Heavy Hitters (HH)

There is no alg to solve HH in one pass with sublinear space.

We relax our requirements \( \Rightarrow \) \( \exists \) -HH

Suppose at time \( t \) \( \forall i \in U \)

\[ x_i^+ = \# \text{ times seen } i \text{ up to time } t \]

Will require estimates

\[ \tilde{x}_i^+ \in x_i^+ \pm \epsilon t \] \( (\star) \)

that used space \( \ll \min(t, n) \)

\( \# \text{ distinct sets} \)

Idea: counting Bloom filter
\( h : \mathbb{U} \rightarrow [m] \) random universal \( A \) array with \( m \) elts

when elt \( i \) arrives

\[ A[h(i)]++ \]

\[ x_i^+ = A[h(i)] \]

\[ = x_i^+ + \sum_{j \neq i} x_j^+ 1_{h(j) = h(i)} \]

\[ E[A[h(i)]] = x_i^+ + \frac{\sum x_j^+}{\frac{n}{m}} \leq x_i^+ + \frac{n}{m} \]

with \( m = \frac{1}{\varepsilon} \)
satisfy \( (\star) \) in expectation

Want low error w/ high probability

\[ \Rightarrow \text{amplify success probability w/ indep repetitions} \]

Do same thing independently \( \ell \) times w/ \( \ell \) independently

\( h_1, h_2, \ldots, h_\ell \)

\( A, A, \ldots, A \)

Selected hash fns

Called Count Min sketch
How should we combine entries? We have overestimates

$$\Rightarrow \bar{z}_i^+ = \min_{i \leq j \leq k} [h_j(i)]$$

$$E(x_i^+ - \bar{x}_i^+) \leq \frac{n}{m}$$

$$\Rightarrow \Pr(x_i^+ - \bar{x}_i^+ > \frac{2n}{m}) \leq \frac{1}{2^k}$$

Markov Inequality

$$\Pr(\bar{z}_i^+ > \frac{2n}{m}) = \Pr(\text{all estimates too high}) \leq \frac{1}{2^k}$$
How to choose parameters?

\[ Z_i^+ \subseteq [x_i^+, x_i^+ + \frac{2n}{m}] \quad \text{with prob } \geq 1 - \frac{1}{\delta^2} \]

\[ \frac{2}{m} = \epsilon \quad \quad l = \log_2 \left( \frac{1}{\delta} \right) \]

\[ Z_i^+ \subseteq [x_i^+, x_i^+ + \epsilon m] \quad \text{with prob } \geq 1 - \delta \]

useful when \( x_i^+ \approx 2\epsilon m \)

Space usage: \( l m \) counters of \( \log_2 T \) bits each

\[ \Rightarrow O \left( \frac{\log \delta}{\epsilon} \log T \right) \]

also space for hash functions: \( O (l \log U) \)

Example: want to output all items that constitute \( \geq 4\% \) of

flow and output no items that constitute \( < 2\% \)