Reliable and Trustworthy Artificial Intelligence

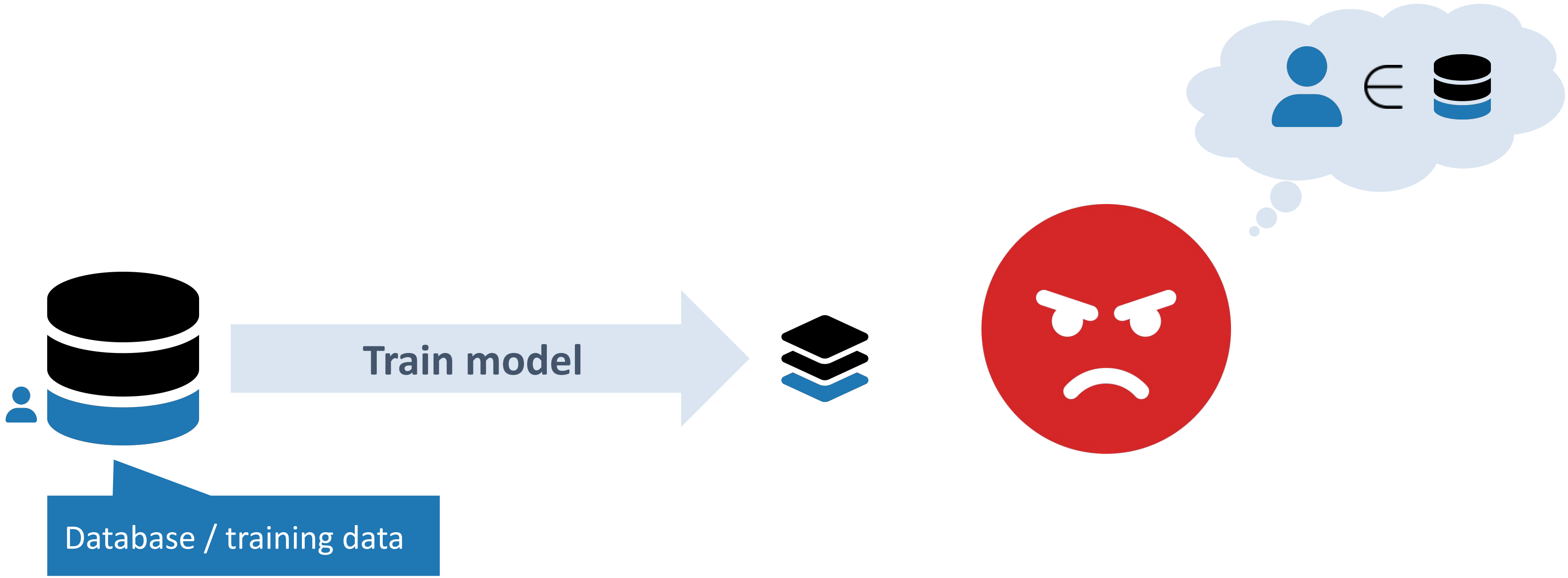
Lecture 8: Differential Privacy

Benjamin Bichsel

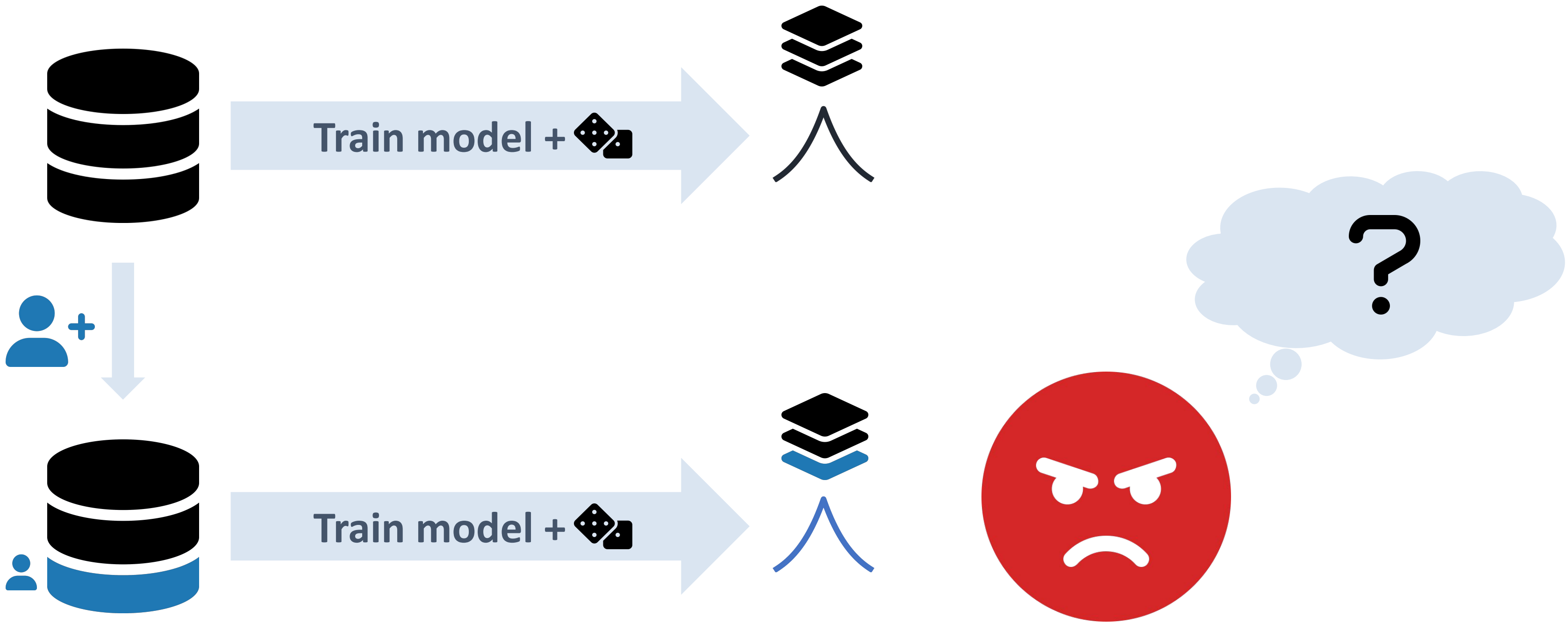
ETH Zurich

Fall 2022

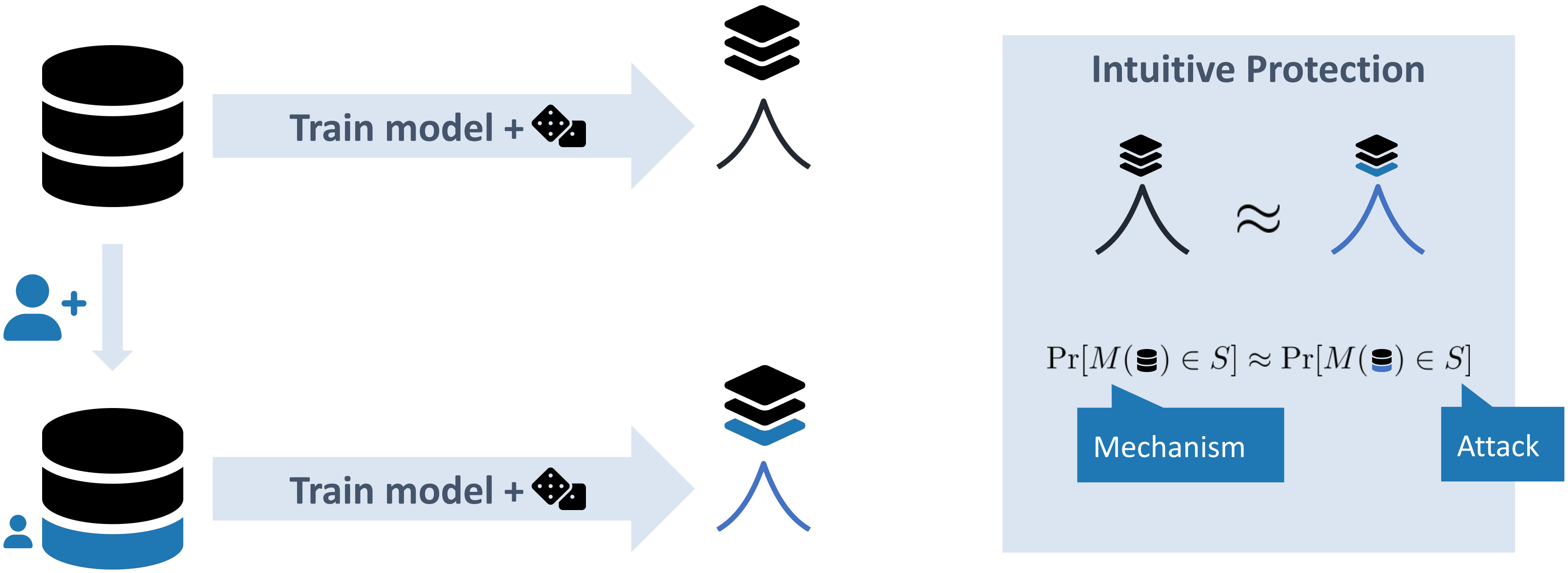
Membership Inference



Differential Privacy



Differential Privacy



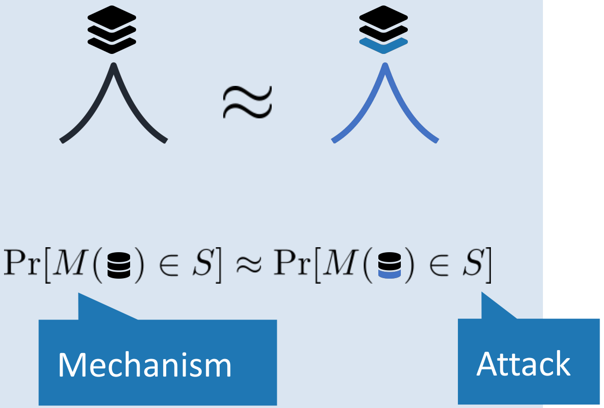
Differential Privacy

M is ε differentially private (ε-DP):

For all "neighboring" (a,a') and for every attack S:

$$\Pr[M(a) \in S] \leq e^{\epsilon} \Pr[M(a') \in S]$$

Intuitive Protection



Neighborhood

Typically symmetric

Which inputs should be indistinguishable?

Examples:

- (a,a') neighboring ⇔ adding/removing one person to/from a yields a'
- (a,a') neighboring ⇔ changing the data/features of one person in a yields a'
- (a,a') neighboring ⇔ $\|a-a'\|_p<R$

Written: (a, a') ∈ Neigh

Intuition behind Inequality

$$\Pr[M(a) \in S] \leq e^{\epsilon} \Pr[M(a') \in S]$$

$$\Pr[M(a') \in S] \leq e^{\epsilon} \Pr[M(a) \in S]$$

$$\implies \underbrace{e^{-\epsilon}}_{\approx 1-\epsilon} \Pr[M(a') \in S] \leq \Pr[M(a) \in S]$$

$$\Pr[M(a) \in S] \leq \underbrace{e^{\epsilon}}_{\approx 1+\epsilon} \Pr[M(a') \in S]$$

$$(1-\epsilon) \Pr[M(a') \in S] \lesssim \Pr[M(a) \in S] \lesssim (1+\epsilon) \Pr[M(a') \in S]$$

Example: Laplace Mechanism

Medical data

Name	Has disease (a)
Jane	1
John	1
Richard	0

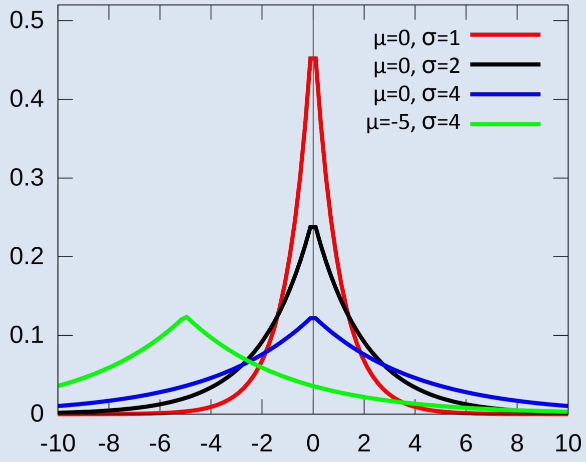
Report number of patients with disease

$$M(a) = \left(\sum_{i=1}^{n} a_i\right) + \mathrm{Lap}(0, 1/\epsilon)$$

$$(a, a') \in \mathrm{Neigh} \iff \|a - a'\|_0 \leq 1$$

Laplace distribution

$$p(\mathrm{Lap}(\mu, \sigma) = t) = \frac{1}{2\sigma} \exp\left(-\frac{|t-\mu|}{\sigma}\right)$$



Example: Laplace (Analysis)

We show: M is ε-DP

$$p(\mathrm{Lap}(\mu,\sigma)=t)=\frac{1}{2\sigma}\exp\left(-\frac{|t-\mu|}{\sigma}\right)$$

$$p(M(a)=b)\le e^{\epsilon}p(M(a')=b)$$

$$\iff \frac{1}{2\cdot 1/\epsilon}\exp\left(-\frac{|b-\sum a_i|}{1/\epsilon}\right)\le e^{\epsilon}\frac{1}{2\cdot 1/\epsilon}\exp\left(-\frac{|b-\sum a_i'|}{1/\epsilon}\right)$$

$$\iff \exp\Bigg(\underbrace{\frac{-|b-\sum a_i|+|b-\sum a_i'|}{1/\epsilon}}_{\le\frac{1}{1/\epsilon}}\Bigg)\le e^{\epsilon}$$

$$\Longleftarrow \exp\left(\frac{1}{1/\epsilon}\right)\le e^{\epsilon}$$

In exercises

$$\forall S\subseteq\mathbb{B}:\Pr[M(a)\in S]\le e^{\epsilon}\Pr[M(a')\in S]$$

$$\iff$$

$$\forall b\in\mathbb{B}:\underbrace{\Pr[M(a)=b]}_{\text{or: } p(M(a)=b)}\le e^{\epsilon}\underbrace{\Pr[M(a')=b]}_{\text{or: } p(M(a')=b)}$$

density

$$-\left|b-\sum a_i\right|+\left|b-\sum a_i'\right|$$
$$=\left|b-\sum a_i'\right|-\left|b-\sum a_i\right|$$
$$\le\left|b-\sum a_i'-(b-\sum a_i)\right|$$
$$=\left|\sum a_i-\sum a_i'\right|_1$$
$$\le 1$$

Reverse triangle inequality

Laplace and Sensitivity

Note: Also works for vector outputs (add noise elementwise)

Theorem: Laplace Mechanism

$f(a) + \text{Lap}(0, \Delta_1/\varepsilon)$ is ε-DP

Sensitivity: Largest possible effect of changing input on output in L1 norm

$$\Delta_1 = \max_{(a,a')\in\text{Neigh}} \|f(a) - f(a')\|_1$$

Generalization: (ε,δ)-DP

M is (ε,δ)-DP iff:

For all "neighboring" (a,a') and for every attack S:

$$\Pr[M(a) \in S] \leq e^{\epsilon} \Pr[M(a') \in S] + \delta$$

Absolute difference in probabilities (vs relative)

Allows support of distributions to differ

Theorem: Gaussian Mechanism is DP

$f(a) + \mathcal{N}(0, \sigma^2 I)$ is (ε,δ)-DP

for

$$\sigma = \frac{\sqrt{2\log(1.25)/\delta} \cdot \Delta_2}{\epsilon}$$

Sensitivity: Largest possible effect of changing input in **L2** norm

Benefits of DP

Also ok: Adaptive composition →

No assumptions on attacker

Attacker may have side information, e.g., know part of the dataset (not discussed)

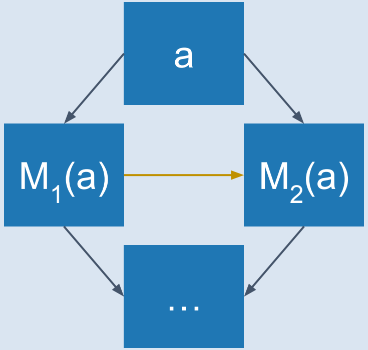
Protected against unbounded computation (see→)

Post-processing

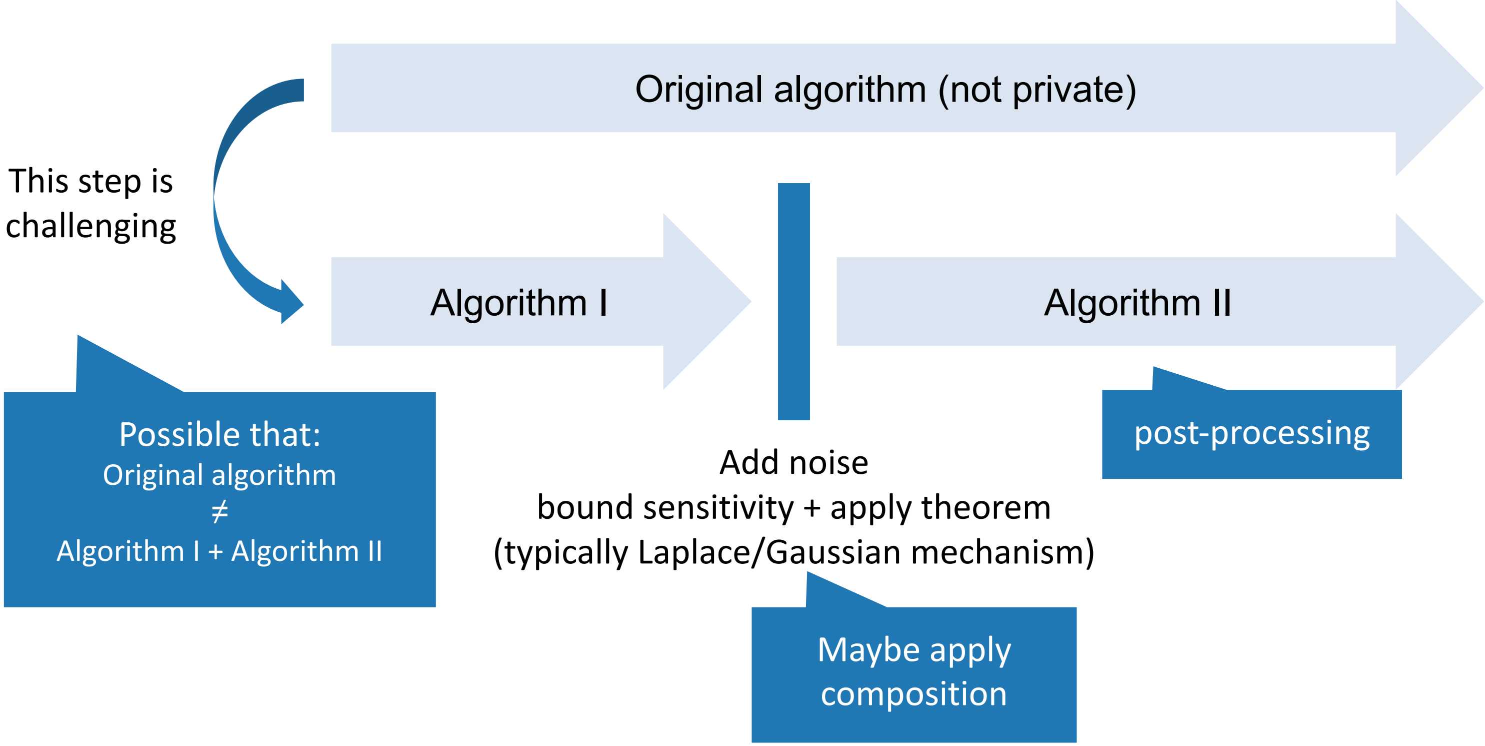
If M is (ε,δ)-DP, then $f \circ M$ is (ε,δ)-DP

Composition

If M_1 and M_2 are $(\varepsilon_1, \delta_1)$ and $(\varepsilon_2, \delta_2)$-DP, then the combined mechanism $M(a) := (M_1(a), M_2(a))$ is $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-DP



Common Pattern* when Creating DP Algorithms



Not always the case. Analysis could be harder and error-prone. May need analysis tools:

Bichse, Steffen, Bogunovic, Vechev. S&P21. DP-Sniper: Black-Box Discovery of Differential Privacy Violations using Classifiers

Next: Methods to Achieve DP in ML

Standard Setting

DP-SGD

Add noise during gradient update step

PATE

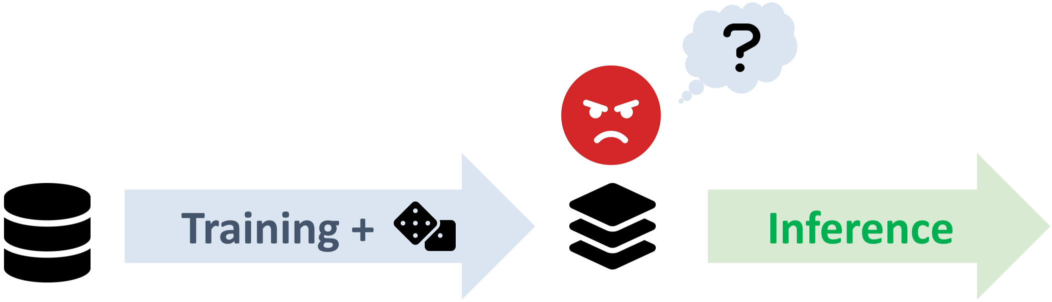
DP via knowledge transfer

Federated Setting

Noise before Aggregation

FedSGD and FedAVG with noise

DP-SGD



Idea

- Introduce noise during SGD **training**
- Can safely re-distribute resulting model
    - Private against **white-box attacker**
    - Private under **arbitrary number of inference queries** (see post-processing)

DP-SGD

Algorithm

> **In practice:**
> Permute inputs and iterate through batches of size L.

Initialize random θ_0

For $t \in \{0, \ldots, T-1\}$:

- Sample a random subset of L data points
- For each input x_i in the subset:
  - Compute gradient of loss: $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t} \mathcal{L}(\theta_t, x_i)$
  - Clip gradient: $\mathbf{g}'_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}\right)$
- Aggregate: $\bar{\mathbf{g}}_t \leftarrow \frac{1}{L} \sum_{i=1}^{L} \mathbf{g}'_t(x_i)$
- Add noise: $\tilde{\mathbf{g}}_t \leftarrow \bar{\mathbf{g}}_t + \mathcal{N}(0, \ \sigma^2 \mathbf{I})$
- Update: $\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$

Return θ_T

> Required to bound the sensitivity of the gradient update step

> Project onto ℓ_2-ball of size C

> C and L are **parameters** affecting privacy

> Add Gaussian noise of scale
>
> $$\sigma = \frac{\sqrt{2 \log(1.25)/\delta} \cdot (C/L)}{\epsilon}$$

> Level of privacy?

Abadi, Chu, Goodfellow. CCS 2016. Deep Learning with Differential Privacy

DP-SGD: Basic Privacy Analysis

1) Assume T = 1 and no sub-sampling (L = N)

Adding/removing an input to/from the training set affects **at most one index** i

Neighborhood: Training example input present vs. not present

$$\bar{\mathbf{g}}_t \leftarrow \frac{1}{L} \sum_{i=1}^{L} \mathbf{g}'_t(x_i)$$

L2 Sensitivity: C/L

$$\tilde{\mathbf{g}}_t \leftarrow \bar{\mathbf{g}}_t + \mathcal{N}(0,\ \sigma^2 \mathbf{I})$$

$$\sigma = \frac{\sqrt{2\log(1.25)/\delta} \cdot (C/L)}{\epsilon}$$

Gaussian mechanism

Result is (ϵ, δ)-DP

DP-SGD: Basic Privacy Analysis

For N inputs, define

$q = L/N$

2) Assume T = 1 but sample random fraction q

Theorem: Privacy Amplification

Applying a (ϵ, δ)-DP mechanism on a random fraction q subset yields a $(\tilde{q}\epsilon, q\delta)$-DP mechanism, where $\tilde{q} \approx q$.

Result is $(\tilde{q}\epsilon, q\delta)$-DP

Thm. 9 from: Balle et al. NeurIPS 2018. Privacy Amplification by Subsampling: Tight Analyses via Couplings and Divergences.

DP-SGD: Basic Privacy Analysis

3) Repeat for T >= 1 iterations

Apply **composition theorem:**
Privacy budgets "sum up"

When selecting $\sigma = \frac{\sqrt{2\log(1.25)/\delta} \cdot (C/L)}{\epsilon}$,

DP-SGD is $(\tilde{q}T\epsilon, qT\delta)$-DP

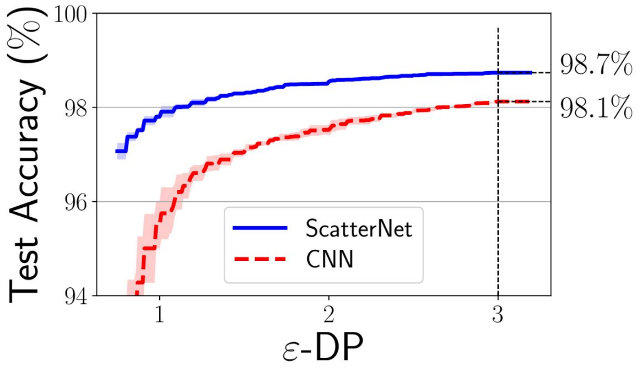
Problem: T large in practice

Why don't we just select ϵ, δ very small?

Problem: Introduces more noise (larger σ)...

Utility vs Privacy

More noise = more privacy :)
More noise = less utility :(



(a) MNIST

Not specific to DP-SGD, applies to all DP approaches (also beyond ML)

Florian Tramèr and Dan Boneh. ICLR 2021. Differentially Private Learning Needs Better Features (or Much More Data)

DP-SGD: Refined Privacy Analysis

DP-SGD is $(\tilde{q}T\epsilon, qT\delta)$-DP

Our analysis was simple, but very **imprecise**

Better bound via **strong composition theorem** (not discussed) and different σ:

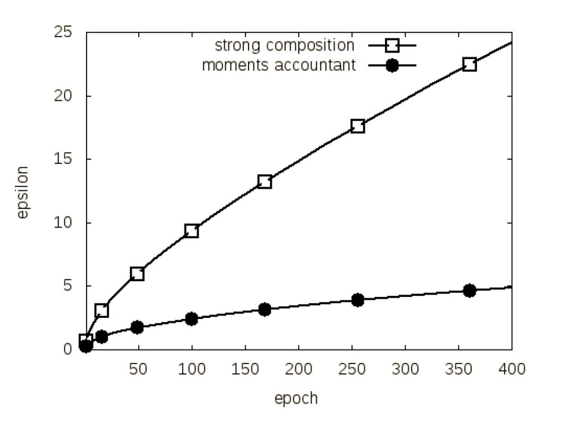
$\left(\mathcal{O}\left(q\epsilon\sqrt{T\log\frac{1}{\delta}}\right), \mathcal{O}(qT\delta)\right)$- DP

Even better bound via **moments accountant** (not discussed) and adaptive σ (data-dependent):

$(\mathcal{O}(q\epsilon\sqrt{T}), \delta)$ - DP

No factor of T any more

Now, privacy level depends on **data**: be careful!



Abadi, Chu, Goodfellow. CCS 2016. Deep Learning with Differential Privacy

DP-SGD: Problems

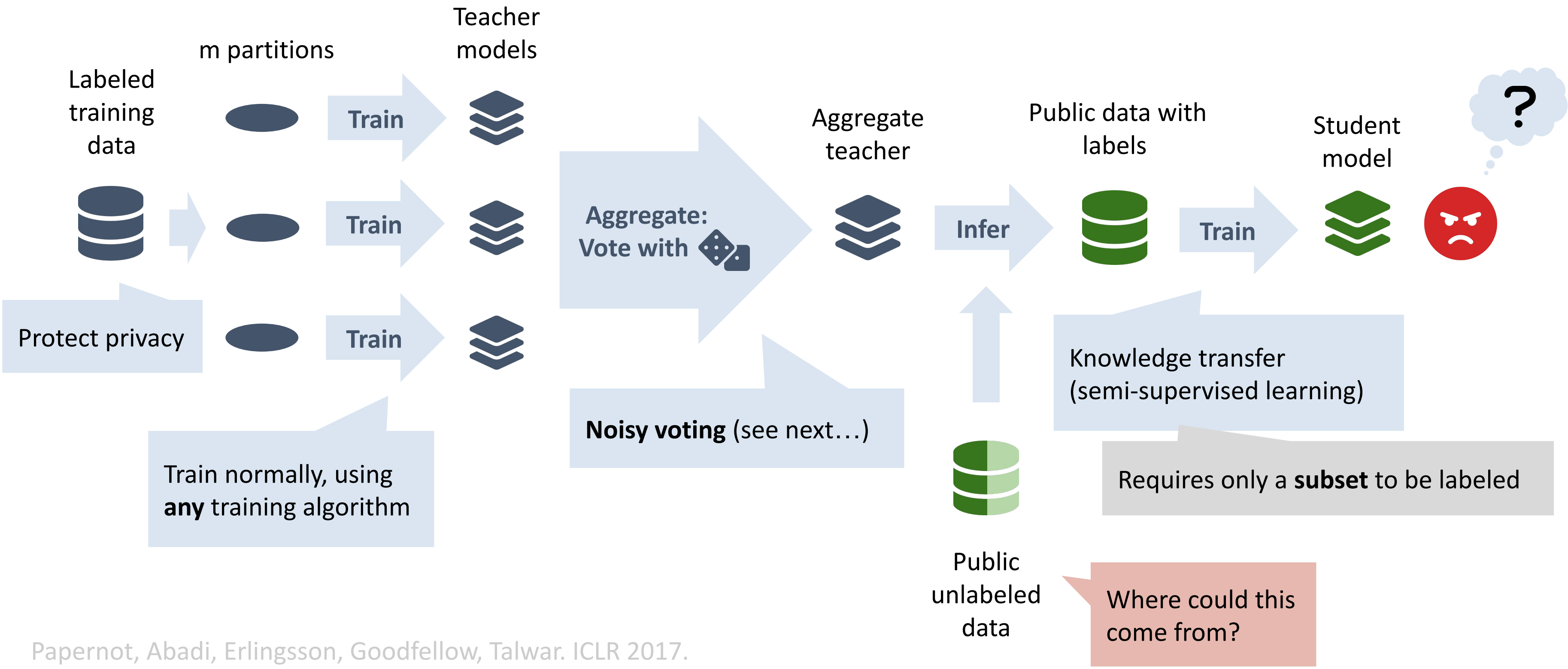
Problems with DP-SGD

- Tailored to **specific training algorithm** (SGD)
- Relatively **weak privacy guarantees** for reasonable utility:
  E.g. $(8, 10^{-5})$-DP for 97% accuracy on MNIST

Next: **PATE**

- Independent of training algorithm
- Better results:
  E.g. $(2.04, 10^{-5})$-DP for 98% accuracy on MNIST

PATE: Private Aggregation of Teacher Models



Papernot, Abadi, Erlingsson, Goodfellow, Talwar. ICLR 2017.
Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data

PATE: Noisy Voting

Let $n_j(\mathbf{x})$ be the number of teachers predicting class j for input $\mathbf{x}$.

The aggregate teacher f should use the votes $n_j(\mathbf{x})$ for prediction. Where to add noise?

Naive attempt: Laplace mechanism after voting

Need to add a lot of noise (c large…)

$$f(\mathbf{x}) = \arg\max_j \{n_j(\mathbf{x})\} + \mathrm{Lap}(0, ?)$$

Neighborhood: Training example input present vs. not present

Sensitivity c
(number of classes)

Better: Noise **before** argmax

By Laplace mechanism + post-processing:
One such inference query is $(\epsilon, 0)$-DP

$$f(\mathbf{x}) = \arg\max_j \{n_j(\mathbf{x}) + \mathrm{Lap}(0,\ 2/\epsilon)\}$$

Sensitivity for vector $n(\mathbf{x})$ is $\Delta_1 = 2$

PATE: Basic Privacy Analysis

Labeling T inputs for training the student is **composition**

One query: $(\epsilon, 0)$-DP

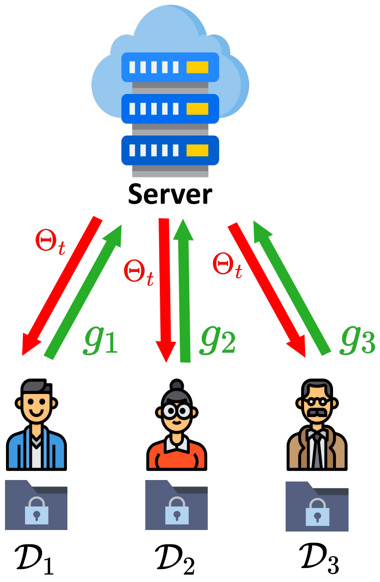
T queries: $(\epsilon T, 0)$-DP

Number of labels required to train student is large in practice ($T \approx 100$)…

Again, can get better bounds via **strong composition theorem** or data-dependent **moments accountant** (not discussed)

After labeling the public dataset, the remaining pipeline is just **postprocessing** and does **not** affect privacy

Idea: FedSGD with Noise



Server aggregation

$$g_c \leftarrow \frac{1}{K} \sum_{k=1}^{K} g_k$$
$$\Theta_{t+1} \leftarrow \Theta_t - \gamma g_c$$

Client update

$$\{x^k, y^k\} \sim \mathcal{D}_k$$
$$g_k \leftarrow \nabla_\Theta \mathcal{L}(f_{\Theta_t}(x^k), y^k)$$

Idea: Make this differentially private using DP-SGD

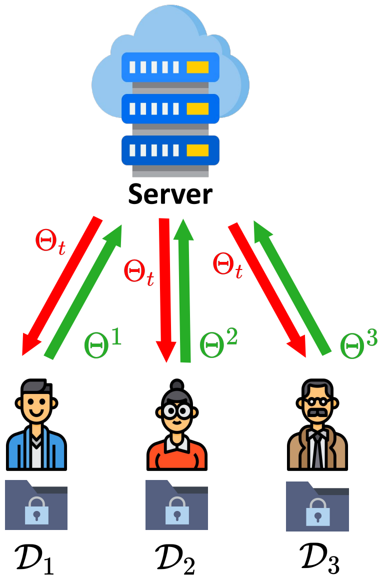
Client update using DP-SGD

$$g_k \leftarrow \nabla_\Theta \mathcal{L}(f_{\Theta_t}(x^k), y^k)$$
$$\bar{g}_k \leftarrow g_k / \max\left(1, \frac{\|g_k\|_2}{C}\right)$$
$$\tilde{g}_k \leftarrow \bar{g}_k + \mathcal{N}(0, \sigma^2 \mathbf{I})$$

Clip and add noise

Analogous analysis as for DP-SGD

Idea: FedAVG with Noise



Server aggregation

$$\Theta_{t+1} \leftarrow \frac{1}{K} \sum_{k=1}^{K} \Theta^k$$

Idea: Make this differentially private by adding noise to weights

Client update

$\Theta^k_{1,1} \leftarrow \Theta_t$
for e in range(E):
  for b in range(B):
    $\{x^k_{e,b}, y^k_{e,b}\} \sim \mathcal{D}_k$
    $\Theta^k_{e,b} \leftarrow \Theta^k_{e,b-1} - \gamma \nabla_\Theta \mathcal{L}(f_{\Theta^k_{e,b-1}}(x^k_{e,b}), y^k_{e,b})$
  end for
end for
$\Theta^k \leftarrow \Theta^k_{E,B}$

Client update

$\Theta^k_{1,1} \leftarrow \Theta_t$
for e in range(E):
  for b in range(B):
    $\{x^k_{e,b}, y^k_{e,b}\} \sim \mathcal{D}_k$
    $\Theta^k_{e,b} \leftarrow \Theta^k_{e,b-1} - \gamma \nabla_\Theta \mathcal{L}(f_{\Theta^k_{e,b-1}}(x^k_{e,b}), y^k_{e,b})$
  end for
end for
$\Theta^k \leftarrow \Theta^k_{E,B}$
$\Theta^k \leftarrow \Theta^k / \max\left(1, \frac{\|\Theta^k\|}{C}\right)$
$\Theta^k \leftarrow \Theta^k + \mathcal{N}(0, \sigma^2 \mathbf{I})$

Clip and add noise

Wei et al. arXiv 2019. Federated Learning with Differential Privacy: Algorithms and Performance Analysis

Analogous for L2 Smoothing (but with Gaussian noise)

ection to Randomized Smoothing

Simple L1 Smoothing

- f: $\mathbb{R}^d \rightarrow Y$
- Bounded attacks: $\|a-a'\|_1<R$
- **Classify** a as c IFF
  $\forall c' \neq c.\ Pr[f(a + \eta)=c] > Pr[f(a + \eta)=c']$
  for $\eta \sim Lap(0,R/\varepsilon)$
- **Robust** IF
  $\forall c' \neq c.\ Pr[f(a + \eta)=c] > \exp(2\varepsilon)\ Pr[f(a + \eta)=c']$

Analysis

- $a + \eta$ is ε-DP (Laplace mechanism)
- $f(a + \eta)$ is ε-DP (post-processing)
- Robust (due to DP, see exercises)

Lecuyer, Atlidakis, Geambasu, Hsu, and Jana. S&P 2019. https://arxiv.org/pdf/1802.03471.pdf
Certified Robustness to Adversarial Examples with Differential Privacy

Summary

- We introduced the notion of Differential Privacy (DP) a principled mechanism to defend against membership inference attacks.
- We discussed basic general mechanisms achieving DP, including the Laplace and Gaussian mechanisms.
- We introduced and applied important properties of DP, especially post-processing and composition, and discussed its inherent utility-privacy tradeoff.
- We analyzed several methods to achieve DP in machine learning, including techniques perturbing gradients or performing noisy voting. Such methods can be used to achieve DP guarantees in the setting of federated learning.
- We discussed the connection of DP to Randomized Smoothing.