Lecture 1: 1/21/25:

Welcome to class!

Me: - Been at JHU 11 years
- work broadly in algorithms, focus on graph algs, approx algs, distributed computing
- Recently interested in differential privacy due to sabbatical at Google Research - NYC
- Learned this area in order to do research combining graph/combinatorial algs with DP: I never learned the basics!

- Goal for this class: teach myself the basics!
- So class not really planned out: definitely going to cover basics, but how fast? What order? What beyond basics?
- Beyond basics, plan is to teach (force myself to learn) stuff related to the kind of DP work that I do. But happy to take requests, evolve class to fit your interests!
(subject to being at least somewhat algorithmic)

Admin stuff:

- Fundamentally: (new) grad class!
  - Figure things out as we go.
  - www.cs.jhu.edu/~mdinitz/classes/DP-class/Spring2025/
  - Online discussion: courselore
  - Office hours: by appointment
  - TA: Shruthi Prusty

- Work:
  - Will be some homeworks, not sure when.
  - Participation
  - Final Project: up to you!
    - Can be small groups
    - Lots of options!
      - Research in DP algs
      - Research combining DP with your area
      - Survey/lecture of some DP topic we're not covering
      - Lecture/overview of recent DP paper(s)

        ⋮

Grades:
- 50% HW
- 30% project
- 20% participation

Textbook: Dwork and Roth, available online

- Will follow pretty closely for basics, but go beyond for advanced topics
- There are other classes and textbooks out there. Feel free to use as resources!

Start of technical content:

Main question: "privacy-preserving data analysis":

- Given a bunch of data, want to analyze it to learn things!
- But data might be sensitive - need to preserve privacy!
    - Medical data
    - Media consumption
    - Friends
    - Voting record

      ⋮
- How can we have privacy but still analyze/release the data?

- Classical approach: "anonymize" data
  - e.g., remove PII ("personally identifiable information").
  - Linkage attacks! Can combine "anonymized" data with "non-private" external data to re-identify people
  - Medical records of governor of Massachusetts: linked anonymized medical records with public voter registration records
  - Netflix "anonymized" viewing histories before releasing as part of Netflix challenge. De-anonymized by linking with IMDB
  - Dangerous even without full re-identification!
    - Ex: anonymized list of encounters at medical facility on one day - maybe only small # distinct diagnoses.
      - If know neighbor visited facility on that day, know something pretty private!
- Classical approach #2: only allow queries for "large" sets.
  - Sps know person X in medical database
    - "How many people in database have trait Y?"
    - "How many people in database not named X have trait Y"?

- Differencing attack!

- Query auditing: check whether queries violate privacy!
  - Computationally difficult (impossible even just for differencing attacks)
  - Refusing to answer can violate privacy!
- Summary statistics: still subject to both differencing and other reconstruction attacks!

- "Just a few": preserve most people's privacy, but not all!
  - Can often be achieved by just sampling small subset of database.
  - But these people get privacy completely compromised!

Differential Privacy: Most modern, popular formalization of privacy.
- Used in US census (controversial!), Google, Facebook, etc.

Database D, held by trusted curator
- think of one row/individual
- Can relax both one way trusted curator.

People want to analyze D, but we want to maintain privacy for people in D.

Two models:

- interactive: Analysts submit queries to curator, who then answers
- non-interactive: curator publishes something once: synthetic database, summary statistics, etc. then true data destroyed.

Private algorithm/mechanism: using D and random bits, output answer to query or synthetic database while preserving privacy.

Main question: What is "preserving privacy"?

Want to be very general, not just robust to exact reconstruction attacks.

Intuition 1: After analyzing D/answering query about D, shouldn't know much more about any individual.

Not possible!

Toy example: - sps believe everyone has 2 left feet.

- Analyze database, learn that everyone has one left, one right.
- Learned about individuals!

Smoking: - Sps learn that action A often causes B:

smoking causes lung cancer.

- Sps know person X does A
- After analysis, know X has good chance of B!

Generalizes: - want to analyze D to learn something about world.

- After learning this, know something more about individuals!

Intuition 2: "Plausible Deniability".

Ex: randomized response.

- want to know how many people have property P.
- Mechanism for each person:
  - with prob. 1/2, answer truthfully
  - with prob. 1/4, answer Yes
  - with prob. 1/4, answer No

Intuitively private!

- If p corresponds to illegal activity, answering Yes not incriminating.

But useful!

- If p fraction have property P,

$$E[\text{fraction say yes}] = p\left(\tfrac{1}{2}+\tfrac{1}{4}\right)+(1-p)\tfrac{1}{4}$$
$$= \tfrac{1}{2}p + \tfrac{1}{4}$$

$\Rightarrow$ given fraction say yes, can figure out p!

Similar intuition: "since plausible deniability, doesn't make much difference whether or not I'm in database"

$\Rightarrow$ Might as well participate!

Formalizing Differential Privacy:

- Let M be a randomized algorithm which takes as input a database and outputs something in Range(M)
- Two databases D, D' are neighboring if exactly one entry has been added/removed $(|D \Delta D'| = 1,\ |D \setminus D'| + |D' \setminus D| = 1)$

Note: can generalize!

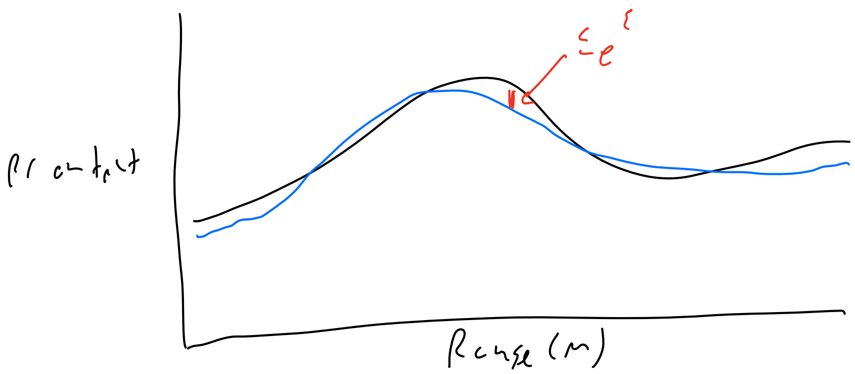
Def: M satisfies (ε, δ)-differential privacy if for all neighboring D, D' and for all $S \subseteq Range(M)$:

$$Pr[M(D) \in S] \leq e^{\varepsilon} Pr[M(D') \in S] + \delta$$ ← approx DP

If M satisfies $(\varepsilon, 0)$-DP, then just say ε-DP

Pure DP

(think of ε small constant, $\delta = \frac{1}{poly(n)}$)



Nonzero δ relaxes significantly for low probability events!

Idea: no matter what the algorithm does, output is basically the same in D and D'. So consider some person $x \in D$, let $D' = D \setminus x$. For any event $(S \subseteq Range(M))$, probability that output is in it is basically the same in D and D'

$\Rightarrow$ doesn't matter to x whether in database or not!

And get plausible deniability!

-Don't learn anything about x that couldn't have otherwise figured out (if x in database)

(cancer example, left foot etc.)

Automatically protects against not just linkage or difference attacks, but all attacks, since no way to tell from output whether x in database!

Formalization: immune to postprocessing! Even if you get more info later, do extra computation, etc., doesn't matter.

Thm: Let $M: \mathcal{D} \to R$ be randomized alg. that is (ε, δ)-DP. Let $f: R \to R'$ be arbitrary randomized mapping. Then $f \circ M: \mathcal{D} \to R'$ is (ε, δ)-DP.

Pf: Sps f deterministic.

Let $D, D' \in \mathcal{D}$ be neighboring databases

Let $S \subseteq R'$

Let $T = \{r \in R : f(r) \in S\}$

$$\Rightarrow \Pr[f(M(D)) \in S] = \Pr[M(D) \in T]$$
$$\leq e^{\varepsilon} \Pr[M(D') \in T] + \delta$$
$$= e^{\varepsilon} \Pr[f(M(D')) \in S] + \delta \quad \checkmark$$

Now sps f randomized.

$\Rightarrow$ convex combination of deterministic g_i's

$$\Rightarrow \Pr_{f,M}[f(M(D)) \in S] = \Pr_{f,M}[g_i(M(D)) \in S]$$

$$= \sum_i \alpha_i \Pr_M[g_i(M(D)) \in S]$$

$$\le \sum_i \alpha_i \left(e^{\varepsilon} \Pr_M[g_i(M(D')) \in S] + \delta\right)$$

$$= \sum_i \alpha_i e^{\varepsilon} \Pr_M[g_i(M(D')) \in S] + \delta$$

$$= e^{\varepsilon} \Pr_{f,M}[f(M(D')) \in S] + \delta \quad \checkmark$$

Other nice things we'll eventually prove about DP:

- composition: running a few DP algs still DP!
- group privacy: even if databases differ in >1, still get some guarantee!

Next time: some simple mechanisms.