DP-ML Proofs

September 30, 2019

1 Privacy Loss when adding $\epsilon_1 + \epsilon_2$ noise

Let:

$A(x) => (\epsilon_1, \delta)$ differentially private mechanism by adding noise from $N(0, \sigma_1^2)$
$B(x) => (\epsilon_2, \delta)$ differentially private mechanism by adding noise from $N(0, \sigma_2^2)$

Theorem 1.1. *If C(x) is a dp-mechanism that adds the sum of noise sampled from $N(0, \sigma_1^2)$ and $N(0, \sigma_2^2)$, then C(x) is $\sqrt{(\frac{\epsilon_1^2 * \epsilon_2^2}{\epsilon_1^2 + \epsilon_2^2})}$ differentially private.*

Proof. We know that the sum of two normal distributed random variable is also normal $=> N(0, \sigma_1^2) + N(0, \sigma_2^2) = N(0, \sigma_1^2 + \sigma_2^2)$

Therefore, summing up noise from two randomly distributed variables is equivalent to sampling noise from $N(0, \sigma_1^2 + \sigma_2^2 = \sigma_3^2)$

From [3] it follows that, if σ is equivalent to

$$\frac{s}{\epsilon}\sqrt{(}2ln\frac{1.25}{\delta})$$

then a step is (ϵ, δ) differentially private.

We know:

$$\sigma_3^2 = \sigma_1^2 + \sigma_2^2$$
$$\sigma_3^2 = \frac{s^2}{\epsilon_1^2}(2ln\frac{1.25}{\delta}) + \frac{s^2}{\epsilon_1^2}(2ln\frac{1.25}{\delta})$$
$$\sigma_3^2 = (2ln\frac{1.25}{\delta})(\frac{s^2}{\epsilon_1^2} + \frac{s^2}{\epsilon_1^2})$$
$$\frac{s^2}{\epsilon_3^2}(2ln\frac{1.25}{\delta}) = (2ln\frac{1.25}{\delta})(\frac{s^2}{\epsilon_1^2} + \frac{s^2}{\epsilon_2^2})$$
$$\frac{s^2}{\epsilon_3^2}(2ln\frac{1.25}{\delta}) = (2ln\frac{1.25}{\delta})(\frac{s^2}{\epsilon_1^2} + \frac{s^2}{\epsilon_2^2})$$
$$\frac{s^2}{\epsilon_3^2}(2ln\frac{1.25}{\delta}) = (2ln\frac{1.25}{\delta})(\frac{s^2}{\epsilon_1^2} + \frac{s^2}{\epsilon_2^2})$$

$$\frac{1}{\epsilon_3^2} = \frac{1}{\epsilon_1^2} + \frac{1}{\epsilon_2^2}$$

$$\frac{1}{\epsilon_3^2} = \frac{\epsilon_1^2 + \epsilon_2^2}{\epsilon_1^2 * \epsilon_2^2}$$

$$\epsilon_3^2 = \frac{\epsilon_1^2 * \epsilon_2^2}{\epsilon_1^2 + \epsilon_2^2}$$

$$\epsilon_3 = \sqrt(\frac{\epsilon_1^2 * \epsilon_2^2}{\epsilon_1^2 + \epsilon_2^2})$$

■

Here are a few examples for the resulting epsilon value when you add two noise vectors each satisfying (ϵ_1, δ) and (ϵ_2, δ) respectively:

ϵ_1	ϵ_2	ϵ_3
0.5	0.75	0.416
1.0	0.5	0.447
2.0	1.0	0.894
2.0	0.5	0.485

MS ► *Not sure if the above result can be used for $\epsilon > 1$ values since $\frac{s}{\epsilon}\sqrt(2ln\frac{1.25}{\delta})$ might only applicable for cases where both $\epsilon_1 < 1$ and $\epsilon_2 < 1$ [3]*◄

2 Next Steps/TODOs

Here are some potential next steps/proofs for designing a private Federated Learning system with an untrusted aggregator:

1. *How much privacy gain do you get when only observing only the noisy aggregate compared to observing all the individual updates?*

Some Ideas => From the secure aggregation paper (Appendix A) [2], we know that we can get the same ϵ- guarantee with secure aggregation by sampling from $N(0, \sigma/\sqrt(n))$ compared to sampling from $N(0, \sigma)$ with no secure aggregation.

By using the same argument, if A(x) satisfies (ϵ, δ)- differential privacy when the adversary observes each update protected by noise sampled from $N(0, \sigma)$, then with the adversary observing only the aggregate of the noisy updates, it becomes equivalent to (ϵ, δ) guarantee by sampling from $N(0, \sigma * \sqrt(n))$. MS ► *??*◄

However, a key assumption to get the gain above would be a shift to computational differential privacy [4]. Not sure currently that if we make this assumption how it would affect privacy calculations?

MS ► *With computation differential privacy can I still use $\frac{s}{\epsilon}\sqrt(2ln\frac{1.25}{\delta})$to get (ϵ, δ) differential privacy here?*◄

2. *How would the privacy loss compose over N rounds such that an adversary observes the individual dp-updates in m out of N rounds and the aggregate in others?*

Would it be possible to use one composition theorem out of the box for this? Some potential compositions that can be used:

1. Advanced composition theorem [5] $=> O(\sqrt(klog(\frac{1}{\delta'})).\epsilon, k * \delta + \delta')$ $(k < 1/\epsilon^2)$
2. Moments Accountant [1] $=> O(q * \epsilon * sqrt(T))$
3. Amplification by sampling $O(q * \epsilon, q * \delta)$ [1]

MS ►*This is a sample comment*◄

References

[1] Abadi, M., Chu, A., Goodfellow, I., McMahan, B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *23rd ACM Conference on Computer and Communications Security* (2016), CCS.

[2] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (2017), CCS.

[3] Dwork, C., and Roth, A. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science 9*, 3-4 (2014).

[4] Mironov, I., Pandey, O., Reingold, O., and Vadhan, S. Computational differential privacy. In *Proceedings of the 29th Annual International Cryptology Conference on Advances in Cryptology* (2009), CRYPTO '09.

[5] Vadhan, S. P. The complexity of differential privacy. In *Tutorials on the Foundations of Cryptography* (2017).