Semi-supervised Knowledge Transfer for Deep Learning from Private Training Data

PATE-G

Nicolas Papernot

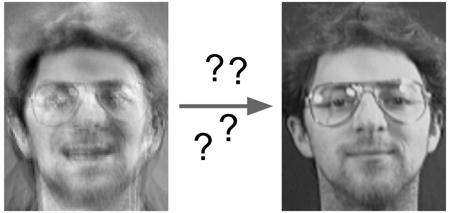
joint work with

Martín Abadi, Úlfar Erlingsson, Ian Goodfellow, Kunal Talwar

Google Brain

Nicolas is at Penn State, was an intern in Brain; Ian did part of the work at OpenAI.

Some challenges of learning from private data



Training-data extraction attacks

Fredrikson et al. (2015) *Model Inversion Attacks*

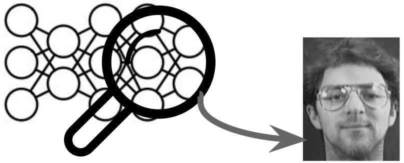
Membership attacks

Shokri et al. (2016) *Membership Inference Attacks against ML Models*

Types of adversaries and our threat model

Model querying (**black-box adversary**)

Shokri et al. (2016) *Membership Inference Attacks against ML Models*
Fredrikson et al. (2015) *Model Inversion Attacks*



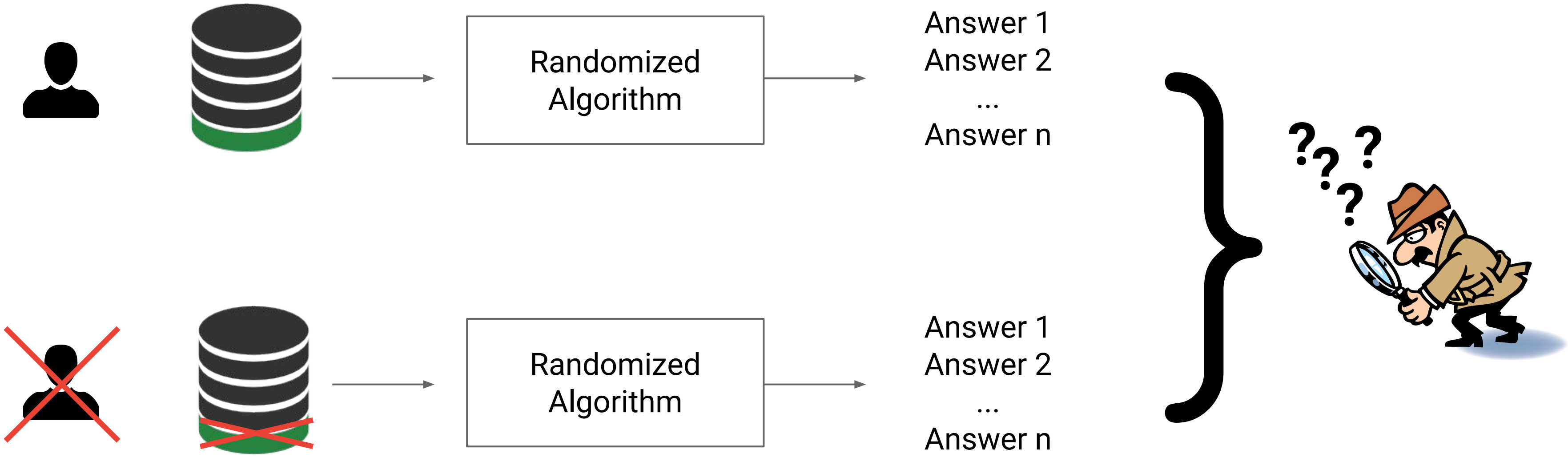
Model inspection (**white-box adversary**)

Zhang et al. (2017) *Understanding DL requires rethinking generalization*

In our work, the threat model assumes:

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

Quantifying privacy



Our design goals

Problem	Preserve **privacy of training data** when learning **classifiers**
Goals	**Differential privacy** protection guarantees **Intuitive privacy** protection guarantees **Generic*** (independent of learning algorithm)

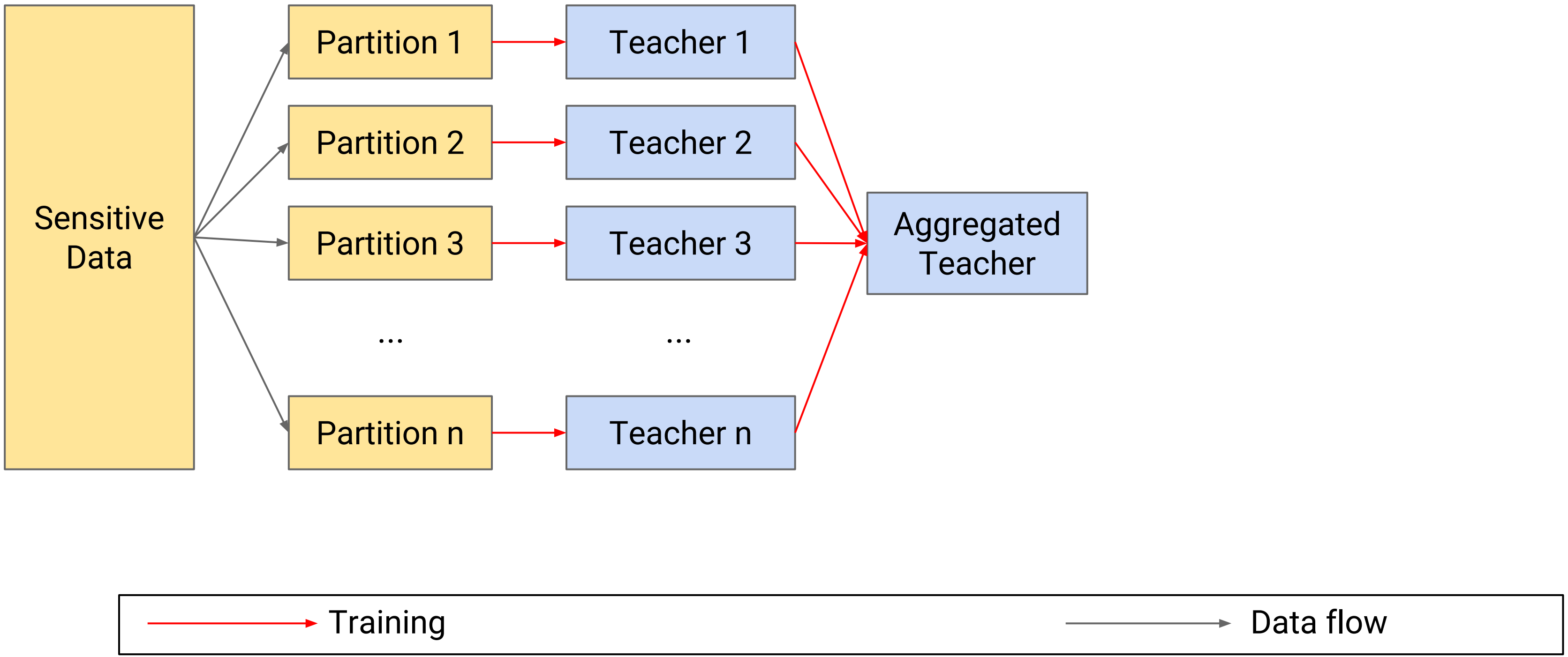
*This is a key distinction from previous work, such as

Pathak et al. (2011) *Privacy preserving probabilistic inference with hidden markov models*
Jagannathan et al. (2013) *A semi-supervised learning approach to differential privacy*
Shokri et al. (2015) *Privacy-preserving Deep Learning*
Abadi et al. (2016) *Deep Learning with Differential Privacy*
Hamm et al. (2016) *Learning privately from multiparty data*

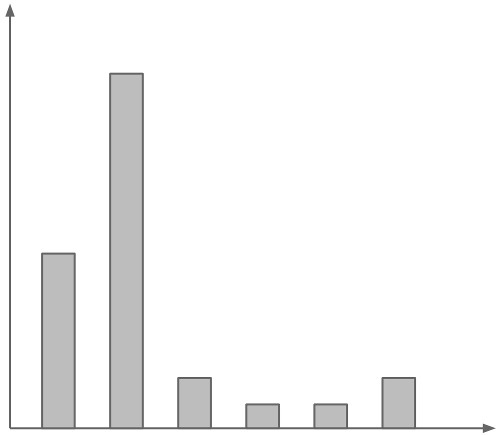
The PATE approach:

Private
Aggregation
of
Teacher
Ensembles

Teacher ensemble

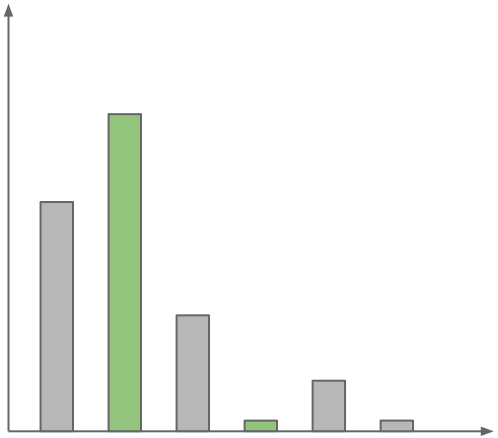


Aggregation



Count votes

$$n_j(\vec{x}) = |\{i : i \in 1..n, f_i(\vec{x}) = j\}|$$

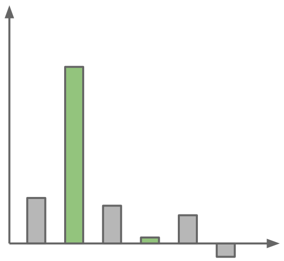


Take maximum

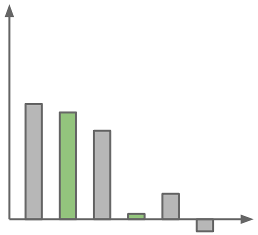
$$f(x) = \arg\max_j \left\{ n_j(\vec{x} \right\}$$

Intuitive privacy analysis

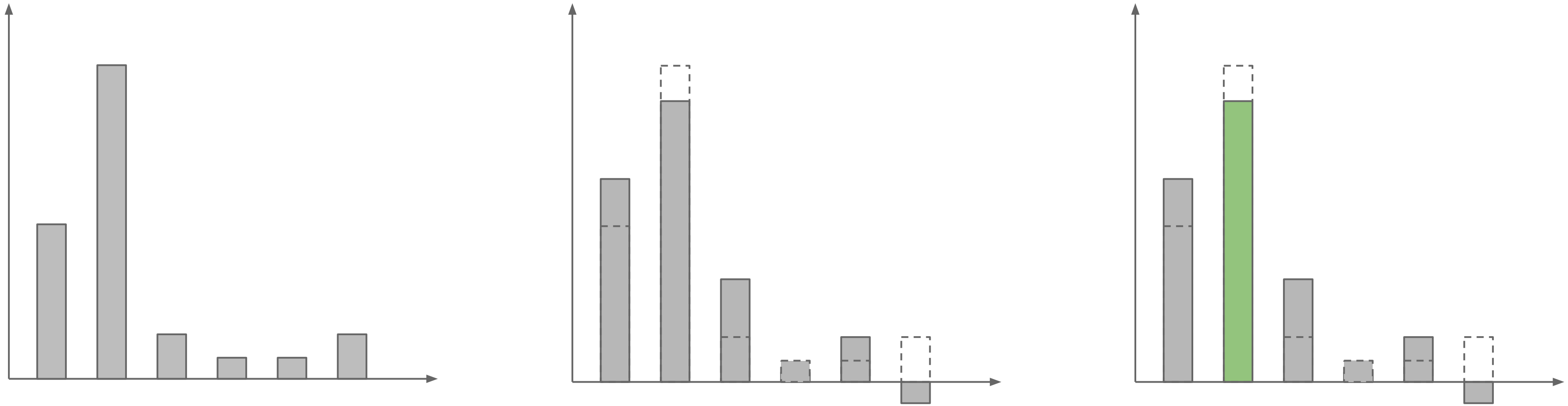
If most teachers agree on the label, it does not depend on specific partitions, so the privacy cost is small.



If two classes have close vote counts, the disagreement may reveal private information.



Noisy aggregation



Count votes

$$n_j(\vec{x}) = |\{i : i \in 1..n, f_i(\vec{x}) = j\}|$$

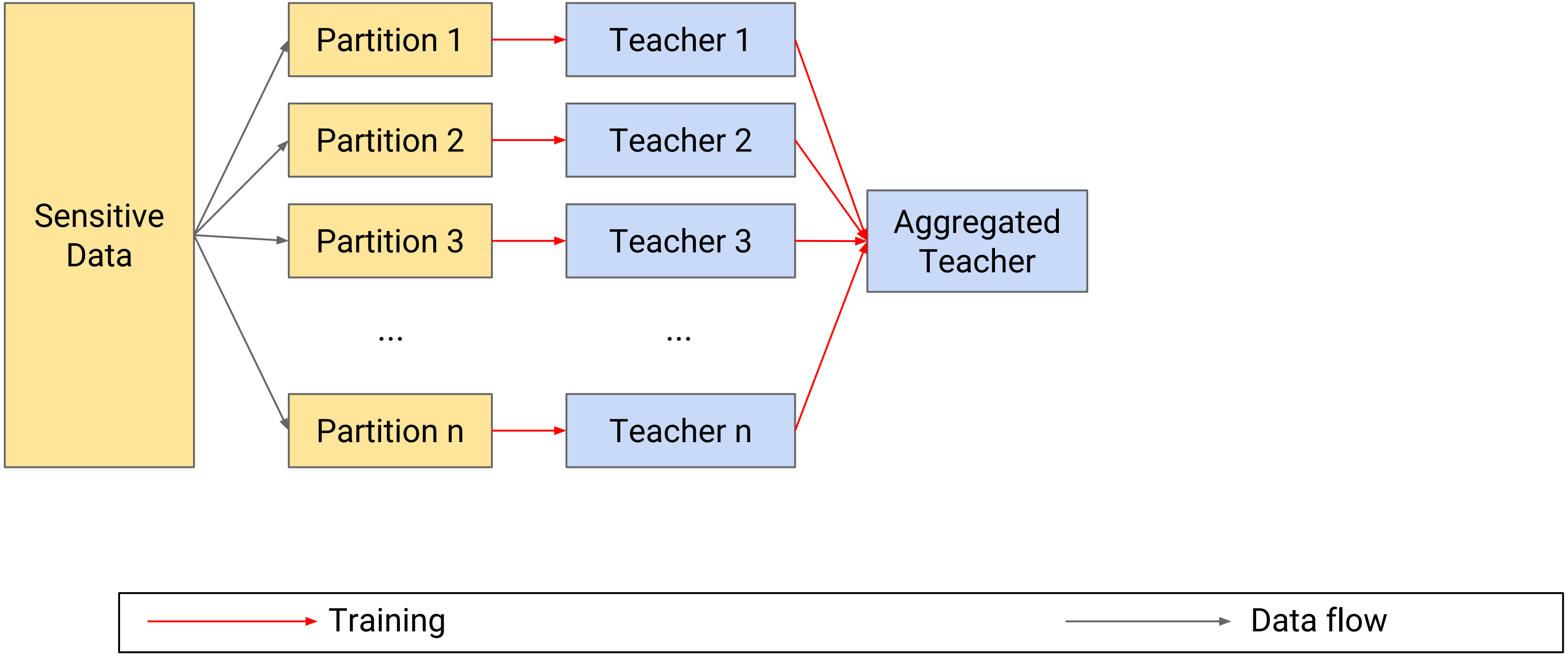
Add Laplacian noise

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

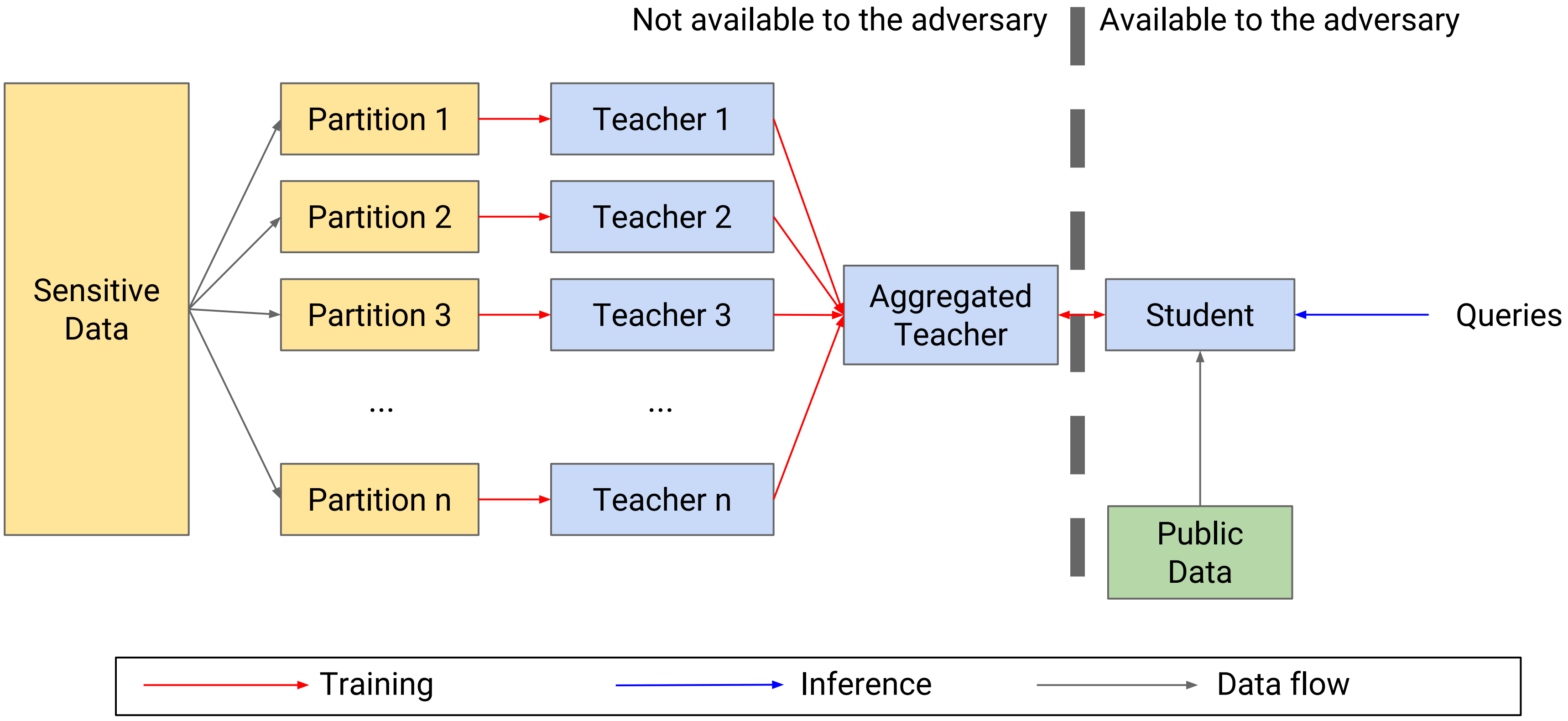
Take maximum

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

Teacher ensemble



Student training



Why train an additional "student" model?

The aggregated teacher violates our threat model:

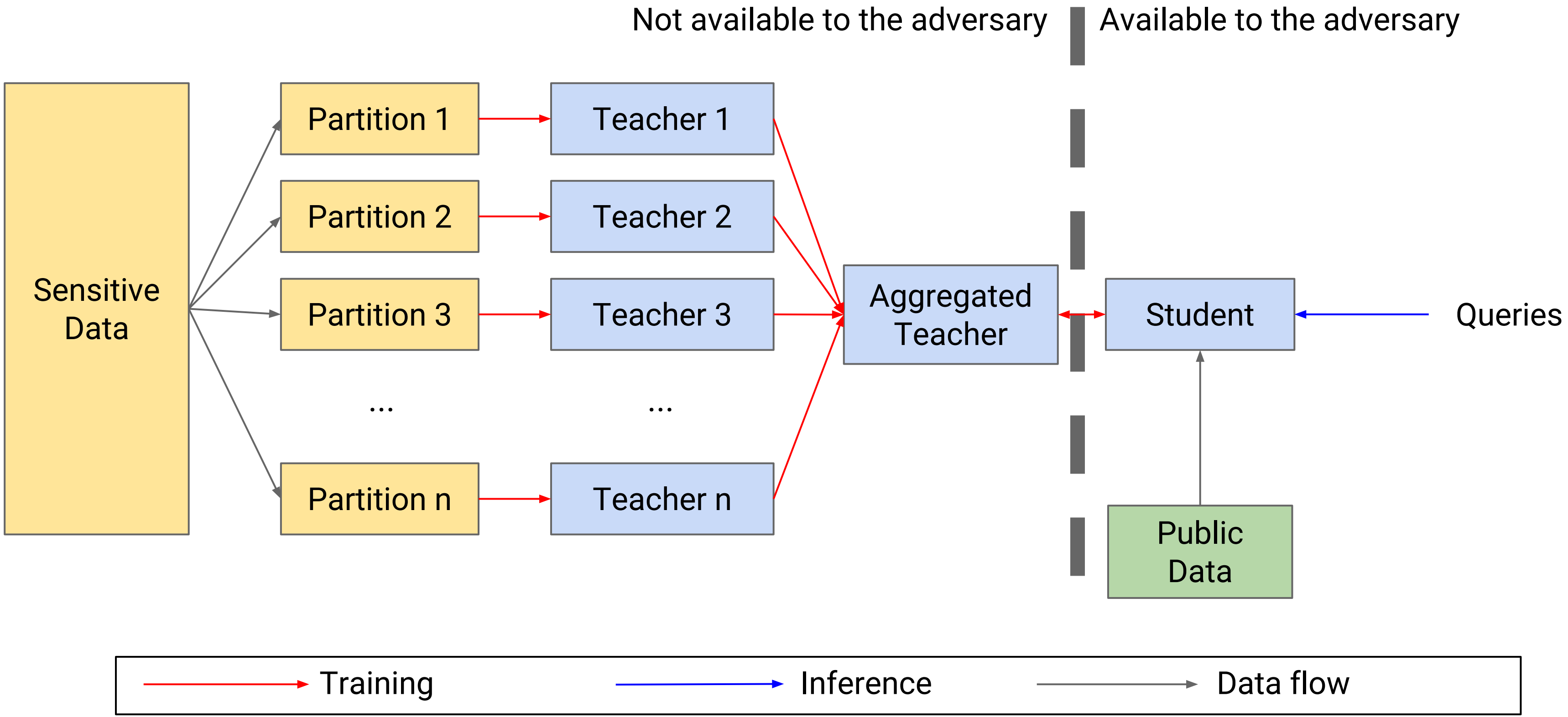
1. Each prediction increases total privacy loss.

   Privacy budgets create a tension between the accuracy and number of predictions.

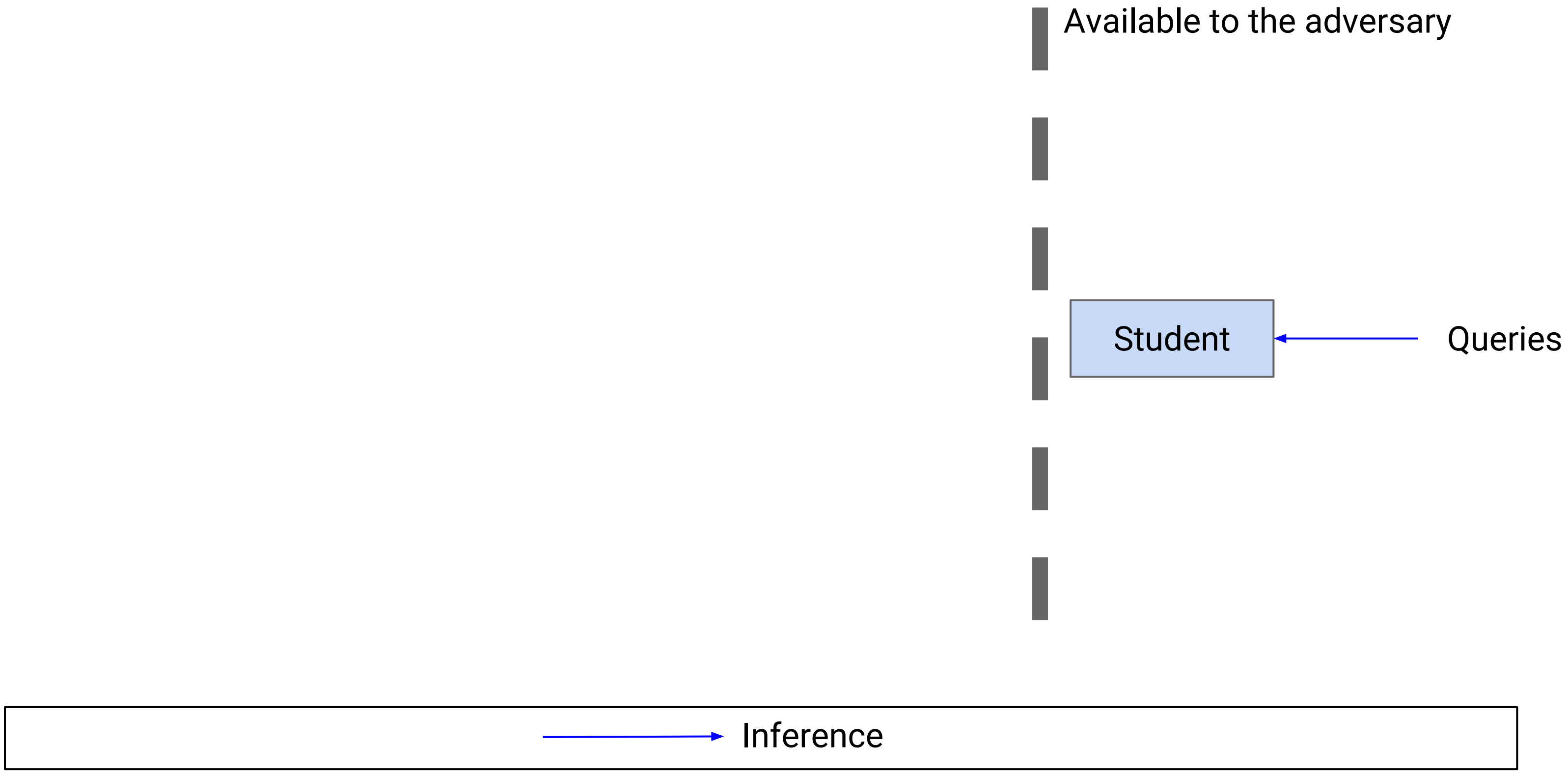
2. Inspection of internals may reveal private data.

   Privacy guarantees should hold in the face of white-box adversaries.

Student training



Deployment



Differential privacy analysis

Differential privacy:
A randomized algorithm M satisfies (ε,δ) differential privacy if for all pairs of neighbouring datasets (d,d'), for all subsets S of outputs:

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

Application of the **Moments Accountant** technique (Abadi et al, 2016)

Strong **quorum** $\Longrightarrow$ Small privacy cost

Bound is **data-dependent**: computed using the empirical quorum

PATE-G:
the generative variant of PATE

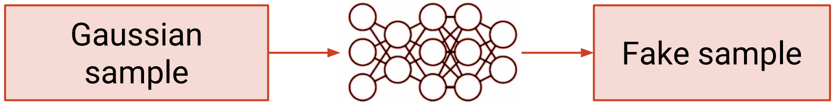
Generative Adversarial Networks (GANs)

2 **competing** models trying to game each other:

Generator:

Input: noise sampled from random distribution

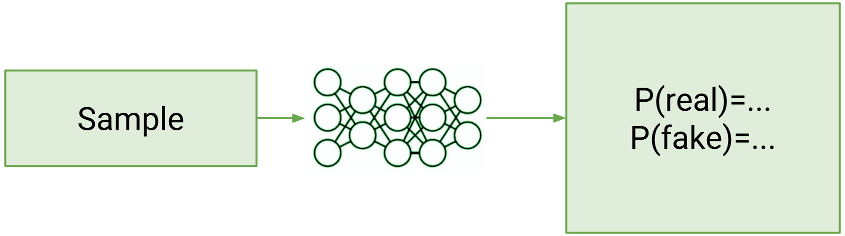
Output: synthetic input close to the expected training distribution



Discriminator

Input: output from generator OR example from real training distribution

Output: in distribution OR fake



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

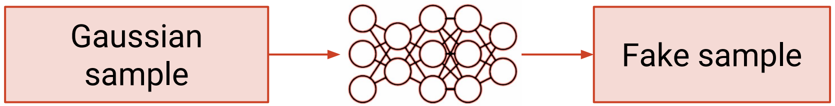
GANs for semi-supervised learning

2 **competing** models trying to game each other:

Generator:

Input: noise sampled from random distribution

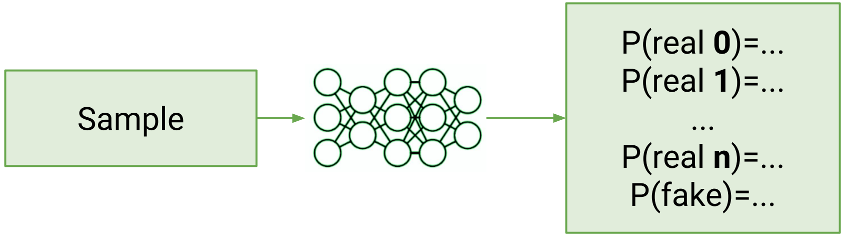
Output: synthetic input close to the expected training distribution



Discriminator

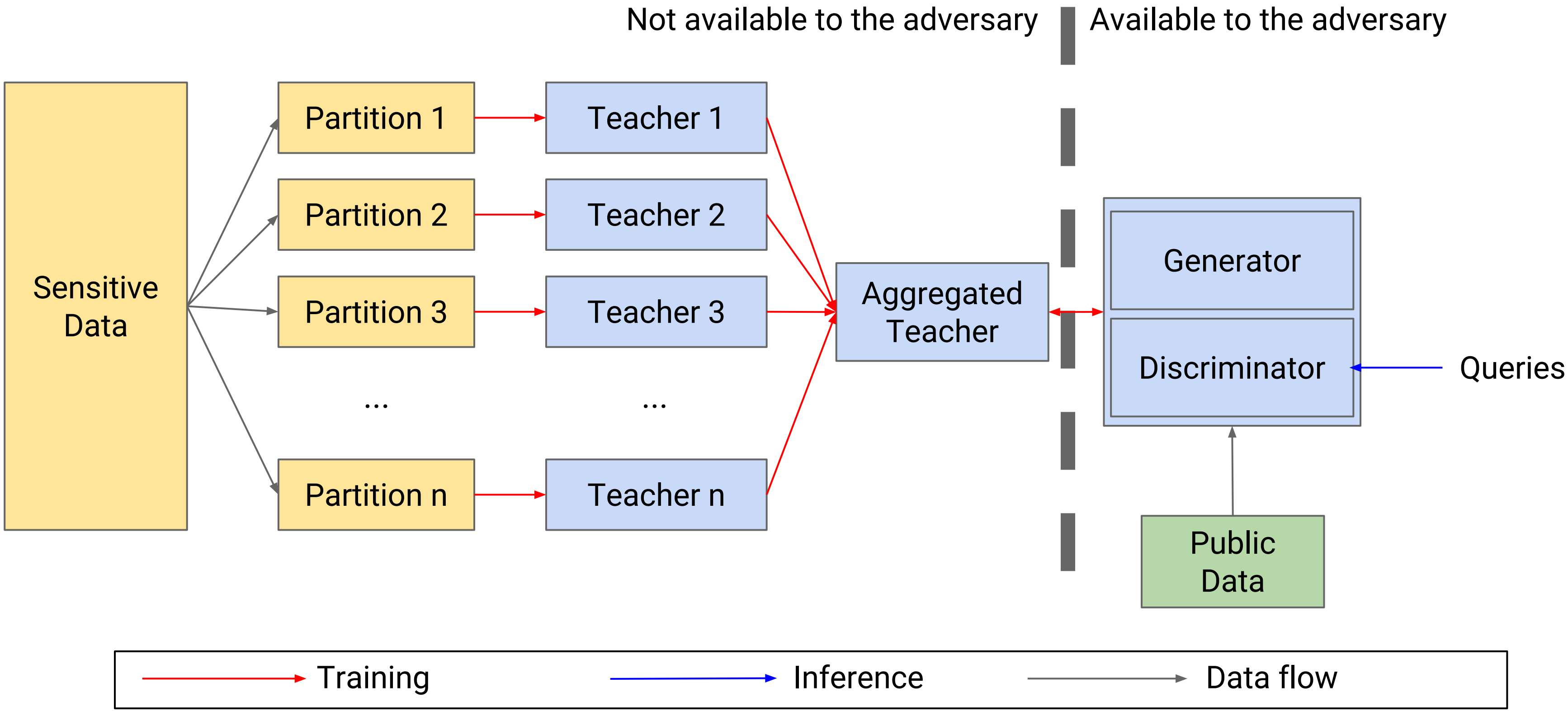
Input: output from generator OR example from real training distribution

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

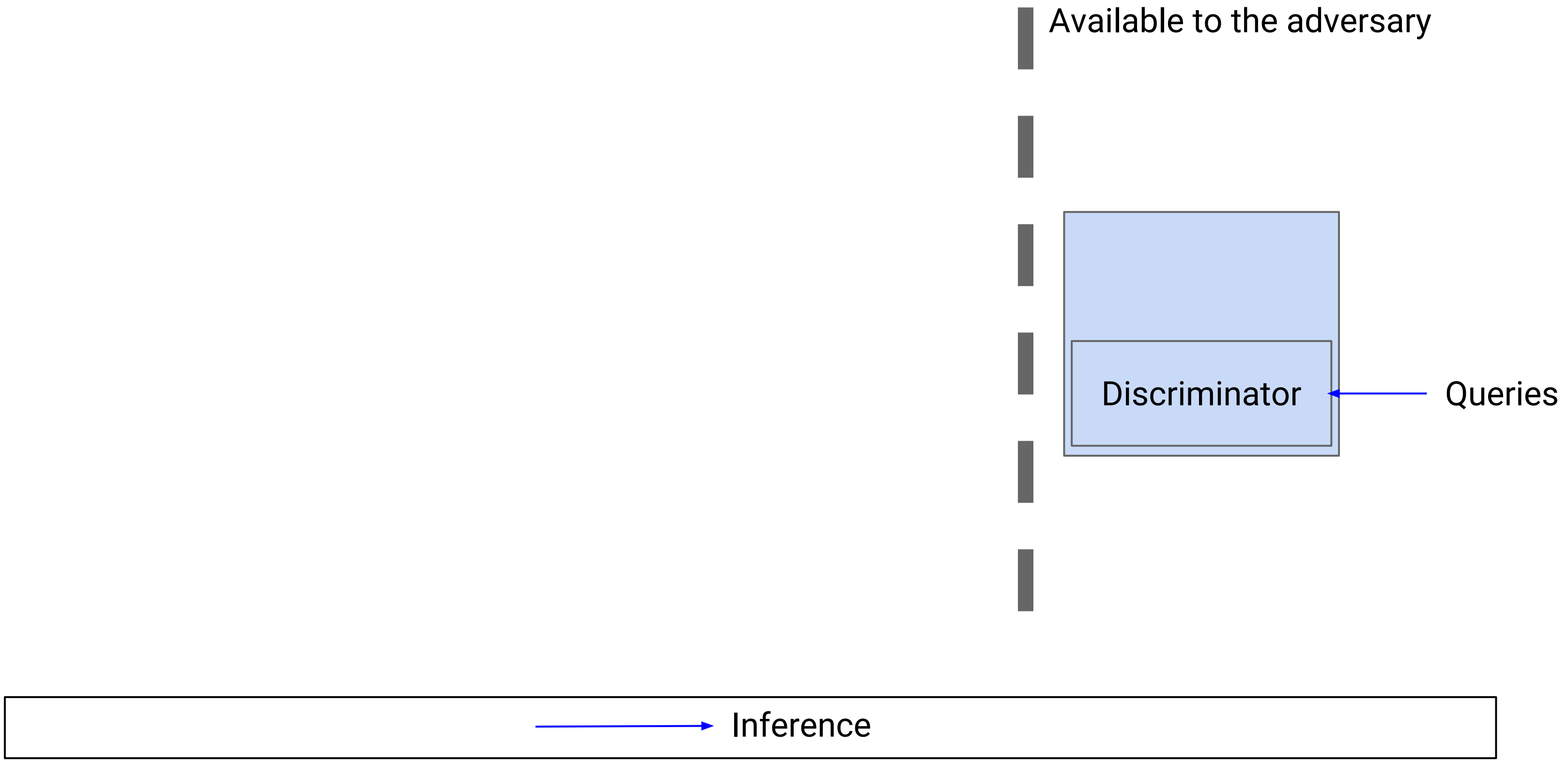


Salimans et al. (2016) *Improved techniques for training GANs*

Student training in PATE-G



Deployment of PATE-G



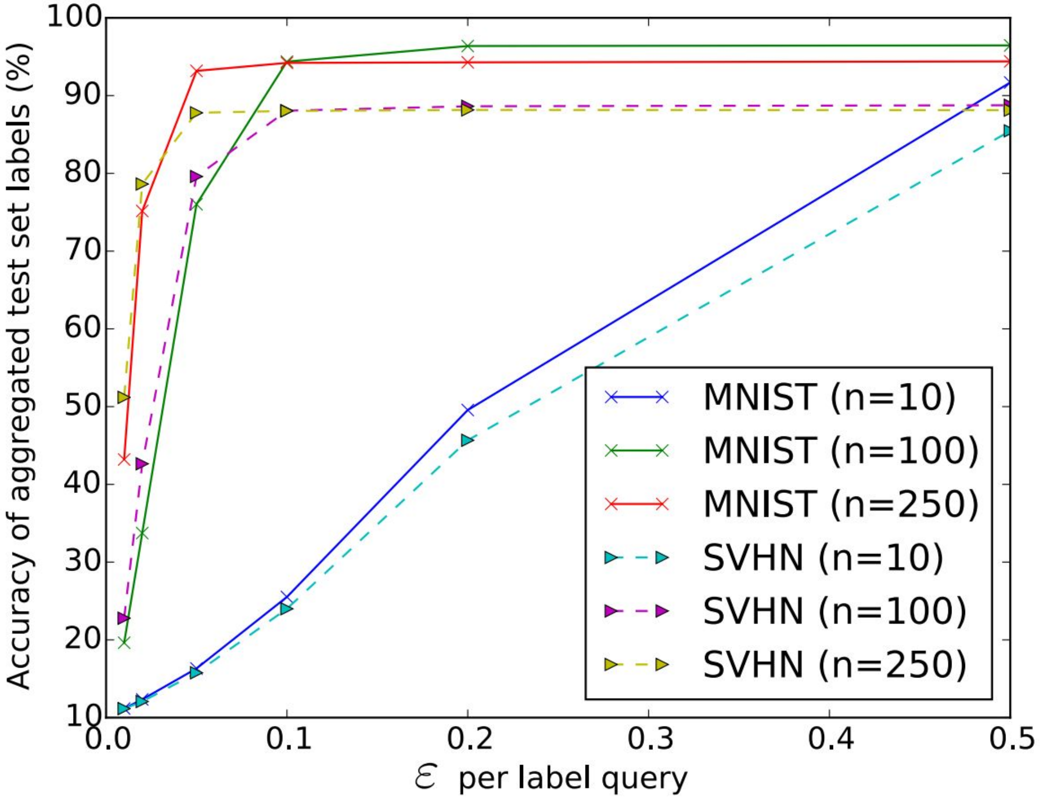
Experimental results

Experimental setup

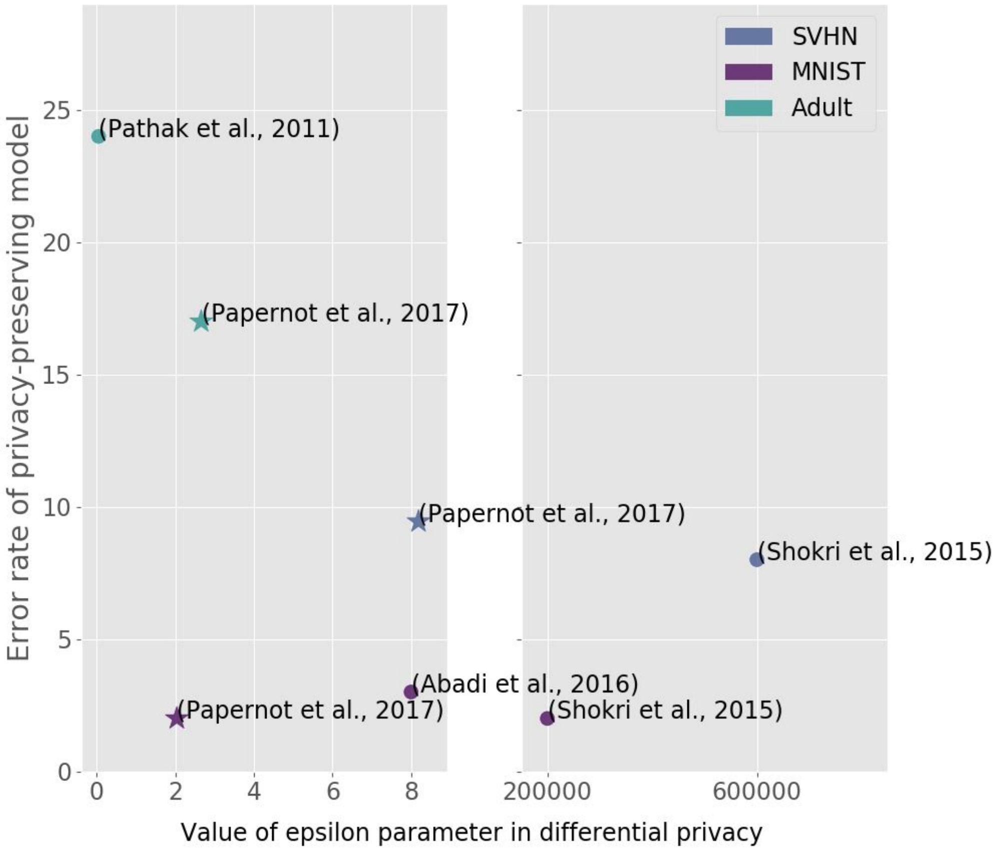
Dataset	Teacher Model	Student Model	Student Public Data	Testing Data
MNIST	2 conv + 1 relu	GANs (6 fc layers)	test[:1000]	test[1000:]
SVHN	2 conv + 2 relu	GANs (7 conv + 2 NIN)	test[:1000]	test[1000:]
UCI Adult	RF (100 trees)	RF (100 trees)	test[:500]	test[500:]
UCI Diabetes	RF (100 trees)	RF (100 trees)	test[:500]	test[500:]

/TensorFlow™/**models**/tree/master/differential_privacy/multiple_teachers

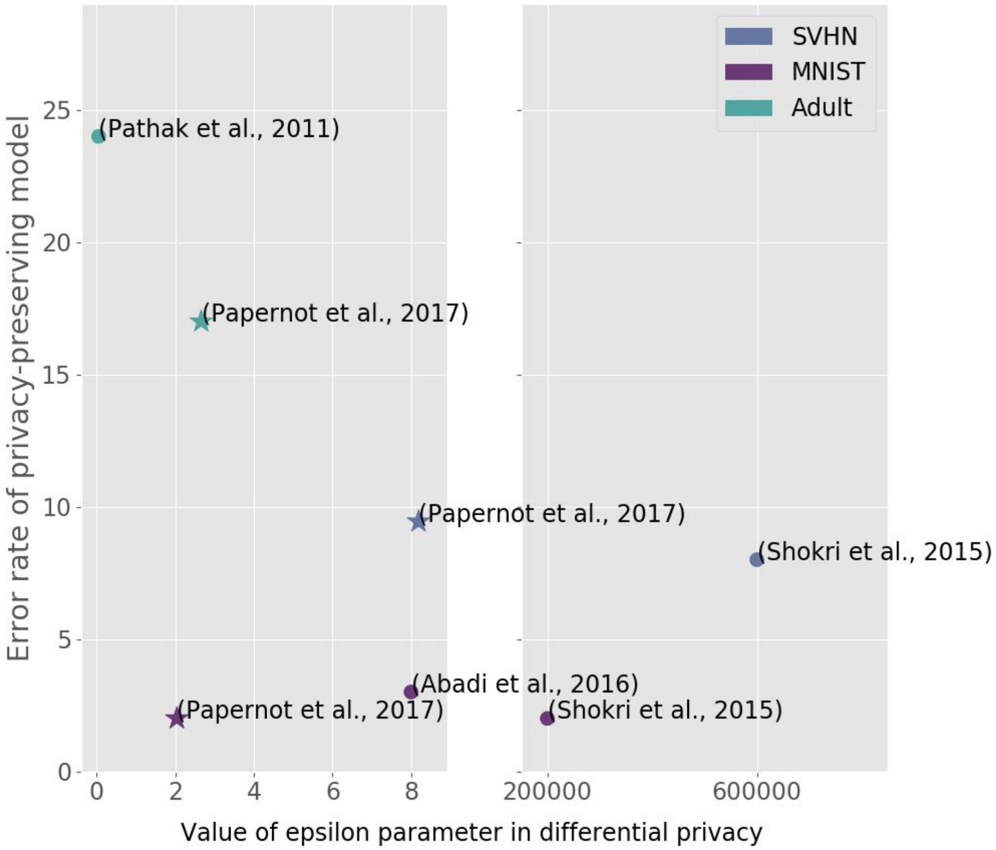
Aggregated teacher accuracy



Trade-off between student accuracy and privacy



Trade-off between student accuracy and privacy



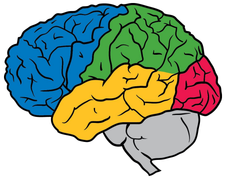
UCI Diabetes	
ε	1.44
δ	10^{-5}
Non-private baseline	93.81%
Student accuracy	93.94%

nicolas@papernot.fr

www.cleverhans.io

@NicolasPapernot

?



Come check out our poster C13
(PATE-G *swag* will be distributed
while supplies last)