Foundations of Privacy

Class 1

Plan of the lectures

- Motivations, a bit of history, main problems, research directions (3 hours)
- Quantitative Information Flow (9 hours)
- Differential Privacy and Extensions (9 hours)
- Location Privacy (3 hours)

Motivations

In the "Information Society", each individual constantly leaves digital traces of his actions that may allow to infer a lot of information about himself

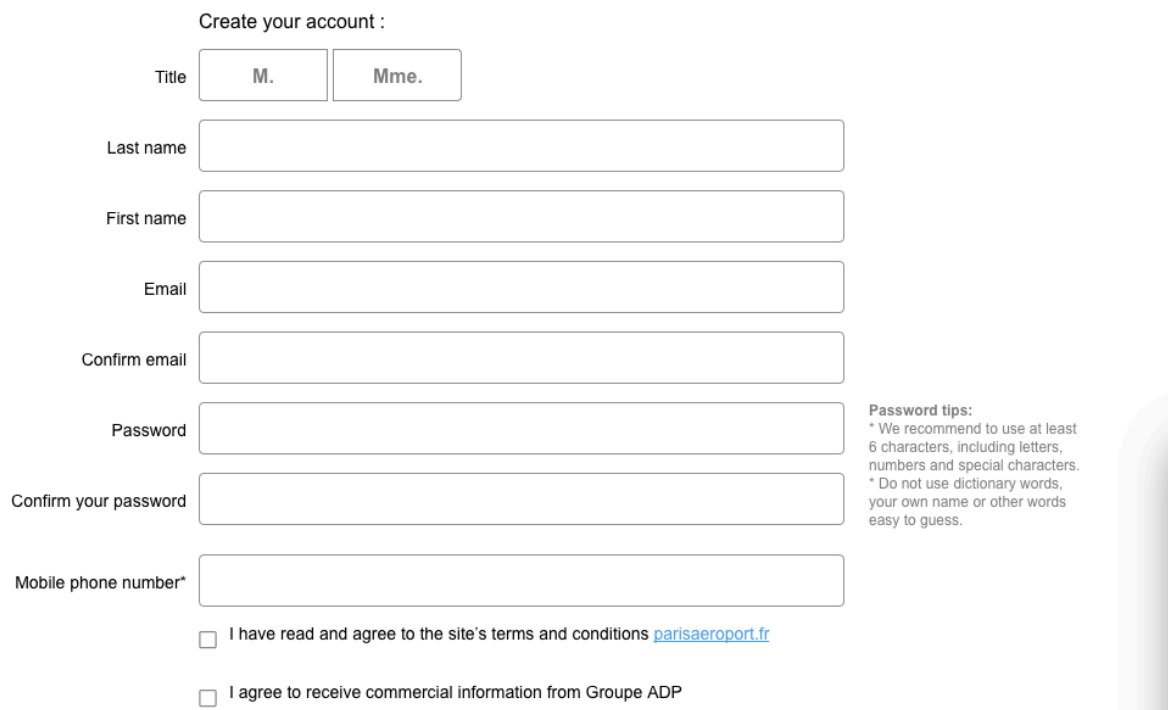
Request to a LBS ⇒ location.

History of requests ⇒ interests.

Activity in social networks ⇒ political opinions, religion, hobbies, . . .

Power consumption (smart meters) ⇒ activities at home.

Example:
Personal information in exchange of a service



We don't know how our information will be used

Concerns about privacy

Risk: collect and use of digital traces for fraudulent purposes.

Examples: targeted spam, identity theft, profiling, discrimination, …

The news are full of problems caused by privacy breaches

The need for privacy is intrinsic to the human nature, although it varies a lot from individual to individual, between cultures, and it evolves with time

Privacy is recognized as one of the fundamental right of individuals:

- Universal Declaration of the Human Rights at the assembly of the United Nations (Article 12), 1948.
- European Directive 95/46/EC on the Protection of Personal Data (currently being revised towards a stricter regulation).
- Japanese Act on the Protection of Personal Information from 2003 (current discussions to amend it and make stricter).

The new European regulation (will be enforced starting from 2018)

What will be the key changes?

- A **'right to be forgotten'** will help you manage data protection risks online. When you no longer want your data to be processed and there are no legitimate grounds for retaining it, the data will be deleted. The rules are about empowering individuals, not about erasing past events, re-writing history or restricting the freedom of the press.
- **Easier access to your own personal data.**
- **A right to transfer personal data** from one service provider to another.
- When your **consent is required, you must be asked to give it by means of a clear affirmative action.**
- More transparency about how your data is handled, with **easy-to-understand information**, especially for **children**.
- Businesses and organisations will need to **inform you about data breaches** that could adversely affect you **without undue delay**. They will also have to notify the relevant data protection supervisory authority.
- Better enforcement of data protection rights through improved **administrative and judicial remedies** in cases of violations
- Increased **responsibility and accountability** for those processing personal data – through **data protection risk assessments, data protection officers**, and the principles of **'data protection by design'** and **'data protection by default'**.

Different types of sensitive data

- Sensitive information about an individual :
  - credit card / bank information, home access code, passwords, …
    - sensitive because it can bring to attacks to the person or his properties
  - ethnicity, religious beliefs, political opinions, medical status, intimate videos ..
    - Sensitive because it can lead to discrimination.
- Identification information : information that can uniquely identify an individual.
  - First and last name, social security number, physical and email address, phone number, biometric data (such as fingerprint and DNA), …
    - Sensitive because it can be used to cross-reference databases, or to identify him as the subject of certain actions
- Sensitive information for organizations
  - Industries: production plans, research, strategies,…
  - Governments. Police. Armies…
- In this course, we will try to encompass the various scenario. We will abstract from the nature of the sensitive information whenever possible, and present the common principles of information protection, but we will also show that the kind of information (and of adversary) induces differences in the approach.

Why it is difficult to protect privacy

- Traditionally, privacy is protected via:
  - Anonymization
  - Encryption
  - Access control

- However, these methods often fail:
  - encryption and access control cannot protect against the inference of private information from public information
  - anonymization has been proved highly ineffective

Privacy via anonymity

Nowadays, many institutions and companies that collect data use anonymization, i.e., they remove all personal identifiers: name, address, SSN, …

"We don't have any raw data on the identifiable individual. Everything is anonymous"
(CEO of NebuAd, a U.S. company that offers targeted advertising based on browsing histories)

Similar practices are used by Facebook, MySpace, Twitter, …

Privacy via anonymity

However, anonymity-based sanitization has been shown to be **highly ineffective:** Several de-anonymization attacks have been carried out in the last decade

- The quasi-identifiers allow to retrieve the identity in a large number of cases.
- More sophisticated methods (k-anonymity, ℓ-diversity, …) take care of the quasi-identifiers, but they are still prone to composition attacks

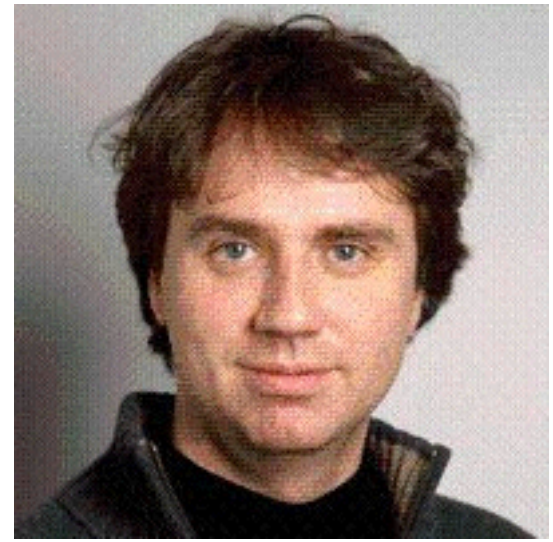
Differential Privacy at Google

RAPPOR

ABSTRACT

Randomized Aggregatable Privacy-Preserving Ordinal Response, or RAPPOR, is a technology for crowdsourcing statistics from end-user client software, anonymously, with strong privacy guarantees. In short, RAPPORs allow the forest of client data to be studied, without permitting the possibility of looking at individual trees. By applying randomized response in a novel manner, RAPPOR provides the mechanisms for such collection as well as for efficient, high-utility analysis of the collected data. In particular, RAPPOR permits statistics to be collected on the population of client-side strings with strong privacy guarantees for each client, and without linkability of their reports.

This paper describes and motivates RAPPOR, details its differential-privacy and utility guarantees, discusses its practical deployment and properties in the face of different attack models, and, finally, gives results of its application to both synthetic and real-world data.



Úlfar Erlingsson

Head of the team
on data security
and privacy at Google

Differential Privacy at Apple

Differential privacy is the statistical science of trying to learn as much as possible about a group while learning as little as possible about any individual in it.

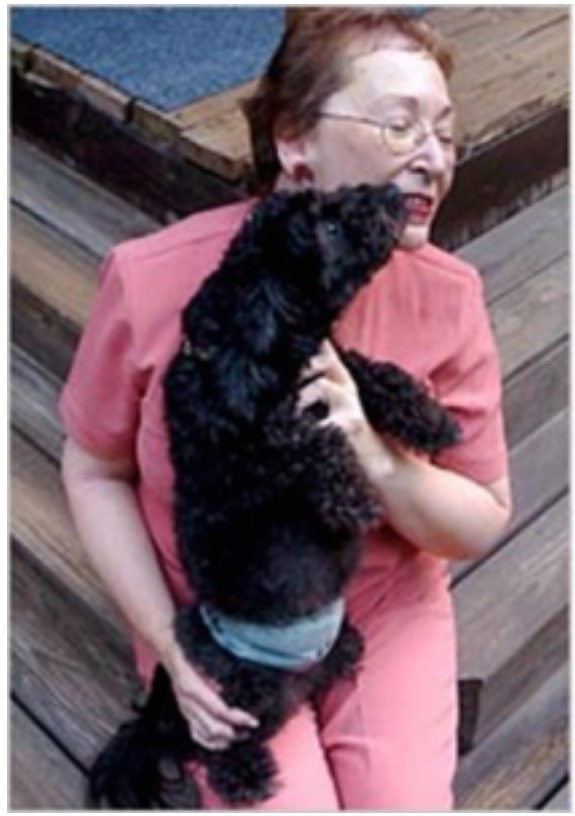
Apple has been doing some important work in this area to enable differential privacy to be deployed at scale."

Craig Federighi,
Vice president of
Software Engineering @Apple

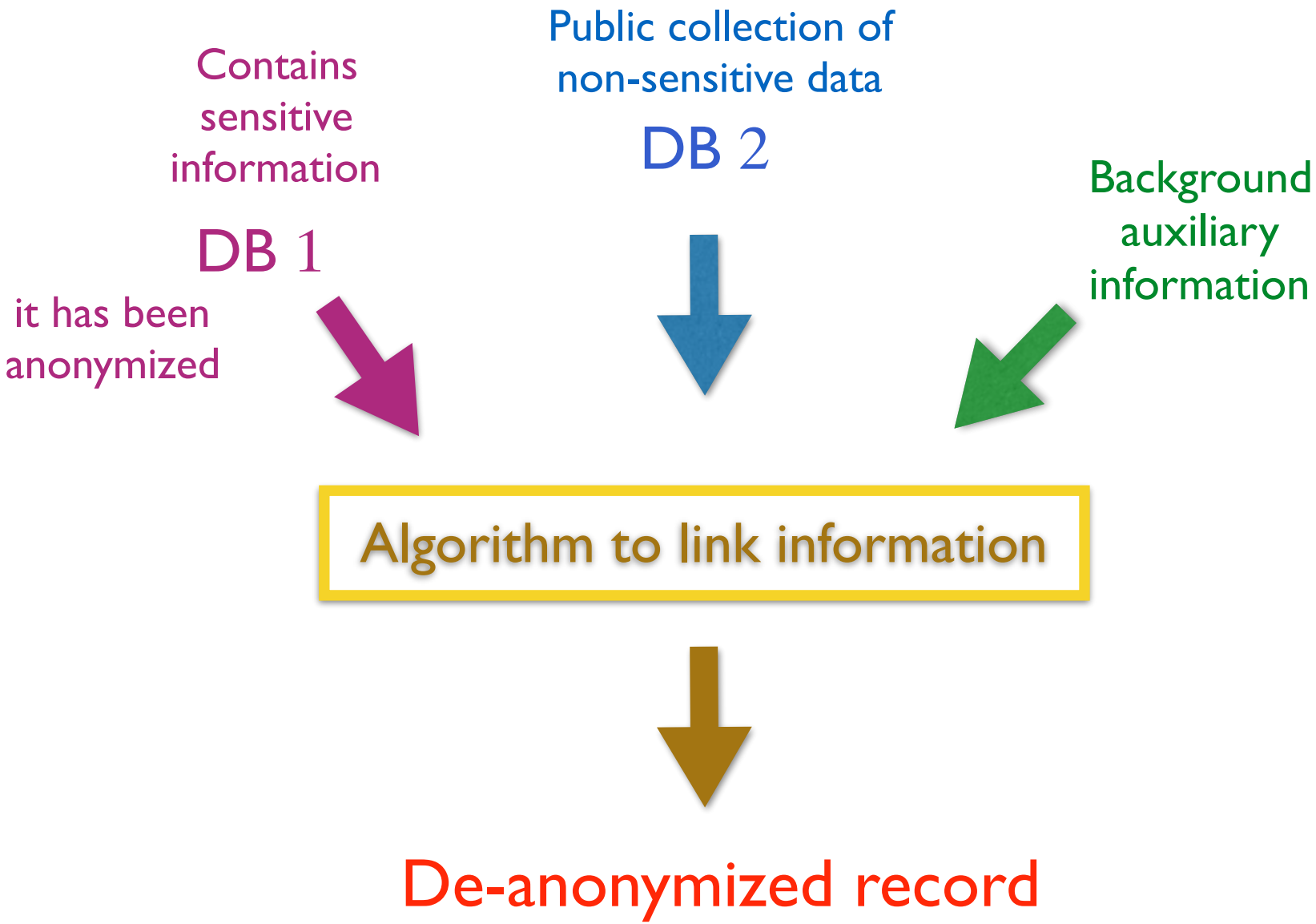
Keynote speech
Annual conference 2016
Apple software developers

Deanonymization attacks (I)

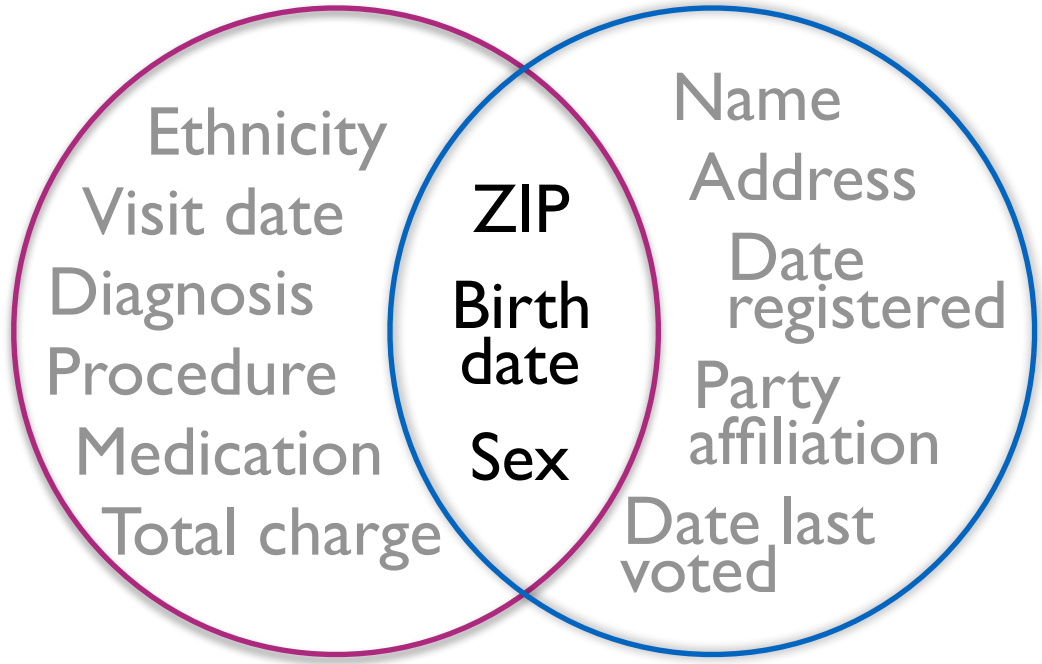
- In 2006, AOL Research released a text file containing twenty million search keywords for over 650,000 users, intended for research purposes.
- The file was anonymized (names where substituted by numbers as pseudonyms), but personally identifiable information was present in many of the queries. The NYT was able to locate an individual from the search records by cross referencing them with phonebook listings
- <<No. 4417749 conducted hundreds of searches over a three-month period on topics ranging from "numb fingers" to "60 single men" to "dog that urinates on everything.", "landscapers in Lilburn, Ga," several people with the last name Arnold and "homes sold in shadow lake" It did not take much investigating to follow that data trail to Thelma Arnold, a 62-year-old widow with three dogs who lives in Lilburn, Ga. >>



Sweeney's de-anonymization attack by linking



Sweeney's de-anonymization attack by linking



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

This attack has lead to the proposal of k-anonymity (that I will present later)

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.

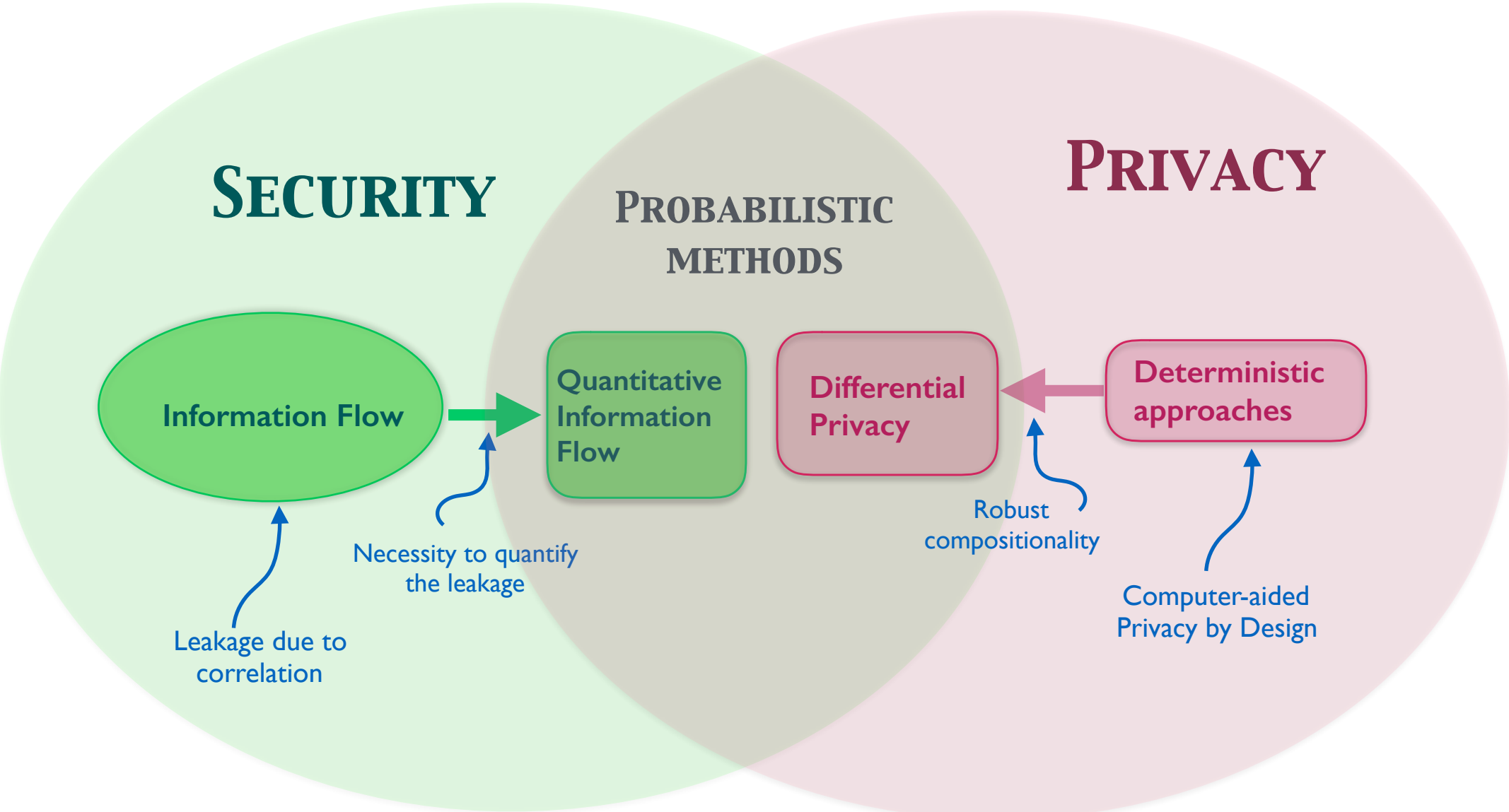
Statistical Databases

- **The problem:** we want to use databases to get statistical information (aka aggregated information), but without violating the privacy of the people in the database
  - We assume that the database itself is hidden. The only way to access information is by querying it
- For instance, medical databases are often used for research purposes. Typically we are interested in studying the correlation between certain diseases, and certain other attributes: age, sex, weight, etc.
  - A typical query would be: "*Among the people affected by the disease, what percentage is over 60 ?* "
  - Personal queries are forbidden. An example of forbidden query would be: " *Does Don have the disease ?* "

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

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

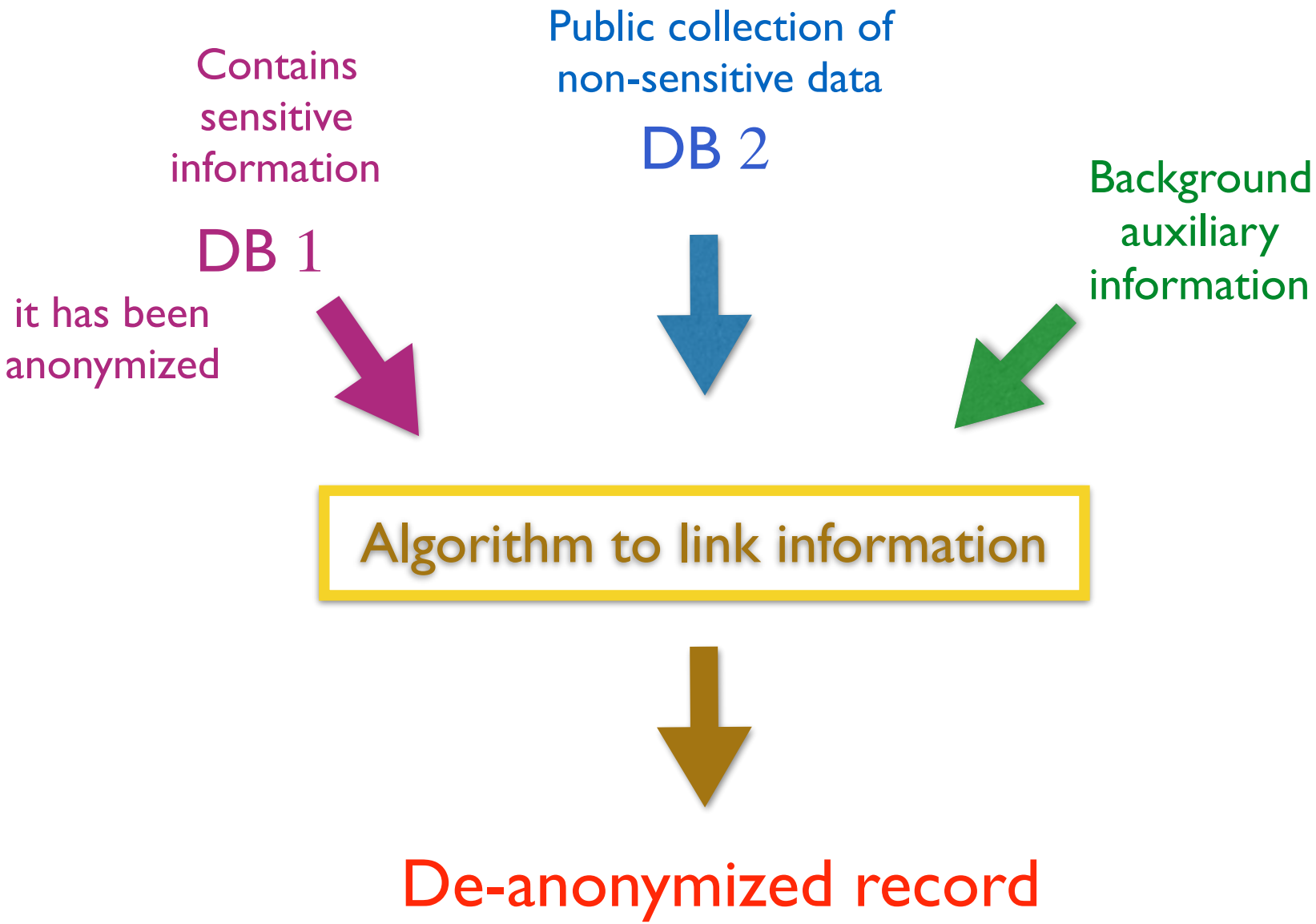
First solution: anonymization

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

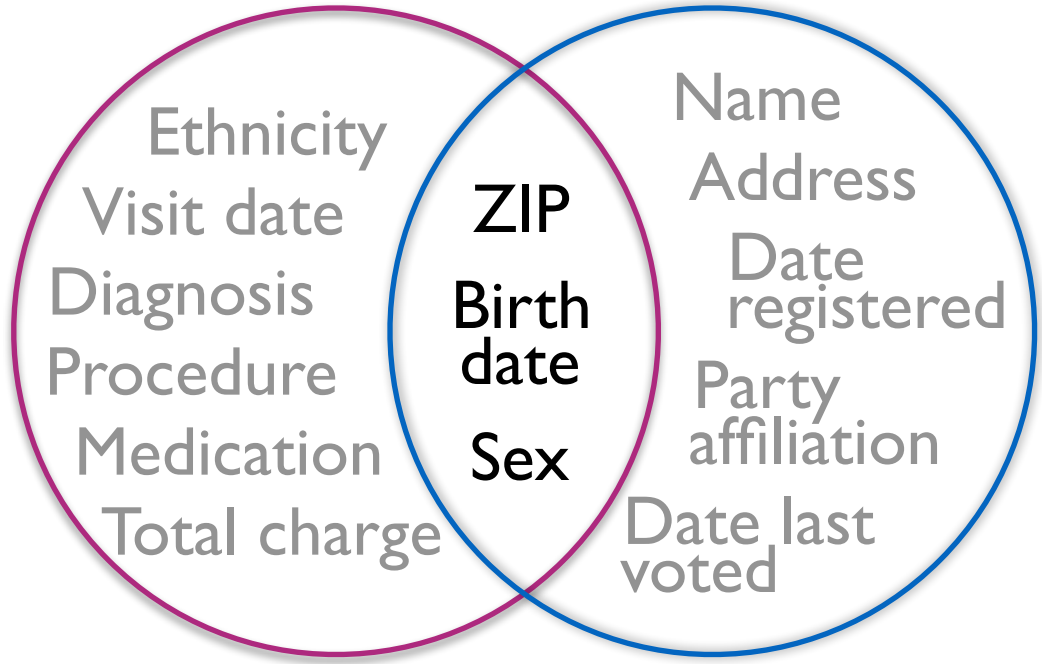
- Hystorically the first method, still used nowadays
- However, this solution has been (already several years ago) shown to be very weak and prone to de-anonymization attacks

	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]



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

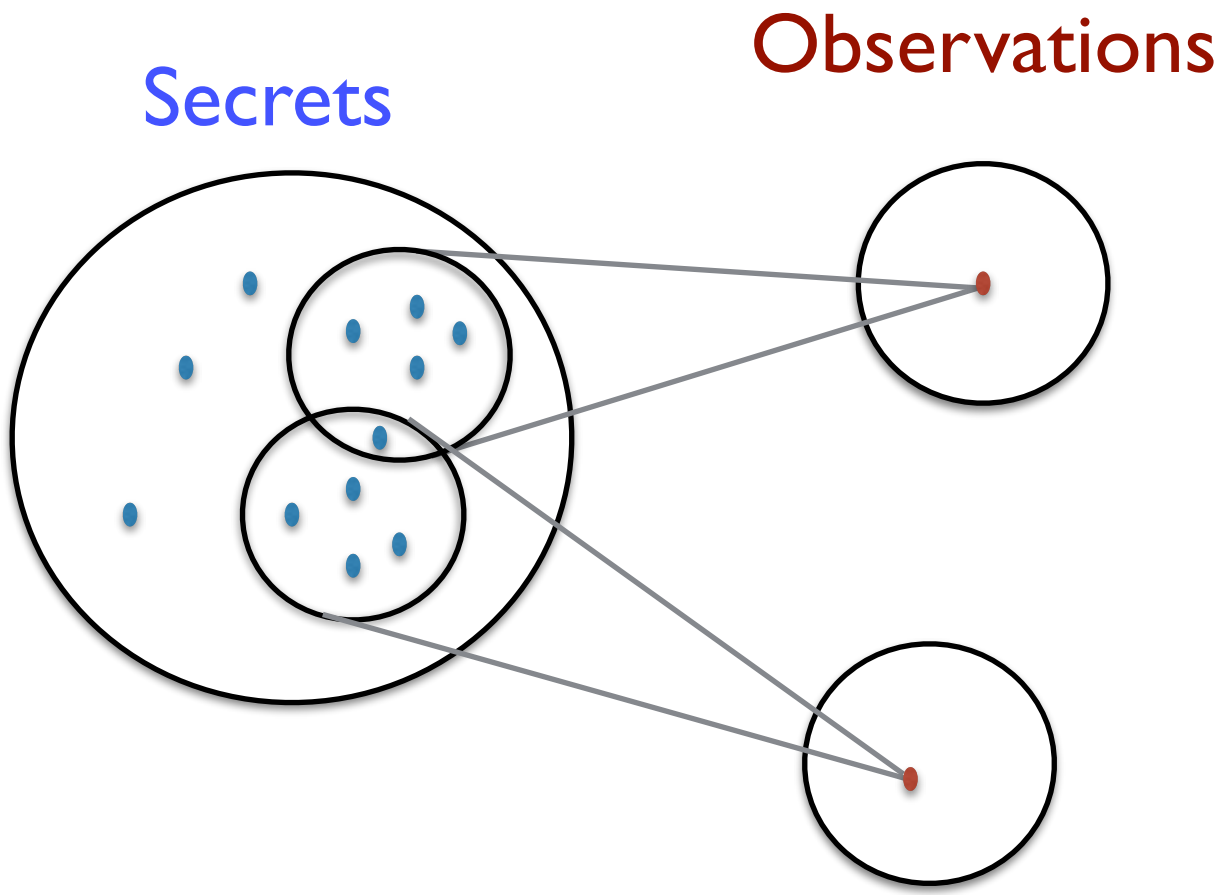
Answers: 40, 100

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

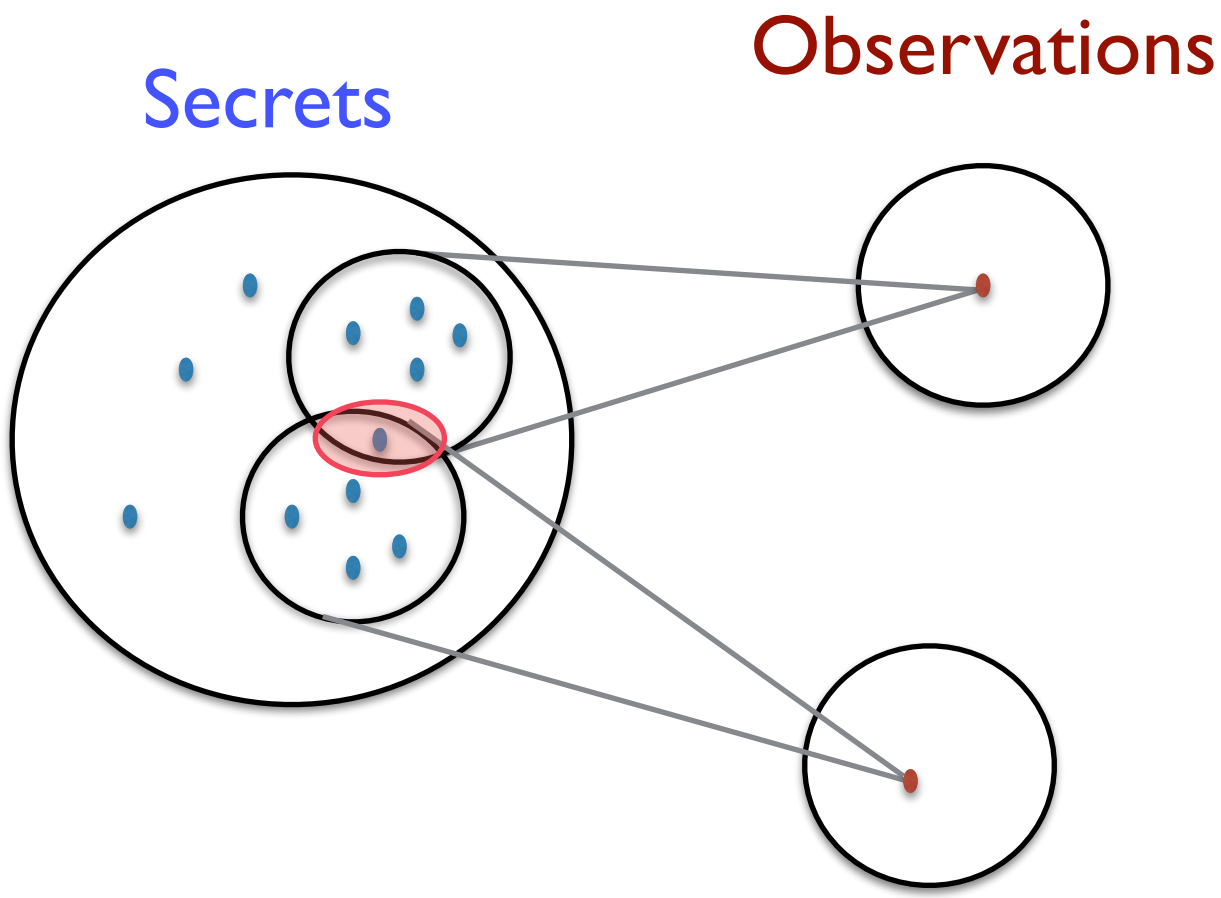
name	weight	disease
Alice	60	no
Bob	90	no
Carl	90	no
Don	100	yes
Ellie	60	no
Frank	100	yes

Alice	Bob
Carl	Don
Ellie	Frank

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	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

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 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,

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

  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 $\mathcal{D}(x)$ 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,

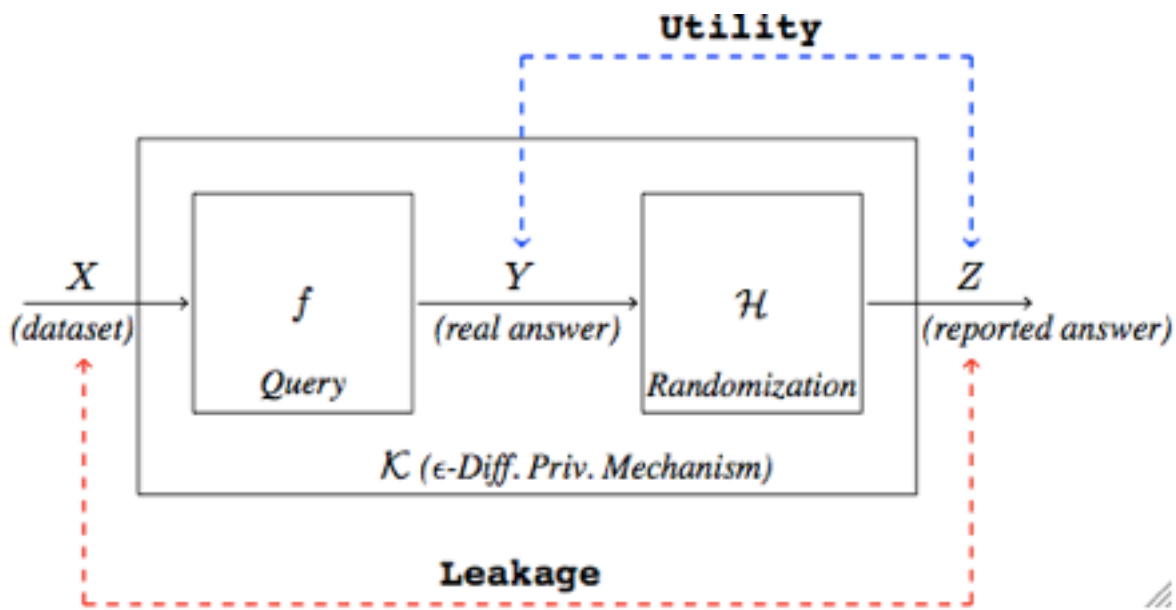
$$p(X_i = x_i | X_{others} = x_{others}, Z = z) \leq e^{\varepsilon} p(X_i = x_i | X_{others} = x_{others})$$

  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 $f: \mathcal{X} \to \mathcal{Y}$ and $\mathcal{K}: \mathcal{X} \to \mathcal{Z}$, 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 $f: \mathcal{X} \rightarrow \mathcal{Y}$.
- 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:

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

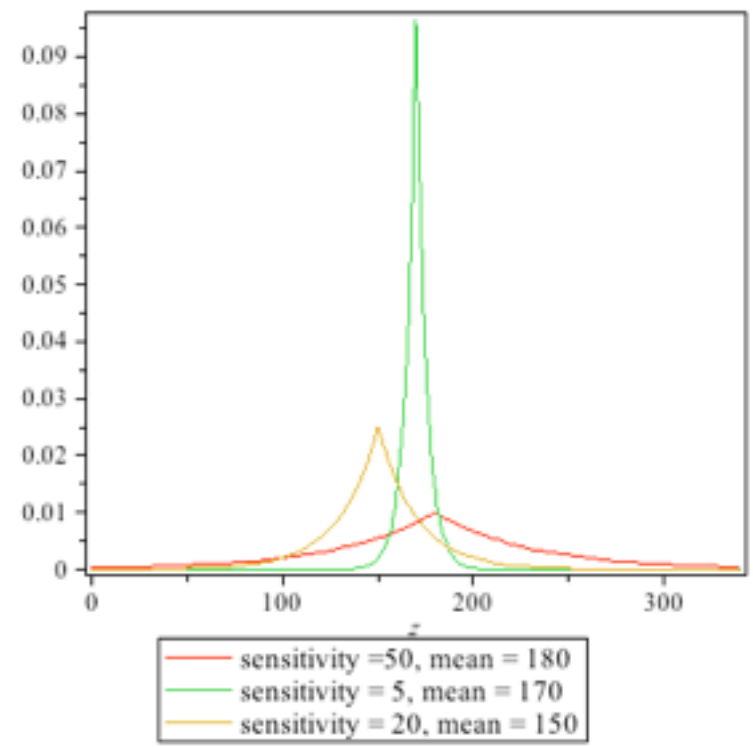
where Δf is the *sensitivity* of f:

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

($x \sim x'$ 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 30, is indeed equivalent to the original formulation of differential privacy

Foundations of Privacy

Lecture 7

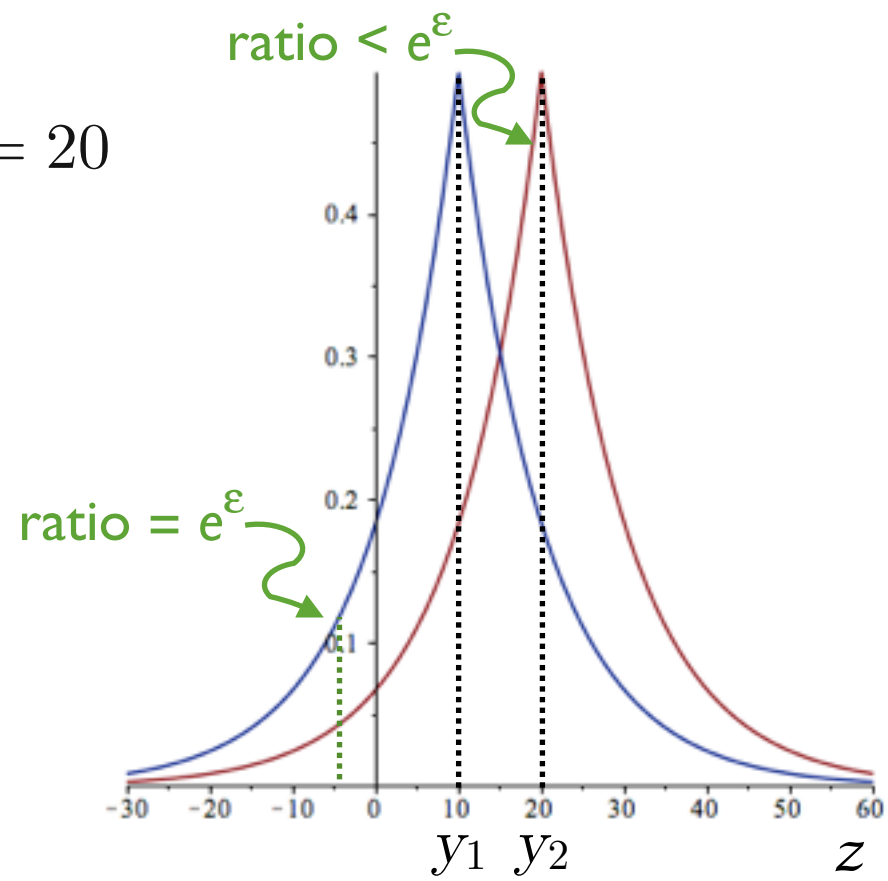
Plan of the lecture

- Solution of the exercise
- Brief recall of the Laplacian mechanism
- Discrete queries and Geometric Mechanism
- Truncated mechanisms
- Utility
- Optimal Mechanisms

Example of Laplacian Mechanism

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

  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

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

Gaussian noise

A gaussian noise would not satisfy differential privacy (although it satisfies a more relaxed form of privacy called (ε,δ)-privacy)

In fact, the formula for gaussian noise would be

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

and we can easily check that it does not satisfy DP for any value of σ

Sensitivity of the query in a Laplacian

- The sensitivity of the query and the level of privacy ε determine how uniform the noise is:
  - higher sensitivity $\Rightarrow$ more uniform noise
  - smaller $\varepsilon \Rightarrow$ more privacy, more uniform noise
- Intuitively, the more uniform is the noise, the less useful is the mechanism (the reported answer is less precise)
- To reduce the sensitivity of the query, we often assume that the database contains a minimum number of individuals
- **Example:** consider the query "What is the average age of the people in the DB ?". Assume that the age can vary from 0 to 120. Check the sensitivity in the following two cases:
  - the DB contains at least 100 records, or
  - there is no restriction.

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 **dense** numerical set, like the Reals or the Rationals.
- 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

Example: 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 are affected by diabetes?
    - How many diabetic people are obese?
- Question: what is the sensitivity of a counting query?

Normalization constant in a geometric mechanism

- In the geometric mechanism, the probability distribution of the noise is:

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

As usual, we can compute c (the normalization factor) by imposing that the sum of the probability on all Z is 1. It turns out that

$$c = \frac{1-\alpha}{1+\alpha} \quad \text{where} \quad \alpha = e^{-\frac{\varepsilon}{\Delta_f}}$$

$$\text{hence} \quad p(z|y) = \frac{1-\alpha}{1+\alpha}\,\alpha^{|z-y|}$$

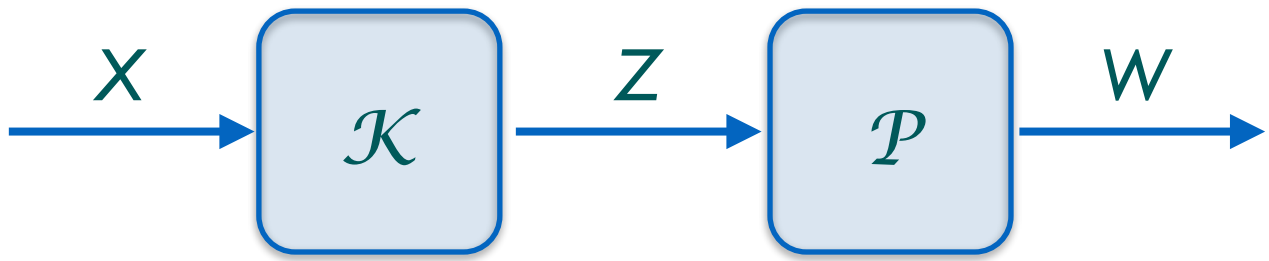
- **Examples:** Compute the geometric mechanism for the following queries:
  - " How many diabetic people weight more than 100 kilos ? "
  - " What is the max weight (in kilos) of a diabetic person ? "

Truncated geometric mechanism

- Often $\mathcal{Y}$ (the set of the true answers) does not coincide with the whole set of integers, but it is just subset, for instance an interval [a,b].
- With the geometric mechanism, however, the set of reported answers $\mathcal{Z}$ is always the whole set of integers
- It is often considered that it does not make much sense to report answers outside $\mathcal{Y}$. If $\mathcal{Y}$ is an interval [a,b], we can **truncate** the mechanism, i.e., set $\mathcal{Z} = \mathcal{Y}$, and transfer on the extremes a and b all the probability that (according to the geometric mechanism) would fall outside the interval: The probability that would fall to the left of a is transferred into a, and probability that would fall to the right of b is transferred into b.
- The same considerations hold for the Laplacian (truncated Laplacian)
- Exercise: Compute the truncated geometric mechanism for a counting query if the interval is [0,100]

Post-processing

- Post-processing a mechanism $\mathcal{K}$ consists in composing $\mathcal{K}$ with another function $\mathcal{P}$
  - $\mathcal{P}$ can be probabilistic or deterministic
  - $\mathcal{K}$ can be oblivious or not — it does not matter for the theorem below



Theorem: Post processing does not harm privacy. Namely, if $\mathcal{K}$ is ε-differentially private, then also $\mathcal{P} \circ \mathcal{K}$ is ε-differentially private

Truncation

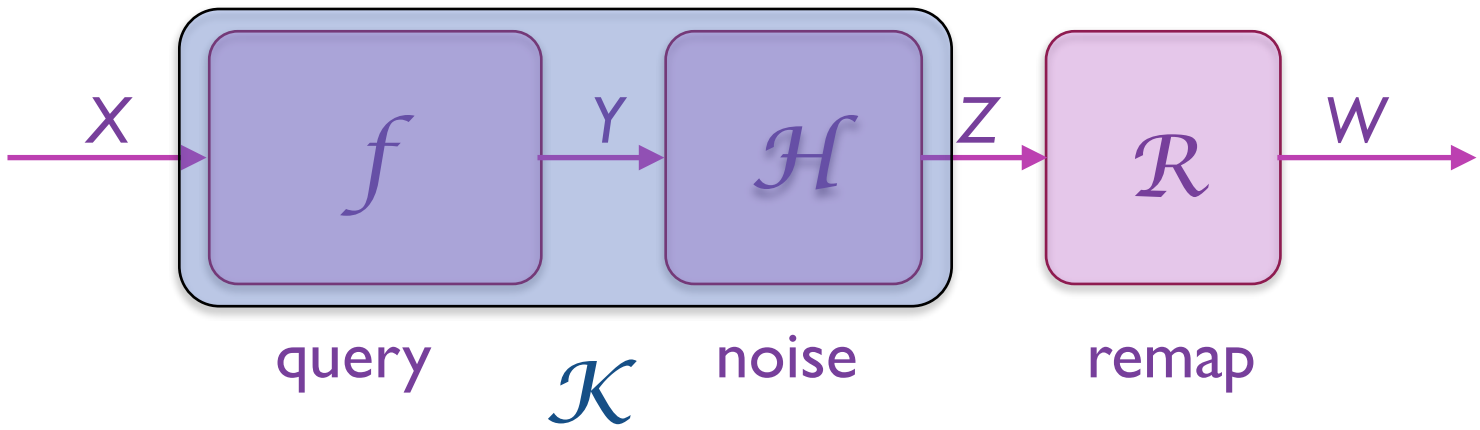
- Truncation is a typical example of post-processing
- In fact, assume that the true answer is in the interval [a,b]. Then truncation can be defined as follows: If the reported is smaller than a, then it gets remapped into a, and if it is greater than b, then it gets remapped into b.
- Because of the above theorem, truncation does not decrease the level of privacy.

Utility

- When a user sees the reported value z of the mechanism, he may take z as it is, or, based on his prior knowledge, he may guess another value w. We say that the user remaps z into w.
  Summarizing, we have:
- $\mathcal{X}$, the set of databases, with associated random variable X
- $\mathcal{Y}$, the set of true answers to the query f. Associated random variable Y
- $\mathcal{Z}$, the set of reported answers to the query f (after we apply the noise). Associated random variable Z
- $\mathcal{W}$, the set of guesses. Associated random variable W. $\mathcal{W}$ often coincides with $\mathcal{Y}$, but W usually does not coincide with Y.

Utility

- When a user sees the reported value z of the mechanism, he may take z as it is, or, based on his prior knowledge, he may guess another value w. We say that the user remaps z into w.
  Summarizing, we have:
- $\mathcal{X}$, the set of databases, with associated random variable X
- $\mathcal{Y}$, the set of true answers to the query f. Associated random variable Y
- $\mathcal{Z}$, the set of reported answers to the query f (after we apply the noise). Associated random variable Z
- $\mathcal{W}$, the set of guesses. Associated random variable W. $\mathcal{W}$ often coincides with $\mathcal{Y}$, but W usually does not coincide with Y.



Schema for an oblivious mechanism. In a non-oblivious one Z depend also on X.

Utility

- A gain function is a function

$$g : \mathcal{W} \times \mathcal{Y} \to \mathbb{R}$$

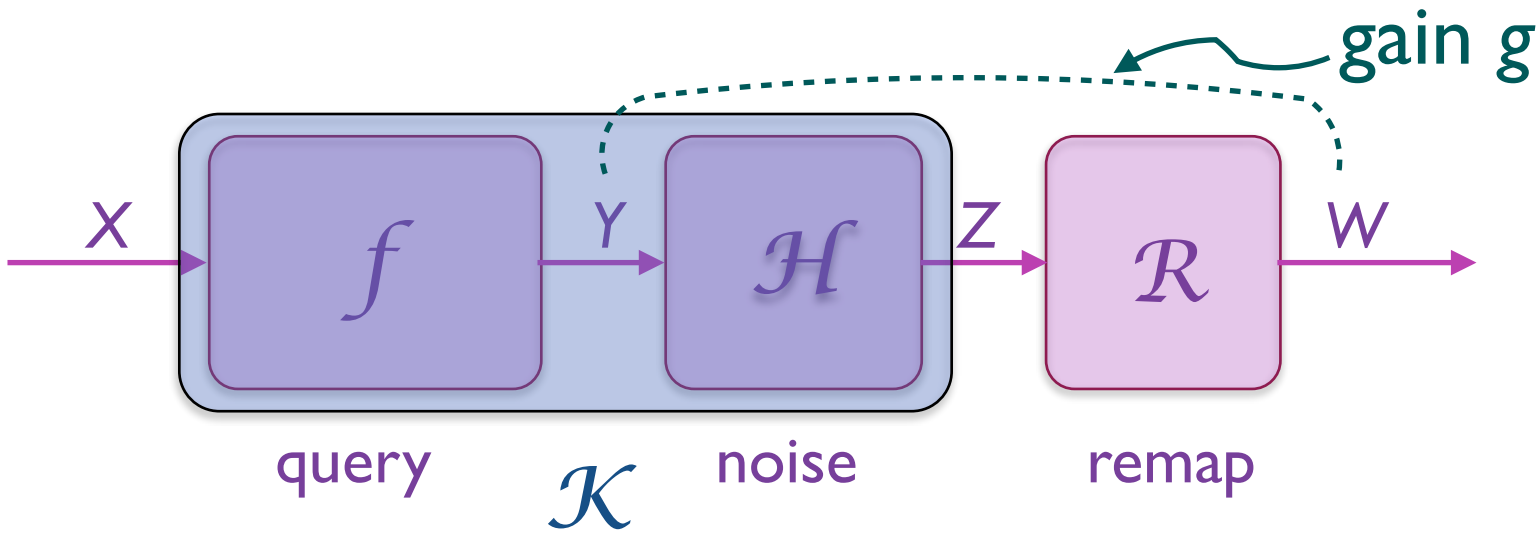
that represents the usefulness of the guess w when the true answer is y.

- Often there is a notion of distance d between w and y, representing how well w approximates y. Formally:

$$d : \mathcal{W} \times \mathcal{Y} \to \mathbb{R}$$

- The gain g is usually assumed to be anti-monotonic with respect to d. Namely:

$$\text{if } d(w, y) \leq d(w', y), \text{ then } g(w, y) \geq g(w', y)$$



Schema for an oblivious mechanism. In a non-oblivious one Z depend also on X.

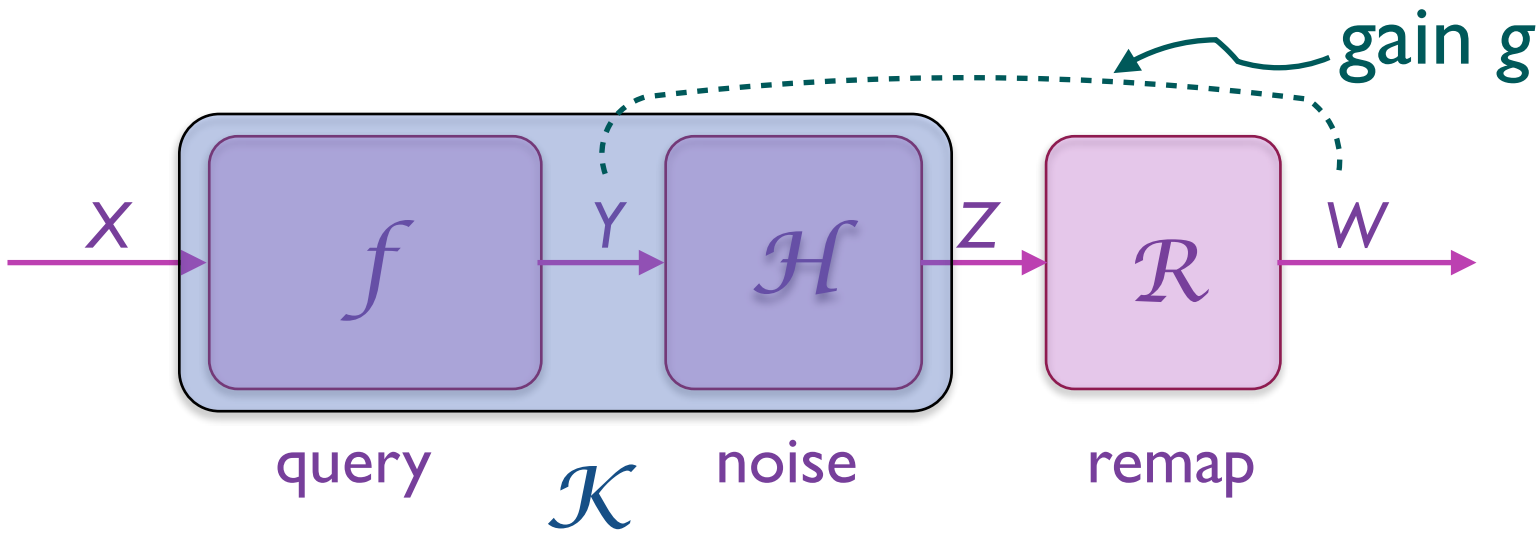
Utility

- Given a database x, consider the expected gain over all possible reported answers, for a certain remapping r. For an oblivious mechanism this is given by the formula:

$$\sum_z p_{\mathcal{H}}(z|f(x))g(r(z), f(x))$$

- For a generic (possibly non oblivious) mechanism, this is given by:

$$\sum_z p_{\mathcal{K}}(z|x)g(r(z), f(x))$$



Schema for an oblivious mechanism. In a non-oblivious one Z depend also on X.

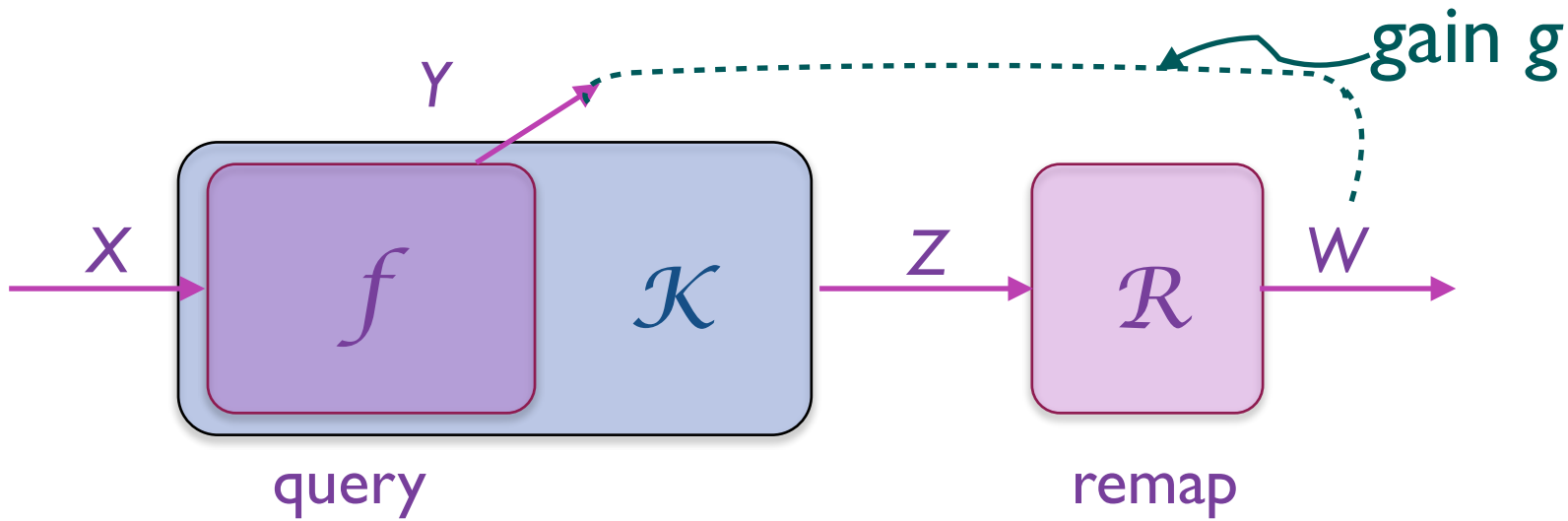
Utility

- The utility $\mathcal{U}$ of a mechanism is the maximum expected gain over all possible databases. The maximum is over all possible remappings: It is assumed that the user is rational and therefore makes the guesses that are the most useful to him. Note that $\mathcal{U}$ depends also on the prior π over $\mathcal{X}$ Formally, let us denote by r a remapping function. For an oblivious mechanism we have:

$$\mathcal{U}(\mathcal{K}, \pi, g) = \max_r \sum_x \pi(x) \sum_z p_{\mathcal{H}}(z|f(x)) g(r(z), f(x))$$

For a general (possibly non-oblivious) mechanism, we have:

$$\mathcal{U}(\mathcal{K}, \pi, g) = \max_r \sum_x \pi(x) \sum_z p_{\mathcal{K}}(z|x) g(r(z), f(x))$$



Example

The simplest gain function is the identity relation:

$$g(w,x) = \begin{cases} 1 & w = x \\ 0 & w \neq x \end{cases}$$

It represents the situation in which we are happy only if we guess the true answer.
With this gain function, the utility becomes (we give the formula for the oblivious case, the non-oblivious one is analogous):

$$\begin{aligned} \mathcal{U}(\mathcal{K},\pi,g) &= \max_r \sum_x \pi(x) \sum_z p_{\mathcal{H}}(z|f(x))\ g(r(z),f(x)) \\ &= \max_r \sum_y p_f(y) \sum_z p_{\mathcal{H}}(z|y)\ g(r(z),y) \\ &= \sum_z \max_y (p_f(y)\ p_{\mathcal{H}}(z|y)) \end{aligned}$$

This utility function essentially gives the expected probability of guessing the true answer. It is the converse of the Bayes risk

Example

Another typical gain function is the converse of the distance:

$$g(w, x) = D - d(w, x)$$

where D is the maximum possible distance between reported answers and true answers (it works well for truncated mechanisms). If such maximum does not exists, we can take $D = 0$. The only problem is that we get negative gains
With this gain function, the utility is the expected distance between our best guess and the true answer. It gives a measure of how good is the approximated of the true answer that we can get with the mechanism.

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
- Note that the privacy does not depend on the prior, but the utility (in general) does.
- In the finite case the optimal mechanism can be computed with linear optimization techniques, where the variables are the conditional probabilities p(z | y)
  where y is the exact answer and z is the reported answer
- A mechanism is universally optimal if it is optimal for all priors π

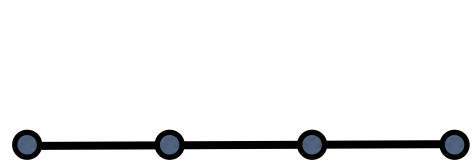
Privacy vs utility: two fundamental results

1. [Ghosh et al., STOC 2009]
   The geometric mechanism and the truncated geometric mechanism are universally optimal for counting queries and any anti-monotonic gain function

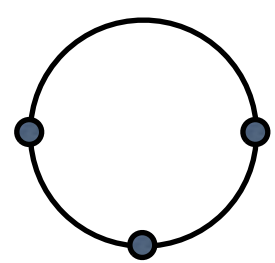
Privacy vs utility: two fundamental results

2. [Brenner and Nissim, STOC 2010] The counting queries are the only kind of queries for which a universally optimal mechanism exists

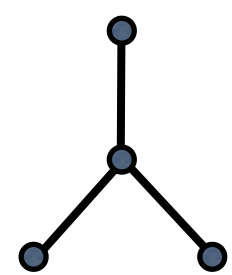
   - This means that for other kind of queries one the optimal mechanism is relative to a specific user.
   - The precise characterization is given in terms of the graph $(\mathcal{Y}, \sim)$ induced by $(\mathcal{X}, \sim)$



ok



not ok



not ok

Exercises

1. Define the noise density function for the Laplacian mechanism for the query "What is the percentile of the people in the DB who earn more than 10K Euro a month", assuming that the database contains at least 1000 elements.

2. Define the truncated Laplacian mechanism for the above query. Note that $\mathcal{Y}$ is the interval [0,100].

3. Prove that ε-differential privacy can be equivalently defined as follows

> $\mathcal{K}$ is ε-differentially private if for every pair of databases $x_1, x_2 \in \mathcal{X}$ (not necessarily adjacents), and for every $z \in \mathcal{Z}$, we have:
>
> $$p(Z = z|X = x_1) \leq e^{\varepsilon h(x_1,x_2)} p(Z = z|X = x_2)$$
>
> where $h(x_1, x_2)$ represents the Hamming distance between x_1 and x_2

Exercises

4. Compute the utility of the geometric mechanism for a counting query, with privacy degree ε, on the uniform prior distribution, with the gain function defined as the identity relation.

5. Same exercise, but with the gain function defined as the converse of the distance.

6. Find a mechanism for the same counting query, with the same degree of privacy, but lower utility.

7. We saw that post-processing cannot decrease privacy. Can it decrease the utility? Motivate your answer.

Foundations of Privacy

Lecture 8

Motivation

Can differential privacy be adapted to different privacy requirements?

Can we use differential privacy on secrets that are not databases?

Outline

- **Generalization of differential privacy**
- Privacy in the context of statistical databases
- Privacy in location-based systems

Differential Privacy, adjacent databases

- Adjacency: $x \sim_h x'$ iff they differ in exactly one individual

$$x = \langle 32, 41, 27 \rangle$$
$$x' = \langle 32, 52, 27 \rangle$$

- $K : \mathcal{X} \to \mathcal{P}(\mathcal{Z})$ satisfies ϵ-differential privacy iff

$$K(x)(Z) \leq e^{\epsilon}\, K(x')(Z) \qquad \forall x \sim_h x'$$

- ϵ : distinguishability level between adjacent databases

Differential Privacy, any databases

- Hamming distance $d_h(x, x')$: # of elements in which x, x' differ

$$x = \langle 32, 41, 27 \rangle$$
$$x' = \langle 21, 52, 27 \rangle \qquad d_h(x, x') = 2$$

- Differential privacy can be equivalently defined as follows:

$$K(x)(Z) \leq e^{\epsilon d_h(x,x')} K(x')(Z) \quad \forall x, x'$$

- $\epsilon d_h(x, x')$: distinguishability level between any databases

Differential Privacy, generalization

- Arbitrary domain of secrets $\mathcal{X}$
- $\boldsymbol{\varepsilon}(x, x')$: distinguishability level between x, x'
- Expected properties:
  - $\boldsymbol{\varepsilon}(x, x) = 0$
  - $\boldsymbol{\varepsilon}(x, x') = \boldsymbol{\varepsilon}(x', x)$
  - $$\left.\begin{array}{l}\boldsymbol{\varepsilon}(x_1, x_2) \leq b \\ \boldsymbol{\varepsilon}(x_3, x_2) \leq b\end{array}\right\} \Rightarrow \boldsymbol{\varepsilon}(x_1, x_3) \leq f(b)$$

Differential Privacy, generalization

$d_{\mathcal{X}}$-privacy

$K(x)(Z) \leq e^{d_{\mathcal{X}}(x,x')}\, K(x')(Z) \quad \forall x, x'$

- the less distinguishable two secrets are, the more similar the outcome should be
- There is no ϵ, but we can just rescale the metric in order to obtain the desired level of privacy: $d_{\mathcal{X}} = \epsilon d_{\mathcal{X}}{}'$
- ϵ-differential privacy $= \epsilon d_h$-privacy

Differential Privacy, generalization

$d_{\mathcal{X}}$-privacy

$K(x)(Z) \leq e^{d_{\mathcal{X}}(x,x')}\ K(x')(Z) \quad \forall x, x'$

This notion of privacy protects the accuracy of the data

- Foundations
  - Compositionality
  - Implementation: Laplacian
  - Optimality results
- Applications
  - Statistical databases - (normalized) Manhattan distance
  - Location privacy - Geographical distance
  - In general, every domain equipped with a metric

Compositionality

If K, K' are $d_{\mathcal{X}}$ and ${d_{\mathcal{X}}}'$ differentially private, then the composition of the two mechanisms, (K, K'), is $d_{\mathcal{X}} + {d_{\mathcal{X}}}'$ differentially private

Answering queries

- Query $f : \mathcal{X} \to \mathcal{Y}$
- f is Δ-sensitivite wrt $d_{\mathcal{X}}, d_{\mathcal{Y}}$ iff:

$$\Delta = \max_{x,x'} \frac{d_{\mathcal{Y}}(f(x), f(x'))}{d_{\mathcal{X}}(x, x')}$$

- If $H : \mathcal{Y} \to \mathcal{P}(\mathcal{Z})$ satisfies $d_{\mathcal{Y}}$-privacy
  then $H \circ f$ satisfies $\Delta d_{\mathcal{X}}$-privacy
- H can be implemented in the usual way as Laplacian noise:

$$H(y)(z) = c \cdot e^{\frac{-d_{\mathcal{Y}}(z,y)}{\Delta}\epsilon}$$

We can easily prove that H satisfies $\frac{d_{\mathcal{Y}}}{\Delta}\epsilon$-privacy , and consequently $H \circ f$ satisfies $d_{\mathcal{X}}\epsilon$-privacy

The normalized Manhattan metric

- The Hamming distance is independent from the actual values

$$x_1 = \langle 32, 0, 27 \rangle$$
$$x_2 = \langle 32, 0.01, 27 \rangle$$
$$x_3 = \langle 32, 10^6, 27 \rangle \qquad d_h(x_1, x_2) = d_h(x_1, x_3) = 1$$

- the disting. level between x_1, x_2 and x_2, x_3 is the same
- Many queries are insensitive to minor changes in values
- If ϵ is "weak", we might require higher protection for x_1, x_2

The normalized Manhattan metric

- Manhattan metric:

$$d_1(x, x') = \sum_{i=1}^{n} d_\nu(x[i], x'[i])$$

- Normalized Manhattan metric:

$$\widetilde{d}_1(x, x') = \frac{d_1(x, x')}{d_\nu(\mathcal{V})}$$

  where $d_\nu(\mathcal{V})$ is the maximum distance among the values

- Stronger that Hamming: $\widetilde{d}_1(x, x') \leq d_h(x, x')$

$$x_1 = \langle 32, 0, 27 \rangle$$
$$x_2 = \langle 32, 0.01, 27 \rangle \qquad \widetilde{d}_1(x_1, x_2) = 10^{-8}$$
$$x_3 = \langle 32, 10^6, 27 \rangle \qquad \widetilde{d}_1(x_1, x_3) = 1$$

Advantages of the normalized Manhattan metric

Sensitivity:

- For a family of queries (sum, average, percentile, ...), the sensitivity wrt $\widetilde{d}_1, d_{\mathbb{R}}$ and $d_h, d_{\mathbb{R}}$ coincide
- In general, $\widetilde{d}_1$ is smaller than d_h
- hence we get stronger privacy with the same noise

Optimality:

- If the set of values is discrete, then sum, average and percentile queries induce a graph structure which is a straight line
- As a consequence, the Geometric mechanism is universally optimal for sum, average and percentile queries wrt $\widetilde{d}_1$
- In contrast, we saw that only counting queries have universally optimal mechanisms wrt d_h

The Manhattan metric

- We can use the Manhattan metric without normalization:

$$d_1(x, x') = \sum_{i=1}^{n} d_\nu(x[i], x'[i])$$

- d_1 can be much higher that Hamming, but Δ will be proportionally smaller than the usual sensitivity, so the protection, with respect to the introduced noise, is comparable.
  Example:

$$x_1 = \langle 32, 0, 27 \rangle$$
$$x_2 = \langle 32, 0.01, 27 \rangle \qquad \widetilde{d}_1(x_1, x_2) = 10^{-2}$$
$$x_3 = \langle 32, 10^6, 27 \rangle \qquad \widetilde{d}_1(x_1, x_3) = 10^6$$

The Manhattan metric

- The Manhattan metric be useful when we need to prevent the attacker from getting very precise data (for instance because they can be used to identify an individual),
- Trade-off between privacy and utility
- Optimality results similar to $\widetilde{d}_1$

Motivation

Geographical information is becoming essential for a variety of services: LBS, advertising, social networks, data mining, ...

Privacy: location data are often sensitive and need protection

Location-Based Systems

A **location-based system** is a system that uses geographical information in order to provide a service.

- Retrieval of Points of Interest (POIs).
- Mapping Applications.
- Deals and discounts applications.
- Location-Aware Social Networks.

Location-Based Systems

- Location information is sensitive. (it can be linked to home, work, religion, political views, etc).
- Ideally: we want to **hide our true location**.
- Reality: we need to **disclose some information**.

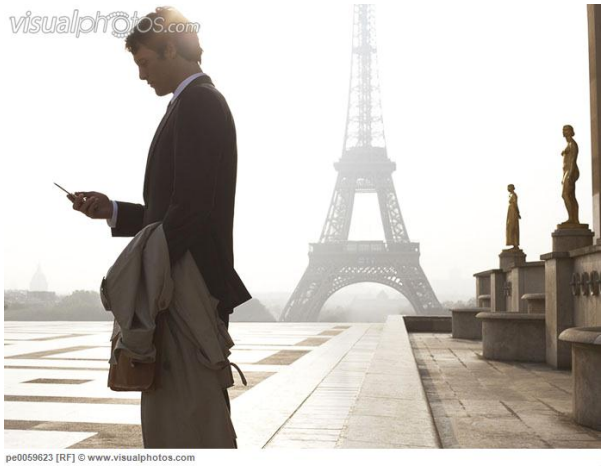
Motivating example

Goal:

- Hide the user's location (not identity) from the service provider
- Formal privacy guarantee

Constraints:

- Implementable in real-time on a smartphone
- No trusted party
- Optimally: no peer-to-peer communication

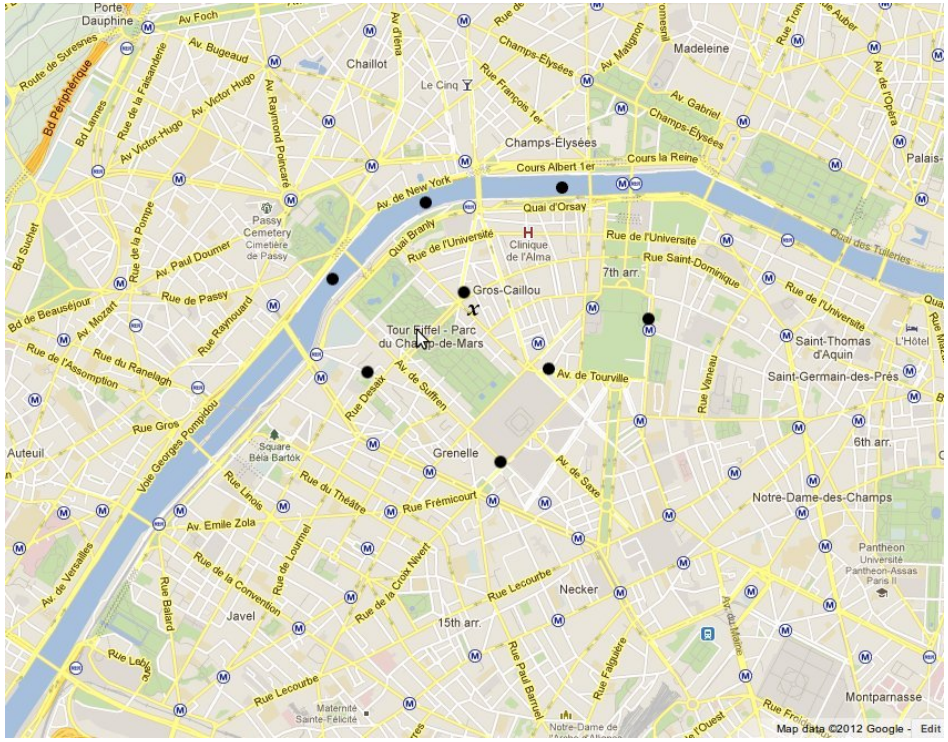


Existing privacy notions

k-anonymity (or l-diversity)

Hide the user's location among k points

- Include $k-1$ randomly generated points in the query
- Use a cloaking region including k points of interest



Problem: depends on the attacker's side information

Existing privacy notions

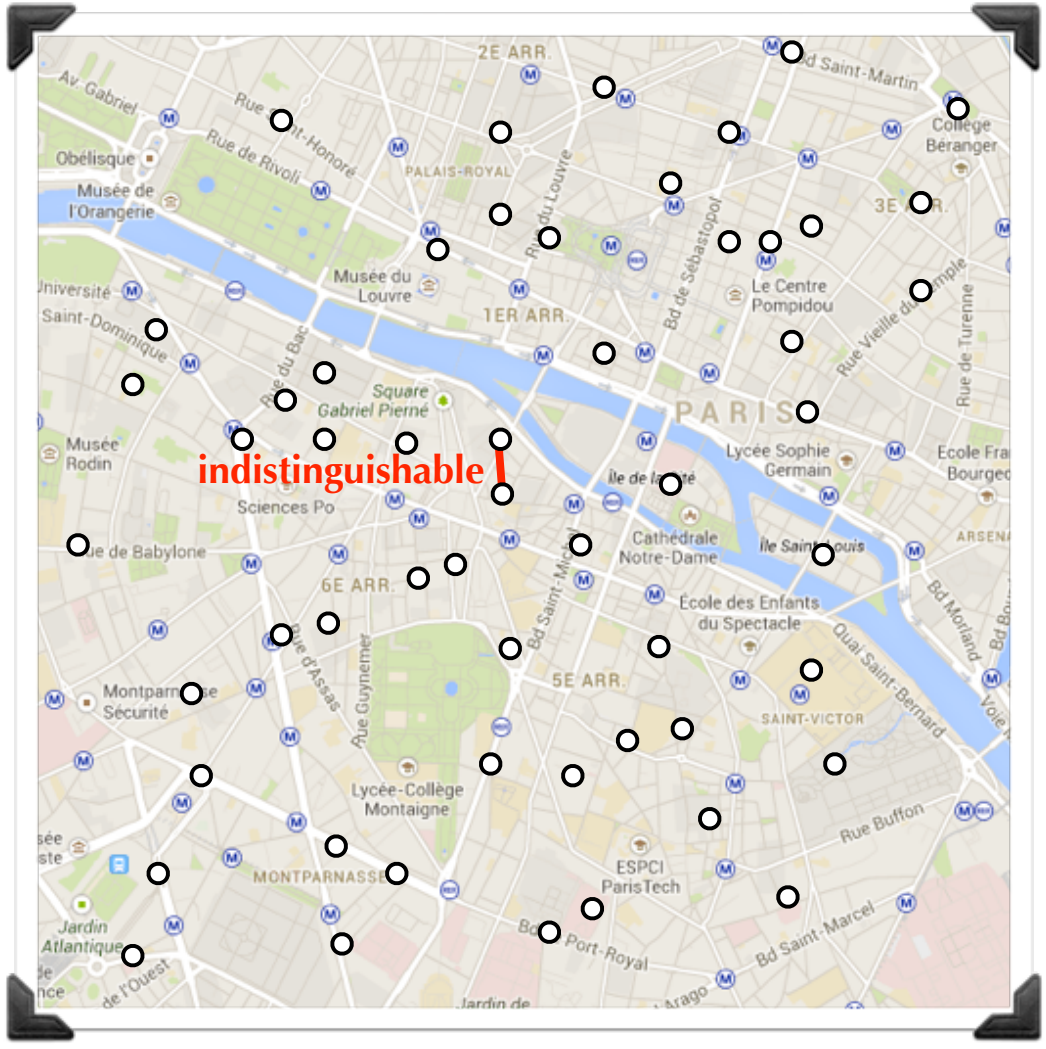
Differential Privacy

Changes in a single user's value should have negligible effect on the reported value

- Useful for publishing aggregate information about a large number of users
- Has been used in the context of geo-location
- Inadequate for our motivating example

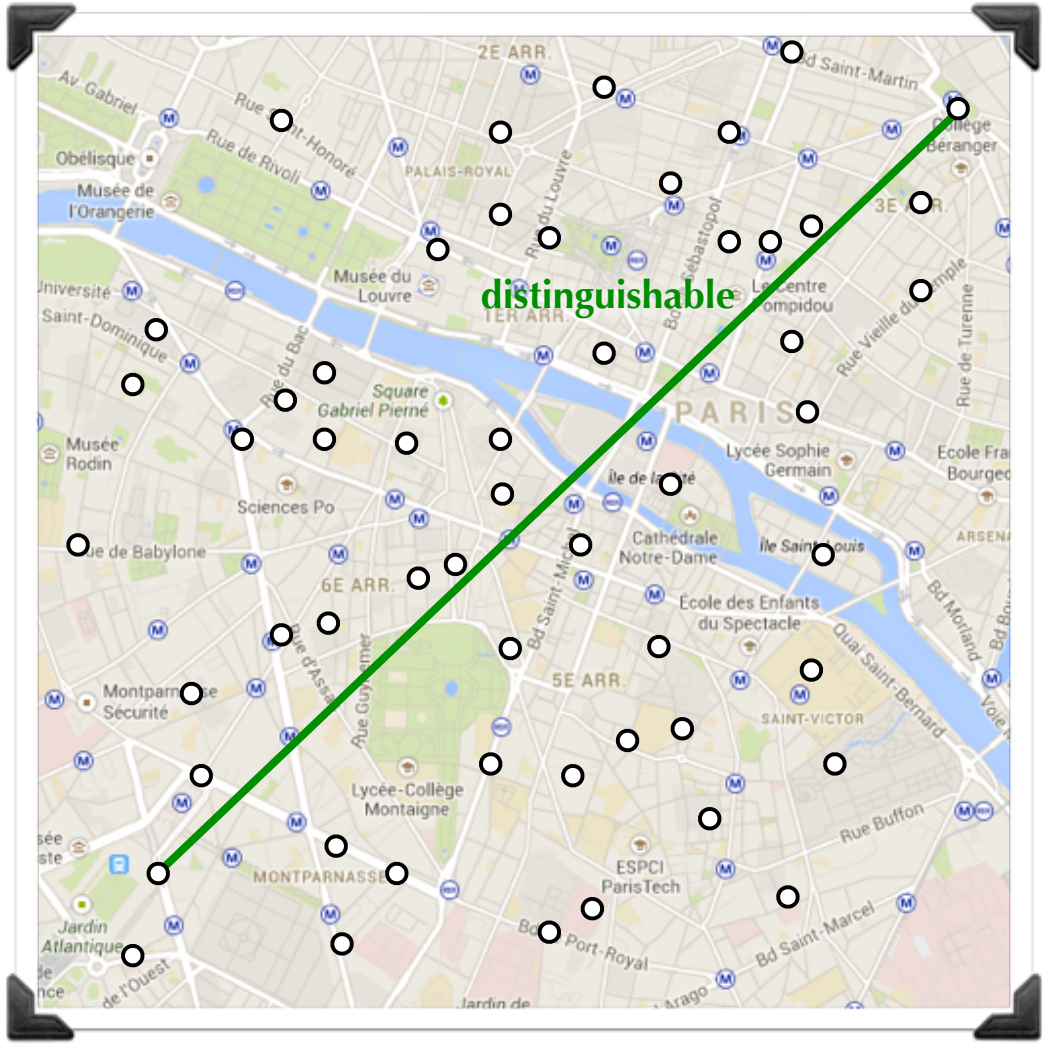
Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.



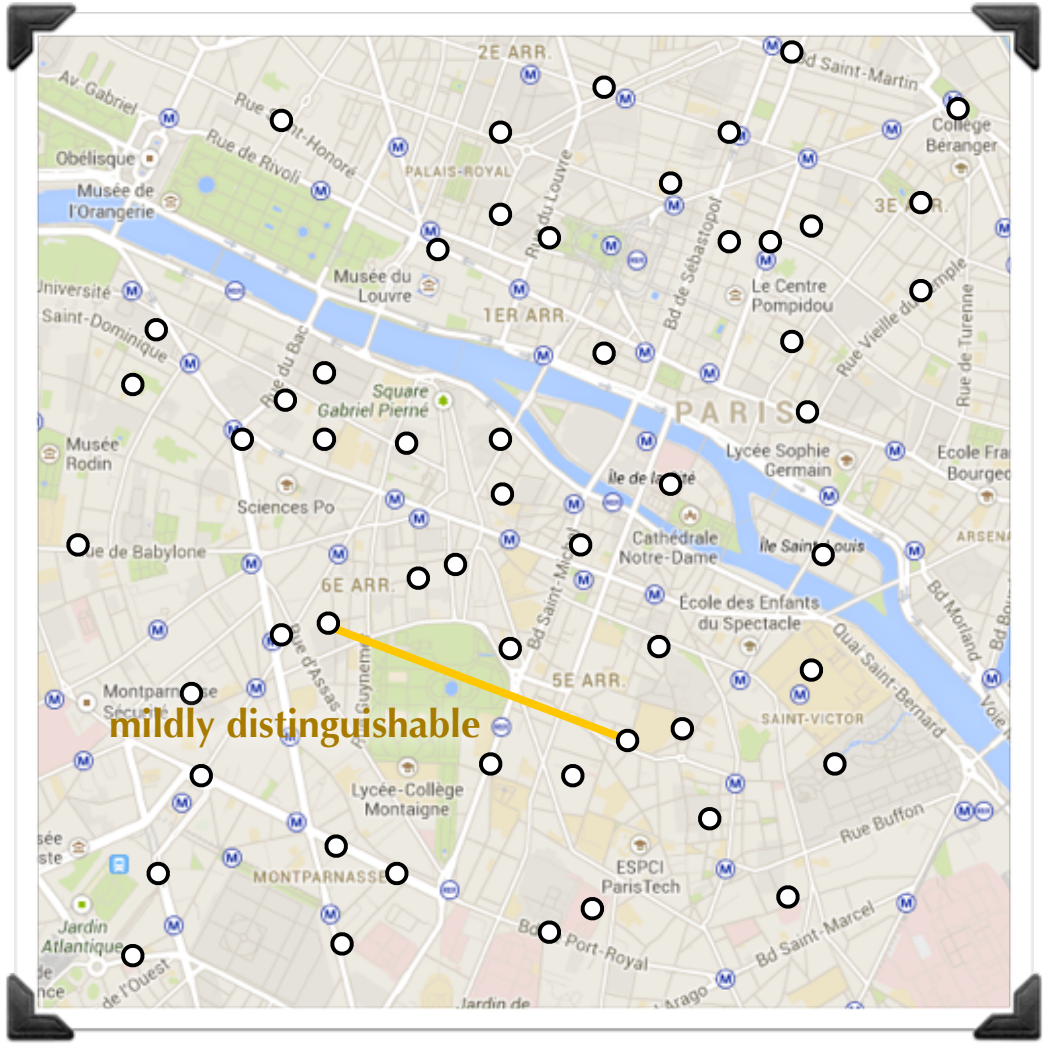
Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.



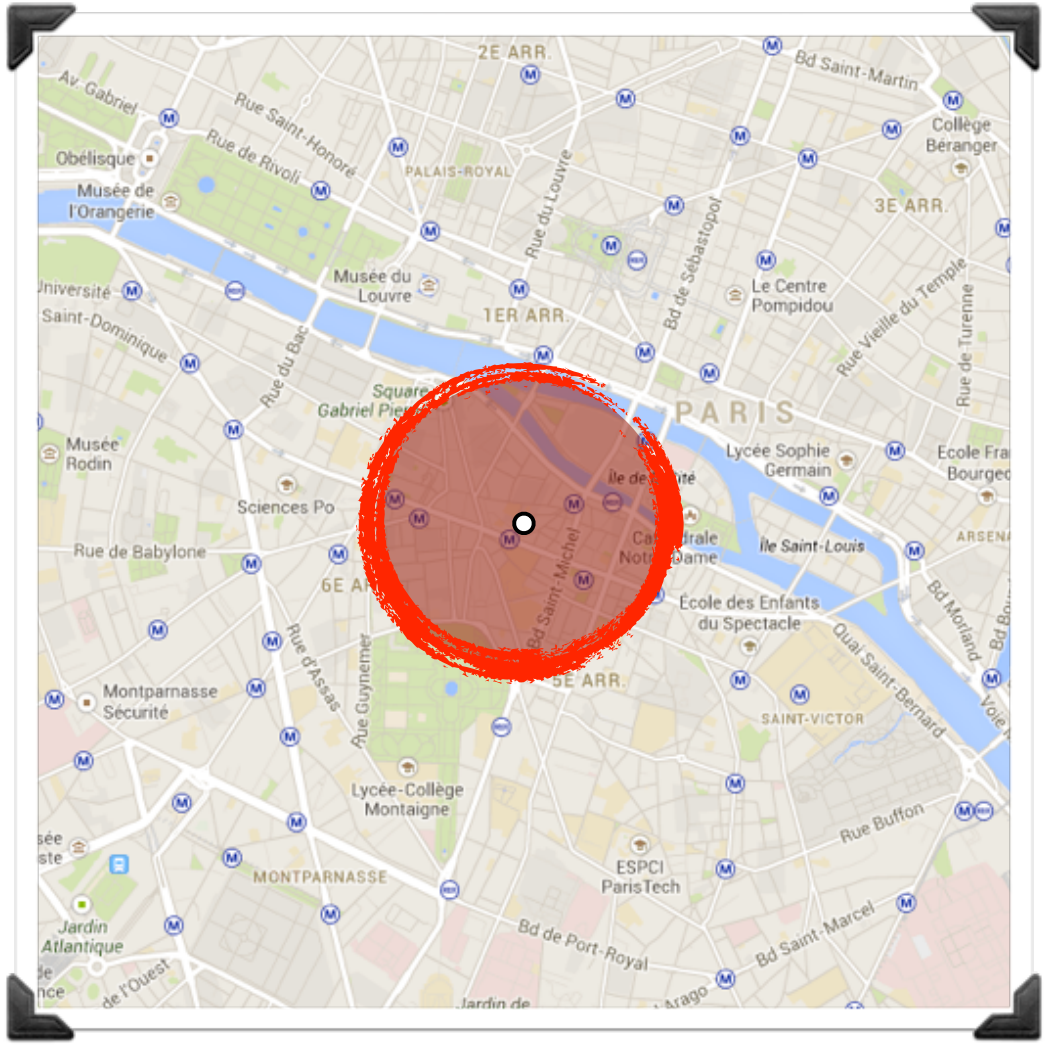
Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.



Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.

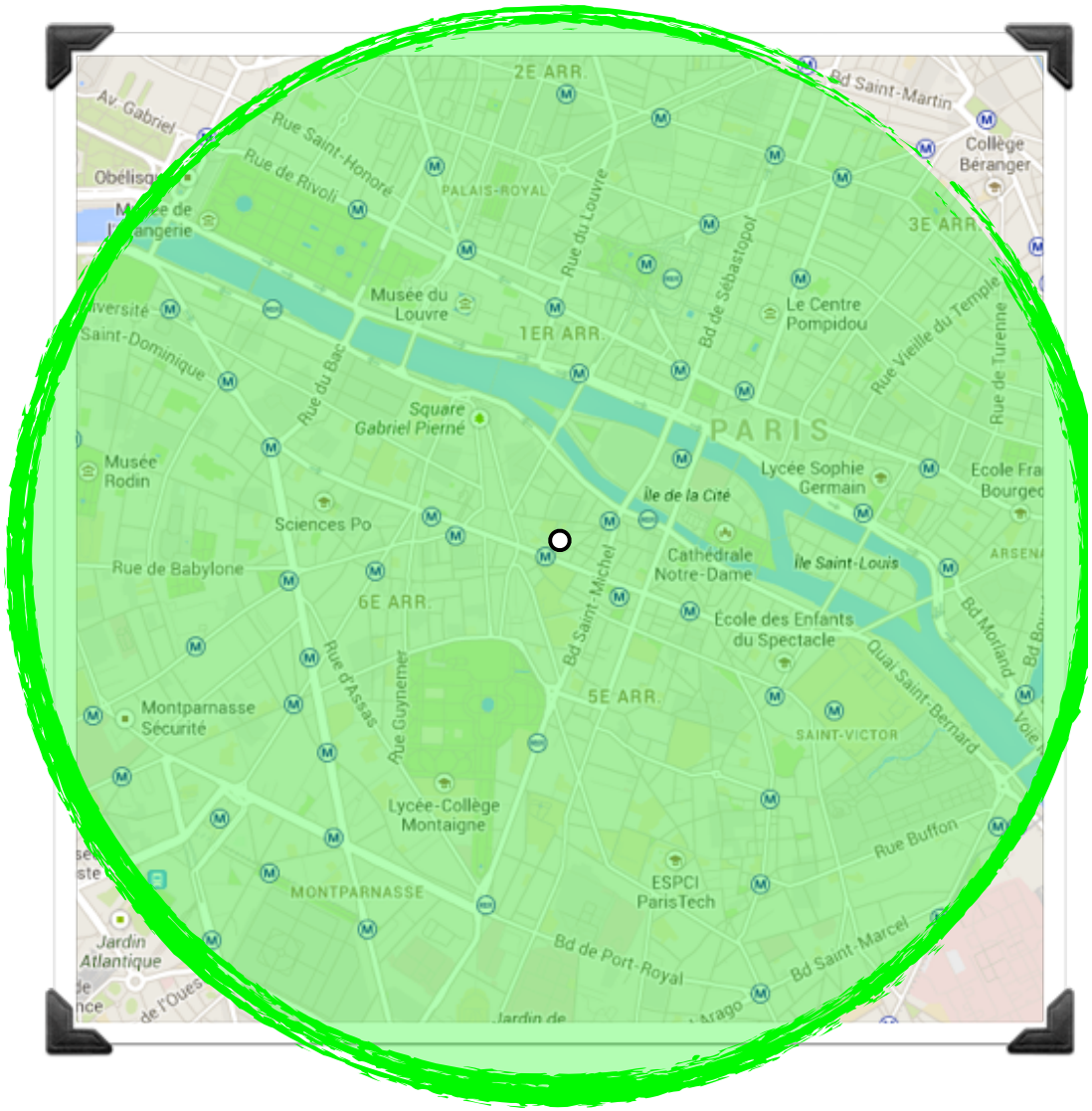


Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.

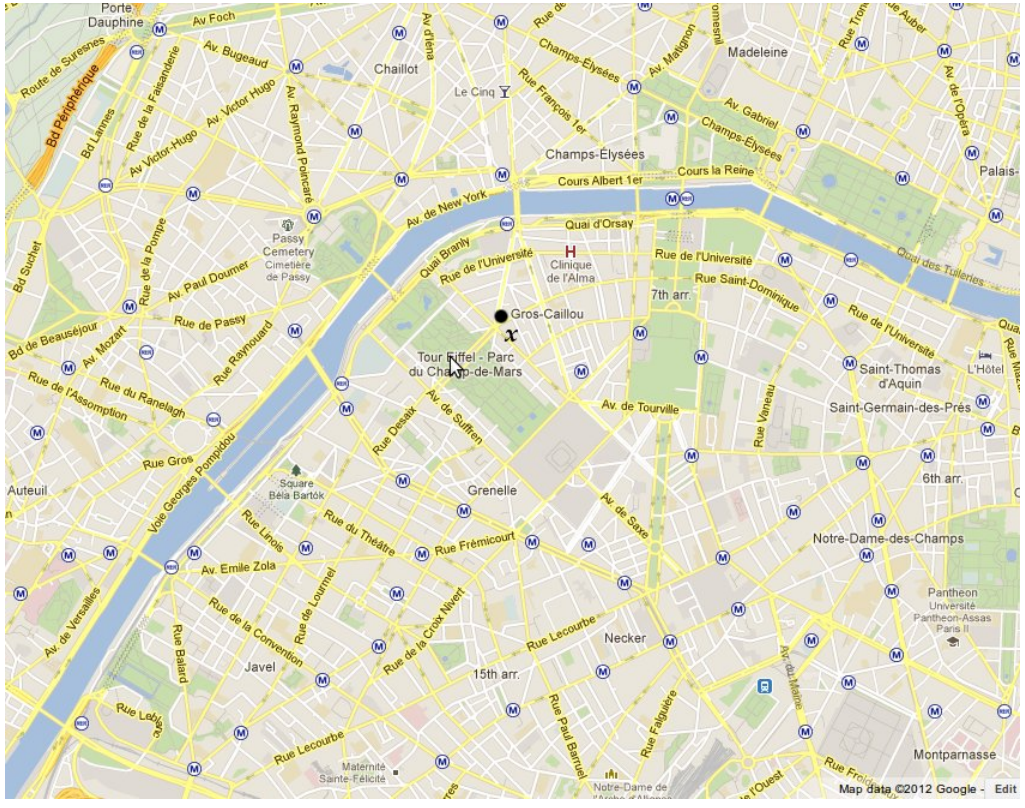
Towards a Definition

- Secrets are **locations**.
- Attacker's goal: **distinguish** location x from x′.
- The closer two locations are, the more indistinguishable they should be.



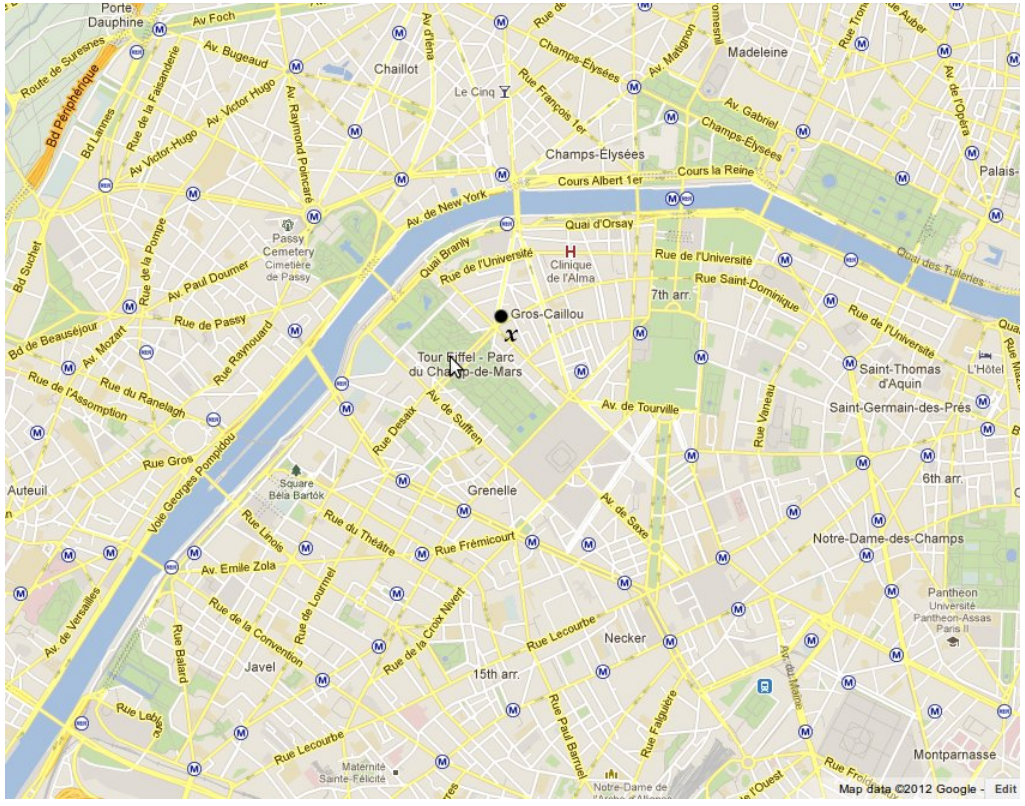
In search for a new definition

- What kind of privacy does the user expect to have?



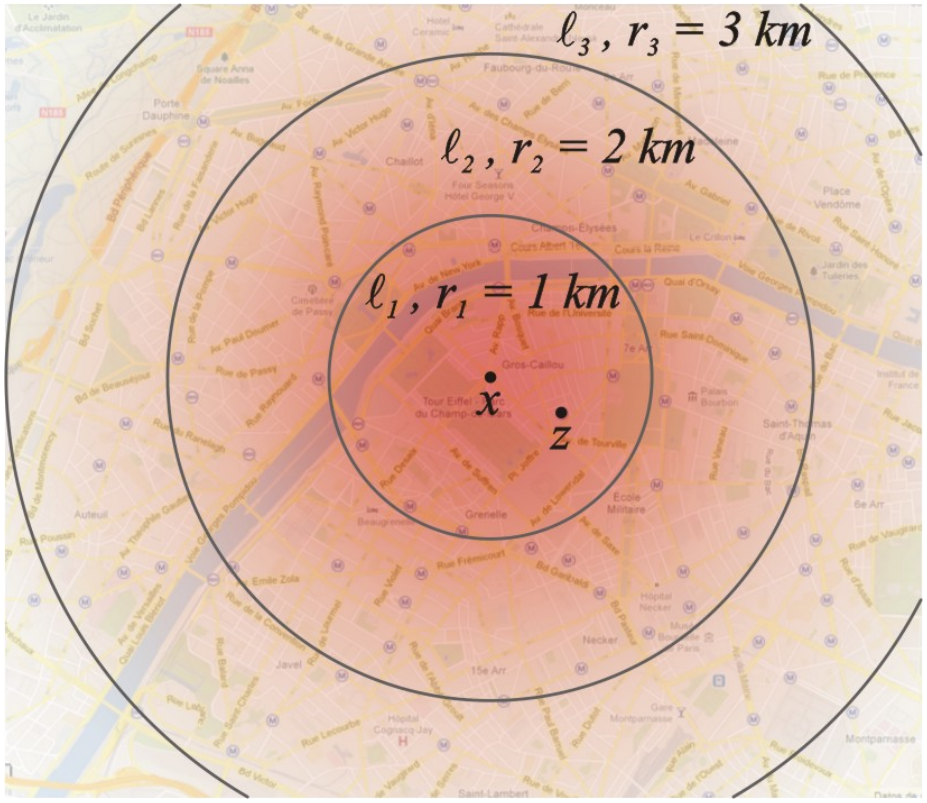
In search for a new definition

- What kind of privacy does the user expect to have?
- Privacy depends on the accuracy of detecting x



In search for a new definition

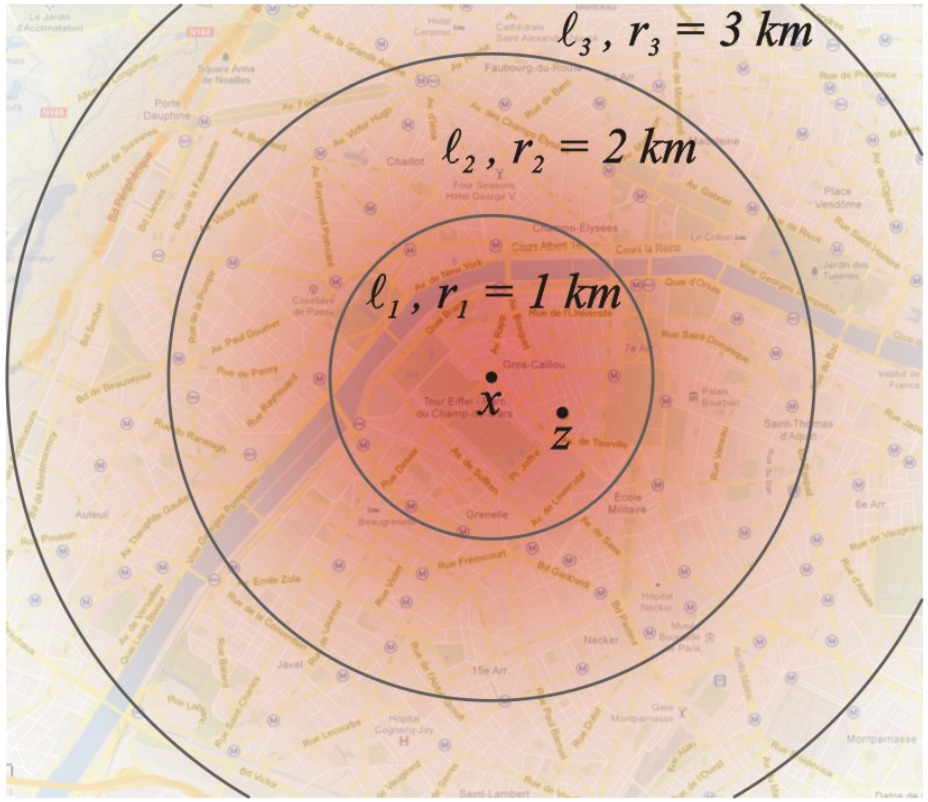
- What kind of privacy does the user expect to have?
- Privacy depends on the accuracy of detecting x
- A different privacy level l for each radius r



In search for a new definition

ϵ-geo-indistinguishability

Require privacy for any radius r with a proportional level $l(r) = \epsilon \cdot r$

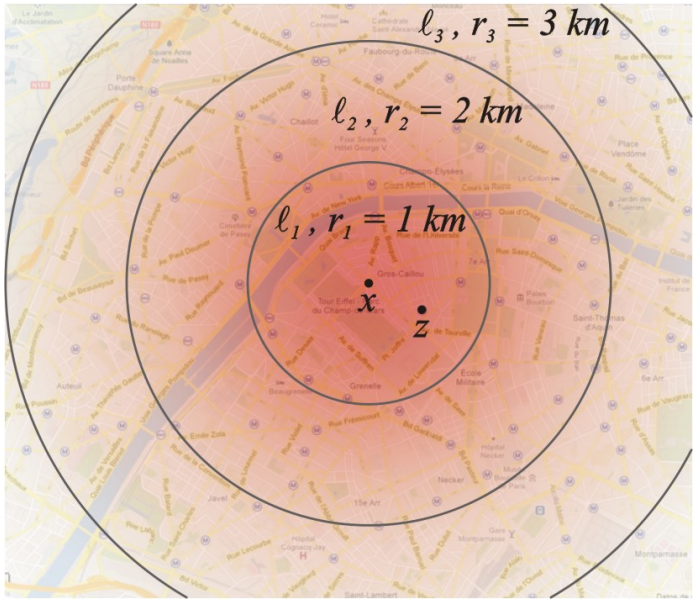


First approach for defining this notion

Intuitively we would like to require:

$$\frac{P(x|z)}{P(x'|z)} \leq e^{\epsilon r} \qquad \forall r \; \forall x, x' : d_2(x, x') \leq r$$

but this might fail because of the prior knowledge $P(x)$

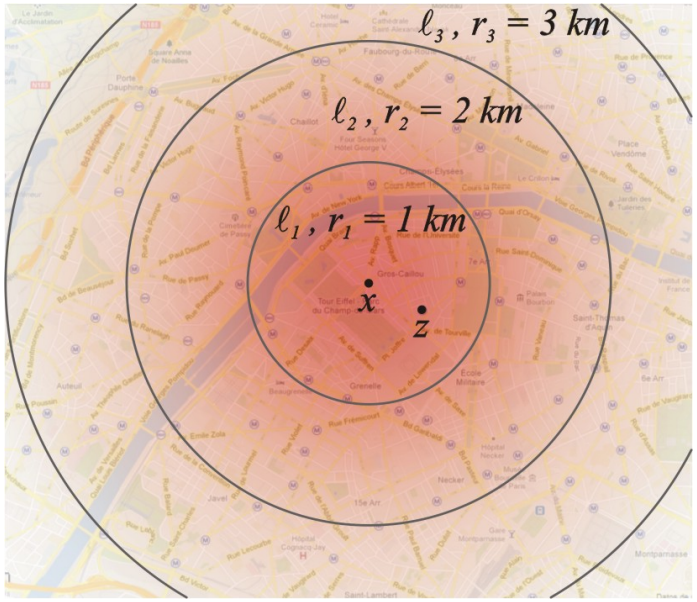


First approach for defining this notion

So we have to take it into account:

$$\frac{P(x|z)}{P(x'|z)} \leq e^{\epsilon r} \frac{P(x)}{P(x')} \qquad \forall r \; \forall x, x' : d_2(x, x') \leq r$$

are require this to hold for any prior $P(x)$

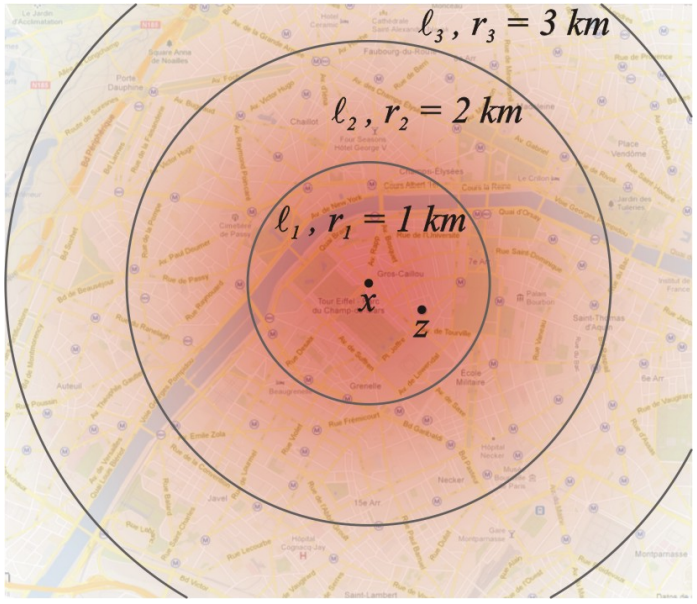


Second approach for defining this notion

Ideally we'd like the attacker's knowledge to be unaffected by z:

$$\frac{P(x|z)}{P(x)} \leq e^{\epsilon r} \qquad \forall r, x$$

but z does provide information (i.e. that the user is in Paris)

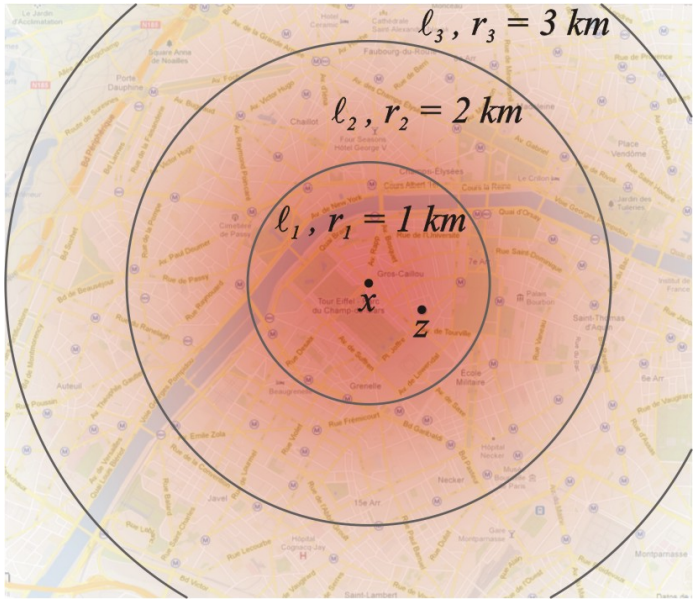


Second approach for defining this notion

So we restrict the increase in knowledge within the radius r:

$$\frac{P(x|z, B_r(x))}{P(x|B_r(x))} \leq e^{\epsilon r} \qquad \forall r, x$$

again, this should hold for any prior $P(x)$

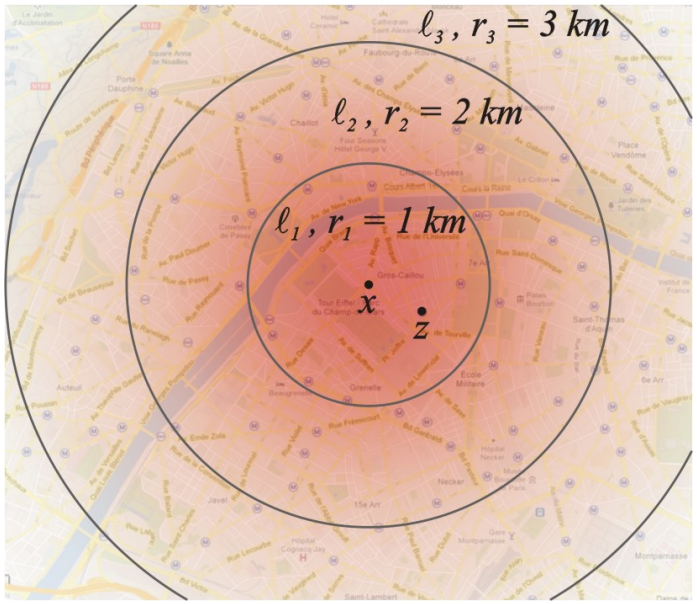


Third approach for defining this notion

Nearby points should produce similar observations:

$$\frac{K(x)(z)}{K(x')(z)} \leq e^{\epsilon r} \qquad \forall r\ \forall x, x' : d_2(x, x') \leq r$$

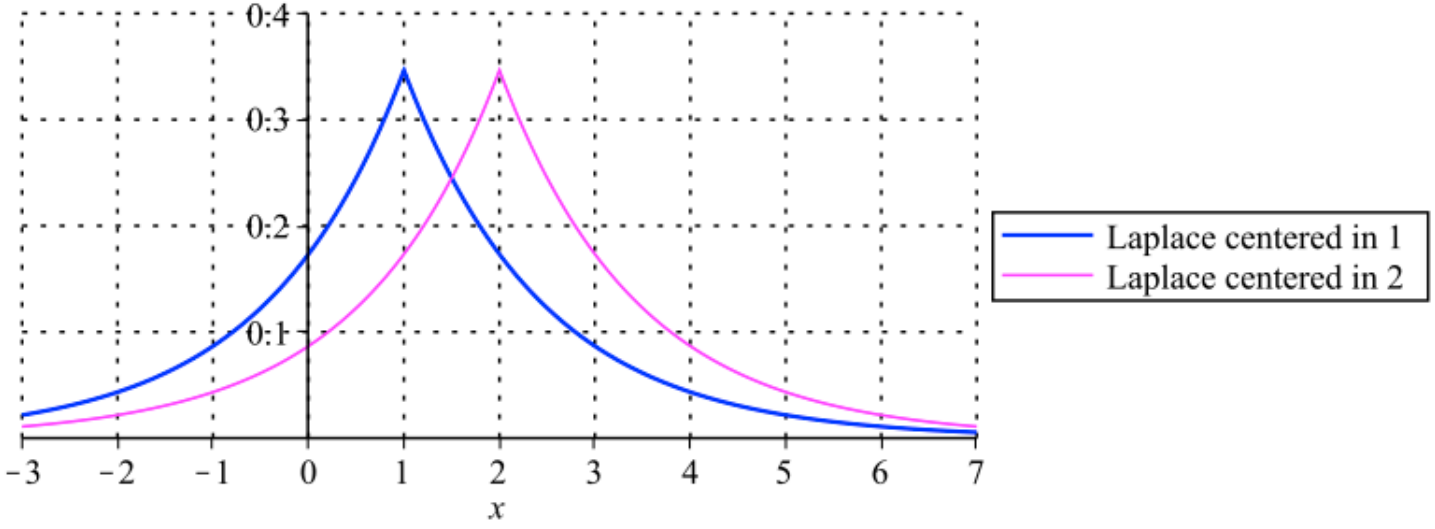
which is the same as ϵd_2-privacy.



All three formulations are equivalent

A mechanism for geo-indistinguishability

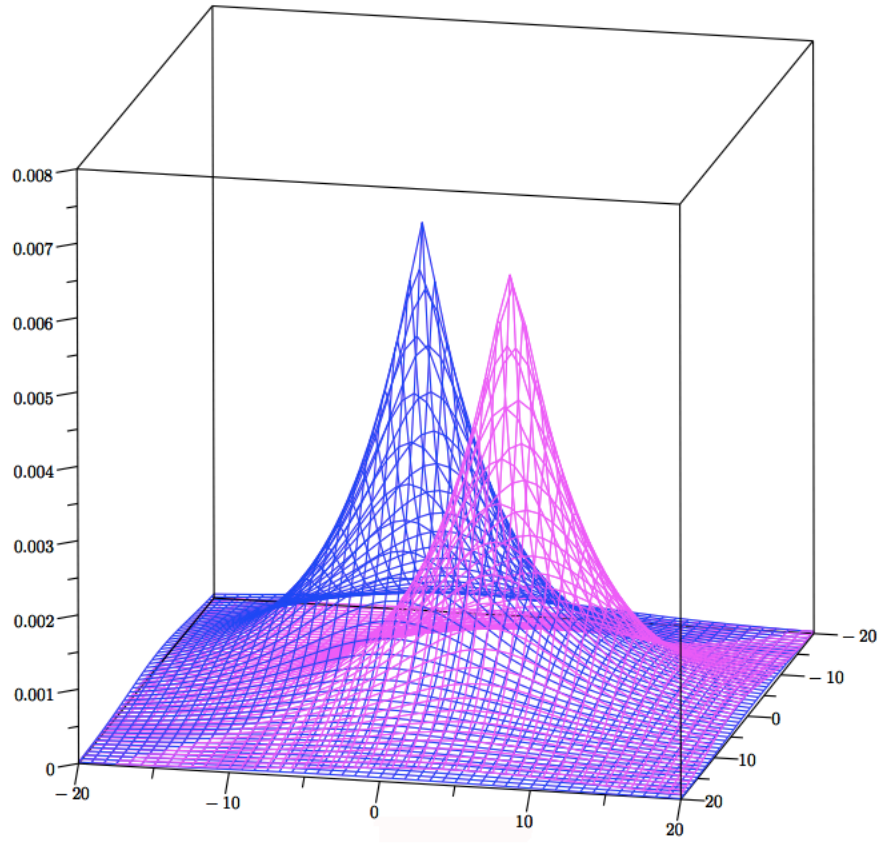
The case of one dimension:



pdf: $\frac{\epsilon}{2} e^{-\epsilon|z-x|}$

A mechanism for geo-indistinguishability

Similarly in two dimensions:

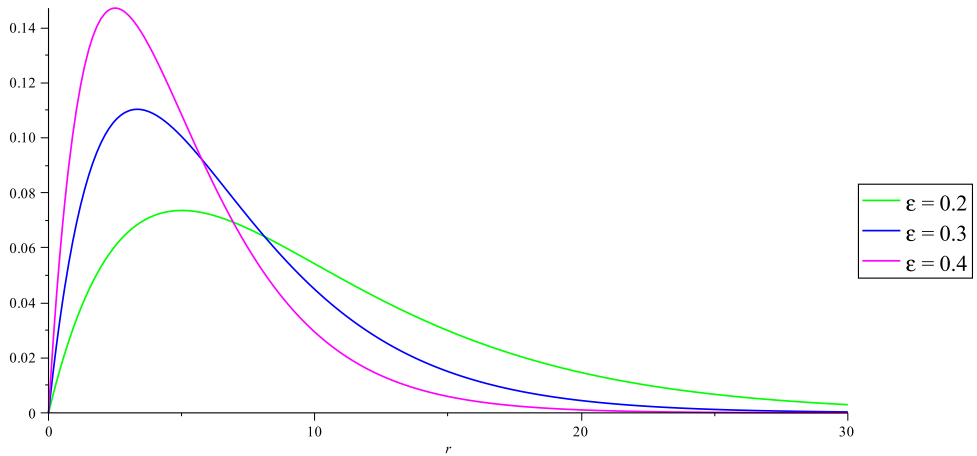


pdf: $\frac{\epsilon^2}{2\pi} e^{-\epsilon\, d_2(\vec{x},\vec{z})}$

A mechanism for geo-indistinguishability

Drawing from this distribution:

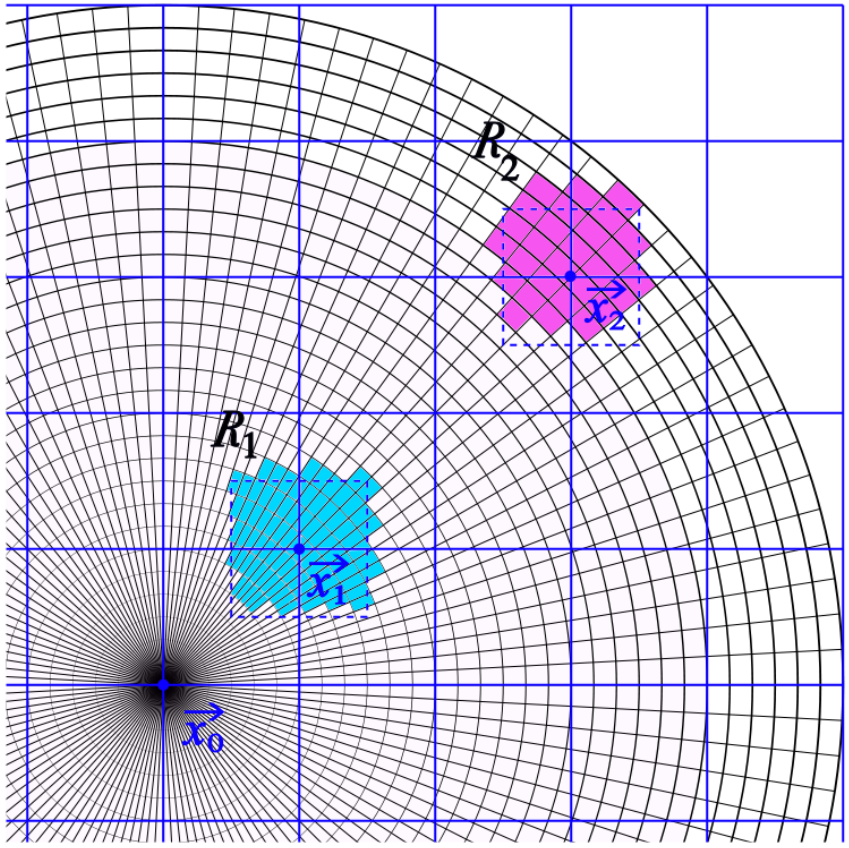
- use polar coordinates
- draw an angle θ uniformly
- draw a radius r from a gamma distribution



pdf: $\epsilon^2\, r\, e^{-\epsilon\, r}$

A mechanism for geo-indistinguishability

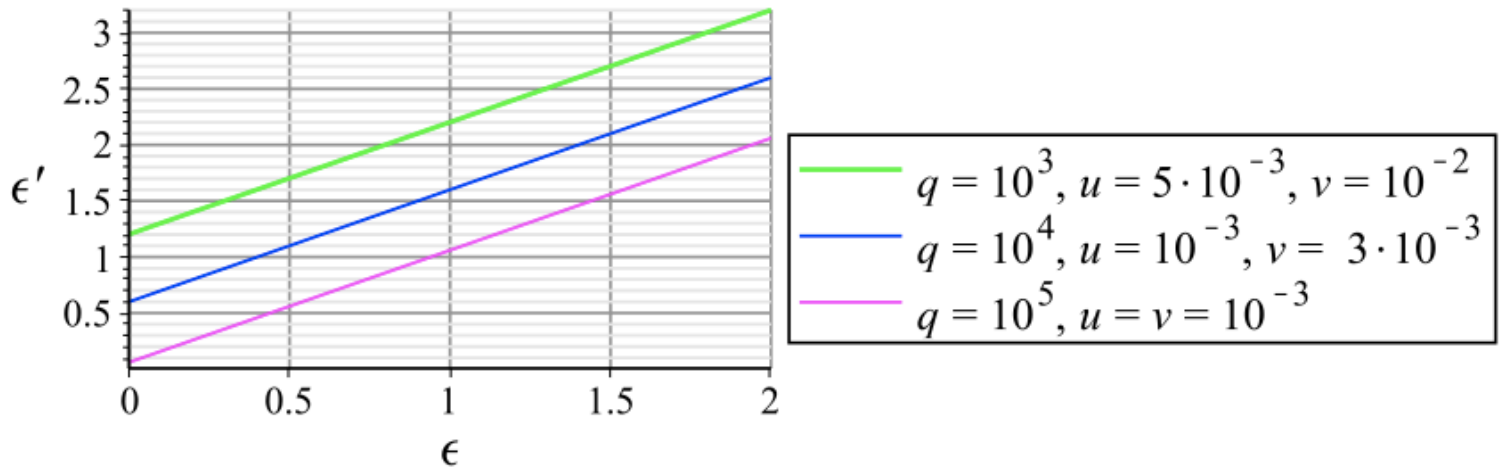
- In practice locations are discretized
- (discretely) draw r, θ, map to the closest point on the grid
- Points correspond to differently shaped areas, leading to a vioation of geo-indistinguishability



A mechanism for geo-indistinguishability

Solution: adjust ϵ to compensate for these differences

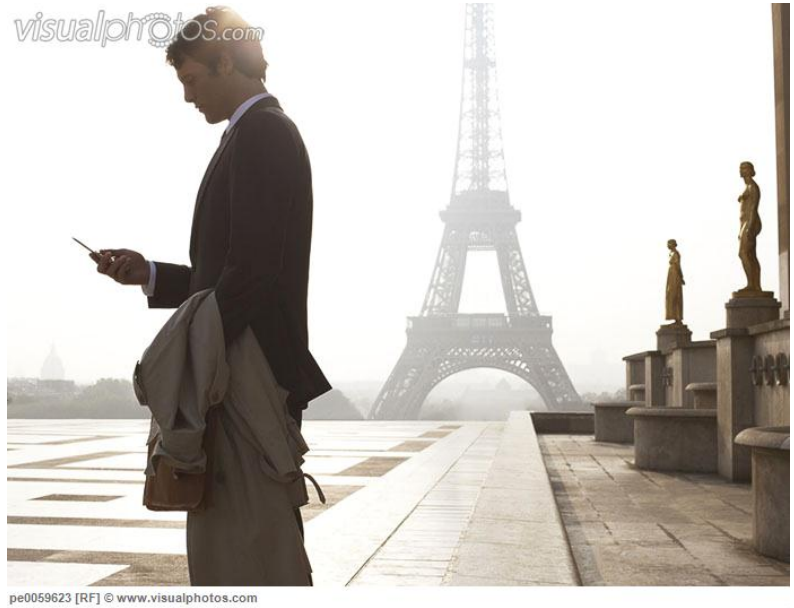
$$\epsilon' = \epsilon + \frac{1}{u} \ln \frac{q - 2 + 3\, e^{\epsilon\, v\, \sqrt{2}}}{q - 5}$$



Case study: Location-Based Services

Retrieve location-dependent information

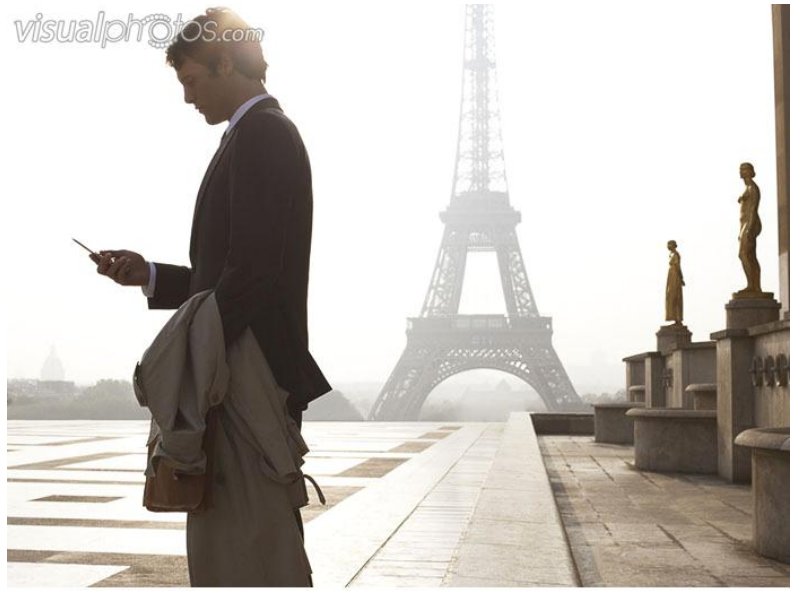
- Restaurants
- Friends
- Gas stations
- Weather
- ...



Case study: Location-Based Services

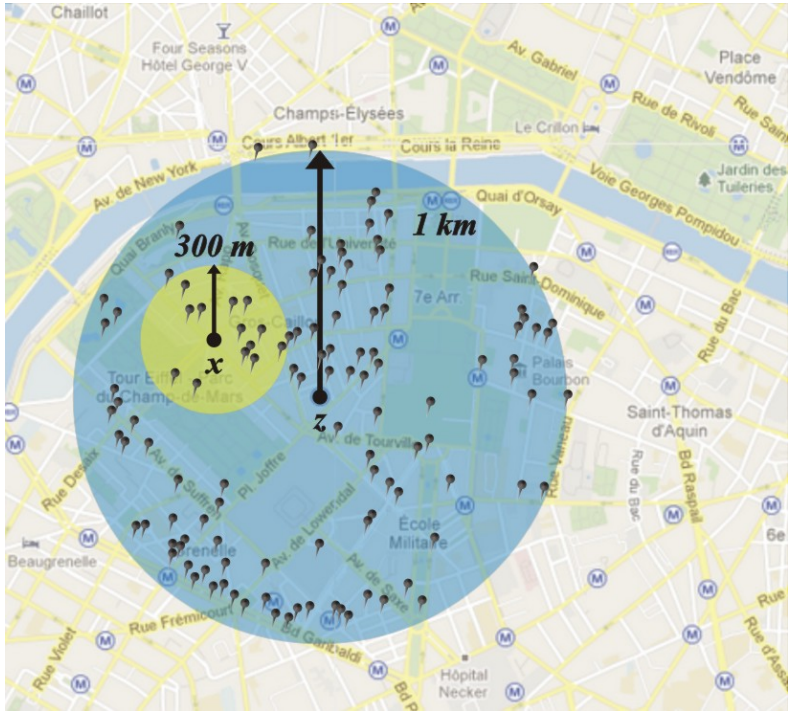
Solution:

- Add noise to the location x to obtain z
- Use z to query the provider
- Some services are insensitive to "small" perturbations (eg. weather, gas stations)
- In this case the quality of the results will not be affected

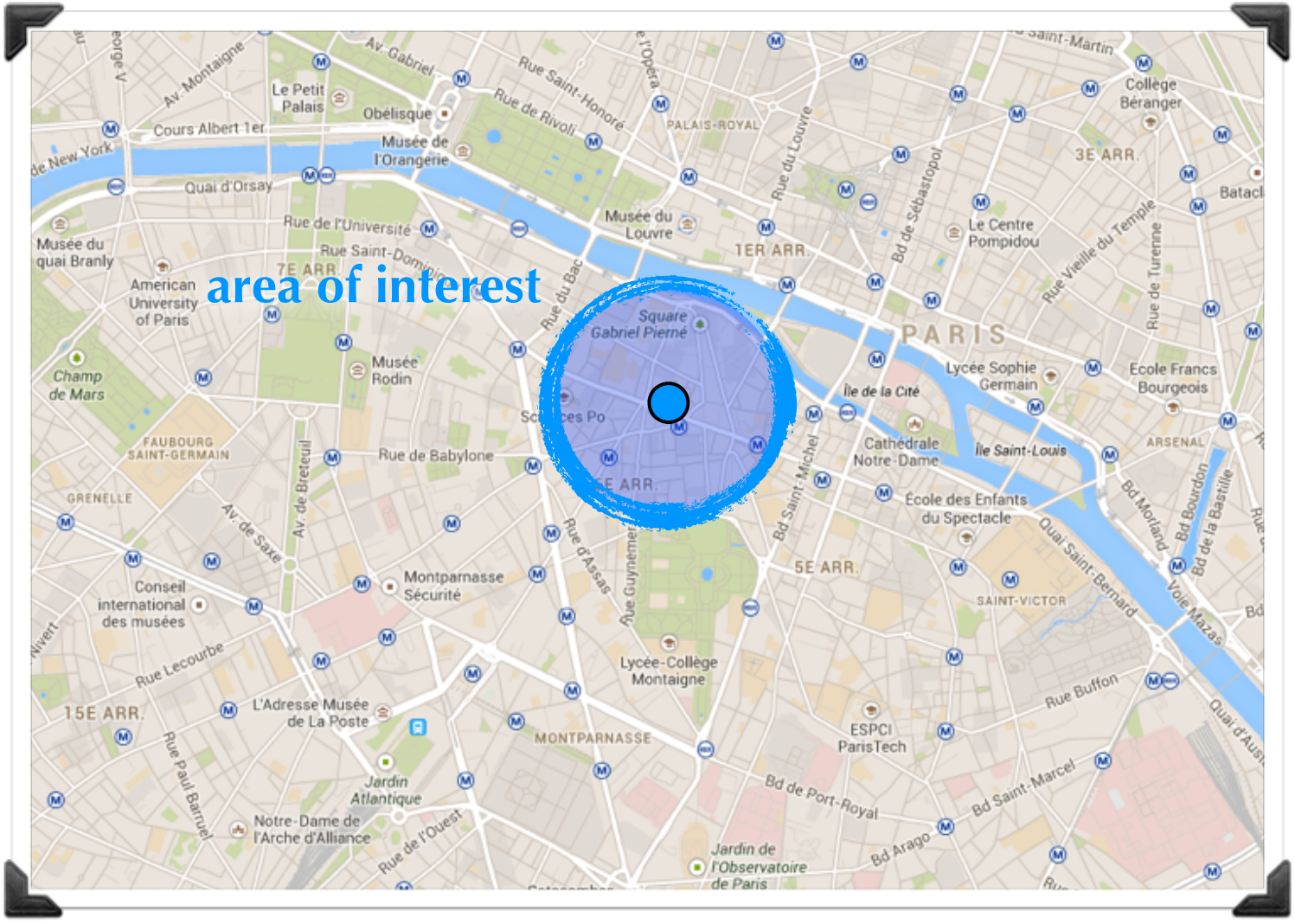


Case study: Location-Based Services

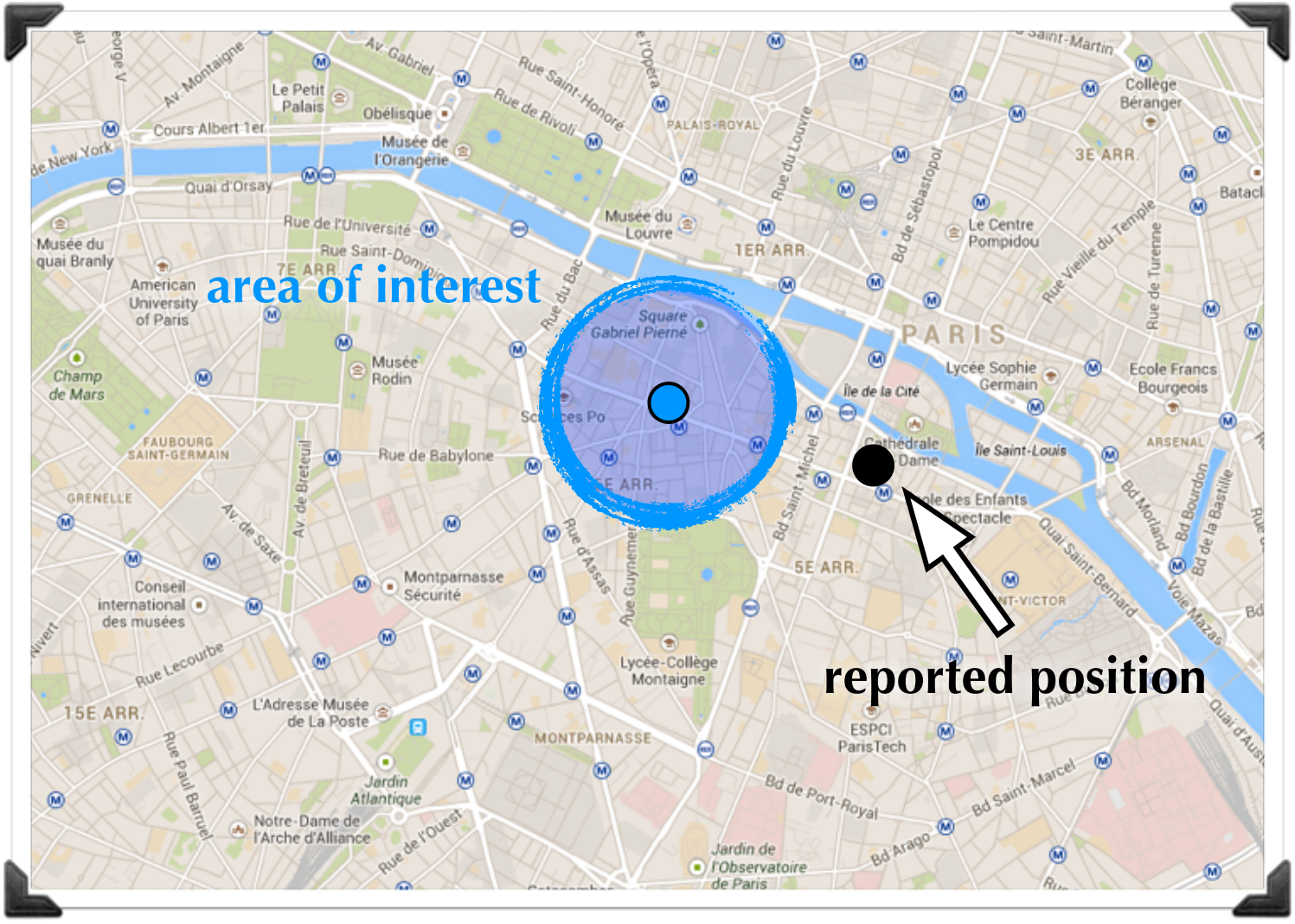
- Many LBS depend on the accuracy of the location
  eg. find restaurants within 300m from x
- In this case the query needs to be extended to a larger area
  eg. get restaurants within 1km from z
- Important: the area needs to be independent from z



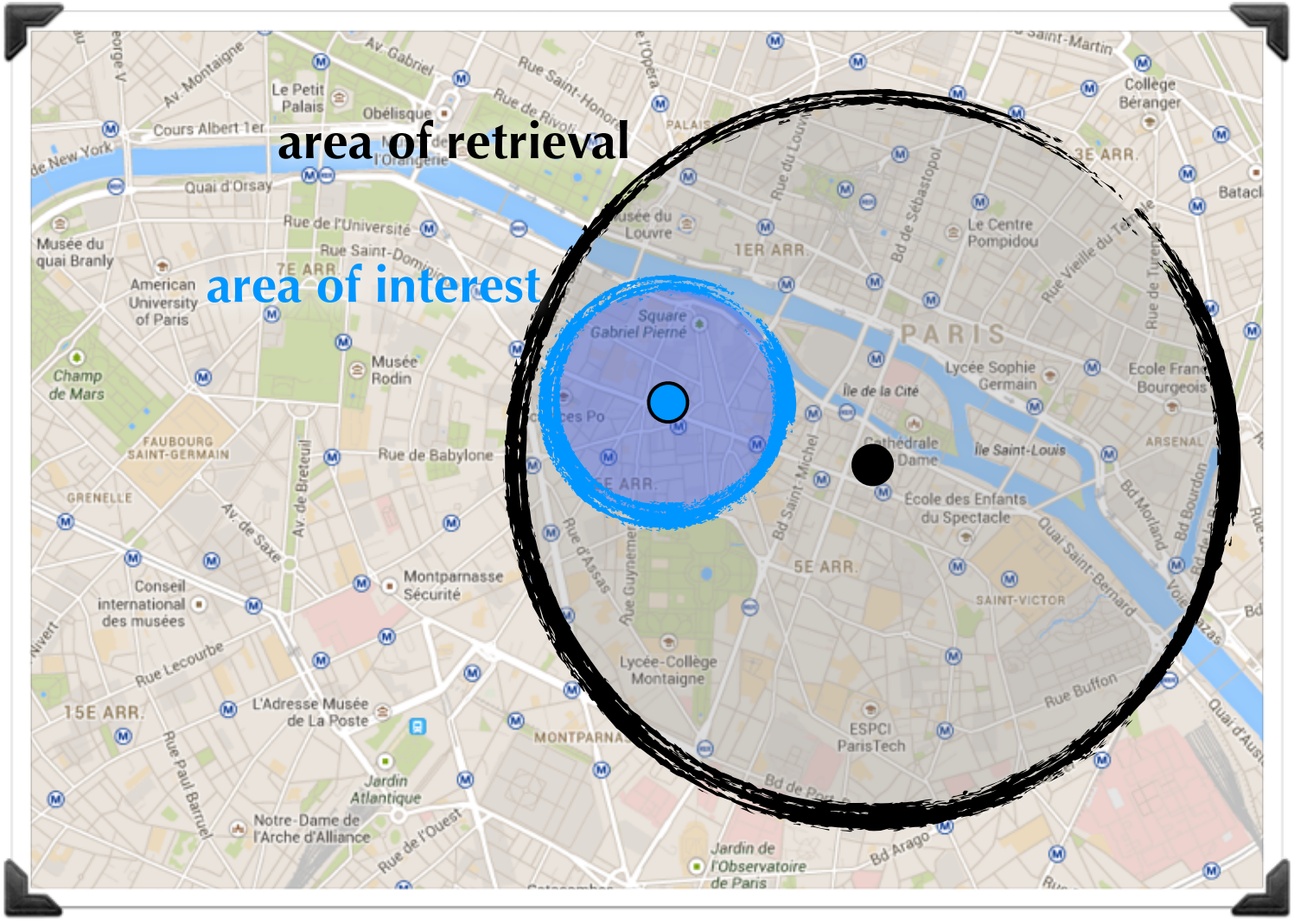
Obfuscation



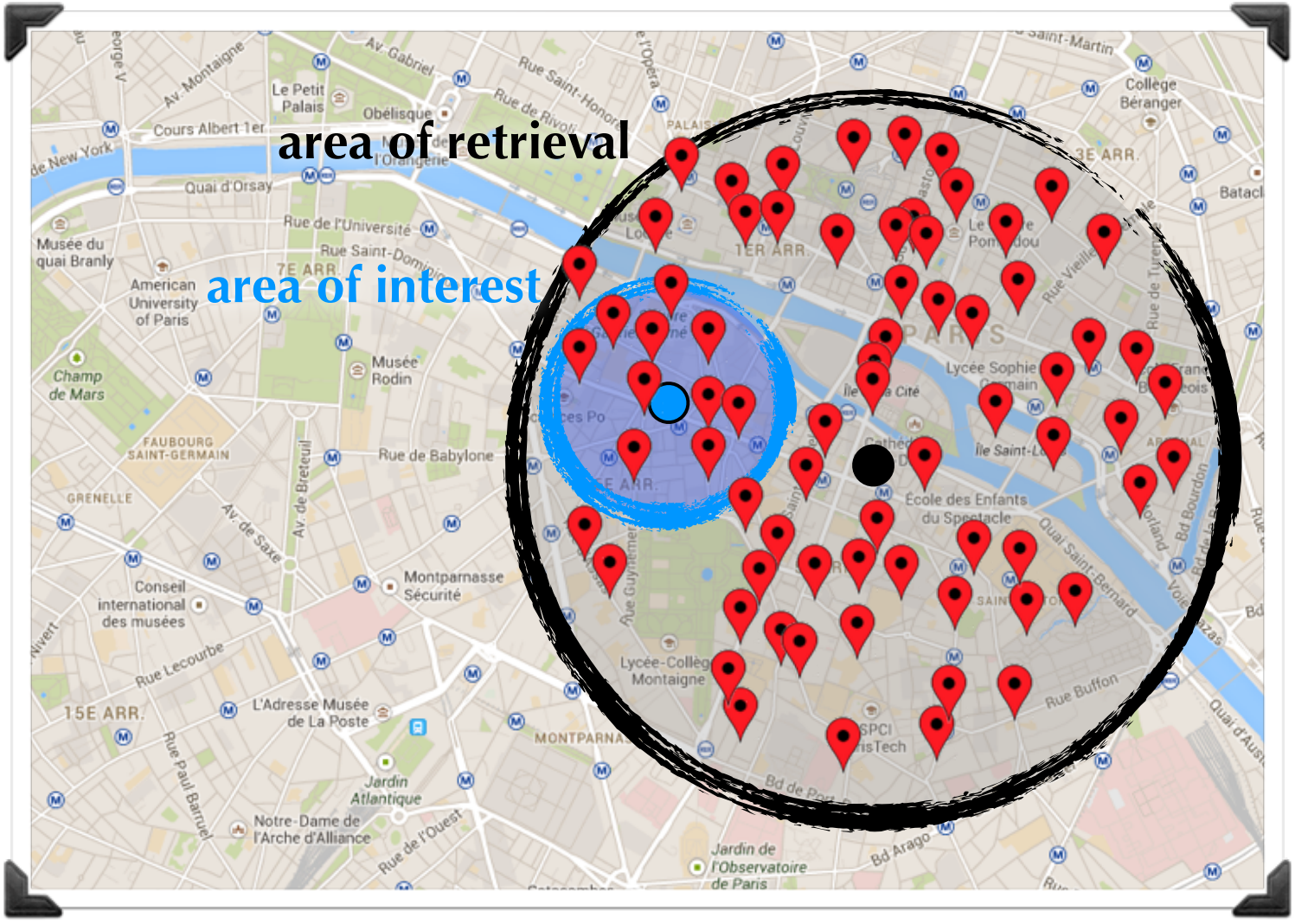
Obfuscation



Obfuscation



Obfuscation



Obfuscation

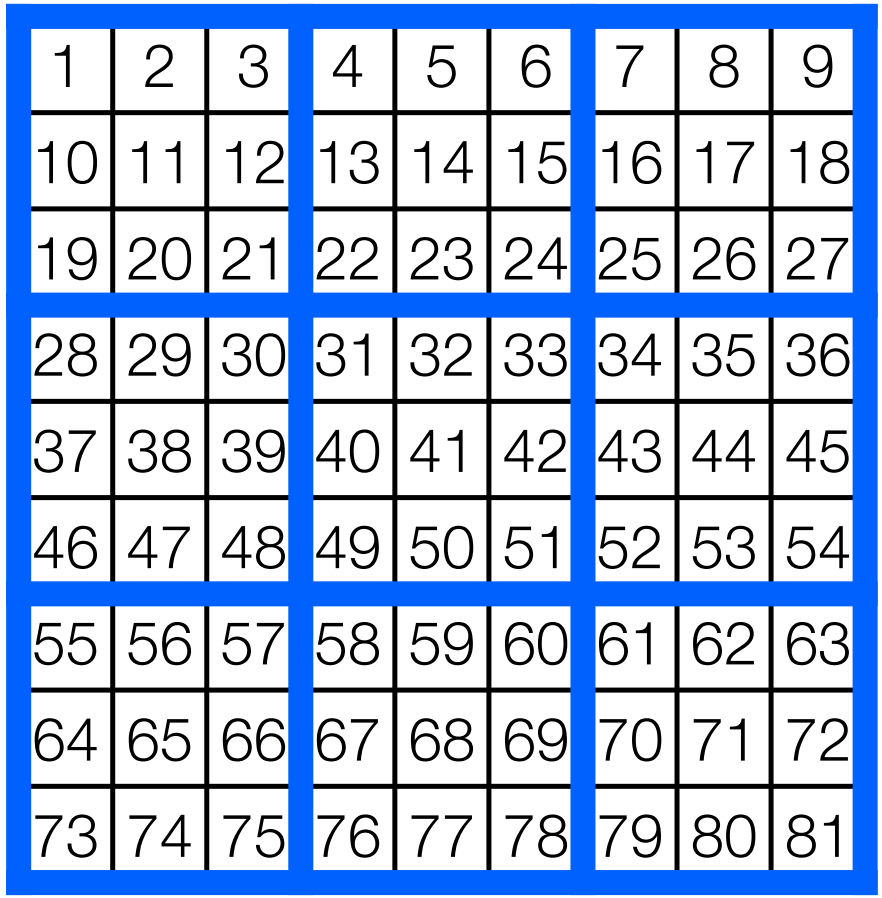
Privacy versus utility: evaluation

- 9x9 = 81 "points".
- We compare 4 mechanisms.
- Configured to the **same utility**.
- Optimal mechanism by [Shroki et al., S&P 2012] for the **corresponding prior**. Obtained by linear optimization techniques.
- Three **prior independent**:

  Planar Laplacian (discretized).

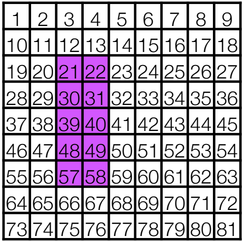
  Optimal under uniform prior.

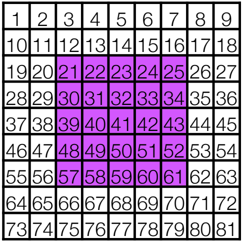
  Simple cloaking.

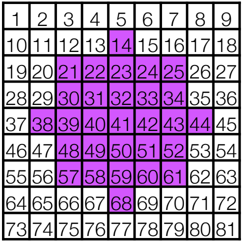


Privacy versus utility: evaluation

- We fix the utility and measured the privacy.
- Utility loss measured as the **expected distance** between the true location and the reported one [Shroki et al., S&P 2012]
- Privacy measured as the **expected error of the attacker** (using prior information) [Shroki et al., S&P 2012]
- Priors: uniform over colored regions



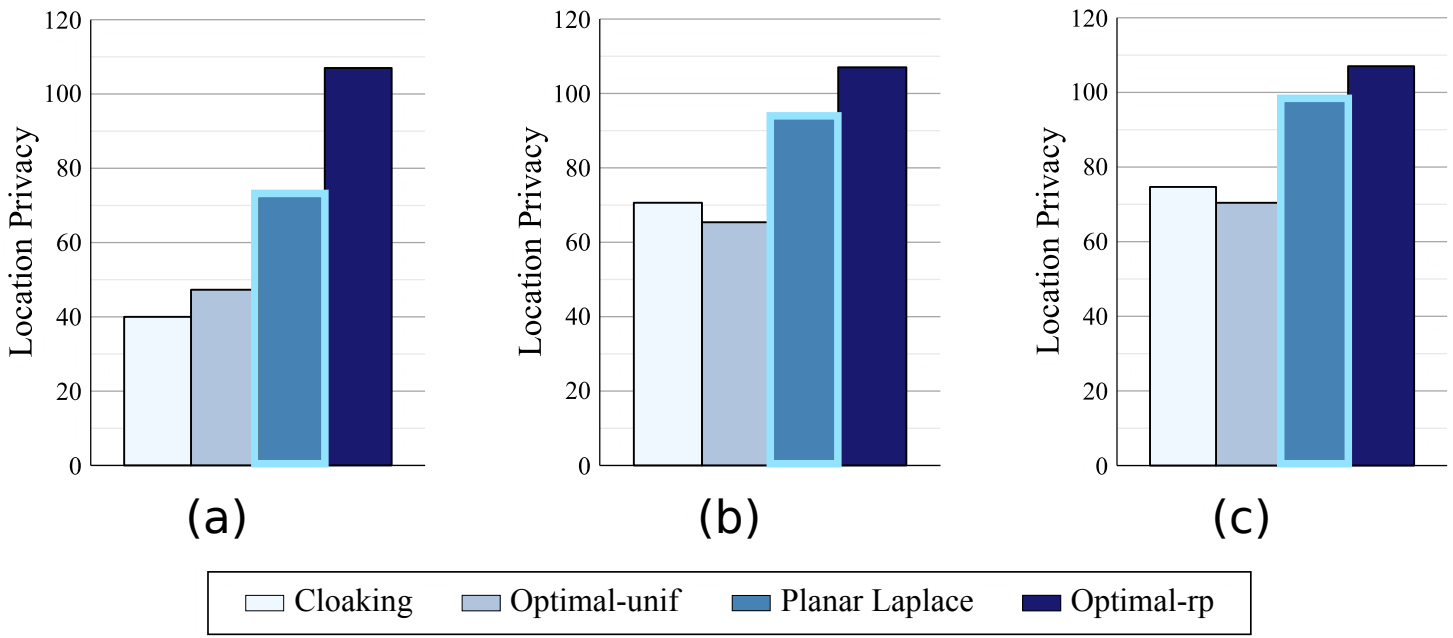




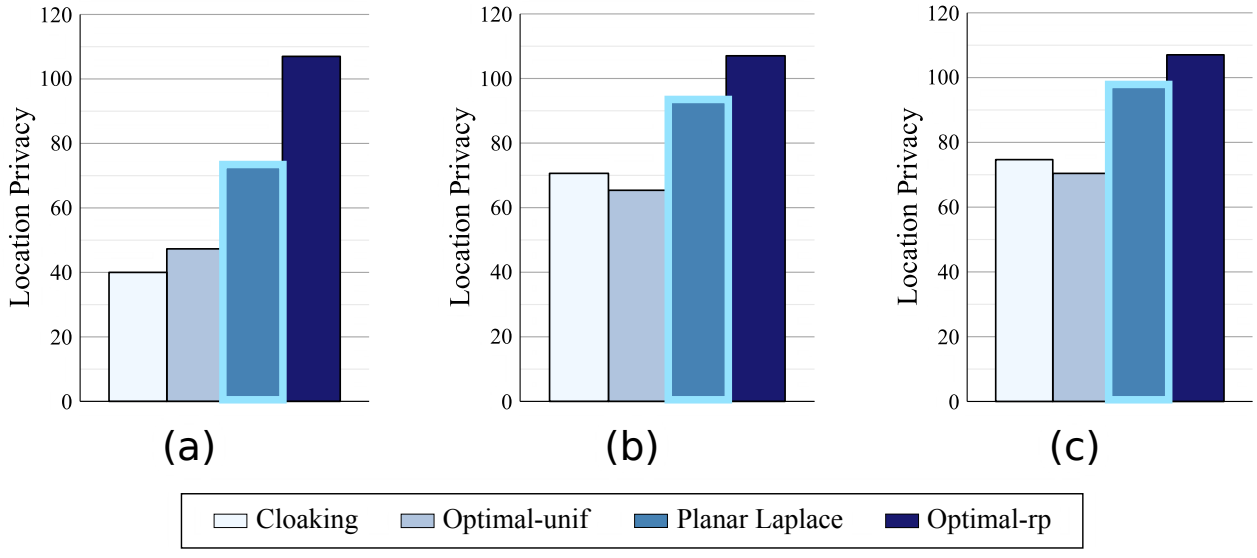
Privacy versus utility: evaluation

The four mechanisms:

- Cloaking,
- Optimal by [Shroki et al. S&P 2012] for the uniform prior
- Ours (Planar Laplacian)
- Optimal by [Shroki et al. S&P 2012] for the given prior



Privacy versus utility: evaluation



With respect to the privacy measures proposed by [Shokri et al, S&P 2012], our mechanism performs better than the other mechanisms proposed in the literature which are independent from the prior (and therefore from the adversary)

The only mechanism that outperforms ours is the optimal by [Shokri et al, S&P 2012] for the given prior, but that mechanism is adversary-dependent

Tool: "Location Guard"

http://www.lix.polytechnique.fr/~kostas/software.html

About 50,000 active users to date