Differential Privacy

CompSci 590.03
Instructor: Ashwin Machanavajjhala

Duke
UNIVERSITY

Announcements

- Please meet with me at least once before you finalize your project (deadline Sep 27).

Outline

- Differential privacy
- Algorithms
  - Deterministic algorithms & Sampling
  - Randomized Response
- Laplace mechanism
- Designing complex algorithms
  - Sequential Composition
  - Parallel Composition

Duke UNIVERSITY

Differential Privacy

An algorithm A satisfies ε-differential privacy if:

For **every pair** of *neighboring tables* D_1, D_2(differ in one individual)

For **every output** O

$$\Pr[A(D_1) = O] \leq e^{\varepsilon} \Pr[A(D_2) = O]$$

Neighboring Tables

An algorithm A satisfies ε-differential privacy if:

For **every pair** of *neighboring tables* D_1, D_2(differ in one individual)

For **every output** O

$$\Pr[A(D_1) = O] \leq e^{\varepsilon} \Pr[A(D_2) = O]$$

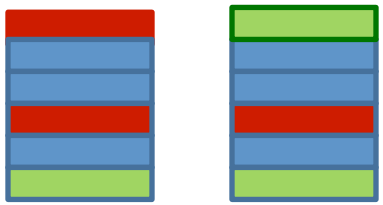
Neighboring tables differ in:

- Presence or absence of a tuple – $D_2 = D_1 - \{t\}$
- Value of a tuple – $D_1 = D \cup \{x\}$ and $D2 = D \cup \{y\}$

Duke UNIVERSITY

Why 2 neighboring tables?

Set of all possible input databases

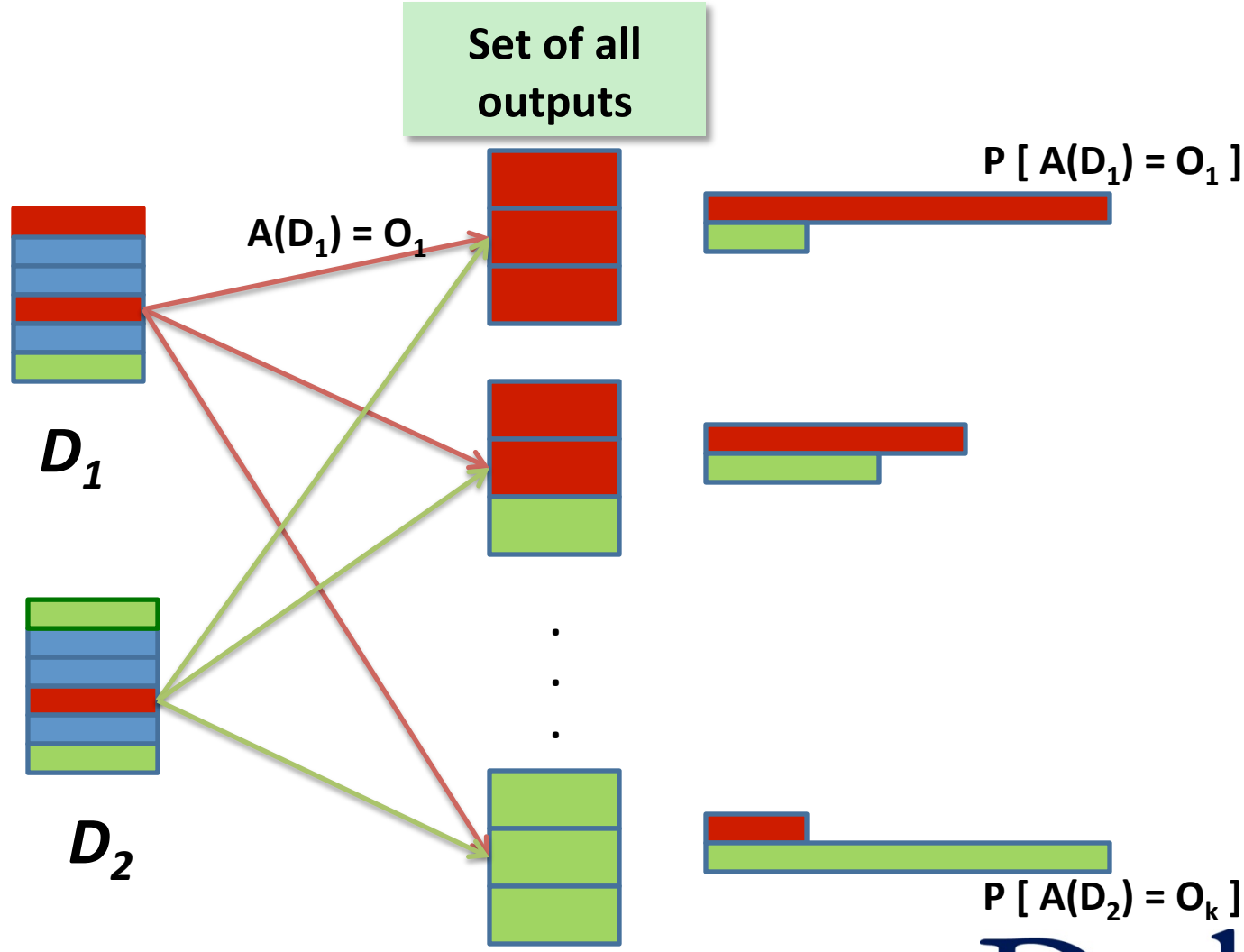


Adversary knows x_1 is **either green or red**.

Adversary knows exact values for $\{x_2, x_3, \ldots, x_n\}$

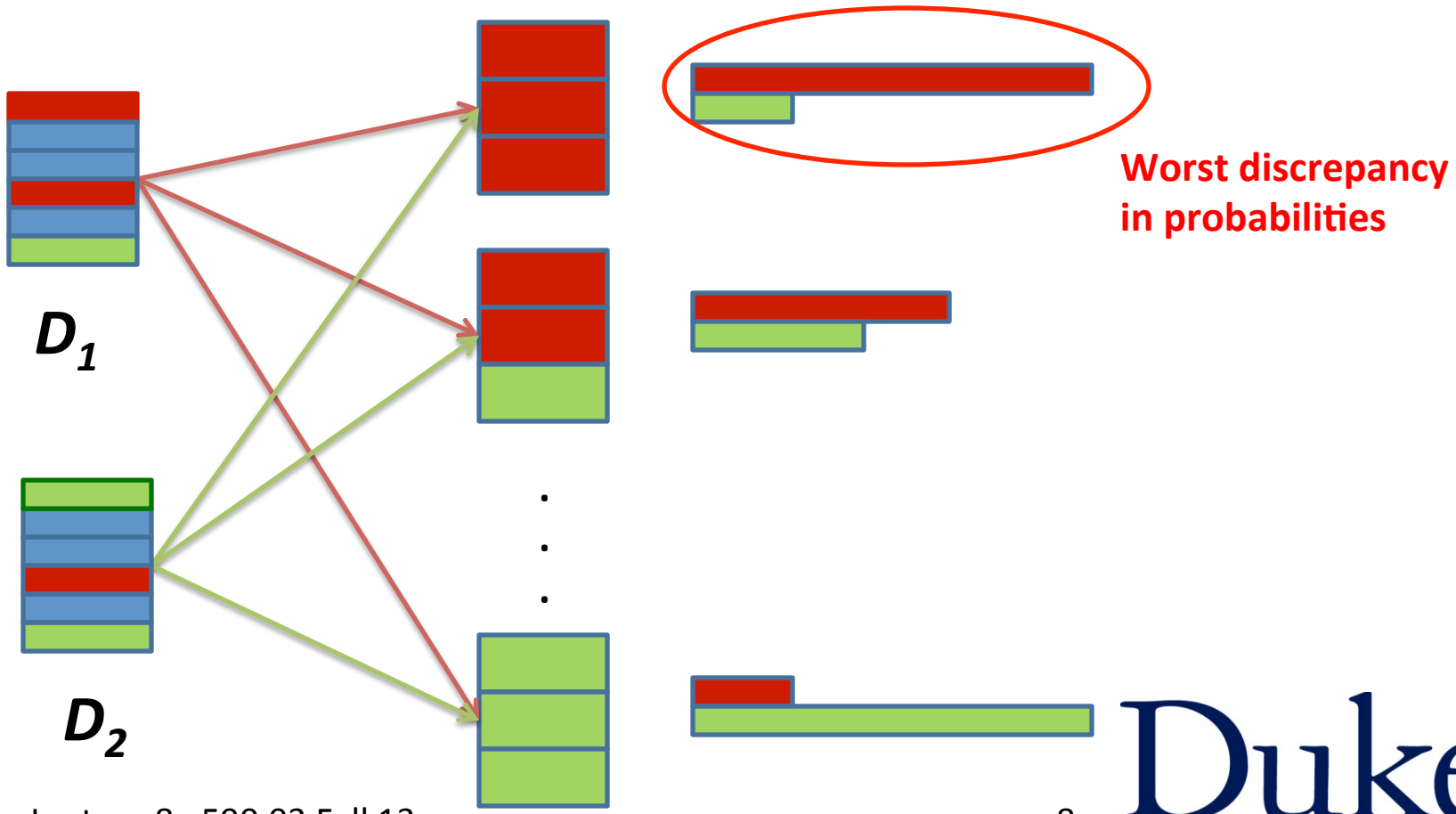
blue, green and red are three possibilities for each x_i.

Why all outputs?



Why all outputs?

For every pair of tables D_1 and D_2,
adversary should not be able to distinguish between D_1 and D_2.



Privacy parameter ε

An algorithm A satisfies ε-differential privacy if:

For **every pair** of *neighboring tables* D_1, D_2(differ in one individual)

For **every output** O

$$\Pr[A(D_1) = O] \leq e^{\varepsilon} \Pr[A(D_2) = O]$$

- Controls the amount of information disclosed to the adversary
- Smaller the ε more the privacy (and better the utility)

Duke UNIVERSITY

Differential Privacy: Summary

An algorithm A satisfies ε-differential privacy if:

For **every pair** of *neighboring tables* D_1, D_2(differ in one individual)

For **every output** O

$$\Pr[A(D_1) = O] \leq e^{\varepsilon} \Pr[A(D_2) = O]$$

what the adversary learns about an individual is the same even if the individual is not in the data (or lied about his/her value)

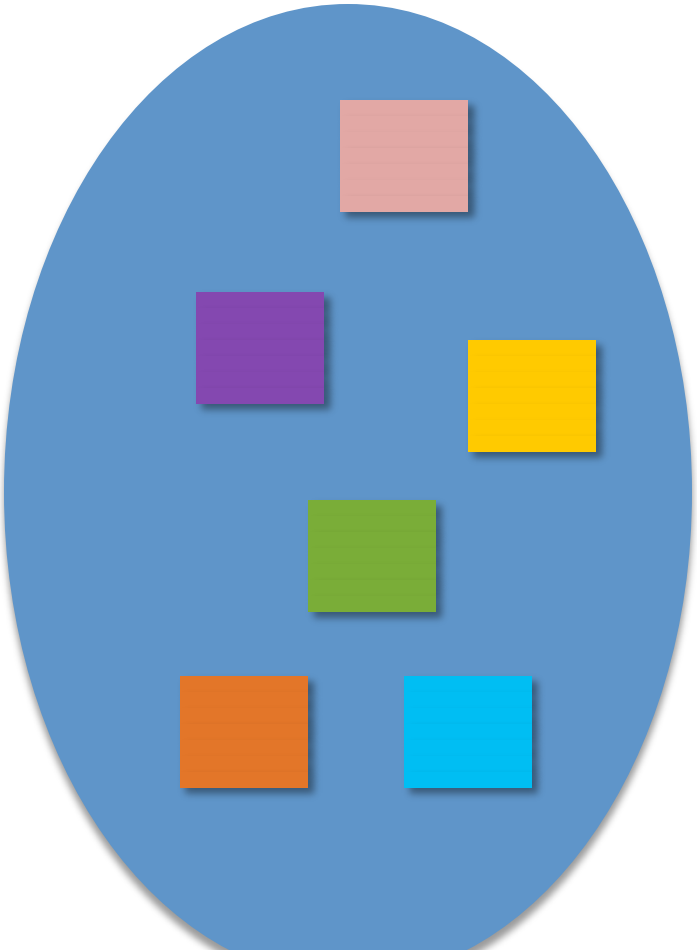
Outline

- Differential privacy
- Algorithms
  - Deterministic algorithms & Sampling
  - Randomized Response
- Laplace mechanism
- Designing complex algorithms
  - Sequential Composition
  - Parallel Composition

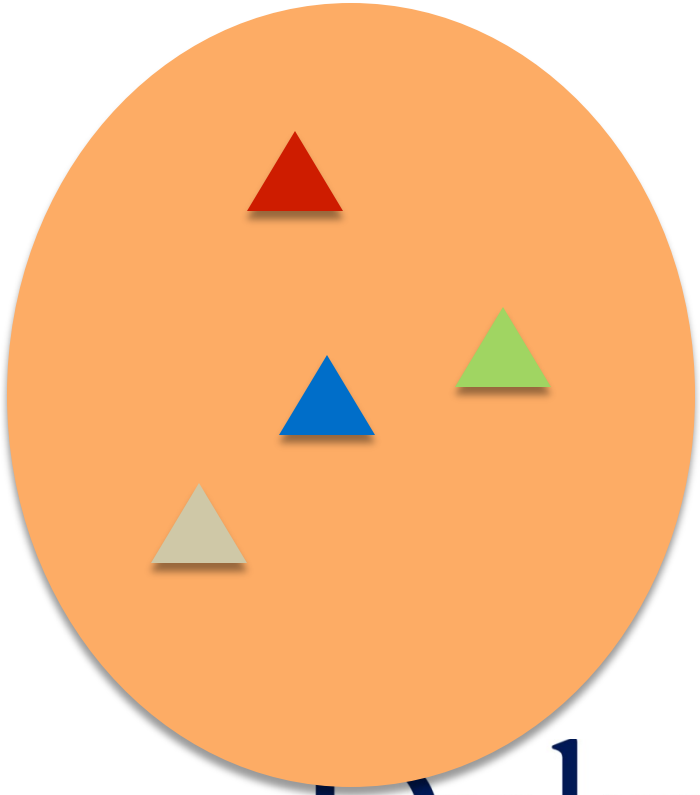
Can deterministic algorithms satisfy differential privacy?

Deterministic Algorithms do not satisfy differential privacy

Space of all inputs

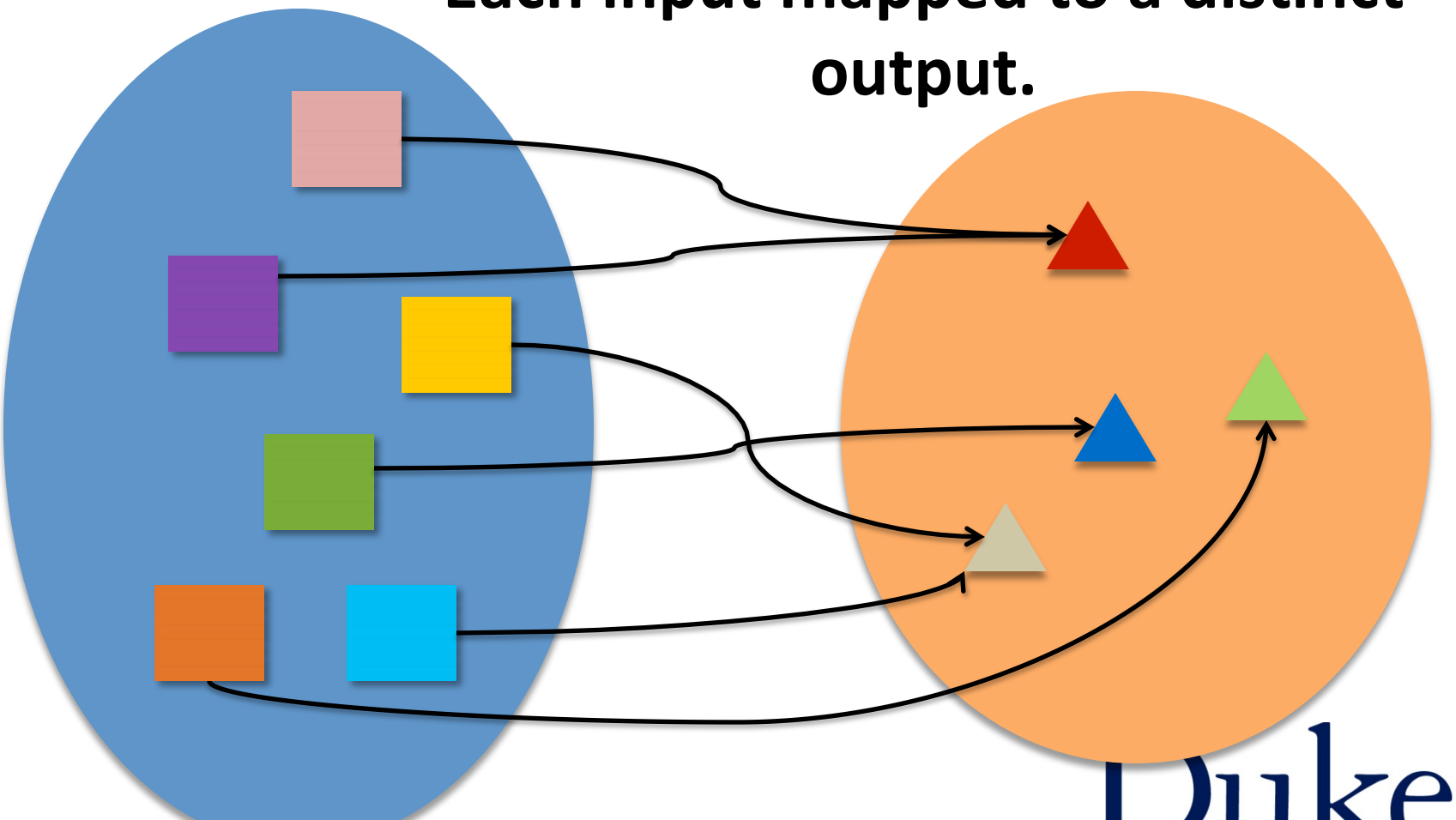


Space of all outputs (at least 2 distinct ouputs)

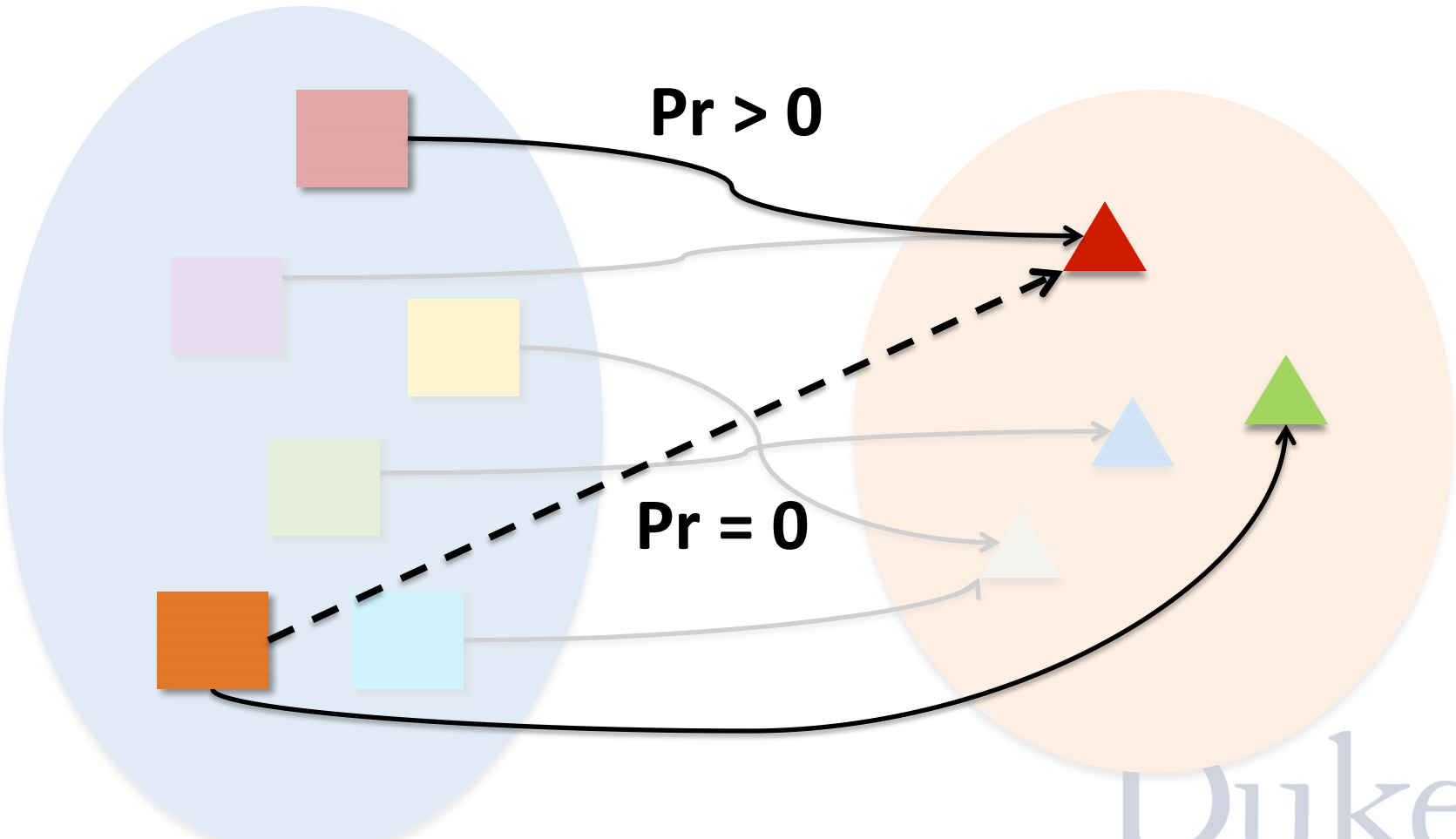


Deterministic Algorithms do not satisfy differential privacy

Each input mapped to a distinct output.

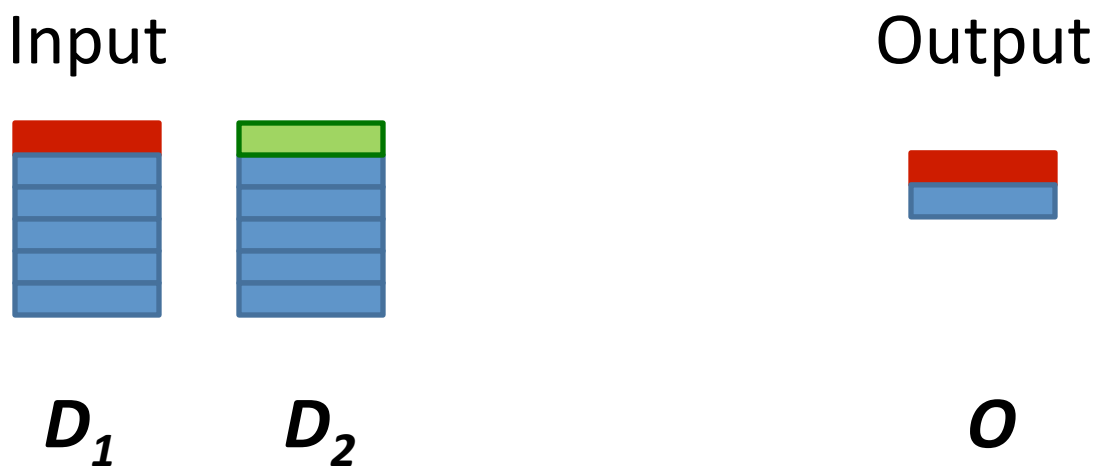


There exist two inputs that differ in one entry mapped to different outputs.



Random Sampling

- Also does not satisfy differential privacy



$$\Pr[D_2 \rightarrow O] = 0 \quad \textbf{implies} \quad \log\left(\frac{\Pr[D_1 \rightarrow O]}{\Pr[D_2 \rightarrow O]}\right) = \infty$$

Random Sampling

- Also does not satisfy differential privacy

[Chauduri et al., 2006]

- **If uniques are rare**, then differential privacy can be guaranteed with high probability.

Most interesting data have many uniques!

Duke UNIVERSITY

Randomized Response (a.k.a. local randomization)

D

Disease (Y/N)
Y
Y
N
Y
N
N

With probability p,
Report true value

With probability 1-p,
Report flipped value

O

Disease (Y/N)
Y
N
N
N
Y
N

Differential Privacy Analysis

- Consider 2 databases D, D' (of size M) that differ in the jth value
  - D[j] ≠ D'[j]. But, D[i] = D'[i], for all i ≠ j

- Consider some output O

$$\frac{P(D \to O)}{P(D' \to O)} = \frac{\prod_{i=1}^{M} P(D[i] \to O[i])}{\prod_{i=1}^{M} P(D'^{[i]} \to O[i])}$$

$$= \frac{P(D[j] \to O[j]) \cancel{\prod_{i \neq j} P(D[i] \to O[i])}}{P(D'^{[j]} \to O[j]) \cancel{\prod_{i \neq j} P(D'^{[i]} \to O[i])}}$$

Differential Privacy Analysis

- Consider 2 databases D, D' (of size M) that differ in the jth value
  - D[j] ≠ D'[j]. But, D[i] = D'[i], for all i ≠ j

- Consider some output O

$$\frac{P(D \rightarrow O)}{P(D' \rightarrow O)} = \frac{P(D[j] \rightarrow O[j])}{P(D'^{[j]} \rightarrow O[j])}$$

$$= \begin{cases} \dfrac{p}{1-p} & if\ D[j] = O[j] \\ \dfrac{1-p}{p} & if\ D'[j] = O[j] \end{cases} \leq e^{\varepsilon}$$

Duke UNIVERSITY

Differential Privacy Analysis

- Consider 2 databases D, D' (of size M) that differ in the jth value
  - D[j] ≠ D'[j]. But, D[i] = D'[i], for all i ≠ j

- Consider some output O

$$\frac{P(D \rightarrow O)}{P(D' \rightarrow O)} \leq e^{\varepsilon} \Leftrightarrow \frac{1}{1+e^{\varepsilon}} < p < \frac{e^{\varepsilon}}{1+e^{\varepsilon}}$$

Utility Analysis

- Suppose n1 out of N people replied "yes", and rest said "no"
- What is the best estimate for π ?

$$\pi_{hat} = \{n1/n - (1-p)\}/(2p-1)$$

- $E(\pi_{hat}) = \pi$
- $Var(\pi_{hat}) = \dfrac{\pi(1-\pi)}{n} + \dfrac{1}{n(16(p-0.5)^2 - 0.25)}$

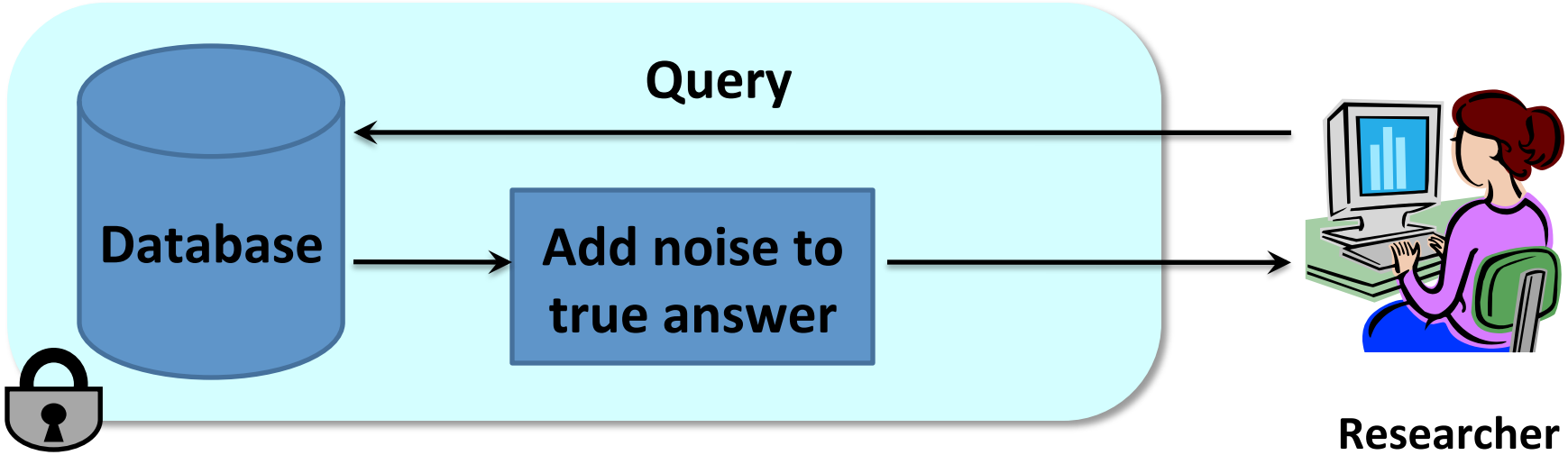
Sampling **Variance due to coin flips**

Duke UNIVERSITY

Outline

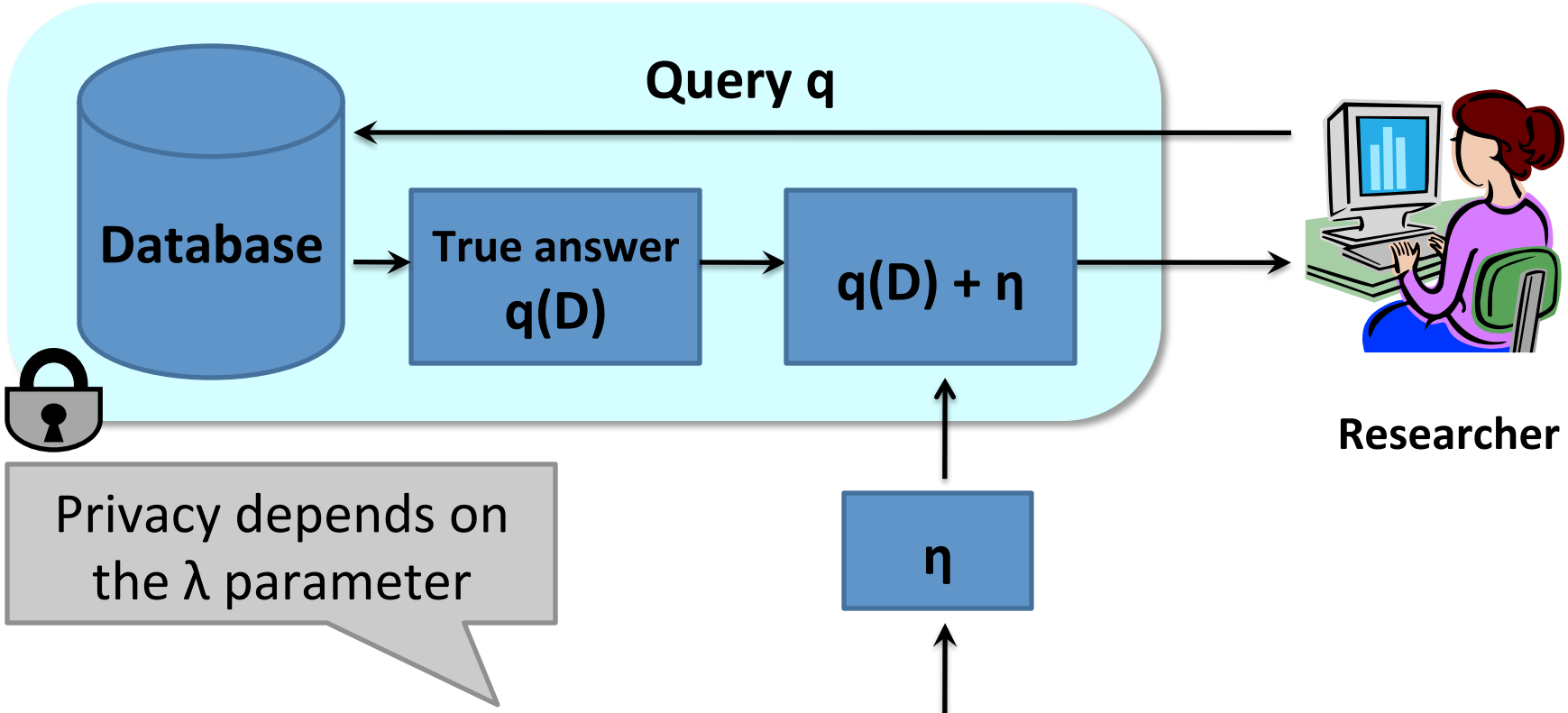
- Differential privacy
- Algorithms
  - Deterministic algorithms & Sampling
  - Randomized Response
- Laplace mechanism
- Designing complex algorithms
  - Sequential Composition
  - Parallel Composition

Output Randomization



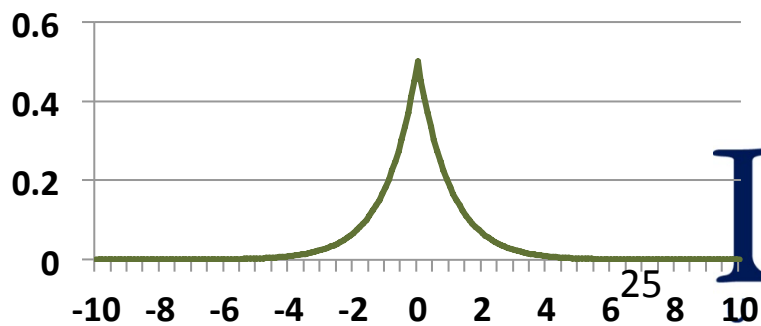
- Add noise to answers such that:
  - Each answer does not leak too much information about the database.
  - Noisy answers are close to the original answers.

Laplace Mechanism



$h(\eta) \propto \exp(-\eta / \lambda)$

Laplace Distribution – Lap(λ)



Mean: 0,
Variance: $2\lambda^2$

How much noise for privacy?

[Dwork et al., TCC 2006]

Sensitivity: Consider a function q: *I* → R. *(call q a query)*
S(q) is the smallest number s.t. for any neighboring tables D, D',

$$| q(D) - q(D') | \leq S(q)$$

Thm: If **sensitivity** of the query is **S**, then the following guarantees ε-differential privacy.

$$\lambda = S/\varepsilon$$

Sensitivity: COUNT query

- Number of people having disease
- Sensitivity = 1

- Solution: $3 + \eta$,
  where η is drawn from $Lap(1/\varepsilon)$
  - Mean = 0
  - Variance = $2/\varepsilon^2$
    (smaller than case of local randomization)

D

Disease (Y/N)
Y
Y
N
Y
N
N

Duke UNIVERSITY

Sensitivity: SUM query

- Suppose all values x are in [a,b]
- Sensitivity depends on the definition of neighbors
- Neighbors D and D U {t}
  Sensitivity = b
- Neighbors D U {x}, and D U {y}
  Sensitivity = (b-a)

Privacy of Laplace Mechanism

- Consider any two neighboring databases D and D'
- Consider some output O

$$\frac{\Pr\left[A(D) = O\right]}{\Pr\left[A(D') = O\right]} = \frac{\Pr\left[q(D) + \eta = O\right]}{\Pr\left[q(D') + \eta = O\right]}$$

$$= \frac{e^{-|O-q(D)|/\lambda}}{e^{-|O-q(D')|/\lambda}}$$

$$\leq e^{|q(D)-q(D')|/\lambda} \leq e^{S(q)/\lambda} = e^{\varepsilon}$$

Utility of Laplace Mechanism

- Laplace mechanism works for **any function** that returns a real number

- Error: E(true answer – noisy answer)^2

$$= \mathrm{Var}(\ \mathrm{Lap}(S(q)/\varepsilon)\ )$$

$$= 2*S(q)^2 / \varepsilon^2$$

- What about sensitivities of more complex functions that return real numbers?

Outline

- Differential privacy
- Algorithms
  - Deterministic algorithms & Sampling
  - Randomized Response
- Laplace mechanism
- Designing complex algorithms
  - Sequential Composition
  - Parallel Composition

HISTOGRAM query

- Each individual in the database takes one out of c categories
  - E.g., {Diabetes, Cancer, Flu, Healthy}
- Return number of people in each category
  - E.g., 4 diabetes, 3 cancer, 2 flu, 3 healthy

- How to answer the histogram query ?

HISTOGRAM query

- Suppose neighboring tables are D, D U {t}
- Adding or removing a value only changes one of the counts
- Sensitivity:

$$\max_{D,D'} |q(D) - q(D')|_1$$

  - Sum of absolute changes in each output.
  - Sensitivity is 1 for the histogram query

HISTOGRAM query

- Laplace Mechanism

  $[m1, m2, \ldots, mc] \rightarrow [m1 + \eta 1, m2 + \eta 2, \ldots, mc + \eta c]$

  where each ηi is drawn independently from Laplace($1/\varepsilon$)

- IMPORTANT: Need to add noise to every count in the histogram (even if the count is 0)

Parallel Composition

Let $A_1, A_2, \ldots, A_k$ be ε differentially private algorithms.

Let $P_1, P_2, \ldots, P_k$ be disjoint subsets of the domain.

Then, releasing $A_i(D \cap P_i)$ in parallel also satisfies ε differential privacy

– The above theorem only works when neighbors are D, D U {t}

HISTOGRAM query

- Suppose neighbors are D U {x}, D U {y}

- How to release histogram query?

- Sensitivity = 2
- Add noise drawn from Lap(2/ε) to each count.

Utility

- Error in the histogram query:

- Neighbors D, D U {t}
  Total error = $2c/\varepsilon^2$.

- Neighbors D U {x}, D U {y}
  Total error = $8c/\varepsilon^2$.

Multiple queries

- Suppose we want to release
  - Histogram of diseases
  - Histogram of ages
  - Histogram of salaries

- Sensitivity(releasing 3 histograms)
  = 3 * Sensitivity(releasing 1 histogram)
  - Adding an individual results in changing one count in each histogram.
  - Changing an individual's value results in changing 2 counts in each histogram.

Duke UNIVERSITY

Sequential Composition

If algorithms $A_1, A_2, \ldots, A_k$ use independent randomness and each A_i satisfies $\boldsymbol{\varepsilon_i}$-differential privacy, resp.

Then, outputting $A_1(D), A_2(D), \ldots, A_k(D)$ together satisfies differential privacy with

$$\boldsymbol{\varepsilon = \varepsilon_1 + \varepsilon_2 + \ldots + \varepsilon_k}$$

Postprocessing

- Suppose A1 satisfies differential privacy, and A2 is another algorithm.
- What can you say about A2 (A1 (D)) ?
- Also satisfies differential privacy

Composition

- Sequential composition allows breaking up complex functions into smaller building blocks and dividing the privacy budget amongst these building blocks.

- Parallel composition ensures that privacy budget is efficiently utilized (when acting on disjoint subsets of the data)

- Postprocessing does not impact privacy
  - Very useful as we will see in later classes.

Duke UNIVERSITY

Summary

- Differential privacy is a formal methodology for privacy protection.
  - Ensures that presence, absence or change in a single individual's value does not significantly impact output of the algorithm
  - Epsilon tunes the privacy utility tradeoff
- Output perturbation is an effective method
  - Adding noise from a Laplace distribution ensures differential privacy
  - Deterministic algorithms and sampling don't satisfy the condition
- Composition helps design complex privacy algorithms
  - Sequential and parallel composition

Duke UNIVERSITY