Local Model for Differentially Private Data Analysis

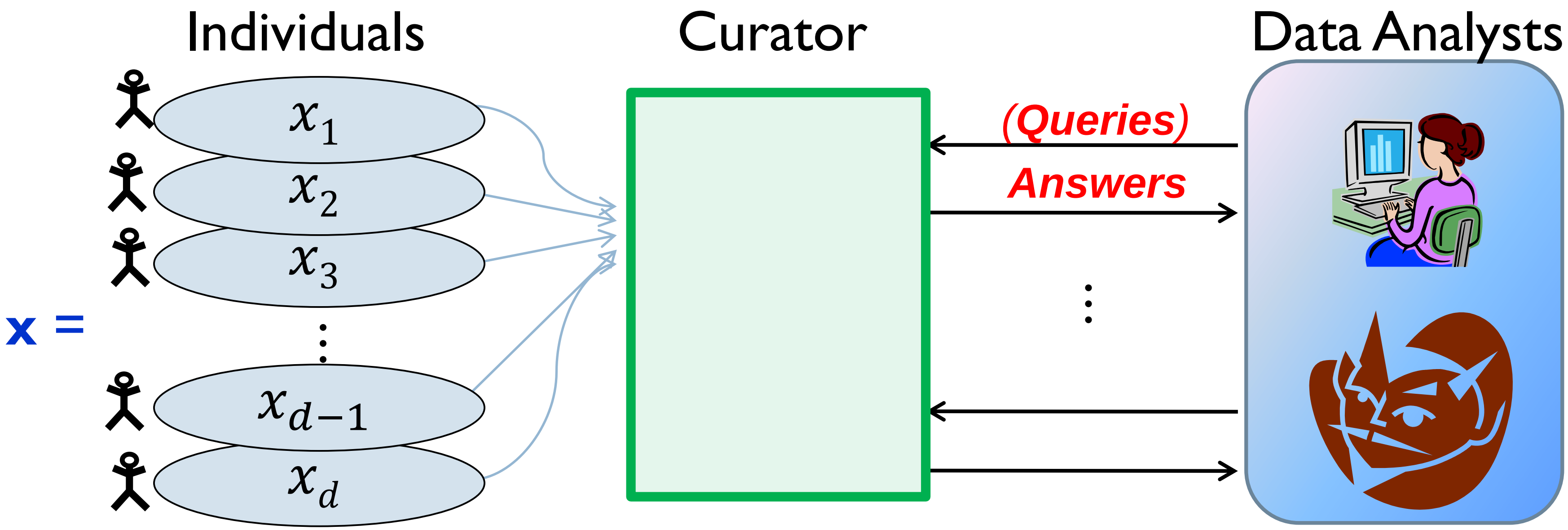
Sofya Raskhodnikova

Boston University

Some slides are based on slides by
Adam Smith (*Boston University*)

BOSTON UNIVERSITY

Private Data Analysis



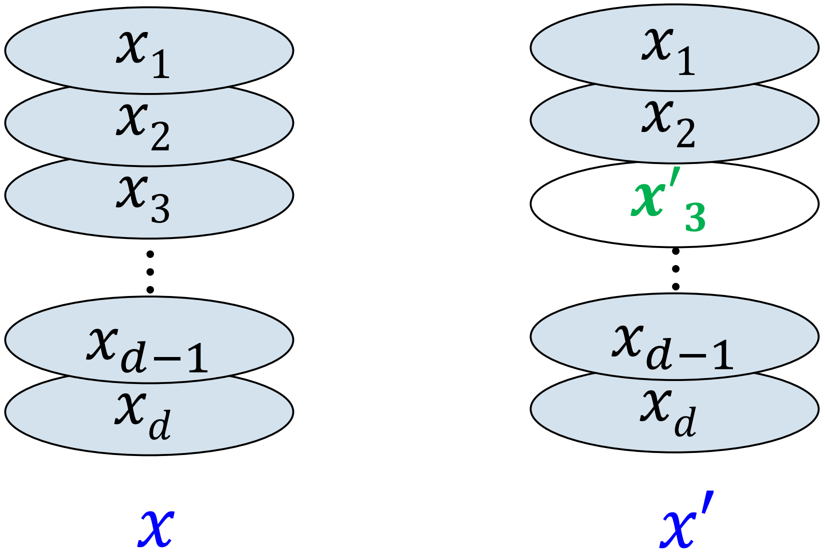
Typical examples: *census*, *medical studies*, *what big companies want to publish about our data*…

Two conflicting goals

- ➢ *Protect privacy of individuals*
  - ***Differential privacy*** *[Dwork McSherry Nissim Smith 06]*
- ➢ *Give accurate answers*

Neighboring Datasets

Two datasets x, x' are **neighbors** if they differ in one person's data.



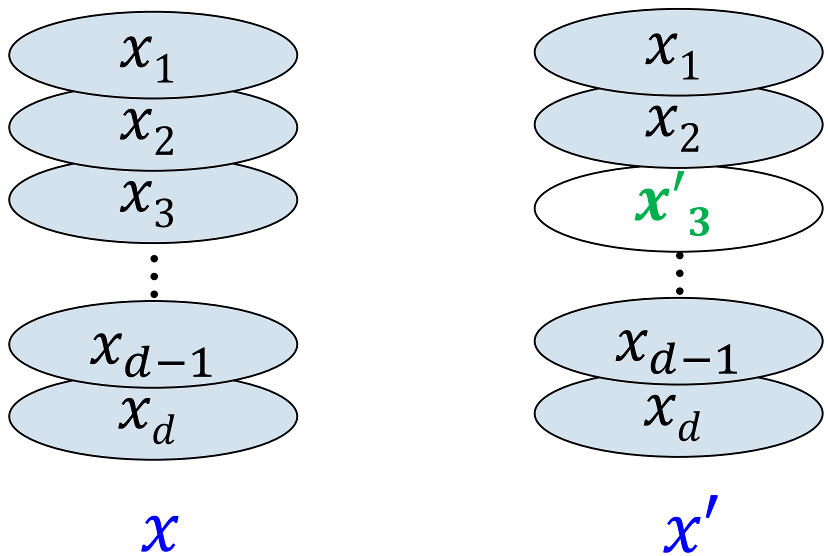
Differential Privacy [Dwork McSherry Nissim Smith 06]

Privacy Definition

An algorithm A is ϵ-differentially private if

for all pairs of **neighbors** $\boldsymbol{x}, \boldsymbol{x}'$ and all sets of answers **S**:

$$\mathbf{Pr}[\boldsymbol{A}(\boldsymbol{x}) \in \boldsymbol{S}] \leq \boldsymbol{e}^{\epsilon}\, \mathbf{Pr}[\boldsymbol{A}(\boldsymbol{x}') \in \boldsymbol{S}]$$



Properties of Differential Privacy

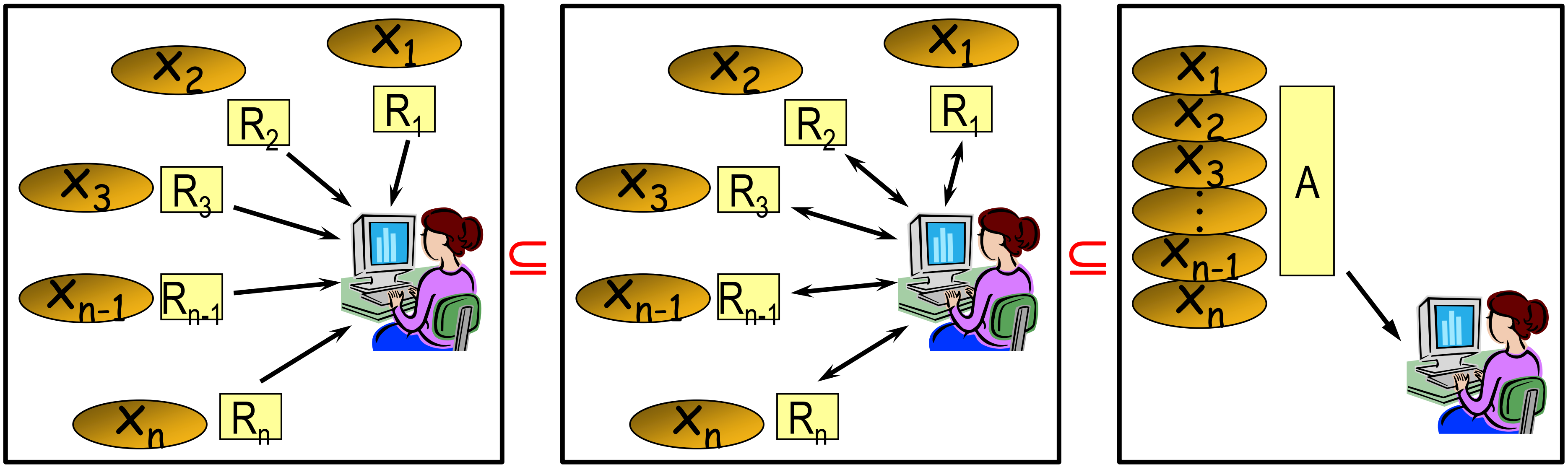
- Composition:

If algorithms A_1 and A_2 are ϵ-differentially private **then** algorithm that outputs $(A_1(x), A_2(x))$ is 2ϵ-differentially private

- Meaningful in the presence of arbitrary external information

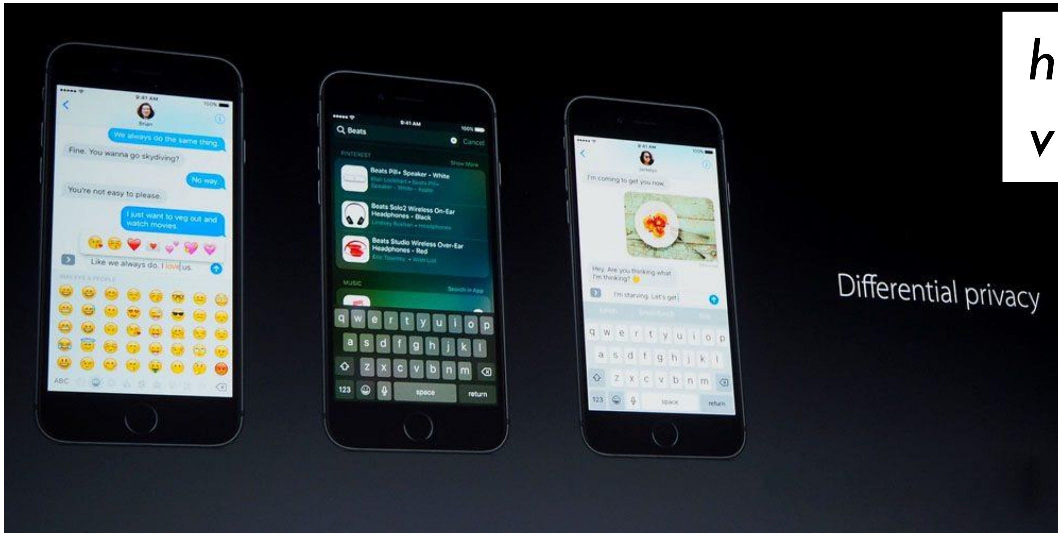
Basic Privacy Models

Local Noninteractive **Local (Interactive)** **Centralized**



- Advantages of the local model:
  - private data never leaves person's hands
  - no single point of failure
  - highly distributed
- Disadvantage of the local model:
  - data-thirsty (more data for the same accuracy)
  - Exponentially more data for learning parity

Deployments of the Local Privacy Model



https://developer.apple.com/videos/play/wwdc2016/709

Chrome

https://github.com/google/rappor

Differential Privacy in the Local Model

Privacy Definition

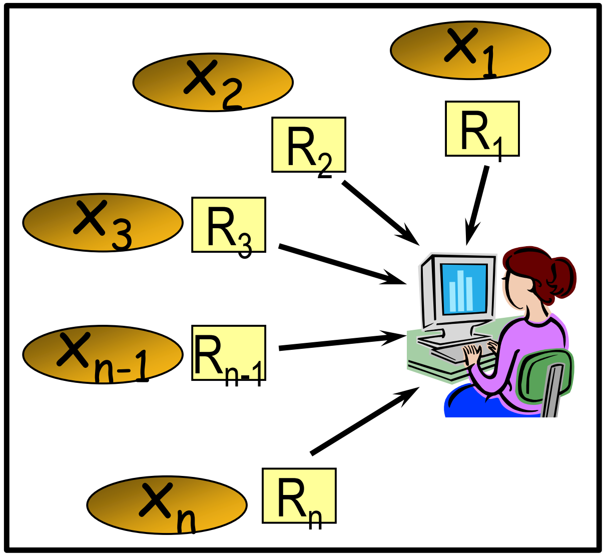
A randomizer R is ϵ-differentially private if

for all pairs of values $x_i, x_i{}'$ and all sets of answers **S**:

$$\mathbf{Pr}[R(x_i) \in S] \le e^{\epsilon}\, \mathbf{Pr}[A(x_i') \in S]$$

- The requirement that the ratio $\frac{\Pr[R(x_i)=a]}{\Pr[A(x_i')=a]}$ be bounded predates differential privacy

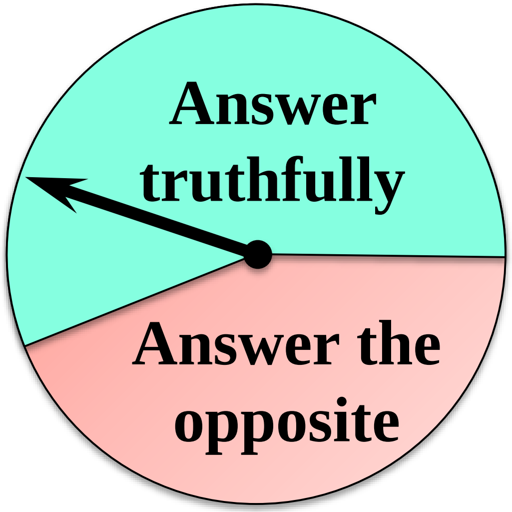
  [Efvimievski Gehrke Srikant 03]



Randomized Response [Warner 65]

- Canonical example of a local algorithm
- Invented to help get truthful answers on sensitive YES/NO survey questions.
- Each person has data $x_i \in \mathcal{X}$
  - Given $f: \mathcal{X} \to \{-1,1\}$, analyst needs the average of $f(x_i)$
  - Can deduce, e.g., the proportion of diabetics
- Randomization operator takes $y \in \{-1,1\}$:

$$R(\boldsymbol{y}) = \begin{cases} +\boldsymbol{y} & w.p.\ \frac{e^{\epsilon}}{e^{\epsilon}+1} \\ -\boldsymbol{y} & w.p.\ \frac{1}{e^{\epsilon}+1} \end{cases}$$

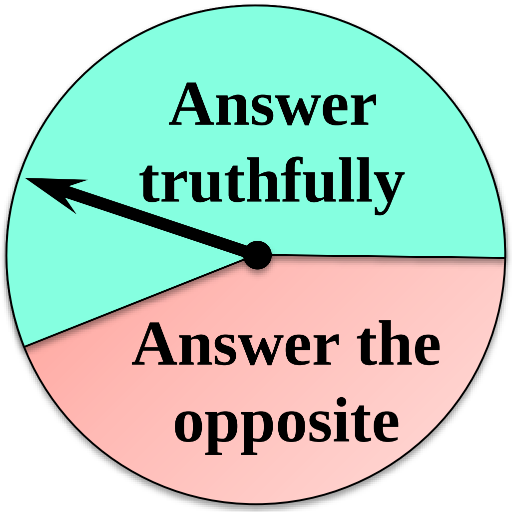


Randomized Response

- Randomization operator takes $y \in \{-1,1\}$:

$$R(\boldsymbol{y}) = \begin{cases} +\boldsymbol{y} & \text{w.p. } \frac{e^{\epsilon}}{e^{\epsilon}+1} \\ -\boldsymbol{y} & \text{w.p. } \frac{1}{e^{\epsilon}+1} \end{cases}$$

ratio is e^{ϵ}



- $\boldsymbol{E}[R(\boldsymbol{y})] = \boldsymbol{y} \cdot \frac{e^{\epsilon}}{e^{\epsilon}+1} - \boldsymbol{y} \cdot \frac{1}{e^{\epsilon}+1} = \boldsymbol{y} \cdot \frac{e^{\epsilon}-1}{e^{\epsilon}+1}$
- If we rescale by $c_{\epsilon} = \frac{e^{\epsilon}+1}{e^{\epsilon}-1}$, then $\boldsymbol{E}[c_{\epsilon} \cdot R(\mathbf{y})] = \mathbf{y}$
- We can estimate the average of $f(x_i)$

$$\boldsymbol{A}(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n) = \frac{1}{n} \sum_i c_{\epsilon} \cdot R\big(\boldsymbol{f}(\boldsymbol{x}_i)\big)$$

Lemma. $\boldsymbol{E}\left[\left|A(\boldsymbol{x}) - \frac{1}{n}\sum_i f(x_i)\right|\right] \leq \frac{c_{\epsilon}}{\sqrt{n}} \approx \frac{1}{\epsilon\sqrt{n}}$.

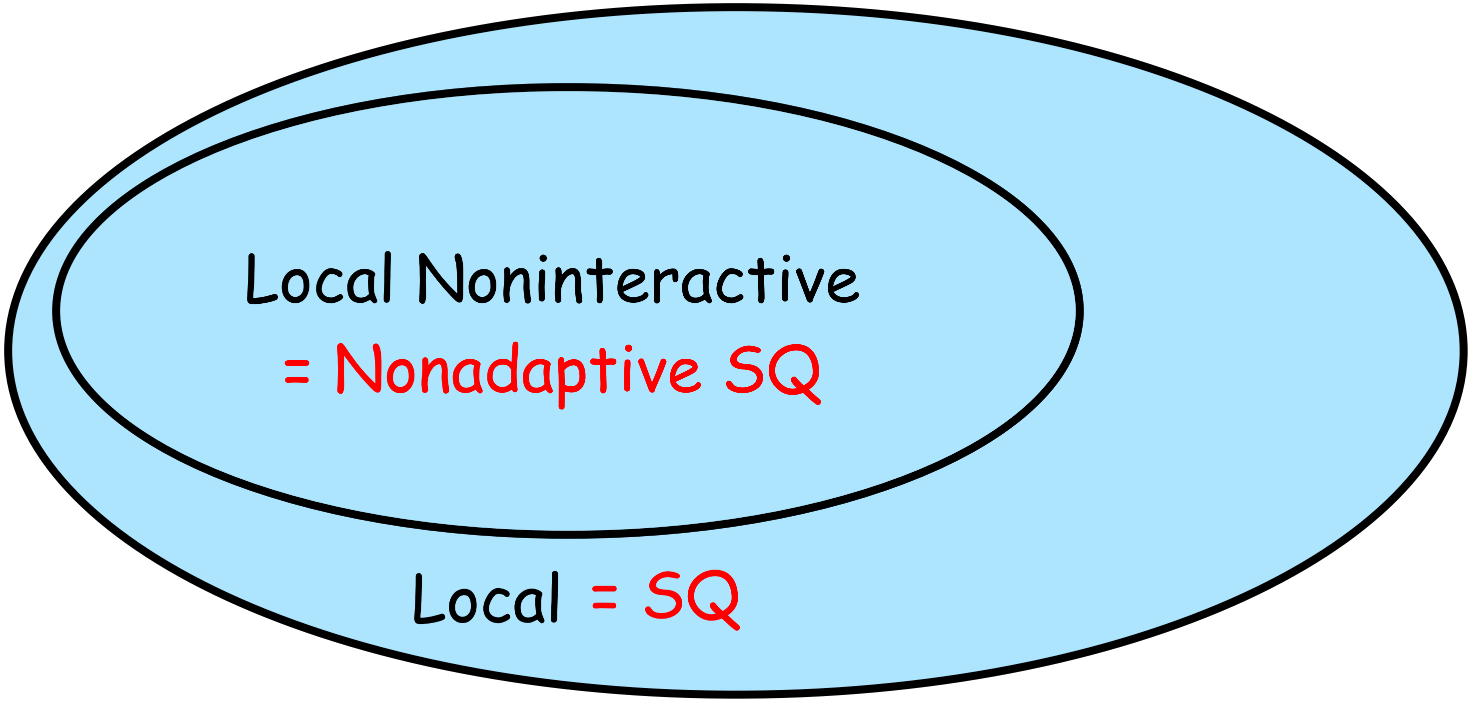
Randomized Response: Generalization

- Can be generalized to estimating the averages of functions of the form $f: \mathcal{X} \rightarrow [-1,1]$
- If $y \in [-1,1]$, first round it to 1 or -1:

$$\text{Round}(\boldsymbol{y}) = \begin{cases} +\mathbf{1} & \text{w.p. } \frac{1+\boldsymbol{y}}{2} \\ -1 & \text{w.p. } \frac{1-\boldsymbol{y}}{2} \end{cases}$$

- Define $\text{RR}(\boldsymbol{y}) = R\big(\text{Round}(\boldsymbol{y})\big)$
- $\boldsymbol{E}[RR(y)] = \boldsymbol{E}[\text{Round}(\boldsymbol{y})] = \frac{1+\boldsymbol{y}}{2} - \frac{1-\boldsymbol{y}}{2} = \boldsymbol{y}$
- We can estimate the average as before.

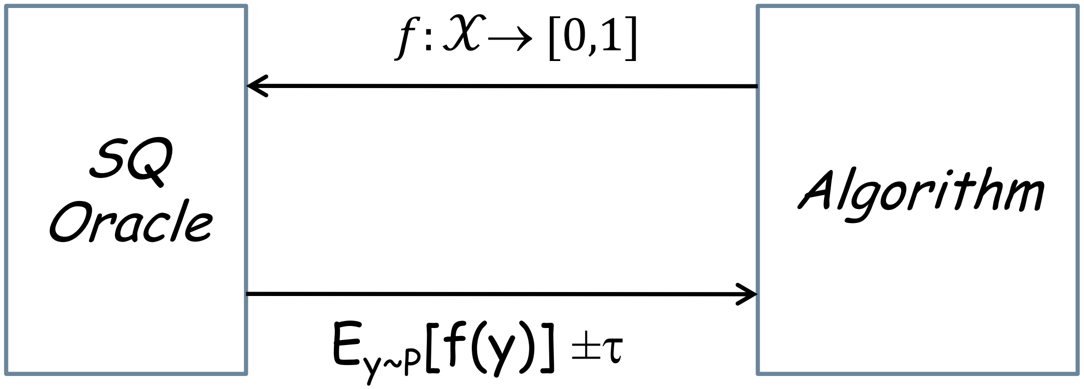
Power of Local Models [Kasiviswanathan Lee Nissim Raskhodnikova Smith 11]



Containment is strict: there are computational tasks for which noninteractive protocols require **exponentially** larger n than interactive ones.

Statistical Query (SQ) Algorithms

- An SQ algorithm can perform its computation by accessing the data via an SQ oracle.



✓ *tolerance τ > 1/poly(…)*
✓ *f can be evaluated in poly time*
✓ *poly running time*

- Distribution P could be the distribution from which the data drawn or the empirical distribution over the data set.
- A *nonadaptive* algorithm specifies all its queries in advance.
- Huge fraction of basic learning/optimization algorithms can be expressed in SQ form [Kearns 93]

Theorem [Blum Dwork McSherry Nissim 05]

Any SQ algorithm can be simulated by a private algorithm.

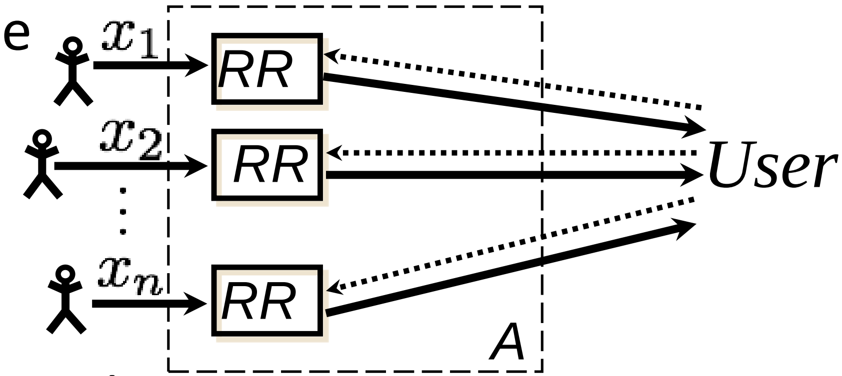
(Noninteractive) Local =(Nonadaptive) SQ

Theorem

Any q-query (nonadaptive) SQ algorithm with tolerance τ can be simulated by an ϵ-DP (noninteractive) local algorithm if $n \geq \frac{q \ln q}{\tau^2 \epsilon^2}$.

Local protocol for an SQ query:

- use a different group of $\boldsymbol{n/q}$ people
- for each i, compute bit RR(f(x_i))
- average the noisy bits and rescale



- Participants can compute noisy bits on their own
- RR (applied by each participant) is differentially private
- If all SQ queries are known in advance (non-adaptive), the protocol is non-interactive

(Noninteractive) Local =(Nonadaptive) SQ

Theorem

When the data is sampled i.i.d. from an unknown distribution P, any (noninteractive) local algorithm can be simulated by a (nonadaptive) SQ algorithm.

Technique: Rejection sampling

Proof idea [noninteractive case]:

- To simulate a randomizer $R: D \to W$ on entry $\boldsymbol{x_i}$, need to output each $\boldsymbol{w} \in \boldsymbol{W}$ with probability $p(w) = \Pr_{\boldsymbol{y \sim P}}[\boldsymbol{R(y) = w}]$.
- Let $q(w) = \mathbf{Pr}[\boldsymbol{R(0) = w}]$. (Approximates $p(w)$ up to factor e^{ϵ}).

  1. Sample w from $q(w)$.
  2. Output w with probability $\frac{\boldsymbol{p(w)}}{\boldsymbol{q(w)e^{\epsilon}}}$.
  3. With the remaining probability, repeat from (1).

- Use SQ queries to estimate $p(w)$.

Idea:

$$p(w) = \Pr_{\boldsymbol{y \sim P}}[R(z) = w] = \sum_{y} \Pr_{y \sim P}[y] \cdot \Pr[R(y) = w]$$

$$= \mathbf{E}_{\boldsymbol{y \sim P}}[\Pr[R(y) = w]]$$

Summary

- We characterized the class of problems solvable in local noninteractive and local interactive models
  - with respect to sample size
  - up to polynomial factors
- Many specific tasks are studied (learning, heavy hitters, histograms, optimization problems, clustering,…)
  - Best algorithms often use sketching techniques
  - Information-theoretic techniques were developed for lower bounds [Beimel Nissim Omri 08, Chan Shi Song 12, Duchi Jordan Wainwright 13,…]
- For specific tasks, need to optimize
  - Amount of data need for specific accuracy
  - Running time
  - Communication
  - Server memory