Privacy Leakage of Machine Learning Models

Esfandiar Mohammadi

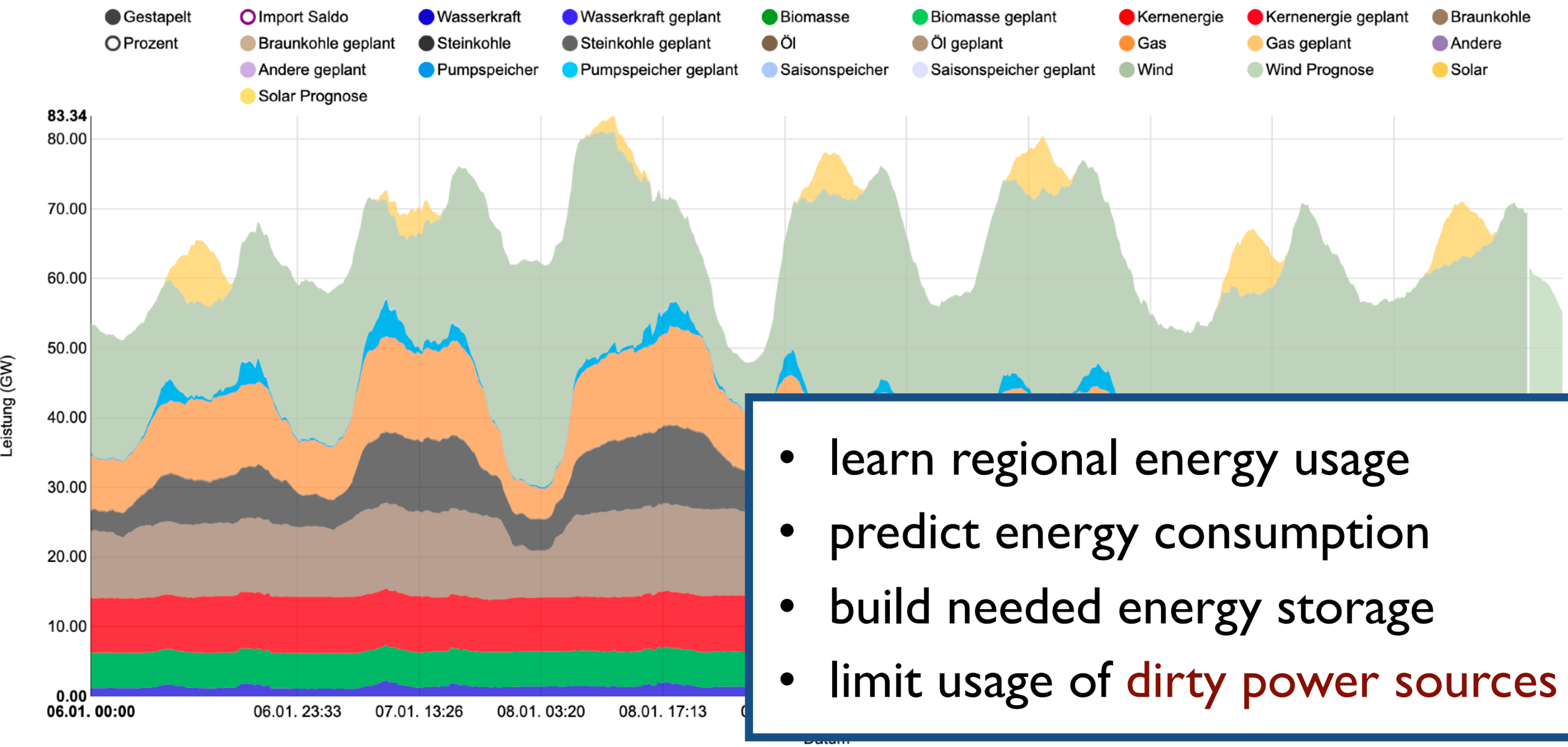
UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR IT-SICHERHEIT

Why Private Learning?

- Open data: everybody shares everything
  - huge privacy concerns

    but useful applications

Predict Regional Energy Usage



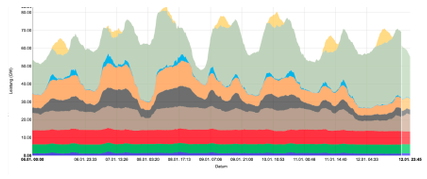
- learn regional energy usage
- predict energy consumption
- build needed energy storage
- limit usage of dirty power sources

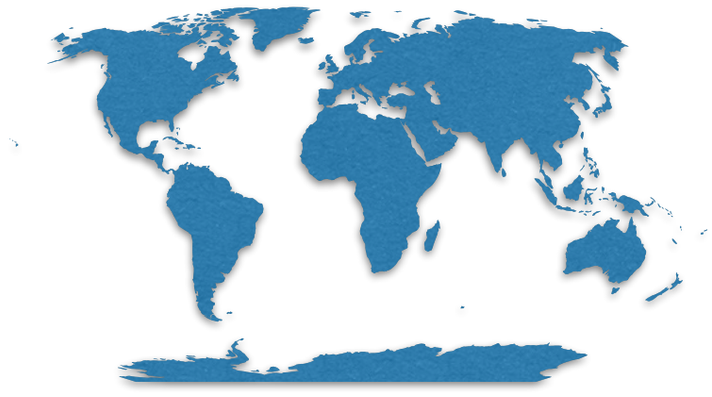
source: https://www.energy-charts.de/power_de.htm

Why Private Learning?

- Open data: everybody shares all its data
  - huge privacy concerns

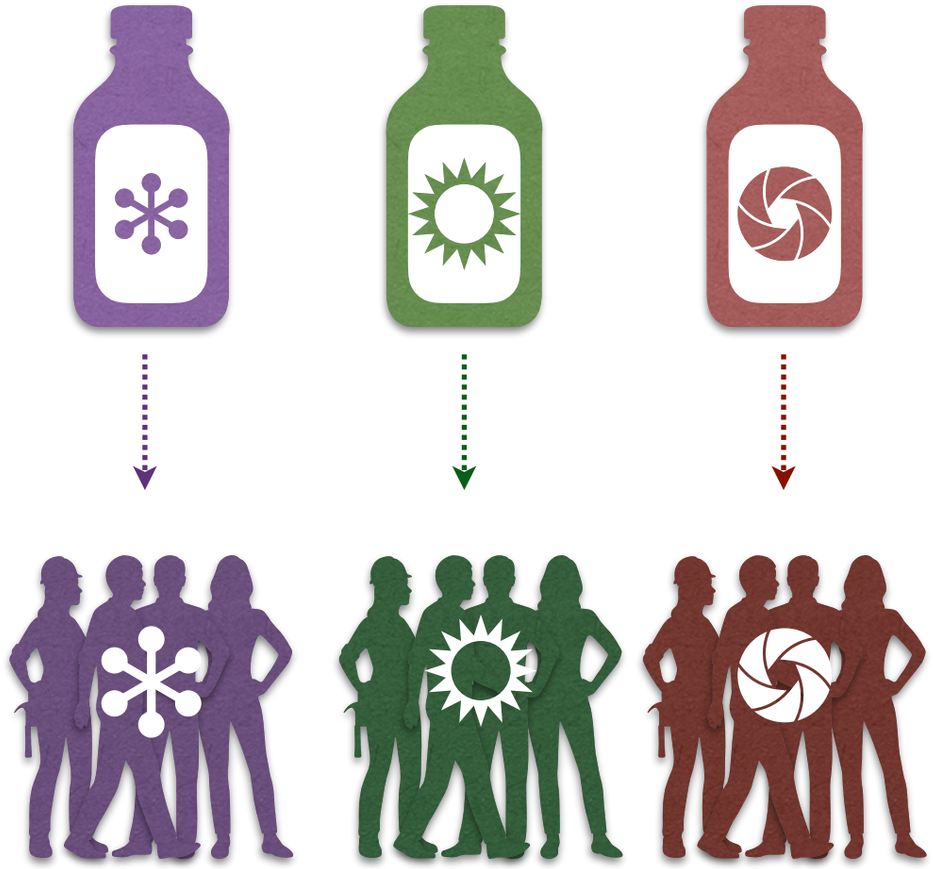
    but useful applications
  - smart grid





Personalized Medicine

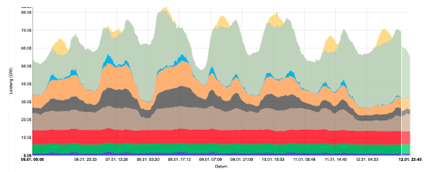
- Learn rare cases
- Interaction between genetic markers and pharmaceuticals
- Symptoms of combinations of conditions
- Combinations of pharmaceuticals



Why Private Learning?

- Open data: everybody shares all its data
  - huge privacy concerns

    but useful applications
  - smart grid



  - personalized medicine

Smart Assistants

- Context-aware personal assistants
  - context-aware health-recommendations
  - context-aware reminders
  - context-aware search assistant
- Information could stay on device
  - training needs a lot of data
  - local training unrealistic
  - too little data
  - use combined data of all users
    $\Longrightarrow$ protect training data

Apple's Siri

Cortana

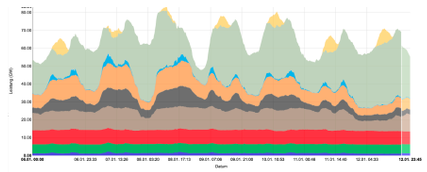
amazon alexa

Google Assistant

Why Private Learning?

- Open data: everybody shares all its data
  - huge privacy concerns

    but useful applications
  - smart grid

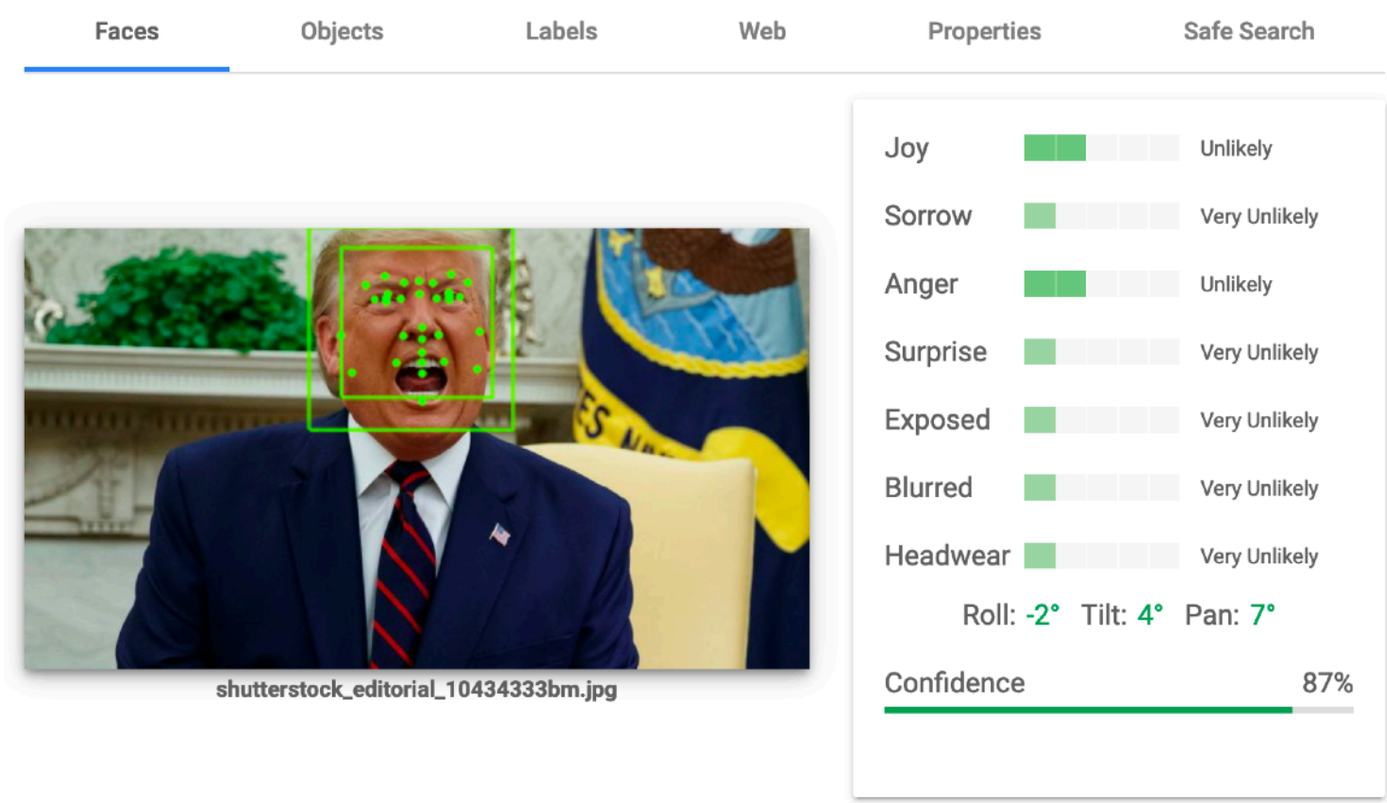


  - personalized medicine

  - smart assistants

Selling Models on Customer Data

- Lucrative business:
  - train specialized models on user data
  - sell or give access to model
  - does user data leak?



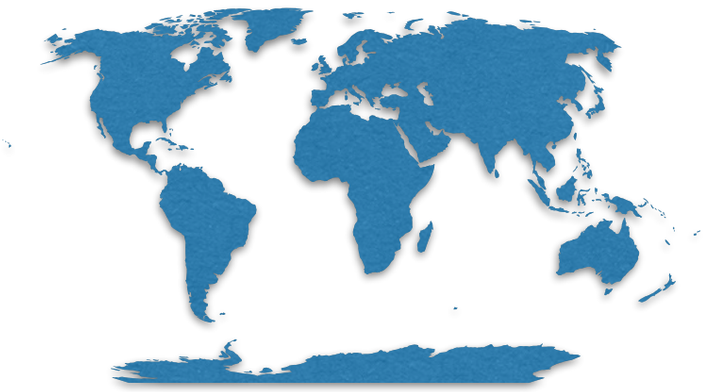
Google Vision API

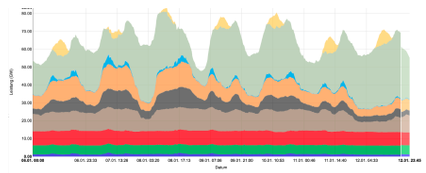
Why Private Learning?

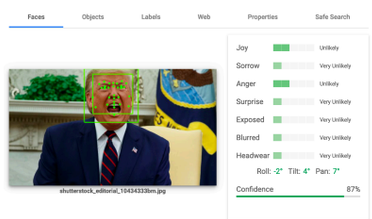
- Open data: everybody shares everything
  - huge privacy concerns

    but useful applications
  - smart grid
  - personalized medicine
  - smart assistants
- Provide access to model

  trained on user-data

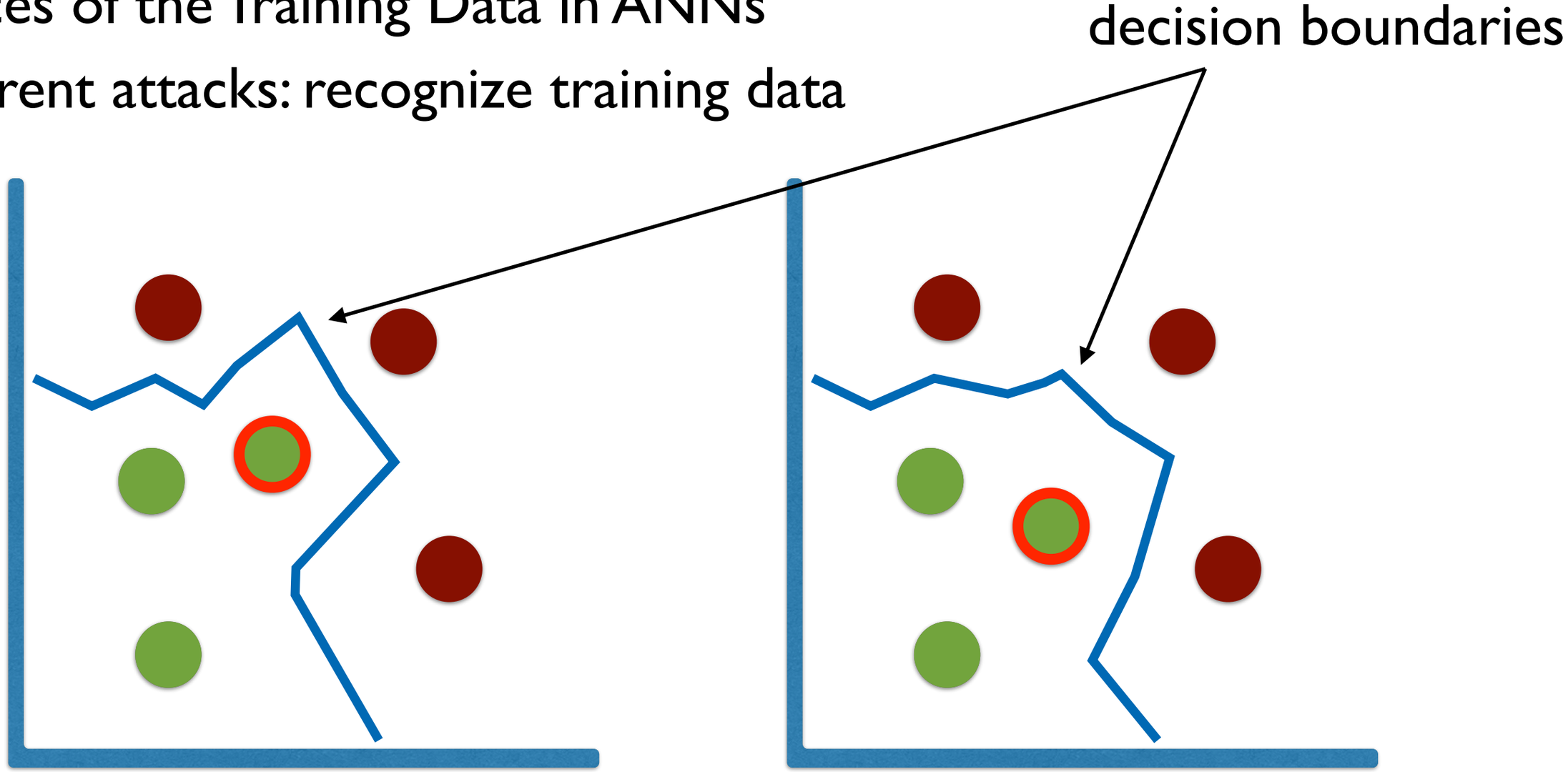






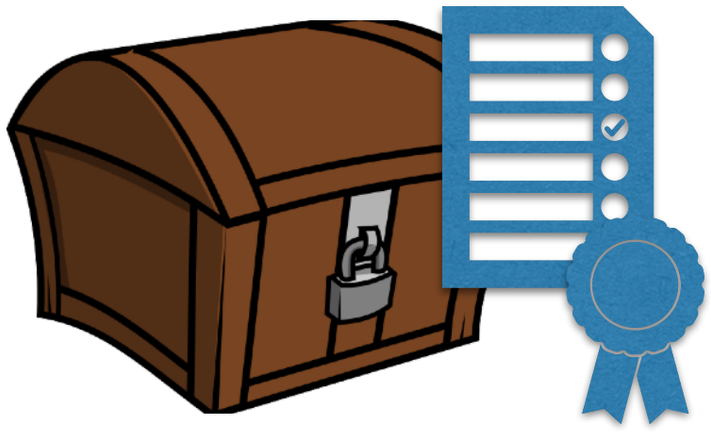
Why Privacy Concerns?

- Traces of the Training Data in ANNs
- Current attacks: recognize training data



(Disclaimer: example completely made up for illustration purposes)

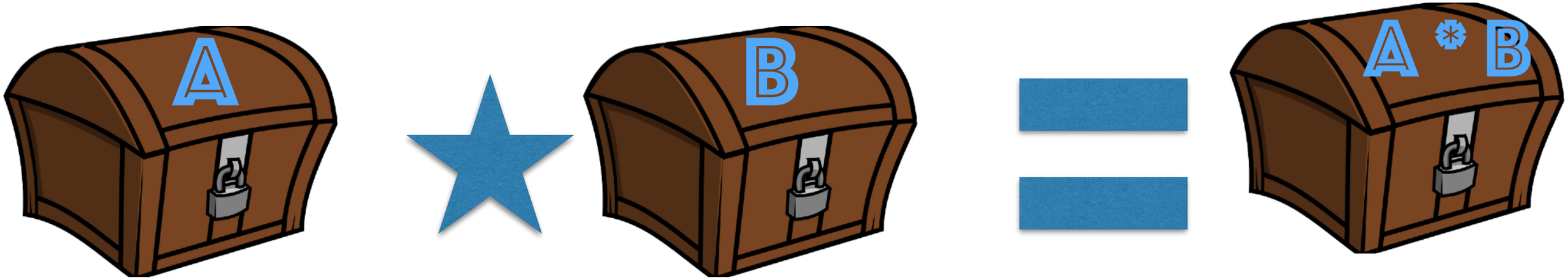
Can Powerful Cryptographic Tools Help?



Zero-Knowledge Proof

- Unforgeable proof about hidden secrets
- Computing on secrets without revealing them (same for SMPC)

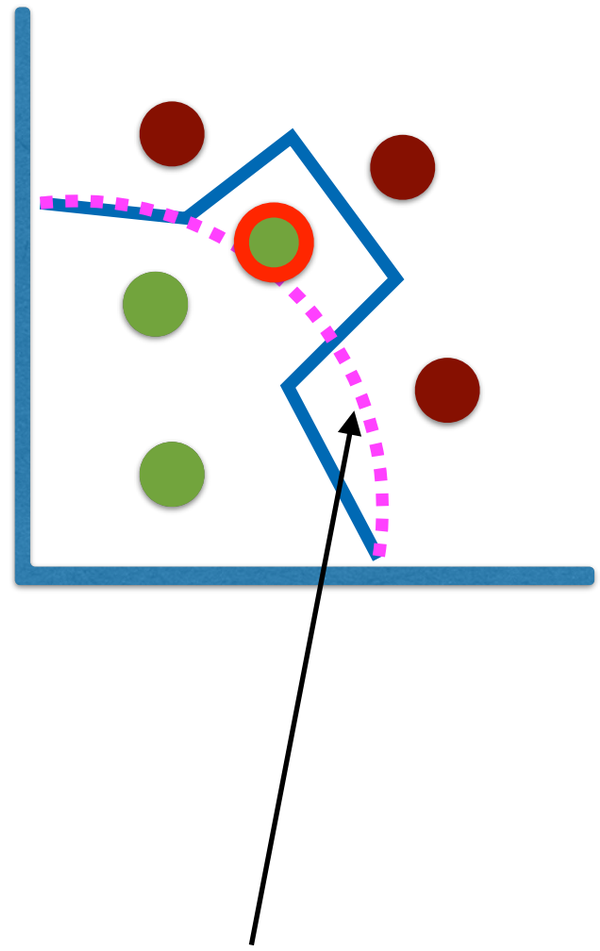
↯ Adversary needs to access the secret (the model)



Homomorphic Encryption

Private Learning

- Give adversary access to the model
- Protect all training data
  - impossible
- Blur the influence of any single element (only learn trends)
- Related (not a topic for today):
learn privacy-preserving version of a given function / protect the inputs



smoothed
decision boundaries

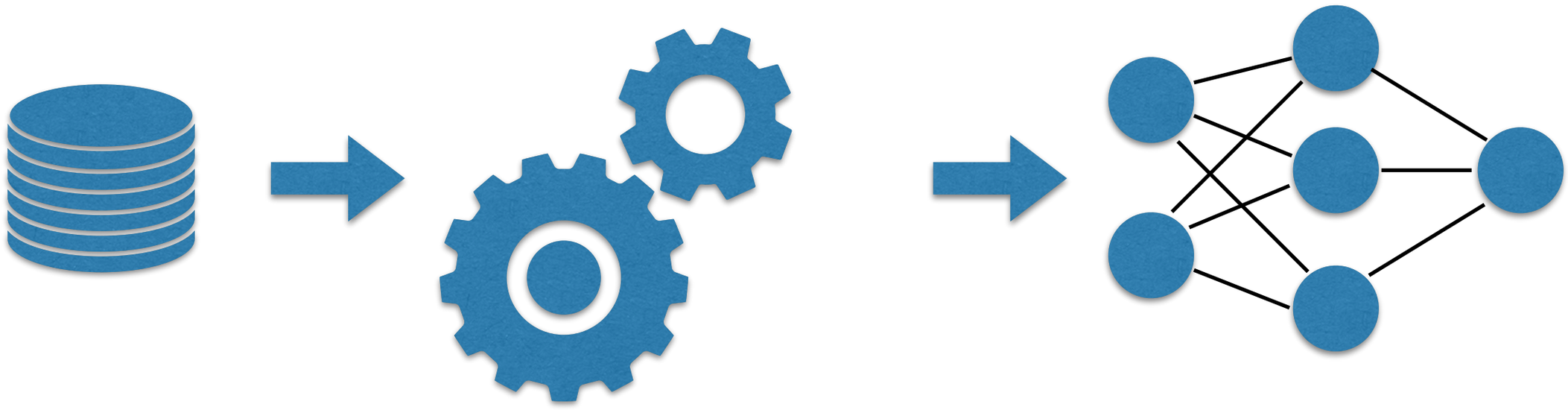
Outline

- Traces of training data in ANNs
- How to formulate privacy?
- Private learning
- Other learning techniques

Outline

- Traces of training data in ANNs
- How to formulate privacy?
- Private learning
- Other learning techniques

Supervised Machine Learning



training data

$(x_i, f(x_i))_{i=1}^{k}$

training

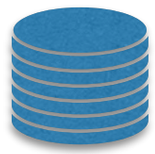
model

$\hat{f}$

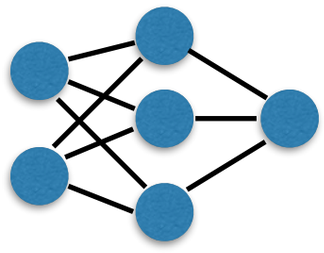
approximation of f

Goal: Approximate f

- training data $(x_i, f(x_i))_{i=1}^{k}$



- $\hat{f}$ approximates f : $\hat{f}(x_i') \sim f(x_i')$
  - for unseen data x_i' (not training data)

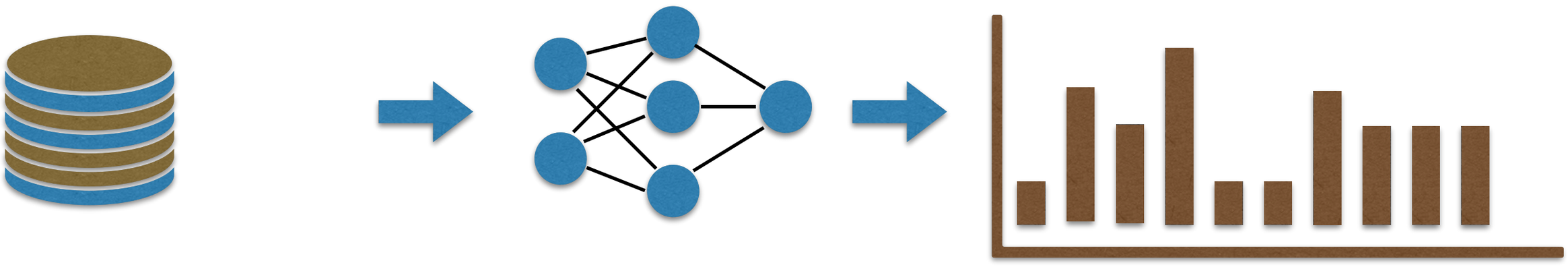


Problem:
Model $\hat{f}$ learns more than f

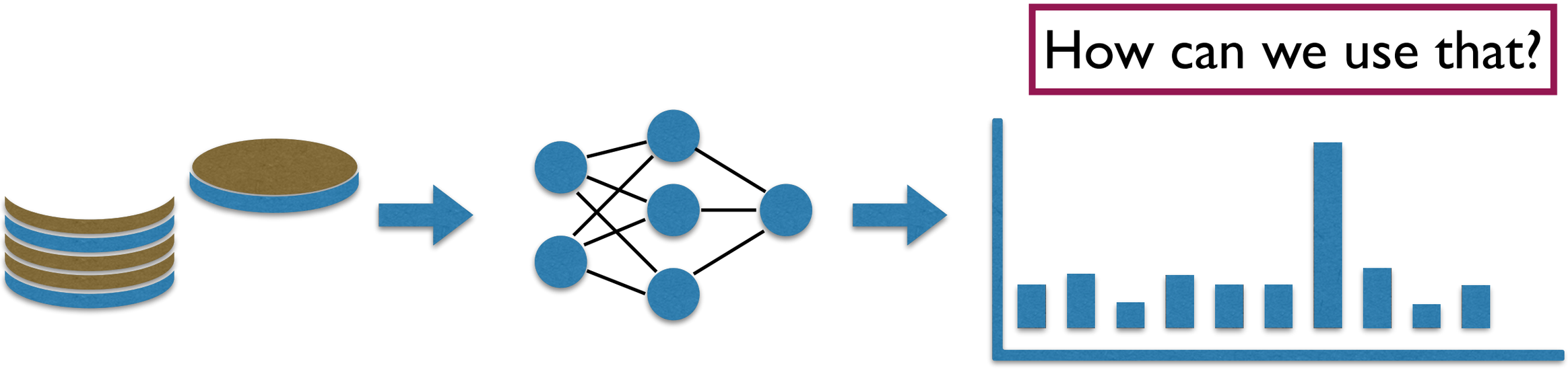
- Popular functions (for some domain X)
  - classification $f : X \to \{0,1\}^k$
  - prediction $f : X^* \to X$
  - regression $f : X \to \mathbb{R}$

likelihood vector:
for each class one weight/probability
(highest weight
→ predicted class)

Insight



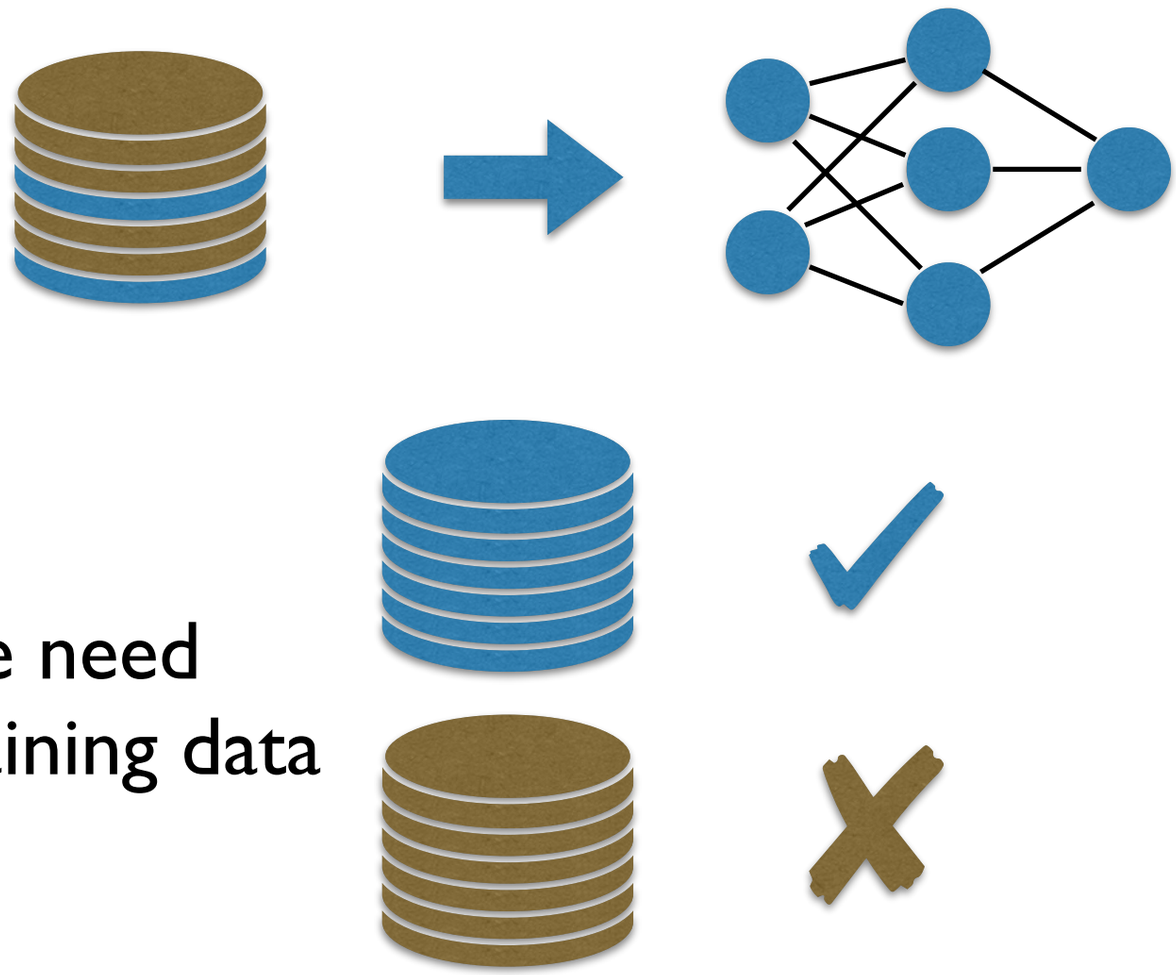
Insight



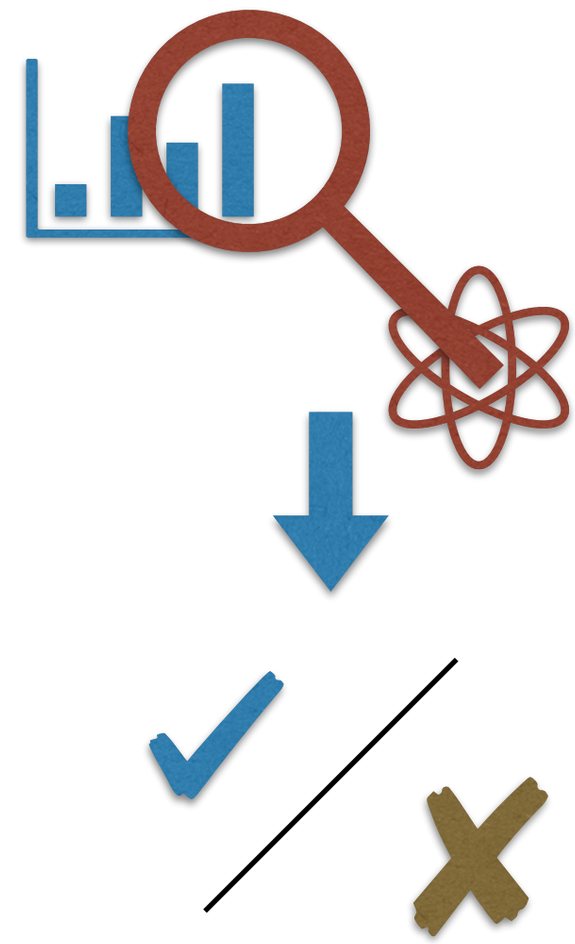
bars more pronounced for trainings data

Distingusher: A Binary Classifier

From where do we get the real training data?



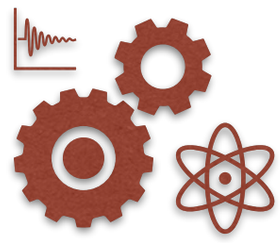
We need the training data

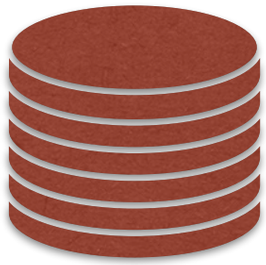


How to Use the Related Data?

Assume, we have this

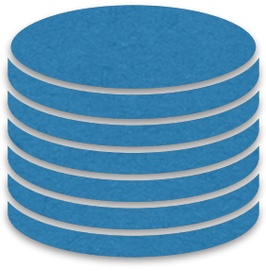
How to use this instead of this?





related labelled data

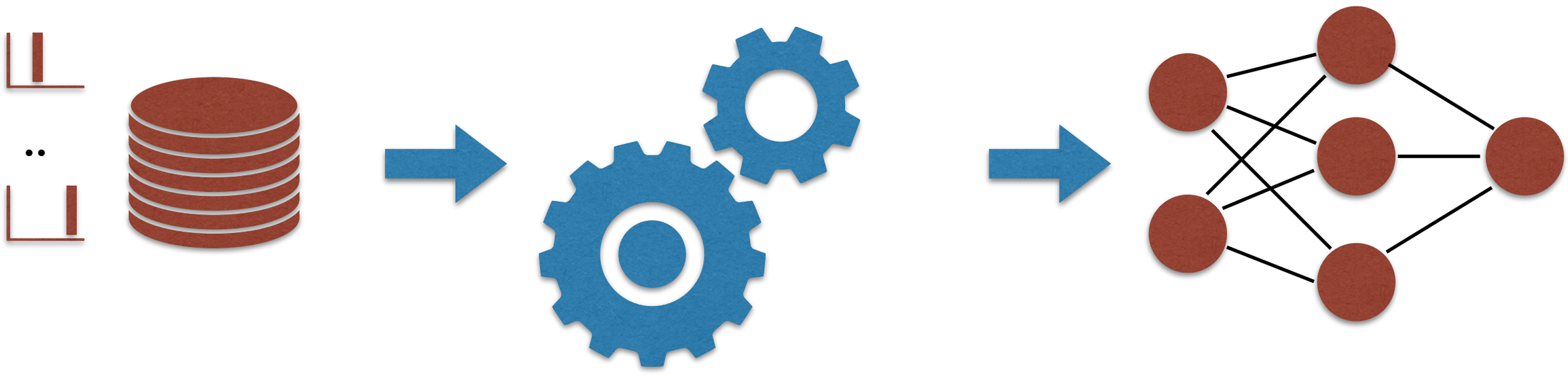
$$(x'_i, f(x'_i))_{i=1}^{k'}$$



(target) training data

$$(x_i, f(x_i))_{i=1}^{k}$$

Idea: Train Your Own Shadow Models



related labelled data

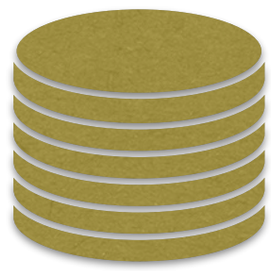
$$(x_i', f(x_i'))_{i=1}^{k'}$$

training

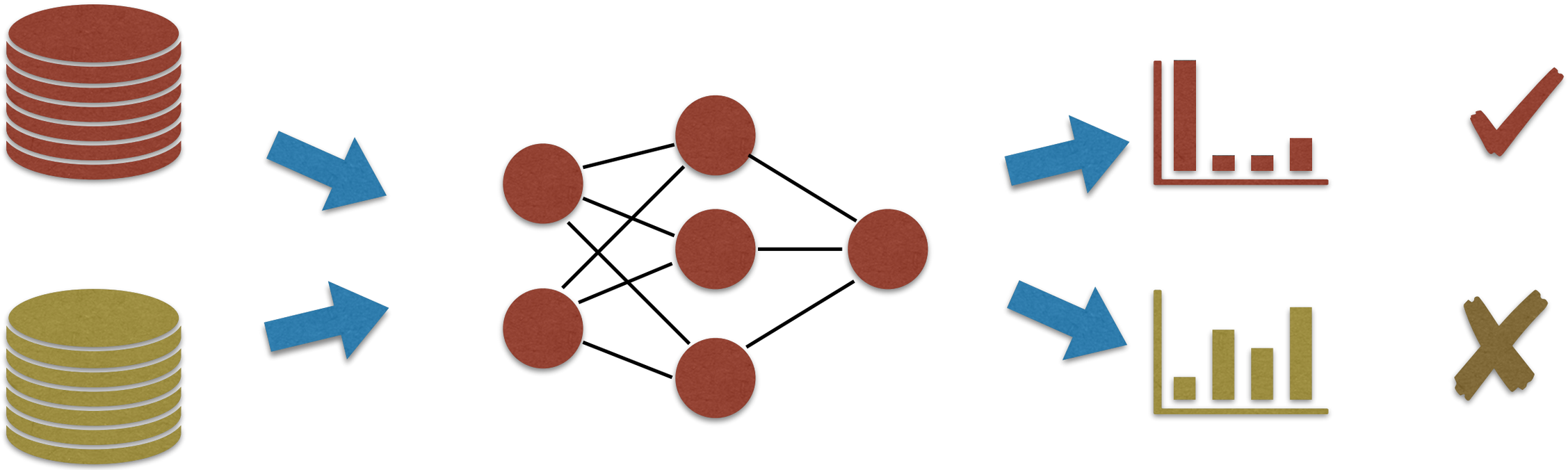
shadow model for

f

and keep a hold-out set
that is not used in training



Construct Likelihood Vectors



related labelled data

$(x'_i, f(x'_i)))_{i=1}^{k'}$

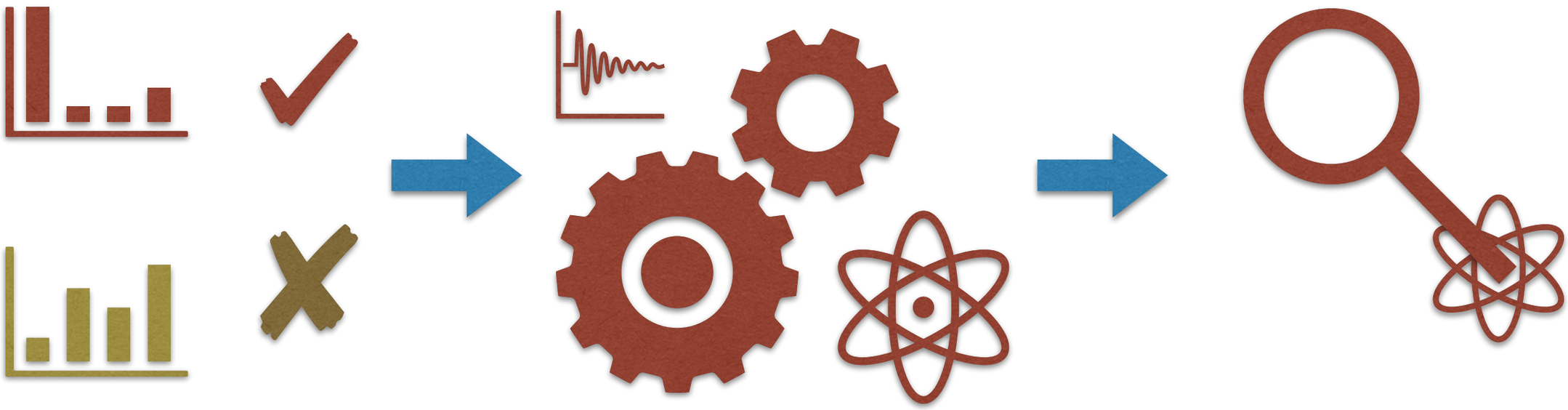
shadow model for

f

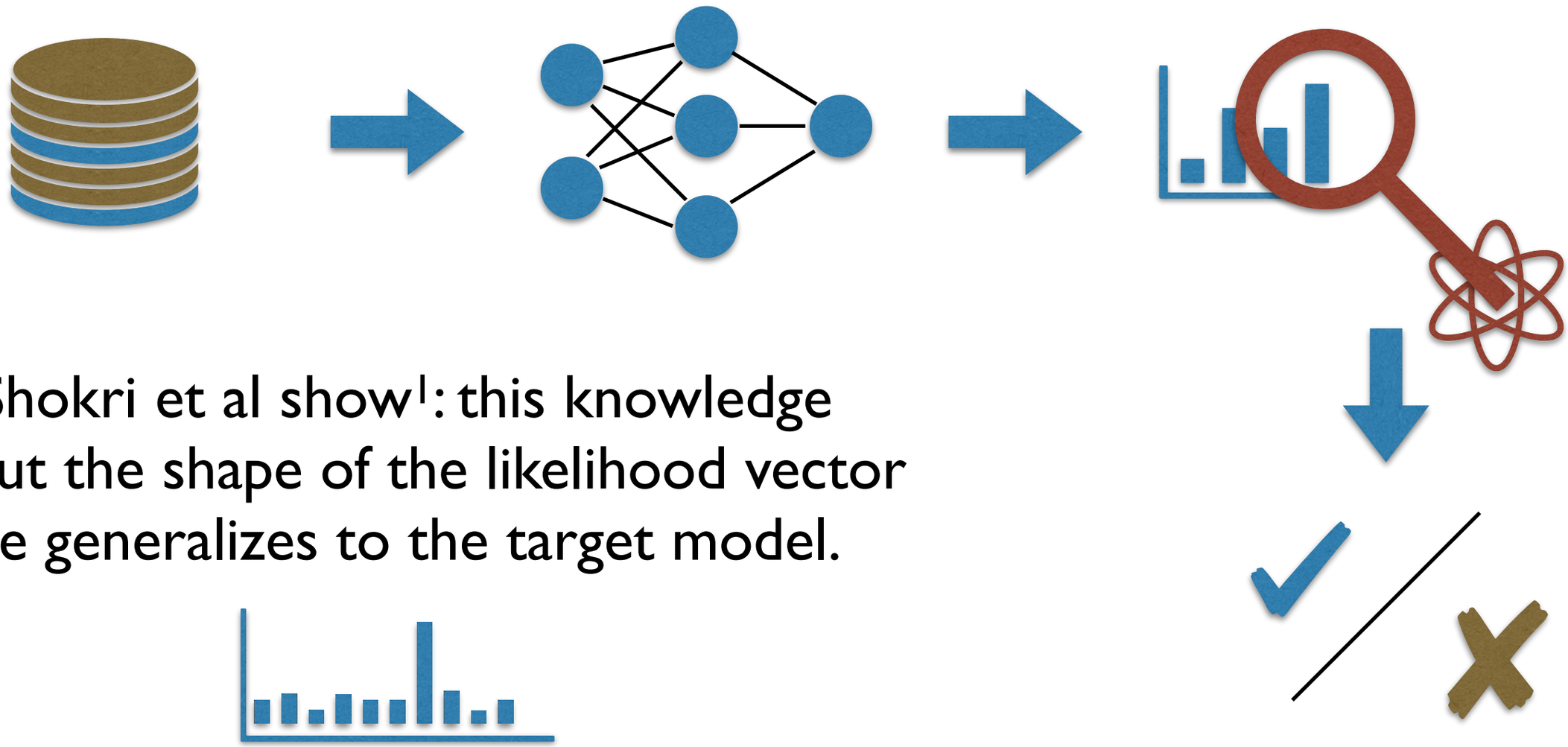
likelihood vectors

labels

Train a Binary Classifier



Apply the Binary Classifier



Shokri et al show[1]: this knowledge about the shape of the likelihood vector be generalizes to the target model.

[1]Membership Inference Attacks against Machine Learning Models, by Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. In IEEE S&P, 2017. available under https://arxiv.org/abs/1610.05820

Outline

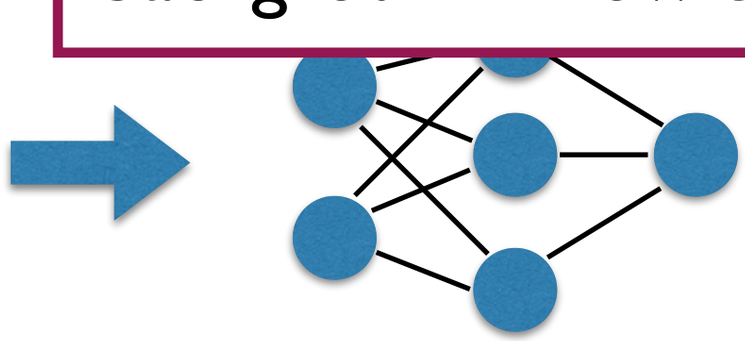
- Traces of training data in ANNs
- How to formulate privacy?
- Private learning
- Other learning techniques

Data Sanitization is Industry-Standard

- Sanitize the original data set
  - e.g., remove identifiers, keep age and address in a range
  - industry-standard
- Train with the sanitized data set

data sanitization

Teaser: Leakage against attackers with background knowledge



K-Anonymity (Definition)

- A dataset satisfies *K-Anonymity for attribute* $X_1, .., X_n$ if for each row, the **value combination** of attributes $X_1, .., X_n$ is contained in at least K-1 other rows.
- A dataset satisifies *K-Anonymity*, for a set of *quasi-identifying attributes* $X_1, .., X_n$, if for each row, the **value combination** of attributes $X_1, .., X_n$ is contained in at least K-1 other rows.
  - *Quasi-identifying attributes* $X_1, .., X_n$: Attributes that could identify a person (first name, age, state of residence, etc.) and could be publicly available.

Achieving K-Anonymity

Approach: Reduce the information of the quasi-identifiers.

Name	Age	Gender	Semester	Grade	Minor
Alice	19	Female	1	1.3	Math
Bob	18	Male	1	2.0	Literature
Charlie	18	Male	1	1.7	Philosophy
Dave	18	Male	1	3.7	CS
Eve	17	Female	1	1.0	CS
Fritz	19	Male	3	1.3	History
Gerd	21	Male	3	2.3	Math
Hans	23	Male	3	3.0	CS
Isa	20	Female	3	failed	CS
John	20	Male	3	1.7	Literature
Petra	22	Female	3	1.0	Physics
Ole	22	Male	5	3.3	Math
Kale	21	Male	5	1.7	CS
Leonard	23	Male	5	failed	History
Martin	20	Male	5	2.7	Literature
Nils	22	Male	5	3.0	Math
Otto	20	Male	5	1.0	Physics

Suppression (Name and Gender):

Name	Age	Gender	Semester	Grade	Minor
*	19	*	1	1.3	Math
*	18	*	1	2.0	Literature
*	18	*	1	1.7	Philosophy
*	18	*	1	3.7	CS
*	17	*	1	1.0	CS
*	19	*	3	1.3	History
*	21	*	3	2.3	Math
*	23	*	3	3.0	CS
*	20	*	3	failed	CS
*	20	*	3	1.7	Literature
*	22	*	3	1.0	Physics
*	22	*	5	3.3	Math
*	21	*	5	1.7	CS
*	23	*	5	failed	History
*	20	*	5	2.7	Literature
*	22	*	5	3.0	Math
*	20	*	5	1.0	Physics

Achieving K-Anonymity

Approach: Reduce the information of the quasi-identifiers.

Name	Age	Gender	Semester	Grade	Minor
Alice	19	Female	1	1.3	Math
Bob	18	Male	1	2.0	Literature
Charlie	18	Male	1	1.7	Philosophy
Dave	18	Male	1	3.7	CS
Eve	17	Female	1	1.0	CS
Fritz	19	Male	3	1.3	History
Gerd	21	Male	3	2.3	Math
Hans	23	Male	3	3.0	CS
Isa	20	Female	3	failed	CS
John	20	Male	3	1.7	Literature
Petra	22	Female	3	1.0	Physics
Ole	22	Male	5	3.3	Math
Kale	21	Male	5	1.7	CS
Leonard	23	Male	5	failed	History
Martin	20	Male	5	2.7	Literature
Nils	22	Male	5	3.0	Math
Otto	20	Male	5	1.0	Physics

Generalization (Age):

Name	Age	Gender	Semester	Grade	Minor
*	17-20	*	1	1.3	Math
*	17-20	*	1	2.0	Literature
*	17-20	*	1	1.7	Philosophy
*	17-20	*	1	3.7	CS
*	17-20	*	1	1.0	CS
*	17-20	*	3	1.3	History
*	21-25	*	3	2.3	Math
*	21-25	*	3	3.0	CS
*	17-20	*	3	failed	CS
*	17-20	*	3	1.7	Literature
*	21-25	*	3	1.0	Physics
*	21-25	*	5	3.3	Math
*	21-25	*	5	1.7	CS
*	21-25	*	5	failed	History
*	17-20	*	5	2.7	Literature
*	21-25	*	5	3.0	Math
*	17-20	*	5	1.0	Physics

Result: K-Anonymity

K-Anonymity for a list of students with K=3. For all quasi-identifying attributes (Name, Gender & Age) there are at least 3 rows with the same value.

Idea/Goal:

Consequently, one cannot be identified, but hides in a group of K=3 people.

.. right?

Name	Age	Gender	Semester	Grade	Minor
*	17-20	*	1	1.3	Math
*	17-20	*	1	2.0	Literature
*	17-20	*	1	1.7	Philosophy
*	17-20	*	1	3.7	CS
*	17-20	*	1	1.0	CS
*	17-20	*	3	1.3	History
*	21-25	*	3	2.3	Math
*	21-25	*	3	3.0	CS
*	17-20	*	3	failed	CS
*	17-20	*	3	1.7	Literature
*	21-25	*	3	1.0	Physics
*	21-25	*	5	3.3	Math
*	21-25	*	5	1.7	CS
*	21-25	*	5	failed	History
*	17-20	*	5	2.7	Literature
*	21-25	*	5	3.0	Math
*	17-20	*	5	1.0	Physics

Not Robust Against Background Knowledge

What can an attacker learn that knows Name, Gender, Age & Minor?

Name	Age	Gender	Semester	Grade	Minor
Alice	19	Female	1	1.3	Math
Bob	18	Male	1	2.0	Literature
Charlie	18	Male	1	1.7	Philosophy
Dave	18	Male	1	3.7	CS
Eve	17	Female	1	1.0	CS
Fritz	*19*	Male	*3*	1.3	*History*
Gerd	21	Male	3	2.3	Math
Hans	23	Male	3	3.0	CS
Isa	20	Female	3	failed	CS
John	20	Male	3	1.7	Literature
Petra	22	Female	3	1.0	Physics
Ole	22	Male	5	3.3	Math
Kale	21	Male	5	1.7	CS
Leonard	23	Male	5	failed	History
Martin	20	Male	5	2.7	Literature
Nils	22	Male	5	3.0	Math
Otto	20	Male	5	1.0	Physics

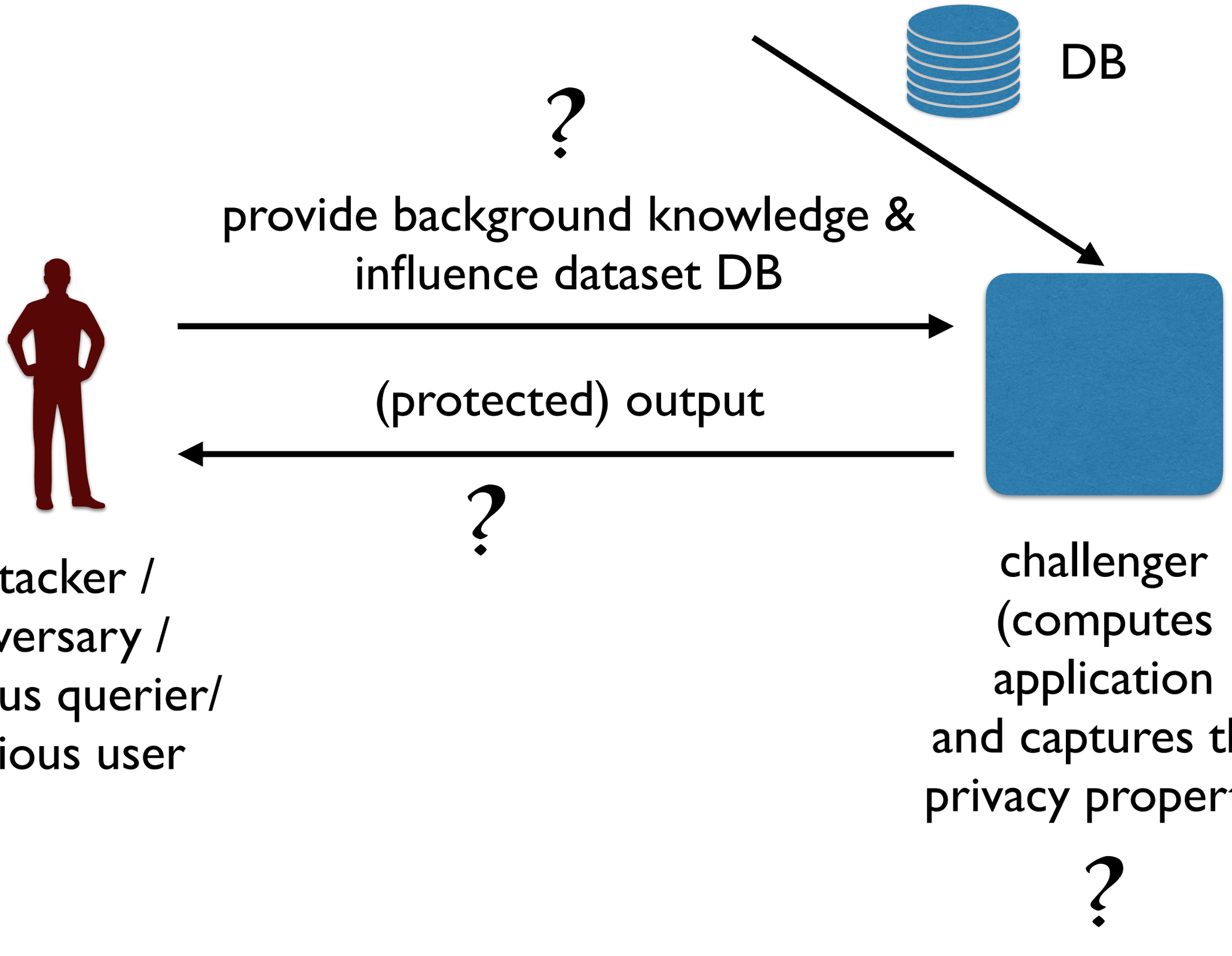
Name	Age	Gender	Semester	Grade	Minor
*	17-20	*	1	1.3	Math
*	17-20	*	1	2.0	Literature
*	17-20	*	1	1.7	Philosophy
*	17-20	*	1	3.7	CS
*	17-20	*	1	1.0	CS
Fritz	***17-20***	*	***3***	***1.3***	***History***
*	21-25	*	3	2.3	Math
*	21-25	*	3	3.0	CS
*	17-20	*	3	failed	CS
*	17-20	*	3	1.7	Literature
*	21-25	*	3	1.0	Physics
*	21-25	*	5	3.3	Math
*	21-25	*	5	1.7	CS
Leonard	***21-25***	*	***5***	***failed***	***History***
*	17-20	*	5	2.7	Literature
*	21-25	*	5	3.0	Math
*	17-20	*	5	1.0	Physics

Background knowledge vs k-Anonymity

- k-anonymous databases can contain too much information
  - Background knowledge can help de-anonymize persons
  - Too much information about single persons is preserved (and too much information thrown away)

- What would be realistic to assume?
- What could the attacker additionally know?
- Can the attacker influence the dataset?
- How to define that?

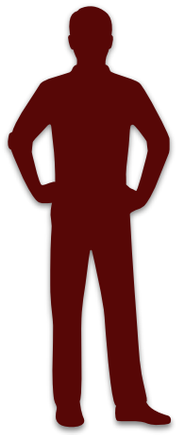
Definition: A Cryptographic Game



Definition: A Cryptographic Game

?

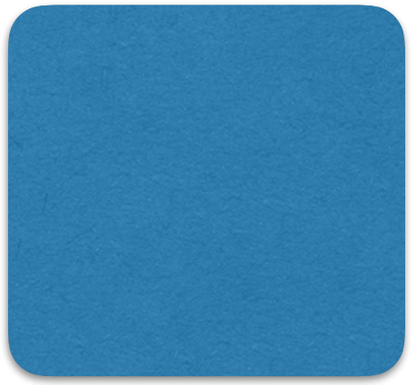
provide dataset DB?
no secrets to be guessed remain



(protected) output

?

attacker /
adversary /
curious querier/
curious user



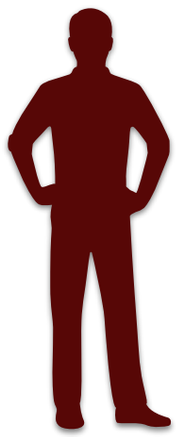
challenger
(computes
application
and captures the
privacy property)

?

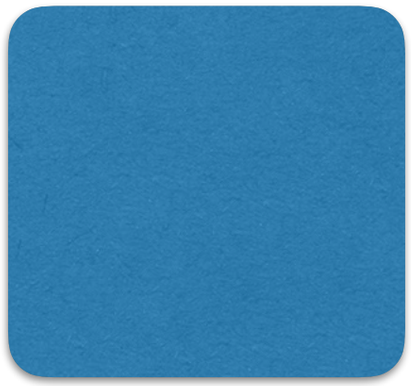
Definition: A Cryptographic Game

$b \in \{0,1\}$

provide a pair of datasets D_0, D_1 and guess which one was chosen



$M(D_b)$



attacker / adversary / curious querier/ curious user

challenger (computes application and captures the privacy property)

Definition: A Cryptographic Game

Indistinguishability?

$$\forall D_0, D_1, \mathcal{A} \; . \; |\Pr[\mathcal{A}(M(D_0)) = 0] - \Pr[\mathcal{A}(M(D_1)) = 0]| \; ?$$

No, impossible in many cases where $M(D_b)$ computes a useful function

What Can we Do?

For, e.g., counting query q *"# of cancer patients in 2019 at UKSH"*

query-result: $q(D) := \sum_{x \in D} 1$ $s.t.\ p(x)$

Here: $p(x) \iff$

x is a cancer patient in 2019

$$q(D) = 3$$

counting highly stable:
one row difference,
minor change in output

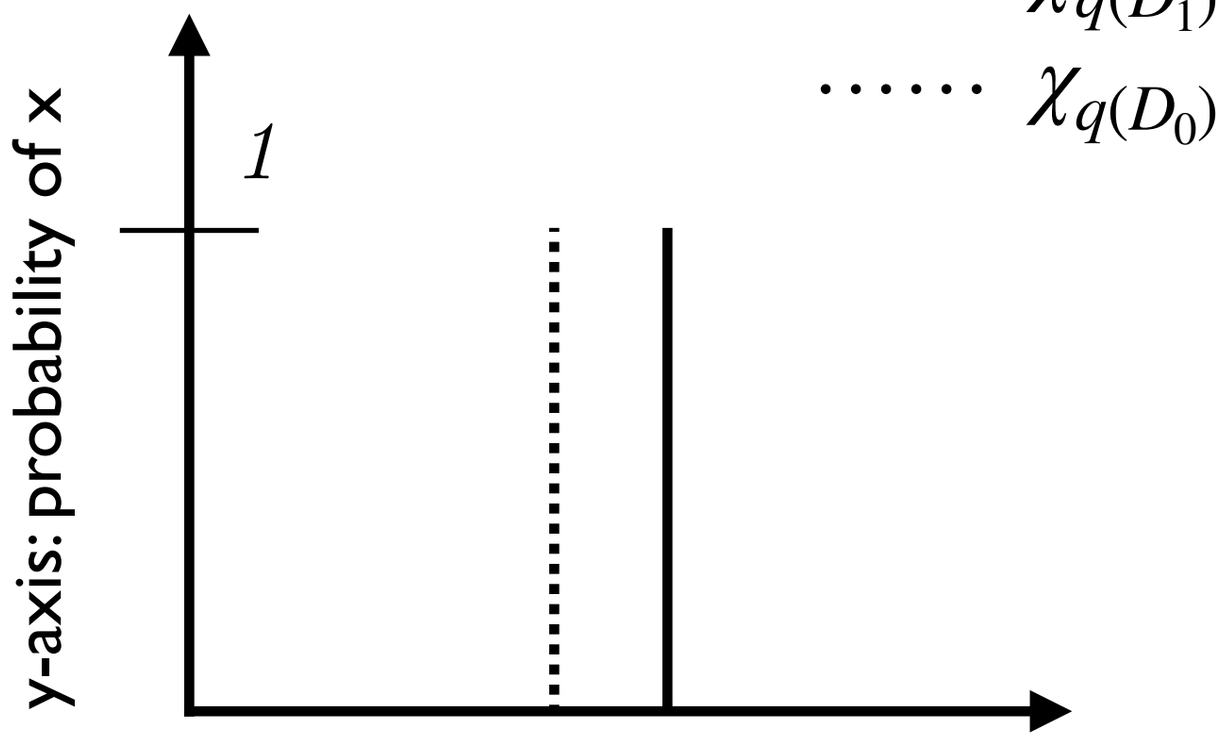
Name	Age	Gender	year	Decease
Alice	60	Female	2017	Cancer
Bob	54	Male	2002	Heart attack
Charlie	70	Male	1982	Cancer
Dave	43	Male	1999	Fracture
Eve	88	Female	2018	Cancer
Fritz	81	Male	2019	Fracture
Gerd	67	Male	2011	Heart attack
Hans	35	Male	2019	Cancer
Isa	64	Female	2003	Allergic reaction
John	72	Male	2005	Food poisoning
Petra	80	Female	1986	Cancer
Ole	74	Male	2019	Cancer
Kale	94	Male	2014	Fracture
Leonard	96	Male	2018	Cancer
Martin	86	Male	2012	Allergic reaction
Nils	78	Male	2009	Heart attack
Otto	40	Male	2019	Cancer

Deterministic Counting Query

For, e.g., counting query q

"# of cancer patients in 2019 at UKSH"

$$M(D) = q(D) := \sum_{x \in D} 1 \quad s.t.\ p(x)$$



datasets only differ in 1 element (all combinations of 1-difference):

$\forall D_0, t\,.\quad D_1 := D_0 \cup \{t\}$

Perturb the Counting Query

For, e.g., counting query q

"# of cancer patients in 2019 at UKSH"

query-result: $q(D) := \sum_{x \in D \; s.t. \; p(x)} 1$

Mechanism $M(D)$:

add Laplace noise $Lap(0, b)$

with mean 0

and scale parameter b

to the query-result $q(D)$

$$M(D) := q(D) + Lap(0, b)$$
$$= Lap(q(D), b)$$

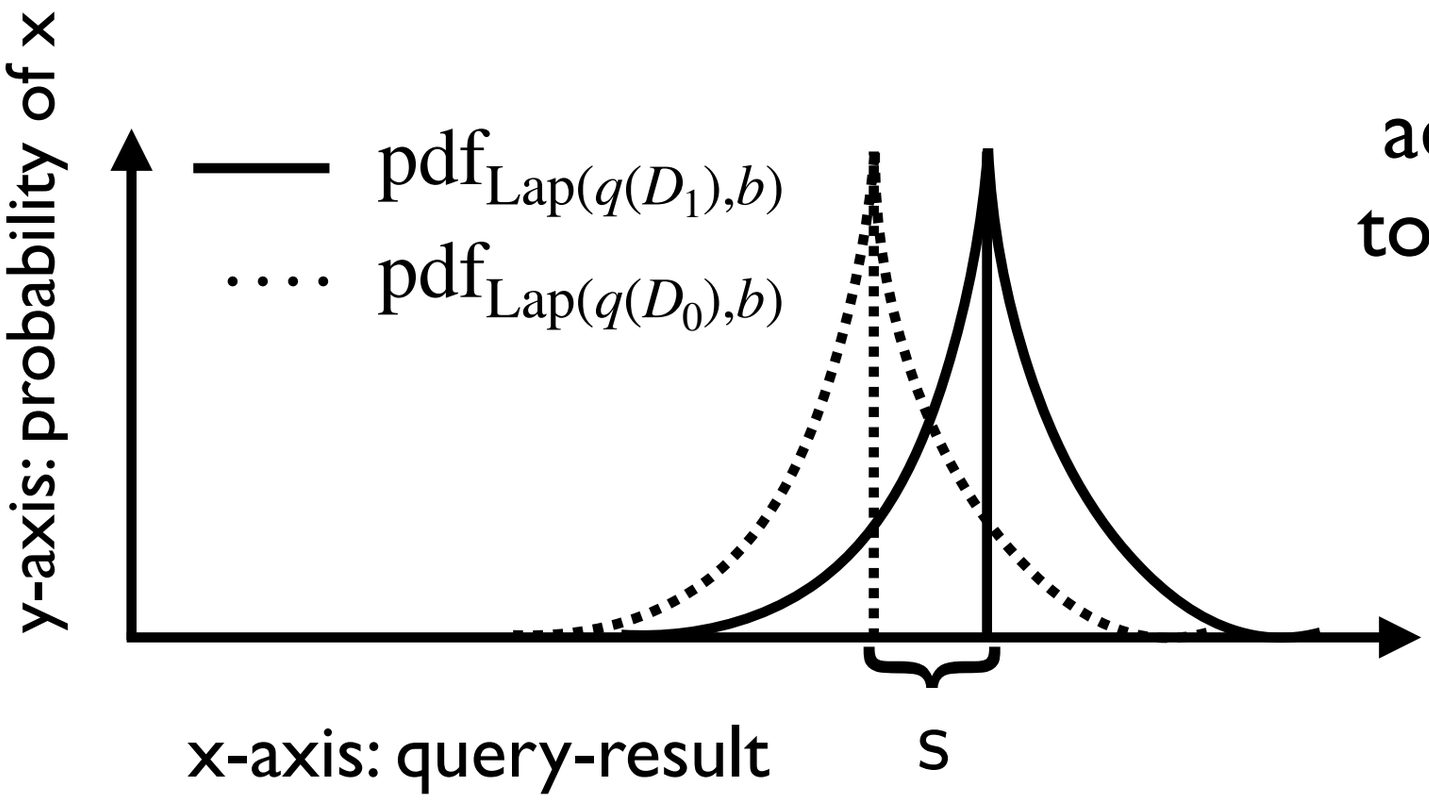
Perturb the Counting Query (cont'd)

For, e.g., counting query q

"*# of cancer patients in 2019 at UKSH*"

pairs of databases D_0, D_1

with sensitivity s (difference in the query-result)

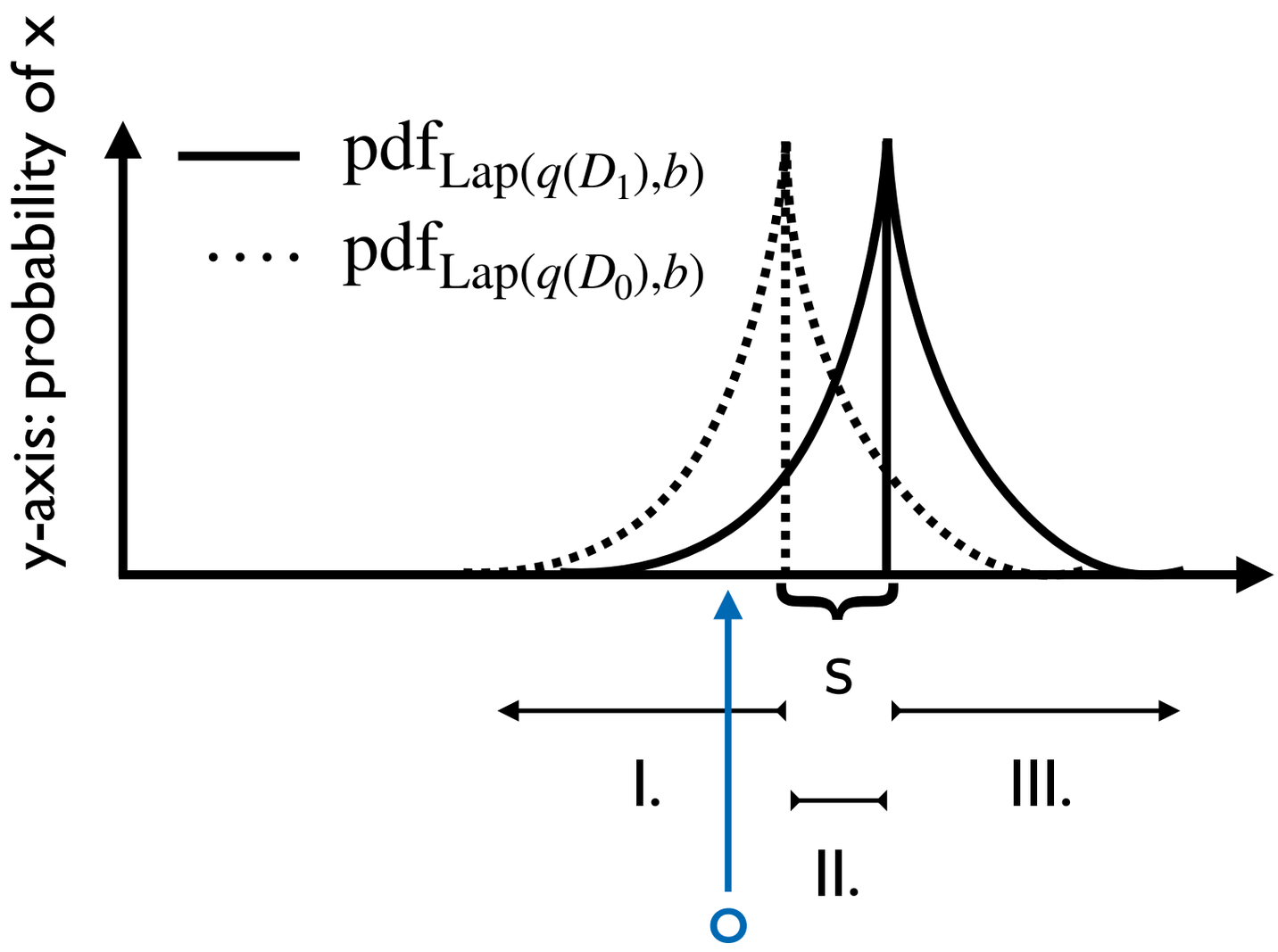


add Laplace noise
to the query-result

Laplace Perturbed Counting Queries

$$q(D) + \mathrm{Lap}(0,b)$$
$$= \mathrm{Lap}(q(D), b)$$

$$\mathrm{pdf}_{\mathrm{Lap}(\mu,b)}(o)$$
$$= \frac{1}{2b} \exp\left(-\frac{|o-\mu|}{b}\right)$$



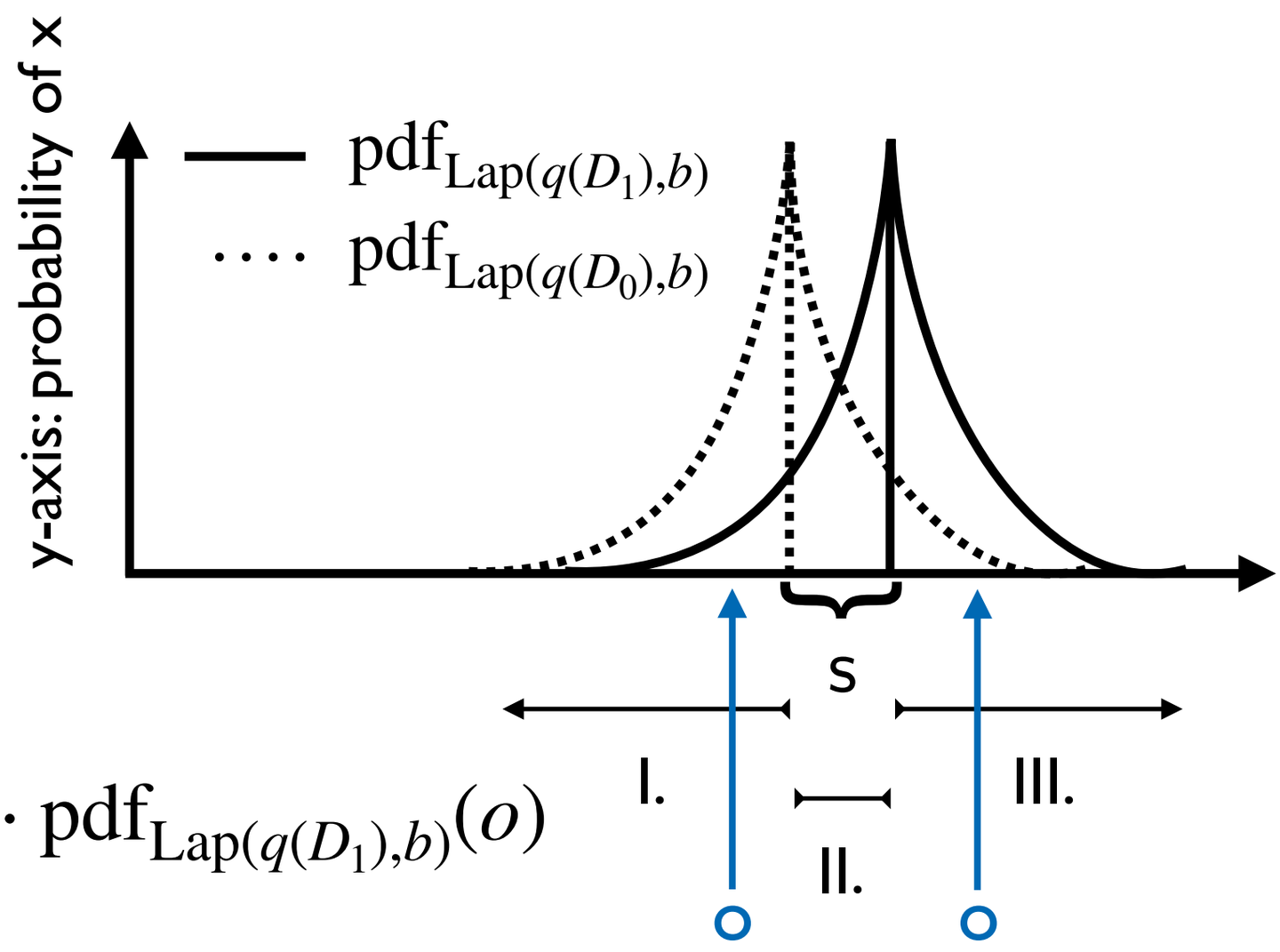
For I.

$$\mathrm{pdf}_{\mathrm{Lap}(q(D_0),b)}(o)$$

Laplace Perturbed Counting Queries

$$q(D) + \mathrm{Lap}(0,b)$$
$$= \mathrm{Lap}(q(D), b)$$

$$\mathrm{pdf}_{\mathrm{Lap}(\mu,b)}(o)$$
$$= \frac{1}{2b} \exp\left(-\frac{|o-\mu|}{b}\right)$$



For I.

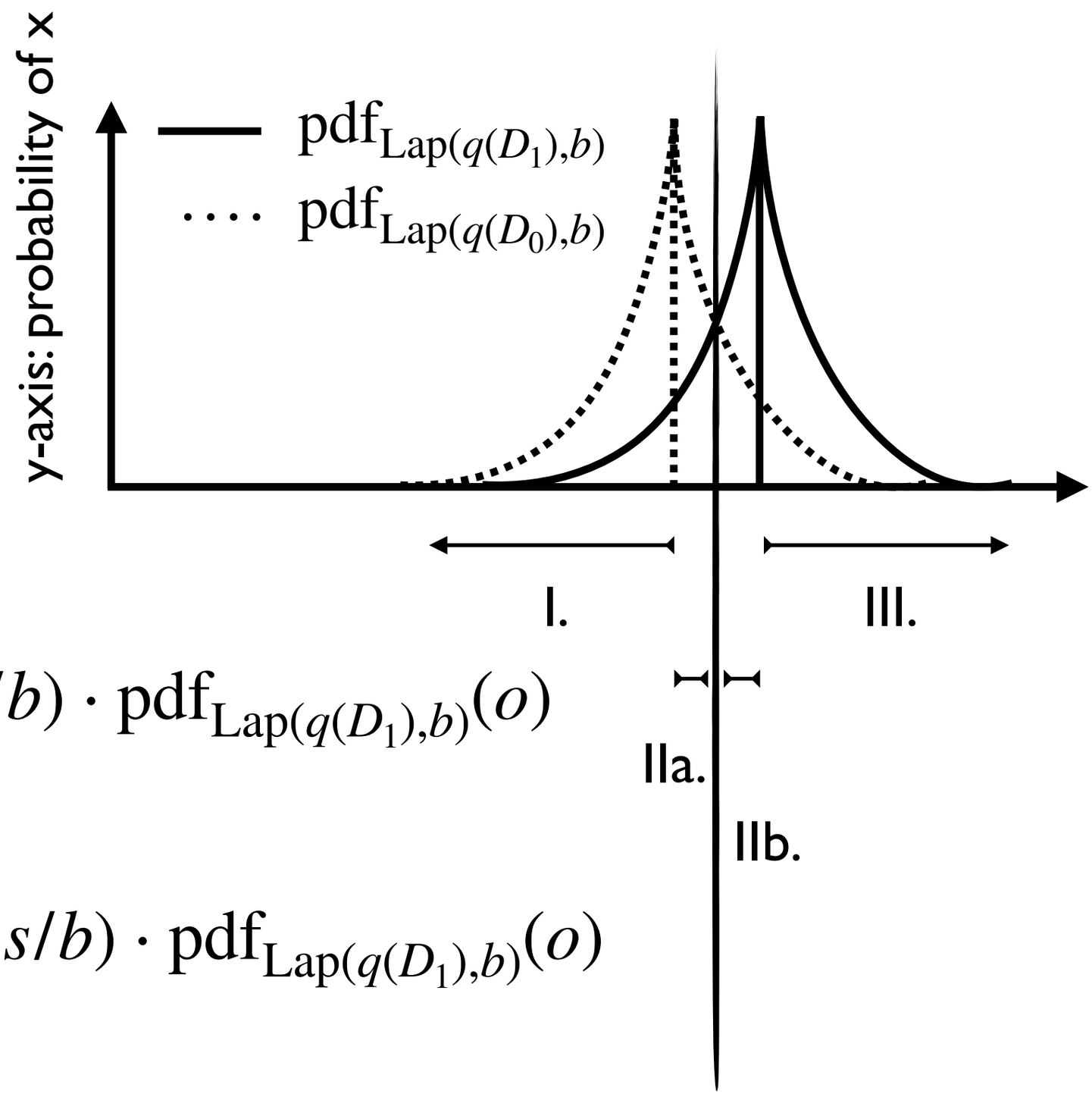
$$\mathrm{pdf}_{\mathrm{Lap}(q(D_0),b)}(o) = \exp(s/b) \cdot \mathrm{pdf}_{\mathrm{Lap}(q(D_1),b)}(o)$$

III. symmetric to
I. with $-s$ instead of s

Laplace Perturbed Counting Queries

$$q(D) + \mathrm{Lap}(0,b)$$
$$= \mathrm{Lap}(q(D), b)$$

$$\mathrm{pdf}_{\mathrm{Lap}(\mu,b)}(o)$$
$$= \frac{1}{2b} \exp\left(-\frac{|o-\mu|}{b}\right)$$



For IIa.

$$\mathrm{pdf}_{\mathrm{Lap}((q(D_0),b)}(o) \leq \exp(s/b) \cdot \mathrm{pdf}_{\mathrm{Lap}(q(D_1),b)}(o)$$

For IIb.

$$\mathrm{pdf}_{\mathrm{Lap}(q(D_0),b)}(o) \leq \exp(-s/b) \cdot \mathrm{pdf}_{\mathrm{Lap}(q(D_1),b)}(o)$$

Laplace Perturbed Counting Queries

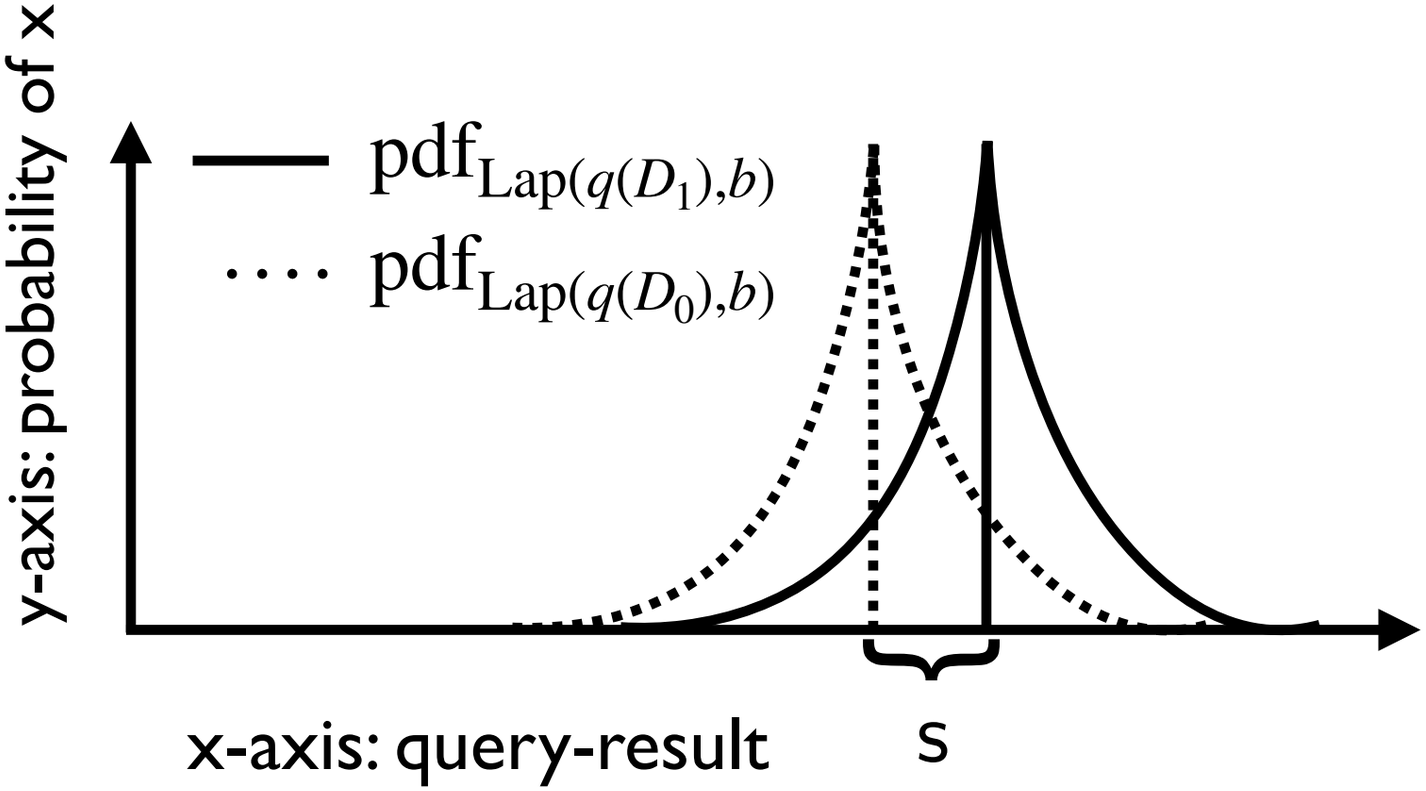
For, e.g., counting query q

"# of cancer patients in 2019 at UKSH"

pairs of databases D_0, D_1
with $D_1 := D_0 \cup \{x\}$
(one element difference)

$$\forall o \,.\, \mathrm{pdf}_{\mathrm{Lap}(q(D_0),b)}(o) \leq \exp(1/b) \cdot \mathrm{pdf}_{\mathrm{Lap}(q(D_1),b)}(o)$$

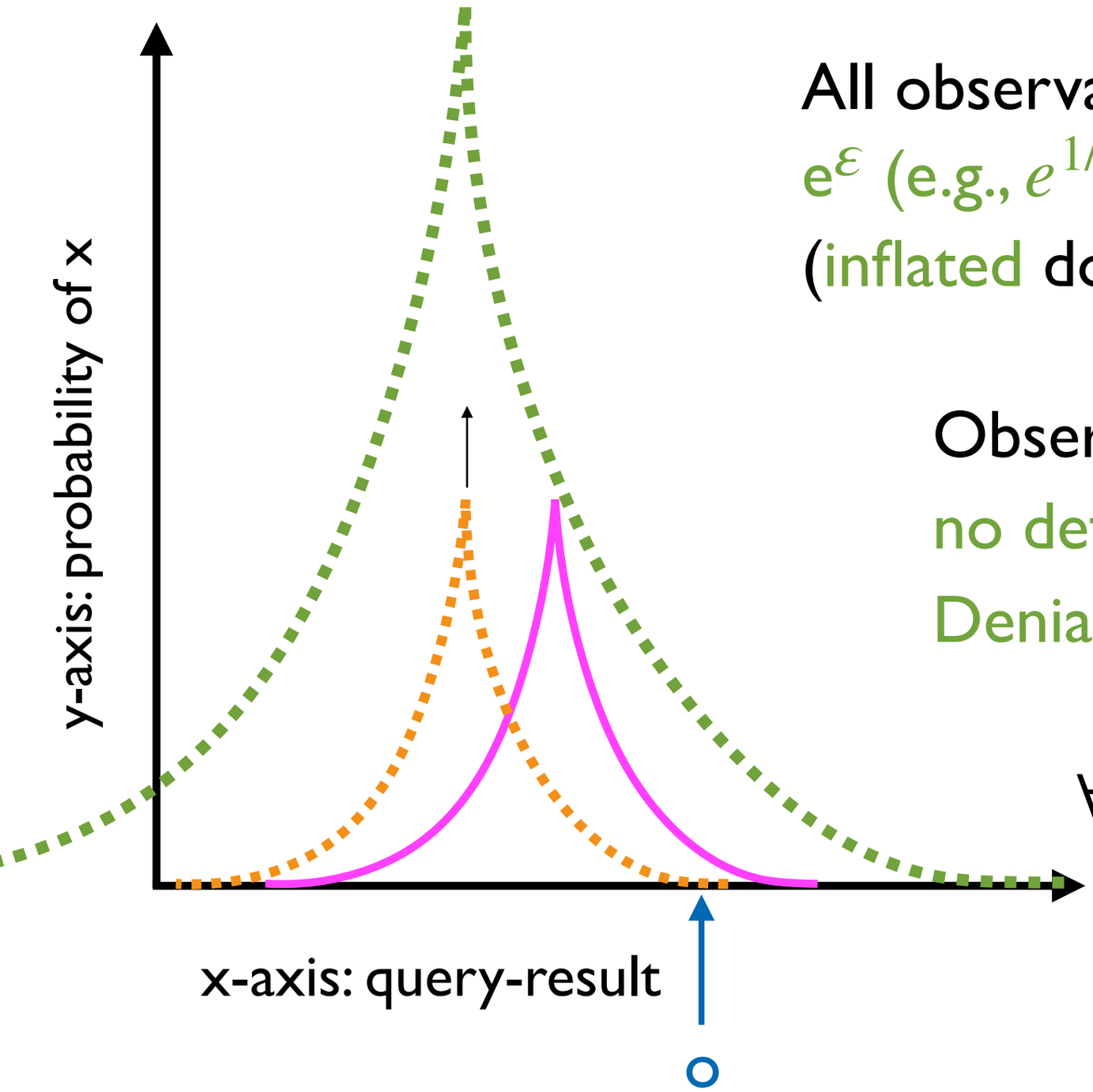
rest of the talk: $s = 1$



add Laplace noise
to the query-result

Differential Privacy

$$\forall o \,.\, \mathrm{pdf}_{\mathrm{Lap}(q(D_0))}(o) \leq \exp(1/b) \cdot \mathrm{pdf}_{\mathrm{Lap}(q(D_1))}(o)$$



All observations o in B are at most
e^{ε} (e.g., $e^{1/b}$) times as likely as in A
(inflated dotted line $2 \cdot$ B above solid line A)

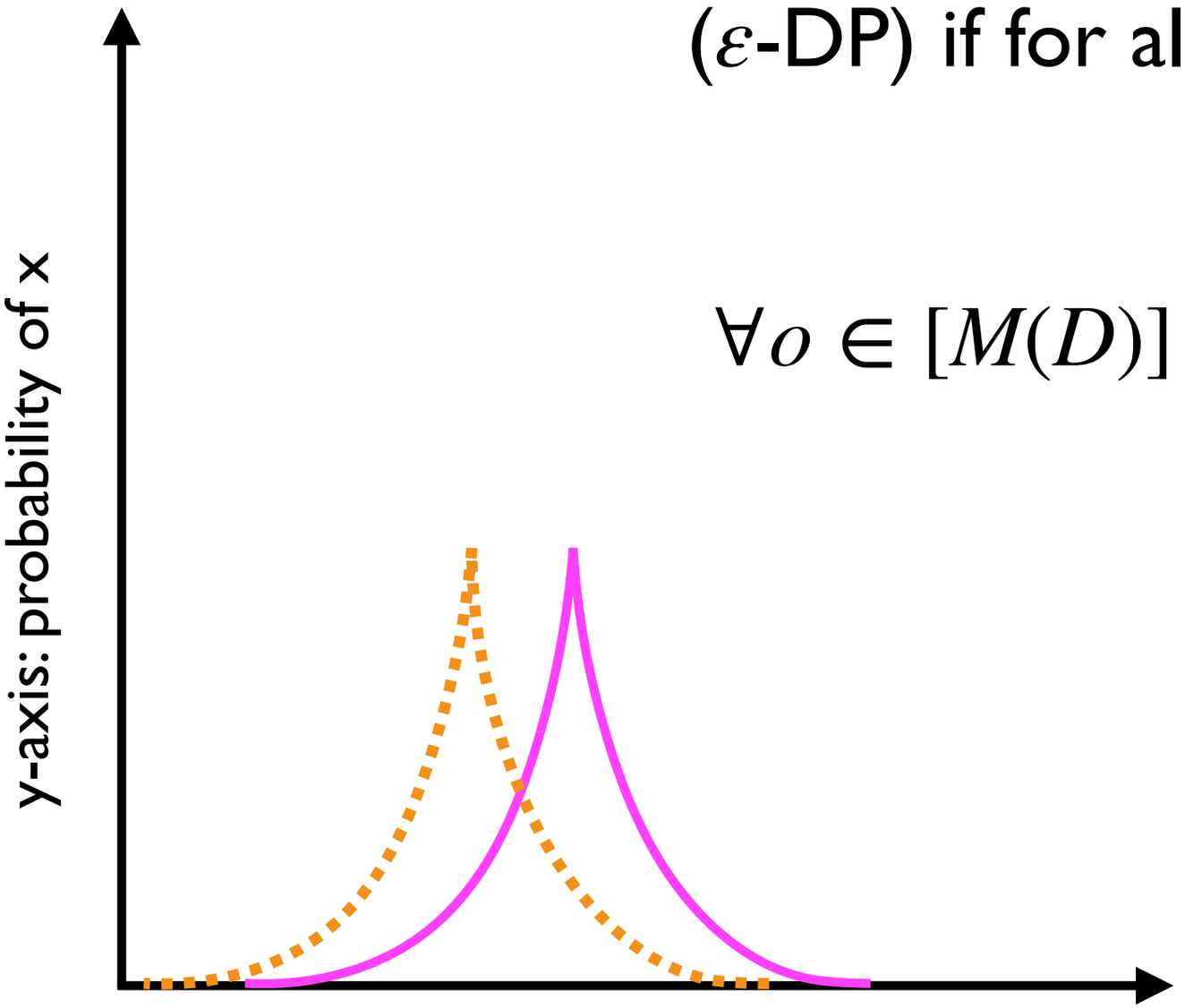
Observing any event (e.g., o):
no definitive proof
Deniability against an observation

$$\forall o \in [B] \,.\, e^{-\varepsilon} \leq \frac{\mathrm{pdf}_A(o)}{\mathrm{pdf}_B(o)} \leq e^{\varepsilon}$$

ε-Differential Privacy

A mechanism M is ε-differentially private (ε-DP) if for all dataset D and all rows x

$$\forall o \in [M(D)] \,.\, e^{-\varepsilon} \leq \frac{\mathrm{pdf}_{M(D \cup \{x\})}(o)}{\mathrm{pdf}_{M(D)}(o)} \leq e^{\varepsilon}$$



(also with $D \cup \{x\}$ and D in switched roles, omitted for the sake of brevity)

Connection to KL Diversity

- ε-Differential privacy
  - worst case (log) ratio (for all $o \in [M(D)]$) is bounded

$$\forall o \in [M(D)] \,.\, e^{-\varepsilon} \leq \frac{\mathrm{pdf}_{M(D \cup \{x\})}(o)}{\mathrm{pdf}_{M(D)}(o)} \leq e^{\varepsilon}$$

$$\forall o \in [M(D)] \,.\, -\varepsilon \leq \ln \frac{\mathrm{pdf}_{M(D \cup \{x\})}(o)}{\mathrm{pdf}_{M(D)}(o)} \leq \varepsilon$$

- KL Divergence (relative entropy)
  - expected case log ratio

$$\int_{-\infty}^{\infty} \mathrm{pdf}_{\mathrm{M(D \cup \{x\})}}(o) \cdot \ln \frac{\mathrm{pdf}_{M(D \cup \{x\})}(o)}{\mathrm{pdf}_{M(D)}(o)} do$$

Post-Processing Theorem

- Leakage can never increase if you further process the output of a computation
- for all S,D,D':

$$\Pr[M(D_0) \in S] < \exp(\varepsilon) \Pr[M(D') \in S]$$

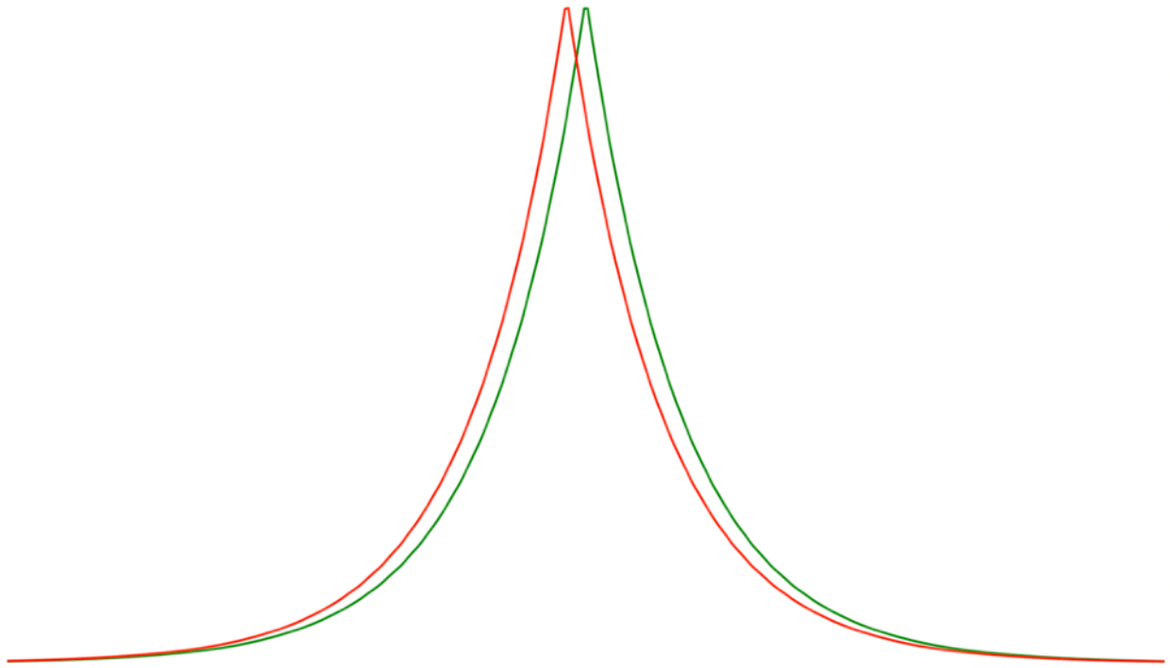
- Then, for all Alg:

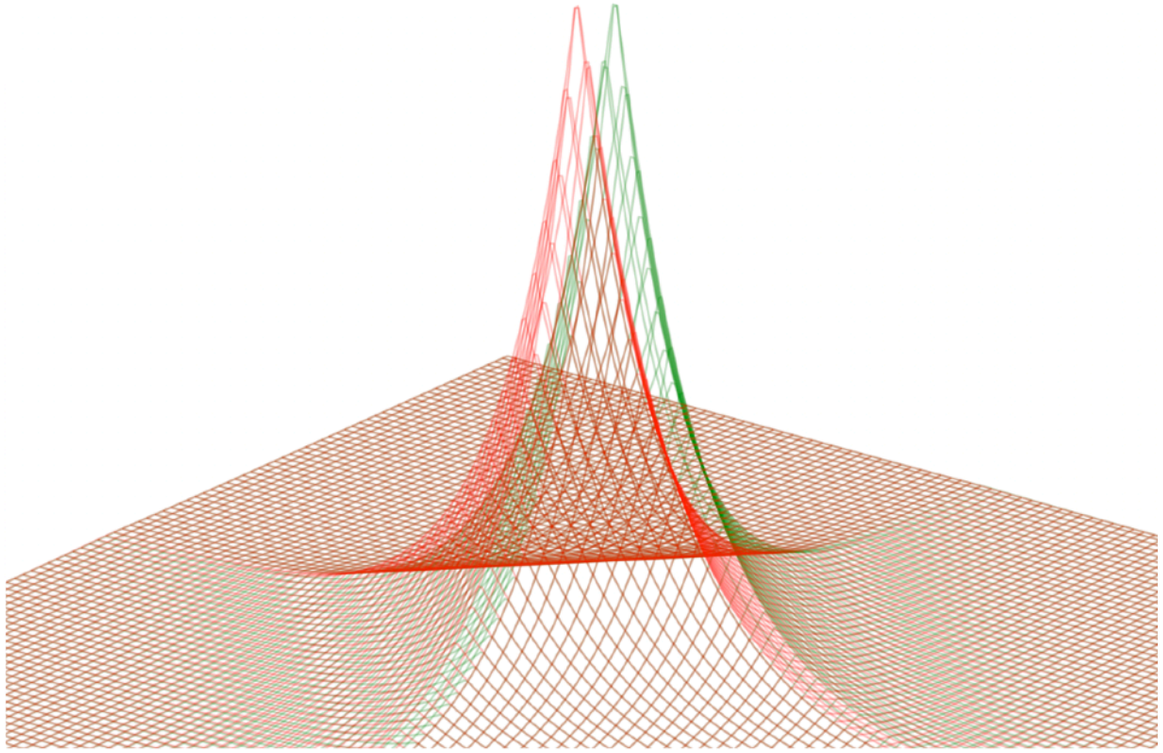
$$\Pr[\mathrm{Alg}(M(D_0)) \in S] < \exp(\varepsilon) \Pr[\mathrm{Alg}(M(D')) \in S]$$

- Useful for proving DP of complicated algorithms

Sequential Composition as a Graph

- One noised counting query: observations $o \in \mathbb{R}$
- Several noised counting queries: observations $(o_1, o_2) \in \mathbb{R}^2$
- Example: Laplace





Sequential Composition

- Attacker can better an better estimate means
  - Attacker knows two candidates for each query
- Ratios increase exponentially with the number of queries
- Leakage increases:

  ε-DP after one query response

  $\Longrightarrow 2\varepsilon$-DP after two query responses (even adaptive query responses)

$$\frac{\mathrm{pdf}_{M_1(D\cup\{x\})}(o_1)}{\mathrm{pdf}_{M_1(D)}(o_1)} \leq e^{\varepsilon_1} \wedge \frac{\mathrm{pdf}_{M_2(D\cup\{x\})}(o_2)}{\mathrm{pdf}_{M_2(D)}(o_2)} \leq e^{\varepsilon_2}$$

$$\Longrightarrow \frac{(\mathrm{pdf}_{M_1(D\cup\{x\})}, \mathrm{pdf}_{M_2(D\cup\{x\})})(o_1, o_2)}{(\mathrm{pdf}_{M_1(D)}, \mathrm{pdf}_{M_2(D)})(o_1, o_2)} \leq e^{\varepsilon_1+\varepsilon_2}$$

Sufficient Condition: Bounded Sensitivity

- $f : \mathbb{R}^a \to \mathbb{R}$ (can be generalized to $f : \mathbb{R}^a \to \mathbb{R}^b$)
- Sensitivity (in the DP community):
  changes in the output if one one element of the input data set changes

$$\Delta_f := \max_{D, D \cup \{x\}} ||f(D) - f(D \cup \{x\})||_2$$

- Bounded sensitivity ($\Delta_f < \infty$) is sufficient to achieve DP:

$$f(D) + \mathrm{Lap}(0, \Delta_f / \varepsilon)$$

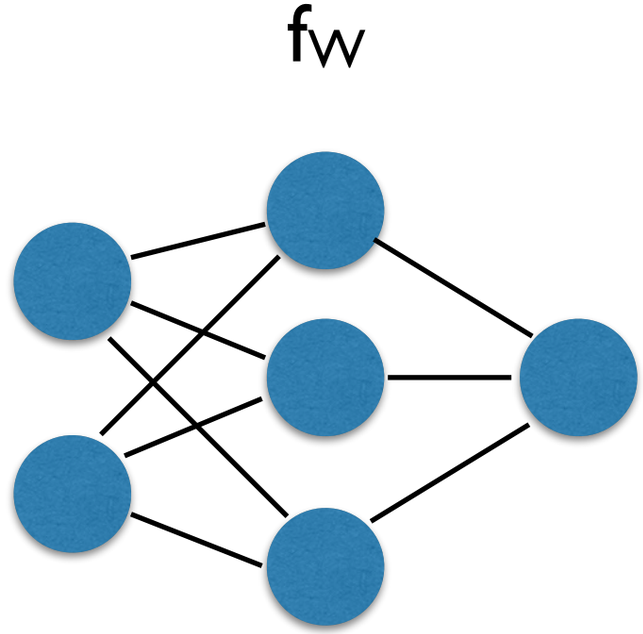
- Bounded sensitivity: very strong form of stability
  - corresponds to *change-one-error-stability* for the output distribution
- Most DP mechanisms bound sensitivity $\Longrightarrow$ achieve stability

Outline

- Traces of training data in ANNs
- How to formulate privacy?
- Private learning
- Other learning techniques

Neural Networks: Training

- A neural network is a function $f_W(x)$ that is parametric in some weights W
- Training an NN with a loss function L and training data points (x,y)
- Find W such that L := L($f_W(x)$,y) is minimized (e.g., L($f_W(x)$,y) = |$f_W(x)$ - y|)
- Plan: Minimize L($f_W(x)$,y) using partial derivatives



Gradient Descent with Partial Derivatives

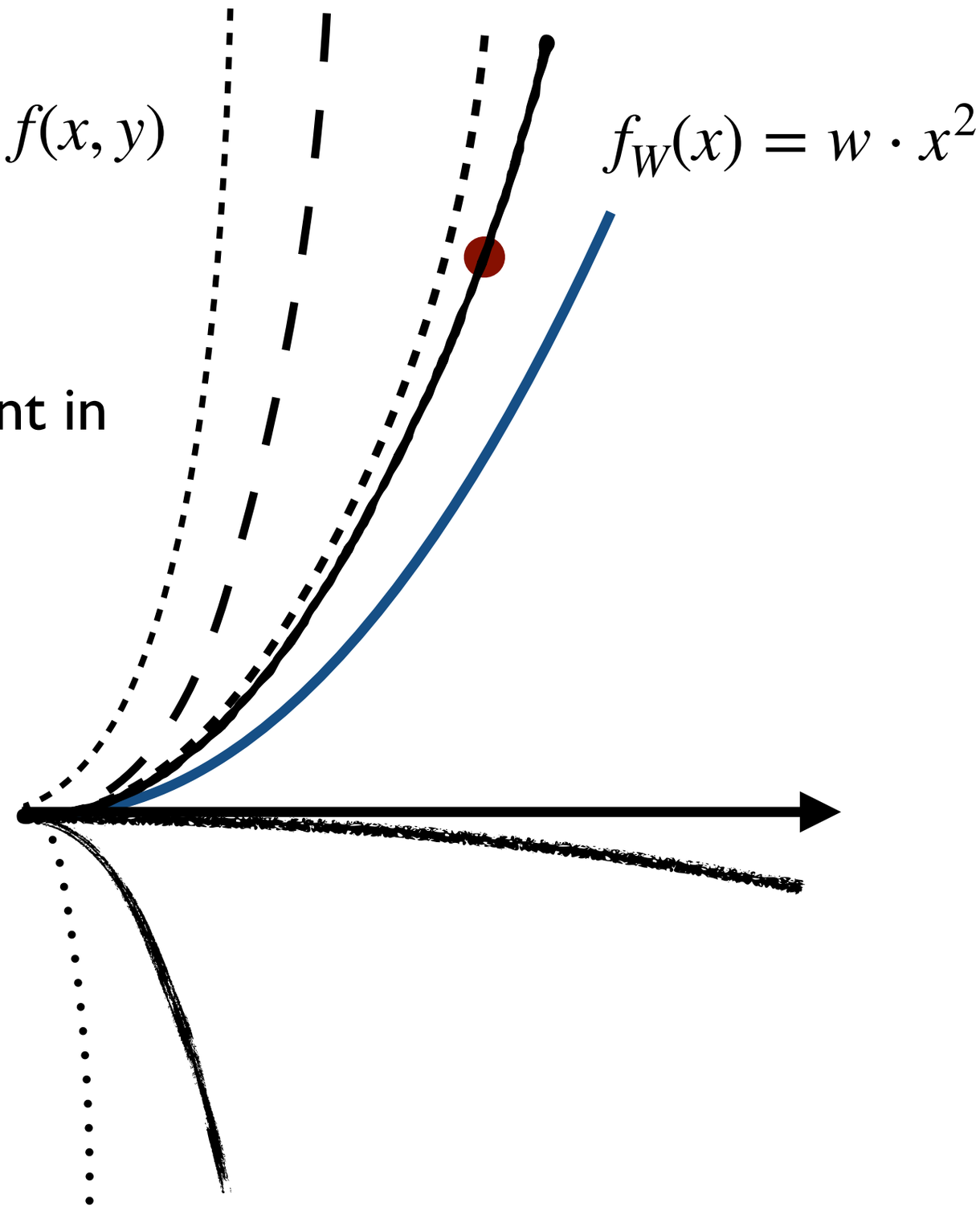
- Gradient Descent
  - in each round t compute $\nabla_{W_t} f(x, y)$
  - update parameters / weights: $W_{t+1} := W_t - \nabla_W f_{W_t}(x, y)$
  - gradients $\nabla_{W_t} f(x, y)$ only point in the right direction

➡ decrease the weight of the update

$W_{t+1} := W_t - \eta_t \nabla_W f_{W_t}(x, y)$

with η_t decreasing with t

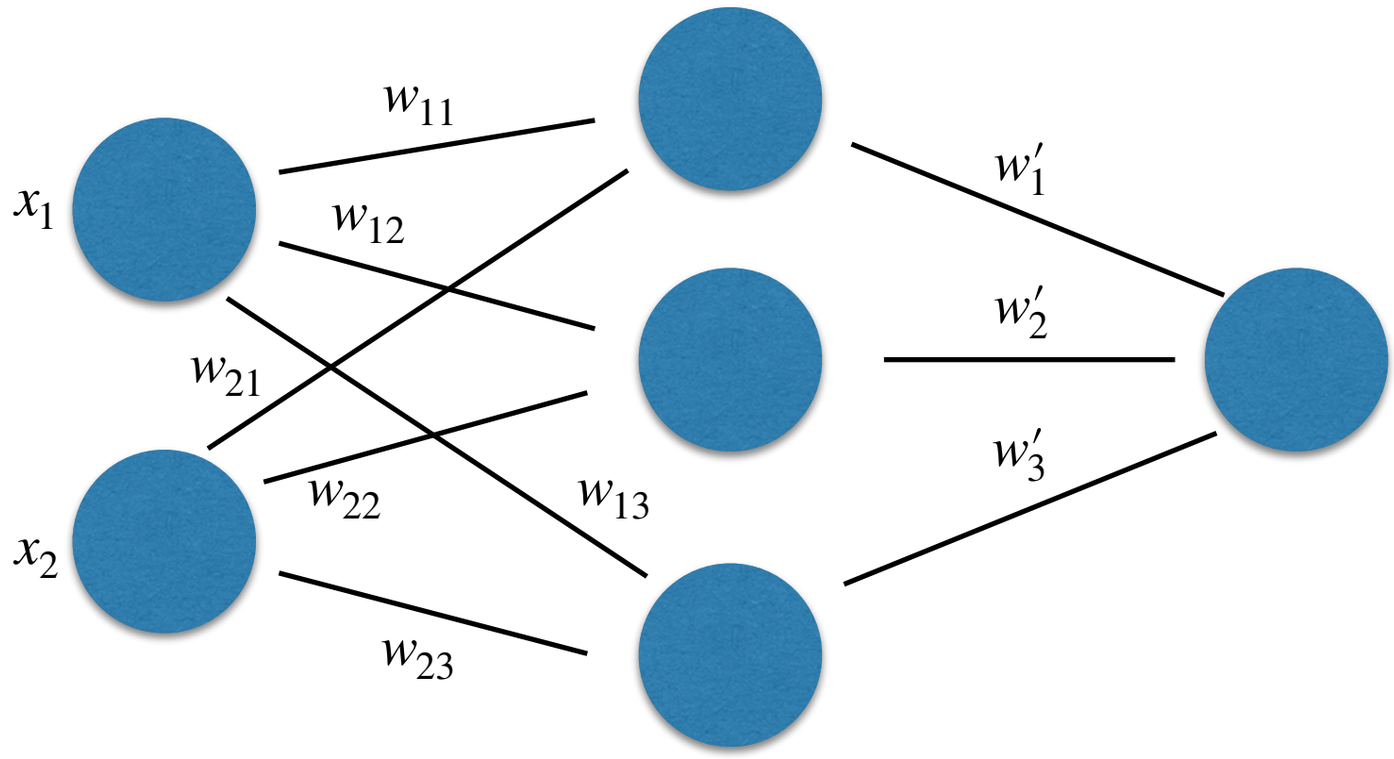
e.g., $\eta_t := \min(1, 1/t + 100)$



Partial Derivatives: Neural Network

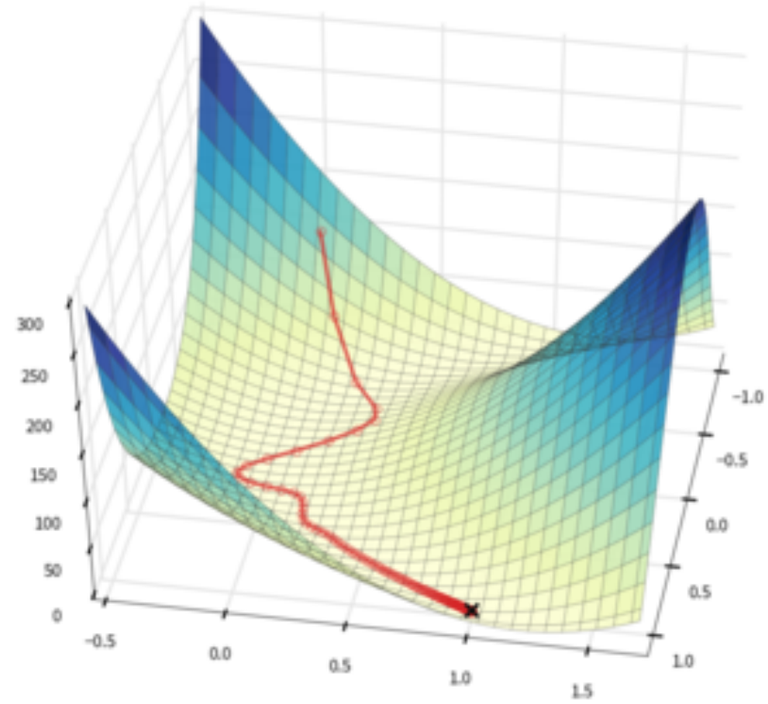
$$\nabla_W f = \begin{pmatrix} \frac{\partial f_W}{\partial w_{11}} \\ \cdot\cdot \\ \frac{\partial f_W}{\partial w_{23}} \\ \frac{\partial f_W}{\partial w'_1} \\ \cdot\cdot \\ \frac{\partial f_W}{\partial w'_3} \end{pmatrix}$$

$$f_W(x_1, x_2) = \max(0, \sum_{j=1}^{3} w'_j \cdot max(0, \sum_{i=1}^{2} w_{ij} x_i))$$



Mini-Batch SGD

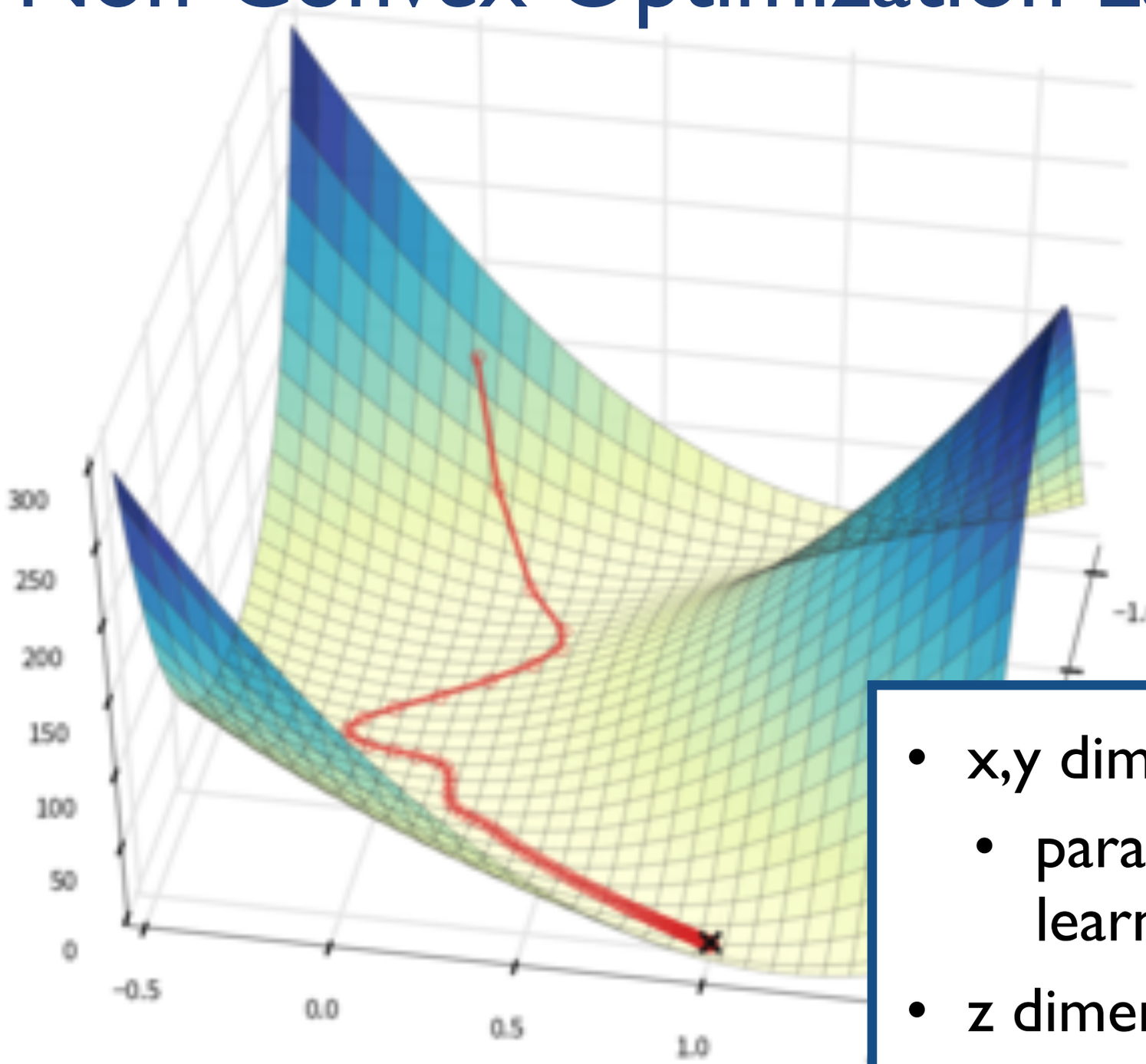
- Gradient descent (GD)
  - Compute gradient $\nabla_W L(x, y)$
  - Update: $W_{t+1} := W_t - \nabla_W L(x, y)$
- Perform Stochastic GD (SGD) with mini-batch
  - Iteratively compute the gradient of a random subsets of the training points $(x_i, y_i)_{i=1}^k$



  - Subtract the average of the gradients

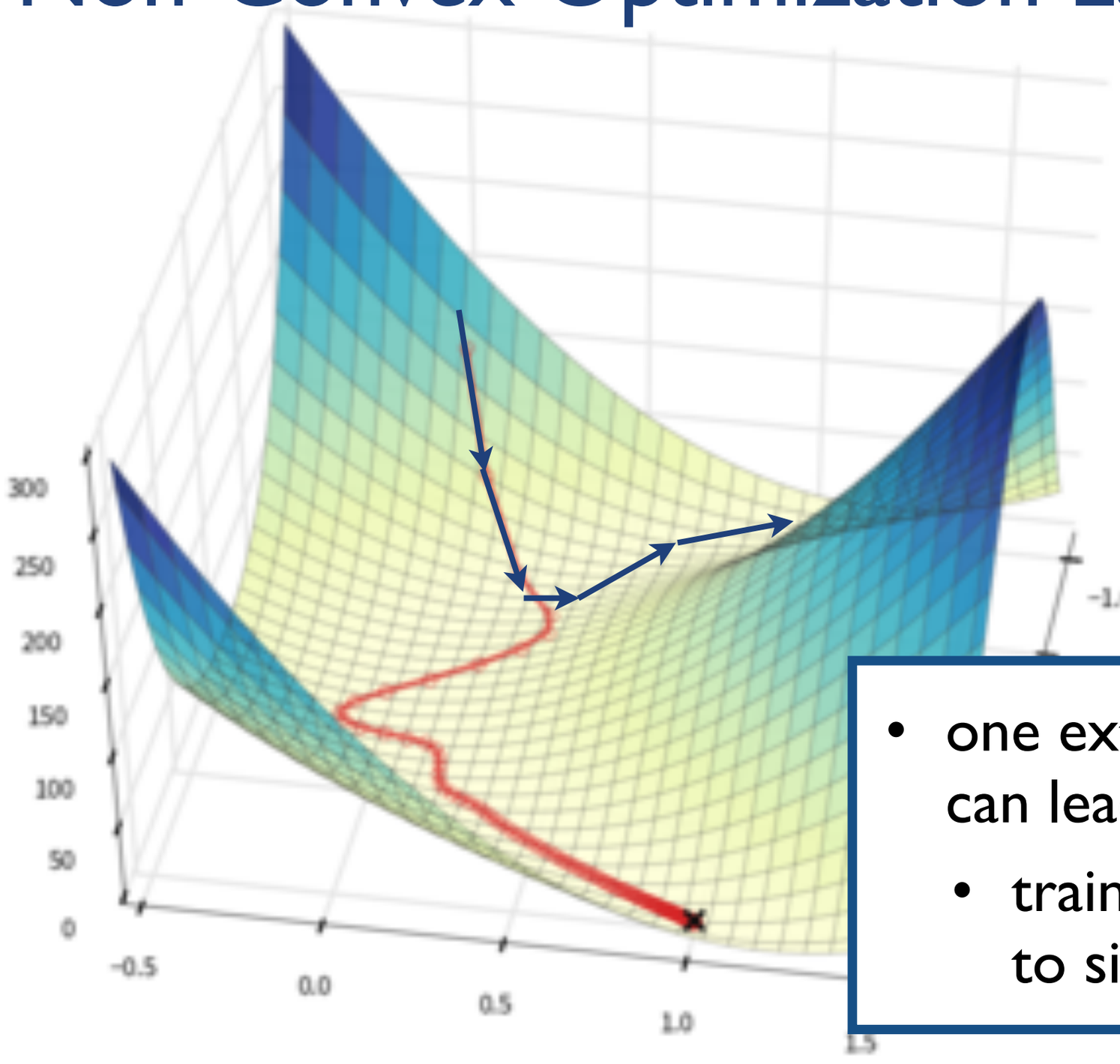
$$W_{t+1} := W_t - \frac{1}{k} \sum_{i=1}^{k} \nabla_W L(x_i, y_i)$$

Non-Convex Optimization Landscape



- x,y dimension: weights
  - parameters of the learned function
- z dimension: loss (L(x,y))

Non-Convex Optimization Landscape



- one extra training point can lead to another path
  - training can be sensitive to single training points

Can we achieve DP for ML?

- A good goal would be
  - a probability distribution over paths (or over local optima)
  - the probabilities of each path / local optimum not much influenced by a single training data point (stability)

DP Empirical Risk Minimization[1]

- Given an objective function L, find model $h^* : \mathbb{R}^a \to \mathbb{R}^b$ via ERM:

$$h^* := \text{argmin}_h \frac{1}{n} \sum_{i=1}^{n} L(h(x_i), y_i)$$

- First work to perturb the objective function L
- Modify gradient descent
  - $W_{i+1} := W_i - \sum_{i=1}^{n} \nabla_L(x_i, y_i) + (\text{Lap}(0, 1/\varepsilon)1_b)^T h(x_i)$
    ($1_b \in \mathbb{R}^b$ is the constant-1 vector)
- Precondition: $|L'(x, y)| \leq 1$ for all x, y
  - after normalizing L, this is applicable for linear regression and SVMs
  - Convex optimization problems, ε-DP, for some $\varepsilon > 0$

[1] Differentially Private Empirical Risk Minimization by Kamalika Chaudhuri, Claire Monteleoni, Anand D. Sarwate. In JMLR 2011.

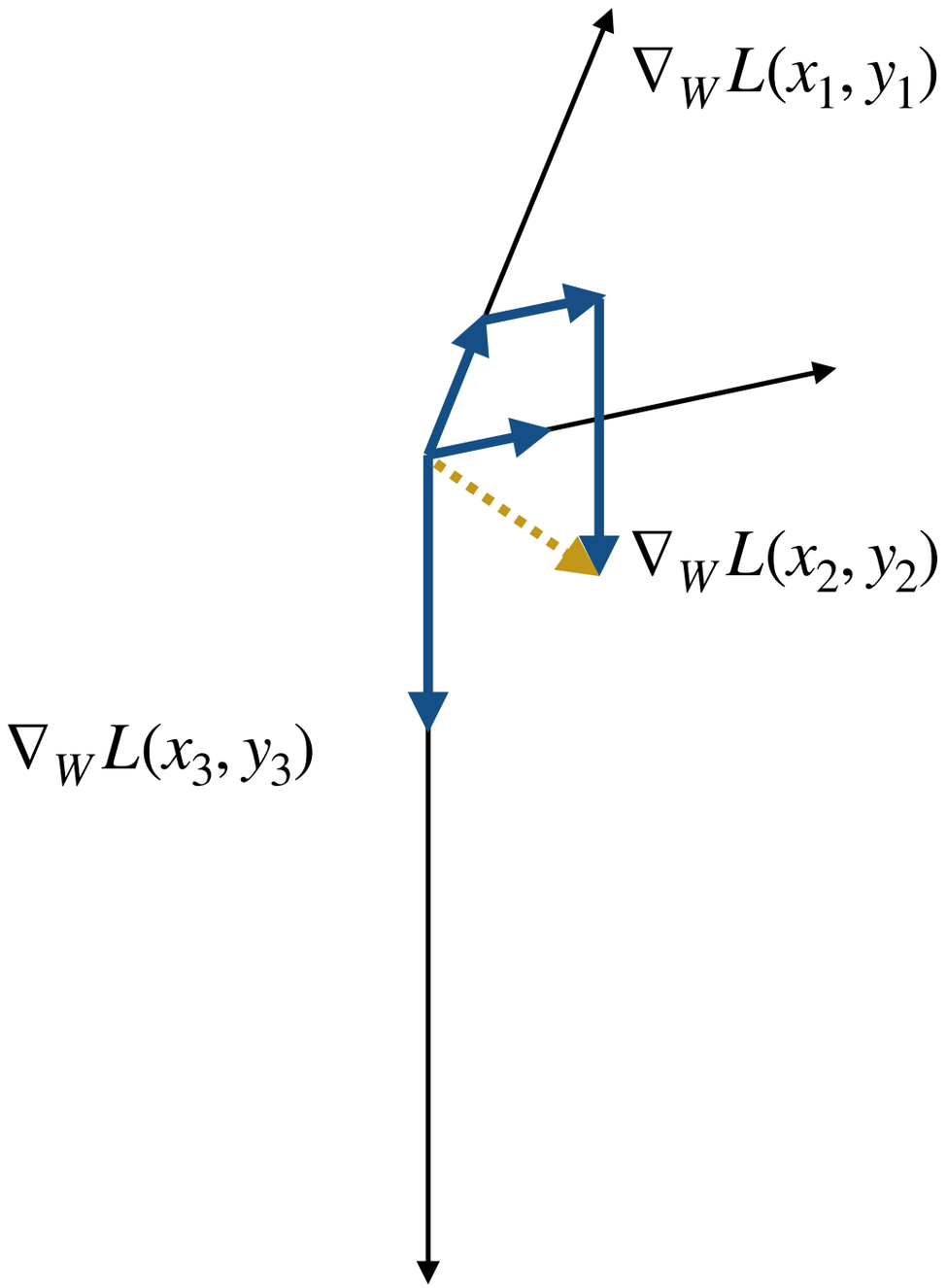
Noisy SGD

- Less strict normalization: norm clipping (winsorized mean)
  - also non-convex optimization problems, e.g., ANNs
- E.g., winsorized mean is one robust statistic
    - other robust statistics might be interesting (influence functions)

Mini-Batch SGD: Computing the Update

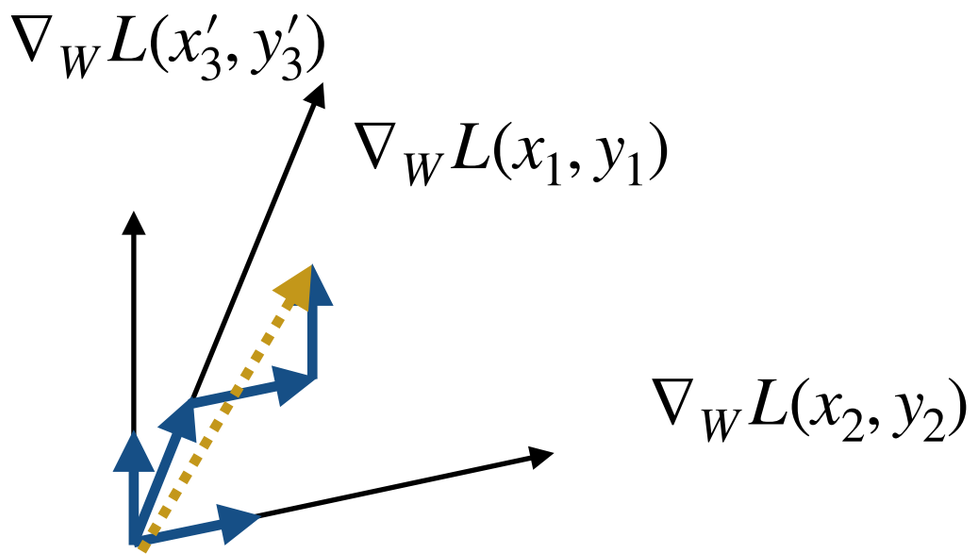
Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$
$\subset D \cup \{(x_3, y_3)\}$



Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x_3', y_3')\} \subset D$

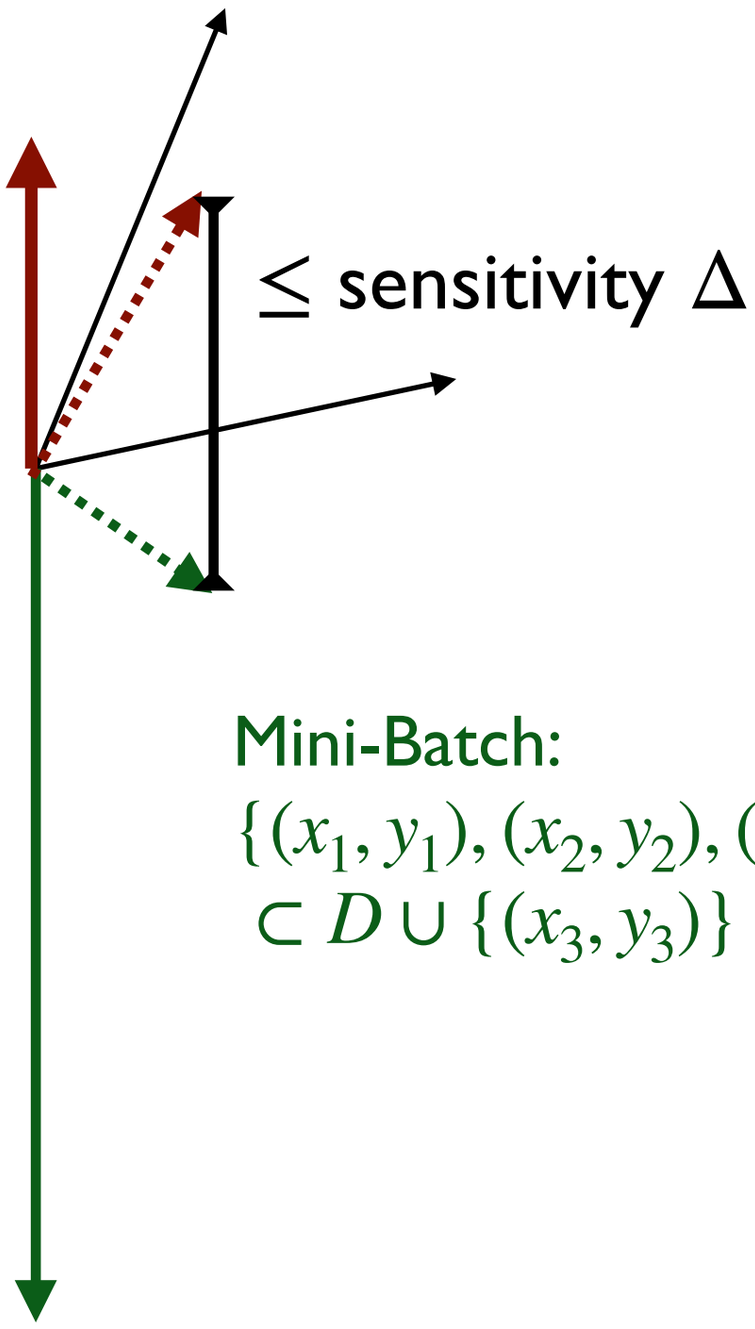


$$W_{t+1} := W_t - \frac{1}{k} \sum_{i=1}^{k} \nabla_W L(x_i, y_i)$$

Mini-Batch SGD: Sensitivity

Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x_3', y_3')\} \subset D$



We could now add noise, but how much?

Mini-Batch:

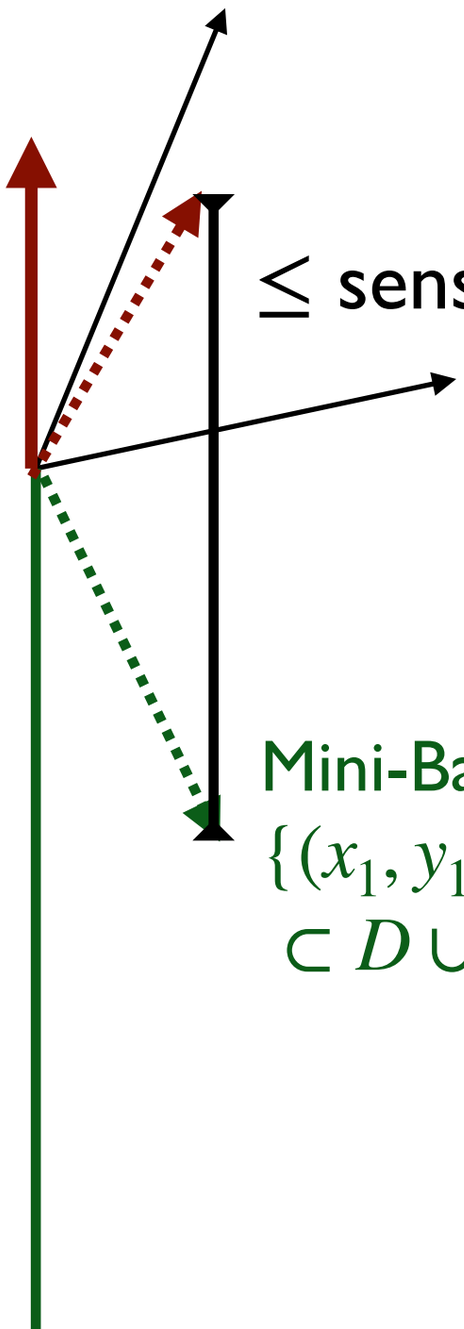
$\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$
$\subset D \cup \{(x_3, y_3)\}$

$$W_{t+1} := W_t - \frac{1}{k} \sum_{i=1}^{k} \nabla_W L(x_i, y_i)$$

Mini-Batch SGD: Unbounded Sensitivity

Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x_3', y_3')\} \subset D$



Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x_3'', y_3'')\}$
$\subset D \cup \{(x_3'', y_3'')\}$

Sensitivity is unbounded. ⚡
How to keep Δ small?

$$W_{t+1} := W_t - \frac{1}{k} \sum_{i=1}^{k} \nabla_W L(x_i, y_i)$$

Mini-Batch SGD: Norm Clipping

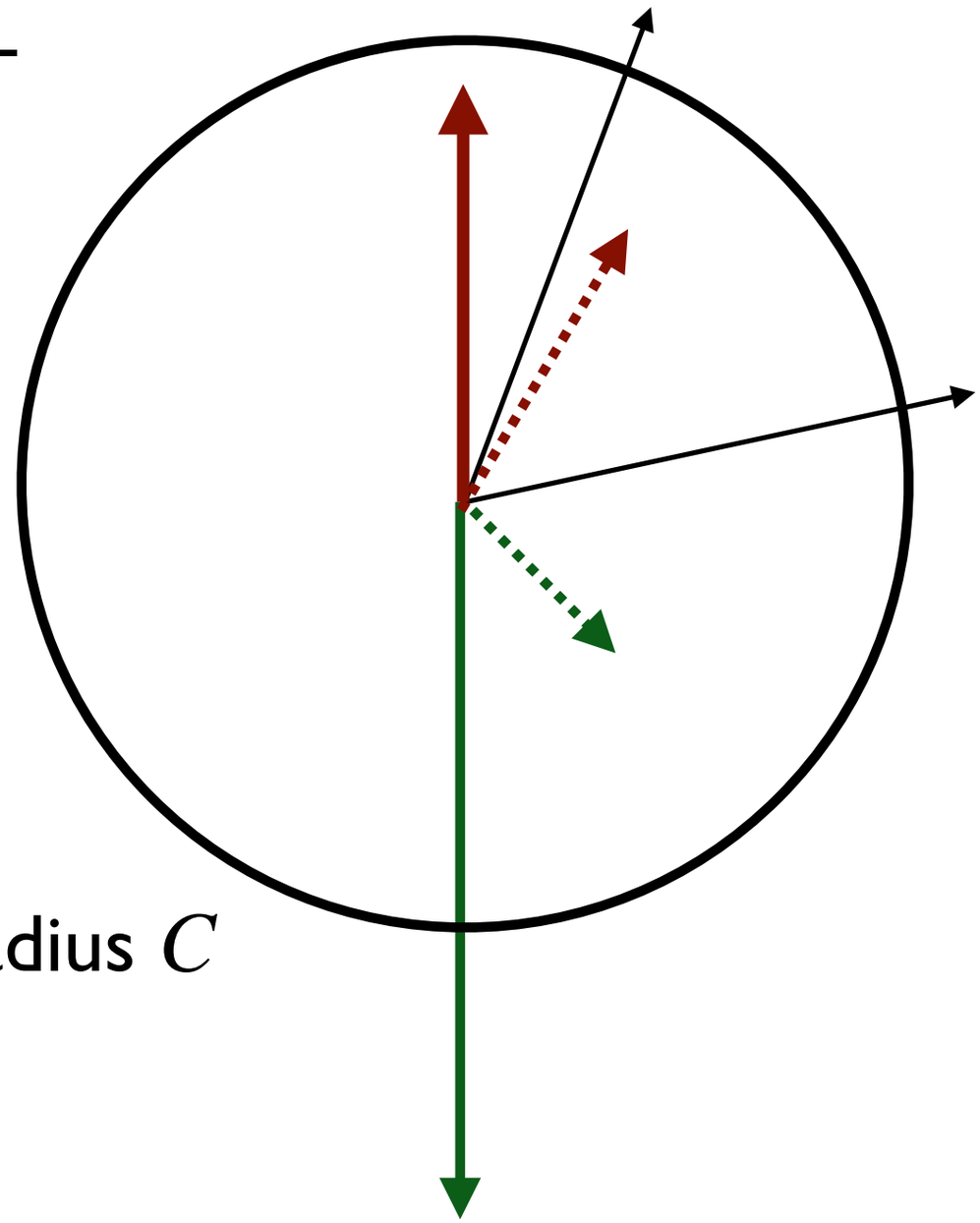
Norm clipping:

$$(x_i, y_i) = \frac{(x_i, y_i)}{||(x_i, y_i)||_2}$$

bound the length of each vector, if $||(x_i, y_i)||_2 > C$

Mini-Batch:

$\{(x_1, y_1), (x_2, y_2), (x'_3, y'_3)\} \subset D$



Mini-Batch:

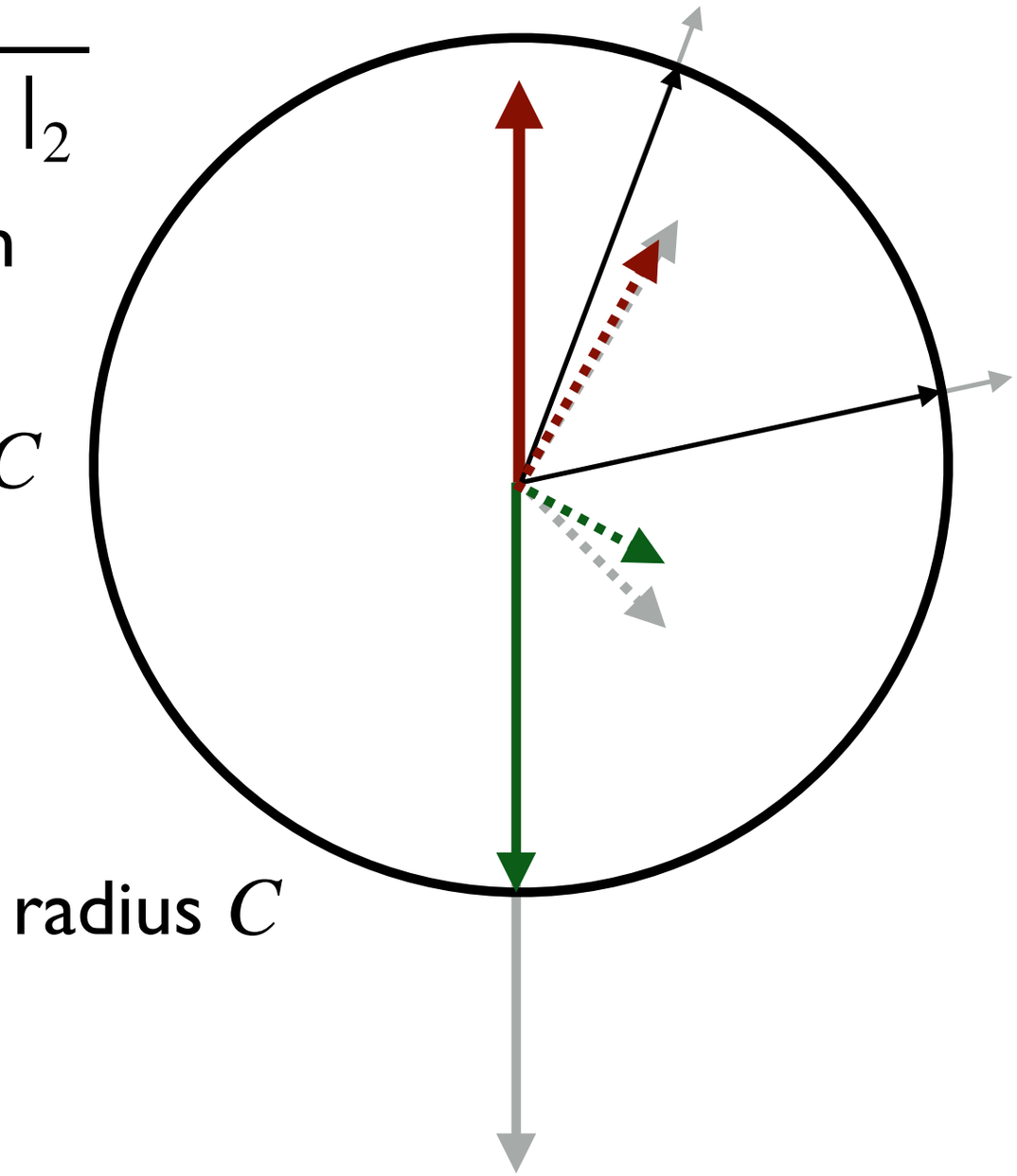
$\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$
$\subset D \cup \{(x_3, y_3)\}$

Mini-Batch SGD: Norm Clipping

Norm clipping:

$$(x_i, y_i) = \frac{(x_i, y_i)}{||(x_i, y_i)||_2}$$

bound the length of each vector, if $||(x_i, y_i)||_2 > C$



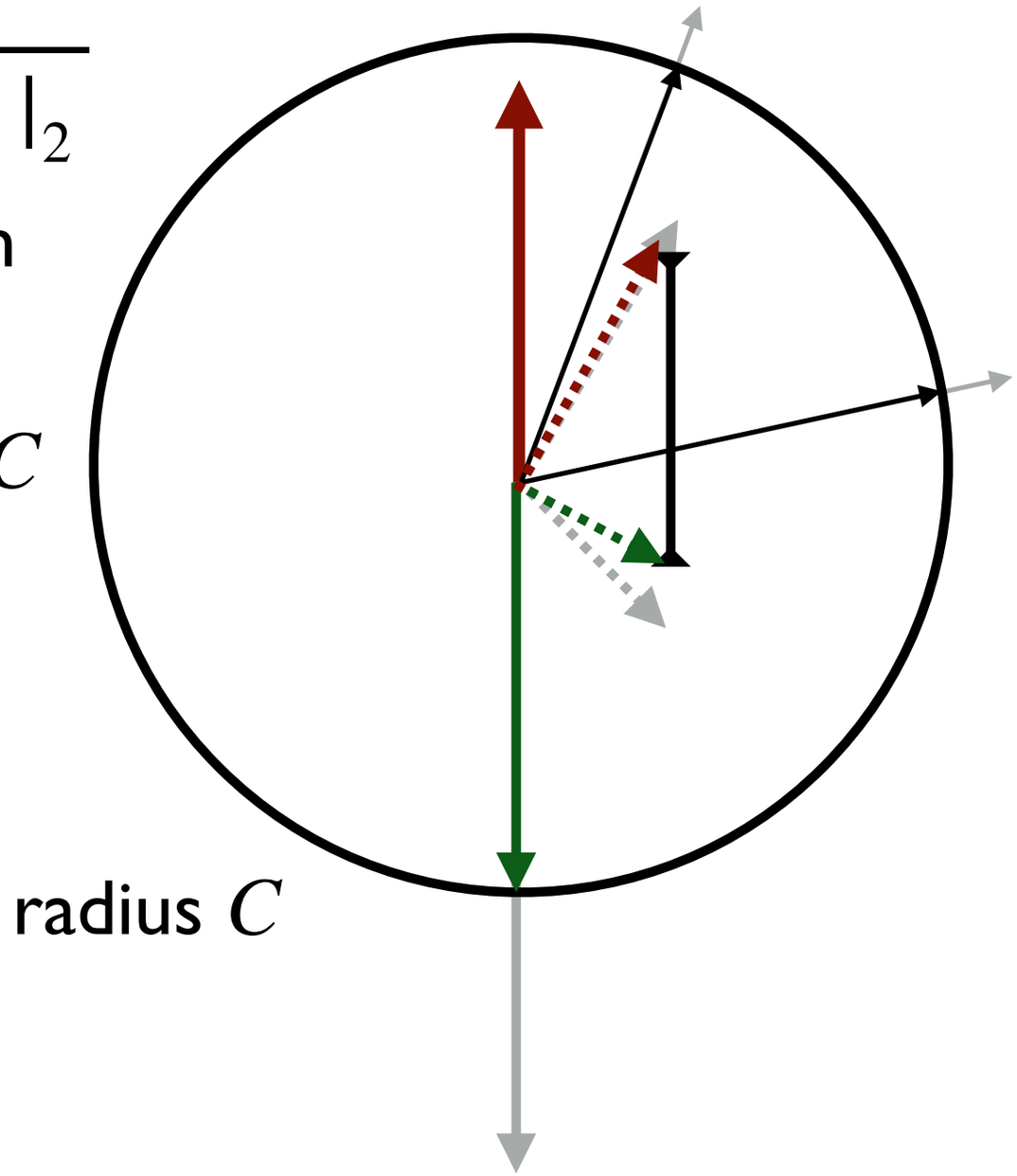
Mini-Batch SGD: Norm Clipping

Norm clipping:

$$(x_i, y_i) = \frac{(x_i, y_i)}{||(x_i, y_i)||_2}$$

bound the length of each vector,

if $||(x_i, y_i)||_2 > C$



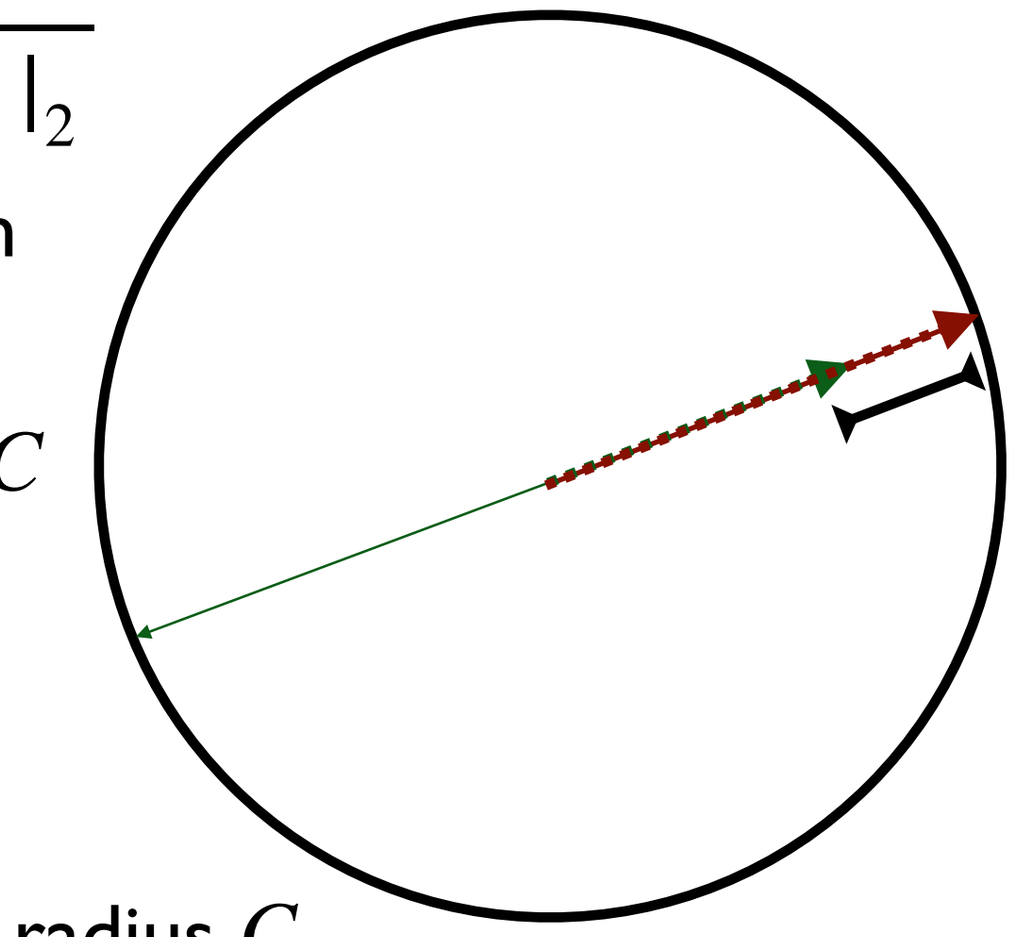
Sensitivity Δ?

Mini-Batch SGD: Bounded Sensitivity

Norm clipping:

$$(x_i, y_i) = \frac{(x_i, y_i)}{||(x_i, y_i)||_2}$$

bound the length of each vector, if $||(x_i, y_i)||_2 > C$



How to blur this difference Δ?

Sensitivity

$$2C/k = 2C/3 = \Delta$$

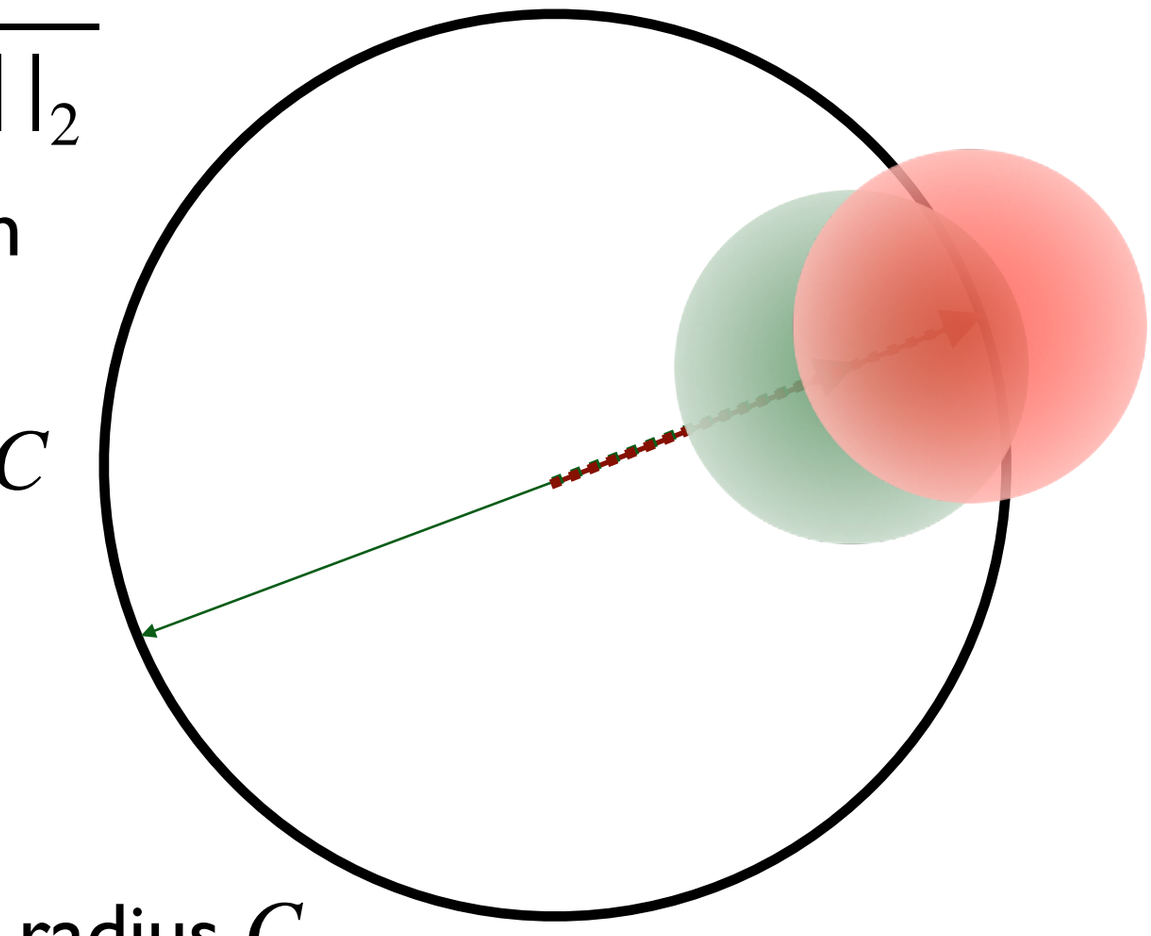
k is the size of the mini-batch

Noisy SGD

Norm clipping:

$$(x_i, y_i) = \frac{(x_i, y_i)}{||(x_i, y_i)||_2}$$

bound the length of each vector, if $||(x_i, y_i)||_2 > C$



Sensitivity

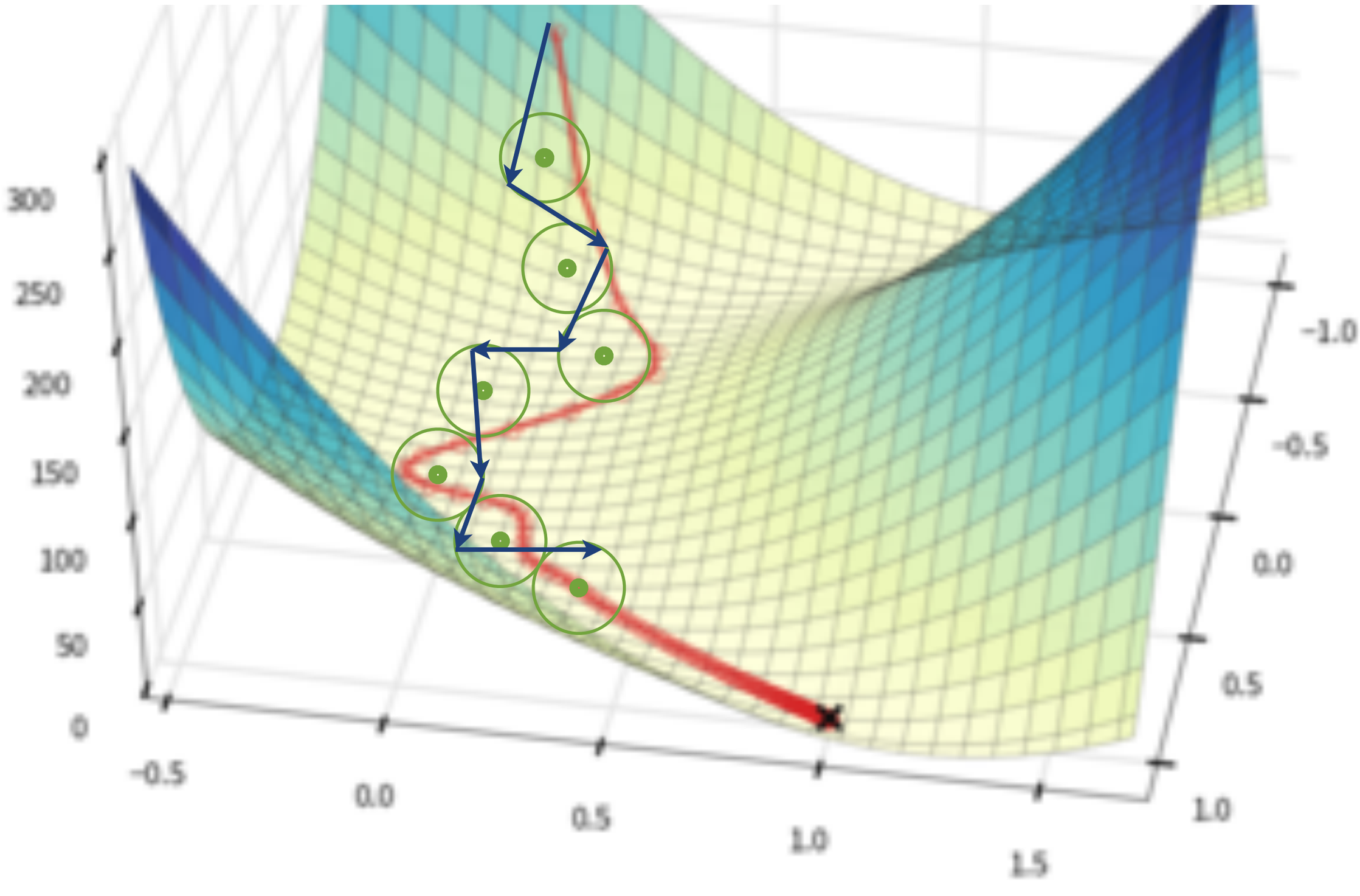
$2C/k = \Delta$

$\tilde{\varepsilon}$ degree of noise

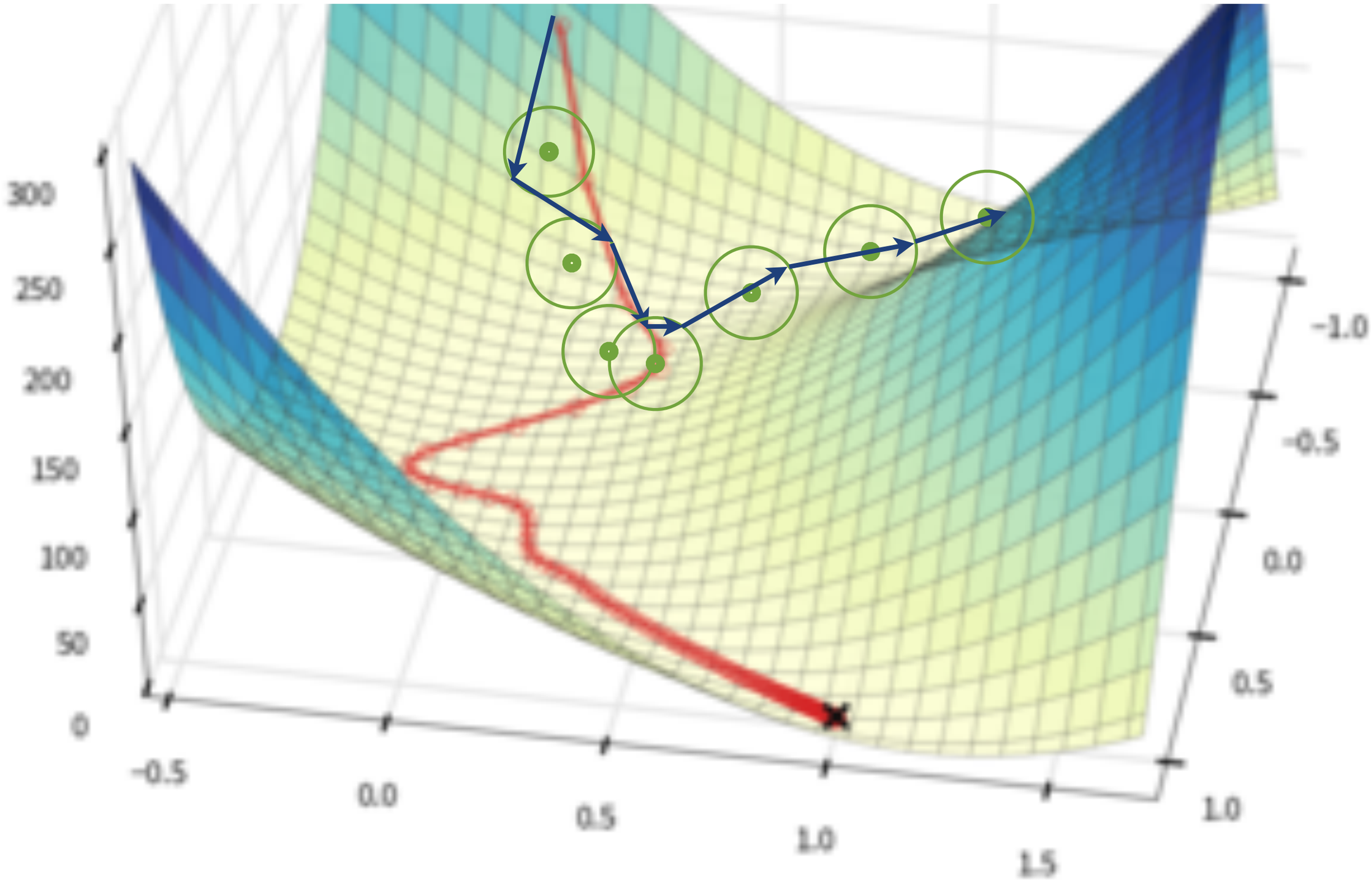
Perturb the update with (Gaussian Noise)

$\mathcal{N}(0, \tilde{\varepsilon} \cdot 2C/k)$

An Example Run: Drunken SGD



An Example Run: Takes Various Turns



Outline

- Traces of training data in ANNs
- How to formulate privacy?
- Private learning
- Other learning techniques

Other Learning Techniques

- Other learning approaches
  - Is it easier to prove DP for Bayesian learning approaches?
    - There is a paper that proves posterior sampling satisfies DP (backup slides)
  - e.g., random ferns: count contexts in which data occurs
    - counting queries lead to good DP guarantees
- Other counting-based ML techniques?

DP on Atomic Events

- Corollary: If for a mechanism $M : A \to RV(B)$ for all $o \in [X]$ we have

$$\exp(-\varepsilon) \leq \frac{pdf_{M(D \cup \{x\})}(o)}{pdf_{M(D)}(o)} \leq \exp(\varepsilon)$$

then for all $S \subseteq [X]$

$$\exp(-\varepsilon) \leq \frac{\Pr[M(D \cup \{x\}) \in S]}{\Pr[M(D) \in S]} \leq \exp(\varepsilon)$$

recall that

$$\Pr[M(D) \in S] = \int_S pdf_{M(D)}(x)dx$$

Think of S as tests.

A Bayesian View on Differential Privacy

- Recall Bayesian statistics
  - update prior belief about a hypothesis with the likelihood of the hypothesis after an observation
  - Normalized with marginalized observation distribution (often ignored since independent of hypothesis)

$$\Pr[A \mid B] \cdot \Pr[B]$$
$$= \Pr[A \wedge B]$$
$$= \Pr[A] \cdot \Pr[B \mid A]$$

$$\Pr[A \mid B]$$
$$= \frac{\Pr[A] \cdot \Pr[B \mid A]}{\Pr[B]}$$

Bayes' rule

$$\underbrace{\Pr[\mathit{hypothesis} \mid \mathit{observation}]}_{\text{posterior belief}} = \frac{\underbrace{\Pr[\mathit{hypothesis}]}_{\text{prior belief}} \cdot \underbrace{\Pr[\mathit{observation} \mid \mathit{hypothesis}]}_{\text{likelihood}}}{\underbrace{\Pr[\mathit{observation}]}_{\text{marginalized observation distribution}}}$$

Bayesian Hypothesis Testing

- Bayesian hypothesis testing
  - update prior odds with likelihood ratio of an observation
- Likelihood ratio: the gained knowledge of an observation

$$\underbrace{\frac{\Pr[hypothesis \mid observation]}{\Pr[\neg hypothesis \mid observation]}}_{\text{posterior odds}} = \frac{\Pr[observation \mid hypothesis] \cdot \Pr[hypothesis] \cdot \Pr[observation]}{\Pr[observation] \cdot \Pr[observation \mid \neg hypothesis] \cdot \Pr[\neg hypothesis]}$$

$$= \underbrace{\frac{\Pr[hypothesis]}{\Pr[\neg hypothesis]}}_{\text{prior odds}} \cdot \underbrace{\frac{\Pr[observation \mid hypothesis]}{\Pr[observation \mid \neg hypothesis]}}_{\text{likelihood ratio}}$$

A Bayesian View on Differential Privacy

- For a mechanism M, e.g., $M(D) \sim q(D) + N(0,\boldsymbol{\sigma}^2)$
- For any dataset D' and row t with
  $D' \cup \{t\} \leftarrow D$ with $1/2$ probability and
  $D' \leftarrow D$ with $1/2$ probability
- For any test S and observation $M(D) \in S$ and hypothesis $t \in D$

$$\frac{\Pr[M(D) \in S \mid t \in D]}{\Pr[M(D) \in S \mid t \notin D]} \leq e^{\varepsilon}$$

Differential Privacy

bound likelihood ratio

$$\underbrace{\frac{\Pr[t \in D \mid M(D) \in S]}{\Pr[t \notin D \mid M(D) \in S]}}_{\text{posterior odds}}$$

$$= \frac{\Pr[M(D) \in S \mid t \in D] \cdot \Pr[t \in D] \cdot \Pr[M(D) \in S]}{\Pr[M(D) \in S] \cdot \Pr[M(D) \in S \mid t \notin D] \cdot \Pr[t \notin D]}$$

$$= \underbrace{\frac{\Pr[t \in D]}{\Pr[t \notin D]}}_{\text{prior odds}} \cdot \underbrace{\frac{\Pr[M(D) \in S \mid t \in D]}{\Pr[M(D) \in S \mid t \notin D]}}_{\text{likelihood ratio}}$$

A Bayesian View on Differential Privacy

- For a mechanism M, e.g., $M(D) \sim q(D) + N(0,\sigma^2)$
- For any dataset D' and row t with
  $D' \cup \{t\} \leftarrow D$ with $1/2$ probability and
  $D' \leftarrow D$ with $1/2$ probability
- For any test S and observation $M(D) \in S$ and hypothesis $t \in D$

➡ Differential privacy bounds the **knowledge gained** (the likelihood ratio) from **any observation** for **any prior**

$$\underbrace{\frac{\Pr[t \in D \mid M(D) \in S]}{\Pr[t \notin D \mid M(D) \in S]}}_{\text{posterior odds}} = \underbrace{\frac{\Pr[t \in D]}{\Pr[t \notin D]}}_{\text{prior odds}} \cdot \underbrace{\frac{\Pr[M(D) \in S \mid t \in D]}{\Pr[M(D) \in S \mid t \notin D]}}_{\text{likelihood ratio}} \leq \frac{\Pr[t \in D]}{\Pr[t \notin D]} \cdot e^{\varepsilon}$$

Thank you!

- Attacks on ML methods
- Privacy notions
- Distributions over gradient descent paths
- ML methods inherently more privacy-preserving