distributed nodes

Many nodes in graph
  - each node knows only small number of neighbors
  - need to communicate or calculate

Key bottleneck is communication

-------------

Mergeable Summaries:

Many unorganized nodes [1,...,k] each with data X_i.
  <Connected in tree structure>

\[ X = \bigcup_{i} X_i \]

Want \( S = \text{summ}(X) \), but don't want to send X.

Key operation:
  - given \( S_1 = \text{summ}(X_1) \) and \( S_2 = \text{summ}(X_2) \)
  - produce \( S_{12} = \text{summ}(X_1 \cup X_2) \)

-------------

Example: \( X_1 = \{1,2,3,8,9\} \)
  \( X_2 = \{4,5,89,90,91\} \)
  \( X_3 = \{6,7,92,93,94\} \)

\( m_1 = \text{median}(X_1) = 3 \)
\( m_2 = \text{median}(X_2) = 89 \)
\( m_3 = \text{median}(X_3) = 92 \)
\( \text{median}\{m_1,m_2,m_3\} = 89 \)
\( \text{median}(X_1 \cup X_2 \cup X_3) = 8 \)

Often error (or size) accumulates

-------------

Goal: \( S = \text{summ}(X) \) is a \( \epsilon \)-approximation of \( X \)

-----

\( X \) multi-subset \([n]\)
\( f_i = |\{x_j \in X \mid x_j = i\}| \)

\( \epsilon \)-approx frequency values
\( |f_i - \bar{f}_i| \leq \epsilon \)
\( F_1 = \epsilon m \)

\( f_i \)
size $S = 1/\varepsilon$
-----
- error is relative
- size depends only on $\varepsilon$

key operation:
given: $S_1 = \text{sum}(X_1)$, $S_2 = \text{sum}(X_2)$
  - $S_i$ is $\varepsilon$-approx of $X_i$
  - size($S_i$) = $f(1/\varepsilon)$
output: $S_{12} = \text{sum}(X_1 \cup X_2)$
  - $S_{12}$ is $\varepsilon$-approx of $X_1 \cup X_2$
  - size($S_{12}$) = $f(1/\varepsilon)$

* neither size, nor error increase

-------------

Misra-Gries Summaries:
$S =$
Let $C$ be array of $k$ counters $C[1], C[2], \ldots, C[k]$
Let $L$ be array of $k$ locations $L[1], L[2], \ldots, L[k]$

$S_1 = (C_1, L_1) = \text{sum}(X_1)$
$S_2 = (C_2, L_2) = \text{sum}(X_2)$

$k = 1/\varepsilon = 3$

$S_{12}$ [1 + 0] [2 + 3] [0 + 4] [0 + 0] [3 + 0] [0 + 2]
  -> [1] [5] [4] [0] [3] [2]*
  -> [0] [3] [2] [0] [1] [0]

- add like counters together (at most $2k$)
- retain just top $k$ after subtracting $C[k+1]$, set rest to 0.

proof:
Each subtraction removes $\geq k$ items
can subtract at most $m/k$ times
each value $\sim f_i$ in $[f_i, f_i - m/k] = [f_i, f_i - \varepsilon m]$

-------------

comutative, associative

Any linear summary:
$\text{sum}(X_{12}) = \text{sum}(X_1) + \text{sum}(X_2)$
Any idempotent summary:
\[ \max(X_{12}) = \max\{\max(X_1), \max(X_2)\} \]

-------------------

count-min sketch

**********************
t independent hash functions \{h_1, ..., h_t\}
each \( h_i : [n] \rightarrow [k] \)

2-d array of counters:
h_1 \rightarrow [C_{1,1}] [C_{1,2}] ... [C_{1,k}]
h_2 \rightarrow [C_{2,1}] [C_{2,2}] ... [C_{2,k}]
... ...
\( h_t \rightarrow [C_{t,1}] [C_{t,2}] ... [C_{t,k}] \)

for each \( a \in A \) -> increment \( C_{i,h_i(a)} \) for \( i \) in \([t]\).

\( \hat{f}_a = \min_{i \in [t]} C_{i,h_i(a)} \)

Set \( t = \log(1/\delta) \)
Set \( k = 2/\varepsilon \)

***************
can add or subtract!

--------------

\(-\) 11111111111
Random Sample size \( k = O(1/\varepsilon^2) = S \)
\( R_{-\text{rank}}(S(v)) = S(v) \)
\( |R_{-\text{rank}}(v) - S(v)| <= \varepsilon \)

\( S_1 = \{(s_1, u_1), (s_2, u_2), ...\} \)
\( S_2 = \{(s_1, u_1), (s_2, u_2), ...\} \)
- \( u_i \) at random for each \( s_i \)
- keep top \( k \) values \( u_i \) (and paired \( s_i \))
easily mergeable, maintain random sample size k.

Maintain sorted list of size \( k = O(1/\epsilon \sqrt{\log(1/\epsilon)}) \)

\( S_1 = \{s_{11}, s_{12}, s_{13}, \ldots, s_{1k}\} \)
\( S_2 = \{s_{21}, s_{22}, s_{23}, \ldots, s_{2k}\} \)

s.t. \( s_{i,j} < s_{i,j+1} \) for \( i = \{1,2\} \)

\( S_{12} = \)
1. merge sort \( S_1, S_2 \) -> ordered list size 2k
2. select even points / odd points at random

***magically, error does not accumulate, nor probability of failure
older merges less important towards relative error

above only works for \(|X_1| = |X_2|\)

if not true, need size \( O((1/\epsilon) (\log(1/\epsilon))^{3/2}) \)