Foundations of Privacy

Catuscia Palamidessi

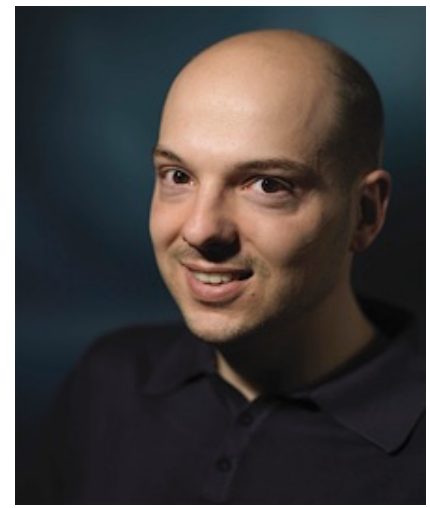
Content of the lectures

- Motivations, a bit of history, main problems
- Differential privacy
- Local differential privacy
- d-privacy and geo-indistinguishability
- Quantitative information flow
- Privacy issues in machine learning

Lectures

Catuscia Palamidessi
INRIA
LIX, Ecole Polytechnique

http://www.lix.polytechnique.fr/~catuscia/



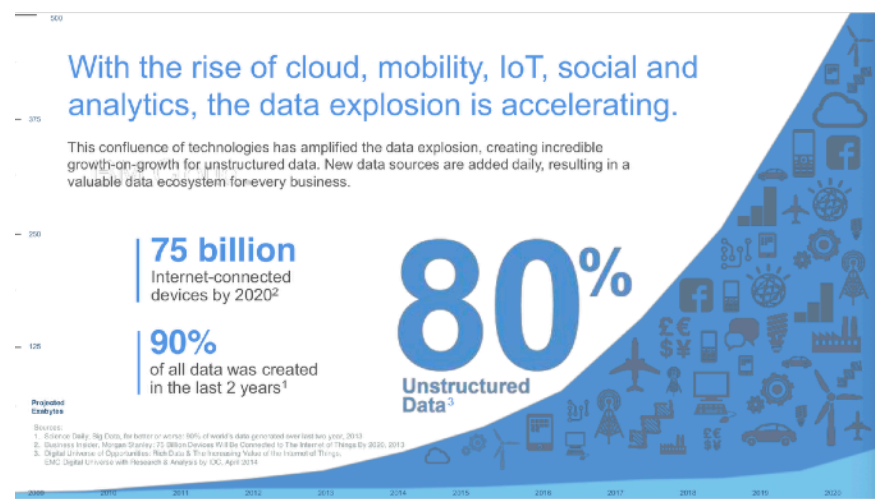
Pablo Piantanida
CentraleSupélec

http://webpages.lss.supelec.fr/perso/pablo.piantanida/Welcome.html

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 is the computing power to analyse them and extract all sort of sensitive information

Also, data are accumulated at an increasing speed. According to a research made by IMB in 2017, 90% of the world data had been generated in the last 2 years!



Collection and use of data

Individuals

Enterprises

Research

data

service

Data analyst

Service provider

Data collection

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

Issue I: Inference attacks

The problem of privacy is complicated because hiding the sensitive data is not sufficient: sensitive information can be derived using *correlated information* that is necessarily public or anyway available to the attacker (inference attacks).

Example: your FB page may give hints about your political opinions, religious beliefs, medical status, etc.

In addition the adversary can use some *side knowledge* (aka background knowledge)

Example: if you live in an area where the majority of the population votes for a certain party, it's more likely that you have the same political inclination.

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

Issue 2: Trade off with utility

The measure 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) and the precision of the analysis

Issue 2: Trade off with utility

Individuals

Enterprises

Research

Utility of analysis

data

service

QoS

Data analyst

Service provider

Data collection

Issue 3: Whom can we trust?

Issue 3: Whom can we trust?

Individuals

Enterprises

Research

data

service

Data analyst

Service provider

Data collection

Issue 3: Whom can we trust?

Individuals

Enterprises

Research

data

service

Data analyst

Service provider

Data collection

Issue 3: Whom can we trust?

Individuals

Enterprises

Research

data

service

Data analyst

Service provider

Data collection

Issue 3: Whom can we trust?

1. **Centralized model**: we trust the server / data curator.
   - The sanitization is done by the curator.
   - Utility is precision of analysis.
   - 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 is done at the user's side
   - Both kinds of utility 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 like that of Cambridge Analitica). Some big companies (e.g., Google and Apple) 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	amputed hand
3	Arya Stark	16	stomac 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**

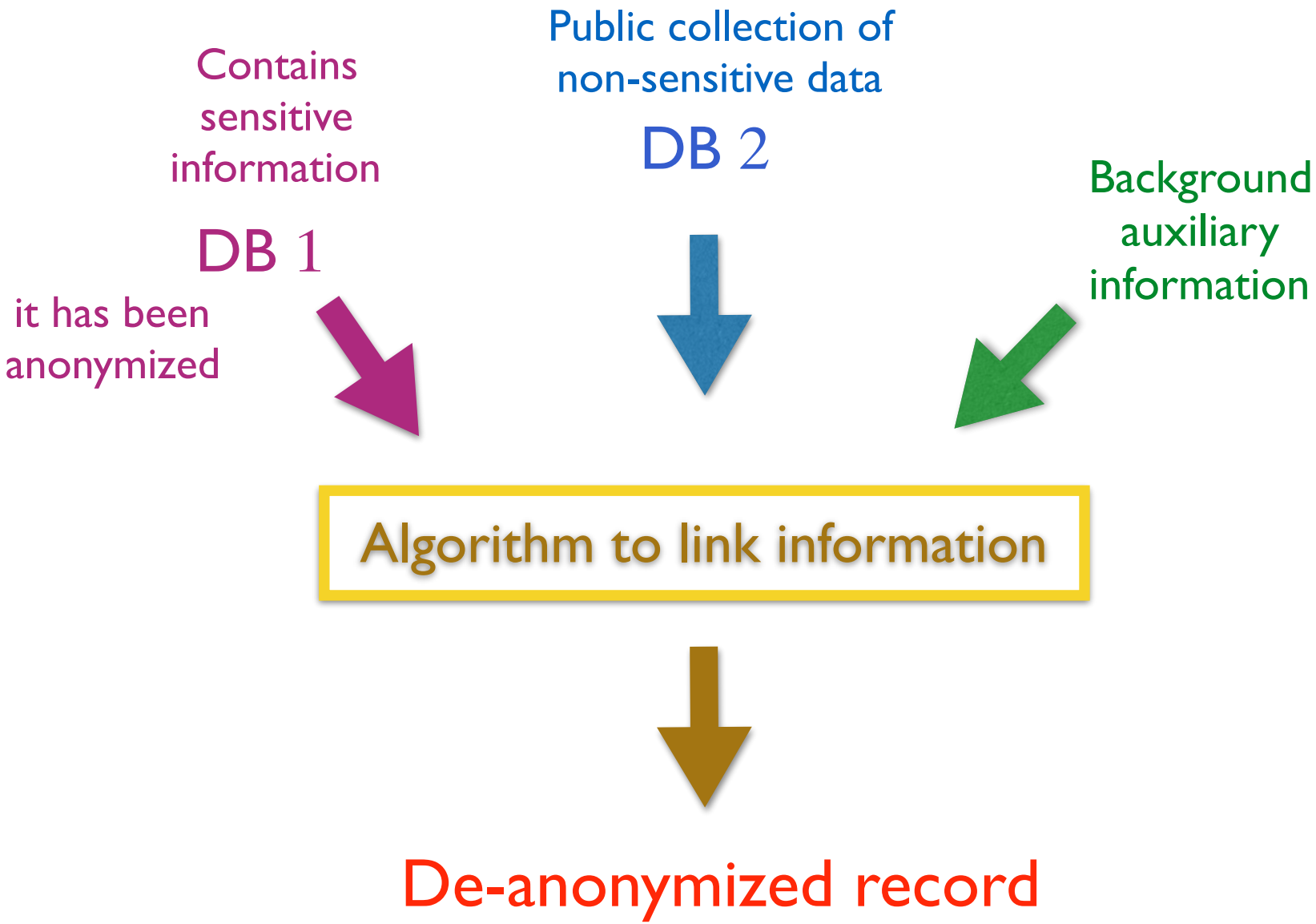
- Anonymization removes the column of the name, so that, for instance, the grayscale disease cannot be directly linked to Jorah Mormont

- Hystorically the first method, still used nowadays

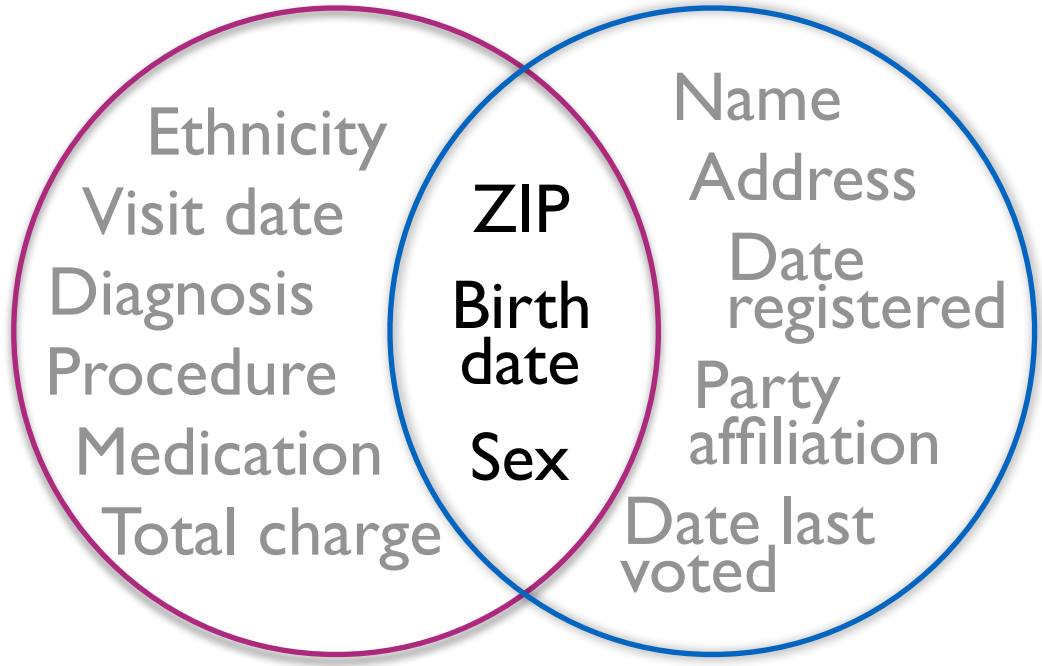
- However, this solution has been (already several years ago) shown to be ineffective, i.e., vulnerable to de-anonymization attacks

	Name	age	Disease
1	-	30	cold
2	-	39	amputed 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



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

- Obvious problem: in the sanitized dataset, all the individual in a group may the same value for the sensitive data, like in this table
- Clearly, the people in that group are not protected from the revelation of their disease

	Non-Sensitive				Sensitive
	Rase	Age	Sex	Zip Code	Disease
1	*	< 40	*	120**	Cancer
2	*	< 40	*	120**	Cancer
3	*	< 40	*	120**	Cancer
4	*	< 40	*	120**	Cancer
5	*	≥ 50	*	151**	Hemophilia
6	*	≥ 50	*	151**	Cancer
7	*	≥ 50	*	151**	Virus
8	*	≥ 50	*	151**	Virus
9	*	4*	*	120**	Hemophilia
10	*	4*	*	120**	Hemophilia
11	*	4*	*	120**	Virus
12	*	4*	*	120**	Virus

Table 2: 4-anonymous inpatient microdata.

ℓ-diversity [Kifer et al.]

- A solution to this problem was proposed under the name of ℓ-diversity.
- The idea is to form the groups in such a way that each group contains a variety of values for the sensitive data

	Non-Sensitive				Sensitive
	Rase	Age	Sex	Zip Code	Disease
1	*	≤ 50	*	120**	Cancer
2	*	≤ 50	*	120**	Cancer
9	*	≤ 50	*	120**	Hemophilia
11	*	≤ 50	*	120**	Virus
5	*	> 50	*	151**	Hemophilia
6	*	> 50	*	151**	Cancer
7	*	> 50	*	151**	Virus
8	*	> 50	*	151**	Virus
3	*	≤ 50	*	120**	Cancer
4	*	≤ 50	*	120**	Cancer
10	*	≤ 50	*	120**	Hemophilia
12	*	≤ 50	*	120**	Virus

Table 5: 3-diverse table

t-closeness

- Also the ℓ-diversity has problems, though:
  - the requirement of ℓ-diversity may be too strict (for instance, certain values of the disease, like having a cold, may not need to be protected)
  - the requirement of ℓ-diversity may not be enough. For instance, if **almost all individuals** in a certain group have cancer, the attacker will infer that information (for a given individual in the group) with high probability
- To amend these problems, the t**-closeness** requirement was proposed: the idea is that the grouping is done in such a way that the distribution for the sensitive values in each group is close to the general distribution

Complexity of ℓ-diversity and t-closedness

The sanitization of the database with ℓ-diversity or t-closedness is NP-hard

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

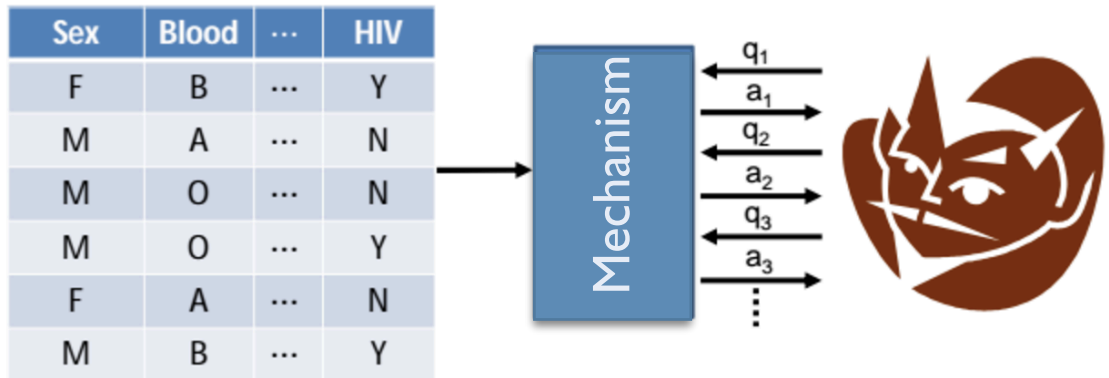
By using only the network topology, they were able to show that 33% of the users who had accounts on both **Twitter** and **Flickr** could be re-identified in the anonymous Twitter graph with only a 12% error rate.

Scenario 1.2:

Centralized model

Micro data not accessible, we can only query the DB

Protection of datasets via an interface



- One can only retrieve aggregated information, not personal records
  - *"What is the average weight of people affected by the disease ?"*
  - *"Does Don have the disease ?"*

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 minimal 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 minimal age of a person with the disease

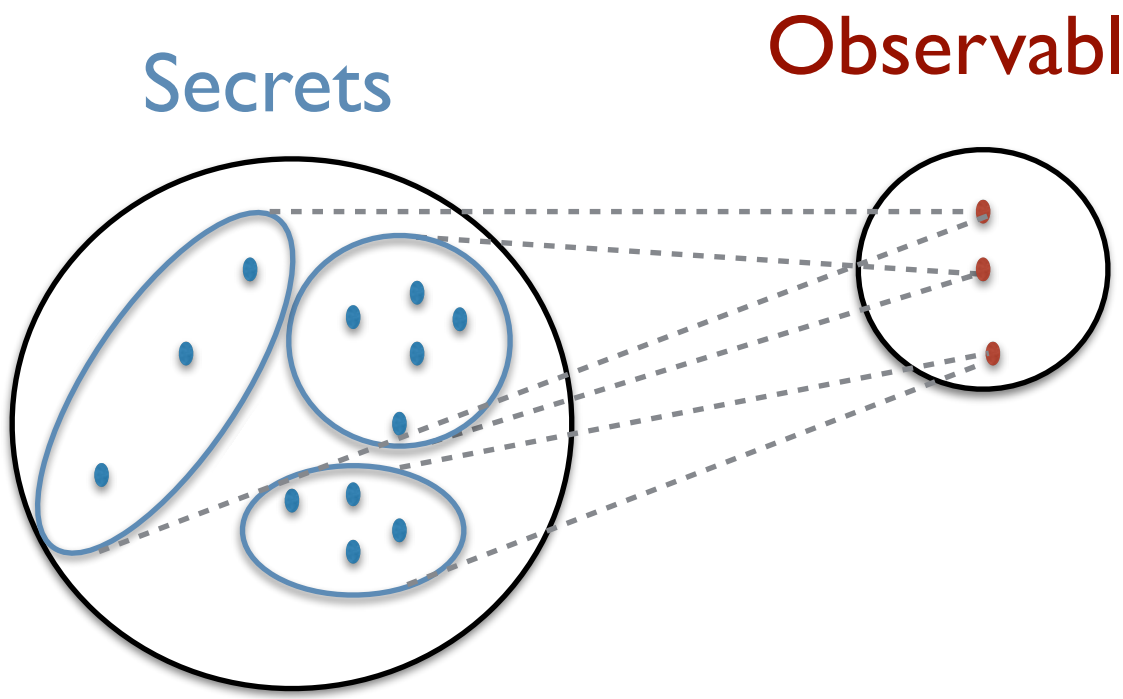
Answers: 40, 100. **Unique!**

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

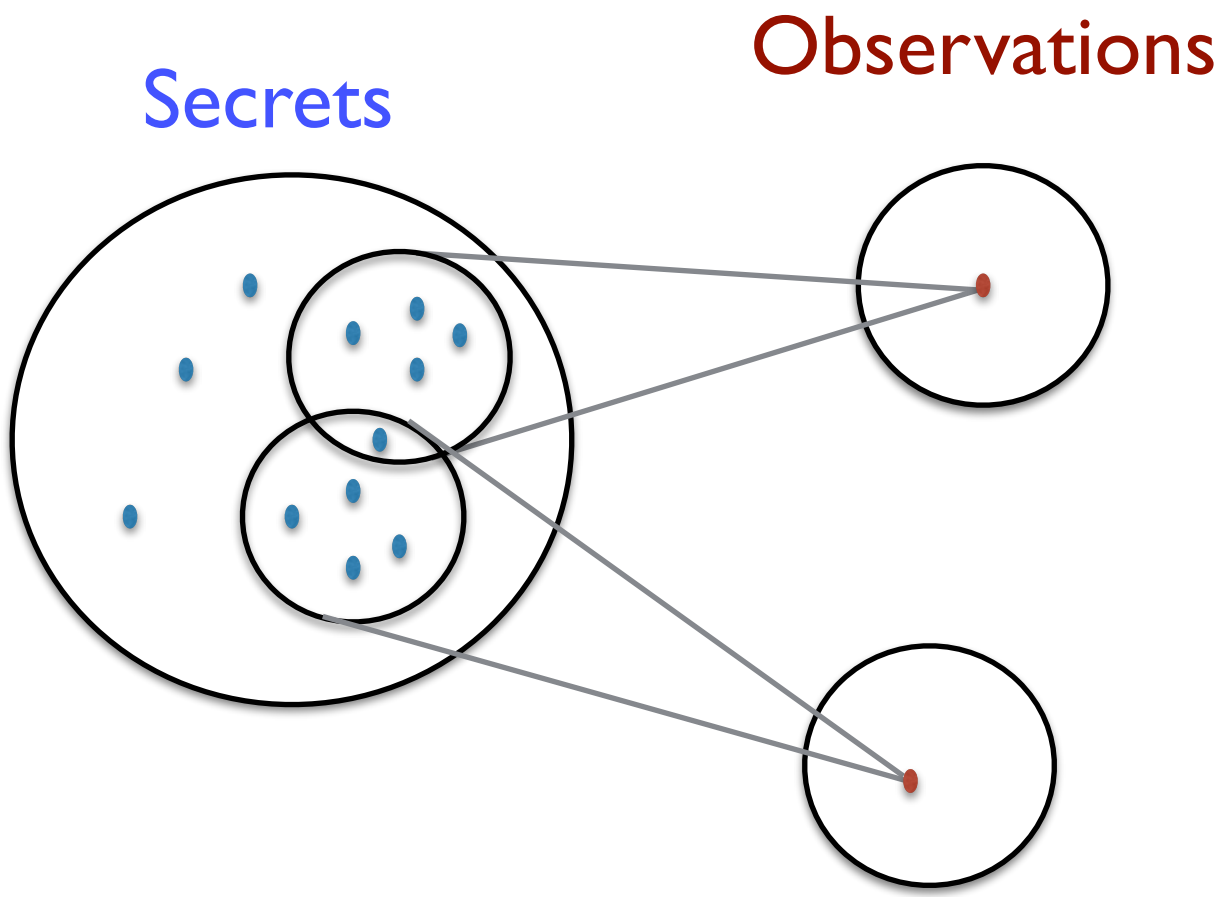
name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

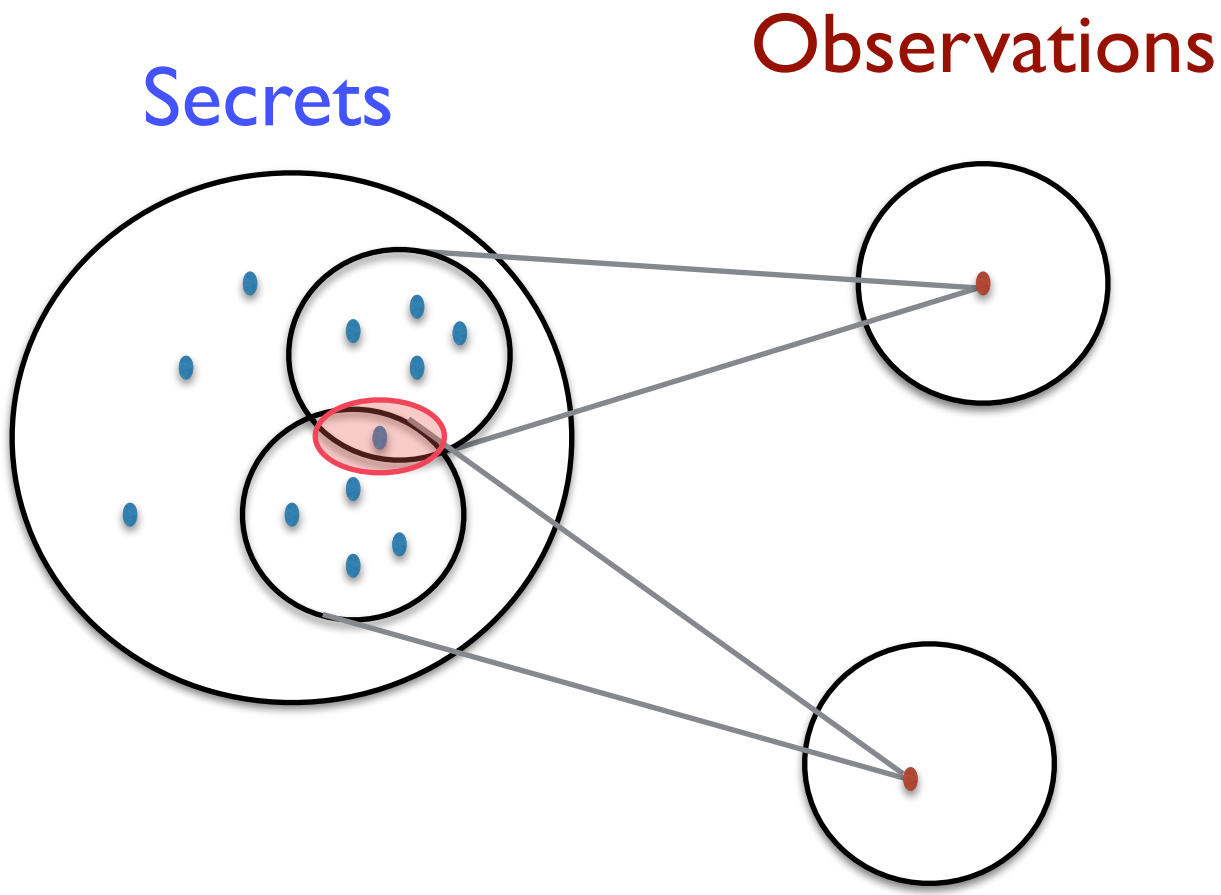
Composition attacks are a general problem of **Deterministic approaches :** They are all based on the principle that one observation corresponds to many possible values of the secret (group anonymity)



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



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



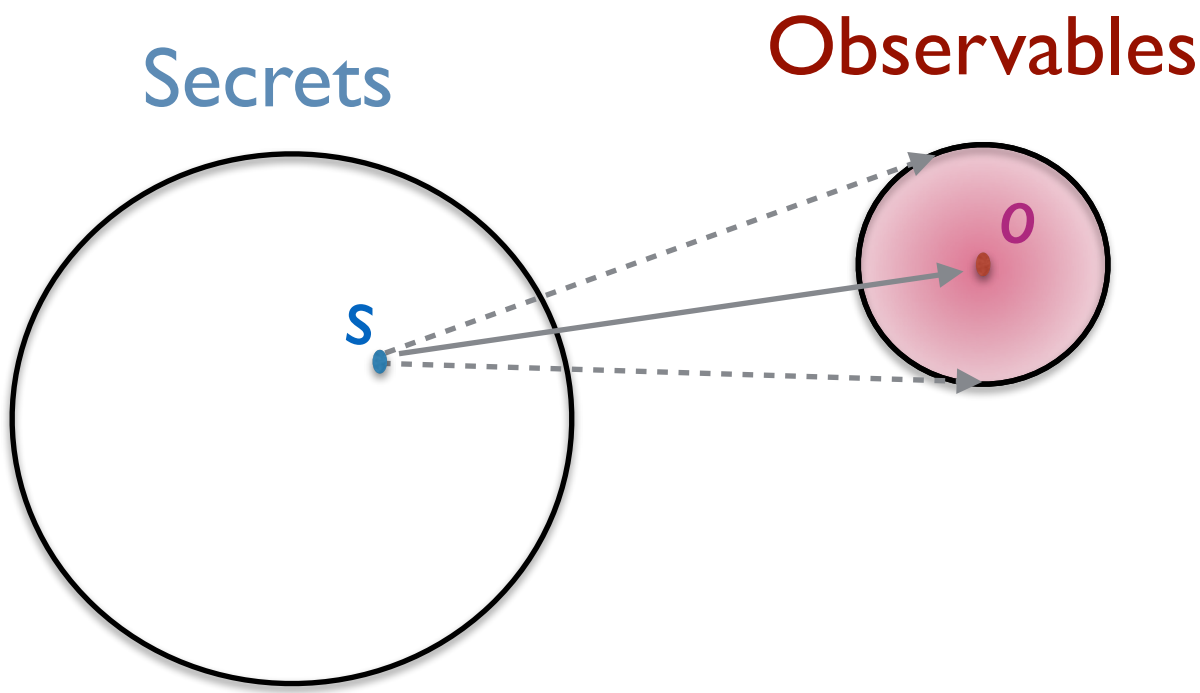
Too bad!!! What can we do?

Use probabilistic approaches!

Most of the state-of-the-art techniques are indeed based on randomization

Probabilistic approaches

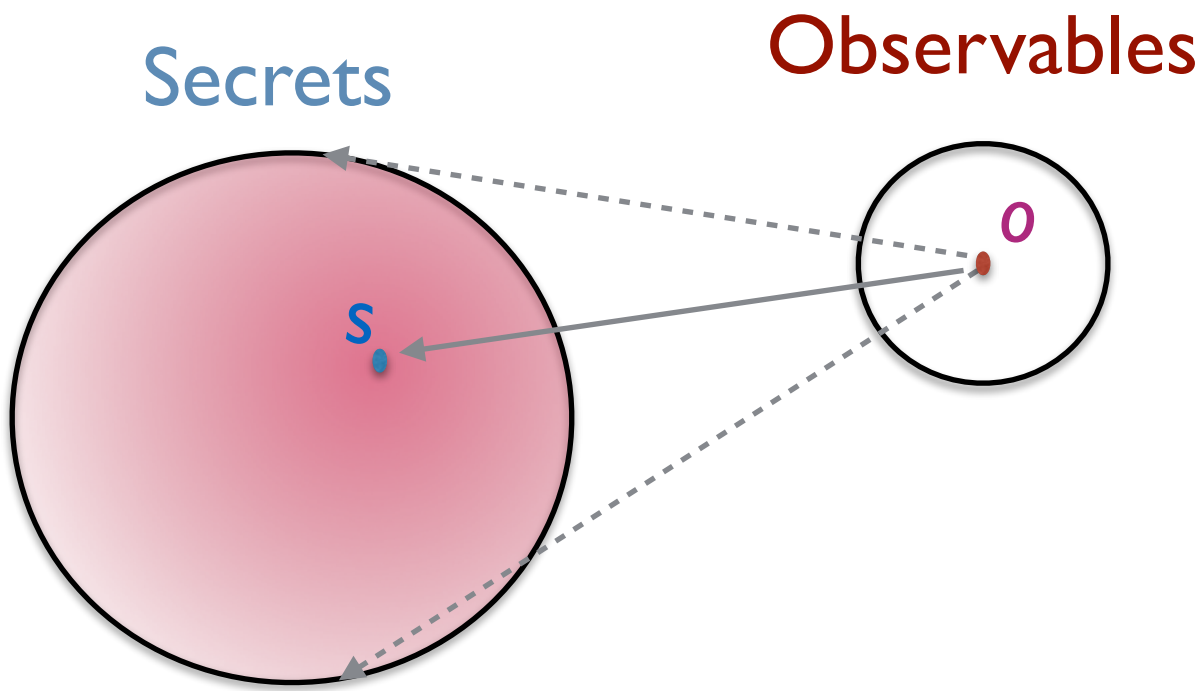
Every secret can generate any observable, according to a certain probability distribution.



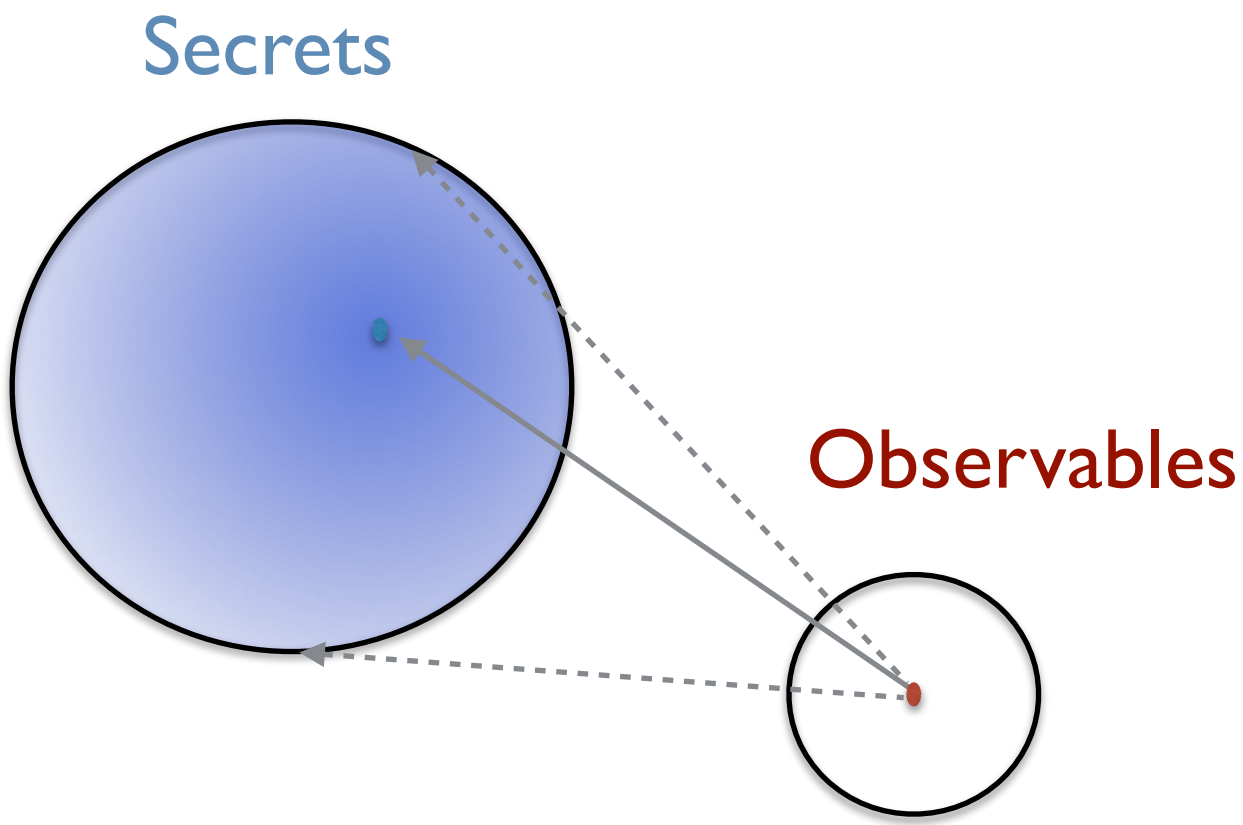
Probabilistic approaches

By the Bayes law

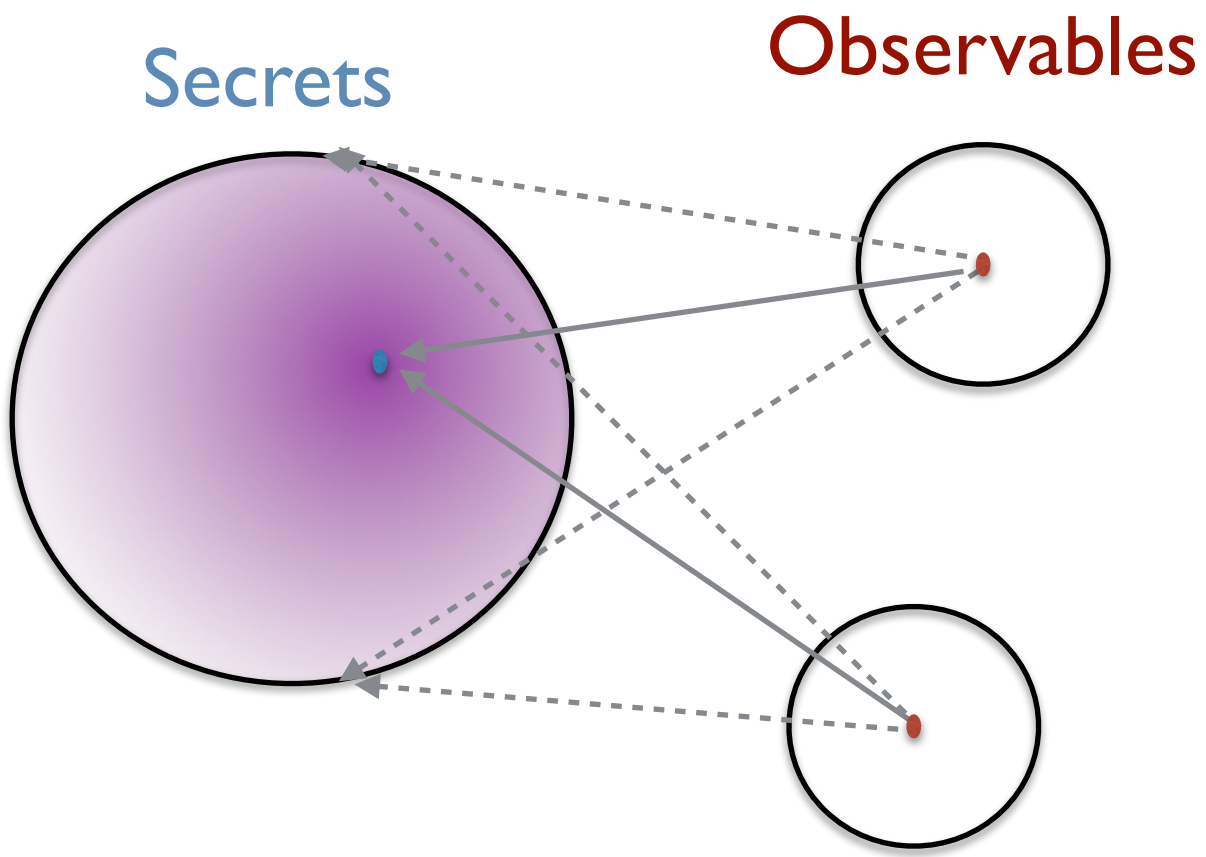
$$p(s|o) \propto p(o|s)$$



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

minimal 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

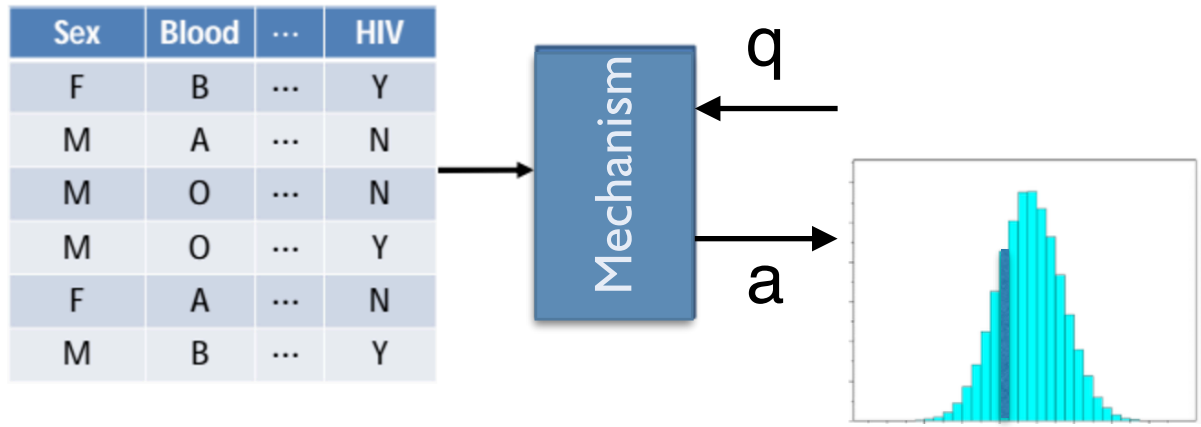
Even if he combines the answers, the adversary cannot tell for sure whether a certain person has the disease

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

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]

Differential Privacy

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 $\mathcal{X} = \bigcup_{n\geq 1}^{<\infty} \mathcal{V}^n$. We will represent the elements of $\mathcal{X}$ by $x, x_1, x_2, \ldots$
- We will assume a probability distribution on distribution on $\mathcal{V}$ and $\mathcal{X}$ and indicate by V, X the respective random variables

Examples:

$\mathcal{V}$ = integers

x

20
14
51
75

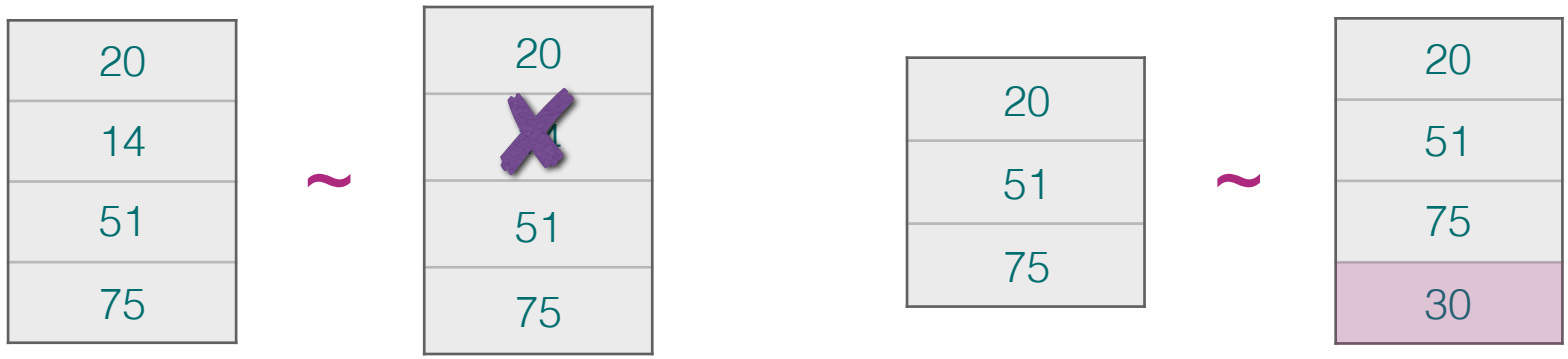
$\mathcal{V}$ = names x integers

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,

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

- $y = f(x)$ 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

x
20
14
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,

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

  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

We are now ready to define differential privacy. We first consider the discrete case, i.e., when the reported answer is discrete

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 $z \in \mathcal{Z}$ we have

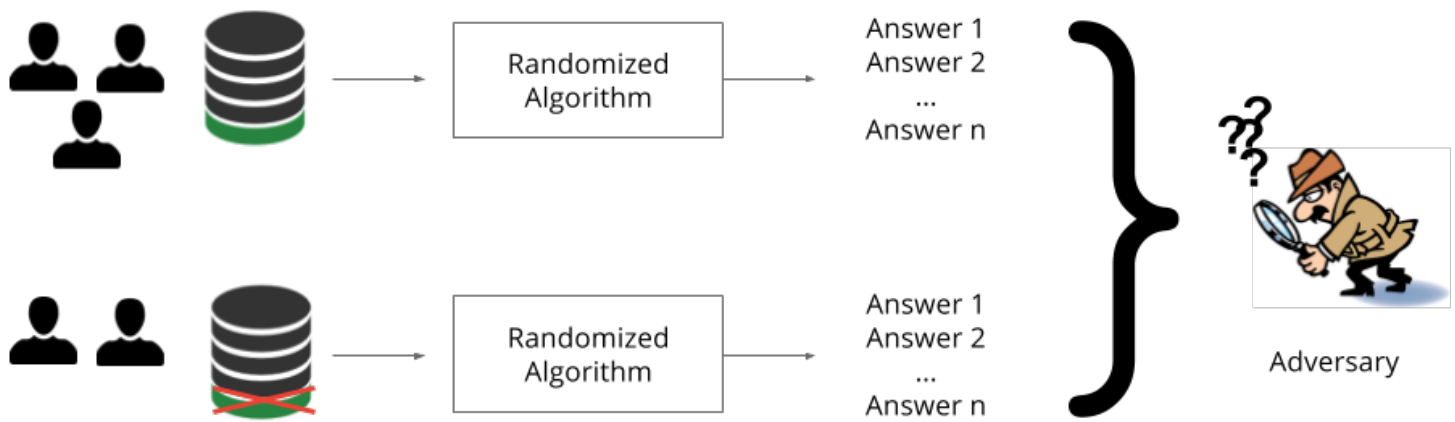
$$p(\mathcal{K}(x_1) = z) \leq e^{\varepsilon}\, p(\mathcal{K}(x_2) = z)$$

where $p(\mathcal{K}(x) = z)$ represents the probability that $\mathcal{K}$ applied to x reports the answer z

Note: $p(\mathcal{K}(x) = z)$ represents a conditional probability. We will write it as $p(Z = z|X = x)$ 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

Is DP what we want?
Is DP the best we can do?

What we really would like is that by querying the DB the adversary cannot derive much information about the individual

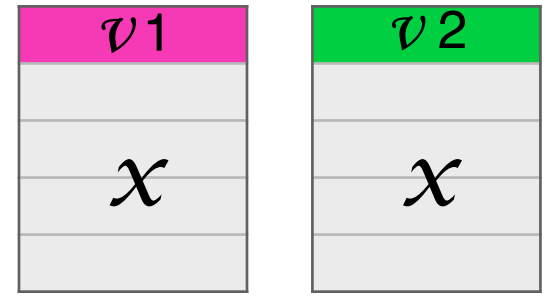
Unfortunately, this is not possible

Example: Assume that the adversary knows that Turing has the same height of the average height of the rest of the people in the DB. Then, by querying the DB, the adversary gains a lot of information about the heigh of Turing (if we want to preserve some utility)

Note that this happens whether or not John is in the DB

Other interpretations of DP

Consider two databases that differ only for the value of one individual record

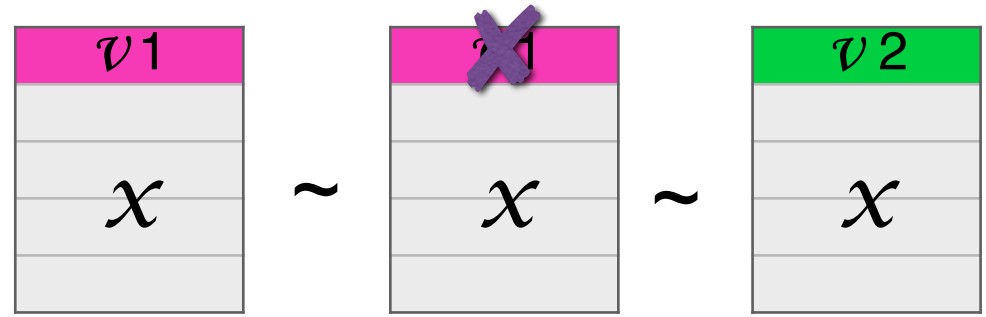


Theorem If $\mathcal{K}$ is ε-differentially-private then $\forall v_1, v_2 \in \mathcal{V}, \forall x \in \mathcal{X}, \forall z \in \mathcal{Z}$

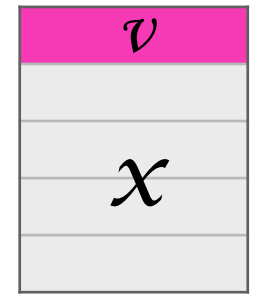
$$p(\mathcal{K}(x \cup v_1) = z) \leq e^{2\varepsilon}\, p(\mathcal{K}(x \cup v_2) = z)$$

Note that also the reverse is true: it is sufficient to enrich $\mathcal{V}$ with an extra value $\perp$ that represents the absence of an individual

The proof of the theorem is immediate, just observe the relation with x, and then apply transitivity of $\leq$



Other interpretations of DP: Bayesian



Consider a database consisting of $\mathcal{X}$ plus a record $\mathcal{V}$

Theorem $\mathcal{K}$ is ε-differentially-private iff $\forall v \in \mathcal{V}, \forall x \in \mathcal{X}, \forall z \in \mathcal{Z}$

$$p(V = v|X = x, Z = z) \leq e^{\varepsilon}\, p(V = v|X = x)$$

$$p(V = v|X = x) \leq e^{\varepsilon}\, p(V = v|X = x, Z = z)$$

Proof: exercise.

This means that, if the adversary knows the value of all the other records of the database, then knowing the reported answer z does not improve much his knowledge of a given individual V

The assumption that the adversary knows the value of all the other records of the database is called **strong adversary model**

Question 1 Is the hypothesis of the strong adversary necessary for the result?

Question 2 How does this result reconcile with the example of the height of Turing ?

Examples of mechanisms

Let us assume that we have databases containing as values $\mathcal{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\varepsilon = 0$.

Examples of mechanisms

Let us assume that we have databases containing as values $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\varepsilon = 0$

  However, it's totally useless !

Examples of mechanisms

Let us assume that we have databases containing as values $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{100}$ if the true answer is less than $\mathbf{150}$, and $\mathbf{200}$ otherwise. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\mathbf{100}$ if the true answer is less than $\mathbf{150}$, and $\mathbf{200}$ 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 $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\frac{e^{\varepsilon}}{200+e^{\varepsilon}}$, and every other integer in $[50, 250]$ with probability $\frac{1}{200+e^{\varepsilon}}$. Is it differentially private ?

Examples of mechanisms

Let us assume that we have databases containing as values $\mathbf{V}$ the heights of people, in cm, ranging from $\mathbf{50}$ to $\mathbf{250}$ (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 $\frac{e^\varepsilon}{200+e^\varepsilon}$, and every other integer in $[50, 250]$ with probability $\frac{1}{200+e^\varepsilon}$. Is it differentially private ?

Yes. It's ε-DP

It is also relatively useful. We will study its utility 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.

Internship and PhD

If you find these topics interesting, you may consider doing an internship (stage) in our teams. There are various internship projects available, and we will describe them during the course

Possibility to continue as a PhD. We have various grants that will provide financial support. In particular:

HYPATIA:

- Statistical utility from noisy data
- Optimal privacy-utility trade-off
- Generation of optimal mechanism via ML

- Analysis of privacy threats in ML

Thanks for the attention

Questions?