Adversarially Robust Dense-Sparse Tradeoffs via Heavy-Hitters

David P. Woodruff, Carnegie Mellon University and Google Research
Samson Zhou, Texas A&M University

NEURAL INFORMATION PROCESSING SYSTEMS

Motivation

- Algorithms are often evaluated with the assumption that the input is independent of the parameters of the algorithm
- Adversary may exploit these parameters to generate adversarial inputs
- Multiple interactions with the algorithm may cause future inputs to depend on previous outputs (and thus internal parameters of the algorithm)

Model

- Input: Elements of an underlying data set S, which arrives sequentially and *adversarially*
- Output: Evaluation (or approximation) of a given function
- Goal: Use space *sublinear* in the size m of the input S

Frequency Moments

- Given a set S of m elements from $[n]$, let x_i be the frequency of element i. (How often it appears)
- Let F_p be the frequency moment of the vector:

$$F_p(x) = x_1^p + x_2^p + \cdots + x_n^p$$

- Goal: Given a set S of m elements from $[n]$, output an approximation to $F_p(x)$
- Motivation: Entropy estimation, linear regression

Heavy-Hitters

- Given a set S of m elements from $[n]$, let x_i be the frequency of element i.
- Let L_2 be the norm of the frequency vector:

$$L_2(x) = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$$

- Goal: Given a set S of m elements from $[n]$ and a threshold ε, output the elements i such that $x_i > \varepsilon\, L_2(x)$...and no elements j such that $x_j < \frac{\varepsilon}{16}\, L_2(x)$
- Motivation: DDoS prevention, iceberg queries

Insertion-Only Streams [BJWY20] [HKM+20] [WZ21]

Distinct Elements	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^3}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^4 n}{\varepsilon^{2.5}}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^2}\right)$
F_p Estimation, $p \in (0,2]$	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^3}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^4 n}{\varepsilon^{2.5}}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^2}\right)$
Shannon Entropy	$\tilde{\mathcal{O}}\left(\frac{\log^6 n}{\varepsilon^5}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^4 n}{\varepsilon^{3.5}}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^3 n}{\varepsilon^2}\right)$
L_2-Heavy Hitters	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^3}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^4 n}{\varepsilon^{2.5}}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log n}{\varepsilon^2}\right)$
F_p Estimation, integer $p > 2$	$\tilde{\mathcal{O}}\left(\frac{n^{1-2/p}}{\varepsilon^3}\right)$	$\tilde{\mathcal{O}}\left(\frac{n^{1-2/p}}{\varepsilon^{2.5}}\right)$	$\tilde{\mathcal{O}}\left(\frac{n^{1-2/p}}{\varepsilon^2}\right)$
F_p Estimation, $p \in (0,2]$, flip number λ	$\tilde{\mathcal{O}}\left(\frac{\lambda \log^2 n}{\varepsilon^2}\right)$	$\tilde{\mathcal{O}}\left(\frac{\log^3 n \sqrt{\lambda \log n}}{\varepsilon^2}\right)$	$\tilde{\mathcal{O}}\left(\frac{\lambda \log^2 n}{\varepsilon}\right)$

Insertion-Deletion Streams

- Each update $u_t = (a_t, \Delta_t)$ can increase or decrease a coordinate $a_t \in [n]$ of the underlying frequency vector $x \in \mathbb{R}^n$ by $\Delta_t \in \mathbb{Z}$
- For simplicity, we assume $\Delta_t \in \{-1, +1\}$
- In the robust setting, each update u_t can be chosen adversarially
- $\tilde{O}\left(m^{p/(2p+1)}\right)$ space algorithm for F_p estimation, where m is the length of the stream [BEO22]
- Nothing known for constant-factor approximation in space polynomial in n

Dense-Sparse Tradeoffs

- [BEO22] observes that the value of the function can change by $(1+\varepsilon)$-multiplicative factor a lot, but only if the value of the function is SMALL
- If the function has SMALL value, it must be somewhat sparse, can use sparse recovery to identify the frequency vector
- Leads to good balancing to handle cases where value of the function changes by $(1+\varepsilon)$-multiplicative factor

https://github.com/samsonzhou/WZ24

Our Results

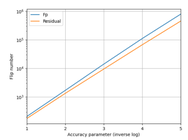
- Let $p \in [1,2]$. Then there exists an adversarially robust algorithm that solves the ε- L_p-heavy hitters problem on turnstile streams, using $\tilde{O}\left(\frac{1}{\varepsilon^{2.5}} m^{\frac{2p-2}{4p-3}}\right)$ bits of space
- Let $p \in [1,2]$ and $c = \frac{24p^2-23p+4}{(4p-3)(12p+3)}$. Then there exists an adversarially robust algorithm that outputs a constant-factor approximation to F_p-estimation on turnstile streams, using $\tilde{O}(m^c)$ bits of space
- Streaming algorithm for estimating the frequency moment of the tail vector, which achieves additive error and uses space independent in the size of the tail

Techniques

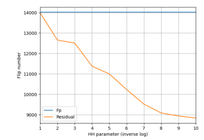
- Observation: if the function has LARGE value, it takes more updates to change the value of the function by $(1+\varepsilon)$-multiplicative factor if the updates are to different coordinates
- We can capture the case where the updates are to the same coordinates through heavy-hitter algorithms
- However, these heavy-hitter algorithms may themselves fail
- Deterministic heavy-hitter algorithm for turnstile streams that uses $\tilde{O}\left(\frac{1}{\varepsilon^2} n^{2-2/p}\right)$ bits of space for $p \in [1,2]$ [GM07]
- Leads to *better* balancing for analyzing cases where value of the function changes by $(1+\varepsilon)$-multiplicative factor

Experiments

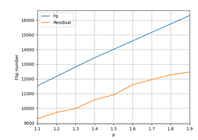
- CAIDA traffic monitoring dataset, with anonymized passive traffic traces from the "equinix-nyc" data center's high-speed monitor
- Extracted sender IP addresses from 12 minutes of the internet flow data, containing roughly 3 million total events
- Compared flip number vs. flip number of residual vector



(a) Flip number across $-\log_{10} \varepsilon$



(b) Flip number across $-\log_4 \alpha$



(c) Flip number across p

References

[BJWY22]: Omri Ben-Eliezer, Rajesh Jayaram, David P. Woodruff, and Eylon Yogev. A framework for adversarially robust streaming algorithms. J. ACM,, 2022
[BEO22]: Omri Ben-Eliezer, Talya Eden, and Krzysztof Onak. Adversarially robust streaming via dense-sparse trade-offs. SOSA, 2022
[HKM+20]: Avinatan Hassidim, Haim Kaplan, Yishay Mansour, Yossi Matias, and Uri Stemmer. Adversarially robust streaming algorithms via differential privacy. NeurIPS, 2020
[WZ21]: David P. Woodruff and Samson Zhou. Tight bounds for adversarially robust streams and sliding windows via difference estimators. FOCS 2021