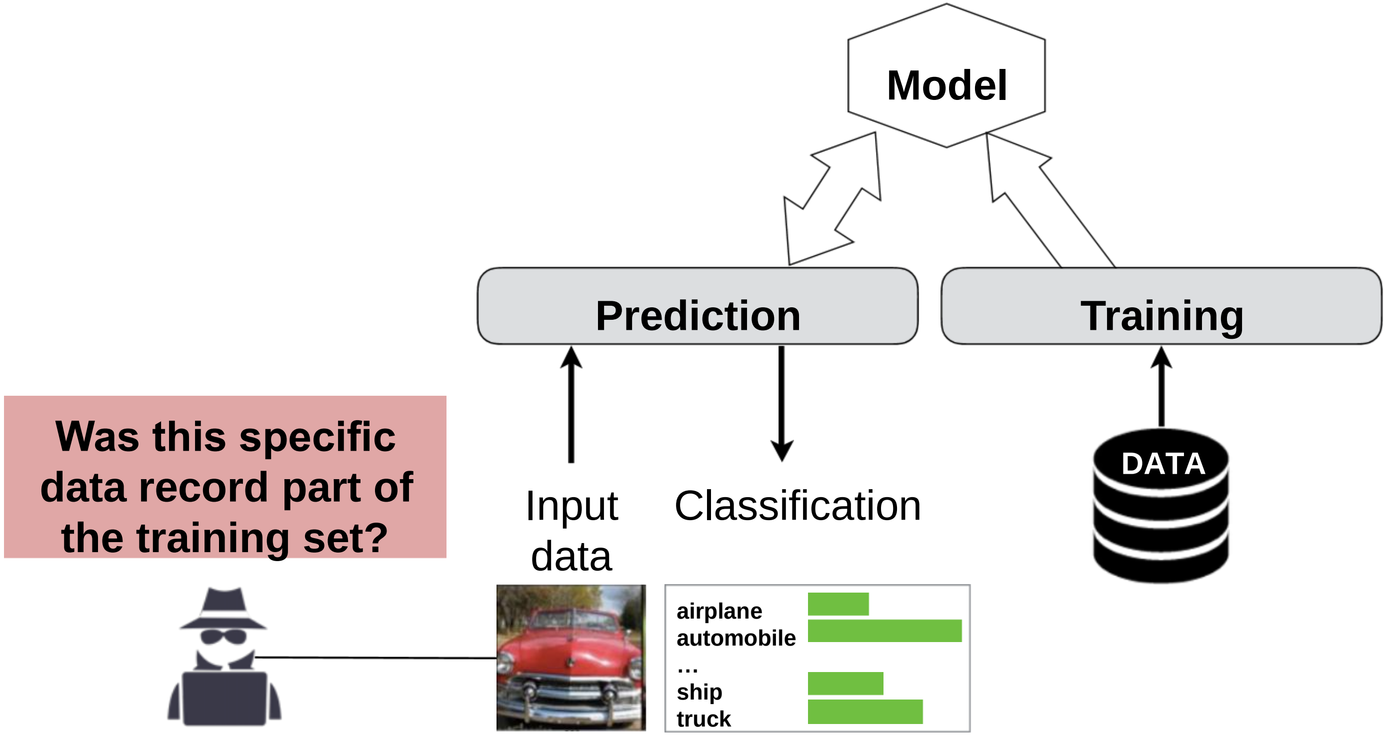
Membership Inference Attacks against Machine Learning Models

Reza Shokri, Marco Stronati, Congzheng Song, Vitaly Shmatikov

CORNELL TECH

Membership Inference Attack



Membership Inference Attack

on Summary Statistics

- Summary statistics (e.g., average) on each attribute
- Underlying distribution of data is known

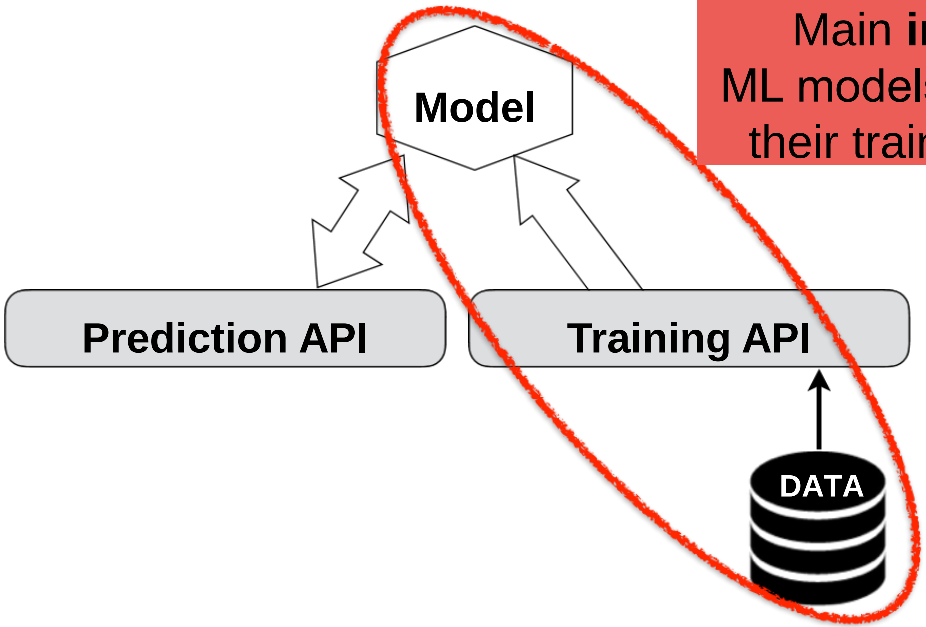
[Homer et al. (2008)], [Dwork et al. (2015)], [Backes et al. (2016)]

on Machine Learning Models

Black-box setting:

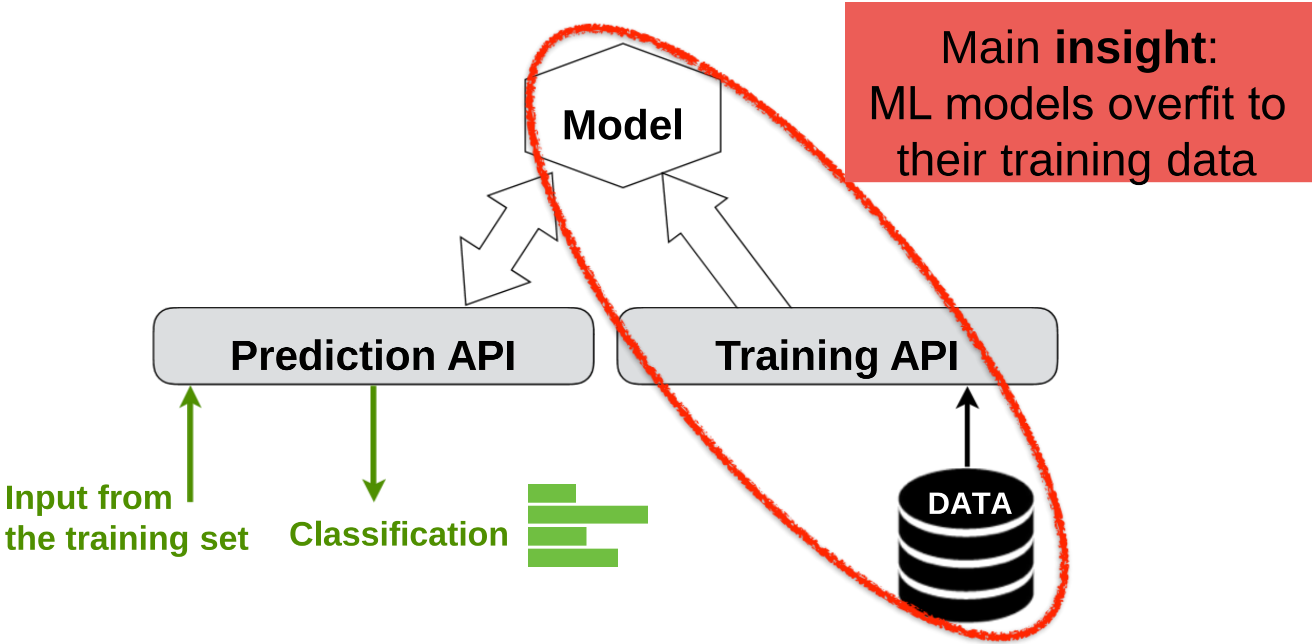
- No knowledge about the models' parameters
- No access to internal computations of the model
- No knowledge about the underlying distribution of data

Exploit Model's Predictions

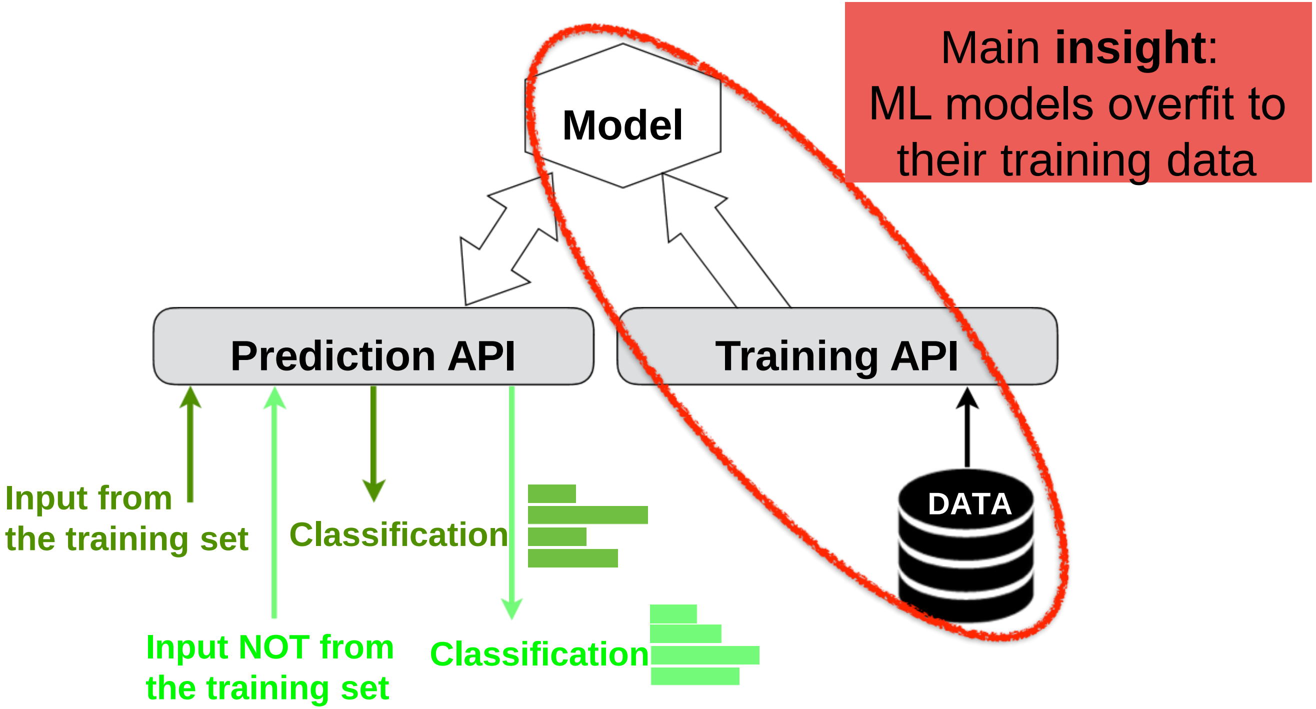


Main **insight**:
ML models overfit to their training data

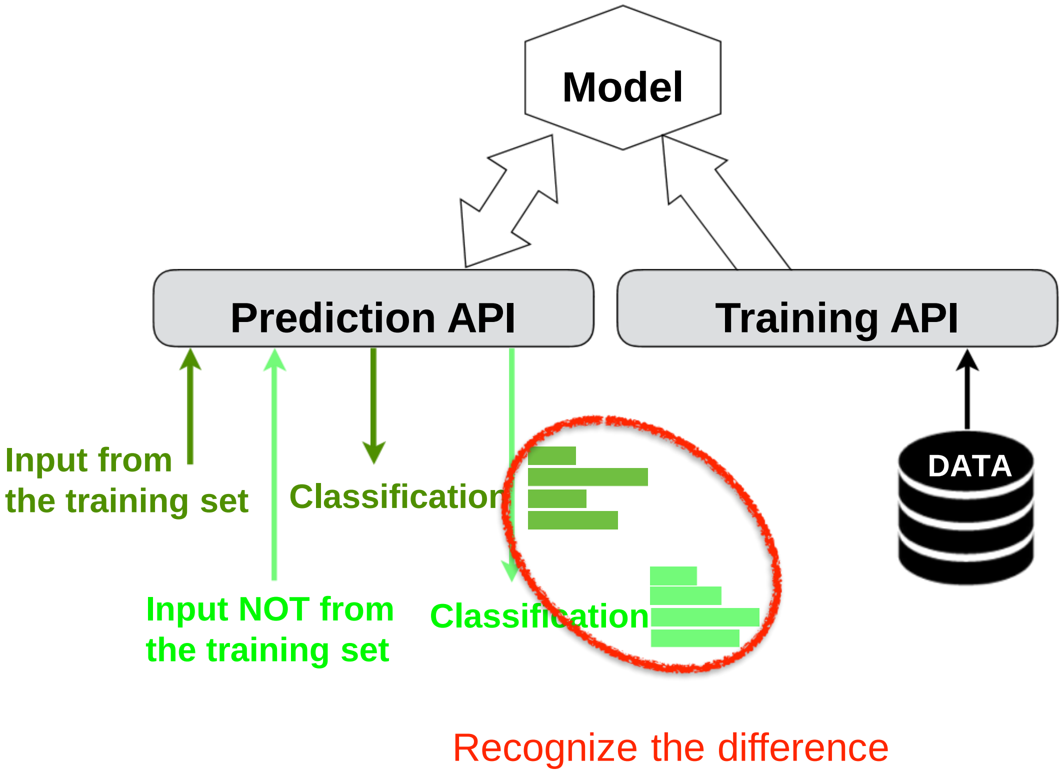
Exploit Model's Predictions



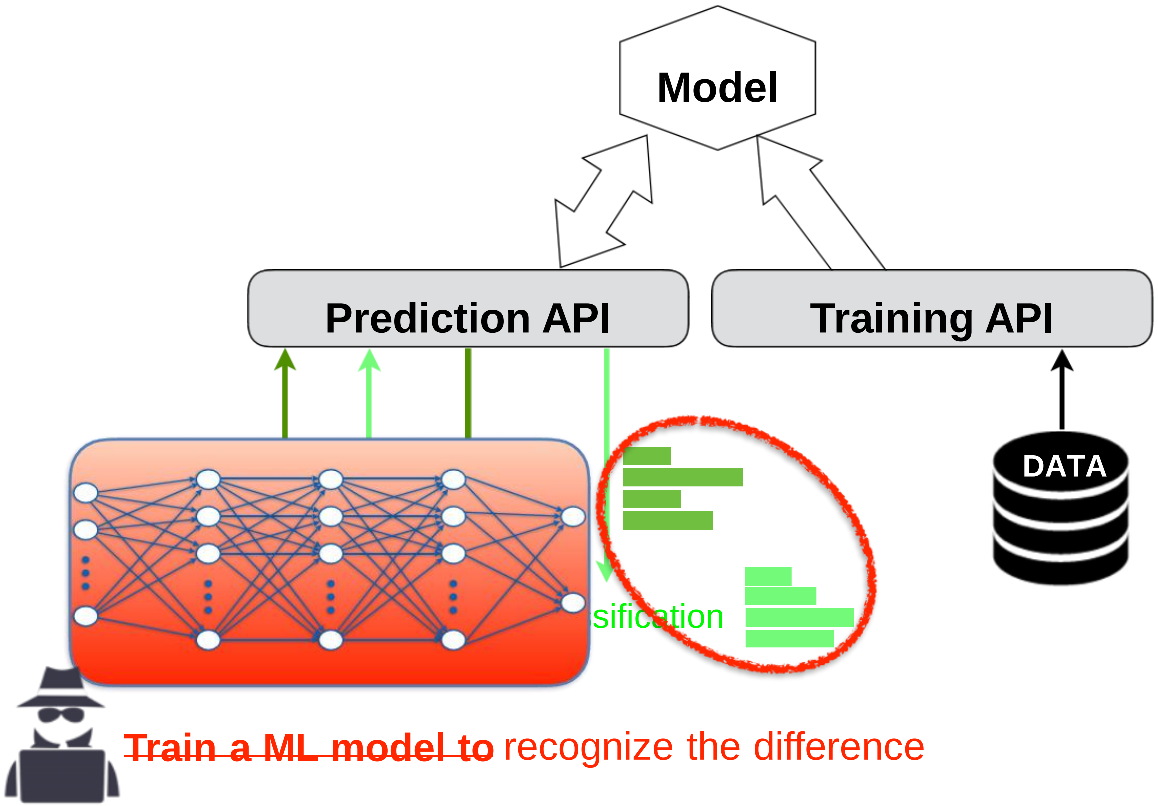
Exploit Model's Predictions



Exploit Model's Predictions

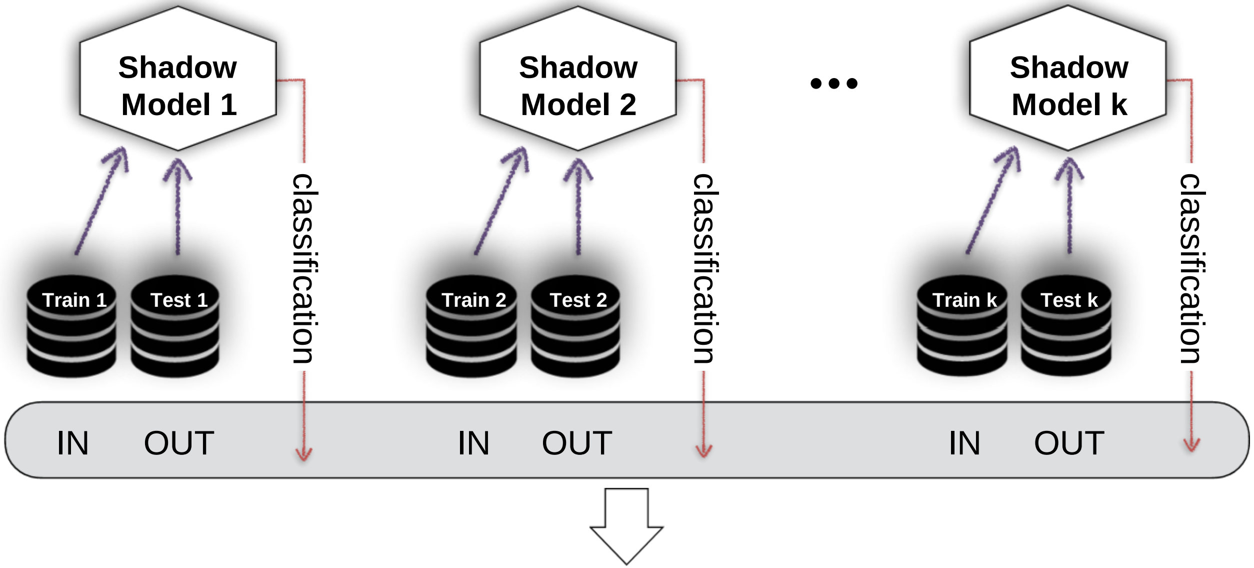


ML against ML



~~Train a ML model to~~ recognize the difference

Train Attack Model using **Shadow Models**



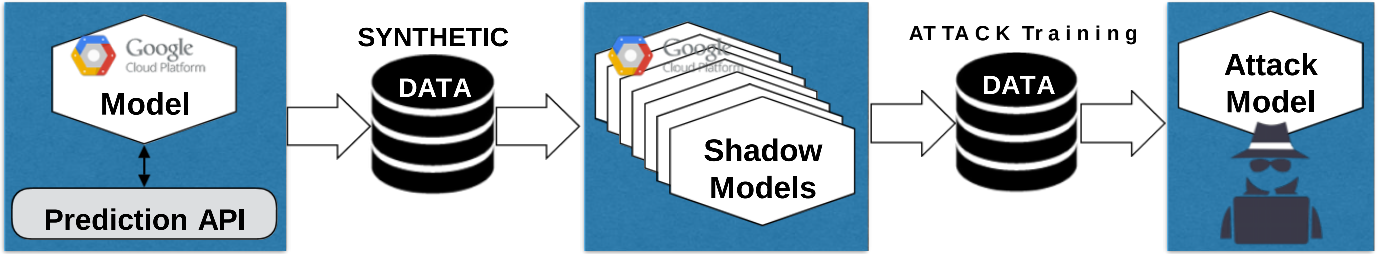
Train the attack model

to predict if an input was a member of the training set (in) or a non-member (out)

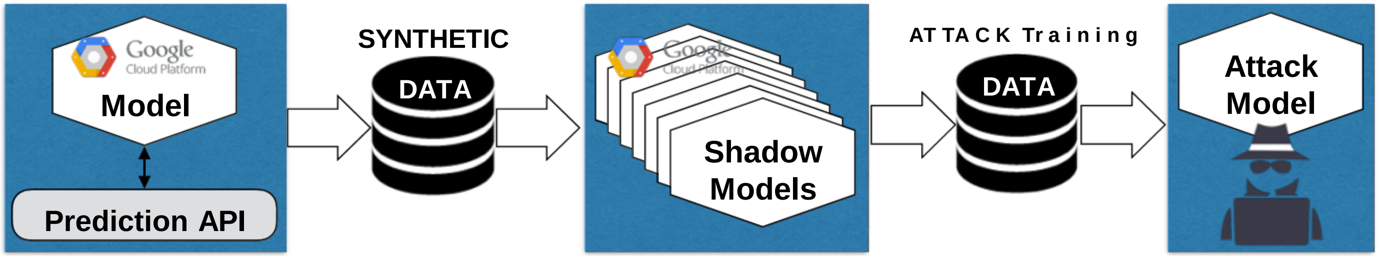
Obtaining Data for Training Shadow Models

- **Real**: similar to training data of the target model (i.e., drawn from same distribution)
- **Synthetic**: use a sampling algorithm to obtain data classified with high confidence by the target model

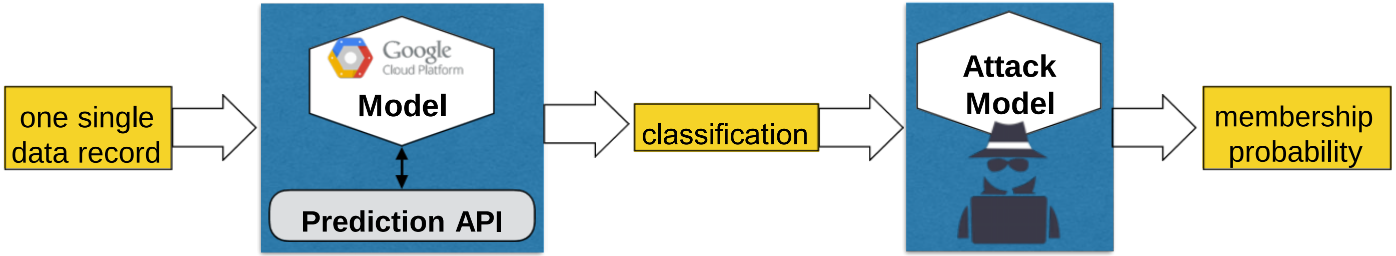
Constructing the Attack Model

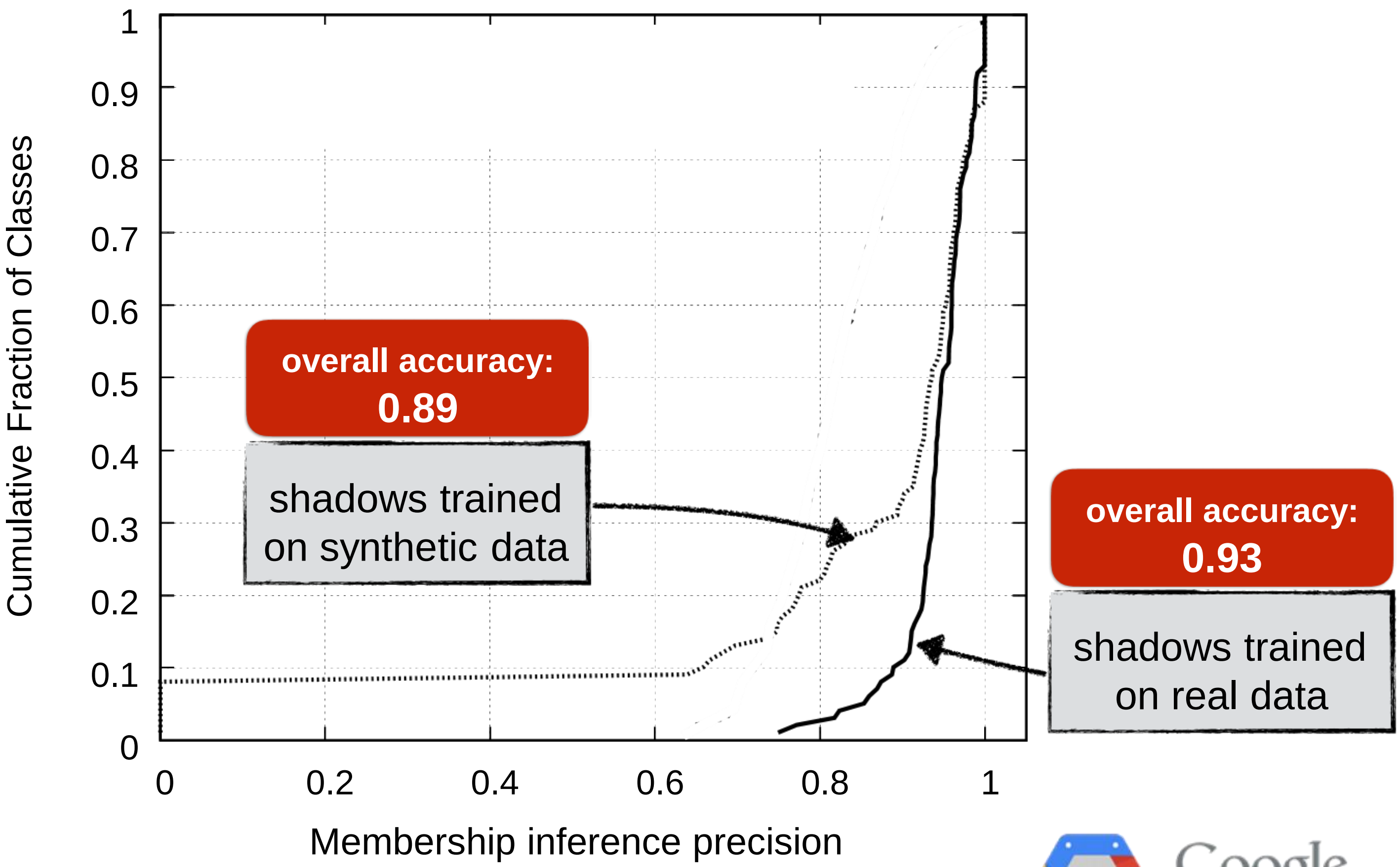


Constructing the Attack Model



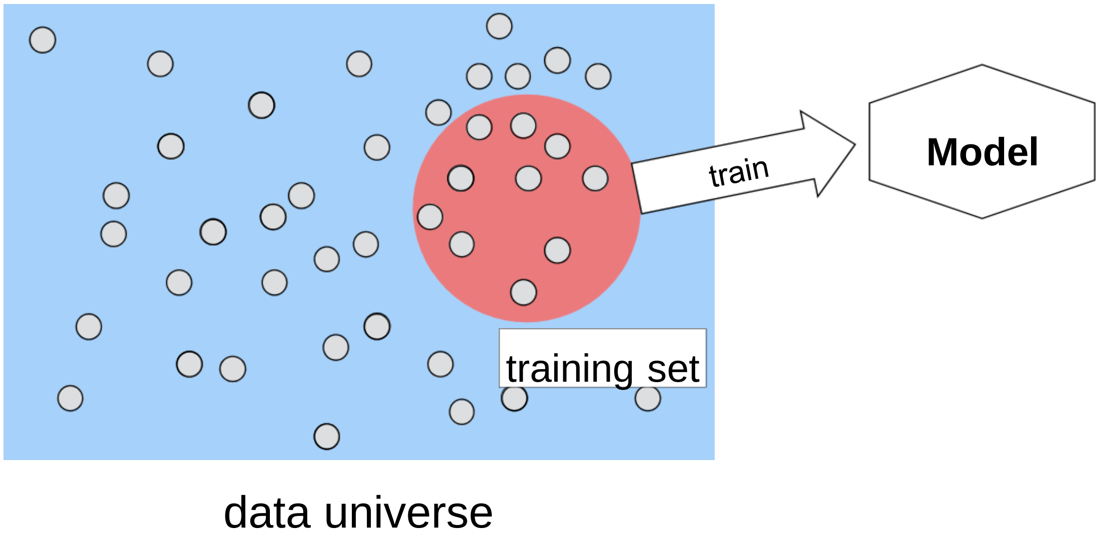
Using the Attack Model





Purchase Dataset — Classify Customers (100 classes)

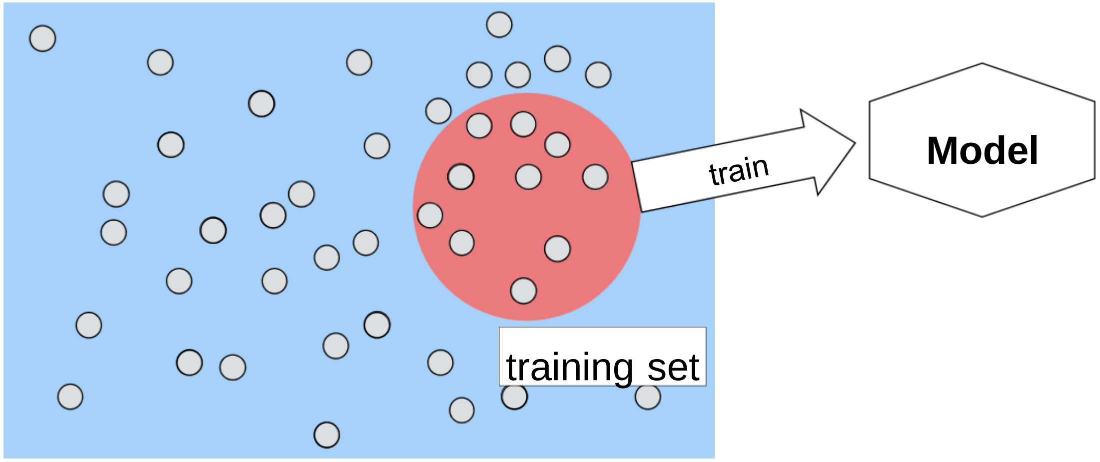
Privacy Learning



Privacy

Learning

Does the model leak information about data in the training set?



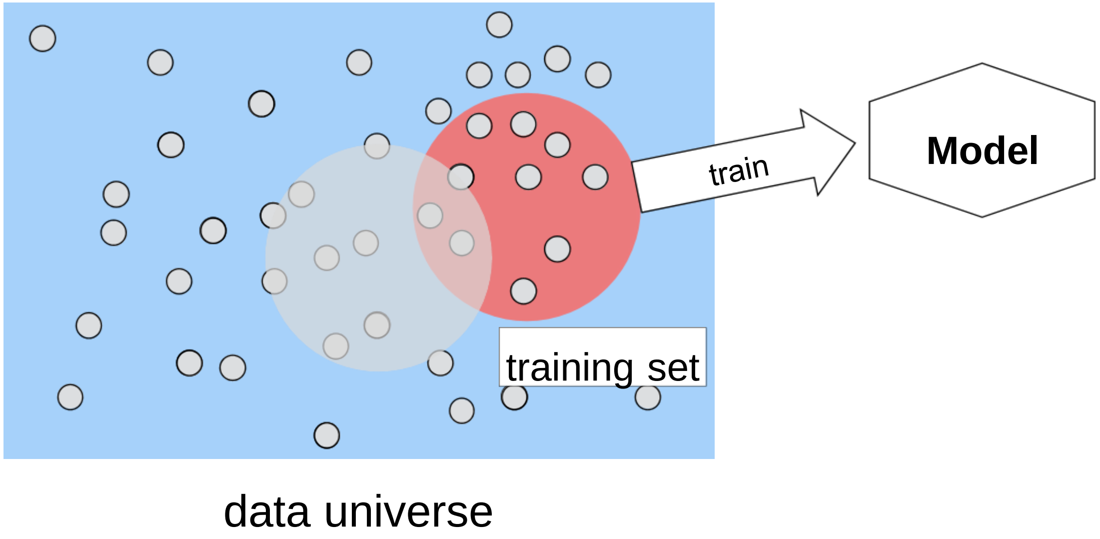
data universe

Privacy

Does the model leak information about data in the training set?

Learning

Does the model generalize to data outside the training set?

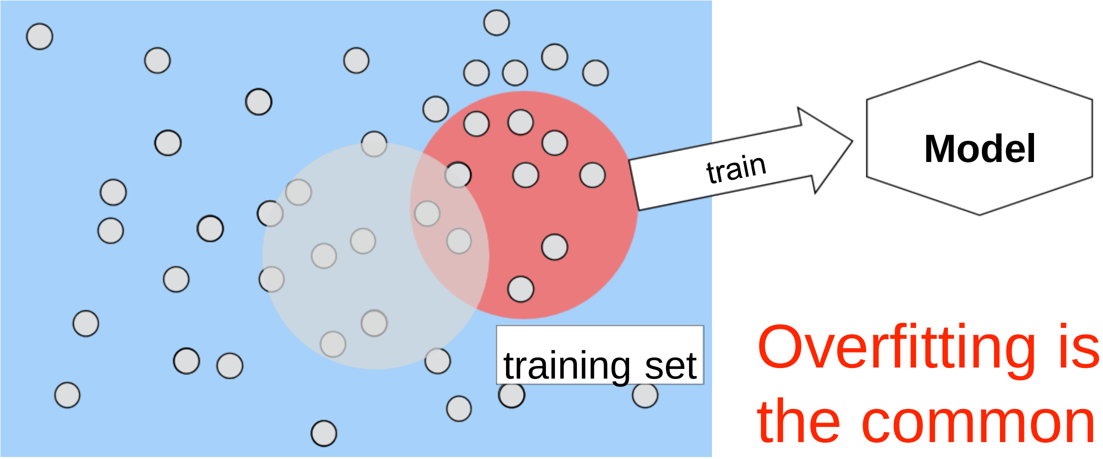


Privacy

Does the model leak information about data in the training set?

Learning

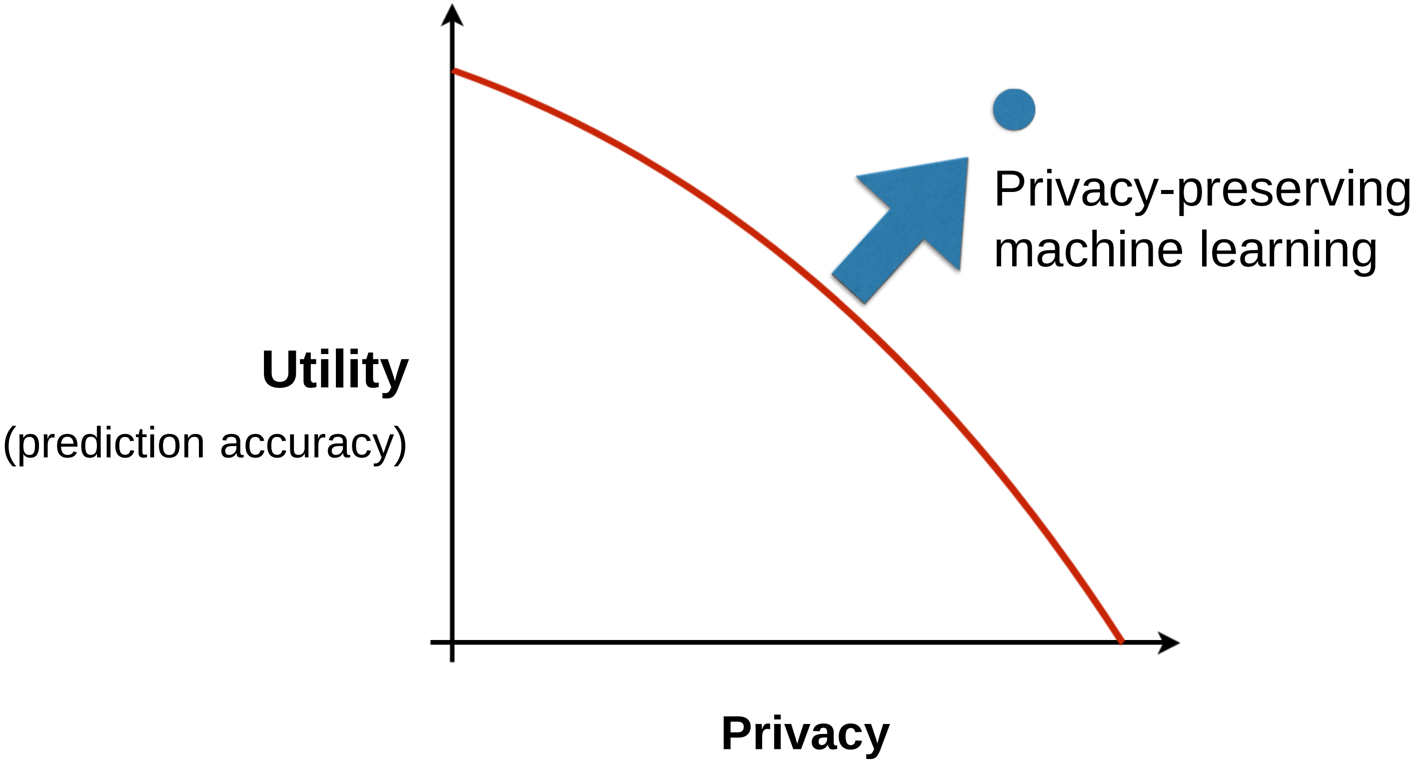
Does the model generalize to data outside the training set?



data universe

Overfitting is the common enemy!

Not in a Direct Conflict!



DEEP LEARNING WITH DIFFERENTIAL PRIVACY

Martin Abadi, Andy Chu, Ian Goodfellow*,
Brendan McMahan, Ilya Mironov, Kunal Talwar, Li Zhang
Google

* OpenAI

Differential Privacy

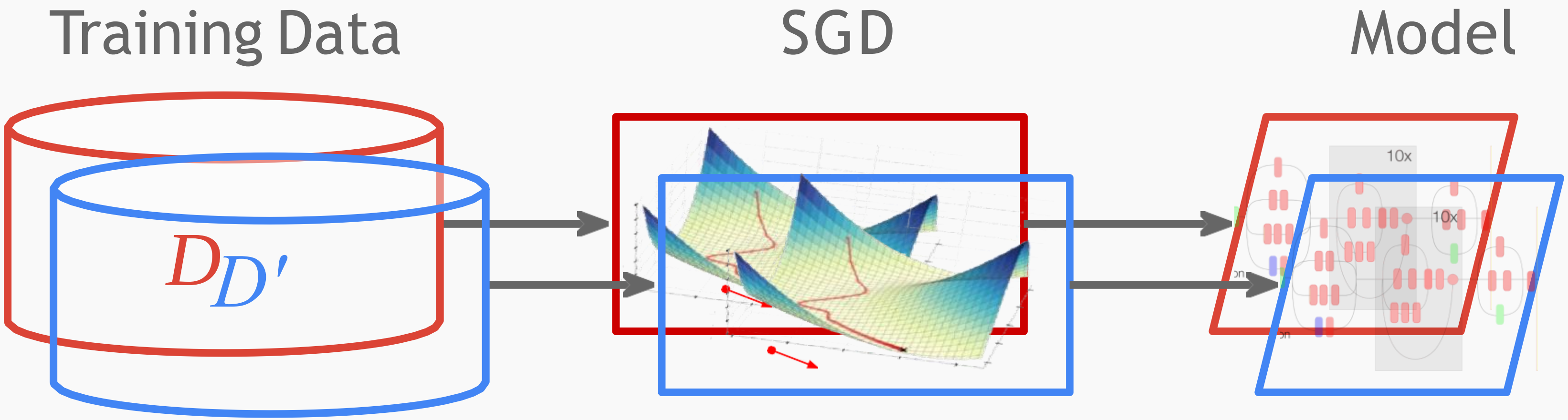
(ε, δ)-Differential Privacy: The distribution of the output $M(D)$ on database D is (nearly) the same as $M(D')$:

$$\forall S: \qquad \Pr[M(D) \in S] \leq \exp(\varepsilon) \cdot \Pr[M(D') \in S] + \delta.$$

quantifies information leakage

allows for a small probability of failure

Interpreting Differential Privacy

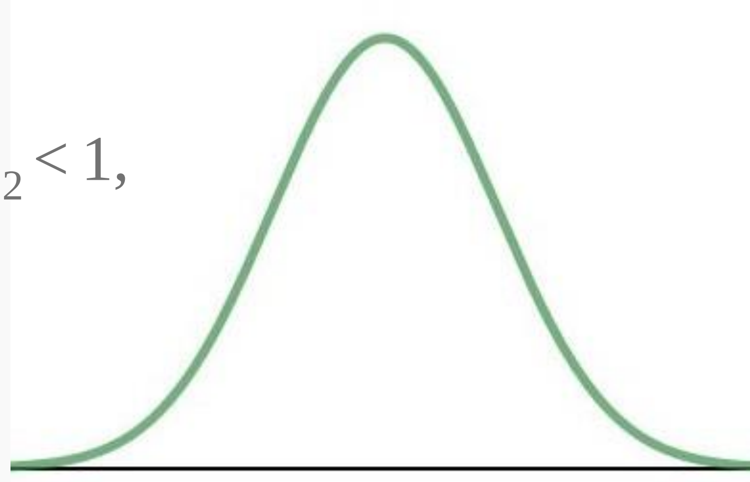


Differential Privacy: Gaussian Mechanism

If ℓ_2-sensitivity of f:D$\rightarrow\mathbb{R}^n$:

$$\max_{D,D'} \|f(D) - f(D')\|_2 < 1,$$

then the Gaussian mechanism



$$f(D) + N^n(0, \sigma^2)$$

offers (ε, δ)-differential privacy, where $\delta \approx \exp(-(\varepsilon\sigma)^2/2)$.

Dwork, Kenthapadi, McSherry, Mironov, Naor, “Our Data, Ourselves”, Eurocrypt 2006

Basic Composition Theorem

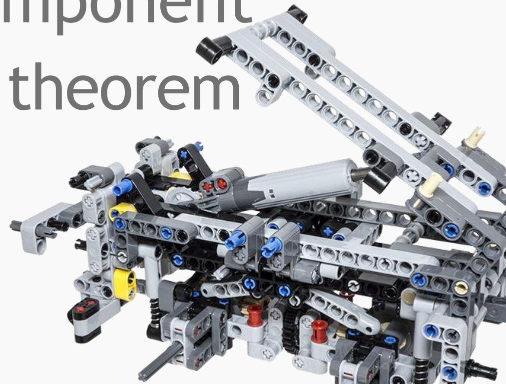
If f is $(\varepsilon_1, \delta_1)$-DP and g is $(\varepsilon_2, \delta_2)$-DP, then

$$f(D), g(D) \text{ is } (\varepsilon_1+\varepsilon_2, \delta_1+\delta_2)\text{-DP}$$

Simple Recipe for Composite Functions

To compute composite f with differential privacy

1. Bound sensitivity of f's components
2. Apply the Gaussian mechanism to each component
3. Compute total privacy via the composition theorem



Deep Learning with Differential Privacy

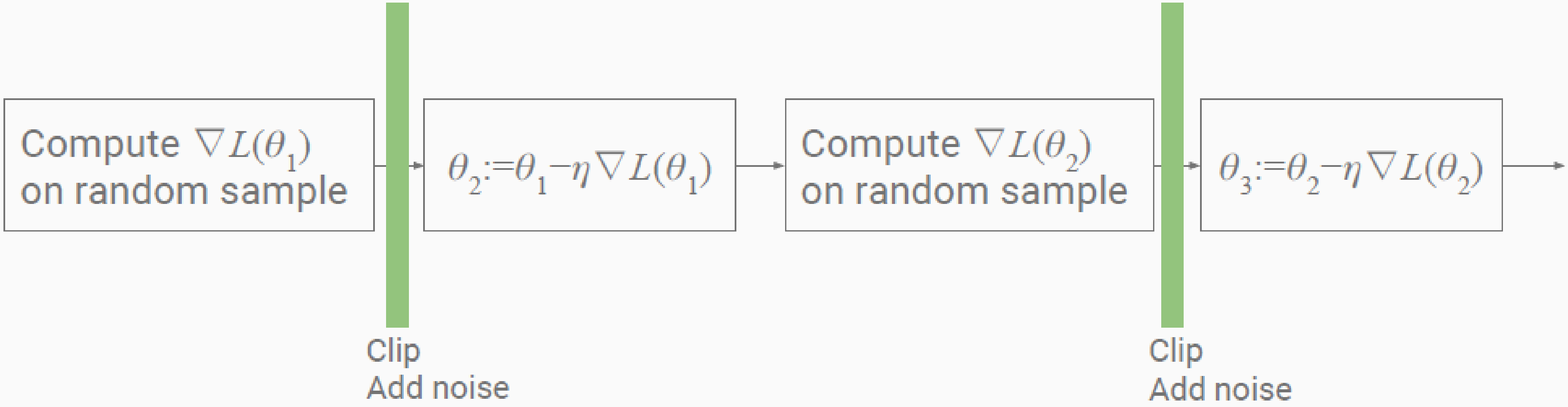
Differentially Private Deep Learning

1. Loss function	softmax loss
2. Training / Test data	MNIST and CIFAR-10
3. Topology	PCA + neural network
4. Training algorithm	Differentially private SGD
5. Hyperparameters	tune experimentally

Stochastic Gradient Descent

Compute $\nabla L(\theta_1)$ on random sample → $\theta_2 := \theta_1 - \eta \nabla L(\theta_1)$ → Compute $\nabla L(\theta_2)$ on random sample → $\theta_3 := \theta_2 - \eta \nabla L(\theta_2)$ →

Stochastic Gradient Descent with Differential Privacy



Algorithm 1 Differentially private SGD (Outline)

Input: Examples $\{x_1, \ldots, x_N\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$. Parameters: learning rate η_t, noise scale σ, group size L, gradient norm bound C.

Initialize θ_0 randomly

for $t \in [T]$ **do**

Take a random sample L_t with sampling probability L/N

Compute gradient

For each $i \in L_t$, compute $\mathbf{g}_t(x_i) \leftarrow \nabla_{\theta_t}\mathcal{L}(\theta_t, x_i)$

Clip gradient

$\bar{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i) / \max\left(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}\right)$

Add noise

$\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L}\left(\sum_i \bar{\mathbf{g}}_t(x_i) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I})\right)$

Descent

$\theta_{t+1} \leftarrow \theta_t - \eta_t \tilde{\mathbf{g}}_t$

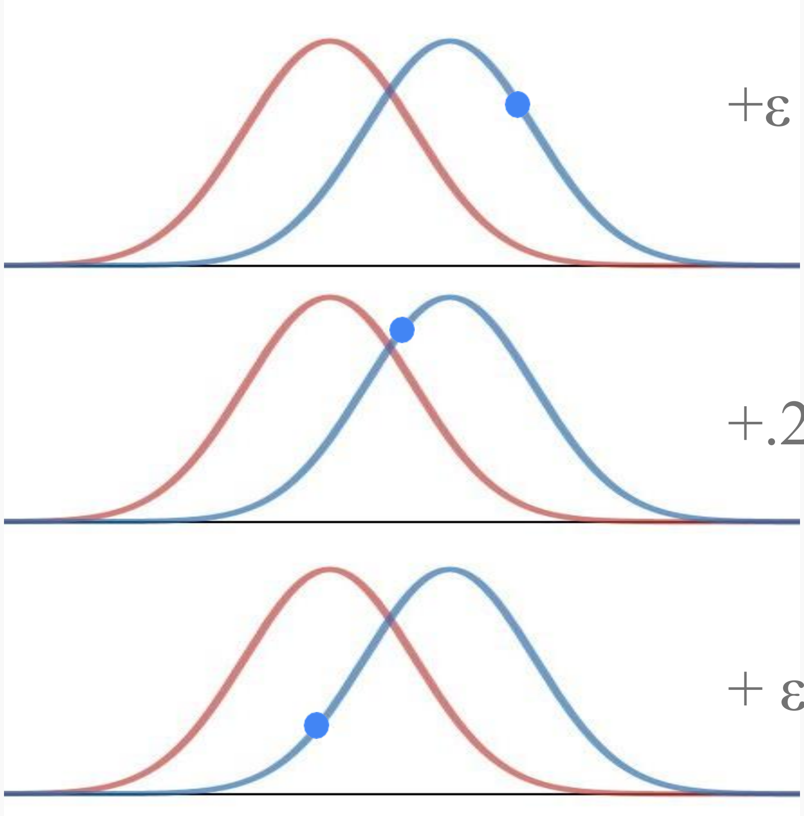
Output θ_T and compute the overall privacy cost (ε, δ) using a privacy accounting method.

Naïve Privacy Analysis

1. Choose $\sigma = \frac{\sqrt{2 \log 1/\delta}}{\varepsilon}$	$= 4$
2. Each step is (ε, δ)-DP	$(1.2, 10^{-5})$-DP
3. Number of steps T	10,000
4. Composition: $(T\varepsilon, T\delta)$-DP	$\boxed{(12{,}000, .1)\text{-DP}}$

Advanced Composition Theorems

Composition theorem



$+\varepsilon$ for Blue

$+.2\varepsilon$ for Blue

$+\ \varepsilon$ for Red

Strong Composition Theorem

1. Choose $\sigma = \frac{\sqrt{2 \log 1/\delta}}{\varepsilon}$	= 4
2. Each step is (ε, δ)-DP	$(1.2, 10^{-5})$-DP
3. Number of steps T	10,000
4. Strong comp: $(\varepsilon\sqrt{T \log 1/\delta}, T\delta)$-DP	$\boxed{(360, .1)\text{-DP}}$

Dwork, Rothblum, Vadhan, "Boosting and Differential Privacy", FOCS 2010
Dwork, Rothblum, "Concentrated Differential Privacy", https://arxiv.org/abs/1603.0188

Amplification by Sampling

1. Choose $\sigma = \frac{\sqrt{2 \log 1/\delta}}{\varepsilon}$	$= 4$
2. Each batch is q fraction of data	1%
3. Each step is $(2q\varepsilon, q\delta)$-DP	$(.024, 10^{-7})$-DP
4. Number of steps T	10,000
5. Strong comp: $(2q\varepsilon\sqrt{T \log 1/\delta}, qT\delta)$-DP	$\boxed{(10, .001)\text{-DP}}$

S. Kasiviswanathan, H. Lee, K. Nissim, S. Raskhodnikova, A. Smith, “What Can We Learn Privately?”, SIAM J. Comp, 2011

Moments Accountant

1. Choose $\sigma = \frac{\sqrt{2 \log 1/\delta}}{\varepsilon}$ = 4
2. Each batch is q fraction of data 1%
3. Keeping track of privacy loss's **moments**
4. Number of steps T 10,000
5. Moments: $(2q\varepsilon\sqrt{T}, \delta)$-DP $\boxed{(1.25, 10^{-5})\text{-DP}}$

Results

Our Datasets: “Fruit Flies of Machine Learning”

MNIST dataset:

70,000 images

28×28 pixels each

CIFAR-10 dataset:

60,000 color images

32×32 pixels each

Summary of Results

	Baseline
	no privacy
MNIST	98.3%
CIFAR-10	80%

Summary of Results

	Baseline	[SS15]	[WKC+16]
	no privacy	reports ε per parameter	ε = 2
MNIST	98.3%	98%	80%
CIFAR-10	80%		

Summary of Results

	Baseline	[SS15]	[WKC+16]	this work		
	no privacy	reports ε per parameter	$\varepsilon = 2$	$\varepsilon = 8$ $\delta = 10^{-5}$	$\varepsilon = 2$ $\delta = 10^{-5}$	$\varepsilon = 0.5$ $\delta = 10^{-5}$
MNIST	98.3%	98%	80%	97%	95%	90%
CIFAR-10	80%			73%	67%	

Contributions

- Differentially private deep learning applied to publicly available datasets and implemented in TensorFlow
  - https://github.com/tensorflow/models
- Innovations
  - Bounding sensitivity of updates
  - Moments accountant to keep tracking of privacy loss
- Lessons
  - Recommendations for selection of hyperparameters
- Full version: https://arxiv.org/abs/1607.00133

SEMI-SUPERVISED KNOWLEDGE TRANSFER FOR DEEP LEARNING FROM PRIVATE TRAINING DATA

Nicolas Papernot*
Pennsylvania State University
ngp5056@cse.psu.edu

Martín Abadi
Google Brain
abadi@google.com

Úlfar Erlingsson
Google
ulfar@google.com

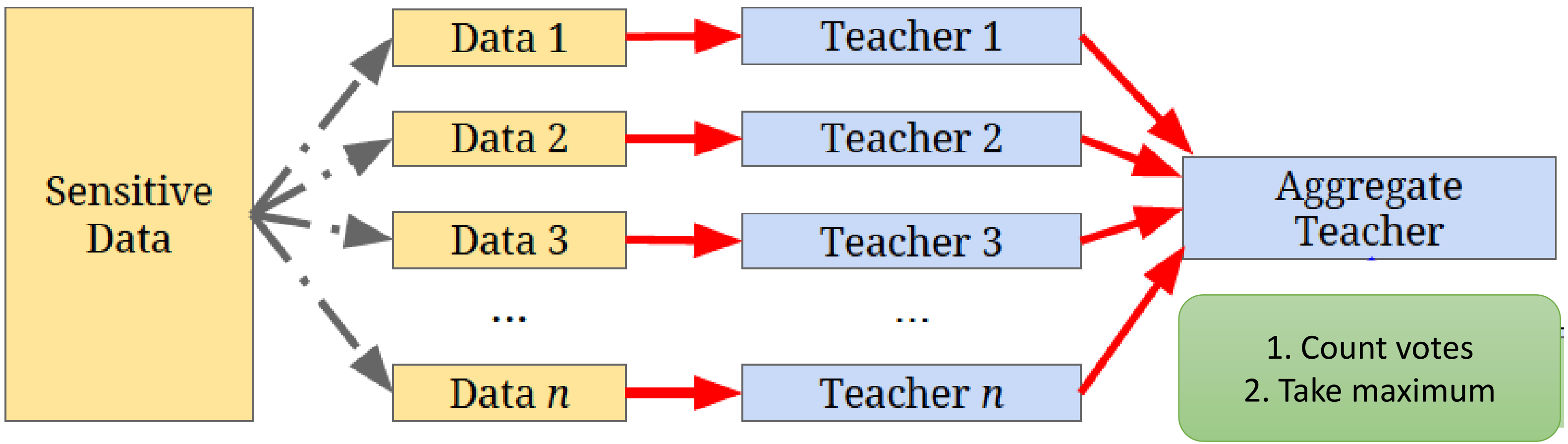
Ian Goodfellow
Google Brain†
goodfellow@google.com

Kunal Talwar
Google Brain
kunal@google.com

In their work, the threat model assumes:

- Adversary can make a potentially unbounded number of queries
- Adversary has access to model internals

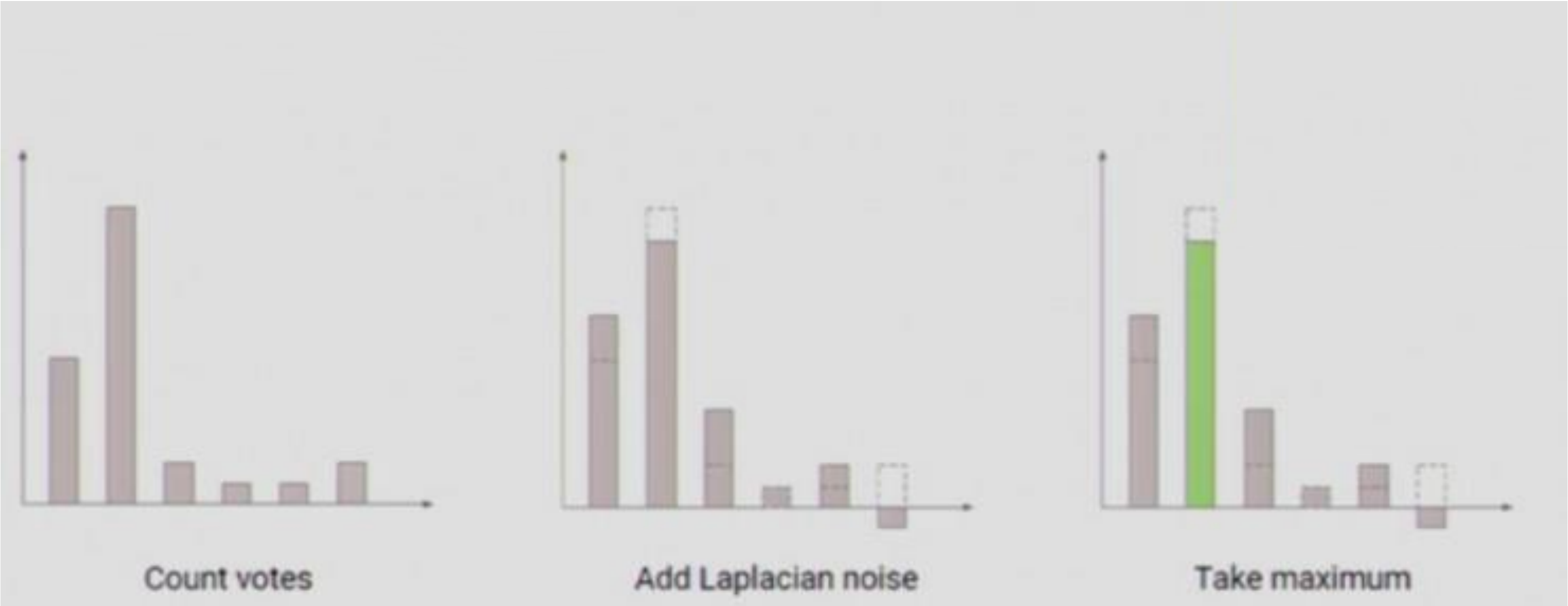
Private Aggregation of Teacher Ensembles (PATE)



Intuitive privacy analysis:

- If most teachers agree on the label, it does not depend on specific partitions, so the privacy cost is small.
- If two classes have close vote counts, the disagreement may reveal private information

Noisy aggregation

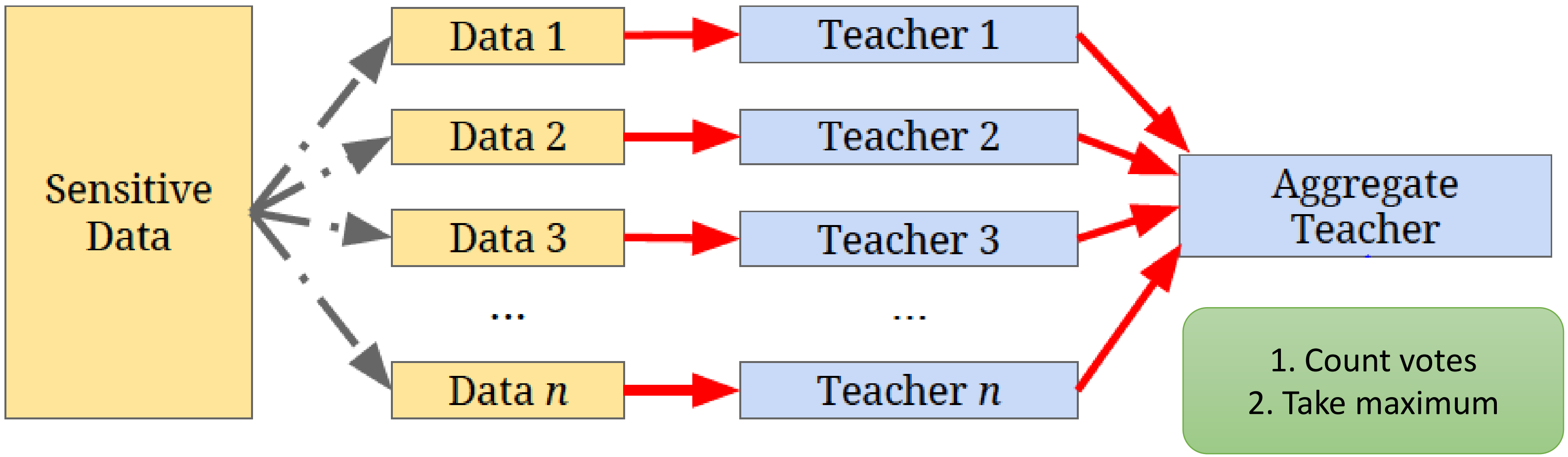


$$n_j(\vec{x}) = |\{i : i \in [n], f_i(\vec{x}) = j\}|$$

$$Lap\left(\frac{1}{\gamma}\right)$$

$$f(x) = \arg\max_j \left\{ n_j(\vec{x}) + Lap\left(\frac{1}{\gamma}\right) \right\}$$

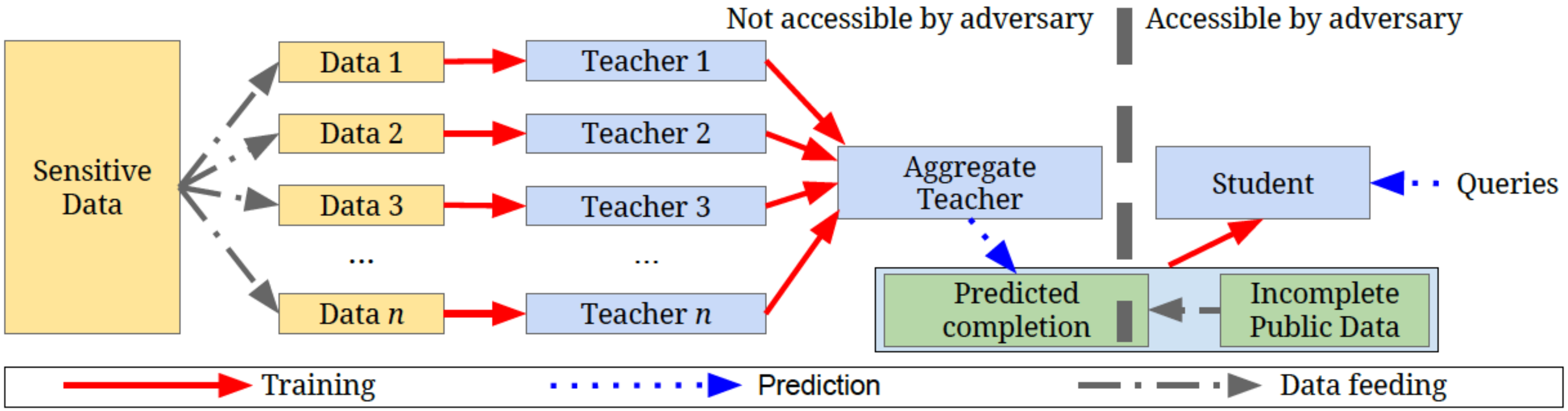
Private Aggregation of Teacher Ensembles (PATE)



The aggregated teacher violates the threat model:

- **Each prediction increases total privacy loss.**
  privacy budgets create a tension between the accuracy and number of predictions
- **Inspection of internals may reveal private data.**
  Privacy guarantees should hold in the face of white-box adversaries

Private Aggregation of Teacher Ensembles (PATE)



Privacy Analysis:

- Privacy loss is fixed after the student model is done training.
- Even if white-box adversary can inspect the model parameters, the information can be revealed from student model is unlabeled public data and labels from aggregate teacher which is protected with privacy

GANs

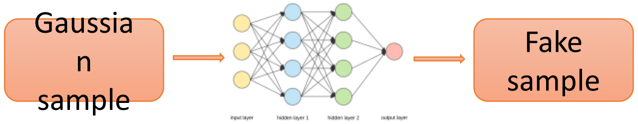
IJ Goodfellow et al. (2014) *Generative Adversarial Networks*

2 computing models

Generator:

Input: noise sampled from random distribution

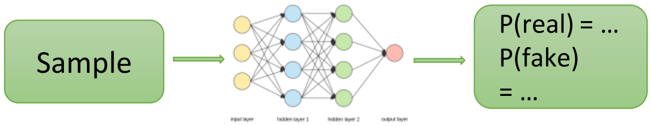
Output: synthetic input close to the expected training distribution



Discriminator:

Input: output from generator OR example from real training distribution

Output: in distribution OR fake



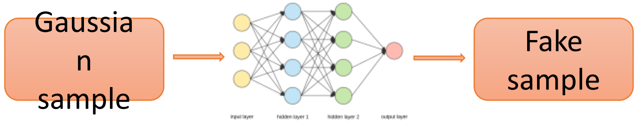
Improved Training of GANs

T Salimans et al. (2016) *Improved Techniques for Training GANs*

Generator:

Input: noise sampled from random distribution

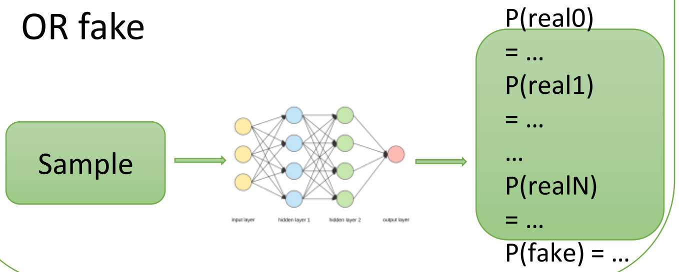
Output: synthetic input close to the expected training distribution



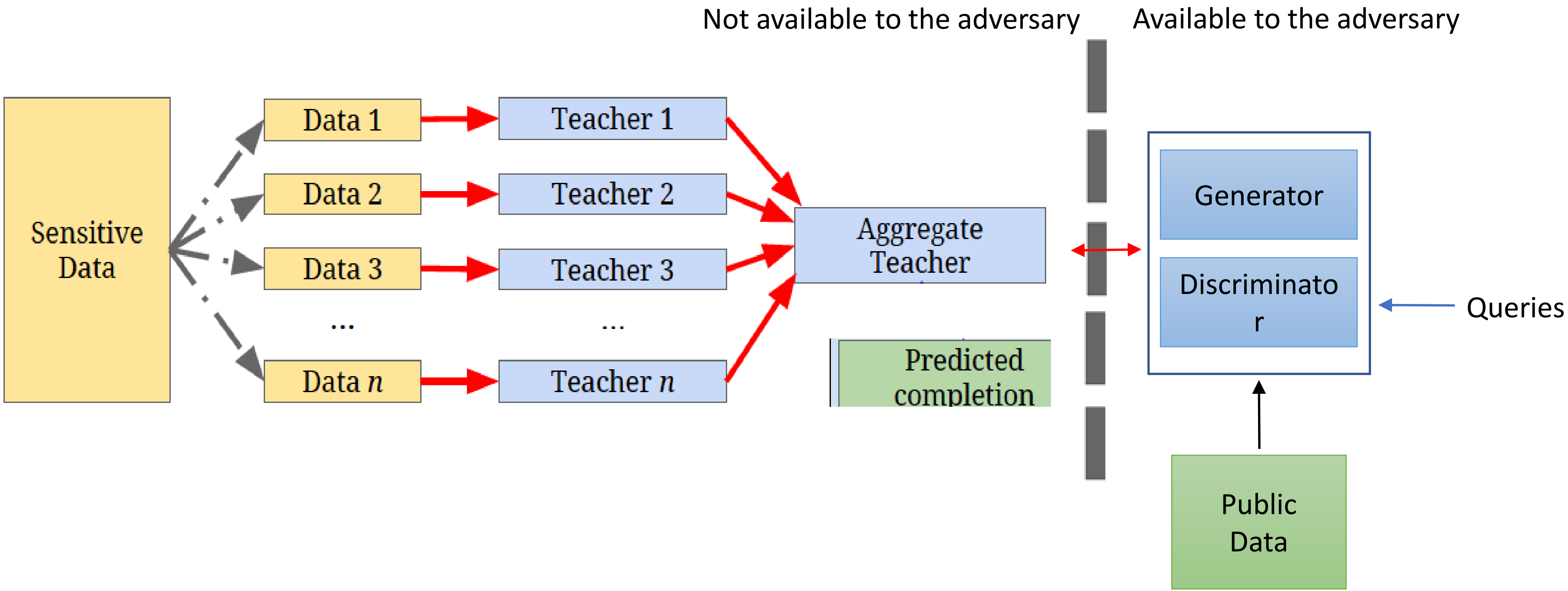
Discriminator:

Input: output from generator OR example from real training distribution

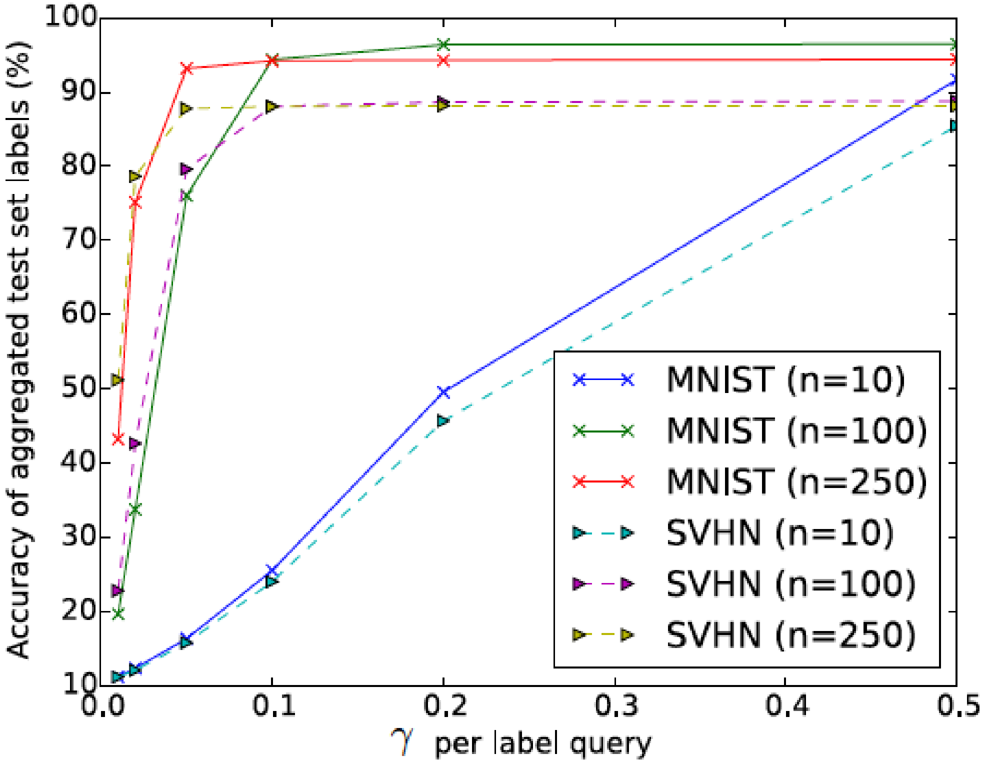
Output: in distribution (**which class**) OR fake



Private Aggregation of Teacher Ensembles using GANs (PATE-G)



Aggregated Teacher Accuracy Before the Student Model is Trained



Evaluation

Dataset	ε	δ	Queries	Non-Private Baseline	Student Accuracy
MNIST	2.04	10^{-5}	100	99.18%	98.00%
MNIST	8.03	10^{-5}	1000	99.18%	98.10%
SVHN	5.04	10^{-6}	500	92.80%	82.72%
SVHN	8.19	10^{-6}	1000	92.80%	90.66%

M Abadi et al. (2016) *Deep Learning with Differential Privacy*

$(0.5, 10^{-5})$ 90%

$(2, 10^{-5})$ 95%

$(8, 10^{-5})$ 97%

increase # teachers will increase privacy guarantee, but decrease model accuracy
teachers is constrained by task's complexity and the available data