Sample-and-Threshold Differential Privacy: Histograms and Applications

Akash Bharadwaj, Graham Cormode (Meta AI)
{akashb, gcormode}@fb.com

Motivation

Federated Analytics (FA) emphasises distributed computation of statistics in a privacy-preserving way.

Releasing histograms is a building block for many FA tasks, including quantiles and heavy hitters.

Our goal is to gather data from a distributed set of clients and achieve a centralized differential privacy (DP) guarantee.

The protocol should minimize communication, and minimize the work of the server to obtain the private results.

It should be a practical building block for other applications.

Background and Applications

Histogram release with DP has been heavily studied, via:

- Noise addition in the central model e.g. [1]
- Randomized response in the local model e.g. [3]
- Distributed noise addition in the shuffle model [2]

We show that sampling itself provides a DP histogram mechanism, similar to the work of [4] on heavy hitters.

Heavy-hitters: Two histogram approaches to heavy hitters:

- Hierarchical search with growing histograms, as in [4]
- Direct histogram materialization at leaf level

Quantiles: Two histogram approaches to quantiles:

- Interactive (binary) search for target quantile
- Materialize hierarchical histograms for offline search

All approaches lead to (ϵ, δ)-DP and accuracy guarantees.

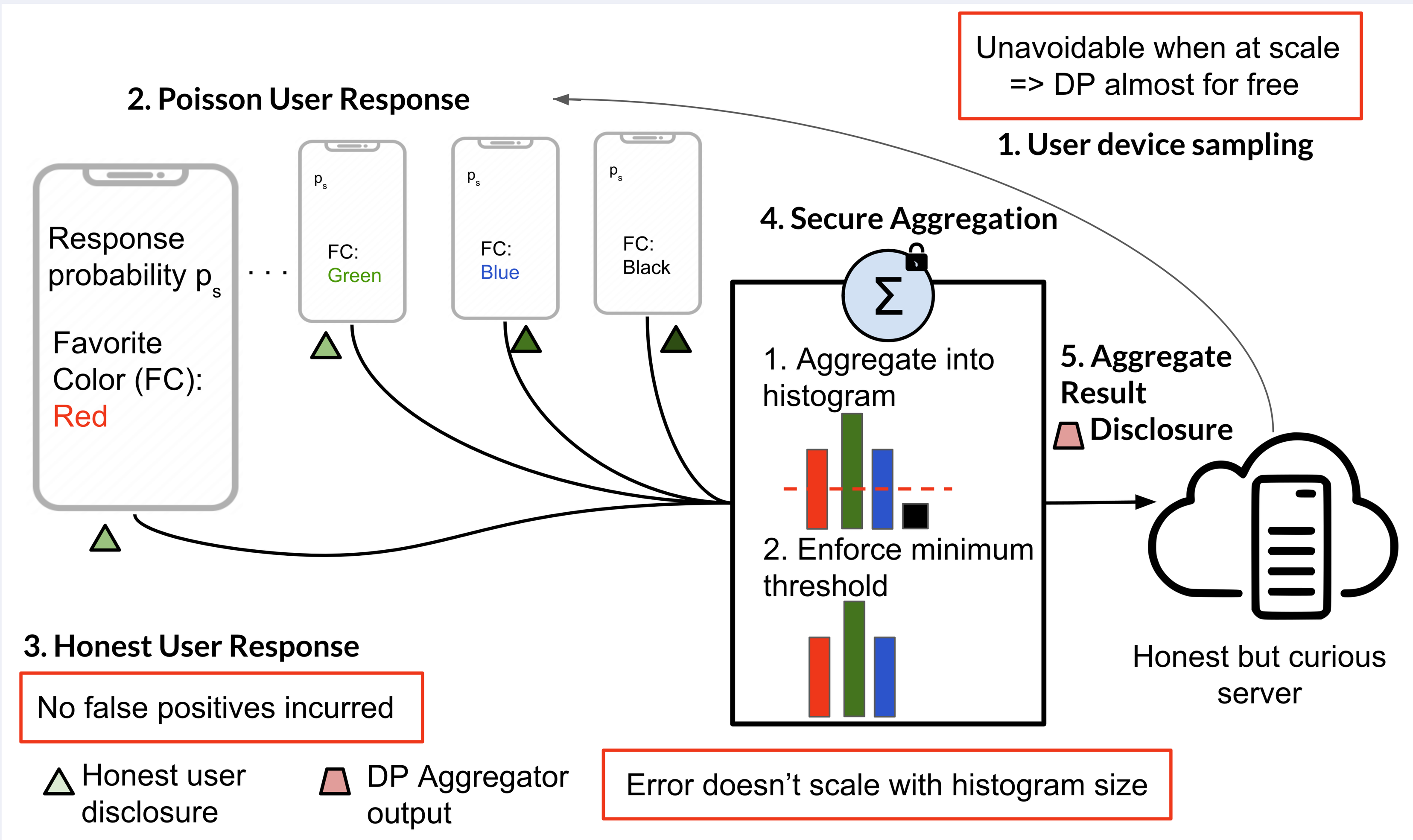
[1] C. Dwork. Differential privacy. In *ICALP*, 2006.

[2] Ú. Erlingsson, V. Feldman, I. Mironov, A. Raghunathan, S. Song, K. Talwar, and A. Thakurta. Encode, shuffle, analyze privacy revisited: Formalizations and empirical evaluation. *CoRR*, abs/2001.03618, 2020.

[3] T. Wang, J. Blocki, N. Li, and S. Jha. Locally differentially private protocols for frequency estimation. In *USENIX Security*, 2017.

[4] W. Zhu, P. Kairouz, B. McMahan, H. Sun, and W. Li. Federated heavy hitters discovery with differential privacy. In *AISTATS*, 2020.

Sample-and-Threshold Histograms



Histogram Protocol for n clients, each holding one item:

— Sample each client with probability $p_s = \alpha(1 - \exp(-\epsilon))$
— Sampled clients report their item truthfully to the server
— The server reports only those items with at least τ reports

Privacy guarantee: the output is (ϵ, δ)-DP, for $\delta = \exp(-\tau O(\ln(1/\alpha)))$.

Output also achieves a k-anonymity property for $k = \tau$.

Intuition: sampling introduces Binomial noise on the counts.

After thresholding, it is hard to tell the difference between inputs containing k or $k+1$ copies of an item.

Sample rate = 0.1 → Threshold value 6

Original data				→	Sampled data				→	Thresholded sample			
99	67	11	2	→	8	7	1	0	→	8	7	0	0

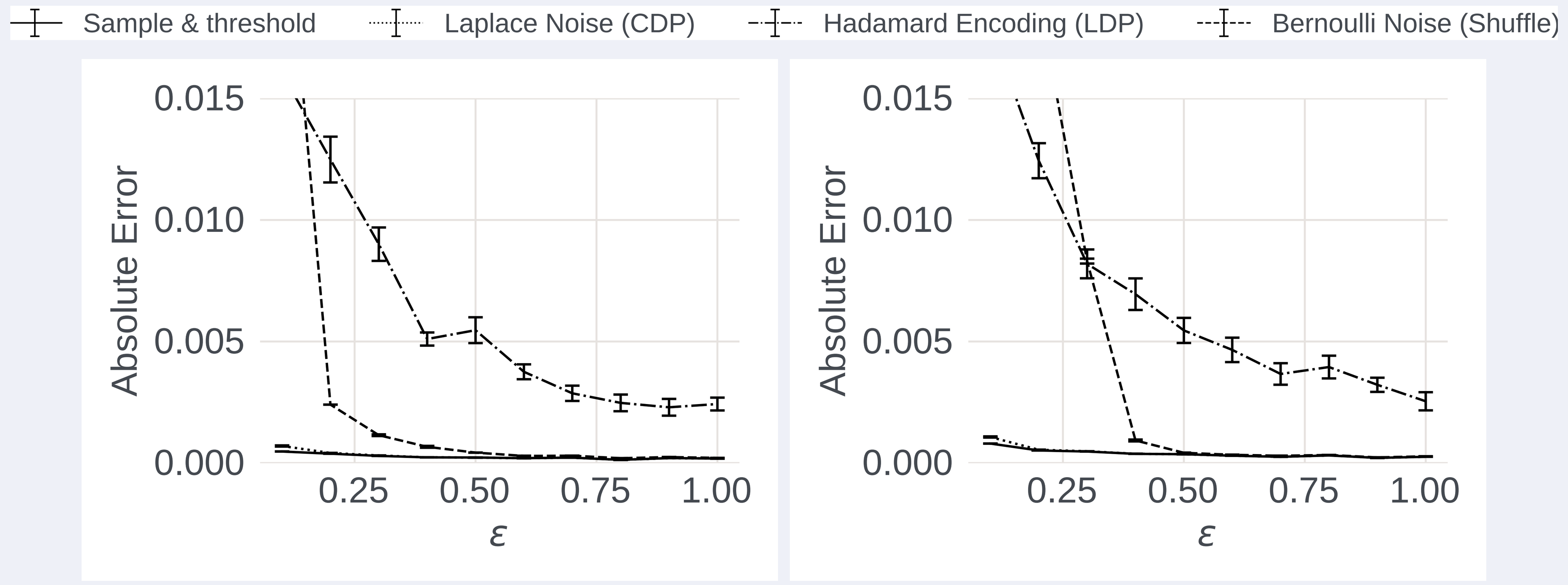
Here, $p_s = 0.1$ and $\tau = 6$ giving $(\epsilon = 1, \delta = 0.0015)$-DP.

Accuracy guarantees are proved with Chernoff bounds.

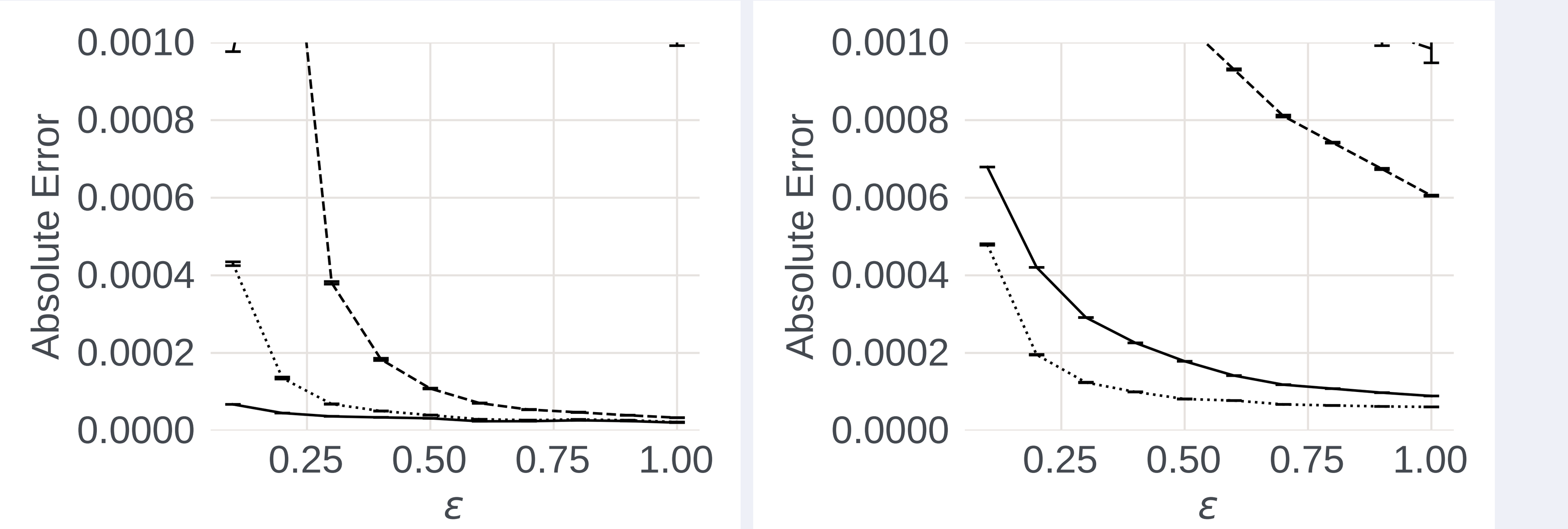
- Probability of not reporting a heavy item decreases exponentially with its expected frequency above τ
- Items are reported with relative error when their frequency is high enough

Results

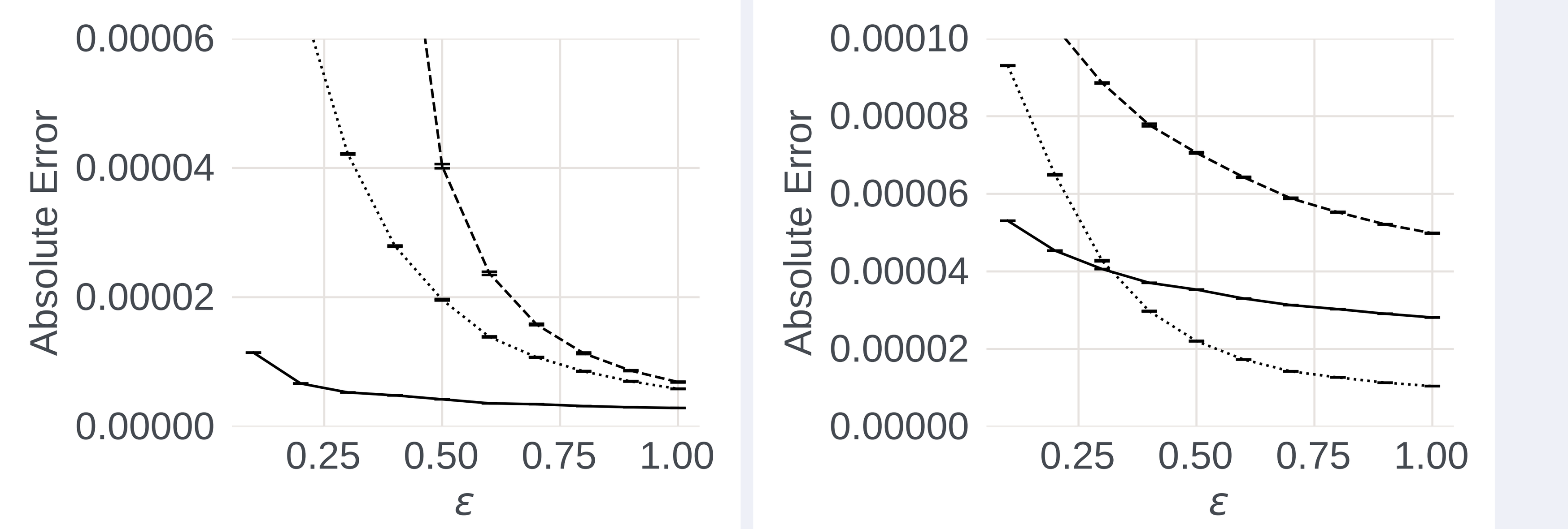
Experiments measure absolute frequency error on real and synthetic data, varying the histogram size (D) from 2^6 to 2^{14}.



For small domain size $D = 2^6$ and Binomial (left) and Shakespeare (right) data, sample-and-threshold has similar or better error than central noise.



For medium domain size ($D = 2^{10}$), sample-and-threshold lags central noise, but improves over randomized response (local DP) and Bernoulli noise (shuffle).



Over large domains ($D = 2^{14}$) and small ϵ, sample-and-threshold is preferred. Errors are due to missing small counts from long-tail items.