Can Byzantine Learning be Private?

Principles of Distributed Computing (PODL)

Rafael PINOT – July 25 2022

EPFL – Ditributed Computing Lab

Contact info: Rafael.pinot@epfl.ch

Based on a joint works with

Youssef Allouah

Sadegh Farhadkhani

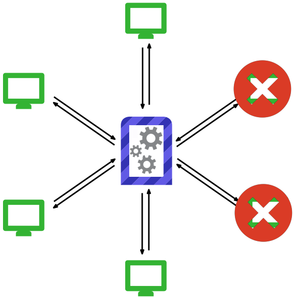
Rachid Guerraoui

Nirupam Gupta

John Stephan

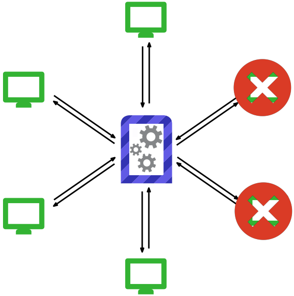
Standards of Byzantine learning

The Byzantine threat model in parameter-server



- n workers - one parameter-server
- Some workers may crash or be malicious
- We consider the standard **Byzantine threat model** Lamport et al. (1982)
- Up to $f < n/2$ workers may be Byzantine

The Byzantine threat model in parameter-server



- n workers - one parameter-server
- Some workers may crash or be malicious
- We consider the standard **Byzantine threat model** Lamport et al. (1982)
- Up to $f < n/2$ workers may be Byzantine

Practical objective: Find, despite the presence of up to f Byzantine workers, an η-**critical** point of Q, i.e., the server outputs $\widehat{\theta} \in \mathbb{R}^d$ such that

$$\mathbb{E}\left[\|\nabla Q\left(\widehat{\theta}\right)\|^2\right] \leq \eta$$

Standard approach to confer Byzantine resilience

Byzantine-resilient parameter-server SGD:

At every step $t = 1, \ldots, T$

1. **Worker** i computes & sends a gradient $g_t^{(i)}$
   $\rightarrow$ *A Byzantine worker j can send anything for $g_t^{(j)}$*
2. **Server** updates with a non-linear rule F & broadcasts

$$\theta_{t+1} = \theta_t - \gamma F\left(g_t^{(1)}, \cdots, g_t^{(n)}\right)$$

Standard approach to confer Byzantine resilience

Byzantine-resilient parameter-server SGD:

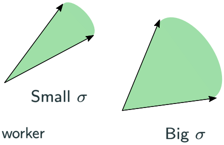
At every step $t = 1, \ldots, T$

1. **Worker** i computes & sends a gradient $g_t^{(i)}$
   $\rightarrow$ *A Byzantine worker j can send anything for $g_t^{(j)}$*
2. **Server** updates with a non-linear rule F & broadcasts

$$\theta_{t+1} = \theta_t - \gamma F\left(g_t^{(1)}, \cdots, g_t^{(n)}\right)$$

One of the main challenges is uncertainty:

$$\mathbb{E}\left[\boldsymbol{g}_t^{(i)} - \mathbb{E}\left[\boldsymbol{g}_t^{(i)}\right]\right] \leq \sigma^2$$

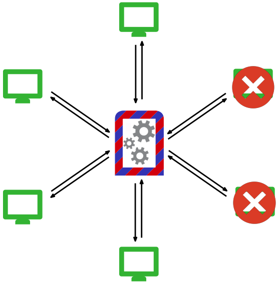


Range of plausible gradients for a honest worker

Essentially: the bigger σ, the harder it is to defend against Byzantine workers

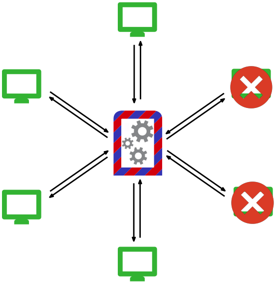
Privacy in distributed ML with honest-but-curious server

Privacy threat(s): External threat and curious server



- Privacy threats can come from **several sources** (internal or external)
- Curious parameter-server:
  - → **hacking/corruption** of the machine
  - → **curious** service provider (e.g, in API)

Privacy threat(s): External threat and curious server



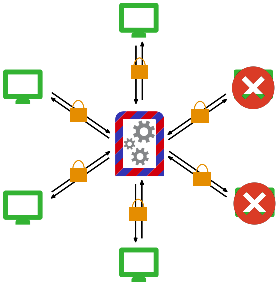
- Privacy threats can come from **several sources** (internal or external)
- Curious parameter-server:
  - → **hacking/corruption** of the machine
  - → **curious** service provider (e.g, in API)

Folklore belief: sending gradients is private because raw data is not shared

→ Massive privacy leakage can occur with gradients Zhu et al. (2019)

→ Need to rethink the scheme to make it more private

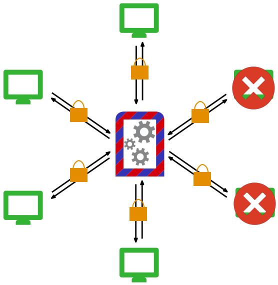
Open problem 1: Cryptographic primitives



Cryptographic scheme on the communications, e.g., **Homomorphic encryption** Paillier (1999)

- very **active** area of research in ML
- **difficult to scale** to large models
- not very adapted to **non-linearity**

Open problem 1: Cryptographic primitives

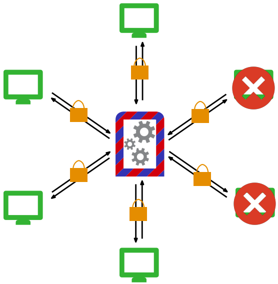


Cryptographic scheme on the communications, e.g., **Homomorphic encryption** Paillier (1999)

- very **active** area of research in ML
- **difficult to scale** to large models
- not very adapted to **non-linearity**

Open problem 1: Find ways to compute F in this very challenging setting

Open problem 1: Cryptographic primitives



Cryptographic scheme on the communications, e.g., **Homomorphic encryption** Paillier (1999)

- very **active** area of research in ML
- **difficult to scale** to large models
- not very adapted to **non-linearity**

Open problem 1: Find ways to compute F in this very challenging setting

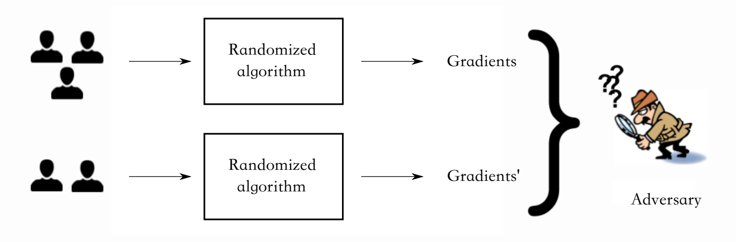
Alternative solution: differential privacy?

Can we combine Byzantine learning and differential privacy?

Differential Privacy (Recall)

Differential privacy, introduced in Dwork et al. (2014), the standard for privacy in ML

Basic idea: randomize the workers' behavior to provide privacy

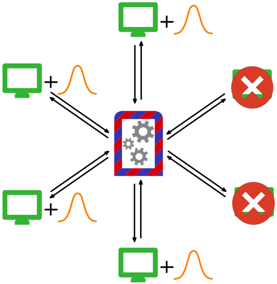


The adversary is not able to say whether the change in the gradients is due to the change in the workers data or to randomization

Differential privacy & Noise injection mechanisms

Gaussian mechanism: Worker i computes and sends a noisy gradient

$$\tilde{g}_t^{(i)} := g_t^{(i)} + \mathcal{N}\left(0, s^2 I_d\right) \; ; \text{ Balle and Wang (2018)}$$

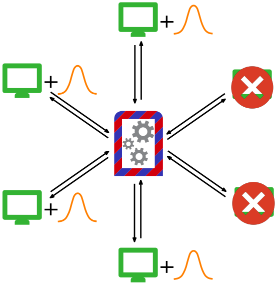


- Easy to implement and **efficient**
- Easy to analyze even for complex models
- Privacy guarantee grows with s^2

Differential privacy & Noise injection mechanisms

Gaussian mechanism: Worker i computes and sends a noisy gradient

$$\tilde{g}_t^{(i)} := g_t^{(i)} + \mathcal{N}\left(0, s^2 I_d\right) \; ; \; \text{Balle and Wang (2018)}$$

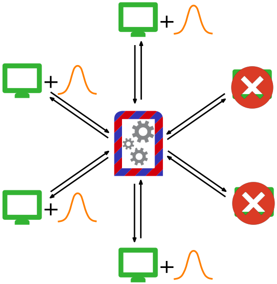


- Easy to implement and **efficient**
- Easy to analyze even for complex models
- Privacy guarantee grows with s^2
- **Great ...**

Differential privacy & Noise injection mechanisms

Gaussian mechanism: Worker i computes and sends a noisy gradient

$$\tilde{g}_t^{(i)} := g_t^{(i)} + \mathcal{N}\left(0, s^2 I_d\right) \; ; \text{ Balle and Wang (2018)}$$



- Easy to implement and **efficient**
- Easy to analyze even for complex models
- Privacy guarantee grows with s^2
- **Great ...**

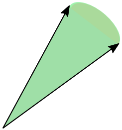
... but does this combine well with Byzantine learning?

Byzantine learning and privacy do not trivially combine

By definition privacy make uncertainty grow:

$$\mathbb{E}\left[\tilde{\boldsymbol{g}}_t^{(i)} - \mathbb{E}\left[\tilde{\boldsymbol{g}}_t^{(i)}\right]\right] \leq \sigma^2$$

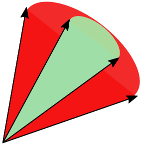
Range of plausible gradients for an honest worker (before noise injection)



Byzantine learning and privacy do not trivially combine

By definition privacy make uncertainty grow:

$$\mathbb{E}\left[\tilde{g}_t^{(i)} - \mathbb{E}\left[\tilde{g}_t^{(i)}\right]\right] \leq \sigma^2 + d \times s^2$$



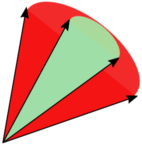
 Range of plausible gradients for an honest worker (before noise injection)

 Range of plausible gradients for an honest worker (after noise injection)

Byzantine learning and privacy do not trivially combine

By definition privacy make uncertainty grow:

$$\mathbb{E}\left[\tilde{g}_t^{(i)} - \mathbb{E}\left[\tilde{g}_t^{(i)}\right]\right] \leq \sigma^2 + d \times s^2$$



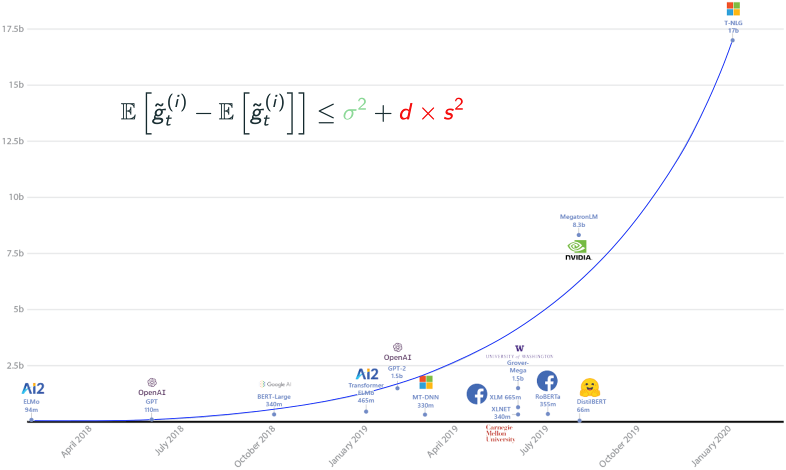
Range of plausible gradients for an honest worker (before noise injection)

\+ Range of plausible gradients for an honest worker (after noise injection)

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
$\rightarrow$ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

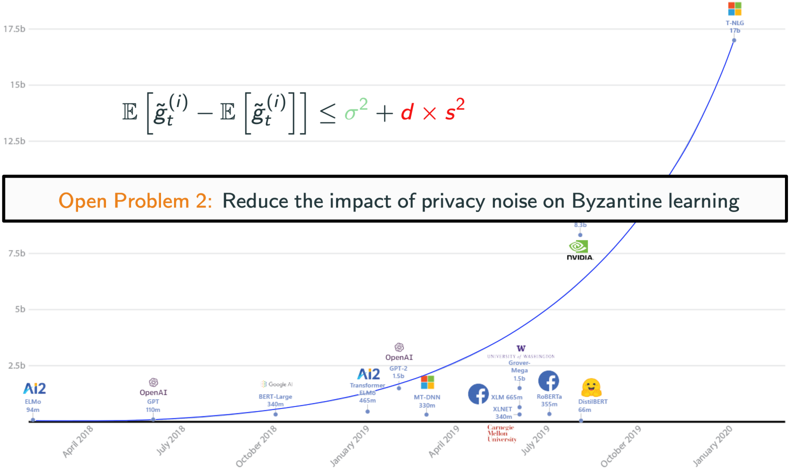
Model size grows exponentially in modern day ML

$$\mathbb{E}\left[\tilde{g}_t^{(i)} - \mathbb{E}\left[\tilde{g}_t^{(i)}\right]\right] \leq \sigma^2 + d \times s^2$$



Based on: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/

Model size grows exponentially in modern day ML



$$\mathbb{E}\left[\tilde{g}_t^{(i)} - \mathbb{E}\left[\tilde{g}_t^{(i)}\right]\right] \leq \sigma^2 + d \times s^2$$

Open Problem 2: Reduce the impact of privacy noise on Byzantine learning

Based on: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/

Avenues for combining Byzantine learning and differential privacy

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
$\rightarrow$ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
→ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Recent developments:

- (α, f)-Byzantine resilience is **too stringent to be met** Karimireddy et al. (2021)
- **Alternative definition** with tightened analysis Farhadkhani et al. (2022)

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
→ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Recent developments:

- (α, f)-Byzantine resilience is **too stringent to be met** Karimireddy et al. (2021)
- **Alternative definition** with tightened analysis Farhadkhani et al. (2022)

  → We believe this definition to better combine with privacy (ongoing work)

Avenues for combining Byzantine learning and differential privacy

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
$\rightarrow$ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
$\rightarrow$ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Recent developments:

- (α, f)-Byzantine resilience is **too stringent to be met** Karimireddy et al. (2021)
- **Alternative definition** with tightened analysis Farhadkhani et al. (2022)

Avenue 1: improving Byzantine learning theory

Injecting noise to get **privacy** makes Byzantine resilience **much harder**
$\rightarrow$ (α, f)-Byzantine resilience in Guerraoui et al. (2021)

Also shows that $(\alpha, f)-$Byzantine resilience tends to **overrate** the impact of noise

Recent developments:

- (α, f)-Byzantine resilience is **too stringent to be met** Karimireddy et al. (2021)
- **Alternative definition** with tightened analysis Farhadkhani et al. (2022)

  $\rightarrow$ We believe this definition to better combine with privacy (ongoing work)

Avenues (2 & 3): adapt the system to overcome the issue

Dimensionality reduction:

- Use compression/dimensionality reduction to have smaller model size d
- Use **coordinate-wise gradient descent** to reduce the size of effective gradients Damaskinos et al. (2021) and Mangold et al. (2022)

Avenues (2 & 3): adapt the system to overcome the issue

Dimensionality reduction:

- Use compression/dimensionality reduction to have smaller model size d
- Use **coordinate-wise gradient descent** to reduce the size of effective gradients Damaskinos et al. (2021) and Mangold et al. (2022)

Rebuild some trust in the parameter-server:

- Use new hardware/system architecture to **enforce verifiable computing**
- With a **trusted server**, we can relate the problem to robust statictics where combining robustness and privacy is **much easier** Dwork and Lei (2009)

References

Balle, B. and Wang, Y.-X. (2018). Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In Dy, J. and Krause, A., editors, Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pages 394–403. PMLR.

Damaskinos, G., Mendler-Dünner, C., Guerraoui, R., Papandreou, N., and Parnell, T. (2021). Differentially private stochastic coordinate descent. Proceedings of the AAAI Conference on Artificial Intelligence, 35(8):7176–7184.

Dwork, C. and Lei, J. (2009). Differential privacy and robust statistics. In Proceedings of the forty-first annual ACM symposium on Theory of computing, pages 371–380.

Dwork, C., Roth, A., et al. (2014). The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 9(3-4):211–407.

Farhadkhani, S., Guerraoui, R., Gupta, N., Pinot, R., and Stephan, J. (2022). Byzantine machine learning made easy by resilient averaging of momentums. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S., editors, International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, volume 162 of Proceedings of Machine Learning Research, pages 6246–6283. PMLR.

Guerraoui, R., Gupta, N., Pinot, R., Rouault, S., and Stephan, J. (2021). Differential privacy and byzantine resilience in sgd: Do they add up? In Proceedings of the 2021 ACM Symposium on Principles of Distributed Computing, PODC'21, page 391–401, New York, NY, USA. Association for Computing Machinery.

Karimireddy, S. P., He, L., and Jaggi, M. (2021). Learning from history for byzantine robust optimization. International Conference On Machine Learning, Vol 139, 139.

Lamport, L., Shostak, R., and Pease, M. (1982). The byzantine generals problem. ACM Trans. Program. Lang. Syst., 4(3):382–401.

Mangold, P., Bellet, A., Salmon, J., and Tommasi, M. (2022). Differentially private coordinate descent for composite empirical risk minimization. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S., editors, International Conference on Machine

Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, volume 162 of Proceedings of Machine Learning Research, pages 14948–14978. PMLR.

Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In Proceedings of EUROCRYPT, pages 223–238.

Zhu, L., Liu, Z., and Han, S. (2019). Deep leakage from gradients. In Wallach, H., Larochelle, H., Beygelzimer, A., d Alché-Buc, F., Fox, E., and Garnett, R., editors, Advances in Neural Information Processing Systems 32, pages 14774–14784. Curran Associates, Inc.