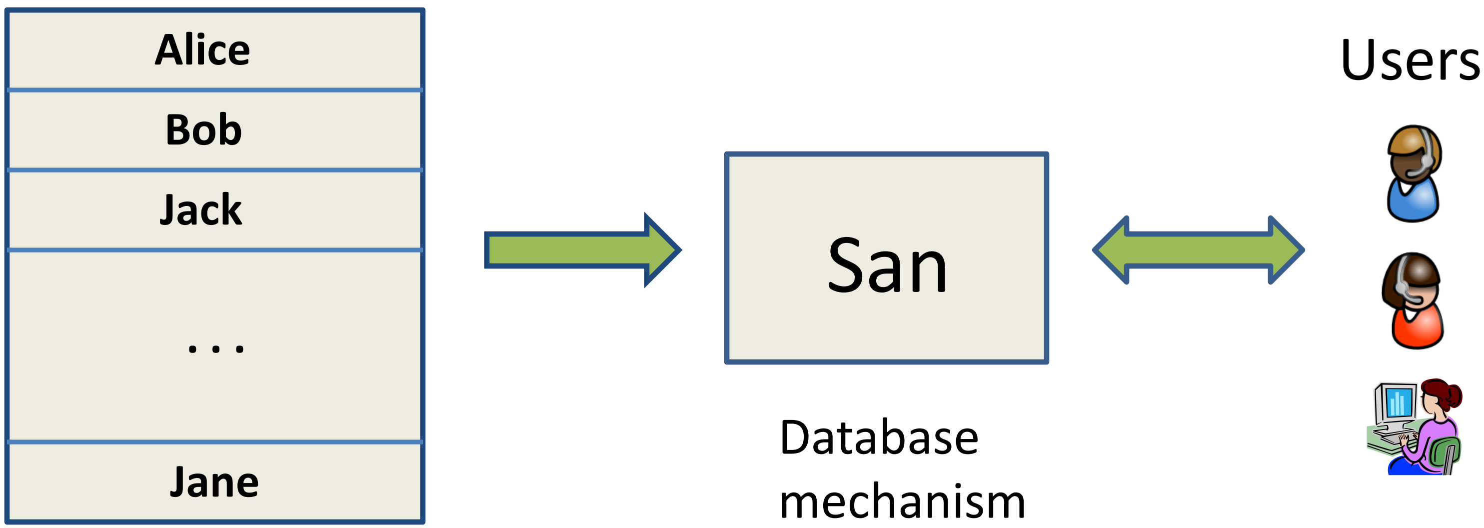
Crowd-Blending Privacy

Johannes Gehrke, Michael Hay, Edward Lui, Rafael Pass

Cornell University

Data Privacy



Database containing
data. E.g., census data,
medical records, etc.

- Utility: Accurate statistical info is released to users
- Privacy: Each individual’s sensitive info remains hidden

Simple Anonymization Techniques are Not Good Enough!

- Governor of Massachusetts Linkage Attack [Swe02]
  - "Anonymized" medical data + public voter registration records
    $\Rightarrow$ Governor of MA's medical record identified!

- Netflix Attack [NS08]
  - "Anonymized" Netflix user movie rating data + public IMDb database
    $\Rightarrow$ Netflix dataset partly deanonymized!

Privacy Definitions

- *k*-anonymity [Sam01, Swe02]
  - Each record in released data table is indistinguishable from *k*-1 other records w.r.t. certain identifying attributes
- Differential privacy [DMNS06]
  - ∀ databases D, D' differing in only one row,

$$\text{San}(D) \approx_{\varepsilon} \text{San}(D')$$

- Zero-knowledge privacy [GLP11]
  - ∀ adversary A interacting with San, ∃ a simulator S s.t. ∀ D, z, i, the simulator S can simulate A's output given just k random samples from D \ {i}:

$$\text{Out}_A(A(z) \leftrightarrow \text{San}(D)) \approx_{\varepsilon} S(z, \text{RS}_k(D \setminus \{i\}))$$

Privacy Definitions

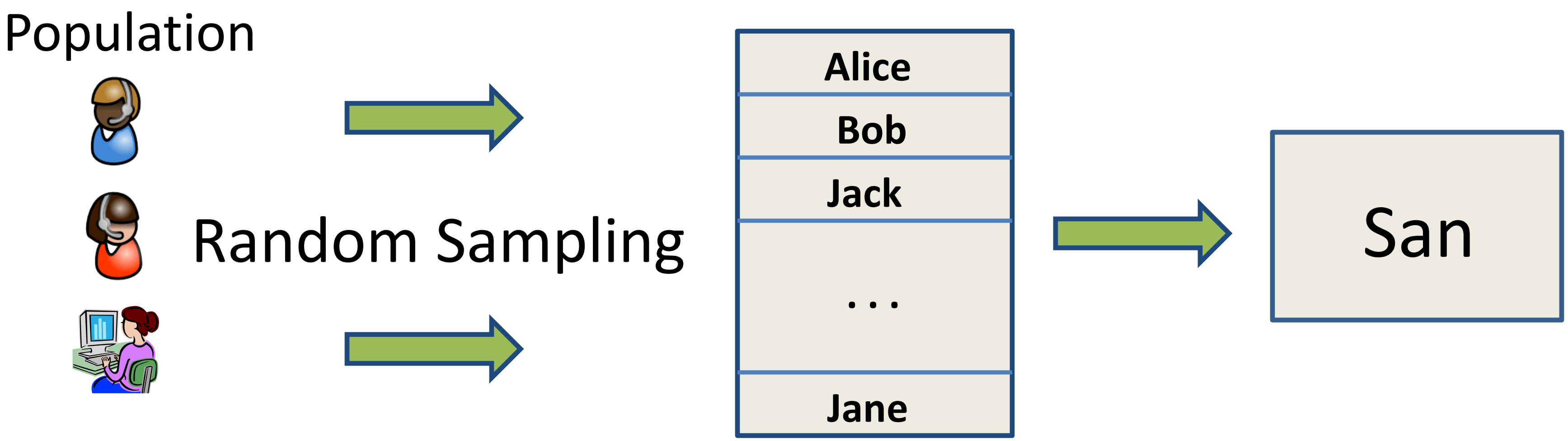
- *k*-anonymity
  - **Good:** Simple; efficient; practical
  - **Bad:** Weak privacy protection; known attacks
- Differential privacy
  - **Good:** Strong privacy protection; lots of mechanisms
  - **Bad:** Have to add noise. Efficient? Practical?
- Zero-knowledge privacy
  - **Good:** Even stronger privacy protection, lots of mechanisms
  - **Bad:** Have to add even more noise. Efficient? Practical?

Practical Sanitization?

- Differential privacy and zero-knowledge privacy
  - Mechanism needs to be randomized
  - noise is added to the exact answer/output (sometimes quite a lot!)
- In practice
  - Don’t want to add (much) noise
  - Want simple and efficient sanitization mechanisms
- Problem: Is there a practical way of sanitizing data while ensuring privacy and good utility?

Privacy from Random Sampling

- In practice, data is often collected via random sampling from some population (e.g., surveys)



- Already known: If San is differentially private, then the random sampling step amplifies the privacy of San [KLNRS08]
- Can we use a qualitatively weaker privacy def. for San and still have the combined process satisfy a strong notion of privacy?

Leveraging Random Sampling

- **Goal:** Provide a privacy definition such that if San satisfies the privacy definition, then:

Random Sampling + San $\Rightarrow$ Differential privacy or zero-knowledge privacy

- Should be weaker than differential privacy
  $\Rightarrow$ Better utility!
- Should be meaningful by itself (without random sampling)
  – Strong fall-back guarantee if the random sampling is corrupted or completely leaked

k-Anonymity Revisited

- *k*-anonymity: Each record in released data table is indistinguishable from *k*-1 other records w.r.t. certain identifying attributes
- Based on the notion of "blending in a crowd"
- Simple and practical
- Problem: Definition restricts the output, not the mechanism that generates it
  - Leads to practical attacks on *k*-anonymity

k-Anonymity Revisited

- A simple example illustrating the problem:
  - Use any existing algorithm to generate a data table satisfying *k*-anonymity
  - At the end of each row, attach the personal data of some fixed individual from the original database

- The output satisfies *k*-anonymity but reveals personal data about some individual!

- There are plenty of other examples!

Towards a New Privacy Definition

- *k*-anonymity does not impose restrictions on mechanism
  - Does not properly capture "blending in a crowd"
- One of the key insights of differential privacy: Privacy should be a property of the mechanism!
- We want a privacy definition that imposes restrictions on the mechanism and properly captures "blending in a crowd"

Our Main Results

- We provide a new privacy definition called crowd-blending privacy

- We construct simple and practical mechanisms for releasing histograms and synthetic data points

- We show:

Random Sampling + Crowd-blending privacy → Zero-knowledge privacy

Blending in a Crowd

- Two individuals (with data values) t and t' are ε-indistinguishable by San if

$$\mathrm{San}(D, t) \approx_{\varepsilon} \mathrm{San}(D, t') \quad \forall D$$

- Differential privacy: Every individual t in the universe is ε-indistinguishable by San from every other individual t' in the universe.
  - In any database D, each individual in D is ε-indistinguishable by San from every other individual in D

Blending in a Crowd

- First attempt of a privacy definition:
  $\forall$ D of size $\geq$ k, each individual in D is ε-indistinguishable by San from at least k-1 other individuals in D.
  - Collapses back down to differential privacy:
    If DP doesn't hold, then $\exists$ t and t' s.t. San can ε-distinguish t and t'; now, consider a database D = (t, t', t', …, t').

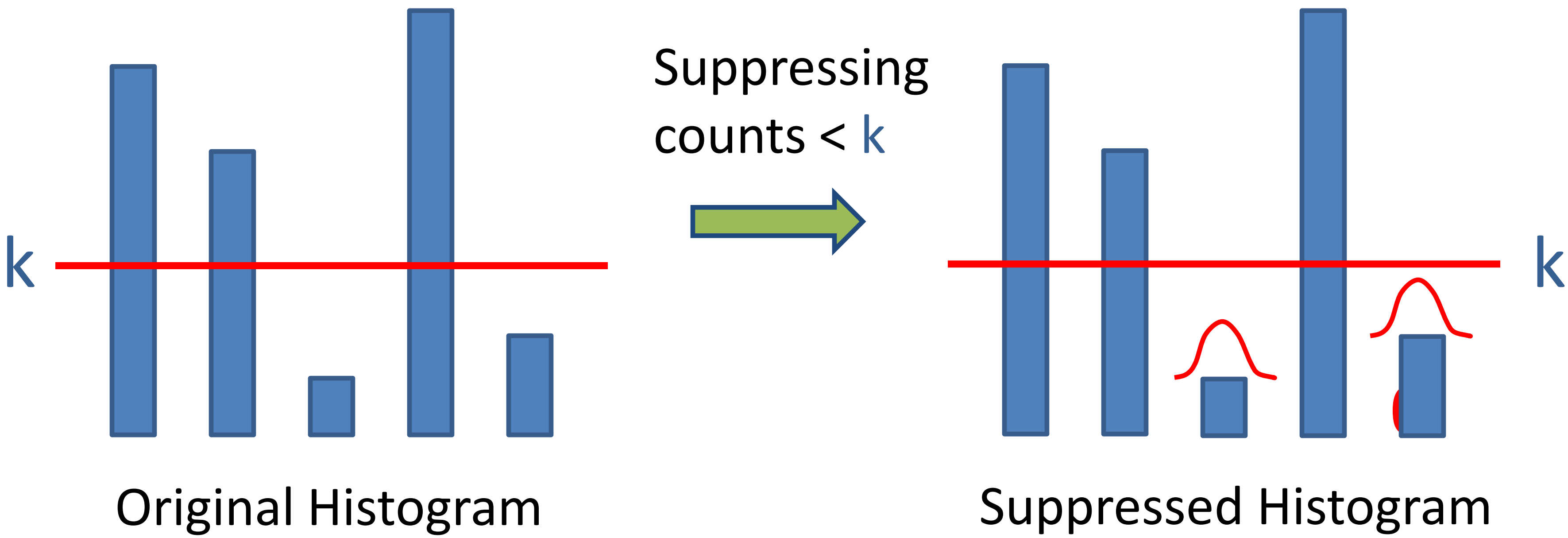
- Solution: D can have "outliers", but we require San to essentially delete/ignore them.

Crowd-Blending Privacy

- **Definition:** San is (k,ε)-crowd-blending private if $\forall$ D, and $\forall$ t in D, either
  - t is ε-indistinguishable from $\geq k$ individuals in D, or
  - t is essentially ignored: $San(D) \approx_{\varepsilon} San(D \setminus \{t\})$.
- Weaker than differential privacy
  $\Rightarrow$ Better utility!
- Meant to be used in conjunction with random sampling, but still meaningful by itself

Privately Releasing Histograms

- (k,0)-crowd-blending private mechanism for releasing histogram:
  - Compute histogram
  - For bin counts < k, suppress to 0



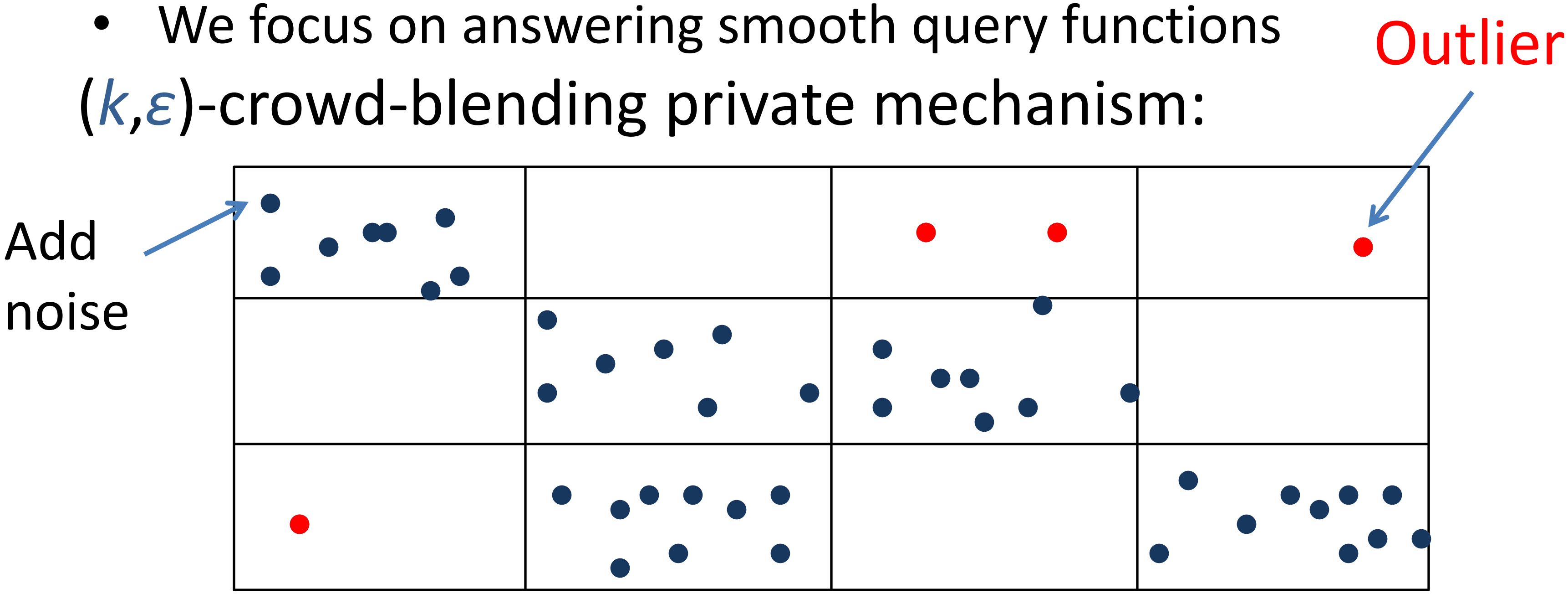
Simple and similar to what is done in practice!

(Not differentially private)

Privately Releasing Synthetic Data Points

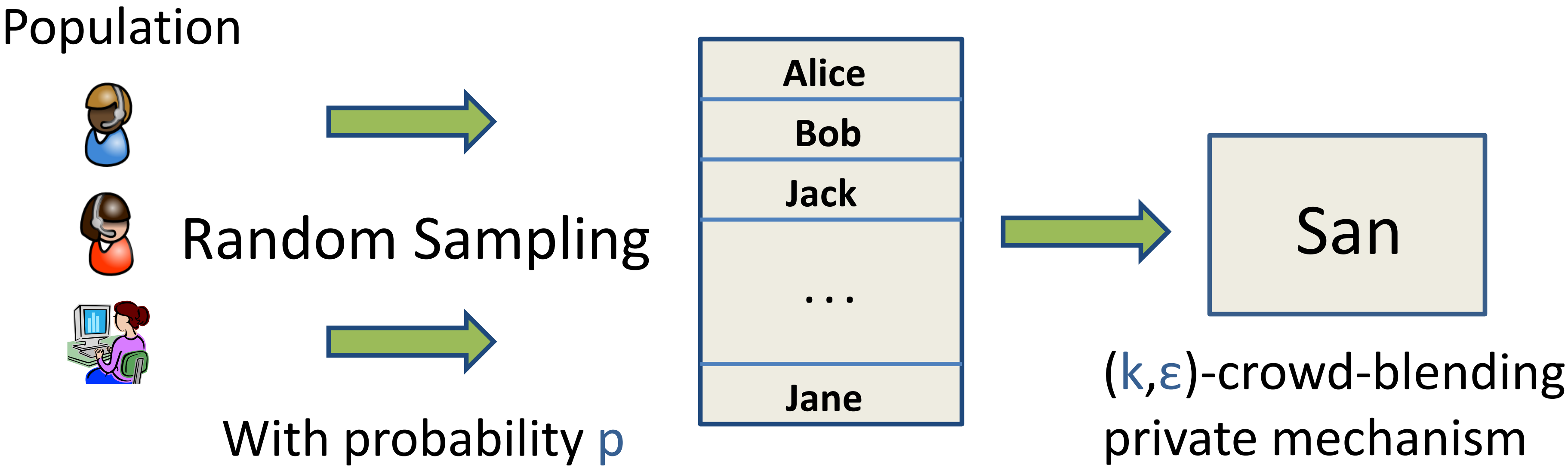
- Impossible to efficiently and privately release synthetic data points for answering general classes of counting queries [DNRRV09, UV11]
- We focus on answering smooth query functions

(k,ε)-crowd-blending private mechanism:



- The above CBP mechanism: Useful for answering all smooth query functions with decent accuracy
  - Not possible with differentially private synthetic data points

Our Main Theorem



Theorem (Informal): The combined process satisfies zero-knowledge privacy, and thus differential privacy as well.

Our theorem holds even if the random sampling is slightly biased as follows:

- Most individuals are sampled w.p. ≈ p
- Remaining are sampled with arbitrary probability

Thank you!