Last time:
- matching
- intro hashing
- universal hashing

Key idea:
- put the randomness in hash function instead of assuming data is random
- give all the nice properties
  - data distributed randomly throughout table
  - hash fn efficient to store, efficient to compute.

Today:

3 applications of hashing & lossy compression
- Bloom filters
- Heavy hitters & count-min sketch
- Distinct els
with short review of variance & tail bounds
Heavy hitters

Stream of elts at any time \( t \),

let \( f^+_x \) \# times seen element \( x \) in \( a_1, a_2, ..., a_t \)

Goal: when elt shows up, output that element

\[ f^+_x > \frac{n}{K} \]

such an element is a heavy hitter.

Space used proportional to \# unique els

Not possible to solve this problem exactly with sublinear space.

Modified goal \((\varepsilon, \delta)\)

1. If \( f^+_x > \frac{n}{K} \) output \( x \).
2. If \( x \) output, then w.prob at least \( 1 - \delta \) it is the case that \( f^+_x \geq \frac{n}{K} - \varepsilon n \)

For example: suppose \( k = 25 \), \( \varepsilon = 0.01 \)

\[ \delta = \frac{1}{2^{10}} \]

1. If \( f^+_x > \frac{n}{25} = 0.04n \), then output \( x \)
2. If \( x \) output then w.prob \[ 1 - \frac{1}{2^{10}} \]

\[ f^+_x \geq 0.04n - \frac{0.01n}{\varepsilon} = 0.03n \]
Count-min sketch: Design specifications \( n, k, \delta, \varepsilon \Rightarrow b, \ell \)

Keep 2D array called CMS

![Diagram of CMS array]

When element \( x \) shows up

\[
\text{Inc}(x): \quad \forall 1 \leq j \leq \ell \quad \text{increment } CMS[j][h_j(x)]
\]

Observe: \( \forall x: CMSG \cup_{j=0}^{\ell} CMS[j][h_j(x)] = f^+_x \)

\[
\text{Count}(x): \quad \text{return } \min_{1 \leq j \leq \ell} CMS[j][h_j(x)]
\]

if this value \( \geq \frac{n}{\varepsilon} \), output \( x \) as \( \text{HT} \).

by observation: \( \text{Count}(x) \geq f^+_x \)

Construction

1. Hash functions behave randomly
   \( \forall x, y, 1 \leq j \leq \ell \quad \Pr(h_j(x) = h_j(y)) = \frac{1}{b} \)

2. Hash functions for \( 1 \leq j \leq \ell \) are independent of each other
   \( \text{Universal class of hash functions} \)
Analysis

Fixing $f^+$, let $y$ trace arrive at $t$.


$Z_j = f^+_x + \sum_{y \neq x} f^+_y W_{xy}$

\[
E(Z_j) = \underbrace{f^+_x + \sum_{y \neq x} f^+_y E[W_{xy}]}_{\text{by linearity of expectation}}
\]

\[
E(Z_j - f^+_x) \leq \frac{\varepsilon}{b}
\]

$Pr(Z_j - f^+_x > \frac{\varepsilon}{b}) \leq \frac{1}{2^c}$

$Pr(\text{Count}(x) - f^+_x > \frac{\varepsilon}{b}) \leq \frac{1}{2^c}$

Conclusion: $Pr(\text{Count}(x) > f^+_x + \frac{2n}{b}) \leq \frac{1}{2^c}$

Modified goal $(E, \delta)$

1. If $f^+_x > \frac{n}{2}$ output $x$.

2. If $x$ is output, then with prob at least $1 - \delta$ it is the case that $f^+_x > \frac{n}{2} - E\cdot \delta$.

Choose $b$ and $c$ so that

\begin{align*}
\delta &= \frac{1}{2^c} \\
\frac{1}{2^c} &= b \cdot \frac{\varepsilon}{b} \\
E &= \frac{1}{2^c}
\end{align*}

Suppose $f^+_x < \frac{n}{2} - E\cdot \delta$

$Pr(\text{Count}(x) > \frac{n}{2}) < Pr(\text{Count}(x) > f^+_x + E\cdot \delta) \leq \delta$
Universal hash family \( \mathcal{H} \)  

\[ \forall h \in \mathcal{H}, \quad h : U \rightarrow \{0, \ldots, b-1\} \]

If \( h \) is chosen uniformly at random from \( \mathcal{H} \)

\[ \forall x \neq y, \quad \Pr_{h \in \mathcal{H}}(h(x) = h(y)) = \frac{1}{b} \]

Choose any prime \( p > n \)

\[ \mathcal{H} = \{ h(x) = (e \cdot x + g) \mod p \mod b, \quad 1 \leq e \leq p-1, \quad 0 \leq g \leq p-1 \} \]

\[ |\mathcal{H}| = p(p-1) \]

Keep 2D array called \( \text{CHS} \)