Foundations of Privacy

Lecture 3

Differential Privacy

- **Definition (Differential Privacy, discrete case)** $\mathcal{K}$ is ε-differentially private if for every pair of databases $x_1, x_2 \in \mathcal{X}$ such that $x_1 \sim x_2$ (i.e., x_1 and x_2 differ for only one record), and for every reported answer $z \in Z$, 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.

- **Definition (Differential Privacy, general case)** $\mathcal{K}$ 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 set $\mathcal{S} \subseteq Z$, we have:

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

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.

- ε-differential privacy is equivalent to the following property (we consider here 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 possible distributions,

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

  and

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

  Namely: the reported answer does not affect significantly the probabilistic knowledge of the value of i, with respect to the 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.

Properties of DP

- The degree of privacy is determined by ε. The smaller ε is, the more privacy we have. ε is non-negative by definition, thus the minimal possible value is 0, which gives total indistinguishability because $\varepsilon^0 = 1$ means that a given answer can be obtained with the same probability from any database

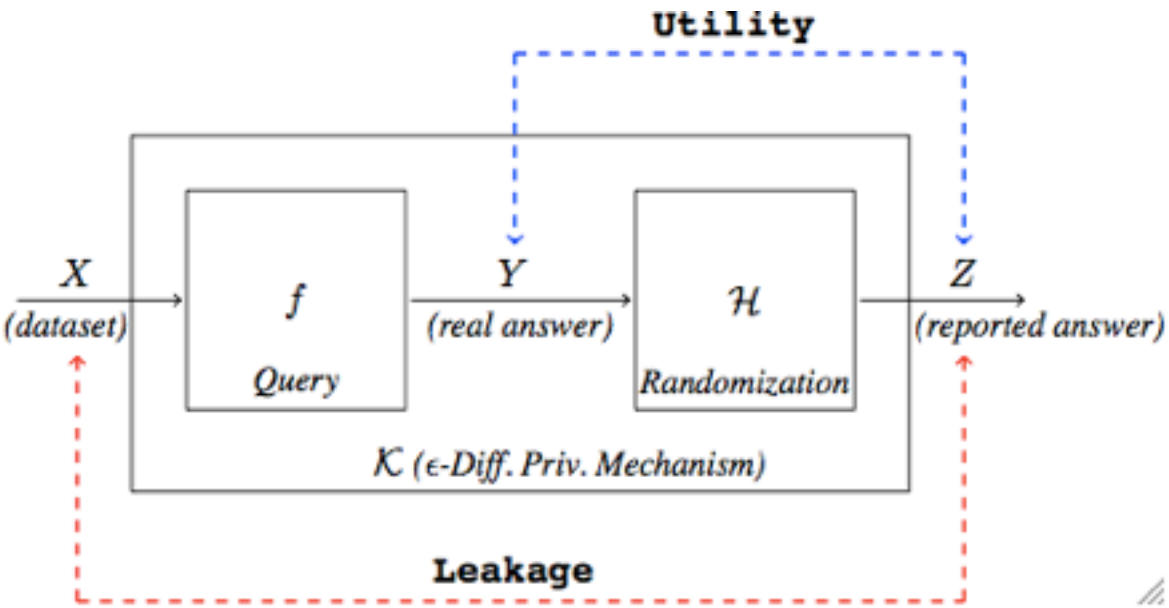
- DP does not depend on the prior distribution. This means that a mechanism satisfying DP will be DP independently from the knowledge of the adversary.

- DP is compositional: combining the answers of two mechanisms which are respectively ε_1 and ε_2 differentially private, yields a mechanism that is $(\varepsilon_1 + \varepsilon_2)$-differentially private.

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



- 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 way to obtain an oblivious ε-DP mechanism: adding Laplacian noise

- Query $f : \mathcal{X} \to \mathcal{Y}$.
- Add **Laplacian noise to the query:** If the exact answer is y, report answer z, with a probability density function defined as:

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

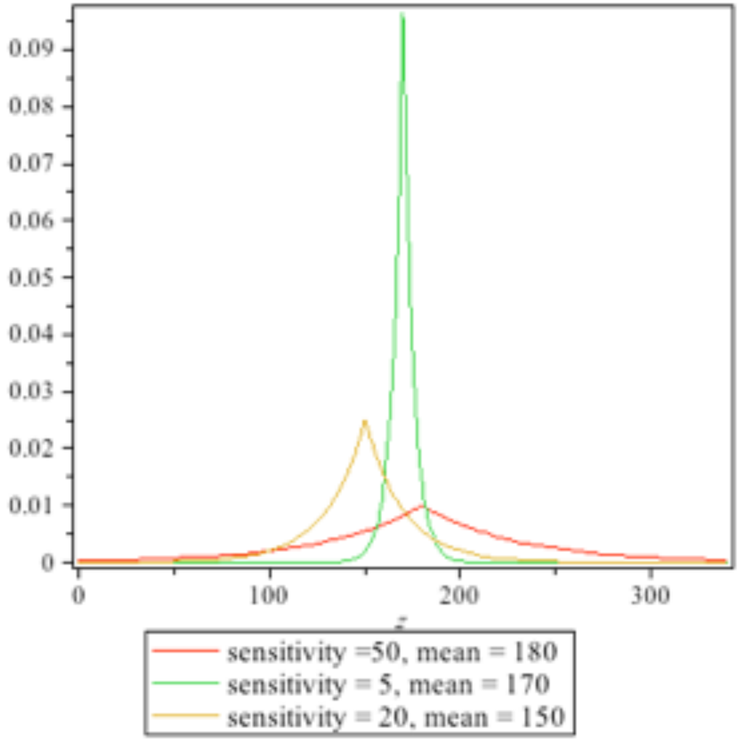
where Δf is the *sensitivity* of f:

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

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

and c is a normalization factor:

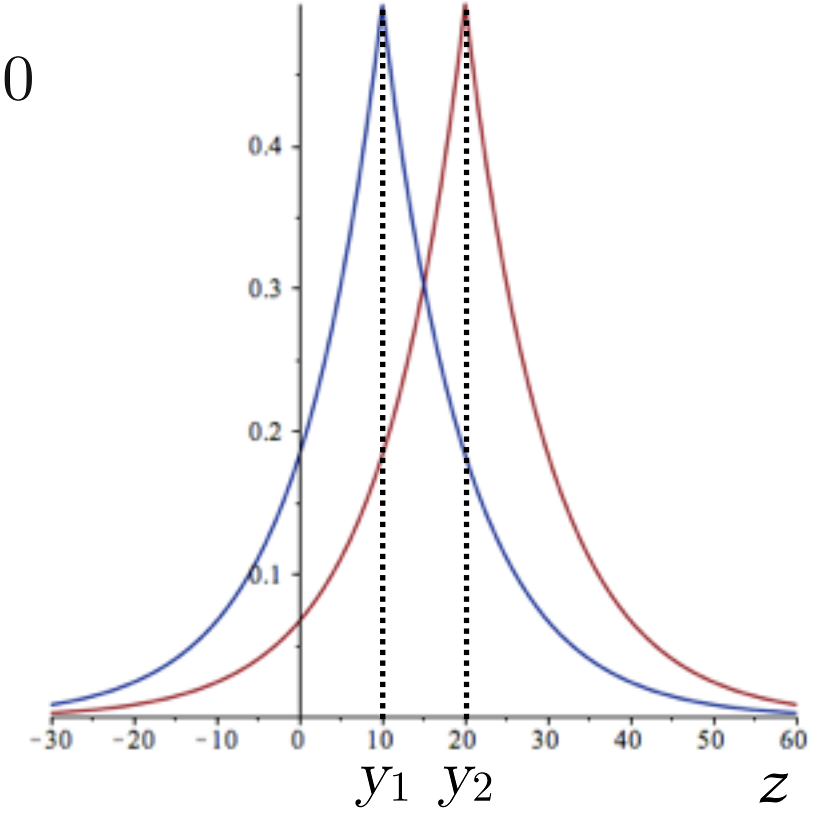
$$c = \frac{\varepsilon}{2\,\Delta f}$$



Example

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

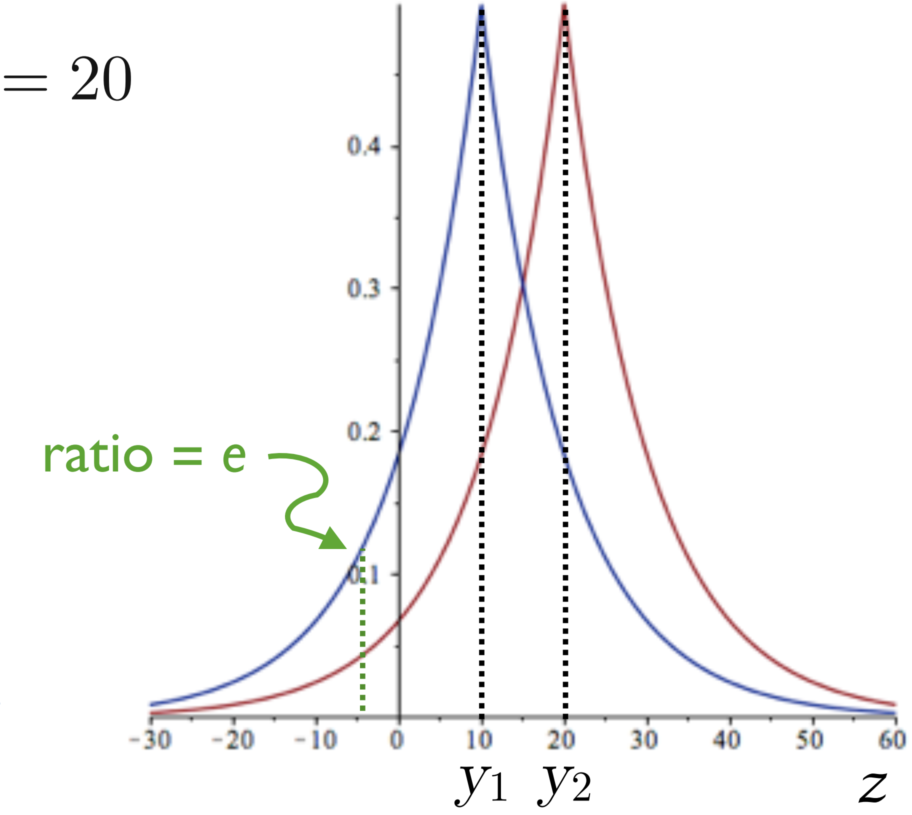
  Then:
- $dP_{y_1} = \frac{1}{2 \cdot 10} e^{\frac{|z-10|}{10}}$
- $dP_{y_2} = \frac{1}{2 \cdot 10} e^{\frac{|z-20|}{10}}$



Example

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

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



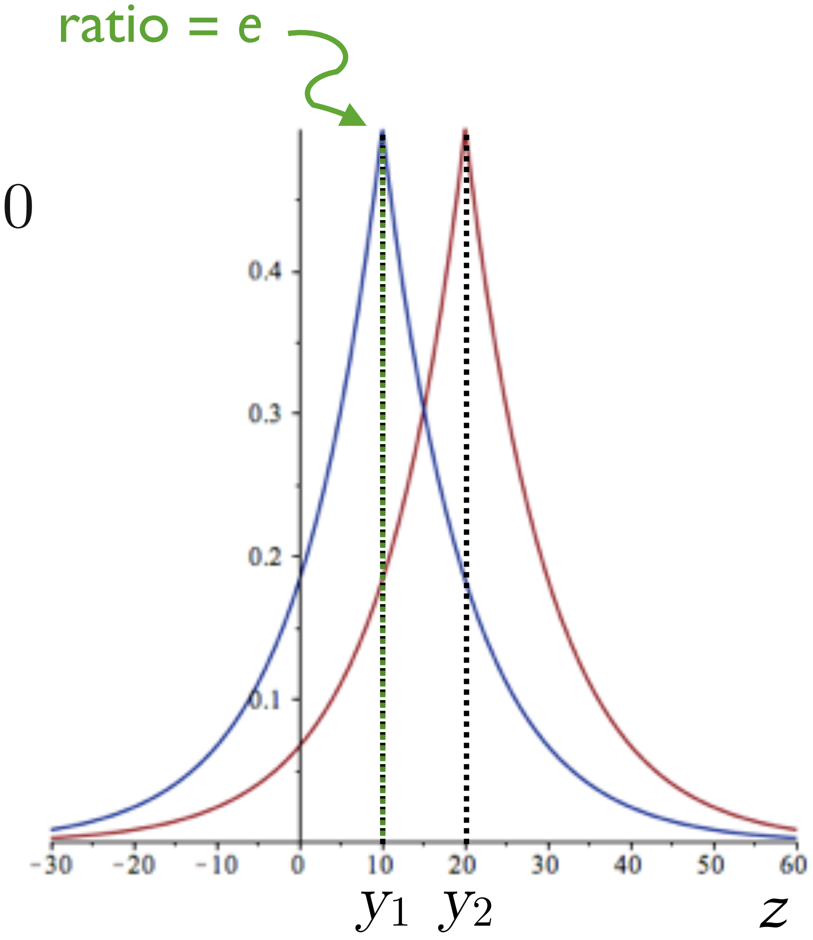
The ratio between these distribution is

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

Example

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

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


ratio = e

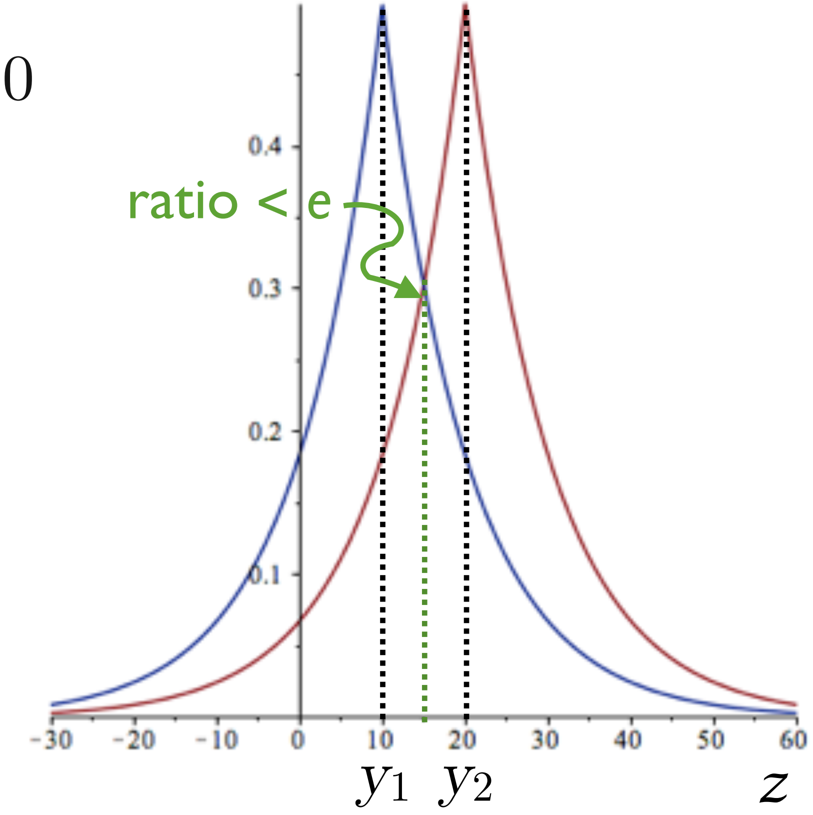
The ratio between these distribution is

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

Example

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

  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

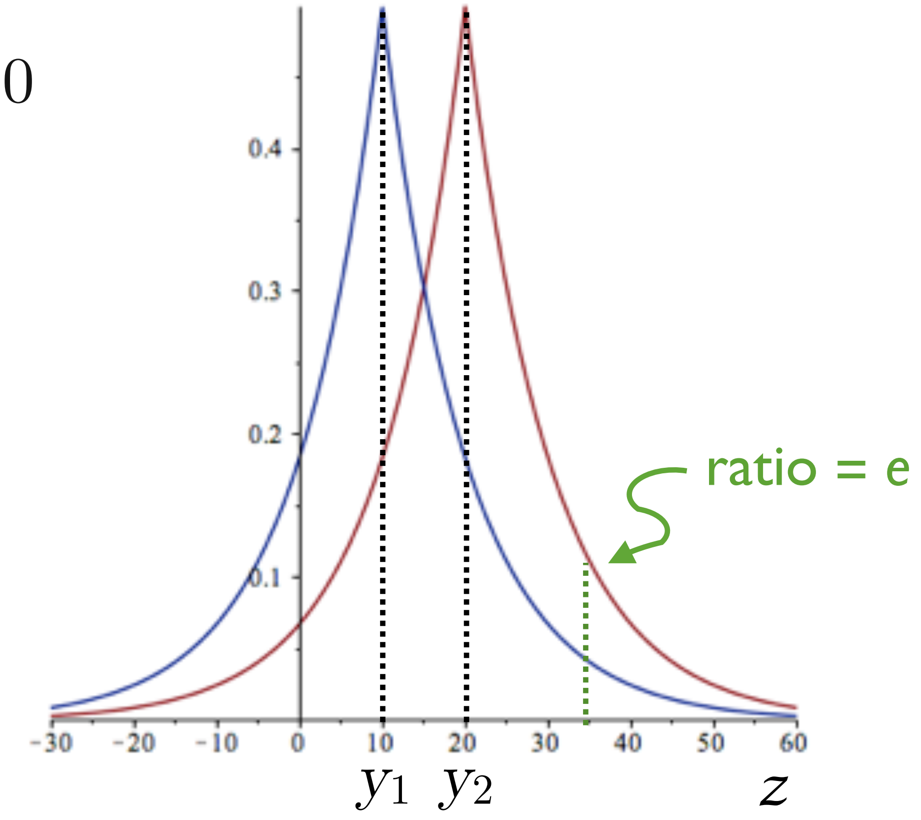
- $= e^{\varepsilon}$ outside the interval $[y_1, y_2]$
- $\leq e^{\varepsilon}$ inside the interval $[y_1, y_2]$

Example

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

  Then:

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



ratio = e

The ratio between these distribution is

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

The geometric mechanism

- The geometric mechanism is a sort of discrete Laplacian.
- Assume that $\mathcal{Y}$ and $\mathcal{Z}$ are sets of integers. In the geometric mechanism, the probability distribution of the noise is:

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

where c is a normalization factor, defined so to obtain a probability distribution. It turns out that

$$c = \frac{1-\alpha}{1+\alpha} \qquad \text{where} \qquad \alpha = e^{-\frac{\varepsilon}{\Delta f}}$$

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

Counting Queries

- A counting query is a query of the form:
  How many individuals (tuples) in the database satisfy the property $\mathcal{P}$?

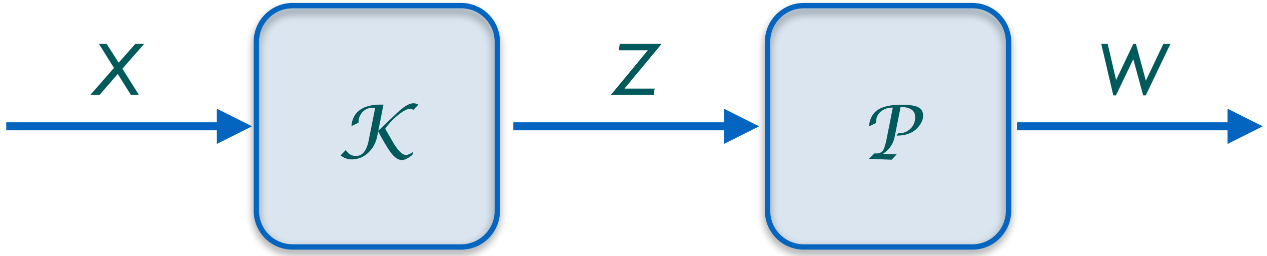
- The sensitivity of a counting query is 1

Plan of the lecture

- Postprocessing
- Truncation
- Revision of the exercises
- The utility of a mechanism
- Trade-off between utility and privacy
- Optimal and universally optimal mechanisms
- Existence and non-existence of u.o. mechanisms
- Examples and exercises

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 typically applied to a geometric mechanism.
- If the true answer is in the interval [0,n], truncation remaps all the elements smaller than 0 into 0, and all the elements greater than n into n.
- Because of the above theorem, truncation does not decrease the level of privacy.

Exercises

1. Define the noise probability distribution for the geometric mechanism for a counting query when $\mathcal{Y}$ is the interval [0,n].

   An example of a counting query: "How many people in the database are affected by the disease?"

2. Define the truncated geometric mechanism for a counting query when $\mathcal{Y}$ and $\mathcal{Z}$ are the the interval [0,n].

3. Prove that the laplacian mechanism is ε-differentially private.

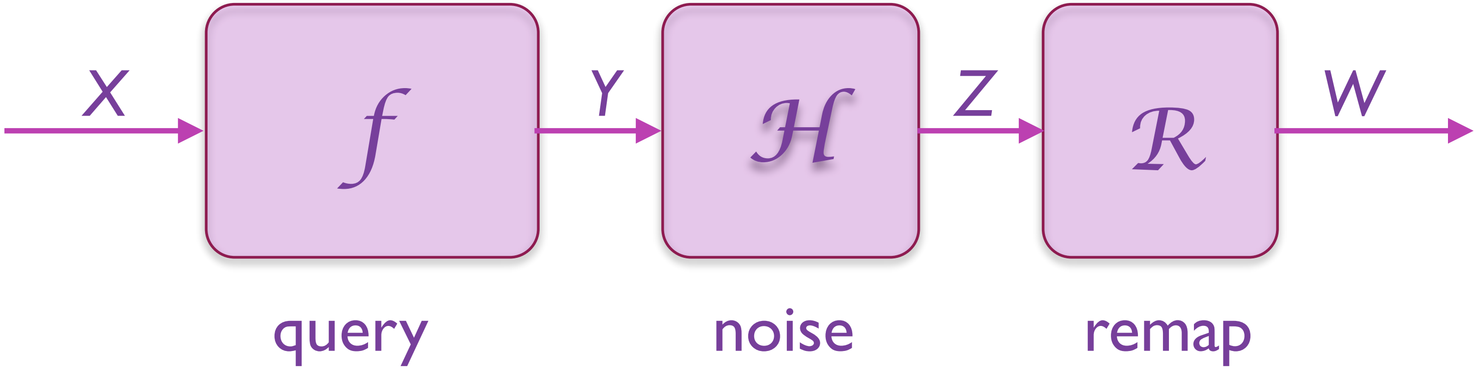
4. John knows that Sue is checking in in a hospital specialized in a certain disease. The hospital keeps a DB of the patients, containing various information including the weight, the age, and the disease status. One can ask statistical query on the DB such as "average height of people with the disease" and counting queries, and they are not sanitized. Find a sequence of queries that John can ask in order to figure out, with a large probability of success, whether Sue has the disease of not.

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

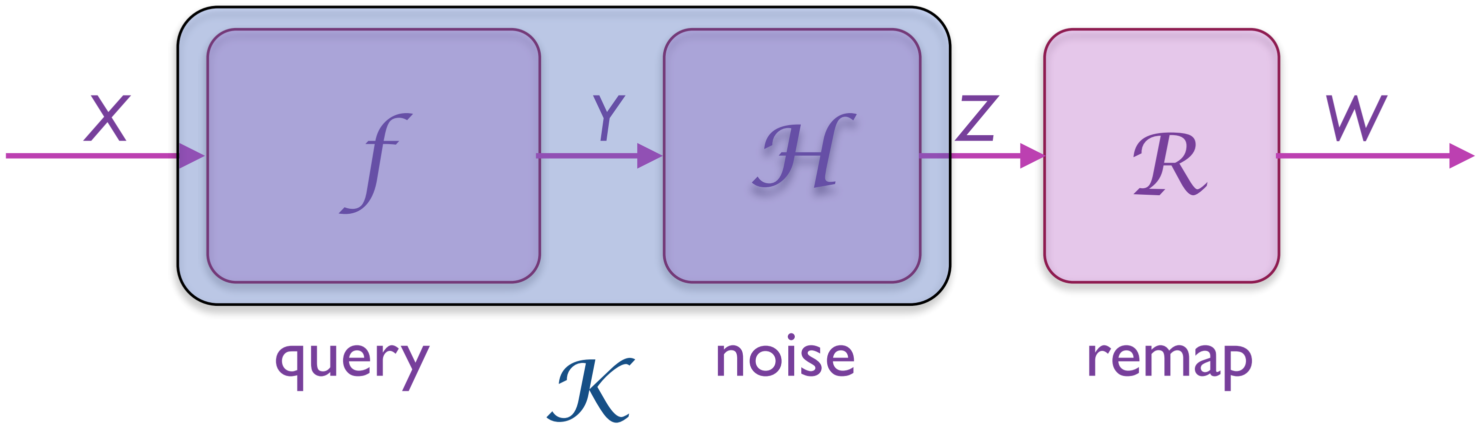
- When a user sees the reported value z of the mechanism, he may take z as it is, or, based on his prior knowledge, he may guess another value w. We say that the user remaps z into w.
  Summarizing, we have:

- $\mathcal{X}$, 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

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

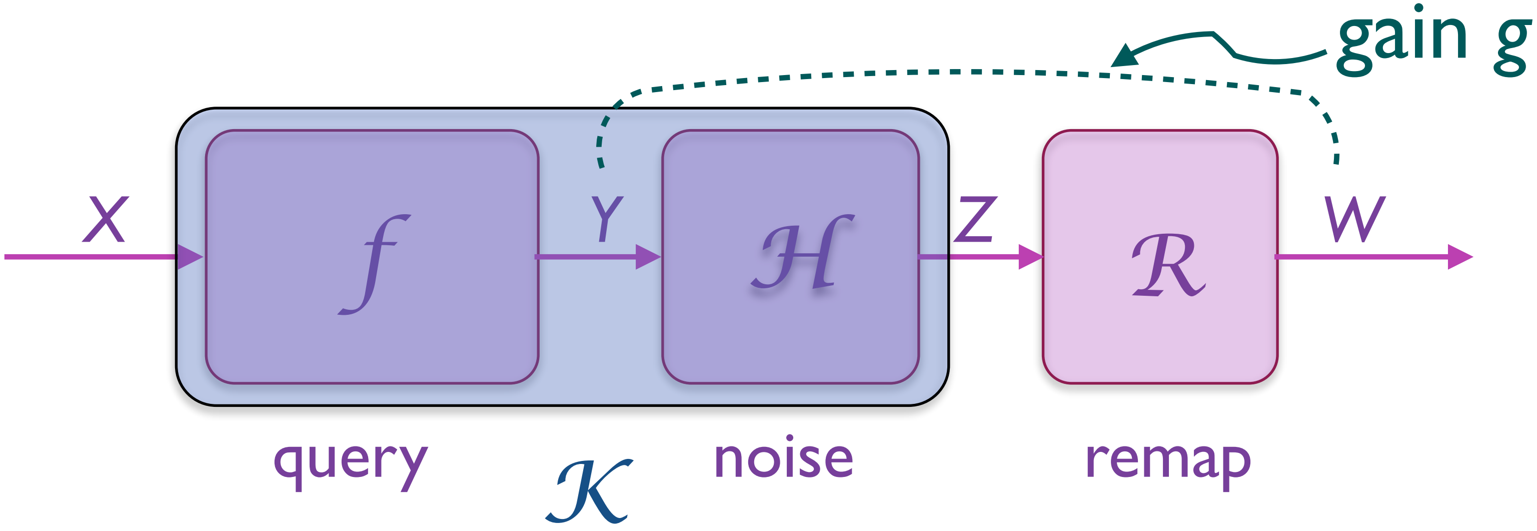
that represents the usefulness of the guess w when the true answer is y.

- 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*.

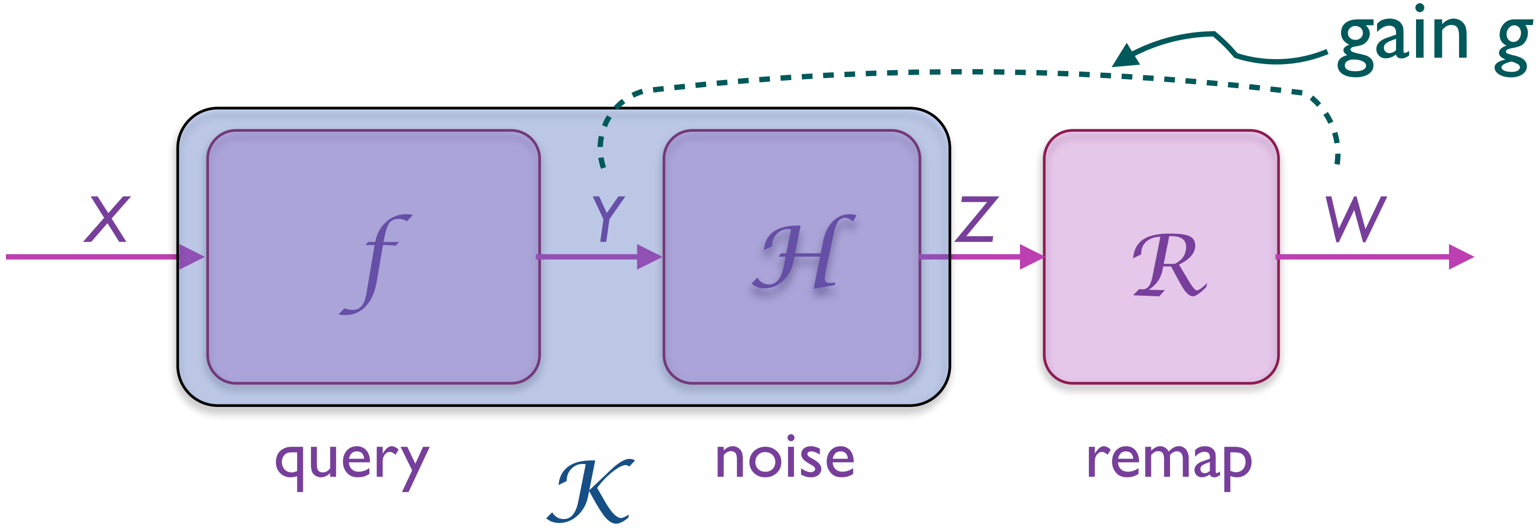
Utility

- 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:

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

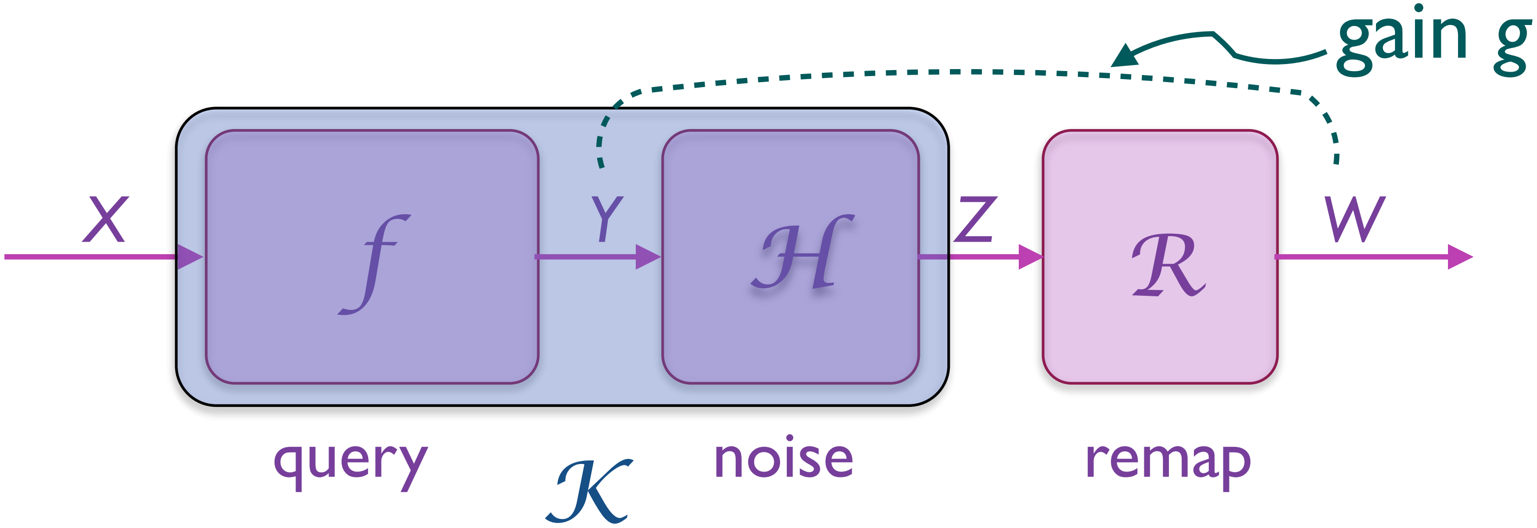


Schema for an oblivious mechanism. In a non-oblivious one *Z* depend also on *X*.

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:

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



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

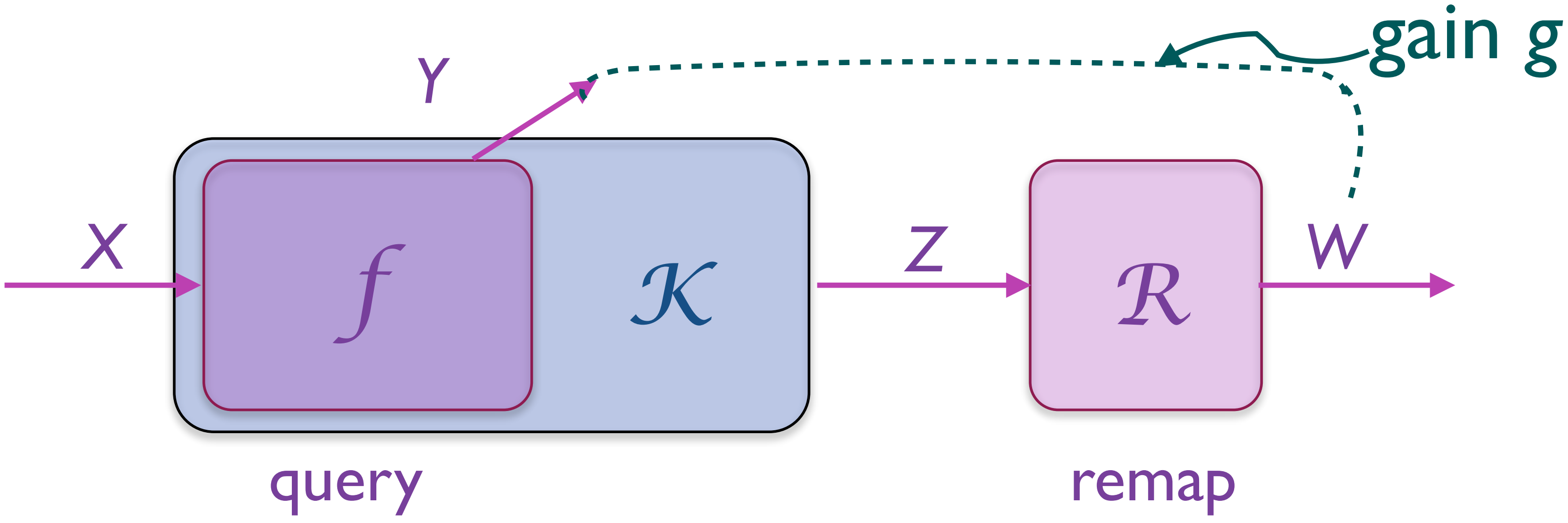
Utility

- 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):

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

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

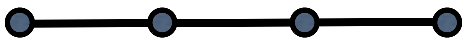
1. [Ghosh et al., STOC 2009]
   The geometric mechanism and the truncated geometric mechanism are universally optimal for counting queries and any anti-monotonic gain function

Privacy vs utility:
two fundamental results

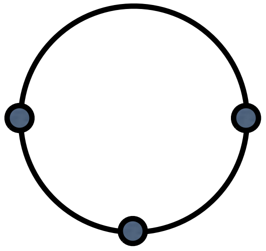
2. [Brenner and Nissim, STOC 2010] The counting queries are the only kind of queries for which a universally optimal mechanism exists

- This means that for other kind of queries one the optimal mechanism is relative to a specific user.

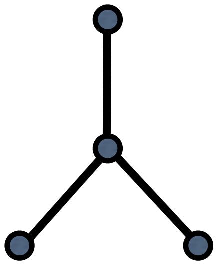
- The precise characterization is given in terms of the graph $(\mathcal{Y},\sim)$ induced by $(\mathcal{X},\sim)$



ok



not ok



not ok

Exercises

1. 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.
2. Same exercise, but with the gain function defined as the converse of the distance.
3. Find a mechanism for the same counting query, with the same degree of privacy, but lower utility
4. We saw that post-processing cannot decrease privacy. Can it decrease the utility? Motivate your answer