Differential Privacy made practical

Slides and notebook for this talk:

github.com/robromijnders/pydata24

What is the problem?

Users want to contribute data

But need a guarantee that their details are not leaked

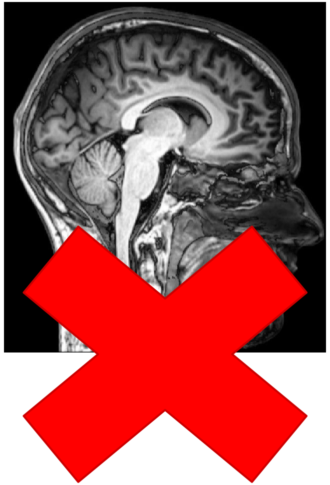


Image credits:
https://vectorportal.com/vector/hospital/34993
https://commons.wikimedia.org/wiki/File:MRI_brain_sagittal_section.jpg

github.com/robromijnders/pydata24

Are you in the stack?

Example: https://huggingface.co/spaces/bigcode/in-the-stack

Instruction: Create a SQL query to get the list of employee names and ids with a monthly income greater than 4,000.
Input: n/a
Output:
```
SELECT id, name FROM Employees WHERE
monthly_income > 4000;
```

Instruction: Write a code to add two numbers without using the \"+\" operator.
Input:
```
num1 = 2
num2 = 7
```
Output:
```
num1 = 2
num2 = 7
# Without using \"+\" operator
sum = num1 - (-num2)
```

Instruction: Optimize the given Python program to improve the speed of execution.

Input:
```
def calc( num1, num2):
    result = 0
    for i in range(num1, num2):
        result += ((i+1) * (i+2))
    return result
```

Output:
```
def calc(num1, num2):
    result = (num1 + num2) * (num2
- num1 + 1) // 2
    return result
```

Example from https://arxiv.org/pdf/2305.07922

The Stack version:

v2.1.0

Your GitHub username:

robromijnders

Check!

Yes, there is code from **28 repositories** in The Stack. Check the links to see when it was archived by Software Heritage:

RobRomijnders/DRAW_1D

RobRomijnders/EDS

RobRomijnders/FCN

RobRomijnders/MCMC

RobRomijnders/RBFN_two_MNIST

RobRomijnders/SOM_MNIST

RobRomijnders/bandit

RobRomijnders/bayesian_model_comparison

RobRomijnders/bbvi

RobRomijnders/dan

RobRomijnders/dpbmm

RobRomijnders/dpm

RobRomijnders/far_away

RobRomijnders/gp_hyper

Basic pattern

Differential Privacy:

“Noise up a learning algorithm such that individual contribution is blurred, but collective contribution can be learned.”

Practical use of Differential Privacy

- Emoji suggestions at **Apple**
- QuickType suggestions at **Apple**
- **US Census** releases data under DP
- Executive order US gov. mentions Differential Privacy multiple times
- **Governments** releasing birth rate data
- **Facebook** releases mobility data of users during covid pandemic
- **Google GBoard** language next word prediction
- **LinkedIn** user analytics
- Telemetry on **Windows**

Ponomareva et al. JAIR 2023
Apple blog on DP
Executive order US white house
Facebook Mobility, 2020
LinkedIn (slides, post)
Windows telemetry (NeurIPS paper)
Israeli Birthrate release
Other great overview

But what is the alternative?

Anonymization

Obfuscation (example next slide)

But what is the alternative?

Anonymization

But doesn't work, e.g., Netflix prize participants uncovered

Obfuscation (example next slide)

Approach 1 [Does not work]

Approach: only allow averages

Approach 1 [Does not work]

Approach: only allow averages

Problem:

AVG(age) WHERE name='Jan Kees' ==> 97

Restrict to averages does not work [Example]

https://en.wikipedia.org/wiki/Reconstruction_attack

Approach (2) [does not work]

Solution: only allow averages of subset size >= 50

Approach (2) [does not work]

Solution: only allow averages of subset size >= 50

Problem:

CNT(1) WHERE name='Jan Kees' OR name='Peter' ==> 96
CNT(1) WHERE name='Peter' ==> 95

Approach (2) [does not work]

Solution: only allow averages of subset size >= 50

Problem:

COUNT(1) WHERE name='Jan Kees' OR name='Peter' ==> 96
COUNT(1) WHERE name='Peter' ==> 95

x = AVG(age) WHERE name='Jan Kees' OR name='Peter' ==> 49.5
y = AVG(age) WHERE name='Peter' ==> 49

Approach (2) [does not work]

$x*96$ = total age of Jan Kees and people named Peter
$y*95$ = total age of people named Peter

$x*96 - y*95 = 4752 - 4655 = 97$

Restrict group size does not work

Approach (3) [Differential Privacy]

x = AVG(age) WHERE name='Jan Kees' OR name='Peter' **+ Noise**

==> 49.5 + L(.) = 49.9

y = AVG(age) WHERE name='Peter' **+ Noise**

==> 49 + L(.) = 48.8

Approach (3) [Differential Privacy]

x = AVG(age) WHERE name='Jan Kees' OR name='Peter' **+ Noise**

==> 49.5 + L(.) = 49.9

y = AVG(age) WHERE name='Peter' **+ Noise**

==> 49 + L(.) = 48.8

And now:

x*96 - y*95 = 154.4

Approach (3) [Differential Privacy]

x = AVG(age) WHERE name='Jan Kees' OR name='Peter' **+ Noise**

==> 49.5 + L(.) = 49.3

y = AVG(age) WHERE name='Peter' **+ Noise**

==> 49 + L(.) = 49.2

And now:

x*96 - y*95 = 58.8

Approach (3) [Differential Privacy]

x = AVG(age) WHERE name=‘Jan Kees’ OR name=‘Peter’ **+ Noise**

==> 49.5 + L(.) = 49.3

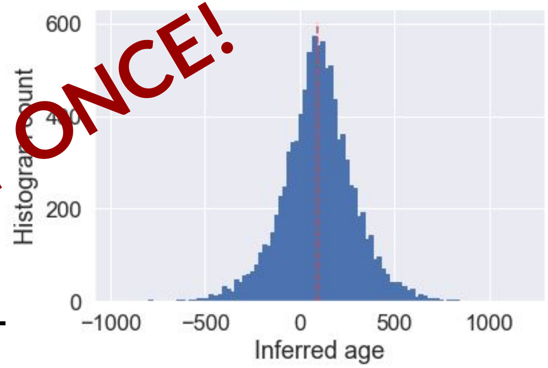
y = AVG(age) WHERE name=‘Peter’ **+ Noise**

==> 49 + L(.) = 49.2

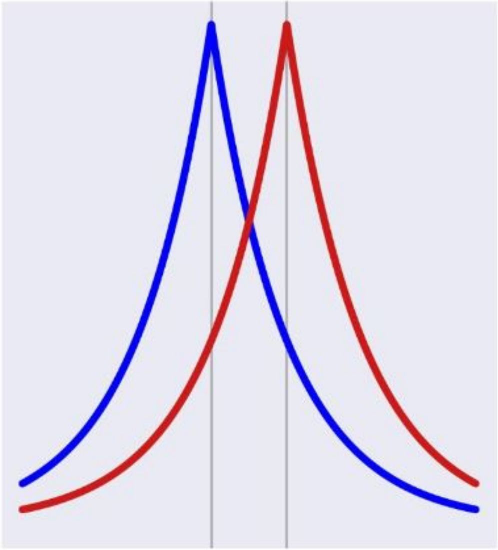
And now:

x*96 - y*95 = 58.8

ONLY ONCE!



Why that Laplace noise?



Adding Laplace noise results in Differential Privacy. It hides the contribution of a single individual:

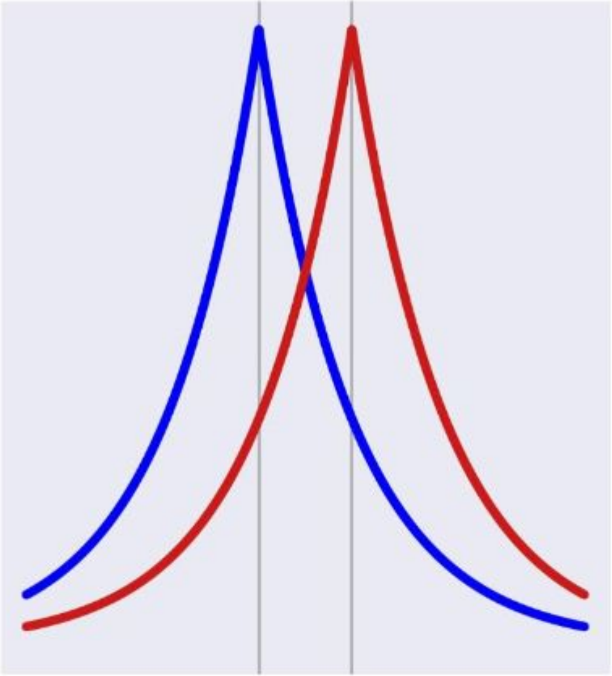
If an individual has age between 2 and 100, then the Laplace noise is proportional to (100-2)

$$\mathcal{L}(\cdot) = \mathrm{Laplace}(b = \frac{(100-2)}{N}) = \frac{1}{2b}e^{-\frac{|x|}{b}}$$

Differential Privacy

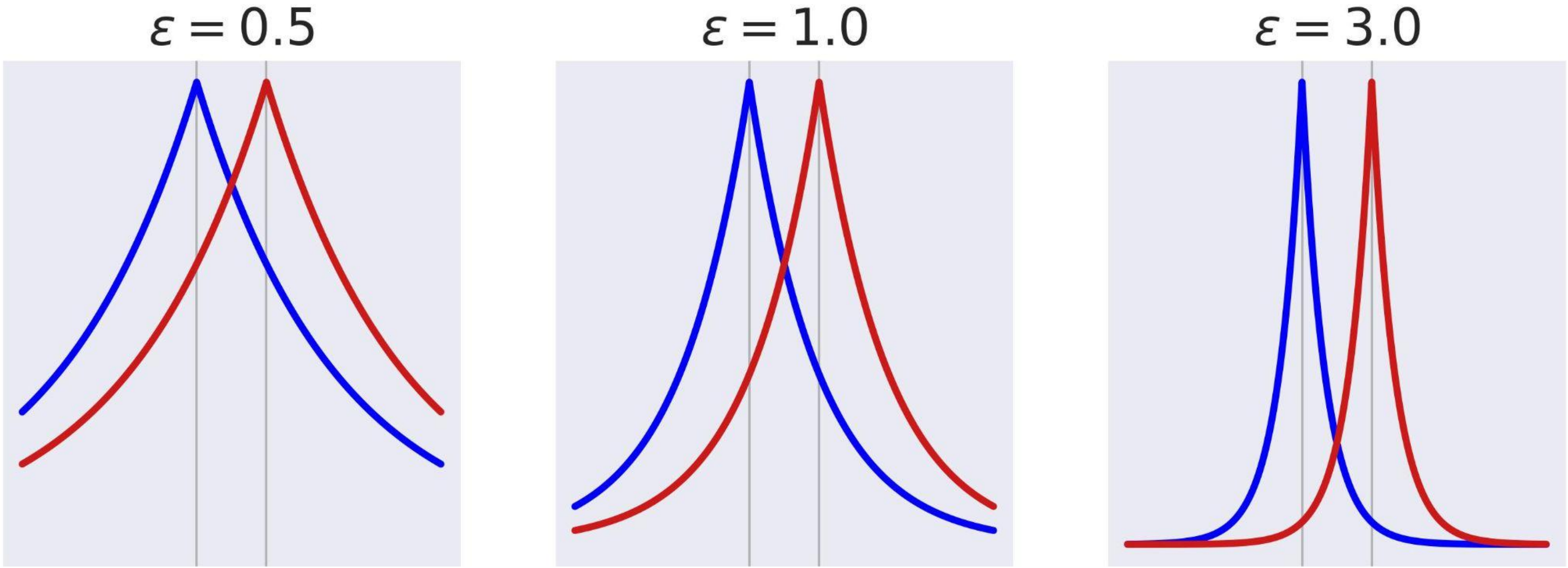
(ε, δ)-DP: an algorithm $A(\cdot)$, with outcome W, and two adjacent data sets D, D' that have at most one element different:

$$\mathrm{P}(A(D) \in W) \leq e^{\varepsilon}\, \mathrm{P}(A(D') \in W) + \delta$$



Dwork & Roth, 2014, "Algorithmic Foundations of Differential Privacy"

Noise proportional to inverse epsilon



github.com/robromijnders/pydata24

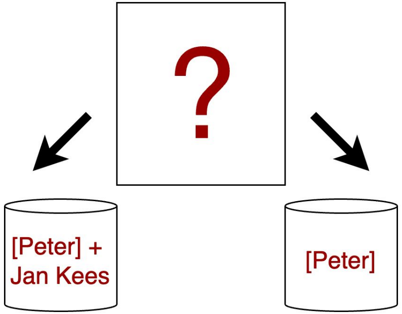
Laplace distribution, proportional to sensitivity Δ

$$\frac{\mathrm{P}_D(z)}{\mathrm{P}_{D'}(z)} = \exp\left(-\frac{|A(D)-z|}{\Delta/\varepsilon}\right) / \exp\left(-\frac{|A(D')-z|}{\Delta/\varepsilon}\right) \tag{1}$$

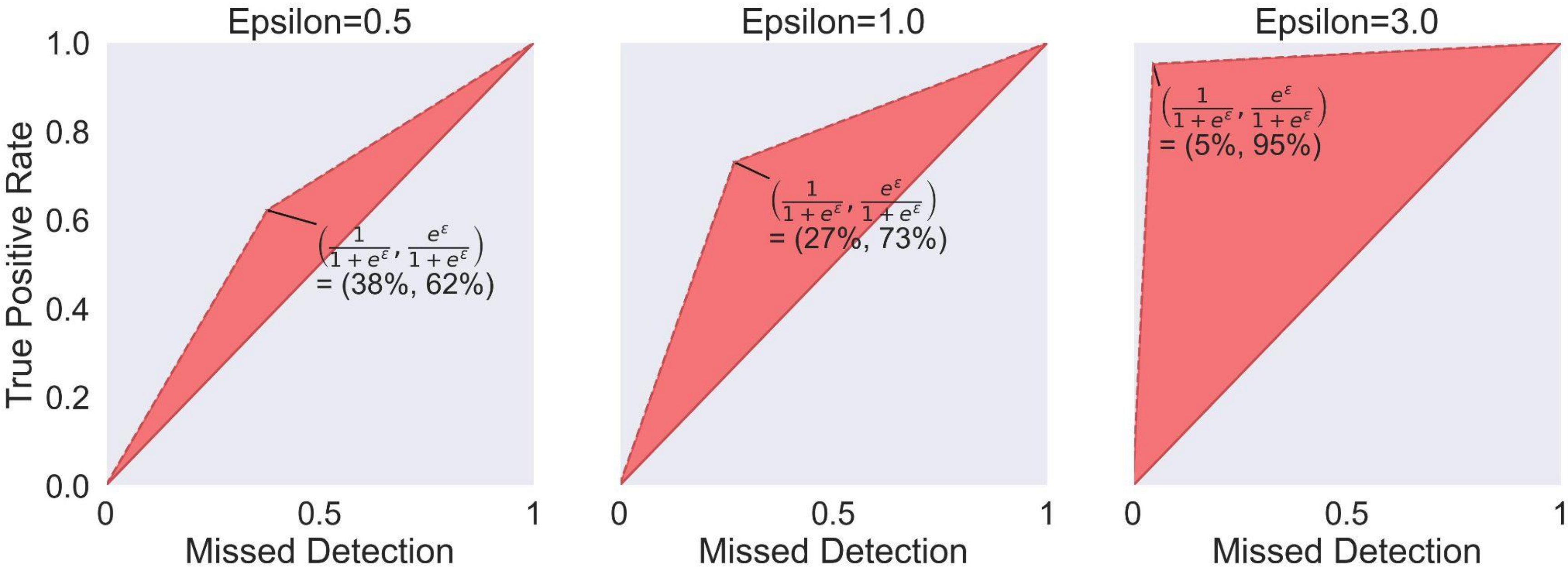
$$= \exp\left(\frac{\varepsilon(|A(D')-z| - |A(D)-z|)}{\Delta}\right) \tag{2}$$

$$\leq \exp\left(\frac{\varepsilon|A(D)-A(D')|}{\Delta}\right) \tag{3}$$

$$\leq \exp(\varepsilon) \tag{4}$$



What does DP mean?



https://en.wikipedia.org/wiki/Differential_privacy#Hypothesis_testing_interpretation

github.com/robromijnders/pydata24

Two applications

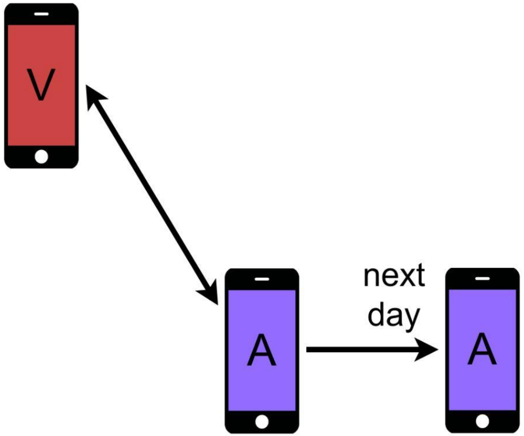
- Contact tracing COVID19 with Differential Privacy

- Deep Learning with Differential Privacy

Contact tracing COVID19 with Differential Privacy

Attack Scenario

Privacy with respect to released covidscore

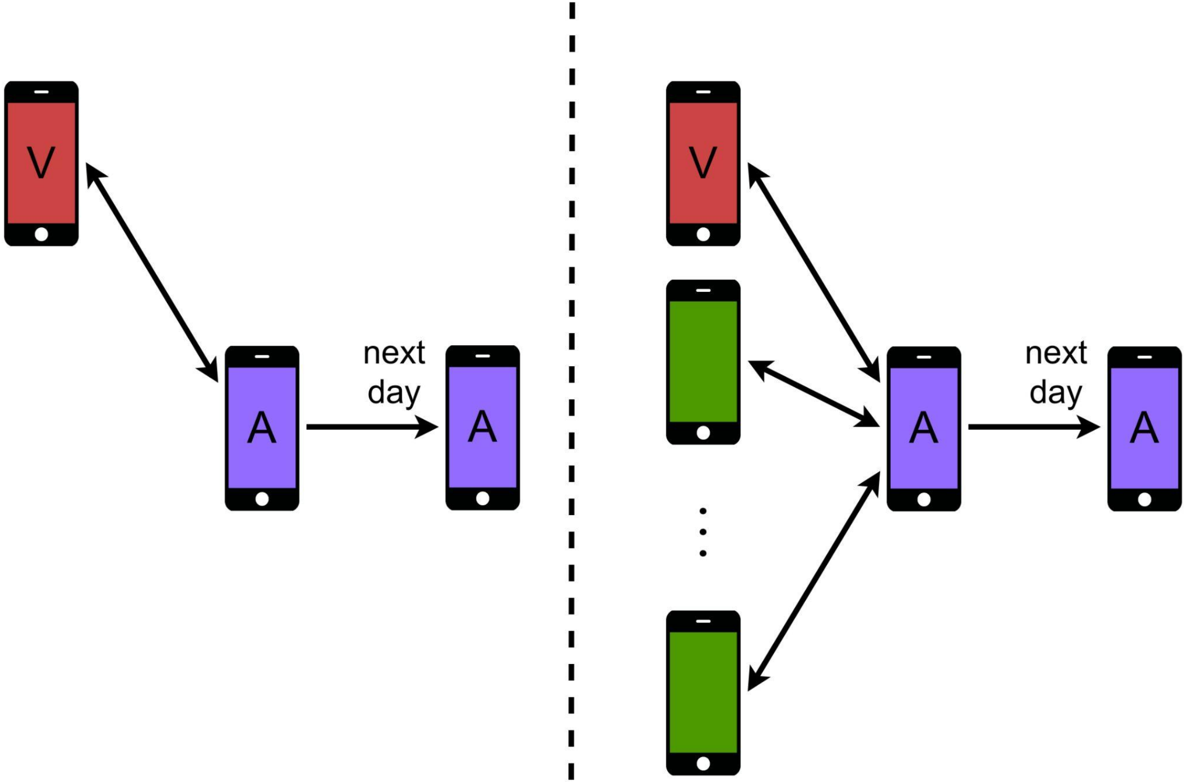


V is victim, A is attacker

Green phones are agents with ‘known’ score

Attack Scenario

Privacy with respect to released covidscore



V is victim, A is attacker

Green phones are agents with ‘known’ score

Add noise to COVID Risk Score

Algorithm 2 DP Contact Tracing COVID19

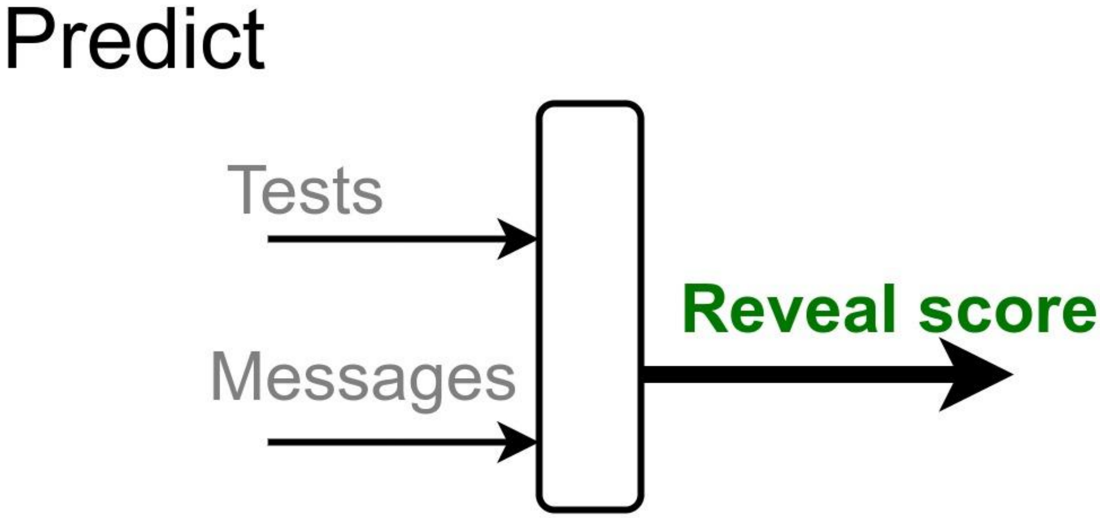
Require: Dataset $D = \{(\mu_i, t_i)\}_{i=1}^{C_T}$, constants $p_1, \gamma_u \in (0, 1)$;

$\mu_i \leftarrow \min(\mu_i, \gamma_u)$

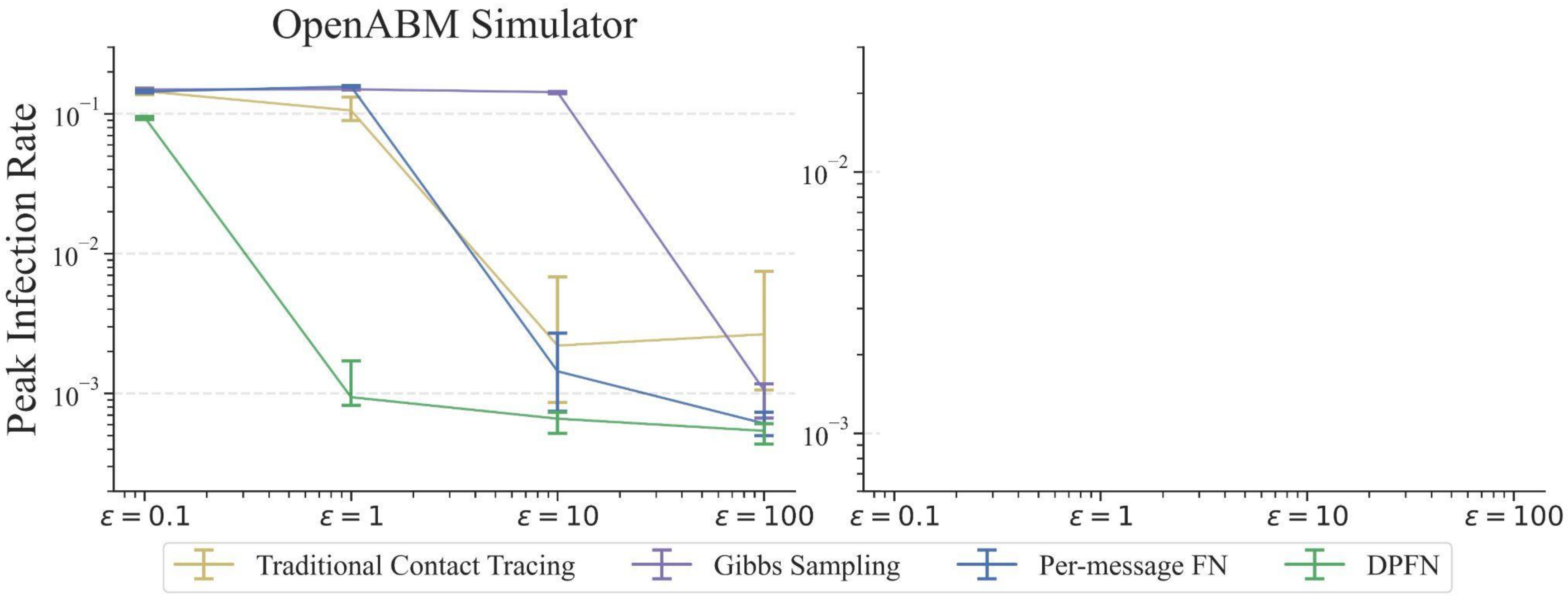
$\bar{\phi} \leftarrow F(\{(\mu_i, t_i)\}_{i=1}^{C_T}) + \mathcal{N}(0, \mathrm{V}(\varepsilon, \delta, p_1, \gamma_u, C_T))$

From https://arxiv.org/abs/2404.13381

Contact tracing COVID19



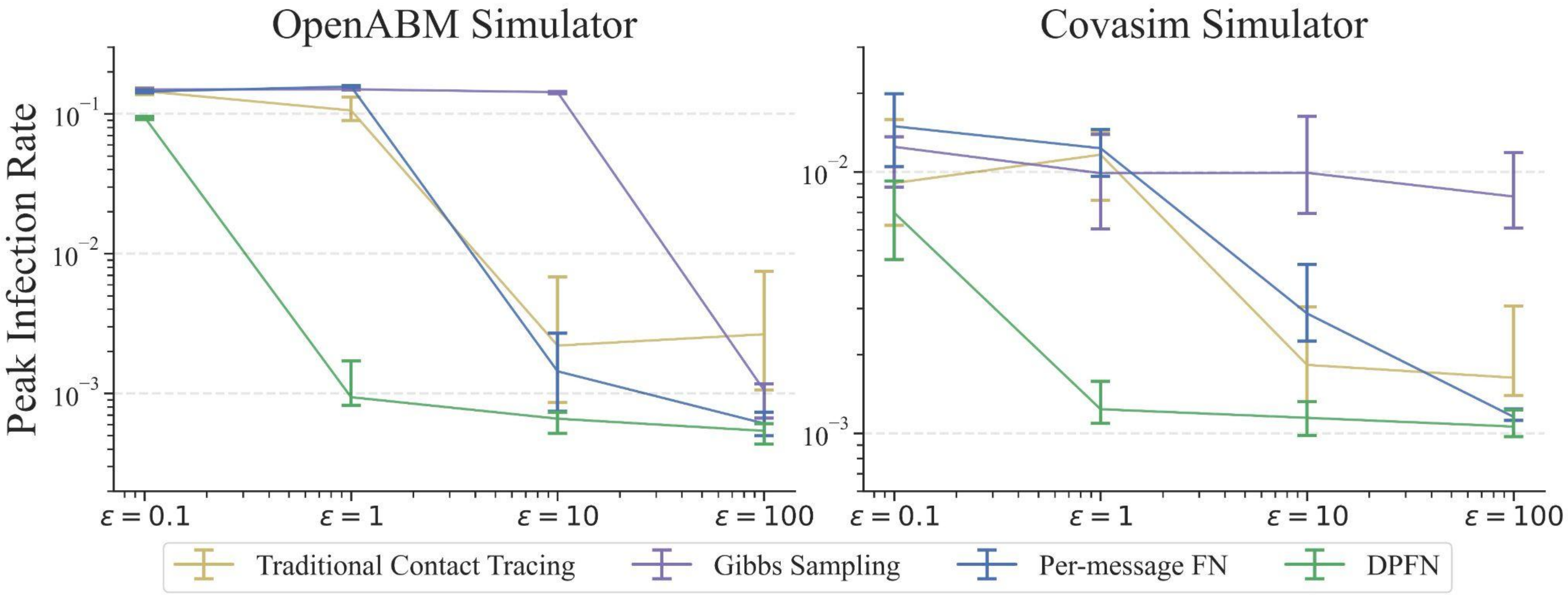
Results on two widely used simulators



From https://arxiv.org/abs/2404.13381

github.com/robromijnders/pydata24

Results on two widely used simulators

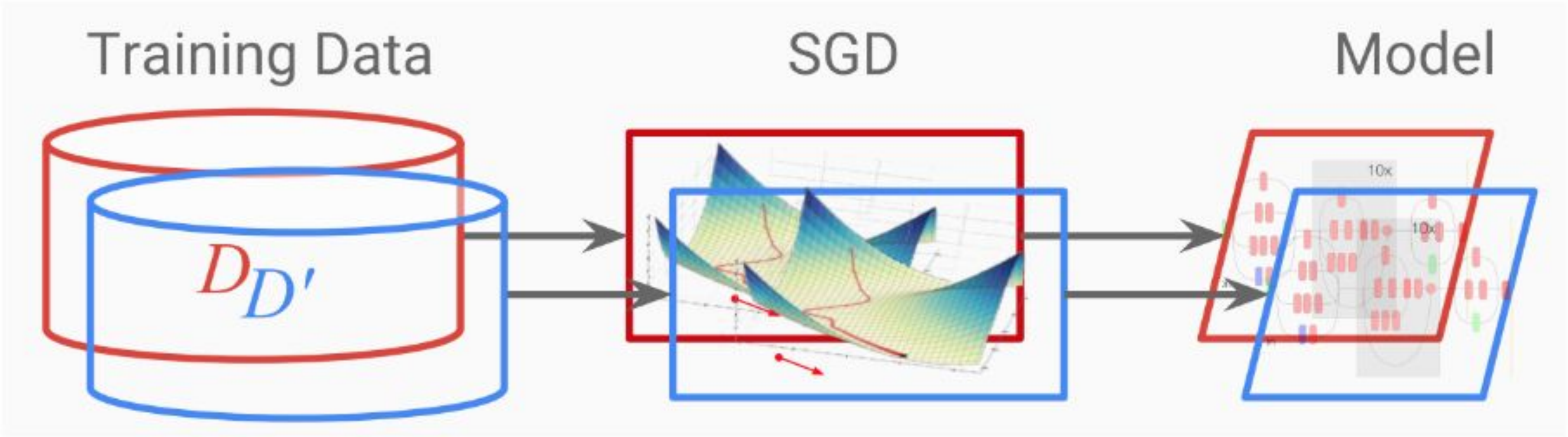


From https://arxiv.org/abs/2404.13381

github.com/robromijnders/pydata24

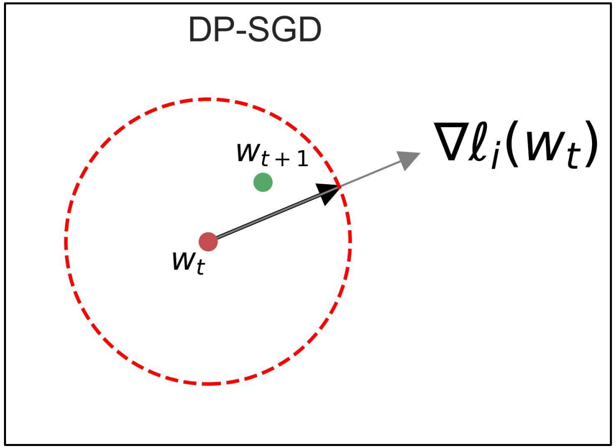
Deep Learning with Differential Privacy

How to achieve DP for Neural Nets?



github.com/robromijnders/pydata24

Clip and noise gradients



Require: Parameters w_0, η, DP parameters C, σ^2

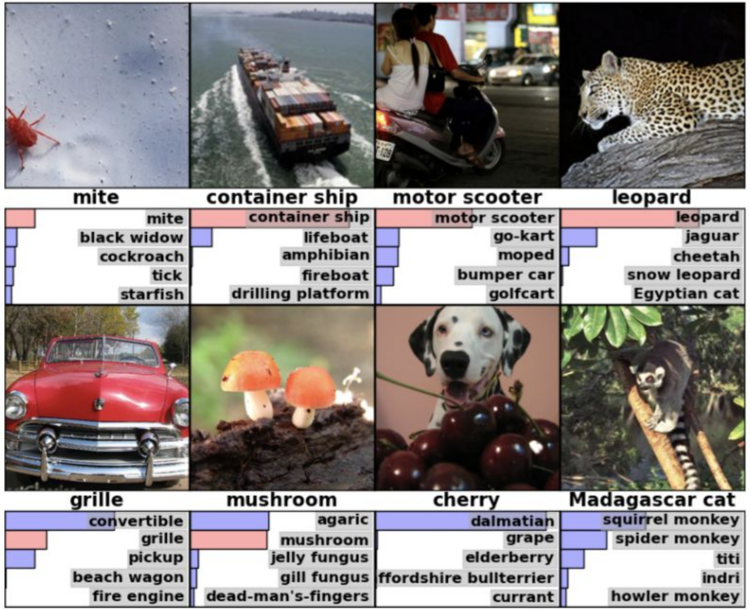
1: **for** $t = 1, 2, \ldots, T$ **do**

2: Construct a random mini-batch I_t

3: $g_t = \mathcal{N}(0, C^2\sigma^2 I) + \frac{1}{|I_t|} \sum_{i \in I_t} \text{clip}(\nabla \ell_i(w_t), C)$

4: $w_{t+1} = w_t - \eta g_t$

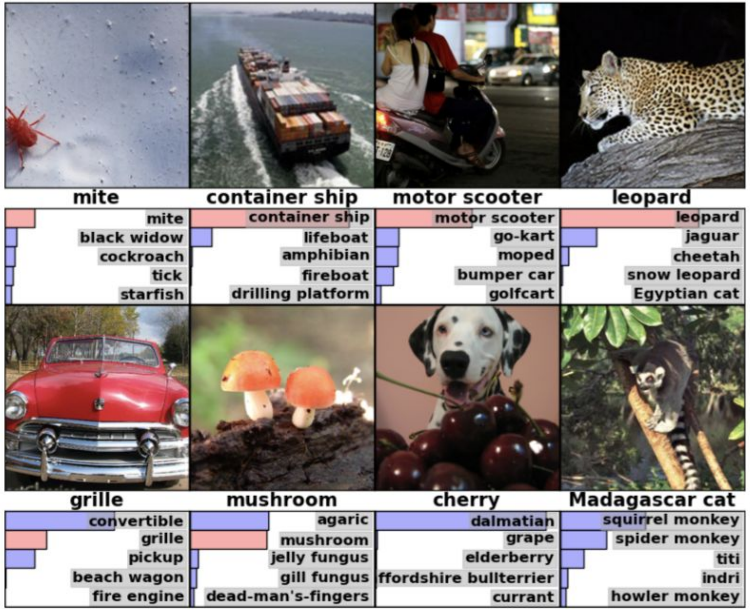
5: **end for**



DP in Neural Nets is bad

DP	privacy loss bound ε							
	4.6	13.2	71	$\approx 10^7$	10^9	10^{11}	10^{13}	10^{15}
Resnet-18	3.7%	6.9%	11.3%	45.7%	55.4%	56.0%	56.3%	56.4%
Resnet-50	2.4%	5.0%	7.7%	44.3%	58.8%	57.8%	58.2%	58.6%

Table from https://arxiv.org/pdf/2201.12328
Image from https://blog.acolyer.org/



DP in Neural Nets is bad

Note: empirical epsilon can be up to 30x lower (link)

DP	privacy loss bound ε							
	4.6	13.2	71	$\approx 10^7$	10^9	10^{11}	10^{13}	10^{15}
Resnet-18	3.7%	6.9%	11.3%	45.7%	55.4%	56.0%	56.3%	56.4%
Resnet-50	2.4%	5.0%	7.7%	44.3%	58.8%	57.8%	58.2%	58.6%

Table from https://arxiv.org/pdf/2201.12328
Image from https://blog.acolyer.org/

Conclusion

* Differential Privacy blurs contribution of a single data point

* Strive for epsilon=1, delta<<1/N

Useful libraries:

tensorflow.org/responsible_ai/privacy/api_docs/
github.com/pytorch/opacus
github.com/RobRomijnders/dna

Appendix
[extra slides]

Practical use of Differential Privacy

- Emoji suggestions at **Apple** (eps=4.)
- QuickType suggestions at **Apple** (eps=8.)
- **US Census** releases data under DP (eps=12.2 per person)
- Executive order US gov. mentions Differential Privacy multiple times
- **Governments** releasing birth rate data (eps=9.98)
- **Facebook** releases mobility data of users during covid pandemic (eps=2.)
- **Google GBoard** language next word prediction (eps=8.9, device level)
- **LinkedIn** user analytics (eps=1.0, record level)
- Telemetry on **Windows** (every six hours, eps=1.0)

Ponomareva et al. JAIR 2023
Apple blog on DP
Executive order US white house
Facebook Mobility, 2020
LinkedIn (slides, post)
Windows telemetry (NeurIPS paper)
Israeli Birthrate release
Other great overview

Example of Membership Inference Attack: how good can be?

Using ML-as-a-service from cloud providers

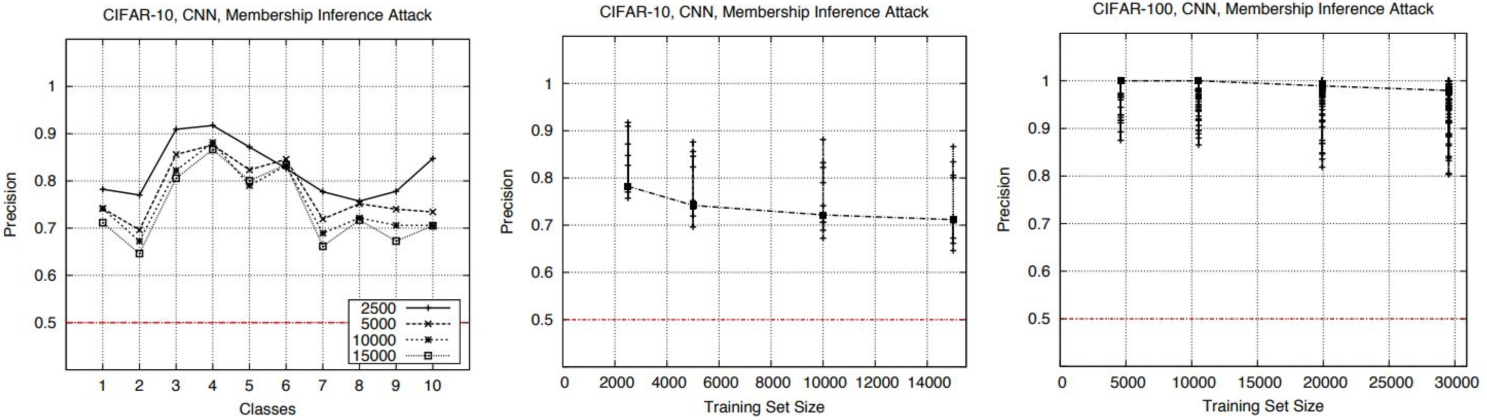
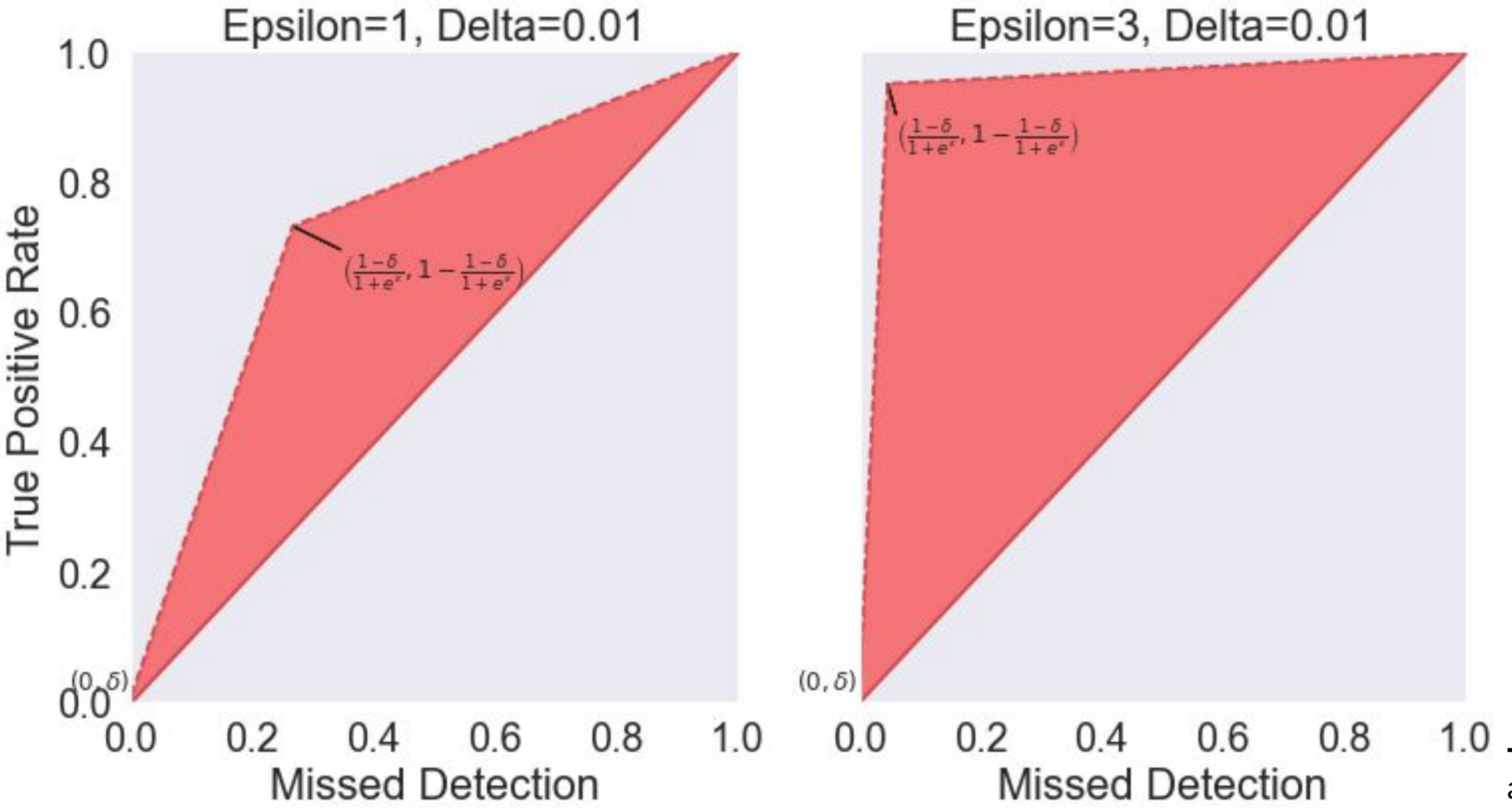


Fig. 4: Precision of the membership inference attack against neural networks trained on CIFAR datasets. The graphs show precision for different classes while varying the size of the training datasets. The median values are connected across different training set sizes. The median precision (from the smallest dataset size to largest) is $0.78, 0.74, 0.72, 0.71$ for CIFAR-10 and $1, 1, 0.98, 0.97$ for CIFAR-100. Recall is almost 1 for both datasets. The figure on the left shows the per-class precision (for CIFAR-10). Random guessing accuracy is 0.5.

Shokri et al.
SSP, 2017

Credits

Image credits online are due to https://drlee.io/differential-privacy-in-machine-learning-intuitive-explanation-use-cases-and-practical-911e5952ae4e


Epsilon=1, Delta=0.01
Epsilon=3, Delta=0.01
True Positive Rate
Missed Detection
Missed Detection
$\left(\frac{1-\delta}{1+e^{\epsilon}}, 1-\frac{1-\delta}{1+e^{\epsilon}}\right)$
$(0,\delta)$
0.0
0.2
0.4
0.6
0.8
1.0