Privacy

Differential Privacy

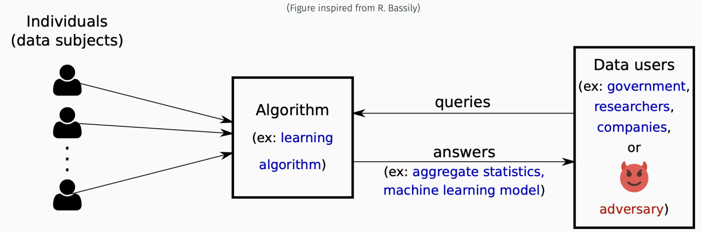
Guillaume Raschia — Nantes Université

Last update: January 3, 2026

original slides from A. Bellet (Inria), M2DS Univ. Lilles
and A. Machanavajjhala, M. Hay, X. He; Differential Privacy in the Wild, VLDB'16 & SIGMOD'17 Tutorial

REMINDER: PRIVATE DATA ANALYSIS

(Figure inspired from R. Bassily)



Goal: **achieve utility** while **preserving privacy** (conflicting objectives!)

REMINDER: REQUIREMENTS FOR PRIVACY DEFINITION

1. **Robustness to any auxiliary knowledge** the adversary may have, since one cannot predict what an adversary knows or might know in the future
2. **Composition over multiple analyses**: keep track of the "privacy budget" when asking several questions about the same data

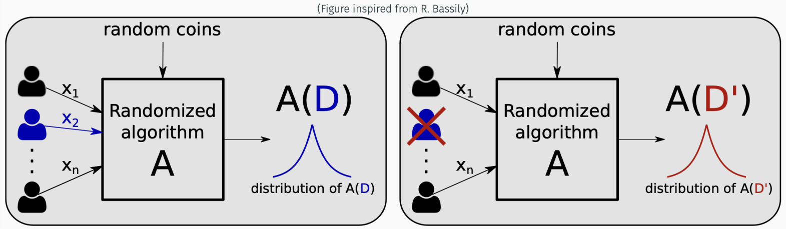
OUTLINE

Next Topic

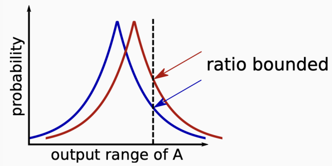
- Differential Privacy (DP)
- A First DP Algorithm
- Properties of DP

SCHEMATIC DIFFERENTIAL PRIVACY

(Figure inspired from R. Bassily)



Requirement: $\mathcal{A}(D)$ and $\mathcal{A}(D')$ should have "close" distributions



DIFFERENTIAL PRIVACY

Definition (Differential Privacy)
A randomized mechanism $\mathcal{A}$ preserves ε-differential privacy if for any pair of neighboring datasets $\mathbf{D}$ and $\mathbf{D}'$, and for all possible sets of outputs S:

$$\Pr[\mathcal{A}(\mathbf{D}) \in S] \leq e^{\varepsilon} \cdot \Pr[\mathcal{A}(\mathbf{D}') \in S], \quad \varepsilon > 0$$

DIFFERENTIAL PRIVACY

Definition (Differential Privacy)
A randomized mechanism $\mathcal{A}$ preserves ε-differential privacy if for any pair of neighboring datasets $\mathbf{D}$ and $\mathbf{D}'$, and for all possible sets of outputs S:

$$\Pr[\mathcal{A}(\mathbf{D}) \in S] \leq e^{\varepsilon} \cdot \Pr[\mathcal{A}(\mathbf{D}') \in S], \quad \varepsilon > 0$$

Parameter ε is called "privacy budget": it controls the degree to which $\mathbf{D}$ and $\mathbf{D}'$ can be distinguished. Smaller ε gives more privacy (and worse utility)

DIFFERENTIAL PRIVACY

Definition (Differential Privacy)
A randomized mechanism $\mathcal{A}$ preserves ε-differential privacy if for any pair of neighboring datasets $\mathbf{D}$ and $\mathbf{D}'$, and for all possible sets of outputs S:

$$\Pr[\mathcal{A}(\mathbf{D}) \in S] \leq e^{\varepsilon} \cdot \Pr[\mathcal{A}(\mathbf{D}') \in S], \quad \varepsilon > 0$$

Parameter ε is called "privacy budget": it controls the degree to which $\mathbf{D}$ and $\mathbf{D}'$ can be distinguished. Smaller ε gives more privacy (and worse utility)

First introduced in [Dwork et al., 2006] by **Dwork, McSherry, Nissim and Smith who won the Gödel prize in 2017**

DECYPHER DP

- What does mean "neighboring" datasets?

DECYPHER DP

- What does mean "neighboring" datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)

DECYPHER DP

- What does mean "neighboring" datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record

DECYPHER DP

- What does mean “neighboring” datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = “one person”, most common and safe but there exist alternatives like “one person-day”

DECYPHER DP

- What does mean “neighboring” datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = “one person”, most common and safe but there exist alternatives like “one person-day”
  - Under one raw = one person: adding or removing means sizes of $\mathbf{D}$ and $\mathbf{D}'$ are different, updating or replacing preserves the size but $\mathbf{D}\Delta\mathbf{D}' = 2$!

DECYPHER DP

- What does mean “neighboring” datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = “one person”, most common and safe but there exist alternatives like “one person-day”
  - Under one raw = one person: adding or removing means sizes of $\mathbf{D}$ and $\mathbf{D}'$ are different, updating or replacing preserves the size but $\mathbf{D}\Delta\mathbf{D}' = 2$!
- Why all pairs of datasets?

DECYPHER DP

- What does mean “neighboring” datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = “one person”, most common and safe but there exist alternatives like “one person-day”
  - Under one raw = one person: adding or removing means sizes of $\mathbf{D}$ and $\mathbf{D}'$ are different, updating or replacing preserves the size but $\mathbf{D}\Delta\mathbf{D}' = 2$!
- Why all pairs of datasets?
  - Privacy guarantee holds no matter what the other records are

DECYPHER DP

- What does mean "neighboring" datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = "one person", most common and safe but there exist alternatives like "one person-day"
  - Under one raw = one person: adding or removing means sizes of $\mathbf{D}$ and $\mathbf{D}'$ are different, updating or replacing preserves the size but $\mathbf{D}\Delta\mathbf{D}' = 2$!
- Why all pairs of datasets?
  - Privacy guarantee holds no matter what the other records are
- Why all outputs?

DECYPHER DP

- What does mean "neighboring" datasets?
  - Pairs of datasets that differ in one row: $\mathbf{D}\Delta\mathbf{D}' \leq 1$ (symmetric difference)
  - Simulate the presence or absence of a single record
  - Unit of privacy = "one person", most common and safe but there exist alternatives like "one person-day"
  - Under one raw = one person: adding or removing means sizes of $\mathbf{D}$ and $\mathbf{D}'$ are different, updating or replacing preserves the size but $\mathbf{D}\Delta\mathbf{D}' = 2$!
- Why all pairs of datasets?
  - Privacy guarantee holds no matter what the other records are
- Why all outputs?
  - Should not be able to distinguish whether input was $\mathbf{D}$ or $\mathbf{D}'$ no matter what the output

ABOUT ε PARAMETER

Privacy budget is actually a privacy loss

$$\varepsilon \geq \ln\left(\frac{\Pr[\mathcal{A}(\mathbf{D}) \in S]}{\Pr[\mathcal{A}(\mathbf{D}') \in S]}\right)$$

Small value of ε requires $\mathcal{A}$ to provide very similar outputs when given similar inputs

How should we set ε to prevent bad outcomes in practice? Nobody knows...

- Remind $e^{\varepsilon} \approx 1 + \varepsilon$ for very small ε values

ABOUT ε PARAMETER

Privacy budget is actually a privacy loss

$$\varepsilon \geq \ln\left(\frac{\Pr[\mathcal{A}(\mathbf{D}) \in S]}{\Pr[\mathcal{A}(\mathbf{D}') \in S]}\right)$$

Small value of ε requires $\mathcal{A}$ to provide very similar outputs when given similar inputs

How should we set ε to prevent bad outcomes in practice? Nobody knows...

- Remind $e^{\varepsilon} \approx 1 + \varepsilon$ for very small ε values
- up to 1.0 gives a strong privacy: $\varepsilon = 0.1$ bounds leak to 10%

ABOUT ε PARAMETER

Privacy budget is actually a privacy loss

$$\varepsilon \geq \ln\left(\frac{\Pr[\mathcal{A}(\mathbf{D}) \in S]}{\Pr[\mathcal{A}(\mathbf{D}') \in S]}\right)$$

Small value of ε requires $\mathcal{A}$ to provide very similar outputs when given similar inputs

How should we set ε to prevent bad outcomes in practice? Nobody knows...

- Remind $e^{\varepsilon} \approx 1 + \varepsilon$ for very small ε values
- up to 1.0 gives a strong privacy: $\varepsilon = 0.1$ bounds leak to 10%
- 1.0 to 10 is “better than nothing”
- more than 10 hardly protects privacy...

WHY S IS A SET?

$\mathcal{A}(\mathbf{D}) \in S$ vs. $\mathcal{A}(\mathbf{D}) = s$?

If $\mathcal{A}$ returns elements from a continuous output domain, $\Pr[\mathcal{A}(\mathbf{D}) = s] = 0$ for all $\mathbf{D}$

The DP definition makes sense for both discrete and continuous distributions.

For discrete outputs, then the definition may be

$$\Pr[\mathcal{A}(\mathbf{D}) = s] \leq e^{\varepsilon} \cdot \Pr[\mathcal{A}(\mathbf{D}') = s]$$

CAN DETERMINISTIC ALGORITHMS SATISFY DP?

Non-trivial deterministic algorithm has at least two distinct outputs in its image

There exist two inputs that differ in one row, mapped to distinct outputs:

- Assume $\mathbf{D} = \mathbf{D}' \cup \{x\}$, x the target row,
- and $\mathcal{A}(\mathbf{D}) = o_1$, $\mathcal{A}(\mathbf{D}') = o_2$ deterministically (so undoubtedly)

Then, a Differencing Attack may disclose the target’s data

Aside, $\Pr[\mathcal{A}(\mathbf{D}) = o_1] = 1.0$ and $\Pr[\mathcal{A}(\mathbf{D}') = o_1] = 0.0$

WHAT ABOUT RANDOM SAMPLING?

Assume $\mathbf{D} = \mathbf{D}' \cup \{x\}$, x the target row;

As soon as row x is sampled in o, then $\Pr[\mathcal{A}(\mathbf{D}') = o] = 0.0$, and

$$\frac{\Pr[\mathcal{A}(\mathbf{D}) \in S]}{\Pr[\mathcal{A}(\mathbf{D}') \in S]} = +\infty$$

Privacy loss is infinite!

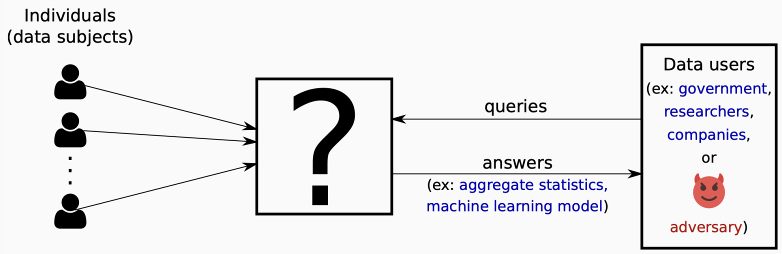
Next Topic

Differential Privacy (DP)

A First DP Algorithm

Properties of DP

HOW TO DESIGN DP ALGORITHMS?



ANSWERING NUMERICAL QUERIES

- Suppose we want to compute a numerical function $f : \mathcal{X}^n \to \mathbb{R}$ of a private dataset $\mathbf{D}$
- How to construct a DP algorithm (or mechanism $\mathcal{A}$) for computing $f(\mathbf{D})$?
  - How much randomness (error) do we add?
  - How to introduce this randomness in the output?

  A popular approach: the Laplace mechanism

THE LAPLACE MECHANISM: ALGORITHM & PRIVACY GUARANTEES

Algorithm: Laplace mechanism $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f : \mathcal{X}^n \to \mathbb{R}, \varepsilon)$

1. Compute $\mathbf{\Delta} = \mathbf{\Delta}_1(f)$, the sensitivity of function f
2. draw $Y \sim \text{Lap}(\mathbf{\Delta}/\varepsilon)$, the added noise
3. Output $f(\mathbf{D}) + Y$, the noisy answer to query f over $\mathbf{D}$

THE LAPLACE MECHANISM: ALGORITHM & PRIVACY GUARANTEES

Algorithm: Laplace mechanism $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f : \mathcal{X}^n \to \mathbb{R}, \varepsilon)$

1. Compute $\mathbf{\Delta} = \mathbf{\Delta}_1(f)$, the sensitivity of function f
2. draw $Y \sim \text{Lap}(\mathbf{\Delta}/\varepsilon)$, the added noise
3. Output $f(\mathbf{D}) + Y$, the noisy answer to query f over $\mathbf{D}$

Idea

perturb $f(\mathbf{D})$ with Laplace noise, to get $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon) := f(\mathbf{D}) + \text{Lap}(\frac{\mathbf{\Delta}}{\varepsilon})$

- noise is calibrated to sensitivity $\mathbf{\Delta}$ of f and the privacy parameter ε

THE LAPLACE MECHANISM: ALGORITHM & PRIVACY GUARANTEES

Algorithm: Laplace mechanism $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f : \mathcal{X}^n \to \mathbb{R}, \varepsilon)$

1. Compute $\mathbf{\Delta} = \mathbf{\Delta}_1(f)$, the sensitivity of function f
2. draw $Y \sim \text{Lap}(\mathbf{\Delta}/\varepsilon)$, the added noise
3. Output $f(\mathbf{D}) + Y$, the noisy answer to query f over $\mathbf{D}$

Idea

perturb $f(\mathbf{D})$ with Laplace noise, to get $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon) := f(\mathbf{D}) + \text{Lap}(\frac{\mathbf{\Delta}}{\varepsilon})$

- noise is calibrated to sensitivity $\mathbf{\Delta}$ of f and the privacy parameter ε

Theorem (DP guarantees for Laplace mechanism)

The Laplace mechanism $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon)$ satisfies ε-differential privacy

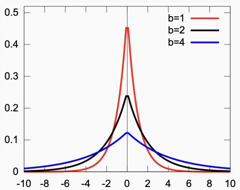
THE LAPLACE DISTRIBUTION

Definition (Laplace distribution)

The Laplace distribution Lap(b) (centered at 0) with scale b is the distribution with probability density function:

$$p(y; b) = \frac{1}{2b} \exp\left(-\frac{|y|}{b}\right), \quad y \in \mathbb{R}.$$

- It is a symmetric version of the exponential distribution
- For $Y \sim \text{Lap}(b)$, we have $\mathbb{E}[Y] = 0$, $\mathbb{E}[|Y|] = b$, $\mathbb{E}[Y^2] = 2b^2$
- Useful property for DP: $\Pr[Y = y]/\Pr[Y + a = y]$ can be bounded by something which does not depend on y



THE LAPLACE MECHANISM: UTILITY GUARANTEES

- This is great but what is the error incurred when using $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon)$ to answer $f(\mathbf{D})$?
- For a given output of $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon)$, we can consider the ℓ_1 error $||\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon) - f(\mathbf{D})||_1$

Theorem (Expected ℓ_1 error of the Laplace mechanism)

Let $\varepsilon > 0$. For a query $f : \mathcal{X}^n \to \mathbb{R}$ and any dataset $\mathbf{D} \in \mathcal{X}^n$, the Laplace mechanism $\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon)$ has the following utility guarantee:

$$\mathbb{E}[||\mathcal{A}_{\text{Lap}}(\mathbf{D}, f, \varepsilon) - f(\mathbf{D})||_1] = \frac{\mathbf{\Delta}_1(f)}{\varepsilon}.$$

- The Laplace mechanism can answer low sensitivity queries, say $\mathbf{\Delta}_1(f) = O(1)$ or smaller, with high utility (as long as ε is not too small)

THE LAPLACE MECHANISM: USE CASE

- Assume $\mathbf{\Delta}_1(f) = 1$ and $\varepsilon = 0.1$
- How much noise do we add? or what is a "typical" noise value?

THE LAPLACE MECHANISM: USE CASE

- Assume $\mathbf{\Delta}_1(f) = 1$ and $\varepsilon = 0.1$
- How much noise do we add? or what is a "typical" noise value?
    - scale $b = \mathbf{\Delta}_1(f)/\varepsilon = 10$
    - "typical" noise is $b\sqrt{2} = 14$

THE LAPLACE MECHANISM: USE CASE

- Assume $\mathbf{\Delta}_1(f) = 1$ and $\varepsilon = 0.1$
- How much noise do we add? or what is a "typical" noise value?
    - scale $b = \mathbf{\Delta}_1(f)/\varepsilon = 10$
    - "typical" noise is $b\sqrt{2} = 14$
- Let's compute the probability of the "tail region", i.e. noise $> b$:

$$2 \cdot \int_b^\infty p(y; b)\, dy = 2 \cdot \frac{1}{2b} \cdot \int_b^\infty \exp\left(-\frac{|y|}{b}\right) dy = -\frac{2b}{2b} \cdot \left[e^{-\frac{y}{b}}\right]_b^\infty = e^{-1} = 0.36$$

THE LAPLACE MECHANISM: USE CASE

- Assume $\mathbf{\Delta}_1(f) = 1$ and $\varepsilon = 0.1$
- How much noise do we add? or what is a "typical" noise value?
    - scale $b = \mathbf{\Delta}_1(f)/\varepsilon = 10$
    - "typical" noise is $b\sqrt{2} = 14$
- Let's compute the probability of the "tail region", i.e. noise $> b$:

$$2 \cdot \int_b^\infty p(y; b)\, dy = 2 \cdot \frac{1}{2b} \cdot \int_b^\infty \exp\left(-\frac{|y|}{b}\right) dy = -\frac{2b}{2b} \cdot \left[e^{-\frac{y}{b}}\right]_b^\infty = e^{-1} = 0.36$$

- In 1 over 3 random samples, the Laplace mechanism adds noise greater than 10
- Is this answer useful?

THE LAPLACE MECHANISM: USE CASE

- Assume $\boldsymbol{\Delta}_1(f) = 1$ and $\varepsilon = 0.1$
- How much noise do we add? or what is a “typical” noise value?
  - scale $b = \boldsymbol{\Delta}_1(f)/\varepsilon = 10$
  - “typical” noise is $b\sqrt{2} = 14$
- Let’s compute the probability of the “tail region”, i.e. noise $> b$:

$$2 \cdot \int_b^\infty p(y; b)\, dy = 2 \cdot \frac{1}{2b} \cdot \int_b^\infty \exp\left(-\frac{|y|}{b}\right) dy$$
$$= -\frac{2b}{2b} \cdot \left[e^{-\frac{y}{b}}\right]_b^\infty = e^{-1} = 0.36$$

- In 1 over 3 random samples, the Laplace mechanism adds noise greater than 10
- Is this answer useful?
  - Yes, if the real answer is $\gg 10$
  - No, otherwise

GLOBAL SENSITIVITY

Definition (Global ℓ_1 sensitivity)
The global ℓ_1 sensitivity of a query (function) $f : \mathcal{X}^n \to \mathbb{R}$ is

$$\boldsymbol{\Delta}_1(f) = \max_{\mathbf{D},\mathbf{D}':\mathbf{D}\Delta\mathbf{D}'\leq 1} |f(\mathbf{D}) - f(\mathbf{D}')|_1$$

- *global* means it holds for all pairs of neighboring datasets
- How much one record can affect the output value of the function
- Intuitively, it gives the amount of uncertainty needed to hide any single contribution

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$
- How many people have blond hair?

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$
- How many people have blond hair?
- How many males, how many people with blond hair?

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$
- How many people have blond hair?
- How many males, how many people with blond hair?
- How many people have blond hair, how many people have dark hair, how many people have brown hair, how many people have red hair?

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$
- How many people have blond hair?
- How many males, how many people with blond hair?
- How many people have blond hair, how many people have dark hair, how many people have brown hair, how many people have red hair?
- What is the sum of the salaries, knowing salaries range between 20K€ and 200K€

INTERPRETING GLOBAL SENSITIVITY

Think about the sensitivity of the following functions/queries:

- $f(x) = x$, for real numbers
- $f(x) = x + x$
- $f(x) = 5x$
- $f(x) = x^2$
- How many people have blond hair?
- How many males, how many people with blond hair?
- How many people have blond hair, how many people have dark hair, how many people have brown hair, how many people have red hair?
- What is the sum of the salaries, knowing salaries range between 20K€ and 200K€
- What is the average age?

CLIPPING

Queries with unbounded sensitivity cannot be straightforwardly answered with the Laplace mechanism

Definition (Clipping)
Enforce lower and upper bounds of a given function, as a *band-pass filter*, to fall back into bounded sensitivity

- Trade-off between information lost in clipping and noise needed to ensure DP
  - aggressive clipping (close bounds) yields to lower sensitivity then less noise
  - conservative clipping (broad range) yields to higher sensitivity then more noise

CLIPPING

Queries with unbounded sensitivity cannot be straightforwardly answered with the Laplace mechanism

Definition (Clipping)
Enforce lower and upper bounds of a given function, as a *band-pass filter*, to fall back into bounded sensitivity

- Trade-off between information lost in clipping and noise needed to ensure DP
  - aggressive clipping (close bounds) yields to lower sensitivity then less noise
  - conservative clipping (broad range) yields to higher sensitivity then more noise
- As a rule of thumb: clipping bounds should include 100% of the dataset

CLIPPING

Queries with unbounded sensitivity cannot be straightforwardly answered with the Laplace mechanism

Definition (Clipping)
Enforce lower and upper bounds of a given function, as a *band-pass filter*, to fall back into bounded sensitivity

- Trade-off between information lost in clipping and noise needed to ensure DP
  - aggressive clipping (close bounds) yields to lower sensitivity then less noise
  - conservative clipping (broad range) yields to higher sensitivity then more noise
- As a rule of thumb: clipping bounds should include 100% of the dataset
- But never ever scan the data to set bounds! or do it properly...i.e. in a “DP manner”

CLIPPING

Queries with unbounded sensitivity cannot be straightforwardly answered with the Laplace mechanism

Definition (Clipping)
Enforce lower and upper bounds of a given function, as a *band-pass filter*, to fall back into bounded sensitivity

- Trade-off between information lost in clipping and noise needed to ensure DP
  - aggressive clipping (close bounds) yields to lower sensitivity then less noise
  - conservative clipping (broad range) yields to higher sensitivity then more noise
- As a rule of thumb: clipping bounds should include 100% of the dataset
- But never ever scan the data to set bounds! or do it properly...i.e. in a “DP manner”

Sensitivity underestimation may break the differential privacy guarantee, while sensitivity overestimation leads to unnecessary inaccuracy in the private analysis

Next Topic

ROBUSTNESS TO AUXILIARY KNOWLEDGE

- DP guarantees are intrinsically robust to arbitrary auxiliary knowledge: it bounds the relative advantage that an adversary gets from observing the output of an algorithm
  - Adversary may know all the dataset except one record
  - Adversary may know all external sources of knowledge, present and future

ROBUSTNESS TO AUXILIARY KNOWLEDGE

- DP guarantees are intrinsically robust to arbitrary auxiliary knowledge: it bounds the relative advantage that an adversary gets from observing the output of an algorithm
  - Adversary may know all the dataset except one record
  - Adversary may know all external sources of knowledge, present and future
- The algorithm $\mathcal{A}$ can be public: only the randomness needs to remain hidden
  - A key requirement of modern security ("security by obscurity" has long been rejected)
  - Allows to openly discuss the algorithms and their guarantees

RESILIENCE TO POSTPROCESSING

Theorem (Postprocessing)
Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{O}$ be ε-DP and let $f : \mathcal{O} \to \mathcal{O}'$ be an arbitrary (randomized) function, independent of A. Then

$$f \circ \mathcal{A} : \mathcal{X}^n \to \mathcal{O}'$$

is ε-DP.

RESILIENCE TO POSTPROCESSING

Theorem (Postprocessing)
Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{O}$ be ε-DP and let $f : \mathcal{O} \to \mathcal{O}'$ be an arbitrary (randomized) function, independent of A. Then

$$f \circ \mathcal{A} : \mathcal{X}^n \to \mathcal{O}'$$

is ε-DP.

- "Thinking about" the output of a differentially private algorithm cannot make it less differentially private

RESILIENCE TO POSTPROCESSING

Theorem (Postprocessing)
Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{O}$ be ε-DP and let $f : \mathcal{O} \to \mathcal{O}'$ be an arbitrary (randomized) function, independent of A. Then

$$f \circ \mathcal{A} : \mathcal{X}^n \to \mathcal{O}'$$

is ε-DP.

- “Thinking about” the output of a differentially private algorithm cannot make it less differentially private
- Can let data users do whatever they want with it

RESILIENCE TO POSTPROCESSING

Theorem (Postprocessing)
Let $\mathcal{A} : \mathcal{X}^n \to \mathcal{O}$ be ε-DP and let $f : \mathcal{O} \to \mathcal{O}'$ be an arbitrary (randomized) function, independent of A. Then

$$f \circ \mathcal{A} : \mathcal{X}^n \to \mathcal{O}'$$

is ε-DP.

- “Thinking about” the output of a differentially private algorithm cannot make it less differentially private
- Can let data users do whatever they want with it
- This holds regardless of attacker strategy and computational power

SEQUENTIAL COMPOSITION

Theorem (Simple composition)
Let $\mathcal{A}_1, \dots, \mathcal{A}_K$ be K independently chosen algorithms where $\mathcal{A}_k$ satisfies ε_k-DP. For any dataset $\mathbf{D}$, let $\mathcal{A}$ be such that

$$\mathcal{A}(\mathbf{D}) = (\mathcal{A}_1(\mathbf{D}), \dots, \mathcal{A}_K(\mathbf{D})).$$

Then $\mathcal{A}$ is ε-DP with $\varepsilon = \sum_{k=1}^{K} \varepsilon_k$.

SEQUENTIAL COMPOSITION

Theorem (Simple composition)
Let $\mathcal{A}_1, \dots, \mathcal{A}_K$ be K independently chosen algorithms where $\mathcal{A}_k$ satisfies ε_k-DP. For any dataset $\mathbf{D}$, let $\mathcal{A}$ be such that

$$\mathcal{A}(\mathbf{D}) = (\mathcal{A}_1(\mathbf{D}), \dots, \mathcal{A}_K(\mathbf{D})).$$

Then $\mathcal{A}$ is ε-DP with $\varepsilon = \sum_{k=1}^{K} \varepsilon_k$.

- This allows to control the cumulative privacy loss over multiple analyses run on the same dataset, including complex multi-step algorithms
- Total budget is an upper bound: actual privacy loss may be smaller
  - $(\text{Lap}(1/\varepsilon_1) + \text{Lap}(1/\varepsilon_2))/2$ is less accurate than $\text{Lap}(1/(\varepsilon_1 + \varepsilon_2))$

PARALLEL COMPOSITION

The previous composition result is worst-case (assumes correlated outputs)

Theorem (Parallel composition)

If $\mathcal{A}_1, \ldots, \mathcal{A}_K$ operate on distinct inputs, then $\mathcal{A}(\mathbf{D})$ is $\max_k \varepsilon_k$-DP

Example (Count by gender and hair color)

	Blond	Dark	Brown	Red
Female	20	33	9	7
Nonbinary	12	7	28	3
Male	17	42	4	8

If for each count the algorithm generating it satisfies ε-DP, then releasing the entire table is also ε-DP (as opposed to 12ε-DP with sequential composition!)

CONCLUSION

- Differential Privacy is robust to auxiliary knowledge
- DP is a property of the algorithm, not the dataset
- DP requires randomization
- Privacy loss is bounded by ε, also called "budget"
- The Laplace Mechanism provides ε-DP to numerical functions (queries)
- Laplace scale is calibrated to sensitivity of the function and ε
- Clipping ensures sensitivity is bounded
- DP mechanisms can be composed
  - in sequence, then $\varepsilon = \sum \varepsilon_k$, or
  - in parallel, then $\varepsilon = \max \varepsilon_k$
- DP is robust to postprocessing

References

Dwork, C., McSherry, F., Nissim, K., and Smith, A. (2006).
Calibrating noise to sensitivity in private data analysis.
In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, page 265–284, Berlin, Heidelberg. Springer-Verlag.