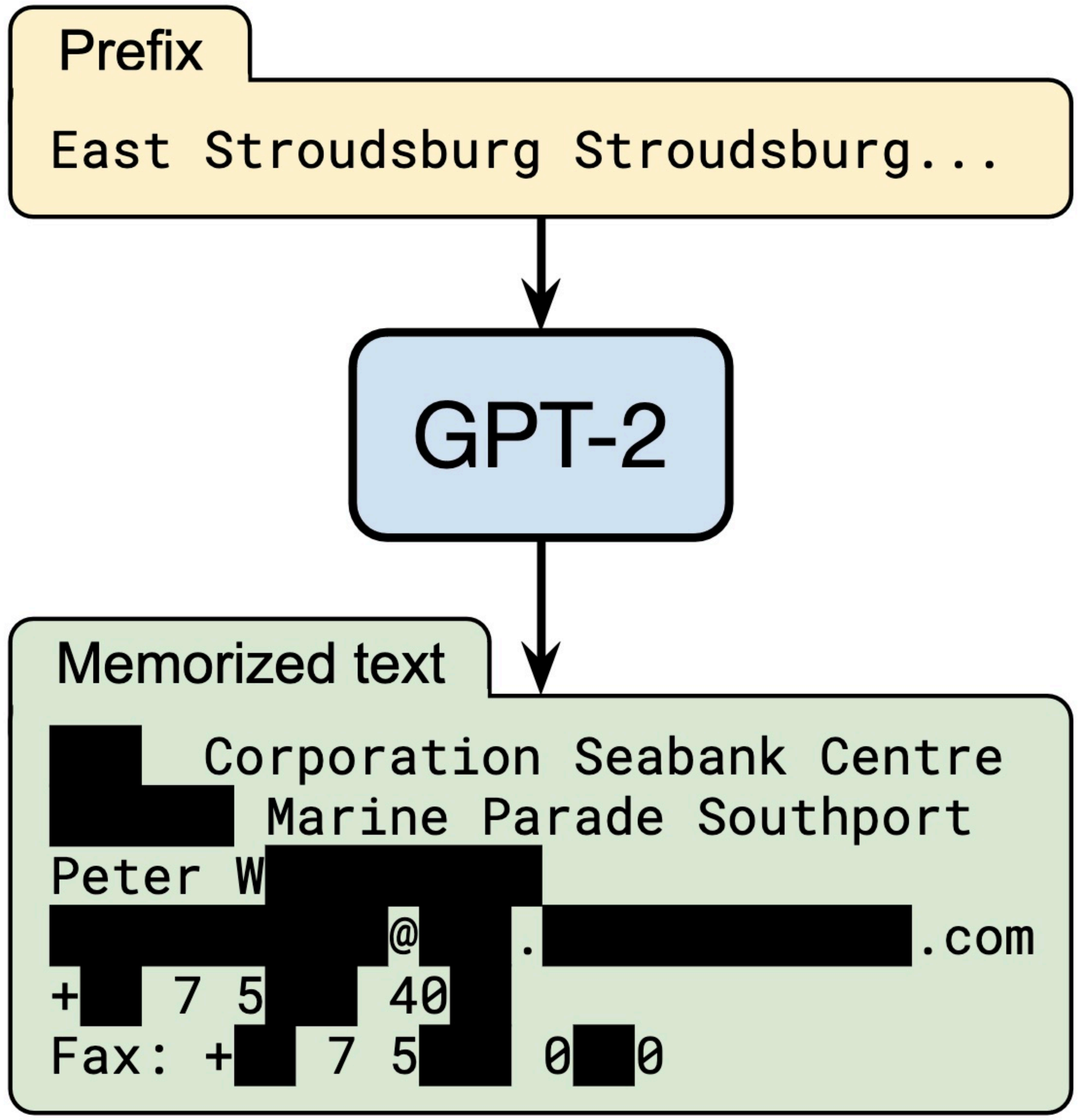
Towards user-level differential privacy at scale

Krishna Pillutla

Wadhwani School of Data Science & AI,
IIT Madras

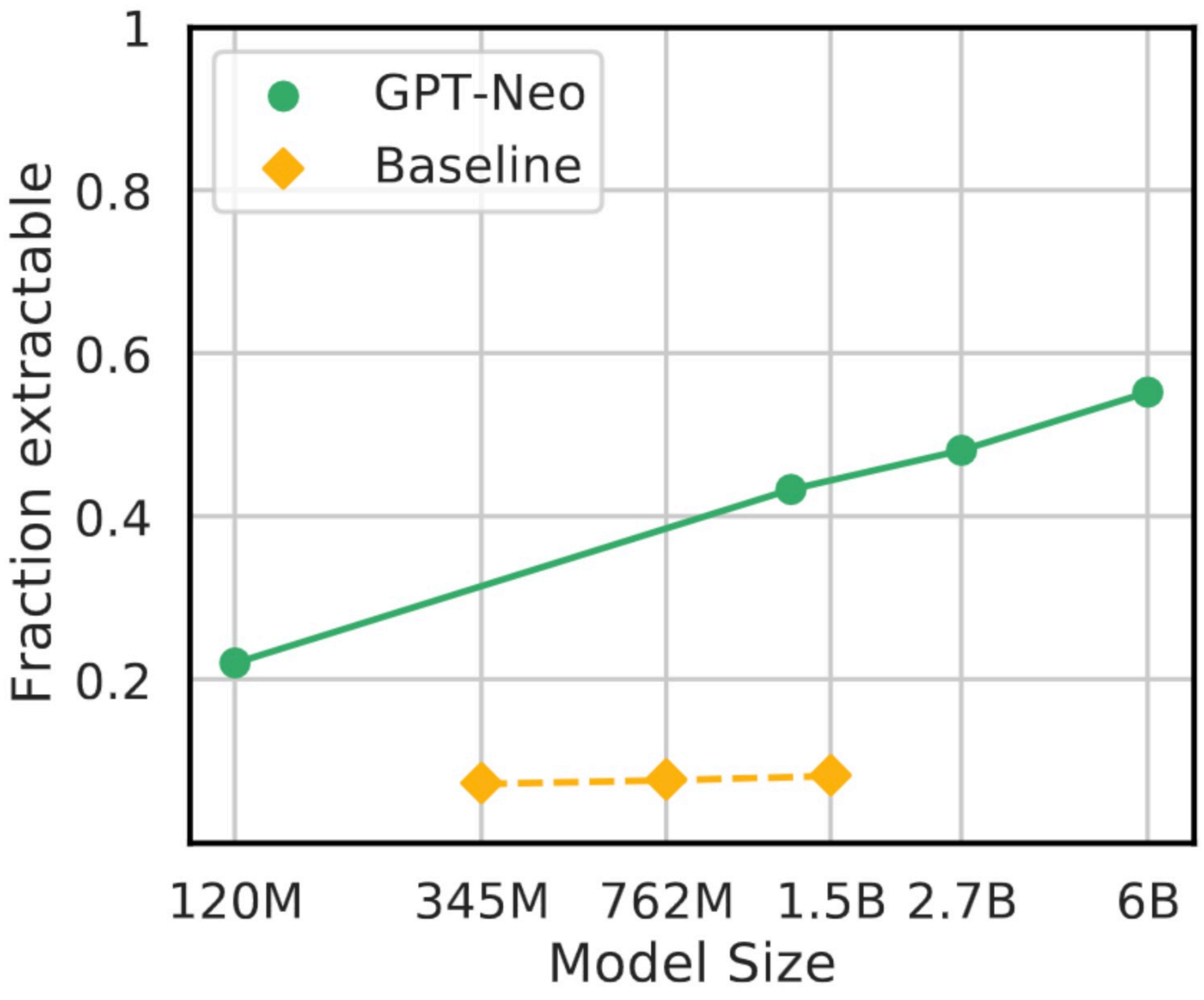
WHEN YOU TRAIN PREDICTIVE MODELS ON INPUT FROM YOUR USERS, IT CAN LEAK INFORMATION IN UNEXPECTED WAYS.

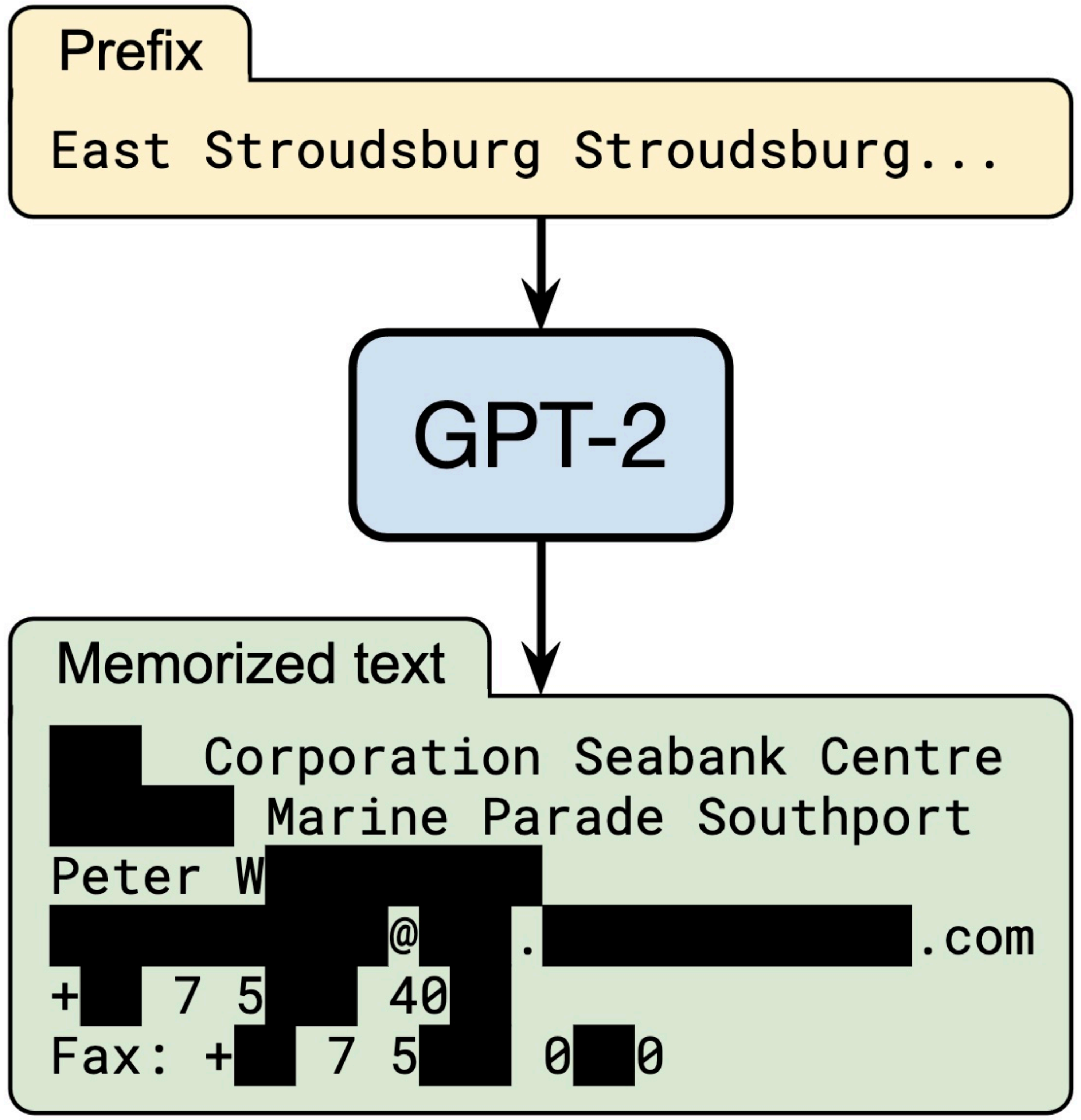
Models leak information about their training data



Carlini et al. (USENIX Security 2021)

Models leak information about their training data ***reliably***





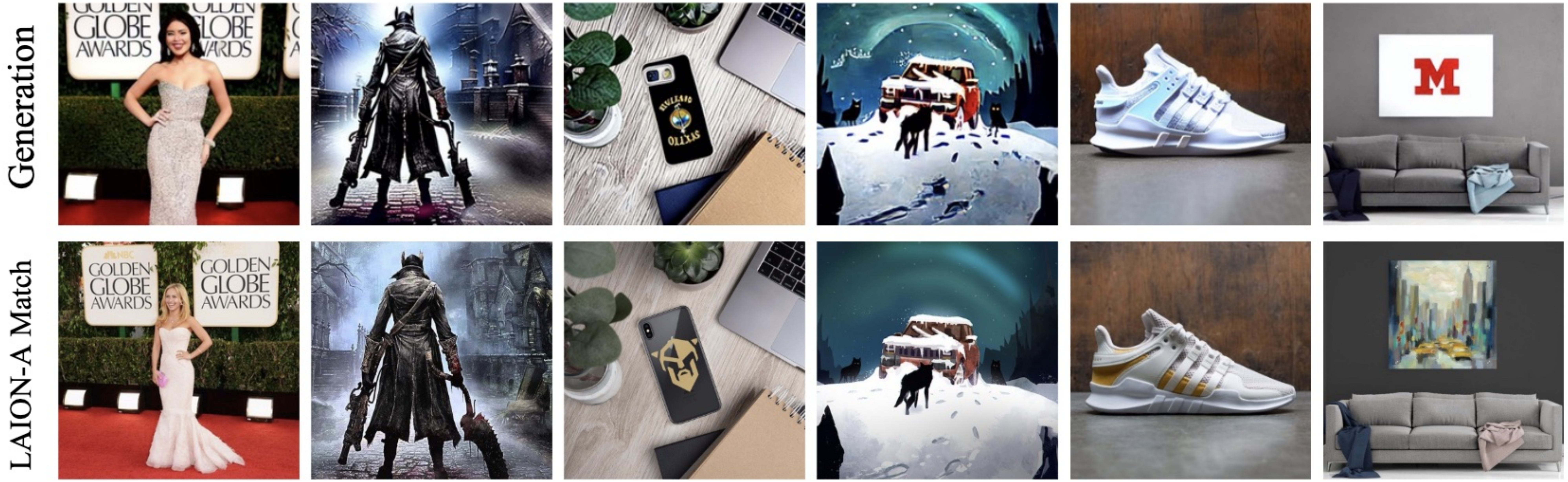
Carlini et al. (USENIX Security 2021)

Diffusion Art or Digital Forgery? Investigating Data Replication in Diffusion Models

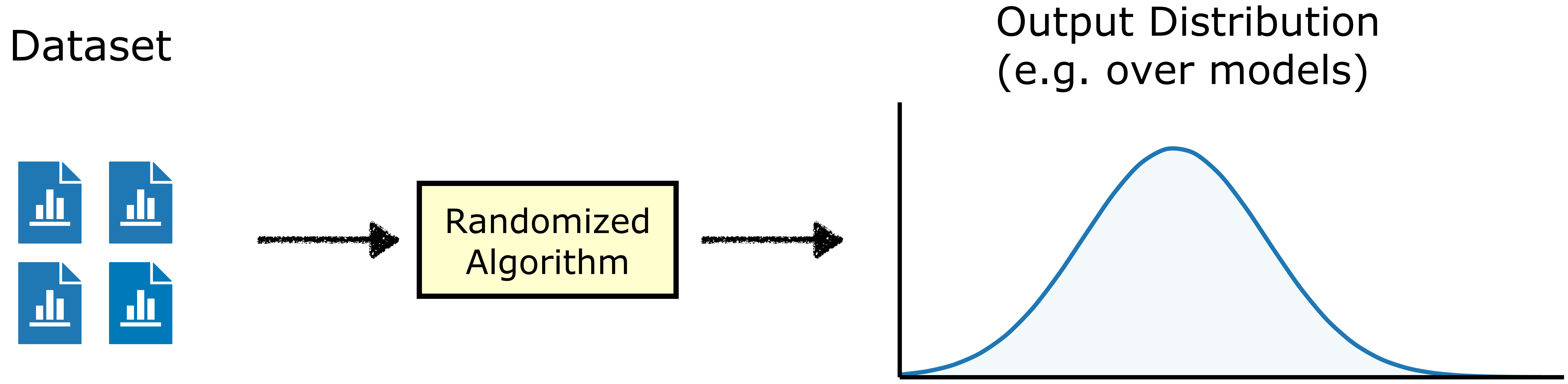
Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, Tom Goldstein

University of Maryland, College Park
{gowthami, vsingla, jgeiping, tomg}@cs.umd.edu

New York University
goldblum@nyu.edu

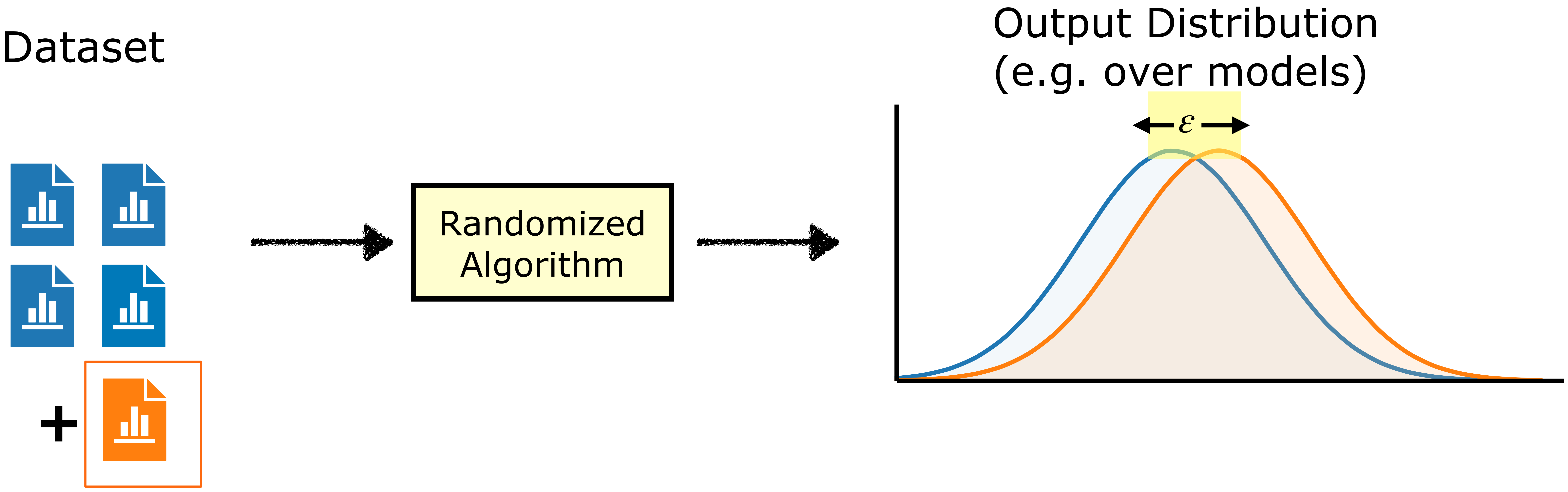


Differential privacy (DP)



Dwork, McSherry, Nissim, Smith. **Calibrating noise to sensitivity in private data analysis**. TCC 2006

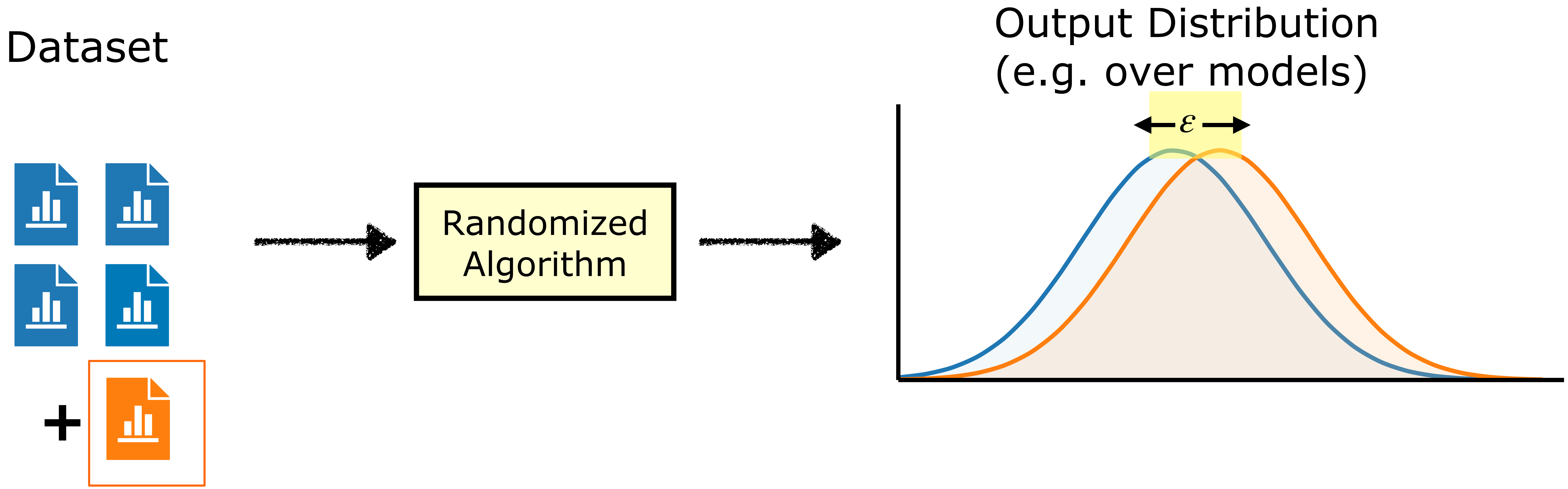
Differential privacy (DP)



A randomized algorithm is ε-**differentially private** if the addition of **one <u>unit of data</u>** does not alter its output distribution by more than ε

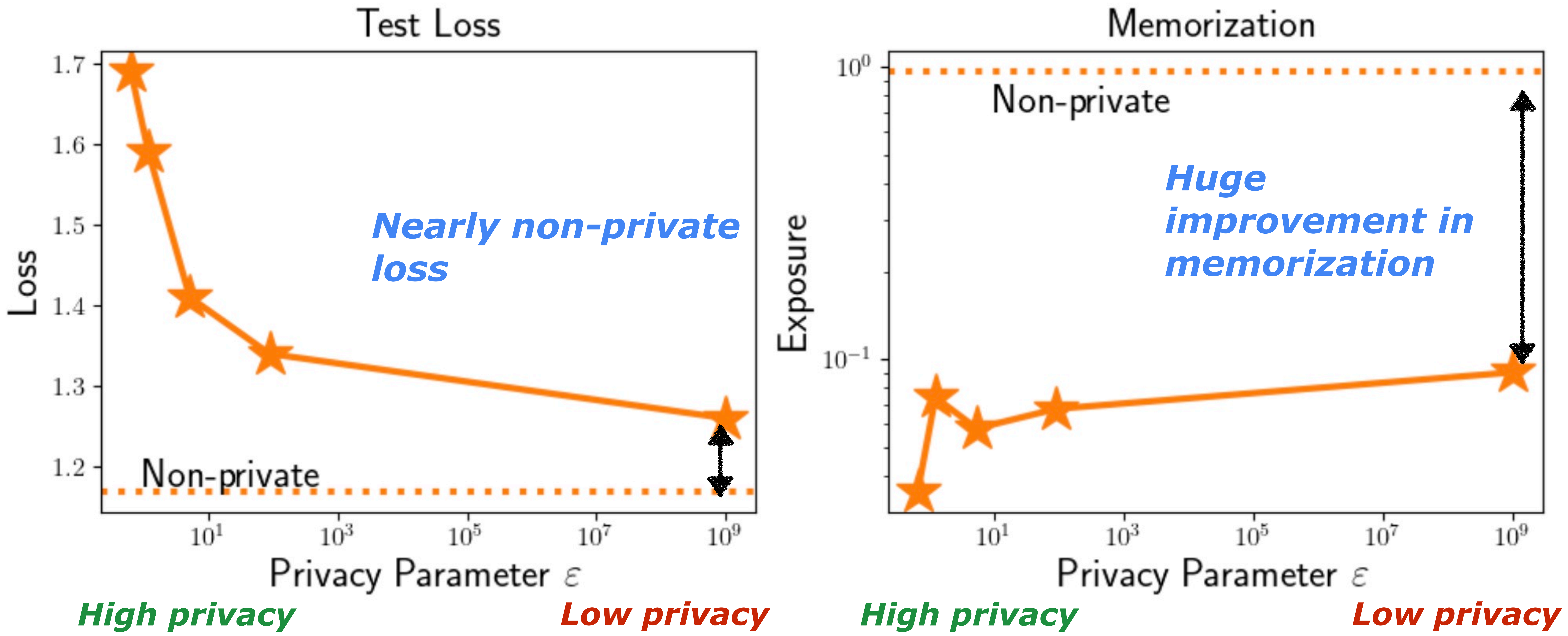
Example-level Differential privacy (DP)

Unit of data = ***example***



A randomized algorithm is ε-**differentially private** if the addition of **one *example*** does not alter its output distribution by more than ε

Differential privacy nearly eliminates memorization

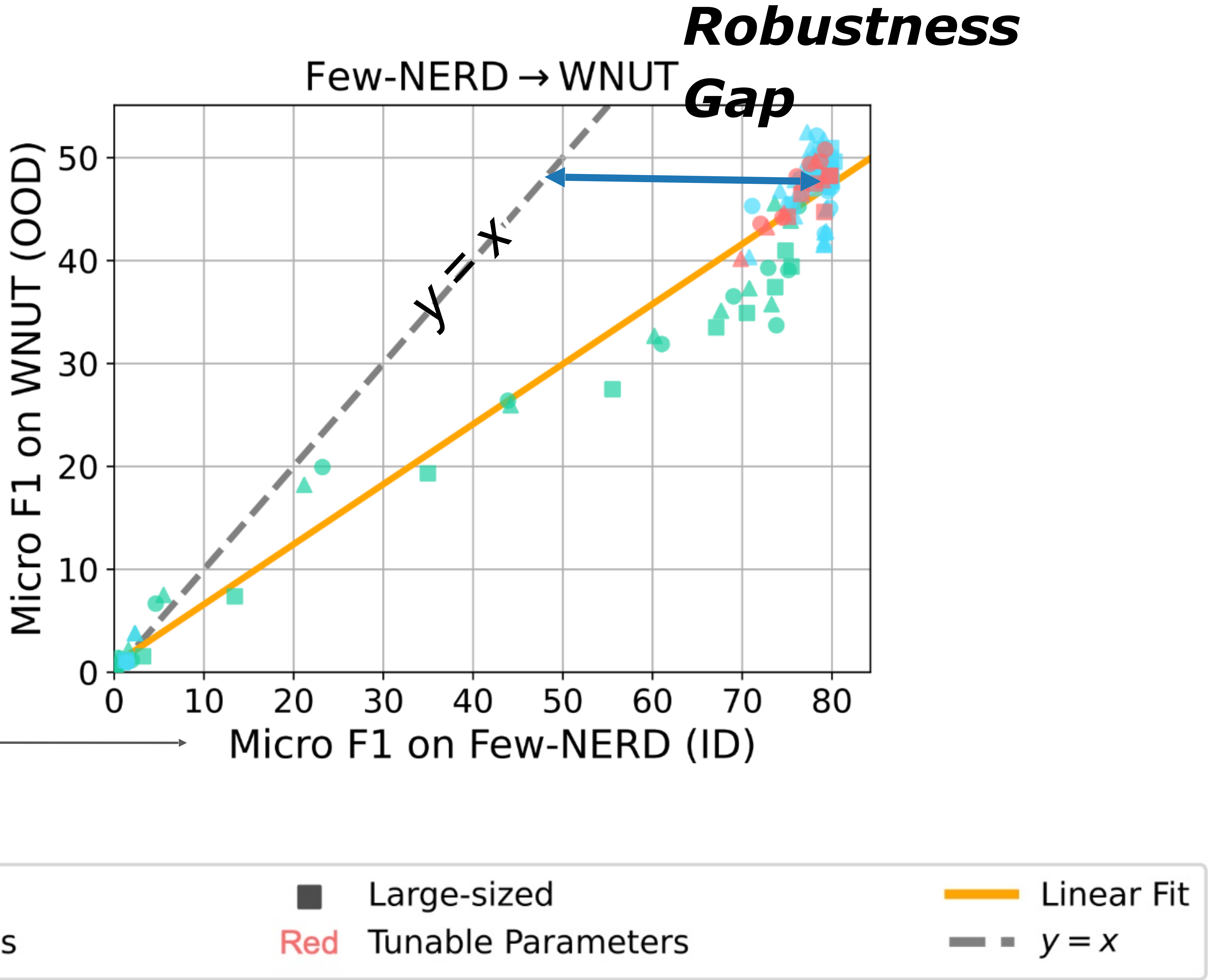


Carlini, Liu, Erlingsson, Kos, Song. **The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks.** USENIX Security 2019.

Which data do we use to train/finetune/align these models?

Test on **shifted distribution** (out-of-domain / **OOD**)

Test on **training distribution** (in-domain / **ID**)



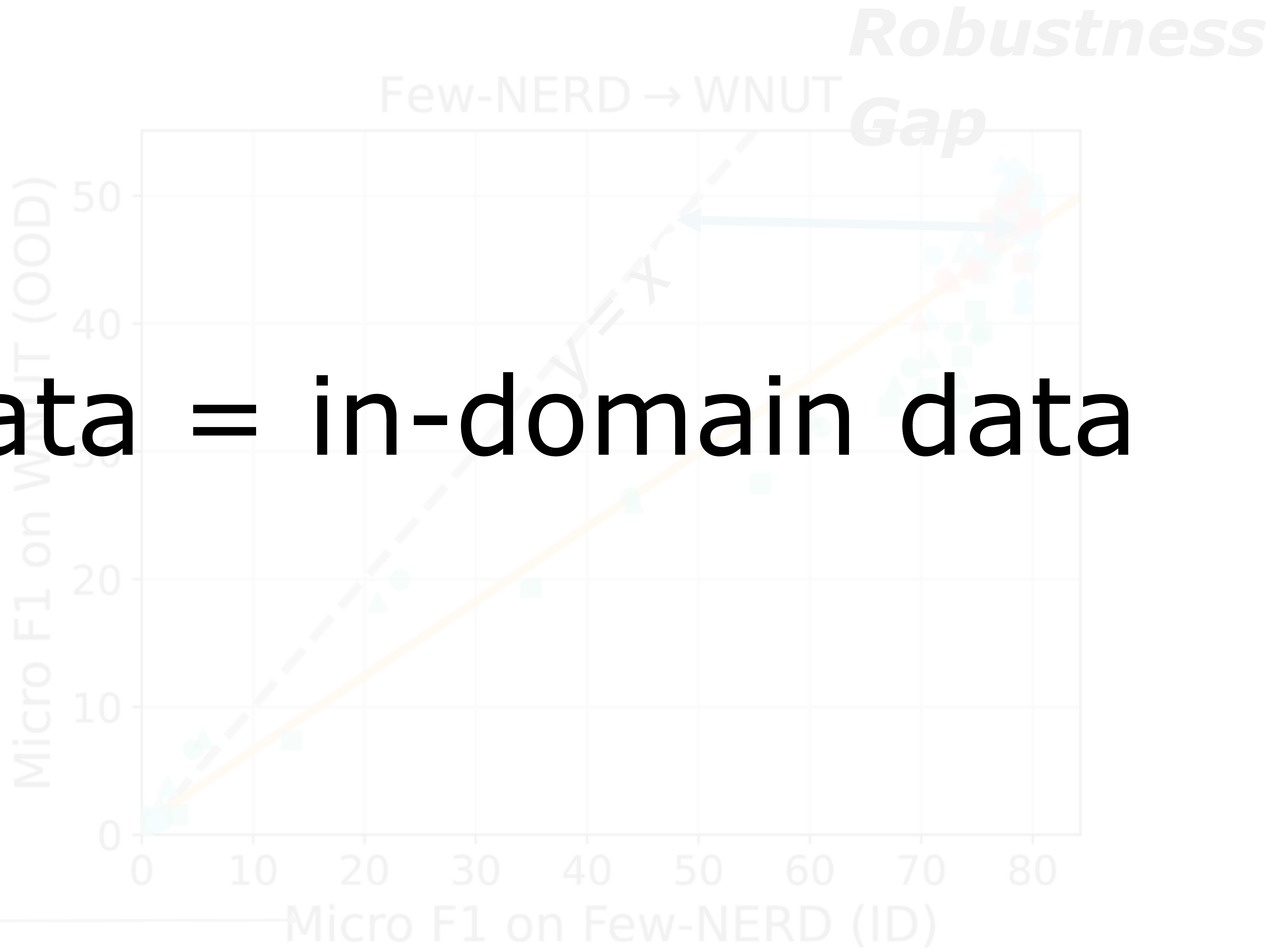
Yuan et al. **Revisiting Out-of-distribution Robustness in NLP: Benchmark, Analysis, and LLM Evaluations**. NeurIPS D&B 2023

Which data do we use to train/finetune/align these models?

Best training data = in-domain data

Test on **shifted distribution**
(out-of-domain / **OOD**)

Test on **training distribution**
(in-domain / **ID**)



Yuan et al. **Revisiting Out-of-distribution Robustness in NLP: Benchmark, Analysis, and LLM Evaluations**. NeurIPS D&B 2023

ChatGPT

Gemini

Claude

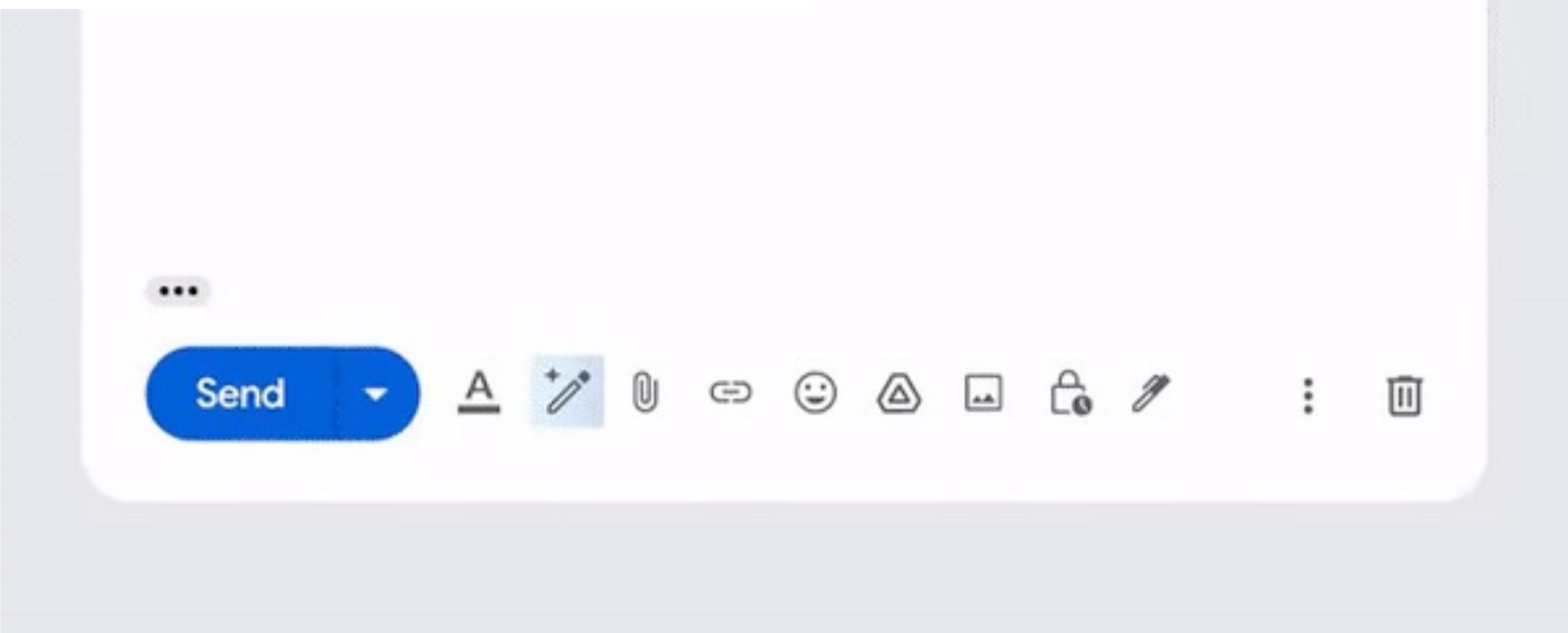
ANTHROP\C

LLAMA

MISTRAL AI_

deepseek

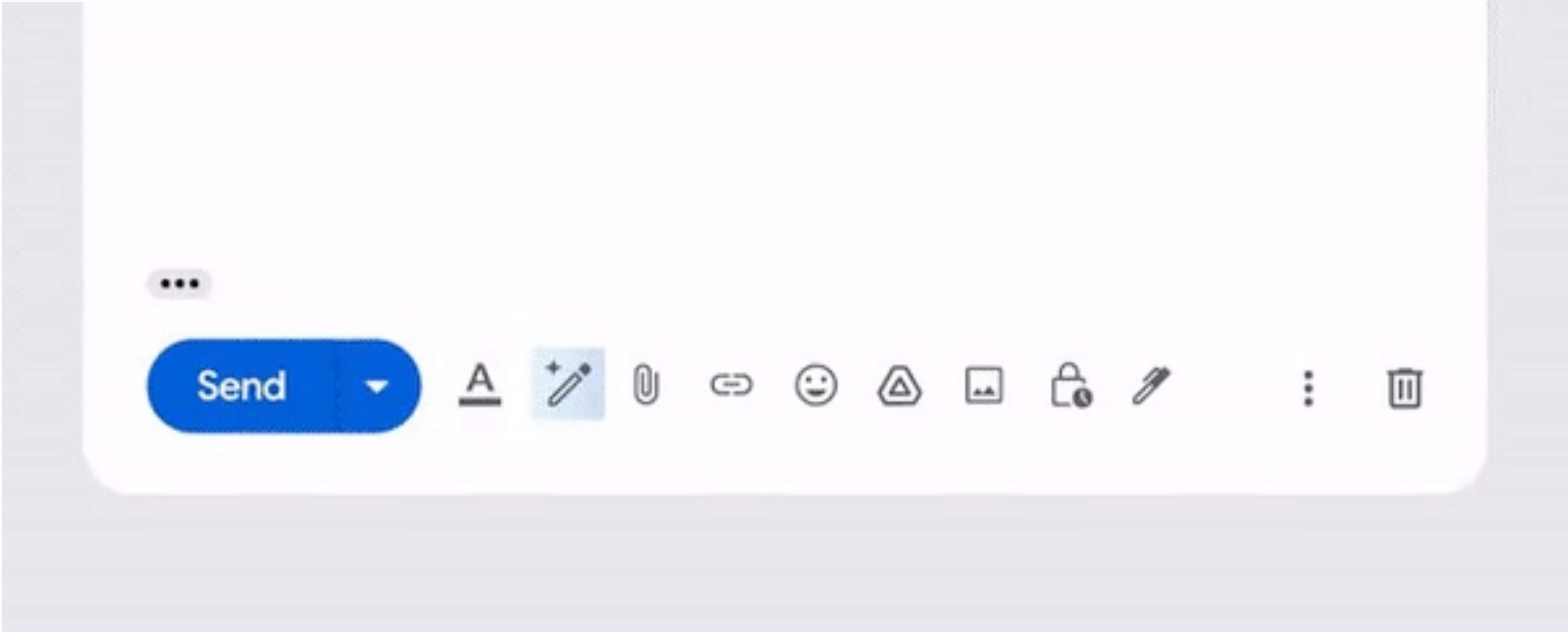
Gmail



https://blog.google/products/gmail/gmail-ai-features/

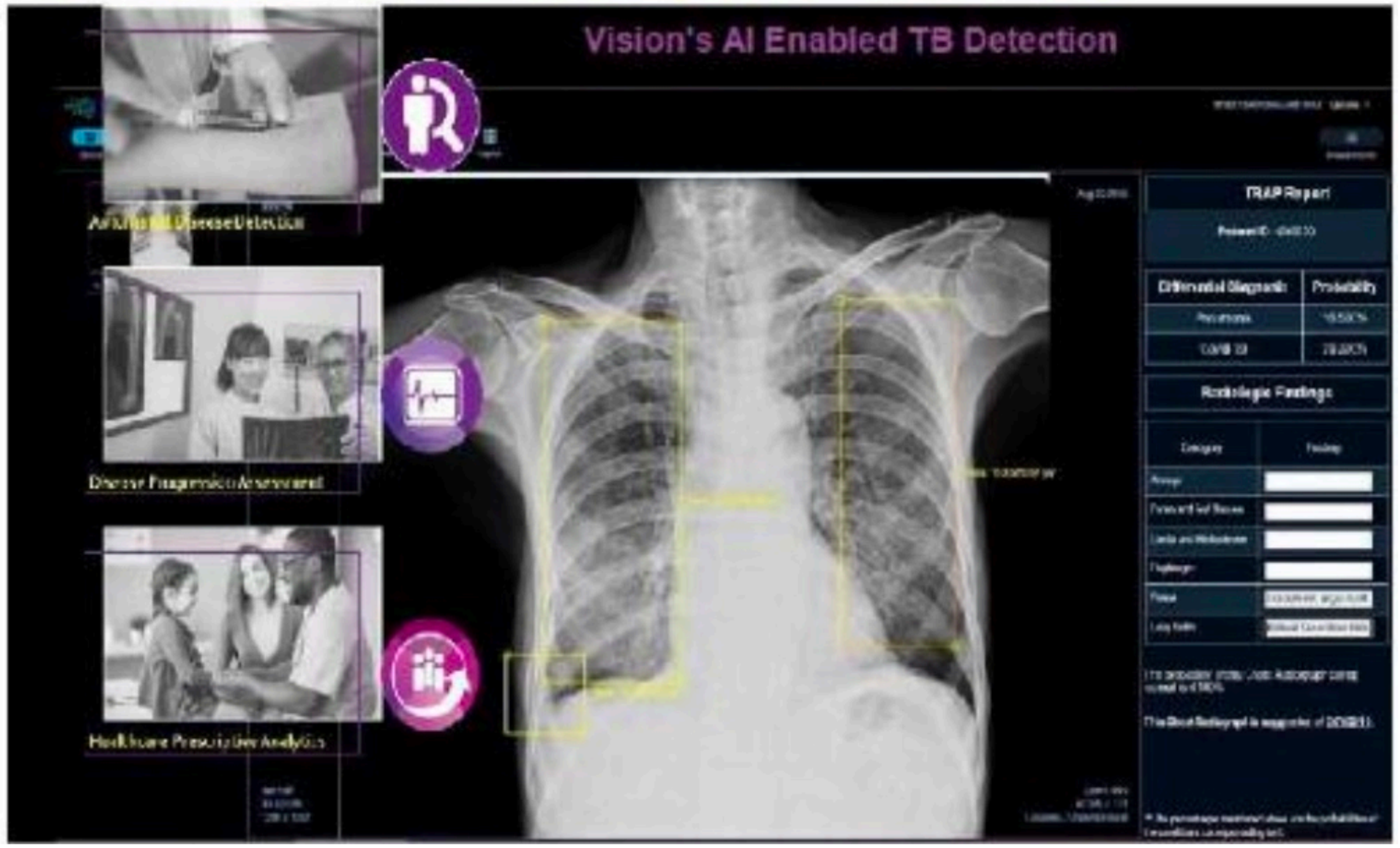
ChatGPT
Gemini
Claude
ANTHROP\C
LLAMA
MISTRAL AI_
deepseek

Gmail



https://blog.google/products/gmail/gmail-ai-features/

For many applications, in-domain data = **user data**



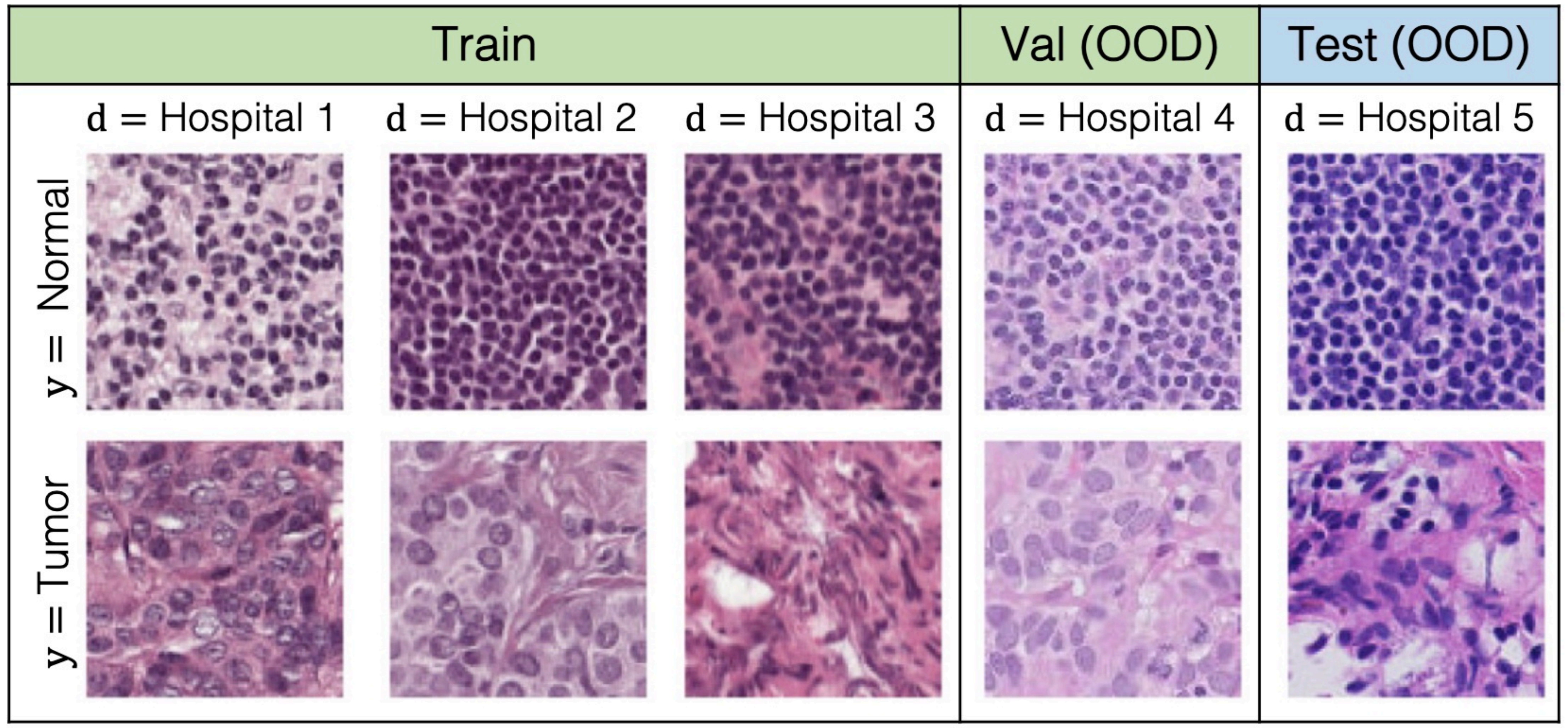
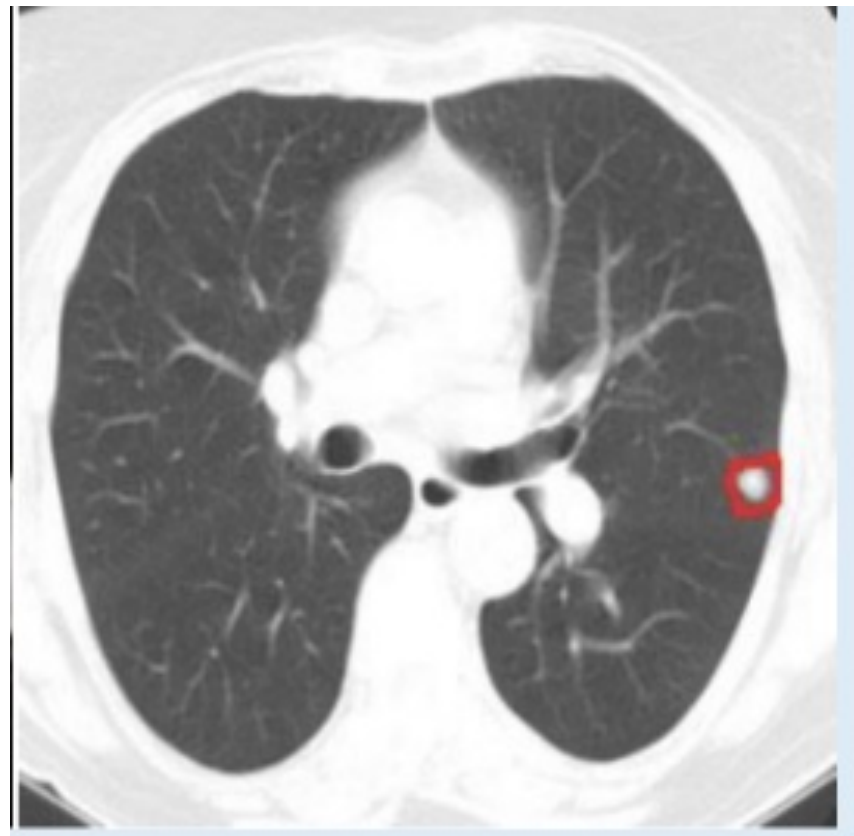
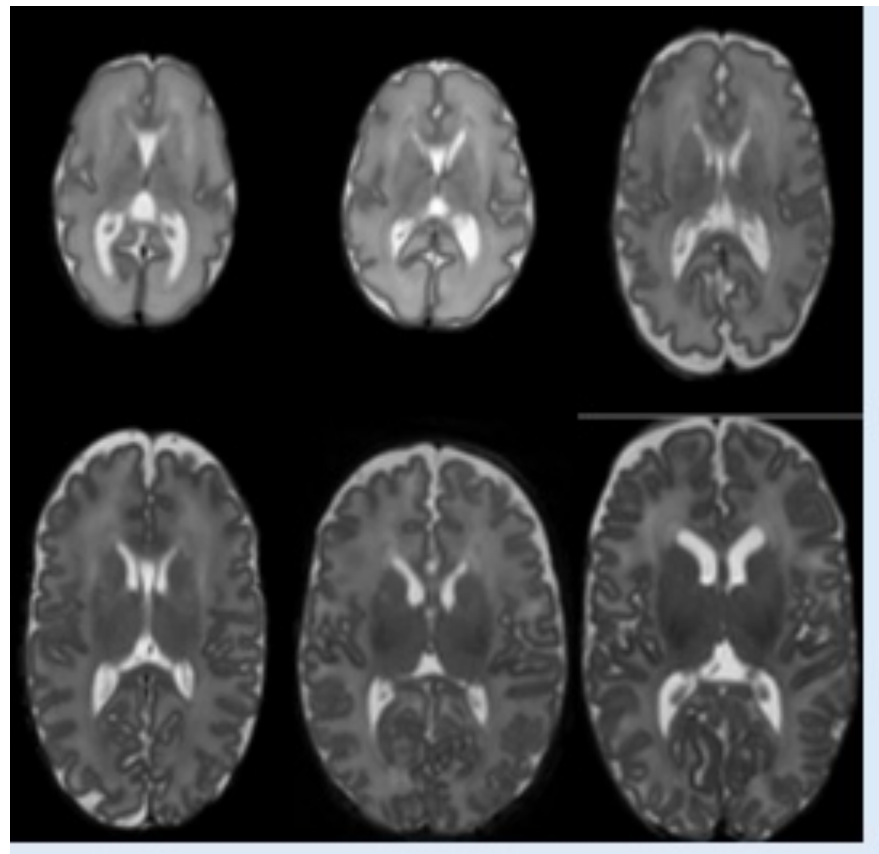
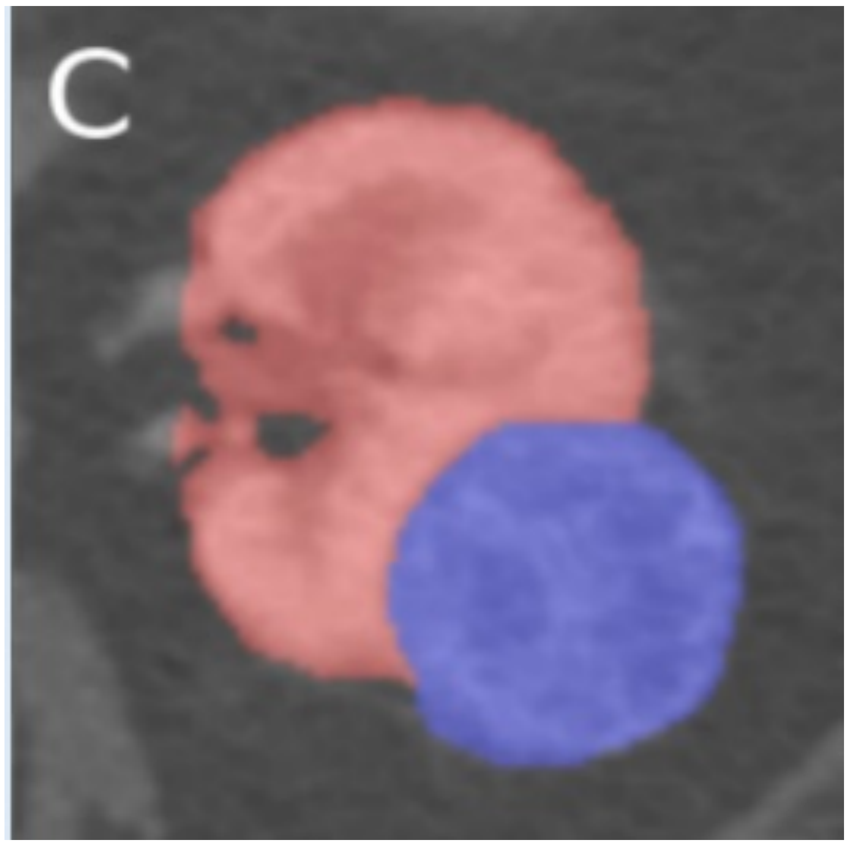
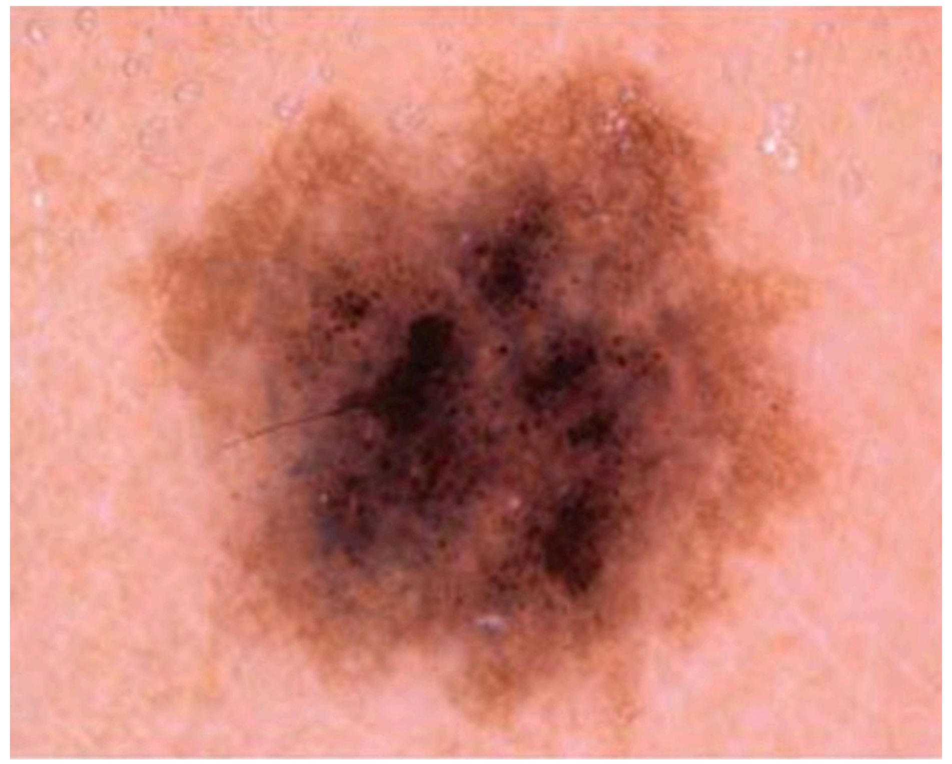


Figure 4: The CAMELYON17-WILDS dataset comprises tissue patches from different hospitals. The goal is to accurately predict the presence of tumor tissue in patches taken from hospitals that are not in the training set. In this figure, each column contains two patches, one of normal tissue and the other of tumor tissue, from the same slide.









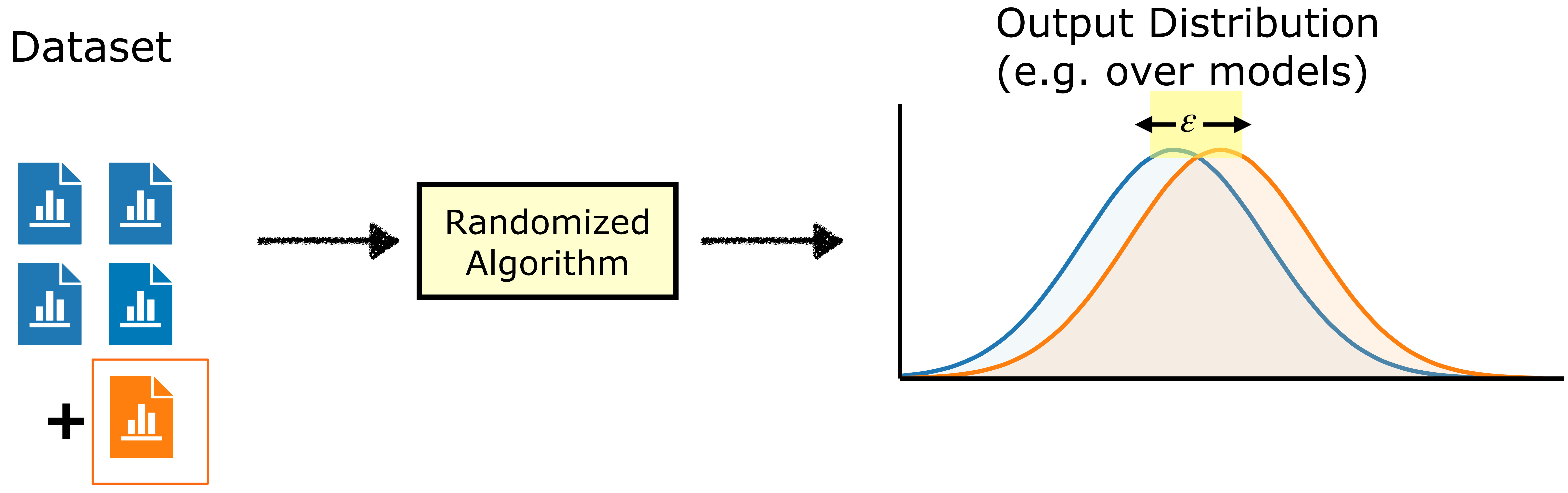
Digital Health Laws and Regulations India 2024

HIPAA
Health Insurance Portability and Accountability Act

For many applications, in-domain data = **user data**

Example-level Differential privacy (DP)

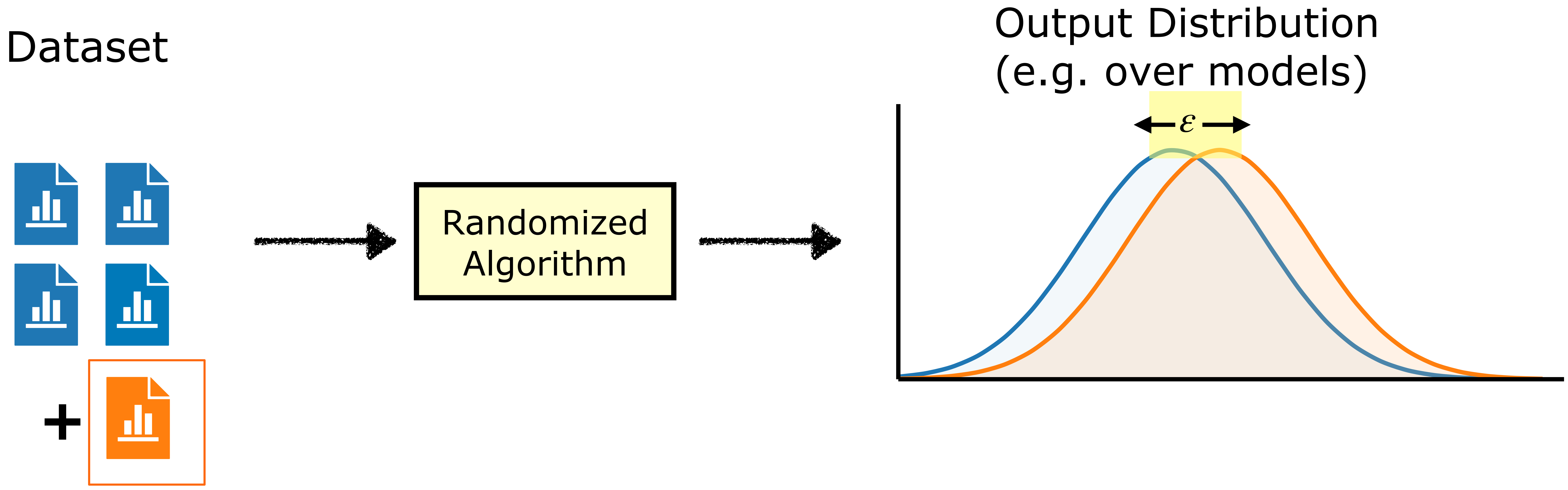
Unit of data = ***example***



A randomized algorithm is ε-**differentially private** if the addition of **one *example*** does not alter its output distribution by more than ε

User ~~**Example-level**~~ Differential privacy (DP)

Unit of data = ***user***



A randomized algorithm is ε-**differentially private** if the addition of **one *user's data*** does not alter its output distribution by more than ε

Why do we need user-level DP?

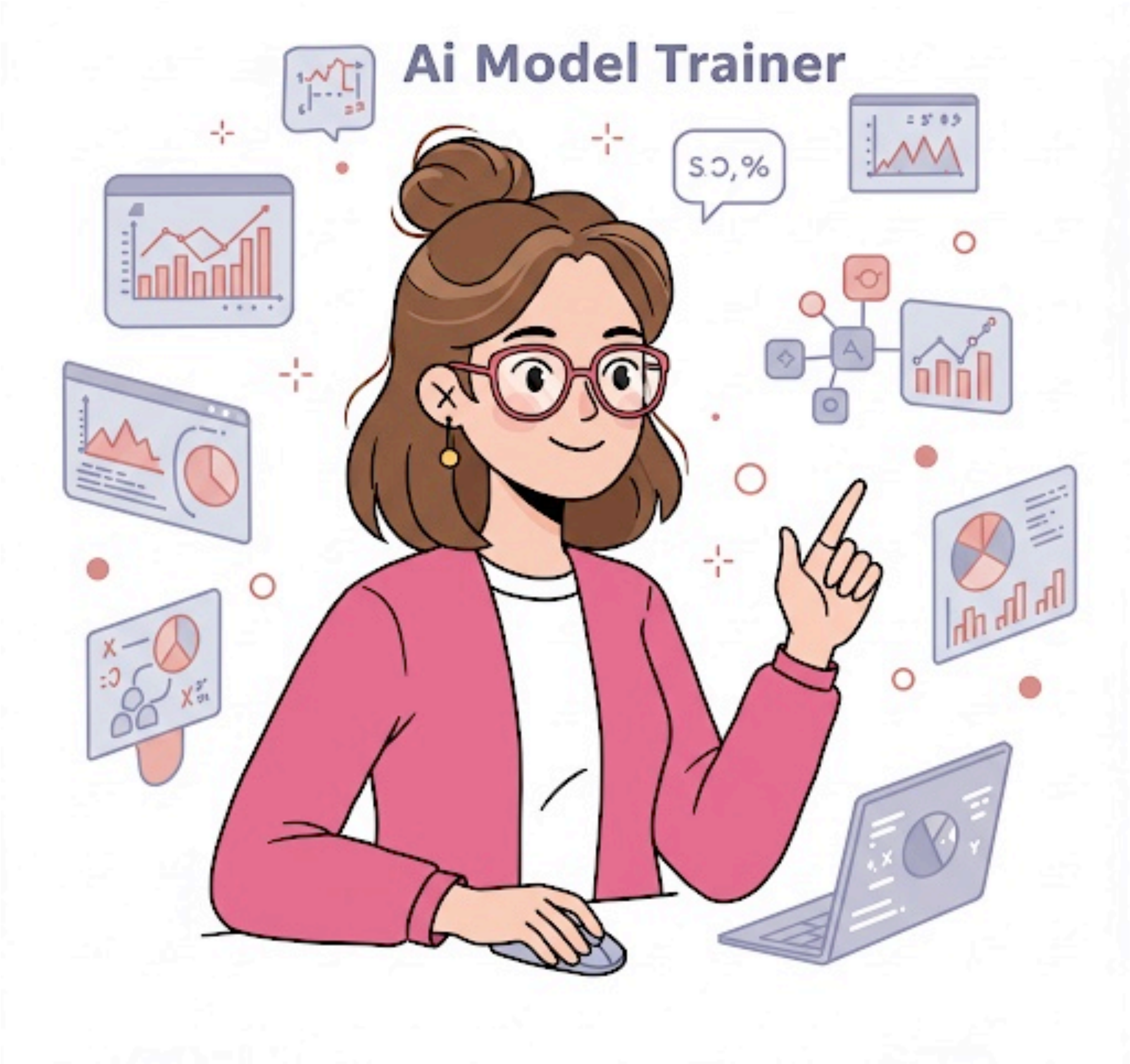
Why do we need user-level DP?

Standard LLM finetuning pipelines are susceptible to
user inference attacks*!*

Nikhil Kandpal, ***P.***, Alina Oprea, Peter Kairouz, Chris Choquette-Choo, Zheng Xu.
EMNLP (2024) Oral

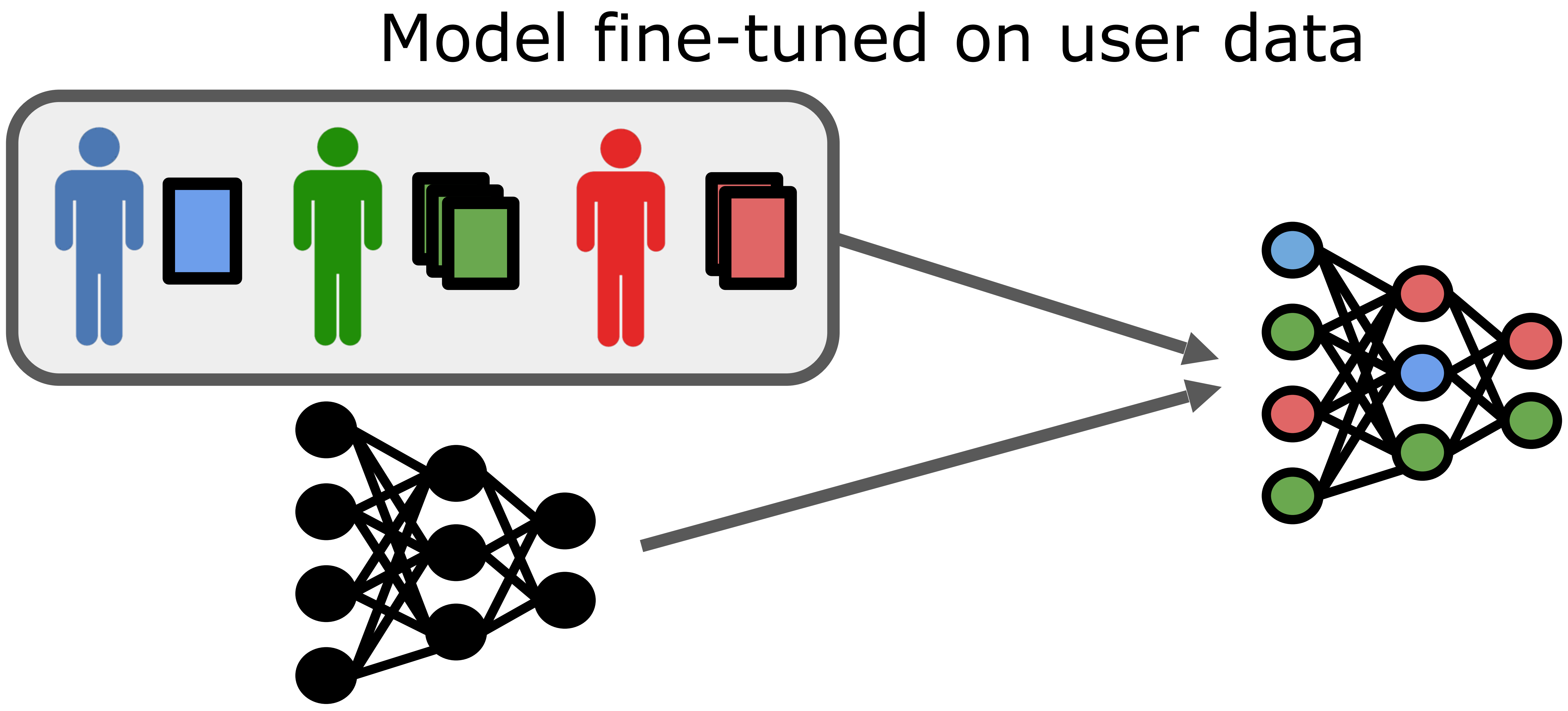
User Inference Attack

You:
Train AI model



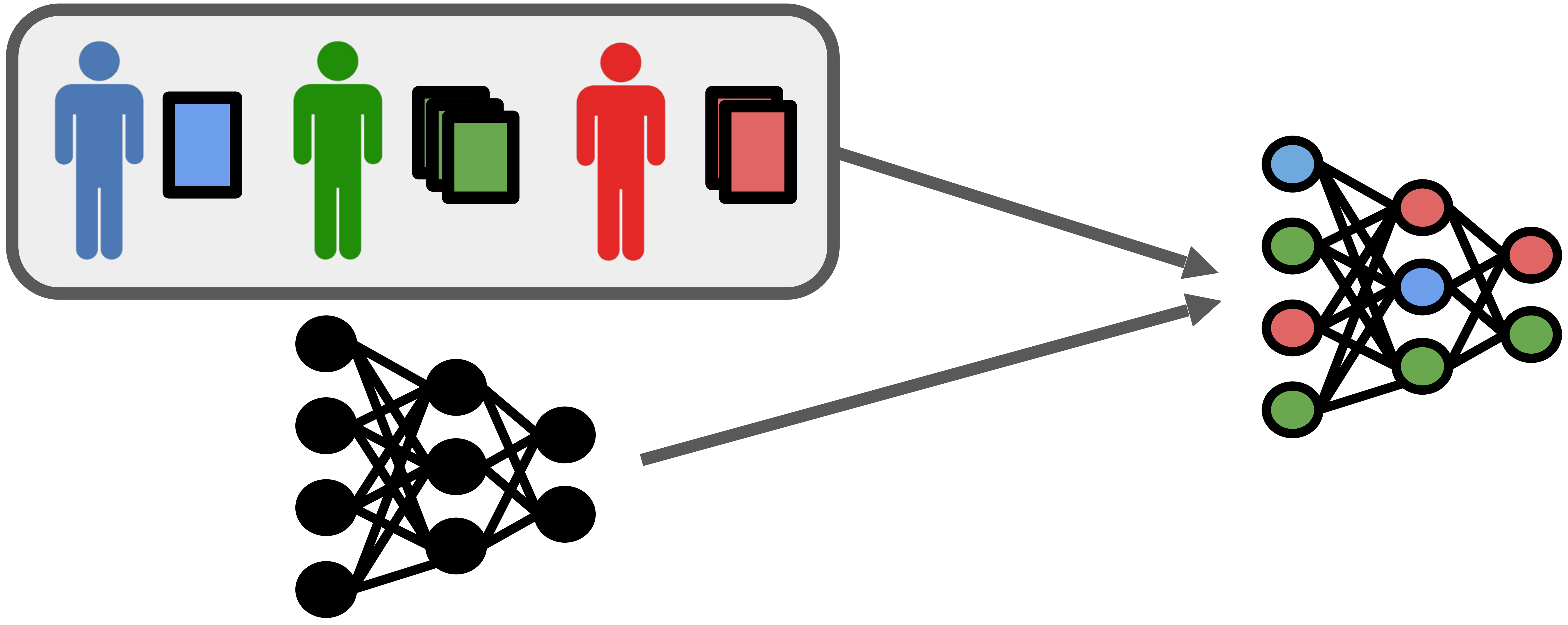
Adversary:
“Attack” the model

User Inference Attack

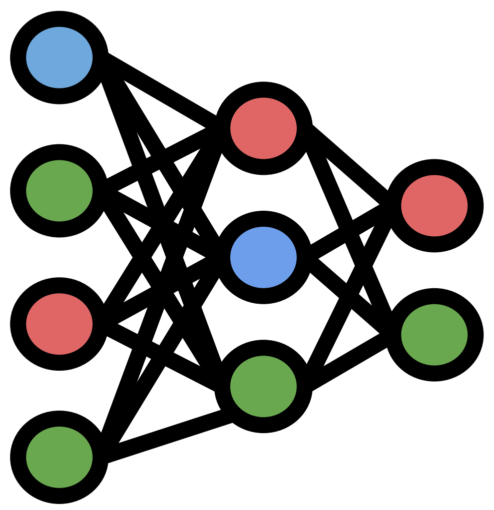


User Inference Attack

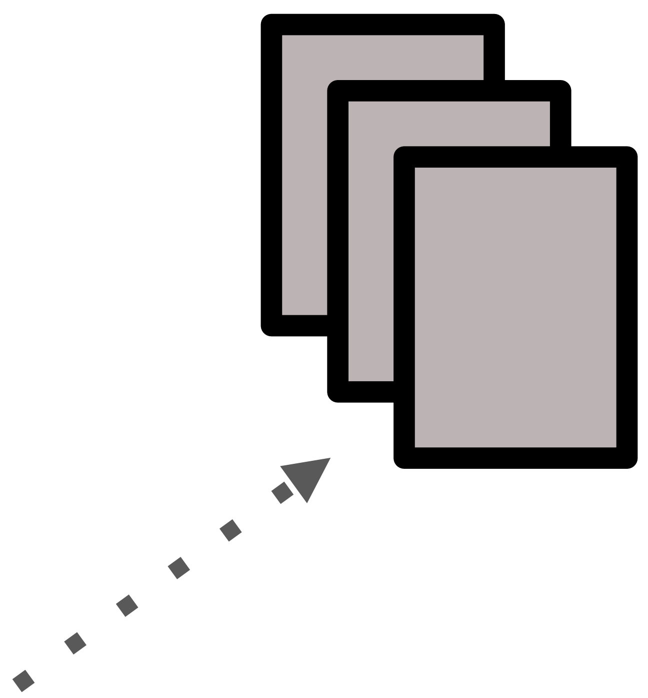
Model fine-tuned on user data



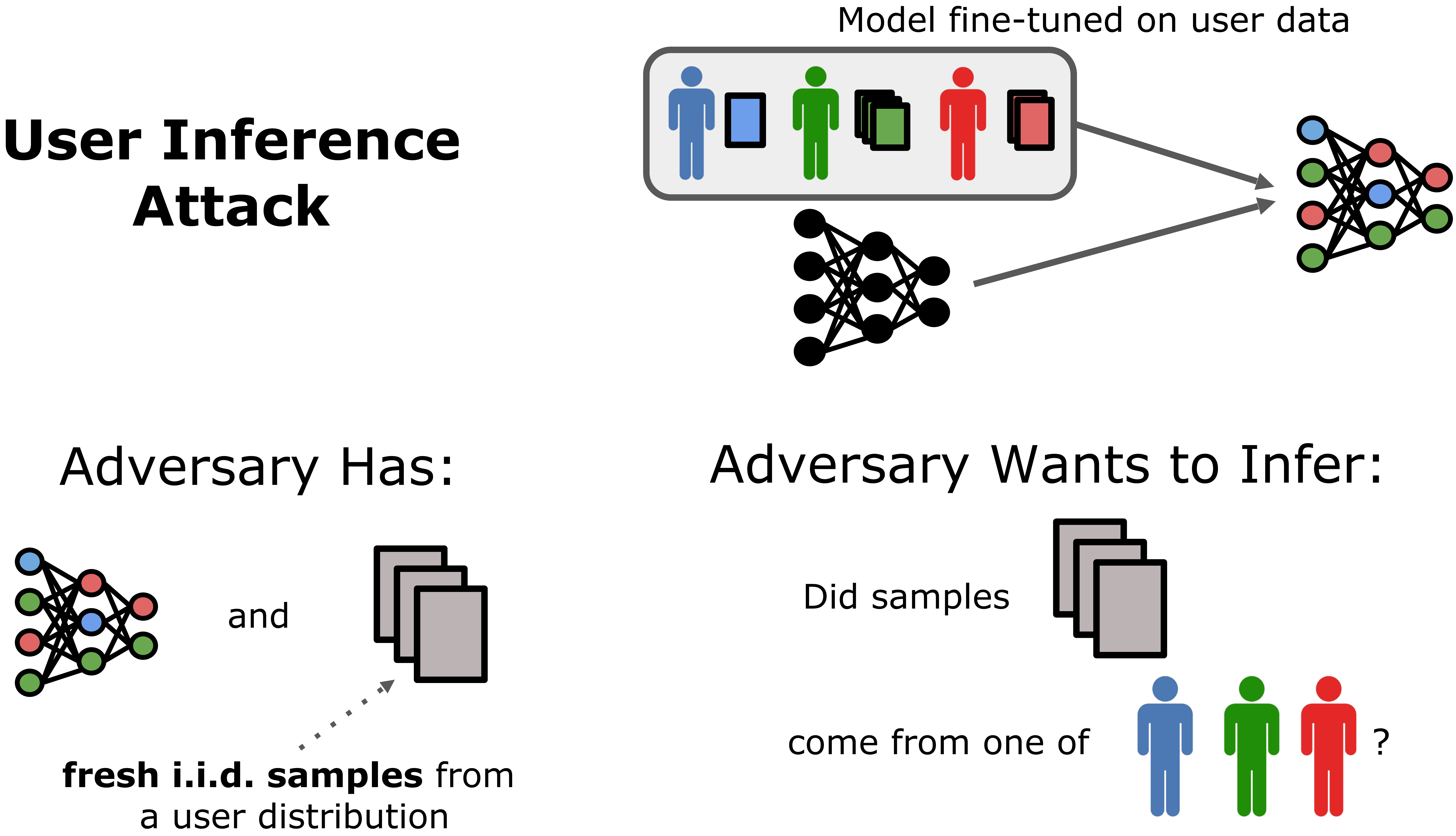
Adversary Has:



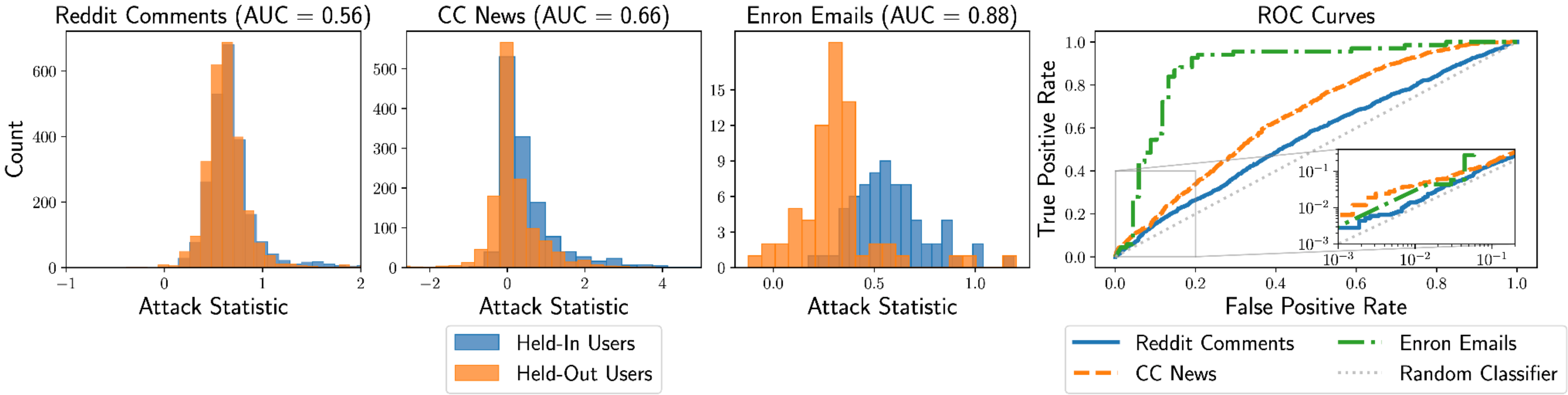
and



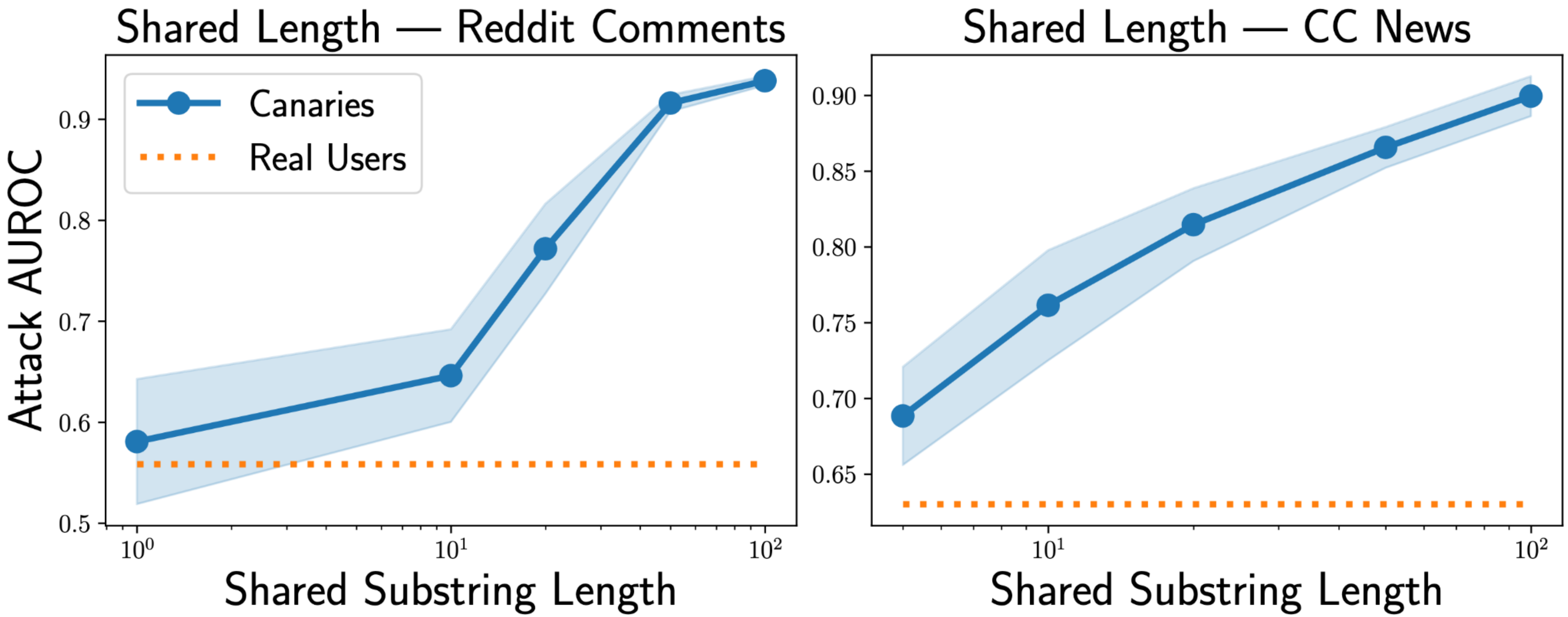
fresh i.i.d. samples from
a user distribution


User Inference Attack
Model fine-tuned on user data
Adversary Has:
and
fresh i.i.d. samples from a user distribution
Adversary Wants to Infer:
Did samples
come from one of
?

User inference is effective when #users is small and data per user is large



Short common phrases can exacerbate user inference



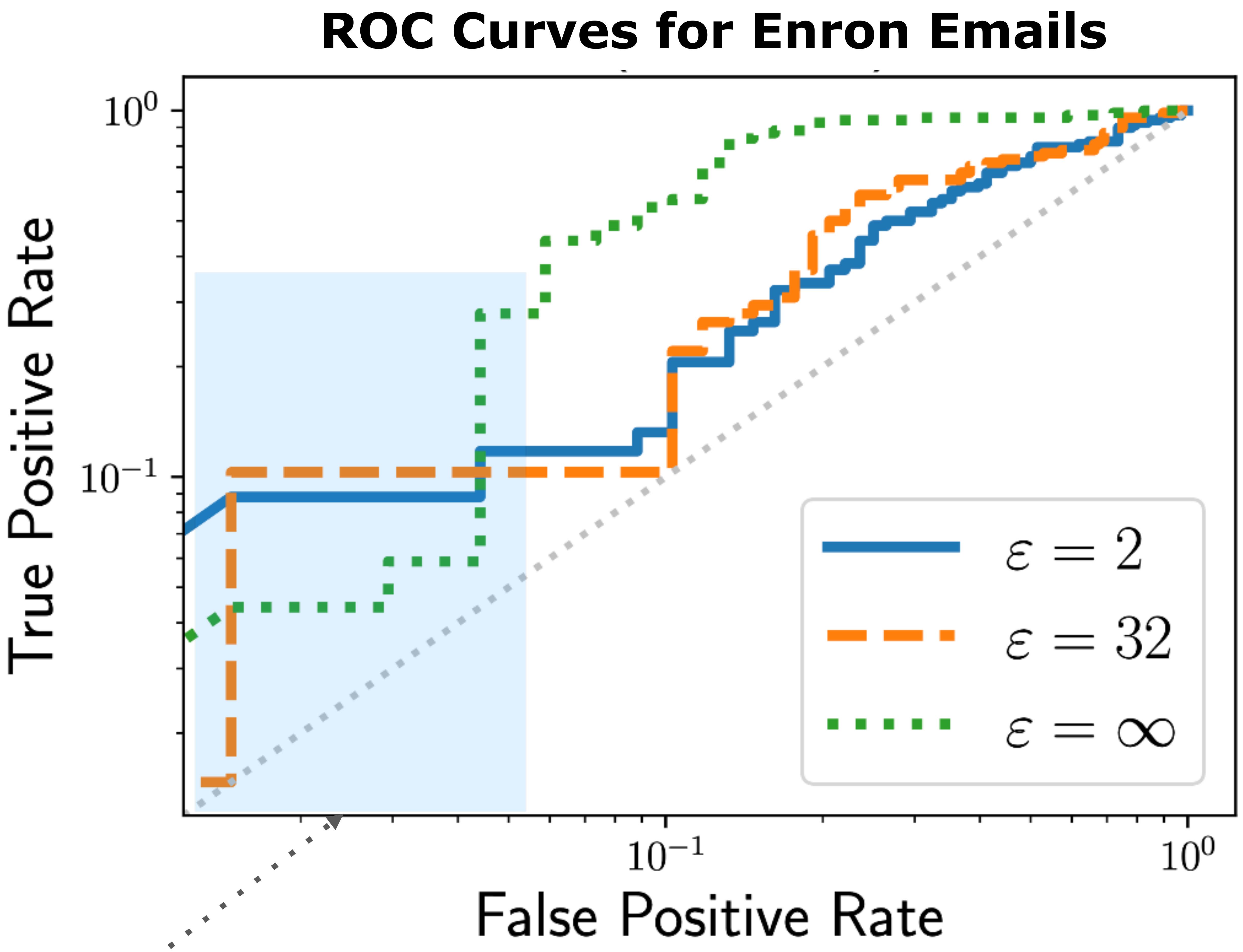
Example-level DP offers limited mitigation

AUROC:

- non-private: 88%
- ε = 32: 70%

Utility:

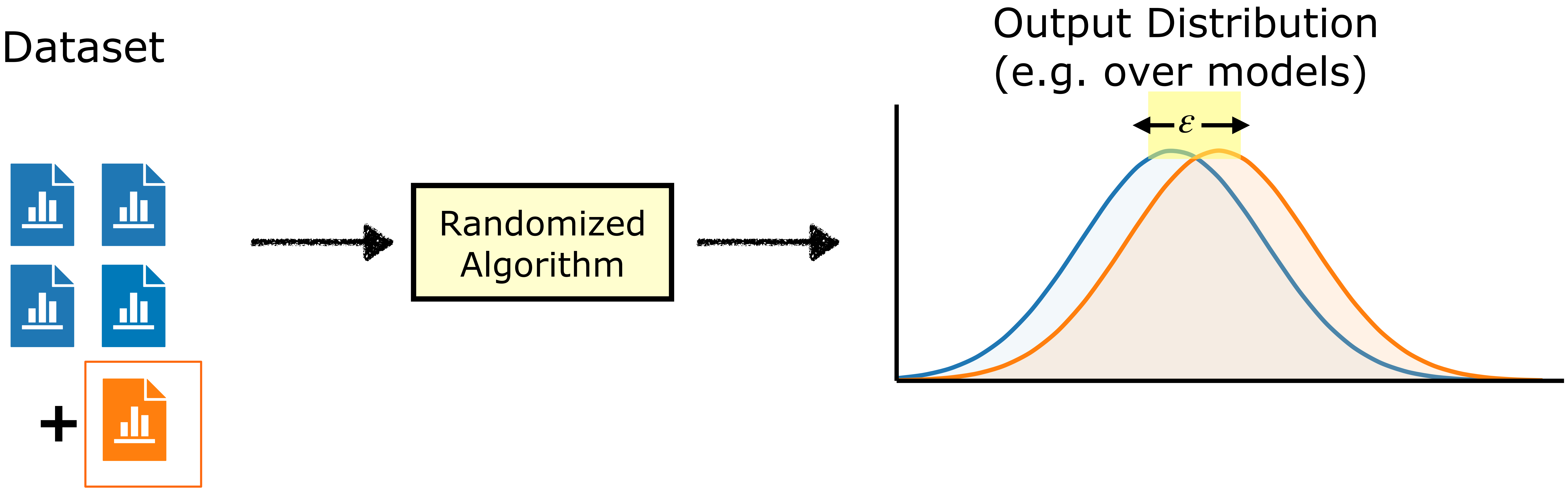
- DP model reaches what the private model achieves in 1/3 epoch



Example-level DP does not help here

User ~~Example-level~~ Differential privacy (DP)

Unit of data = ***user***



A randomized algorithm is ε-**differentially private** if the addition of **one *user's data*** does not alter its output distribution by more than ε

How do we realize user-level DP?

Outline: how do we realize user-level DP?

Learning algorithms:

Improve the runtime of SoTA correlated noise algorithms from $O(n^2)$ to $O(n \log^2 n)$
at the same performance

(n = number of steps)

Noise	***Error***	***Time / step***
Independent	$\Theta\left(\sqrt{n}\right)$	$O(1)$
Optimal Correlated	$\frac{\log(n)}{\pi}$	$O(n)$
Ours	$\frac{\log(n)}{\pi} + c$	$O\left(\log^2(n/c)\right)$

Outline: how do we realize user-level DP?

Learning algorithms:

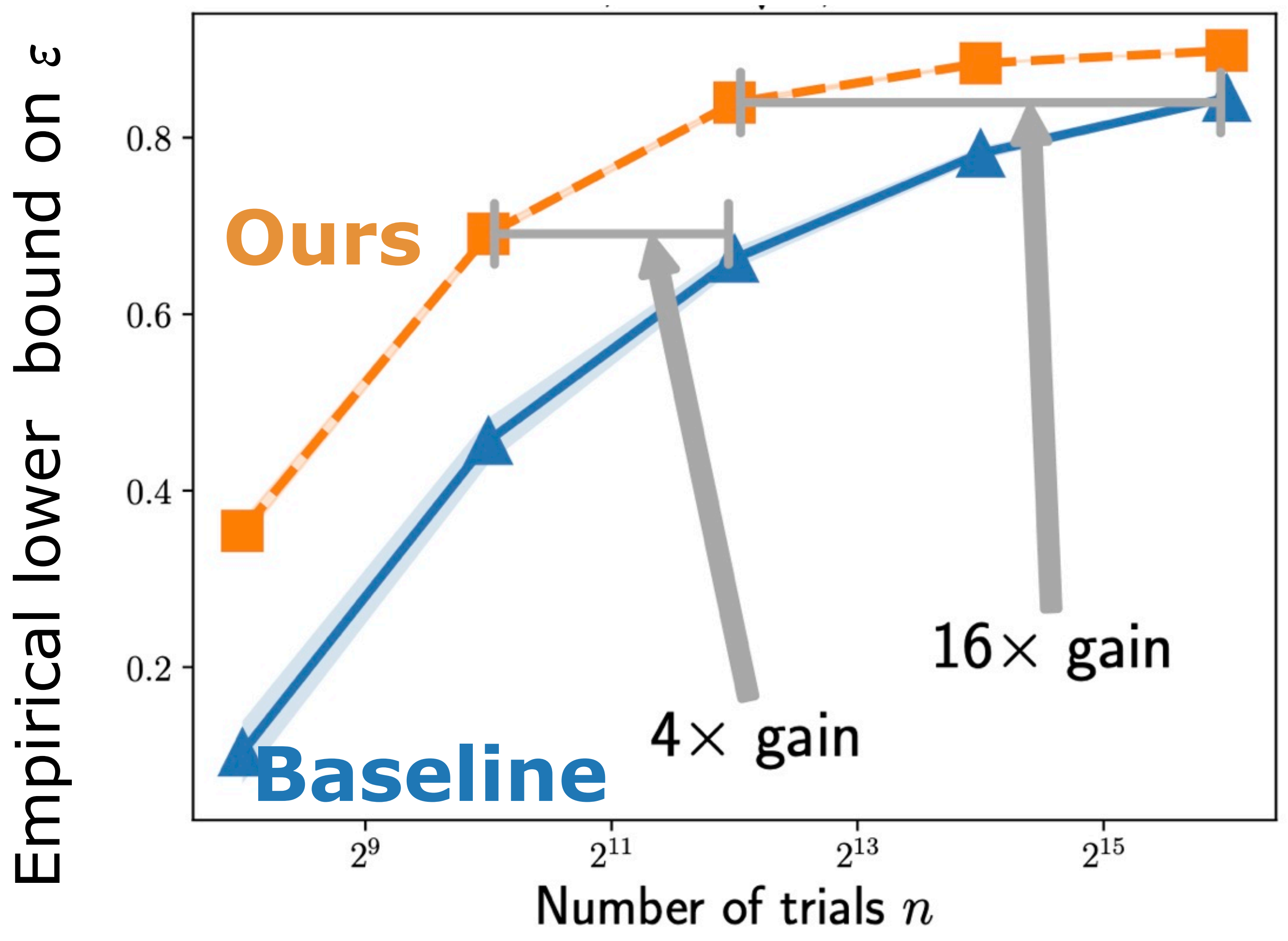
Improve the runtime of SoTA correlated noise algorithms from $O(n^2)$ to $O(n \log^2 n)$ at the same performance

(n = number of steps)

Noise	***Error***	***Time / step***
Independent	$\Theta(\sqrt{n})$	$O(1)$
Optimal Correlated	$\frac{\log(n)}{\pi}$	$O(n)$
Ours	$\frac{\log(n)}{\pi} + c$	$O\left(\log^2(n/c)\right)$

Auditing:

Randomness makes the audit more computationally efficient



Part 1: Faster learning algorithms

FOCS (2024)

Dj
Dvijotham

Brendan
McMahan

Krishna
Pillutla

Thomas
Steinke

Abhradeep
Thakurta

DP-SGD: How do we train models with **example-level** DP?

Stochastic gradient clipped to $\|g\|_2 \leq 1$ **per-example**

Independent Gaussian noise

$$\theta_{t+1} = \theta_t - \eta \left(g_t + z_t \right)$$

Learning rate

Song et al. (2013), Bassily et al. (FOCS 2014), Abadi et al. (CCS 2016)

DP-FedAvg: How do we train models with **user-level** DP?

Stochastic gradient clipped to $\|g\|_2 \leq 1$ **per-user**

Independent Gaussian noise

$$\theta_{t+1} = \theta_t - \eta \left(g_t + z_t \right)$$

Learning rate

Song et al. (2013), Bassily et al. (FOCS 2014), Abadi et al. (CCS 2016)

DP-SGD: DP Training with ***Independent*** Noise

Independent
Gaussian noise

$$\theta_{t+1} = \theta_t - \eta \left(g_t + z_t \right)$$

Song et al. (2013), Bassily et al. (FOCS 2014), Abadi et al. (CCS 2016)

DP-FTRL: DP Training with ***Correlated*** Noise

(Anti-)correlated
Gaussian noise
(z_t i.i.d. Gaussian)

$$\theta_{t+1} = \theta_t - \eta\left(g_t + \boxed{z_t - \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}}\right)$$

Kairouz, McMahan, Song, Thakkar, Thakurta, Xu. **Practical and Private (Deep) Learning without Sampling or Shuffling**. ICML 2021.
Denisov, McMahan, Rush, Smith, Thakurta. **Improved Differential Privacy for SGD via Optimal Private Linear Operators on Adaptive Streams**. NeurIPS 2022.

Prior work: (Empirically) correlated noise outperforms independent noise

Experiment: user-level DP + language modeling

Dataset: StackOverflow

Choquette-Choo, Ganesh, McKenna, McMahan, Rush, Thakurta, Xu.
(Amplified) Banded Matrix Factorization: A unified approach to private training. NeurIPS 2023

Production Training

"the first production neural network trained directly on user data announced with a formal DP guarantee."

- Google AI Blog post, Feb 2022

Google AI Blog

The latest from Google Research

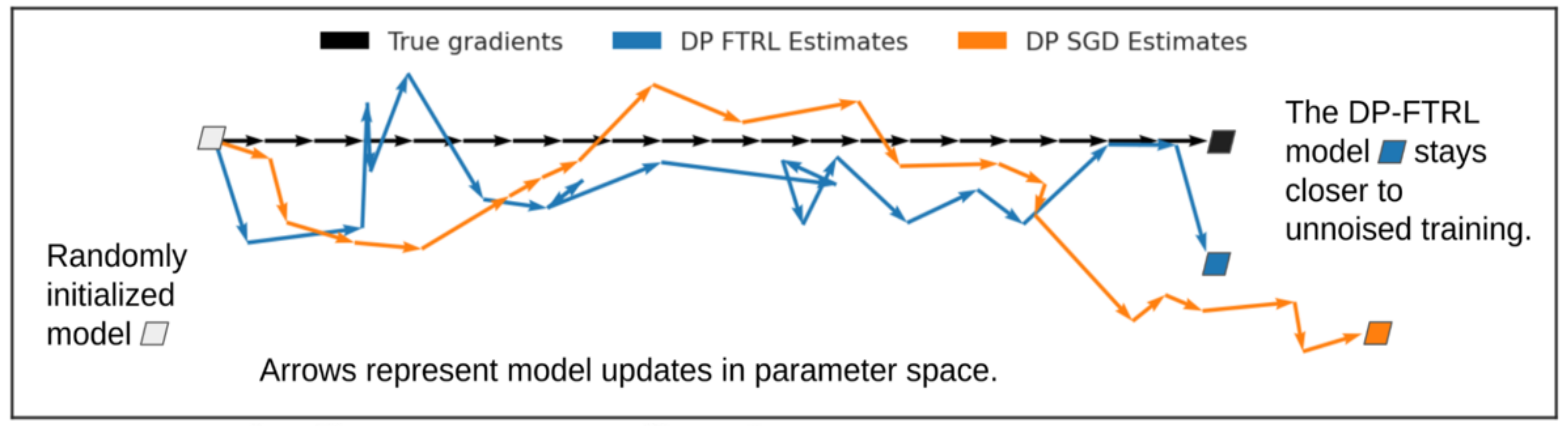
Federated Learning with Formal Differential Privacy Guarantees

Monday, February 28, 2022

Posted by Brendan McMahan and Abhradeep Thakurta, Research Scientists, Google Research

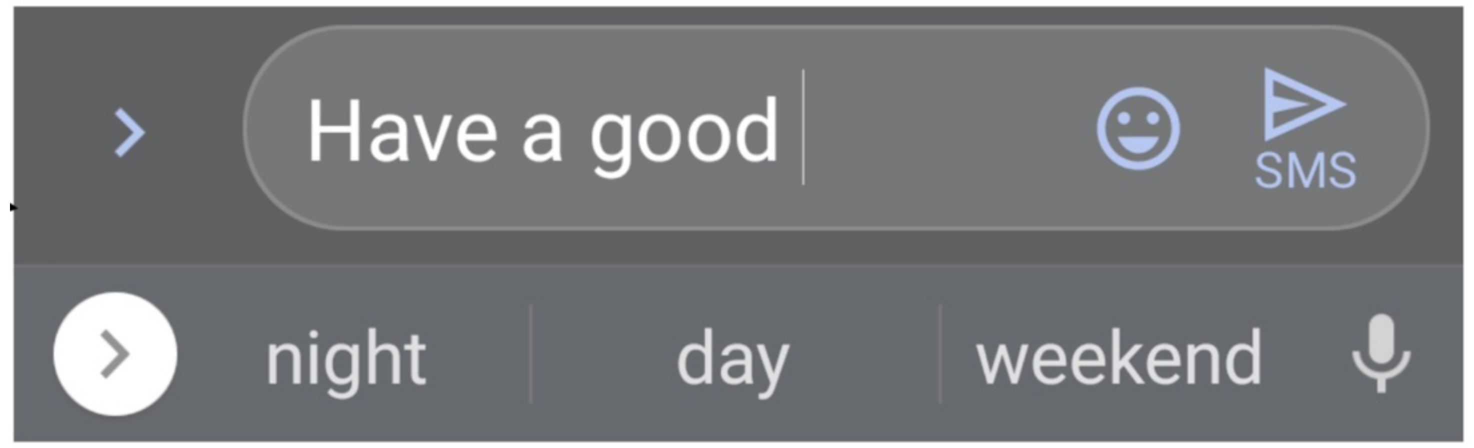
In 2017, Google introduced federated learning (FL), an approach that enables mobile devices to collaboratively train machine learning (ML) models while keeping the raw training data on each user's device, decoupling the ability to do ML from the need to store the data in the cloud. Since its introduction, Google has continued to actively engage in FL research and deployed FL to power many features in Gboard, including next word prediction, emoji suggestion and out-of-vocabulary word discovery. Federated learning is improving the "Hey Google" detection models in Assistant, suggesting replies in Google Messages, predicting text selections, and more.

While FL allows ML without raw data collection, differential privacy (DP) provides a quantifiable measure of data anonymization, and when applied to ML can address concerns about models memorizing sensitive user data. This too has been a top research priority, and has yielded one of the first production uses of DP for analytics with RAPPOR in 2014, our open-source DP library, Pipeline DP, and TensorFlow Privacy.



Data Minimization and Anonymization in Federated Learning

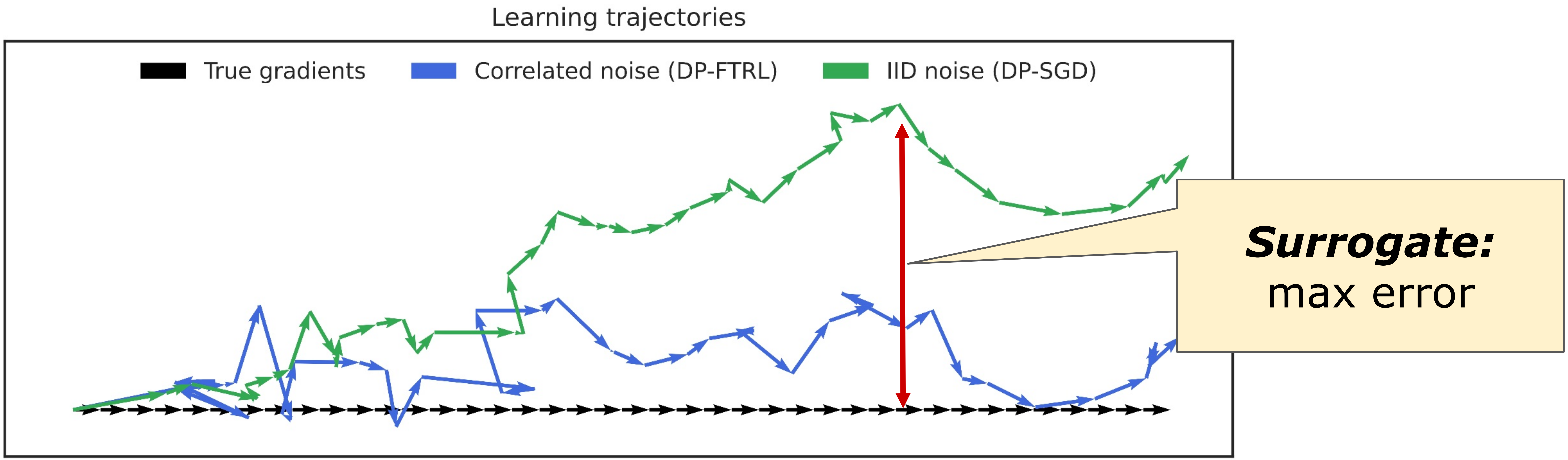
Along with fundamentals like transparency and consent, the privacy principles of data minimization and anonymization are important in ML applications that involve sensitive data.



How do we find the noise coefficients?

How do we find the noise coefficients?

Current Approach:
Find the noise coefficients to ***minimize the cumulative noise*** added to the learning trajectory (such that a given DP constraint is satisfied)



How do we find the noise coefficients?

$$\theta_{t+1} = \theta_t - \eta\Big(g_t + \underbrace{\boxed{z_t - \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}}}_{=: w_t}\Big)$$

Find the noise coefficients β_t to ***minimize the max error*** (i.e. cumulative noise added to the learning trajectory):

Surrogate Objective

$$\mathcal{E}(\beta)^2 = \max_{t \le n} \mathbb{E}_{z_\tau \sim \mathcal{N}(0,\sigma^2 I)} \left\|\sum_{\tau=0}^{t} w_\tau\right\|_2^2$$

where the variance σ^2 is chosen so that θ_t's satisfy a given DP constraint

Toeplitz mechanism: optimal max error

$$\theta_{t+1} = \theta_t - \eta\left(g_t + z_t - \sum_{\tau=1}^{t} \boxed{\beta_\tau} z_{t-\tau}\right)$$

Theorem

[Fichtenberger, Henzinger, Upadhyay (ICML '23); Dvijotham, McMahan, ***P***., Steinke, Thakurta (FOCS '24)]

For any number *n* of steps, the optimal max error is obtained by coefficients $\beta_t{}^* = t^{-3/2}$ and satisfies the bounds

$$\mathcal{E}(\beta^*) = \frac{\log n}{\pi} + \text{constant}$$

Toeplitz mechanism: optimal max error

$$\theta_{t+1} = \theta_t - \eta\Big(g_t + z_t - \sum_{\tau=1}^{t} \boxed{\beta_\tau} z_{t-\tau}\Big)$$

Theorem

[Fichtenberger, Henzinger, Upadhyay (ICML '23); Dvijotham, McMahan, ***P***., Steinke, Thakurta (FOCS '24)]

For any number *n* of steps, the optimal max error is obtained by coefficients $\beta_t{}^* = t^{-3/2}$ and satisfies the bounds

$$\mathcal{E}(\beta^*) = \frac{\log n}{\pi} + \text{constant}$$

Exponential improvement over independent noise

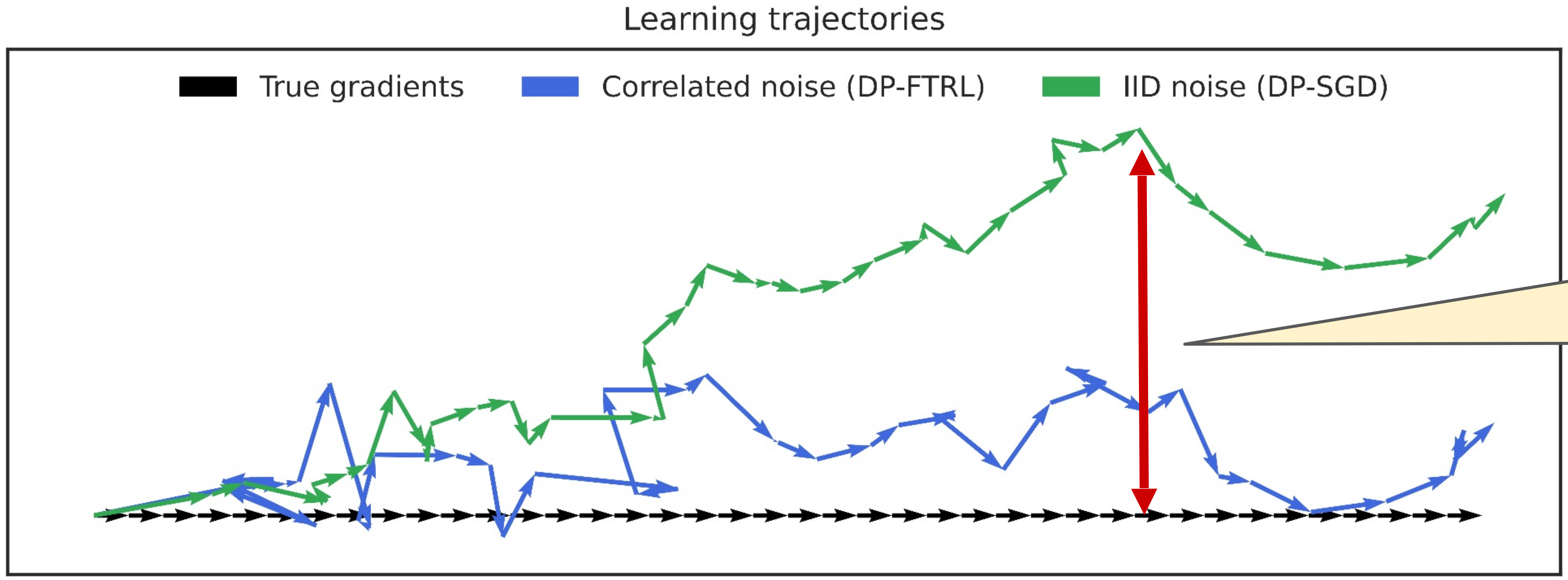
$$\mathcal{E}(\beta^{\mathrm{SGD}}) = \Theta(\sqrt{n})$$

Our challenge: running time

$$\theta_{t+1} = \theta_t - \eta\left(g_t + z_t - \boxed{\sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}}\right)$$

$\beta_t = t^{-3/2}$

Quadratic time complexity:
Noise generation requires $O(t)$ time in iteration t



Surrogate:
max error

	Max Error	***Noise generation time*** (in iteration t)
Independent noise	$\Theta(\sqrt{n})$	$O(\mathrm{dim})$
Optimal correlated noise	$\frac{\log n}{\pi} + c$	$O(t \cdot \mathrm{dim})$

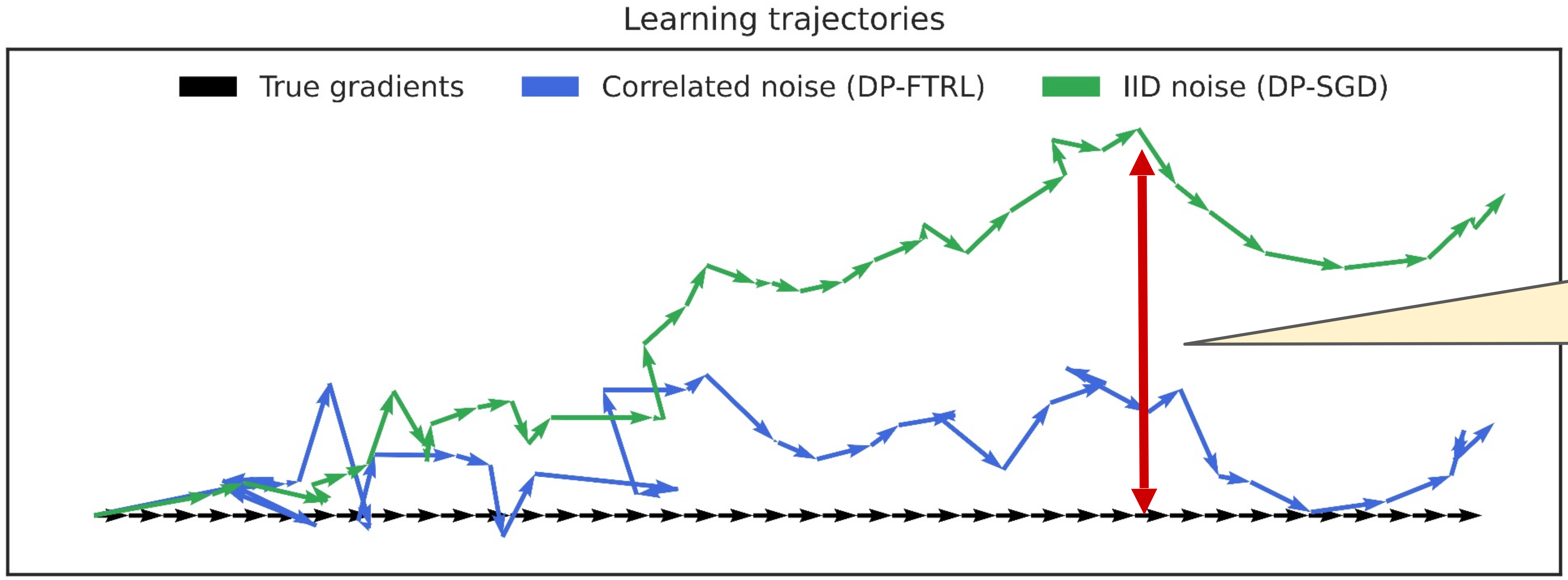
A first attempt: the banded mechanism

Set $\beta_t = 0$ for $t > b$

Then, we only have to sum b terms in $\sum_{\tau=1}^{b} \beta_\tau z_{t-\tau}$

Linear complexity:
Noise generation requires $O(b)$ time in each iteration

Choquette-Choo, Ganesh, McKenna, McMahan, Rush, Thakurta, Xu.
(Amplified) Banded Matrix Factorization: A unified approach to private training. NeurIPS 2023



Surrogate:
max error

	Max Error	***Noise generation time*** (in iteration t)
Independent noise	$\Theta(\sqrt{n})$	$O(\mathrm{dim})$
Optimal correlated noise	$\frac{\log n}{\pi} + c$	$O(t \cdot \mathrm{dim})$
b-***Banded***	$O\Big((\sqrt{n/b} - 1)\log b\Big)$	$O(b \cdot \mathrm{dim})$

Kalinin and Lampert. **Banded Square Root Matrix Factorization for Differentially Private Model Training.** NeurIPS 2024

Our approach: Intuition

Consider an exponentially decaying sequence $\beta_t = \alpha\, \lambda^{t-1}$.

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using the recurrence $w_{t+1} = \alpha\, z_t + \lambda\, w_{t-1}$

Linear complexity:
Noise generation requires *O*(dim) time in each iteration

Our approach: Intuition

Consider an exponentially decaying sequence $\beta_t = \alpha\, \lambda^{t-1}$.

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using the recurrence $w_{t+1} = \alpha\, z_t + \lambda\, w_{t-1}$

Linear complexity:
Noise generation requires *O*(dim) time in each iteration

Our approach: Intuition

Consider sums of exponentials: $\beta_t = \alpha_1 \lambda_1^{t-1} + \alpha_2 \lambda_2^{t-1}$

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using

Our approach: Intuition

$$\boxed{\beta_t'} + \boxed{\beta_t''}$$

Consider sums of exponentials: $\beta_t = \boxed{\alpha_1 \lambda_1^{t-1}} + \boxed{\alpha_2 \lambda_2^{t-1}}$

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using

Our approach: Intuition

$$\boxed{\beta_t'} + \boxed{\beta_t''}$$

Consider sums of exponentials:

$$\beta_t = \boxed{\alpha_1 \lambda_1^{t-1}} + \boxed{\alpha_2 \lambda_2^{t-1}}$$

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using

$$w_t' = \sum_{\tau=1}^{t} \beta_\tau' z_{t-\tau}$$

$$w_t'' = \sum_{\tau=1}^{t} \beta_\tau'' z_{t-\tau}$$

$$w_t = w_t' + w_t''$$

Our approach: Intuition

$$\boxed{\beta_t'} + \boxed{\beta_t''}$$

Consider sums of exponentials: $\beta_t = \boxed{\alpha_1 \lambda_1^{t-1}} + \boxed{\alpha_2 \lambda_2^{t-1}}$

Then, we can compute the correlated noise $w_t = \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}$

using

$$w_t' = \sum_{\tau=1}^{t} \beta_\tau' z_{t-\tau}$$

$$w_t'' = \sum_{\tau=1}^{t} \beta_\tau'' z_{t-\tau}$$

$$w_t = w_t' + w_t''$$

Linear time + **space**

Our approach: **B**uffered **L**inear **T**oeplitz (BLT) Mechanism

Approximate the optimal noise coefficients with *d* exponentials as

$$\beta_t \approx \beta'_t = \sum_{i=1}^{d} \alpha_i \lambda_i^{t-1}$$

Time & space complexity:
O(*d* x dimension)

Our approach: **B**uffered **L**inear **T**oeplitz (BLT) Mechanism

	Max Error	***Noise generation time*** (in iteration t)
Independent noise	$\Theta(\sqrt{n})$	$O(\mathrm{dim})$
Optimal correlated noise	$\frac{\log n}{\pi} + c$	$O(t \cdot \mathrm{dim})$
b-***Banded***	$O\Big((\sqrt{n/b} - 1)\log b\Big)$	$O(b \cdot \mathrm{dim})$
BLT of degree *d*	***??????***	$O(d \cdot \mathrm{dim})$

Our approach: **B**uffered **L**inear **T**oeplitz (BLT) Mechanism

	Max Error	***Noise generation time*** (in iteration *t*)
Independent noise	$\Theta(\sqrt{n})$	$O(\mathrm{dim})$
Optimal correlated noise	$\frac{\log n}{\pi} + c$	$O(t \cdot \mathrm{dim})$
b-***Banded***	$O\Big((\sqrt{n/b} - 1)\log b\Big)$	$O(b \cdot \mathrm{dim})$
BLT of degree *d*	***??????***	$O(d \cdot \mathrm{dim})$

Approximation Theory!!

From sequences to functions

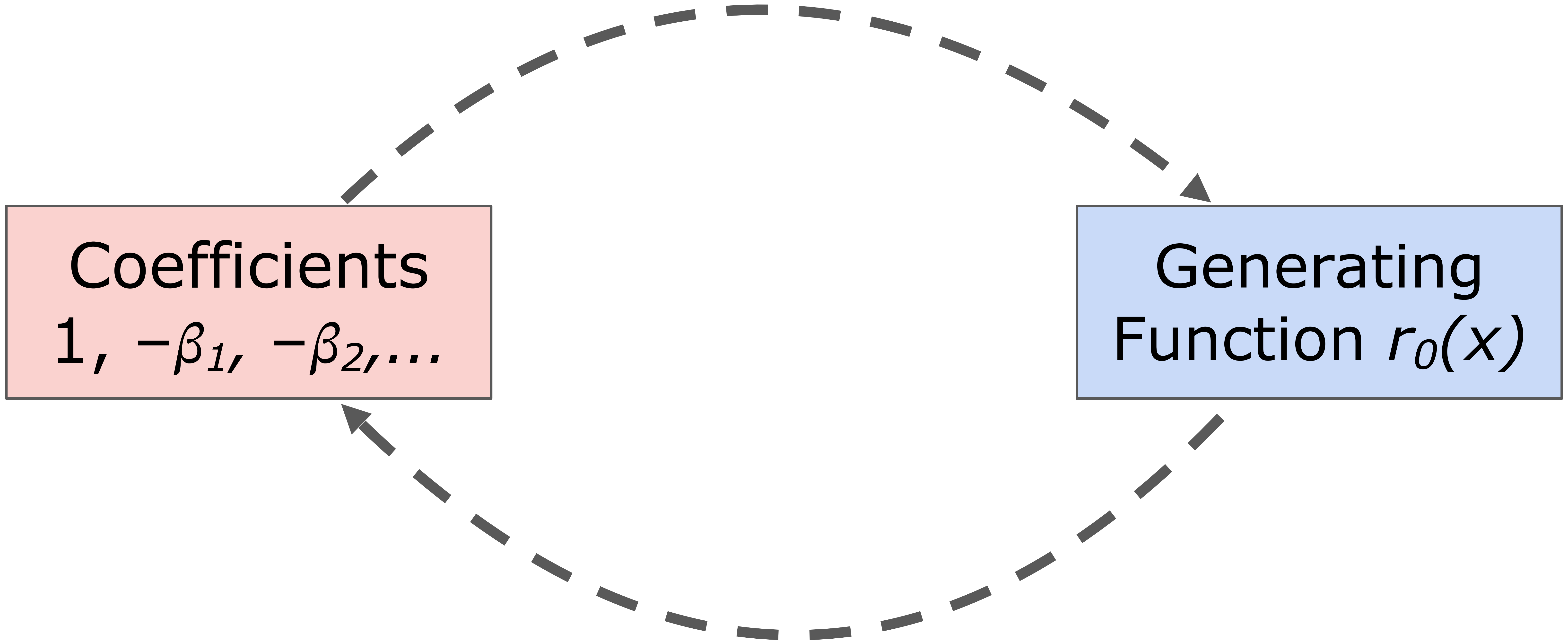
$$r(x) = 1 - \beta_1 x - \beta_2 x^2 - \cdots$$

Coefficients
1, $-\beta_1$, $-\beta_2$,...

Generating
Function $r_0(x)$

From sequences to functions

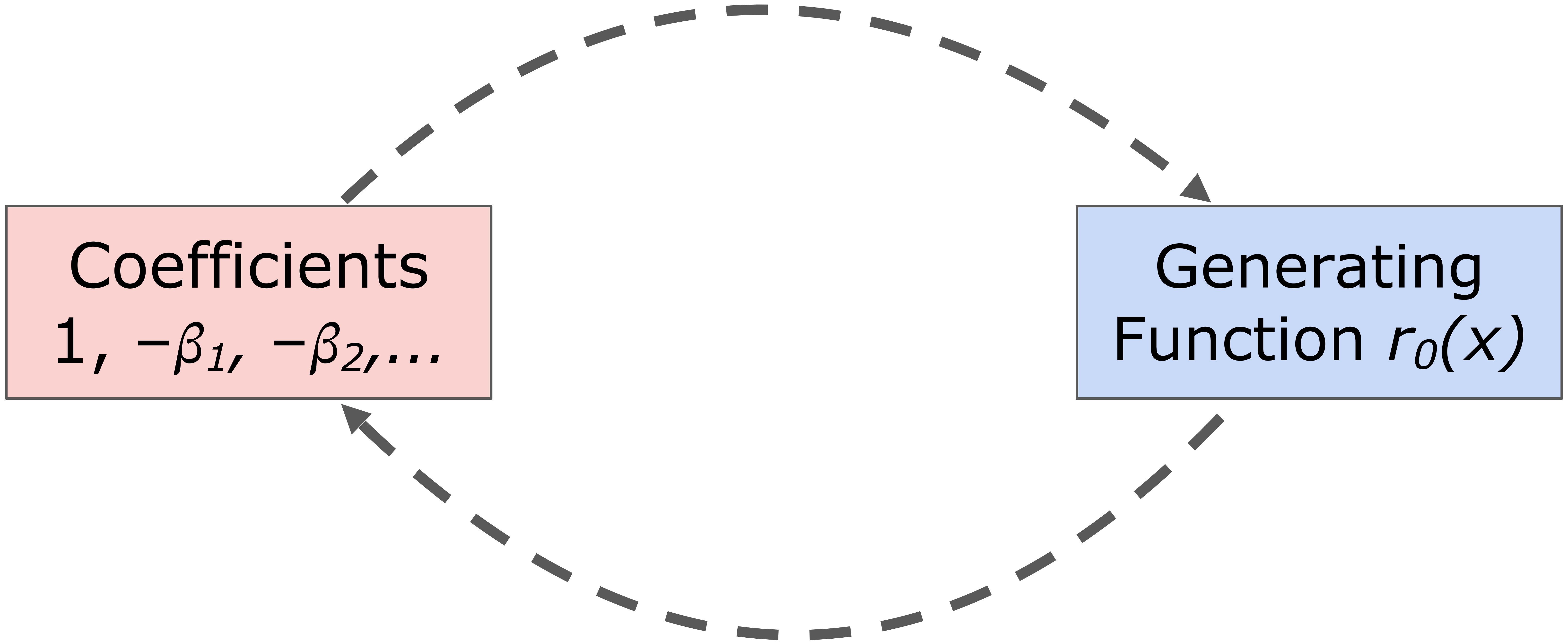
$$r(x) = 1 - \beta_1 x - \beta_2 x^2 - \cdots$$



Taylor expansion around $x = 0$

From sequences to functions

$$r(x) = 1 - \beta_1 x - \beta_2 x^2 - \cdots$$

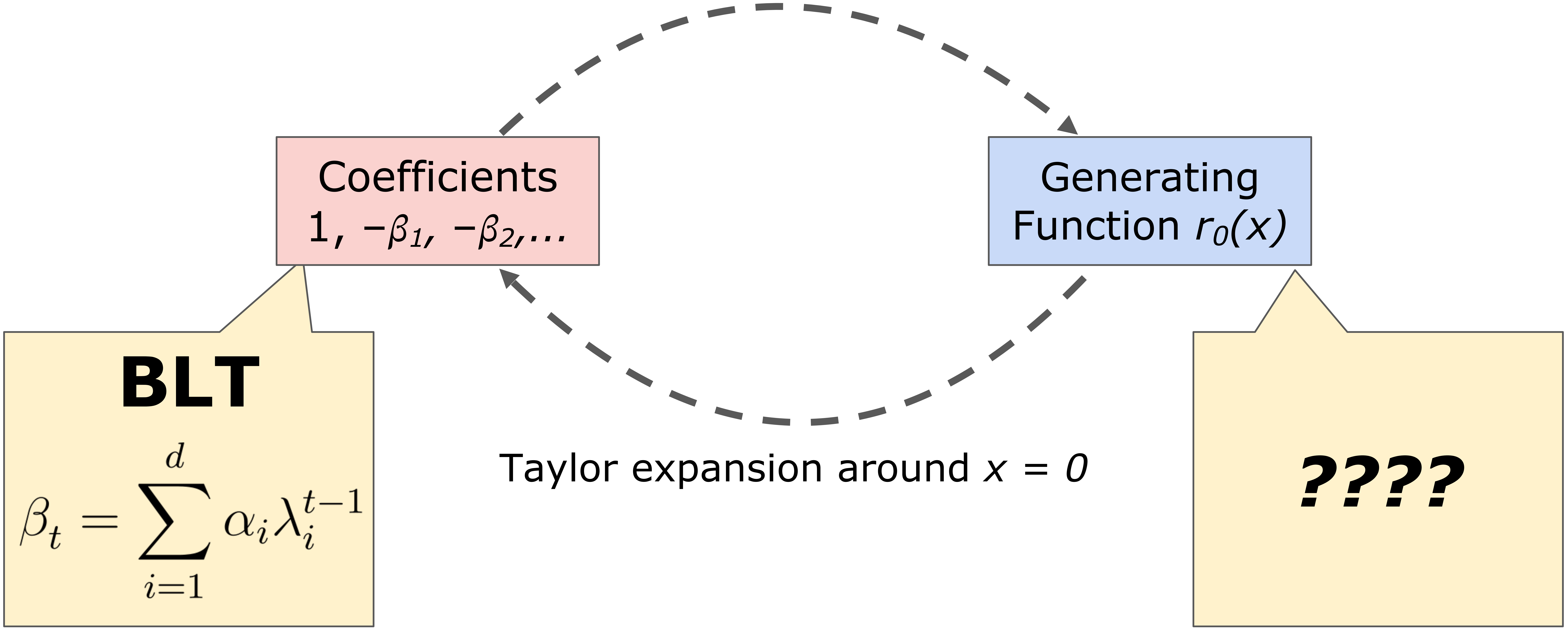


Taylor expansion around $x = 0$

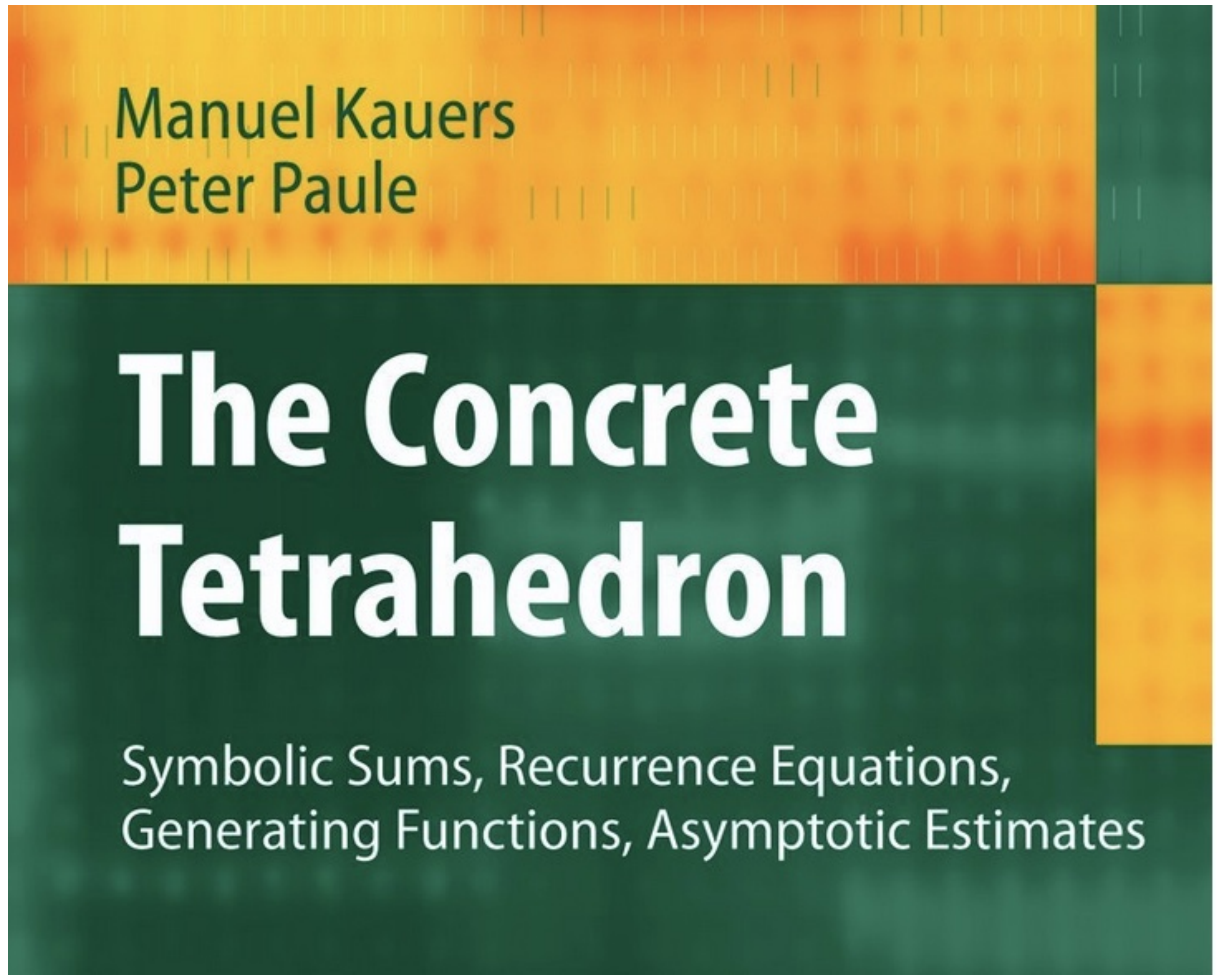
Coefficients $\beta_t = \Theta(t^{-3/2})$ $\Leftrightarrow$ generating function $r_0(x) = (1 - x)^{1/2}$

From sequences to functions

$$r(x) = 1 - \beta_1 x - \beta_2 x^2 - \cdots$$



BLT generating functions



Theorem (Informal):

The following properties are equivalent:

- β's are a (complex) BLT sequence: $$\beta_t = \sum_{i=1}^{d} \alpha_i \lambda_i^{t-1}$$
- Its generating function $r(x)$ is a **rational function** of degree d
- β's satisfy a linear recurrence $$\beta_t = \sum_{i=1}^{d} q_i \, \beta_{t-i}$$

From functions to efficient noise generation

Generating Function
$r_0(x) = (1 - x)^{1/2}$

How good is it?

Rational approximation
$r(x) \approx (1 - x)^{1/2}$

Theorem [Dvijotham, McMahan, ***P***., Steinke, Thakurta 2024]

The max error of a sequence (β_t) with generating function $r(x)$ is

$$\mathcal{E}(\beta) \leq \frac{\log n}{\pi} + O(n \cdot \mathrm{err}(r))$$

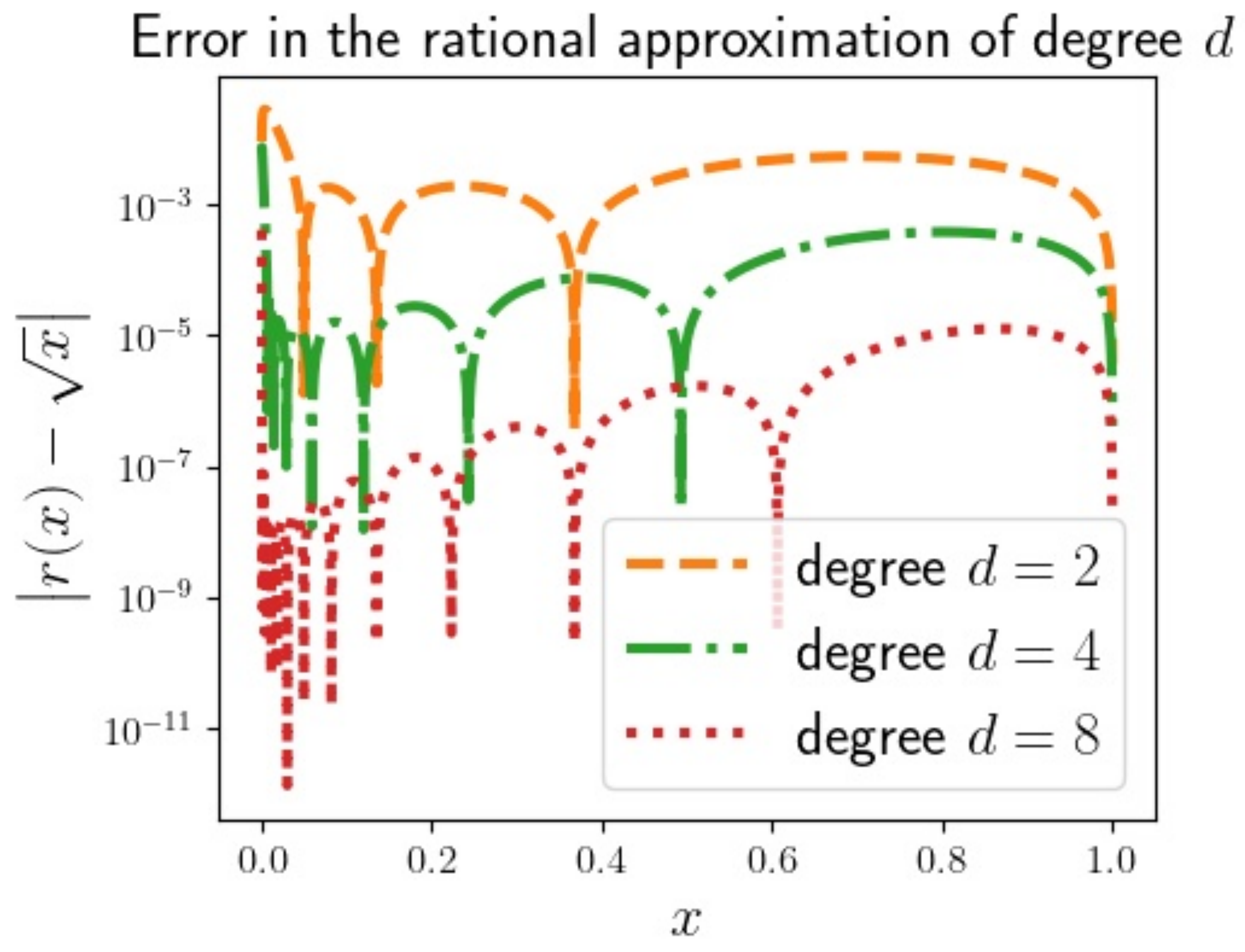
where err(r) quantifies the **approximation quality**

$$\mathrm{err}(r) = \max_{x \in \mathbb{C} \,:\, |x| = 1 - n^{-1}} \left| r(x) - \sqrt{1 - x} \right|$$

There exists a degree-*d* rational function that satisfies the tight approximation bound:

$$\sup_{x\in[0,1]} |r(x) - \sqrt{x}| \leq 3 \cdot \exp(-\sqrt{d}).$$

Newman. **Rational approximation to |*x*|**. Michigan Math. J. (1964)



where err(r) quantifies the **approximation quality**

$$\mathrm{err}(r) = \max_{x\in\mathbb{C}\,:\,|x|=1-n^{-1}} \left|r(x) - \sqrt{1-x}\right|$$

Our approach: **B**uffered **L**inear **T**oeplitz (BLT) Mechanism

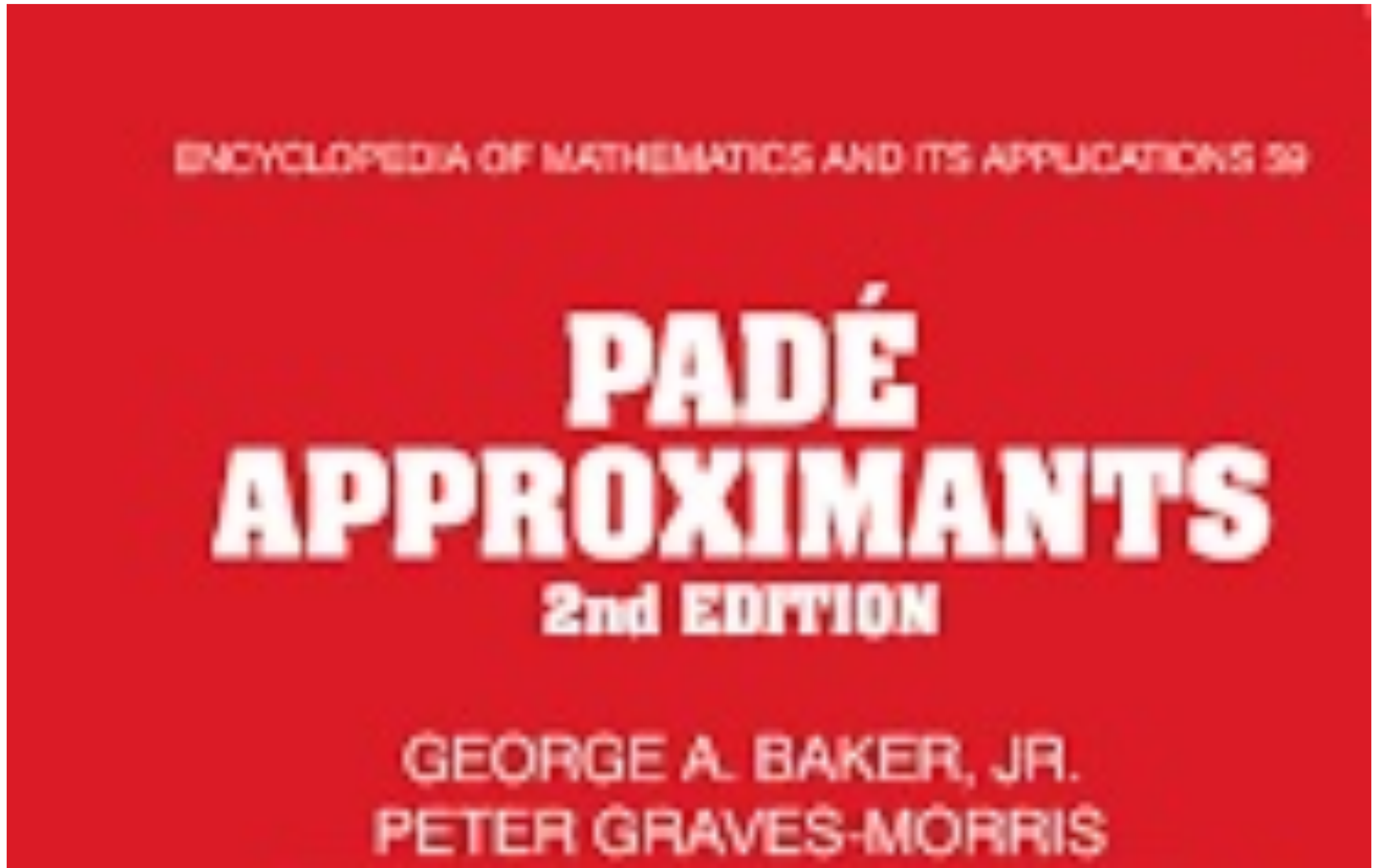
	Max Error	***Noise generation time*** (in iteration *t*)
Independent noise	$\Theta(\sqrt{n})$	$O(\mathrm{dim})$
Optimal correlated noise	$\frac{\log n}{\pi} + c$	$O(t \cdot \mathrm{dim})$
b-***Banded***	$O\Big((\sqrt{n/b} - 1)\log b\Big)$	$O(b \cdot \mathrm{dim})$
BLT of degree *d*	$\frac{\log n}{\pi} + O\big(n \cdot \exp(-\sqrt{d})\big)$	$O(d \cdot \mathrm{dim})$

Suffices to take $d=O(\log^2 n)$!

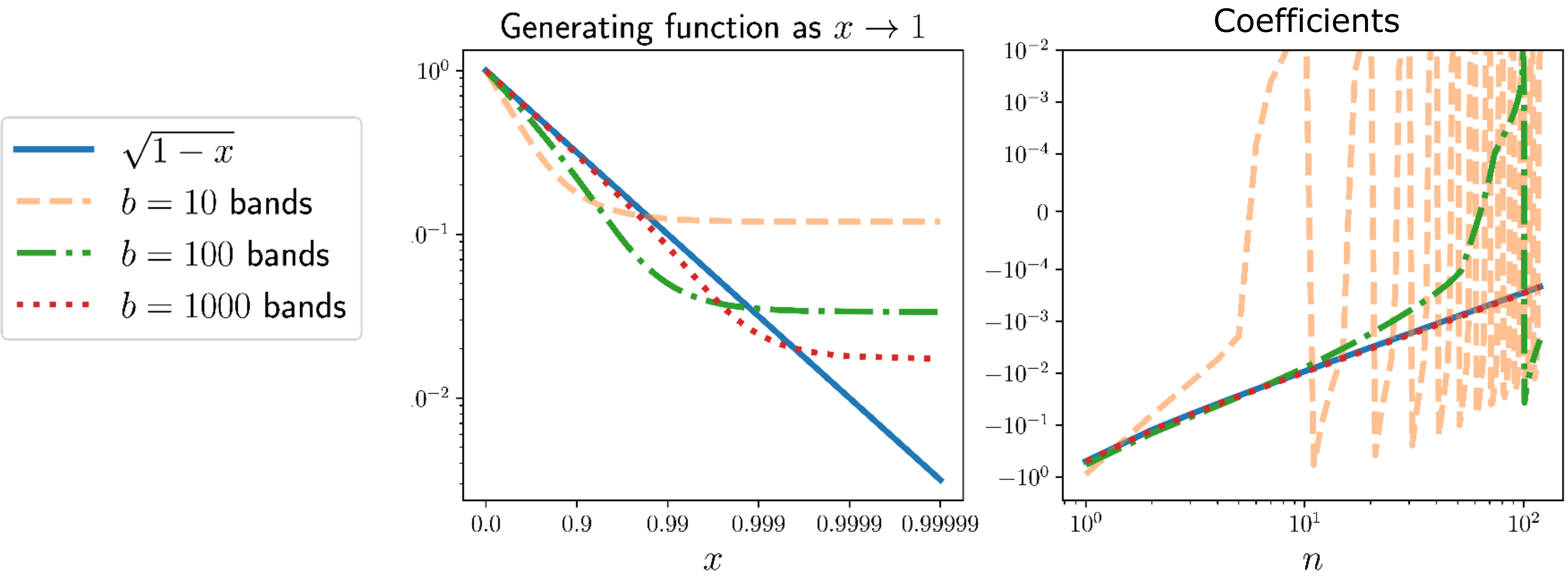
Key difference: approximation quality

Banded: Set $\beta_t = 0$ for $t > b$ $\Rightarrow$ polynomial approximation

BLT: $\beta'_t = \sum_{i=1}^{d} \alpha_i \lambda_i^{t-1}$ $\Rightarrow$ rational approximation

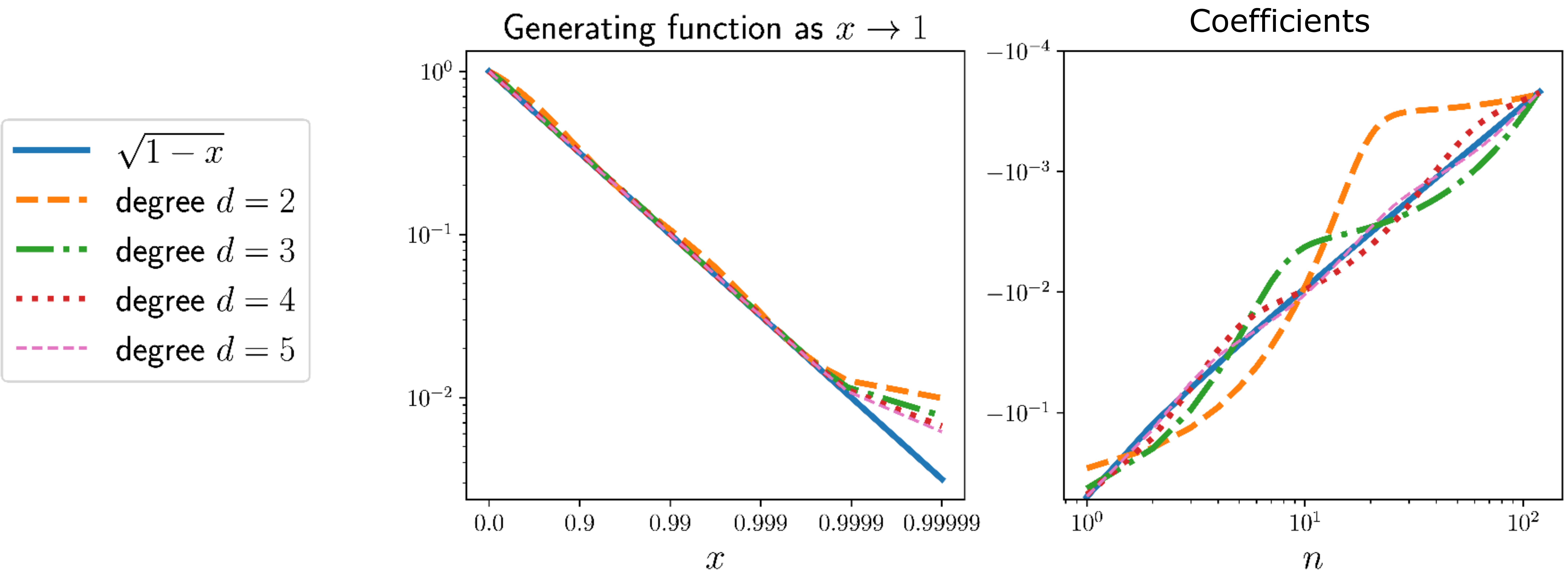


Approximation quality: banded mechanism



Note: here, we use a polynomial approximation to 1 / $(1 - x)^{1/2}$ rather than $(1 - x)^{1/2}$

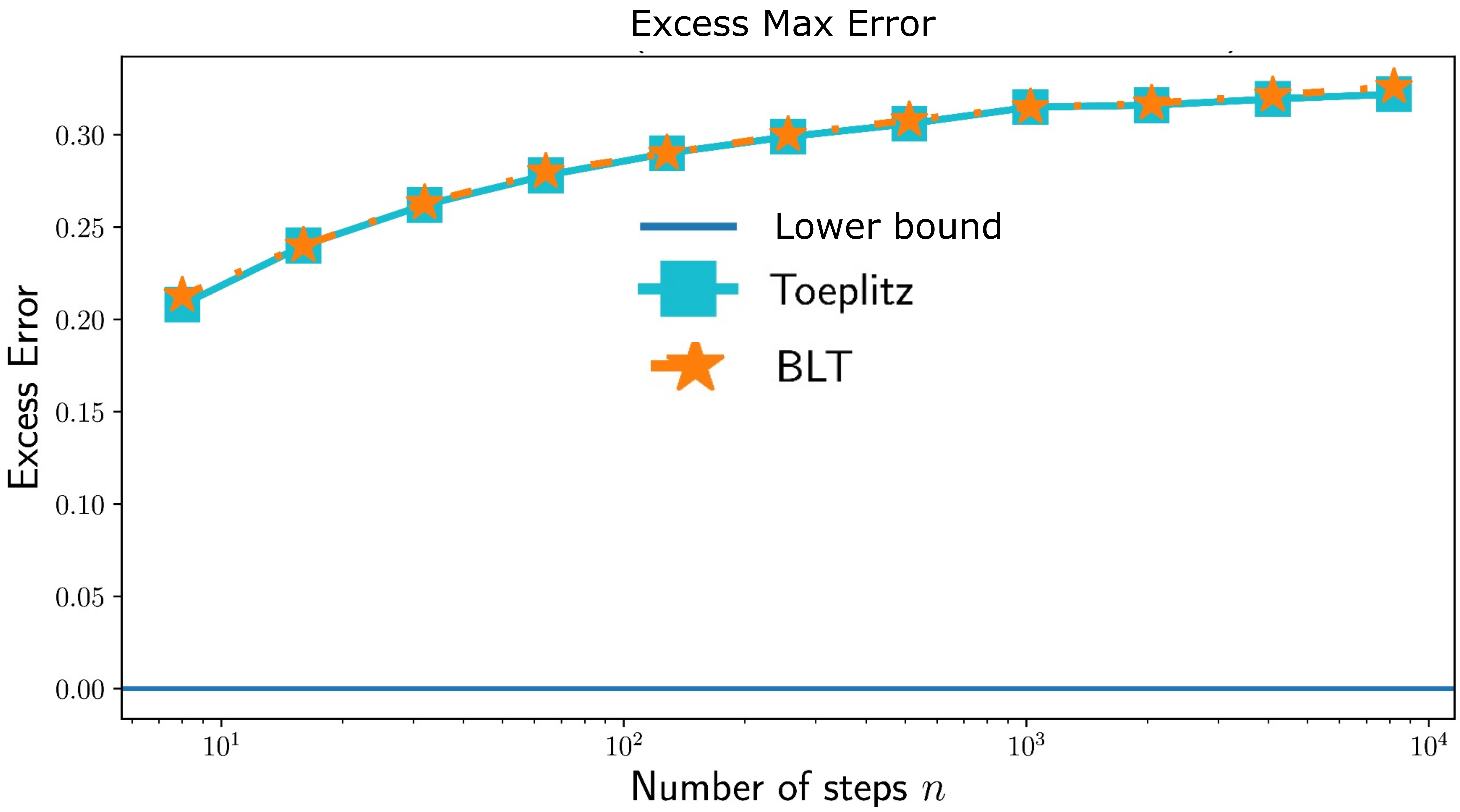
Approximation quality: BLT mechanism



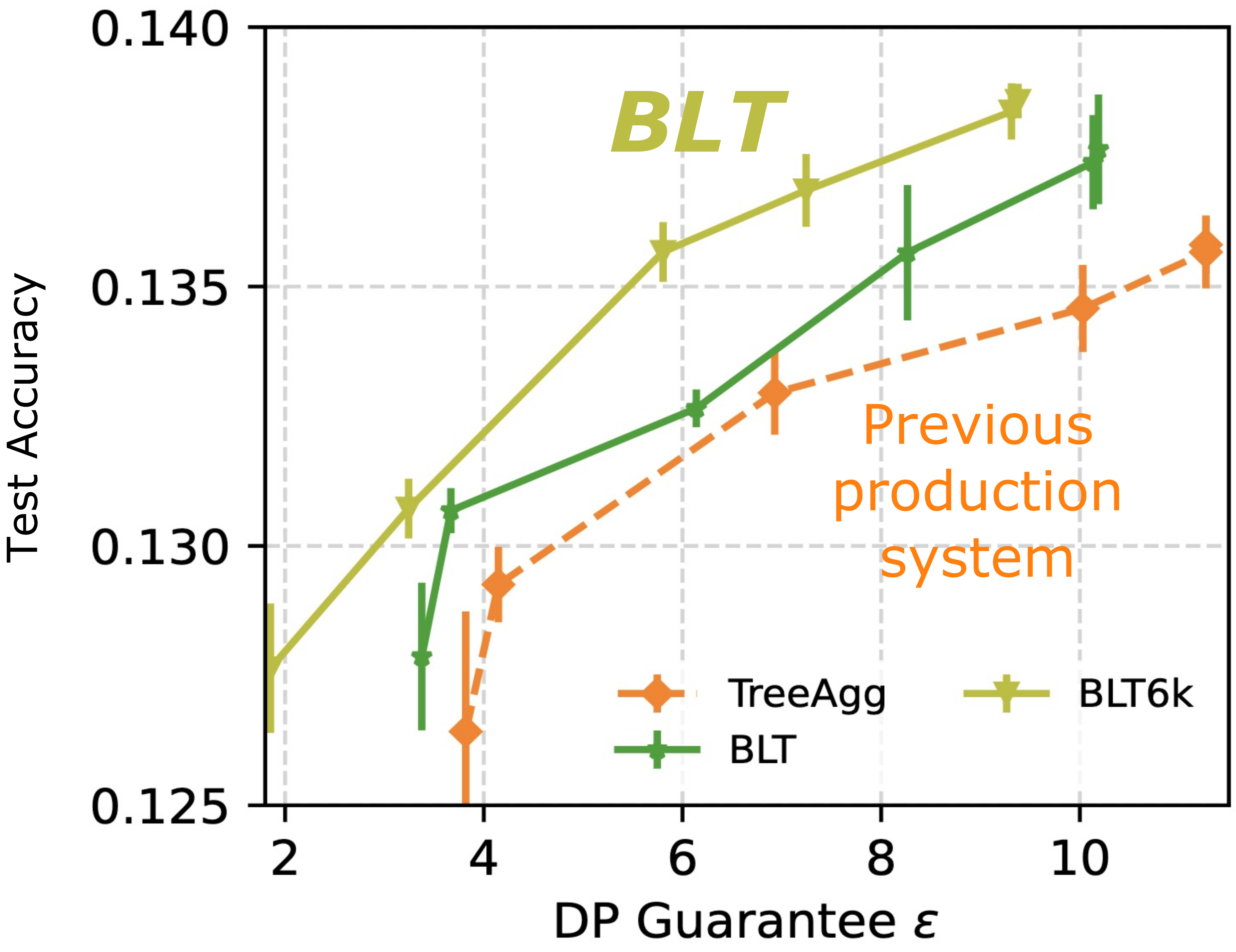
Note: BLT approximation of $1 / (1 - x)^{1/2}$ $\Leftrightarrow$ BLT approximation of $(1 - x)^{1/2}$

[McMahan and ***P***. (2025)]

Empirical Results



Practical Impact:
Google's production language model (Portuguese)



Plot: McMahan, Xu, Zhang (2024)

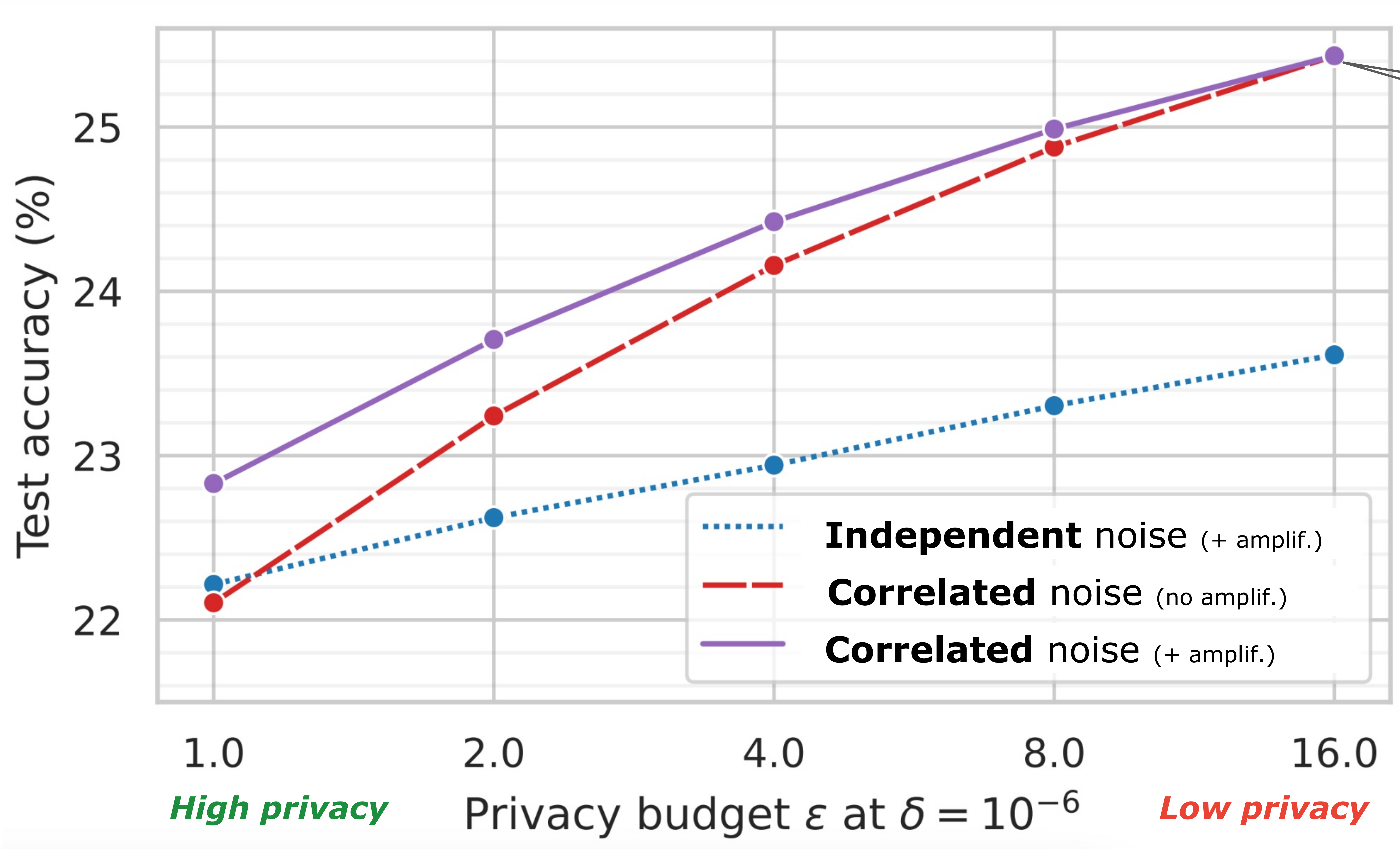
Aside: Theoretical evidence in favour of correlated noise for ***learning problems***

Choquette-Choo*, Dvijotham*, ***P.****, Ganesh, Steinke, Thakurta.
Correlated Noise Provably Beats Independent Noise for Differentially Private Learning.
ICLR (2024)

Prior work: (Empirically) correlated noise outperforms independent noise

Experiment: user-level DP + language modeling

Dataset: StackOverflow

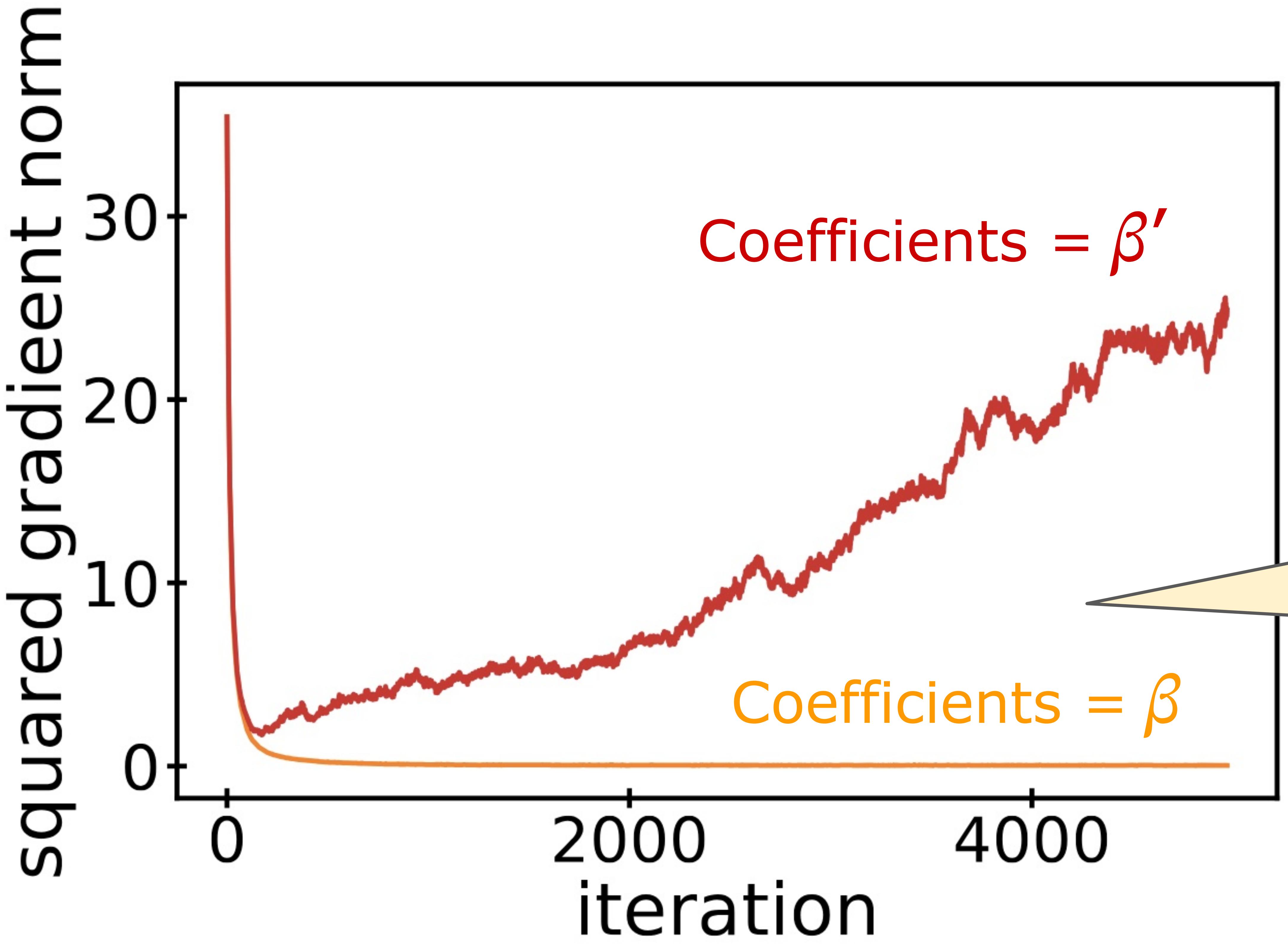


Correlated noise uniformly beats **independent noise**

Choquette-Choo, Ganesh, McKenna, McMahan, Rush, Thakurta, Xu.
(Amplified) Banded Matrix Factorization: A unified approach to private training. NeurIPS 2023

Is correlated noise provably better for learning problems?

The surrogate objective is not related to the learning objective



$$\mathcal{E}(\beta) = \mathcal{E}(\beta')$$

Same ***max error*** but different ***learning performance***

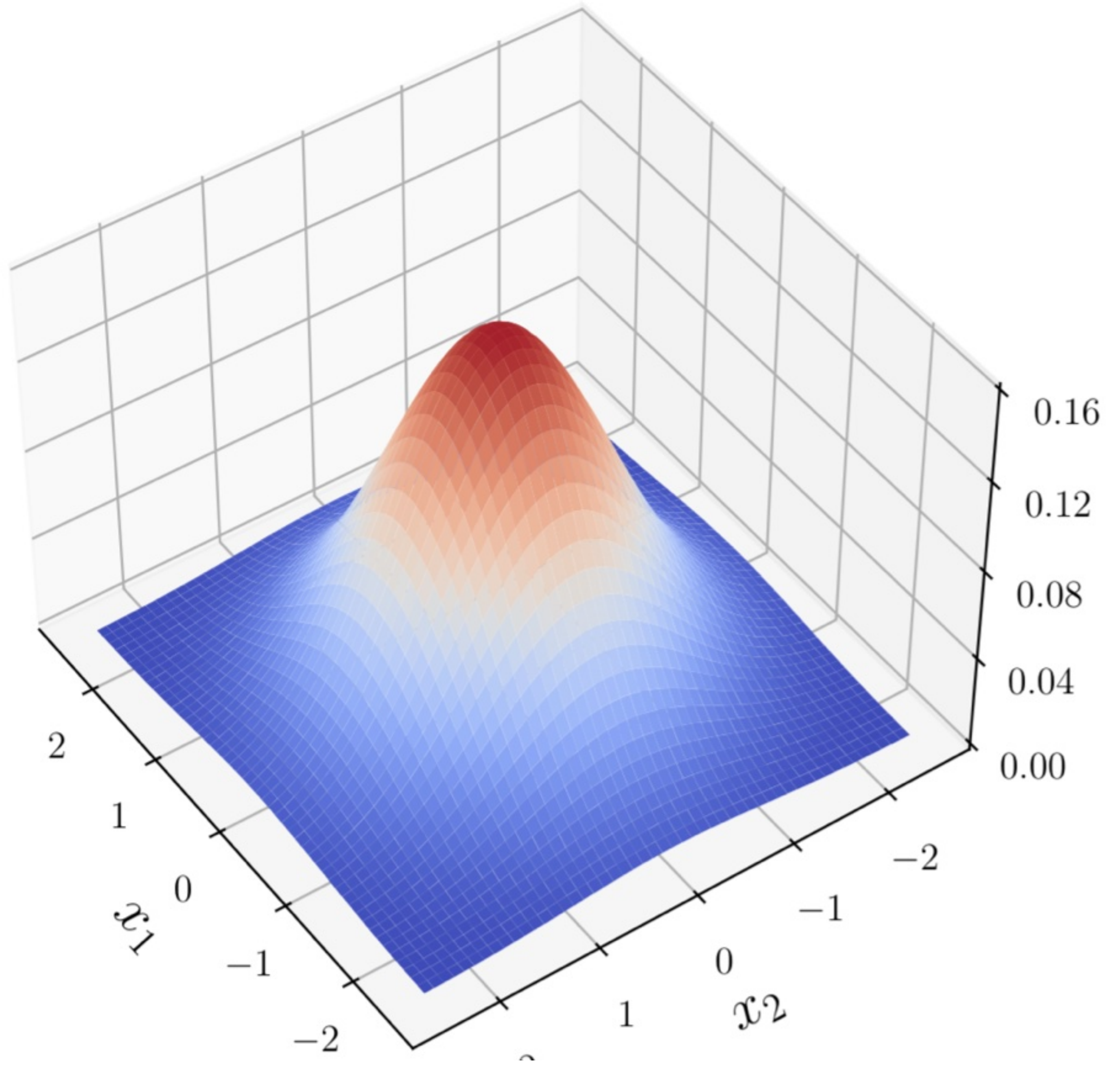
Koloskova, McKenna, Charles, Rush, McMahan. **Gradient Descent with Linearly Correlated Noise: Theory and Applications to Differential Privacy**. NeurIPS 2023

Our result: Correlated noise ***is*** provably better for learning problems

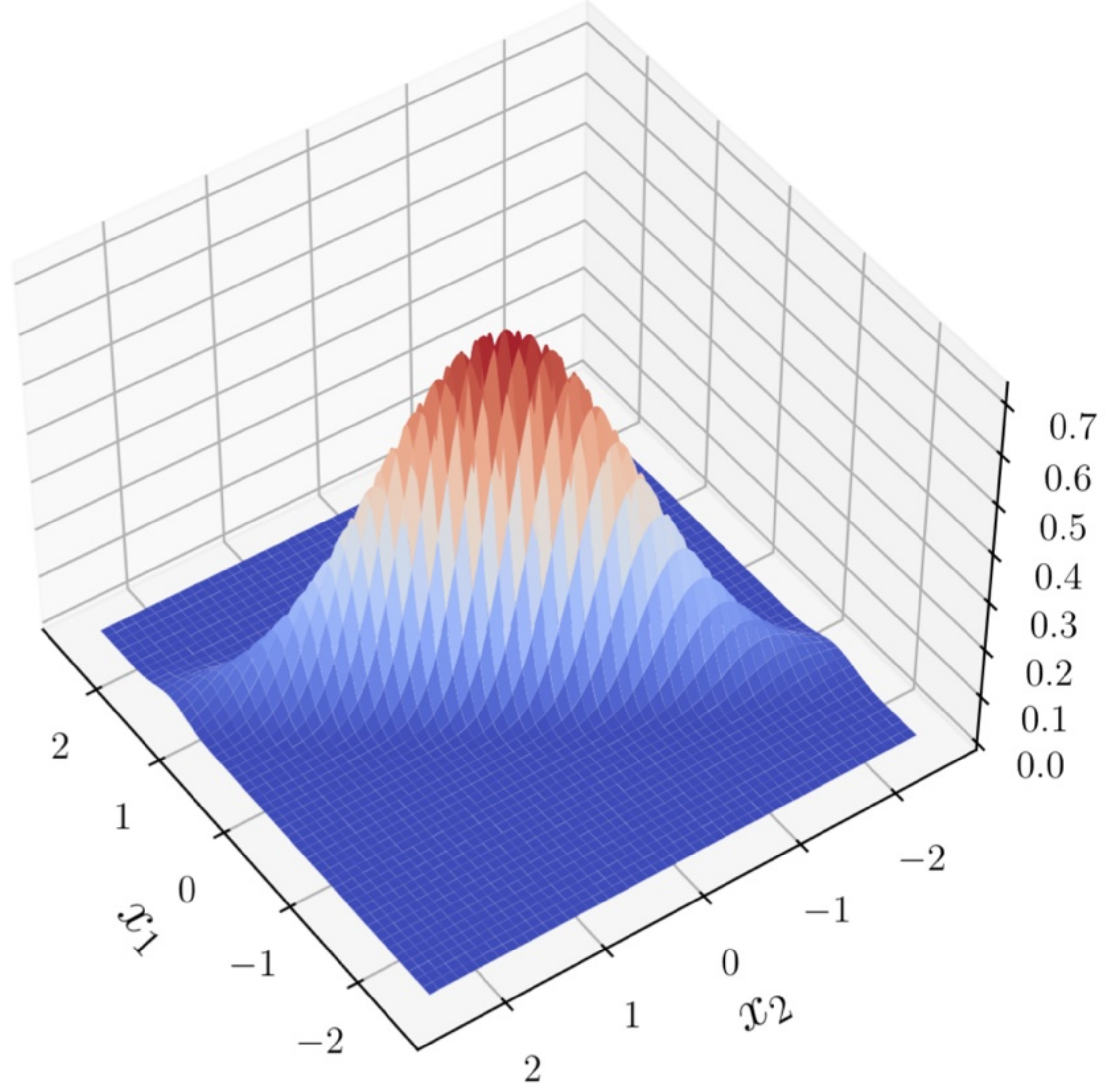
(Anti-) correlated noise provably beats independent noise

For linear regression, dimension d improves to problem-dependent ***effective dimension*** *d_{eff}*

Independent noise	$\Theta(d)$
Correlated noise	$\tilde{O}(d_{\text{eff}})$
Lower bound	$\Omega(d_{\text{eff}})$



High effective dimension



Low effective dimension

Aside 2: Fine-tuning LLMs with user-level DP

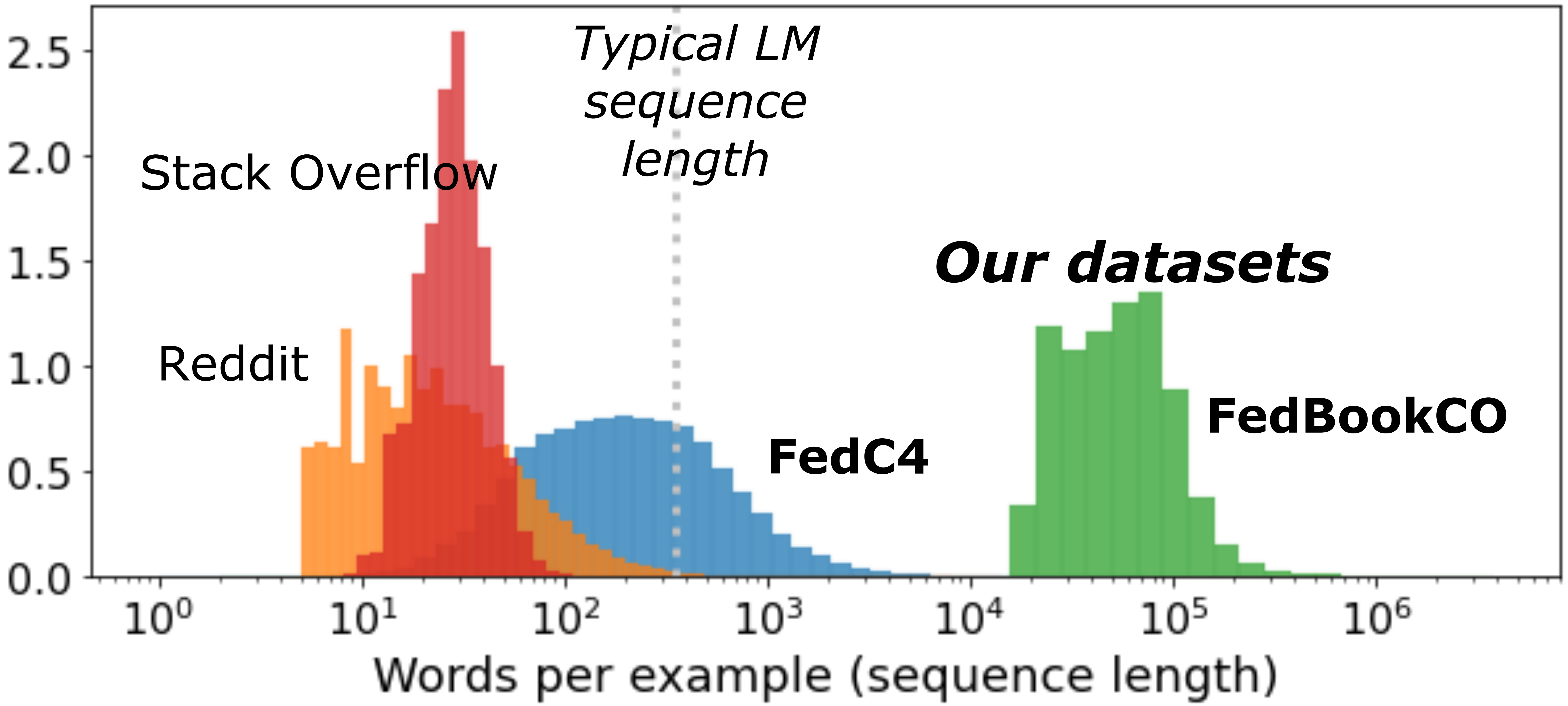
Scaling up user-level DP to LLMs (on a budget)

NeurIPS D&B 2023

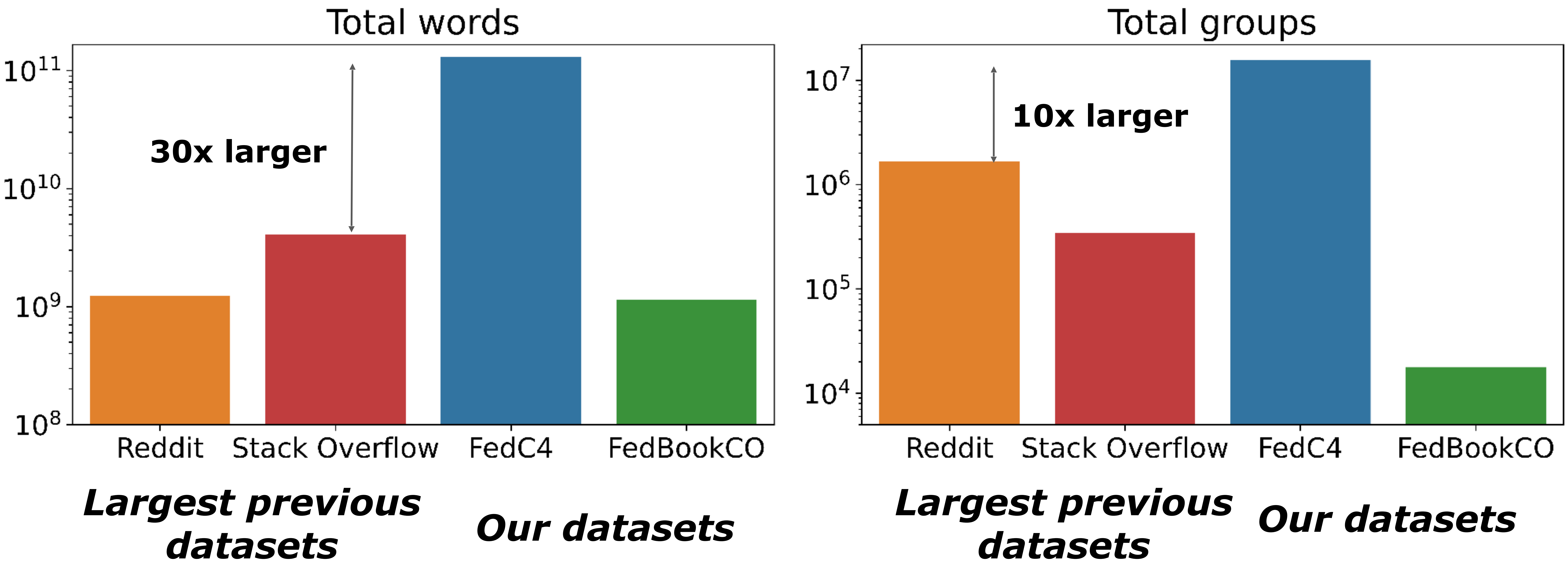
SaTML 2025

- *First user-level DP benchmarks for LLMs*
- *Training with **O(0.5B) params** and **O(100K) users***

Largest previous datasets



More words & groups than any previous benchmarks



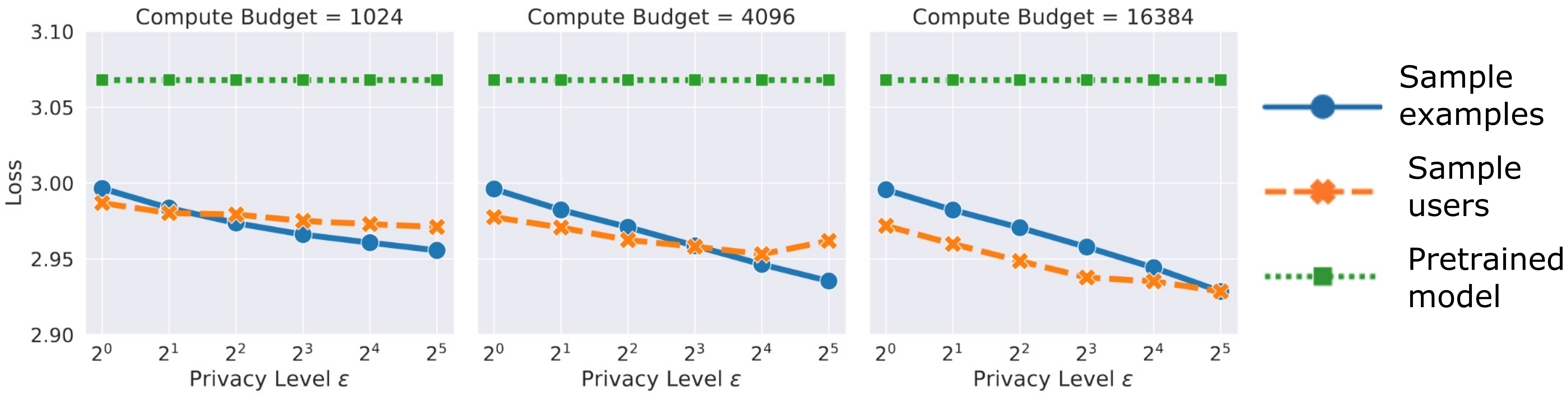
Scaling up user-level DP to LLMs (on a budget)

NeurIPS D&B 2023

SaTML 2025

Scaling user-level DP to LLMs (on a budget)
with independent noise:

- *First user-level DP benchmarks for LLMs*
- *Training with **O(0.5B) params** and **O(100K) users***



Coming soon: Monograph and tutorial on correlated noise mechanisms!

Contents

Open Problem: Continuous time limits

$$\theta_{t+1} = \theta_t - \eta\Big(g_t + \boxed{z_t - \sum_{\tau=1}^{t} \beta_\tau z_{t-\tau}}\Big)$$

Proceedings of Machine Learning Research vol 195:1–44, 2023 36th Annual Conference on Learning Theory

Universality of Langevin Diffusion for Private Optimization, with Applications to Sampling from Rashomon Sets

Arun Ganesh ARUNGANESH@GOOGLE.COM
Google Research

Abhradeep Thakurta ATHAKURTA@GOOGLE.COM
Google DeepMind

Jalaj Upadhyay JALAJ.UPADHYAY@RUTGERS.EDU
Rutgers University

Precise analysis
(better rates)

Algorithm design

Open Problem: Adaptive Gradient Algorithms

SGD update (without noise)

$$\theta_t - \theta_0 = -\sum_{\tau=0}^{t-1} g_\tau$$

Adam update (without noise)

$$v_t = (1-\beta_1)v_{t-1} + \beta_1 g_t$$

$$s_t = (1-\beta_2)s_{t-1} + \beta_2 g_t^2$$

$$\theta_{t+1} = \theta_t - \eta \frac{v_t}{\sqrt{s_t} + \delta}$$

Non-linear functions of the injected noise

Part 2: How audit user-level DP?

Unleashing the power of randomness in auditing DP

NeurIPS 2023

Krishna Pillutla

Galen Andrew

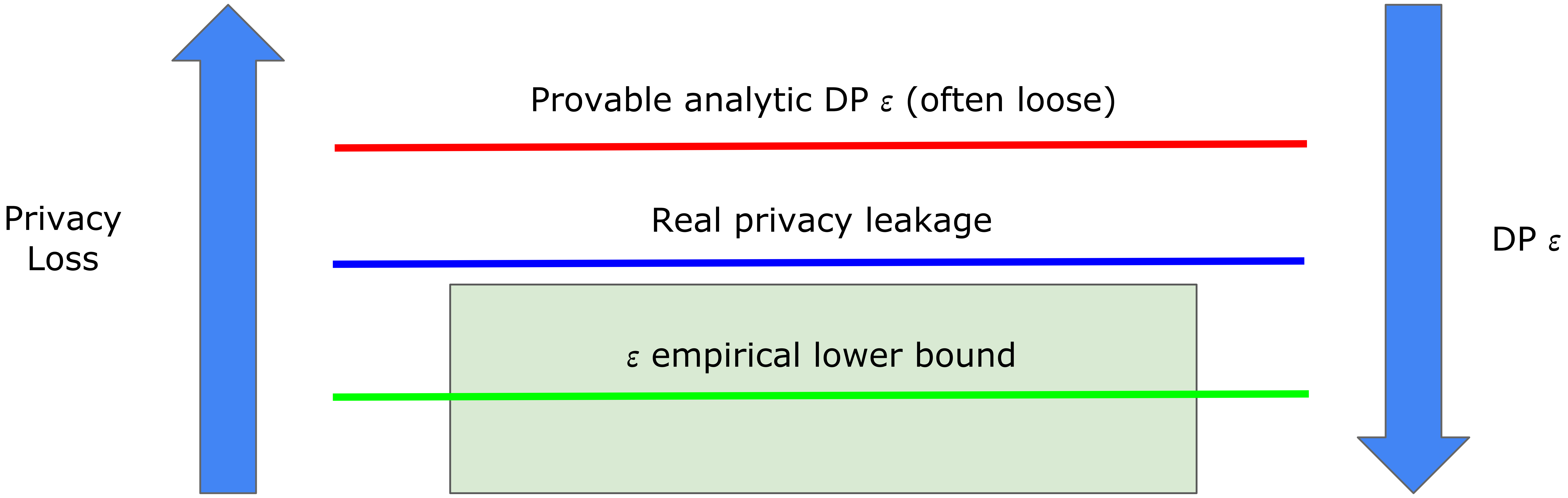
Peter Kairouz

Brendan McMahan

Alina Oprea

Sewoong Oh

Empirical privacy auditing



Our focus

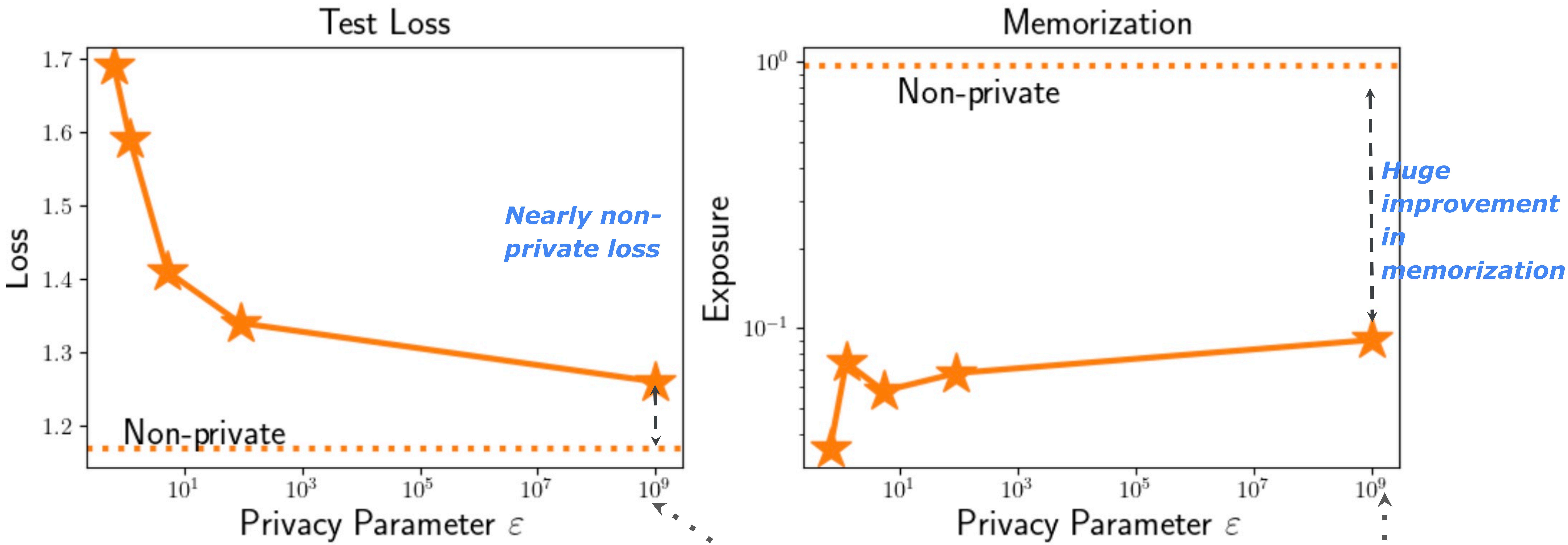
Why empirical privacy auditing?

To verify that we actually provide the guarantee we claim
(no bugs in proofs/implementation)

```diff
2 mnist_experiment.py
@@ -71,7 +71,7 @@ def forward(self, x):
                                                     rho_i,
                                                     epochs,
                                                     inp_clip,
-                                                    grad_clip
+                                                    grad_clip/BATCH_SIZE
                                                 )
                         tl, correct, set_len = uc.test(model, test_loader)
                         print(f'MNIST_{BATCH_SIZE}_{epochs}_{grad_clip}_{inp_clip}_{rho_i}', correct/set_len)

2 upstream_clipping.py
@@ -110,7 +110,7 @@ def run_experiment(model, train_loader, rho_i, epochs, input_bound, grad_bound):

     model.train()
     # sensitivity for everything with weights is just:
-    sensitivity = input_bound * grad_bound / train_loader.batch_size
+    sensitivity = input_bound * grad_bound
     sigma = np.sqrt(sensitivity**2 / (2*rho_i))
     print('sensitivity:', sensitivity)

```

Tramèr et al. **Debugging Differential Privacy: A Case Study for Privacy Auditing**. Preprint 2022

Gap between DP guarantees and empirical behavior: Memorization



High privacy *Low privacy* *High privacy* *Low privacy*

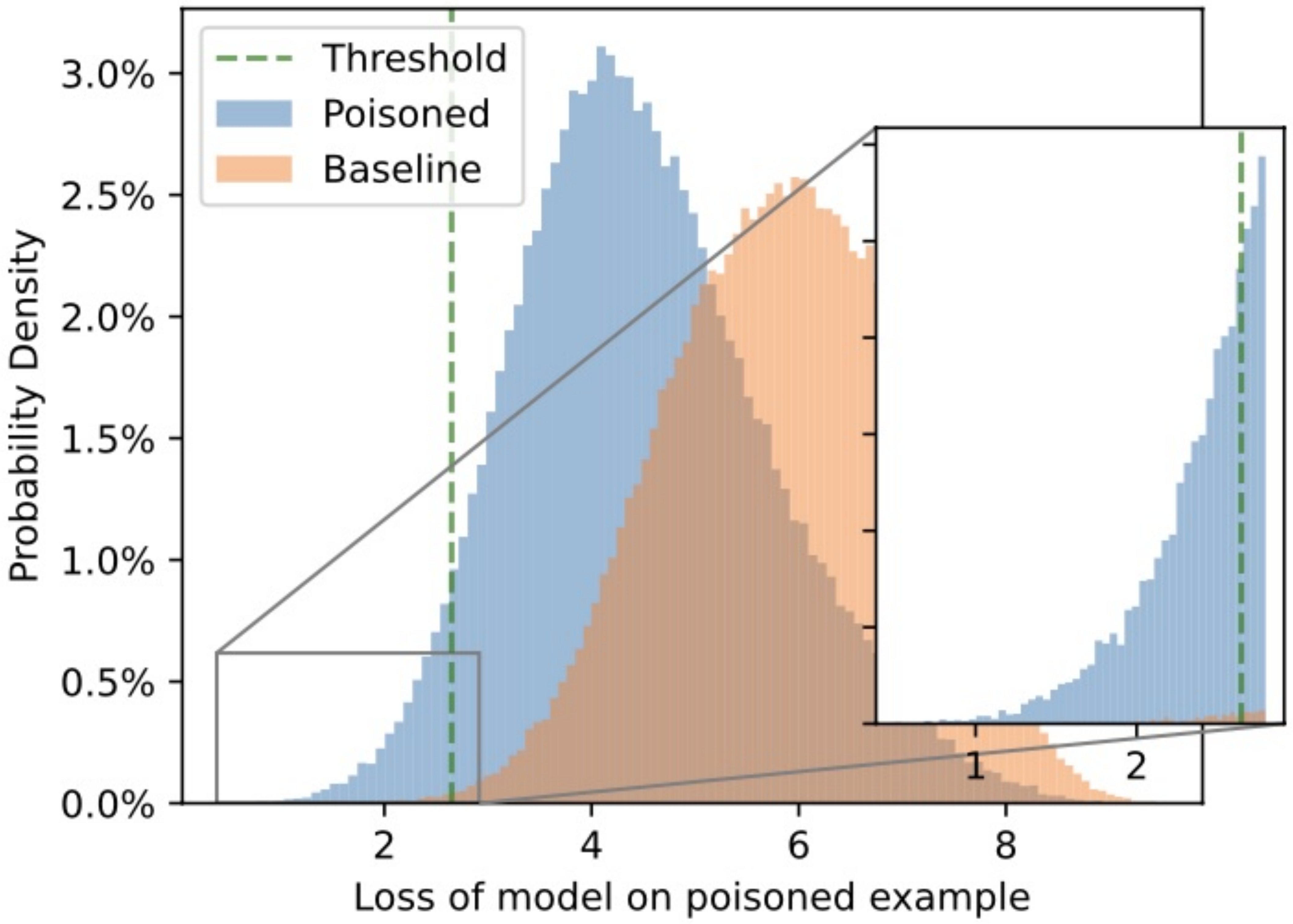
Carlini, Liu, Erlingsson, Kos, Song. **The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks.** USENIX Security 2019

Privacy guarantee is vacuous at this ε!

Empirical Privacy Auditing requires **many samples**

- Trained w/ $(0.21, 10^{-5})$-DP but empirically $\varepsilon > 2.79$ with confidence $1\text{-}10^{-8} \Rightarrow$ **bug in implementation**

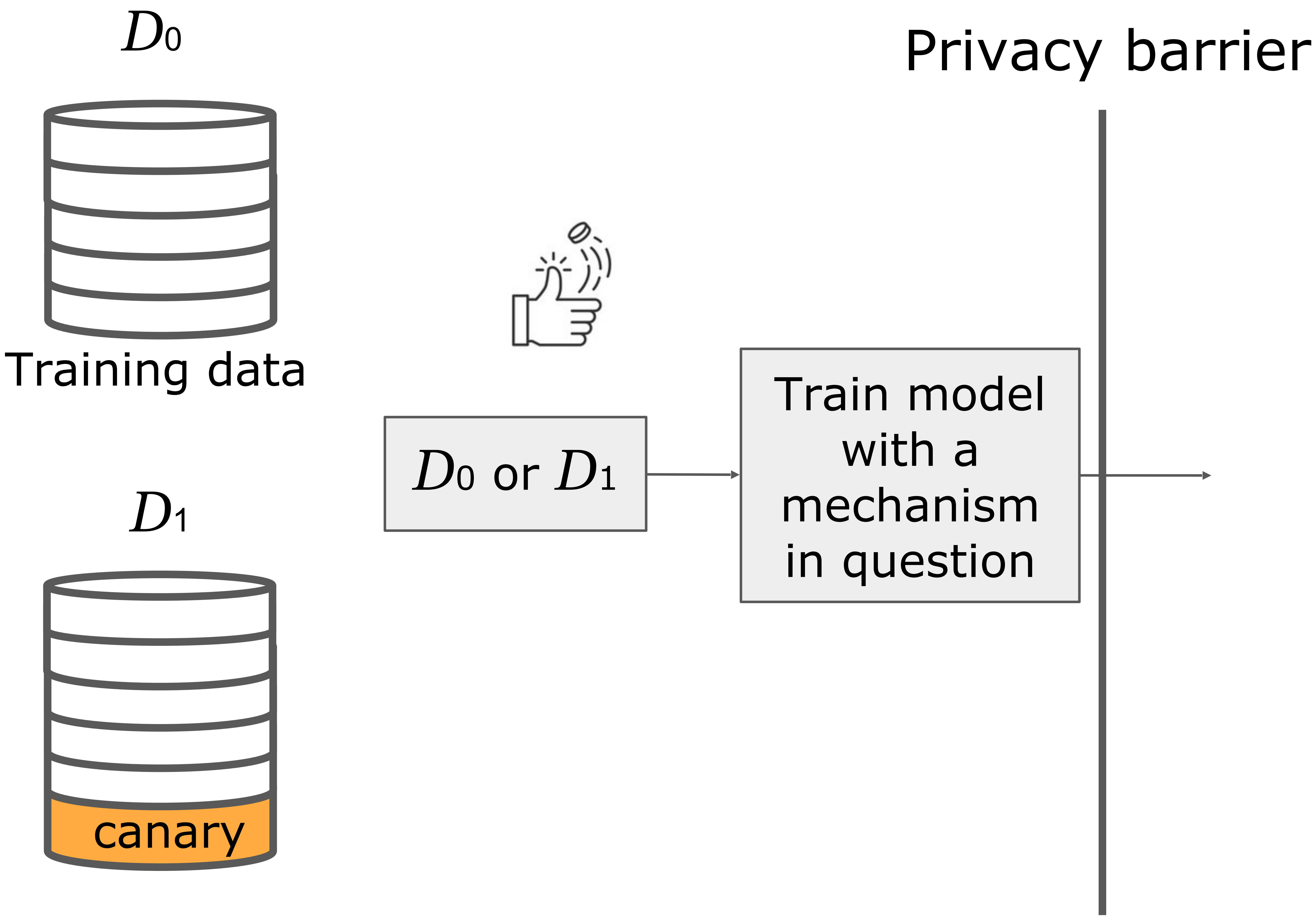
- This required training n**=200,000 models**



Tramèr et al. **Debugging Differential Privacy: A Case Study for Privacy Auditing**. Preprint 2022

Our goal: make empirical privacy auditing more ***sample-efficient***

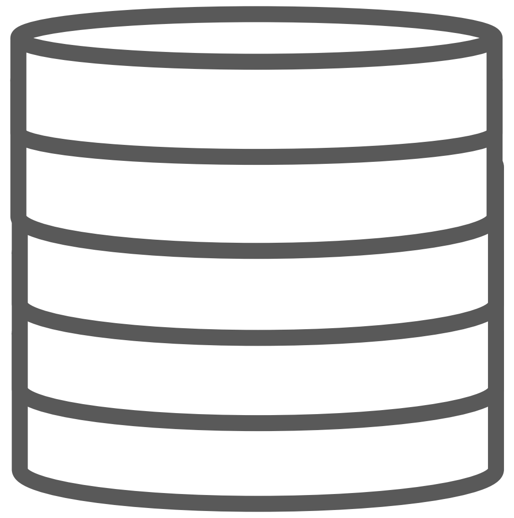
Standard approaches for auditing privacy: **binary hypothesis testing**



E.g., Nasr, Song, Thakurta, Papernot, Carlini. **Adversary Instantiation: Lower Bounds for Differentially Private Machine Learning**. IEEE S&P 2021
Jagielski, Ullman, Oprea. **Auditing differentially private machine learning: How private is private SGD?** NeurIPS 2020

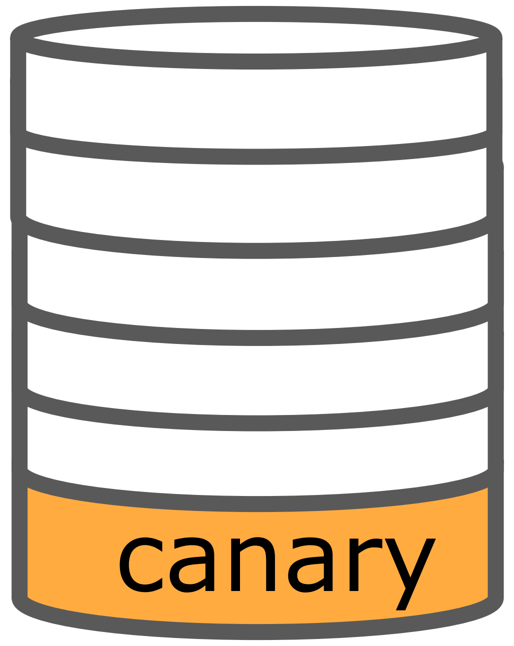
Standard approaches for auditing privacy: **binary hypothesis testing**

D_0



Training data

D_1



D_0 or D_1 → Train model with a mechanism in question →

Privacy barrier

$$\underbrace{\mathbb{P}(\mathcal{A}(D_1) \in R)}_{\text{True Positive Rate}} \leq e^{\varepsilon} \underbrace{\mathbb{P}(\mathcal{A}(D_0) \in R)}_{\text{False Positive Rate}} + \delta$$

D_0 (Null Hypothesis)

or

D_1 (Alternative Hypothesis) ? → Repeat many times and measure privacy leakage

E.g., Nasr, Song, Thakurta, Papernot, Carlini. **Adversary Instantiation: Lower Bounds for Differentially Private Machine Learning**. IEEE S&P 2021

Jagielski, Ullman, Oprea. **Auditing differentially private machine learning: How private is private SGD?** NeurIPS 2020

Bottleneck: Bernoulli confidence intervals

- Confidence intervals based on *n* trials

$$\mathrm{TPR} \approx \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(\text{Guess } i \text{ correct})}_{} + \sqrt{\frac{\text{Variance}}{n}}$$

Actual TPR/FPR

Empirical TPR/FPR

Actual TPR/FPR

$$\varepsilon \geq \log\left(\frac{\mathrm{TPR} - \delta}{\mathrm{FPR}}\right)$$
$$\geq \log\left(\frac{\hat{\mathrm{TPR}}_n - \boxed{\frac{c}{\sqrt{n}}} - \delta}{\hat{\mathrm{FPR}}_n + \boxed{\frac{c}{\sqrt{n}}}}\right)$$

Empirical TPR/FPR

Sample size *n* needs to be large for good estimates

Our approach: leverage randomness

- **Lifted DP**: Equivalent notion of DP with randomized datasets
- Multiple randomized hypothesis tests
- **Adaptive confidence intervals** capitalizing on low correlations

Multiple hypothesis tests for auditing Lifted DP

- **Leave-One Out** construction with **i.i.d. random** canaries

Multiple hypothesis tests for auditing Lifted DP

If the statistics are independent ⇒ better confidence intervals

Unfortunately, they are **dependent**
(but highly uncorrelated)

Average
test statistics

k **Random**
canaries c_1,c_2,c_3

k**-1 Random**
canaries

Is c_3 in D_1?
Is c'_3 in D_0?

Novel higher-order confidence interval

- 2nd-order confidence interval using empirical correlations between two tests

$$|\mathrm{TPR} - \widehat{\mathrm{TPR}}_{n,k}| \lesssim \sqrt{\frac{1}{n}\left(\mathrm{Correlation} + \frac{1}{k} + \sqrt{\frac{\text{4th moment}}{n}}\right)}$$

- Ideally, when **correlation=**O(*1/k*), the confidence interval improves as

$$|\mathrm{TPR} - \widehat{\mathrm{TPR}}_{n,k}| \lesssim \sqrt{\frac{1}{nk}} + \frac{1}{n^{3/4}}$$

Takeaway: **Reduces variance** from randomness in trials

Standard approach:

$$\varepsilon \geq \log\left(\frac{\widehat{\mathrm{TPR}}_n - \boxed{\frac{c}{\sqrt{n}}} - \delta}{\widehat{\mathrm{FPR}}_n + \boxed{\frac{c}{\sqrt{n}}}}\right)$$

c - Universal constant

c' - Data-dependent constant

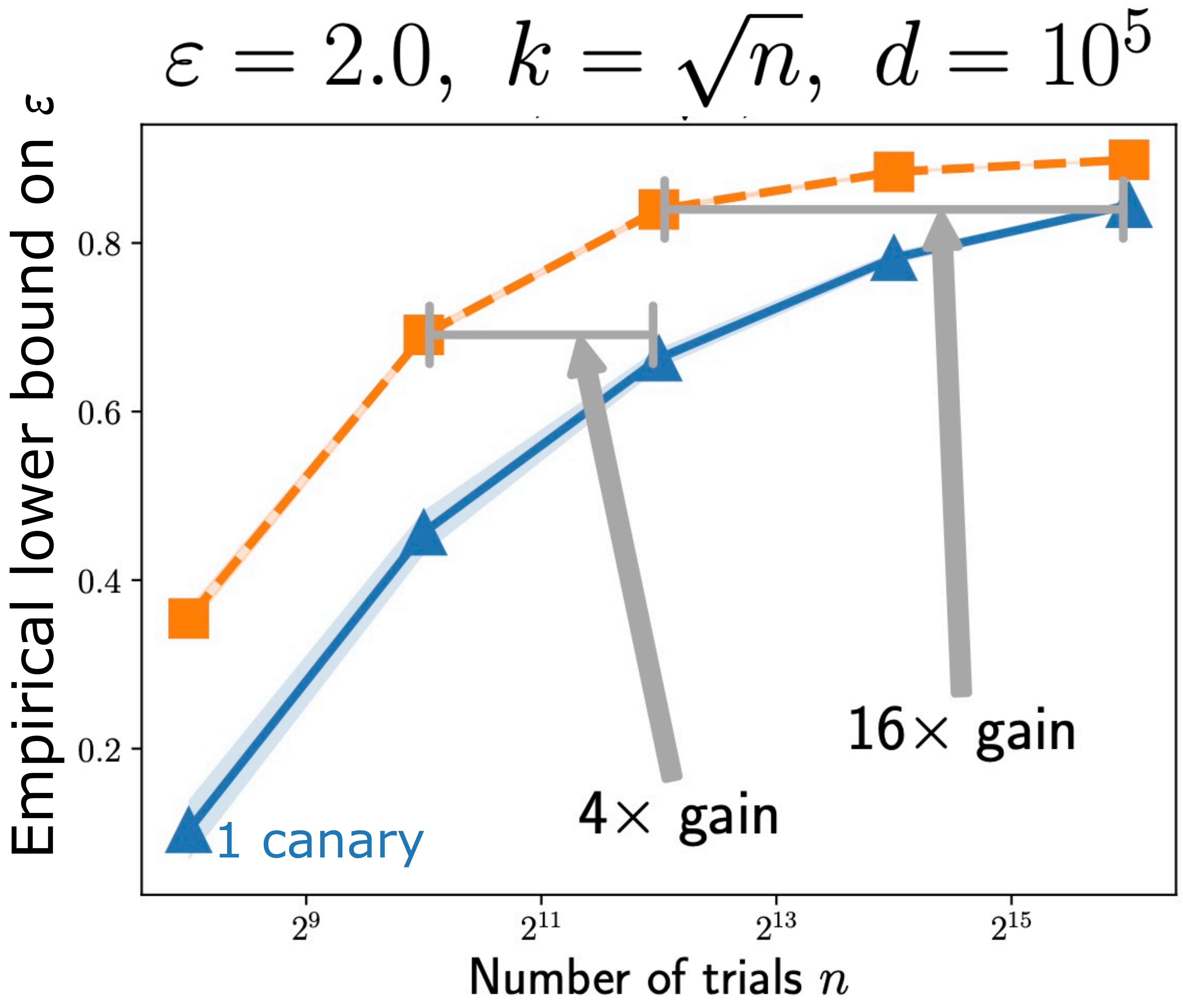
Lower variance =>
Tighter confidence intervals

Our approach:

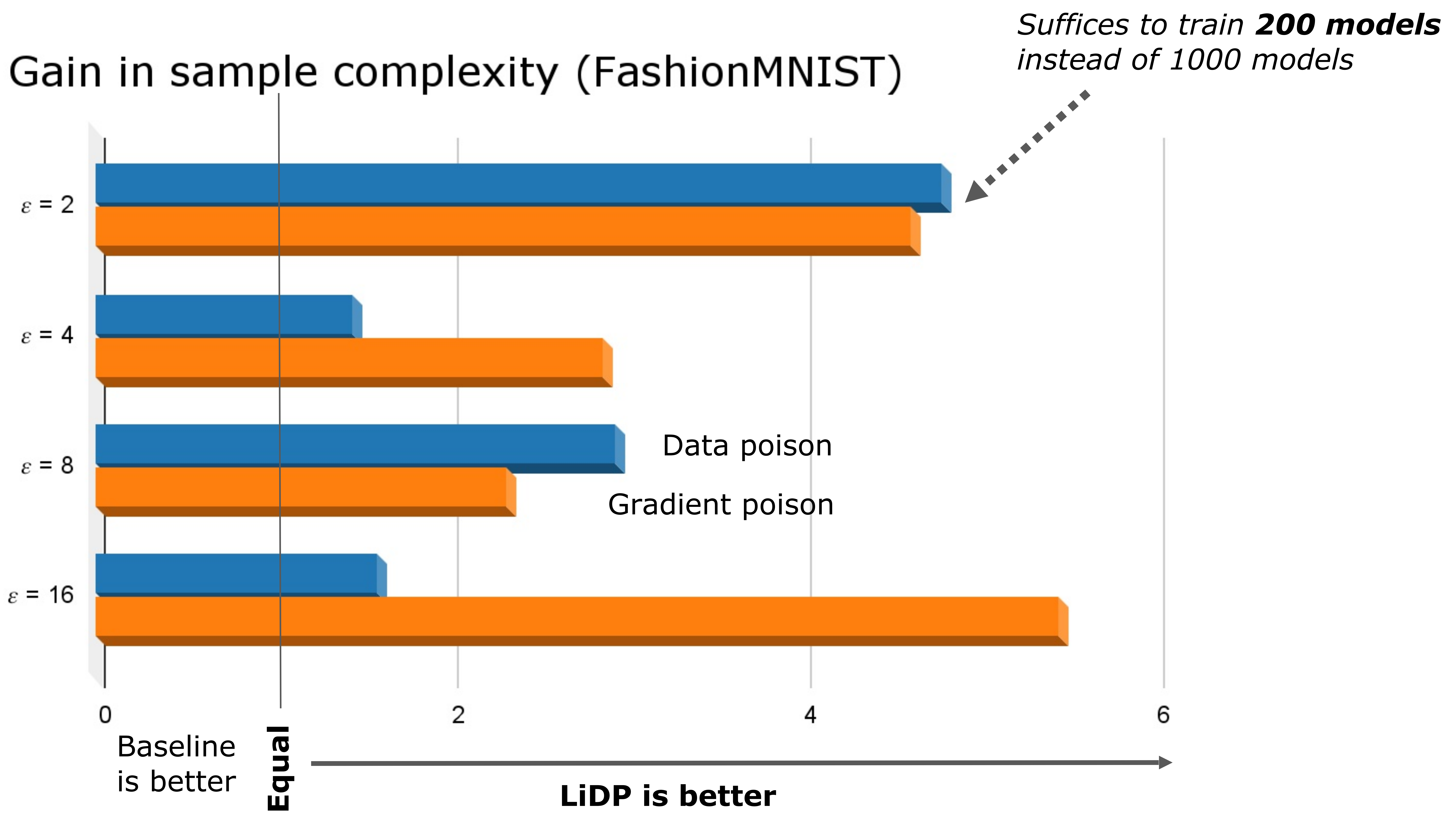
$$\varepsilon \geq \log\left(\frac{\widehat{\mathrm{TPR}}_{n,k} - \boxed{\frac{c}{\sqrt{nk}} - \frac{c'}{n^{3/4}}} - \delta}{\widehat{\mathrm{FPR}}_{n,k} + \boxed{\frac{c}{\sqrt{nk}} + \frac{c'}{n^{3/4}}}}\right)$$

Proof of concept with Gaussian mechanisms

- Sum query with sensitivity 1
- Gaussian mechanism
- k **canaries** uniformly random on the sphere
- **Test statistic** is inner product



Dwork, Smith, Steinke, Ullman, Vadhan. **Robust traceability from trace amounts**. FOCS 2015


Suffices to train **200 models** instead of 1000 models
Gain in sample complexity (FashionMNIST)
ε = 2
ε = 4
ε = 8
ε = 16
Data poison
Gradient poison
0
2
4
6
Baseline is better
Equal
LiDP is better

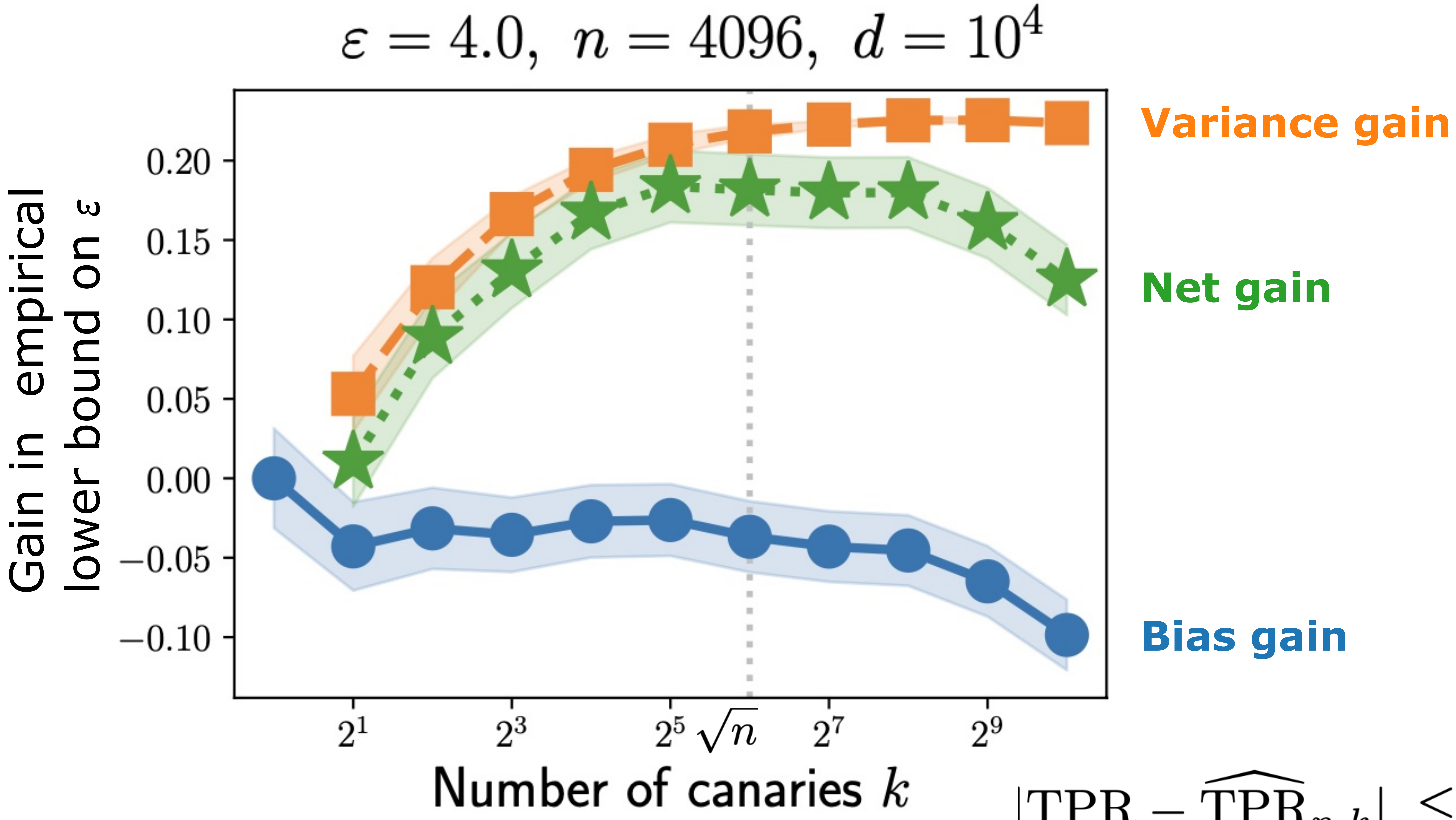
Privacy Auditing with One (1) Training Run

Thomas Steinke* Google DeepMind steinke@google.com

Milad Nasr* Google DeepMind srxzr@google.com

Matthew Jagielski* Google DeepMind jagielski@google.com

Bias-variance tradeoff in the number of canaries k



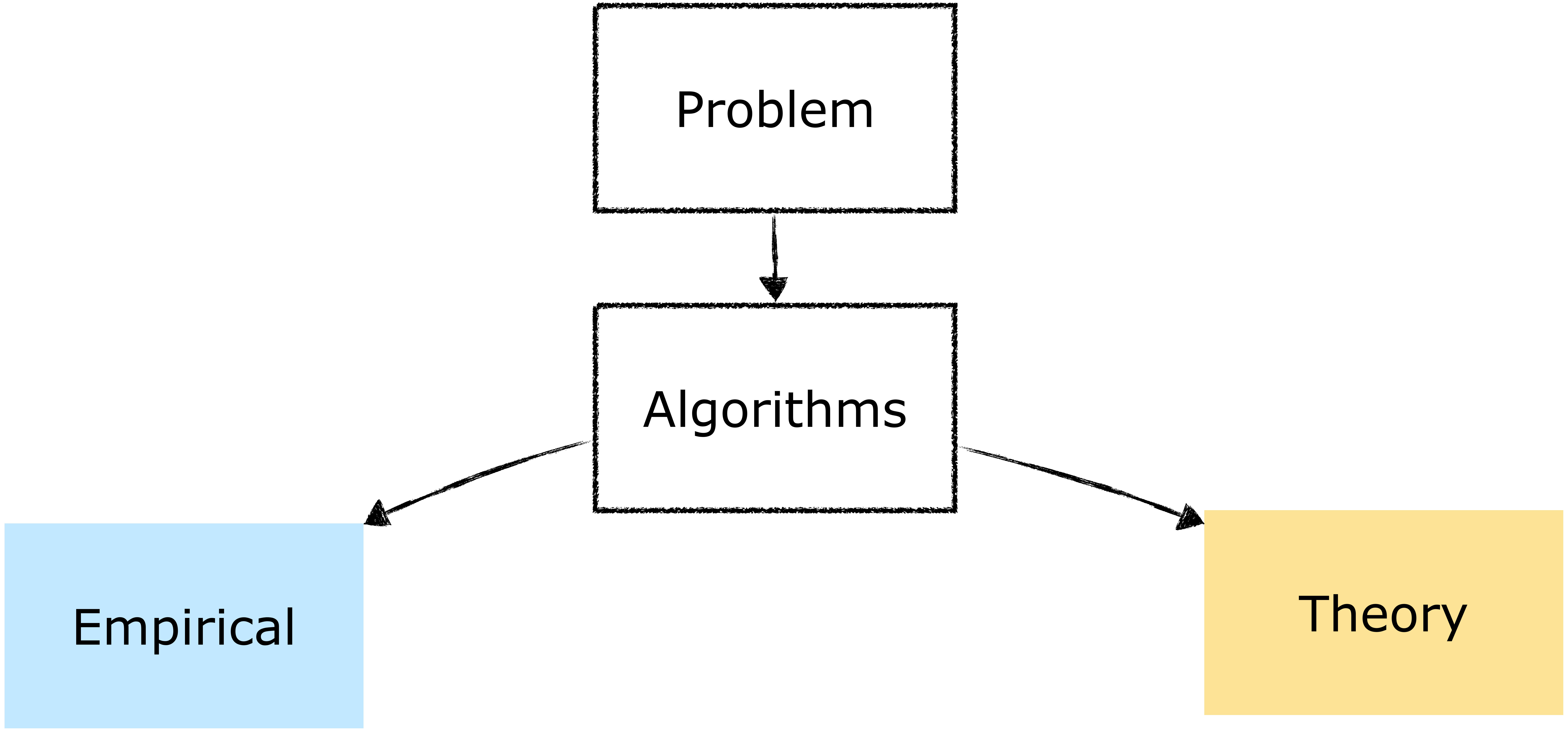
$$|\mathrm{TPR} - \widehat{\mathrm{TPR}}_{n,k}| \lesssim \sqrt{\frac{1}{nk} + \frac{1}{n^{3/4}}}$$

Summary

- **Auditing Lifted DP** (equivalent to usual DP) using multiple **i.i.d. random canaries** to improve sample dependence of the confidence intervals

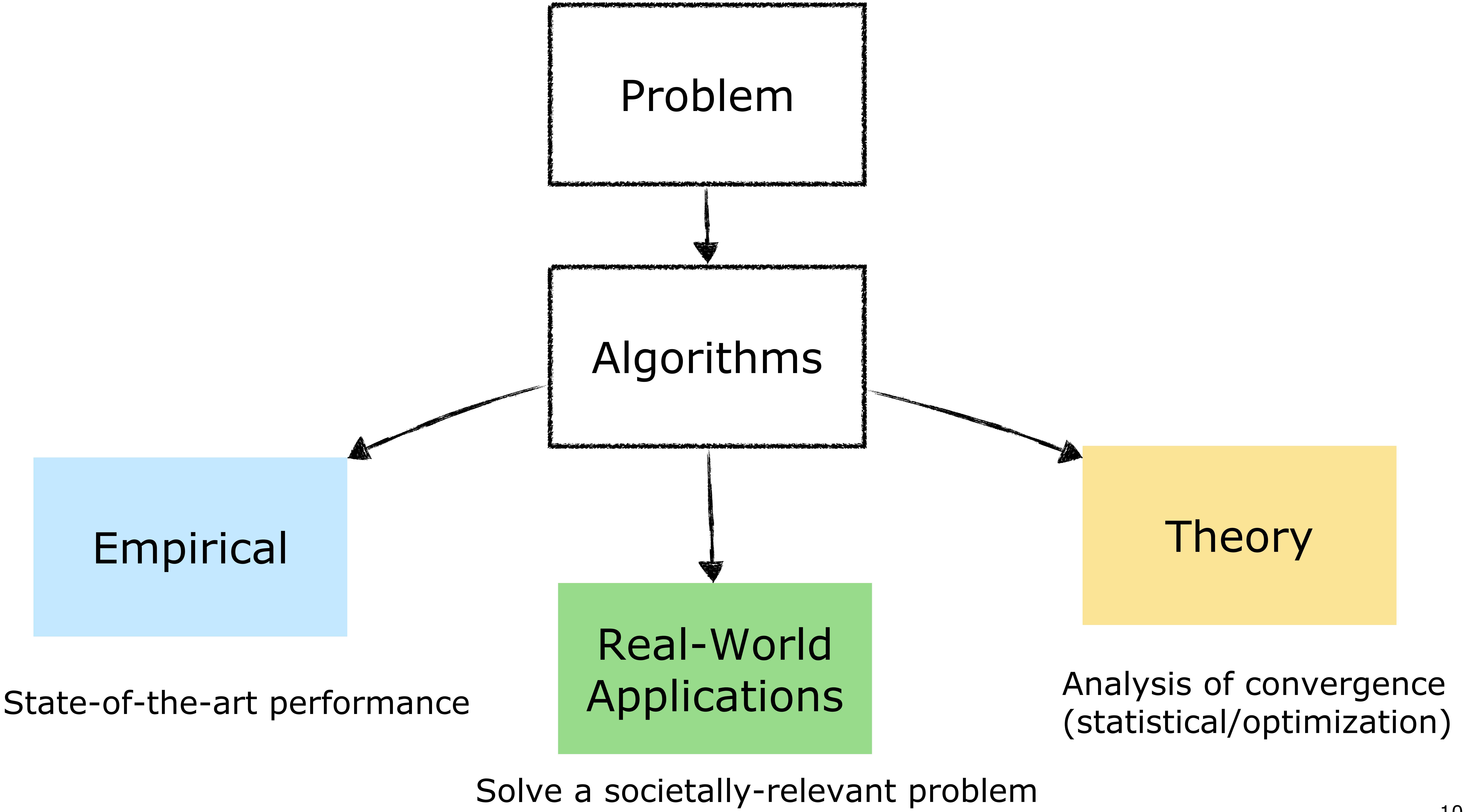
- Can integrate with existing recipes for designing canaries

Ongoing Projects



State-of-the-art performance

Analysis of convergence
(statistical/optimization)

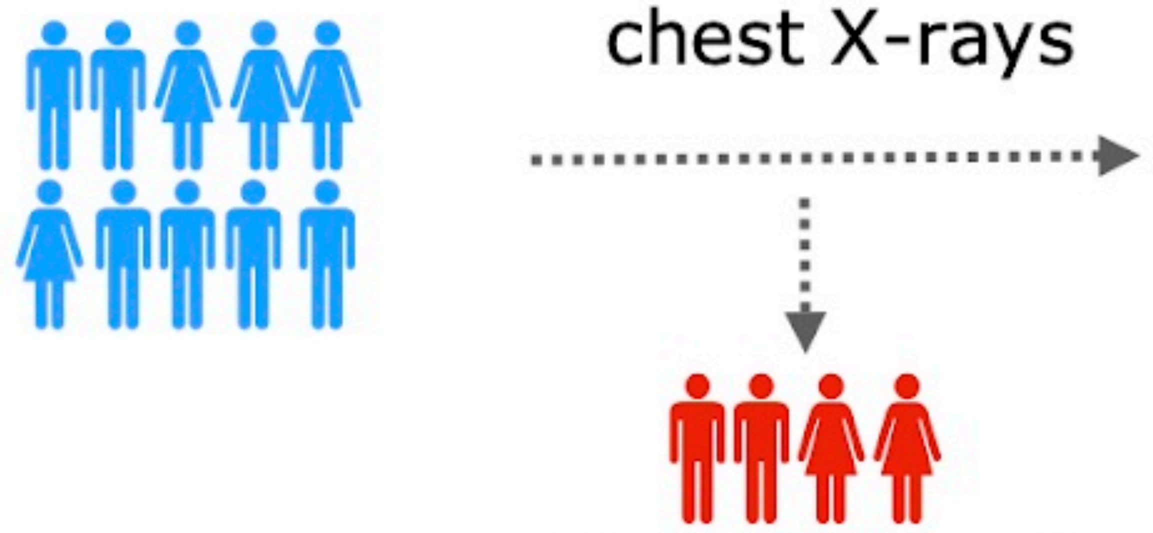

Problem
Algorithms
Empirical
State-of-the-art performance
Real-World Applications
Solve a societally-relevant problem
Theory
Analysis of convergence (statistical/optimization)

TB Detection with Privacy-Preserving AI

Usual Approach

TB positive patients

Detection with chest X-rays



Missed detections

× spread to 10-15 others
× 50% mortality rate

TB Detection with Privacy-Preserving AI

Usual Approach

TB positive patients

Detection with chest X-rays

Missed detections

× spread to 10-15 others
× 50% mortality rate

Our Approach

TB positive patients

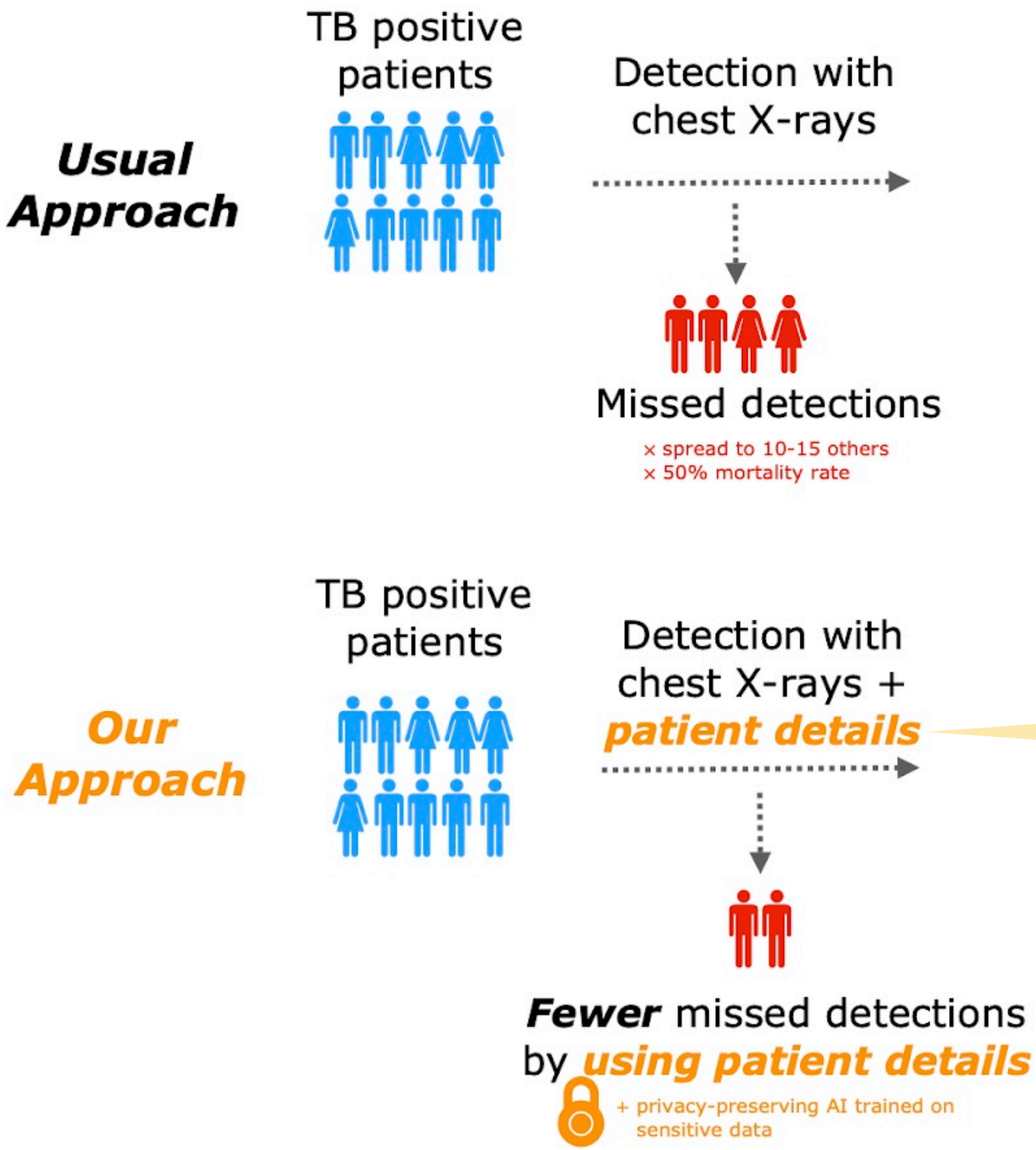
Detection with chest X-rays + ***patient details***

Privacy-sensitive!

Fewer missed detections by ***using patient details***

+ privacy-preserving AI trained on sensitive data

TB Detection with Privacy-Preserving AI

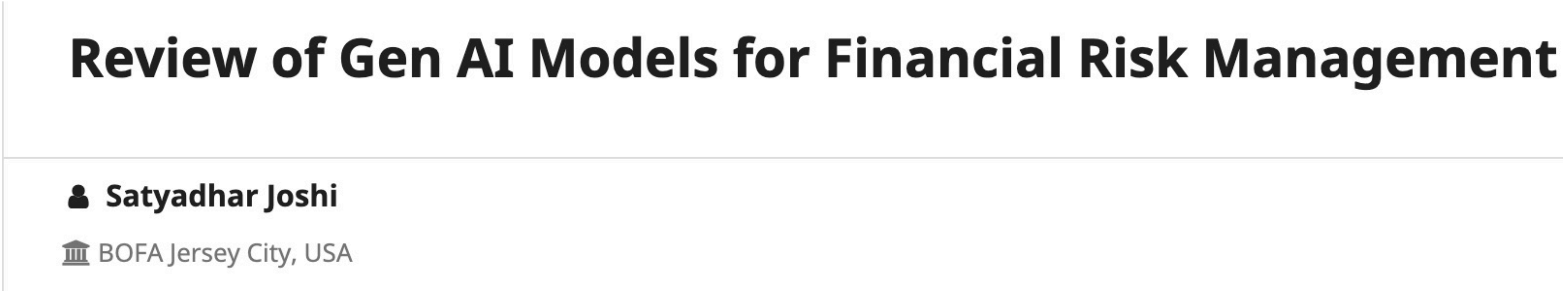


Technical Problem:
Mixed public-private
multi-modal learning

Privacy-sensitive synthetic data generation

Text-to-speech data augmentation

Financial fraud detection



Review of Gen AI Models for Financial Risk Management

Satyadhar Joshi

BOFA Jersey City, USA

Useful to address data imbalances, bias

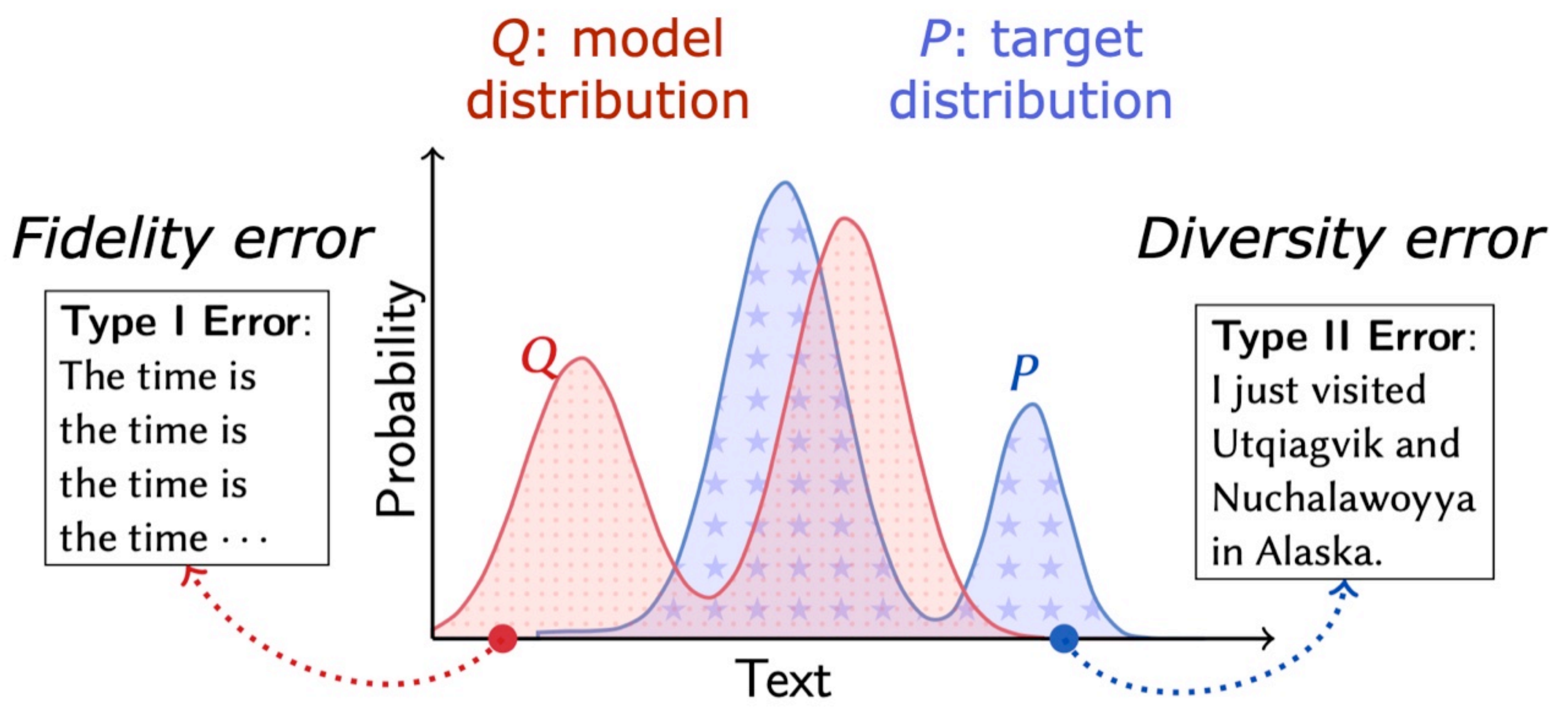
Evaluating synthetic data from LLMs/Gen AI

Project Page: *https://krishnap25.github.io/mauve-overview/*

- ***Pillutla****, Liu*, Thickstun, Welleck, Swayamdipta, Zellers, Oh, Choi, Harchaoui. **MAUVE Scores for Generative Models: Theory and Practice**. *Journal of Machine Learning Research* (2023).

- ***Pillutla***, Swayamdipta, Zellers, Thickstun, Welleck, Choi, Harchaoui. **MAUVE: Measuring the Gap Between Neural Text and Human Text using Divergence Frontiers**. *NeurIPS* (2021). ***Outstanding Paper Award***.

- Liu, ***Pillutla***, Welleck, Oh, Choi, Harchaoui. **Divergence Frontiers for Generative Models: Sample Complexity, Quantization Effects, and Frontier Integrals**. *NeurIPS* (2021).



Type I Error $= D_f(Q\|P)$ Type II Error $= D_f(P\|Q)$

mauve-text 0.4.0

```
pip install mauve-text
```

```
import mauve

# call mauve.compute_mauve using raw text on GPU 0; eac
out = mauve.compute_mauve(p_text=p_text, q_text=q_text,
print(out.mauve) # prints 0.9917
```

Evaluate

Outstanding Paper

NEURAL INFORMATION PROCESSING SYSTEMS

NeurIPS 2021

Privacy-sensitive synthetic data generation

Text-to-speech data augmentation

Financial fraud detection

Review of Gen AI Models for Financial Risk Management

Satyadhar Joshi
BOFA Jersey City, USA

Useful to address data imbalances, bias

Thank you!

https://krishnap25.github.io

@KrishnaPillutla

https://www.linkedin.com/in/krishna-pillutla-a0b1b2a8/

Advertisement: MS/PhD Openings in my group at IIT Madras

- Areas of interest in ML/AI:
  - Privacy-preserving AI
  - Making (generative) AI more robust
  - Applications in healthcare + public good

- Flavour:
  - Theoretical foundations +
  - State of the art empirical performance +
  - Real-world applications