Part IV

Advanced Topics

11 Privacy

One of the motivating factors for developing distributed algorithms for optimization and learning was the desire to avoid the exchange of raw data due to privacy concerns. Indeed, all distributed algorithms we encountered so far, ranging from federated to decentralized architectures, exchange intermediate estimates in lieu of raw data. One may then conjecture that all distributed algorihtms fully preserve the privacy of participating agents, since no data is exchanged. As we will see in this chapter, this is more nuanced than it may seem at first side. In order to make inference about private information (such as local data), it is sufficient to observe *correlated* information (such as models).

We must hence develop a more rigorous framework for quantifying privacy and the leakage of information. We will use *differential privacy* as a privacy notion. This concept was developed by **Cynthia Dwork** and co-authors in a sequence of works in the 2000's, and has since gained significant traction in both academia and industry. Differential privacy was originally developed in the context of probing databases, rather than a learning context, and hence we will adapt concepts and notation to our setting.

11.1 LEAKAGE OF PRIVATE INFORMATION

Let us consider a simple example to illustrate how information can leak in a distributed setting where no raw data is exchanged. Recall the centralized stochastic gradient algorithm of Chapter 4, where a fusion center sends the current model $\boldsymbol{w}_{i-1}$ to all agents. Each agent then performs a local update using the gradient approximation of its local cost function:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{i-1} - \mu\widehat{\nabla J}_k(\boldsymbol{w}_{i-1}) \tag{11.1}$$

The locally updated models are sent back to the fusion center, where they are aggregated according to:

$$\boldsymbol{w}_i = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{\psi}_{k,i} \tag{11.2}$$

Note that from (11.1), the parameter server can reconstruct the local gradient approximation via:

$$\widehat{\nabla J}_k(\boldsymbol{w}_{i-1}) = \frac{\boldsymbol{w}_{i-1} - \boldsymbol{\psi}_{k,i}}{\mu} \tag{11.3}$$

Similarly, if we implement the gradient-exchange variant (4.37)–(4.16) of the algorithm, the server will immediately have access to the gradient approximation $\widehat{\nabla J}_k(\boldsymbol{w}_{i-1})$. This means that observing the evolution of models over time allows a parameter server (or someone intercepting the message) to reconstruct the locally constucted gradient approximations. These in turn are generally correlated with the data itself. For the mean squared error, for example, where:

$$J_k(w) = \mathbb{E}Q(w;\boldsymbol{x}_k) = \mathbb{E}\frac{1}{2}\left(\boldsymbol{\gamma}_k - \boldsymbol{h}_k^{\mathsf{T}}w\right)^2 \tag{11.4}$$

we have:

$$\widehat{\nabla J}_k^{\,\mathrm{ord}}(w) = \nabla Q(w;\boldsymbol{x}_k) = -\boldsymbol{h}_k\left(\boldsymbol{\gamma}_k - \boldsymbol{h}_k^{\mathsf{T}}w\right) \tag{11.5}$$

For the logistic regression problem on the other hand, where:

$$J_k(w) = \mathbb{E}Q(w;\boldsymbol{x}_k) = \mathbb{E}\ln\left(1 + e^{-\boldsymbol{\gamma}_k\boldsymbol{h}_k^{\mathsf{T}}w}\right) + \frac{\rho}{2}\|w\|^2 \tag{11.6}$$

we have:

$$\widehat{\nabla J}_k^{\,\mathrm{ord}}(w) = \nabla Q(w;\boldsymbol{x}_k) = \rho w - \frac{\boldsymbol{h}_k}{1 + e^{\boldsymbol{\gamma}_k\boldsymbol{h}_k^{\mathsf{T}}w}} \tag{11.7}$$

We observe that in both cases, an observer with access to the model w and gradient approximation $\widehat{\nabla J}_k^{\,\mathrm{ord}}(w)$ is able to reconstruct the feature vector $\boldsymbol{h}_k$ up to a scaling factor $\boldsymbol{\gamma}_k - \boldsymbol{h}_k^{\mathsf{T}}w$ in the case of the mean squared error cost, and a factor of $1/1 + e^{\boldsymbol{\gamma}_k\boldsymbol{h}_k^{\mathsf{T}}w}$ in the case of logistic regression.

These observations imply that simply sharing intermediate estimates does not guarantee privacy by default. More care needs to be taken in quantifying *exactly* how much privacy is lost by participating in a distributed learning scheme. Are certain architectures more private than others? Is there something we can do if we are unhappy with the level of privacy inherent in a particular algorithm? As we will see, the framework of differential privacy will allow us to address both of these questions.

11.2 DIFFERENTIAL PRIVACY

Let us consider a collection of K agents indexed by k. Each agent has access to private data, which we model through the random variable $\boldsymbol{x}_k$. Any agent will have the option of participating in collaborative effort, which we describe generically as $\mathrm{Alg}\,(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K)$. Consider also an alternative scenario, where

an arbitrary agent, let us say without loss of generality, agent 1, refuses to participate in the learning protocol, and is replaced by some other agent $1'$ with different local data $\boldsymbol{x}_{1'} \in \mathcal{X}$ from a set of permissible local data sets $\mathcal{X}$. We can then state the formal definition of ϵ-differential privacy.

DEFINITION 11.1 (ϵ-**differential privacy**). We say that an algorithm $\mathcal{A}\}(\cdot)$ is ϵ-differentially private, if it holds that:

$$\frac{f\left(\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\text{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} \leq e^{\epsilon} \tag{11.8}$$

for all $\boldsymbol{x}_{1'} \in \mathcal{X}$, where $f(\cdot)$ denotes the probability density function of the argument. □

For small ϵ it holds that $e^{\epsilon} \approx 1 - \epsilon \approx 1$, and hence the condition of ϵ-differential privacy essentially ensures that the distribution of the output of the algorithm is essentially unaffected by the presence of agent 1. We conclude from this definition, that if ϵ is small, agent 1 does not relinquish much privacy by participating in the learning protocol, since the output of the algorithm is largely unaffected by its participation. In other words, agent 1 by partcipating does not reveal particluarly unique information that would allow an observer of the output $f\left(\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)$ to identify the presence of agent 1. A useful corollary of this fact is that if the local data sets happen to be perfectly homogeneous, we can expect little privacy loss.

Example 11.1 (**Homogeneous agents**) Suppose the local data distributions are i.i.d, meaning $\boldsymbol{x}_1 \sim \boldsymbol{x}_2 \sim \ldots \sim \boldsymbol{x}_K \sim \boldsymbol{x}_{1'}$. It then follows that:

$$f\left(\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right) = f\left(\text{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right) \tag{11.9}$$

and hence the procedue is 0-differentially private. This essentially formalized the fact that if all information provided by agents is common knowledge, there is no privacy loss incurred by any given agent's partciipation.

Example 11.2 (**Randomized response**) Let us consider a scenario where $\boldsymbol{x}_k$ contains potentially embarassing information, such as the answer to the question "Have you comitted a crime?". For agents who have comitted a crime, we will then have $\boldsymbol{x}_k = 1$ with probability one, and for those who have not we will have $\boldsymbol{x}_k = -1$ with probability one. Since not all agents have comitted a crime, and hence the $\boldsymbol{x}_k$ is not identical, agents may be hesitant to reveal this private information. We may then envision the following randomized response mechanism and algorithm for estimating the rate of criminals among the collection of agents:

- Each agent k flips a fair coin.

  - If the coin falls on heads, the agent responds truthfully, i.e., returns $\boldsymbol{m}_k = \boldsymbol{x}_k$.
  - If the coin falls on tails, the agent flips another coin and returns $\boldsymbol{m}_k = 1$ on heads, and $\boldsymbol{m}_k = -1$ on tails.
- The server aggregates the messages and estimates the rate of criminals as from $\{\boldsymbol{m}_k\}_{k=1}^K$.

We can show that the above mechanism is differentially private. The source of privacy protection here arises from the plausible deniability, which in turn is the result of the randomized response mechanism. If an agent answers $\boldsymbol{m}_k = 1$, this may be because it has commited a crime, but it also be merely as a result of flipping two heads in a row, which occurs with probability $\frac{1}{4}$. More formally, we have:

$$\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right) = \{\boldsymbol{m}_k\}_{k=1}^K \tag{11.10}$$

and

$$\begin{aligned}\frac{f\left(\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\text{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} &= \frac{f\left(\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_K\right)}{f\left(\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_K\right)} \\ &\overset{(a)}{=} \frac{f\left(\boldsymbol{m}_1\right) f\left(\boldsymbol{m}_2\right) \cdots f\left(\boldsymbol{m}_K\right)}{f\left(\boldsymbol{m}_{1'}\right) f\left(\boldsymbol{m}_2\right) \cdots f\left(\boldsymbol{m}_K\right)} \\ &= \frac{f\left(\boldsymbol{m}_1\right)}{f\left(\boldsymbol{m}_{1'}\right)}\end{aligned} \tag{11.11}$$

where (a) follows since the messages are generated independently at each agent. Then, if $\boldsymbol{x}_1 = \boldsymbol{x}_{1'}$, it is immediate that $\frac{f(\boldsymbol{m}_1)}{f(\boldsymbol{m}_{1'})} = 1$. For the other cases, we compute:

$$\Pr\left\{\boldsymbol{m}_1 = 1 | \boldsymbol{x}_1 = 1\right\} = \Pr\left\{\boldsymbol{m}_{1'} = 1 | \boldsymbol{x}_{1'} = 1\right\} = \frac{3}{4} \tag{11.12}$$

$$\Pr\left\{\boldsymbol{m}_1 = 1 | \boldsymbol{x}_1 = 0\right\} = \Pr\left\{\boldsymbol{m}_{1'} = 1 | \boldsymbol{x}_{1'} = 0\right\} = \frac{1}{4} \tag{11.13}$$

$$\Pr\left\{\boldsymbol{m}_1 = 0 | \boldsymbol{x}_1 = 1\right\} = \Pr\left\{\boldsymbol{m}_{1'} = 0 | \boldsymbol{x}_{1'} = 1\right\} = \frac{1}{4} \tag{11.14}$$

$$\Pr\left\{\boldsymbol{m}_1 = 0 | \boldsymbol{x}_1 = 0\right\} = \Pr\left\{\boldsymbol{m}_{1'} = 0 | \boldsymbol{x}_{1'} = 0\right\} = \frac{3}{4} \tag{11.15}$$

It follows that $\frac{1}{3} \leq \frac{f(\boldsymbol{m}_1)}{f(\boldsymbol{m}_{1'})} \leq 3$, and hence:

$$\frac{f\left(\text{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\text{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} \leq 3 = e^{\ln 3} \tag{11.16}$$

We conclude that the mechanism is $\ln 3$-differentially private. □

11.2.1 Laplace Mechanism

The randomized response mechanism we encountered in the previous section was useful, as it provides indication that adding randomness to messages can

result in quantifiable privacy gain (in the sense of differential privacy). There are two remaining limitations that need to be addressed before we can apply this concept to learning algorithms. First, we would like to be able to control the level of privacy ϵ, rather than simply accepting $\ln 3$-differential privacy. Second, the randomized response mechanism is natural when dealing with binary (yes or no) questins, but while learning, we encounter continuous, vector-valued quantities. To this end, we introduce the Laplace mechanism.

DEFINITION 11.2 (ℓ_1-**sensitivity**). The sensitivity of $\mathrm{Alg}(\cdot)$ is defined as:

$$\Delta = \max_{\boldsymbol{x}_{1'} \in \mathcal{X}} \|\mathrm{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right) - \mathrm{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\|_1 \tag{11.17}$$

□

LEMMA 11.1 (**Laplace mechanism**). *Suppose* $\mathrm{Alg}(\cdot)$ *has* ℓ_1*-sensitivity* Δ*, and define:*

$$\mathrm{LAlg}(\cdot) = \mathrm{Alg}(\cdot) + \boldsymbol{v}_p \tag{11.18}$$

where $\boldsymbol{v}_p$ *is a vector of suitable dimension* M_v*, where each entry follows the Laplace distribution:*

$$f_{\boldsymbol{v}_p}(v) = \frac{1}{(2b_v)^{M_v}} e^{-\frac{\|v\|_1}{b_v}} \tag{11.19}$$

Then, $\mathrm{LAlg}(\cdot)$ *is* $\left(\frac{\Delta}{b_v}\right)$*-differentially private.*

Proof: We have:

$$\begin{aligned}
\frac{f\left(\mathrm{LAlg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\mathrm{LAlg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} &= \frac{e^{-\frac{\|\mathrm{LAlg}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K) - x\|}{b_v}}}{e^{-\frac{\|\mathrm{LAlg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right) - x\|}{b_v}}} \\
&= e^{\frac{\|\mathrm{LAlg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right) - x\| - \|\mathrm{LAlg}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K) - x\|}{b_v}} \\
&\overset{(a)}{\leq} e^{\frac{\|\mathrm{LAlg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right) - \mathrm{LAlg}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K)\|}{b_v}} \\
&\overset{(11.17)}{\leq} e^{\frac{\Delta}{b_v}}
\end{aligned} \tag{11.20}$$

□

It follows that each base-algorithm with bounded sensitivity Δ can be *privatized* via the addition of Laplacian noise to the output. The level of noise necessary is related to the sensitivity Δ of the algorithm and the desired level of privacy ϵ.

11.2.2 Repeated Probing

Most algorithms for distributed optimization and learning we have encountered so far have been iterative, meaning that we repeatedly sample and compute functions of the local data, and repeatedly exchange model estimates. We hence

need to quantify as well the effect of these repeated interactions on privacy. It turns out that differential privacy is quite amenable to such settings.

LEMMA 11.2 (**Multiple algorithm evaluations**). *Suppose* $\mathrm{Alg}_1(\cdot)$ *is* ϵ_1*-differentially private,* $\mathrm{Alg}_2(\cdot)$ *is* ϵ_2*-differentially private, and the evaluations are independent. It then follows that:*

$$\mathrm{Alg}\,(\cdot) = \mathrm{col}\,\{\mathrm{Alg}_1(\cdot), \mathrm{Alg}_2(\cdot)\} \tag{11.21}$$

is $(\epsilon_1 + \epsilon_2)$*-differentially private.*

Proof: We have:

$$\begin{aligned}
\frac{f\left(\mathrm{Alg}\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\mathrm{Alg}\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} &= \frac{f\left(\mathrm{Alg}_1\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right), \mathrm{Alg}_2\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\mathrm{Alg}_1\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right), \mathrm{Alg}_2\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} \\
&= \frac{f\left(\mathrm{Alg}_1\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right) \cdot f\left(\mathrm{Alg}_2\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)}{f\left(\mathrm{Alg}_1\left(\boldsymbol{x}_{1'}, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right) \cdot f\left(\mathrm{Alg}_2\left(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_K\right)\right)} \\
&\leq e^{\epsilon_1} \cdot e^{\epsilon_2} = e^{\epsilon_1 + \epsilon_2}
\end{aligned} \tag{11.22}$$

□

11.3 PRIVATE DISTRIBUTED LEARNING

For simplicity, we will illustrate private distributed learning for the diffusion algorithm (8.62)–(8.63) from Chapter 8, though analogous privacy considerations can be applied to other algorithms. Let us repeat the diffusion algorithm for reference:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) \tag{11.23}$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell,i} \tag{11.24}$$

As described when motivating the privacy developments in this chapter, agents may be concerned to share their models $\boldsymbol{\psi}_{k,i}$, as an eavesdropper or malicious agent may leverage these to reconstruct the gradient approximation $\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$, which will in turn reveal information about the locally available data $\boldsymbol{x}_k$ at agent k. Inspired by the Laplace mechanism of Section 11.2.1, we can then consider the following variant:

$$\boldsymbol{\phi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) \tag{11.25}$$

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{\phi}_{k,i-1} + \boldsymbol{v}_{k,i} \tag{11.26}$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell,i} \tag{11.27}$$

where $\boldsymbol{v}_{k,i}$ follows a Laplace distribution as Lemma 11.1. We will present a theorem further below which establishes a privacy guarantee for this kind of construction. Before we proceed, however, note that if we combine (11.25) and (11.26),

we obtain after some reformulations:

$$\boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu\left(\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) - \frac{\boldsymbol{v}_{k,i}}{\mu}\right) \tag{11.28}$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell,i} \tag{11.29}$$

We can then define the privatized gradient approximation:

$$\widehat{\nabla J}_k^{\,\text{priv}}(\boldsymbol{w}_{k,i-1}) \triangleq \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) - \frac{\boldsymbol{v}_{k,i}}{\mu} \tag{11.30}$$

This adjusted gradient approximation satisfies all conditions of Chapter (3). However, in this construction, the gradient approximation $\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$ is perturbed by the privacy noise $\boldsymbol{v}_{k,i}$ divided by the step-size μ. For small step-sizes this can result in serious amplification of the privacy noise component, yielding:

$$\sigma^2_{k,\text{priv}} = \sigma_k^2 + \frac{\sigma_v^2}{\mu^2} \tag{11.31}$$

where σ_k^2 denotes the absolute component of the gradient approximation $\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$ and σ_v^2 denotes the variance of the Laplacian privacy noise $\boldsymbol{v}_{k,i}$. We can then conclude from Theorem 9.1, that the limiting performance of the privatized diffusion algorithm (for small step-sizes) will be given by:

$$\lim_{i\to\infty} \mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|^2 = O(\mu\sigma^2) + O\left(\mu^{-1}\sigma_v^2\right) \tag{11.32}$$

We note that the privacy preserving noise added to the distributed recursion as a very detrimental effect on the limiting performance of the algorithm. This begs the question of whether we can do any better. After all, the fact that models are computed and exchanged in a decentralized manner should give us some freedom to obtain improved privacy.

11.3.1 Graph-Homomorphic Perturbations

The idea here will be to tune perturbations to the network topology in order to minimize their impact on learning performance while preserving privacy. The first step is to allow

$$\boldsymbol{\phi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) \tag{11.33}$$

$$\boldsymbol{\psi}_{k\ell,i} = \boldsymbol{\phi}_{k,i-1} + \boldsymbol{v}_{k\ell,i} \tag{11.34}$$

$$\boldsymbol{w}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{\psi}_{\ell k,i} \tag{11.35}$$

Note that agent k in this construction sends different models $\boldsymbol{\psi}_{k\ell,i}$ to different neighbors, each perturbed by a different privacy noise $\boldsymbol{v}_{k\ell,i}$. For the network

centroid, we have:

$$\begin{aligned}
\boldsymbol{w}_{c,i} &\triangleq \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{w}_{k,i} \\
&\overset{(11.35)}{=} \frac{1}{K}\sum_{k=1}^{K}\sum_{\ell=1}^{K}a_{\ell k}\boldsymbol{\phi}_{\ell,i} + \frac{1}{K}\sum_{k=1}^{K}\sum_{\ell=1}^{K}a_{\ell k}\boldsymbol{v}_{\ell k,i} \\
&= \frac{1}{K}\sum_{\ell=1}^{K}\left(\sum_{k=1}^{K}a_{\ell k}\right)\boldsymbol{\phi}_{\ell,i} + \frac{1}{K}\sum_{\ell=1}^{K}\sum_{k=1}^{K}a_{\ell k}\boldsymbol{v}_{\ell k,i} \\
&\overset{(a)}{=} \frac{1}{K}\sum_{\ell=1}^{K}\boldsymbol{\phi}_{\ell,i} + \frac{1}{K}\sum_{\ell=1}^{K}\sum_{k=1}^{K}a_{\ell k}\boldsymbol{v}_{\ell k,i} \\
&\overset{(11.33)}{=} \boldsymbol{w}_{c,i-1} - \frac{\mu}{K}\sum_{\ell=1}^{K}\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) + \frac{1}{K}\sum_{\ell=1}^{K}\sum_{k=1}^{K}a_{\ell k}\boldsymbol{v}_{\ell k,i}
\end{aligned} \tag{11.36}$$

where (a) follows for symmetric adjacency matrices. We saw in Chapter 9 that the network centroid, in many cases describes the dominant dynamics of the network of learners. It is then reasonable to construct perturbations $\boldsymbol{v}_{\ell k,i}$ so their impact on the centroid $\boldsymbol{w}_{c,i}$ is minimized. This motivates the following definition:

DEFINITION 11.3 (**Graph-Homomorphic Perturbations**). A set of perturbations $\boldsymbol{v}_{\ell k,i}$ is homomorphic for the the graph defined by the adjacency matrix $A \triangleq [a_{\ell k}]$ if it holds with probability one that:

$$\frac{1}{K}\sum_{\ell=1}^{K}\sum_{k=1}^{K}a_{\ell k}\boldsymbol{v}_{\ell k,i} = 0 \tag{11.37}$$

While other constructions are possible, we present here a simple construction, which can be implemented locally and independently at every agent k.

LEMMA 11.3 (**Constructing Graph-Homomorphic Perturbations**). *Let each agent ℓ sample independently from the Laplace distribution $\boldsymbol{v}'_{\ell,i} \sim \text{Lap}\,(0, b_v)$ with variance $\sigma_v^2 = 2b_v^2$. Then, the construction:*

$$\boldsymbol{v}_{\ell k,i} = \begin{cases} \boldsymbol{v}'_{\ell,i}, & \text{if } k \in \mathcal{N}_\ell \text{ and } k \neq \ell, \\ -\frac{1-a_{\ell\ell}}{a_{\ell\ell}}\boldsymbol{v}'_{\ell,i}, & \text{if } k = \ell. \end{cases} \tag{11.38}$$

is homomorphic for the graph described by the symmetric adjacency matrix $A = A^{\mathsf{T}}$.

Proof: The result can be verified immediately by substitution. □

It is important to note that graph-homomorphic perturbations are not completely unimpactful to the recursion, since the network centroid recursion (11.36) is driven by gradient approximations evaluated at $\boldsymbol{w}_{k,i-1}$, rather than $\boldsymbol{w}_{c,i-1}$.

As a result, the privacy perturbation will affect the learning dynamics through the network disagreement, rather than the centroid directly. This will result in reduced, albeit non-zero effect of the privacy pertubrations. Adjusting the arguments of Chapter 9 to allow for privacy parturbation of this manner is fairly straightforward, resulting in:

$$\lim_{i\to\infty} \mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|^2 = O(\mu\sigma^2) + O\left(\sigma_v^2\right) \tag{11.39}$$

Comparing the two performance expressions(11.32) and (11.39) we note that the effect of the privacy perturbations in is reduced by a factor of μ, and now constant for small step-sizes. This results in improved performance for a given level of privacy.

11.3.2 Differential Privacy Guarantee for Privatized Diffusion

We motivated the privatized diffusion implementations through the Laplacian mechanism, but have not yet established an actual privacy guarantee. We will provide a sketch of the proof under the simplifying assumption that gradient approximations are bounded almost surely:

$$\|\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})\| \leq G \tag{11.40}$$

This condition is in fact not necessary, and can be removed at the expense of more cumbersome privacy analysis. We refer the reader to [Rizk, Vlaski, Sayed 2023 [1]] for this.

Let us now consider the setting, where agent 1 has decided not to volunteer its private information for the diffusion of information, and its data $\boldsymbol{x}_1$ is replaced by some other data $\boldsymbol{x}_{1'}$, following a different distribution. In this setting, implementing the perturbed diffusion recursions, would naturally result in a different learning trajectory $\boldsymbol{w}'_{k,i}$ at every agent k, since the data $\boldsymbol{x}_{1'}$ propagates through the gradient approximations and the diffusion of estimates through the entire network. We can then define and bound the sensitivity of the diffusion algorithms as.

LEMMA 11.4 (**Sensitivity of the diffusion algorithm**). *The distance between the trajectories $\boldsymbol{w}_{k,i}$ and $\boldsymbol{w}'_{k,i}$ is bounded with probability one by:*

$$\Delta(i) \triangleq \max_k \|\boldsymbol{w}_{k,i} - \boldsymbol{w}'_{k,i}\| \leq \mu 2Gi \tag{11.41}$$

Proof: For reference, we repeat:

$$\boldsymbol{w}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\left(\boldsymbol{w}_{\ell,i-1} - \mu\widehat{\nabla J}_\ell(\boldsymbol{w}_{\ell,i-1})\right) + \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{v}_{\ell k,i} \tag{11.42}$$

$$\boldsymbol{w}'_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\left(\boldsymbol{w}'_{\ell,i-1} - \mu\widehat{\nabla J}_\ell(\boldsymbol{w}'_{\ell,i-1})\right) + \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{v}_{\ell k,i} \tag{11.43}$$

[1] E. Rizk, S. Vlaski, A. H. Sayed, "Enforcing Privacy in Distributed Learning with Performance Guarantees", available as arXiv:2301.06412, 2023.

Where $\boldsymbol{x}'_{\ell,i}$ differs from $\boldsymbol{x}_{\ell,i}$ only for one agent, say agent 1, and $\boldsymbol{w}'_{k,i}$ differs from $\boldsymbol{w}_{k,i}$ as a result of the propagation of the alternative data $\boldsymbol{x}'_{1,i}$ through $\widehat{\nabla J}_\ell(\boldsymbol{w}'_{\ell,i-1})$. The sensitivity of the algorithm is then defined as the deviation of $\boldsymbol{w}'_{k,i}$ from $\boldsymbol{w}_{k,i}$, and measures the influence that agent 1 has on the evolution of the algorithm. A higher sensitivity is generally associated with a need for larger privacy perturbations, since a larger influence needs to be masked. We can bound:

$$
\begin{aligned}
&\|\boldsymbol{w}_{k,i} - \boldsymbol{w}'_{k,i}\| \\
&= \Bigg\| \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \Big(\boldsymbol{w}_{\ell,i-1} - \mu\widehat{\nabla J}_\ell(\boldsymbol{w}_{\ell,i-1}) \Big) \\
&\quad - \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \Big(\boldsymbol{w}'_{\ell,i-1} - \mu\widehat{\nabla J}_\ell(\boldsymbol{w}'_{\ell,i-1}) \Big) \Bigg\| \\
&\overset{(b)}{\leq} \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \|\boldsymbol{w}_{\ell,i-1} - \mu\widehat{\nabla J}_\ell(\boldsymbol{w}_{\ell,i-1}) \\
&\qquad\qquad - \boldsymbol{w}'_{\ell,i-1} + \mu\widehat{\nabla J}_\ell(\boldsymbol{w}'_{\ell,i-1})\| \\
&\overset{(c)}{\leq} \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \left\|\boldsymbol{w}_{\ell,i-1} - \boldsymbol{w}'_{\ell,i-1}\right\| \\
&\quad + \mu \sum_{\ell\in\mathcal{N}_k} a_{\ell k} \left\|\widehat{\nabla J}_\ell(\boldsymbol{w}_{\ell,i-1}) - \widehat{\nabla J}_\ell(\boldsymbol{w}'_{\ell,i-1})\right\| \\
&\overset{(d)}{\leq} \left(\sum_{\ell\in\mathcal{N}_k} a_{\ell k} \right) \max_\ell \left\|\boldsymbol{w}_{\ell,i-1} - \boldsymbol{w}'_{\ell,i-1}\right\| + \mu \left(\sum_{\ell\in\mathcal{N}_k} a_{\ell k} \right) 2G \\
&= \Delta(i-1) + \mu 2G
\end{aligned}
\tag{11.44}
$$

where (b) follows from Jensen's inequality, (c) makes use of the triangle inequality and (d) follows for bounded gradient approximations. We conclude that:

$$
\Delta(i) \triangleq \max_k \|\boldsymbol{w}_{k,i} - \boldsymbol{w}'_{k,i}\| \leq \Delta(i-1) + \mu 2G \tag{11.45}
$$

and hence (11.41) after iterating with $\Delta(0) = 0$. □

Definition 11.4 (ϵ-**differential privacy of diffusion**). We say that the diffusion recursion (11.33)–(11.35) is $\epsilon(i)$-differentially private for agent 1 at time i if:

$$
\frac{f\left(\left\{ \left\{ \boldsymbol{\psi}_{1\ell,n} \right\}_{\ell\neq 1\in\mathcal{N}_1} \right\}_{n=0}^{i} \right)}{f\left(\left\{ \left\{ \boldsymbol{\psi}'_{1\ell,n} \right\}_{\ell\neq 1\in\mathcal{N}_1} \right\}_{n=0}^{i} \right)} \leq e^{\epsilon(i)} \tag{11.46}
$$

where $f(\cdot)$ denotes the probability density function and $\left\{ \left\{ \boldsymbol{\psi}_{1\ell,n} \right\}_{\ell\neq 1\in\mathcal{N}_1} \right\}_{n=0}^{i}$ collects all quantities transmitted by agent 1 to any of its neighbors during the operation of the algorithm, while excluding its local iterates $\boldsymbol{\psi}_{11,n}$, which are kept private.

Theorem 11.1 (**Privacy cost of the diffusion algorithm**). *Suppose* (11.33)–(11.35) *employs homomorphic perturbations constructed as in* (11.38). *Then, at*

time i, algorithm (11.33)–(11.35) is $\epsilon(i)$-differentially private according to (11.46), with:

$$\epsilon(i) = \mu \frac{G(i^2+i)}{b_v} \tag{11.47}$$

Proof: Under construction (11.38), every agent shares the same perturbed estimate $\boldsymbol{\psi}_{1\ell,n} = \boldsymbol{\phi}_{1,n} + \boldsymbol{v}_n$ with all of its neighbors (excluding itself), and we have:

$$\frac{f\left(\left\{\left\{\boldsymbol{\psi}_{1\ell,n}\right\}_{\ell\neq 1\in\mathcal{N}_1}\right\}_{n=0}^{i}\right)}{f\left(\left\{\left\{\boldsymbol{\psi}'_{1\ell,n}\right\}_{\ell\neq 1\in\mathcal{N}_1}\right\}_{n=0}^{i}\right)} = \frac{f\left(\left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i}\right)}{f\left(\left\{\boldsymbol{\phi}'_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i}\right)} \tag{11.48}$$

We construct an iterative argument by induction. Suppose we have $\epsilon(i-1)$ differential privacy at time $i-1$. We can expand by Bayes' rule:

$$\begin{aligned}
&\frac{f\left(\left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i}\right)}{f\left(\left\{\boldsymbol{\phi}'_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i}\right)} \\
&= \frac{f\left(\left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)\cdot f\left(\boldsymbol{\phi}_{1,i}+\boldsymbol{v}_i|\left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)}{f\left(\left\{\boldsymbol{\phi}'_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)\cdot f\left(\boldsymbol{\phi}'_{1,i}+\boldsymbol{v}_i|\left\{\boldsymbol{\phi}'_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)} \\
&\overset{(a)}{\leq} e^{\epsilon(i-1)}\frac{f\left(\boldsymbol{\phi}_{1,i}+\boldsymbol{v}_i|\left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)}{f\left(\boldsymbol{\phi}'_{1,i}+\boldsymbol{v}_i|\left\{\boldsymbol{\phi}'_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}\right)} \\
&\overset{(b)}{\leq} e^{\epsilon(i-1)}\frac{f\left(\boldsymbol{\phi}_{1,i}+\boldsymbol{v}_i|\boldsymbol{\mathcal{F}}_{i-1}\right)}{f\left(\boldsymbol{\phi}'_{1,i}+\boldsymbol{v}_i|\boldsymbol{\mathcal{F}}'_{i-1}\right)}
\end{aligned} \tag{11.49}$$

where in (a) we used the induction hypothesis and in (b) we defined $\boldsymbol{\mathcal{F}}_{i-1} \triangleq \left\{\boldsymbol{\phi}_{1,n}+\boldsymbol{v}_n\right\}_{n=0}^{i-1}$ for brevity. Since $\boldsymbol{v}_i$ is sampled in an i.i.d. fashion, it is independent of $\boldsymbol{\mathcal{F}}_{i-1}, \boldsymbol{\mathcal{F}}'_{i-1}$ and $\boldsymbol{\phi}_{1,i}, \boldsymbol{\phi}'_{1,i}$, hence we have:

$$\begin{aligned}
\frac{f\left(\boldsymbol{\phi}_{1,i}+\boldsymbol{v}_i|\boldsymbol{\mathcal{F}}_{i-1}\right)}{f\left(\boldsymbol{\phi}'_{1,i}+\boldsymbol{v}_i|\boldsymbol{\mathcal{F}}'_{i-1}\right)} &= \frac{\frac{1}{2b_v}e^{-\frac{\|\boldsymbol{\phi}_{1,i}-v_i\|}{b_v}}}{\frac{1}{2b_v}e^{-\frac{\|\boldsymbol{\phi}'_{1,i}-v_i\|}{b_v}}} \\
&= e^{\frac{\|\boldsymbol{\phi}'_{1,i}-v_i\|-\|\boldsymbol{\phi}_{1,i}-v_i\|}{b_v}} \\
&\overset{(a)}{\leq} e^{\frac{\|\boldsymbol{\phi}'_{1,i}-\boldsymbol{\phi}_{1,i}\|}{b_v}}
\end{aligned} \tag{11.50}$$

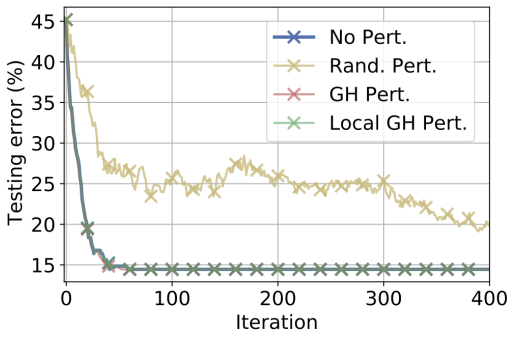
where (a) follows from $\|x\| = \|x-y+y\| \leq \|x-y\| + \|y\|$. We can then bound with probability one:

$$\begin{aligned}
&\|\boldsymbol{\phi}'_{1,i} - \boldsymbol{\phi}_{1,i}\| \\
&= \|\boldsymbol{w}'_{k,i-1} - \mu\nabla Q_k(\boldsymbol{w}'_{k,i-1};\boldsymbol{x}'_{k,i}) - \boldsymbol{w}_{k,i-1} + \mu\nabla Q_k(\boldsymbol{w}_{k,i-1};\boldsymbol{x}_{k,i})\| \\
&\leq \|\boldsymbol{w}'_{k,i-1} - \boldsymbol{w}_{k,i-1}\| + \mu\|\nabla Q_k(\boldsymbol{w}'_{k,i-1};\boldsymbol{x}'_{k,i}) - \nabla Q_k(\boldsymbol{w}_{k,i-1};\boldsymbol{x}_{k,i})\| \\
&\leq \Delta(i-1) + \mu 2G = \mu 2Gi
\end{aligned} \tag{11.51}$$

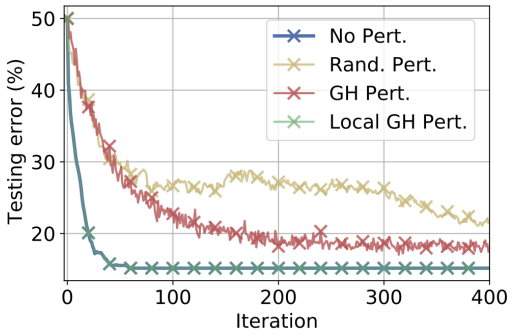
We conclude:

$$\epsilon(i) \leq \epsilon(i-1) + \frac{\mu 2Gi}{b_v} \Longrightarrow \epsilon(i) \leq \frac{\mu 2G}{b_v}\sum_{n=1}^{i} i = \mu\frac{G(i^2+i)}{b_v} \tag{11.52}$$

□



(a) Centroid testing error



(b) Average individual testing error

Figure 11.1 Performance of privatized variants of the diffusion algorithm. Taken from [Rizk, Vlaski, Sayed 2023].