CS719: Data Stream Processing

Lecture 14,15, part of 16
Heavy Hitters

IIT Kanpur
Jan-Apr 2010

Outline

Outline

Problem Definition and Introduction

ℓ_1 point query and heavy-hitters: COUNT-MIN sketch

- COUNT-MIN-sketch: ApproxHH$_1$ by domain iteration
- COUNT-MIN sketch: Approx HH by Group Testing

ℓ_2 Point Query and Heavy Hitters: the COUNTSKETCH structure

- COUNTSKETCH: Structure and Simple Analysis
- COUNTSKETCH and COUNT-MINsketch: Residual F_2 based bounds

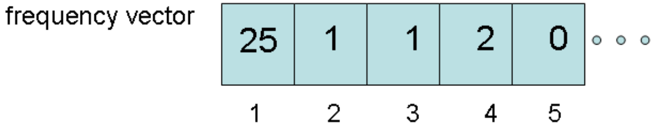
Comparing ℓ_1 and ℓ_2 point query estimators

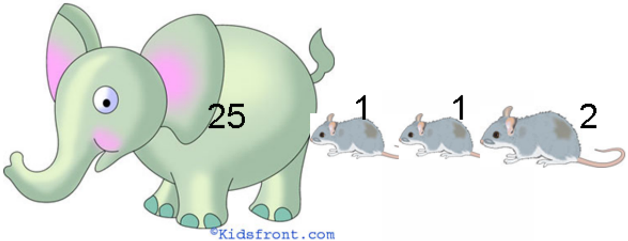
Sparse Approximation

Heavy Hitters: Illustration

Heavy Hitters are items with large absolute frequencies (Elephants)

stream: $(1, 10)(2, 1)(3, 1)(4, 2)(1, 10) \ldots$





Elephants and mice

Heavy Hitter Problem: Find the elephants

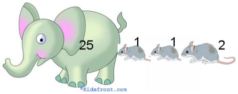
Applications

- Among the most popular applications of data streaming.

    1. Find the IP-addresses that send the most traffic.
    2. Find source-IP, dest-IP pairs that send the most traffic to each other.
    3. Find the most visited web sites.

    $\vdots$

Heavy Hitters: Definition

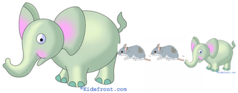


- ℓ_p heavy hitters with parameter ϕ:

$$HH_p^\phi(f) = \left\{ i \in [n] : |f_i|^p \geq \phi \sum_{i \in [n]} |f_i|^p \right\} \quad .$$

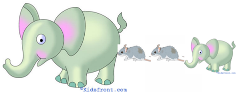
- ℓ_p-heavy hitters problem. First Attempt: Given ϕ, find the set HH_p^ϕ over stream.
- We want to achieve this in low space, close to $O(\frac{1}{\phi}$.
- Finding HH_p^ϕ exactly requires $\Omega(n)$ space. [Proof later lectures.]
- Settle for *Approximate Heavy Hitters*.

Approximate Heavy Hitters: Definition



- Recall: $HH_p^{\phi}(f) = \left\{ i \in [n] : |f_i|^p \geq \phi \sum_{i \in [n]} |f_i|^p \right\}$.
- Approximate heavy hitters: $\text{ApproxHH}_p^{\phi,\phi'}$. Two parameters ϕ and ϕ' with $\phi' < \phi$.
- Return set S such that
  1. S includes HH_p^{ϕ}. Meaning: Do not miss i with $|f_i|^p > \phi F_p$.
  2. S is included in $HH_p^{\phi'}$. Do not include i with $|f_i|^p < \phi' F_p$.

Simple Fact



- Since $HH_p^{\phi}(f) = \left\{ i \in [n] : |f_i|^p \geq \phi \sum_{i \in [n]} |f_i|^p \right\}$.

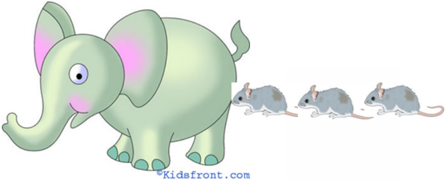
$$\left| \mathrm{HH}_p^{\phi} \right| \leq \left\lfloor \frac{1}{\phi} \right\rfloor .$$

- Let $S =$ Approximate Heavy Hitters (ϕ, ϕ').
- Since $S \subseteq \mathrm{HH}_p^{\phi'}$, number of approximate HH (ϕ, ϕ') is at most $\lfloor 1/\phi' \rfloor$.

ℓ_p Point Query/Estimating Frequencies

- Point query: Estimate frequency of any item i. Cannot be done exactly in $o(n)$ space. Allow bounded error:

$$\hat{f}_i^p = f_i^p \pm \phi F_p, \quad \forall i \in [n] \quad .$$



Frequency Vector

Estimated frequencies

An obvious question

- Which is better: $\mathrm{PtQuery}_1(\phi)$ or $\mathrm{PtQuery}_2(\phi)$?
- i.e., which is smaller: ϕF_1 or $(\phi F_2)^{1/2}$?

Plan for this topic

- ℓ_1 heavy-hitters.
  1. COUNT-MIN Algorithm for ℓ_1 point query.
  2. Application to ℓ_1 heavy-hitters.
- ℓ_2 heavy-hitters.
  1. COUNTSKETCH algorithm for ℓ_2 point query.
  2. Application to ℓ_2 heavy-hitters.
- Brief description of a few other algorithms.

References

- COUNT-MIN algorithm: Cormode, Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. J. Algorithms 55(1): 58-75 (2005).
- ℓ_1-HH: Cormode, Muthukrishnan: What's New: Finding Significant Differences in Network Data Streams. IEEE INFOCOM 2004.
- COUNTSKETCH algorithm: Charikar, Chen, Farach-Colton. Finding frequent items in data streams. Theor. Comput. Sci. 312(1): 3-15 (2004).

Outline

Problem Definition and Introduction

ℓ_1 point query and heavy-hitters: COUNT-MIN sketch

- COUNT-MIN-sketch: ApproxHH$_1$ by domain iteration
- COUNT-MIN sketch: Approx HH by Group Testing

ℓ_2 Point Query and Heavy Hitters: the COUNTSKETCH structure

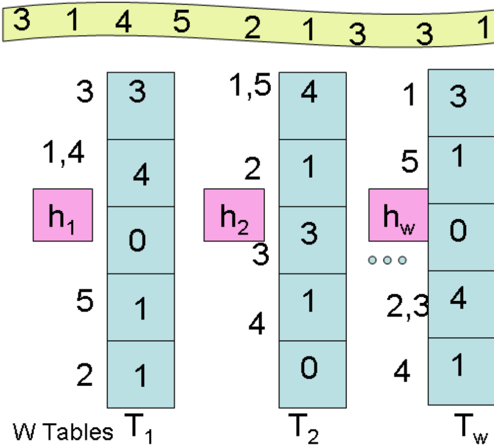
- COUNTSKETCH: Structure and Simple Analysis
- COUNTSKETCH and COUNT-MINsketch: Residual F_2 based bounds

Comparing ℓ_1 and ℓ_2 point query estimators

Sparse Approximation

Count-Min Sketch: Basic algorithm

- w hash tables $T_1, T_2, \ldots, T_w$.
- Each table T_j:
  1. B buckets.
  2. hash fn. $h_j : [n] \rightarrow [B]$.
  3. $h_j \in_R$ pair-wise indep. family.
- h_j's independent.
- UPDATE(i, v): $T_j[h_j(i)]$ **+=** v, $j = 1, 2, \ldots, w$.
- ESTIMATE(i):



All non-negative frequencies
$\hat{f}_i = \min_{j=1}^{w} T_j[h_j(i)]$

General frequencies
$\hat{f}_i = \mathrm{median}_{j=1}^{w} T_j[h_j(i)]$

Simple Analysis: Non-negative frequencies

- Fix table index $j \in \{1, 2, \ldots, w\}$. Let $T = T_j$ and $h = h_j$.

$$T[h(i)] = f_i + \sum_{h(k)=h(i), i \neq k} f_k \ .$$

So, $\mathrm{E}\left[T[h(i)] - f_i\right] = \sum_{k \neq i} f_k \Pr\left[h(k) = h(i)\right] = \dfrac{F_1 - f_i}{B}$.

- By Markov's ineq. $\Pr\left[T[h(i)] - f_i > \frac{2F_1}{B}\right] < \frac{1}{2}$.
- Consider all w tables.

$$\Pr\left[\min_{j=1}^{w}(T_j[h_j(i)] - f_i) > \frac{2F_1}{B}\right] = \Pr\left[\forall j \ \ T_j[h_j(i)] - f_i > \frac{2F_1}{B}\right] < \left(\frac{1}{2}\right)^2$$

- $$f_i \leq \min_{j=1}^{w} T_j[h_j(i)] \leq f_i + \frac{2F_1}{B}, \text{ with prob. } 1 - 1/2^w \ .$$

Count-Min sketch: point query, f_i's non-negative

- $\hat{f}_i = \min_{j=1}^{w} T_j[h_j(i)]$.
- For non-negative frequency vector:

$$f_i \le \hat{f}_i \le f_i + \frac{2F_1}{B}, \text{ with prob. } 1 - 1/2^w \quad .$$

- To solve point query s.t. $\hat{f}_i \in f_i \pm \phi F_1$ with prob. $1 - \delta$ set $B = \frac{1}{2\phi}$ and $w = \log \frac{1}{\delta}$.
- Space requirement: $Bw = \frac{1}{\phi} \log \frac{1}{\delta}$ counters of size $\log(mn)$, where, $m = \max_i f_i$.

Analysis: $f \in \mathbb{Z}^n$

- Fix table $T = T_j$. Now,

$$T[h(i)] - f_i = \sum_{\substack{i \neq k \\ h_j(i) = h_j(k)}} f_k = \sum_{i \neq k} f_k x_k$$

where, x_k's are indicator variables, $x_k = 1$ if $h(k) = h(i)$ and 0 otherwise.

- So,

$$\mathsf{E}\left[\left|T[h(i)] - f_i\right|\right] = \mathsf{E}\left[\left|\sum_{i \neq k} f_k x_k\right|\right] \leq \mathsf{E}\left[\sum_{i \neq k} |f_k| x_k\right]$$

$$= \sum_{i \neq k} \frac{f_k}{B} = \frac{F_1 - |f_i|}{B} \ .$$

- By Markov's inequality,

$$\Pr\left[\left|T[h(i)] - f_i\right| > \frac{4F_1}{B}\right] \leq \frac{1}{4} \ .$$

COUNT-MIN-Sketch: Point query analysis

- jth hash table gives good estimate $\equiv \left|T_j[h_j(i)] - f_i\right| > \frac{4F_1}{B}$ with prob. at least $3/4$.
- By classical boosting argument (Chernoff bound), using $w = O(\log \frac{1}{\delta})$ tables

$$\Pr\left[\left|\left(\text{median}_{j=1}^{w} T_j[h_j(i)]\right) - f_i\right| > \frac{4F_1}{B}\right] < \delta \ .$$

- Recall $\hat{f}_i = \text{median}_{j=1}^{w} T_j[h_j(i)]$.
- Therefore, $\left|\hat{f}_i - f_i\right| < \frac{4F_1}{B}$ with prob. $1 - \delta$.
- Solves point query problem with parameter ϕ for general frequency vectors. Choose $B = \frac{4}{\phi}$, $w = O\big(\log \frac{1}{\delta}\big)$.

COUNT-MIN-Sketch: Space and Update time

- Each counter uses $\log(mn)$ bits. There are Bw counters. So space is $O(\frac{1}{\phi}\log(\frac{1}{\delta}))$ counters.
- Each item is inserted in the right bucket in each table.
- Since h_j's are from pair-wise indep. family, $h_j(i)$ is computed in using $O(1)$ $+$ and $\cdot$ operations over a finite field of size $O(n)$.
- Time for inserting/updating a table is $O(1)$.
- Update time is $O(w) = O(\log(1/\delta))$.
- Number of random bits: $O(\log n)$ random bits per hash table.

Count-Min-Sketch: ℓ_1 ApproxHH

Simple (but inelegant) approach:

1. Estimate F_1 as $\hat{F}_1$ correct to within $1 \pm \frac{1}{8}$ with high prob.
2. Keep $B = \lceil \frac{12}{\phi} \rceil$ buckets per table, and $w = O(\log(n/\delta))$ buckets.
3. Iterate over domain $[n]$ and obtain $\hat{f}_i$ for each i.
4. Return i with $\hat{f}_i \geq \frac{2\phi}{3} \hat{F}_1$.

COUNT-MIN sketch: ApproxHH$_1^{\phi,\phi/3}$ by domain iteration

Assumption for simplicity: $\hat{F}_1 = F_1$. [1]

1. $\left|\hat{f}_i - f_i\right| \leq \frac{\phi F_1}{3}$ with prob. $1 - \frac{\delta}{n}$.
2. If $|f_i| \geq \phi F_1$, then,

$$\hat{f}_i \geq f_i - \frac{\phi F_1}{3} \geq \frac{2\phi F_1}{3} \text{ with prob. } 1 - \frac{\delta}{n}$$

3. If $|f_i| < \frac{\phi F_1}{3}$, then,

$$\hat{f}_i < f_i + \frac{\phi F_1}{3} \leq \frac{\phi F_1}{3} + \frac{\phi F_1}{3} = \frac{2\phi F_1}{3} \quad .$$

4. All n inferences hold jointly with prob. $\geq 1 - \frac{n\delta}{n} = 1 - \delta$.

[1] Otherwise, constants change a little.

COUNT-MIN sketch: ApproxHH$_1^{\phi,\phi'}$

- A possible design:
  1. $f_i > \phi F_1$ should imply
  $$\hat{f}_i > \frac{(\phi + \phi')F_1}{2} = \phi F_1 - \frac{(\phi - \phi')F_1}{2} \quad .$$
  2. $f_i < \phi' F_1$ should imply
  $$\hat{f}_i < \frac{(\phi + \phi')F_1}{2} = \phi' F_1 + \frac{(\phi - \phi')F_1}{2} \quad .$$
- Suffices to solve point query with parameter $\frac{\phi-\phi'}{2}$ with prob. $1 - \frac{\delta}{n}$.
- Space: $O(\frac{1}{\phi-\phi'}(\log \frac{n}{\delta}))$ counters.

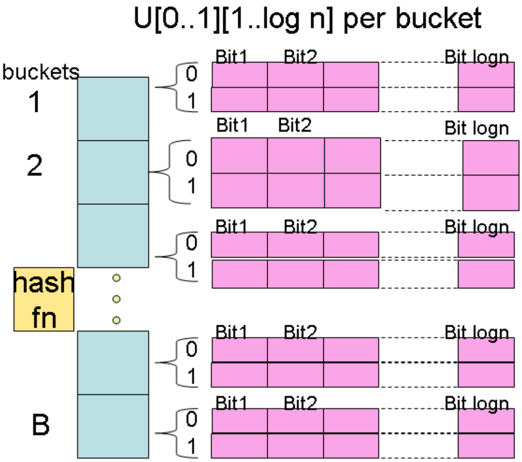
COUNT-MIN sketch: domain iteration

- Solves approximate $\text{ApproxHH}_1^{\phi,\phi/3}$ with prob. 1-δ.
- Resources used
  1. Space: $O\big(\frac{1}{\phi-\phi'}\log(\frac{n}{\delta})\big)$ counters.
  2. Update time: $O\big(\log(\frac{n}{\delta})\big)$.
  3. Discovery time of approximate HH: $O(n)$–the problem of this approach!
  4. Implicit: F_1 has to be estimated.

HH: COUNT-MIN sketch with Group testing

- Each bucket $T_j[b]$ now a $2 \times \log n$ array U.
- UPDATE(i, v):

for each table T_j {
    Go to bucket $h_j(i)$
    for each bit position l {
        if lth bit of x is 1 then
            increment $U[1][l]$ by v
        else
            increment $U[0][l]$ by v.
    }
}



Discovering Candidate Approximate HH: Basic Idea

- Heavy Hitter threshold ϕF_1.
- For simplicity, let F_1 be known exactly. [2]
- Use $U[2][\log n]$ array to find one possible candidate.
- Candidate Item: abs. value of frequency is at least $(\phi + \phi')F_1/2$.
- If multiple possible candidate, then no inference is made from this array.

[2] F_1 is estimated using within accuracy of $1 \pm 1/8$ with prob. $1 - \delta/2$. Replace F_1 by $\hat{F}_1$.

Phase I: Discovering Candidate Approx. HH

<u>Assume: Non-negative frequencies</u>

Ex. 1

	Bit1	Bit2		Bit4		Bit6
0	7	8	4	10	1	4
1	3	2	6	0	9	6

Majority Item: found

0	0	1	0	1	1

Ex. 2

	Bit1	Bit2		Bit4		Bit6
0	7	8	4	5	1	4
1	3	2	6	5	9	6

Majority Item: Amibguous

0	0	1	?	1	1

Candidate HH: an umambiguous majority item.

Analysis

- Suffices to show that every heavy hitter is a majority item in some table bucket, with high probability.
- Let i be a HH_1^{ϕ} item, so $|f_i| > \phi F_1$.
- Prior analysis: $\left| T_j[h_j(i)] - |f_i| \right| < \frac{4(F_1 - |f_i|)}{B}$ with prob. 3/4.
- So i is majority item in bucket $h_j(i)$ with prob. 3/4 if

$$\frac{4(F_1 - |f_i|)}{B} < \frac{\phi F_1}{2} \quad .$$

- Suffices if $B > \lceil \frac{8}{\phi} \rceil + 1$.
- Prob. that i is not a majority item in any of its buckets is at most $\frac{1}{4^w}$.

Width of majority structure

- Height of each hash table can be $\lceil \frac{8}{\phi} \rceil + 1$ or larger.
- Width w:
  1. Prob. that a heavy hitter is a majority item in some bucket is $1 - 4^{-w}$.
  2. There are at most $\lceil \frac{1}{\phi} \rceil$ heavy hitters.
  3. Prob. that each heavy hitter is a majority item in some bucket is $1 - \frac{4^{-w}}{\phi}$.
  4. To ensure that all heavy hitters are detected with a prob. of $1 - \delta$, set

$$\frac{4^{-w}}{\phi} < \delta \text{ or, } w > \frac{1}{2} \log \frac{1}{\phi\delta} \quad .$$

- Number of candidate heavy hitters at most $Bw = O\left(\frac{1}{\phi} \log \frac{1}{\phi\delta}\right)$.

Phase 2: Verifying Candidate Heavy Hitters

- Discovery phase gives at most $K = O\left(\frac{1}{\phi}\log\frac{1}{\phi\delta}\right)$ candidate heavy hitters.
- We now use a COUNT-MIN sketch structure to solve the $\text{PtQuery}_1(\phi, \phi')$ for each of the K candidate heavy hitters.
- Items that cross the threshold $\hat{f}_i \geq \frac{(\phi+\phi')F_1}{2}$ are returned, rest are rejected.

Resources Consumed: ApproxHH$_1^{\phi,\phi'}$

- Space required by PtQuery$_1(\phi, \phi')$ data structure is $O\left(\frac{1}{\phi-\phi'} \log \frac{K}{\delta}\right)$ counters.
- Discovering the heavy hitters requires time $O(K \log \frac{K}{\delta})$.
- Updating the structures requires time $O\left(\log \frac{1}{\phi\delta}\right)$.
- Final Guarantee: ApproxHH$_1^{\phi,\phi'}$ is solved with prob. $1 - 2\delta$.

Outline

Problem Definition and Introduction

ℓ_1 point query and heavy-hitters: COUNT-MIN sketch

- COUNT-MIN-sketch: ApproxHH$_1$ by domain iteration
- COUNT-MIN sketch: Approx HH by Group Testing

ℓ_2 Point Query and Heavy Hitters: the COUNTSKETCH structure

- COUNTSKETCH: Structure and Simple Analysis
- COUNTSKETCH and COUNT-MINsketch: Residual F_2 based bounds

Comparing ℓ_1 and ℓ_2 point query estimators

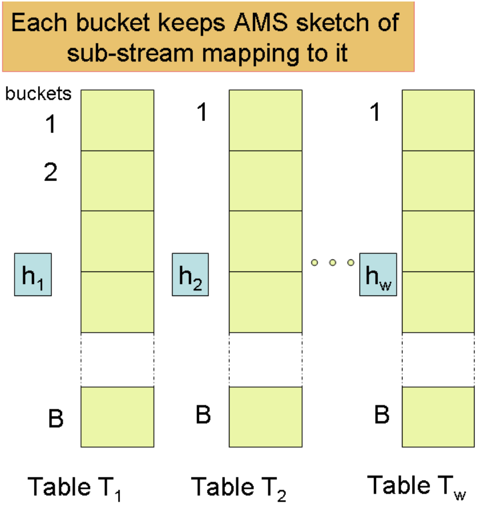
Sparse Approximation

ℓ_2 heavy hitters: the CountSketch structure

- CountSketch structure:
  1. w tables $T_1, \ldots, T_w$.
  2. $h_j : [n] \rightarrow [B]$ corresponding to T_j.
  3. h_j randomly chosen from a pair-wise indep. family.
  4. $h_1, \ldots, h_w$ are independently chosen.
  5. Sketch fn. $\xi_j : [n] \rightarrow \{-1, +1\}$ corresponding to T_j, 4-wise independent.
  6.

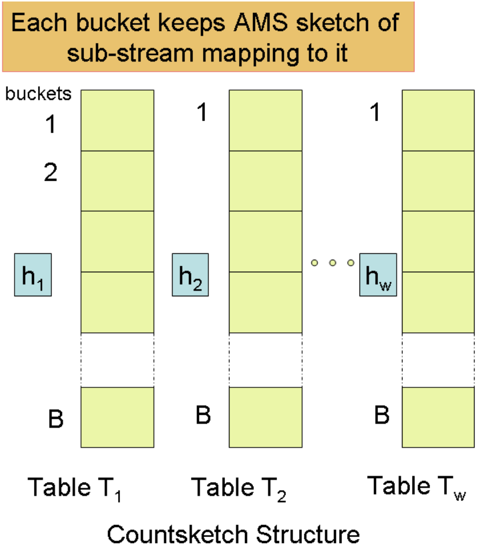
$$T_j[b] = \sum_{i : h_j(i) = b} f_i \xi_j(i)$$

$b = 1, \ldots, B$, $j = 1, 2, \ldots, w$.



Countsketch Structure

CountSketch structure



- Update(i, v):

  for $j = 1$ **to** w {
  $T_j[h_j(i)]$ **+=** $v \cdot \xi_j(i)$
  }

- Estimate(i):

  $\hat{f}_i = \text{median}_{j=1}^{w} T_j[h_j(i)] \cdot \xi_j(i)$

Frequency recovery: Basic idea

- Let $X = \sum_{k \in [n]} f_k \xi(k)$ be AMS sketch.
- Consider $X \cdot \xi(i)$.

$$X \cdot \xi(i) = f_i(\xi(i))^2 + \sum_{\substack{k \in [n] \\ k \neq i}} f_k \xi(k) \xi(i) \quad .$$

- By linearity of expect. and by pair-wise indep. of $\xi(\cdot)$, $\mathrm{E}[\xi(k)\xi(i)] = \mathrm{E}[\xi(k)]\,\mathrm{E}[\xi(i)] = 0 \cdot 0 = 0$, we have

$$\mathrm{E}[X \cdot \xi(i)] = f_i \quad .$$

- Variance calculation.

$$\mathrm{Var}\big[X \cdot \xi(i)\big] = \mathrm{E}\left[(X \cdot \xi(i))^2\right] - f_i^2 = F_2 - f_i^2 \quad .$$

CountSketch: Simple Analysis

- Useful fact: h and ξ use independent random bits.
- Fix table index j. Let $T = T_j$, $h = h_j$, $\xi_j = \xi$.
- Fix arbitrary item i.
- Let $X_i = T[h(i)]$ be sketch of sub-stream mapping to bucket i.
- Then, $\mathrm{E}_\xi\left[X_i \cdot \xi(i)\right] = f_i$, from previous slide. This is independent of h.
- And

$$\mathrm{Var}_\xi\left[X_i \cdot \xi(i)\right] = \sum_{\substack{k: h(k) = h(i) \\ k \neq i}} f_k^2$$

  which is a random function of h (and not of ξ).

CountSketch: Simple Analysis

- Define event LowVar(i) as

$$\mathrm{Var}_{\xi}\left[X_i \cdot \xi(i)\right] \leq 5\mathrm{Var}_{h,\xi}\left[X_i \cdot \xi(i)\right]$$

  or equivalently

$$\sum_{\substack{k:h(k)=h(i)\\ k\neq i}} f_k^2 \leq \frac{5(F_2 - f_i^2)}{B} \quad .$$

- By Markov's inequality, $\Pr_h\left[\textsc{LowVar}(i)\right] \geq 4/5$.

CountSketch: Simple Analysis

- $$\mathrm{E}_h\left[\mathrm{Var}_\xi\left[X_i \cdot \xi(i)\right]\right] = \mathrm{E}_h\Bigg[\sum_{\substack{k:h(k)=h(i)\\ k\neq i}} f_k^2\Bigg] = \sum_{k\neq i} f_k^2 \cdot \Pr\left[h(k)=h(i)\right] = \frac{F_2 - f_i^2}{B} \;.$$
- Also
$$\mathrm{Var}_{h,\xi}\left[X_i \cdot \xi(i) \mid \textsc{LowVar}(i)\right] = \mathrm{E}_h\left[\mathrm{Var}_\xi\left[X_i \cdot \xi(i)\right] \mid \textsc{LowVar}(i)\right] \le 5(F_2 - f_i^2)/B \;.$$

COUNTSKETCH :Simple Analysis

- By Chebychev's inequality

$$
\begin{aligned}
&\Pr_{h,\xi}\left[\left|X_i \cdot \xi(i) - f_i\right| > \left(\frac{25(F_2 - f_i)}{B}\right)^{1/2} \middle| \textsc{LowVar}(i)\right] \\
&\le \frac{\mathrm{Var}_{h,\xi}\left[X_i \cdot \xi(i) \mid \textsc{LowVar}(i)\right]}{25(F_2 - f_i^2)/B} \\
&\le \frac{5(F_2 - f_i^2)/B}{25(F_2 - f_i^2)/B} \\
&= \frac{1}{5} \ .
\end{aligned}
$$

Point query estimator

- Unconditioning w.r.t. event LOWVAR(i)

$$\Pr\left[\left|X_i \cdot \xi(i) - f_i\right| \le \left(\frac{25F_2}{B}\right)^{1/2}\right] \ge 1 - \frac{1}{5} - \frac{1}{5} = \frac{3}{5} \ .$$

- Let $X_{j,i} = T_j[h_j(i)]$.
- $\hat{f}_i = \text{median}_{j=1}^{w} X_{j,i} \cdot \xi_j(i)$.
- Then, by independence of h_j's and ξ_j's, by standard boosting argument we have

$$\Pr\left[\left|\hat{f}_i - f_i\right| \le \left(\frac{25F_2}{B}\right)^{1/2}\right] > 1 - \delta \ .$$

- PtQuery$_2(\phi)$: $\hat{f}_i \in f_i \pm (\phi F_2)^{1/2}$.
- COUNTSKETCH solves PtQuery$_2(\phi)$. Set $B = \lceil \frac{25}{\phi} \rceil$.

COUNTSKETCH Point Query Estimator

- Basic guarantee: Let $B = \lceil \frac{25}{\phi} \rceil$. Then, $\left|\hat{f}_i - f_i\right| < (\phi F_2)^{1/2}$ with prob. $1 - \delta$.
- Space required: $O\left(\frac{1}{\phi} \log \frac{1}{\delta}\right)$.
- Random bits: $2 \log n$ random bits for each h_j and $4 \log n$ random bits for each ξ_j. Total = $O(\log n \cdot \log \frac{1}{\delta})$.
- Update time: $O(\log \frac{1}{\delta})$.
- Solving point query: time $O(\log \frac{1}{\delta})$.

Point query : (slightly) better analysis

- Order items in non-increasing order of absolute frequencies

$$|f_{s_1}| \geq |f_{s_2}| \geq \ldots \geq |f_{s_n}| \ .$$

- A TOP-k item is one of $s_1, s_2, \ldots, s_k$, item with one of the top-k frequencies.
- Hash table T with *pairwise independent* hash fn h.
- Say that item i *collides* with item j if $h(i) = h(j)$ and $i \neq j$.
- Event NOCOLLISION(i) $\equiv$ i does not collide with any of the TOP-k items.
- Property: $\Pr[\text{NOCOLLISION}(i)] \geq 1 - k/B$.

No Collision with any Top-k items

- Fix item i.
- Let j be a fixed TOP-k item, $j \neq i$. Then,

$$\Pr[i \text{ collides with } j] = \Pr[h(i) = h(j)] = \frac{1}{B} \ .$$

- There are k TOP-k items. So by union bound

$$\Pr[\ i \text{ collides with one of the TOP-}k \text{ items}] \leq \frac{k}{B} \ .$$

- So, $\Pr[\textsc{NoCollision}(i)] \geq 1 - \frac{k}{B}$, for any fixed i.

Consequence of NoCollision(i)

- Leads to smaller error in inference made from bucket $h(i)$.
- Notion of residual moments.
- Define

$$F_p^{\mathrm{res}}(k) = \sum_{i=k+1}^{n} |f_{s_i}|^p \quad .$$

- So

$$F_1^{\mathrm{res}}(k) = \sum_{i=k+1}^{n} |f_{s_i}| \text{ and } F_2^{\mathrm{res}}(k) = \sum_{i=k+1}^{n} |f_{s_i}|^2 \quad .$$

COUNTSKETCH $\text{PtQuery}_2(\phi)$

- Condition analysis on LOWVAR(i) and NOCOLLISION(i).
- Set $B = 8k$. Then, $\Pr[\text{NOCOLLISION}(i)] \geq \frac{7}{8}$, $k = \lceil 1/\phi \rceil$.
- From previous calculation

$$\text{Var}_\xi\left[T[h(i)] \cdot \xi(i)\right] = \sum_{k:h(k)=h(i),k\neq i} f_k^2 \ .$$

- Assume NOCOLLISION(i) holds. Then,

$$\text{Var}_\xi\left[T[h(i)] \cdot \xi(i)\right] = \sum_{\substack{k:h(k)=h(i),k\neq i \\ k\notin \text{TOP}-\text{k}}} f_k^2 \ .$$

- So

$$\mathsf{E}_h\left[\text{Var}_\xi\left[T[h(i)] \cdot \xi(i)\right] \,\middle|\, \text{NOCOLLISION}(i)\right] \leq \frac{F_2^{\text{res}}(k)}{B} \ .$$

Analysis contd.

- By Markov's inequality

$$\Pr_h\left[\mathrm{Var}_\xi\left[T[h(i)]\cdot\xi(i)\right] > \frac{8F_2^{\mathrm{res}}}{B}\,\middle|\,\textsc{NoCollision}(i)\right] \le \frac{1}{8} \;.$$

- Change constant in defn of $\textsc{LowVar}(i)$:

$$\textsc{LowVar}(i) \equiv \mathrm{Var}_\xi\left[T[h(i)]\cdot\xi(i)\right] < \frac{8F_2^{\mathrm{res}}(k)}{B} \;.$$

Analysis

By Chebychev's inequality,

$$\Pr_{h,\xi}\left\{\left|T[h(i)]\cdot\xi(i)-f_i\right| > \left(\frac{64F_2^{\text{res}}(k)}{B}\right)^{1/2}\middle|\textsc{NoCollision}(i)\wedge\textsc{LowVar}(i)\right\}$$

$$\leq$$

$$\frac{\operatorname{Var}_{h,\xi}\left[\left|T[h(i)]\cdot\xi(i)\right| \;\middle|\; \textsc{NoCollision}(i)\wedge\textsc{LowVar}(i)\right]}{64F_2^{\text{res}}(k)/B}$$

$$\leq$$

$$\frac{8F_2^{\text{res}}(k)/B}{64F_2^{\text{res}}(k)/B}=\frac{1}{8}\quad .$$

Analysis contd.

Already shown (assuming $B = 8k$)

$$\Pr_{h,\xi}\left\{ \left| T[h(i)] \cdot \xi(i) - f_i \right| \le \left(\frac{64 F_2^{\mathrm{res}}(k)}{8k} \right)^{1/2} \middle| \textsc{NoCollision}(i) \wedge \textsc{LowVar}(i) \right\} \ge \frac{7}{8} \ .$$

$$\Pr\left[\textsc{NoCollision}(i)\right] \ge \frac{7}{8}, \quad \Pr\left[\textsc{LowVar}(i) \mid \textsc{NoCollision}(i)\right] \ge \frac{7}{8}$$

- From basic probability:

$$\Pr[E] = \Pr[E \mid A \wedge B] \cdot \Pr[B|A] \cdot \Pr[A] \quad .$$

- Therefore,

$$\Pr_{h,\xi}\left\{ \left| T[h(i)] \cdot \xi(i) - f_i \right| > \left(\frac{64 F_2^{\mathrm{res}}(k)}{8k} \right)^{1/2} \right\} \ge \frac{7}{8} \cdot \frac{7}{8} \cdot \frac{7}{8} > 2/3 \ .$$

Point Query: Detailed Analysis

- We have $\Pr_{h,\xi}\left\{ \left| T[h(i)] \cdot \xi(i) - f_i \right| > \left(\frac{8F_2^{\text{res}}(k)}{k} \right)^{1/2} \right\} \geq \frac{2}{3}$.
- Keep $w = O(\log \frac{1}{\delta})$ independent tables $T_1, T_2, \ldots, T_w$.
- Let $\hat{f}_i = \text{median}_{j=1}^{w} T_j[h_j(i)] \cdot \xi_j(i)$.
- Then,

$$\Pr\left\{ \left| \hat{f}_i - f_i \right| > \left(\frac{8F_2^{\text{res}}(k)}{k} \right)^{1/2} \right\} \geq 1 - \delta \ .$$

- No change in algorithm, Space increases by a factor of 2, update time does not change. Only analysis is stronger.

COUNT-MIN Sketch: Detailed Analysis

- Same idea can be applied. Choose $B = 4k$.
- Final guarantee obtained as

$$\Pr\left\{ \left|\hat{f}_i - f_i\right| > \left(\frac{F_1^{\text{res}}(k)}{4k}\right)\right\} \geq 1 - \delta \;\; .$$

- No change in point query estimator. Space increases by factor of 2. No change in update time.

Outline

Which is better?

- We have shown estimators for PtQuery_1^{ϕ} and PtQuery_2^{ϕ} as follows.

$$\text{PtQuery}_1 : \quad |\hat{f}_i - f_i| \le \frac{F_1^{\text{res}}(k)}{k}$$
$$\text{PtQuery}_2 : \quad |\hat{f}_i - f_i| \le \left(\frac{F_2^{\text{res}}(k)}{k}\right)^{1/2} \quad .$$

- Which is more accurate (more than just constant factors)?

Comparing different norms: Two results

- $$\left(\frac{F_2^{\text{res}}(k)}{k}\right)^{1/2} \leq \frac{F_1}{k} \quad .$$

  Many examples where, $F_1^{\text{res}}(k) \approx F_1$.

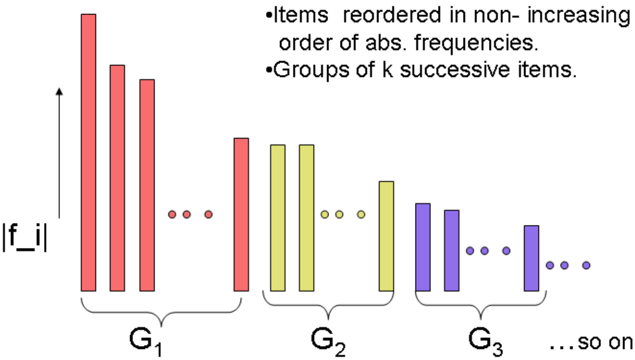
- Follows from inequality:

  $$F_q^{\text{res}}(k) \leq \frac{F_p^{q/p}}{k^{q/p-1}}, \quad q \geq p \quad .$$

- A version of Holder's inequality:

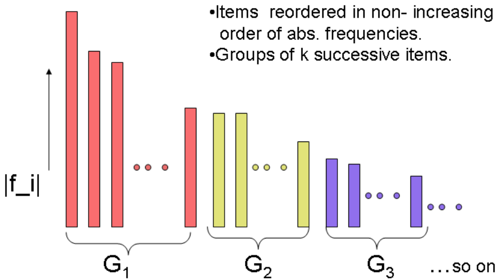
  $$\left(\frac{F_p}{F_0}\right)^{1/p} \leq \left(\frac{F_q}{F_0}\right)^{1/q}, \quad q \geq p \quad .$$

Proof of inequality 1



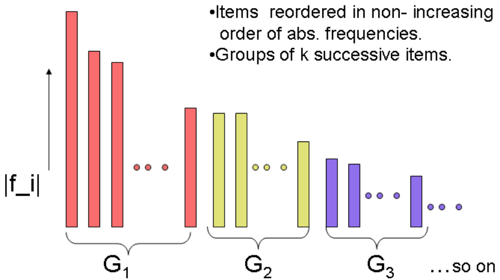
- Arrange items 1 to n in non-increasing order of absolute values of frequency: $|f_{s_1}| \geq |f_{s_2}| \geq \ldots \geq |f_{s_n}|$.
- Items are grouped k at a time (in non-increasing order), $G_1, G_2, \ldots, G_{\lceil n/k \rceil}$ groups.
- Last group may have less than k items.

Proof of CS inequality



$$\sum_{k+1}^{n}|f_i|^q = \sum_{l=2}^{\lceil n/k \rceil}\sum_{i\in G_l}\left(|f_{s_i}|^p\right)^{q/p} \le \sum_{l=2}^{\lceil n/k \rceil}\sum_{i\in G_l}\left(\text{ avg. } |f_j^p| \text{ of } G_{l-1}\right)^{q/p}$$

Proof of CS inequality

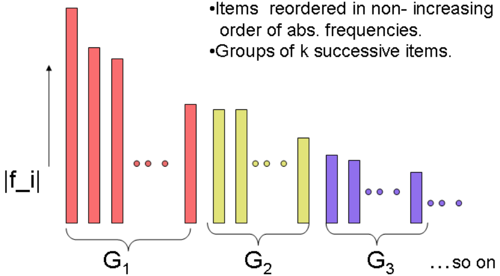


$$\sum_{k+1}^{n} |f_i|^q = \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(|f_{s_i}|^p\right)^{q/p} \leq \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(\text{avg. } |f_j^p| \text{ of } G_{l-1} \right)^{q/p}$$

$$\leq \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(\frac{1}{k} \sum_{k \in G_{l-1}} f_k^p \right)^{q/p} \leq \sum_{l=2}^{\lceil n/k \rceil} \frac{1}{k^{q/p-1}} \left(\sum_{k \in G_{l-1}} f_k^p \right)^{q/p}$$

Proof of CS inequality

- Items reordered in non- increasing order of abs. frequencies.
- Groups of k successive items.

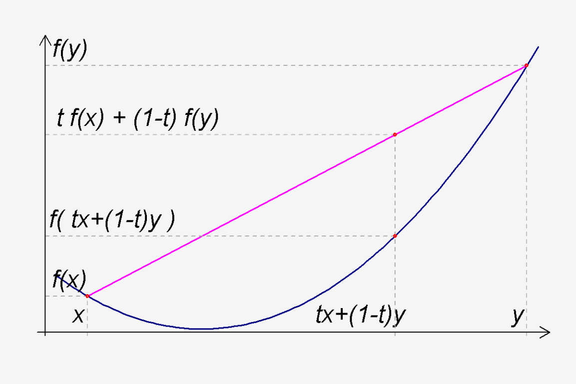


$$\sum_{k+1}^{n} |f_i|^q = \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(|f_{s_i}|^p\right)^{q/p} \le \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(\text{avg. } |f_j^p| \text{ of } G_{l-1} \right)^{q/p}$$

$$\le \sum_{l=2}^{\lceil n/k \rceil} \sum_{i \in G_l} \left(\frac{1}{k} \sum_{k \in G_{l-1}} f_k^p \right)^{q/p} \le \sum_{l=2}^{\lceil n/k \rceil} \frac{1}{k^{q/p-1}} \left(\sum_{k \in G_{l-1}} f_k^p \right)^{q/p}$$

$$\le \frac{1}{k^{q/p-1}} \sum_{l=1}^{\lceil n/k \rceil} \left(\sum_{k \in G_l} f_k^p \right)^{q/p} \le \frac{1}{k^{q/p-1}} F_p^{q/p}, \quad \text{since, } q \ge p$$

Jensen's inequality and convex functions [3]



- A function ϕ is convex in an interval I of $\mathbb{R}$ if for any $x_1, x_2 \in I$ and $0 \leq t \leq 1$,

$$\phi(tx_1 + (1-t)x_2) \leq t\phi(x_1) + (1-t)\phi(x_2), \quad t \in [0, 1] \ .$$

- Convex function $\phi(X)$ of a random variable X. ϕ is convex in an interval that contains the support of X.

[3] Figure: Source Wikipedia-Convex Functions

Jensen's inequality

- Let ϕ be a convex function of a random variable X. Then, $\mathrm{E}[\phi(X)] \geq \phi(\mathrm{E}[X])$.
- Proof:

$$\begin{aligned}
\mathrm{E}[\phi(X)] &= \sum_{x \in \mathrm{Supp}(X)} \Pr[X = x]\, \phi(x) \\
&\geq \phi\Bigg(\sum_{x \in \mathrm{Supp}(X)} \Pr[X = x]\Bigg), \quad \text{by convexity of } \phi \\
&= \phi\big(\mathrm{E}[X]\big) \quad .
\end{aligned}$$

Application of Jensen's inequality

- Let f be an n-dimensional vector. [4]
- Random experiment: choose a random element $i \in_R [n]$. Let $X = |f_i|^p$.
- $\mathsf{E}[X] = \frac{F_p}{n}$.
- Consider $\phi(X) = X^{q/p}$, where, $q \geq p$. ϕ is convex.
- By Jensen's inequality,

$$\begin{aligned} \mathsf{E}\left[X^{q/p}\right] &\geq \left(\mathsf{E}[X]\right)^{q/p} \\ \text{or, } \frac{F_q}{n} &\geq \left(\frac{F_p}{n}\right)^{q/p} \\ \text{or, } \left(\frac{F_q}{n}\right)^{1/q} &\geq \left(\frac{F_p}{n}\right)^{q/p} \quad . \end{aligned}$$

[4] n can be replaced by F_0.

Outline

Sparse Approximation: Basic Idea

- x: n-dimensional frequency vector.
- Wish to summarize x using another vector x' that has "low complexity" k.

  $$\|x - x'\|_p \leq (1+\alpha)\mathrm{Err}_p^k$$

  where,

  $$\mathrm{Err}_p^k = \min_{x'' \text{ with complexity } k} \|x - x''\|_p$$

  and α is a parameter.
- Notion of complexity: $\|\cdot\|_0$ or number of non-zero coordinates . That is, find x' s.t.

  $$\|x - x'\|_p \leq (1+\alpha) \min_{\|x''\|_0 \leq k} \|x - x''\|_p \quad .$$

Definition

- A vector is said to be k-sparse if it has at most k non-zero coordinates, i.e., $\|x\|_0 \leq k$.
- ℓ_1-sparse approximation problem: Find x' such that

$$\|x - x'\|_1 \leq (1+\phi) \min_{x'' \text{ } k\text{-sparse}} \|x - x''\|_1 \ .$$

- ℓ_2-sparse approximation problem: Find x' such that

$$\|x - x'\|_2 \leq (1+\phi) \min_{x'' \text{ } k\text{-sparse}} \|x - x''\|_2 \ .$$

ℓ_1 sparse approximation

- Min value of $\|x - x''\|_p$ for k-sparse x is attained for x^* set to the top-k values of x (and 0's elsewhere).
- Let $x_{s_1} \geq x_{s_2} \geq \ldots \geq x_{s_n}$.
- So
$$\min_{x'' \ k\text{-sparse}} \|x - x''\|_p = \sum_{j=k+1}^{n} |x_{s_j}|^p, \quad p \geq 0.$$
- From PtQuery$_1$ estimator: $|\hat{x}_i - x_i| \leq \frac{\sum_{j=k+1}^{n} |x_{s_j}|}{4k}$ with prob. $1 - 2^{-\Omega(w)}$.
- ℓ_1-sparse approximation: $x' =$ top-k of the $|\hat{x}|_i$'s.

ℓ_1 Sparse Approximation

- x' : top-k by estimated values, x^*: actual top-k values.
- Arrange x in descending order: $|x_{s_1}| \geq |x_{s_2}| \ldots \geq |x_{s_k}|$.
- Keep PtQuery$_1\left(\lceil \frac{\phi}{k} \rceil\right)$ data structure. So,

$$|x_i^* - x_i| \leq \frac{\phi}{4k} \sum_{j=\lfloor k/\phi \rfloor} |x_{s_j}| \quad .$$

- Arrange x' in descending order: $|\hat{x}|_{t_1} \geq |\hat{x}|_{t_2} \ldots \geq |\hat{x}|_{t_k}$. So,

$$\|x^* - x'\| \leq \sum_{j=1}^{k} |x_{s_j} - \hat{x}_{t_j}| \leq k \cdot \frac{\phi}{4k} \sum_{j=\lfloor k/\phi \rfloor} |x_{s_j}| \leq \frac{\phi}{4} \|x - x^*\|$$

ℓ_1 sparse approximation

- Min value of $\|x - x''\|_p$ for k-sparse x is attained for x^* set to the top-k values of x (and 0's elsewhere).
- Let $x_{s_1} \geq x_{s_2} \geq \ldots \geq x_{s_n}$.
- So
$$\min_{x''\ k\text{-sparse}} \|x - x''\|_p = \sum_{j=k+1}^{n} |x_{s_j}|^p, \quad p \geq 0.$$
- From PtQuery$_1$ estimator: $|\hat{x}_i - x_i| \leq \frac{\sum_{j=k+1}^{n} |x_{s_j}|}{4k}$ with prob. $1 - 2^{-\Omega(w)}$.
- ℓ_1-sparse approximation: $x' =$ top-k of the $|\hat{x}_i|$'s.

Estimating $F_1^{\text{res}}(k)$

- Use $\text{PtQuery}_1(\phi/k)$.
- Obtain top-k estimated frequencies: $\hat{x}_{t_1} \geq \hat{x}_{t_2} \geq \ldots \geq \hat{x}_{t_k}$. Denote vector as x'.
- Skimming: Remove contribution of estimated frequencies.

$$X = X - \sum_{i=1}^{k} \hat{x}_{t_1} \xi_{t_1}, \quad \text{for all sketches } X \ .$$

- Apply to Cauchy sketches.
- After skimming, effective frequency is $\bar{x}_i$:

$$X = \sum_{i \in [n]} \bar{x}_i \xi(i) \ .$$

$$\bar{x}_i = \begin{cases} x_i & \text{if } i \text{ is not among top-}k \text{ estimated freq.} \\ x_i - \hat{x}_i & \text{otherwise.} \end{cases}$$

Estimating residual ℓ_1 norm

- $$\begin{aligned} \|\bar{x}\|_1 &= \sum_{j=1}^{k} |x_{t_j} - \hat{x}_{t_j}| + \|x - x'\|_1 \\ &\le \frac{\phi}{4}\|x - x^*\|_1 + (1 + \frac{\phi}{4})\|x - x^*\|_1 \text{ from previous slide} \\ &\le \left(1 + \frac{\phi}{2}\right)\|x - x^*\| \ . \end{aligned}$$

- Guarantee is probabilistic: above result holds with high probability for any x.
  Note: Argument is a variation on Chandan Saha's original argument [CS719 course: 2004 Sem-I].