

CS489/698

Privacy, Cryptography, Network and Data Security

Differential Privacy

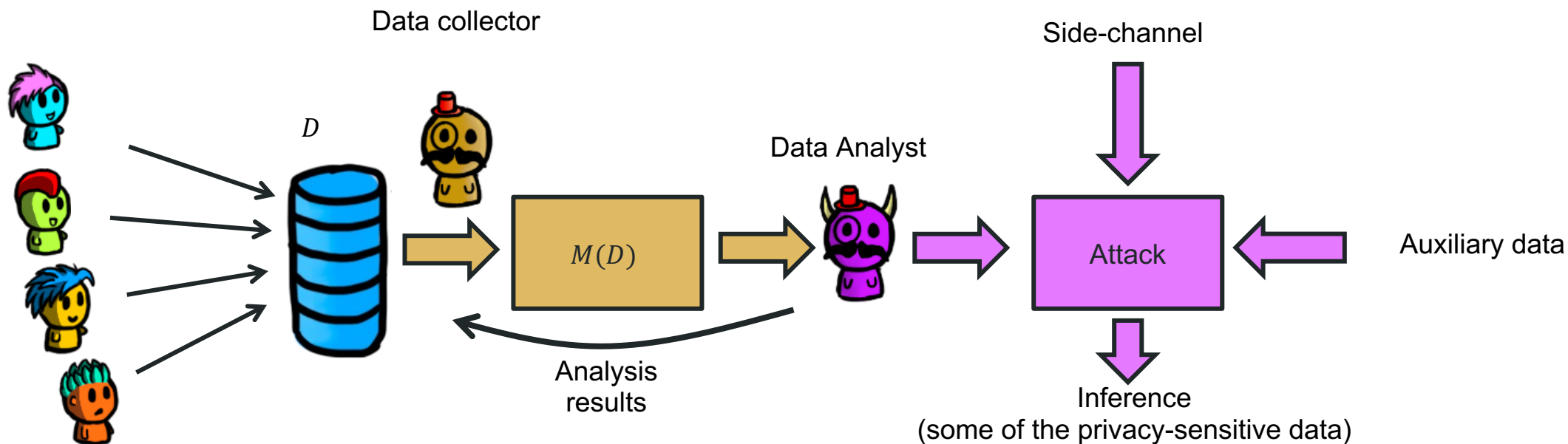
Spring 2024, Monday/Wednesday 11:30am-12:50pm

Issues with syntactic notions of privacy

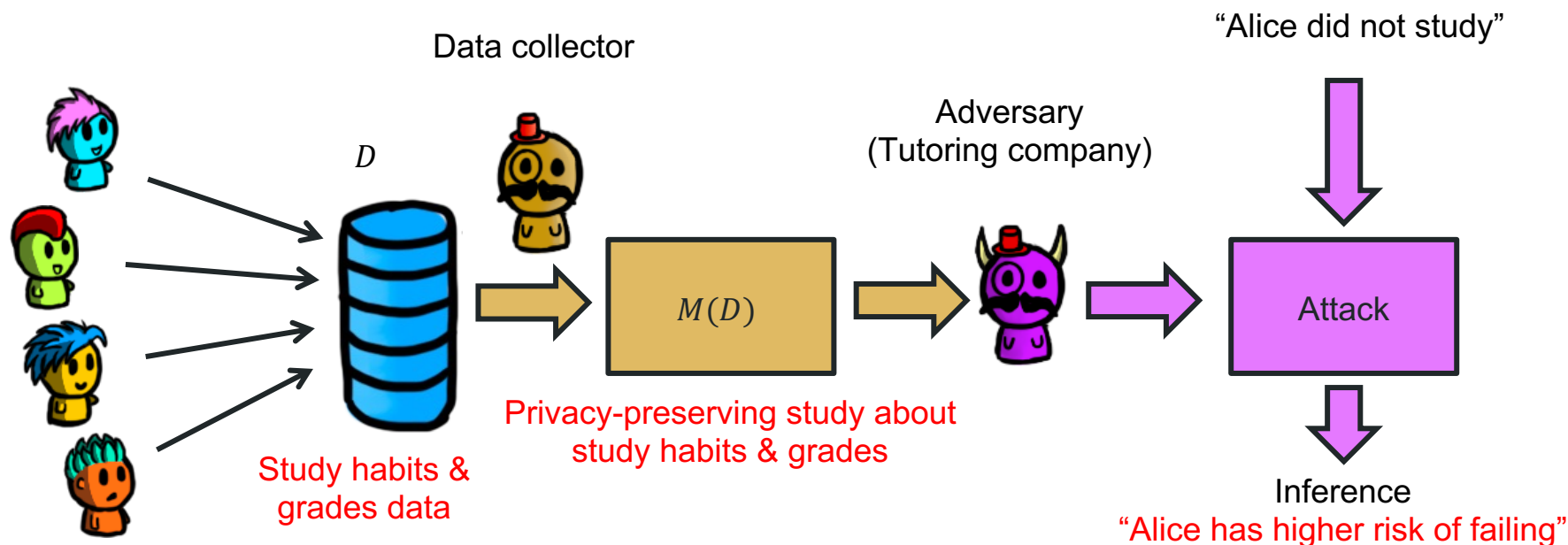
- As seen in the last lecture, syntactic notions of privacy have some issues:
 - Defining which attributes are quasi-identifiers and which are sensitive attributes is hard
 - Mostly apply to relational databases; what about general data releases like machine learning?
 - What if the adversary has arbitrary auxiliary information?
- We need a formal notion of privacy, that provides formal guarantees against (all) attacks.
 - But how do we achieve this?

Can we protect against auxiliary information?

- Each user contributes to one entry (row) of a database D .
- The release mechanism M publishes some data $R = M(D)$.
 - Note: we can characterize the mechanism by $\Pr(M(D) = R)$, which is the same as $\Pr(R|D)$ on inference attacks
- Can we provide privacy when the adversary has **auxiliary information**?

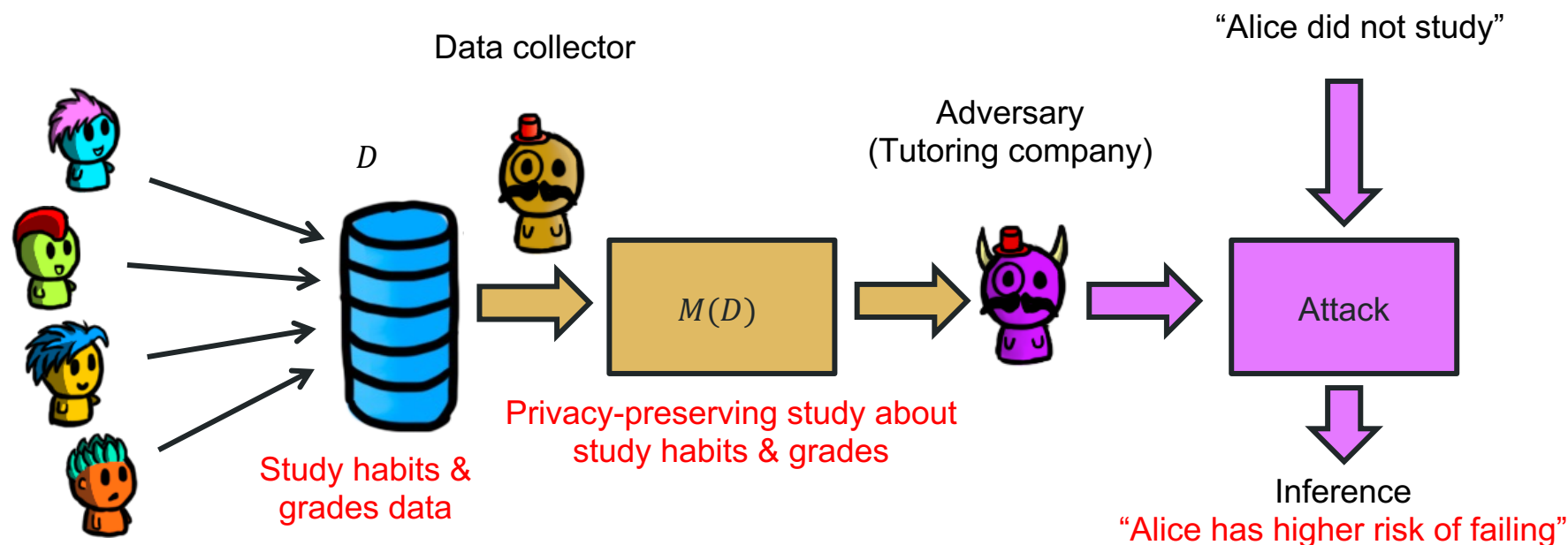


Example: strong auxiliary information



Q: Can we design a mechanism M that prevents this? Does it make sense to design a mechanism M that prevents this?

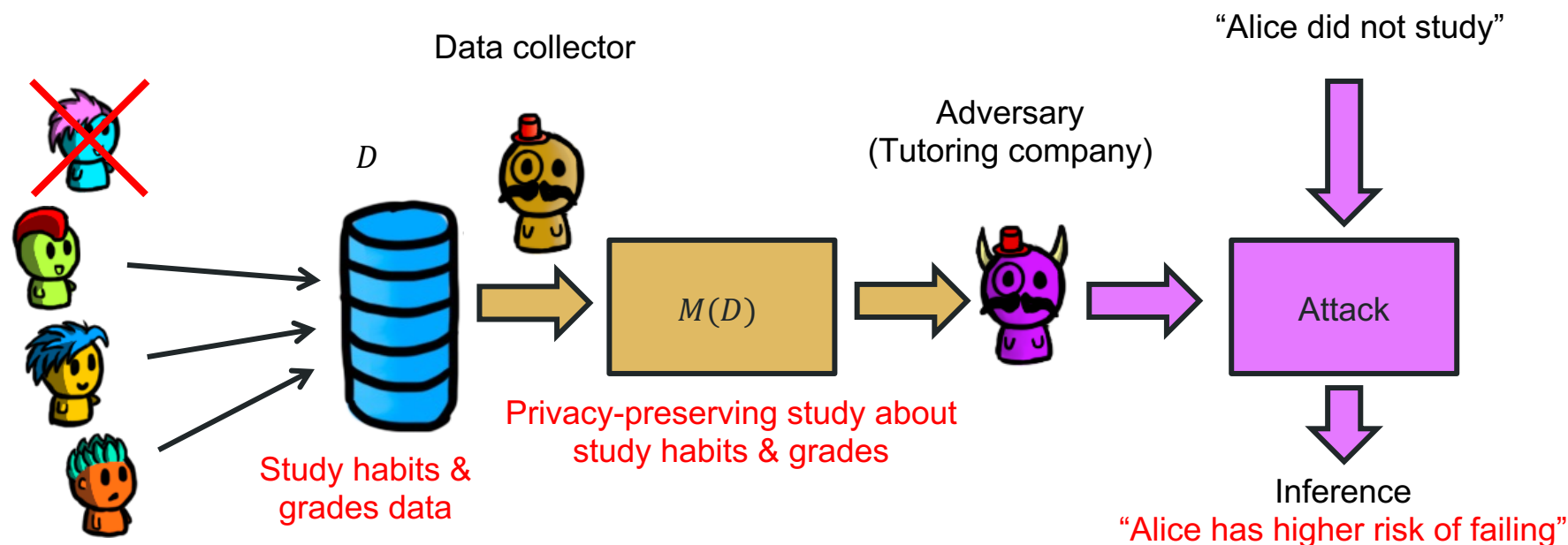
Example: strong auxiliary information



Q: Can we design a mechanism M that prevents this? Does it make sense to design a mechanism M that prevents this?

A: The adversary would've reached the same conclusion even if Alice hadn't participated in the study! We cannot prevent this unless we destroy utility (e.g., not doing the study)

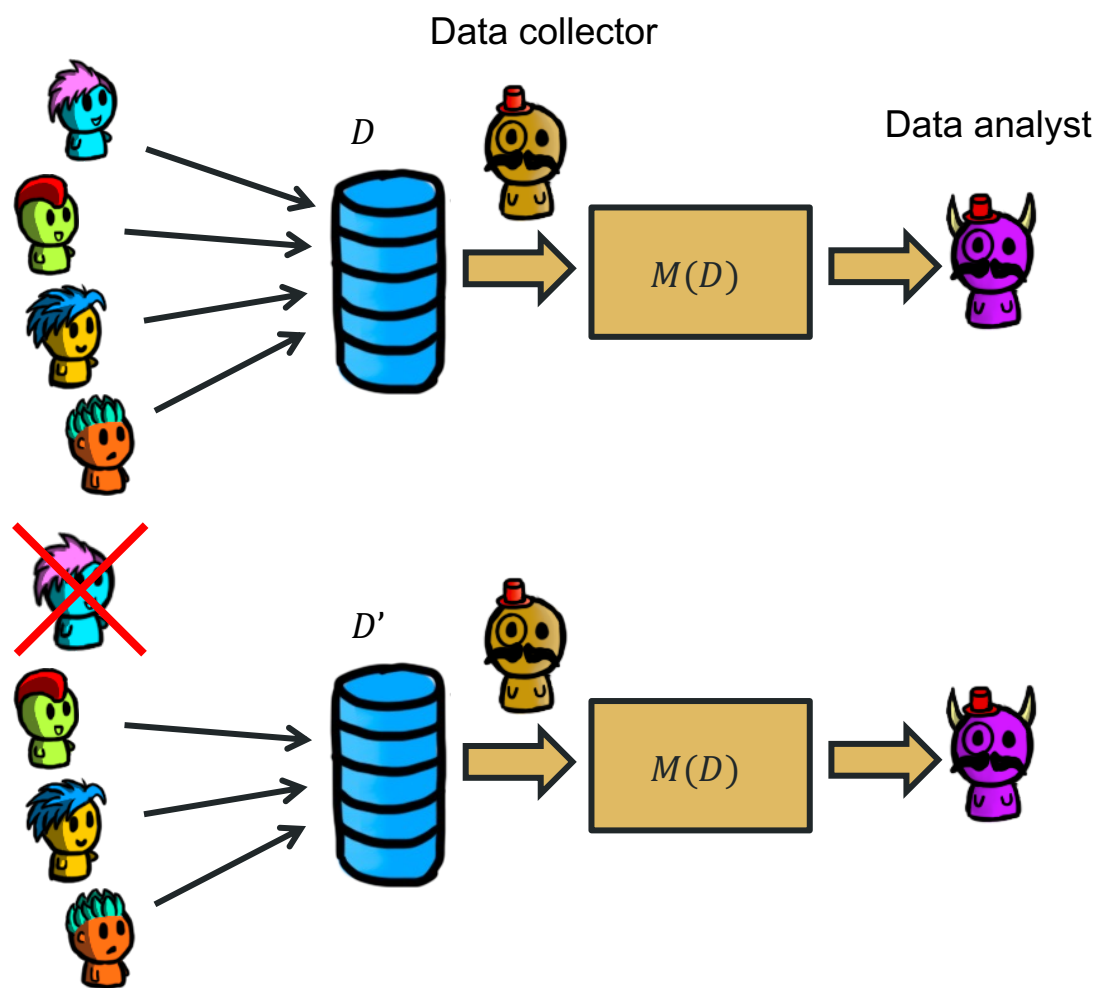
Example: strong auxiliary information



- Note that the adversary reaches the same conclusion in this case, even though Alice has not participated!

Q: Any ideas of how we could define privacy taking this into account?

Possible Idea:



- If the analyst learns similar things in these two cases about Alice, then M provides enough privacy
- If the adversary learns “a lot” about Alice in both cases, then we cannot prevent this anyway
- Given $R = M(D)$, the adversary should be unable to distinguish whether or not Alice was in the dataset!
- Note that this means that $M(D)$ has to be randomized (or always report the same value, but this makes R constant – independent of D – which is not useful.)

An example from the attacker's perspective

- **Background knowledge 1:** You know that Alice is a top-performer and always gets ≥ 90 in course scores.
- **Background knowledge 2:** CS489 is super-challenging and historical records show that most students score in the range of [45, 55].

An example from the attacker's perspective

- **Background knowledge 1:** You know that Alice is a top-performer and always gets ≥ 90 in course scores.
- **Background knowledge 2:** CS489 is super-challenging and historical records show that most students score in the range of $[45, 55]$.
- **Algorithm:** You are given an algorithm that
 - allows you to make 5 queries
 - each query returns the average score of 3 randomly selected students (out of 30 scores in total).

An example from the attacker's perspective

- **Background knowledge 1:** You know that Alice is a top-performer and always gets ≥ 90 in course scores.
- **Background knowledge 2:** CS489 is super-challenging and historical records show that most students score in the range of [45, 55].
- **Algorithm:** You are given an algorithm that
 - allows you to make 5 queries
 - each query returns the average score of 3 randomly selected students (out of 30 scores in total).

Q: How can you infer whether Alice is enrolled in CS489 or not?

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Q: What will happen if Alice IS NOT enrolled (i.e., D')?

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Q: What will happen if Alice IS NOT enrolled (i.e., D')?

A: Expect [50, 50, 50, 50, 50] in response.

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Q: What will happen if Alice IS NOT enrolled (i.e., D')?

A: Expect [50, 50, 50, 50, 50] in response.

Q: What will happen if Alice IS enrolled (i.e., D)?

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Q: What will happen if Alice IS NOT enrolled (i.e., D')?

A: Expect [50, 50, 50, 50, 50] in response.

Q: What will happen if Alice IS enrolled (i.e., D)?

A: For a single response, we either get:

$$63 \leftarrow \frac{C_{30}^2}{C_{30}^3} = 10.7 \%$$

50 ← otherwise

The attack

Just send 5 queries and observe what is returned by the database.

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Q: What will happen if Alice IS NOT enrolled (i.e., D')?

A: Expect [50, 50, 50, 50, 50] in response.

Q: What will happen if Alice IS enrolled (i.e., D)?

A: For a single response, we either get:

$$63 \leftarrow \frac{C_{30}^2}{C_{30}^3} = 10.7 \%$$

50 ← otherwise

A (cont.): For all 5 responses, the chance of getting at least one 63 is: $1 - \left(1 - \frac{C_{30}^2}{C_{30}^3}\right)^5 = 43.26\%$

What went wrong?

- Alice's score has **too much impact** on the output! As a result, seeing the output of the algorithm allows the attacker to differentiate which database is the underlying database representing the class score.
- This is exactly what **Differential Privacy (DP)** tries to capture!
 - Informally, the DP notion requires any single element in a dataset to have only a limited impact on the output.

The strawman defense

- **Background knowledge 1:** You know that Alice is a top-performer and always gets ≥ 90 in course scores.
- **Background knowledge 2:** CS489 is super-challenging and historical records show that most students score in the range of [45, 55].
- **Algorithm:** You are given an algorithm that
 - allows you to make 5 queries
 - each query returns the average score of 3 randomly selected students (out of 30 scores in total).

The strawman defense

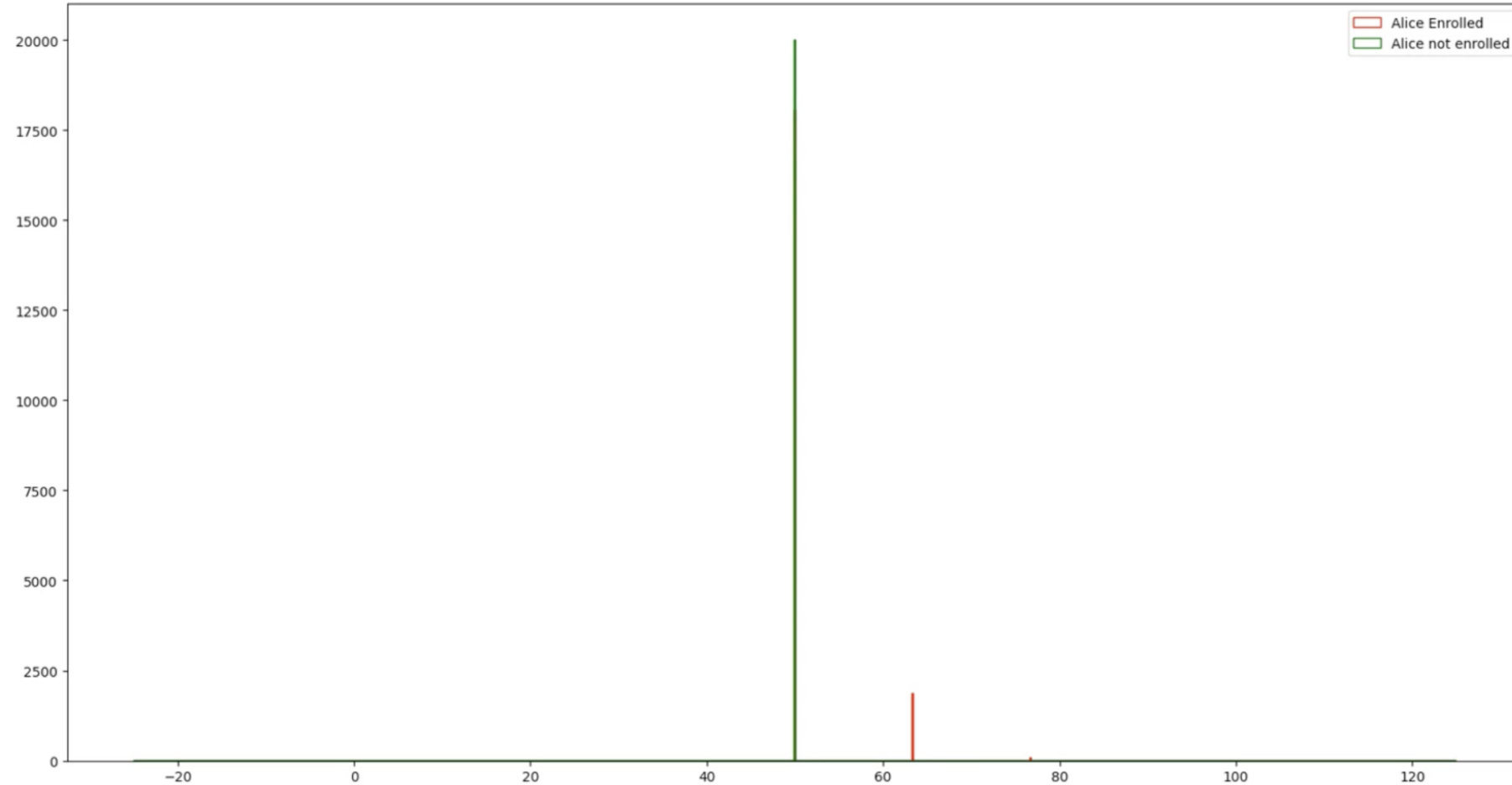
- **Background knowledge 1:** You know that Alice is a top-performer and always gets ≥ 90 in course scores.
- **Background knowledge 2:** CS489 is super-challenging and historical records show that most students score in the range of [45, 55].
- **Algorithm:** You are given an algorithm that
 - allows you to make 5 queries
 - each query returns the average score of 3 randomly selected students (out of 30 scores in total) **plus a random value (i.e., noise)**

Intuition: No noise

When Alice IS in the database:

Noticeable!

- For a given query, most times it will return 50
- Sometimes ($\approx 10\%$) it will return 63

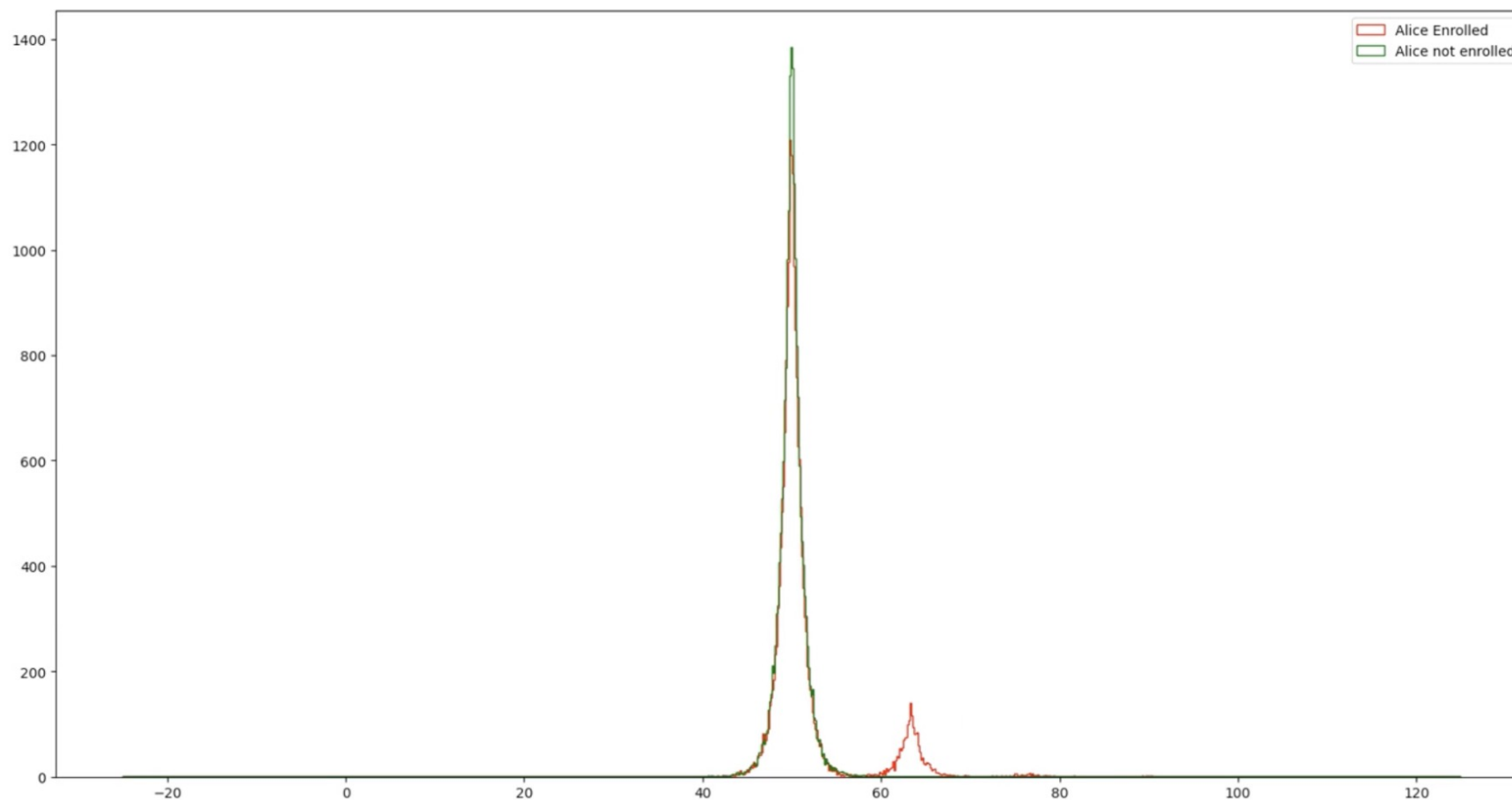


Intuition: Small noise

When Alice IS in the database:

Still noticeable!

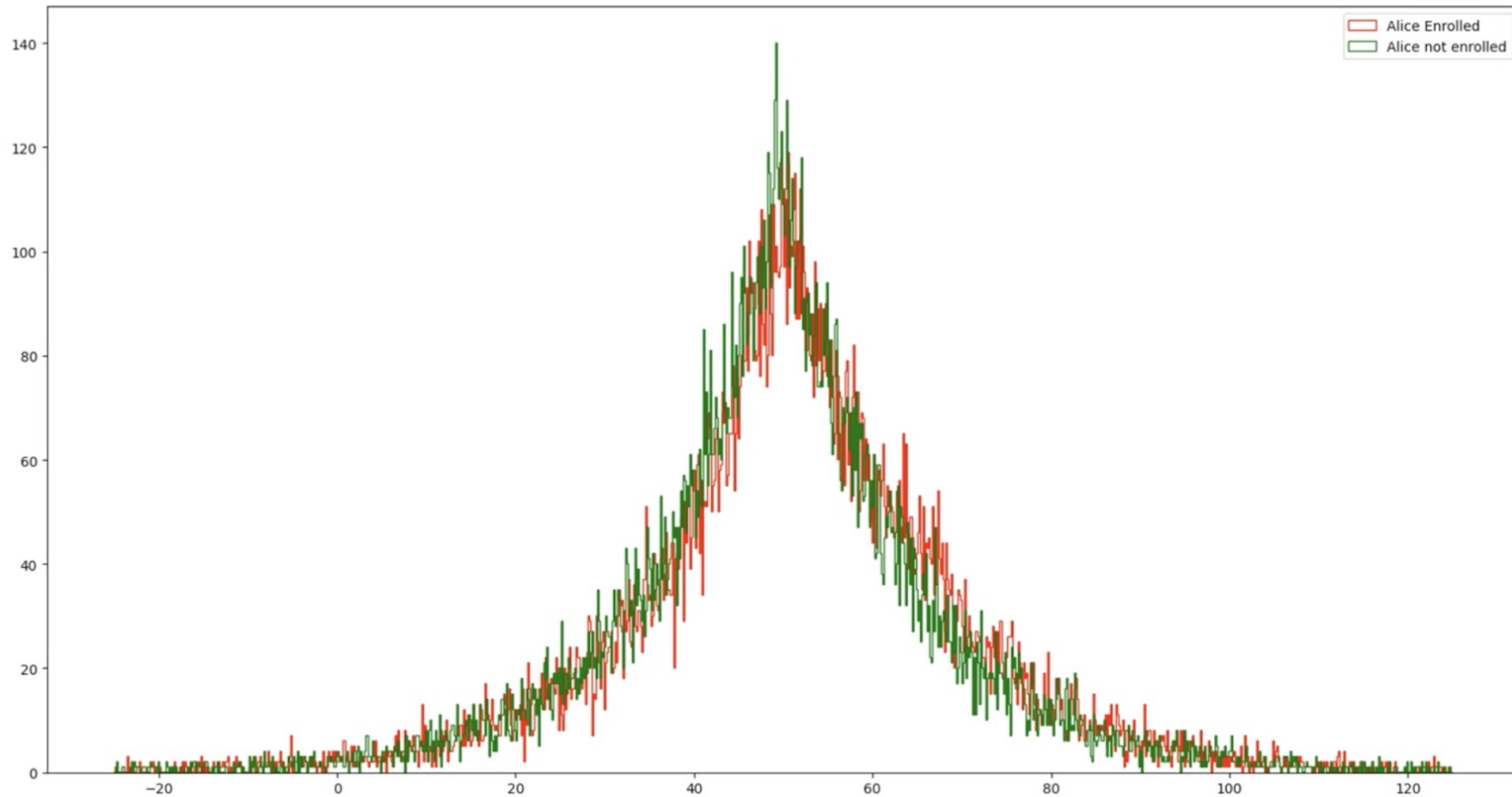
- For a given query, most times it will return ~ 50
- Sometimes it will return ~ 63



Intuition: Large noise

When Alice IS in the database: **Hardly noticeable!**

- Query results have a \sim probability whether Alice is in the database or not (with reasonable utility)



Intuition: Very large noise

When Alice IS in the database:

Unnoticeable!

- We can't tell if Alice is in the database
- But we completely destroy utility



Takeaway

- One should set an **appropriate amount of noise** depending on each particular use case.
 - We want to preserve data privacy
 - We don't want to destroy utility

The data collectors' argument

... on trying to persuade you to join a differentially private survey:

- *You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available. (bla bla... differential privacy ... bla bla)*

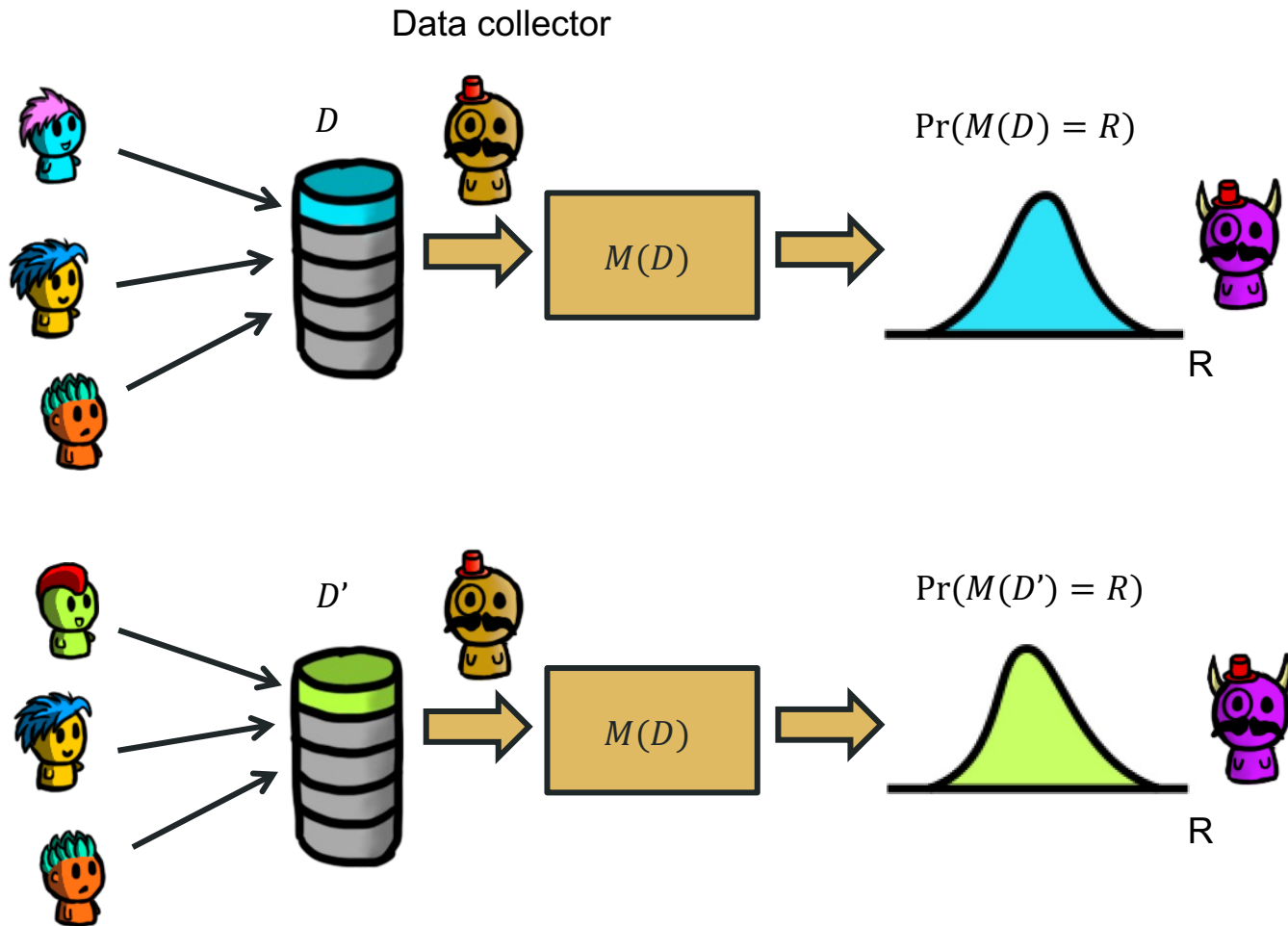
The data collectors' argument

... on trying to persuade you to join a differentially private survey:

- *You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available. (bla bla... differential privacy ... bla bla)*
- But this is only true if they tell you what algorithm they use to release your data and you have verified that their algorithm is indeed differentially private.

Back on topic: We want similar output distributions!

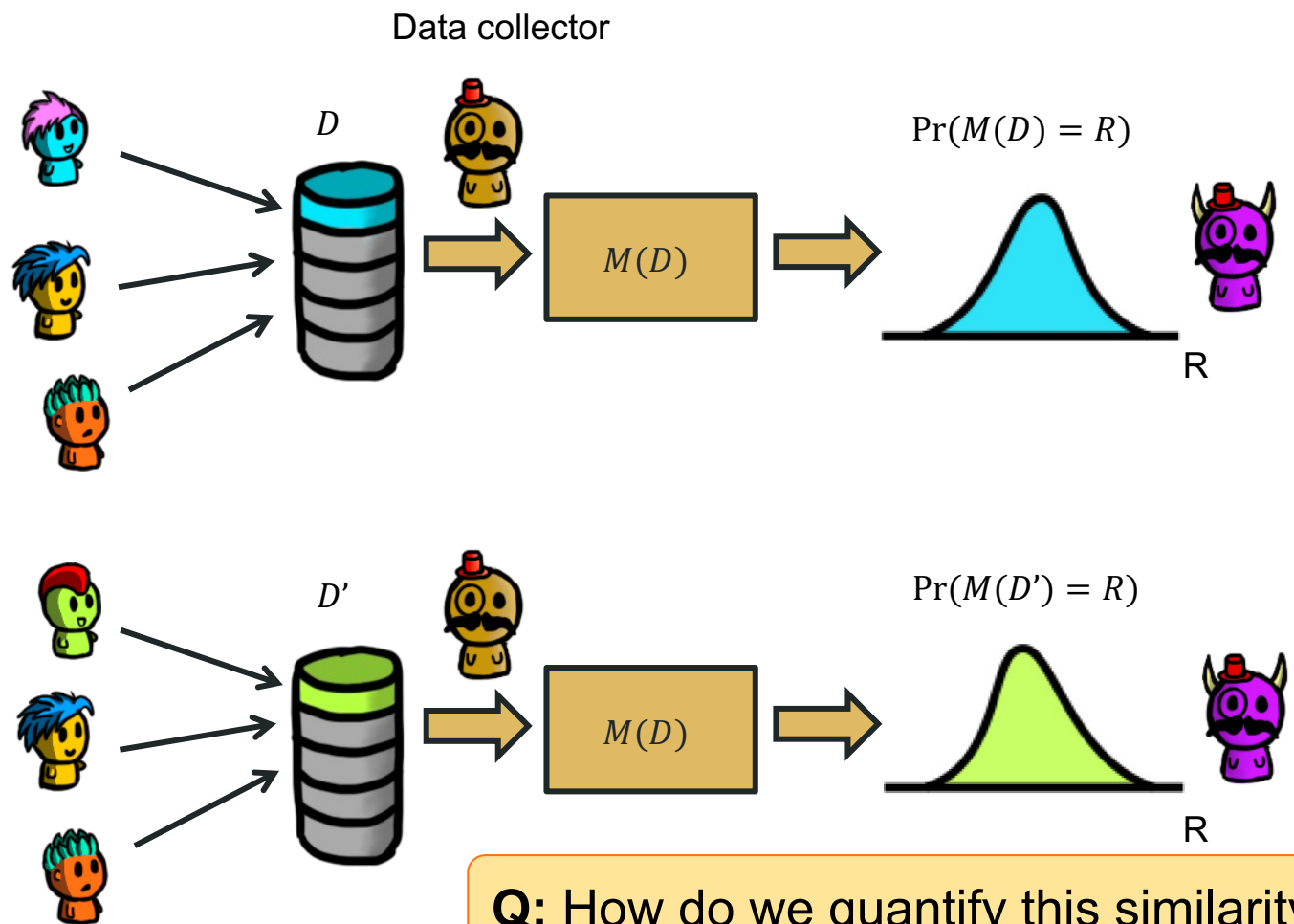
(assume for now that the databases differ on one single record)



- These datasets are usually called **neighboring datasets** (and usually denoted by D and D')
- We want these distributions to be “similar” (for all R)
- If the mechanism M behaves **nearly identically** for D and D' , then an attacker can't tell whether D or D' was used (and hence can't learn much about the individual).

Back on topic: We want similar output distributions!

(assume for now that the databases differ on one single record)



Q: How do we quantify this similarity?

- These datasets are usually called **neighboring datasets** (and usually denoted by D and D')
- We want these distributions to be “similar” (for all R)
- If the mechanism M behaves **nearly identically** for D and D' , then an attacker can't tell whether D or D' was used (and hence can't learn much about the individual).

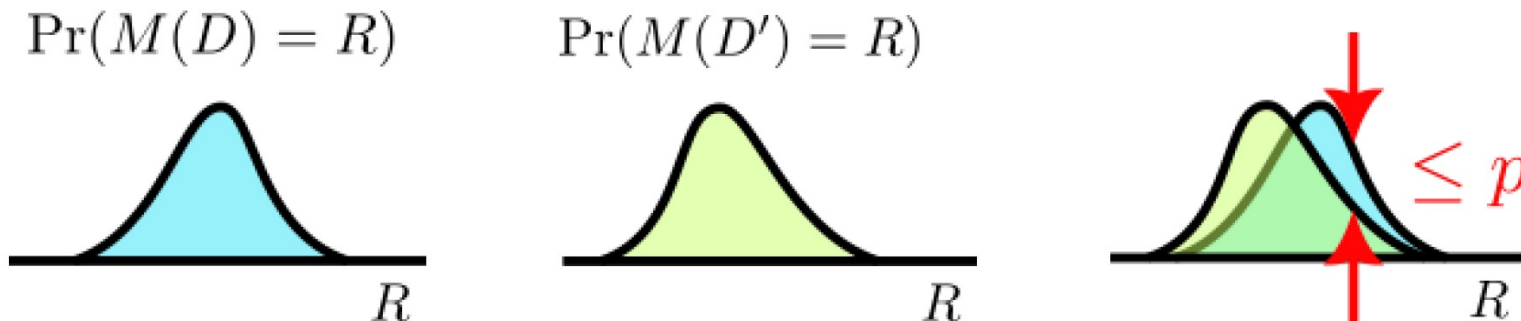
How do we define “similar” distributions?

Tentative privacy definition (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\Pr(M(D') = R) - p < \Pr(M(D) = R) < \Pr(M(D') = R) + p$$

- What does this mean?



Q: What gives more privacy, small or large p ?

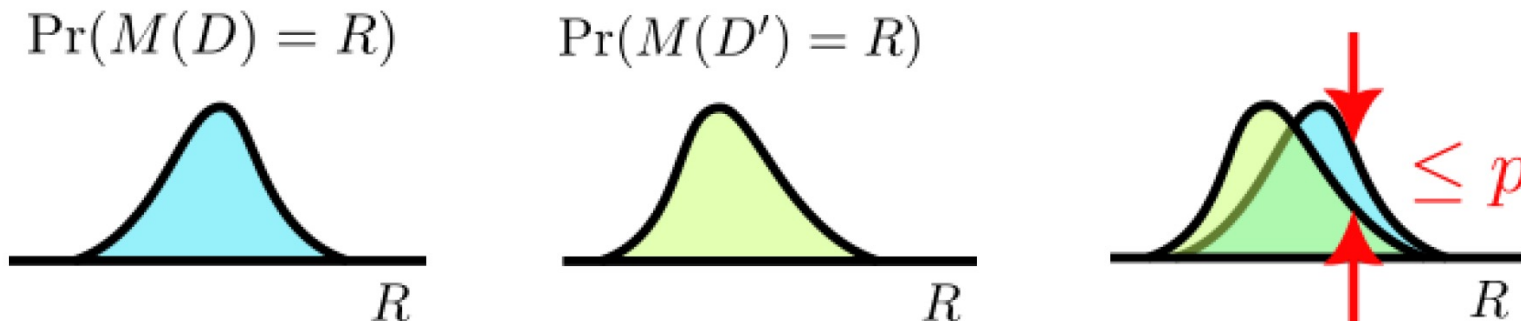
How do we define “similar” distributions?

Tentative privacy definition (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\Pr(M(D') = R) - p < \Pr(M(D) = R) < \Pr(M(D') = R) + p$$

- What does this mean?



Q: What gives more privacy, small or large p ?

A: Small p , the distributions are more alike

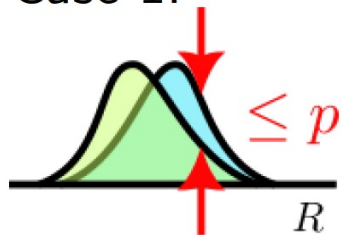
Does this really work?

Tentative privacy definition (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

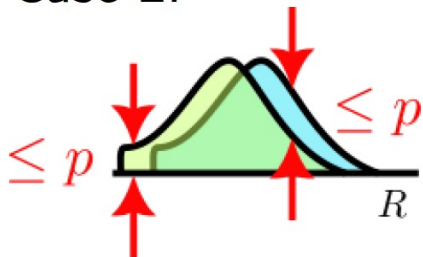
$$\Pr(M(D') = R) - p < \Pr(M(D) = R) < \Pr(M(D') = R) + p$$

Case 1:



Q: Case 1 seems fine. What is the issue with case 2?

Case 2:



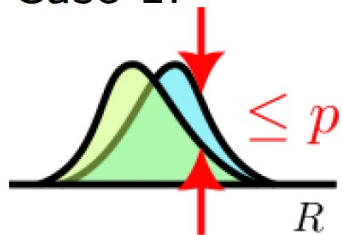
Does this really work?

Tentative privacy definition (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

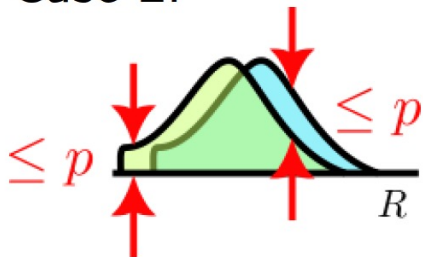
$$\Pr(M(D') = R) - p < \Pr(M(D) = R) < \Pr(M(D') = R) + p$$

Case 1:



Q: Case 1 seems fine. What is the issue with case 2?

Case 2:



A: There are some outputs R that can only happen if the input was D (e.g., if Alice was not in the dataset). This allows the adversary to distinguish between D and D' with 100% certainty.

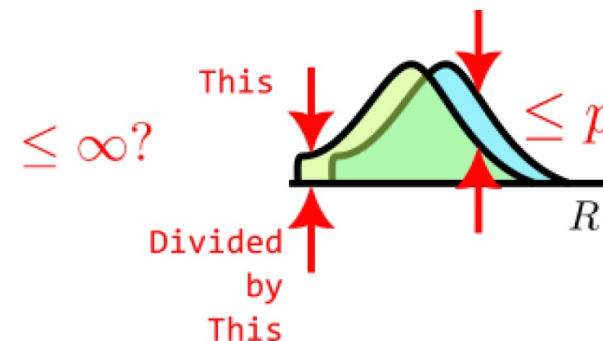
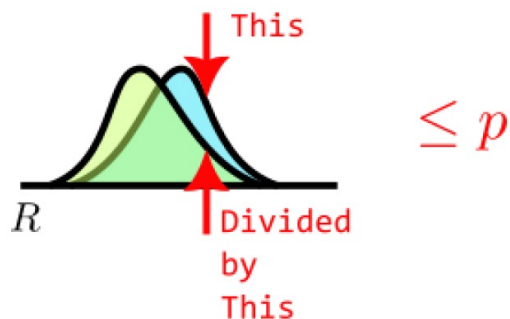
In other words, the attacker can find a **perspective** through which the two databases behave differently.

What if we make the distance multiplicative?

Tentative privacy definition II (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\frac{\Pr(M(D') = R)}{p} < \Pr(M(D) = R) < \Pr(M(D') = R) \cdot p$$



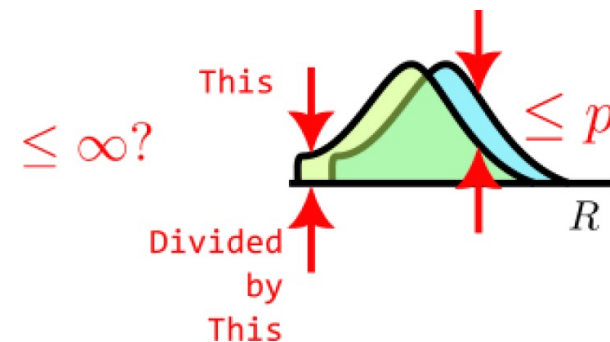
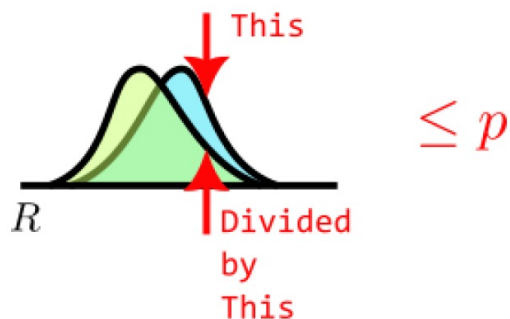
Q: what does provide more privacy, small (but larger than 1) or large p ?

What if we make the distance multiplicative?

Tentative privacy definition II (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\frac{\Pr(M(D') = R)}{p} < \Pr(M(D) = R) < \Pr(M(D') = R) \cdot p$$



Q: what does provide more privacy, small (but larger than 1) or large p ?

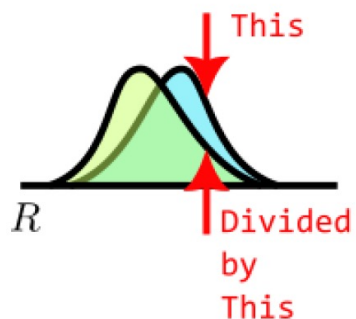
A: Small p

What if we make the distance multiplicative?

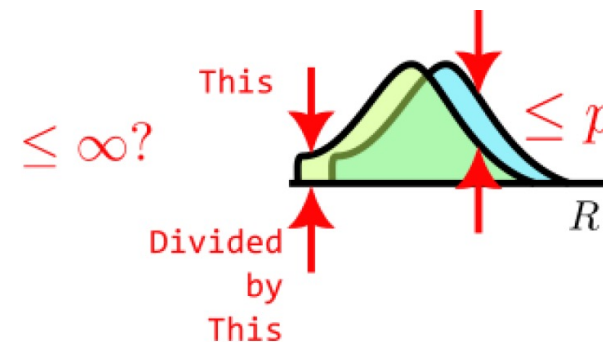
Tentative privacy definition II (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\frac{\Pr(M(D') = R)}{p} < \Pr(M(D) = R) < \Pr(M(D') = R) \cdot p$$



\leq **Q: Does this make sense?**

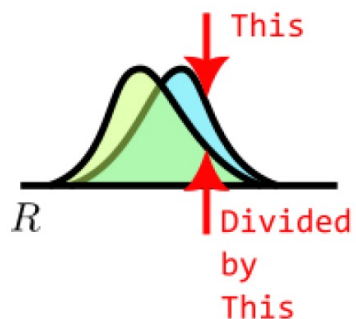


What if we make the distance multiplicative?

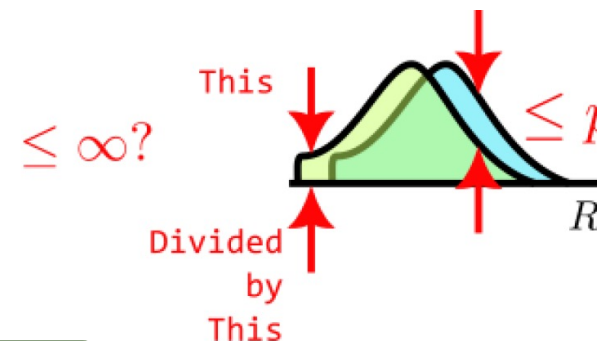
Tentative privacy definition II (with parameter p)

A mechanism M is p -private if the following holds for all possible outputs R and all pairs of neighboring datasets (D, D') :

$$\frac{\Pr(M(D') = R)}{p} < \Pr(M(D) = R) < \Pr(M(D') = R) \cdot p$$



\leq **Q: Does this make sense?**



A: Yes, because this is the case where we get no privacy, and that's what $p = \infty$ means

Finally: Differential Privacy

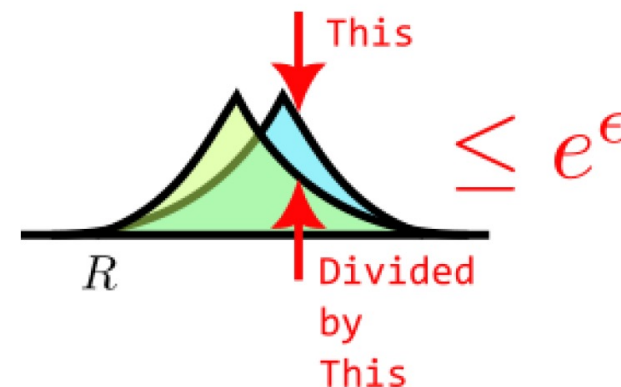
- Same definition, but instead of “ p ” we use e^ϵ

Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible outputs $R \in \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

$$\Pr(M(D) = R) \leq \Pr(M(D') = R) e^\epsilon$$

- Some notes:
 - We use e^ϵ , instead of just ϵ , because this makes it easier to formulate some useful theorems
 - We do not need the $e^{-\epsilon}$ on the left, since this must hold for all pairs (D, D') . This includes (D', D) .
 - $\epsilon \in [0, \infty)$; this ensures that $e^\epsilon \in [1, \infty)$



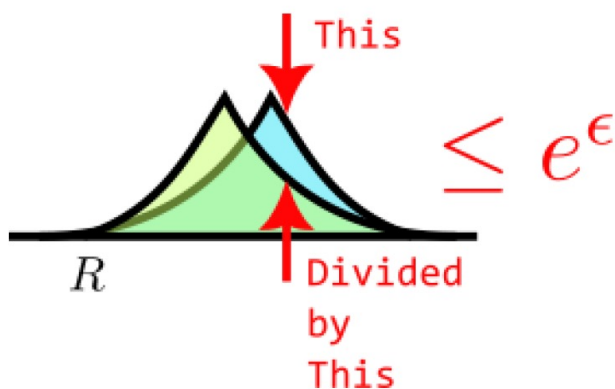
Differential privacy: some questions

Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible sets of outputs $R \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

$$\Pr(M(D) \in R) \leq \Pr(M(D') \in R) e^\epsilon$$

Q: which provides more privacy? $\epsilon = 1$ or $\epsilon = 2$?

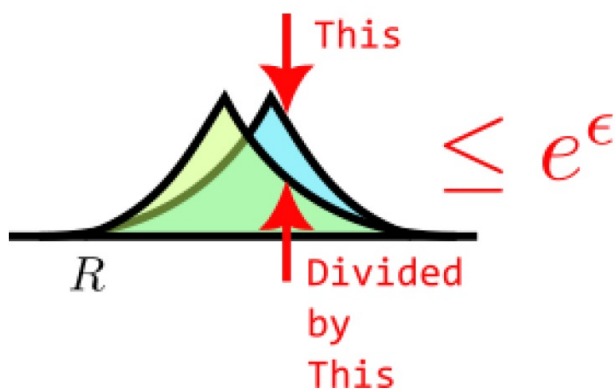


Differential privacy: some questions

Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible sets of outputs $R \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

$$\Pr(M(D) \in R) \leq \Pr(M(D') \in R) e^\epsilon$$



Q: which provides more privacy? $\epsilon = 1$ or $\epsilon = 2$?

A: Smaller ϵ means more privacy; larger means less privacy

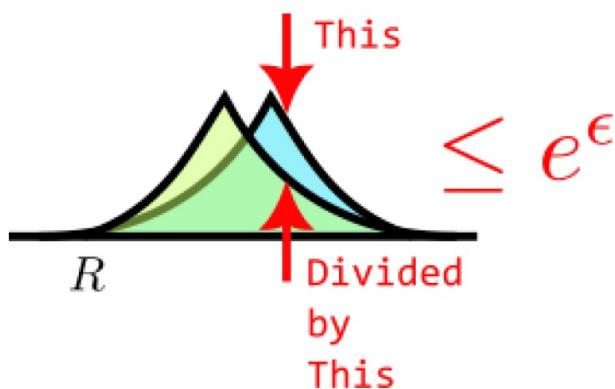
Q: What does $\epsilon = 0$ mean?

Differential privacy: some questions

Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible sets of outputs $R \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

$$\Pr(M(D) \in R) \leq \Pr(M(D') \in R) e^\epsilon$$



Q: which provides more privacy? $\epsilon = 1$ or $\epsilon = 2$?

A: Smaller ϵ means more privacy; larger means less privacy

Q: What does $\epsilon = 0$ mean?

A: Perfect privacy! The output is independent of the dataset! Utility will be very bad.

Some notes on Differential Privacy

- DP was proposed in 2006 by Cynthia Dwork et al. [\[DMNS06\]](#)
- The authors won the Test-of-Time Award in 2016 and the Godel Price in 2017.
- Adopted by big tech like Apple, Google, Microsoft, Facebook, LinkedIn, and by the US Census Bureau for the 2020 US Census
- There is no consensus on how small ϵ should be.

DP Mechanisms

or in other words, how to add noise and how much?

Sensitivity

- Q: How much noise to add? → Measure sensitivity!

Sensitivity

- Q: How much noise to add? → Measure sensitivity!
- Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$, and two neighboring datasets $D \in \mathcal{D}$ and $D' \in \mathcal{D}$, the ℓ_1 -**sensitivity** of f is the maximum change that replacing D for D' can cause in the output:

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

Sensitivity

- **Q:** How much noise to add? → Measure sensitivity!
- Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$, and two neighboring datasets $D \in \mathcal{D}$ and $D' \in \mathcal{D}$, the ℓ_1 -**sensitivity** of f is the maximum change that replacing D for D' can cause in the output:

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

- **Note 1:** The range of f is k -dimensional
 - e.g., Avg. and Sum. of different attributes in a public data release
- **Note 2:** ℓ_1 -sensitivity is the ℓ_1 -norm
 - $\|\vec{x}_1 - \vec{x}_2\|_1 = \sum_i |\vec{x}_1[i] - \vec{x}_2[i]|$

Sensitivity w/ one pair of neighboring databases

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Algorithm: You are allowed to make a query that returns the average score of this course.

Q: What is the ℓ_1 -sensitivity here?

Sensitivity w/ one pair of neighboring databases

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Algorithm: You are allowed to make a query that returns the average score of this course.

Q: What is the ℓ_1 -sensitivity here?

A: $|\text{Avg}(D) - \text{Avg}(D')| = 51.33 - 50 = 1.33$

Sensitivity w/ one pair of neighboring databases

- **D** with Alice **enrolled**:
 - Alice: 90
 - Everyone else (29 of them): 50
- **D'** with Alice **not enrolled**:
 - Everyone (30 of them): 50

Algorithm: You are allowed to make a query that returns the average score of this course.

Q: What is the ℓ_1 -sensitivity here?

A: $|\text{Avg}(D) - \text{Avg}(D')| = 51.33 - 50 = 1.33$

Q: How can we add noise?

DP Mechanisms

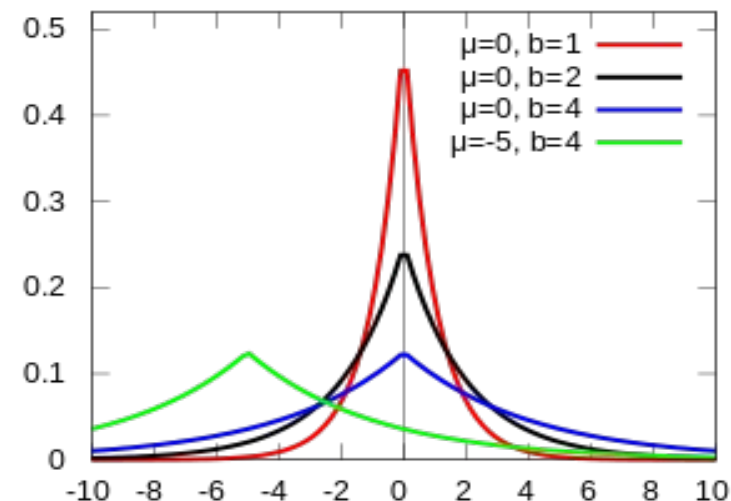
- Multiple mechanisms provide Differential Privacy and can be applied to various systems.
- A few examples:
 1. The Laplace Mechanism (DP, continuous outputs)
 2. The Randomized Response Mechanism (DP, binary inputs/outputs)
 3. General Discrete Mechanisms
 4. The Exponential Mechanism (DP, discrete outputs)
 5. The Gaussian Mechanism (approximate DP, continuous)

DP Mechanisms

- Multiple mechanisms provide Differential Privacy and can be applied to various systems.
- A few examples:
 1. [The Laplace Mechanism \(DP, continuous outputs\)](#)
 2. The Randomized Response Mechanism (DP, binary inputs/outputs)
 3. General Discrete Mechanisms
 4. The Exponential Mechanism (DP, discrete outputs)
 5. The Gaussian Mechanism (approximate DP, continuous)

Example: the Laplacian mechanism

- Let $Y \sim \text{Lap}(\mu, b)$
 - A Laplace distribution!
- With PDF: $p_Y(y) = \frac{1}{2b} e^{-\frac{|y-\mu|}{b}}$
- Usually, for DP, we set $\mu = 0$
 - So you may see $\text{Lap}(b)$ which is essentially $\text{Lap}(0, b)$
- $\text{Lap}(\mu, b)$ has variance $\sigma^2 = 2b^2$
- As b increases, the distribution becomes more flat



The Laplace Mechanism

- Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$, and two neighboring datasets $D \in \mathcal{D}$ and $D' \in \mathcal{D}$, the ℓ_1 -sensitivity of f is the maximum change that replacing D for D' can cause in the output:

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

- Given any function f and its ℓ_1 -sensitivity, we can turn it into a DP mechanism if we add Laplacian noise to its output:

Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$ with ℓ_1 -sensitivity Δ_1 , the **Laplace mechanism** is defined as $M(D) = f(D) + (Y_1, Y_2, \dots, Y_k)$ where each Y_i is independently distributed following $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$.

The Laplace Mechanism

- Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$, and two neighboring datasets $D \in \mathcal{D}$ and $D' \in \mathcal{D}$, the ℓ_1 -sensitivity of f is the maximum change that replacing D for D' can cause in the output:

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

- Given any function f and its ℓ_1 sensitivity, we can turn it into a DP mechanism if we add Laplacian noise to its output:

Given a function $f: \mathcal{D} \rightarrow \mathbb{R}^k$ with ℓ_1 -sensitivity Δ_1 , the **Laplace mechanism** is defined as $M(D) = f(D) + (Y_1, Y_2, \dots, Y_k)$ where Y_i is independently distributed following $Y \sim \text{Lap}(b)$ with $b = \Delta_1 / k$.

The Laplace mechanism provides ϵ -DP

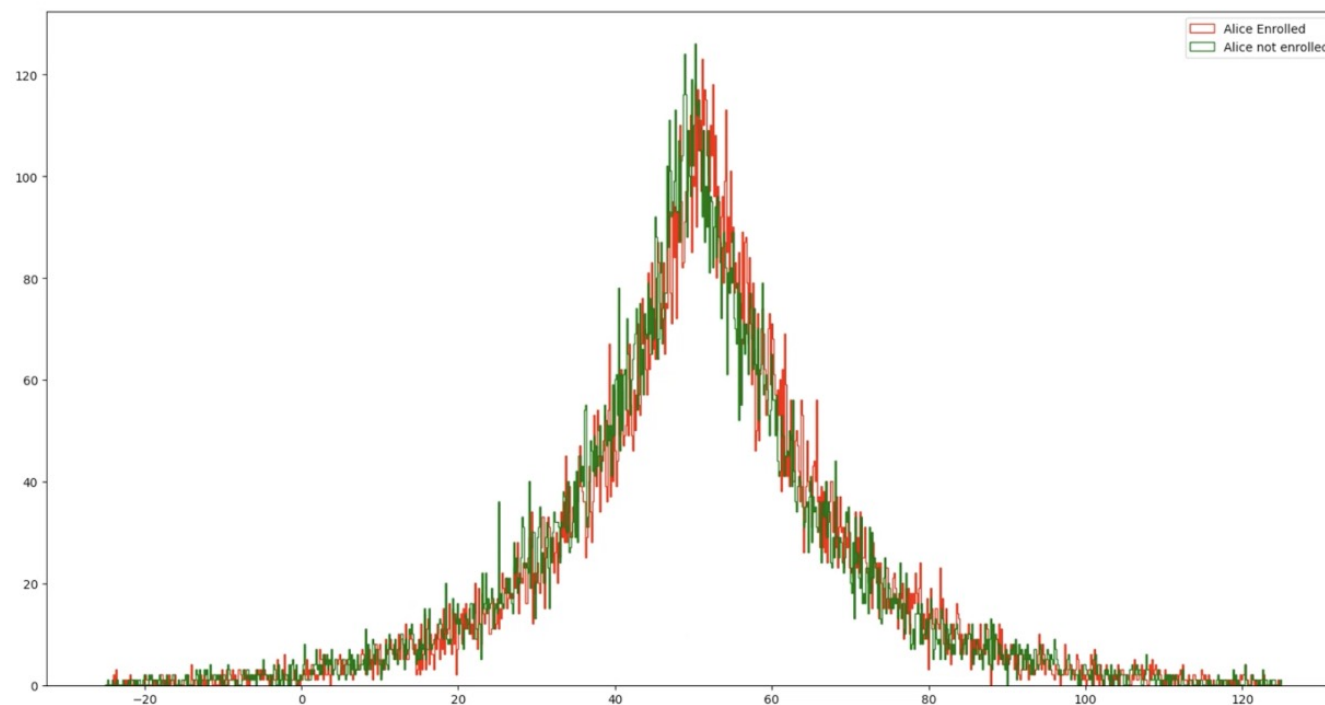
The Laplace Mechanism in our running example

- In our CS489 grades example:
 - let's take $\epsilon = 0.1$, and together with $\Delta_1 = 1.33$, we have:

$$M(D) = f(D) + \text{Lap}(b = \frac{\Delta_1}{\epsilon}) \Leftrightarrow$$

$$\Leftrightarrow M(D) = f(D) + \text{Lap}(\frac{1.33}{0.1}) \Leftrightarrow$$

$$\Leftrightarrow M(D) = f(D) + \text{Lap}(13.3)$$



Curves for D and D' mostly overlap

A Few Other Nice Properties

Compositional privacy

- Given:

- $M_1 : D \rightarrow R_1$ being ϵ_1 -DP, and
- $M_2 : D \rightarrow R_2$ being ϵ_2 -DP

We can define a new mechanism:

$M : D \rightarrow R_1 \times R_2$ as $M(D) = (M_1(D), M_2(D))$.

Then, M is $(\epsilon_1 + \epsilon_2)$ -DP.

- This has a **gossip** analogy:

If A tells you something (potentially with noise), and then B tells you some other things (again, with noise), you may learn more by combining both pieces of information.

One may want to set a **total privacy loss budget** $\epsilon = \epsilon_1 + \epsilon_2 + \dots + \epsilon_n$.

Group privacy

Theorem

Suppose mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private. Suppose D, D' are two neighboring datasets $\in \mathcal{D}$ which differ in exactly k positions. Then:

$$\Pr(M(D) = R) \leq \Pr(M(D') = R) e^{k\epsilon}$$

- **TLDR:** If you need to hide the “effects” caused by a whole group of records, you need to prepare a larger privacy budget.

Approximate DP

- The following is a relaxation of the DP definition, that allows some tolerance:

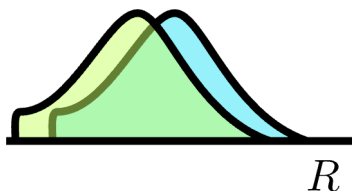
(Approximate) Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is (ϵ, δ) -differentially private (ϵ, δ) -DP if the following holds for all sets of possible outputs $S \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

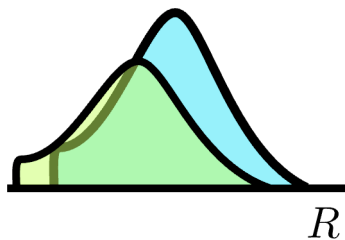
$$\Pr(M(D) \in S) \leq \Pr(M(D') \in S) e^\epsilon + \delta$$

- When $\delta = 0$, this is the same as ϵ -DP (called pure DP).
- What does this mean?

We have two distributions
 $f(R|D)$ vs $f(R|D')$



We multiply one
(e.g., blue) by e^ϵ



The area of the green one not covered by
the blue one now will be $\leq \delta$



Approximate DP: interpretation

(Approximate) Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is (ϵ, δ) -differentially private $((\epsilon, \delta)$ -DP) if the following holds for all sets of possible outputs $S \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

$$\Pr(M(D) \in S) \leq \Pr(M(D') \in S) e^\epsilon + \delta$$

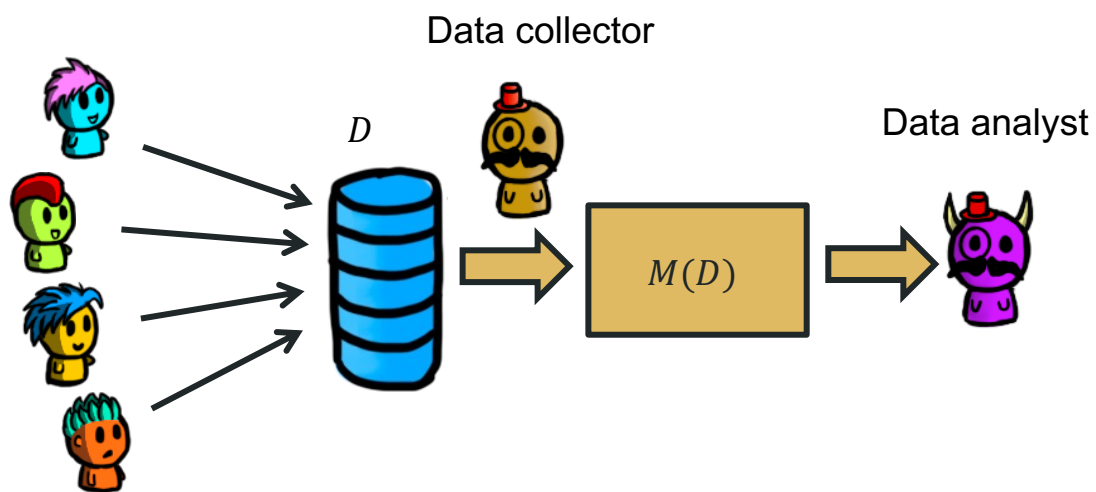
- A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ that provides ϵ -DP except for certain "bad" outcomes $B \subset \mathcal{R}$, where $\Pr(M(D) \in B) \leq \delta$ (for any $D \in \mathcal{D}$) also provides (ϵ, δ) -DP.
- This definition allows us to add less noise, if we are comfortable with the probability of bad outcomes

A Note on Differential Privacy Settings

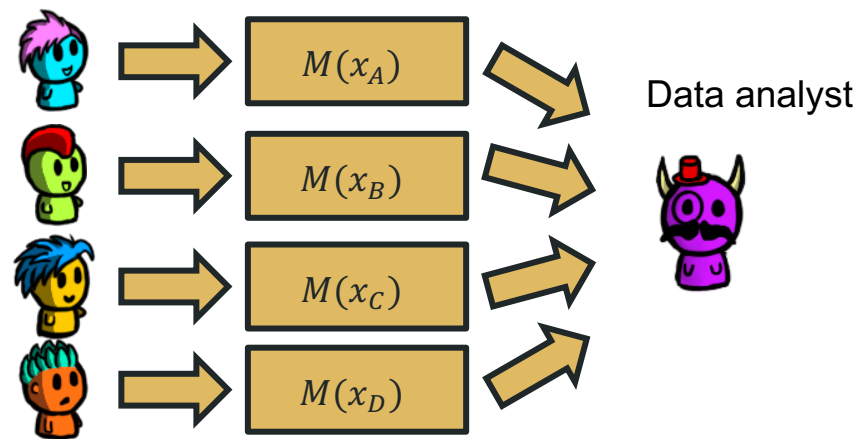
Central DP vs. Local DP

- Depending on who runs the mechanism, there are two broad models for differential privacy.

Central Differential Privacy: there is a centralized (trusted) aggregator



Local Differential Privacy: each user runs the mechanism themselves and reports the result to the adversary/analyst

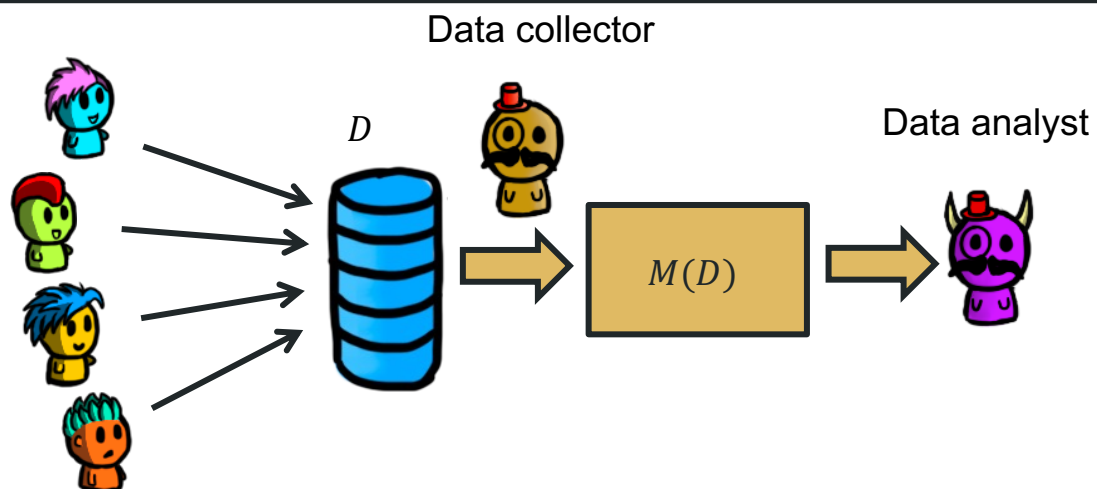


Central DP vs. Local DP

(Central) Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible sets of outputs $R \subset \mathcal{R}$ and all pairs of neighboring datasets $D, D' \in \mathcal{D}$:

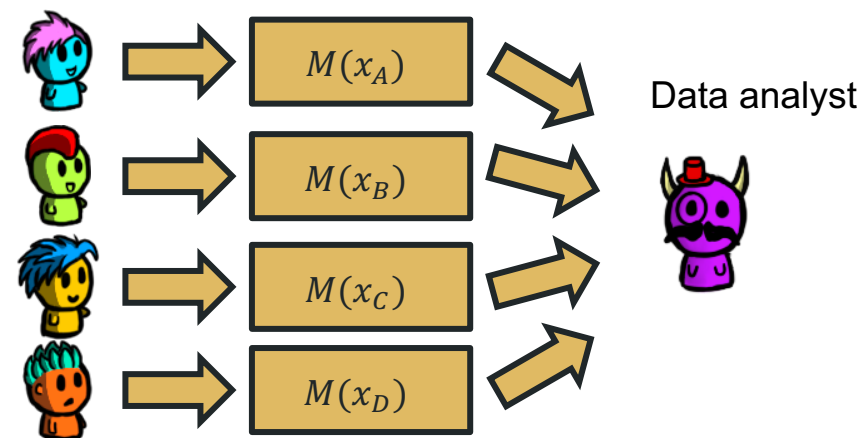
$$\Pr(M(D) \in R) \leq \Pr(M(D') \in R) e^\epsilon$$



(Local) Differential Privacy

A mechanism $M: \mathcal{D} \rightarrow \mathcal{R}$ is ϵ -differentially private (ϵ -DP) if the following holds for all possible sets of outputs $R \subset \mathcal{R}$ and all pairs of neighboring inputs $x, x' \in \mathcal{D}$:

$$\Pr(M(x) \in R) \leq \Pr(M(x') \in R) e^\epsilon$$



- They are “the same definition”, it’s just that the inputs to the mechanism and what we define as “neighbouring” inputs/datasets is usually different.

Central DP vs. Local DP

- Central DP

- Best accuracy, aggregation allows to hide in the crowd before we add noise.
- Need to trust the data collector.
- Hard to verify if noise was added.

- Local DP

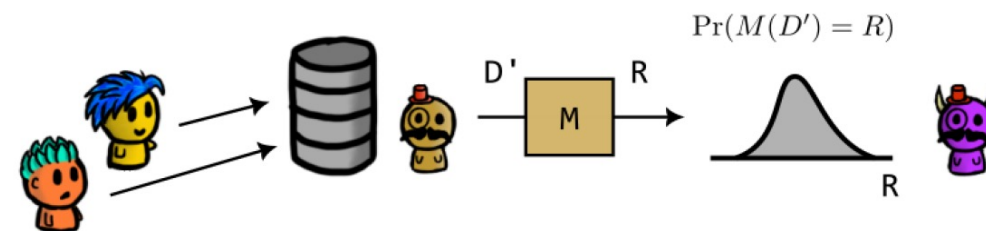
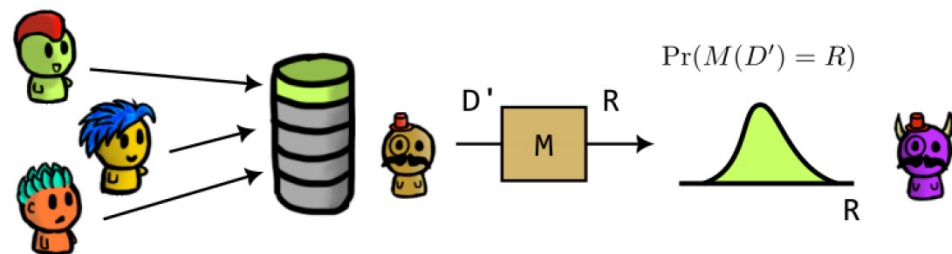
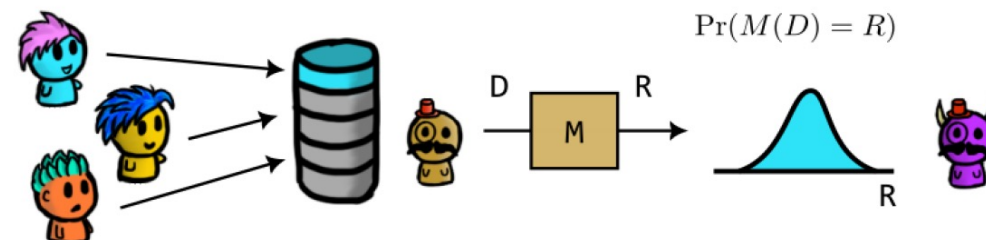
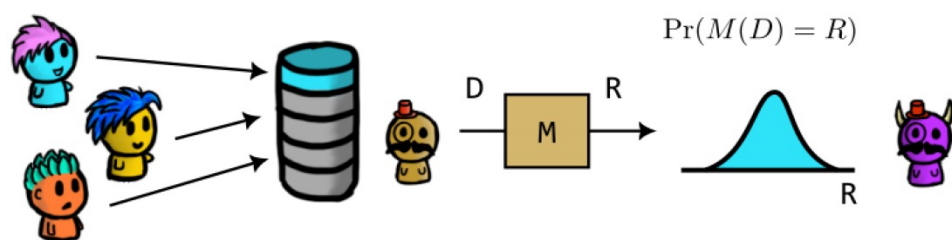
- Accuracy not as good. Each user adds noise which can compound in the final result.
- User doesn't need to trust anybody and knows they added noise.

Bounded DP vs. Unbounded DP

- There are two “main” definitions for how we define neighboring datasets in the central model.

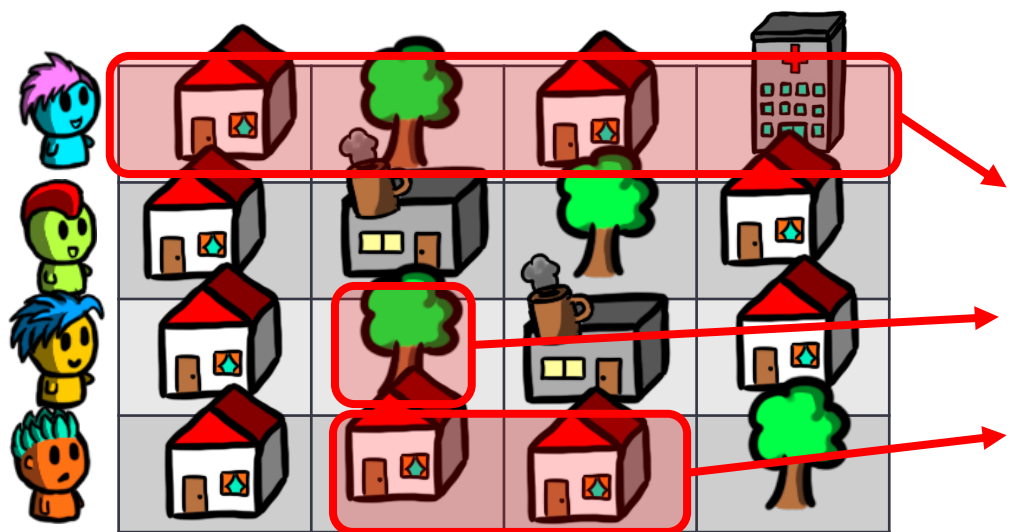
Bounded DP: D and D' have the same number of entries but differ in the value of one.

Unbounded DP: D and D' are such that you get one by deleting an entry from the other one.



Other notions of DP

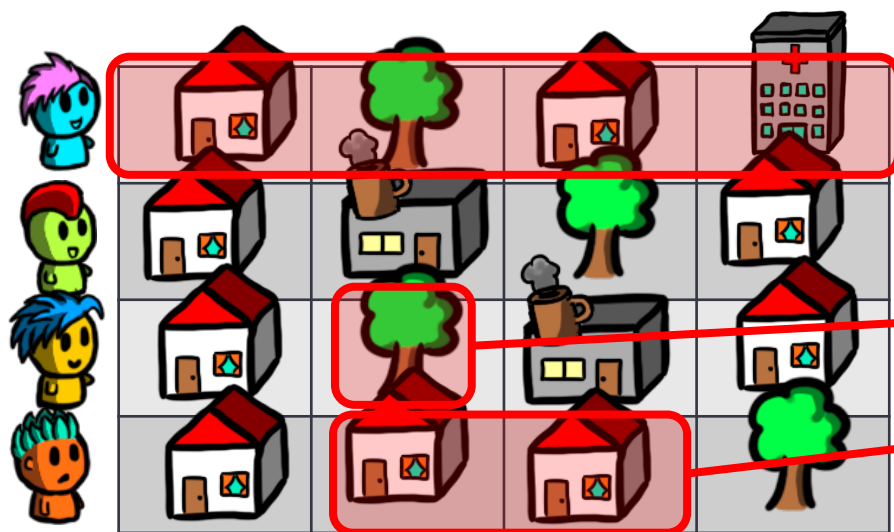
- Many possible neighbouring definitions.
- For example, in location privacy:



Depending on how we define neighboring datasets D and D' , we get a different DP guarantee:

- **User-level DP:** we replace a user trajectory for another user's trajectory
 - **Event-level DP:** we replace the location of a user for another location
 - **w-event DP:** we replace a window of w consecutive locations of a user for another
- These are all DP and have their uses. It is important to understand, for each system/application, which notion of DP it provides.

Other notions of DP - question

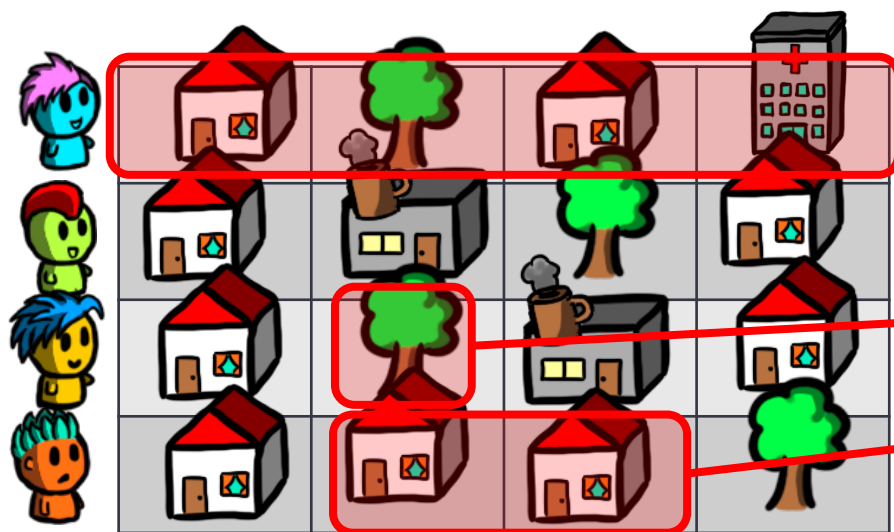


Depending on how we define neighboring datasets D and D' , we get a different DP guarantee:

- **User-level DP:** we replace a user trajectory for another user's trajectory
- **Event-level DP:** we replace the location of a user for another location
- **w-event DP:** we replace a window of w consecutive locations of a user for another

Q: Which notions of DP imply the others?

Other notions of DP - question



Depending on how we define neighboring datasets D and D' , we get a different DP guarantee:

- **User-level DP:** we replace a user trajectory for another user's trajectory
- **Event-level DP:** we replace the location of a user for another location
- **w-event DP:** we replace a window of w consecutive locations of a user for another

Q: Which notions of DP imply the others?

A: User-level DP implies Event- and w-event DP.
W-event DP implies Event-level DP.

A lot more about differential privacy!

- You may want to check CS860 (F'20) – Algorithms for Private Data Analysis, as taught by Prof. Gautam Kamath here at the School.
- The course's contents are available online!
 - https://www.youtube.com/playlist?list=PLmd_zeMNzSvRRNpoEWkVo6QY_6rR3SHjp


Checkpoint on the Laplace Mechanism

(self-study)

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: what does smaller ϵ mean?

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim Lap(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: what does smaller ϵ mean?

A: more privacy

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: if we want more privacy, would we need to add more or less noise?

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: if we want more privacy, would we need to add more or less noise?

A: more noise. That's why $b \propto \frac{1}{\epsilon}$.

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: if changing D for D' can cause a huge change in $f(\cdot)$, is that a large or small sensitivity?

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!




Q: if changing D for D' can cause a huge change in $f(\cdot)$, is that a large or small sensitivity?

A: large sensitivity

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!



Q: if changing D for D' can have a huge impact in f , do we need a lot or a little noise to hide this impact?

The Laplace Mechanism – checkpoint!

The Laplace Mechanism: $M(D) = f(D) + Y$ where $Y \sim \text{Lap}(b)$ with $b = \frac{\Delta_1}{\epsilon}$ provides ϵ -DP

The variance is $2b^2$; higher b means more noise!

Q: if changing D for D' can have a huge impact in f , do we need a lot or a little noise to hide this impact?

A: a lot of noise.
That's why $b \propto \Delta_1$

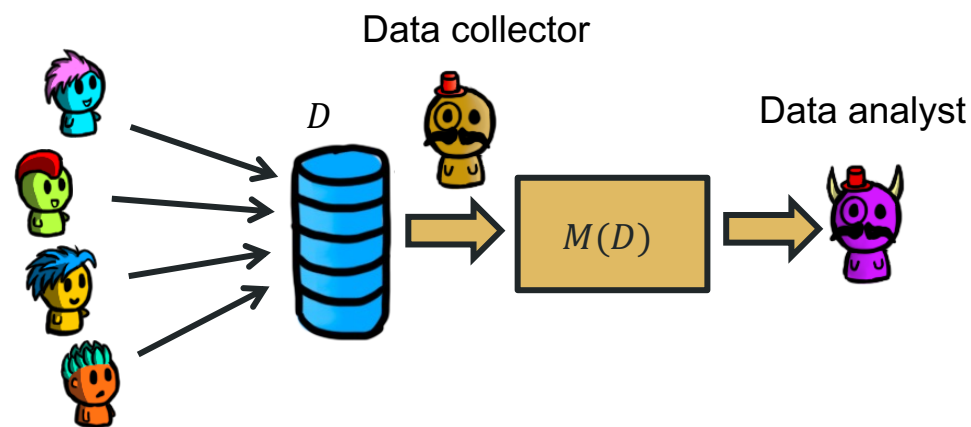
Laplace Mechanism: examples

Example 1: D contains the test results for virus X of a set of users. We want to release the total number of users that tested positive. How do we make this ϵ -DP?

- Under unbounded DP
- Under bounded DP

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



Laplace Mechanism: examples

Example 1: D contains the test results for virus X of a set of users. We want to release the total number of users that tested positive. How do we make this ϵ -DP?

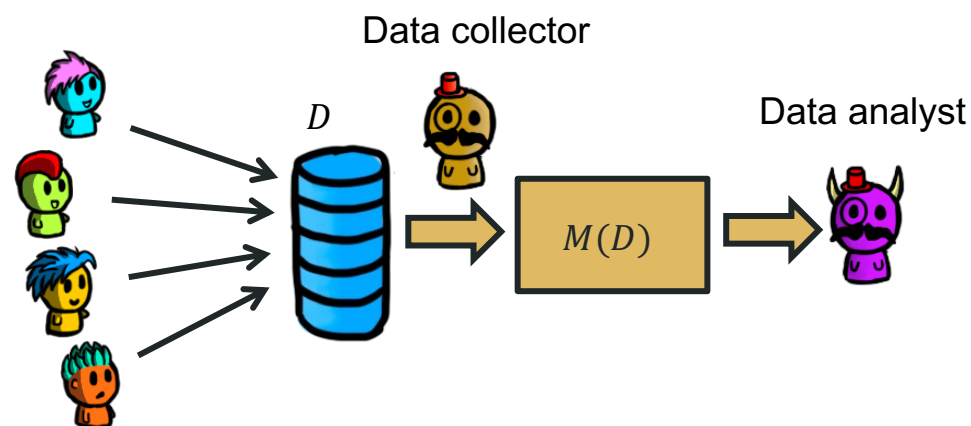
- Under unbounded DP
- Under bounded DP

A: sensitivity is 1 in both cases

Add $Y \sim \text{Lap}\left(\frac{1}{\epsilon}\right)$

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



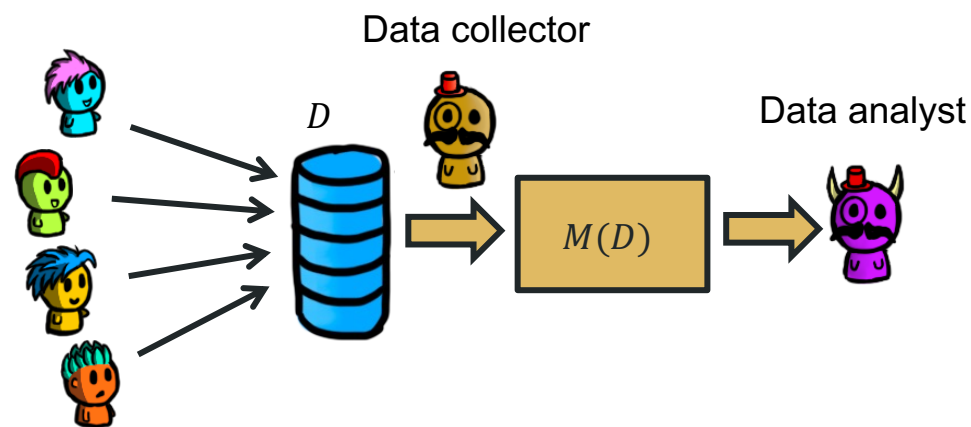
Laplace Mechanism: examples

Example 2: D contains the salaries of a set of users. The salaries range from 20k to 200k. We want to release the **total** salary of the users. How do we make this ϵ -DP?

- Under unbounded DP
- Under bounded DP

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



Laplace Mechanism: examples

Example 2: D contains the salaries of a set of users. The salaries range from 20k to 200k. We want to release the **total** salary of the users. How do we make this ϵ -DP?

- Under unbounded DP
- Under bounded DP

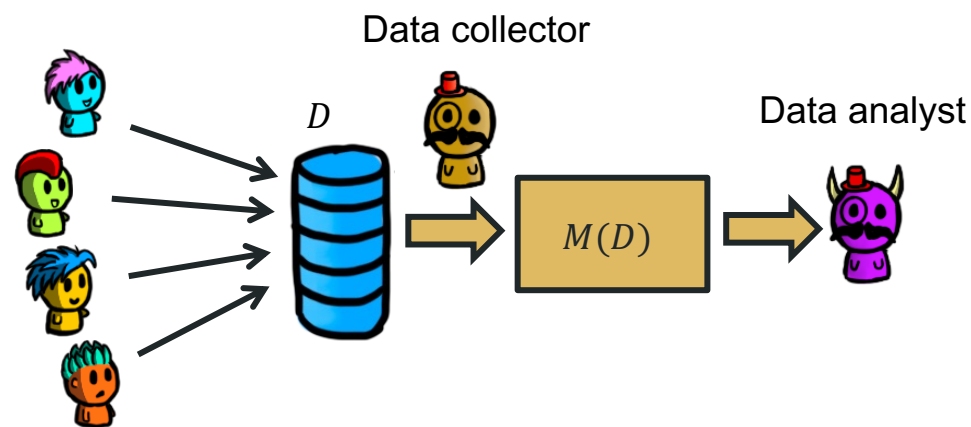
A: sensitivity is bounded by 180k in bounded DP and 200k in unbounded DP

Add $Y \sim \text{Lap}\left(\frac{180k}{\epsilon}\right)$ or

$$Y \sim \text{Lap}\left(\frac{200k}{\epsilon}\right)$$

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



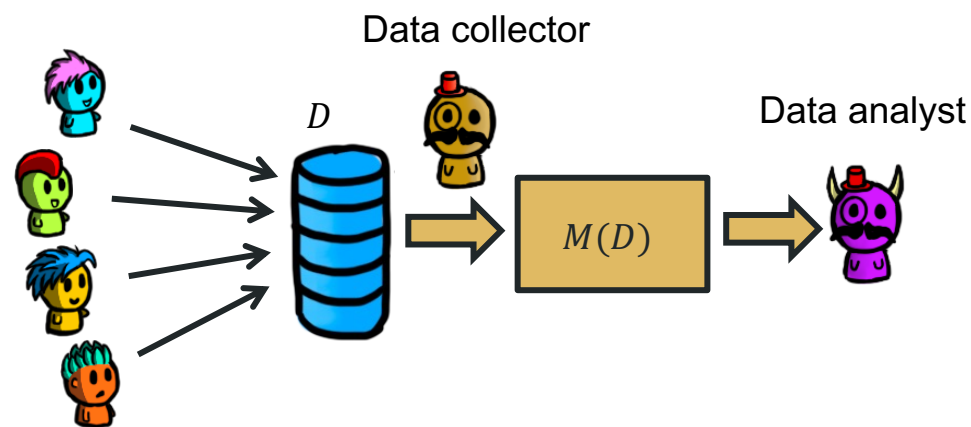
Laplace Mechanism: examples

Example 3: D contains the salaries of n users (n is public knowledge). The salaries range from 20k to 200k. We want to release the **average** salary of users. How do we make this ϵ -DP?

- Under bounded DP

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



Laplace Mechanism: examples

Example 3: D contains the salaries of n users (n is public knowledge). The salaries range from 20k to 200k. We want to release the **average** salary of users. How do we make this ϵ -DP?

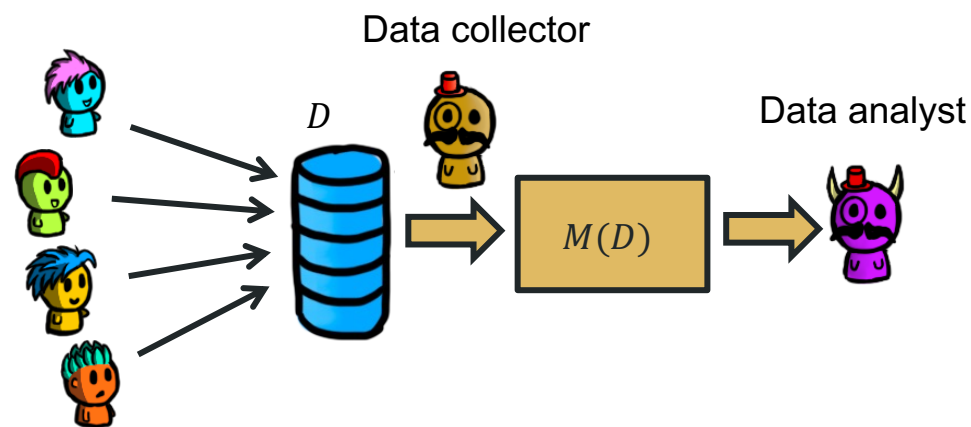
- Under bounded DP

A: sensitivity is bounded by $180k/n$

Add $Y \sim \text{Lap}\left(\frac{180k}{n\epsilon}\right)$

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



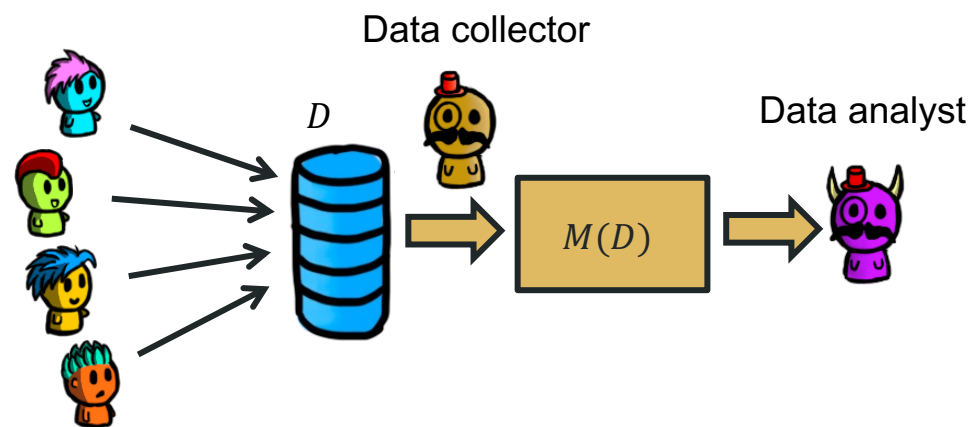
Laplace Mechanism: examples

Example 4: D contains the age of a set of users. We want to release the histogram of ages $[0-10)$, $[10-20)$... $[100,110)$. How do we make this ϵ -DP?

- Under unbounded DP
- Under bounded DP

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



Laplace Mechanism: examples

Example 4: D contains the age of a set of users. We want to release the histogram of ages $[0-10)$, $[10-20)$... $[100,110)$. How do we make this ϵ -DP?

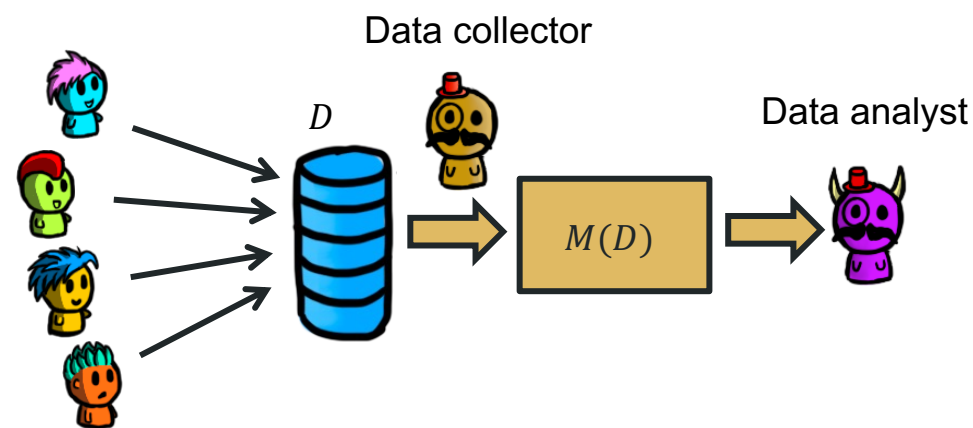
- Under unbounded DP
- Under bounded DP

A: sensitivity is 1 in unbounded 2 in bounded

Add $Y \sim \text{Lap}\left(\frac{1}{\epsilon}\right)$ or $Y \sim \text{Lap}\left(\frac{2}{\epsilon}\right)$ to each bucket in the histogram (drawn fresh for each bucket)

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$

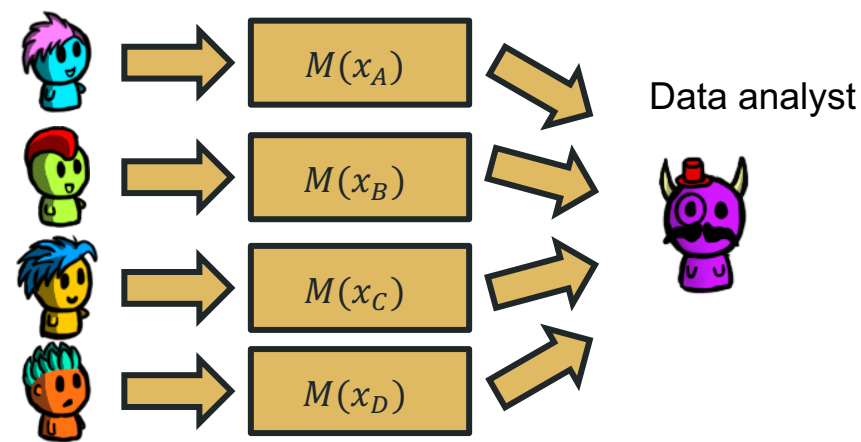


Laplace Mechanism: examples

Example 5: Alice wishes to report her age x_A in a differentially private way. It is public information that she is between 18 and 100 years old. She adds Laplacian noise with $b = 3$ to her age, and reports the resulting value. What is the level of DP that she gets?

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if } Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$



Laplace Mechanism: examples

Example 5: Alice wishes to report her age x_A in a differentially private way. It is public information that she is between 18 and 100 years old. She adds Laplacian noise with $b = 3$ to her age, and reports the resulting value. What is the level of DP that she gets?

A: sensitivity is bounded by 82

$$b = \frac{82}{\epsilon} = 3$$
$$\epsilon = 82/3$$

$$\Delta_1 \doteq \max_{D, D'} \|f(D) - f(D')\|_1$$

$$f(D) + Y \text{ is } \epsilon\text{-DP if}$$
$$Y \sim \text{Lap}\left(\frac{\Delta_1}{\epsilon}\right)$$

