The Skellam Mechanism for Differentially Private Federated Learning

Naman Agarwal, Peter Kairouz, Ziyu Liu[‡]
{namanagarwal, kairouz}@google.com, ziyuliu@cs.cmu.edu
[†]Alphabetical authorship [‡]Work done while at Google

Google Research

Background

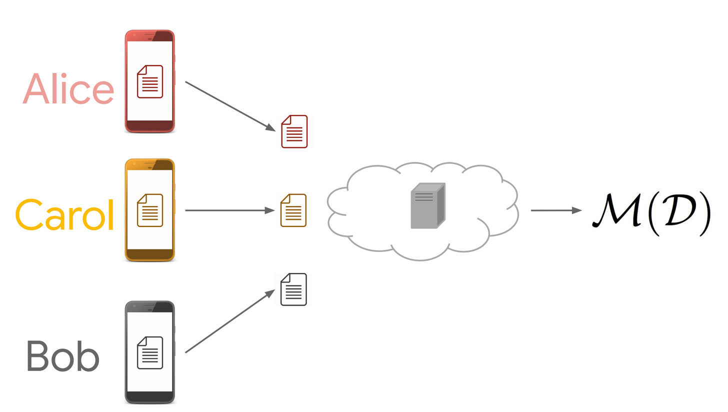
Differentially Private FL

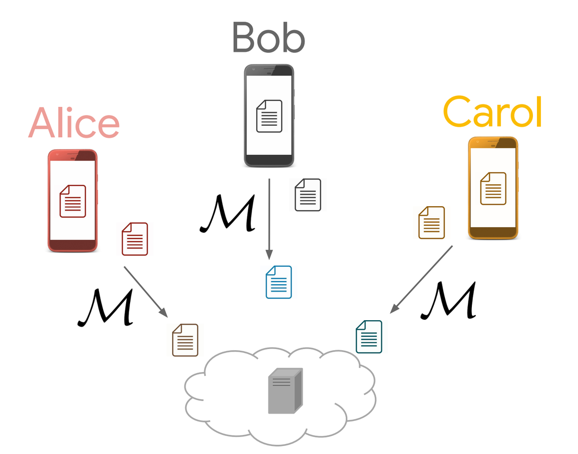
- While Federated Learning (FL) ensures raw data are kept decentralized, it may not provide formal privacy guarantees.
- Differentially Private FL: client updates (e.g. gradients) are clipped and noised appropriately to give quantifiable, user-level DP guarantees.

Privacy Models

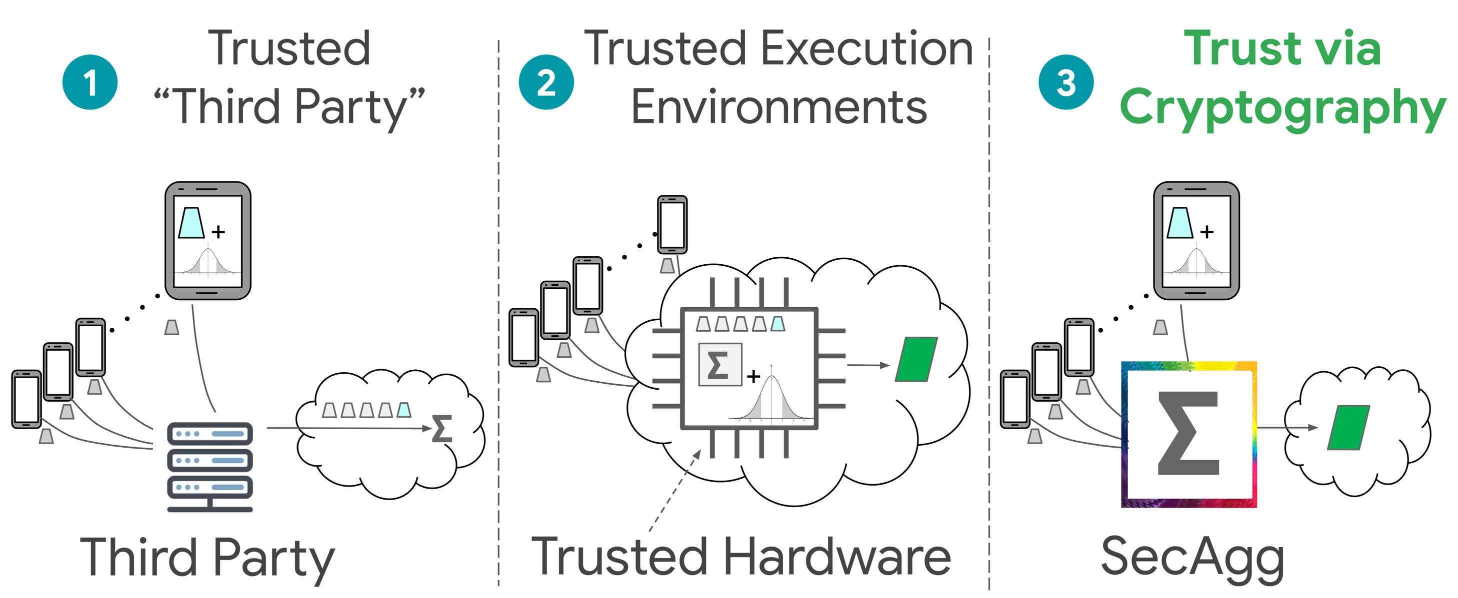
Central DP: Noise@Server
- Full trust on server
- Better utility

Local DP: Noise@Clients
- No trust on server
- Poor utility





Distributed DP
Aims to achieve the utility of Central DP without fully trusting the server by "distributing" trust:



Some Challenges

- Gaussian can't be stored exactly on computers
- Secure Aggregation (SecAgg) operates on a finite group (integers with modular arithmetic)
  - Need **discrete** DP mechanisms
- **Communication efficiency** is vital for practical FL
  - Need to consider the trade-off against privacy and utility (both modular & quantization errors)

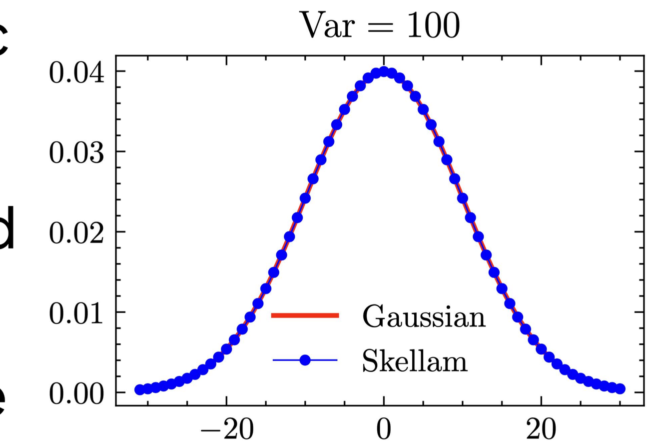
Symmetric Skellam Distribution

- With mean Δ and variance μ, a symmetric *Skellam* RV is given by

$$X \sim \mathrm{Sk}_{\Delta,\mu} \quad \text{with} \quad P(X_i = k) = e^{-\mu} \underbrace{I_{k-\Delta_i}(\mu)}$$

modified Bessel functions of the first kind

- A Skellam RV is the difference between two independent Poisson RVs; if the Poissons have the same parameter, then the resulting Skellam is symmetric
- **Easy to sample:** efficient/vetted samplers like np.random.poisson
- **Closed under summation:** easily switch between **central DP** and **distributed DP** (adding noise centrally vs locally, see left section)
- Skellam approaches the continuous Gaussian with larger variance



Skellam Mechanism for Federated Learning

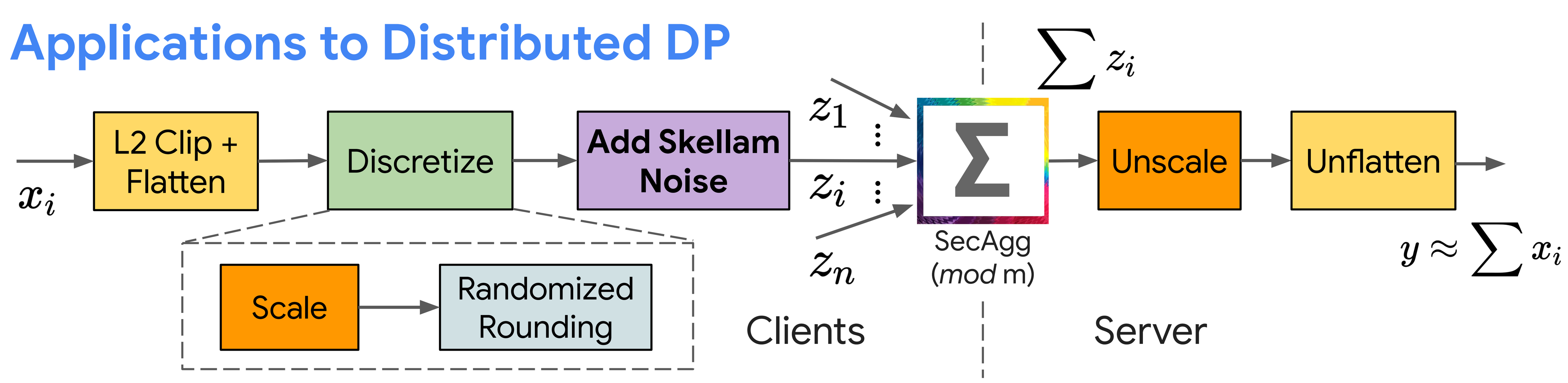
- **Skellam Mechanism:** $\mathrm{Sk}_{0,\mu}(f(D)) = f(D) + Z \text{ where } Z \sim \mathrm{Sk}_{0,\mu}$
- **Prior work:** Analysis for scalar queries only, no **Rényi DP / zCDP** analysis available, no tight compositions → not suitable for FL and high-dim queries. Direct generalizations of existing results to vector queries with composition gives poor performance.
- **Our contribution:** A practical alternative to discrete Gaussians for central/distributed DP
  1. **Tight Rényi DP analysis:** Our RDP guarantee of multi-dim Skellam mechanism is at most $1 + O(1/\mu)$ times that of the Gaussian mechanism (μ = noise variance)

For ℓ_1, ℓ_2 sensitivities Δ_1, Δ_2, central variance μ, and order $\alpha > 1, \alpha \in \mathbb{Z}$,

$$\varepsilon(\alpha) \le \underbrace{\boxed{\frac{\alpha \Delta_2^2}{2\mu}}}_{\text{Gaussian RDP}} + \min\underbrace{\boxed{\left(\frac{(2\alpha - 1)\Delta_2^2 + 6\Delta_1}{4\mu^2}, \frac{3\Delta_1}{2\mu}\right)}}_{\text{Goes to 0 with larger } \mu \text{ (higher privacy)}}$$

  2. **Large-scale empirical evaluation:** We show that Skellam works well in practice and performs as good as the continuous/discrete Gaussian in FL applications
- **Proof Idea**
  - RDP analysis requires bounding ratios of successive bessel functions $\frac{I_{\nu-1}(x)}{I_\nu(x)}$
  - Previous work uses a bound that leads to a loose 2nd term and strong L1 dependence
  - We use a tighter bound capturing finer deviations, giving rapid decay of the 2nd term

Applications to Distributed DP



Empirical Results

Stack Overflow Next Word Prediction

>10^8 training question/answer sentences grouped by >340k Stack Overflow users

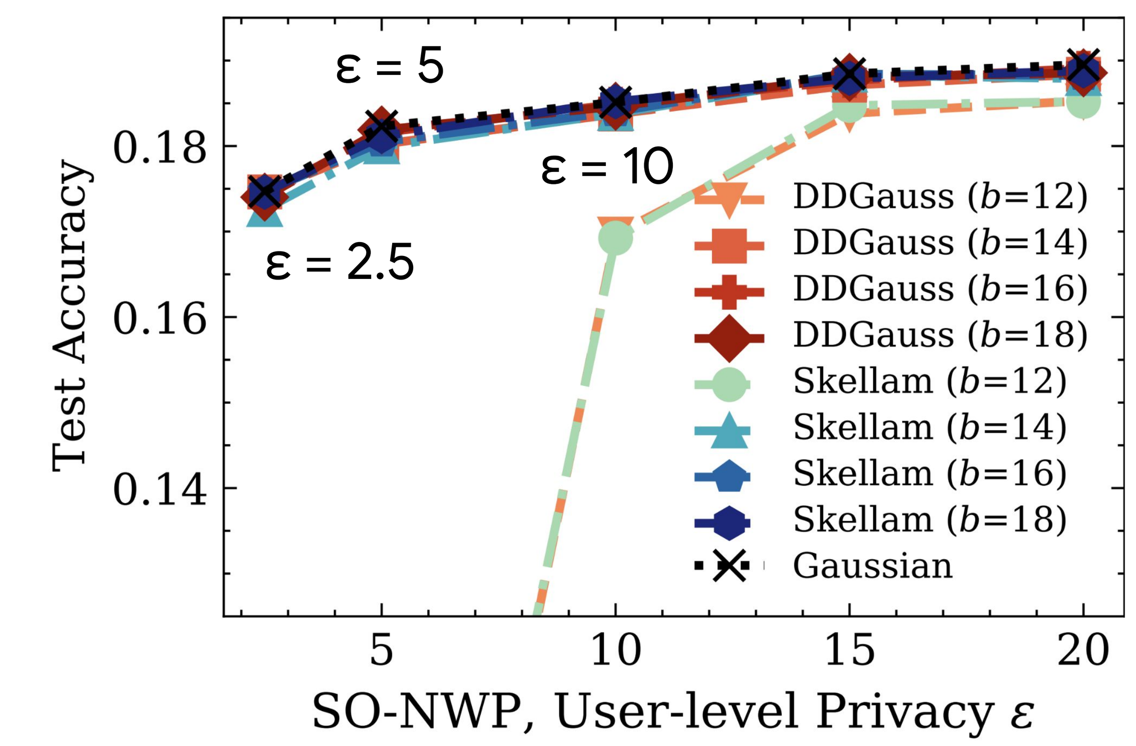


Fig. 1. Skellam matches the central continuous & distributed discrete Gaussian. $\delta = 10^{-6}$. $n = 100$.

Distributed Mean Estimation

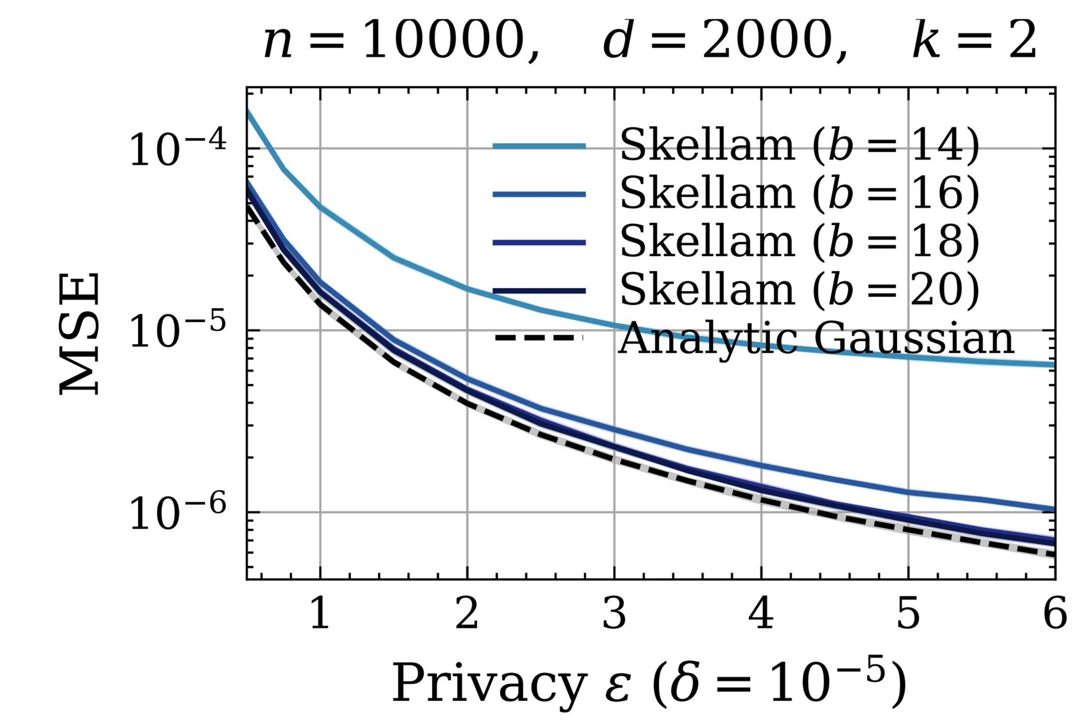


Fig. 2. Skellam matches the Analytic Gaussian Mechanism at n=10000 clients with enough bit-width

See full version (arXiv:2110.04995) for more!

Conclusion

- Skellam performs as good as continuous / discrete Gaussians in realistic settings
- Skellam is a practical alternative to discrete Gaussian for central/distributed DP due to (a) ease of sampling: friendly to DP and ML developers; (b) closure under summation: suitable for highly distributed DP settings.
- **Code:** github.com/google-research/federated/