Bit-efficient Numerical Aggregation and Stronger Privacy for Trust in Federated Analytics

Graham Cormode, Igor Markov (Facebook)

Motivation

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

We study the basic question of mean and variance estimation while minimizing data sharing.

Our solution emphasises bit-efficiency: sending as little as 1 bit per client.

This allows privacy metering at the *bit level*.

Background

Prior work typically assumes inputs in $[0, 1]$.

Subtractive dithering: samples a random threshold and reports whether the client value is $>$ or $\leq$ [1].

Piece-wise mechanism: an optimized LDP method for reporting fractions [2].

Our work more *adaptively* locates the mean in the range. In practice, this improves efficiency when only loose bounds are known.

Our simulations with several thousand clients confirm the trend and give high accuracy.

References

[1] R. Ben-Basat, M. Mitzenmacher, and S. Vargaftik. How to send a real number using a single bit (and some shared randomness). *CoRR*, abs/2010.02331, 2020.

[2] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu. Collecting and analyzing multidimensional data with local differential privacy. In *IEEE ICDE*, pages 638–649. IEEE, 2019.

Contact: {gcormode, imarkov}@fb.com

Bit-pushing algorithms

Each client i out of N hold a b-bit integer value x_i.

Seek mean $\bar{x} = \sum_{i=1}^{N} \frac{x_i}{N}$, variance $\sigma^2 = \sum_{i=1}^{N} \frac{(x_i - \bar{x})^2}{N}$

We write $x^{(j)}$ for the j'th bit of x.

Basic bit-pushing algorithm.

With probability p_j, client i sends $x^{(j)}$ to the server.

Server computes $X^{(j)}$ as the mean of all reports of bit j, and estimate of $\bar{x}$ as $X = \sum_{j=1}^{b} 2^j X^{(j)}$.

Picking $p_j \propto 2^{\alpha j}$ minimizes the variance as $O(2^b \bar{x}/N)$ when $\alpha = 1$.

Adaptive Bit-pushing.

We use a first round of bit-pushing to estimate bit means, and choose p_j for round two based on them.

Adaptive bit-pushing improves the variance to $O(b\sigma^2/N)$ plus lower-order terms.

Local Differential Privacy.

We can apply *randomized response* to each client report to ensure Local Differential Privacy.

The variance is now $O(\frac{4^b}{\epsilon^2 N})$ for ϵ-LDP, independent of the data distribution.

Variance Estimation.

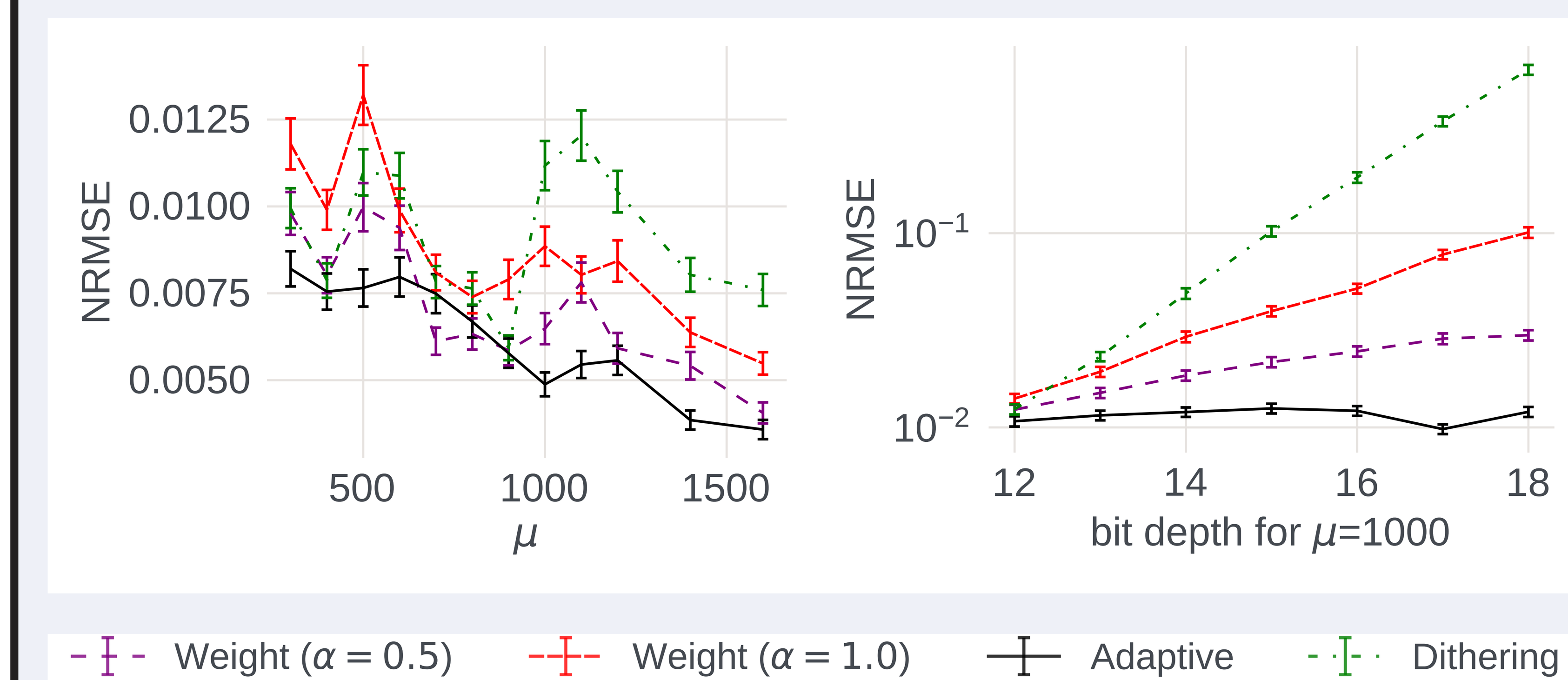
We use a first phase of bit-pushing to estimate $\bar{x}$, then a second round is applied to $(x_i - \bar{x})^2$.

The variance (of the estimated variance) is proportional to $(\sigma^2 + \bar{x}^2/N)^2/N$

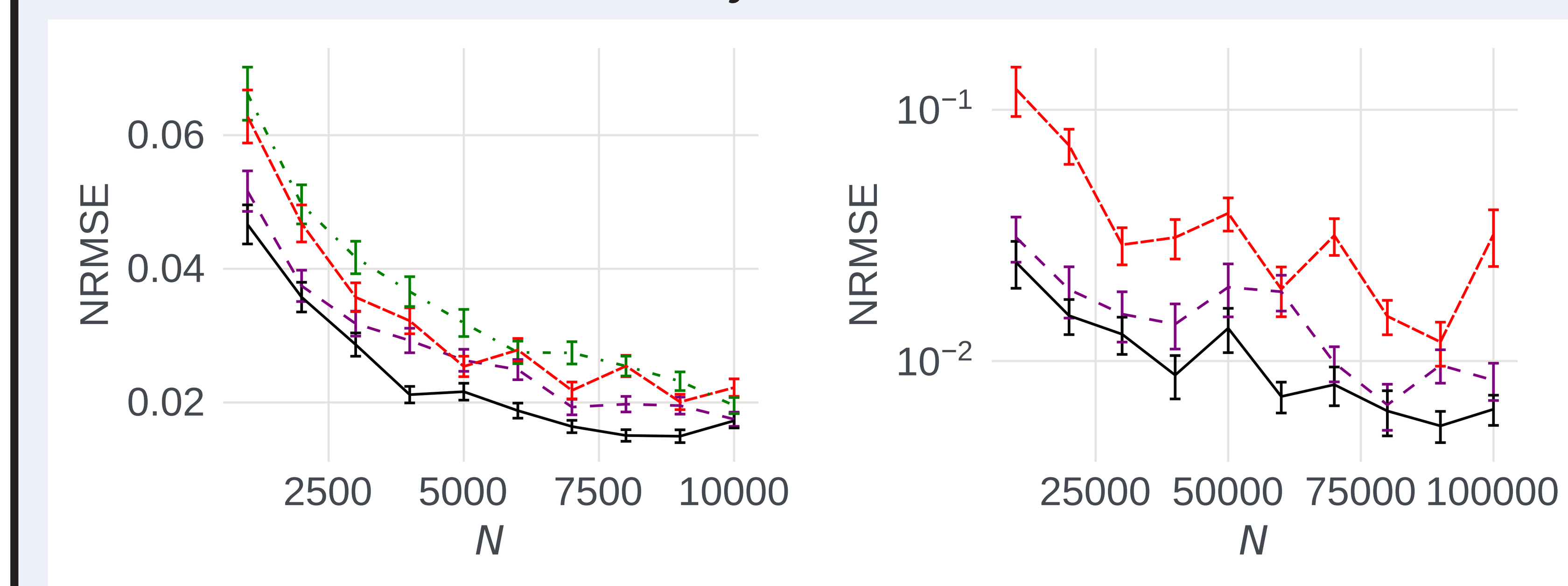
Extensions.

Bit-pushing can also be applied to signed values, higher moments, products, and geometric means.

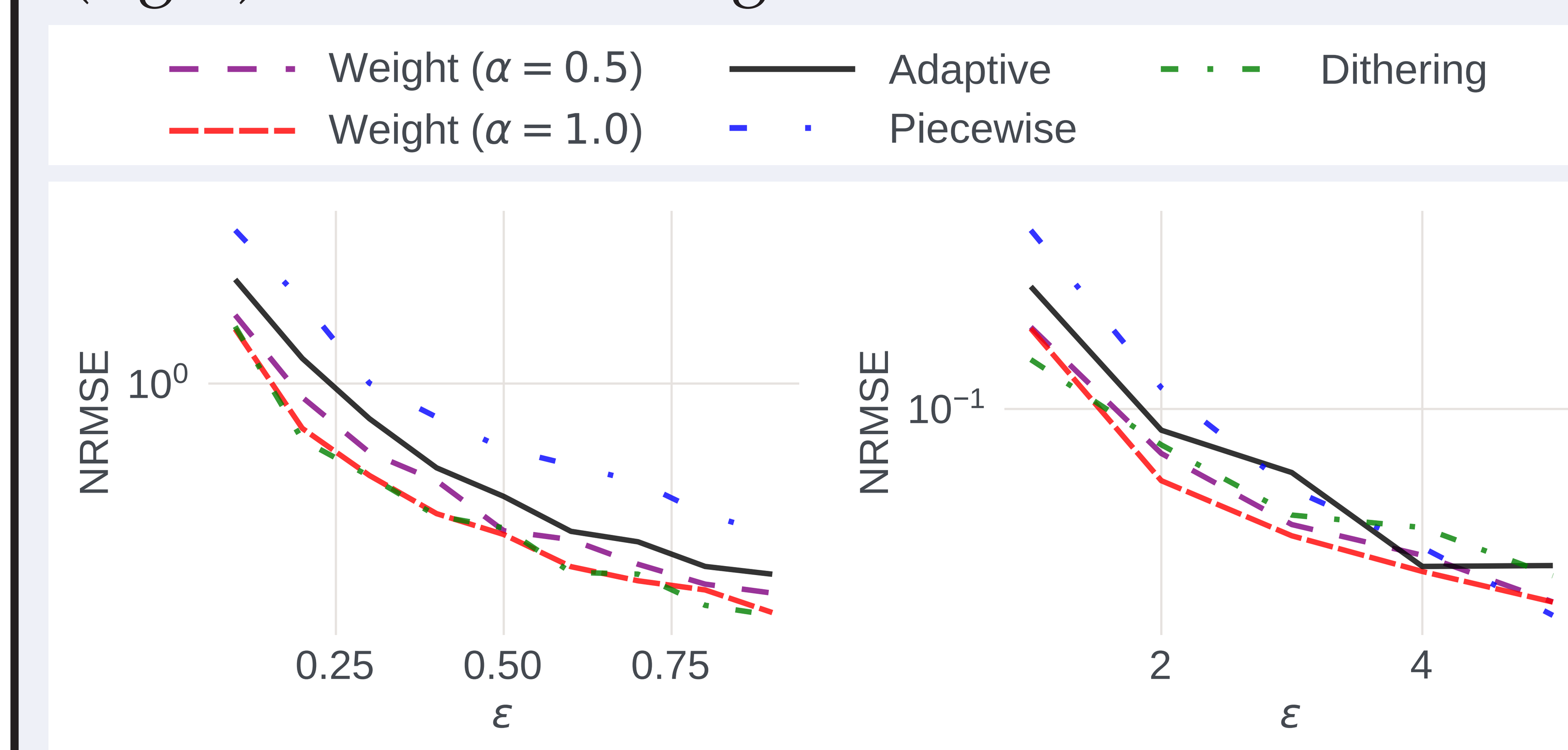
Results



On synthetic (Normal) data, root-mean-square error (RMSE), normalized by true mean value, is low.



Normalized RMSE for mean (left) and variance (right) of US Census age data decreases with N.



Normalized RMSE for ϵ-LDP mean estimation decreases with ϵ, as predicted.