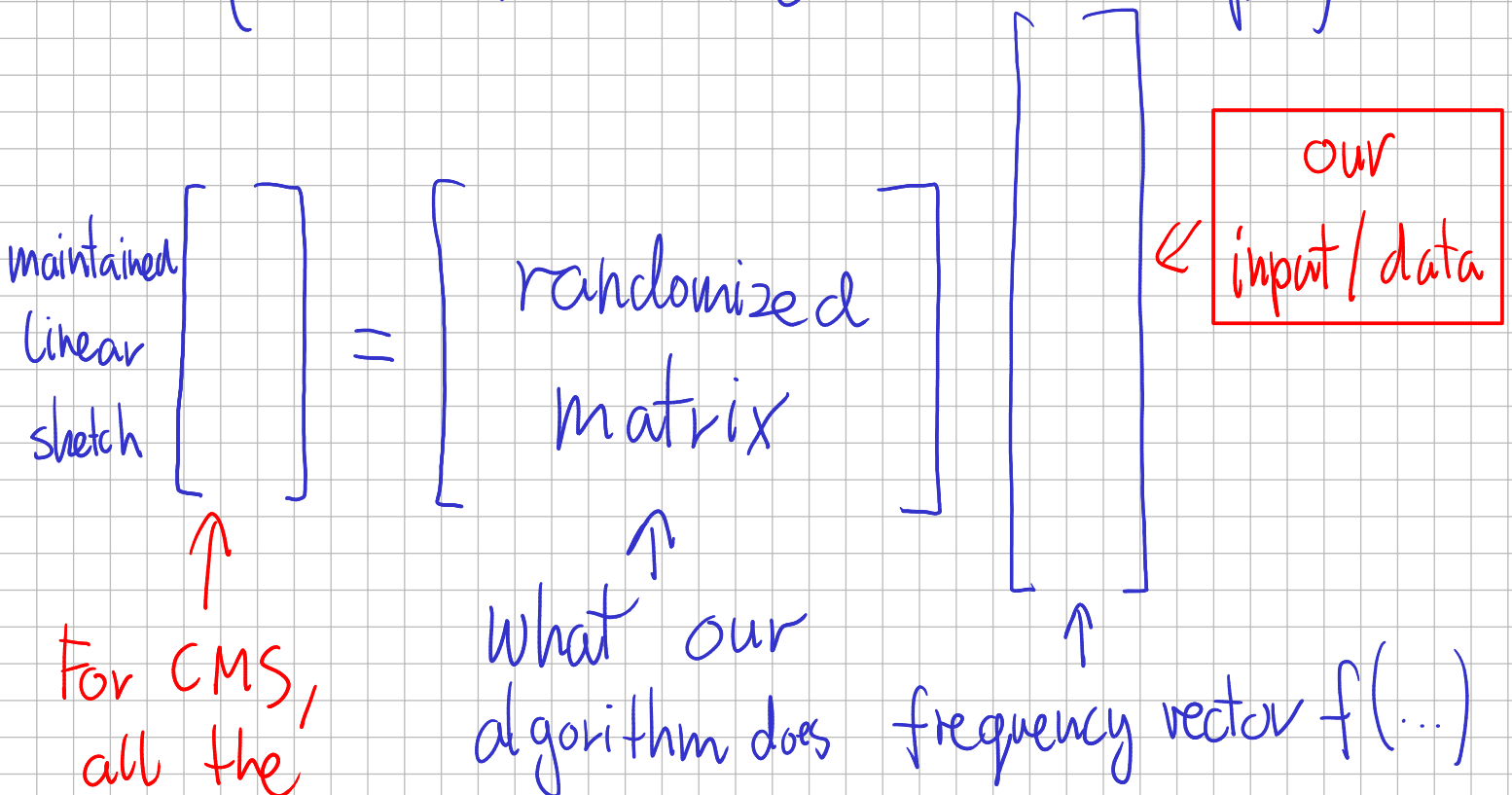
pairwise independence

Later this week: How to construct such hash function.

Note: Obtaining $\mathbb{E}[C_{x,y}] = \frac{c}{k}$ ← constant > 1 may sometimes be easier and can be countered by increasing the number of buckets by a constant

Important concept: Linear sketches

(Count-Min Sketch is an example)



For CMS, all the kt counters

What our algorithm does

frequency vector $f(\ldots)$

Tomorrow: properties of linear sketching algorithms