Information Security and Privacy (COM-402)

Part 4: Privacy enhancing technologies
Data anonymization

Carmela Troncoso

SPRING Lab

carmela.troncoso@epfl.ch

Some slides/ideas adapted from: JP Hubaux, Bryan Ford, Vitaly Shmatikov

PETs for Data anonymization

Scenario:

You have a set of data that contains personal data and you would like to anonymize it to:

- not be subject to data protection while processing
- make it public for profit
- make it public for researchers

Goal:

Produce a dataset that **preserves the utility** of the original dataset **without leaking information** about individuals. *This process is known as "*database sanitization*"*

REMEMBER: ANONYMITY IS ABOUT DECOUPLING IDENTITY AND ACTION!

To achieve anonymity we must **decouple** users from their attributes

✗ Let's make users pseudonymous!

✗ Let's remove identities!

Some attributes are quasi-identifiers!

✗ Let's remove some attributes!

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	*	Ovarian Cancer
47602	22	*	Ovarian Cancer
47678	27	*	Prostate Cancer
47905	43	*	Flu
47909	52	*	Heart Disease
47906	47	*	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** users from their attributes

- Let’s make users pseudonymous!
- Let’s remove identities!

Some attributes are quasi-identifiers!

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	*	Ovarian Cancer
47602	22	*	Ovarian Cancer
47678	27	*	Prostate Cancer
47905	43	*	Flu
47909	52	*	Heart Disease
47906	47	*	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

- Let’s remove some attributes!

Impossible to know what will be a QID

Caucasian	HIV+	Flu
Asian	HIV-	Flu
Asian	HIV+	Herpes
Caucasian	HIV-	Acne
Caucasian	HIV-	Herpes
Caucasian	HIV-	Acne

Anonymization: k-anonymity

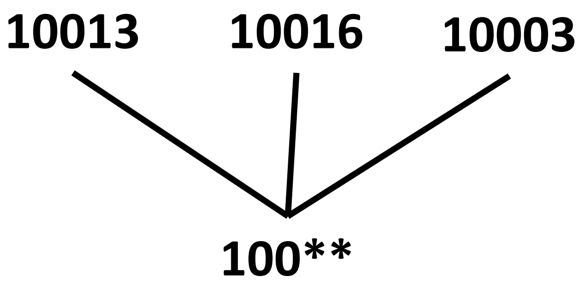
Key Attribute / Identifier	Quasi-identifier		Sensitive attribute
name	**gender**	**zipcode**	**problem**
John	Male	1012	Cancer
Zoey	Female	1013	Flu
Nathan	Male	1016	Heart Disease
Lucas	Male	1015	Heart Disease
Sam	Female	1003	Flu
Max	Male	1012	Flu
Mathias	Male	1014	HIV+
Sarah	Female	1012	Herpes
Julia	Female	1012	Flu

Anonymization: k-anonymity

- Each person contained in the database cannot be distinguished from at least k-1 other individuals whose information also appears in the released database.

***Generalization**: replace attributes with less specific, but semantically consistent values*

Example zipcode



name	gender	zipcode	problem
	Male	1012	Cancer
	Female	100**	Flu
	Male	100**	Heart Disease
	Male	100**	Heart Disease
	Female	100**	Flu
	Male	1012	Flu
	Male	100**	HIV+
	Female	1012	Herpes
	Female	1012	Flu

k=2

What is the rationale?

name	gender	zipcode	Favourite color
John	Male	1012	Blue
Zoey	Female	1013	Red
Nathan	Male	1016	Red
Lucas	Male	1015	Black
Sam	Female	1003	Yellow
Max	Male	1012	Red
Mathias	Male	1014	Black
Sarah	Female	1012	Blue
Julia	Female	1012	Red

Who has cancer, John or Max???

name	gender	zipcode	problem
	Male	1012	Cancer
	Female	100**	Flu
	Male	100**	Heart Disease
	Male	100**	Heart Disease
	Female	100**	Flu
	Male	1012	Flu
	Male	100**	HIV+
	Female	1012	Herpes
	Female	1012	Flu

k=2

What is the rationale?

name	gender	zipcode	Favourite color
John	Male	1012	Blue
Zoey	Female	1013	Red
Nathan	Male	1016	Red
Lucas	Male	1015	Black
Sam	Female	1003	Yellow
Max	Male	1012	Red
Mathias	Male	1014	Black
Sarah	Female	1012	Blue
Julia	Female	1012	Red

Does John have Cancer or Flu?

name	gender	zipcode	problem
	Male	1012	Cancer
	Female	100**	Flu
	Male	100**	Heart Disease
	Male	100**	Heart Disease
	Female	100**	Flu
	Male	1012	Flu
	Male	100**	HIV+
	Female	1012	Herpes
	Female	1012	Flu

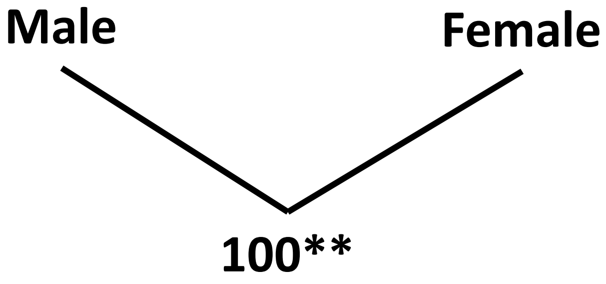
k=2

Anonymization: k-anonymity

- To improve anonymity, identifying attributes can be ***suppressed***

(note that suppression is the ultimate generalization!)

EXAMPLE
gender



name	gender	zipcode	problem
	*	1012	Cancer
	*	100*	Flu
	*	100*	Heart Disease
	*	100*	Heart Disease
	*	100*	Flu
	*	1012	Cancer
	*	100*	HIV+
	*	1012	Herpes
	*	1012	Flu

k=4

This is 3-anonymous, any problem?
(think about the rationale)

Homogeneity attack

Bob	
Zipcode	***Age***
47678	27

Background knowledge attack

Carl	
Zipcode	***Age***
47673	36

A 3-anonymous patient table

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	≥40	Flu
4790*	≥40	Heart Disease
4790*	≥40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

If you have background, e.g., “heart diseases are very unlikely in populations of 30 year old”

It is highly likely that Carl has cancer!!

ℓ-Diversity

- An equivalence class has *l*-diversity if there are at least ℓ well-represented values for the sensitive attribute.
- A database has ℓ-diversity if every equivalence class has ℓ–diversity.

Zipcode	Age	Salary	Disease
476**	2*	20K	Gastric Ulcer
476**	2*	30K	Gastritis
476**	2*	40K	Stomach Cancer
4790*	≥40	50K	Gastritis
4790*	≥40	100K	Flu
4790*	≥40	70K	Bronchitis
476**	3*	60K	Bronchitis
476**	3*	80K	Pneumonia
476**	3*	90K	Stomach Cancer

A 3-diverse hospital records dataset

ℓ-Diversity: problems

Similarity attack

Bob	
Zip	***Age***
47678	27

A 3-diverse patient table

Zipcode	Age	Salary	Disease
476**	2*	20K	Gastric Ulcer
476**	2*	30K	Gastritis
476**	2*	40K	Stomach Cancer
4790*	≥40	50K	Gastritis
4790*	≥40	100K	Flu
4790*	≥40	70K	Bronchitis
476**	3*	60K	Bronchitis
476**	3*	80K	Pneumonia
476**	3*	90K	Stomach Cancer

Conclusion

1. Bob's salary is in [20k,40k], which is relatively low
2. Bob has some stomach-related disease

l-diversity does not consider semantics of sensitive values!

ℓ-Diversity: problems

Q1: 423, >50**
Q2: 423, <60**

Original dataset

…	Cancer
…	Cancer
…	Cancer
…	Flu
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Cancer
…	Flu
…	Flu

99% have cancer

Anonymization A

Q1	Flu
Q1	Flu
Q1	Cancer
Q1	Flu
Q1	Cancer
Q1	Cancer
Q2	Cancer
[illegible]	[illegible]

Anonymization B

Q1	Flu
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q1	Cancer
Q2	Cancer
[illegible]	[illegible]
Q2	Flu

99% cancer ⇒ quasi-identifier group is not “diverse”
…yet anonymized database does not leak anything

50% cancer ⇒ quasi-identifier group is “diverse”
This leaks a ton of information

l-diversity does not consider distribution of semantic values!

t-Closeness

- An equivalence class has *t-closeness* if the distance between the distribution of a sensitive attribute in this class and the distribution of the attribute in the whole table is no more than a *threshold* t.

- A table has *t-closeness* if all equivalence classes have t-*closeness.*

Caucasian	787XX	HIV+	Flu
Asian	787XX	HIV-	Flu
Asian	787XX	HIV+	Herpes
Caucasian	787XX	HIV-	Acne
Caucasian	787XX	HIV-	Herpes
Caucasian	787XX	HIV-	Acne

This is k-anonymous, l-diverse and t-close…

…so secure, right?

What Does the Attacker Know?

Bob is Caucasian
and
I heard he was
admitted to hospital
with flu…

Caucasian	787XX	HIV+	Flu
Asian	787XX	HIV-	Flu
Asian	787XX	HIV+	Herpes
Caucasian	787XX	HIV-	Acne
Caucasian	787XX	HIV-	Herpes
Caucasian	787XX	HIV-	Acne

Takeaways

Anonymizing a dataset via generalization and suppression is extremely hard

The k-anonymity idea focuses on transforming the dataset not its semantics

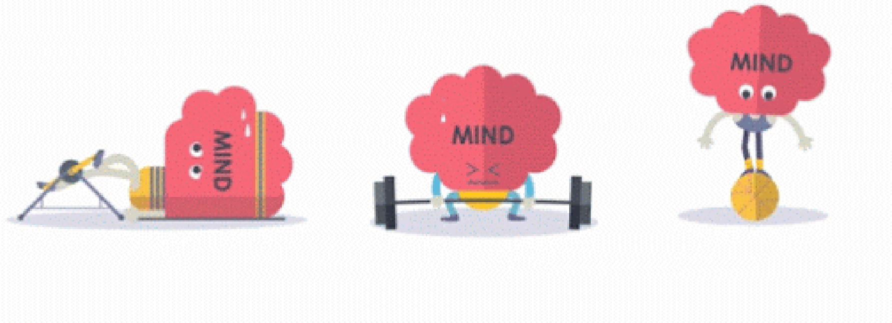
Achieving k-anonymity, l-diversity, t-closeness is hard, and still does not guarantee privacy

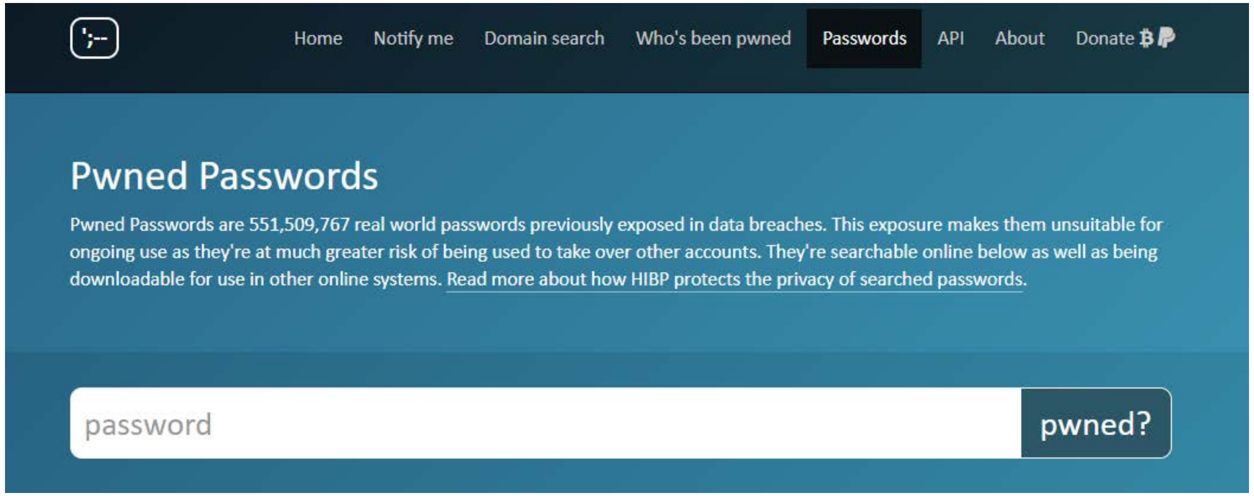
The adversary's background **can be anything**

Being able to fully anonymize a high-dimensional database is as likely as being able to find a unicorn in the galaxy

If we cannot publish the data, can we do something with it?

Let’s exercise your privacy brain





From the point of view of the server (that receives the 5-bytes suffix

What is the privacy of the password?

The password is 475-anonymous!!

Would you send your password in the clear?

Would you send a hash?

What they do: send the first 5 bytes of the hash of the password and receive a list of 475 suffixes to check offline

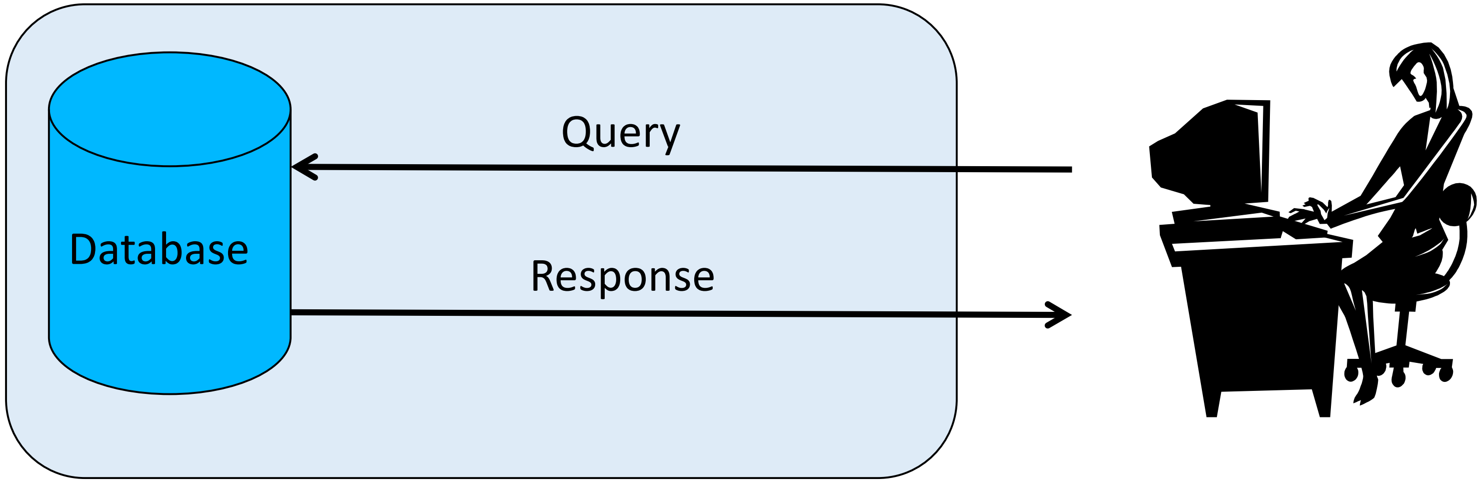
https://haveibeenpwned.com/
https://blog.cloudflare.com/validating-leaked-passwords-with-k-anonymity/

The interactive scenario

Many times we do not want the data, we want statistics!

Redefined Goal for the interactive case:

Produce an **answer** that **preserves the utility** of the **statistics without leaking information** about individuals.

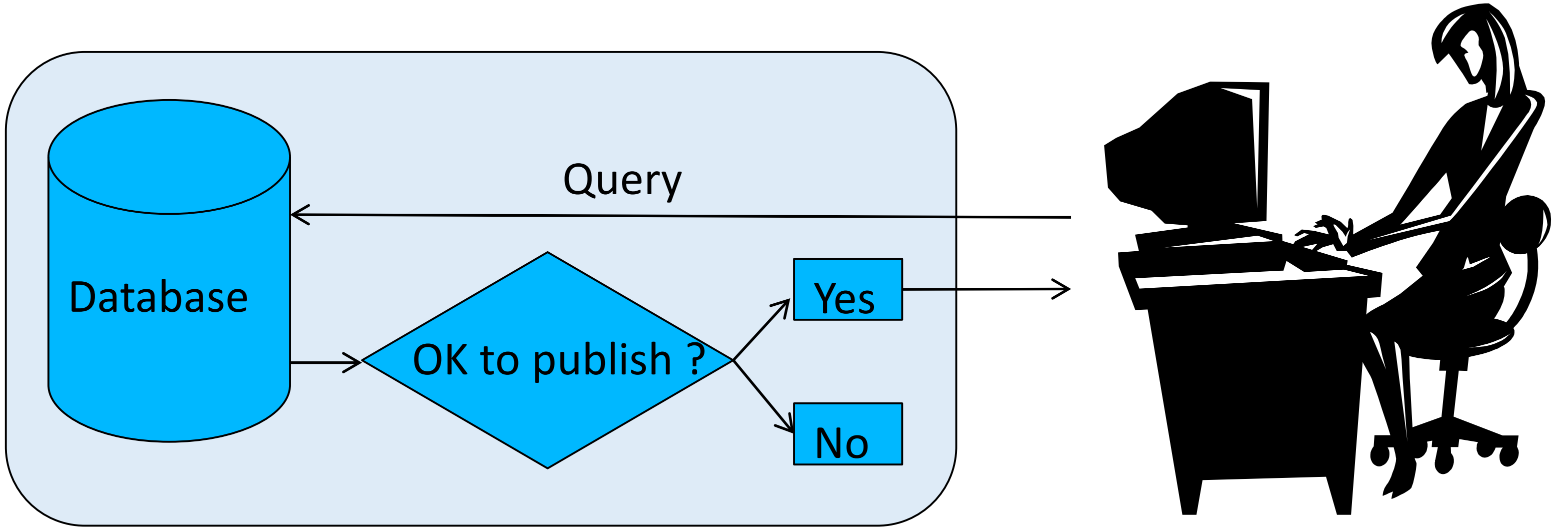


Query = **What is the average salary of women professors at IC@EPFL with Spanish nationality?**

Is there a privacy problem?

The interactive scenario

Let's audit the queries, if the query will leak, deny!
Either answer truthfully or state that there will be no answer

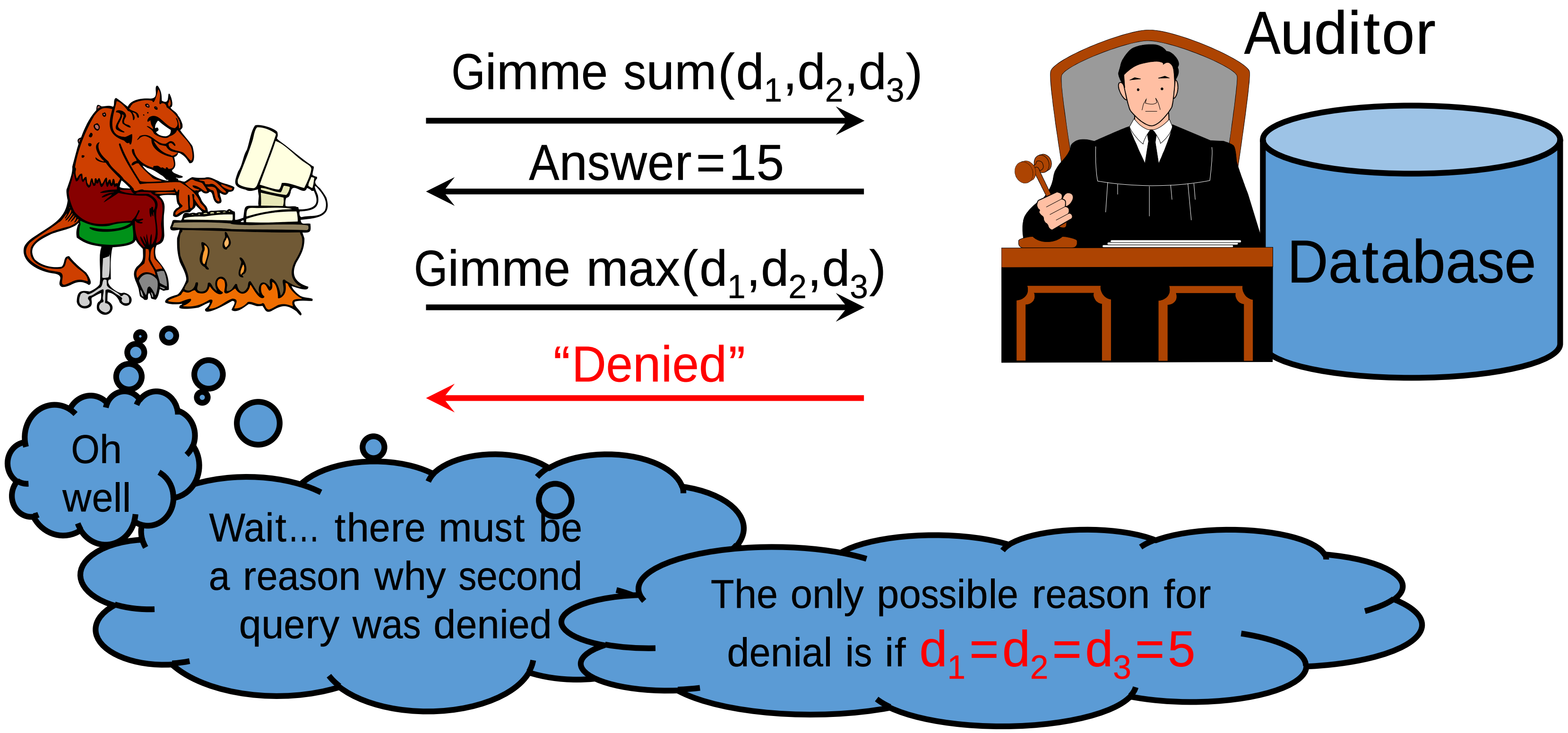


Database assumed to contain *numeric* values.

! Not answering already reveals some information !

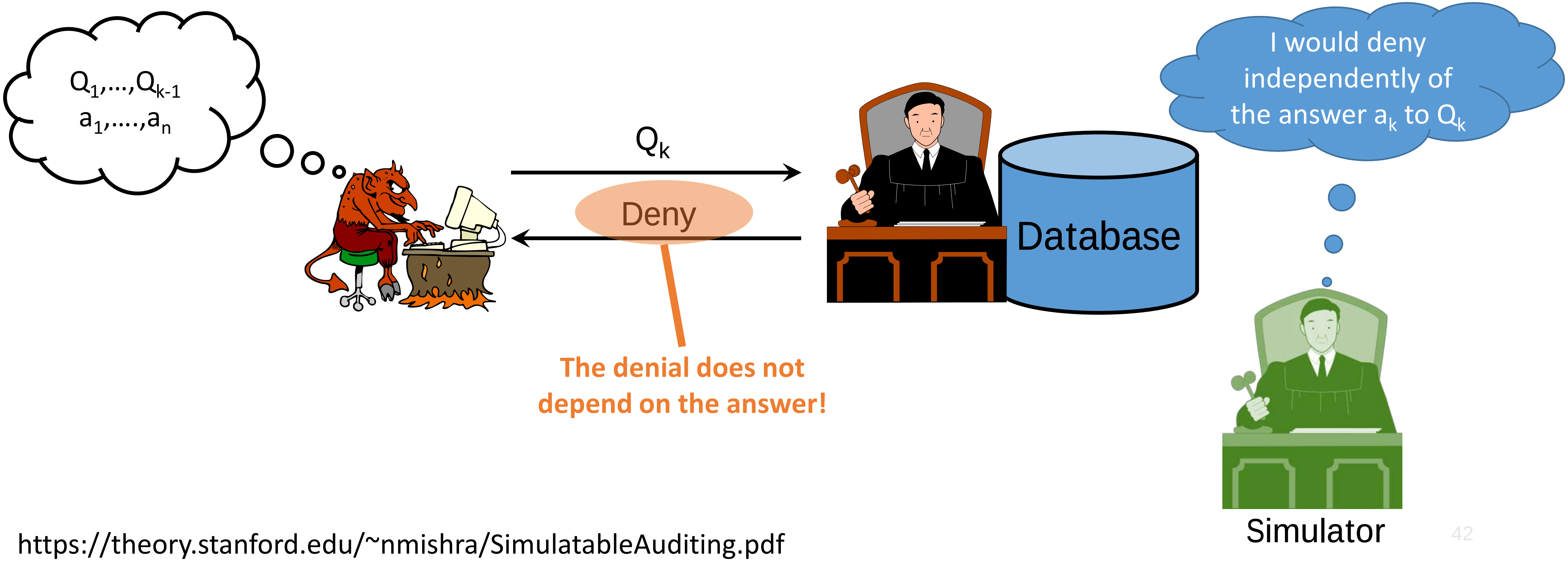
When denying fails: learning exact values

- Variables d_i are real, privacy breached if adversary learns some d_i



Can I make sure that the next query does not leak?
Simulatable auditing

One cannot learn anything from the denial **if the decision to deny or give an answer is independent of the actual data set and the real answer**.



https://theory.stanford.edu/~nmishra/SimulatableAuditing.pdf

Auditing has problems

- Privacy definition? Privacy of Values? Groups? Exact?
- Algorithmic limitations
  - Secure deniability implies using algorithms computationally prohibitive
  - Feasible focus mostly on simple queries
- Collusion? Either high cost or no security
- Utility?
  - Percentage of denials may not be the best measure

What else can we do? Modifying inputs

- **Subsampling**
  - A subset of the rows is chosen at random and released and **statistics are computed on the subsample**
  - Uneven privacy for users, being in a subsample may have unfortunate consequences
    - ➢ Not being may too!
- **Input perturbation**
  - **Data or queries are modified before** a response is generated
  - How can we quantify the leakage?
  - How to balance for utility?

What else can we do? Modifying outputs

- **Adding random noise to the output**
  - **Naively**, this approach will fail
    - E.g., if the same query is asked repeatedly, then the responses can be averaged, and the true answer will eventually emerge.
  - This cannot be fixed by recording each query and providing the same response each time a query is re-issued.
    - **Syntactically different queries may be semantically equivalent**, and, if the query language is sufficiently rich, then the equivalence problem itself is undecidable.
- **Randomized response**
  - Respondents to a query **flip a coin and, based on the outcome, they either honestly respond or respond randomly**
  - Privacy comes from the uncertainty of how to interpret a reported individual value.
  - Yet, data can be useful because **randomness can be averaged out**
  - **Not usable for every case, or combined with other techniques**

Differential privacy

Remember the Goal for the interactive case:

Produce an **answer** that preserves the utility of the statistics without leaking information about individuals.

To have any utility we must allow the leakage of some information, but we can set a bound on the extent of leakage!

Differential Privacy:

Output is similar whether any single individual's record is included in the database or not.

Guarantees minimal similarity

Differential Privacy

- **Basic philosophy:** instead of the real answer to a query, output a random answer, such that by a small change in the database (someone joins or leaves), the distribution of the answer does not change much.

- **A new privacy goal:** minimize the increased risk incurred by an individual when joining (or leaving) a given database.

- **Motivation:** A privacy guarantee that limits risk incurred by joining, therefore encourages participation in the dataset, increasing social utility.

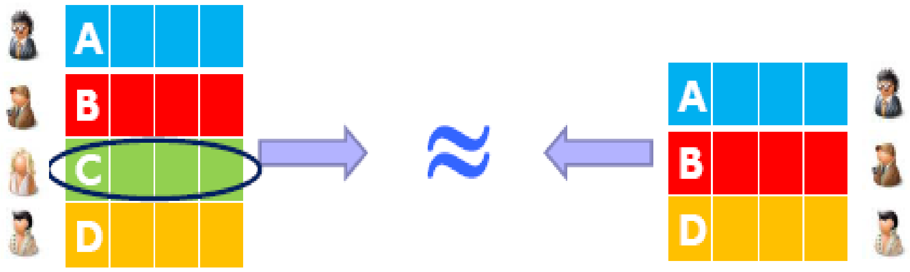
Important!!!!

Differential Privacy is a privacy notion **NOT** a mechanism

You use mechanisms to achieve differential privacy

Differential Privacy - Informal Definition

Output is similar whether any single individual's record is included in the database or not.



C's inclusion of her record in the computation does not make her *significantly* worse off.

If there is already some risk of revealing a secret of C by combining auxiliary information and something learned from DB, then **that risk is still there** but not *significantly* increased by C's participation in the database.

$\boldsymbol{\epsilon}$-Differential Privacy – Formal Definition

- $\mathcal{D}$: The set of input databases
- R: Output space of the query
- F: Query function

  $F: \mathcal{D} \rightarrow R$
- d: Distance function on the set of databases
- *Neighboring databases*: Pairs of databases ($\mathcal{D}$, $\mathcal{D}_{-r}$) differing only in one row *r* (e.g., individual)

$$d(\mathcal{D}\text{-}\mathcal{D}_{-r}) = 1$$

ϵ-Differential Privacy – Formal Definition

- Principle
  - The **removal/addition** of a **single record** in the **database should/does *not* substantially affect the values** of the computed function/statistics.

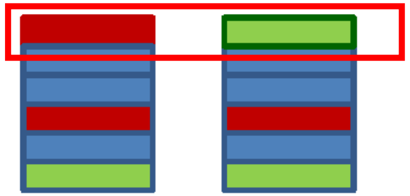
- Formalization
  - Let A be the **randomized function** (namely a **mechanism**) to be computed on a set of records.
    - A is the actual function to be computed f + **noise**.
  - Let S be a subset of the possible values taken by A.
  - A provides ϵ-differential privacy if for all r, S:

$$\mathrm{P}[A(D) \in S] \leq e^{\epsilon} \times \mathrm{P}[A(D_{-r}) \in S]$$

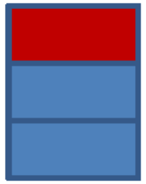
Differential Privacy - Intuition

For every pair of inputs that differ in one value

For every output …



D_1 D_2

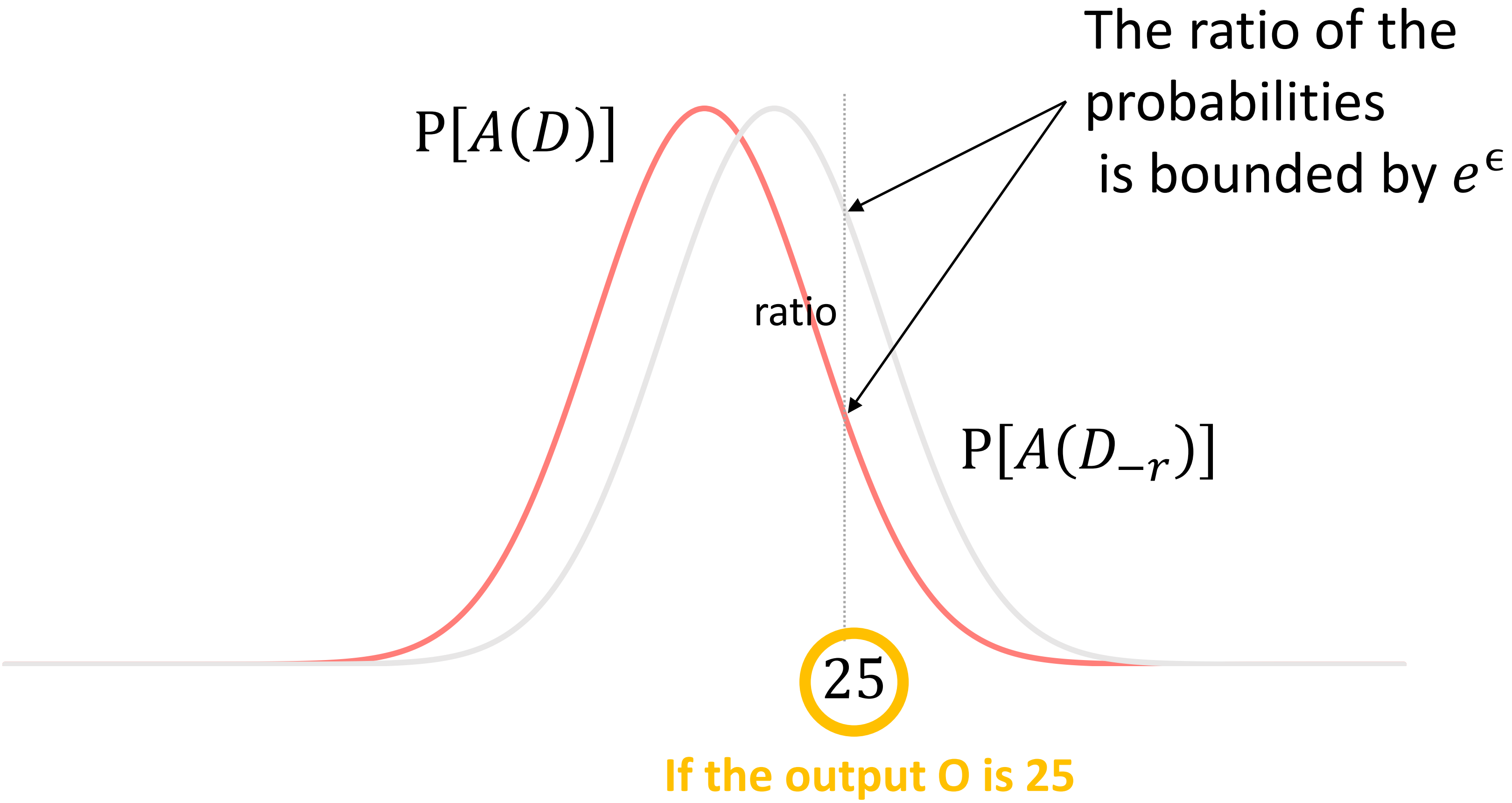


O

Adversary should not be able to distinguish between any D_1 and D_2 based on any O

$$\log\left(\frac{\Pr[A(D_1) = O]}{\Pr[A(D_2) = O]}\right) < \varepsilon \qquad (\varepsilon>0)$$

ϵ-Differential Privacy



How to achieve $\boldsymbol{\epsilon}$-Differential Privacy (simple case)

How to achieve ϵ-differential privacy (simple case)?

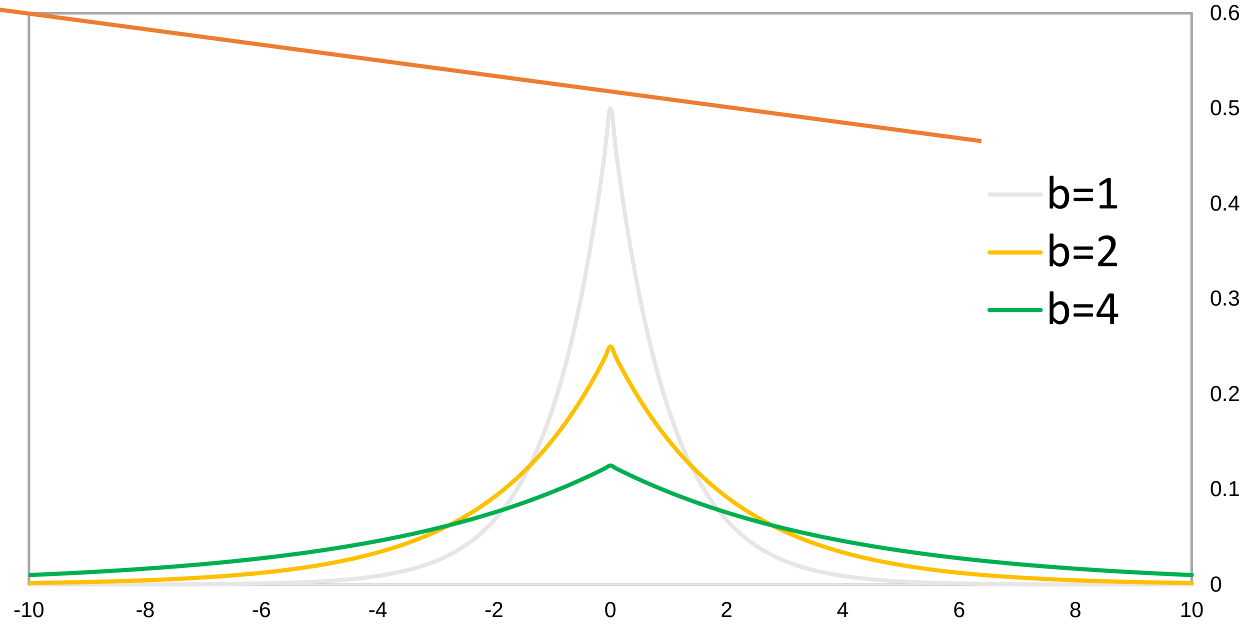
Assume f is a scalar function, i.e., $f: \mathcal{D} \rightarrow \mathbb{R}$ (e.g., "number of records with cancer"?)

Return $\boldsymbol{A(D) = f(D) + \mathbf{Lap}\left(\frac{\Delta f}{\epsilon}\right)}$

$\mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right)$ is **noise** drawn from a **Laplacian** distribution of parameter $\frac{\Delta f}{\epsilon}$

Δf is the **sensitivity** of function f:

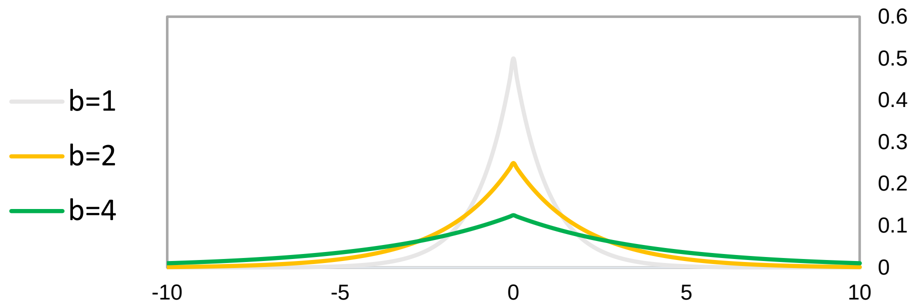
$$\Delta f = \max_{r} |f(D) - f(D_{-r})|$$



Why Laplacian Distribution?

- The Laplacian distribution is: $\mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right) = \frac{\epsilon}{2\Delta f}$ exp(- $\frac{x\epsilon}{\Delta f}$).
- The distortion of the result depends on both the sensitivity and privacy guarantee:
  - The higher the sensitivity, the higher the distortion
  - The higher the privacy guarantee (the lower ϵ), the higher the distortion

- This distribution has highest density at 0 (good for accuracy).
- This distribution is symmetric about 0 and has a heavy tail.



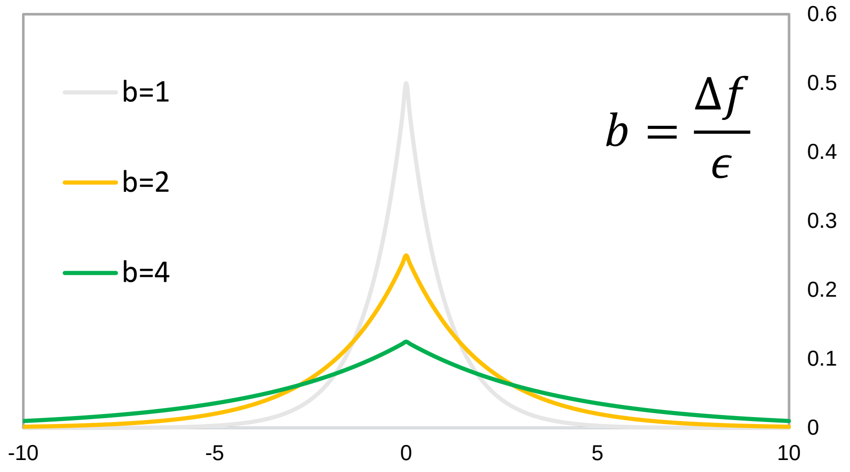
How to choose the parameters ?

Selecting ε

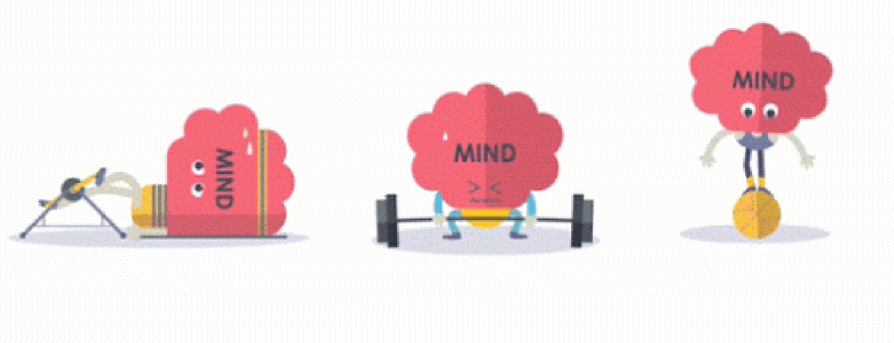
- The parameter ε is public (remember: no security by obscurity
- Selection of ε by Cynthia Dwork:
  - *“We tend to think of ε as 0.01, 0.1, or in some cases, ln2 or ln3”*
  - Smaller ε means better privacy
  - But, what about the utility ?

It depends on the sensitivity!

$$\Delta f = \max_r |f(D) - f(D_{-r})|$$



What is the sensitivity of... ?



For any two **neighboring databases** $\boldsymbol{(D, D_{-r})}$**:**

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

Sensitivity of counting queries:

- The number of elements in the database with a given property *P.*

Sensitivity of histogram queries:

- Suppose each entry in d takes values in $\{c_1, c_2, \ldots, c_n\}$.
- *Histogram*(d) = $\{m_1, m_2, \ldots, m_n\}$, m_i = (#entries in d with value c_i)

Composability of Differential Privacy

Theorem: If algorithms F_1, F_2, …, F_k use independent randomness and each F_i satisfies ε_i-differential privacy, respectively. Then outputting all the answers together satisfies differential privacy with

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

Composability of Differential Privacy

Theorem: If algorithms F_1, F_2, …, F_k use independent randomness and each F_i satisfies ε_i-differential privacy, respectively. Then outputting all the answers together satisfies differential privacy with

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

Does privacy increase or decrease?

How to ensure differential privacy ?

- **Input perturbation**
  - Add noise directly to the database ($\neq$ 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 algo

    \+ algorithm can be optimized with the noise addition

    \- difficult to generalize & depends on the inputs

More on these algorithms and variants in CS-523

Why is DP possible (while anonymization was impossible):

The final result depends on multiple personal records

However it does not depend much on any particular one (sensitivity)

Therefore adding a little bit of noise to the result, suffices to hide any record contribution

For full anonymization.... one would need to add a lot of noise to all the entries

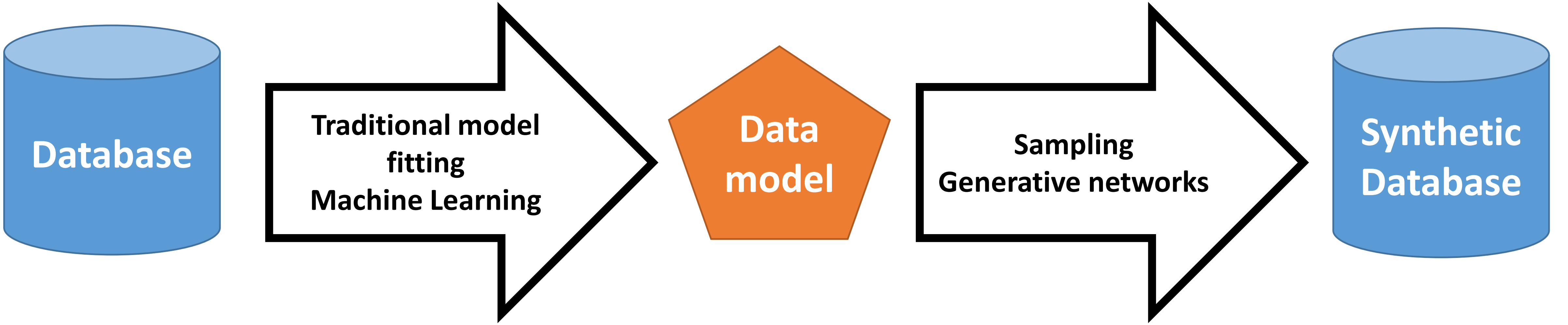
But… the architecture is different: **one Trusted-Third-Party holds the data**!

Also… after some uses utility drops

best use: one-time, **data collection**!!

Google RAPPOR ← Collect data from phones
Apple ← Collect data from phones
Federated learning ← Share models
Smart energy ← Collect measurements

Synthetic databases: A new hope



Problems:

How to know which features to model? Features determine utility!

How to measure privacy? Some processes provide DP, but for what attribute?

Takeaways

Anonymizing is difficult, but privacy-preserving statistical querying is possible

Differential privacy: a notion to reason about privacy

- Key idea: given an output not possible to learn about one individual's participation
- Algorithms available for protecting different types of queries

Synthetic data can be an option for improving data sharing

- Very early days…