

Trustworthy AI: Privacy and Fairness

Catuscia Palamidessi



Content of the lectures

- Privacy
 - Motivations
 - Central Differential Privacy
 - Local Differential Privacy
 - Privacy vs Utility
- Fairness
 - Motivations
 - Some notions of fairness

Content of the lectures

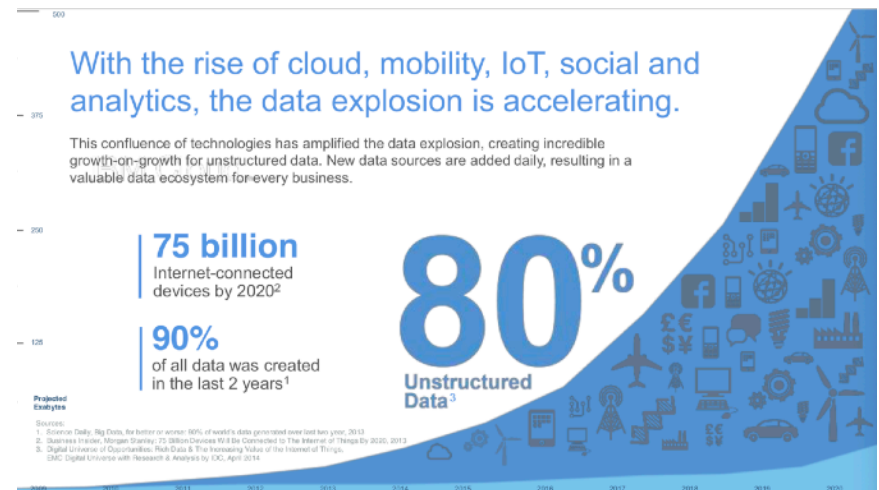
- **Privacy**
 - Motivations
 - Central Differential Privacy
 - Local Differential Privacy
 - Privacy vs Utility
- **Fairness**
 - Motivations
 - Some notions of fairness

Motivations

Privacy is not a new issue, but in our times the problem is exacerbated by the Big Data revolution: data are collected and stored in enormous amounts, and there are the computing resources, and in particular the power of machine learning, allowing to analyse them and extract all sort of sensitive information



Also, data are accumulated at an increasing speed. According to a research made recently by IMB, 90% of the world data had been generated in the last 2 years!



Risks about privacy breaches

Sensitive information can be used for fraudulent purposes.

- **Credentials**

Examples: credit card numbers, home access code, passwords, ...

Risks: Stealing personal property

- **Information about the individual**

Examples: medical status, intimate videos, religious beliefs, political opinions

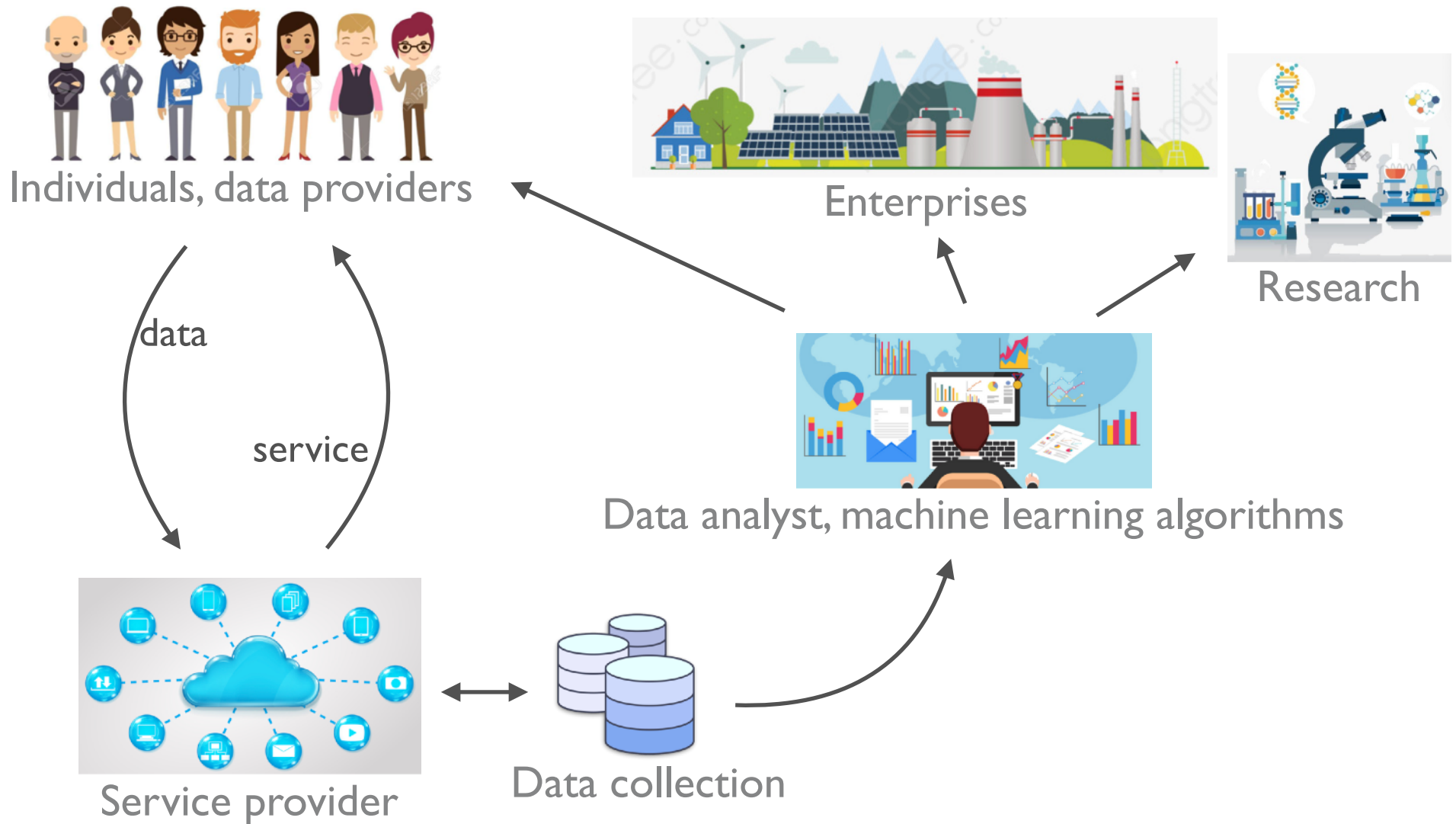
Risks: discrimination, blackmailing, public shame

- **Identification information, i.e., information that can uniquely identify an individual**

Examples: name, SSN, bank information, biometric data (such as fingerprint and DNA)

Risks: Identity theft

Privacy: stakeholders



Issue I: Inference attacks

The problem of Privacy is complicated because sensitive information can be derived using **side information**, i.e., correlated information that is necessarily public or anyway available to the attacker (inference attacks).

Example: all voters vote for the same candidate

- The typical countermeasures used in security (e.g., encryption, access control) do not help against this kind of information leakage
- The side knowledge of the adversary can increase with time

Issue 2: Trade off with utility

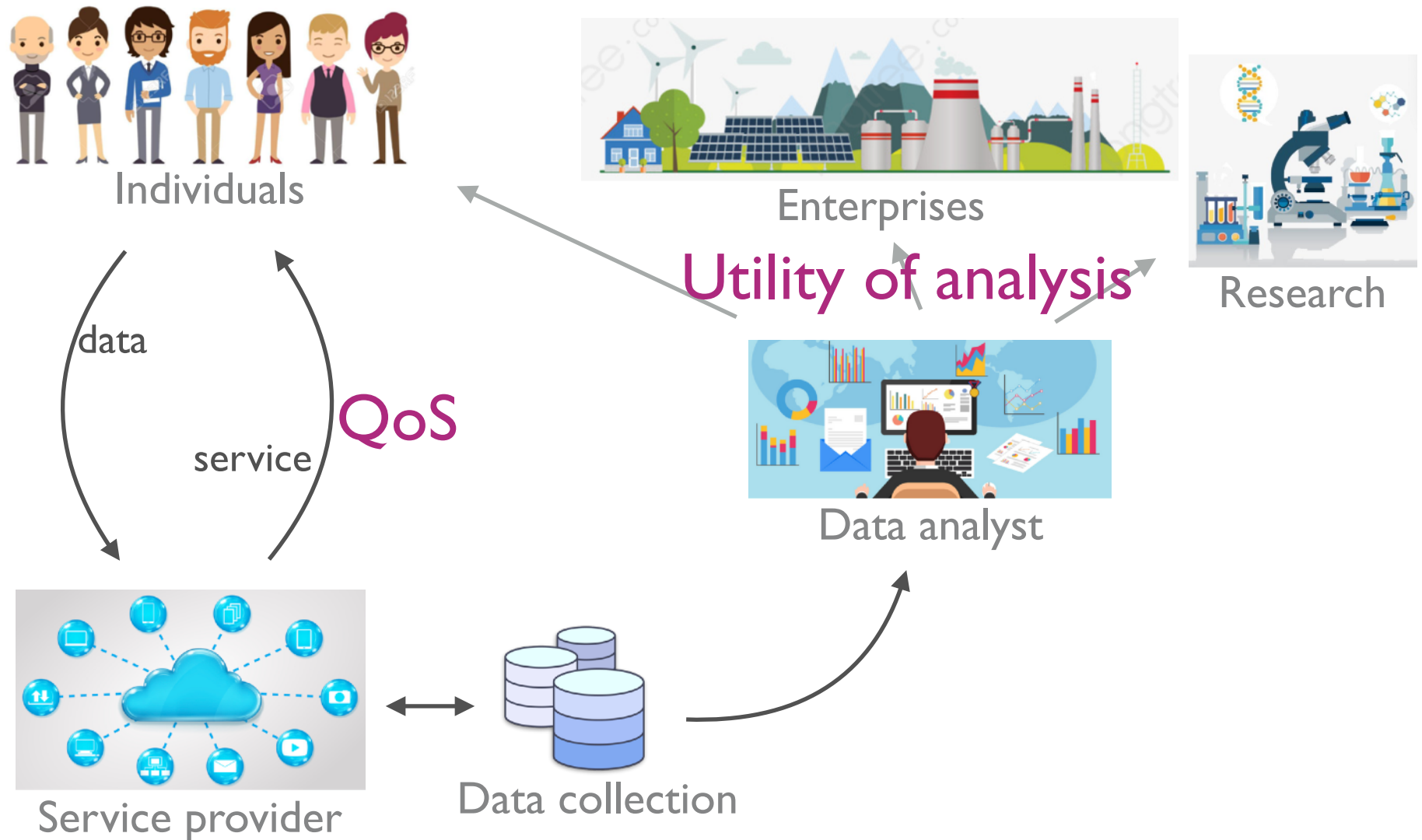
The methods to protect privacy should not destroy the utility of the data.

One of the main issues in the research about privacy-protection mechanisms is to find a good trade-off with utility

In general we consider two kinds of utility:

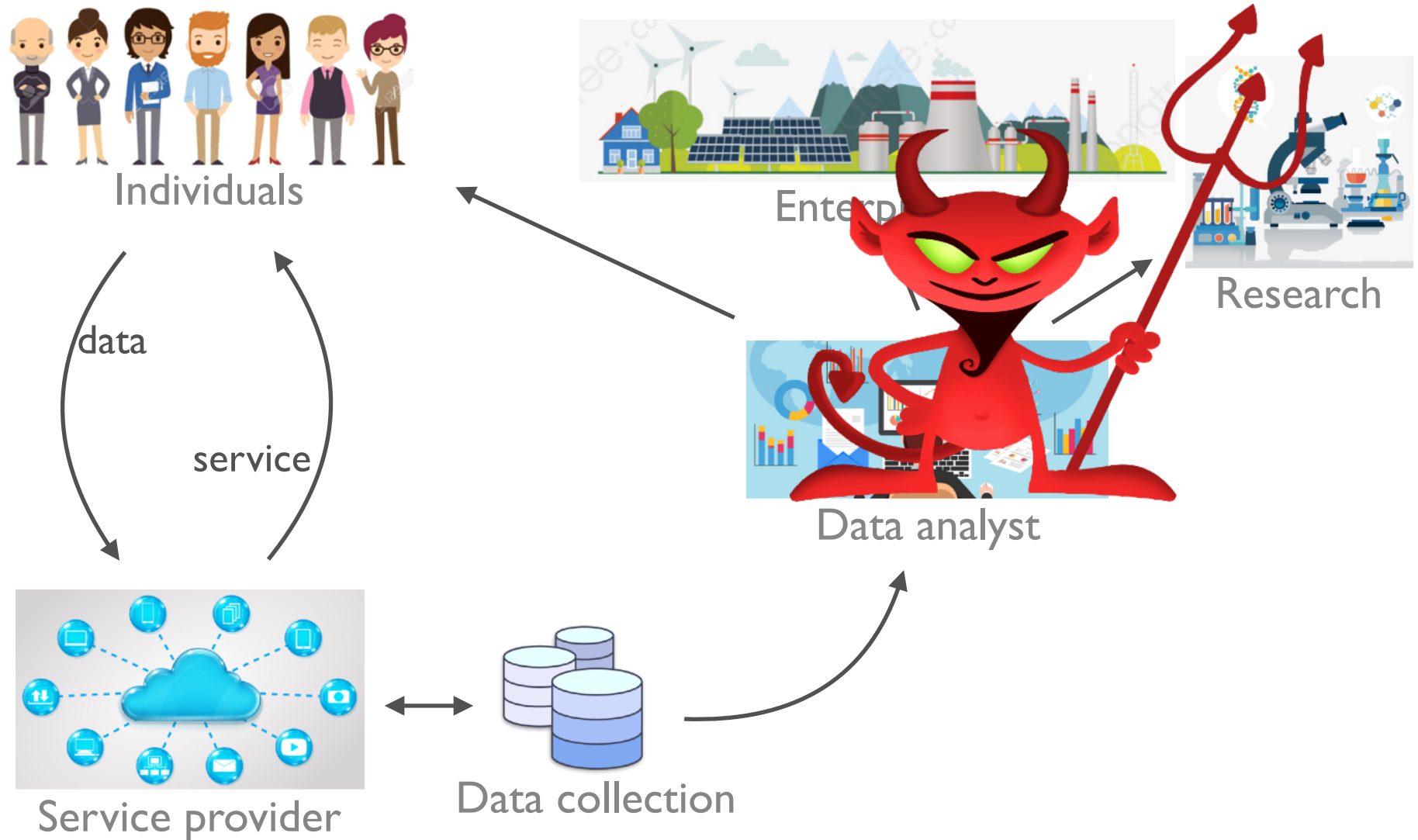
- the **Quality of Service (QoS)**
- the **precision of the analysis**, including the **accuracy** in case of machine learning models

Issues concerning privacy



Issue 3: Whom can we trust?

Issues concerning privacy



Issue 3: Whom can we trust?

1. **Centralized model:** we trust the server / data curator.

- The sanitization is done by the curator.
- Utility is the precision of analysis / accuracy of ML models.
- Two cases:
 1. the (sanitized) micro data are made available, or
 2. they are not available, we can only query the database

2. **Local model:** the server / curator may be corrupted or unable to protect the data.

- The sanitisation must be done at the user's side
- Both kinds of utility (QoS, precision/accuracy) should be taken into account
- The sanitised micro data are made publicly accessible.

The local model has become more popular recently since people tend to trust less and less the service providers and curators (also due to recent scandals). Some big companies (e.g., Google and Apple, Amazon) have developed their own LDP systems.

Scenario 1.1:

Global model

The micro data are made available

First solution: anonymization

- This is the most obvious solution: remove the identity of individuals from the database, so that the sensitive information cannot be directly linked to the individual

- Example: assume that we have a medical database, where the sensitive information is disease that has been diagnosed

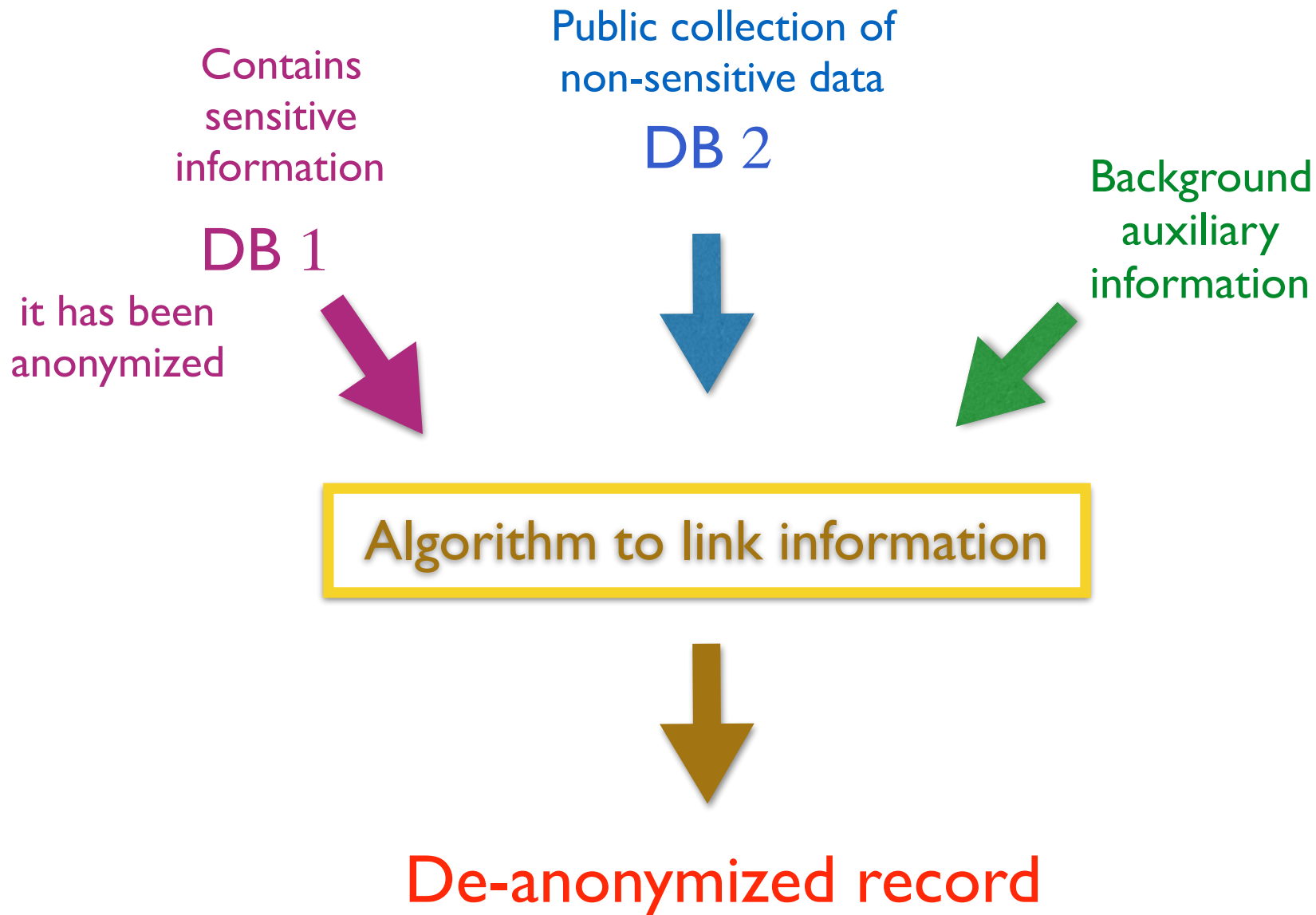
	Name	age	Disease
1	Jon Snow	30	cold
2	Jamie Lannister	39	amputated hand
3	Arya Stark	16	stomach ache
4	Bran Stark	14	crippled
5	Sandor Clegane	45	ignifobia
6	Jorah Mormont	48	gleyscale
7	Eddad Stark	32	headache
8	Ramsay Bolton	32	psychopath
9	Daenerys Targaryen	25	mania of grandeur

First solution: anonymization

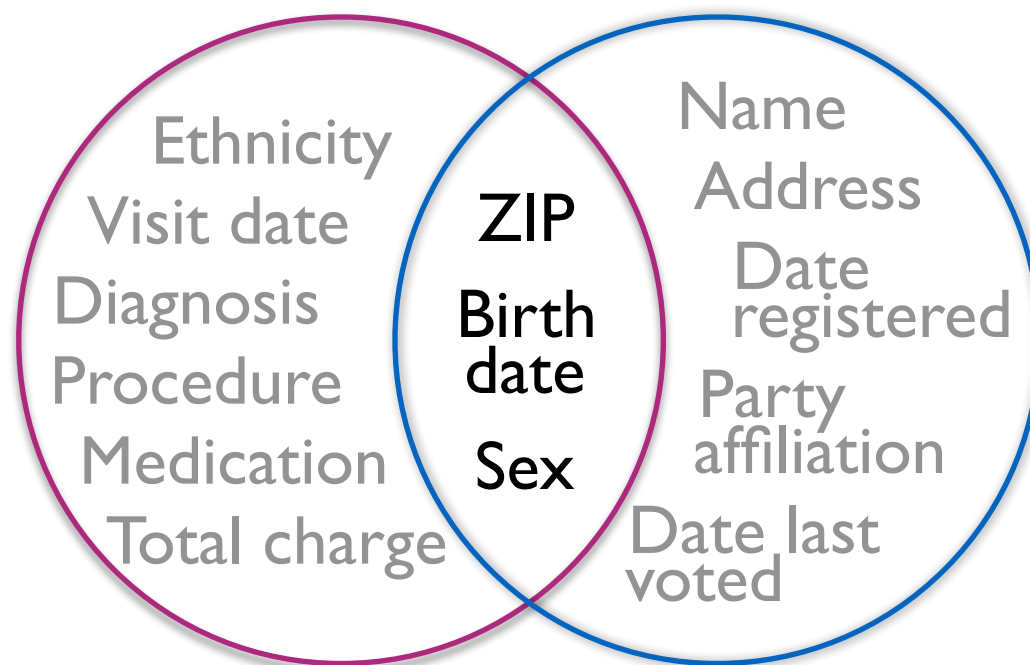
- Anonymization removes the column of the name, so that, for instance, the grayscale disease cannot be directly linked to Jorah Mormont
- Historically the first method, still used nowadays
- However, this solution has been (already several years ago) shown to be ineffective, i.e., vulnerable to de-anonymization attacks

	Name	age	Disease
1	-	30	cold
2	-	39	amputated hand
3	-	16	stomac ache
4	-	14	crippled
5	-	45	ignifobia
6	-	48	gleyscale
7	-	32	headache
8	-	32	psychopath
9	-	25	mania of grandeur

De-anonymization attack (I). Sweeney'98



De-anonymization attack (I). Sweeney'98



DB 1: Medical data

DB 2: Voter list

87 % of US population is uniquely identifiable by 5-digit ZIP, gender, DOB

This attack has lead to the proposal of k-anonymity

K-anonymity [Samarati & Sweeney]

- **Quasi-identifier:** Set of attributes that can be linked with external data to uniquely identify individuals
- Make every record in the table indistinguishable from a least $k-1$ other records with respect to quasi-identifiers. This can be done by:
 - suppression of attributes, and/or
 - generalization of attributes, and/or
 - addition of dummy records
- Linking on quasi-identifiers yields at least k records for each possible value of the quasi-identifier

K-anonymity

Example: 4-anonymity w.r.t. the quasi-identifiers (nationality, ZIP, age)

- achieved by suppressing the nationality and generalizing ZIP and age

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Figure 1. Inpatient Microdata

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

Figure 2. 4-anonymous Inpatient Microdata

Problems with k-anonymity and similar methods

- **Everything can turn out to be a quasi-identifier**
 - Especially in high-dimensional and sparse databases.
- **Composition attacks**
 - Combination of knowledge coming from different sources
 - Open world: Even if present data are protected, in the future there may be some new knowledge available

De-anonymization attacks (II)

Robust De-anonymization of Large Sparse Datasets.
Narayanan and Shmatikov, 2008.

Showed the limitations of K-anonymity

De-anonymization of the **Netflix Prize dataset** (500,000 anonymous records of movie ratings), using **IMDB** as the source of background knowledge.

They demonstrated that an adversary who knows just a few preferences about an individual subscriber can identify his record in the dataset.



De-anonymization attacks (III)

De-anonymizing Social Networks. Narayanan and Shmatikov, 2009



By using only the network topology, they were able to show that 33% of the users who had accounts on both **Twitter** and **Flickr** could be re-identified in the anonymous Twitter graph with only a 12% error rate.

What if we adopt a more controlled setting ?

Scenario 1.2:

Centralized model

Micro data not accessible, we can only query the DB

We will see that, even in this setting, k-anonymity may fail to provide privacy

There is still the problem of composition attacks

Example

- A medical database D1 containing correlation between a certain disease and age.
- Query: “what is the minimal age of a person with the disease”

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

D1 is **2-anonymous with respect to the query**. Namely, every possible answer partitions the records in groups of at least 2 elements

Alice	Bob
Carl	Don
Ellie	Frank

- A medical database D2 containing correlation between the disease and weight.
- Query: “what is the maximal weight of a person with the disease”

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Also D2 is **2-anonymous**

Alice	Bob
Carl	Don
Ellie	Frank

k-anonymity is not compositional

Combine with the two queries:
minimal weight and the maximal
age of a person with the disease

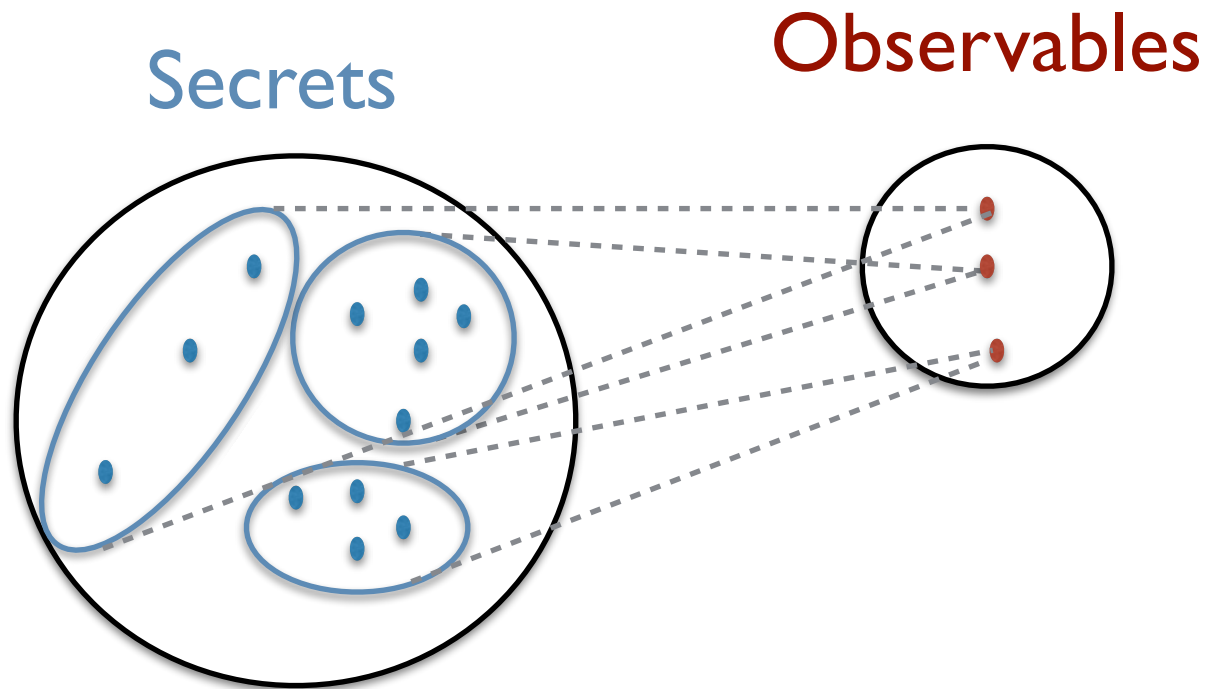
Answers: 40, 100. **Unique!**

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

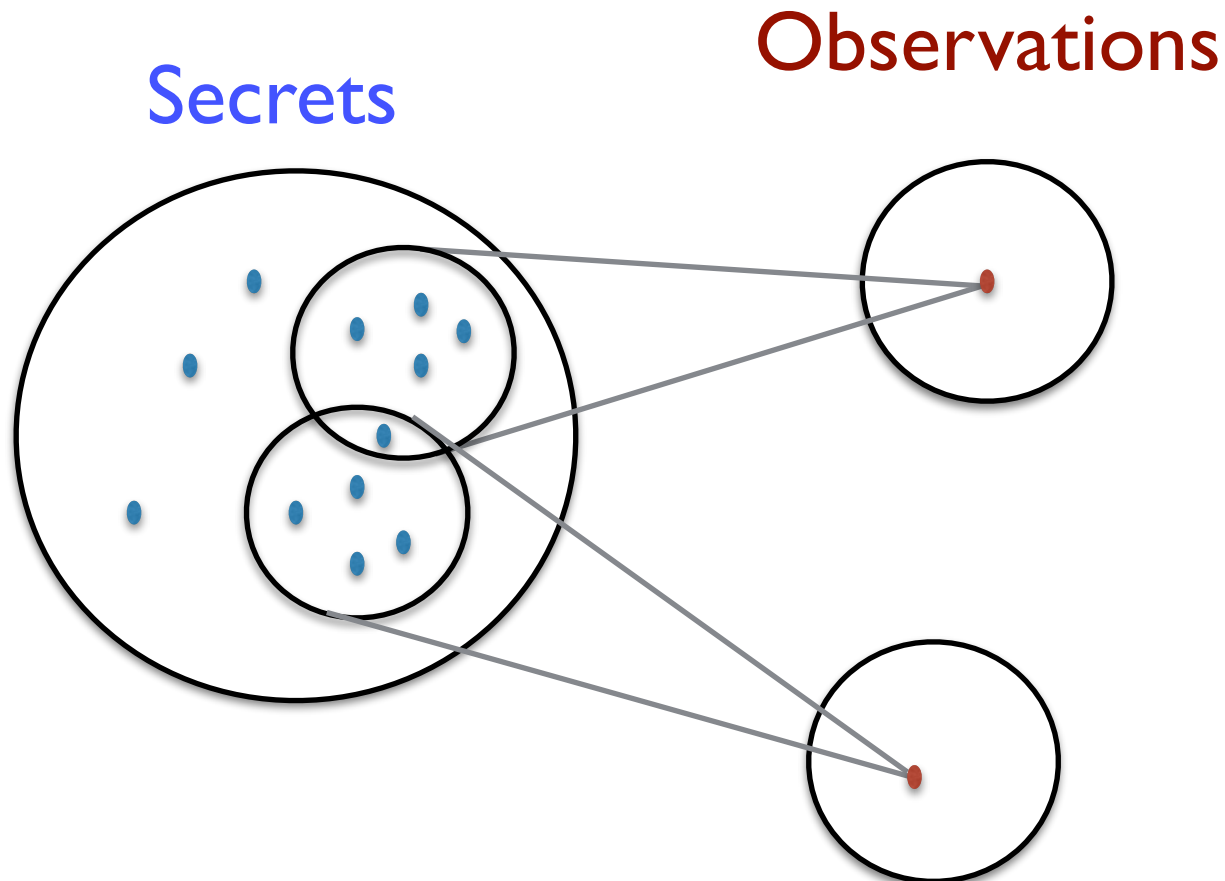
name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

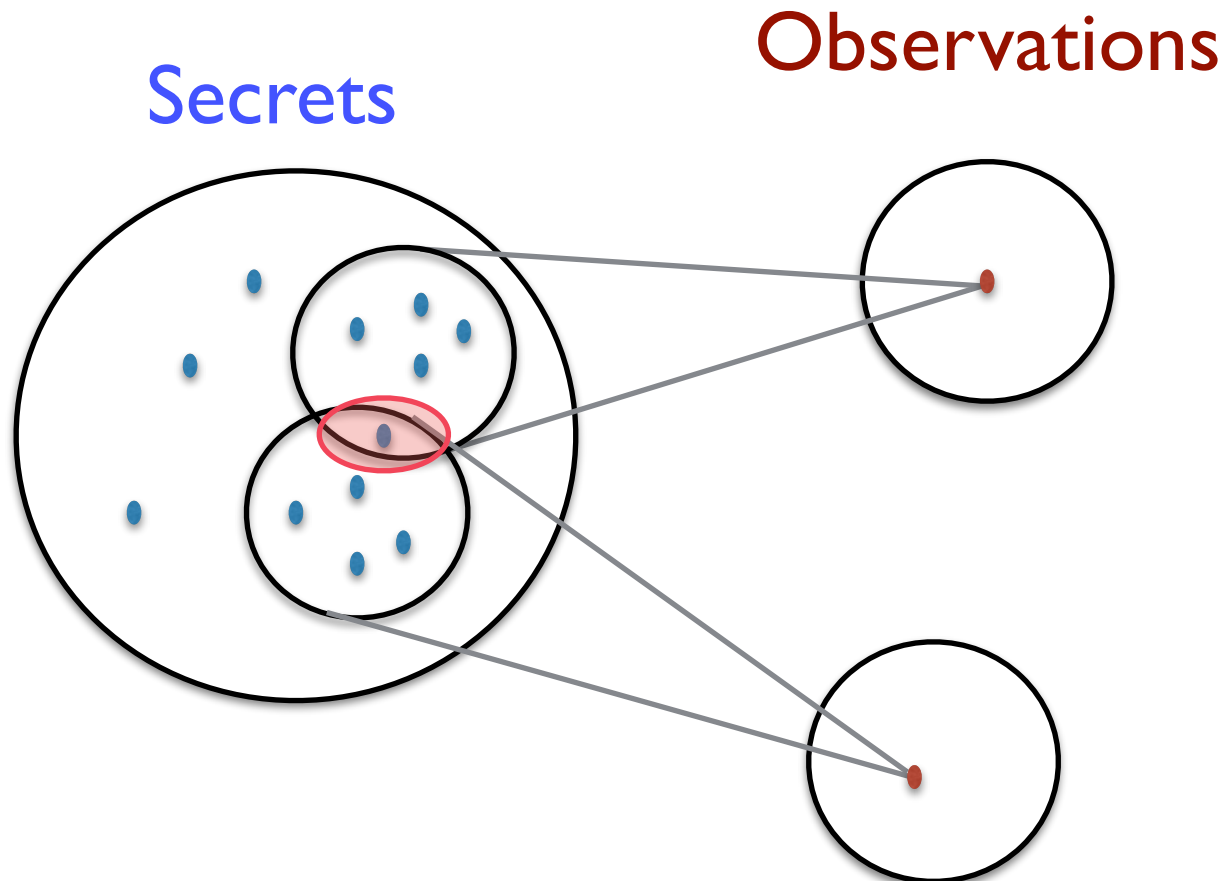
Composition attacks are a general problem of **Deterministic approaches** : They are all based on the principle that one observation corresponds to many possible values of the secret (group anonymity)



Problem of the deterministic approaches: the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletons



Problem of the deterministic approaches: the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletons



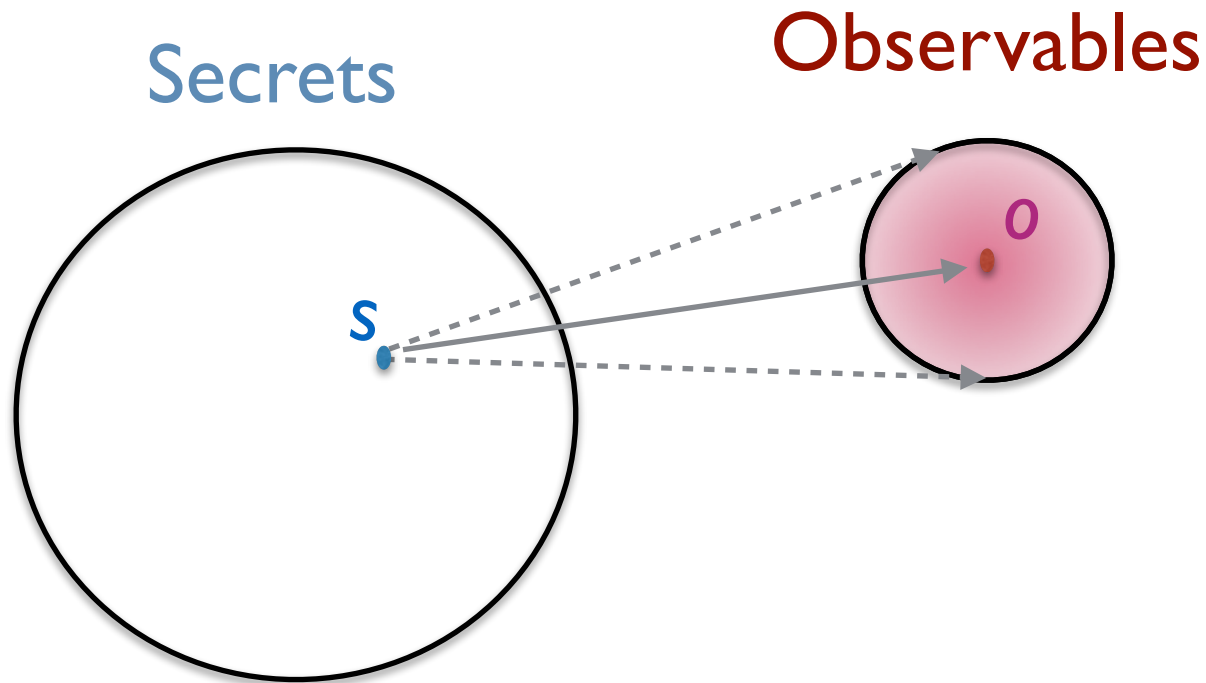
Too bad!!! What can we do?

Use probabilistic approaches!

Most of the state-of-the-art techniques, and in particular differential privacy, are indeed based on randomization

Probabilistic approaches

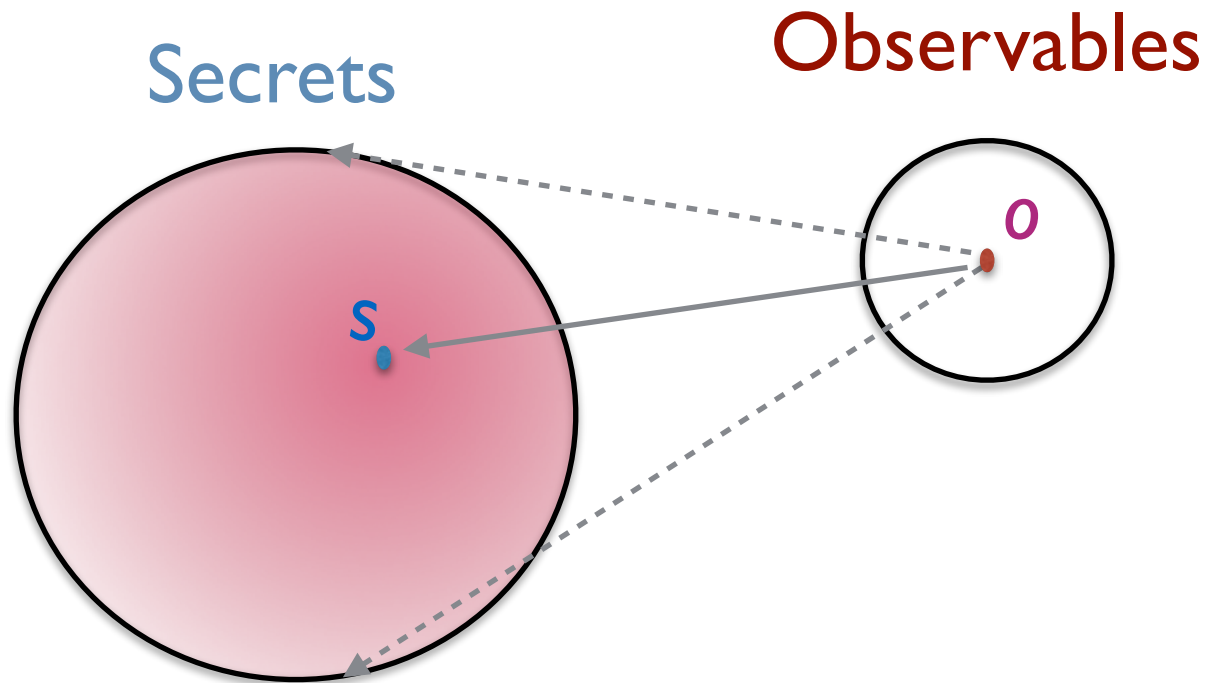
Every secret can generate any observable, according to a certain probability distribution.



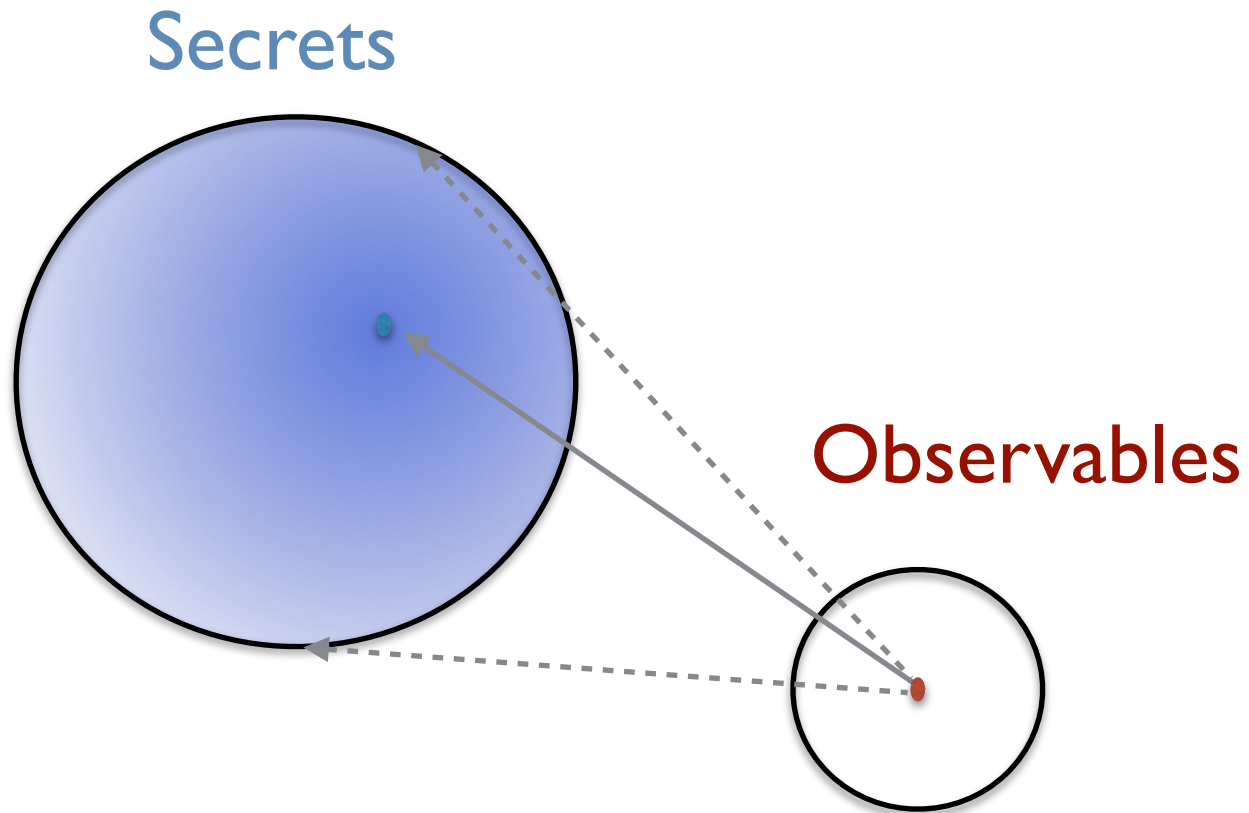
Probabilistic approaches

By the Bayes law

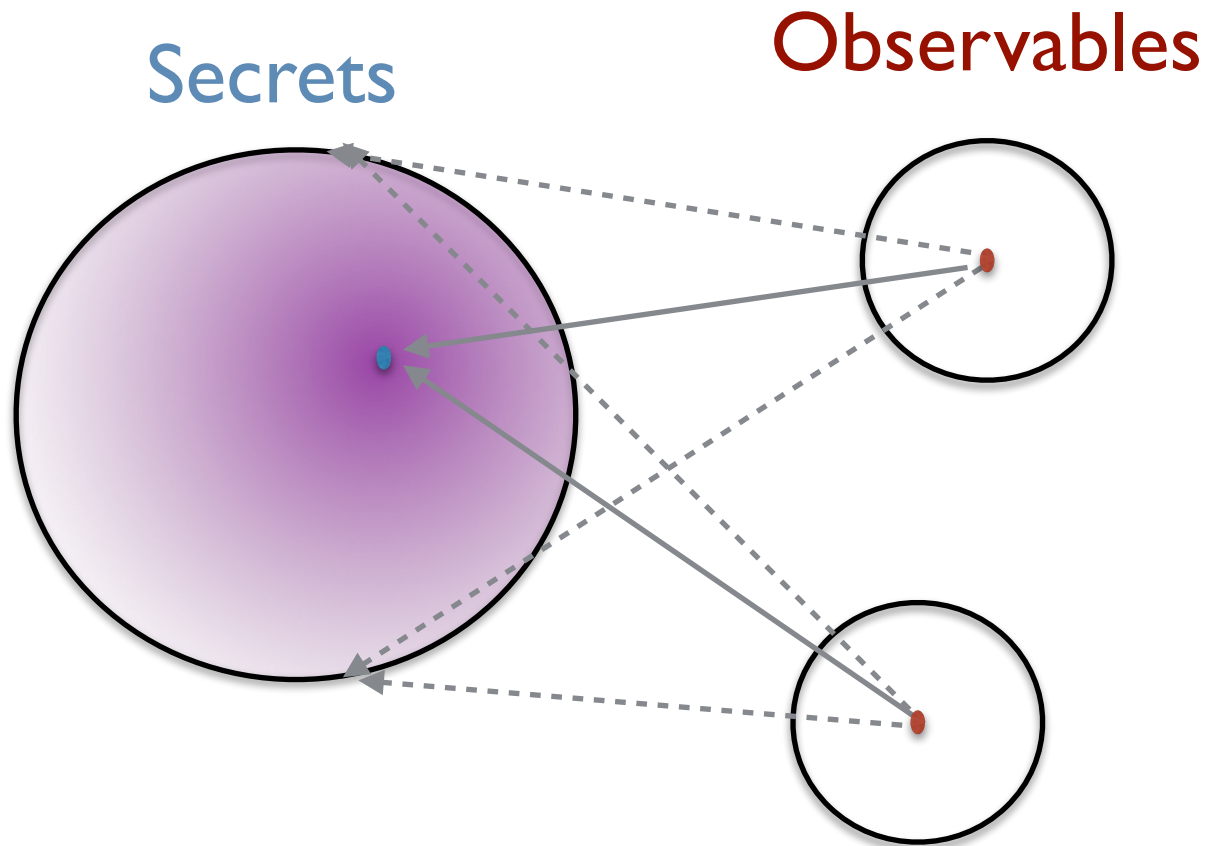
$$p(s|o) \propto p(o|s)$$



Probabilistic approaches



Probabilistic approaches



Randomized approach for DB sanitisation

- Allow accessing the DB only by queries
- Introduce some probabilistic noise on the answer so to obfuscate the link with any particular individual

Noisy answers

minimal age:

40 with probability 1/2

30 with probability 1/4

50 with probability 1/4

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

maximal weight:

100 with prob. $4/7$

90 with prob. $2/7$

60 with prob. $1/7$

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

Noisy answers

Even if he combines the answers, the adversary cannot tell for sure whether a certain person has the disease

name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

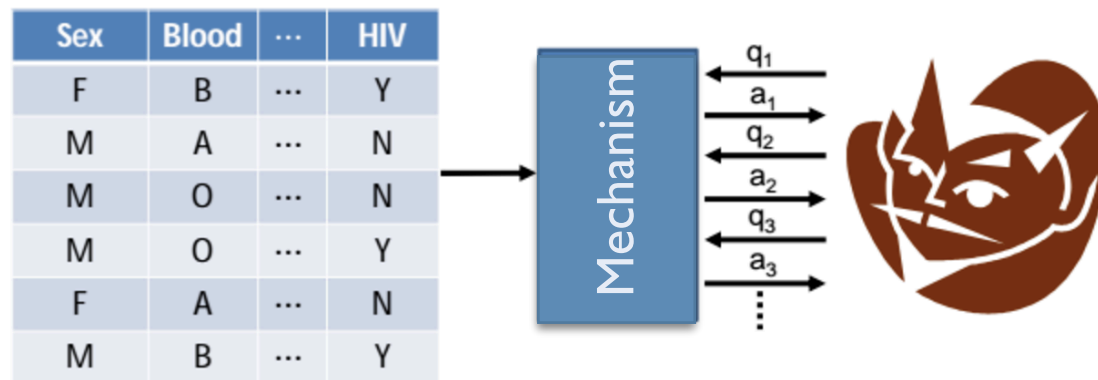
Alice	Bob
Carl	Don
Ellie	Frank

Content of the lectures

- Privacy
 - Motivations
 - **Central Differential Privacy**
 - Local Differential Privacy
 - Privacy vs Utility
- Fairness
 - Motivations
 - Some notions of fairness

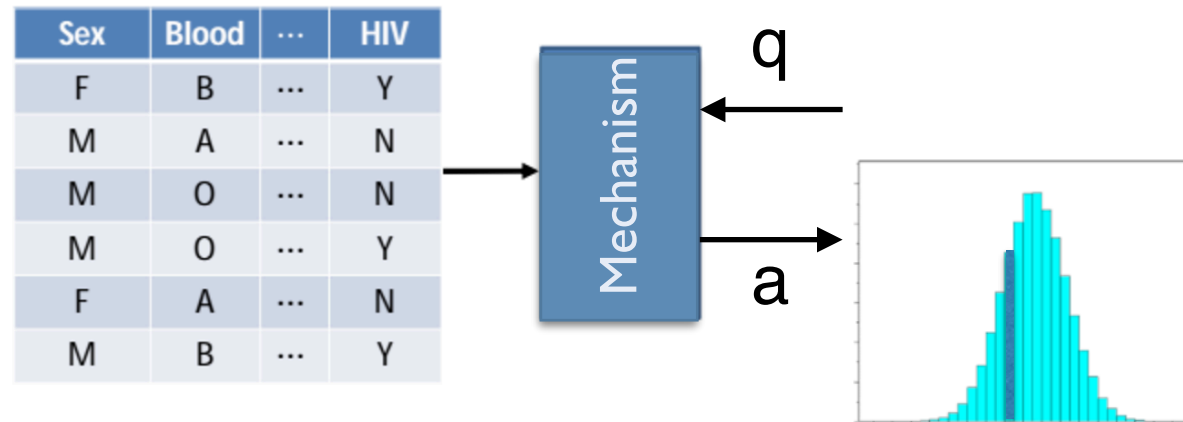
We assume the following setting:

- Centralized model (i.e., the data curator is *trusted*)
- Micro-data are not publicly accessible. The information can only be accessed by querying the DB



In this context, the access to the DB is via an interface (**mechanism**) which receives the queries, computes the answers and sanitises them before reporting them

Randomized mechanisms



- A randomized mechanism (for a certain query) reports an answer generated randomly according to some **probability distribution**
- We need to choose carefully the distribution, so to get the desired **privacy guarantees**, while maintaining a good **utility** for the query
- To find a good trade-off between privacy and utility, and to reason about them, we need formal, rigorous definitions of these notions.
- A definition of privacy that has become very popular: **Differential Privacy [Cynthia Dwork, ICALP 2006]**

Databases

- A record is an element v from some domain \mathcal{V} of values. In general \mathcal{V} is a structured domain, i.e., it is a product of domains corresponding to the attributes. But for our purposes the structure is not relevant and in general we will ignore it
- A **database** (or dataset) of n records is an element of $\mathcal{X} = \mathcal{V}^n$. We will represent the elements of \mathcal{X} by x, x_1, x_2, \dots
- We will assume a probability distribution on distribution on \mathcal{V} and \mathcal{X} and indicate by V, X the respective random variables

Examples:

$\mathcal{V} = \text{integers}$

x

20
14
51
75

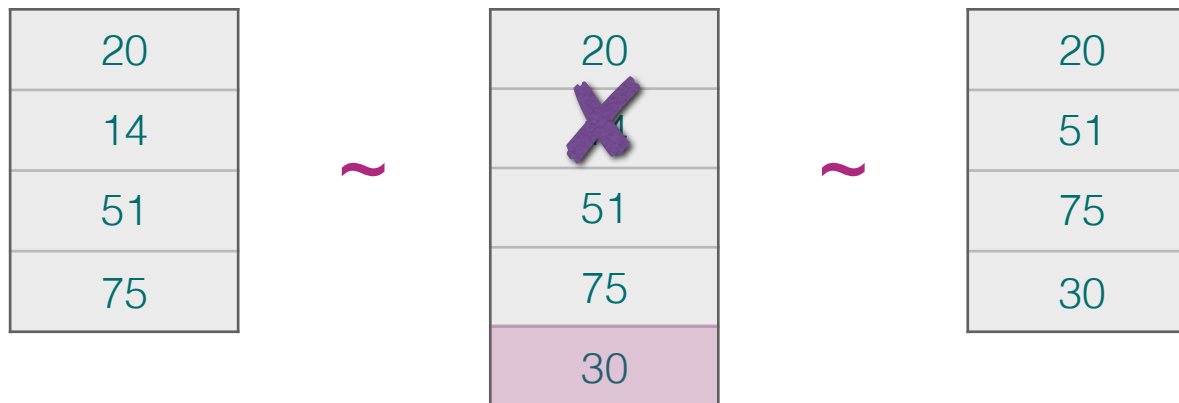
$\mathcal{V} = \text{names} \times \text{integers}$

x

John	20
Mary	14
Dale	51
Anna	75

Adjacency

- Two databases x_1, x_2 are **adjacent** if they differ for exactly one record. We will indicate this property with the notation $x_1 \sim x_2$
- $x_1 \sim x_2$ represent the fact that x_1 and x_2 differ for the information relative to an individual. Either this individual has been added to x_2 , or he has been removed from x_2 .



The adjacency relation is symmetric but not transitive

Queries

- (The answer to) a query f can be seen as a function from the set of databases $\mathcal{X} = V^n$ to a set of values \mathcal{Y} . Namely,

$$f : \mathcal{X} \rightarrow \mathcal{Y}$$

- $y = f(x)$ is the **true answer** of the query f on the database x .
- For a given f , the distribution π on \mathcal{X} also induces a distribution on \mathcal{Y} . We will denote by Y the random variable associated to the distribution on \mathcal{Y} .

Example:

f = average of all values in the DB

	20
	14
x	51
	75

$$f(x) = (20+14+51+75)/4 = 40$$

Randomized mechanisms

- A randomized mechanism for the query f is any probabilistic function \mathcal{K} from \mathcal{X} to a set of values \mathcal{Z} . Namely,

$$\mathcal{K} : \mathcal{X} \rightarrow \mathcal{D}\mathcal{Z}$$

where $\mathcal{D}\mathcal{Z}$ represents the set of probability distributions on \mathcal{Z} .

- \mathcal{Z} does not necessarily coincide with \mathcal{Y} .
- z drawn from $\mathcal{K}(x)$ is a **reported answer** for the query on the DB x .
- Note that π and \mathcal{K} induce a probability distribution also on \mathcal{Z} . We will denote by Z the random variable associated to this probability distribution

Differential Privacy

Definition (Differential Privacy) \mathcal{K} is ε -differentially-private iff for every pair of databases $x_1, x_2 \in \mathcal{X}$ s.t. $x_1 \sim x_2$ and for every measurable $\mathcal{S} \subseteq \mathcal{Z}$ we have

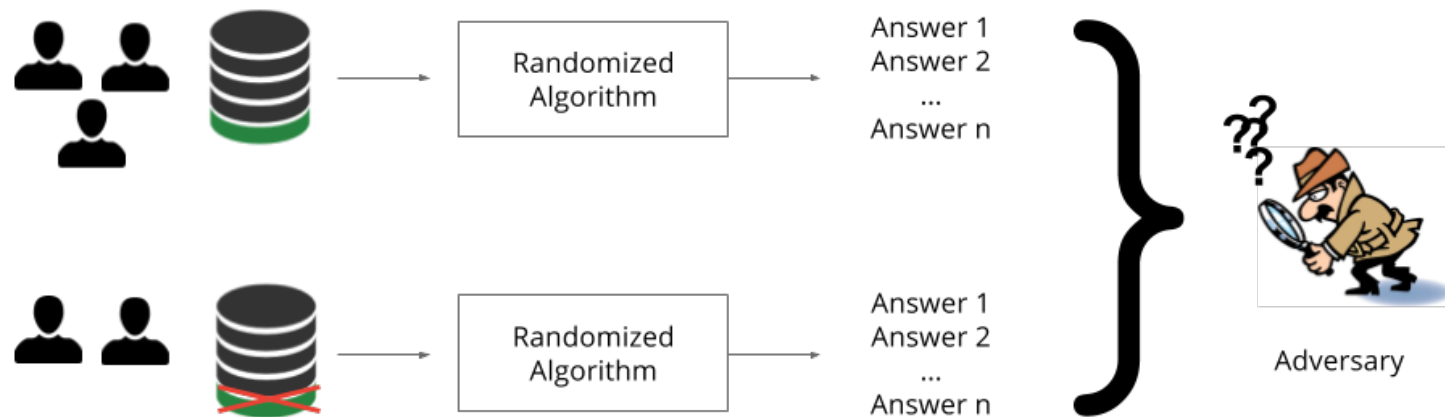
$$p(\mathcal{K}(x_1) \in \mathcal{S}) \leq e^\varepsilon p(\mathcal{K}(x_2) \in \mathcal{S})$$

where $p(\mathcal{K}(x) \in \mathcal{S})$ represents the probability that \mathcal{K} applied to x report an answer in \mathcal{S}

Note: $p(\mathcal{K}(x) \in \mathcal{S})$ represents a conditional probability. We will write it as $p(Z \in \mathcal{S} | X = x)$ when we need to make this fact more explicit.

Meaning of Differential Privacy

Differential privacy essentially means that the presence or absence of an individual in a DB, does not make much difference for the information that the adversary acquires by querying the DB.



Hence an individual does not risks much by accepting that his data are collected in the DB

Examples of mechanisms

Let us assume that we have databases containing as values \mathcal{V} the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer

Examples of mechanisms

Let us assume that we have databases containing as values \mathcal{V} the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer

- Consider the mechanism that always reports the true answer. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer

- Consider the mechanism that always reports the true answer. Is it differentially private ?

No. It's not ϵ -DP for any ϵ

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

- Consider the mechanism that always reports 150. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from 50 to 250 (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

- Consider the mechanism that always reports 150. Is it differentially private ?

Yes. It's ϵ -DP in the strong sense, i.e., for $\epsilon = 0$.

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

- Consider the mechanism that always reports 150. Is it differentially private ?

Yes. It's ϵ -DP in the strong sense, i.e., for $\epsilon = 0$

However, it's totally useless !

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

- Consider the mechanism that reports **100** if the true answer is less than **150**, and **200** otherwise. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

- Consider the mechanism that reports **100** if the true answer is less than **150**, and **200** otherwise. Is it differentially private ?

No. It's a bit more useful than the previous one, but it is not ϵ -DP for any ϵ

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

Consider the mechanism that reports the true answer with probability $\frac{e^\epsilon}{200+e^\epsilon}$, and every other integer in $[50, 250]$ with probability $\frac{1}{200+e^\epsilon}$. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values V the heights of people, in cm, ranging from **50** to **250** (integers). Let us assume that the query is: the average age of the people in the data base, rounded to the next integer.

Consider the mechanism that reports the true answer with probability $\frac{e^\epsilon}{200+e^\epsilon}$, and every other integer in $[50, 250]$ with probability $\frac{1}{200+e^\epsilon}$. Is it differentially private ?

Yes. It's ϵ -DP

Not very useful though. The probability that we get a totally useless answer is very high.

We will see some more useful mechanisms later

Properties of differential privacy

- Two important properties that have made differential privacy so successful:
 - Independence from the side knowledge of the adversary
 - Compositionality

Independence from the side knowledge of the adversary

- The distribution π on the databases is called prior, i.e., prior to the reported answer
- π represents the knowledge that a potential adversary has about the database (before knowing the answer of \mathcal{K})
- We note that the definition of DP does not depend on π . This is a very good property, because it means that we can design mechanisms that satisfy DP without taking the knowledge of the adversary into account: the same mechanism will be good for all adversaries.

Compositionality

- Differential privacy is **compositional**, namely: given two mechanisms \mathcal{K}_1 and \mathcal{K}_2 on \mathcal{X} that are respectively ε_1 and ε_2 -differentially private, their composition $\mathcal{K}_1 \times \mathcal{K}_2$ is $(\varepsilon_1 + \varepsilon_2)$ -differentially private.

Note: $\mathcal{K}_1 \times \mathcal{K}_2$ is defined by the following property: if $\mathcal{K}_1(x)$ reports z_1 and $\mathcal{K}_2(x)$ reports z_2 , then $(\mathcal{K}_1 \times \mathcal{K}_2)(x)$ reports (z_1, z_2) .

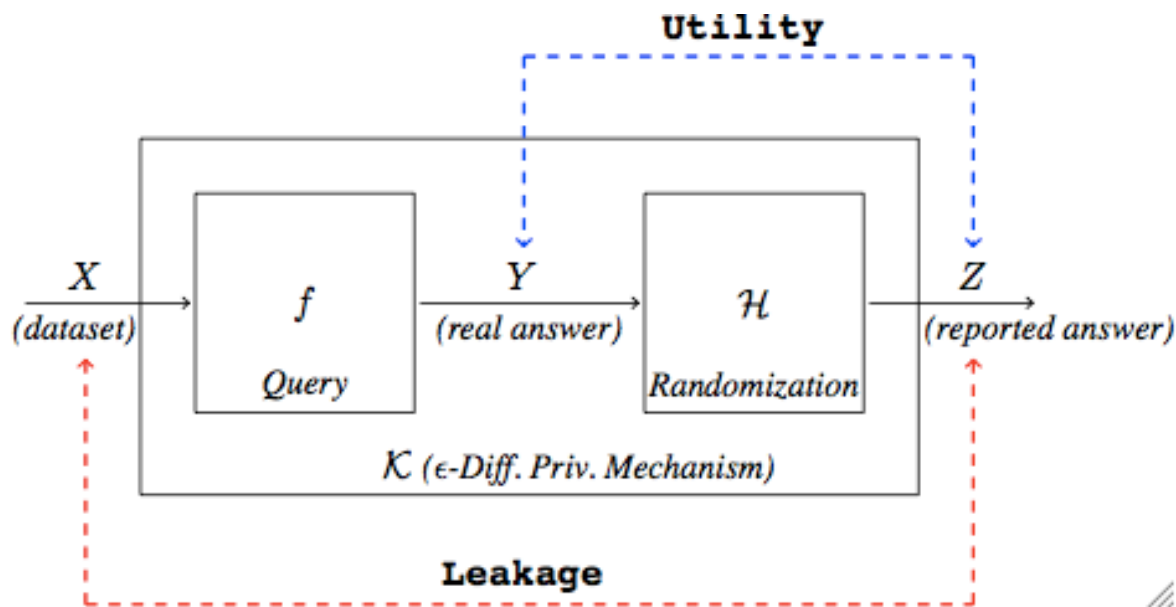
Proof: exercise

- **Privacy budget:** There is an initial budget α associated to the DB. Each time a user asks a query, answered by ε -differentially private mechanism, the budget is decreased by ε . When the budget is exhausted, users are not allowed to ask queries anymore.
Note that the budget is per DB and not per user because users may be colluded.

Some "real" DP mechanisms

Oblivious Mechanisms

- Given $f: \mathcal{X} \rightarrow \mathcal{Y}$ and $\mathcal{K}: \mathcal{X} \rightarrow \mathcal{Z}$, we say that \mathcal{K} is oblivious if it depends only on \mathcal{Y} (not on \mathcal{X})
- If \mathcal{K} is oblivious, it can be seen as the composition of f and a randomized mechanism \mathcal{H} (noise) defined on the exact answers $\mathcal{K} = \mathcal{H} \circ f$



- Privacy concerns the information flow between the databases and the reported answers, while utility concerns the information flow between the correct answer and the reported answer

A typical oblivious DP mechanism: Laplace noise

- Randomized mechanism for a query $f: \mathcal{X} \rightarrow \mathcal{Y}$.
- A typical randomized method: **add Laplace noise to $y=f(x)$** .
Namely, report z with a probability density function defined as:

$$dP_y(z) = c e^{-\frac{|z-y|}{\Delta f} \varepsilon}$$

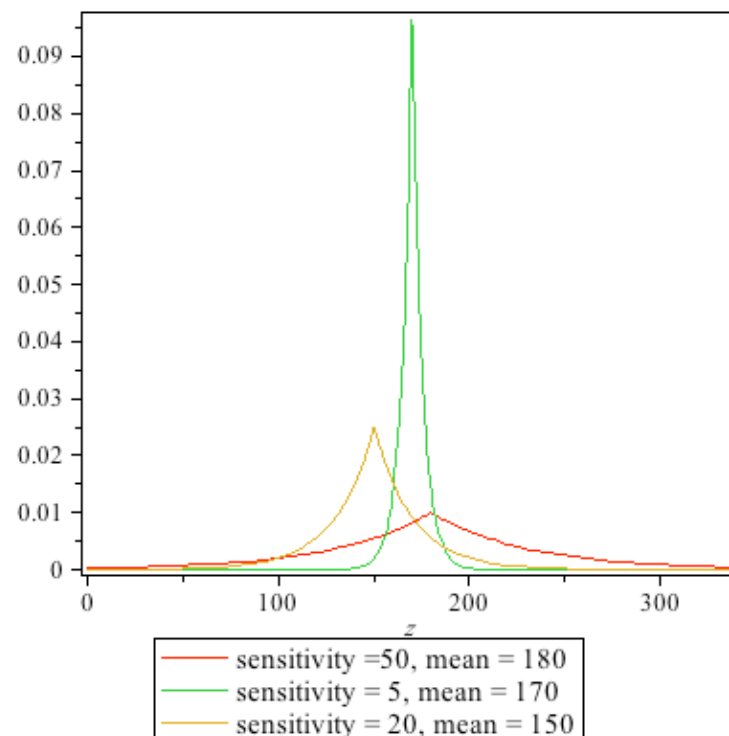
where Δf is the *sensitivity* of f :

$$\Delta f = \max_{x \sim x' \in \mathcal{X}} |f(x) - f(x')|$$

($x \sim x'$ means x and x' are adjacent,
i.e., they differ only for one record)

and c is a normalization factor:

$$c = \frac{\varepsilon}{2 \Delta f}$$



Example of Laplace Mechanism

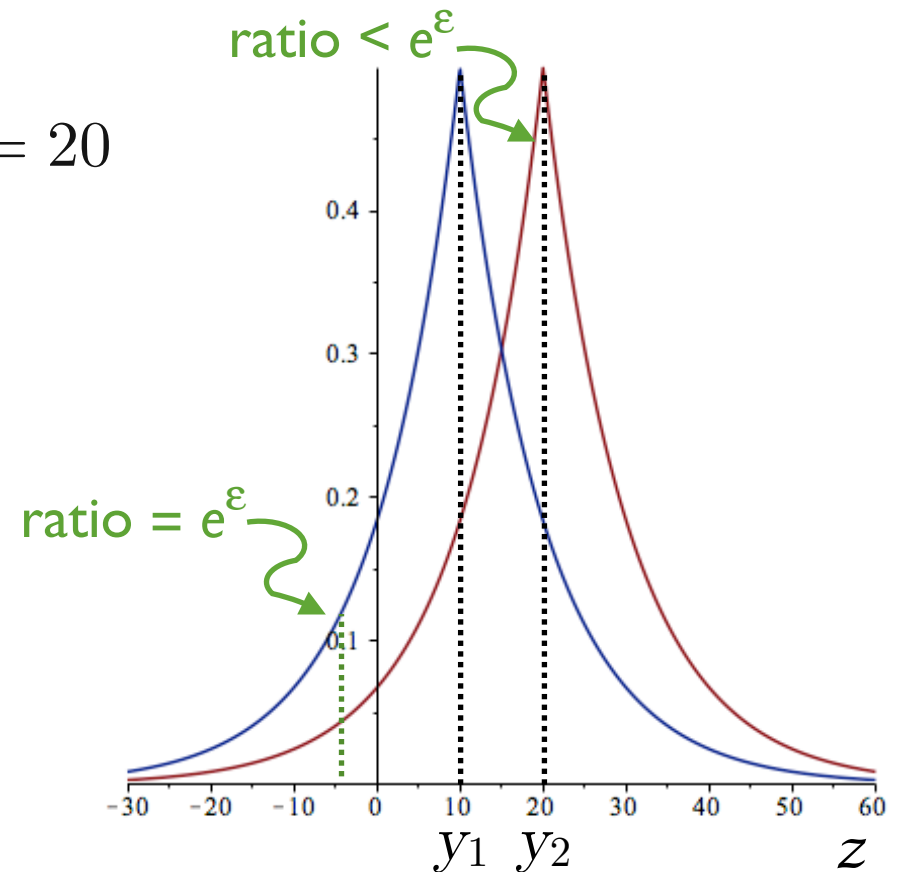
- $\epsilon = 1$
- $\Delta_f = |f(x_1) - f(x_2)| = 10$
- $y_1 = f(x_1) = 10, y_2 = f(x_2) = 20$

Then:

- $dP_{y_1} = \frac{1}{2 \cdot 10} e^{-\frac{|z-10|}{10}}$
- $dP_{y_2} = \frac{1}{2 \cdot 10} e^{-\frac{|z-20|}{10}}$

The ratio between these distribution is

- $= e^\epsilon$ outside the interval $[y_1, y_2]$
- $\leq e^\epsilon$ inside the interval $[y_1, y_2]$



The geometric mechanism

- The Laplacian noise is typically used in the case that \mathcal{Y} (the set of true answers of the query) is a **continuous** numerical set, like the Reals.
- If \mathcal{Y} is a **discrete** numerical set, like the Integers, then the typical mechanism used in this case is the **geometric mechanism**, which is a sort of discrete Laplacian.
- In the geometric mechanism, the probability distribution of the noise is:

$$p(z|y) = c e^{-\frac{|z-y|}{\Delta f} \varepsilon}$$

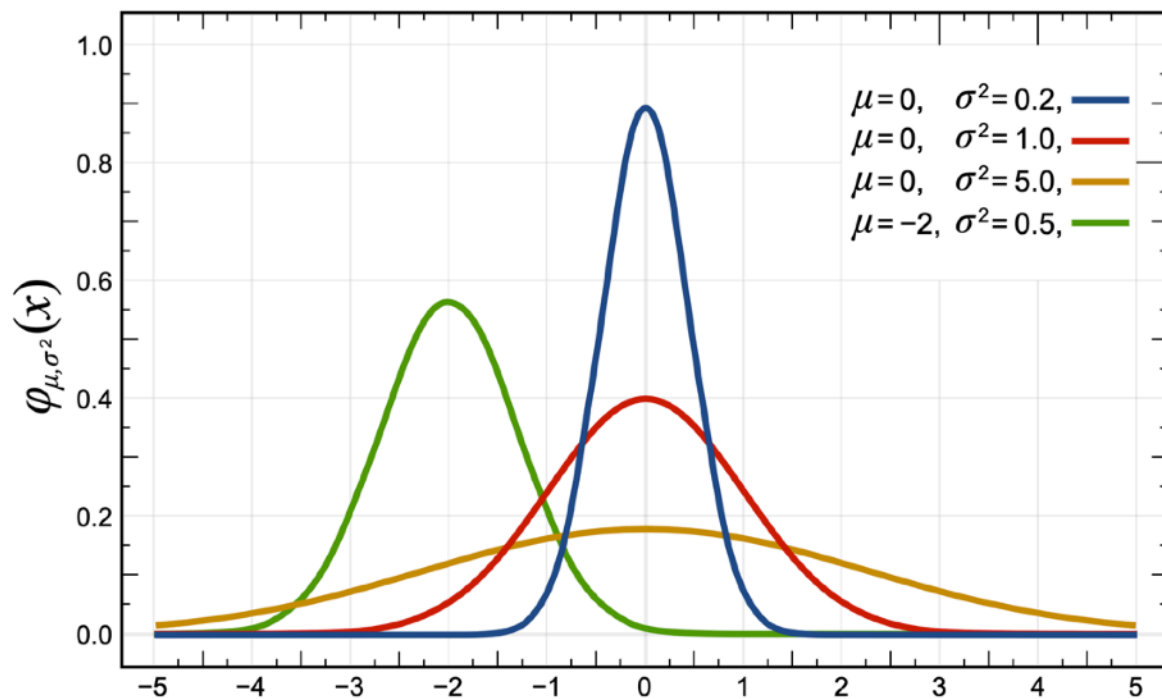
- In this expression, c is a normalization factor, defined so to obtain a probability distribution,
- Δf is the sensitivity of query f

Gaussian noise

The formula for gaussian noise is

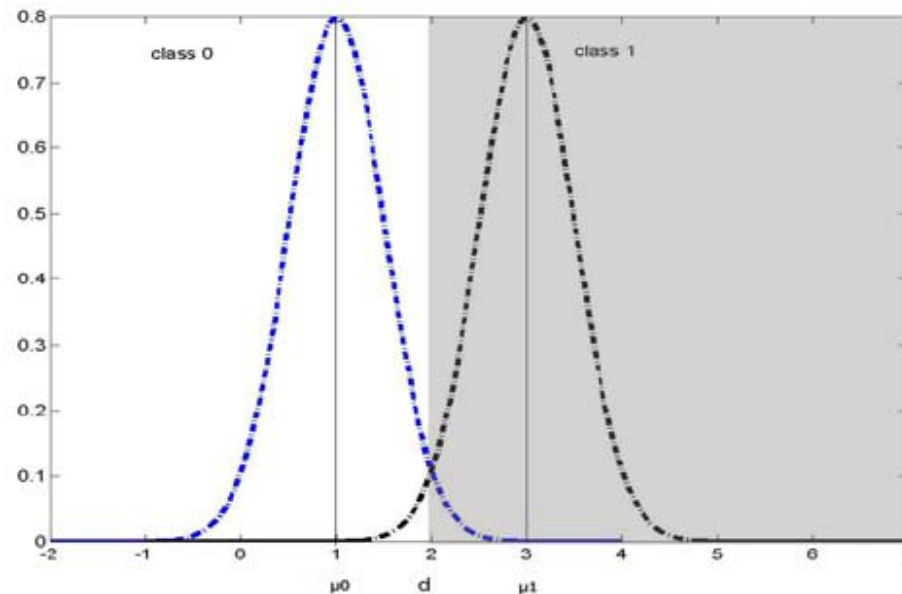
$$c e^{-\frac{(y-z)^2}{\sigma}} \varepsilon$$

where c is a normalization factor and σ is a suitable constant.



Gaussian noise

The gaussian mechanism does not satisfy differential privacy, because on the tail it does decrease too quickly.



However it satisfies a more relaxed form of privacy called (ϵ, δ) -DP

$$p(\mathcal{K}(x_1) \in \mathcal{S}) \leq e^\epsilon p(\mathcal{K}(x_2) \in \mathcal{S}) + \delta$$

Utility

There are various notions of utility.

We will focus on one of the most common ones, namely the utility as expected loss.

Utility as expected loss

Assume:

- π is the prior on \mathcal{Y} (the true answers)
- $p_{\mathcal{K}}$ is the probability associated to the mechanism.
- ℓ is a loss function, that measures how much we “loose” in reporting a noisy answer. Namely, $\ell(y, z)$ is the loss of precision when the true answer is y but the mechanism reports z .

Then: the expected utility loss $\mathcal{U}(\pi, p_{\mathcal{K}}, \ell)$ is defined as:

$$\begin{aligned}\mathcal{U}(\pi, p_{\mathcal{K}}, \ell) &= \mathbb{E}_{\pi, p_{\mathcal{K}}} \ell(y, z) \\ &= \sum_{y, z} \pi(y) p_{\mathcal{K}}(z|y) \ell(y, z)\end{aligned}$$

Optimal mechanisms

- Given a prior π , and a privacy level ϵ , an ϵ -differentially private mechanism K is called **optimal** if it provides the **best utility** among all those which provide ϵ -differential privacy
- A mechanism is **universally optimal** if it is optimal for all priors π

Counting Queries

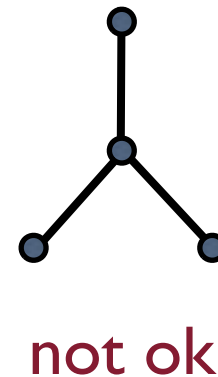
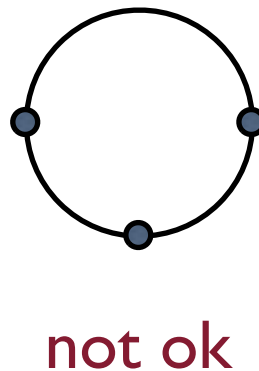
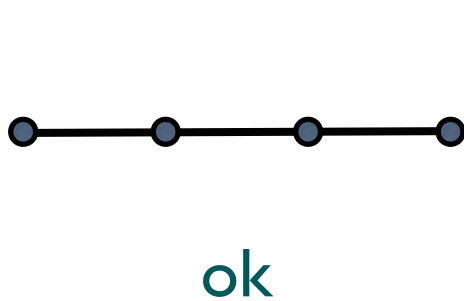
- Counting queries are typical examples of discrete queries. They are of the form: How many individuals in the database satisfy the property \mathcal{P} ?
- Examples:
 - How many individuals in the DB are affected by diabetes?
 - How many diabetic people are obese?

Privacy vs utility: two fundamental results

- I. [Ghosh et al., STOC 2009]
The geometric mechanism is **universally optimal** for counting queries and any monotonic loss function

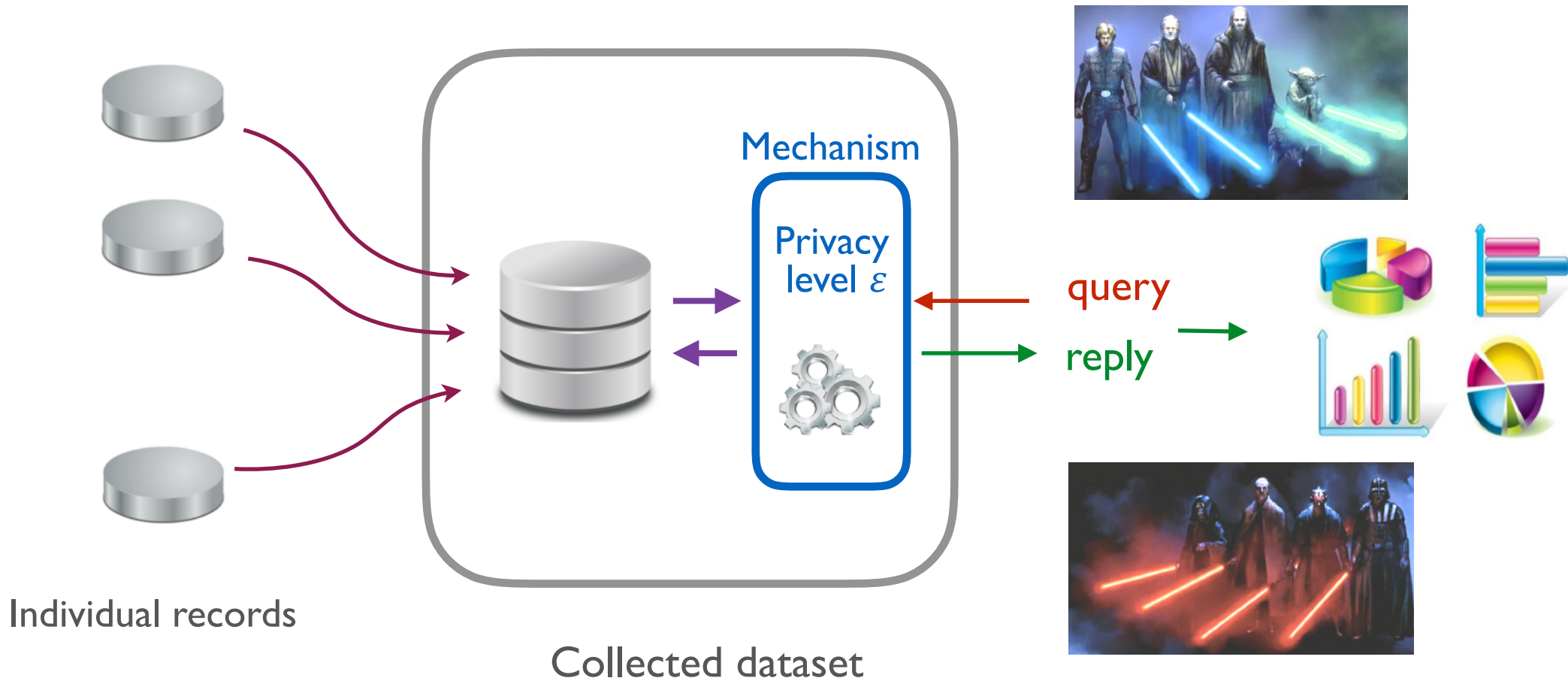
Privacy vs utility: two fundamental results

2. [Brenner and Nissim, STOC 2010] The counting queries are the only kind of queries for which a universally optimal mechanism exists
- This means that for other kind of queries one the optimal mechanism is relative to a specific user.
 - The precise characterization is given in terms of the graph (\mathcal{Y}, \sim) induced by (\mathcal{X}, \sim)

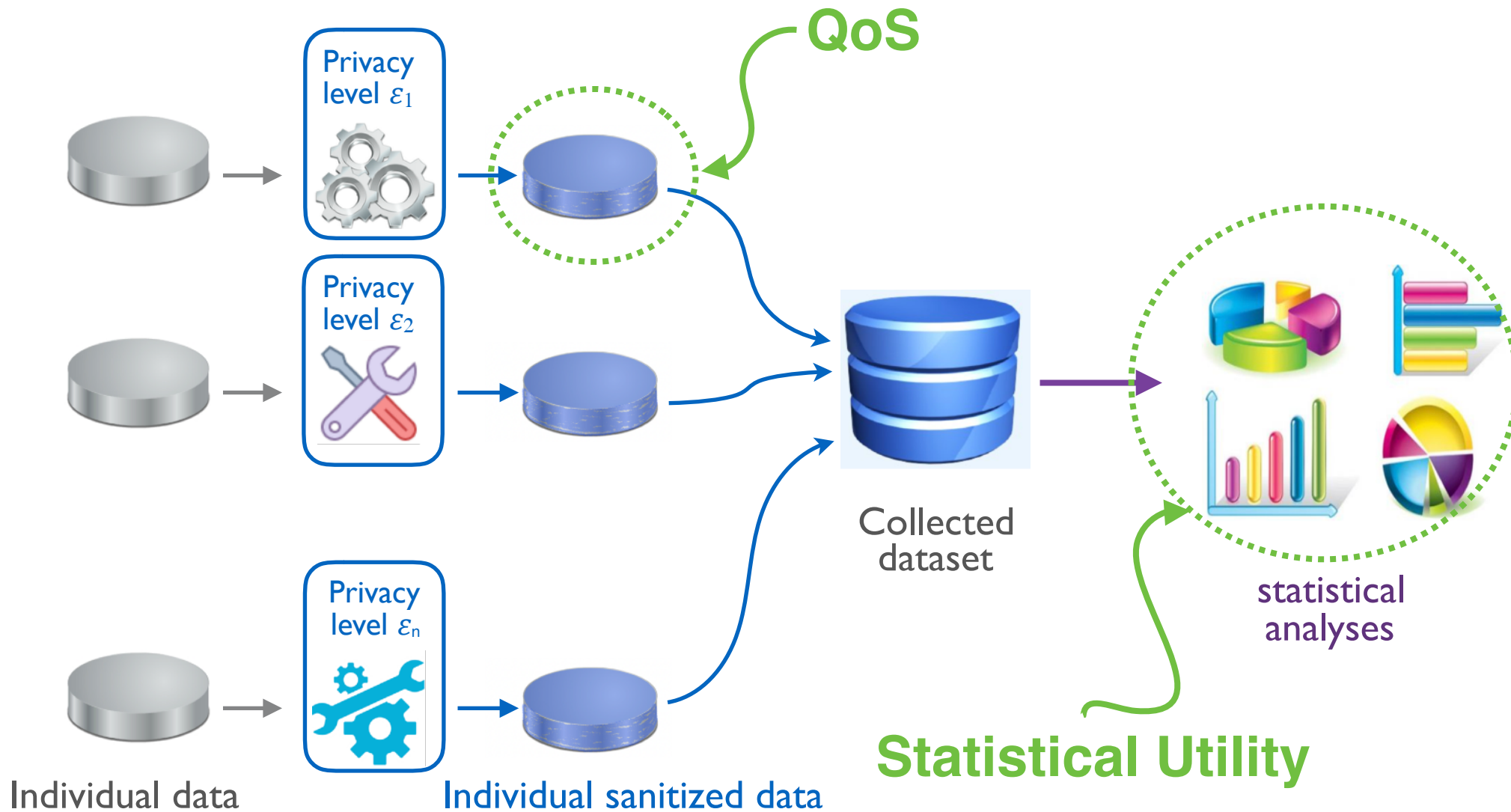


Local Differential Privacy

DP in the Global Model



Local Differential Privacy



Local Differential Privacy

[Jordan & Wainwright '13]

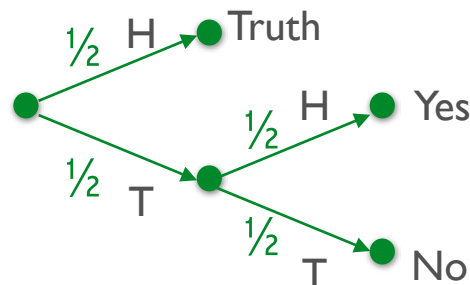
Definition Let \mathcal{X} be a set of possible values and \mathcal{Y} the set of noisy values. A mechanism \mathcal{K} is ϵ -locally differentially private (ϵ -LDP) if for all $x_1, x_2 \in \mathcal{X}$ and for all $y \in \mathcal{Y}$

$$P[\mathcal{K}(x) = y] \leq e^\epsilon P[\mathcal{K}(x') = y]$$

or equivalently, using the conditional probability notation:

$$p(y | x) \leq e^\epsilon p(y | x')$$

For instance, the Randomized Response protocol is $(\log 3)$ -LDP



		y	
		yes	no
x	yes	3/4	1/4
	no	1/4	3/4

The k-RR mechanism (aka flat mechanism)

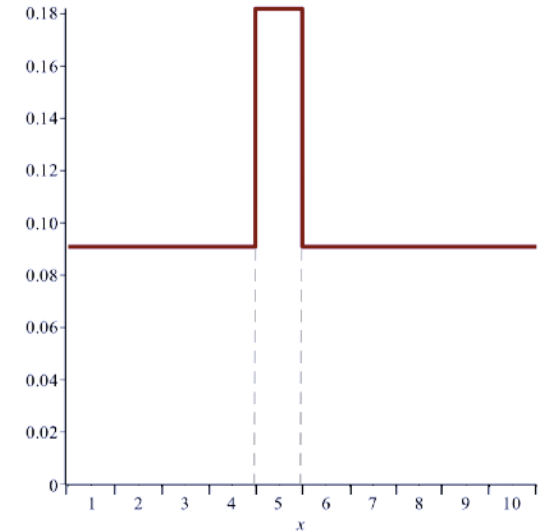
[Kairouz et al, '16]

The flat mechanism is the simplest way to implement LPD.
It is defined as follows:

$$p(y|x) = \begin{cases} c e^\epsilon & \text{if } x = y \\ c & \text{otherwise} \end{cases}$$

where c is a normalization constant.

namely $c = \frac{1}{k - 1 + e^\epsilon}$ where k is the size of the domain



Privacy Properties:

- Compositionality
- Independence from the side knowledge of the adversary

d -privacy: a generalization of DP and LDP [Chatzikokolakis et al., '13]

d -privacy

On a generic domain \mathcal{X} provided with a distance d :

$$\forall x, x' \in \mathcal{X}, \forall z \quad \frac{p(z | x)}{p(z | x')} \leq e^{\varepsilon d(x, x')}$$

generalizes

Differential Privacy

- x, x' are databases
- d is the Hamming distance

Local Differential Privacy

- d is the discrete distance

Properties

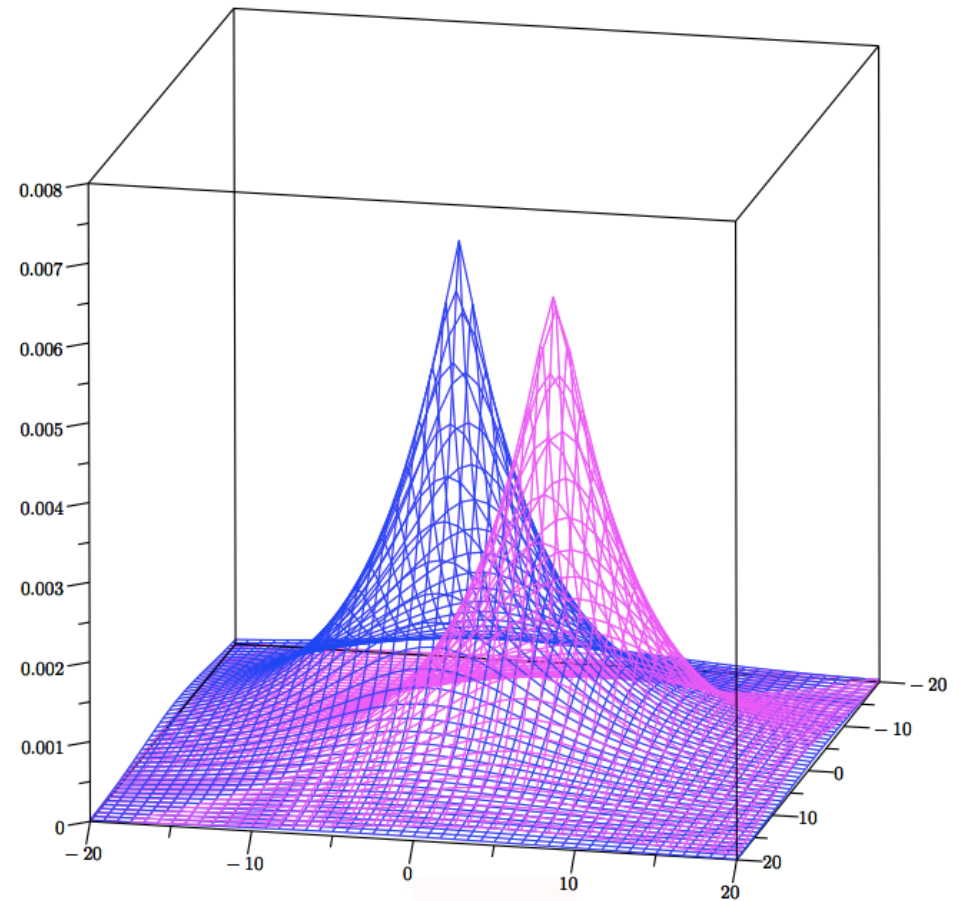
- Like LDP, it can be applied at the user side
- Like DP and LDP, it is compositional

Typical d -private mechanisms

Laplace, Geometric, and their higher-dimension versions

Planar Laplace

$$dp_x(z) = \frac{\epsilon^2}{2\pi} e^{-\epsilon d(x,z)}$$



Used especially for location privacy, where d -privacy is called *geo-indistinguishability*

Statistical Utility

This notion is particularly important for local DP. The goal is to estimate as precisely as possible the true distribution on data from the reported answers.

Statistical utility: The matrix inversion method

[Kairouz et al, '16]

- Let C be the stochastic matrix associated to the mechanism
- Let q be the empirical distribution (derived from the noisy data).
- Compute the approximation of the true distribution as $r = q C^{-1}$

Example Assume $q(Yes) = \frac{6}{10}$ and $q(No) = \frac{4}{10}$. Then:

$$\frac{3}{4} p(Yes) + \frac{1}{4} p(No) = \frac{6}{10}$$

$$\frac{1}{4} p(Yes) + \frac{3}{4} p(No) = \frac{4}{10}$$

From which we derive $p(Yes) = \frac{7}{10}$ and $p(No) = \frac{3}{10}$

		y	
		yes	no
x	yes	$\frac{3}{4}$	$\frac{1}{4}$
	no	$\frac{1}{4}$	$\frac{3}{4}$

Statistical utility: The matrix inversion method

Problem 1: C must be invertible

Problem 2: Assume $q(Yes) = \frac{4}{5}$ and $q(No) = \frac{1}{5}$. Then:

$$\begin{aligned}\frac{3}{4} p(Yes) + \frac{1}{4} p(No) &= \frac{4}{5} \\ \frac{1}{4} p(Yes) + \frac{3}{4} p(No) &= \frac{1}{5}\end{aligned}$$

	y	
	yes	no
x	yes	$\frac{3}{4}$ $\frac{1}{4}$
	no	$\frac{1}{4}$ $\frac{3}{4}$

From which we derive $p(Yes) = \frac{11}{10}$ and $p(No) = -\frac{1}{10}$

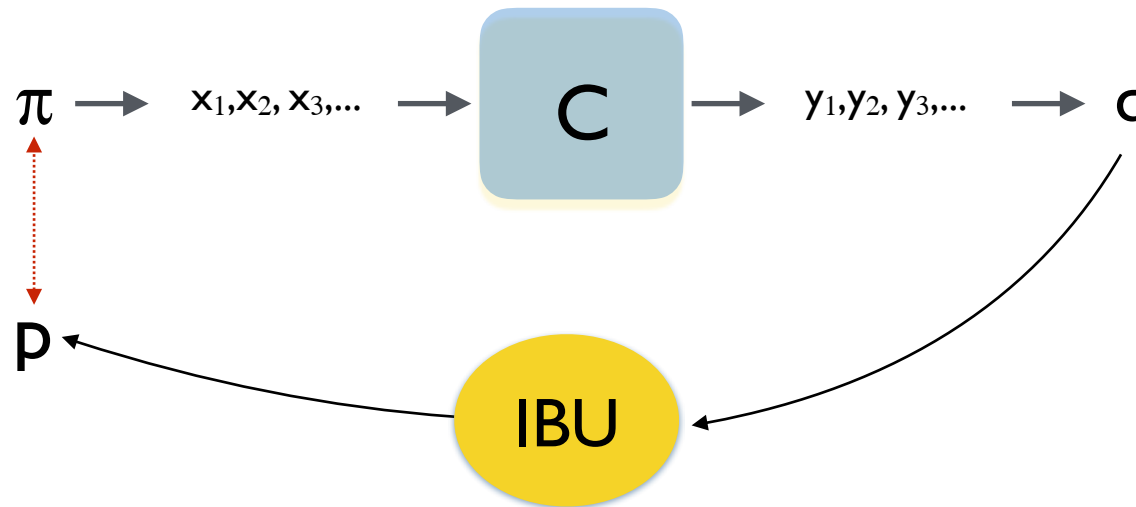
Statistical utility: The matrix inversion method

$r = q C^{-1}$ may not be a distribution because it may contain negative elements. In order to try to obtain the true distribution π we can either:

- set to 0 all the negative elements, and renormalize, or
- project r on the simplex.

The resulting distribution however usually is not the best approximation of the original distribution.

A more general and principled approach: Iterative Bayesian Update



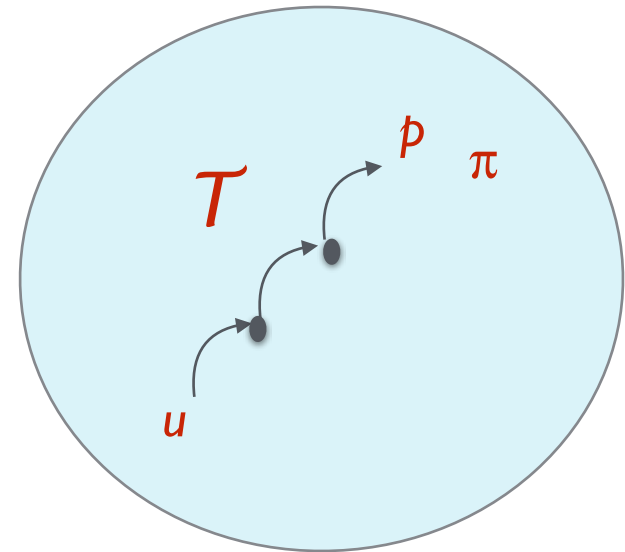
The IBU:

- is based on the **Maximization-Expectation** method
- produces a **Maximum Likelihood Estimator** p of the true distribution π
- If C is invertible, the MLE is unique and as the number of samples grows it converges to π

The Iterative Bayesian Update

- Define $p^{(0)}$ = any distribution (for ex. the uniform distribution)
- Repeat: Define $p^{(n+1)}$ as the Bayesian update of $p^{(n)}$ weighted on the corresponding element of q , namely:

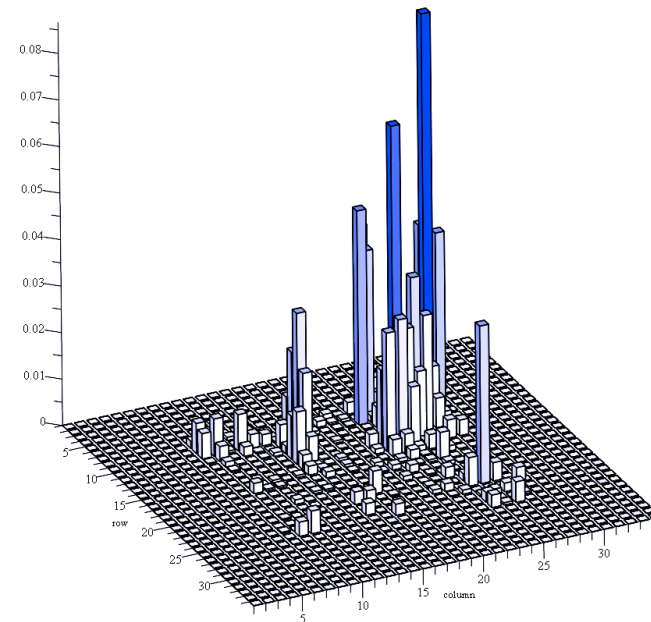
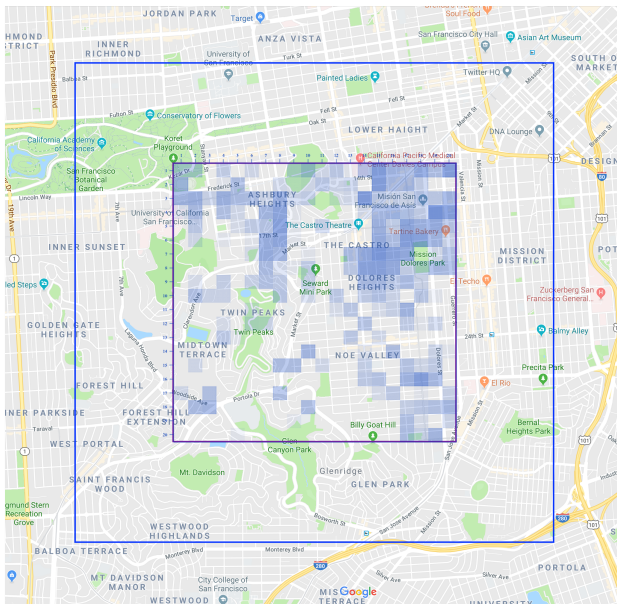
$$p_x^{(n+1)} = \sum_y q_y \frac{p_x^{(n)} C_{xy}}{\sum_z p_z^{(n)} C_{zy}}$$



- Note that $p^{(n+1)} = T(p^{(n)})$
- When C is invertible, T has unique fix point (the MLE)
- Open problem: in some cases (with few samples) the MLE may not be the best estimation of the true distribution. We are trying to devise corrective methods.

Comparison between LPD and d -privacy Experiments on the Gowalla dataset

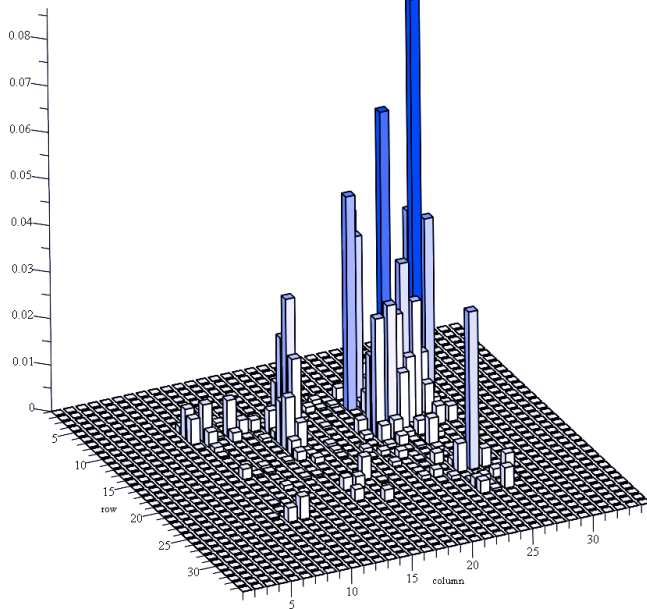
- Gowalla is a dataset of geographical checkins in several cities in the world
- We have used it to compare the statistical utility of kRR and Planar Laplace with the respective ϵ calibrated so to satisfy the same privacy constraint:
same level of privacy within about 1 Km²



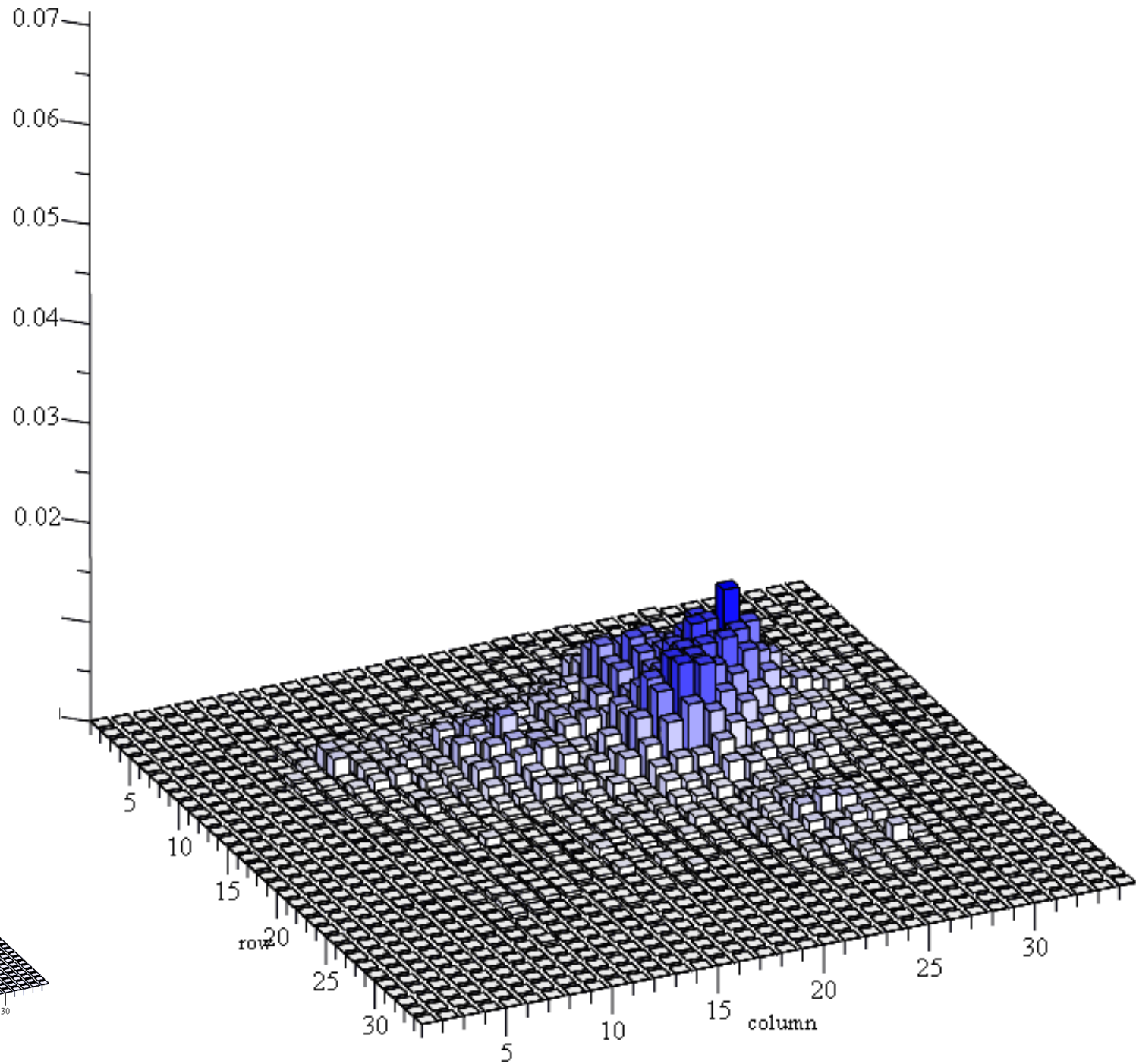
Gowalla checkins in an area of 3x3 km² in San Francisco downtown (about 10K checkins)

The Planar Laplace mechanism

$$\epsilon = \ln(2)$$



The real distribution

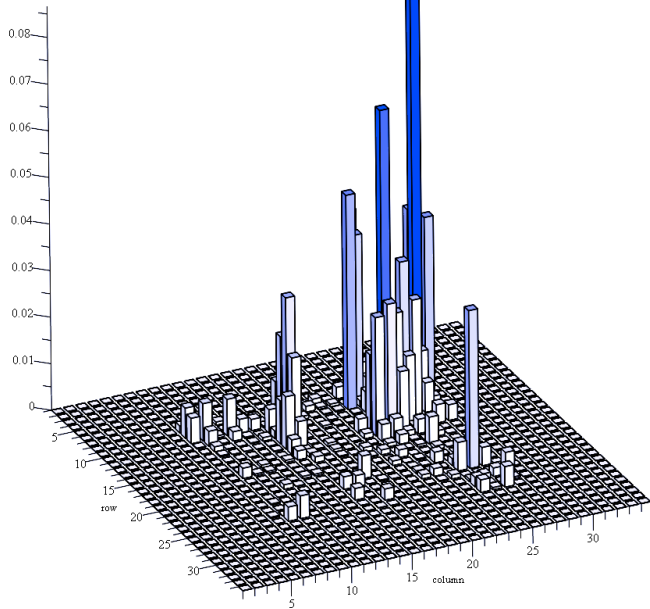


The noisy distribution and the result of the IBU (300 iterations)

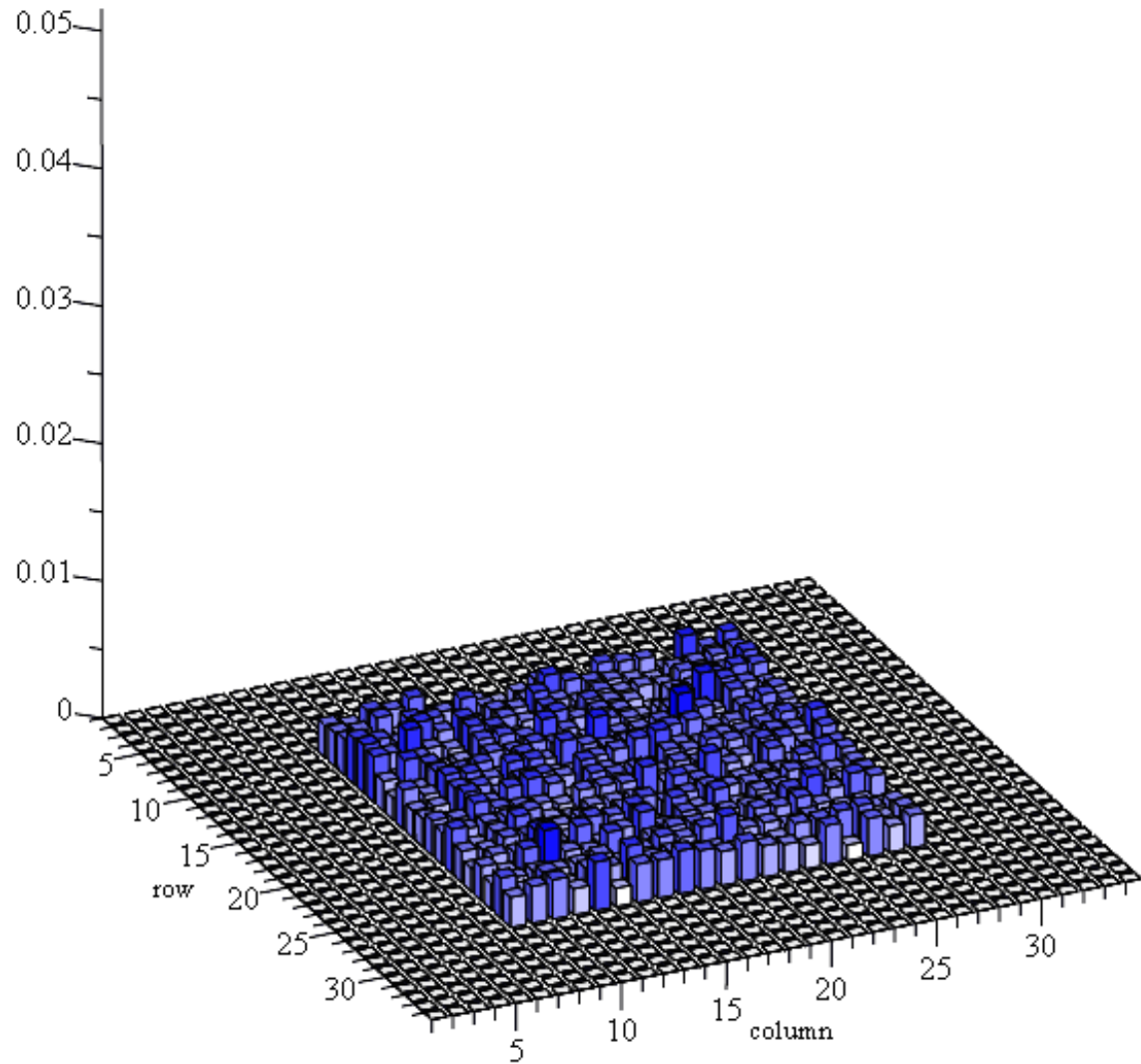
$n = 0.$

The kRR mechanism

$$\epsilon = \ln(8)$$

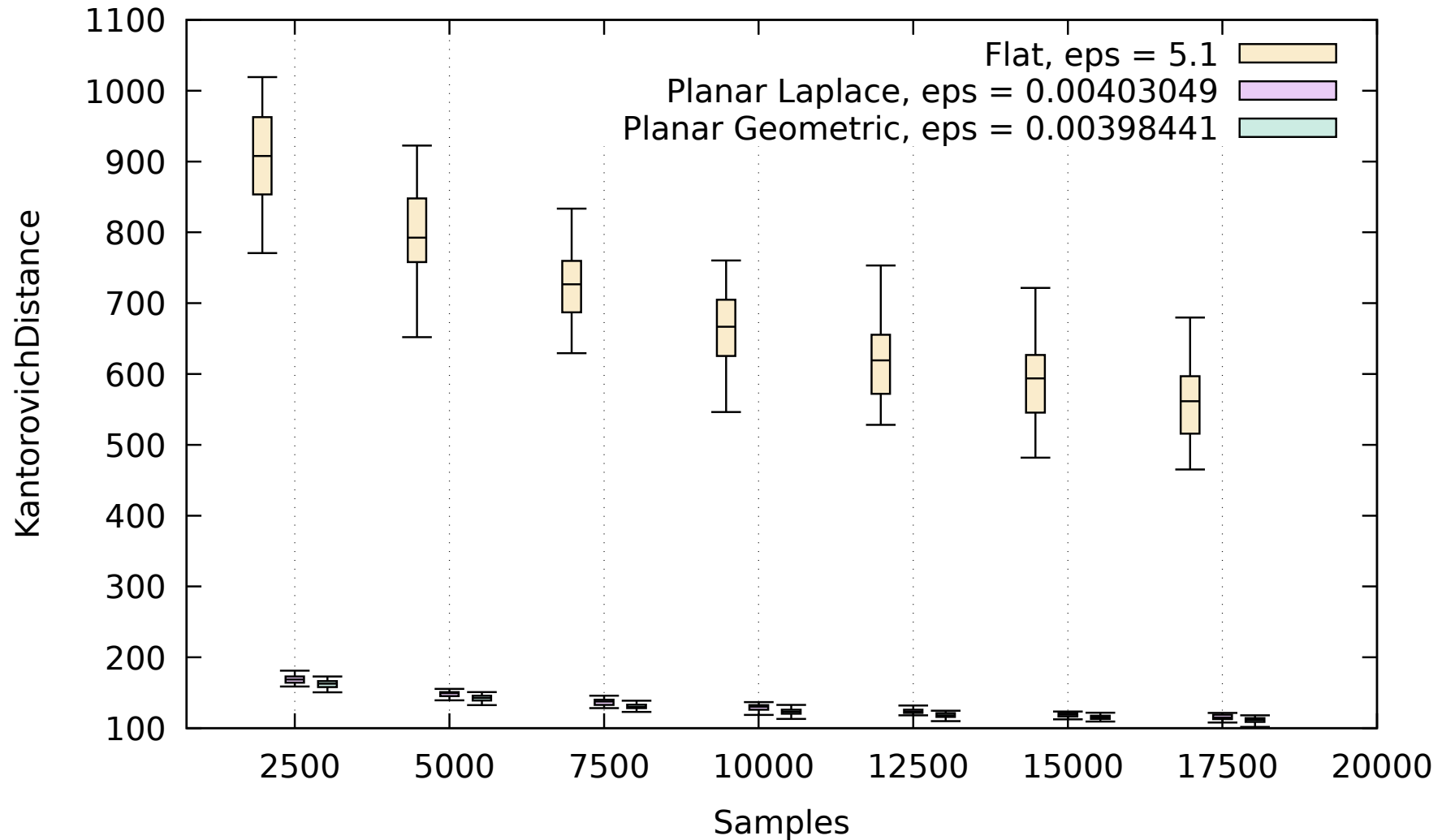


The real distribution

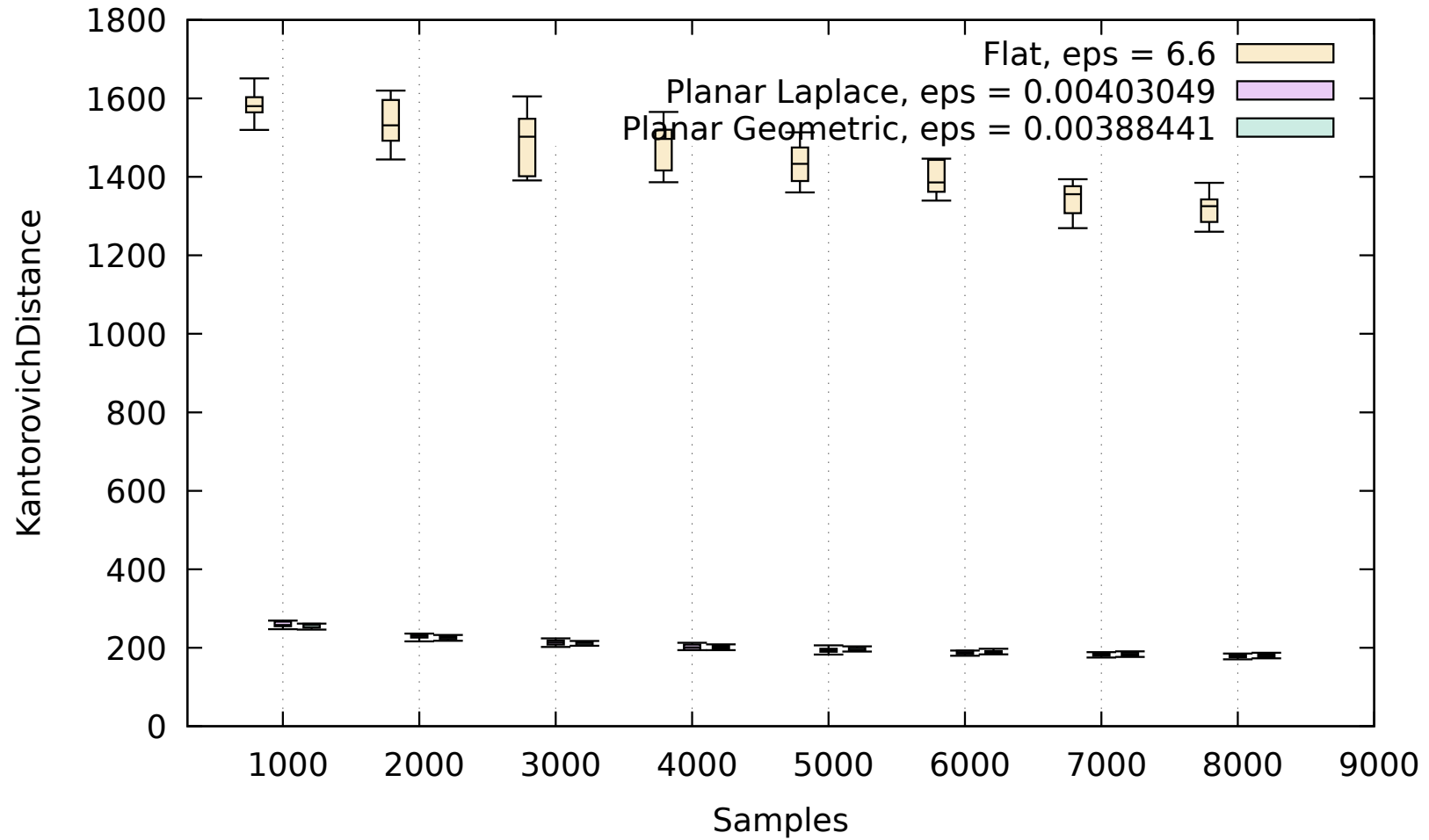


The noisy distribution and the result of the IBU (500 iterations)

Evaluation: San Francisco



Evaluation: Paris



Privacy preserving machine learning

In machine learning, the typical privacy issue is the **membership inference** in the training set. Namely the attacker is trying to infer whether a certain sample is part of the training set or not.

Models of attacks:

- **Black box:** The attacker can only see the answers of the model on a test set
- **White box:** The attacker has access also to the internals of the model (architecture, weights of the nodes, etc.)
- **Grey box:** A combination of the above

The effectiveness of these attacks has been widely documented. Of course the white box attacks are more effective than the black box ones, but also in the black box, surprisingly, the attacker can guess the presence or absence of the target sample with high probability of being correct.

Counter measures

Central differential privacy has been successfully applied in "standard" machine learning. The method typically consists in adding Laplace or Gaussian noise to the weights updates during the gradient descent.

In ML, the typical metric for **utility** is the **accuracy of the model**

Federated learning

A recent topic of research: (white box) attacks and counter measures in **federated learning**.

- In federated learning, we assume that there are various individuals or organizations, each with its own training data, that they do not want to share
- The gradient descent is done in a distributed way: each organization computes the update based on its own data, then sends it to the central coordinator, which combines them
- In federated learning makes sense to consider a very powerful white box attacker that can **access the individual updates** each time.

Content of the lectures

- Privacy
 - Motivations
 - Central Differential Privacy
 - Local Differential Privacy
 - Privacy vs Utility
- **Fairness**
 - Motivations
 - Some notions of fairness

Content of the lectures

- Privacy
 - Motivations
 - Central Differential Privacy
 - Local Differential Privacy
 - Privacy vs Utility
- Fairness
 - **Motivations**
 - Some notions of fairness

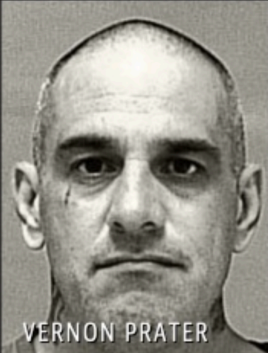
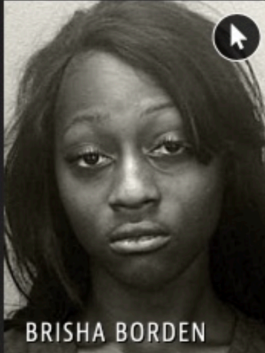
Motivation: risk of unfair decision in ML

The risk of unfair decisions is amplified by Machine Learning. Possible causes are:

- ML is based on correlation and not on causality (risk of bias-inducing correlations)
- Data used for training may be already biased.

Example: ML used to predict recidivity in the USA

Two Petty Theft Arrests

 VERNON PRATER	 BRISHA BORDEN
LOW RISK 3	HIGH RISK 8

Borden was rated high risk for future crime after she and a friend took a kid's bike and scooter that were sitting outside. She did not reoffend.

Two Petty Theft Arrests

 VERNON PRATER	 BRISHA BORDEN
Prior Offenses 2 armed robberies, 1 attempted armed robbery	Prior Offenses 4 juvenile misdemeanors
Subsequent Offenses 1 grand theft	Subsequent Offenses None
LOW RISK 3	HIGH RISK 8

Borden was rated high risk for future crime after she and a friend took a kid's bike and scooter that were sitting outside. She did not reoffend.

Non recidivating black people twice as likely to be labelled high risk than non recidivating white people

<https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>

Content of the lectures

- Privacy
 - Motivations
 - Central Differential Privacy
 - Local Differential Privacy
 - Privacy vs Utility
- Fairness
 - Motivations
 - Some notions of fairness

Notation and basic notions

Data model and predictor

- X Legitimate attributes
- A Sensitive attribute (binary)
- Y Decision (binary)
- \hat{Y} Prediction of the classifier (binary)

Example

Loan	X	employment, salary (income)
	A	race
	Y	loan decision
	\hat{Y}	prediction



Statistical parity SP

$$\mathbb{P}[\hat{Y} = \hat{y} \mid A = 0] = \mathbb{P}[\hat{Y} = \hat{y} \mid A = 1] \quad \hat{Y} \perp A$$

Statistical parity is usually too strong:

Example

In the example of the loan, if the income status is unbalanced between the races, in order to satisfy SP the predictor should grant loans also to some of the people with insufficient income



Some fairness notions

Conditional statistical parity CSP

$$\mathbb{P}[\hat{Y} = \hat{y} \mid X = x, A = 0] = \mathbb{P}[\hat{Y} = \hat{y} \mid X = x, A = 1] \quad \hat{Y} \perp A \mid X$$

Example

The predictor can grant loans less frequently to a certain race, as long as this disparity is justified by the legitimate attributes (insufficient income)



Some fairness notions

Previous notions usually have a negative impact on the accuracy of a classifier. In order to avoid this problem, Hardt, Price and Srebro [NIPS'16] introduced the following notion:

Equalized odds EOdds

$$\mathbb{P}[\hat{Y} = y \mid Y = y, A = 0] = \mathbb{P}[\hat{Y} = y \mid Y = y, A = 1] \quad \hat{Y} \perp A \mid Y$$

EOdds assumes implicitly that Y is unbiased. If the training data do not respect this assumption, we should correct them.

Example

The probability that the predictor takes the “right” decision does not depend on the race



Thanks for the
attention!

Questions ?