Sketching (1)

Alex Andoni
(Columbia University)

MADALGO Summer School on Streaming Algorithms 2015

131.107.65.14

Challenge: log statistics of the data, using *small* space

0.1.12

131.107.65.14

80.97.56.20

18.0.1.12

80.97.56.20

131.107.65.14

IP	Frequency
131.107.65.14	3
18.0.1.12	2
80.97.56.20	2
127.0.0.1	9
192.168.0.1	8
257.2.5.7	0
16.09.20.11	1

Streaming statistics

- Let x_i = frequency of IP i

IP	Frequency
131.107.65.14	3
18.0.1.12	2
80.97.56.20	2

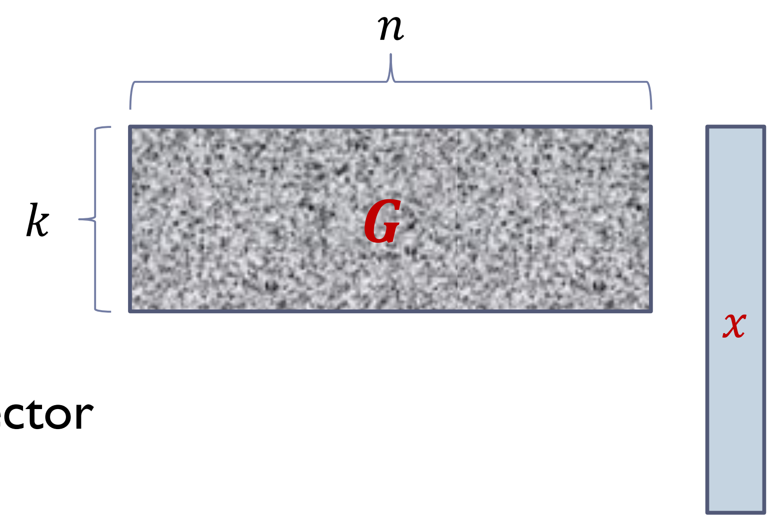
- 1st moment (sum): $\sum x_i$
  - Trivial: keep a total counter
- 2nd moment (variance): $\sum x_i^2 = ||x||^2$
  - Trivially: n counters → too much space
    - Can't do better
  - Better with small approximation!
    - Via dimension reduction in ℓ_2

$\sum x_i = 7$

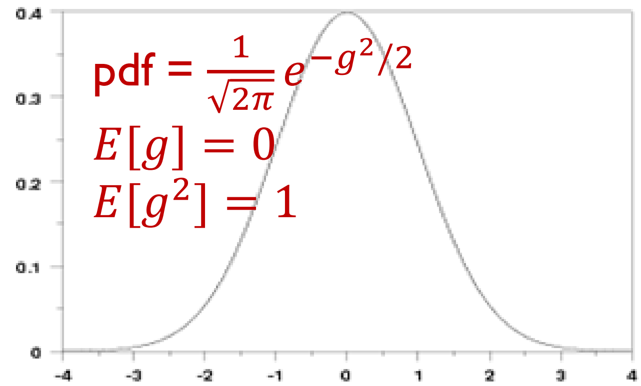
$\sum x_i^2 = 17$

2^{nd} frequency moment

- Let x_i = frequency of IP i
- 2^{nd} moment: $\sum x_i^2 = ||x||^2$
- Dimension reduction
  - Store a sketch of x
    - $S(x) = (G_1 x, G_2 x, \ldots G_k x) = \boldsymbol{G}x$
    - each G_i is n-dimensional Gaussian vector
  - Estimator:
    - $\frac{1}{k}||\boldsymbol{G}x||^2 = \frac{1}{k}\left((G_1 x)^2 + (G_2 x)^2 + \cdots + (G_k x)^2\right)$
  - Updating the sketch:
    - Use linearity of the sketching function S
    - $\boldsymbol{G}(x + e_i) = \boldsymbol{G}x + \boldsymbol{G}e_i$

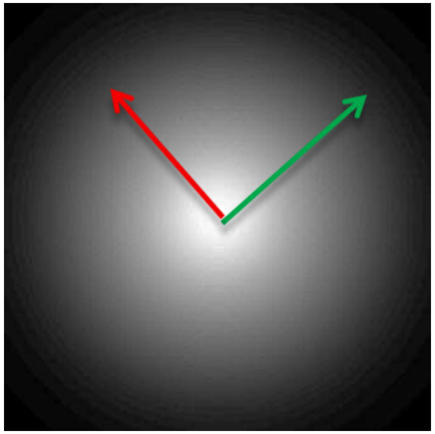


Correctness

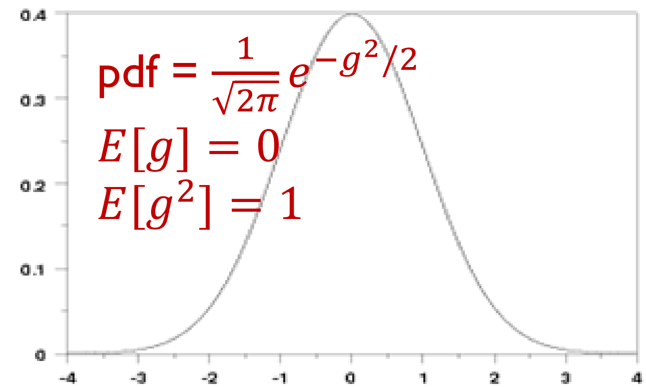


- Theorem [Johnson-Lindenstrauss]:
  - $||\boldsymbol{G}x||^2 = (1 \pm \epsilon)||x||^2$ with probability $1 - e^{-O(k\epsilon^2)}$
- Why Gaussian?
  - Stability property: $G_i x = \sum_j G_{ij} x_j$ is distributed as $||x|| \cdot g$, where g is also Gaussian
  - Equivalently: G_i is centrally distributed, i.e., has random direction, and projection on random direction depends only on length of x

$$P(a) \cdot P(b) =$$
$$= \frac{1}{\sqrt{2\pi}} e^{-a^2/2} \frac{1}{\sqrt{2\pi}} e^{-b^2/2}$$
$$= \frac{1}{2\pi} e^{-(a^2+b^2)/2}$$



Proof [sketch]



- Claim: for any $x \in \Re^n$, we have
  - Expectation: $\mathrm{E}[|G_i \cdot x|^2] = \|x\|^2$
  - Standard deviation: $\sigma[|G_i x|^2] = O(\|x\|^2)$
- Proof:
  - Expectation $= \mathrm{E}[(G_i \cdot x)^2] = \mathrm{E}[\|x\|^2 \cdot g^2]$
    $= \|x\|^2$
- $\boldsymbol{G}x$ is distributed as
  - $\frac{1}{\sqrt{k}}(||x|| \cdot g_1, \ldots, ||x|| \cdot g_k)$
  - where each g_i is distributed as 1D Gaussian
- Estimator: $||Gx||^2 = ||x||^2 \cdot \frac{1}{k} \sum_i g_i^2$
  - $\sum_i g_i^2$ is called chi-squared distribution with k degrees
- Fact: chi-squared very well concentrated:
  - Equal to $1 + \epsilon$ with probability $1 - e^{-\Omega(\epsilon^2 k)}$
  - Akin to central limit theorem

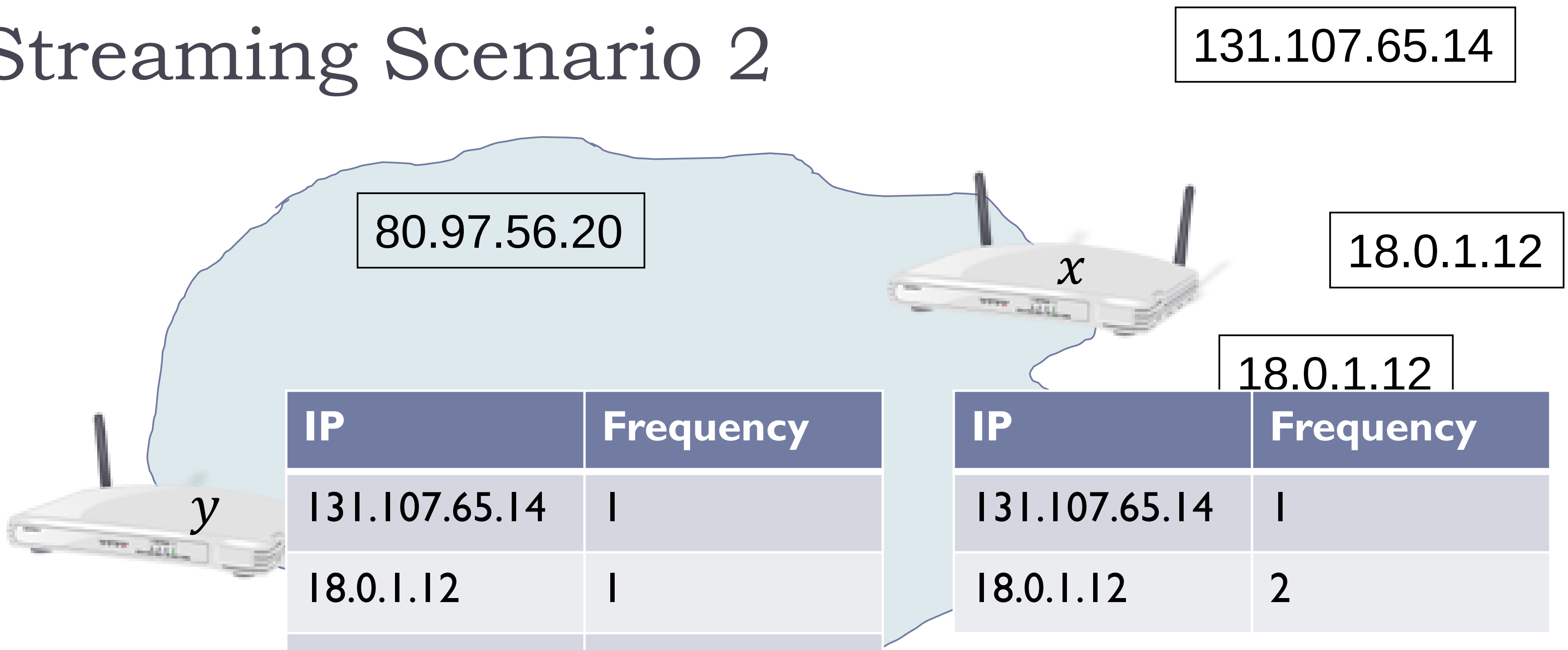
2^{nd} frequency moment: overall

- Correctness:
  - $||\boldsymbol{G}x||^2 = (1 \pm \epsilon)||x||^2$ with probability $1 - e^{-O(k\epsilon^2)}$
  - Enough to set $k = O(1/\epsilon^2)$ for const probability of success
- Space requirement:
  - $k = O(1/\epsilon^2)$ counters of $O(\log n)$ bits
  - What about $\boldsymbol{G}$: store $O(nk)$ reals ?
- Storing randomness [AMS'96]
  - Ok if g_i "less random": choose each of them as 4-wise independent
  - Also, ok if g_i is a random ± 1
  - Only $O(k)$ counters of $O(\log n)$ bits

More efficient sketches?

- Smaller Space:
  - No: $\Omega\left(\frac{1}{\epsilon^2}\log n\right)$ bits [JW'11] ← David's lecture
- Faster update time:
  - Yes: Jelani's lecture

Streaming Scenario 2



IP	Frequency
131.107.65.14	1
18.0.1.12	1
80.97.56.20	1

IP	Frequency
131.107.65.14	1
18.0.1.12	2

Focus: *difference* in traffic

1st moment: $\sum |x_i - y_i| = \|x - y\|_1$ $\qquad \|x - y\|_1 = 2$

2nd moment: $\sum |x_i - y_i|^2 = \|x - y\|_2^2$ $\qquad \|x - y\|_2^2 = 2$

Similar Qs: average delay/variance in a network

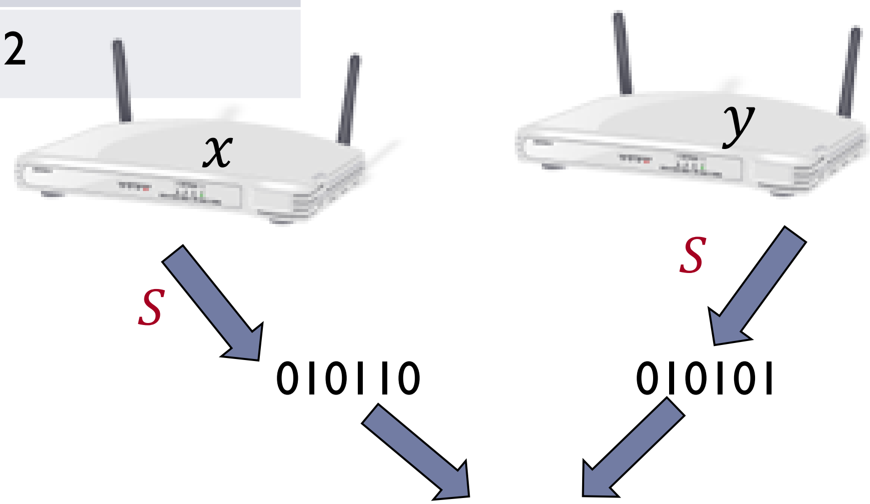
differential statistics between logs at different servers, etc

Definition: Sketching

- Sketching:
  - S : objects → short bit-strings
  - given $S(x)$ and $S(y)$, should be able to estimate some function of x *and* y

IP	Frequency
131.107.65.14	1
18.0.1.12	2

IP	Frequency
131.107.65.14	1
18.0.1.12	1
80.97.56.20	1



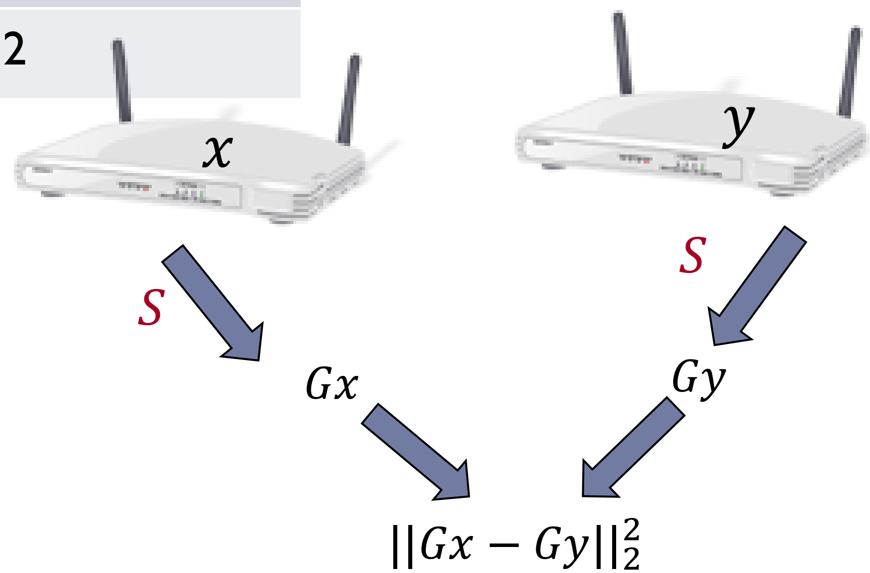
Estimate $\|x - y\|_2^2$?

Sketching for ℓ_2

- As before, dimension reduction
  - Pick $\boldsymbol{G}$ (using common randomness)
  - $S(x) = \boldsymbol{G}x$
- Estimator: $||S(x) - S(y)||_2^2 = ||\boldsymbol{G}(x - y)||_2^2$

IP	Frequency
131.107.65.14	1
18.0.1.12	2

IP	Frequency
131.107.65.14	1
18.0.1.12	1
80.97.56.20	1



Sketching for Manhattan distance (ℓ_1)

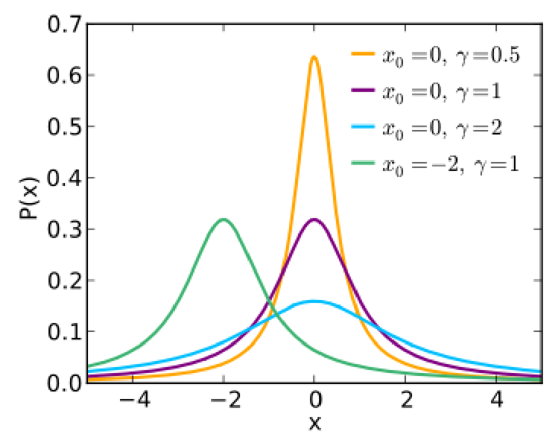
- Dimension reduction?
  - Essentially no: [CS'02, BC'03, LN'04, JN'10…]
  - For n points, D approximation: between $n^{\Omega(1/D^2)}$ and $O(n/D)$ [BC03, NR10, ANN10…]
    - even if map depends on the dataset!
  - In contrast: [JL] gives $O(\epsilon^{-2}\log n)$
  - No distributional dimension reduction either

  - *Weak* dimension reduction is the rescue…

Dimension reduction for ℓ_1 ?

- Can we do the "analog" of Euclidean projections?
- For ℓ_2, we used: Gaussian distribution
  - has stability property:
  - $g_1 z_1 + g_2 z_2 + \cdots g_d z_d$ is distributed as $g \cdot ||z||$
- Is there something similar for 1-norm?
  - Yes: Cauchy distribution!
  - 1-stable:
  - $c_1 z_1 + c_2 z_2 + \cdots c_d z_d$ is distributed as $c \cdot ||z||_1$
- What's wrong then?
  - Cauchy are **heavy-tailed…**
  - doesn't even have finite expectation (of abs)

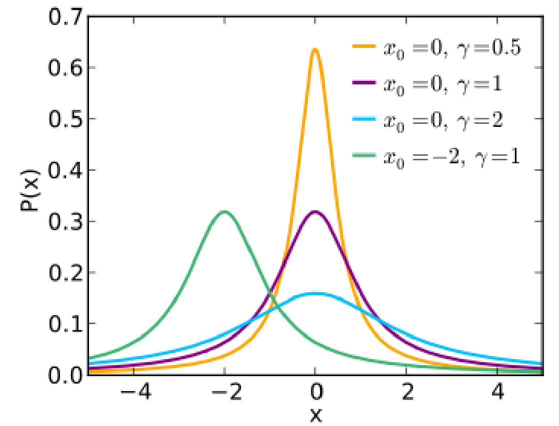
$$pdf(s) = \frac{1}{\pi(s^2 + 1)}$$



Sketching for ℓ_1 [Indyk'00]

- Still, can consider map as before
  - $S(x) = (C_1 x, C_2 x, \dots, C_k x) = \boldsymbol{C}x$
- Consider $S(x) - S(y) = \boldsymbol{C}x - \boldsymbol{C}y = \boldsymbol{C}(x - y) = \boldsymbol{C}z$
  - where $z = x - y$
  - each coordinate distributed as $||z||_1$ ×Cauchy
  - Take 1-norm $||\boldsymbol{C}z||_1$?
    - does not have finite expectation, but…
- Can estimate $||z||_1$ by:
  - *Median* of absolute values of coordinates of $\boldsymbol{C}z$!
- Correctness claim: for each i
  - $\Pr[|C_i z| > ||z||_1 \cdot (1 - \epsilon)] > 1/2 + \Omega(\epsilon)$
  - $\Pr[|C_i z| < ||z||_1 \cdot (1 + \epsilon)] > 1/2 + \Omega(\epsilon)$

Estimator for ℓ_1



- Estimator: median$(|C_1 z|, |C_2 z|, \ldots |C_k z|)$
- Correctness claim: for each i
  - $\Pr[|C_i z| > ||z||_1 \cdot (1-\epsilon)] > 1/2 + \Omega(\epsilon)$
  - $\Pr[|C_i z| < ||z||_1 \cdot (1+\epsilon)] > 1/2 + \Omega(\epsilon)$
- Proof:
  - $|C_i z| = abs(C_i z)$ is distributed as $\mathrm{abs}(||z||_1 c) = ||z||_1 \cdot |c|$
  - Easy to verify that
    - $\Pr[|c| > (1-\epsilon)] > 1/2 + \Omega(\epsilon)$
    - $\Pr[|c| < (1+\epsilon)] > 1/2 + \Omega(\epsilon)$
- Hence, if we have $k = O(1/\epsilon^2)$
  - median$(|C_1 z|, |C_2 z|, \ldots |C_k z|) \in (1 \pm \epsilon)||z||_1$

    with probability at least 90%

To finish the ℓ_p norms…

- p-moment: $\Sigma x_i^p = \|x\|_p^p$
- $p \leq 2$
  - works via p-stable distributions [Indyk'00]
- $p > 2$
  - Can do (and need) $\tilde{O}(n^{1-2/p})$ counters

    [AMS'96, SS'02, BYJKS'02, CKS'03, IW'05, BGKS'06, BO10, AKO'11, G'11, BKSV'14]
  - Will see a construction via Precision Sampling

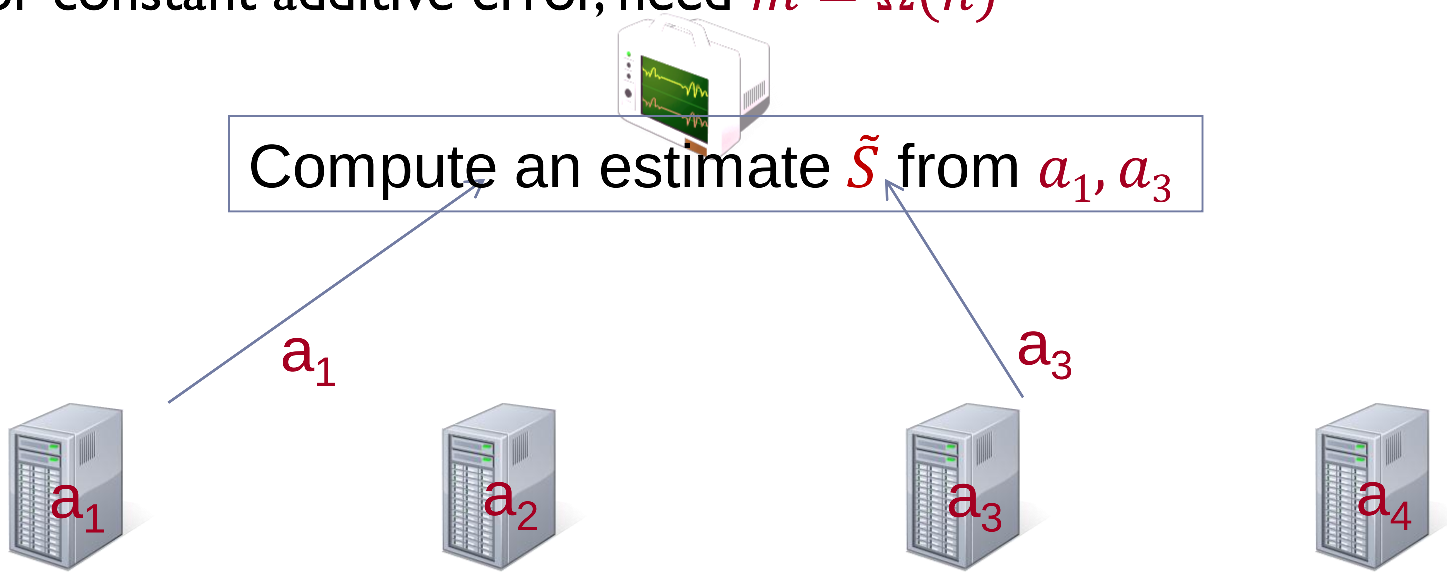
A task: estimate sum

- Given: n quantities $a_1, a_2, \dots a_n$ in the range $[0,1]$
- Goal: estimate $S = a_1 + a_2 + \cdots a_n$ "cheaply"

- Standard sampling: pick random set $J = \{j_1, \dots j_m\}$ of size m
  - Estimator: $\tilde{S} = \frac{n}{m} \cdot (a_{j_1} + a_{j_2} + \cdots a_{j_m})$
- Chebyshev bound: with 90% success probability

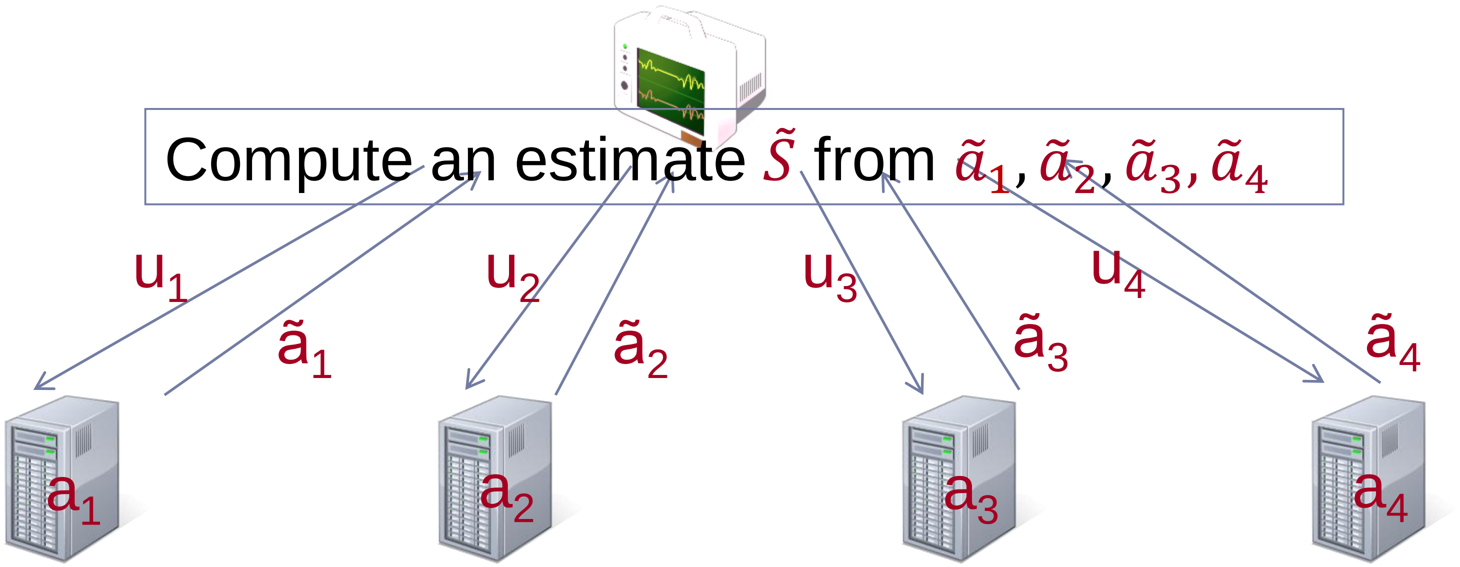
$$\frac{1}{2} S - O(n/m) < \tilde{S} < 2S + O(n/m)$$

- For constant additive error, need $m = \Omega(n)$



Precision Sampling Framework

- Alternative "access" to a_i's:
  - For each term a_i, we get a (rough) estimate $\tilde{a}_i$
  - up to some *precision* u_i, chosen in advance: $|a_i - \tilde{a}_i| < u_i$
- Challenge: achieve good trade-off between
  - quality of approximation to S
  - use only weak precisions u_i (minimize "cost" of estimating $\tilde{a}$)

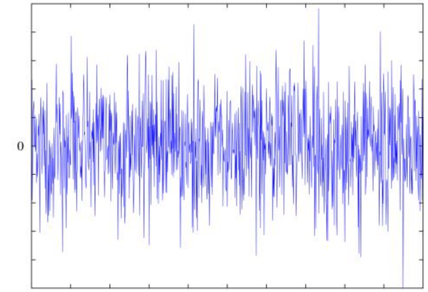


Formalization

Sum Estimator 

1. fix precisions u_i

3. given $\tilde{a}_1, \tilde{a}_2, \ldots \tilde{a}_n$, output $\tilde{S}$ s.t. $\left|\sum_i a_i - \gamma\tilde{S}\right| < 1$ (for some small γ)

Adversary 

1. fix $a_1, a_2, \ldots a_n$

2. fix $\tilde{a}_1, \tilde{a}_2, \ldots \tilde{a}_n$ s.t. $|a_i - \tilde{a}_i| < u_i$

- What is cost?
  - Here, average cost = $1/n \cdot \sum 1/u_i$
  - to achieve precision u_i, use $1/u_i$ "resources": e.g., if a_i is itself a sum $a_i = \sum_j a_{ij}$ computed by subsampling, then one needs $\Theta(1/u_i)$ samples
- For example, can choose all $u_i = 1/n$
  - Average cost $\approx n$

Precision Sampling Lemma

[A-Krauthgamer-Onak'11]

- Goal: estimate $\sum a_i$ from $\{\tilde{a}_i\}$ satisfying $|a_i - \tilde{a}_i| < u_i$.
- Precision Sampling Lemma: can get, with 90% success:
  - ϵ additive error a $1+\epsilon$ multiplicative error:
    $S - \epsilon < \tilde{S} < (1+\epsilon)S + \epsilon$
  - with average cost equal to $O(\epsilon^{-3} \log n)$

- Example: distinguish $\Sigma a_i = 3$ vs $\Sigma a_i = 0$
  - Consider two extreme cases:
    - if three $a_i = 1$: enough to have crude approx for all $(u_i = 0.1)$
      if all $a_i = 3/n$: only few with good approx $u_i = 1/n$, and the rest with $u_i = 1$

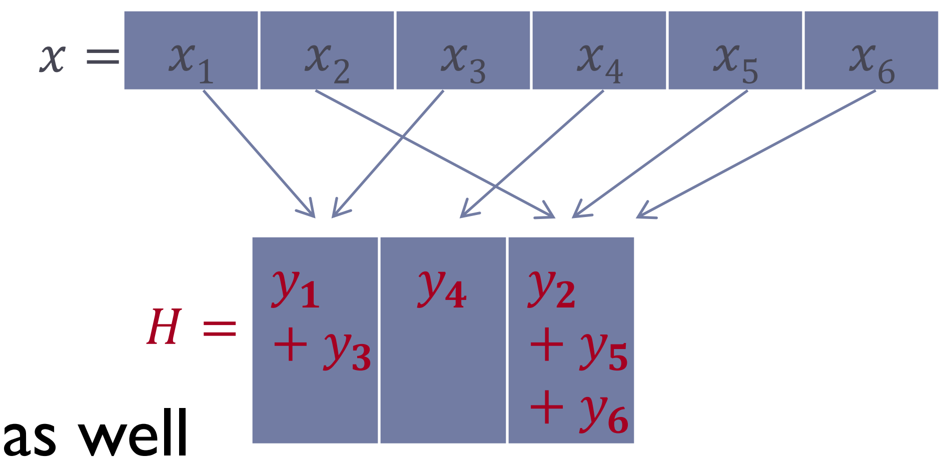
Precision Sampling Algorithm

- Precision Sampling Lemma: can get, with 90% success:
  - ϵ additive error and $1+\epsilon$ ultiplicative error:
    $S-\epsilon < \tilde{S} < (1+\epsilon)\cdot S + O(1)$
  - with average cost equal to $O(\epsilon^{-3}\log n)$
- Algorithm:
  - Choose each $u_i \in$ concrete distrib. = minimum of $O(\epsilon^{-3})$ u.r.v.
  - Estimator: $\tilde{S} =$ function of $[\tilde{a}_i/u_i - 4/\epsilon]^+$ and u_i's normalization constant)
- Proof of correctness:
  - we use only $\tilde{a}_i$ which are 1.5-approximation to a_i
  - $E[\tilde{S}] \approx \sum \Pr[a_i / u_i > 6] = \sum a_i/6$.
  - $E[1/u_i] = O(\log n)$ w.h.p.

ℓ_p via precision sampling

- **Theorem:** linear sketch for ℓ_p with $O(1)$ approximation, and $O(n^{1-2/p}\log n)$ space (90% succ. prob.).
- Sketch:
  - Pick random $r_i \in \{\pm 1\}$, and u_i as exponential r.v.
  - let $y_i = x_i \cdot r_i / u_i^{1/p}$
  - throw into one hash table H,
  - $k = O(n^{1-2/p}\log n)$ cells
- Estimator:
  - $\max_c |H[c]|^p$
- Linear: works for difference as well
- Randomness: bounded independence suffices

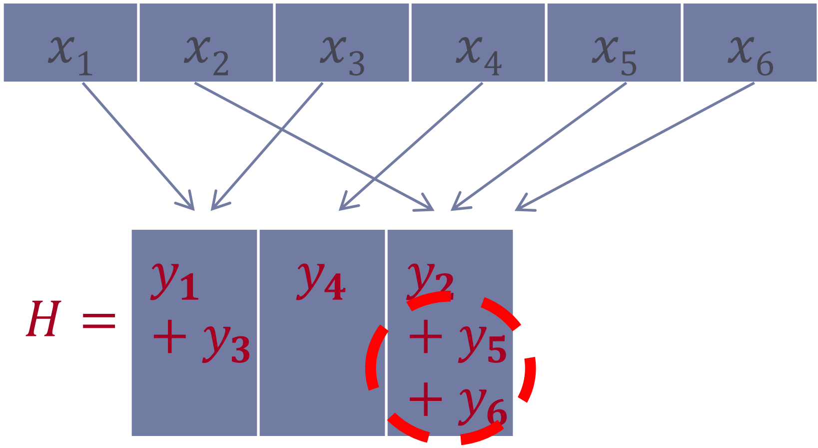
$\boxed{u \sim e^{-u}}$



Correctness of ℓ_p estimation

- Sketch:
  - $y_i = x_i \cdot r_i / u_i^{1/p}$ where $r_i \in \{\pm 1\}$, and u_i exponential r.v.
  - Throw into hash table H
- **Theorem:** $\max_c |H[c]|^p$ is $O(1)$ approximation with 90% probability, for $k = O(n^{1-2/p} \log^{O(1)} n)$ cells
- Claim 1: $\max_i |y_i|$ is a const approx to $||x||_p$
  - $\max_i |y_i|^p = \max_i |x_i|^p / u_i$
  - **Fact [max-stability]:** $\max \lambda_i / u_i$ distributed as $\sum \lambda_i / u$
  - $\max_i |y_i|^p$ is distributed as $||x||_p^p / u$
  - u is $\Theta(1)$ with const probability

Correctness (cont)



- Claim 2:
  - $\max_c |H[c]| = \Theta(1) \cdot ||x||_p$
- Consider a hash table H, and the cell c where y_{i^*} falls into
  - for i^* which maximizes $|y_{i^*}|$
- How much "extra stuff" is there?
  - $\delta^2 = (H[c] - y_{i^*})^2 = (\sum_{j \neq i^*} y_j \cdot \chi[j \to c])^2$
  - $E[\delta^2] = \sum_{j \neq i^*} y_j^2 \cdot \chi[j \to c] = \sum_{j \neq i^*} y_j^2 / k \leq ||y||^2 / k$
  - We have: $E_u ||y||^2 \leq ||x||^2 \cdot E\left[1/u^{1/p}\right] = O(\log n) \cdot ||x||^2$
  - $||x||^2 \leq n^{1-2/p} ||x||_p^2$
  - By Markov's: $\delta^2 \leq ||x||_p^2 \cdot \boxed{n^{1-2/p} \cdot O(\log n)/k}$ with prob 0.9.
  - Then: $H[c] = y_{i^*} + \delta = \Theta(1) \cdot ||x||_p$.
- Need to argue about *other* cells too → concentration

$y_i = x_i \cdot r_i / u_i^{1/p}$
where $r_i \in \{\pm 1\}$
u_i exponential r.v.