EPFL

SPRING
SECURITY AND PRIVACY ENGINEERING LABORATORY

CS-523 Advanced topics on Privacy Enhancing Technologies

Privacy-preserving data publishing (Part II)

Theresa Stadler
SPRING Lab
theresa.stadler@epfl.ch

Introduction
Differential privacy

Course aim: learn **toolbox for privacy engineering**

tool
to publish aggregates with formal privacy guarantees

mechanism
to evaluate privacy

Application Layer

Network Layer

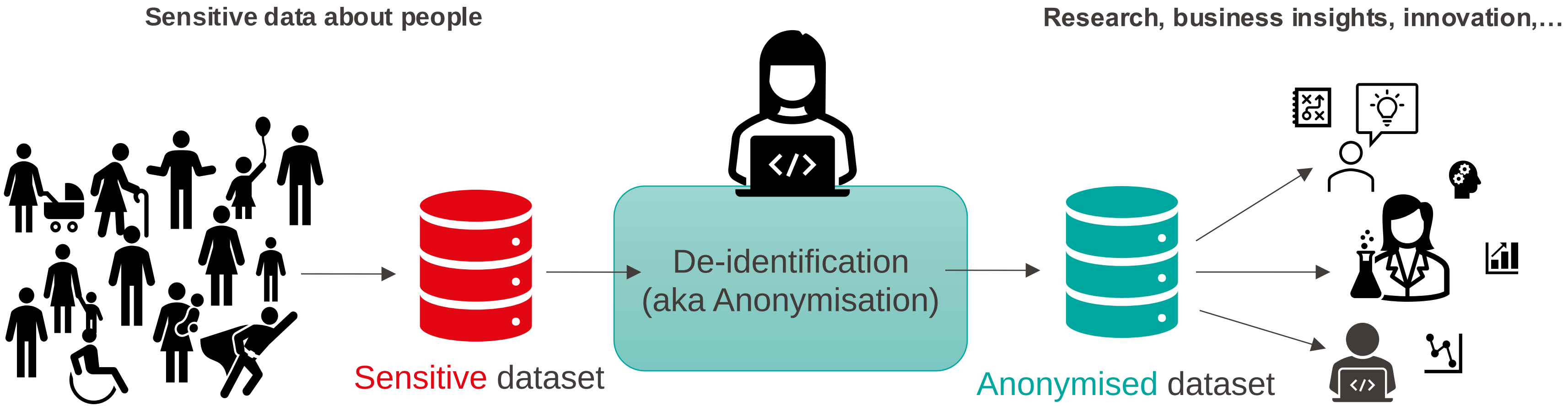
Goals

What should you learn today?

- Basic understanding of **differential privacy** and **its key properties**
  - Composition
  - Post-processing
- Understand **the meaning of ε** and how to use it to measure privacy loss
- Understand basic methods to **achieve differential privacy**
- Understand **practical issues** when using differential privacy

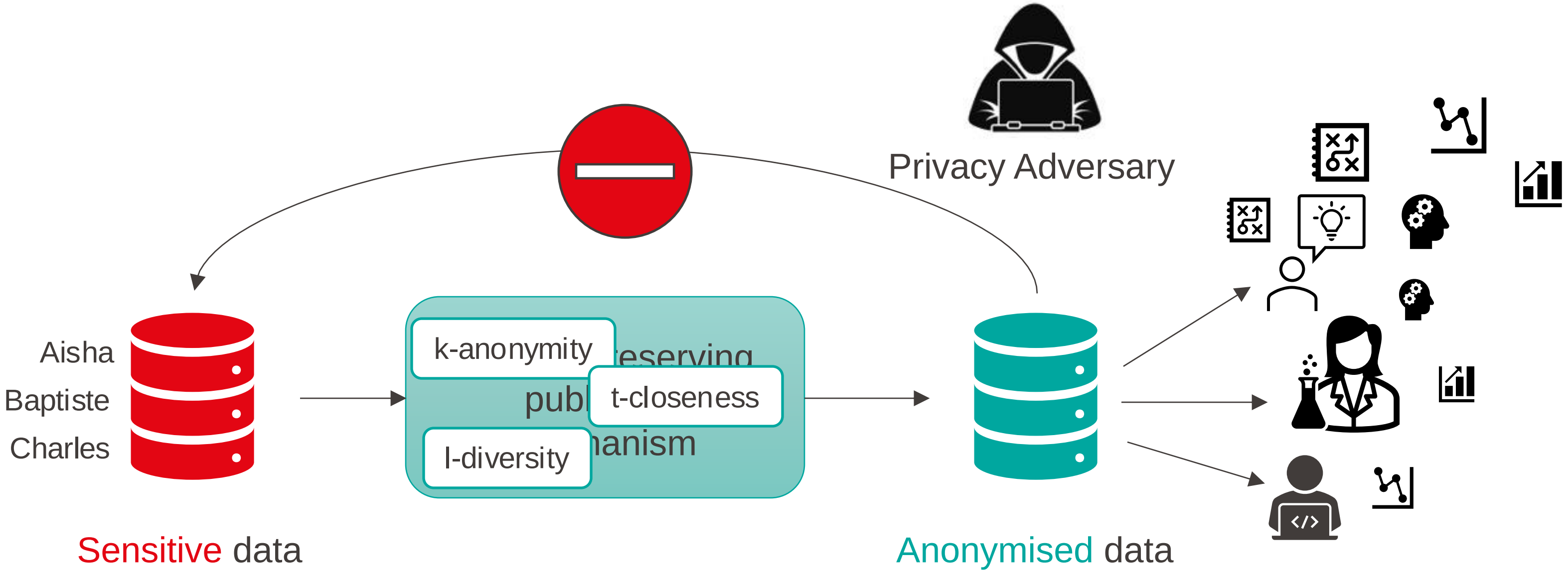
Privacy-preserving microdata sharing

Recap

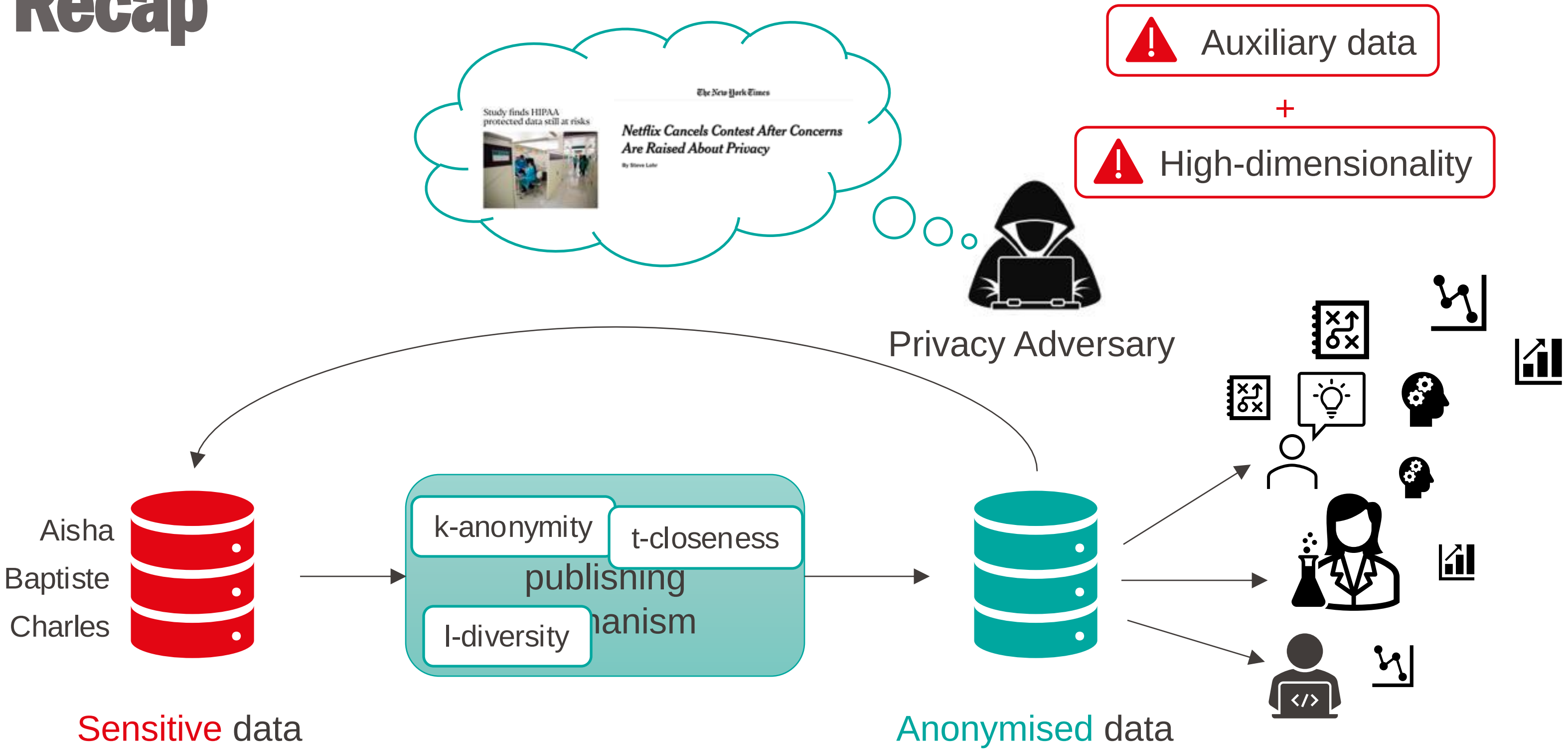


Mask or Remove Personally Identifiable Information (PII):
name, SSN, phone number, address, email, twitter handle,…

Privacy-preserving microdata sharing
Recap

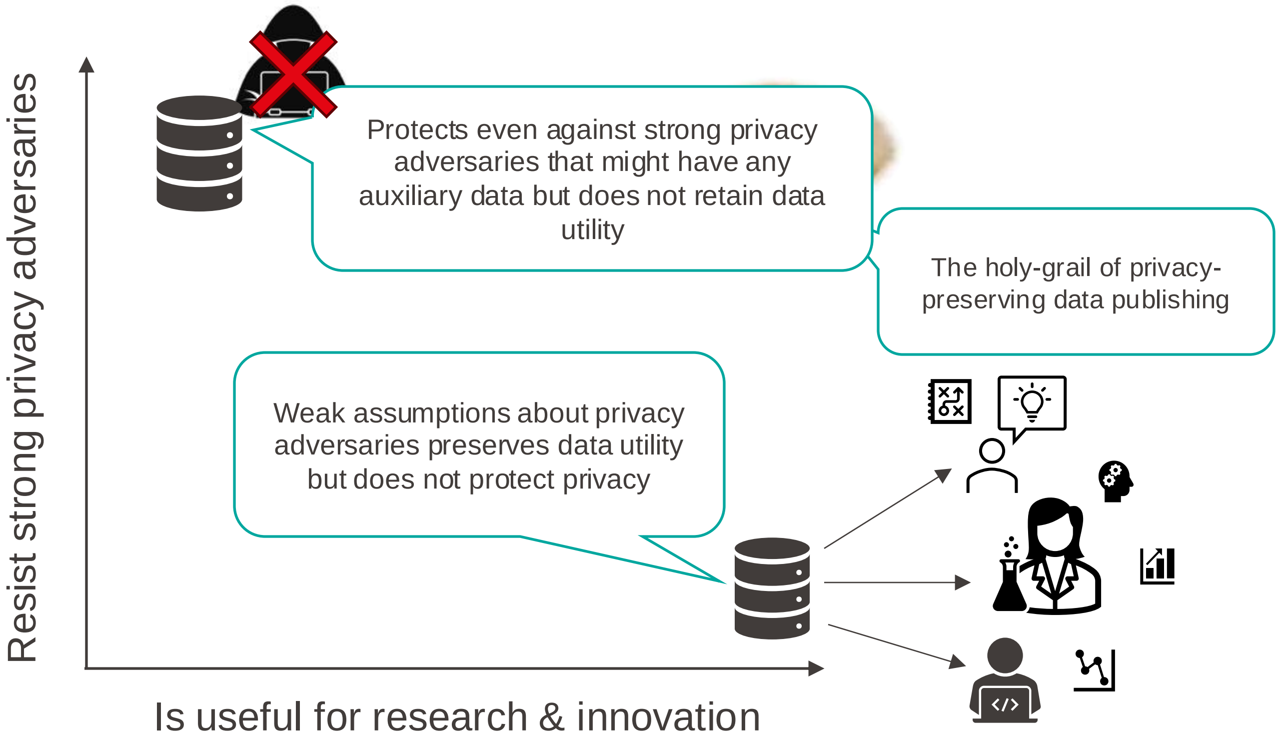


Privacy-preserving microdata sharing
Recap



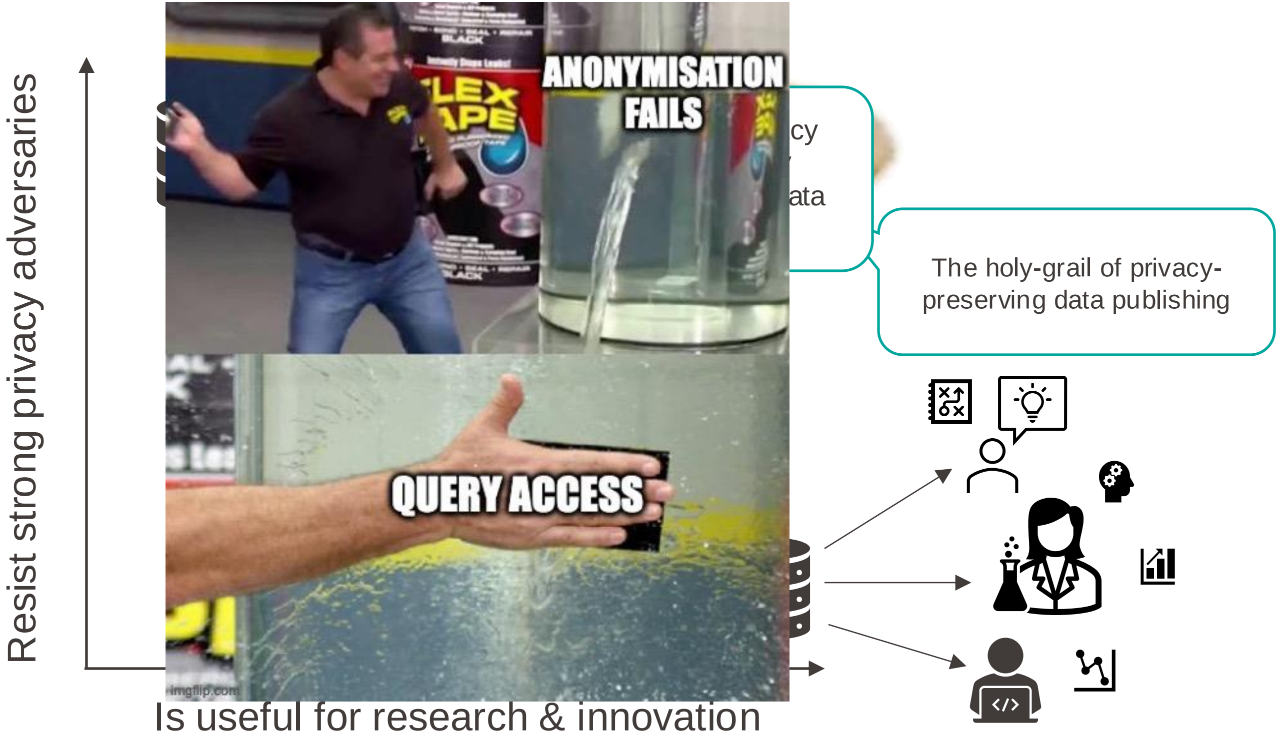
The privacy-utility trade-off

Recap



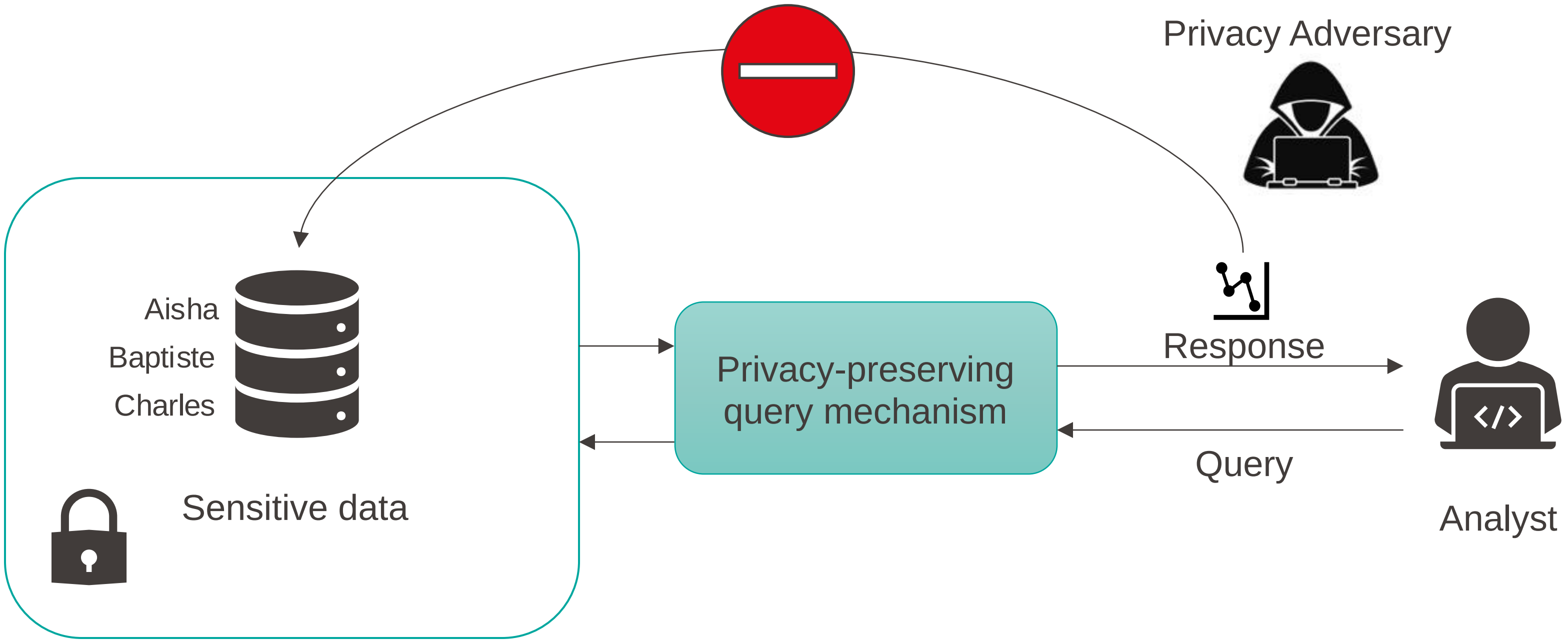
The privacy-utility trade-off

Recap



Aggregate Data Publishing

Change of paradigm: Query access



Aggregate Data Publishing
Differencing Attacks

Have we solved the privacy problem if we just switch to query access?

Name	Commune	COVID+
Alice	1025	Yes
Baptiste	1026	Yes
Charles	1028	No
David	1023	No
Eric	1025	No
Francois	1015	Yes
Geraldine	1028	No

Privacy-preserving query mechanism

Response Q2: 2

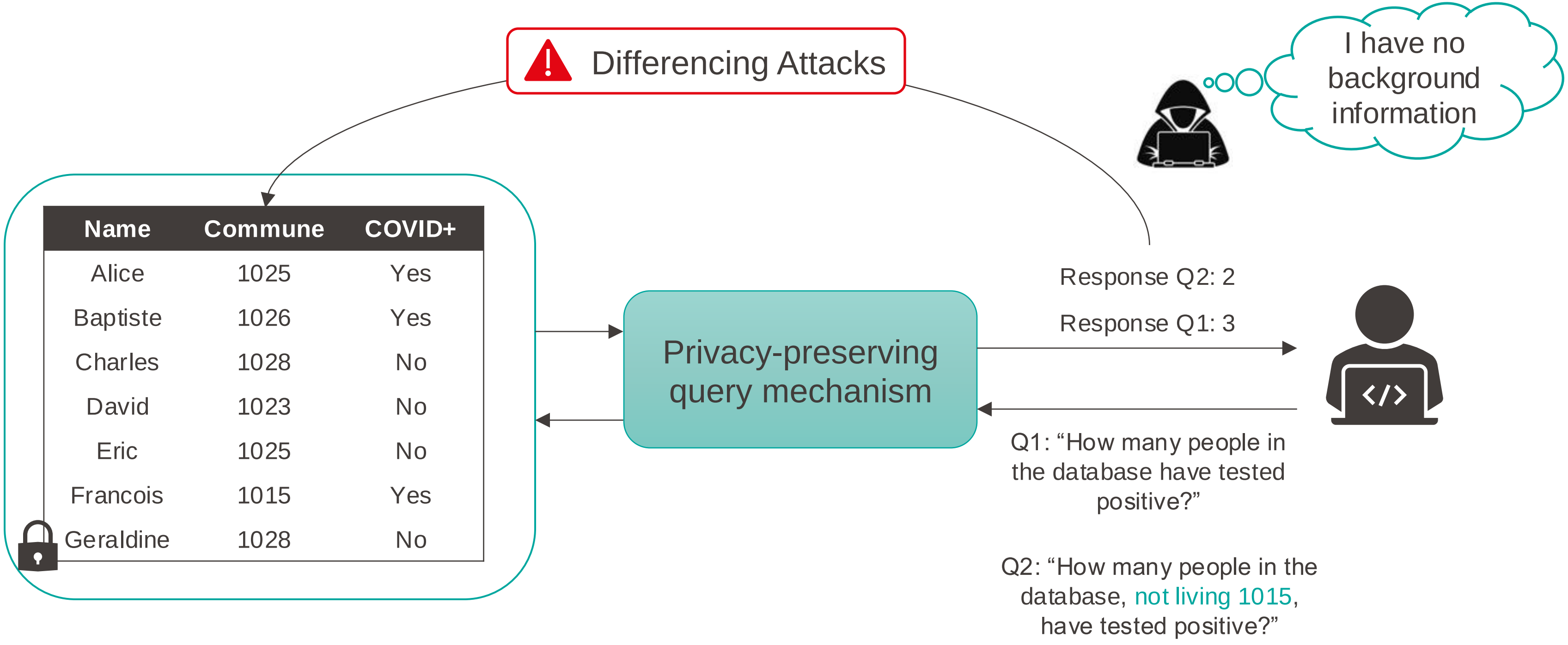
Response Q1: 3

Q1: “How many people in the database have tested positive?”

Q2: “How many people in the database, not living 1015, have tested positive?”

Aggregate Data Publishing
Differencing Attacks

Have we solved the privacy problem if we just switch to query access?



Name	Commune	COVID+
Alice	1025	Yes
Baptiste	1026	Yes
Charles	1028	No
David	1023	No
Eric	1025	No
Francois	1015	Yes
Geraldine	1028	No

Aggregate Data Publishing
Differencing Attacks

Have we solved the privacy problem if we just switch to query access?

Differencing Attacks

“François is the only one who lives in 1015.”

Name	Commune	COVID+
Alice	1025	Yes
Baptiste	1026	Yes
Charles	1028	No
David	1023	No
Eric	1025	No
François	1015	Yes
Geraldine	1028	No

Privacy-preserving query mechanism

Response Q2: 2

Response Q1: 3

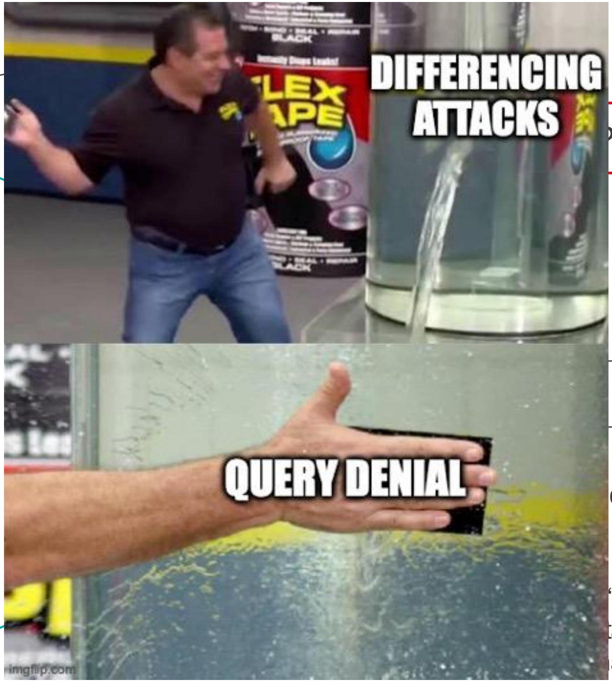
Q1: “How many people in the database have tested positive?”

Q2: “How many people in the database, not living 1015, have tested positive?”

Aggregate Data Publishing
Differencing Attacks

Have we solved the privacy problem if we just switch to query access?

Name	Commune	COVID+
Alice	1025	Yes
Baptiste	1026	Yes
Charles	1028	No
David	1023	No
Eric	1025	No
Francois	1015	Yes
Geraldine	1028	No



"Francois is the only one who lives in 1015."

Response Q2: 2

Response Q1: 3

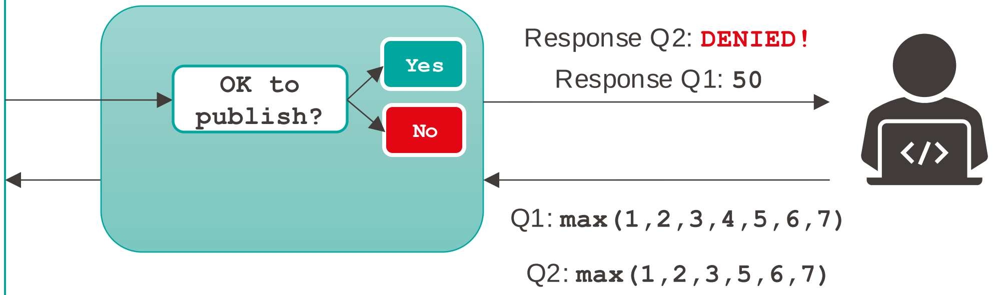
"How many people in database have tested positive?"

'How many people in the abase, not living 1015, ave tested positive?"

Aggregate Data Publishing
Query Auditing

Have we solved the privacy problem if we just switch to query access with query auditing?

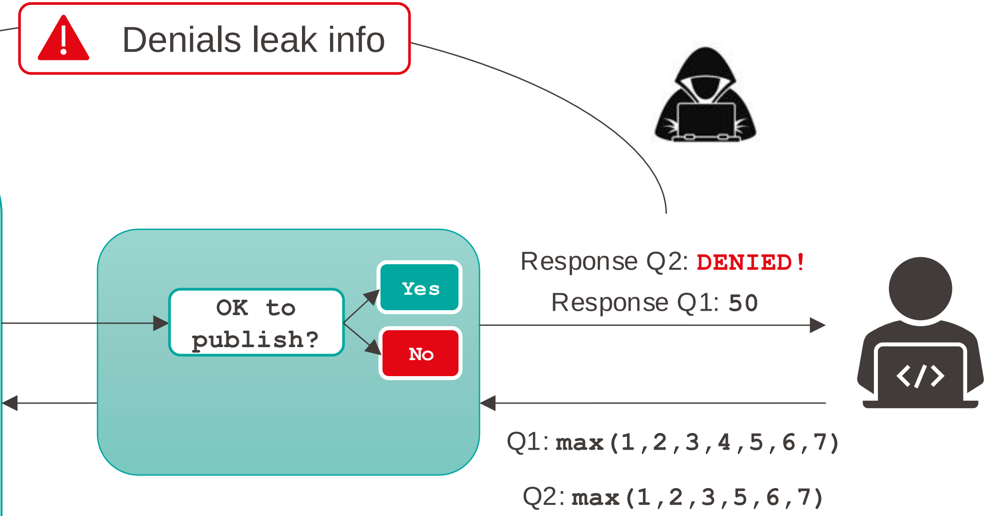
Name	Index	Secret
Alice	1	10
Baptiste	2	13
Charles	3	1
David	4	50
Eric	5	8
Francois	6	23
Geraldine	7	36



Aggregate Data Publishing
Query Auditing

Have we solved the privacy problem if we just switch to query access with query auditing?

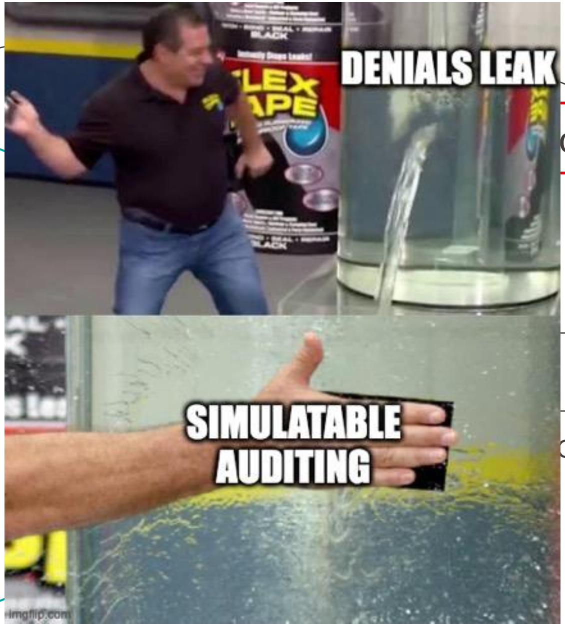
Name	Index	Secret
Alice	1	10
Baptiste	2	13
Charles	3	1
David	4	50
Eric	5	8
Francois	6	23
Geraldine	7	36



Aggregate Data Publishing
Query Auditing

Have we solved the privacy problem if we just switch to query access with query auditing?

Name	Index	Secret
Alice	1	10
Baptiste	2	13
Charles	3	1
David	4	50
Eric	5	8
Francois	6	23
Geraldine	7	36



on?

Response Q2: **DENIED!**

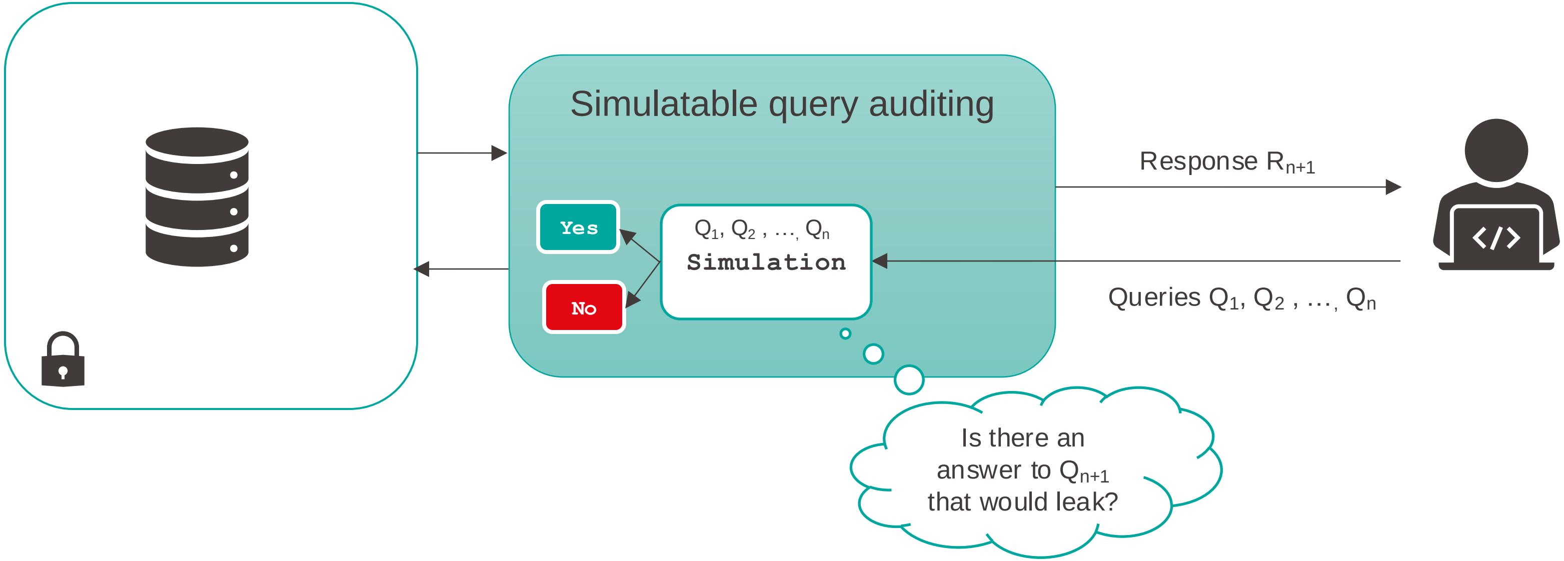
Response Q1: **50**

Q1: `max(1,2,3,4,5,6,7)`

Q2: `max(1,2,3,5,6,7)`

Aggregate Data Publishing
Query Auditing

Have we solved the privacy problem if we just switch to query access with simulatable query auditing?



EPFL

Aggregate Data Publishing
Query Auditing

- Audits are limited to a fixed privacy definition
  - Individual (record) vs. group (record) privacy
  - Rely on heuristics
- Algorithmic limitations
  - Secure deniability implies using algorithms computationally prohibitive
  - Feasible methods focused on simple queries
- Utility loss not quantifiable
  - Literature uses percentage of denials but this may not be representative
  - No good way to quantify the privacy-utility trade-off

Aggregate Data Publishing

…

How do we avoid this?

Aggregate Data Publishing

…

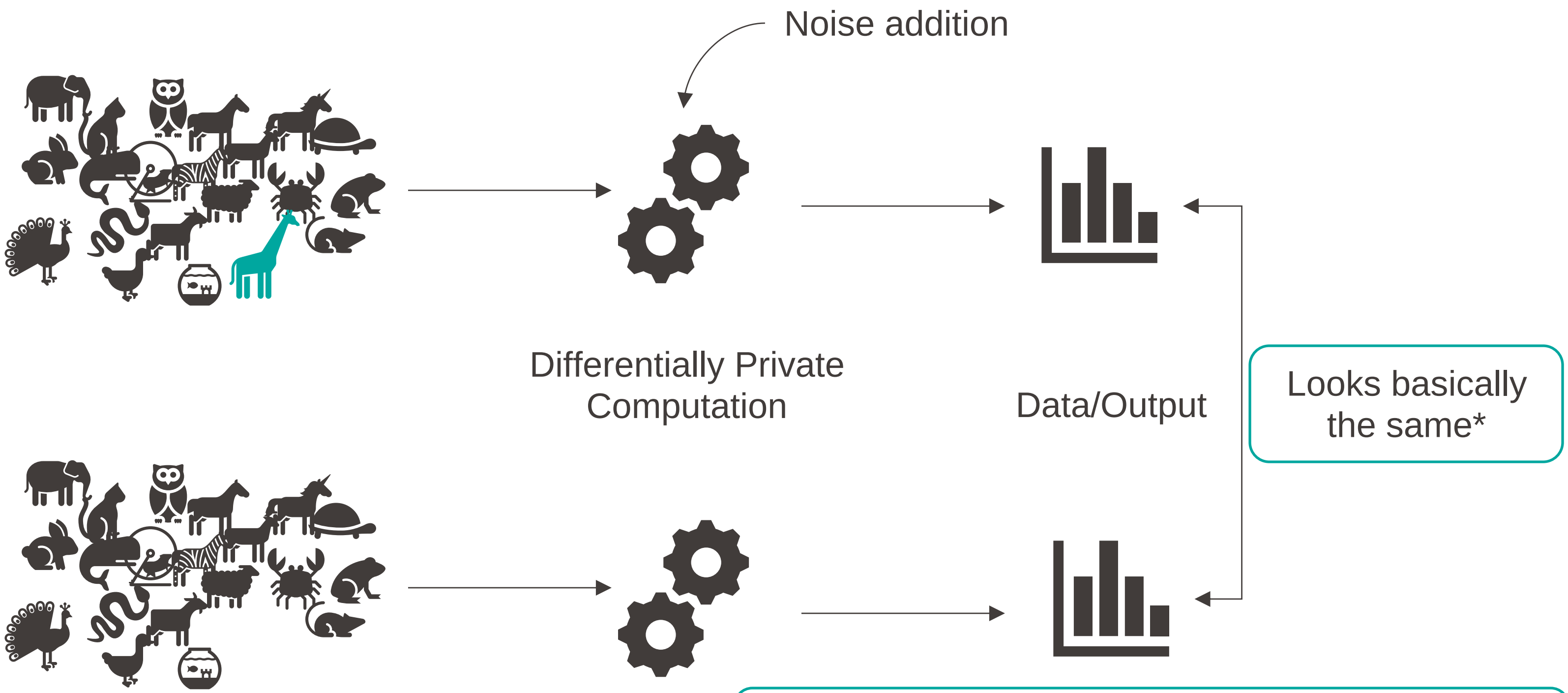
How do we avoid this?

Motivation

Differential Privacy
Motivation

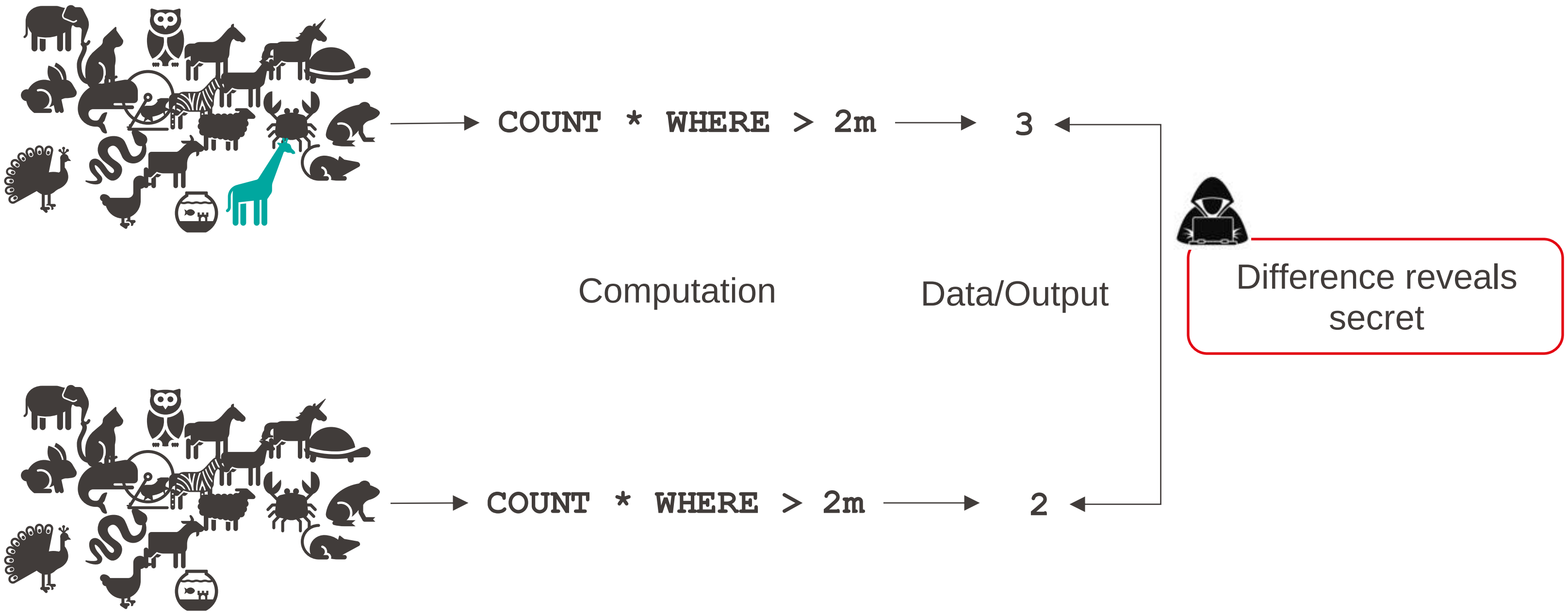
Previous Techniques	Differential Privacy
Heuristic privacy definition fixed to one adversary	Formal privacy guarantee
Repeated failures	Set the right social incentives
Utility loss not quantifiable	Quantify inherent trade-offs

Differential Privacy
Motivation

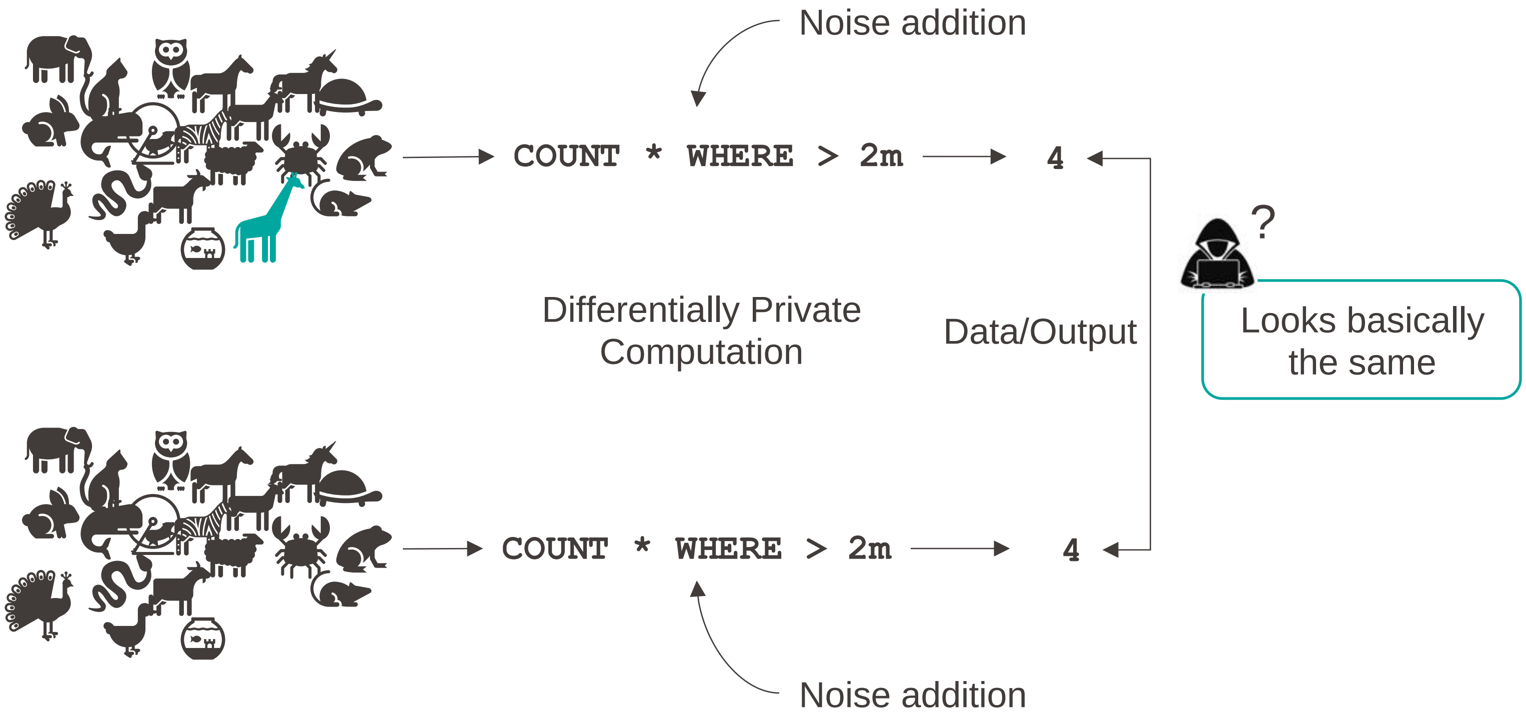


*no matter whom you removed from your database and what database you had in the first place

Differential Privacy
Motivation



Differential Privacy
Motivation



Understanding Differential Privacy

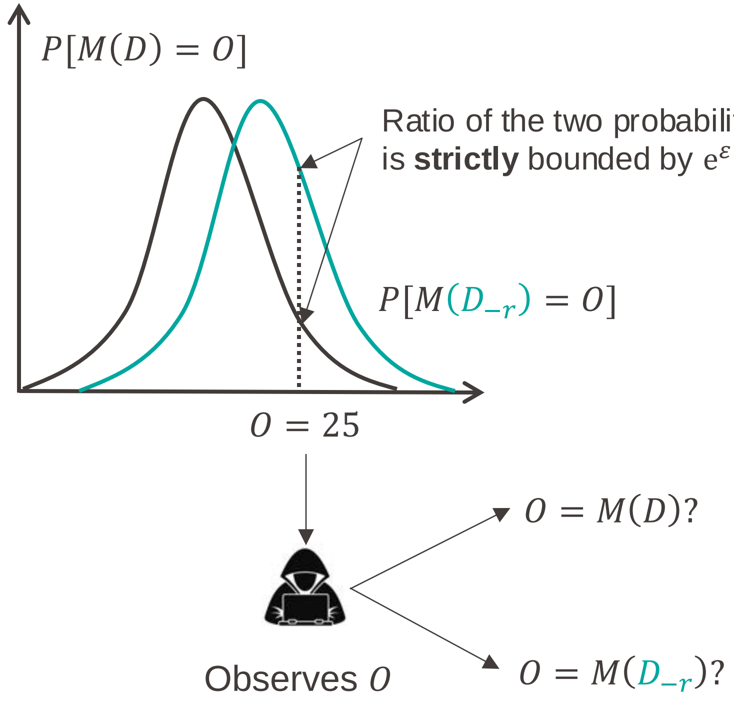
Differential Privacy
Formal Definition

A mechanism M is ε-differentially private if for all neighbouring databases D and D_{-r} which differ in only one individual

$$\mathbb{P}[M(D) = O] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D_{-r}) = O]$$

… and this must be true for all possible outputs O

Understanding Differential Privacy
The Privacy Loss



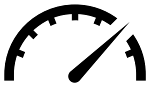
Maximal knowledge gain of the attacker

For any neighbouring databases D, D_{-r} and any possible output O

$$Privacy\ Loss = \log \frac{P[M(D) = O]}{P[M(D_{-r}) = O]} < \varepsilon$$

Understanding Differential Privacy
The Privacy Budget

$$Privacy\ Loss = \log \frac{P[M(D) = O]}{P[M(D_{-r}) = O]} < \varepsilon$$

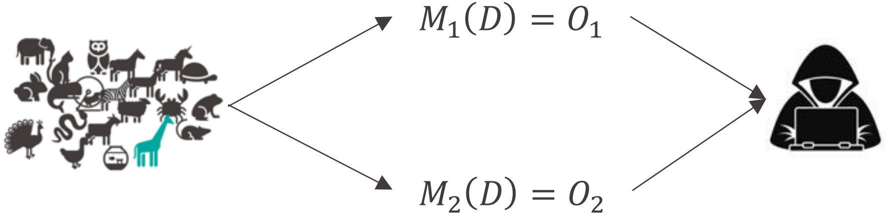


Privacy as a consumable resource The parameter ε measures leakage and can be treated as a “privacy budget” which is consumed as analyses are performed.

Understanding Differential Privacy
Sequential Composition

Theorem: Suppose that we have k algorithms $M_1, M_2, \dots, M_k$ *where* each $\mathrm{M_i}$ satisfies ε_i -differential privacy, respectively. Consider the sequence of computations $\{O_1 = M_1(D), \dots, O_k = M_k(D, O_{k-1})\}$ run on dataset D and the auxiliary input O_i. Then the algorithm $M(D) = O_k$ is ε-differentially private with

$$\varepsilon = \varepsilon_1 + \varepsilon_2 + \dots + \varepsilon_k$$

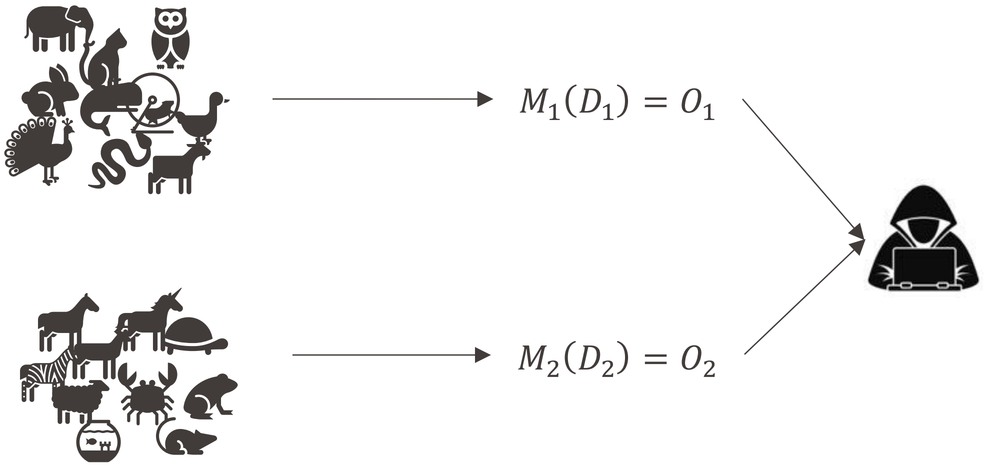


$$\log \frac{P[M_1(D) = O_1]}{P[M_1(D_{-r}) = O_1]} + \log \frac{P[M_2(D) = O_2]}{P[M_2(D_{-r}) = O_2]} < ?$$

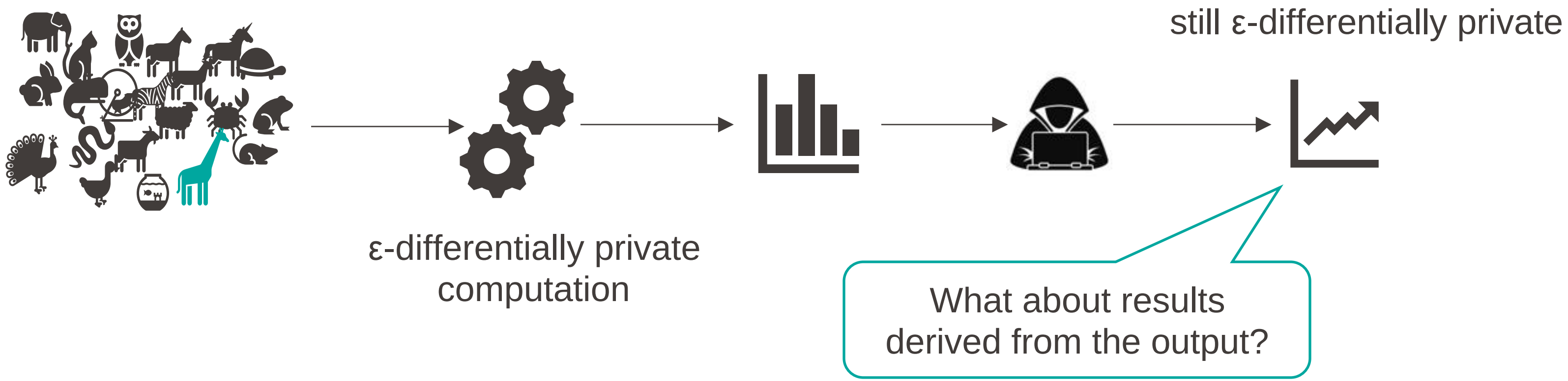
Understanding Differential Privacy
Parallel Composition

Theorem: Suppose that we have k algorithms $M_1, M_2, \dots, M_k$ *where* each $\mathrm{M_i}$ satisfies ε -differential privacy, respectively. Consider the sequence of computations $\{O_1 = M_1(D_1), \dots, O_k = M_k(D_k)\}$ where $D_1, \dots, D_k$ are k disjoint subsets of the data D. Then the algorithm $M(D) = \{O_1, \dots, O_k\}$ is ε-differentially private with

$$\varepsilon = \varepsilon_1 = \cdots = \varepsilon_k$$



Understanding Differential Privacy
Post-Processing



Differential Privacy Properties

Summary

- Formal notion of privacy that allows us to quantify the inherent privacy-utility trade-off
- Privacy loss random variable gives us a bound on the maximal advantage of the adversary
- Privacy budget ε allows to keep track of leakage
- Composition and post-processing theorems important in practice

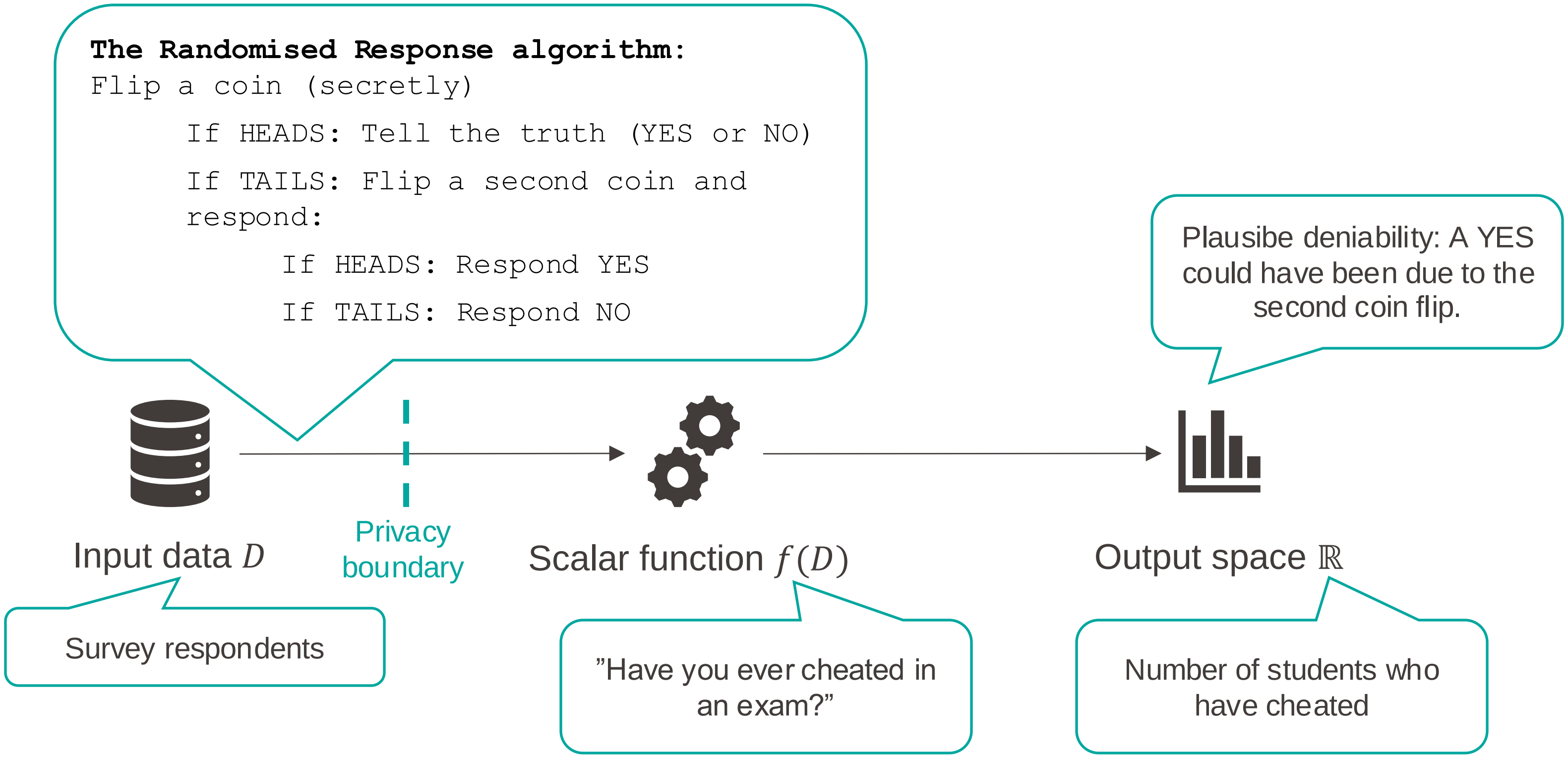
Differential privacy is a notion of privacy **not** a tool → Next part: How to achieve differential privacy

How to achieve Differential Privacy

How to achieve Differential Privacy
Overview

- **Input perturbation**
  - Add noise directly to the database (≠ perturbed dataset can be published)
    + independent of the algorithm & easy to reproduce
    - determining the amount of required noise is difficult
- **Output perturbation**
  - Add noise to the function (statistic) output
    + easier to control privacy & better guarantees than input perturbation
    - results cannot be reproduced
- **Algorithm Perturbation**
  - Inherently add noise to the algorithm
    + algorithm can be optimized with the noise addition
    - difficult to generalize & depends on the inputs

How to achieve Differential Privacy
Input Perturbation



How to achieve Differential Privacy
Input Perturbation

The Randomised Response algorithm:

```
Flip a coin (secretly)
    If HEADS: Tell the truth (YES or NO)
    If TAILS: Flip a second coin and
    respond:
        If HEADS: Respond YES
        If TAILS: Respond NO
```

The math
Assume the true answer is $\text{truth} = YES$

With probability $p = 50\%$ they will **truthfully** answer YES

With probability $p = 50\%$ they will answer **randomly**

With $p = 50\%$ the random answer is YES
With $p = 50\%$ the random answer is NO

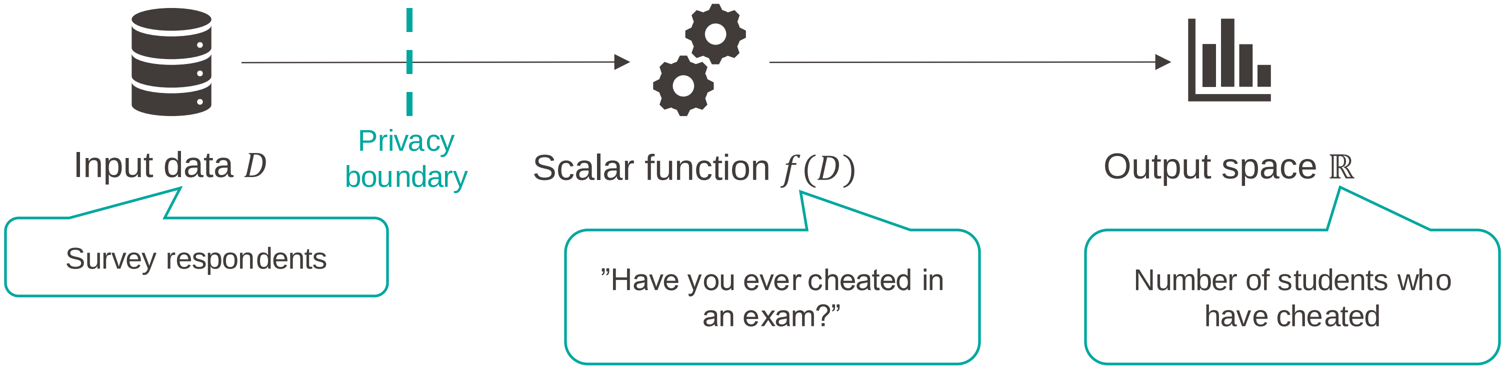
Privacy loss $$\frac{\mathbb{P}[answer{=}YES \mid truth{=}YES]}{\mathbb{P}[answer{=}YES \mid truth{=}NO]} = \frac{0.75}{0.25} = 3 = e^{\varepsilon} \rightarrow \varepsilon \sim 1.1$$

First coin TAILS
Second coin HEADS

How to achieve Differential Privacy
Input Perturbation

What about utility?

- Aggregate result is noisy
- However, if you have enough answers, with high probability, the noise will cancel itself out

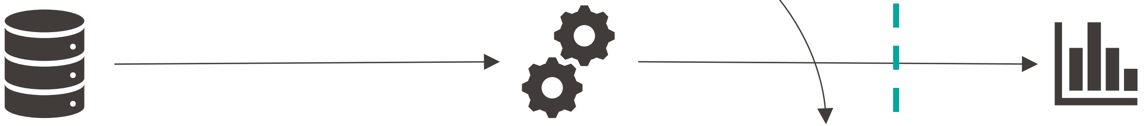


How to achieve Differential Privacy
Output Perturbation

$$\Delta f \; \coloneqq \max_{D,\, D_{-r}} |f(D) - f(D_{-r})|$$

Δf – Sensitivity of the function f

Privacy boundary



Input data D

Mechanism $\mathrm{A(D)} = f(D) + Lap(\frac{\Delta f}{\varepsilon})$

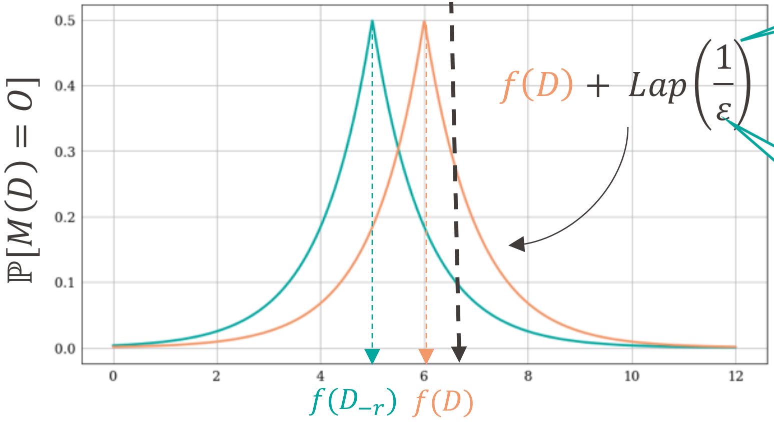
Output space $\mathbb{R}$

Output of the original computation

Noise drawn from a Laplace distribution with scale parameter $\frac{\Delta f}{\varepsilon}$

How to achieve Differential Privacy
Output Perturbation

`COUNT users WHERE rating = 0`



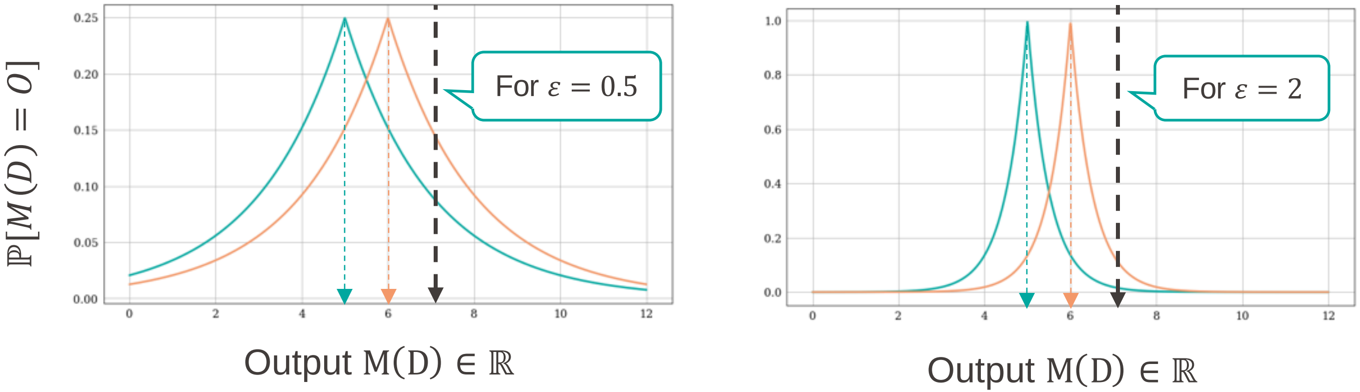
Why 1?

For $\varepsilon = 1$

Output $\mathrm{M(D)} \in \mathbb{R}$

How to achieve Differential Privacy
Output Perturbation

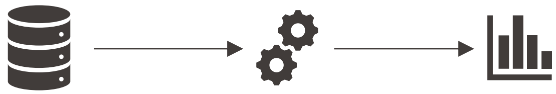
`COUNT users WHERE rating = 0`



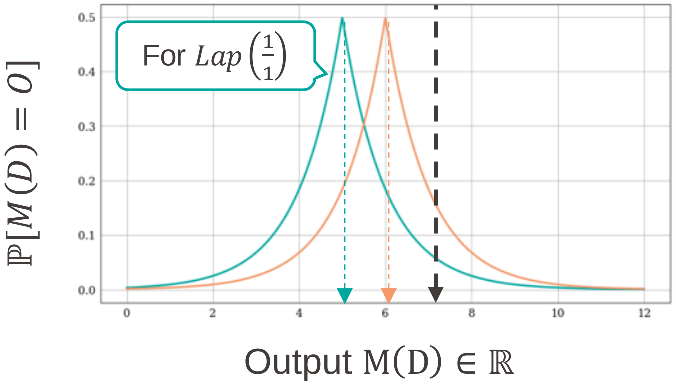
$\downarrow \varepsilon: \uparrow privacy$

$\uparrow \varepsilon: \downarrow privacy$

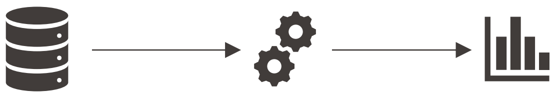
How to achieve Differential Privacy
Output Perturbation



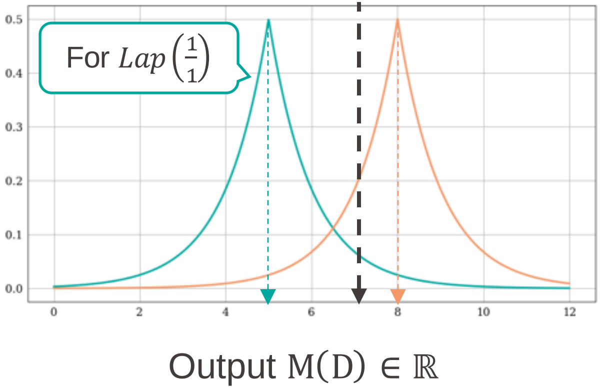
`COUNT users WHERE rating = 0`



$$\Delta f \;:=\; \max_{D,\,D_{-r}} |f(D) - f(D_{-r})| = 1$$

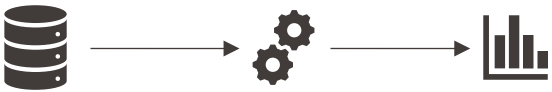


`COUNT ratings WHERE rating = 0`

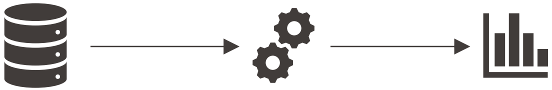


$$\Delta f \;:=\; \max_{D,\,D_{-r}} |f(D) - f(D_{-r})| = 3$$

How to achieve Differential Privacy
Output Perturbation

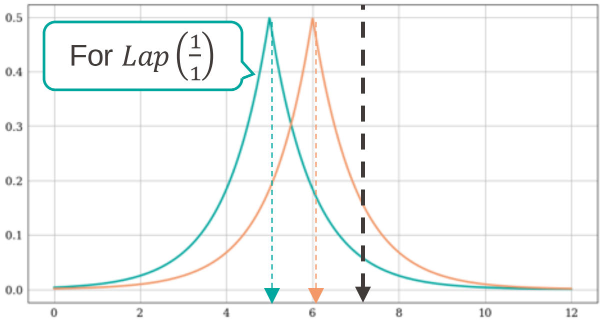


`COUNT users WHERE rating = 0`

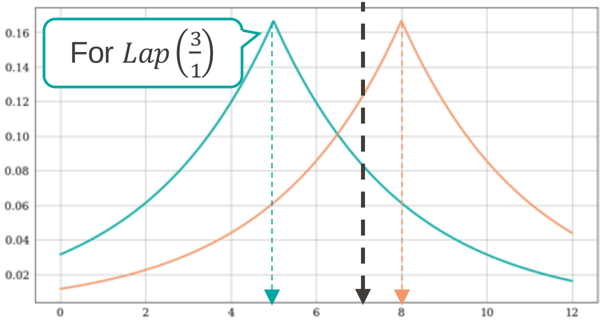


`COUNT ratings WHERE rating = 0`

$\mathbb{P}[M(D) = O]$



Output $\mathrm{M(D)} \in \mathbb{R}$



Output $\mathrm{M(D)} \in \mathbb{R}$

$$\Delta f \coloneqq \max_{D,\, D_{-r}} |f(D) - f(D_{-r})| = 1$$

$$\Delta f \coloneqq \max_{D,\, D_{-r}} |f(D) - f(D_{-r})| = 3$$

How to achieve Differential Privacy

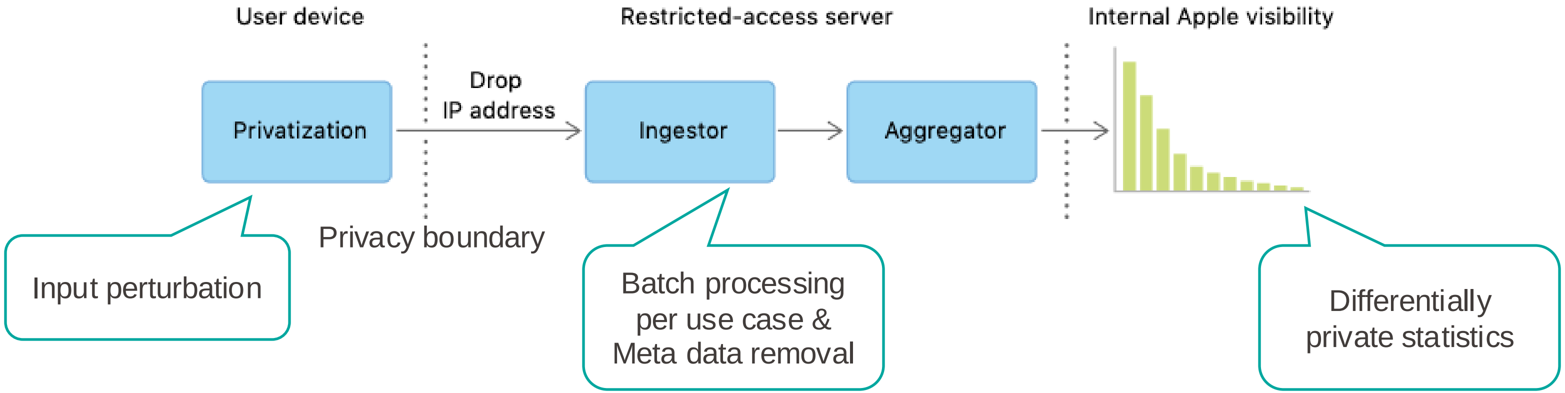
Summary

- Whether we use input or output perturbation shifts the privacy boundary
  - Input perturbation: The aggregator is **not** trusted
  - Output perturbation: Trusted aggregator.
- The randomised response algorithm is a simple way to perturb inputs that gives plausible deniability for sharing sensitive inputs and satisfies the differential privacy notion of privacy
- For output perturbation, the level of noise that is added depends on
  - Δf: The sensitivity of the computation (maximum influence a single individual can have on result)
  - ε: The privacy budget we want to spend on the computation

Differential Privacy in Practice

Differential Privacy in Practice
Input Perturbation

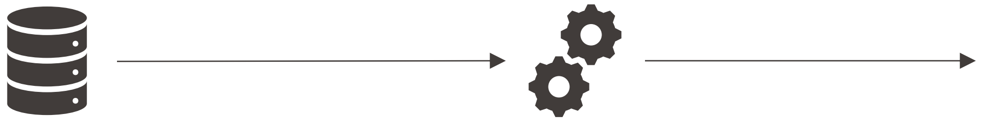
- Apple uses DP to crowdsource data from user devices (iOS, macOS) with privacy for various analytics
  - Discovering new words, popular emojis, web domains that consume high energy in Safari, etc.



- https://machinelearning.apple.com/research/learning-with-privacy-at-scale

Differential Privacy in Practice
Input Perturbation – Private Count Min Sketch

Function $f(d)\colon \mathcal{D} \rightarrow \mathbb{R}$
Counts occurrence of data element d
in dataset D



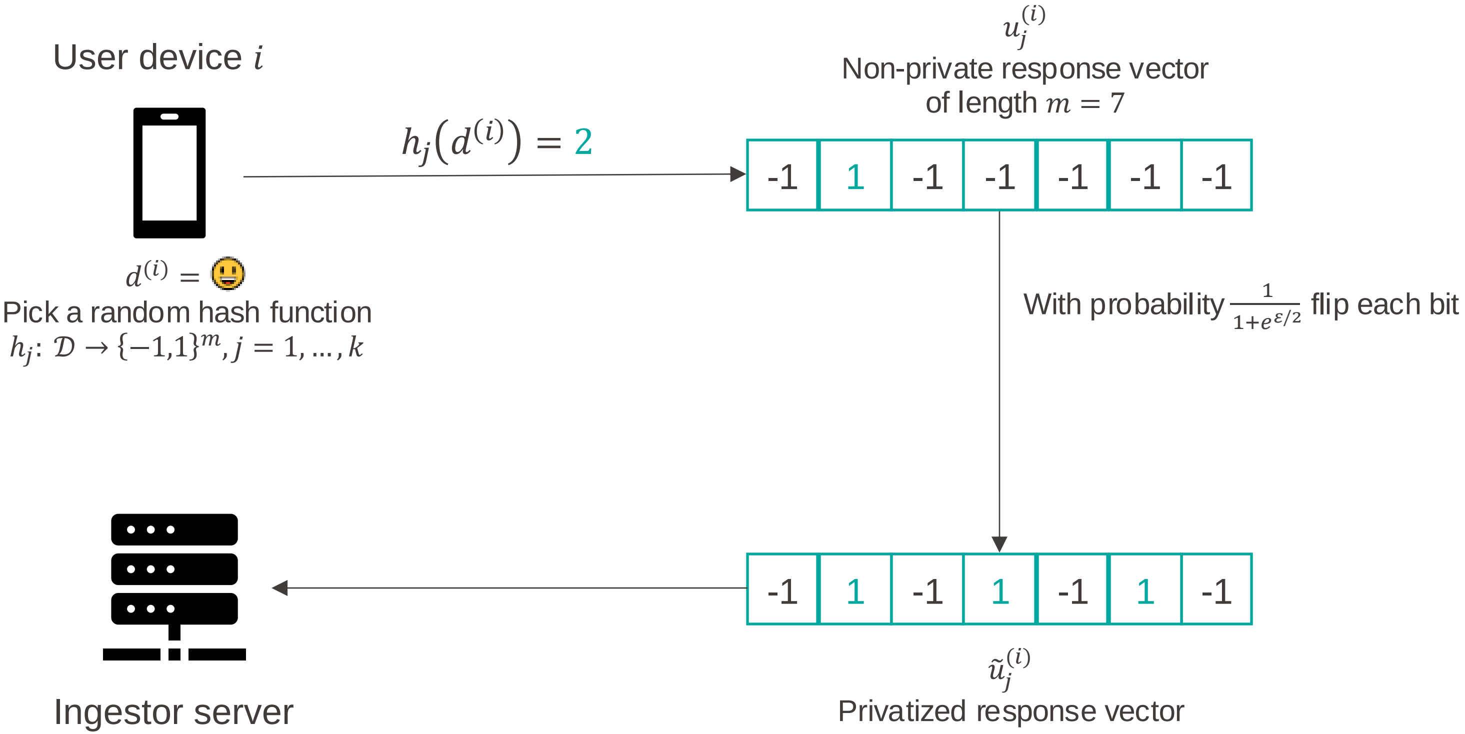
Output

Dataset $\mathrm{D} = \{$🙂, 😃, 😏, 😡, 😃, 😏, 😃, 😏$\}$ of size $\mathrm{n} = 8$
from data universe $\mathcal{D} = \{$🙂, 😃, 😏, 😡$\}$ of size $p = 4$

- https://docs-assets.developer.apple.com/ml-research/papers/learning-with-privacy-at-scale.pdf

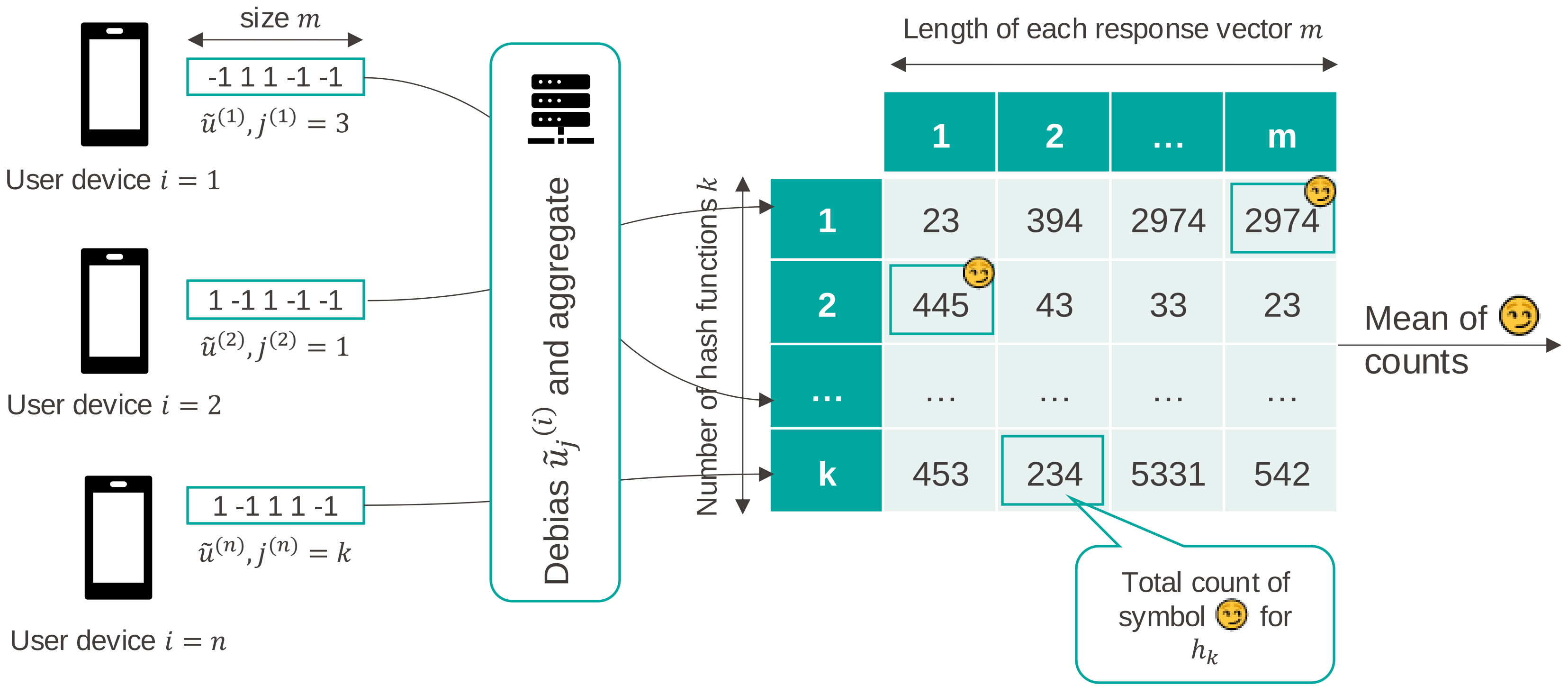
Differential Privacy in Practice
Input Perturbation – Private Count Min Sketch

Client side algorithm M_{client}



Differential Privacy in Practice
Input Perturbation

Server side algorithm M_{server}



Differential Privacy in Practice
Input Perturbation

Privacy Analysis

We need to show that $M_{client}: \mathcal{D} \rightarrow \{-1, +1\}^m$ is ε-differentially private.

Input to the algorithm is an element from the data universe $d^{(i)} \in \mathcal{D}$

Output of the algorithm is the privatised vector $\tilde{u}_j^{(i)}$

$$\Longrightarrow \ln \frac{\mathbb{P}[M_{client}(d) = \tilde{u}]}{\mathbb{P}[M_{client}(d') = \tilde{u}]} \leq \varepsilon, \qquad \forall \tilde{u} \in \{-1, 1\}^m$$

Proof intuition:

$$\frac{\mathbb{P}[J = j] \prod_{l=1}^{m} \mathbb{P}[u_l B_l = \tilde{u}_l | J = j]}{\mathbb{P}[J = j] \prod_{l=1}^{m} \mathbb{P}[u'_l B_l = \tilde{u}_l | J = j]} \leq e^{\varepsilon}$$

Differential Privacy in Practice
Input Perturbation

Privacy Analysis ctd.

Proof intuition:

Pick the same hash function j

$$\frac{\mathbb{P}[J=j]\prod_{l=1}^{m}\mathbb{P}[u_l B_l=\tilde{u}_l|J=j]}{\mathbb{P}[J=j]\prod_{l=1}^{m}\mathbb{P}[u'_l B_l=\tilde{u}_l|J=j]}\leq e^{\varepsilon}$$

Over all bits in $u_j^{(i)}$

Flip each bit with probability $\frac{1}{1+e^{\varepsilon/2}}$

$\boldsymbol{u}, l = h(d)$

-1 1 -1 -1 -1

Differ in at most two locations

-1 -1 1 -1 -1

$\boldsymbol{u}', l' = h(d')$

Case 1: $l = l'$

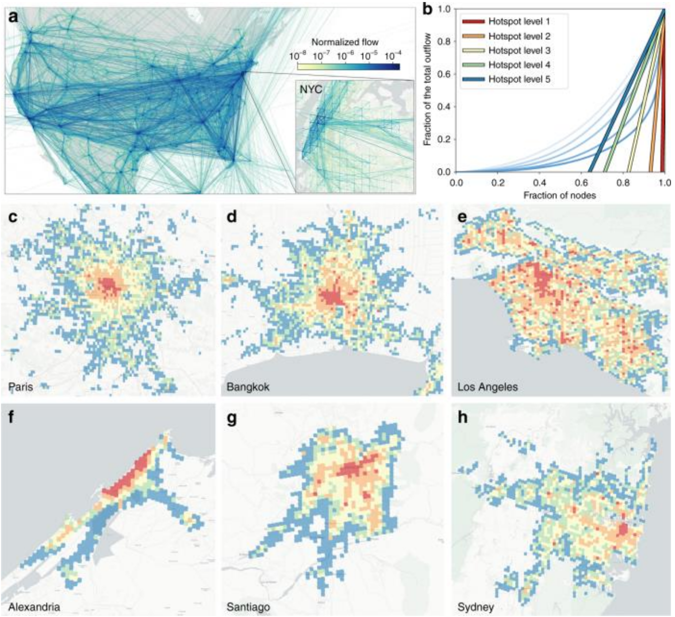
Then $\frac{\prod_{l=1}^{m}\mathbb{P}[u_l B_l=\tilde{u}_l|J=j]}{\prod_{l=1}^{m}\mathbb{P}[u_l B_l=\tilde{u}_l|J=j]}=1$

Case 2: $l \neq l'$

Consider probability that we flip bit l or l' with $\mathbb{P}[B_l=-1]=\frac{1}{1+e^{\varepsilon/2}}$

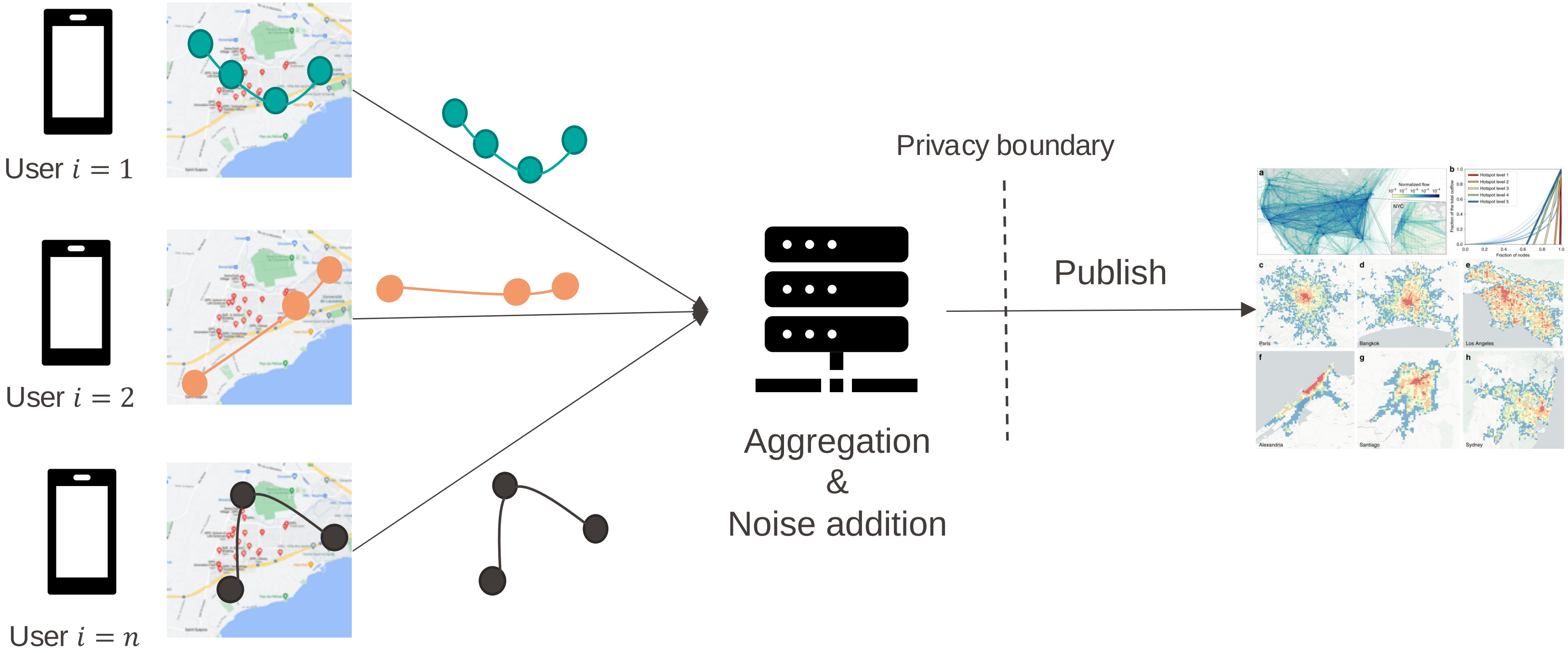
to derive a bound on $\frac{\mathbb{P}[u_l B_l=\tilde{u}_l|J=j]}{\mathbb{P}[u'_l B_l=\tilde{u}_l|J=j]}$

Differential Privacy in Practice
Output Perturbation



- In 2019, Google shared aggregated data from 300M Google Maps users with researchers to analyse human mobility patterns
  - Aggregate data from end-to-end trips taken by users
  - Privacy protected through differentially private output perturbation

Differential Privacy in Practice
Output Perturbation



Differential Privacy in Practice
Output Perturbation

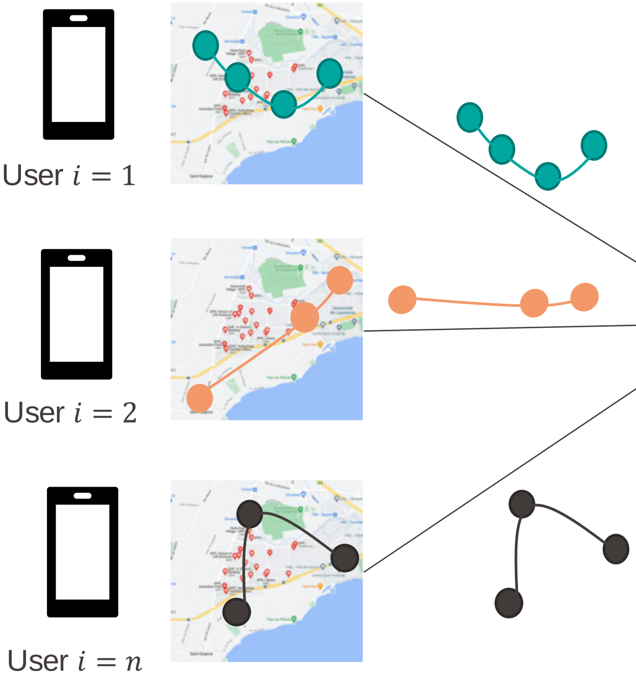
Aggregation over POIs

$$M(D) = f(D) + Lap(\frac{\Delta f}{\varepsilon})$$

What might be a problem here?

POI	M(D)
POI1	3094
POI2	3349
POI3	1782
…	…
POI	987

Differential Privacy in Practice
Output Perturbation




nature COMMUNICATIONS

MATTERS ARISING

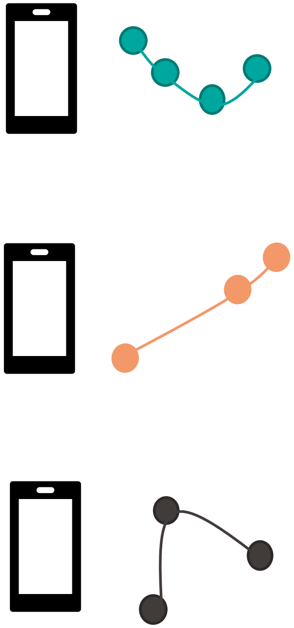
Check for updates

https://doi.org/10.1038/s41467-021-27566-0 OPEN

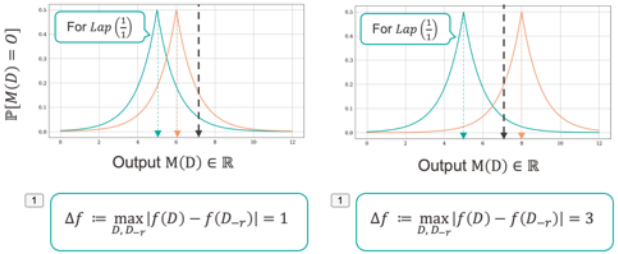
On the difficulty of achieving Differential Privacy in practice: user-level guarantees in aggregate location data

Florimond Houssiau[1,2], Luc Rocher[1,3] & Yves-Alexandre de Montjoye[1✉]

Differential Privacy Pitfalls
Units of Privacy & Unbounded Sensitivity



Remember?

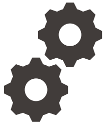


What is Δf?

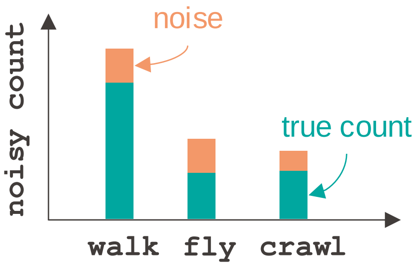
$$\mathrm{M(D)} = f(D) + Lap\left(\frac{\Delta f}{\varepsilon}\right)$$

POI	M(D)
POI1	3094
POI2	3349
POI3	1782
…	…
POI	987

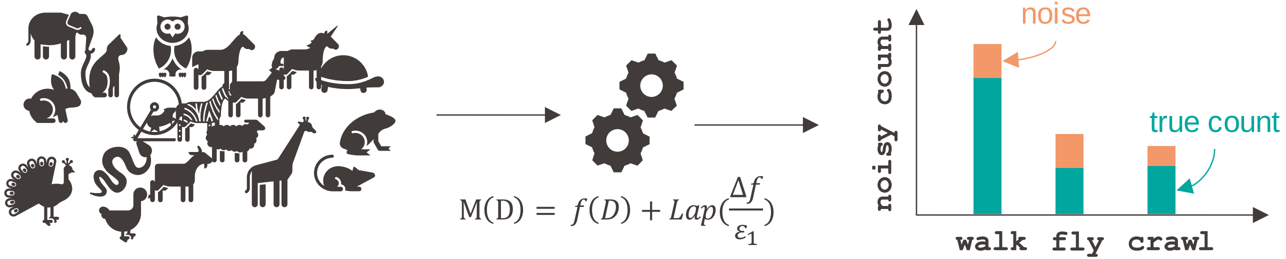
Differential Privacy Pitfalls
Unknown Categories

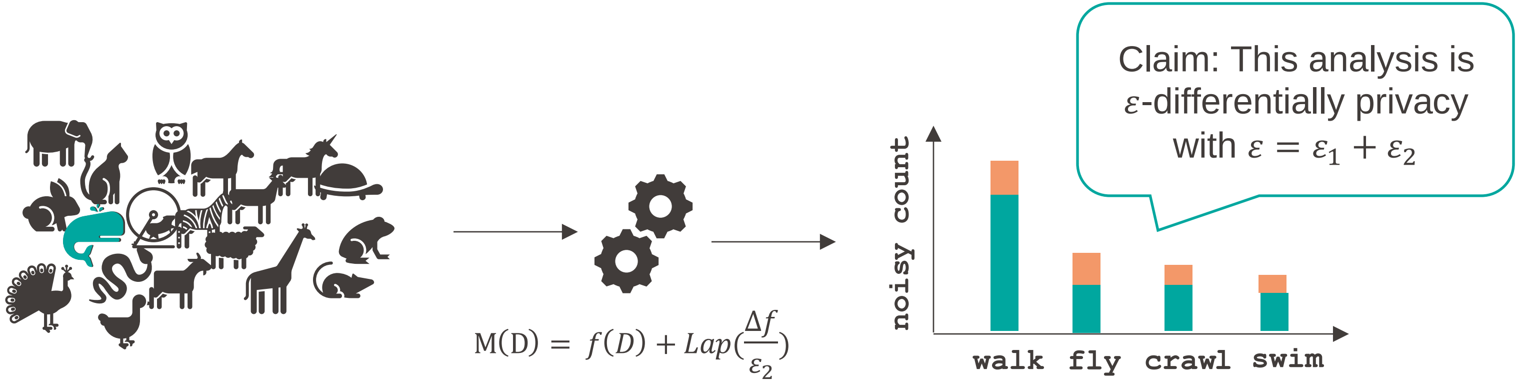


$$\mathrm{M(D)} = f(D) + Lap(\frac{\Delta f}{\varepsilon_1})$$

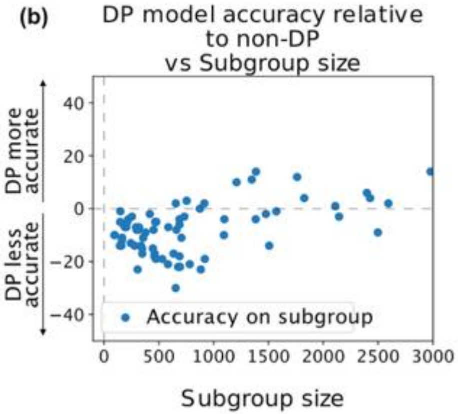


Differential Privacy Pitfalls
Unknown Categories

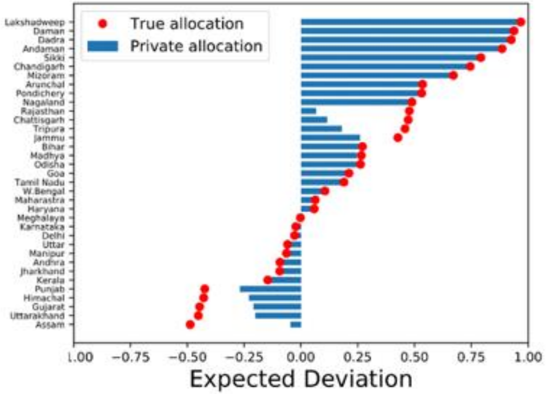




Differential Privacy Pitfalls
Disparate Impact



Disparate impact of DP on a computer vision problem trained with DP-SGD, epsilon ≈ 6
"Differential Privacy Has Disparate Impact on Model Accuracy"
Eugene Bagdasaryan, Vitaly Shmatikov 2019



Disparate impact of hypothetical Indian parliament seat apportionment if Census data had central Laplace "Fair Decision Making Using Privacy-Protected Data"
David Pujol et al. 2020

Differential Privacy in Practice

Summary

- Examples of input and output perturbation in practice show that
  - Very large user base offsets the utility costs of noise addition
  - Differential privacy in practice is hard
- Many pitfalls to avoid
  - User- versus record-level privacy and unbounded sensitivty
  - Unknown categories
  - Disparate impact on subpopulations (DP techniques might not be the right fit for use case)

Takeaways

Takeaways

- Differential privacy is a **formal notion of privacy** that brings **many benefits** in comparison to previous heuristic privacy definitions
  - Protects even against worst-case adversaries
  - Allows to quantify inherent trade-offs between privacy and utility
- However, it is not a good fit for all use cases
  - Limited to computing a **well-defined statistical function** over the data that must be known at time of data publishing
    → no secondary data use for research or other purposes
  - By design, **hides** fine-grained statistical patterns such as information about **outliers**
    → no anomaly detection
- Many **pitfalls** to avoid when it comes to implementation
  - User- versus record-level privacy and unbounded sensitivty
  - Unknown categories
  - Disparate impact on subpopulations