Differentially Private Federated Quantiles with the Distributed Discrete Gaussian Mechanism

Krishna Pillutla*, Yassine Laguel*, Jérôme Malick, Zaid Harchaoui

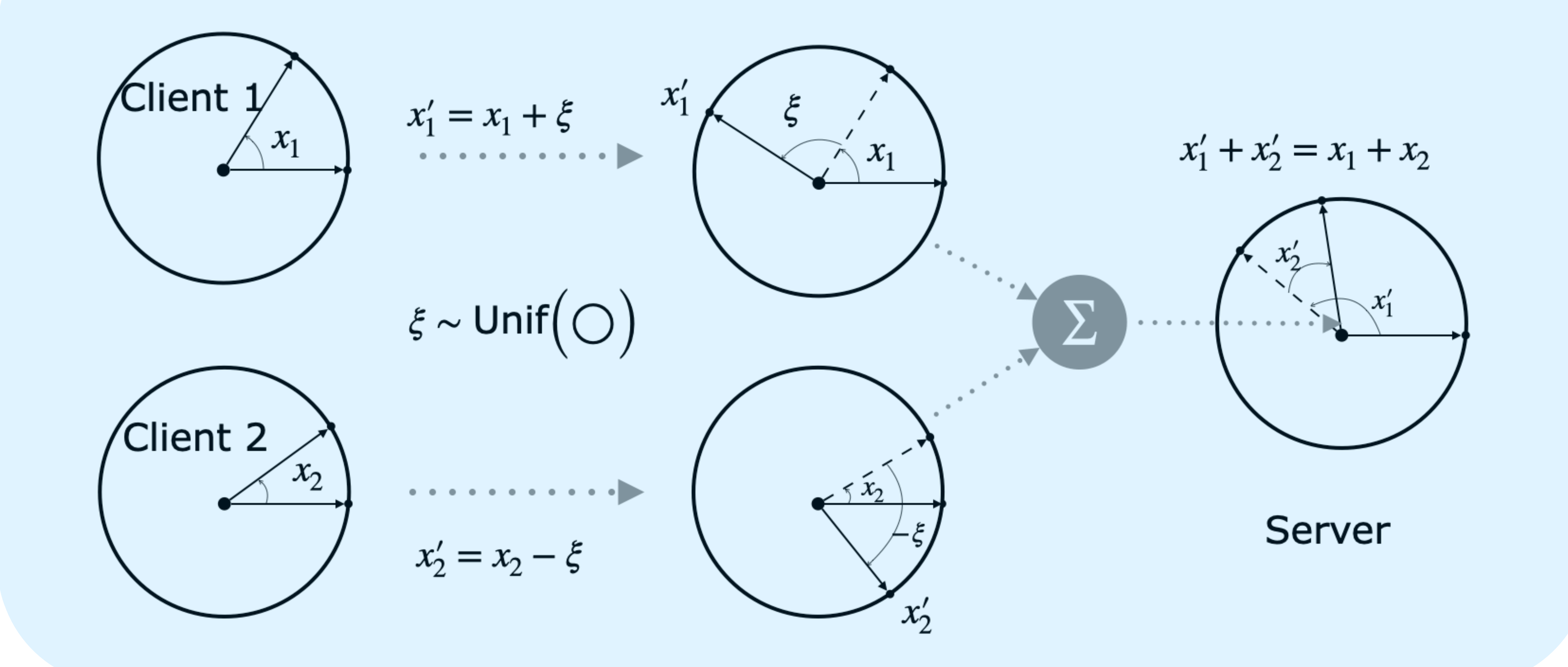
Google Research, Rutgers, CNRS, W

Federated quantiles

Each client i has a scalar s_i

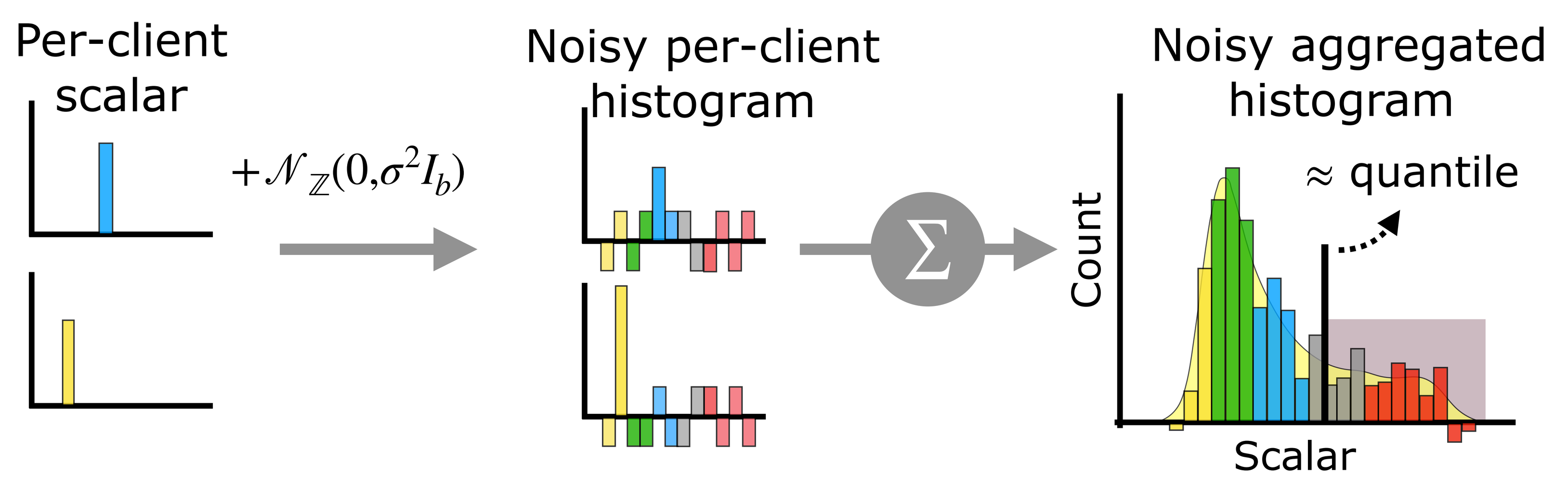
Goal: Compute the $(1-\theta)$-quantile $Q_\theta(s_1, \dots, s_n)$ with **distributed differential privacy**

- Emulate trusted server with crypto
- Primitive: **Secure summation**

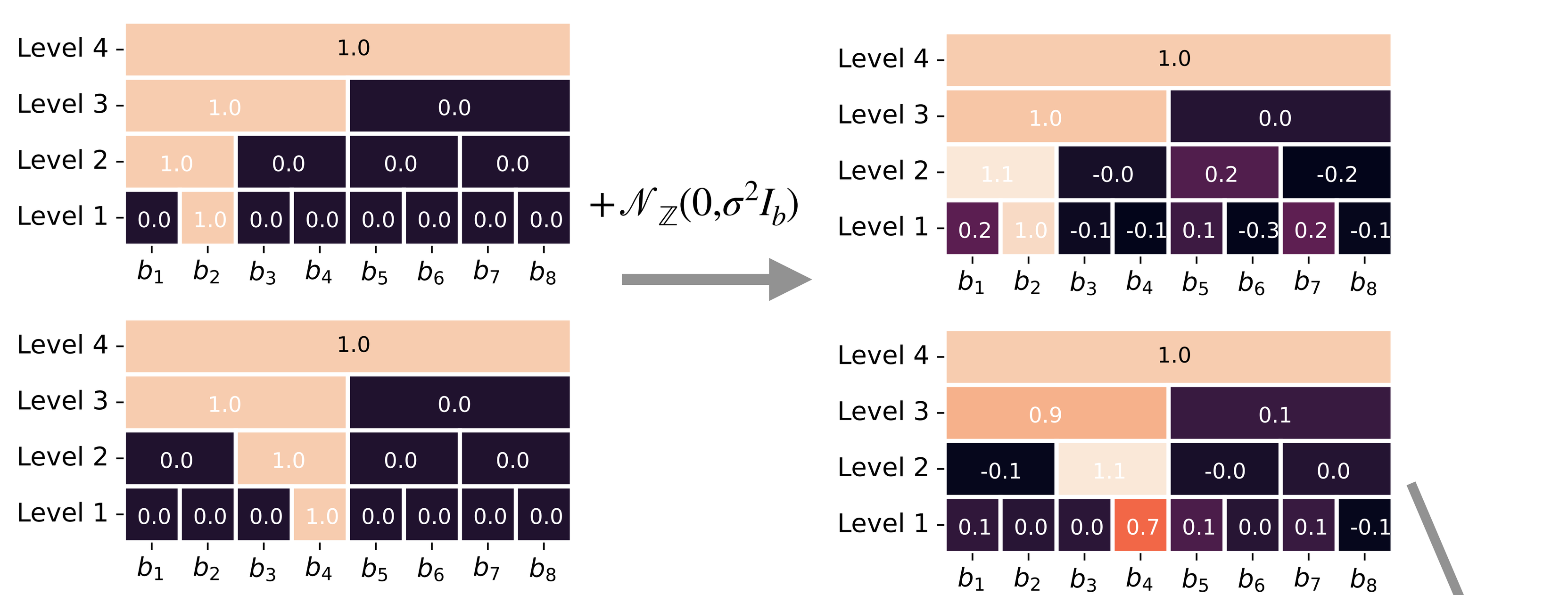


Algorithms

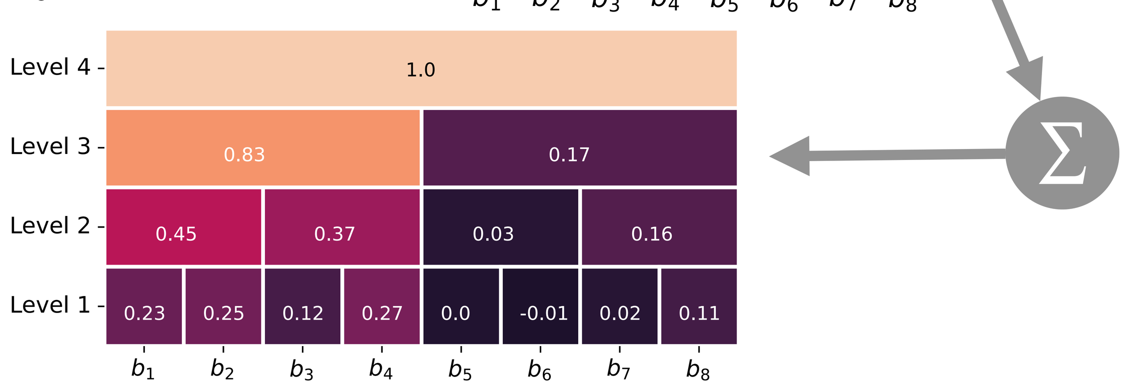
Flat histograms + distributed discrete Gaussian



Hierarchical histograms | Tree aggregation



Hay et al. (2010),
Dwork et al. (2010),
Chan et al. (2011),
Smith et al. (2017),
Cormode et al. (2019)



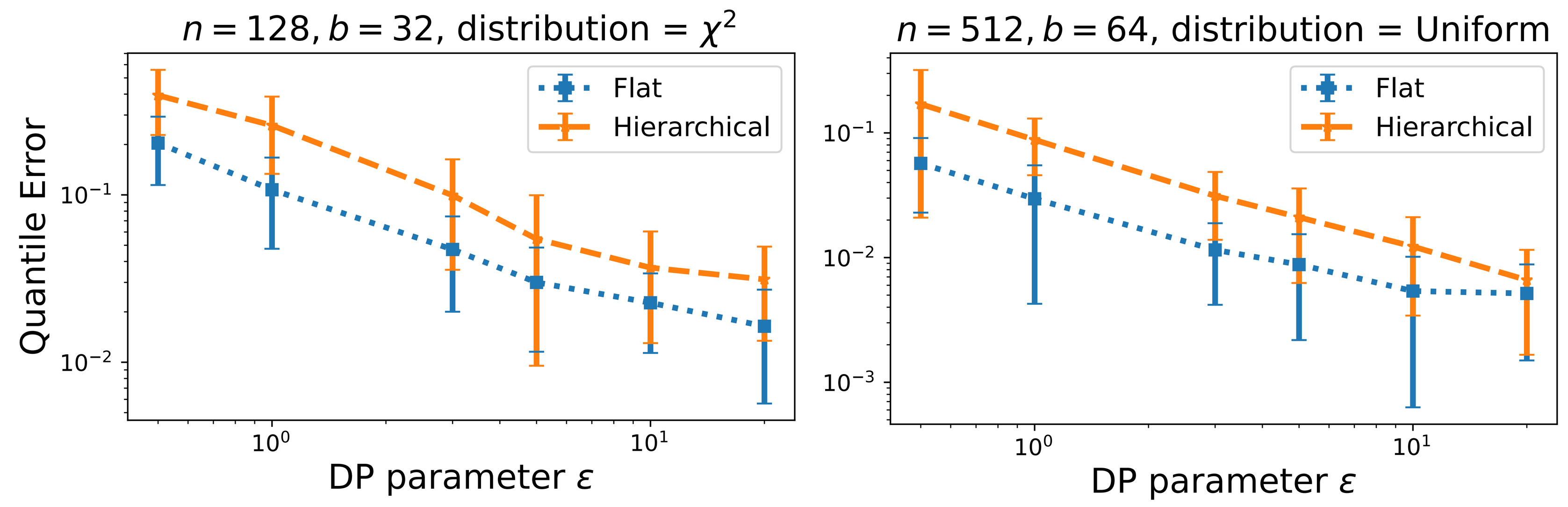
Main results

Theorem: For $(1/2)\epsilon^2$-zCDP quantiles, w.p. $\geq 1-\alpha$

	Flat	**Hierarchical**
σ	$\dfrac{1}{\epsilon\sqrt{n}}$	$\dfrac{\log_2 b}{\epsilon\sqrt{n}}$
Quantile Error	$\dfrac{\sqrt{b \log \frac{1}{\alpha}}}{\epsilon n}$	$\dfrac{\sqrt{\log_2^3 b \, \log \frac{b}{\alpha}}}{\epsilon n}$

Asymptotics: Flat is suboptimal: $\sqrt{b}$

Finite sample: Flat is better for $n \lesssim 2.5 \times 10^6$



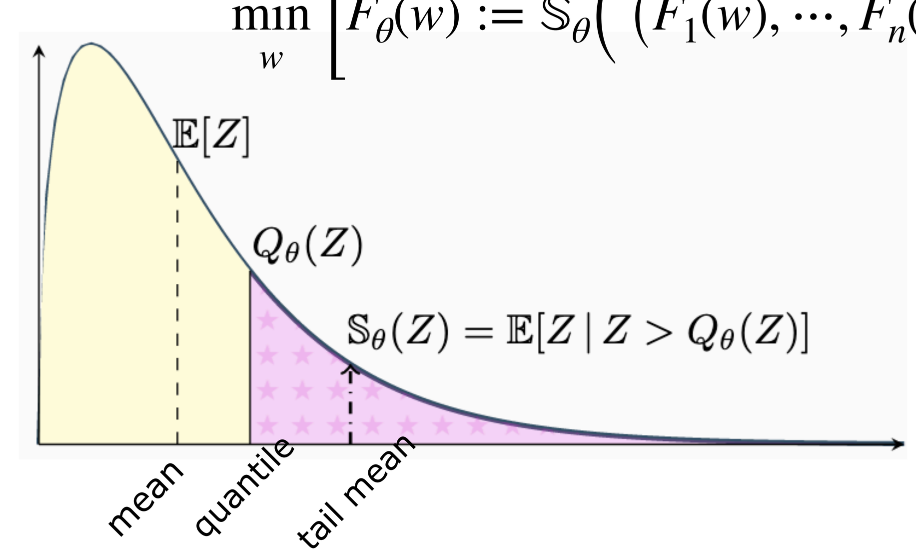
Distributionally robust FL

Setting: Client objectives $F_i(w) = \mathbb{E}_{z\sim p_i}\left[f(w;z)\right]$

Goal: Minimize the tail error [Laguel, Pillutla et al. (2021)]

$$\min_w \left[F_\theta(w) := \mathbb{S}_\theta\Big(\big(F_1(w), \cdots, F_n(w)\big) \Big) \right]$$

Superquantile | Conditional Value at Risk



θ = tail fraction

$Q_\theta(Z) = (1-\theta)$-quantile of Z

$\mathbb{S}_\theta(Z) = (1-\theta)$-superquantile

Distributional robustness: for a new client with distribution $p_\pi = \sum_{i=1}^n \pi_i p_i$, the objective is equivalent to

$$F_\theta(w) = \max_{\pi : \pi_i \leq (\theta n)^{-1}} \mathbb{E}_{z\sim p_\pi}\left[f(w;z)\right]$$

End-to-end DP Optimization

Subgradient expression: if θn is an integer then

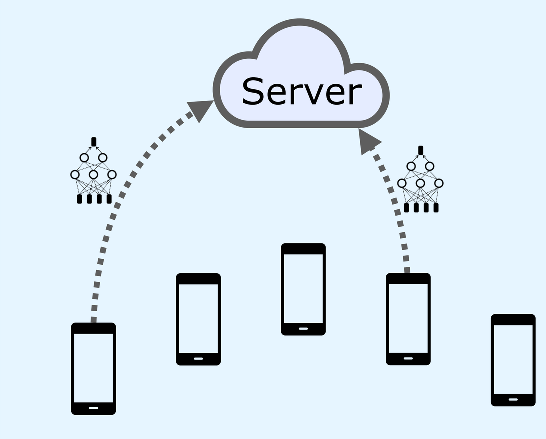
$$\partial F_\theta(w) \ni \sum_{i=1}^n \pi_i^\star \nabla F_i(w) \quad \text{where} \quad \pi_i^\star \propto \mathbb{I}(F_i(w) > q), \quad q = Q_\theta\big(F_1(w), \dots, F_n(w)\big)$$

Algorithm: Like FedAvg but in each round

Tail $= \Big\{ i : F_i(w) > Q_\theta\big(F_1(w), \dots, F_n(w)\big) \Big\}$

- Estimate $q \approx Q_\theta\big(F_1(w), \dots, F_n(w)\big)$ distributed discrete Gaussian mechanism
- Aggregate updates from the tail with the Gaussian mechanism (similar to DP-FedAvg)

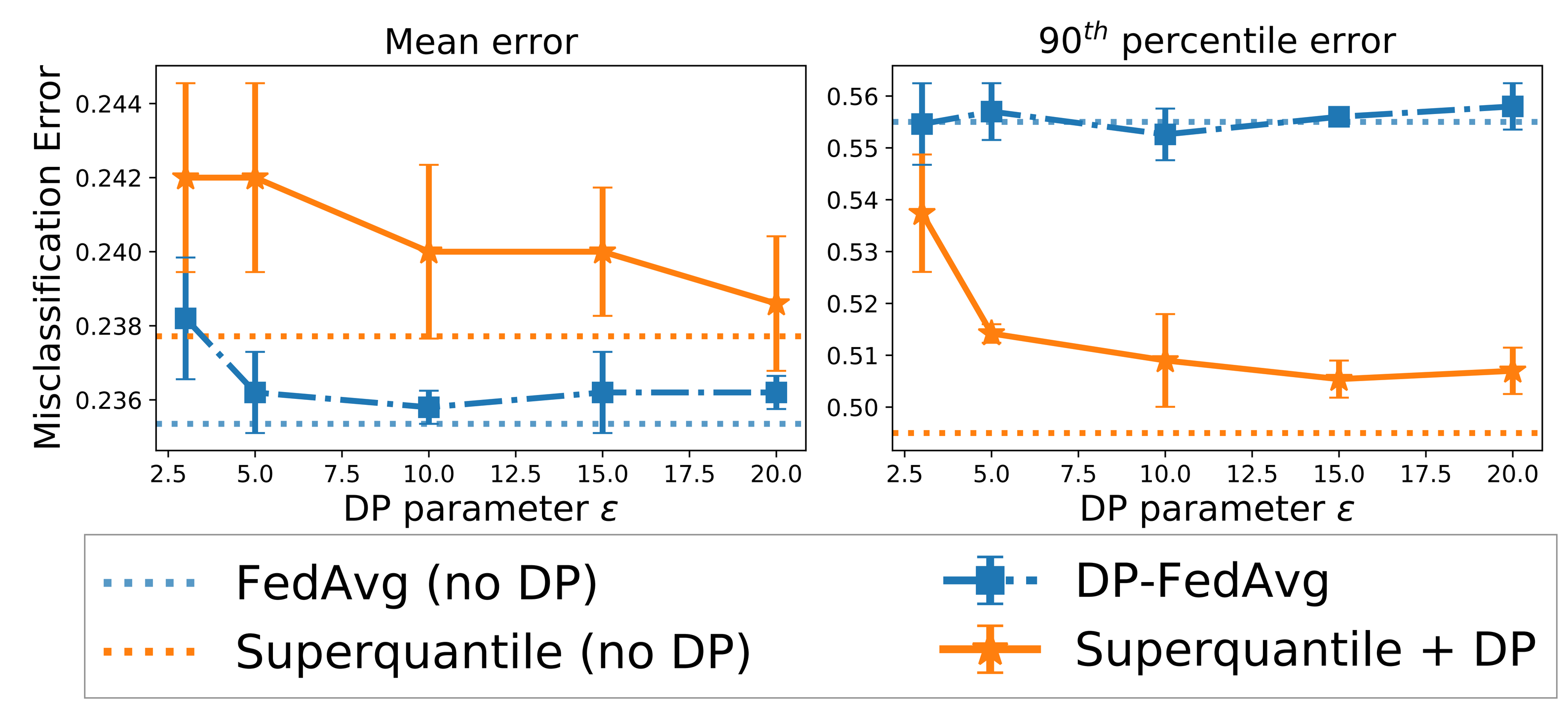
Aggregate updates from tail clients only



Hyperparameters: Number of bins b,
Fraction of privacy budget spent on the quantile,
Loss upper bound (clip losses to $[0,B]$),

Experiments

Synthetic 10-class classification



Pillutla*, Laguel*, Malick, Harchaoui. *Federated Learning with Superquantile Aggregation for Heterogenous Data.* Mach. Learn. (To appear, 2022)

Code

SCAN ME

krishnap25.github.io

KrishnaPillutla