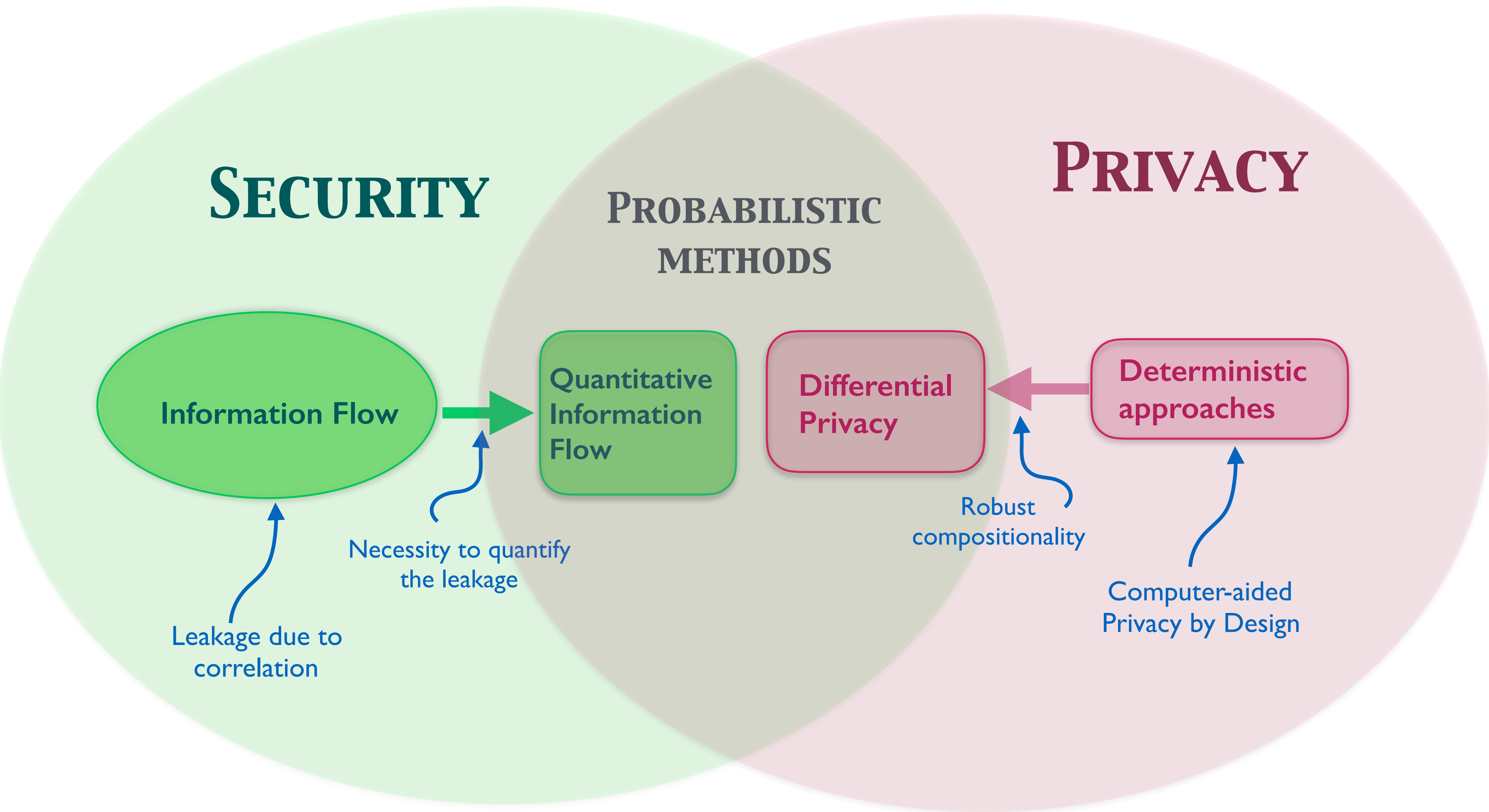
Foundations of Privacy

Lecture 6

Relation between the main topics of this course



Plan of the lecture

- A brief panoramic of the main deterministic approaches to privacy
- Differential Privacy (DP)
- The Bayesian interpretation of DP
- Compositionality and independence from prior
- The privacy budget
- Implementation of DP: Laplacian noise
- Examples and exercises

Plan of the lecture

- A brief panoramic of the main deterministic approaches to privacy
- Differential Privacy (DP)
- The Bayesian interpretation of DP
- Compositionality and independence from prior
- The privacy budget
- Implementation of DP: Laplacian noise
- Examples and exercises

The problem

- In general, the problem of privacy is to protect the disclosure of **sensitive information** of individuals when a collection of data about these individuals (*dataset*) is made **publicly available**
- The process of transforming the dataset in order to avoid such disclosure is called **sanitization**

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

- For instance, Jorah Mormont may not want to reveal that he is affected by greyscale, because he may be

	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 very weak and prone 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

Sweeney's de-anonymization attack by linking [around year 2000]

Contains sensitive information

DB 1

it has been anonymized

Public collection of non-sensitive data

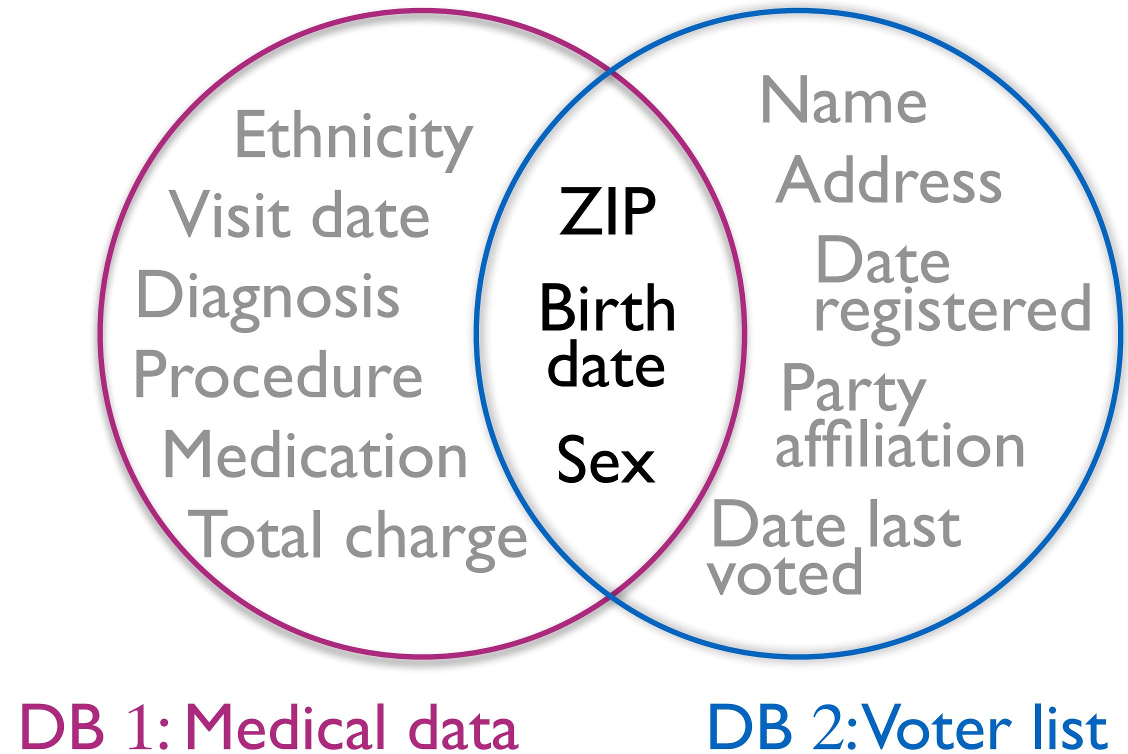
DB 2

Background auxiliary information

Algorithm to link information

De-anonymized record

Sweeney's de-anonymization attack by linking [around year 2000]



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

- 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

- 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 in each group is close to the general distribution

Problems with previous methods

- High-dimensional and sparse databases.
  - Example: Netflix movies preferences.
  - The quasi-identifiers contain too many columns
- Composition attacks (I will come back to these later)
- These problems (and others) have lead to the development of Differential Privacy

Differential Privacy

- Problem of statistical databases: we want to make available aggregate information, but without compromising the private data of the individual participating in the database

- This is not so easy to do. Naive deterministic methods, such as k-anonymity, are vulnerable to combination 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

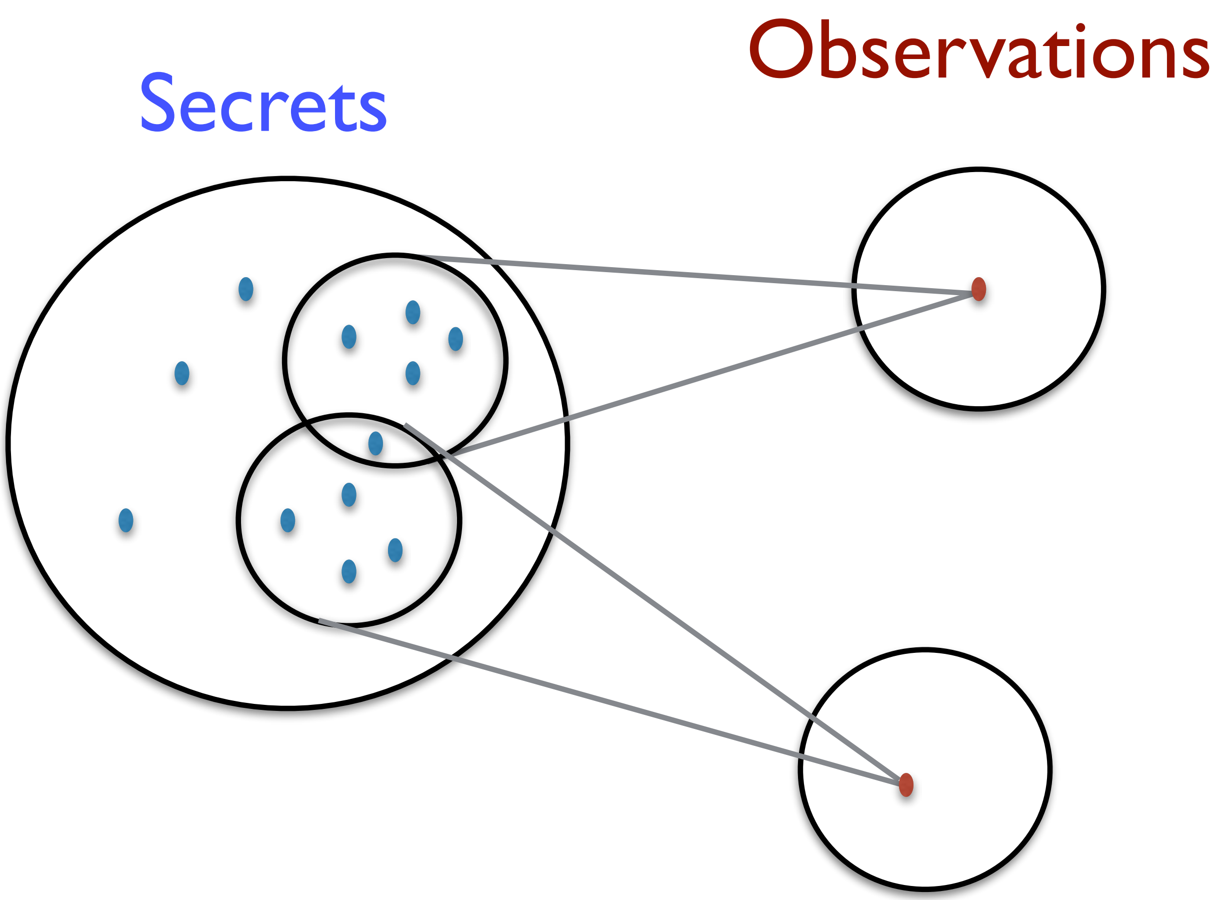
Answers: 40, 100

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

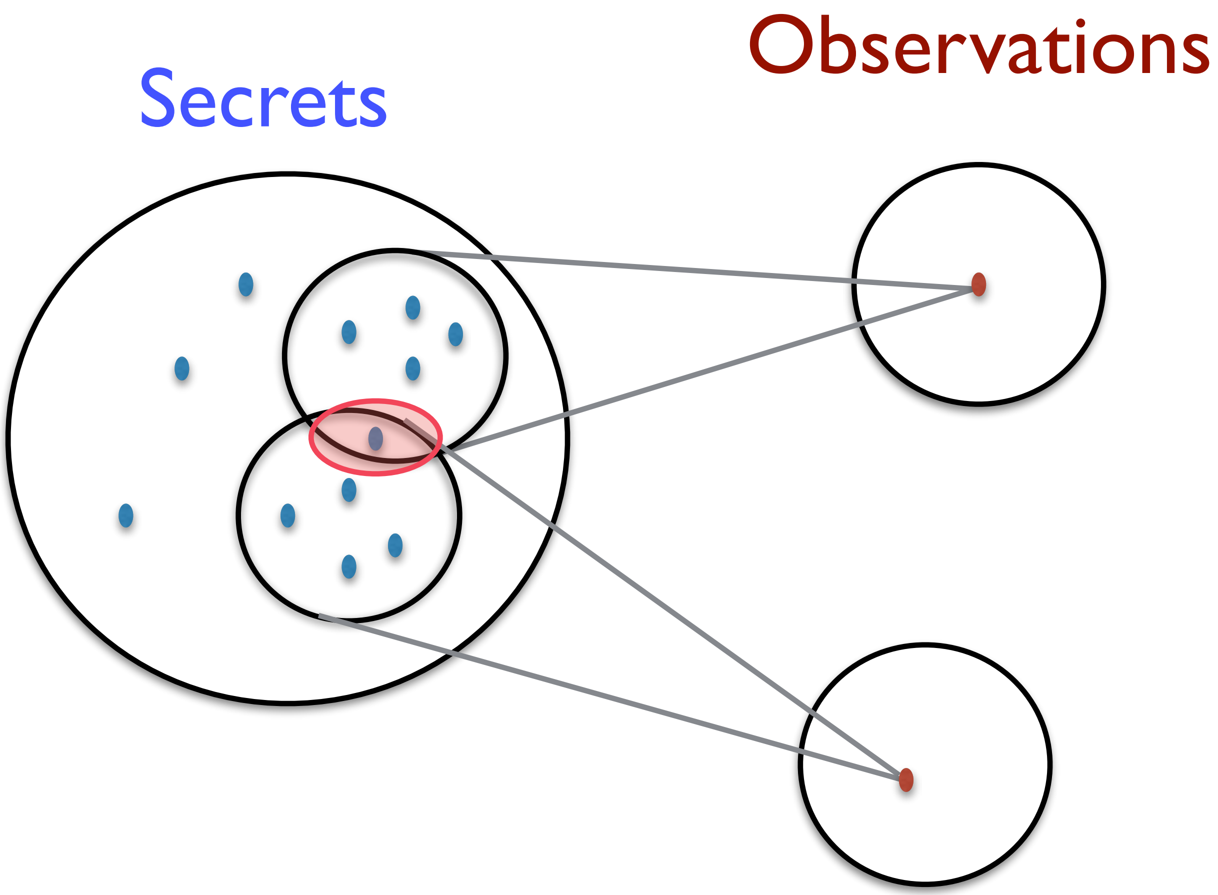
name	age	disease
Alice	30	no
Bob	30	no
Carl	40	no
Don	40	yes
Ellie	50	no
Frank	50	yes

Alice	Bob
Carl	Don
Ellie	Frank

This is a general problem of the deterministic approaches (based on the principle of many-to-one): the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletones



This is a general problem of the deterministic approaches (based on the principle of many-to-one): the combination of observations determines smaller and smaller intersections on the domain of the secrets, and eventually result in singletones



A better solution

Introduce some probabilistic noise on the answer, so that the answers of minimal age and minimal weight can be given also by other people with different age and weight

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

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

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

Randomized mechanisms

- A randomized mechanism (for a certain query) reports an answer which is an approximation of the true answer and is generated randomly according to some probability distribution
- Randomized mechanisms are more robust to combination attacks than the deterministic ones
- However, we need to choose carefully the probability distribution, in order to get the desired degree of privacy, and in order to maintain a certain degree of utility for the query
- There is a trade-off between utility and privacy, but it is not strict: for a certain degree of privacy, one mechanism can give a better utility than another. It is therefore interesting to try to find the optimal mechanism (the mechanism with highest utility), among those that offer the desired degree of privacy.
- To solve the above problem, and more in general to reason about privacy and utility, we need formal, rigorous definitions of these notions.
- A definition of privacy that has become very popular: Differential Privacy [Cynthia Dwork, ICALP 2006]

Databases

- V is a set whose elements represent all possible **values of the records** ($v \in V$ can be a tuple, i.e. it can be composed by various fields). We assume that V contains a special element $\perp$ representing a dummy record, or the absence of the corresponding record.

- A **database** of n records is an element of V^n. We will represent the databases by x, x_1, x_2, ...

- We assume a probability distribution π on the databases. We will indicate by X the corresponding random variable.

- 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, or has changed value.

- The number of records in which two databases x_1, x_2 differ from each other is called "Hamming distance" between x_1, x_2.

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 $\mathcal{Y}$.

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,

  $$\mathcal{K} : \mathcal{X} \to \mathcal{DZ}$$

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

- $\mathcal{Z}$ does not necessarily coincide with $\mathcal{Y}$.

- z drawn from $\mathcal{D}(x)$ is a **reported answer** of the query $\mathcal{K}$ on the database 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** for a randomized mechanism $\mathcal{K}$.

- Let us first consider the discrete case. Namely, $\mathcal{K}(x)$ is discrete, for every database x.

- **Definition (Differential Privacy)** $\mathcal{K}$ is ε-differentially private if for every pair of databases $x_1, x_2 \in \mathcal{X}$ such that $x_1 \sim x_2$, and for every $z \in Z$, we have:

$$p(Z = z|X = x_1) \leq e^{\varepsilon} p(Z = z|X = x_2)$$

  where $p(Z = z|X = x)$ represents the conditional probability of z given x, namely the probability that on the database x the mechanism reports the answer z

- This definition therefore means that the value (or the presence) of an individual does not affect significantly the probability of getting a certain reported value.

Bayesian interpretation

- Let X_i be the random variable representing the value of the individual i, and let X_{others} be the random variable representing the value of all the other individuals in the database.
  Similarly, let x_i and x_{others} represent possible values for X_i and X_{others}.
  Note that (x_i, x_{others}) represents and element in $\mathcal{X}$.
  Analogously, let π_i represent the component of the prior distribution that concerns the value of the individual i.

- ε-differential privacy is equivalently characterized by the following property (we consider the discrete case, the continuous case is analogous): For all $(x_i, x_{others}) \in \mathcal{X}$, for all $z \in Z$, and for all π_i,

$$e^{-\varepsilon} \leq \frac{p(X_i = x_i | X_{others} = x_{others}, Z = z)}{p(X_i = x_i | X_{others} = x_{others})} \leq e^{\varepsilon}$$

  Namely: assuming that the adversary knows the value of all the other individuals in the database, the reported answer does not increase significantly his probabilistic knowledge of the value of i, with respect to his prior knowledge

  Note: $p(X_i = x_i | X_{others} = x_{others})$ is called *prior* of x_i, and $p(X_i = x_i | X_{others} = x_{others}, Z = z)$ is called *posterior* of x_i.

Differential Privacy

- Let us now consider the continuous case. Namely, $\mathcal{K}(x)$ is a probability density function on $\mathcal{Z}$. The only thing that changes is that we consider a measurable subset $\mathcal{S}$ of $\mathcal{Z}$ instead than a single z:

- **Definition (Differential Privacy)** $\mathcal{K}$ is ε-differentially private if for every pair of databases $x_1, x_2 \in \mathcal{X}$ such that $x_1 \sim x_2$, and for every measurable $\mathcal{S} \subseteq Z$, we have:

$$p(Z \in \mathcal{S} | X = x_1) \leq e^{\varepsilon} p(Z \in \mathcal{S} | X = x_2)$$

  where $p(Z \in \mathcal{S} | X = x)$ represents the probability that on the database x the mechanism reports an answer in $\mathcal{S}$

- This definition therefore means that the value (or the presence) of an individual does not affect significantly the probability that the reported value satisfy a certain property.

Independence from the prior

- The distribution π on the databases is called prior, meaning: *before* the reported answer
- π represents the knowledge that a potential adversary (aka user, in the case of DP) 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**: An user is given an initial budget α. Each time he asks a query, answered by ε-differentially private mechanism, his budget is decreased by ε. When his budget is exhausted, he is not allowed to ask queries anymore.

Bayesian interpretation

- Let X_i be the random variable representing the value of the individual i, and let X_{others} be the random variable representing the value of all the other individuals in the database.
  Similarly, let x_i and x_{others} represent possible values for X_i and X_{others}.
  Note that (x_i, x_{others}) represents and element in $\mathcal{X}$.
  Analogously, let π_i represent the component of the prior distribution that concerns the value of the individual i.

- ε-differential privacy in the discrete case is equivalently characterized by the following property: For all $(x_i, x_{others}) \in \mathcal{X}$, for all $z \in Z$, and for all π_i,

$$p(X_i = x_i | X_{others} = x_{others}, Z = z) \leq e^{\varepsilon} p(X_i = x_i | X_{others} = x_{others})$$

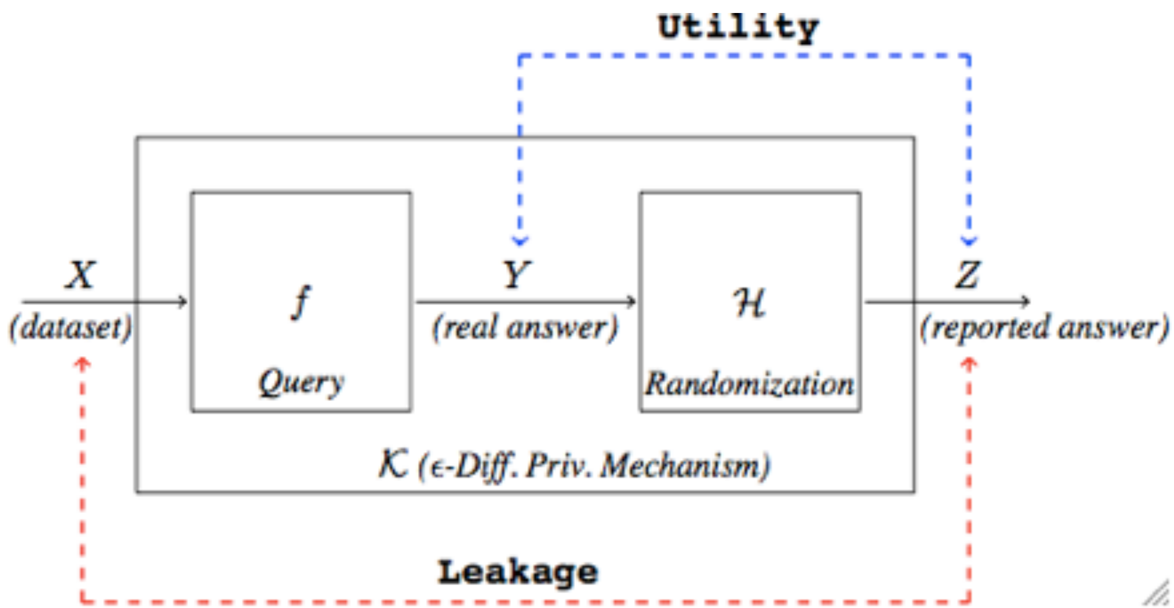
  Namely: assuming that the adversary knows the value of all the other individuals in the database, the reported answer does not increase significantly his probabilistic knowledge of the value of i, with respect to his prior knowledge

  Note: $p(X_i = x_i | X_{others} = x_{others})$ is called *prior* of x_i, and $p(X_i = x_i | X_{others} = x_{others}, Z = z)$ is called *posterior* of x_i.

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 $\mathcal{K} = f \times \mathcal{H}$



- 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 differentially private mechanism: Laplacian noise

- Randomized mechanism for a query $f : \mathcal{X} \to \mathcal{Y}$.
- A typical randomized method: **add Laplacian noise.** If the exact answer is y, the reported answer is z, with a probability density function defined as:

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

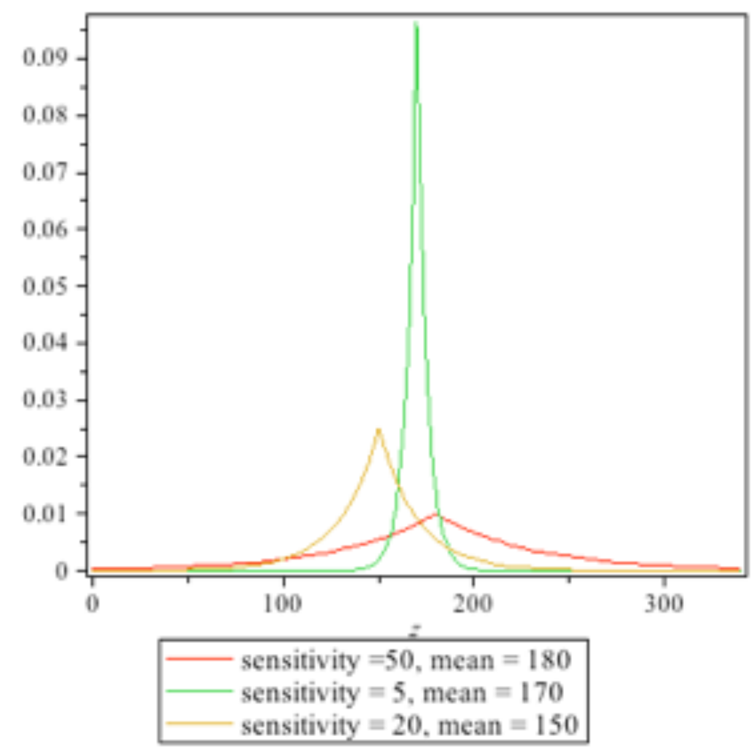
where Δf is the *sensitivity* of f:

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

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

and c is a normalization factor:

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



Laplacian mechanism

The probability density function of a Laplacian mechanism is:

$$p(Z=z|X=x) = dP_{f(x)}(z) = c\,e^{-\frac{|z-f(x)|}{\Delta f}\varepsilon}$$

where $c = \dfrac{\varepsilon}{2\,\Delta f}$

Theorem: The Laplacian mechanism is ε–differentially private

Proof: Let $x_1 \sim x_2$ and $y_1 = f(x_1), y_2 = f(x_2)$ We have:

$$\begin{aligned}
\frac{p(Z=z|X=x_1)}{p(Z=z|X=x_2)} &= \frac{c\,e^{-\frac{|z-f(x_1)|}{\Delta f}\varepsilon}}{c\,e^{-\frac{|z-f(x_2)|}{\Delta f}\varepsilon}} \\
&= e^{\frac{|z-y_2|}{\Delta f}\varepsilon - \frac{|z-y_1|}{\Delta f}\varepsilon} \\
&\leq e^{\frac{|y_1-y_2|}{\Delta f}\varepsilon} \\
&\leq e^{\varepsilon}
\end{aligned}$$

Exercise

- Show that the Bayesian interpretation of differential privacy, explained at Page 31, is indeed equivalent to the original formulation of differential privacy