The Distributed Discrete Gaussian Mechanism for Federated Learning with Secure Aggregation

Peter Kairouz, Ziyu Liu, Thomas Steinke
(kairouz, klz, steinke)@google.com
https://github.com/google-research/federated/tree/master/distributed_dp

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:** Noisy@Server
  - Full trust on server
  - Poor utility
- **Local DP:** Noisy@Clients
  - No trust on server
  - Better utility

Distributed DP (this work)
Aims to achieve the utility of Central DP without fully trusting the server by “distributing” trust:
- Trusted “Third Party”
- Trusted Hardware
- Trusted Execution Environments
- Trust via Cryptography

Some Challenges
- Secure Aggregation (SecAgg) operates on a finite group (integers with modular arithmetic)
  - Need discrete DP mechanisms
- Sums of Discrete Gaussians ≠ Discrete Gaussians
  - Need to carefully analyze the effects on DP
- Communication efficiency is vital for practical FL
  - Need to consider the trade-off against privacy and utility (both modular & quantization errors)

Method: Distributed Discrete Gaussian

<table>
<thead>
<tr>
<th>L2 Clip</th>
<th>Flatten</th>
<th>Discretize</th>
<th>Add Local DGaussian</th>
<th>SecAgg (mod m)</th>
<th>Unscale</th>
<th>Unflatten</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Summary: An end-to-end system for differentially private FL combining compression, SecAgg, and local noising that matches the privacy / accuracy of Central DP.

Procedure
1. **L2 Clipping:** Initial bound on the client vector L2 sensitivity $c$
2. **Flattening:** Random unitary transform to spread values across vector dimensions
   - Controls the L-inf norm to Lower quantization errors / Less modular wrap-around
3. **Discretization:** Round input values to the discrete grid (rounding granularity $\gamma$)
   - Corresponds to scaling by $1/\gamma +$ rounding to integers
   - Scaling: Smaller $\gamma$ → Less rounding errors, but larger values (more communication)
   - Randomized rounding: Unbiased discretization (e.g. 4.2 to 4/5 with 80%/20% prob)
   - Norm inflation: Rounding may increase norm → more DP noise for same privacy
   - Conditional rounding: we give a tighter probabilistic bound and retry rounding until the norm is smaller (less DP noise):
   - $\delta^2$: rounding bias $d$: vector dimension
   - $\Delta^2 := \min \left\{ \frac{c^2 + \gamma^2 d/4 + \sqrt{2\log(1/\delta) - \gamma (c + \gamma d/2)}}{c + \gamma d}, \frac{c^2 + \gamma^2 d}{2c} \right\}$

4. **Local Noising**
   - Each client adds their own local discrete Gaussian noise
   - We give a **tight bound** on the sums of discrete Gaussians, which leads to extremely close privacy guarantees to central DP (central continuous/discrete Gaussian noise):
   - $D_{\infty}(Z|W) = \sup_{z,w \in \mathbb{Z}} \log \left| \frac{\mathbb{P}[Z = z]}{\mathbb{P}[W = w]} \right| \leq 5c^2 - c^2/(4\pi^2 + 1)$

5. **SecAgg**
   - Securely sums locally clipped, scaled, rounded, and noised client vectors
   - SecAgg group size $m = 2^k$ determines the communication bit-width (for the sum)
   - Scaling $(1/\gamma)$ is chosen to keep modular wrapping infrequent (often < 0.05% prob)

6. **Server Post-processing**
   - Unscale and undo the flattening transform
   - Extension: may optionally collect metrics to update $c$ and $\gamma$ for the next iteration

Empirical Results
Stack Overflow Next Word Prediction
>10^8 training question/answer sentences grouped by >340k Stack Overflow users

**Fig. 1.** Our method matches the central continuous Gaussian if bit-width $B$ is sufficient (≥ 14), $\delta = 10^{-6}$.

**Fig. 2.** DDGauss works in production-scale (1000 clients) and low-noise (utility-first) settings. $z$: noise multiplier.

Conclusion & Future Directions
- **Distributed DP** achieves accuracy similar to central DP with only 16 bits per value
- **Next steps:** (a) Discrete Fourier Transform instead of Walsh-Hadamard Transform for better compression efficiency, (b) lower bound on communication, privacy, and accuracy trade-offs, (c) exploring the role of sparsity under distributed DP.

See full version (arXiv:2102.06387) for more!