Privacy of Dynamic Data: Continual Observation and Pan Privacy

Moni Naor

Weizmann Institute of Science

Based on Joint Work With:

Cynthia Dwork

Toni Pitassi

A lot of the work done at MSR SVC

Guy Rothblum

Sergey Yekhanin

What is Privacy?

Extremely overloaded term

Hard to define

“Privacy is a value so **complex**, so entangled in **competing and contradictory dimensions**, so engorged with **various and distinct meanings**, that I sometimes despair whether it can be usefully addressed at all.”

Robert C. Post, *Three Concepts of Privacy*,

“Privacy is like oxygen – you only feel it when it is gone”

Charles J. Sykes

Lots of Data

Recent years: a lot of data is available to companies and government agencies

- Census data
- Huge databases collected by companies
  - Data deluge
- Public Surveillance Information
  - Cameras
  - RFIDs
- Social Networks

Mandatory participation

Must not reveal individual data

Statistical Data Analysis

Huge social benefits from analyzing large collections of data:

Finding co

E.g. medi

Providing

Improve

Publishing

Census, c

Dataminin

Clustering ... ors, principal component analysis

WHAT ABOUT PRIVACY?

Better Privacy Better Data

However: data contains **confidential** information

Almost any usage of the data that is no carefully rafted will leak something about it

AOL Search History Release (2006)

- 650,000 users, 20 Million queries, 3 months
- **AOL's goal**:
  - provide real query logs from real users
- Privacy?
  - "Identifying information" replaced with random identifiers
  - **But:** different searches by the same user still linked

AOL Search History Release (2006)

A Face Is Exposed for AOL Searcher No. 4417749

By MICHAEL BARBARO and TOM ZELLER Jr. The New York Times
Published: August 9, 2006

Buried in a list of 20 million Web search queries collected by AOL and recently released on the Internet is user No. 4417749. The number was assigned by the company to protect the searcher's anonymity, but it was not much of a shield.

No. 4417749 conducted hundreds of searches over a three-month period on topics ranging from "numb fingers" to "60 single men" to "dog that urinates on

Name: Thelma Arnold
Age: 62
Widow
Residence: Lilburn, GA

Privacy of Public Data Analysis

The holy grail:

Get **utility** of statistical analysis while **protecting privacy** of every **individual** participant

Ideally:

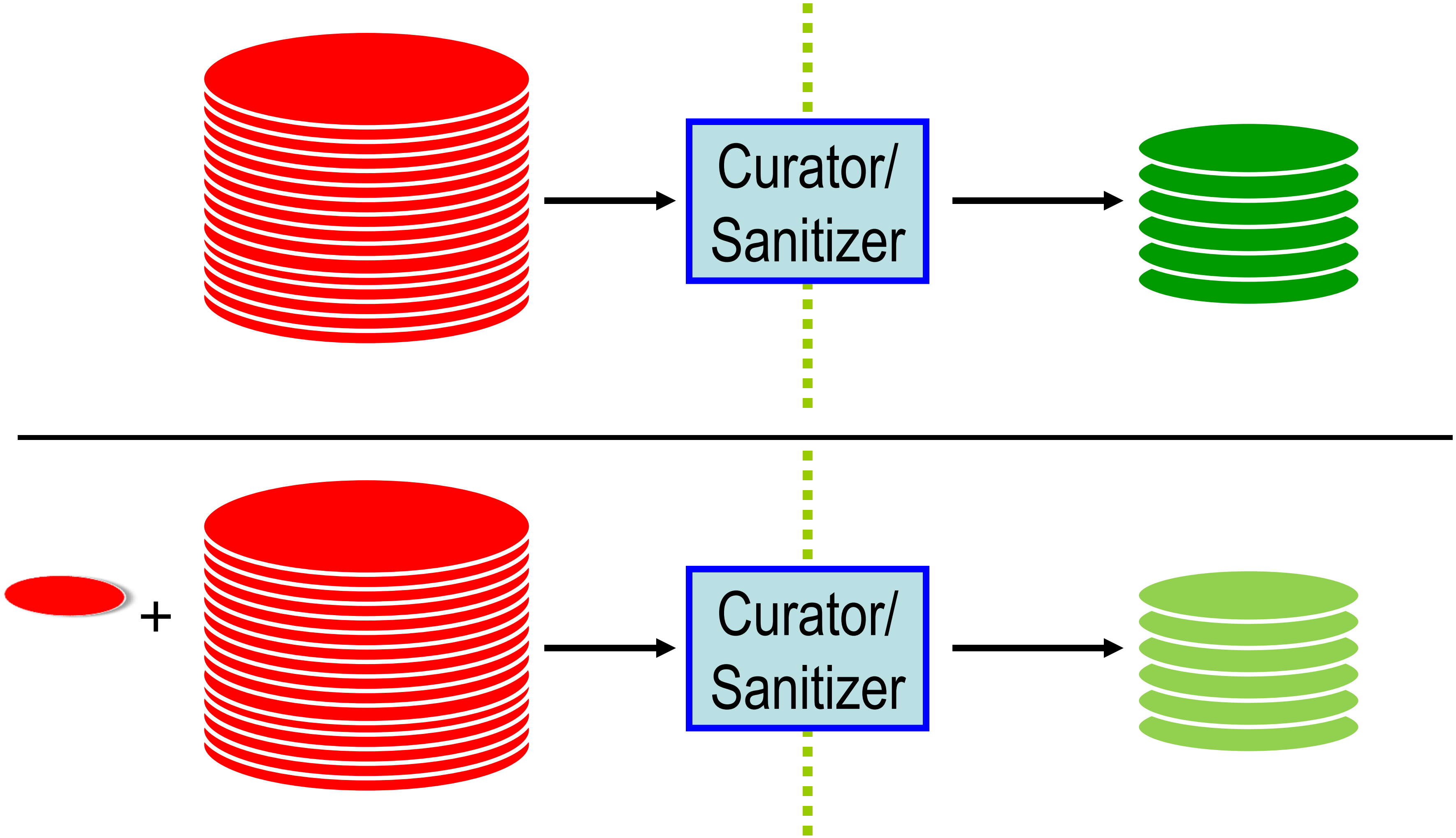
"privacy-preserving" sanitization allows reasonably **accurate answers** to **meaningful information**

Is it possible to phrase the goal in a meaningful and achievable manner?

Differential Privacy

Dwork, McSherry
Nissim & Smith
2006

Protect *individual* participants:



Differential Privacy [DwMcNiSm06]

Protect individual participants:

Probability of every bad event - or any event - **increases only by** small multiplicative factor **when I enter the DB.**

May as well participate in DB…

Adjacency: D+Me and D-Me

ε-differentially private sanitizer *A*

For **all** DBs D, all Me and **all** outputs T

Handles aux input

$$e^{-\varepsilon} \leq \frac{\Pr_A[A(D+Me)\ 2\ T]}{\Pr_A[A(D-Me)\ 2\ T]} \leq e^{\varepsilon} \approx 1+\varepsilon$$

Example: NO Differential Privacy

X set of $(name, tag\ 2\{0, 1\})$ tuples

One query: #of participants with $tag=1$

Sanitizer A: release a few random tuples with no names

Bad event T: Only my tag is 1, my tag released

$Pr_A[A(D+Me)\ 2\ T] \geq 1/n$

$Pr_A[A(D-Me)\ 2\ T] = 0$

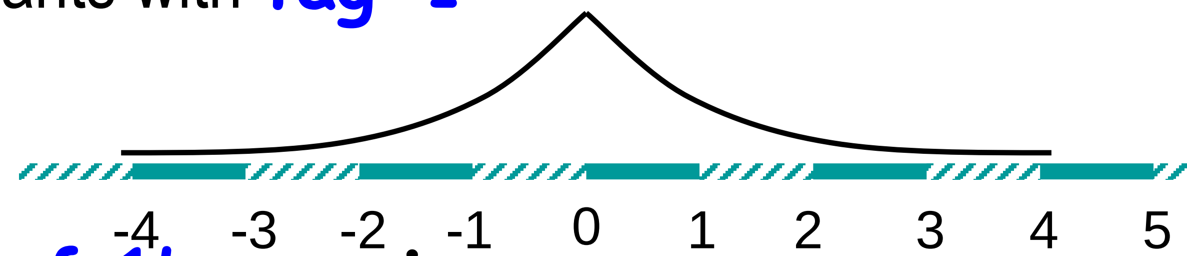
Not diff private **for any ε!**

$$e^{-\varepsilon} \leq \cancel{\frac{Pr_A[A(D+Me)\ 2\ T]}{Pr_A[A(D-Me)\ 2\ T]}} \leq e^{\varepsilon}$$

Example: YES Differential Privacy

X set of **(name, tag 2{0, 1})** tuples

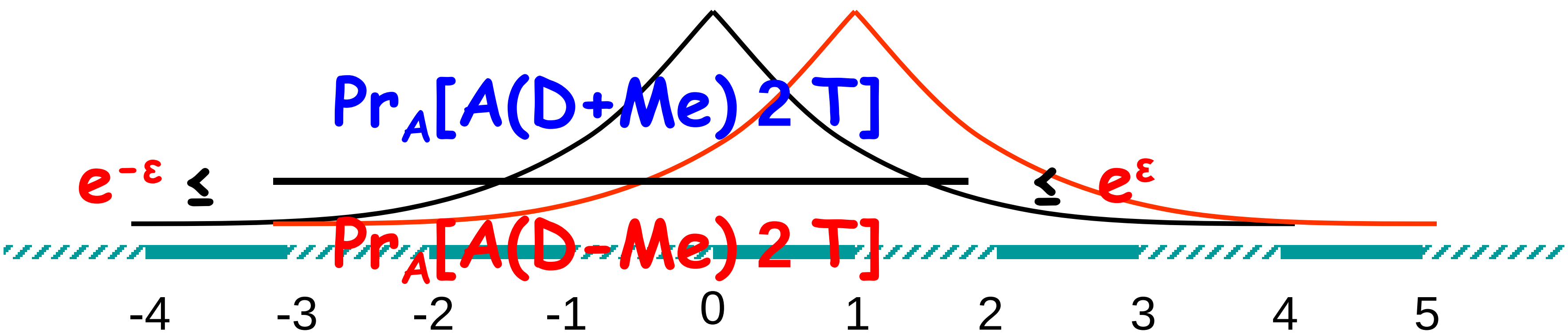
One query: #of participants with **tag=1**



Sanitizer A: output **#of 1's + noise**

- noise from **Laplace distribution** with parameter $1/\varepsilon$
- $Pr[noise = k-1] \approx e^{\varepsilon} \, Pr[noise = k]$

$$e^{-\varepsilon} \leq \frac{Pr_A[A(D+Me)\ 2\ T]}{Pr_A[A(D-Me)\ 2\ T]} \leq e^{\varepsilon}$$

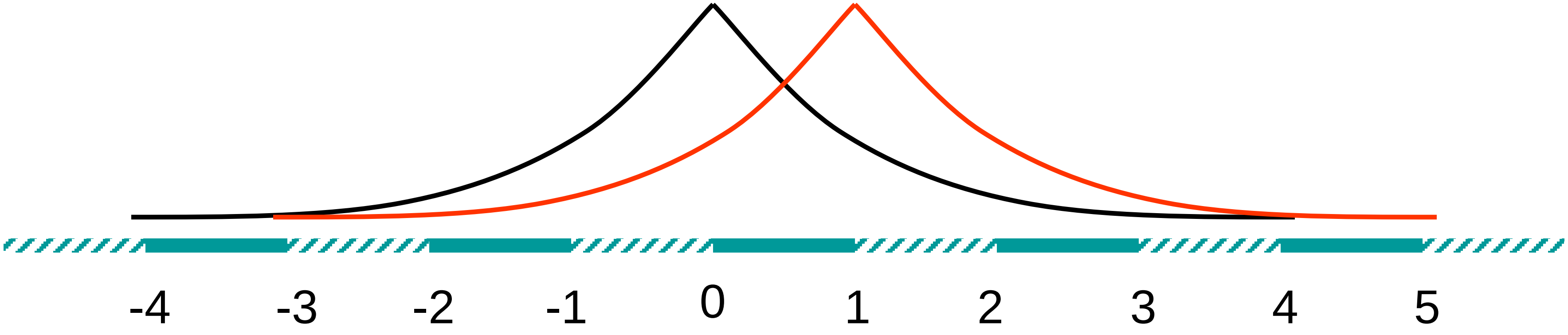


Laplacian Noise

- Laplace distribution $Lap(b)$: has density function

$$\Pr[z|b] = 1/2b\ e^{-|z|/b}$$

- Variance: $2b^2$
- Taking $b = 1/\varepsilon$ density at z is proportional to $e^{-\varepsilon|z|}$



Desirable Properties from a sanitization mechanism

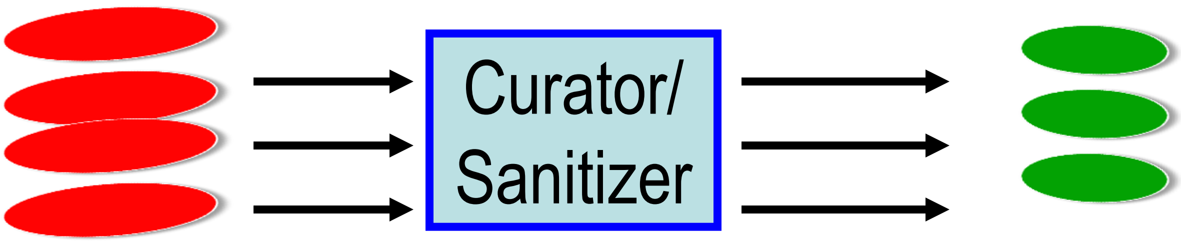
- **Composability**
  - Applying the sanitization several time yields a graceful degradation
  - q releases , each ε-DP, are q¢ε -DP

- Robustness to **side information**
  - No need to specify **exactly** what the adversary knows

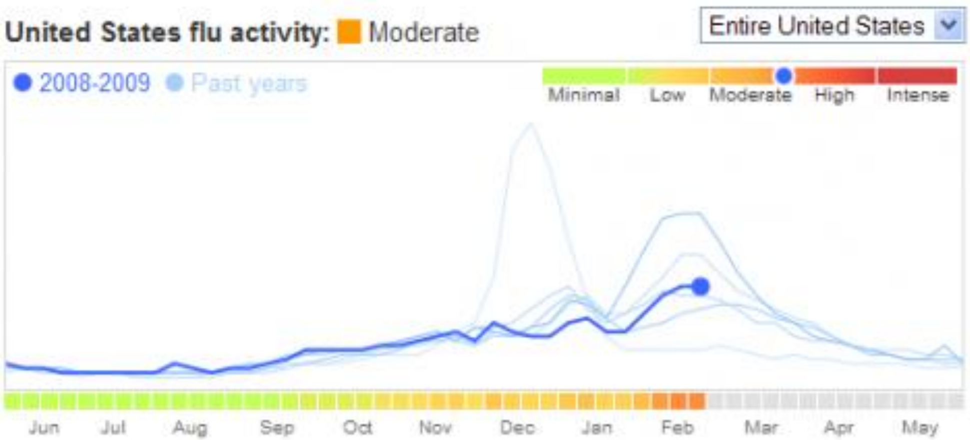
Differential Privacy: satisfies both…

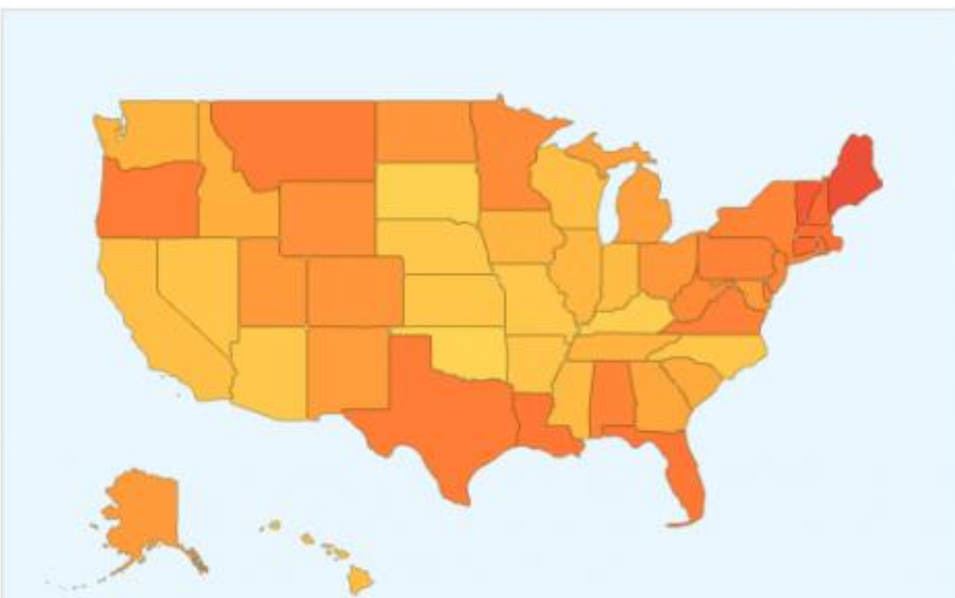
What if the data is dynamic?

- Want to handle situations where the data **keeps changing**
  - Not all data is available at the time of sanitization



Google Flu Trends





Data current through: February 15, 2009

"We've found that certain search terms are good indicators of flu activity.

Google Flu Trends uses aggregated Google search data to estimate current flu activity around the world in near real-time."

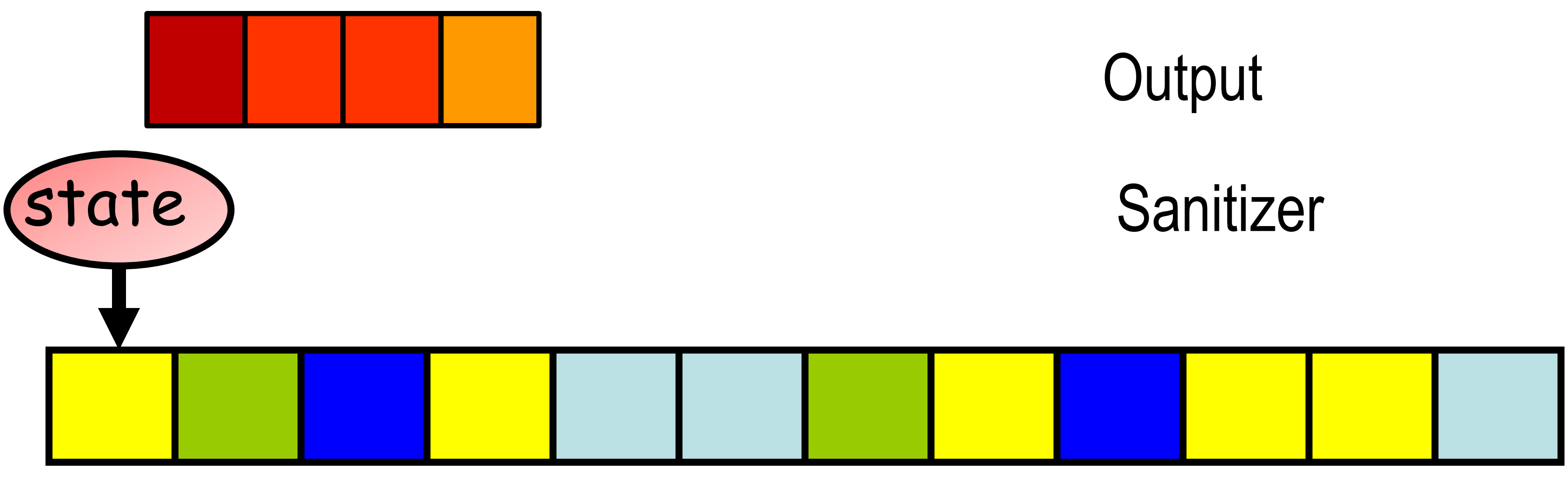
Three new issues/concepts

- **Continual Observation**
  - The adversary gets to examine the **output** of the sanitizer **all the time**
- **Pan Privacy**
  - The adversary gets to examine the **internal state** of the sanitizer. Once? Several times? All the time?
- **"User" vs. "Event" Level Protection**
  - Are the items "singletons" or are they related

Continual Output Observation

Data is a **stream** of items

- Sanitizer sees each item, updates internal state.
- Produces an output **observable to the adversary**



Continual Observation

- ***Alg*** - algorithm working on a stream of data
  - Mapping prefixes of data streams to outputs
  - Step i output σ_i

Adjacent data streams: can get from one to the other by changing **one element**

- ***Alg*** is **ε-differentially private against continual observation** if for all
  - adjacent data streams S and S'
  - for all prefixes t outputs $\sigma_1 \sigma_2 \dots \sigma_t$

S= acgtbxcde

S'= acgtbycde

$$e^{-\varepsilon} \leq \frac{\Pr[Alg(S)=\sigma_1 \sigma_2 \dots \sigma_t]}{\Pr[Alg(S')=\sigma_1 \sigma_2 \dots \sigma_t]} \leq e^{\varepsilon} \approx 1+\varepsilon$$

The Counter Problem

0/1 input stream

011001000100000011000000100101

Goal : **a publicly observable** counter, approximating the total number of **1**'s so far

Continual output: each time period, output total number of **1**'s

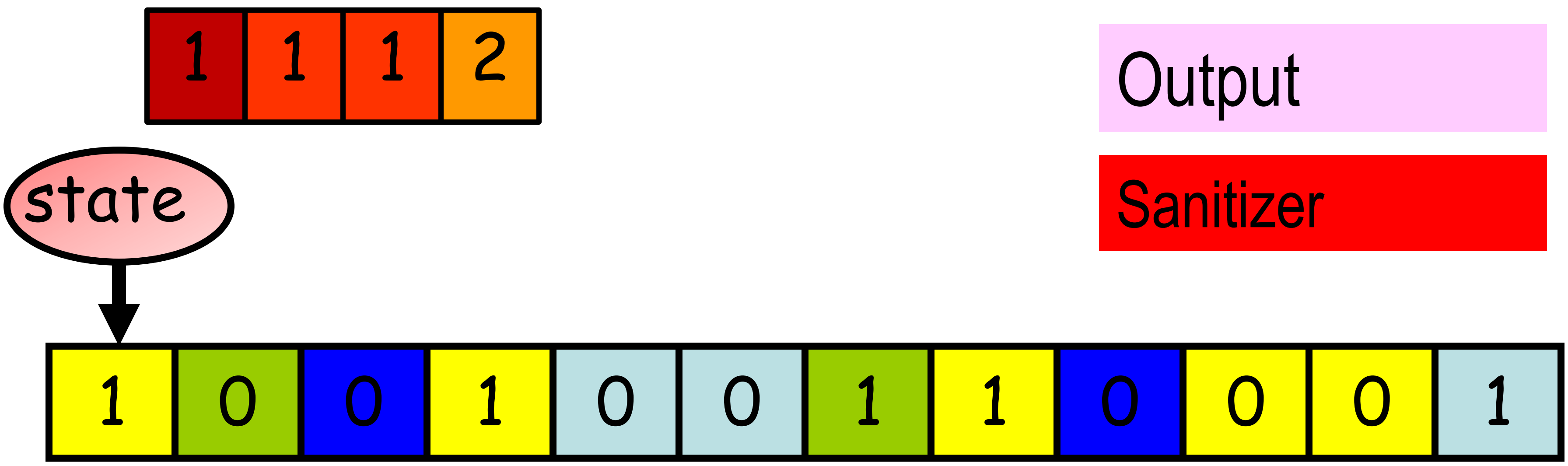
Want to hide individual increments while providing reasonable accuracy

Counters w. Continual Output Observation

Data is a **stream** of **0/1**

Sanitizer sees each x_i, updates internal state.

Produces a **value observable to the adversary**



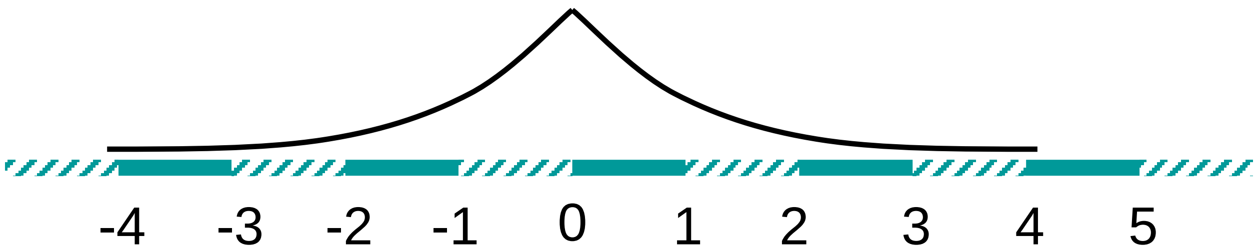
Counters w. Continual Output Observation

Continual output: each time period, output total **1**'s

Initial idea: at each time period, on input x_i 2 $\{0, 1\}$

Update counter by input x_i

Add **independent** Laplace noise with magnitude $1/\varepsilon$



Privacy: since each increment protected by Laplace noise – differentially private whether x_i is **0** or **1**

Accuracy: noise cancels out, error $\tilde{O}(\sqrt{T})$

For sparse streams: this error too high.

T – total number of time periods

Why ***So*** Inaccurate?

- Operate essentially as in **randomized response**
  - **No utilization of the state**

- Problem: we do the same operations when the stream is **sparse** as when it is **dense**
  - Want to act **differently** when the stream is dense

- The times where the counter is updated are **potential** leakage

Dynamic from Static

Accumulator measured when stream is in the time frame

- Run many accumulators in parallel:
  - each accumulator: counts number of 1's in a fixed segment of time plus noise.

Idea: apply ~~conversion of~~ static algorithms into dynamic ones

  - Value of the output counter at any point in time: **sum of the accumulators of few segments**

Only completed segments used

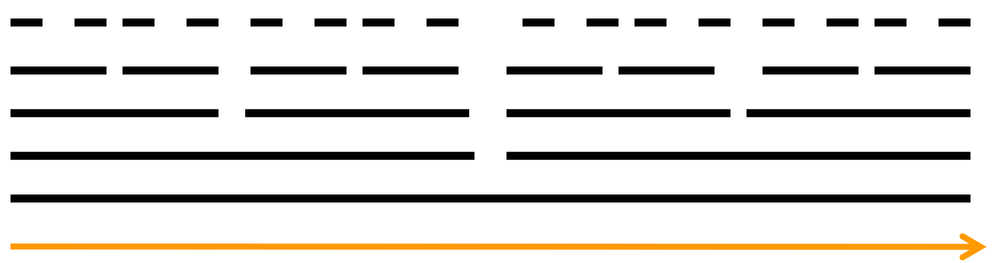
- Accuracy: depends on number of segments in **summation** and the accuracy of accumulators
- Privacy: depends on the number of accumulators that a point **influences**

The Segment Construction

Based on the bit representation:

Each point t is in d$\log t$e segments

$\Sigma_{i=1}^{t} x_i$ - Sum of at most $\log t$ accumulators



By setting ε' ¼ $\varepsilon / \log T$ **can get the desired privacy**

Accuracy: With all but negligible in T probability the error at every step t is at most $O((\log^{1.5} T)/\varepsilon))$

canceling

Pan-Privacy

“think of the children”

In privacy literature: data curator trusted

In reality:

even well-intentioned curator subject to **mission creep,** subpoena, security breach…

Goal: curator **accumulates** statistical information, but **never stores sensitive data** about individuals

Pan-privacy: algorithm private inside and out

- **internal state is privacy-preserving.**

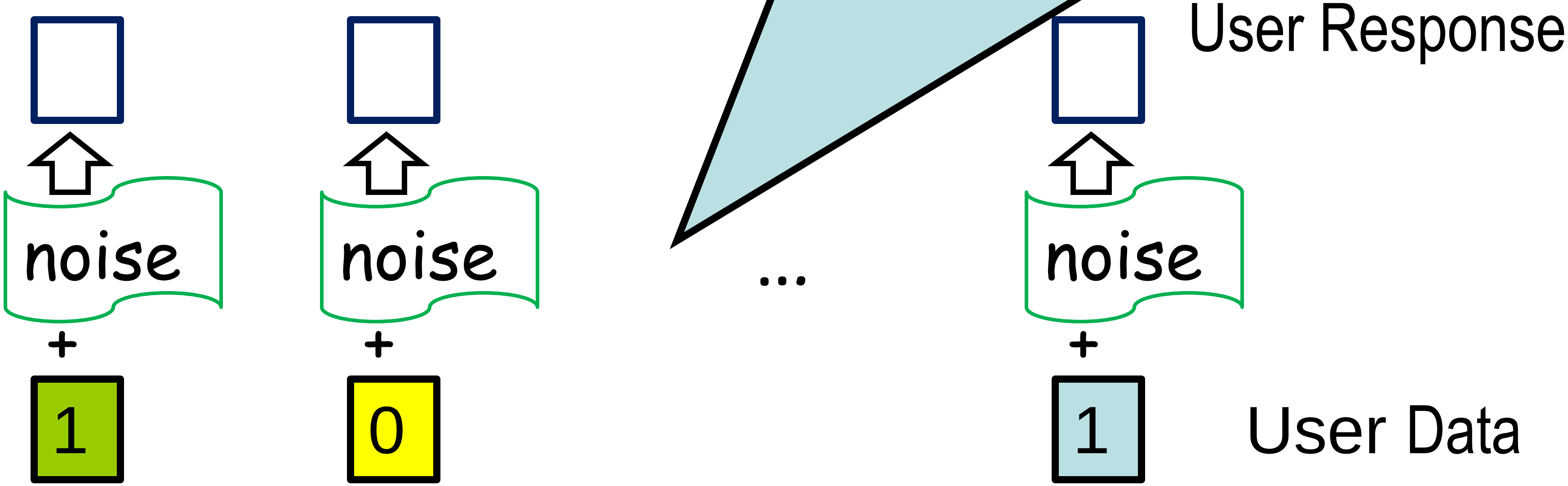
Randomized Response [Warner 1965]

Strong guarantee: **no trust in curator**

Makes sense when each user's data appears only once, otherwise **limited utility**

New idea: curator **aggregates** statistical information, but **never stores sensitive data** about individuals

popular in DB literature [MiSa06]



Example: stream of queries

- Suppose we want to compute some statistics on a query stream

(user, query)

“User level”

Do not wish to expose anything about a particular **user**

Not only about a particular pair (user, query)

“Event level”

Aggregation Without Storing Sensitive Data?

Streaming algorithms: small storage

- Information stored can still be sensitive
- "My data": many appearances, arbitrarily interleaved with those of others

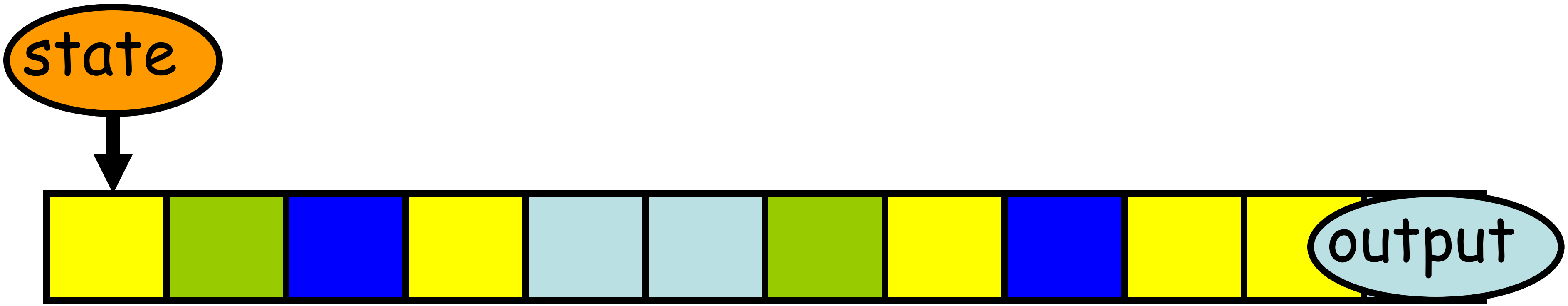
"User level"

Pan-Private Algorithm

- Private "inside and out"
- Even internal state **completely hides the appearance pattern** of any individual: presence, absence, frequency, etc.

Pan-Privacy Model

Data is **stream** of items, each item belongs to a user
Data of different users interleaved arbitrarily
Curator sees items, updates internal state, output at stream end



Can also consider multiple intrusions

Pan-Privacy

For every possible behavior of user in stream, joint distribution of **the internal state** at any single point in time and **the final output** is differentially private

Adjacency: User Level

Universe U of users whose data in the stream; x 2 U

- Streams **x-adjacent** if same projections of users onto $U \backslash \{x\}$

  Example: **axbxcxdxxxex** and **abcdxe** are x-adjacent

  - Both project to **abcde**
  - Notion of "corresponding locations" in x-adjacent streams

- U -adjacent: 9 x 2 U for which they are x-adjacent
  - Simply "adjacent," if U is understood

Note: Streams of different lengths can be adjacent

Example: Stream Density or # Distinct Elements

Universe **U** of users, estimate how many distinct users in **U** appear in data stream

Application: # distinct users who searched for "flu"

Ideas that don't work:

- **Naïve**
  Keep list of users that appeared (bad privacy and space)
- **Streaming**
  - Track random sub-sample of users (bad privacy)
  - Hash each user, track minimal hash (bad privacy)

Pan-Private Density Estimator

Inspired by randomized response.

Store for each user x 2 U a single bit b_x

Initially all b_x — Distribution D_0:
$$\begin{cases} 0 & \text{w.p. } \frac{1}{2} \\ 1 & \text{w.p. } \frac{1}{2} \end{cases}$$

When encountering x redraw b_x — Distribution D_1:
$$\begin{cases} 0 & \text{w.p. } \frac{1}{2}-\varepsilon \\ 1 & \text{w.p. } \frac{1}{2}+\varepsilon \end{cases}$$

Final output: $[($fraction of 1's in table - $\frac{1}{2})/\varepsilon] +$ **noise**

Pan-Privacy

If user never appeared: entry drawn from D_0

If user appeared **any # of times**: entry drawn from D_1

D_0 and D_1 are 4ε-differentially private

Pan-Private Density Estimator

Inspired by randomized response.

Store for each user x 2 U a single bit b_x

Initially all b_x $\begin{cases} 0 \text{ w.p. } \frac{1}{2} \\ 1 \text{ w.p. } \frac{1}{2} \end{cases}$

When encountering x redraw b_x $\begin{cases} 0 \text{ w.p. } \frac{1}{2}-\varepsilon \\ 1 \text{ w.p. } \frac{1}{2}+\varepsilon \end{cases}$

Final output: [(fraction of 1's in table - $\frac{1}{2}$)/ε] + noise

Improved accuracy and Storage

Multiplicative accuracy using hashing

Small storage using sub-sampling

Pan-Private Density Estimator

Theorem [density estimation streaming algorithm]

ε pan-privacy, multiplicative error α

space is poly($1/\alpha, 1/\varepsilon$)

What other statistics have pan-private algorithms?

Density: # of users appeared at least once

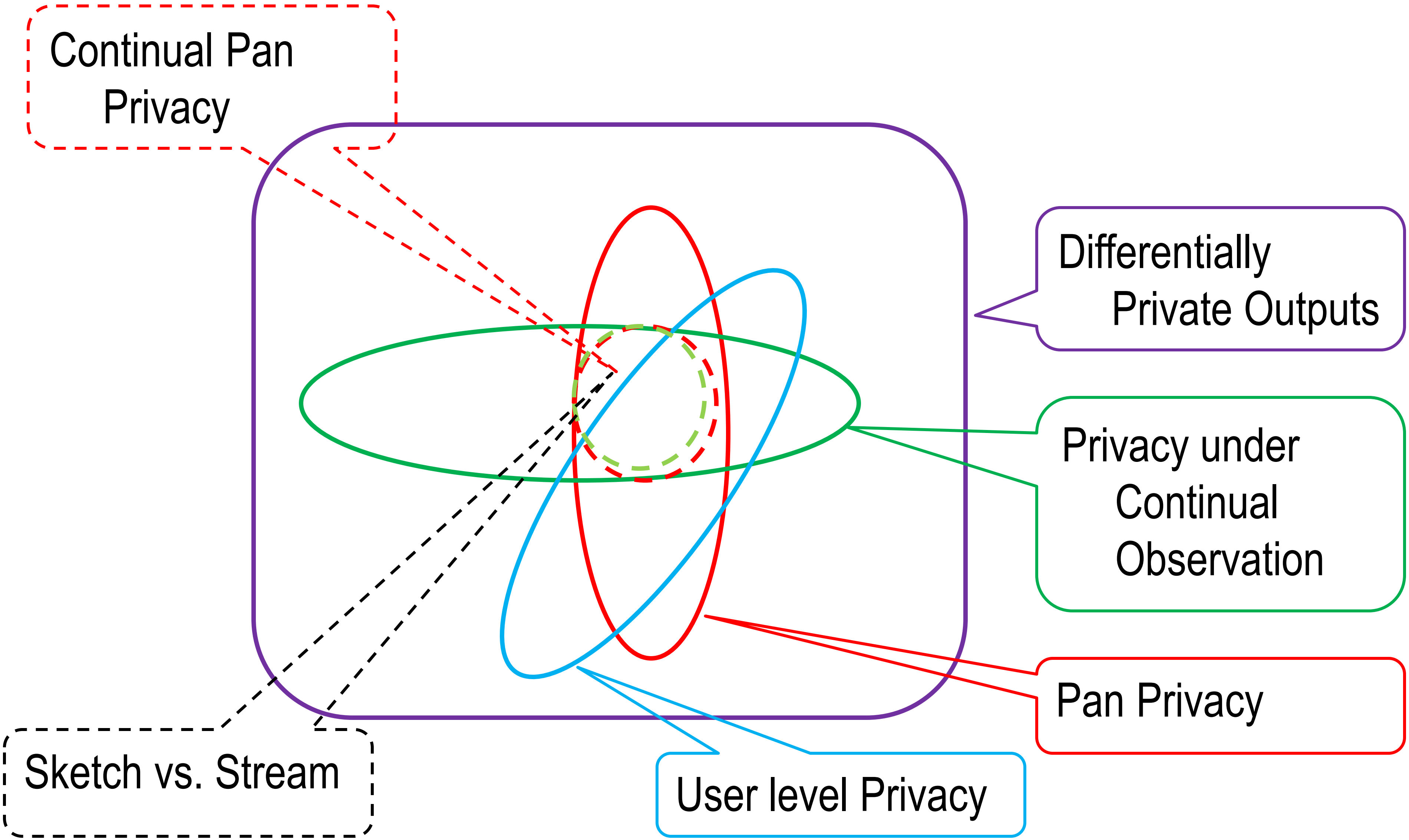
Incidence counts: # of users appearing **k** times exactly

Cropped means: mean, over users, of min(**t**,#appearances)

Heavy-hitters: users appearing at least **k** times

Petting

The Dynamic Privacy Zoo



Sources

Based on

- Cynthia Dwork, Moni Naor, Toni Pitassi, Guy Rothblum and Sergey Yekhanin, **Pan-private streaming algorithms,** ICS 2010
- Cynthia Dwork, Moni Naor, Toni Pitassi and Guy Rothblum, **Differential Privacy Under Continual Observation,** STOC 2010**.**

Pan-private Algorithms

$\overbrace{\textbf{Continual Observation}}$

- ✓ **Density:** # of users appeared at least once
- ✓ **Incidence counts**: # of users appearing **k** times exactly
- ✓ **Cropped means**: mean, over users, of
  min(**t**,#appearances)
- ✓ **Heavy-hitters**: users appearing at least **k** times

Microsoft® Research
Faculty Summit 2010