Privacy-Preserving Data Mining

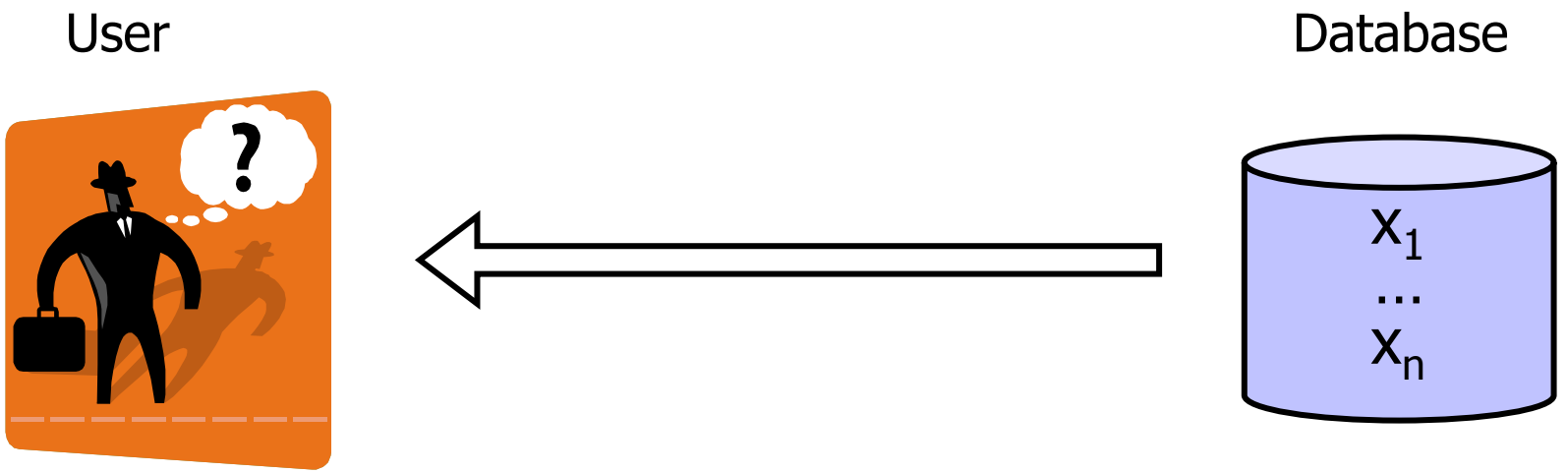
Slides mostly by

Vitaly Shmatikov

Problem definition

A user makes query to a DB

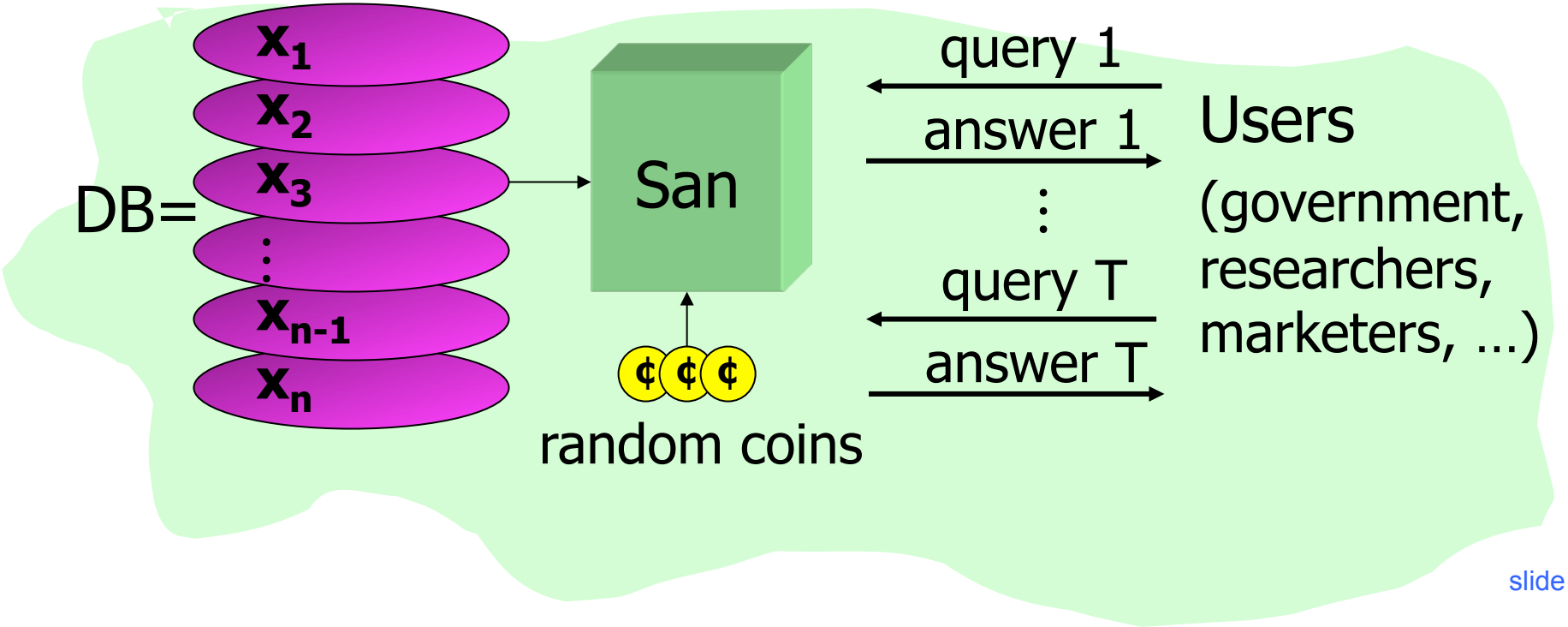
- He is allowed to make statistical queries (privacy reasons)
- He is smart so he is able to infer information he should not know by repeteatedly making queries questions
- We want to allow him to query but we want to preserve privacy of the DB



Data sanitization

Data Sanitization is the process of disguising sensitive information in databases by overwriting it with realistic looking but false data of a similar type

Data sanitization is different from K-anonymity



Examples of Sanitization Methods

- Input perturbation
  - Add random noise to database, release
- Reveal only summary statistics (not data)
  - Means, variances
  - Marginal totals
  - Regression coefficients
- Output perturbation
  - Summary statistics with noise
- Interactive versions of the above methods
  - Auditor decides which queries are OK, type of noise

Strawman Definition

- Assume $x_1,\ldots,x_n$ are drawn i.i.d. from unknown distribution
- Candidate definition: sanitization is safe if it only reveals the distribution
- Implied approach:
  - Learn the distribution
  - Release description of distribution or re-sample points
- This definition is tautological!
  - Estimate of distribution depends on data… why is it safe?

Blending into a Crowd

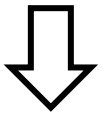
Frequency in DB or frequency in underlying population?

- Intuition: "I am safe in a group of k or more"
  - k varies (3… 6… 100… 10,000?)
- Many variations on theme
  - Adversary wants predicate g such that $0 < \#\{i \mid g(x_i)=\text{true}\} < k$
- Why?
  - Privacy is "protection from being brought to the attention of others" [Gavison]
  - Rare property helps re-identify someone
  - Implicit: information about a large group is public
    - E.g., liver problems more prevalent among diabetics

Data sanitization vs K-anonymity

- Given sanitization S, look at all databases consistent with S
- Safe if no predicate is true for all consistent databases
- k-anonymity
  - Partition D into bins
  - Safe if each bin is either empty, or contains at least k elements
- Cell bound methods
  - Release marginal sums

	brown	blue	Σ
blond	2	10	12
brown	12	6	18
Σ	14	16	



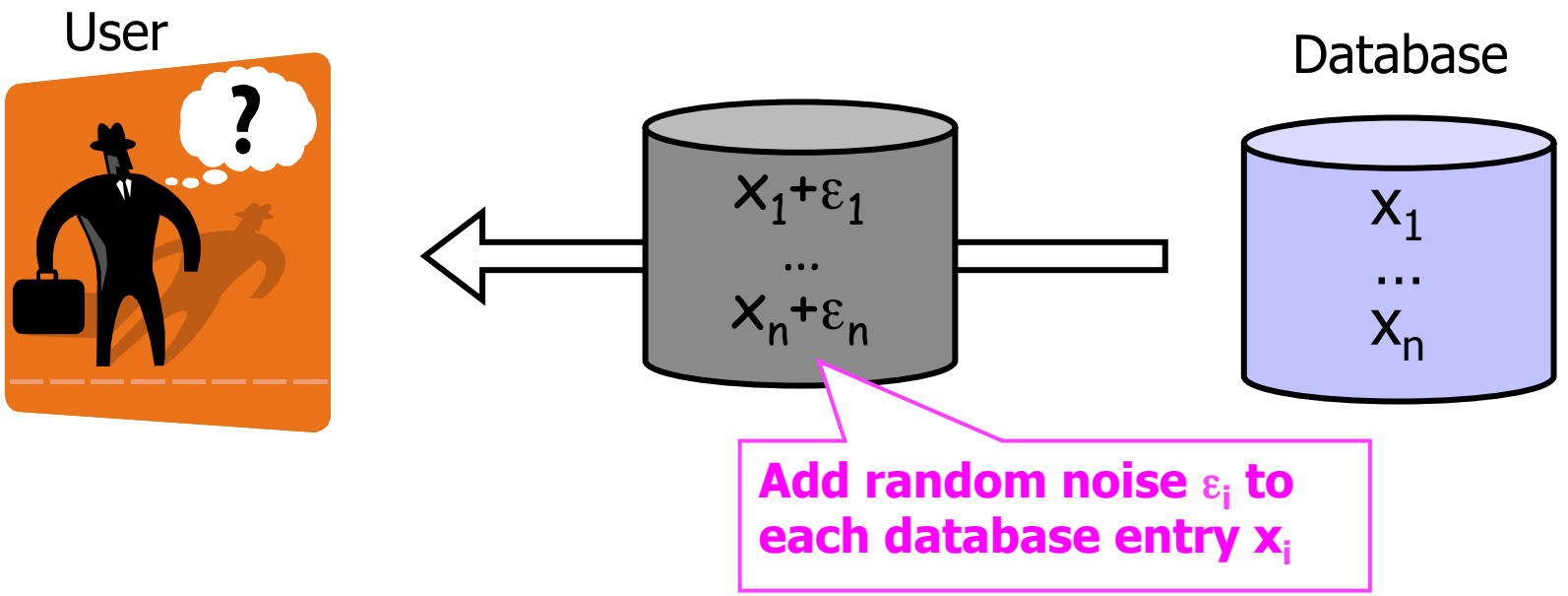
	brown	blue	Σ
blond	[0,12]	[0,12]	12
brown	[0,14]	[0,16]	18
Σ	14	16	

Summary

1. Reveal entire database, but randomize entries (either in input or answers to queries)
2. A strong definition of privacy shows that it is impossible to modify in such a way to be unable to learn something on some specific user
3. A weaker definition shows that it is possible to get positive results
4. However use of additional information can allow to know too many things

Input Perturbation

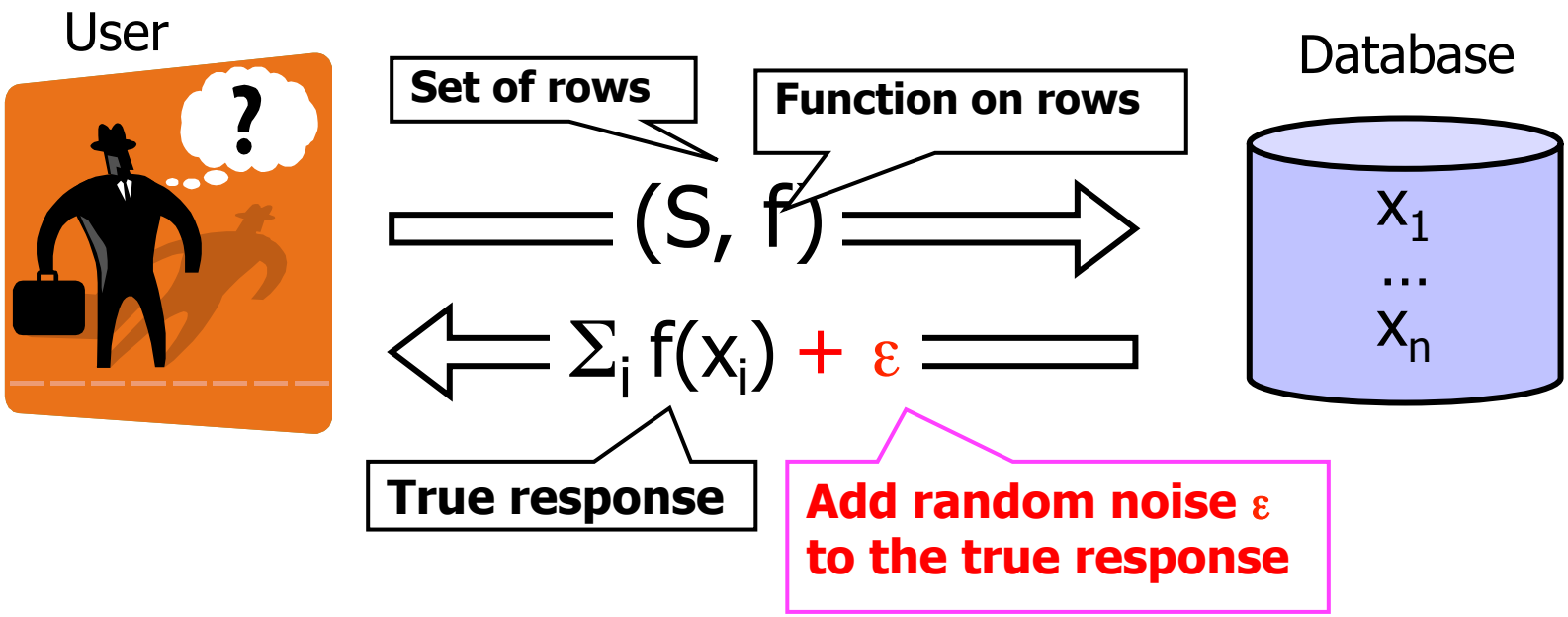
- Reveal entire database, but randomize entries



For example, if distribution of noise has mean 0, user can compute average of x_i

Output Perturbation

- Randomize response to each query



Concepts of Privacy

- Weak: no single database entry has been revealed
- Stronger: no single piece of information is revealed (what's the difference from the "weak" version?)
- Strongest: the adversary's beliefs about the data have not changed

Kullback-Leibler Distance (KL distance)

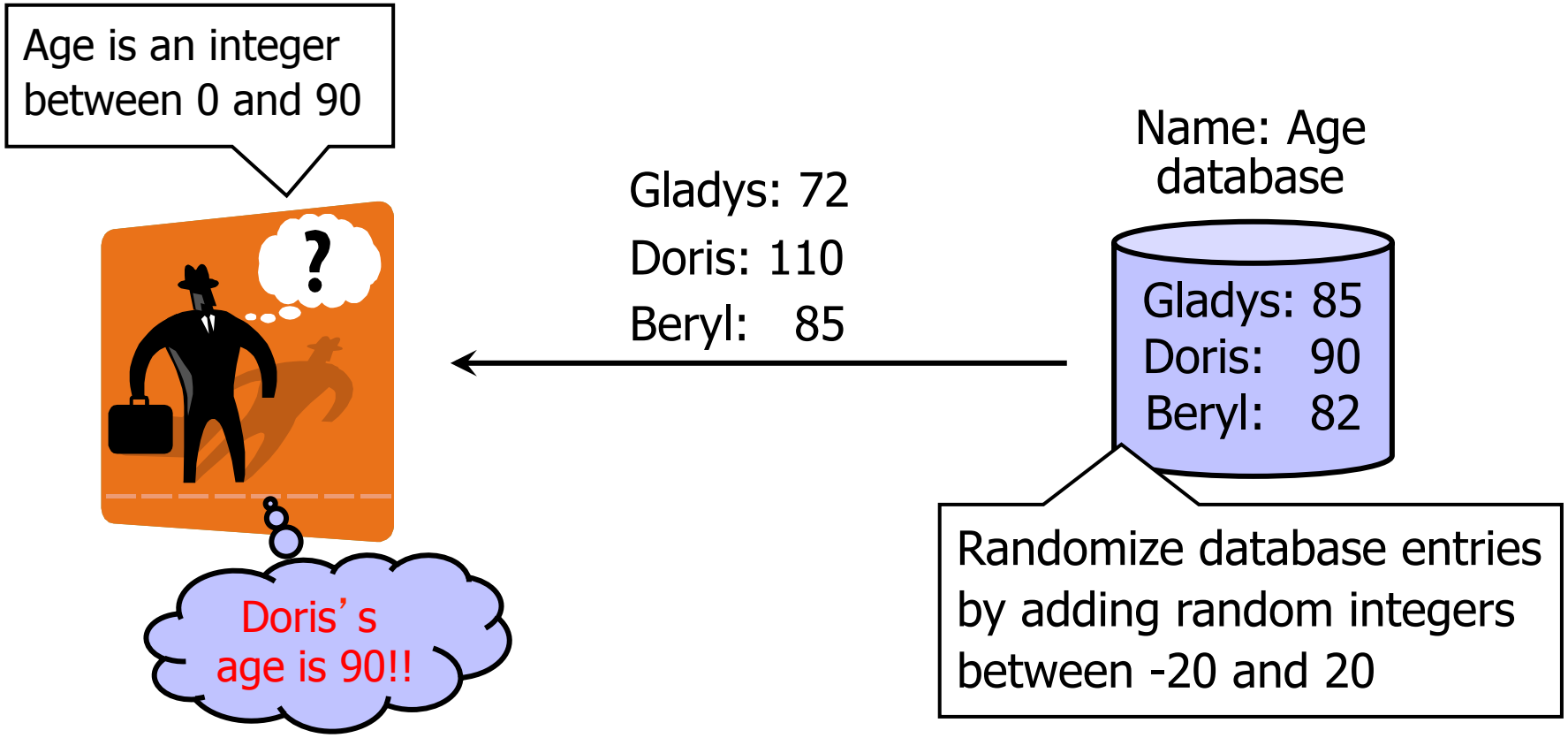
- KL distance Measures the "difference" between two probability distributions

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

- If distance is small then two distributions are similar (P and Q identical implies distance =0)
- KL measures mutual information between original and randomized databases
- Intuition: if this distance is small, then Y leaks little information about actual values of X

Why is this definition problematic?

Example



Randomization operator has to be public (why?)

Privacy Definitions

- Mutual information can be small on average, but an individual randomized value can still leak a lot of information about the original value
- Better: consider some property Q(x)
  - Adversary has a priori probability P_i that $Q(x_i)$ is true
- Privacy breach if revealing $y_i=R(x_i)$ significantly changes adversary's probability that $Q(x_i)$ is true
  - Intuition: adversary learned something about entry x_i (namely, likelihood of property Q holding for this entry)

Example

Knowing the value after randomization changes the information we have

- Example: x= {0,1,2},

  p(x=0)=0.5, p(x=1)=p(x=2)=0.25

- Consider randomization R(x)

  R(x) = x with prob. 50%; x+1 mod 3 prob 50%

- Now we know that R(x) = 0, we have that
  - p(x=0)= 3/4
  - p(x=1)=0
  - p(x=2) =1/4

Example: game three boxes

The reasoning is similar to the three box game

- There are three closed boxes: one with gold two with paper
- You choose one box
- Then one box with no gold is open
- After this you can change your box
- Question: it is convenient to change box or keep yur choice?
- Answer: change your box!!! (you double the probability of choosing the box with gold!)

Example

- Data: $0 \le x \le 1000$,

  $p(x=0)=0.01$, $p(x=i, i \neq 0)=0.00099$
- Reveal $y=R(x)$
- Three possible randomization operators R
  - $R_1(x)$ = x with prob. 20%; uniform with prob. 80%
  - $R_2(x) = x+\xi \bmod 1001$, ξ uniform in [-100,100]
  - $R_3(x) = R_2(x)$ with prob. 50%, uniform with prob. 50%
- Which randomization operator is better?

Some Properties

- $Q_1(x)$: $x=0$; $Q_2(x)$: $x \notin \{200, \ldots, 800\}$
- What are the a priori probabilities for a given x that these properties hold?
  - $Q_1(x)$: 1%, $Q_2(x)$: 40.5%
- Now suppose adversary learned that $y=R(x)=0$. What are probabilities of $Q_1(x)$ and $Q_2(x)$?
  - If $R = R_1$ then $Q_1(x)$: 71.6%, $Q_2(x)$: 83%
  - If $R = R_2$ then $Q_1(x)$: 4.8%, $Q_2(x)$: 100%
  - If $R = R_3$ then $Q_1(x)$: 2.9%, $Q_2(x)$: 70.8%

Privacy Breaches

- $R_1(x)$ leaks information about property $Q_1(x)$
  - Before seeing $R_1(x)$, adversary thinks that probability of $x=0$ is only 1%, but after noticing that $R_1(x)=0$, the probability that $x=0$ is 72%
- $R_2(x)$ leaks information about property $Q_2(x)$
  - Before seeing $R_2(x)$, adversary thinks that probability of $x \notin \{200, ..., 800\}$ is 41%, but after noticing that $R_2(x)=0$, the probability that $x \notin \{200, ..., 800\}$ is 100%
- Randomization operator should be such that posterior distribution is close to the prior distribution for <u>any</u> property

Privacy Breach: Definitions

- Q(x) is some property, ρ_1, ρ_2 are probabilities
  - ρ_1~"very unlikely", ρ_2~"very likely"
- Straight privacy breach:

  $P(Q(x)) \leq \rho_1$, but $P(Q(x) \mid R(x)=y) \geq \rho_2$
  - Q(x) is unlikely a priori, but likely after seeing randomized value of x
- Inverse privacy breach:

  $P(Q(x)) \geq \rho_2$, but $P(Q(x) \mid R(x)=y) \leq \rho_1$
  - Q(x) is likely a priori, but unlikely after seeing randomized value of x

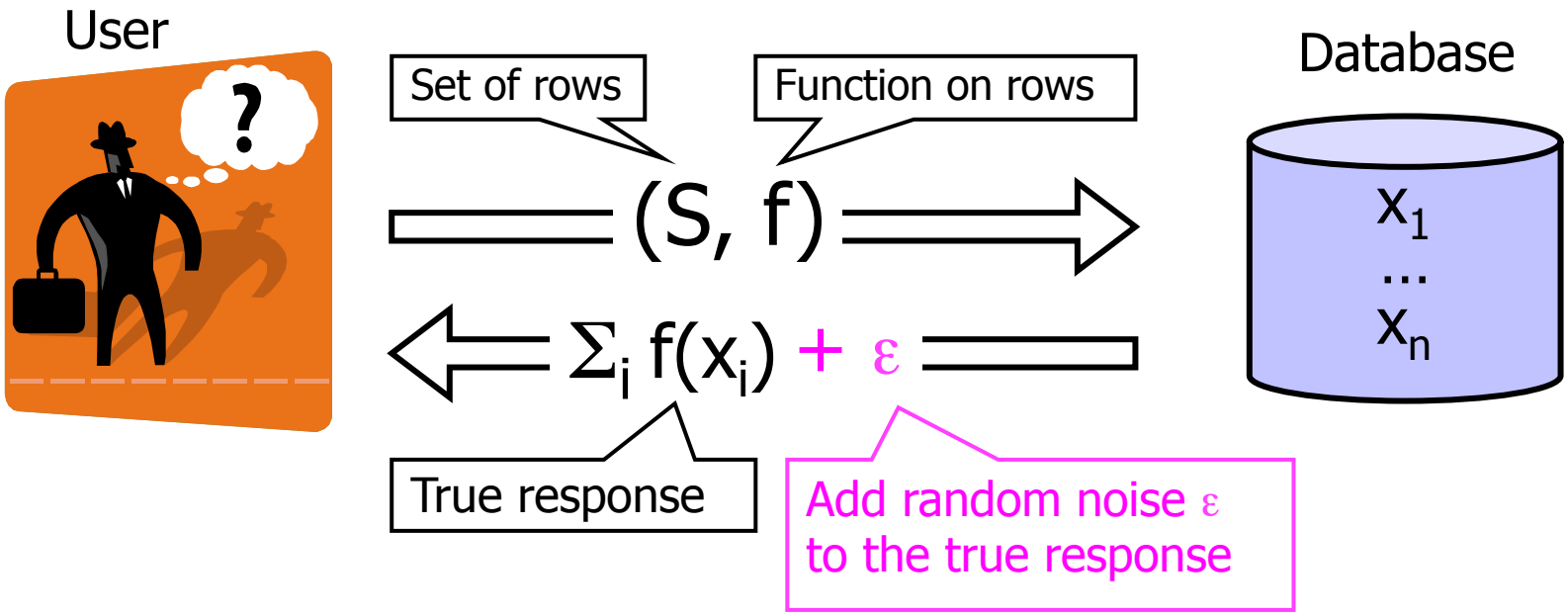
Transition Probabilities

Problems

- How to ensure that randomization operator hides every property?
  - There are $2^{|X|}$ properties
  - Often randomization operator has to be selected even before distribution P_x is known (why?)
- Idea: look at operator's transition probabilities
  - How likely is x_i to be mapped to a given y?
  - Intuition: if all possible values of x_i are equally likely to be randomized to a given y, then revealing $y=R(x_i)$ will not reveal much about actual value of x_i

Output Perturbation

- Randomize response to each query



Formally…

- Database is n-tuple $D = (d_1, d_2 \ldots d_n)$
  - Elements are not random; adversary may have a priori beliefs about their distribution or specific values
- For any predicate $f: D \rightarrow \{0,1\}$, $p^{i,f}(n)$ is the probability that $f(d_i)=1$, given the answers to n queries as well as all other entries d_j for $j \neq i$
  - $p^{i,f}(0)$=a priori belief, $p^{i,f}(t)$=belief after t answers
  - Why is adversary given all entries except d_i?
- For each query we assume that $p^{i,f}(t)$=belief after t answers increases
  - From raw probability to "belief"

Privacy Definition Revisited

- Goal: after each query, adversary's gain in knowledge about any individual database entry should be small (otherwise adversary will know with certainty)
  - Gain in knowledge about d_i as the result of $(n+1)^{st}$ query = increase from $conf(p^{i,f}(n))$ to $conf(p^{i,f}(n+1))$
  - What about if the same query is repeated many times? Same answer or different answer?
- (ε,δ,T)-privacy: for every set of independent a priori beliefs, for every d_i, for every predicate f, with at most T queries

$$\Pr[conf(p_T^{i,f}) - conf(p_0^{i,f}) > \varepsilon] \leq \delta$$

Limits of Output Perturbation

Dinur and Nissim established fundamental limits on output perturbation (PODS 2003)

- Let n be the size of the database (# of entries)
- If $O(n^{1/2})$ perturbation applied, adversary can extract entire database after poly(n) queries
- …but even with $O(n^{1/2} \log n)$ perturbation, it is unlikely that user can learn anything useful from the perturbed answers (too much noise)

Summary

1. Reveal entire database, but randomize entries (either in input or answers to queries)
2. A strong definition of privacy shows that it is impossible to modify in such a way to be unable to learn something on some specific user
3. A weaker definition (differential privacy) shows that it is possible to get positive results
4. However use of additional information can allow to know too many things

Classical Intuition for Privacy

- Dalenius: "If the release of statistics S makes it possible to determine the value [of private information] more accurately than is possible without access to S, a disclosure has taken place." [Dalenius 1977]
  - Privacy means that anything that can be learned about a respondent from the statistical database can be learned without access to the database
- Similar to semantic security of encryption
  - Anything about the plaintext that can be learned from a ciphertext can be learned without the ciphertext

Impossibility Result

[Dwork]

- Dwork: a natural formalization of Dalenius' goal cannot be achieved if the database is useful.
- The key obstacle is the *side information* that may be available to an adversary. The results hold under very general conditions regarding the database, the notion of privacy violation.
- Example
  - Vitaly knows that Alex Benn is 2 inches taller than the average Russian
  - DB allows computing average height of a Russian
  - This DB breaks Alex's privacy according to this definition... even if his record is not in the database!

New Intution for Privacy

IMPOSSIBLE

Privacy means that anything that can be learned about a respondent from the statistical database can be learned without access to the database

SECOND APPROACH (DIFFERENTIAL PRIVACY)

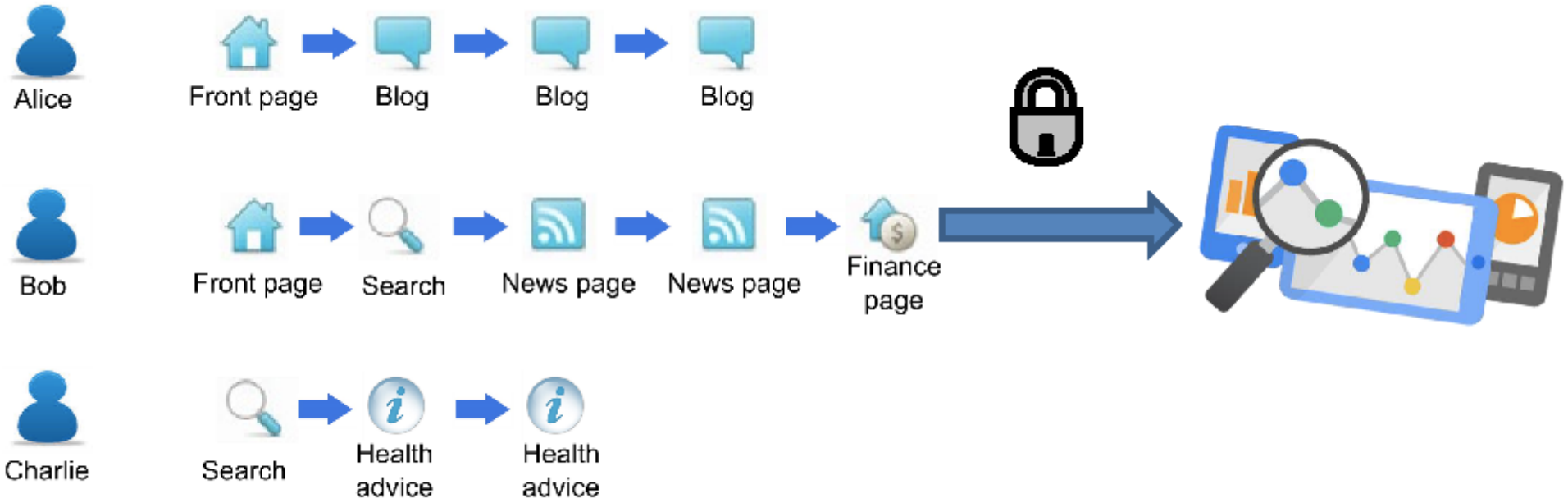
Whatever is learned about one respondent A would be learned regardless of whether or not A participates to the database

Differential Privacy

- Promise: an individual will not be affected, adversely or otherwise, by allowing his/her data to be used in any study or analysis, no matter what other studies, datasets, or information sources, are available
- Paradox: learning nothing about an individual while learning useful statistical information about a population

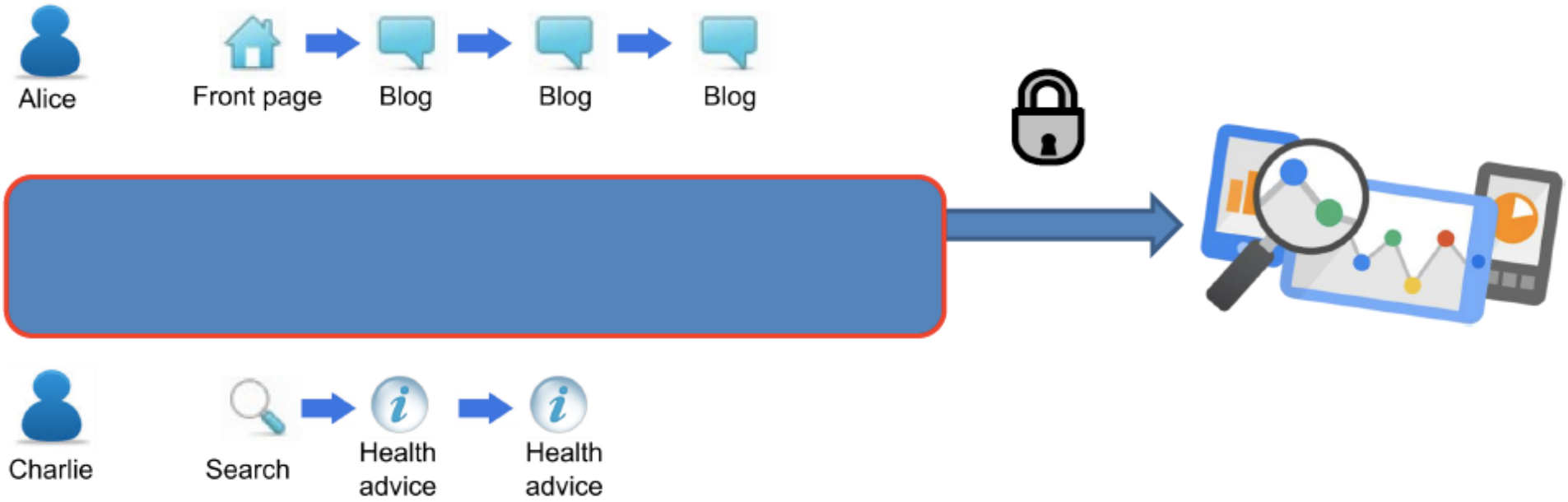
Differential Privacy

- Statistical outcome is indistinguishable regardless whether a particular user (record) is included in the data



Differential Privacy

- Statistical outcome is indistinguishable regardless whether a particular user (record) is included in the data



- Similar to indistinguishabilityfor encryption: input messages are indistinguishable from encrypted messages

Differential Privacy

For every pair of inputs D1 D2 that differ in one row

For every output O

D1 D2 O

If algorithm A satisfies differential privacy then

$$\frac{\Pr[A(D1) = O]}{\Pr[A(D2) = O]} < \exp(\varepsilon) \quad (\varepsilon > 0)$$

Intuition: adversary should not be able to use output O to distinguish between any D1 and D2

Differential Privacy

Why pairs of datasets that differ in one row?

For every pair of inputs D1 D2 that differ in one row

For every output O

D1 D2

O

Simulate the presence or absence of a single record

Differential Privacy

Why all pairs of datasets …?

For every pair of inputs D1 D2 that differ in one row

D1 D2

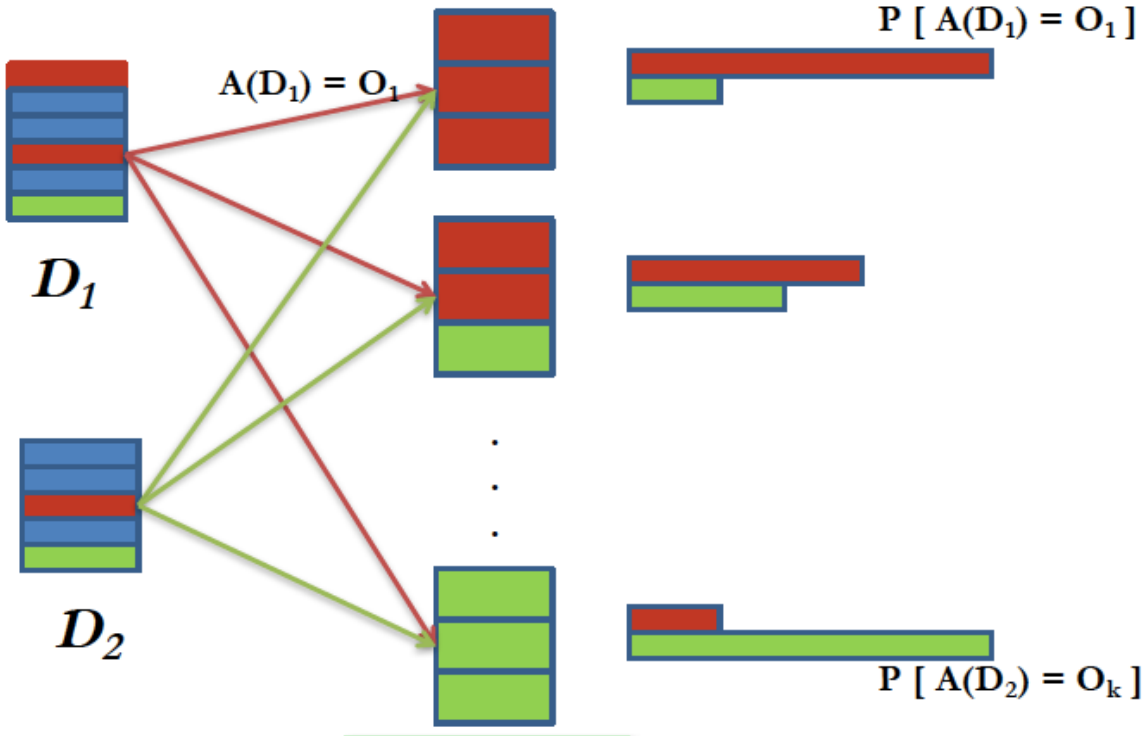
For every output O

O

Guarantee holds no matter what the other records are

Differential Privacy

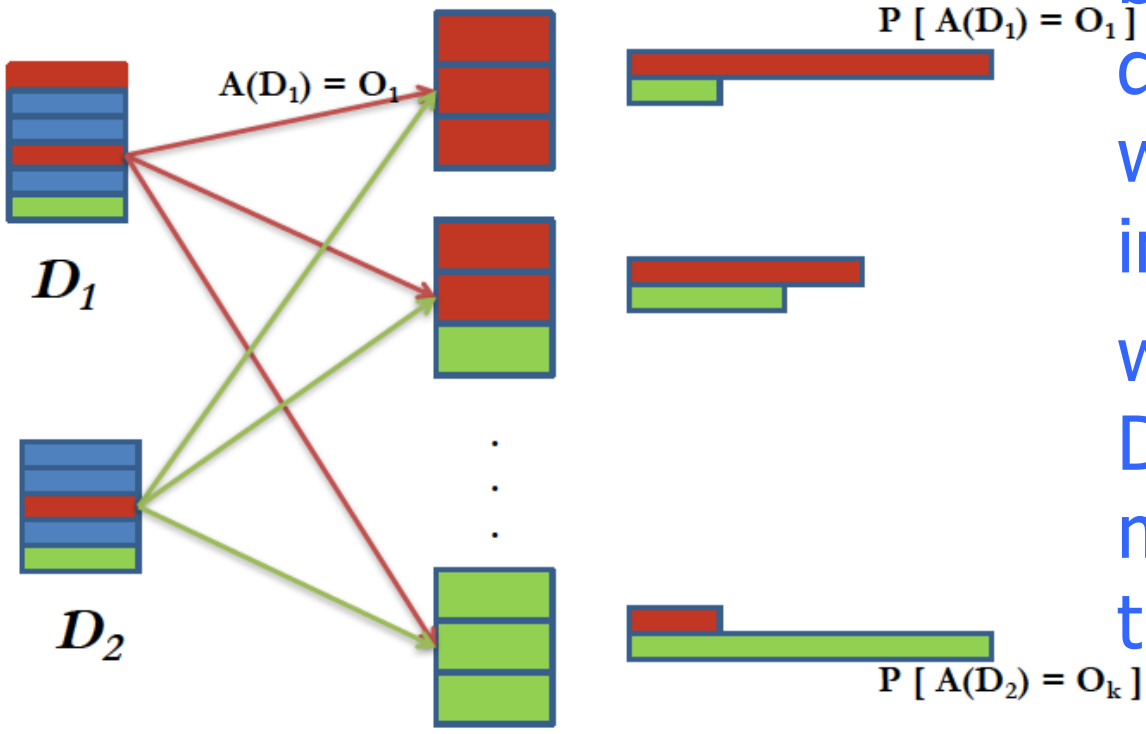
Why all outputs?



Set of all outputs

Differential Privacy

Why all outputs?



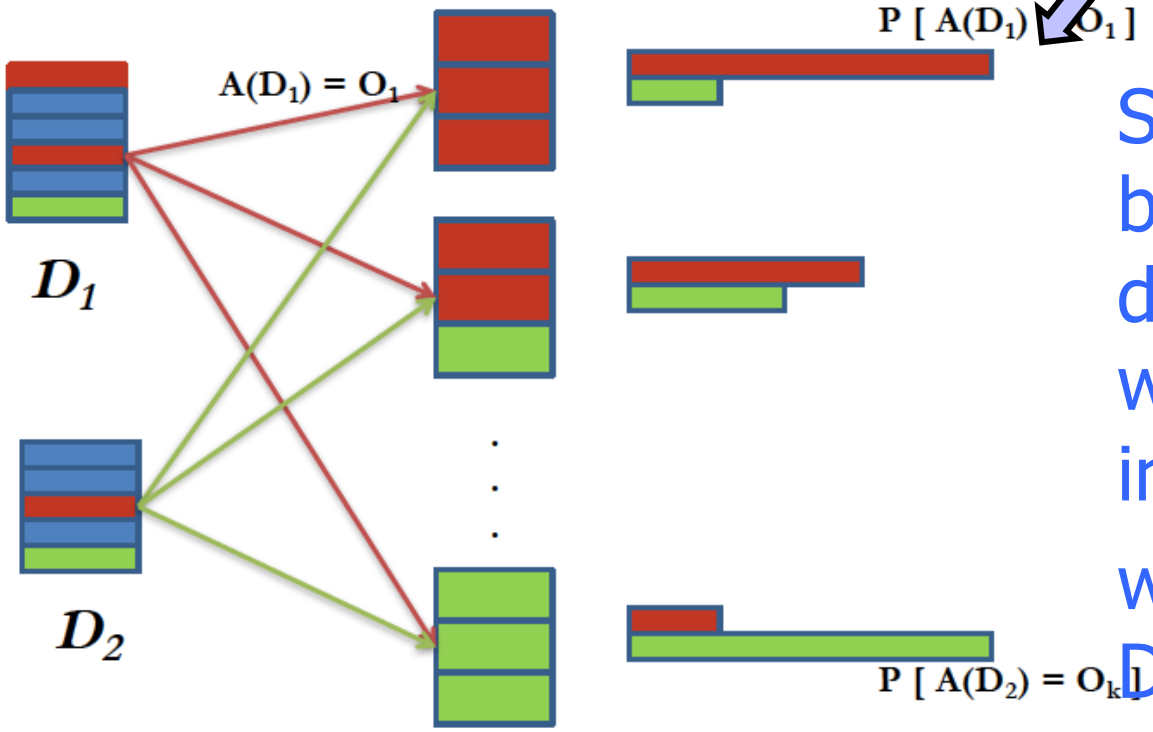
Should not be able to distinguish whether input was D1 or D2 no matter what the output

Set of all outputs

Differential Privacy

Why all outputs?

Worst case: max discrepbancy



Set of all outputs

Should not be able to distinguish whether input was D1 or D2 no matter what the output

Differential Privacy: ε parameter

For every pair of inputs D1 D2 that differ in one row

For every output O

If algorithm A satisfies differential privacy then

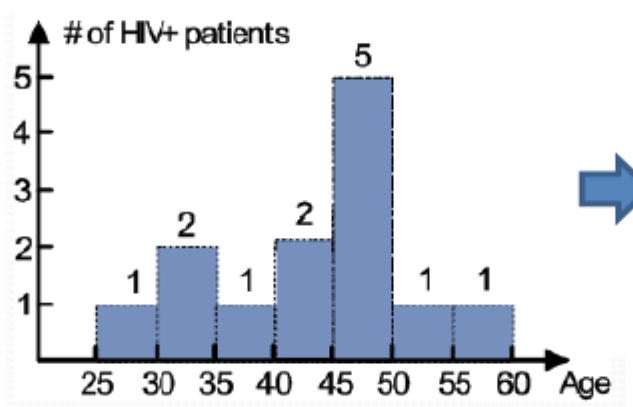
$$\frac{\Pr[A(D1) = O]}{\Pr[A(D2) = O]} < \exp(\varepsilon) \quad (\varepsilon > 0)$$

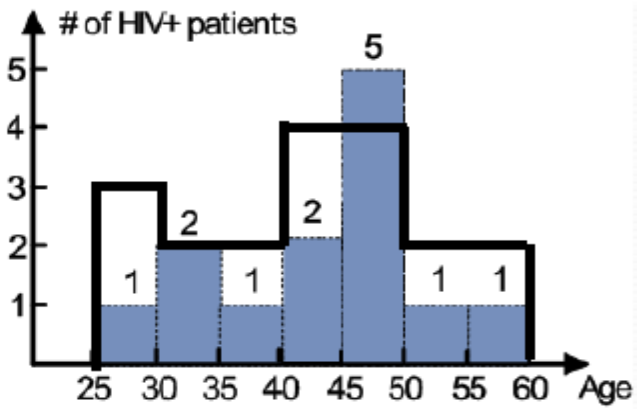
Controls the degree to which D1 and D2 can be distinguished.

Smaller ε gives more privacy (and worse utility)

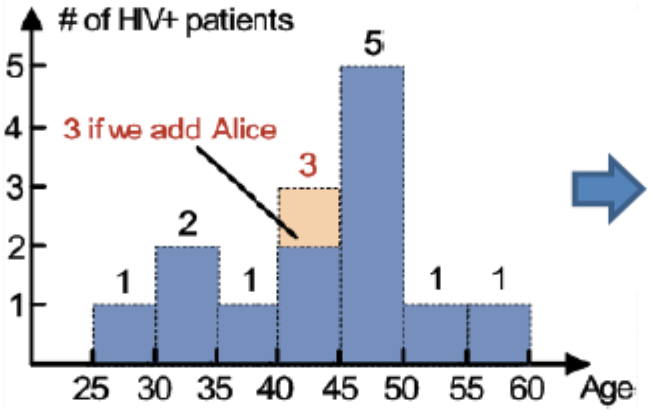
Differential Privacy: example

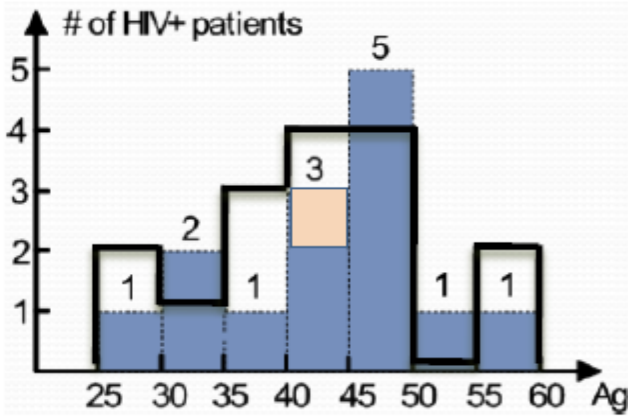
Name	Age	HIV+
Frank	42	Y
Bob	31	Y
Mary	28	Y
Dave	43	N
…	…	…





Name	Age	HIV+
Alice	43	Y
Frank	42	Y
Bob	31	Y
Mary	28	Y
Dave	43	N
…	…	…



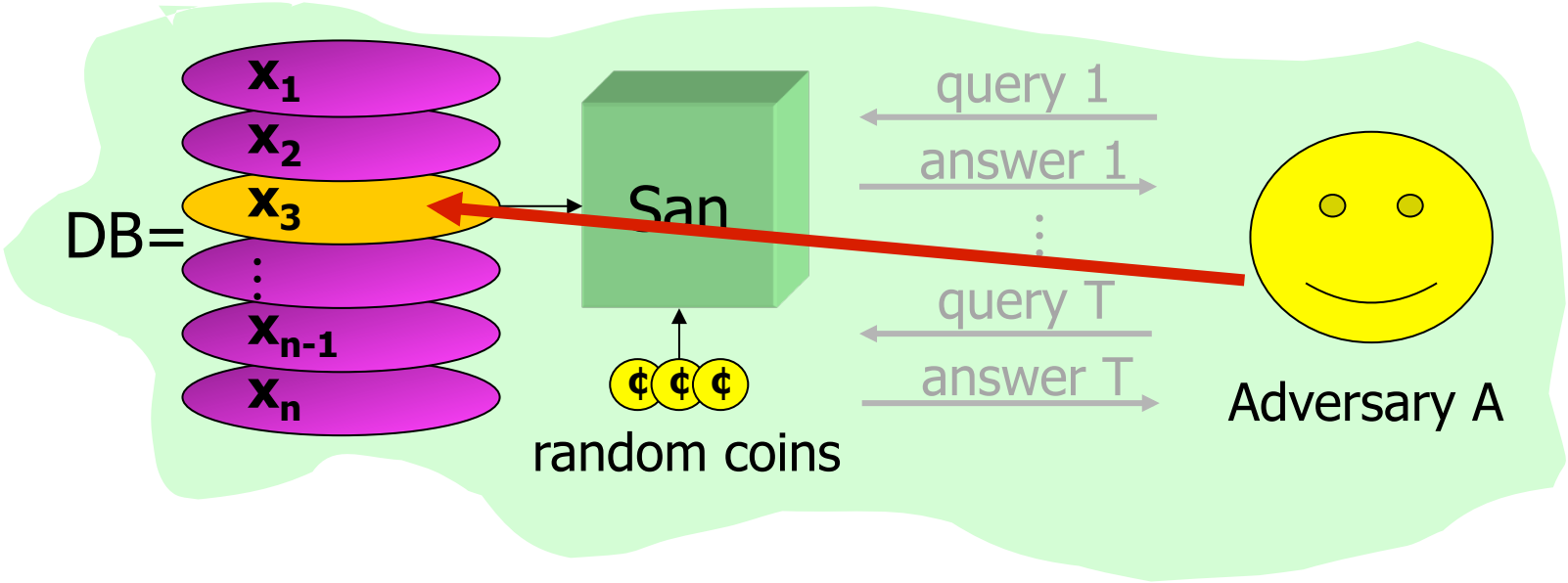


Original records

Original histogram

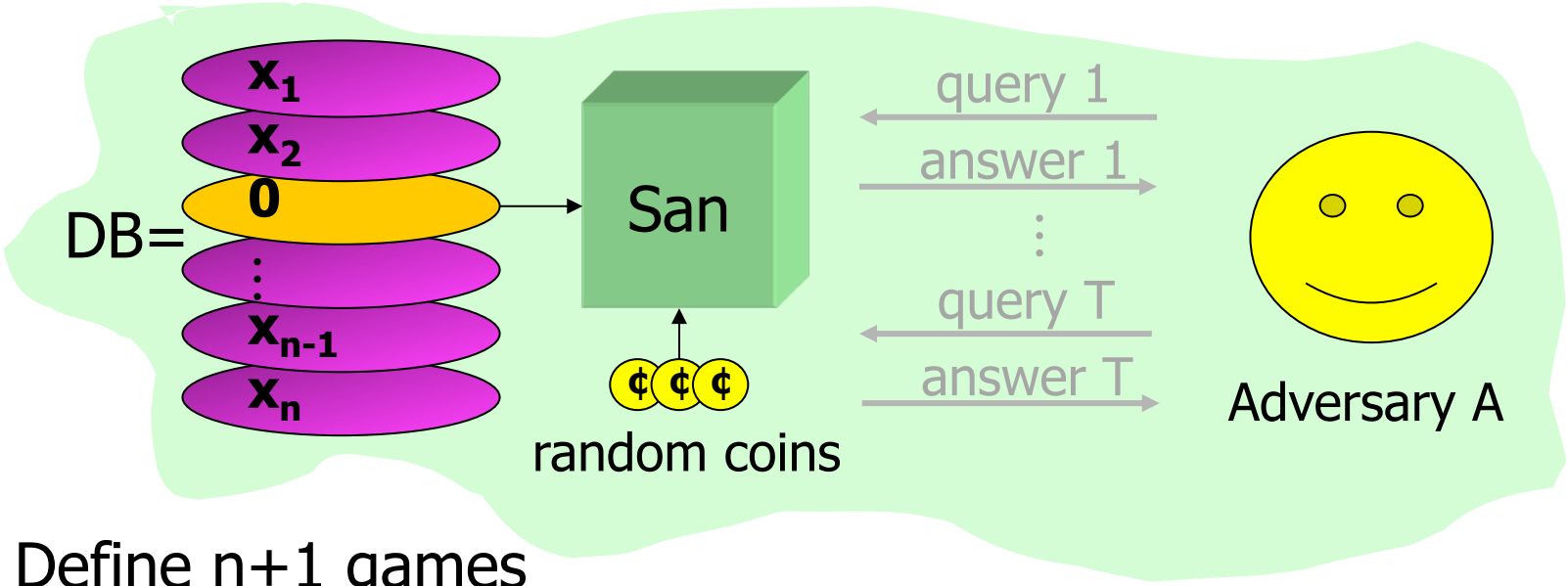
Perturbed histogram with differential privacy

Differential Privacy:



- Example with Russians and Alex Benn
  - Adversary learns Alex's height even if he is not in the database
- Intuition: "Whatever is learned would be learned regardless of whether or not Alex participates"
  - Dual: Whatever is already known, situation won't get worse

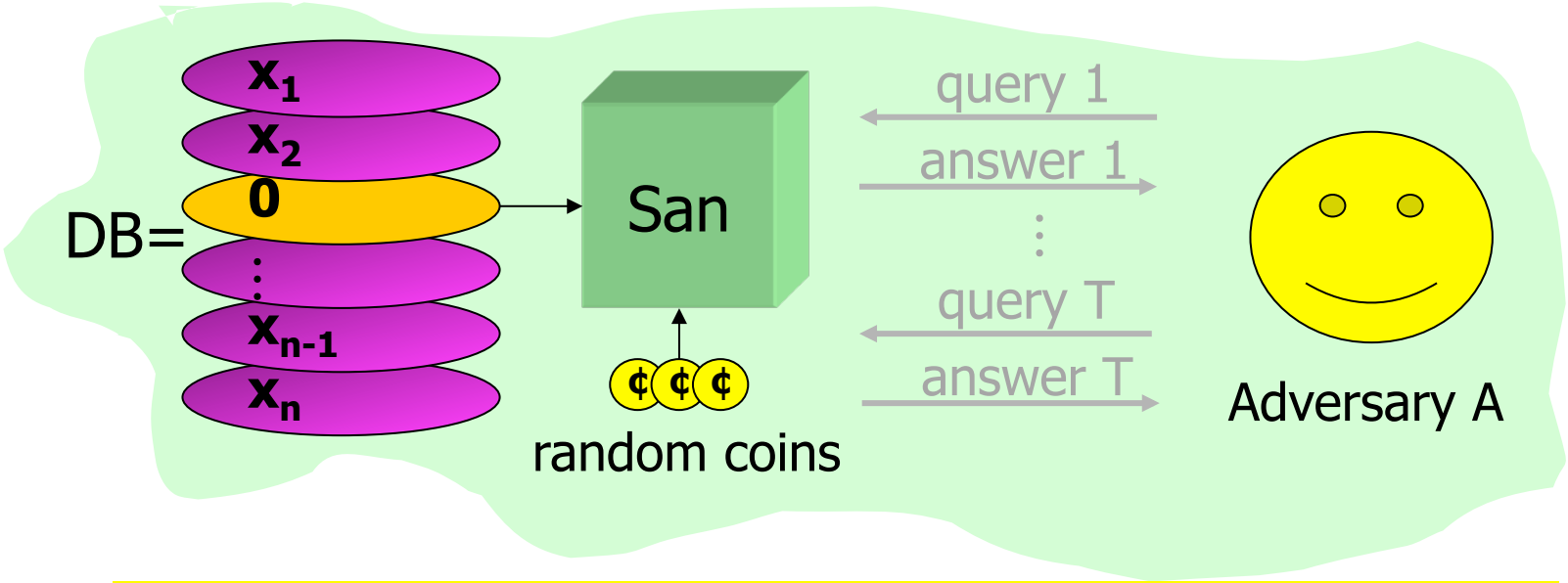
Differential Privacy (2)



- Define n+1 games
  - Game 0: Adv. interacts with San(DB) – all data are present
  - Game i: Adv. interacts with San(DB_{-i}); DB_{-i} = (x_1,…,x_{i-1},0,x_{i+1},…,x_n)
- Given S and prior p() on DB, define n+1 posterior distrib's

$$p_i(DB|S) = p(DB|S \text{ in Game } i) = \frac{p(San(DB_{-i}) = S) \times p(DB)}{p(S \text{ in Game } i)}$$

Differential Privacy (3)



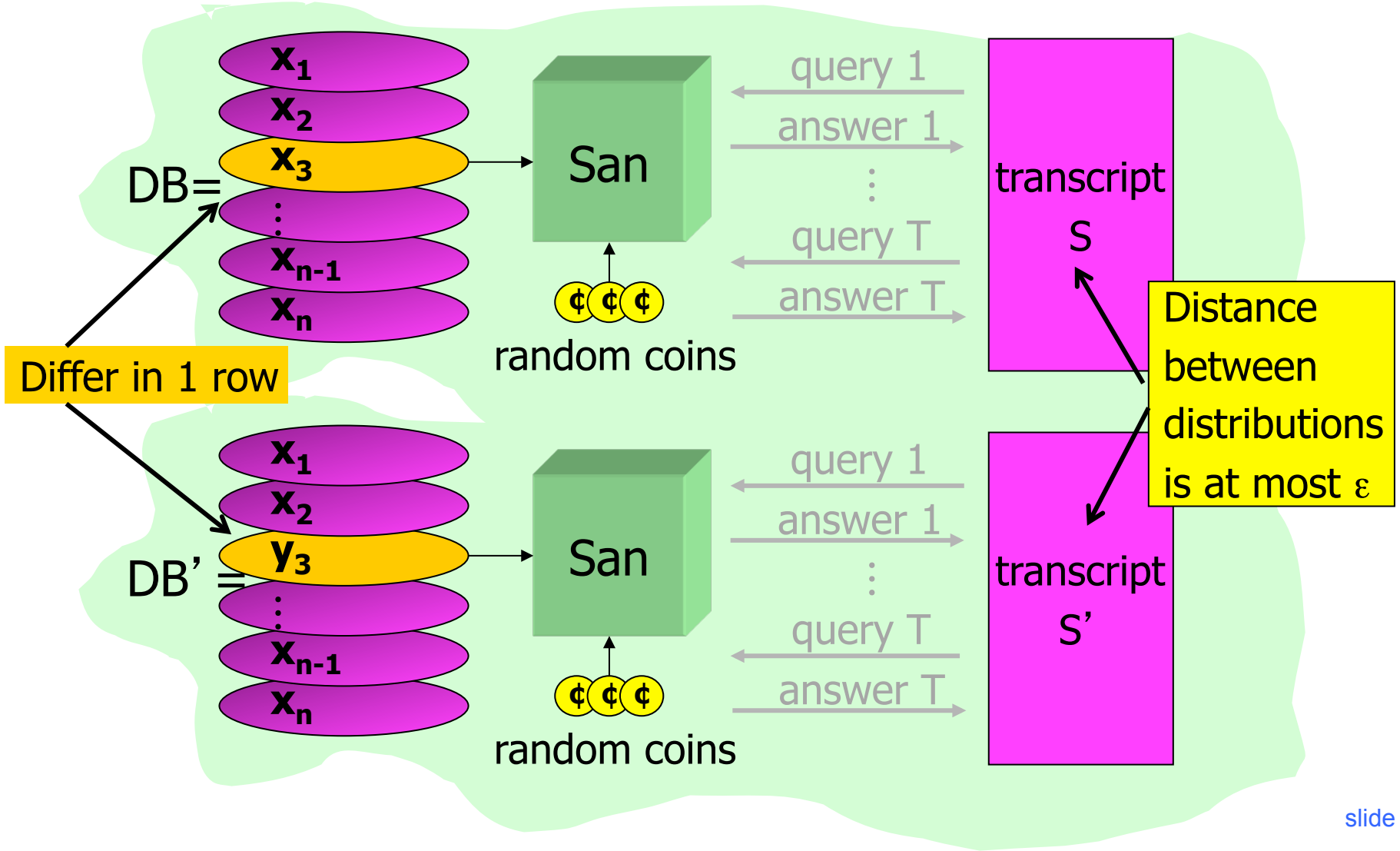
Definition: San is safe if

∀ prior distributions p(¢) on DB,

∀ transcripts S, ∀ i =1,…,n

StatDiff($p_0(¢|S)$, $p_i(¢|S)$) ≤ ε

Indistinguishability



Formalizing Indistinguishability

Definition: San is ε-indistinguishable if

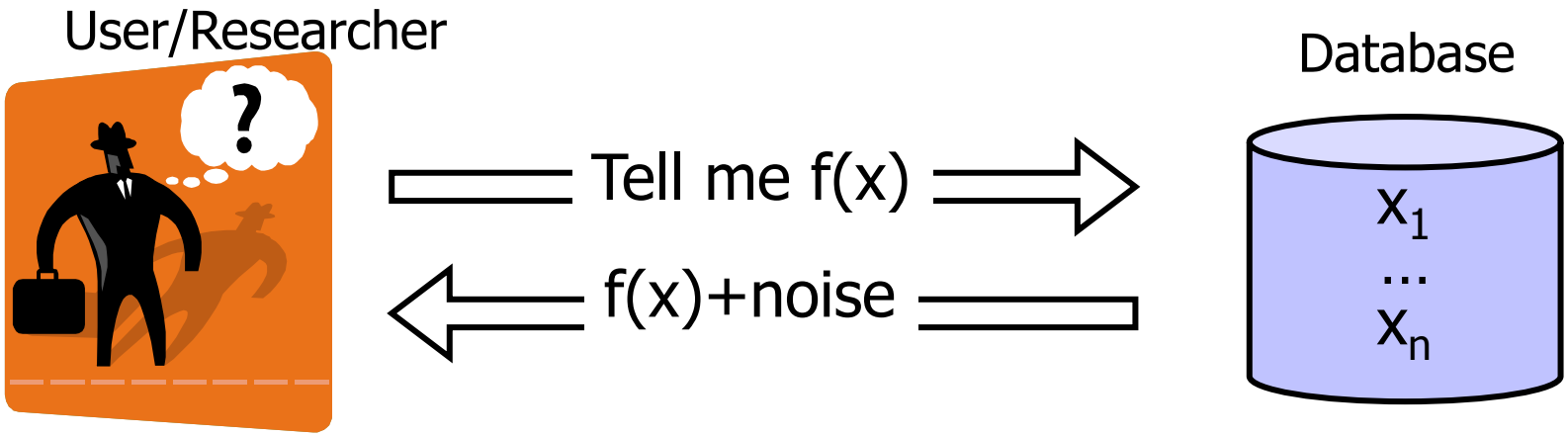
$\forall$ A, $\forall$ DB, DB' which differ in 1 row, $\forall$ sets of transcripts S

$$p(\text{San}(DB) \in S) \in (1 \pm \varepsilon)\, p(\text{San}(DB') \in S)$$

Equivalently, $\forall$ S:

$$\frac{p(\text{San}(DB) = S)}{p(\text{San}(DB') = S)} \in 1 \pm \varepsilon$$

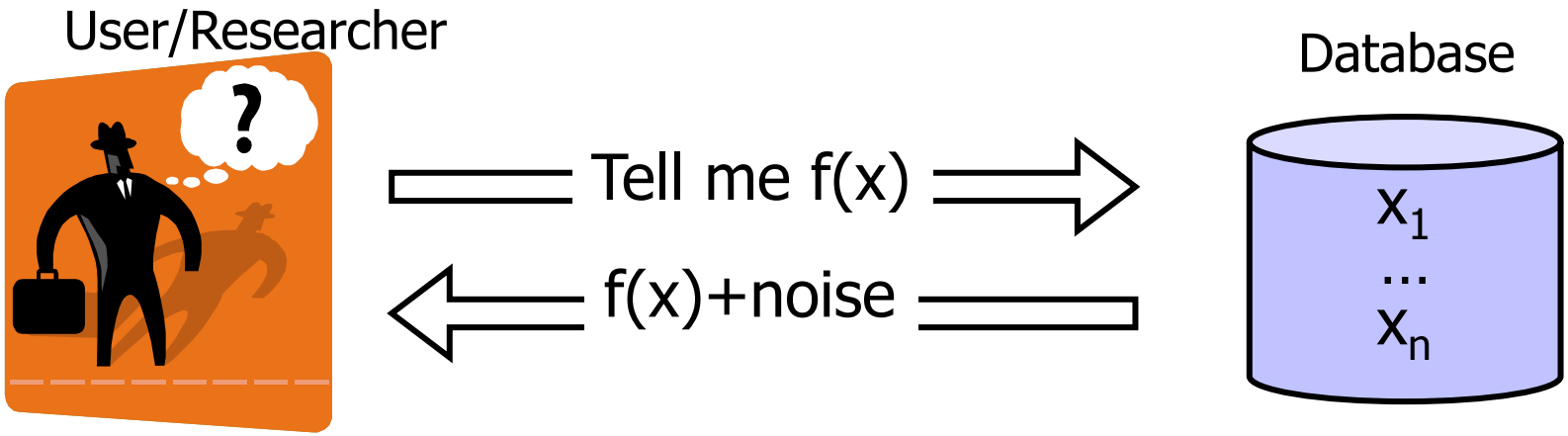
Diff. Privacy in Output Perturbation



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; so they are useful to the user
- Privacy depends on the noise
- Larger noise → higher privacy & less useful answer

Diff. Privacy in Output Perturbation

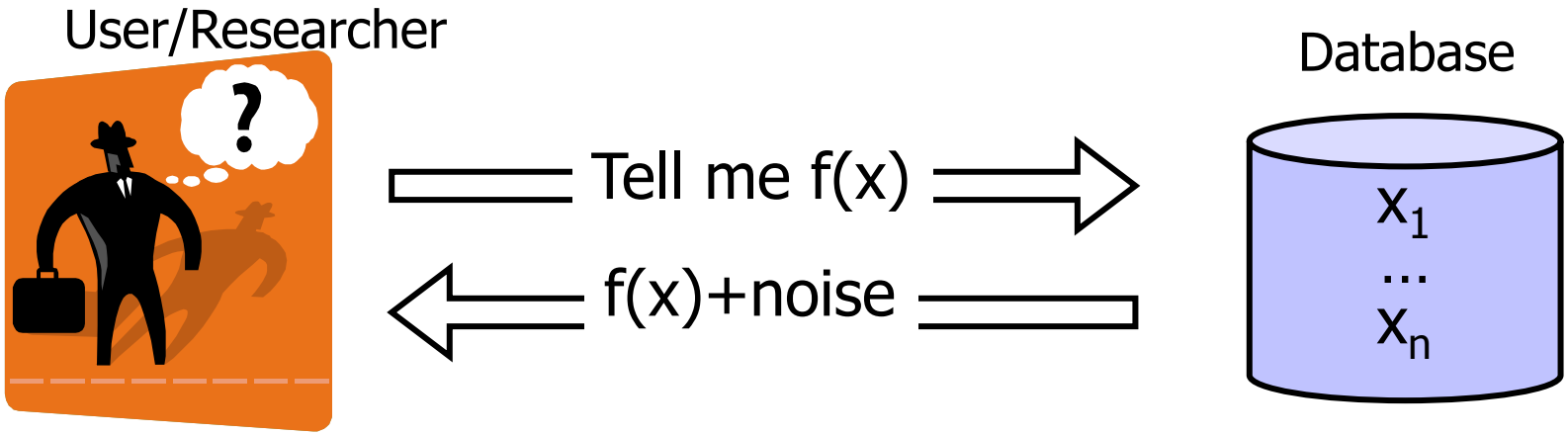


- Intuition: f(x) can be released accurately when f is insensitive to individual entries x_1, … x_n
- Sensitivity: Consider a query q: I → R. S(q) is the smallest number s.t. for any neighboring tables D, D’,

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

- Thm: If sensitivity of the query is S, then the following guarantees ε-differential privacy $\lambda = S/\varepsilon$

Diff. Privacy in Output Perturbation



- Intuition: f(x) can be released accurately when f is insensitive to individual entries x_1, ... x_n
- Global sensitivity $GS_f = \max_{\text{neighbors } x,x'} ||f(x) - f(x')||_1$
  - Example: $GS_{average} = 1/n$ for sets of bits
- Theorem: $f(x) + Lap(GS_f / \varepsilon)$ is ε-indistinguishable
  - Lap: Noise generated from Laplace distribution

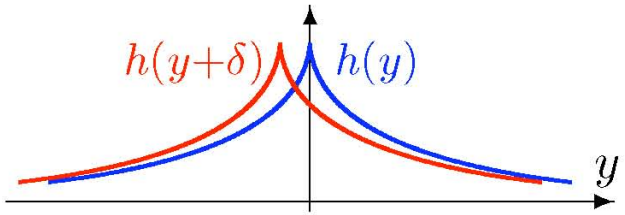
Lipschitz constant of f

Sensitivity with Laplace Noise

Theorem

If $A(x) = f(x) + \mathsf{Lap}\left(\frac{\mathsf{GS}_f}{\varepsilon}\right)$ *then* A *is* ε*-indistinguishable.*

Laplace distribution $\mathsf{Lap}(\lambda)$ has density $h(y) \propto e^{-\frac{\|y\|_1}{\lambda}}$



Sliding property of $\mathsf{Lap}\left(\frac{\mathsf{GS}_f}{\varepsilon}\right)$: $\frac{h(y)}{h(y+\delta)} \leq e^{\varepsilon \cdot \frac{\|\delta\|}{\mathsf{GS}_f}}$ for all y, δ

Proof idea:

$A(x)$: blue curve

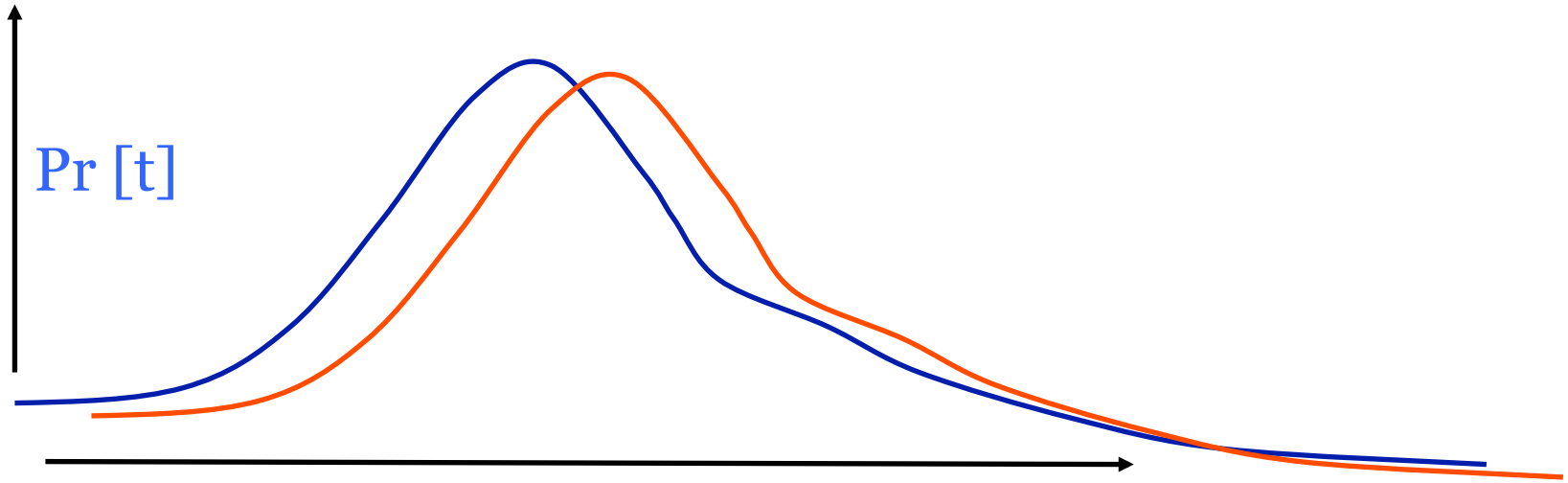
$A(x')$: red curve

$\delta = f(x) - f(x') \leq \mathsf{GS}_f$

Differential Privacy: Summary

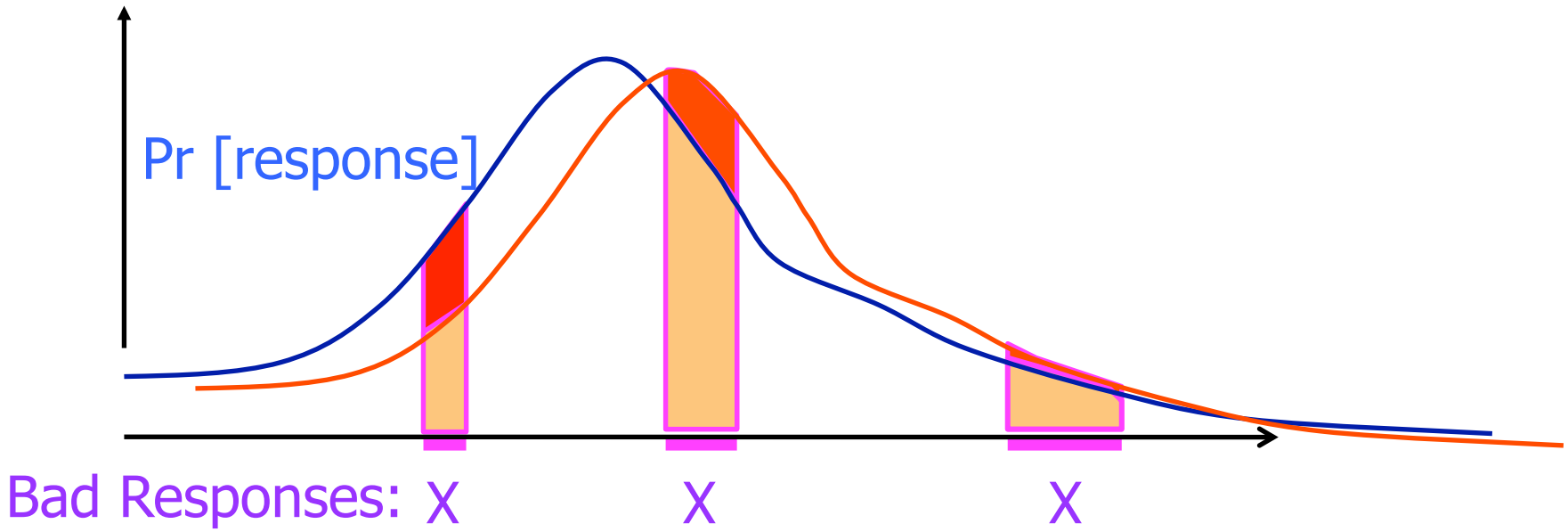
- San gives ε-differential privacy if for all values of DB and Me (Me is any entry in the data base) and all transcripts t:

$$\frac{\Pr[\mathit{San}(\mathrm{DB} - \mathrm{Me}) = t]}{\Pr[\mathit{San}(\mathrm{DB} + \mathrm{Me}) = t]} \leq e^{\varepsilon} \approx 1 \pm \varepsilon$$



Intuition

- No perceptible risk is incurred by joining DB
- Anything adversary can do to me, it could do without me (my data)



How much noise for privacy?

Sensitivity: Count Queries

Number of people having disease

- True answer in the example =3
- Real Answer : 3 + η, where η is drawn from Lap(1/ε)
  - Mean = 0
  - Variance = 2/ε2

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

How much noise for privacy?

Sensitivity:

- Consider a query q: I → R. 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 $\lambda = S/\varepsilon$

- Intuition: sensitivity measures the maximum deviation of a single database entry

How much noise for privacy?

Count Queries

Sensitivity = 1 (adding one person change count of 0 or 1)

- True answer in the example =3
- Real Answer : 3 + η, where η is drawn from Lap(1/ε)

– Mean = 0

– Variance = 2/ε2

Problem we need integer values

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

How much noise for privacy?

Sensitivity: Sum Queries

- Suppose all values x are in [a,b]
- Sensitivity = b

- Example: suppose we count total income
  - if all people have income between 10k and 100K we add a noise depending on 100K
  - If max income is 100000 K then the noise must be much larger for not revealing information on the richest person

Problems

More noise we add less interesting answers we get

1. Count queries OK; count intger values??
2. SUM queries a problem: consider a query income of men vs women; a single person can earn a lot and change the evaluation; you need a lot of randomization
3. MAX queries even worse: answer does not depend on size of data set
4. MULTIPLE QUERIES: If I ask a series of queries then I can disambiguate; this requires to add more noise. A possible approach is to give user a privacy budget; every time she makes a query reduces the privacy budget; when it is zero she stops querying

Utility of Laplace Mechanism

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

$$E(\text{true answer} - \text{noisy answer})^2$$

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

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

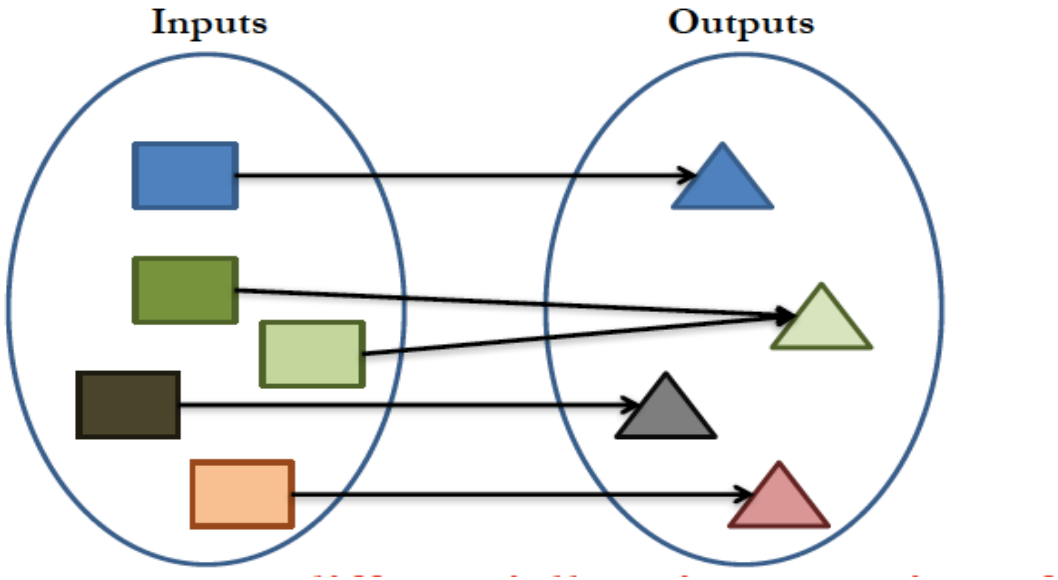
Exponential Mechanism

For functions that do not return a real number …

- "what is the most common nationality in this room":
  - Answers: Italian/Indian/…
- When perturbation leads to invalid outputs …
  - To ensure integrality/non-negativity of output

Exponential Mechanism

- Consider some function f (deterministic or probabilistic): example most common nationality



- How to construct a differentially private version of f ?

Exponential Mechanism: example

Scoring function w: Inputs x Outputs →R

- D: nationalities of a set of people
- #(D, O): # people with nationality O
- f(D): most frequent nationality in D
- w(D, O) = |#(D, O) - #(D, f(D))|
- Sensitivity of w

$$\Delta_w = \max_{O\&D,D'} |w(D,O) - w(D,O')|$$

where D, D' differ in one tuple

Exponential Mechanism

- Given an input D, and a scoring function w,
- Randomly sample an output O from Outputs with probability

$$\frac{e^{\frac{\varepsilon}{2\Delta}\cdot w(D,O)}}{\sum_{Q \epsilon Outputs} e^{\frac{\varepsilon}{2\Delta}\cdot w(D,Q)}}$$

- Note that for every output O, probability O is output > 0.

There are simpler approaches?

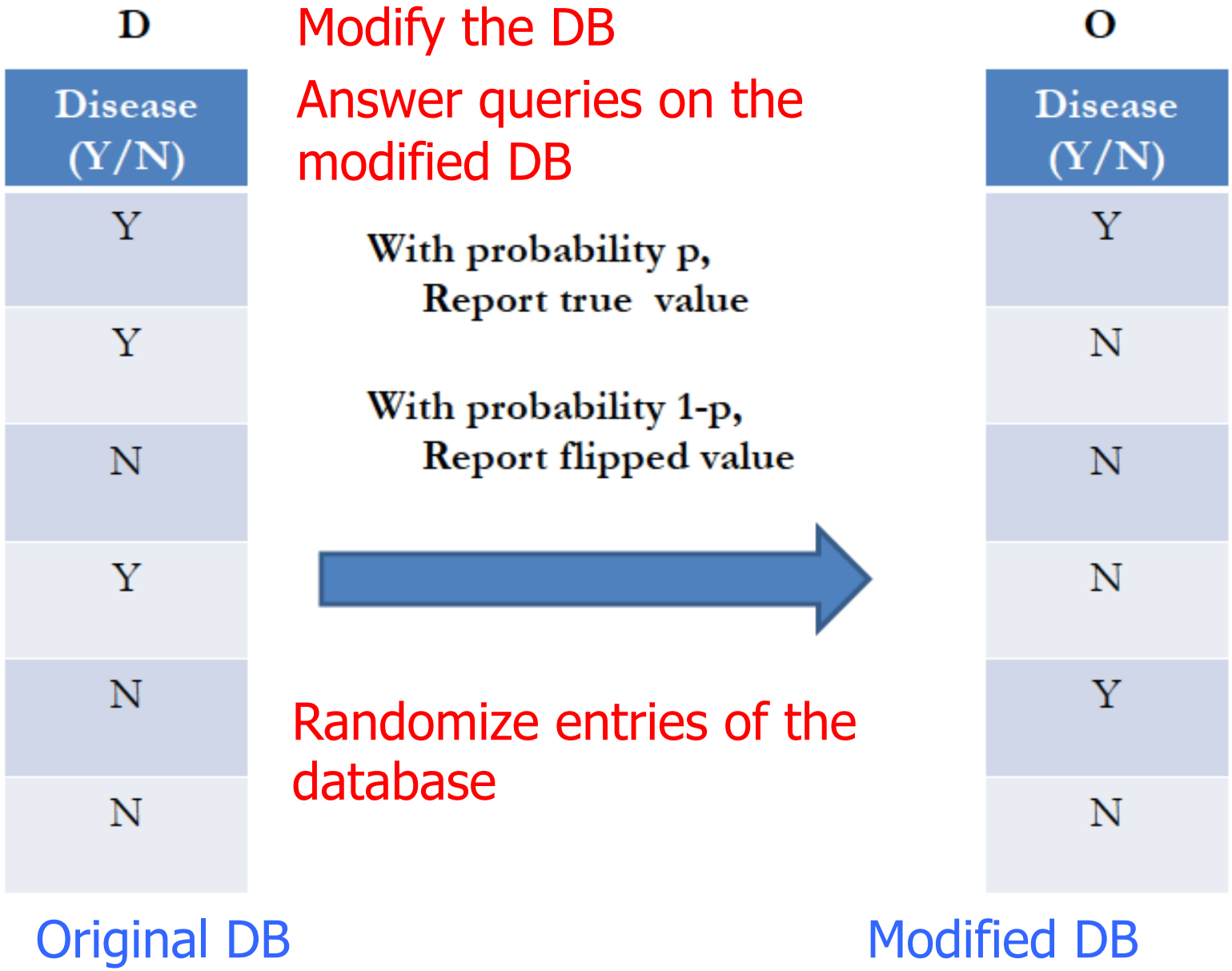
Adding a random noise based on the sensitivity is complicated. Are there simpler approaches?

- Random database: answer query on a DB with a subset of data randomly chosen

Example

- DB values D=(1,1,1,….,1,0) Query Q: how many different values? Q(D)=2
- If we random choose a subset of entries most likely we get Q(D')=2 that does not verify differential privacy

There are simpler approaches?



Randomized database

Laplace Mechanism vs Randomized Response

- Utility

Suppose a database with N records where μN records have disease = Y.

- Query: # rows with Disease=Y
  - Standard dev of Laplace mechanism answer: O(1/ε)
  - Standard dev of Randomized Response answer: O(√N)

Problem: Composition

Why Composition?

- Reasoning about privacy of a complex algorithm is hard.
- Helps software design
  - If building blocks are proven to be private, it would be easy to reason about privacy of a complex
  - algorithm built entirely using these building blocks.

Composition

There MUST be a bound on the number of queries

- In order to ensure utility, a statistical database must leak some information about each individual
- We can only hope to bound the amount of disclosure
- Hence, there is a limit on number of queries that can be answered

Dinur Nissim Result

- A vast majority of records in a database of size n can be reconstructed when n $\log(n)^2$ queries are answered by a statistical database …
- … even if each answer has been arbitrarily altered to have up to $o(\sqrt{n})$ error

Sequential Composition

- If M1, M2, ..., Mk are algorithms that access a private database D such that each Mi satisfies ε_i –differential privacy,

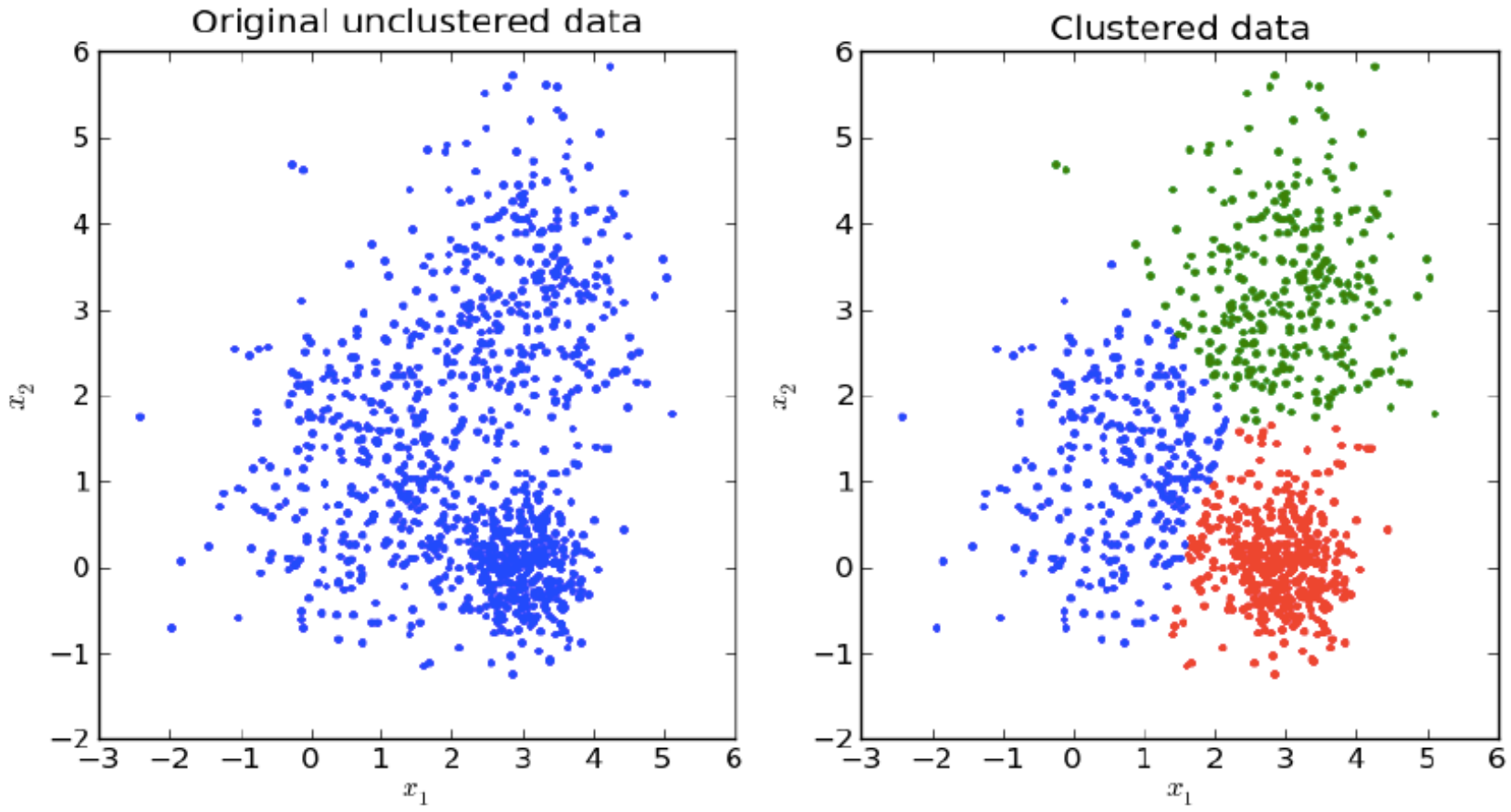
- then running all k algorithms sequentially satisfies ε-differential privacy with $\varepsilon=\varepsilon_1+...+\varepsilon_k$

Parallel Composition

- If M1, M2, …, Mk are algorithms that access disjoint databases D1, D2, …. Dk such that each Mi satisfies ε_i – differential privacy,

- then running all k algorithms sequentially satisfies ε-differential privacy with $\varepsilon = \max \{\varepsilon_1 + \ldots + \varepsilon_k\}$

Case Study: K-means Clustering



K-means Clustering

- Partition a set of points x1, x2, …, xn into k clusters S1, S2, …, Sk such that the following is minimized:

$$\sum_{i=1}^{k} \sum_{x_j \in S_i} \|x_j - \mu_i\|_2^2$$

Mean of the cluster S_i

K-means Clustering

Algorithm (Lloyd):

- Initialize a set of k centers
- Repeat
  - Assign each point to its nearest center
  - Recompute the set of centers

  Until convergence …
- Output final set of k centers

Differentially Private K-means

- Suppose we fix the number of iterations to T
- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters
  - 2. Noisily compute the size of each cluster
  - 3. Compute noisy sums of points in each cluster

Differentially Private Kmeans

- Suppose we fix the number of iterations to T
- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters
  - 2. Noisily compute the size of each cluster
  - 3. Compute noisy sums of points in each cluster

Each iteration uses ε/T privacy budget, total privacy loss is ε

Differentially Private Kmeans

- Question: Which of these steps expends privacy budget?

- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters
  - 2. Noisily compute the size of each cluster
  - 3. Compute noisy sums of points in each cluster

Differentially Private Kmeans

- Question: Which of these steps expends privacy budget?

- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters **NO**
  - 2. Noisily compute the size of each cluster **YES**
  - 3. Compute noisy sums of points in each cluster **YES**

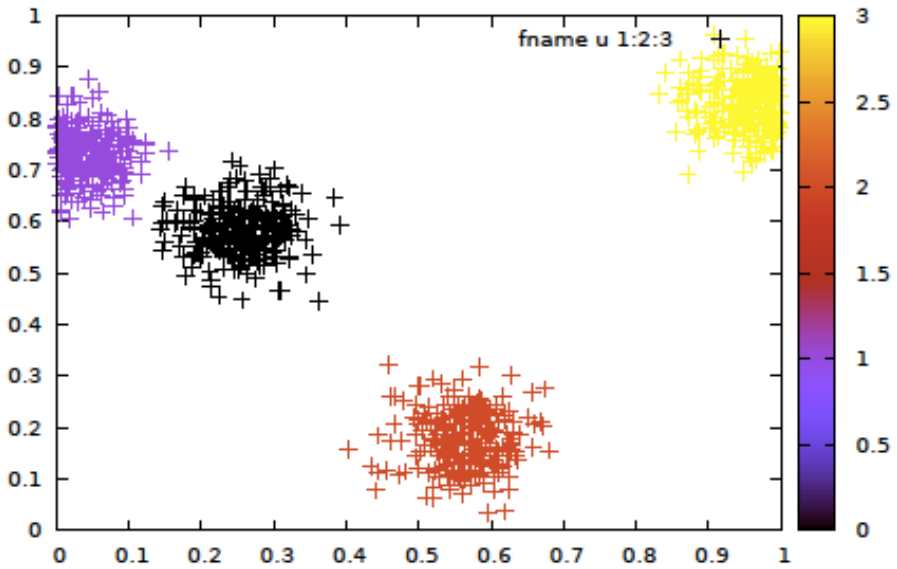
Differentially Private Kmeans

- Question: What is the sensitivity?

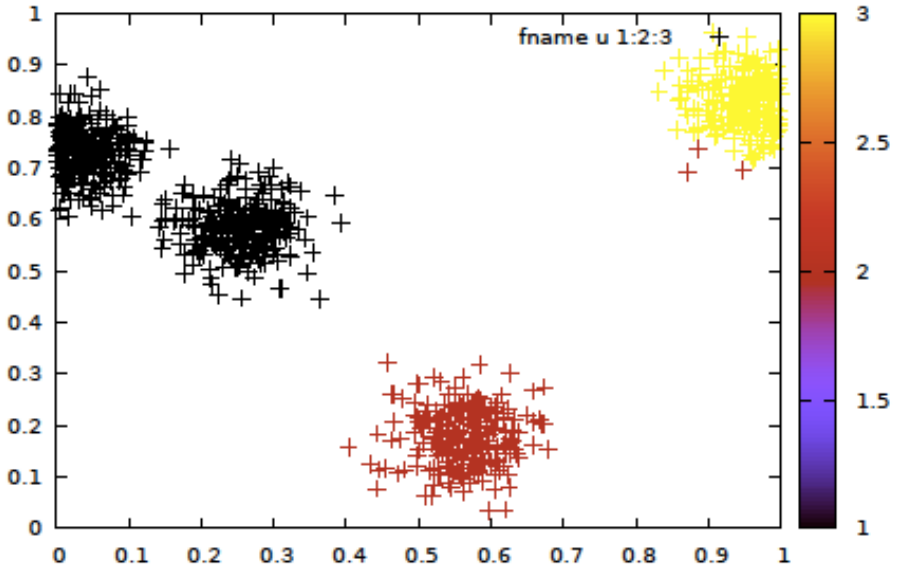
- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters **0**
  - 2. Noisily compute the size of each cluster **1**
  - 3. Compute noisy sums of points in each cluster **Domain size**

Differentially Private Kmeans

Original K-means alg.



Laplace Kmeans



- Even though we noisily compute centers, Laplace k-means can distinguish clusters that are far apart.
- Since we add noise to the sums with sensitivity proportional to |dom|, Laplace k-means can't distinguish small clusters that are close by.

Differentially Private Kmeans

- Question: What is the sensitivity?

Each iteration uses ε/T privacy budget, total privacy loss is ε

- In each iteration (given a set of centers):
  - 1. Assign the points to the new center to form clusters **Laplace(2T/ε)**
  - 2. Noisily compute the size of each cluster **1**
  - 3. Compute noisy sums of points in each cluster **Laplace(2T |dom|/ε)**

Answering queries on Tabular data

- Input: Private database D consisting of a single table (each tuple represents data of single individual)
- Workload W of counting queries with arbitrary predicates

  SELECT COUNT(*) FROM D WHERE <P>;
- Output: (noisy) answers to W
- Requirement: query answering algorithm satisfies differential privacy

Statistical agencies: data publishing

Many many statistics are published that can typically be derived from marginals

A marginal over attributes $A1,\ldots, Ak$ reports count for each combination of attribute values.

– aka cube, contingency table

– E.g. 2-way marginal on EmploymentStatus and Gender

Subject	ZCTA5 13346			
	Estimate	Margin of Error	Percent	Percent Margin of Error
EMPLOYMENT STATUS				
Population 16 years and over	5,676	+/-301	5,676	(X)
In labor force	2,715	+/-223	47.8%	+/-3.7
Civilian labor force	2,715	+/-223	47.8%	+/-3.7
Employed	2,529	+/-228	44.6%	+/-3.6
Unemployed	186	+/-93	3.3%	+/-1.7
Armed Forces	0	+/-16	0.0%	+/-0.5
Not in labor force	2,961	+/-288	52.2%	+/-3.7
Civilian labor force	2,715	+/-223	2,715	(X)
Percent Unemployed	(X)	(X)	6.9%	+/-3.4
Females 16 years and over	2,921	+/-216	2,921	(X)
In labor force	1,312	+/-140	44.9%	+/-4.5
Civilian labor force	1,312	+/-140	44.9%	+/-4.5
Employed	1,245	+/-135	42.6%	+/-4.3
Own children under 6 years	325	+/-117	325	(X)
All parents in family in labor force	241	+/-99	74.2%	+/-17.3
Own children 6 to 17 years	476	+/-102	476	(X)
All parents in family in labor force	389	+/-95	81.7%	+/-8.5
COMMUTING TO WORK				
Workers 16 years and over	2,449	+/-217	2,449	(X)
Car, truck, or van -- drove alone	1,518	+/-176	62.0%	+/-5.2
Car, truck, or van -- carpooled	116	+/-57	4.7%	+/-2.3
Public transportation (excluding taxicab)	17	+/-19	0.7%	+/-0.8
Walked	531	+/-116	21.7%	+/-4.3
Other means	132	+/-58	5.4%	+/-2.4
Worked at home	135	+/-64	5.5%	+/-2.5
Mean travel time to work (minutes)	14.2	+/-1.9	(X)	(X)
OCCUPATION				
Civilian employed population 16 years and over	2,529	+/-228	2,529	(X)

Genome Wide Association Studies

Goal: to study genetic factors associated with a given disease

- Collect subsets of the population with diseases (case) and without (control)
- Extract SNPs (a specific DNA subsequences)
  - For each SNP, usually find 2 alleles (alternative forms of the gene)
- Counting queries: Compute allele frequencies in both the case and control groups (marginal over SNPxDisease)
- Perform association test using these frequencies (e.g., Chi Square Test) to identify SNPs highly associated with disease

Problem variant: offline vs. online

- Offline (batch):
  - Entire W given as input, answers computed in batch

- Online (adaptive):
  - W is sequence q1, q2, … that arrives online
  - Adaptive: analyst's choice for qi can depend on answers $a1$,..., $a'()$
- Answering linear queries online is strictly harder than answering them offline [BSU16].

Data and query complexity

- Data complexity
  - Dimensionality: number of attributes
  - Domain size: number of distinct attribute combinations
  - Many techniques specialized for low dimensional data

- Query complexity
  - Many techniques designed to work well for a specific class of queries
  - Classes (in rough order of difficulty): histograms, range queries, marginals, counting queries, linear queries

Solution variants: query answers vs. synthetic data

Two high-level approaches to solving problem

- 1**. Direct:**
  - Output of the algorithm is list of query answers

- 2. **Synthetic data:**
  - Algorithm constructs a synthetic dataset D', which can be queried directly by analyst
  - Analyst can pose additional queries on D' (though answers may not be accurate)

Theory

Given negative result of Dinur-Nissim, is there any

- Hope? Yes!
  - The key to Dinur-Nissim is that query answers have independent noise (which can cancel out)
  - To answer more queries, query error must be correlated
- Examples of correlation
  - Use some query answers to approximate others
  - Construct a synthetic database that is approximately accurate for queries of interest

Answering Many Queries Offline

Key technical insight: For any set of count queries W, there exists a small database D' consistent with D on every query in W.

- Small: $O(\log(|W|)/\alpha 2)$
- Consistent: error for any q in W is at most α

Result follows from learning theory:

- Estimates on small random sample will generalize to population

Answering Many Queries Offline

Input: W, ε Output: D'

The Mechanism:

- – T = {all small databases D'}
- – $f\ D, D' = -max@\in B|q(D) - q(D')|$
- – Output $D' \in T$ using Exponential Mechanism applied to f
- Theorem: Is ε-private and w.h.p. error α is at most $O(\log |domain| \log W\ \varepsilon\ D))/J$ [BLR08]

Answering Many Queries Offline

Input: W, ε Output: D'

The Mechanism:

– T = {all small databases D'}

– $f\ D, D' = -max@\in B|q(D) - q(D')|$

– Output $D' \in T$ using Exponent. Mechanism applied to f

Theorem: Is ε-private and w.h.p. error α is at most O(log |domain| log $W\ \varepsilon\ D$))/J [BLR08]

Limitations: - Offline

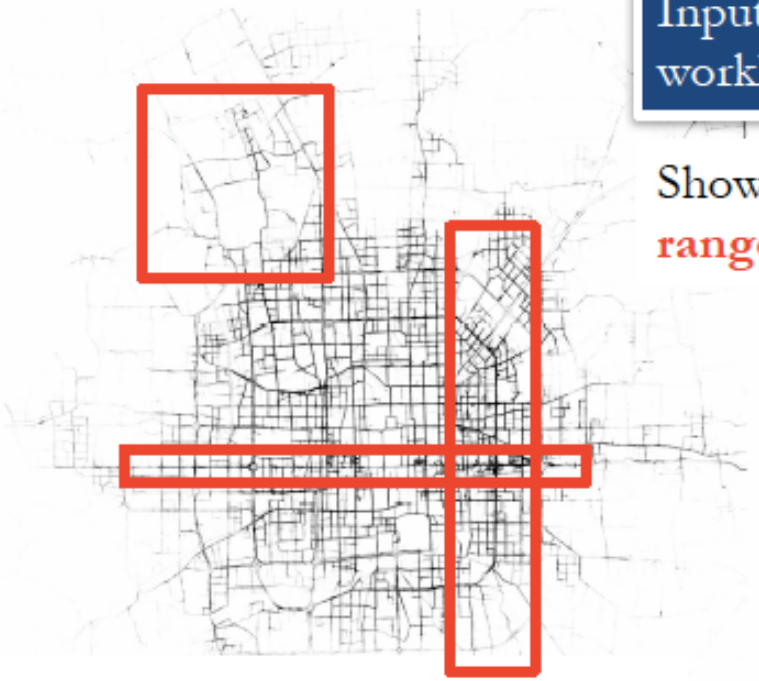
- Impractical: runtime exponential

Case study: range queries over spatial data

Input: sensitive data D

	Latitude	Longitude
2	39.98105	116.30142
3	39.9424	116.30587
4	39.93691	116.33438
5	39.94354	116.3532
6	...	...

BeijingTaxi dataset[1]: 4,268,780 records of (lat,lon) pairs of taxi pickup locations in Beijing, China in 1 month.



Scatter plot of input data

Input: range query workload W

Shown is workload of 3 range queries

Task: compute answers to workload W over private input D

Case study: range queries over spatial data

1. Discretize attribute domain into cells
2. Add noise to cell counts (Laplace mechanism)
3. Use noisy counts to either…

    - Answer queries directly (assume distribution is uniform within cell)

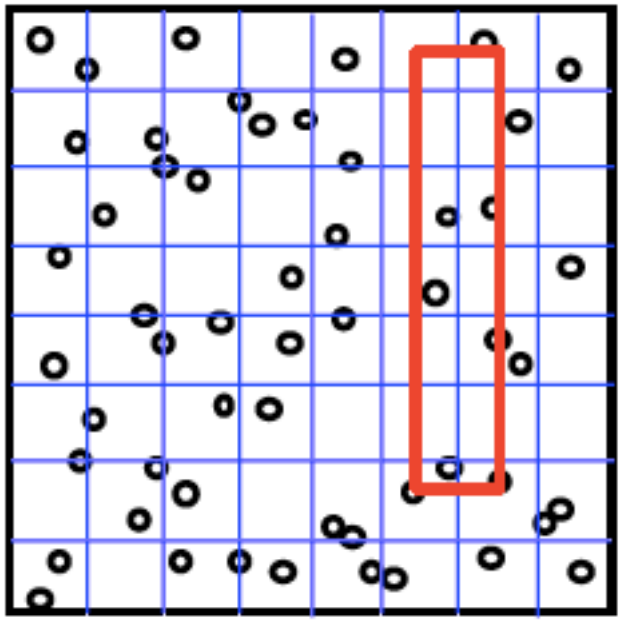
    - Generate synthetic data (derive distribution from counts and sample)

Case study: range queries over spatial data

Limitations:

- Granularity of discretization: detail lost, noise overwhelms signal

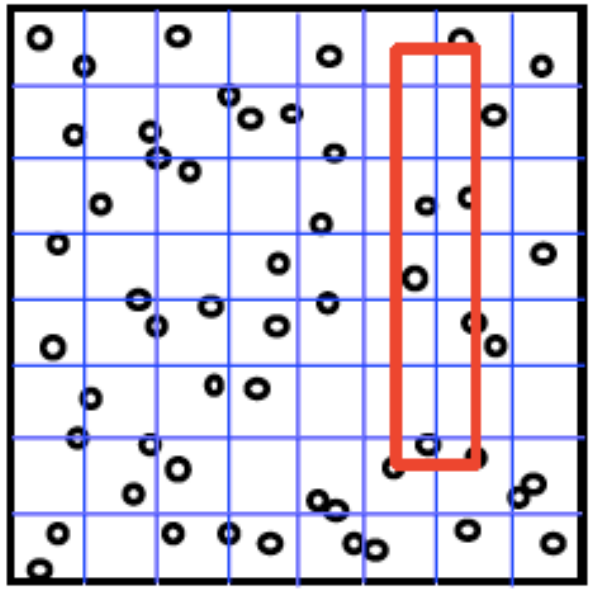
- Noise accumulates: squared error grows linearly with range



Scatter plot of input data

Case study: range queries over spatial data

1. Discretize attribute domain into cells
2. Add noise to cell counts (Laplace mechanism)
3. Use noisy counts to either
- Answer queries directly (assume distribution is uniform within cell)
- Generate synthetic data (derive distribution from counts and sample)



Scatter plot of input data

Limitations: Granularity of discretization

– Coarse: detail lost

– Fine: noise overwhelms signal

Noise accumulates: squared error grows linearly with range

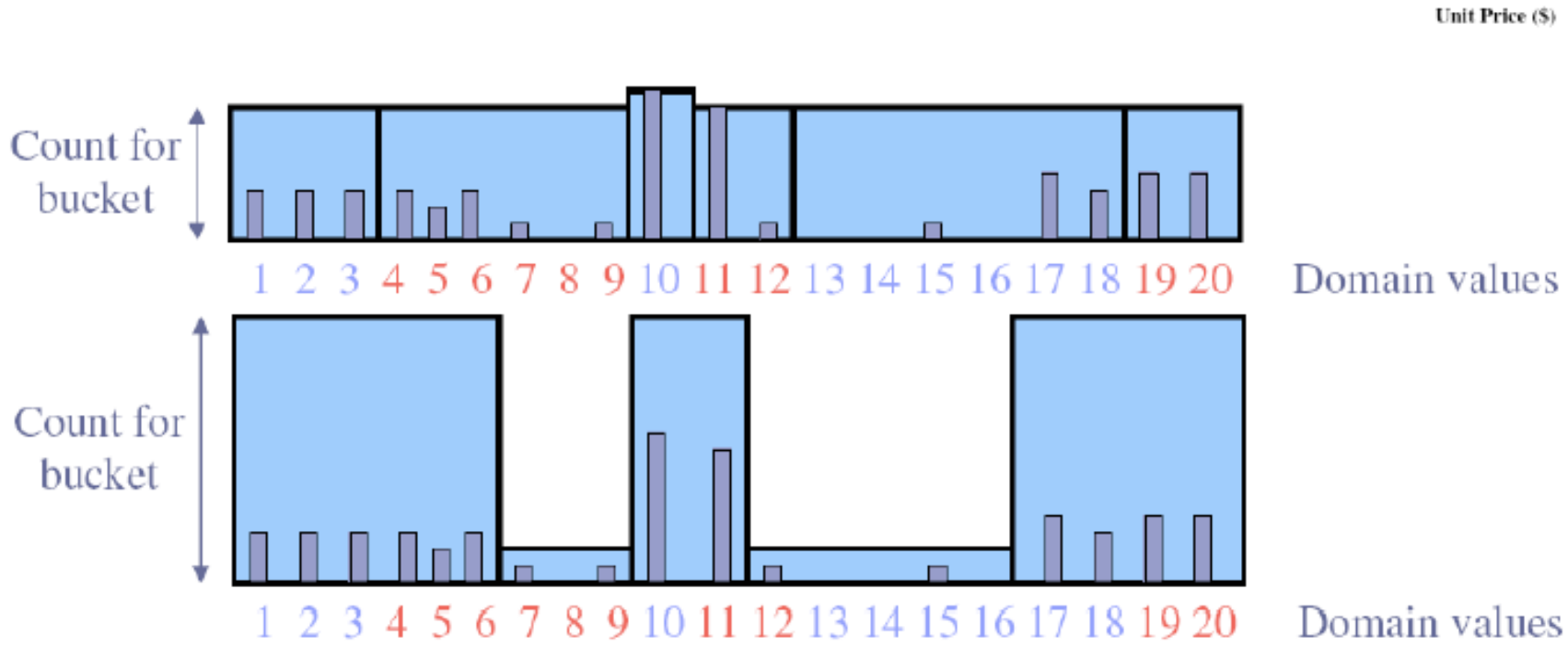
Case study: histogram

Divide data into buckets

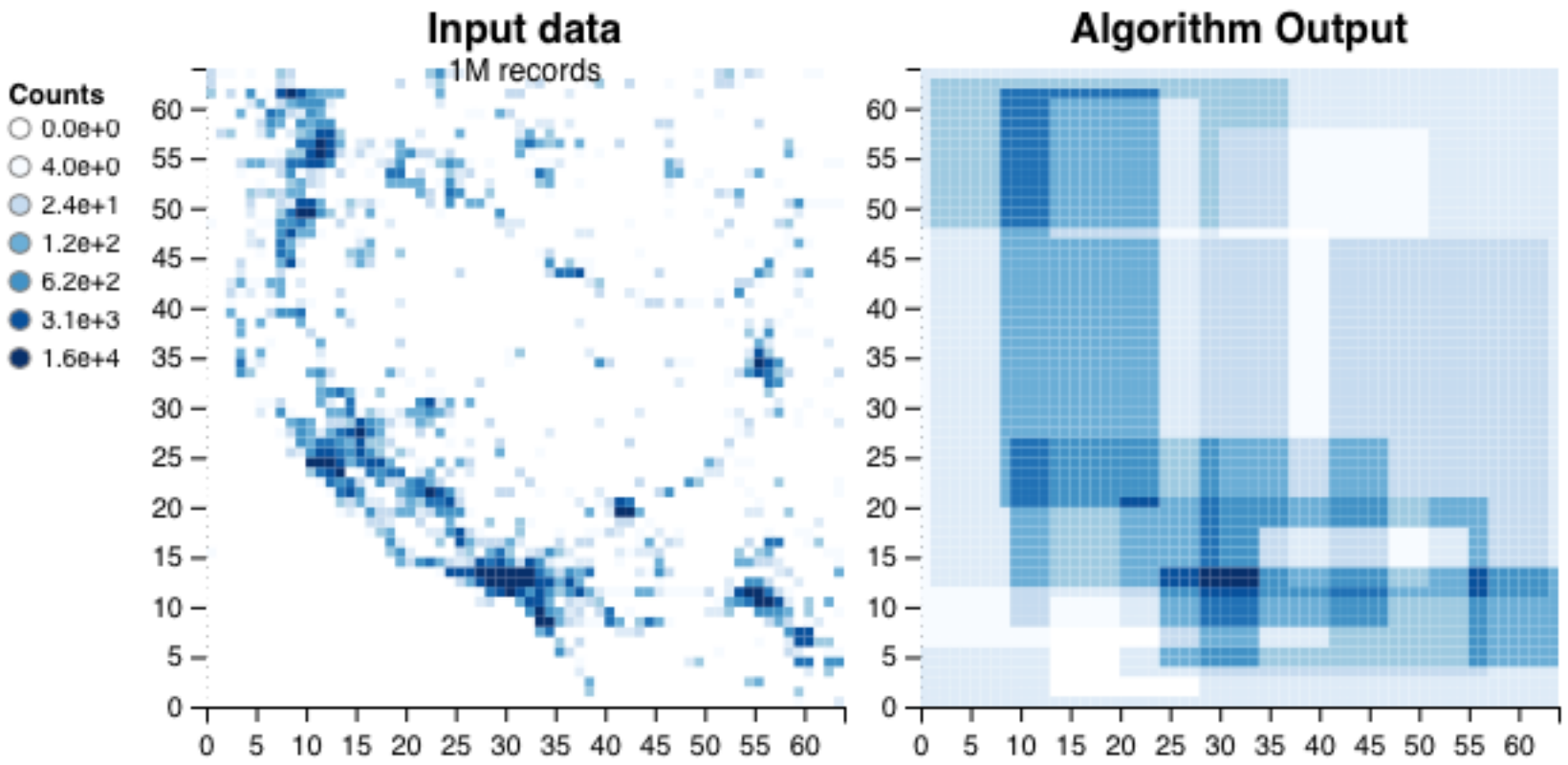
–Equi-width: equal bucket range

–Equi-depth: equal frequency

–V-optimal: with the least *frequency variance*



Case study: 2-dimensional data

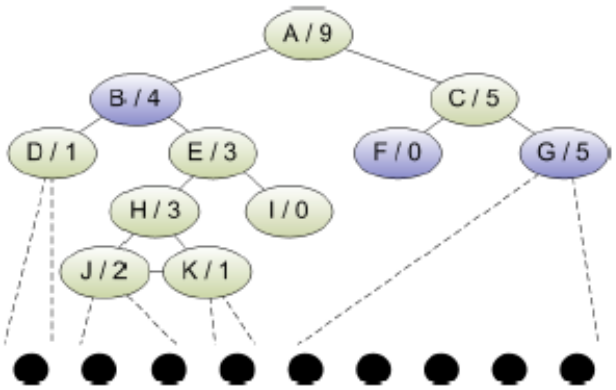


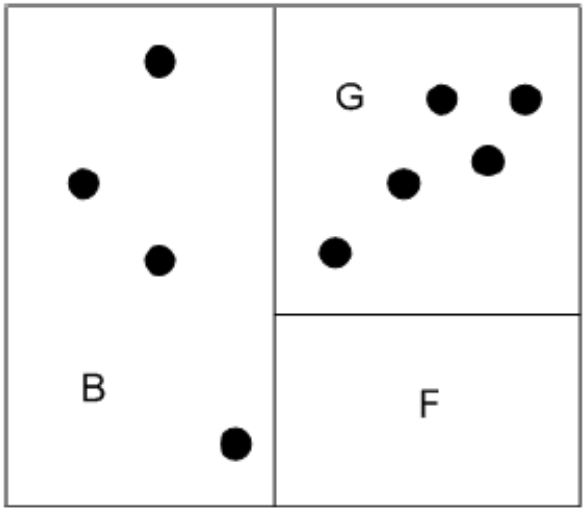
Case study: 2-dimensional data

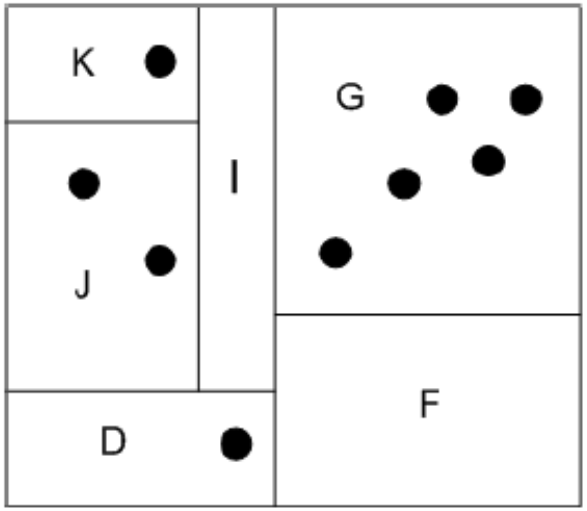
Choose dimension and splitting point to split a region (in order to minimize variance)

Repeat until:

- count on this node less than treshold
- variance or entropy of this node less than threshold







Summary

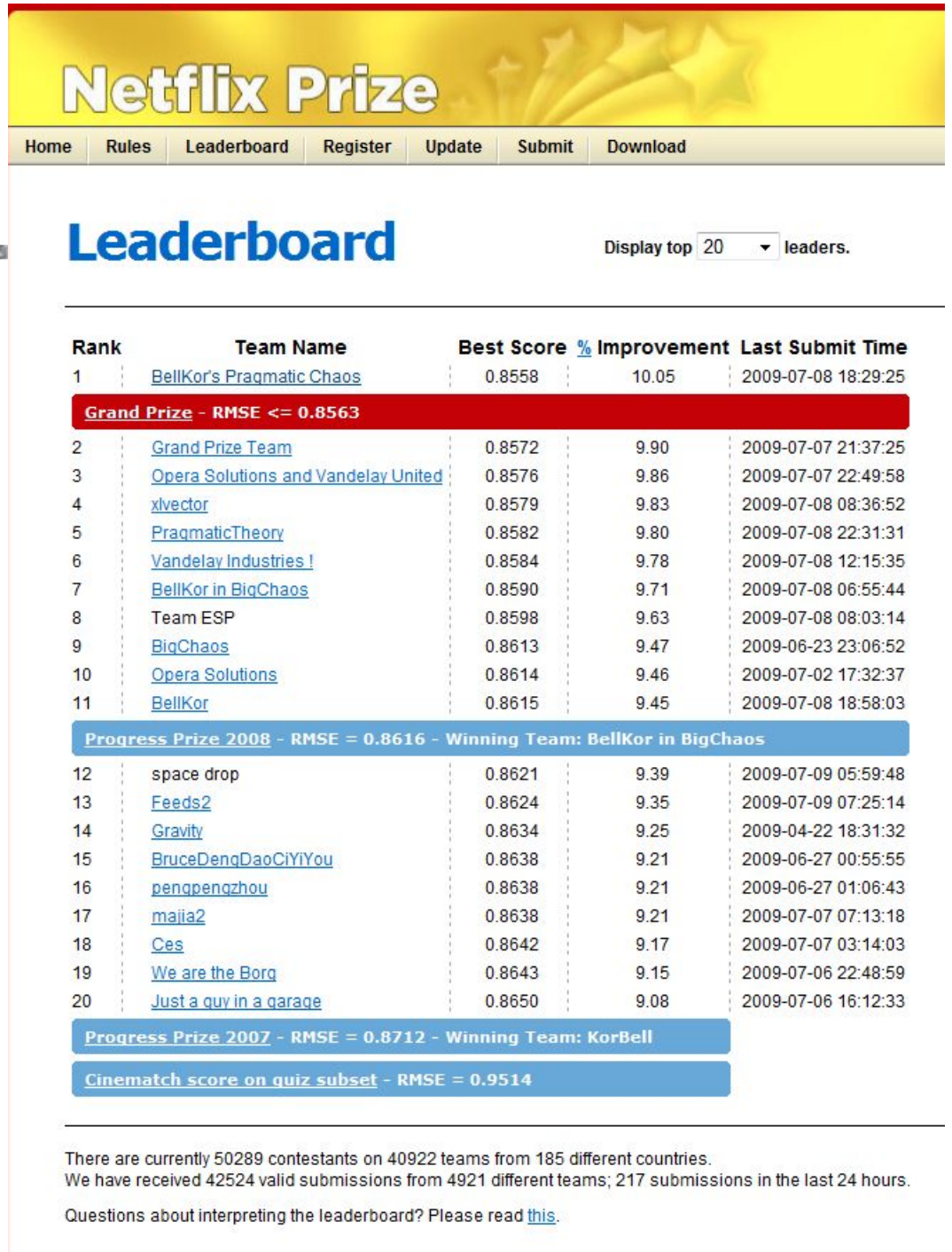
1. Reveal entire database, but randomize entries (either in input or answers to queries)
2. A strong definition of privacy shows that it is impossible to modify in such a way to be unable to learn something on some specific user
3. A weaker definition (differential privacy) shows that it is possible to get positive results
4. However use of additional information can allow to know too many things

Netflix Prize Dataset

- Netflix: online movie rental service
- In October 2006, released real movie ratings of 500,000 subscribers
  - 10% of all Netflix users as of late 2005
  - Names removed
  - Information may be perturbed
  - Numerical ratings as well as dates
  - Average user rated over 200 movies
- Task is to predict how a user will rate a movie
  - Beat Netflix's algorithm (called Cinematch) by 10%
  - You get 1 million dollars

Netflix Prize

- Dataset properties
  - 17,770 movies
  - 480K people
  - 100M ratings
  - 3M unknowns
- 40,000+ teams
- 185 countries
- $1M for 10% gain



Netflix Prize

Home | Rules | Leaderboard | Register | Update | Submit | Download

Leaderboard

Display top 20 leaders.

Rank	Team Name	Best Score	% Improvement	Last Submit Time
1	BellKor's Pragmatic Chaos	0.8558	10.05	2009-07-08 18:29:25
Grand Prize - RMSE <= 0.8563				
2	Grand Prize Team	0.8572	9.90	2009-07-07 21:37:25
3	Opera Solutions and Vandelay United	0.8576	9.86	2009-07-07 22:49:58
4	xlvector	0.8579	9.83	2009-07-08 08:36:52
5	PragmaticTheory	0.8582	9.80	2009-07-08 22:31:31
6	Vandelay Industries !	0.8584	9.78	2009-07-08 12:15:35
7	BellKor in BigChaos	0.8590	9.71	2009-07-08 06:55:44
8	Team ESP	0.8598	9.63	2009-07-08 08:03:14
9	BigChaos	0.8613	9.47	2009-06-23 23:06:52
10	Opera Solutions	0.8614	9.46	2009-07-02 17:32:37
11	BellKor	0.8615	9.45	2009-07-08 18:58:03
Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos				
12	space drop	0.8621	9.39	2009-07-09 05:59:48
13	Feeds2	0.8624	9.35	2009-07-09 07:25:14
14	Gravity	0.8634	9.25	2009-04-22 18:31:32
15	BruceDengDaoCiYiYou	0.8638	9.21	2009-06-27 00:55:55
16	pengpengzhou	0.8638	9.21	2009-06-27 01:06:43
17	majia2	0.8638	9.21	2009-07-07 07:13:18
18	Ces	0.8642	9.17	2009-07-07 03:14:03
19	We are the Borg	0.8643	9.15	2009-07-06 22:48:59
20	Just a guy in a garage	0.8650	9.08	2009-07-06 16:12:33
Progress Prize 2007 - RMSE = 0.8712 - Winning Team: KorBell				
Cinematch score on quiz subset - RMSE = 0.9514				

There are currently 50289 contestants on 40922 teams from 185 different countries.
We have received 42524 valid submissions from 4921 different teams; 217 submissions in the last 24 hours.

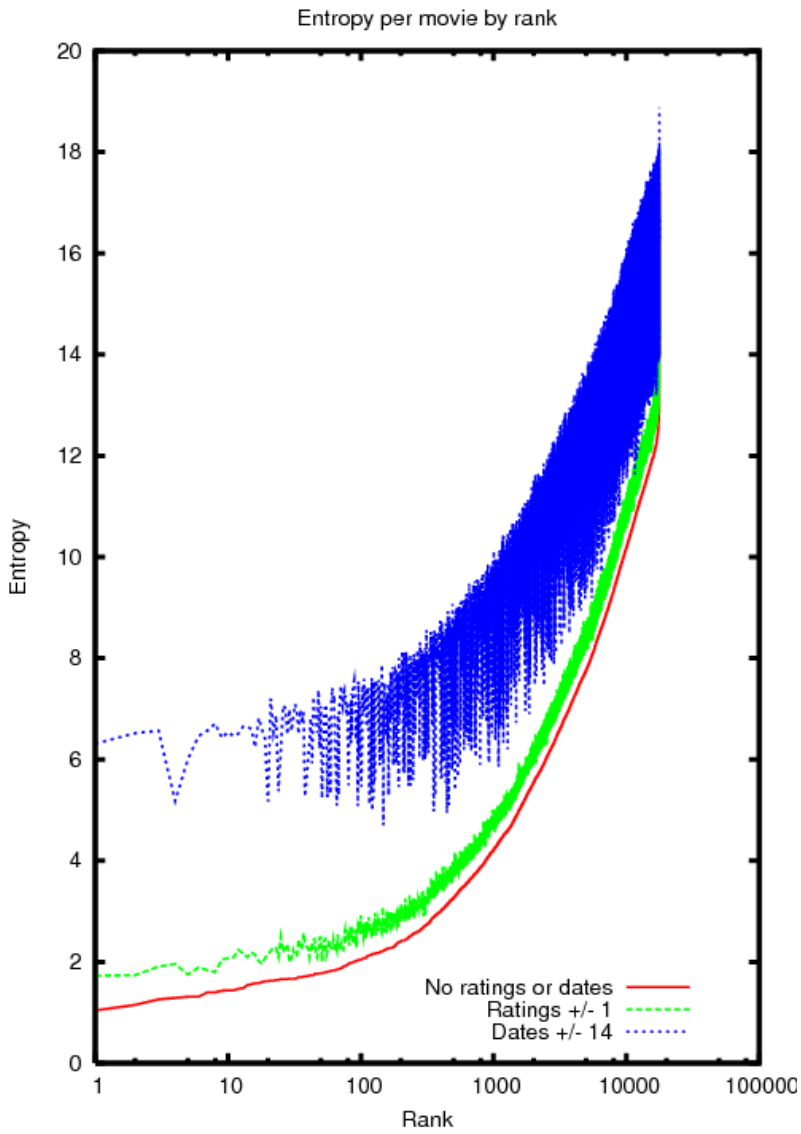
Questions about interpreting the leaderboard? Please read this.

How do you rate a movie?

- Report global average [-15%]
- Report movie average (Movie effects) [-10%]
- User effects
  - Find each user's average
  - Subtract average from each rating
- Movie + User effects is 5% worse than Cinematch
- More sophisticated techniques use covariance matrix

Netflix Dataset: Attributes

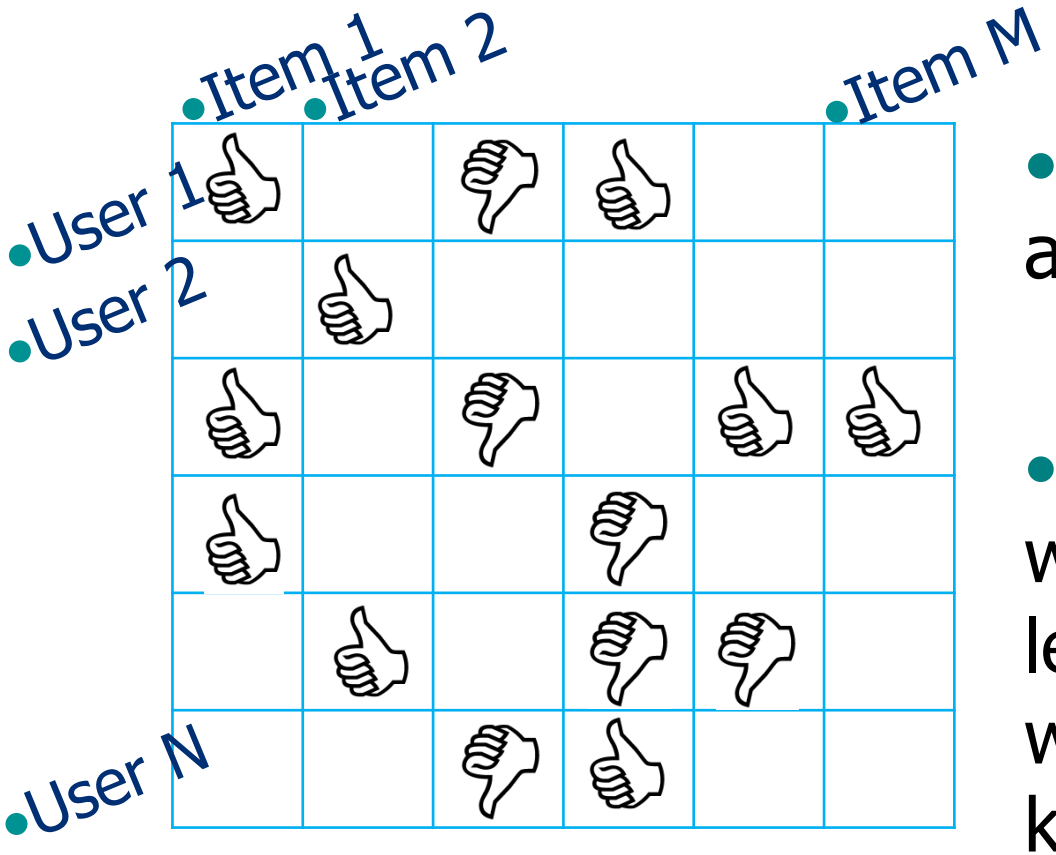
- Most popular movie rated by almost half the users!
- Least popular: 4 users
- Most users rank movies outside top 100/500/1000



Confounding prediction

- Some movies are quirky
  - I Heart Huckabees
  - Napoleon Dynamite
  - Lost In Translation
  - These movies have intermediate average, but high standard deviation
- Users polarize on these movies
- Lovers and Haters hard to determine
  - The Dark Knight might predict X-men II
  - Hard to find predictors for some movies
- Maybe use social networks to weight ratings

Why is Netflix database private?



- No explicit identifiers

- Provides some anonymity

- Privacy question: what can the adversary learn by combining with background knowledge?

Netflix's Take on Privacy

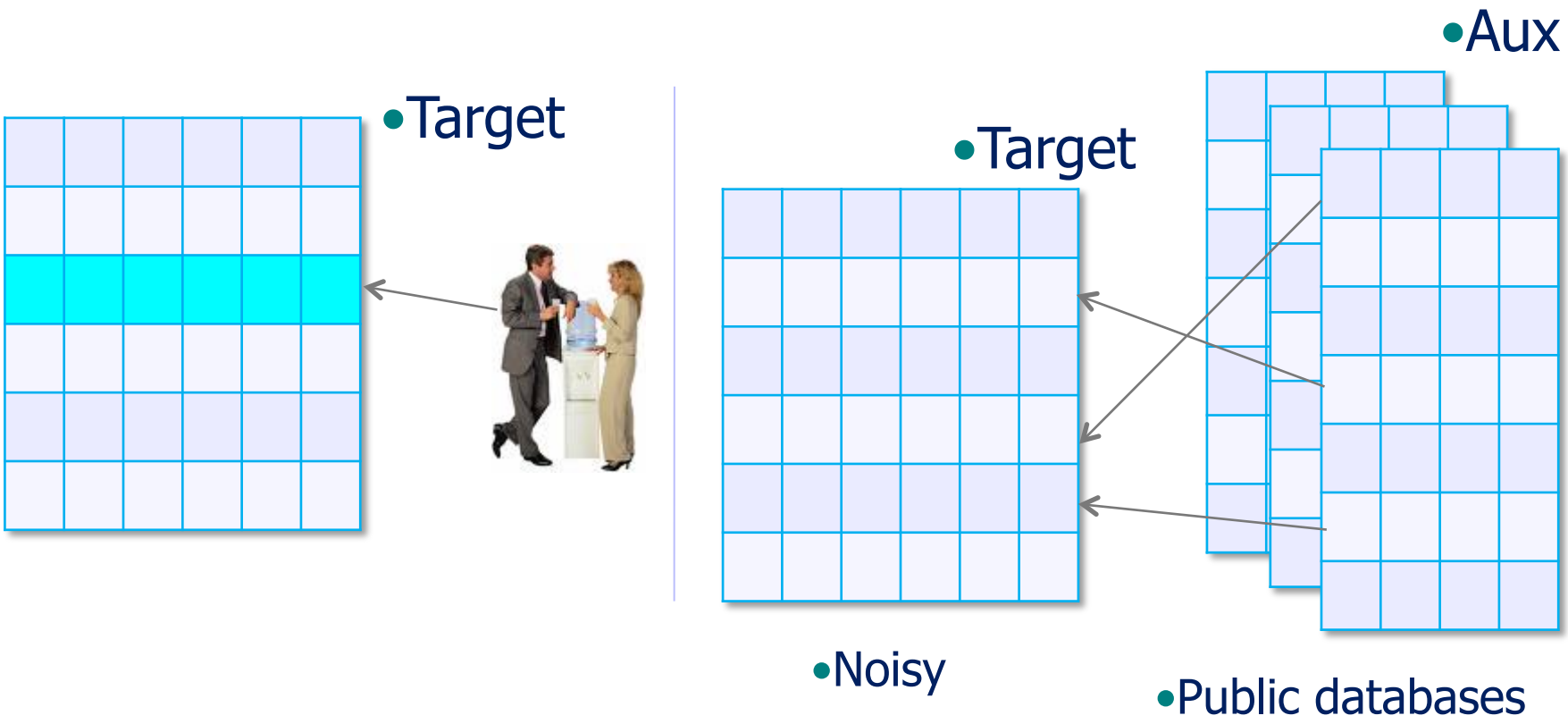
Even if, for example, you knew all your own ratings and their dates you probably couldn't identify them reliably in the data because only a small sample was included (less than one-tenth of our complete dataset) and that data was subject to perturbation. Of course, since you know all your own ratings that really isn't a privacy problem is it?

- -- Netflix Prize FAQ

COURTESY: NETFLIX

Background Knowledge (Aux. Info.)

Information available to adversary outside of normal data release process

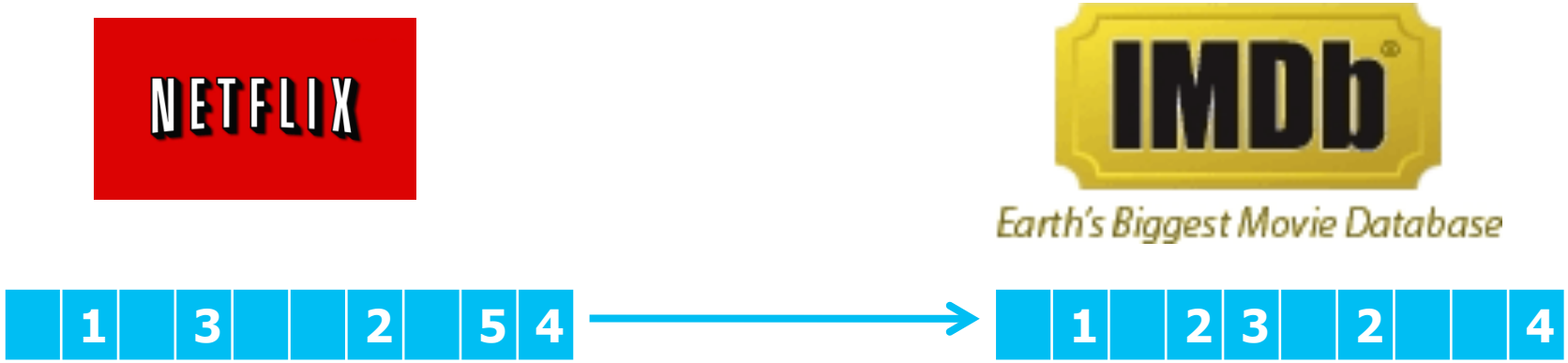


De-anonymization Objective

- Fix some target record r in the original dataset
- Goal: learn as much about r as possible
- Subtler than "find r in the released database"

- Background knowledge is noisy
- Released records may be perturbed
- Only a sample of records has been released
- False matches

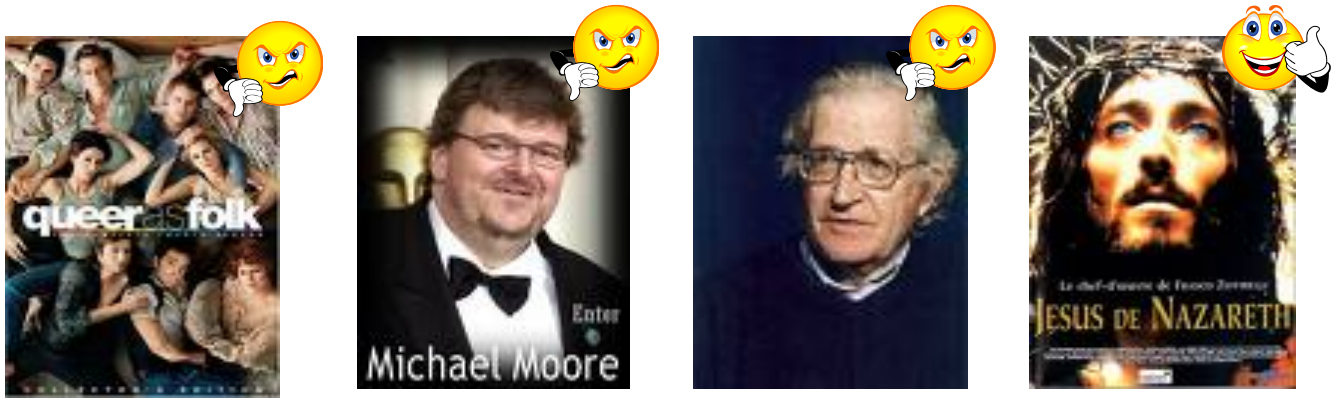
Narayanan & Shmatikov 2008



martinwilliamrandall	
Email	martinwilliamrandall@yahoo.co.uk
Biography	i went to st peters & st pauls primary from 1982 to 1985

Using IMDb as Aux

- Extremely noisy, some data missing
- Most IMDb users are not in the Netflix dataset
- Here is what they learn from the Netflix record of one IMDb user (not in his IMDb profile)

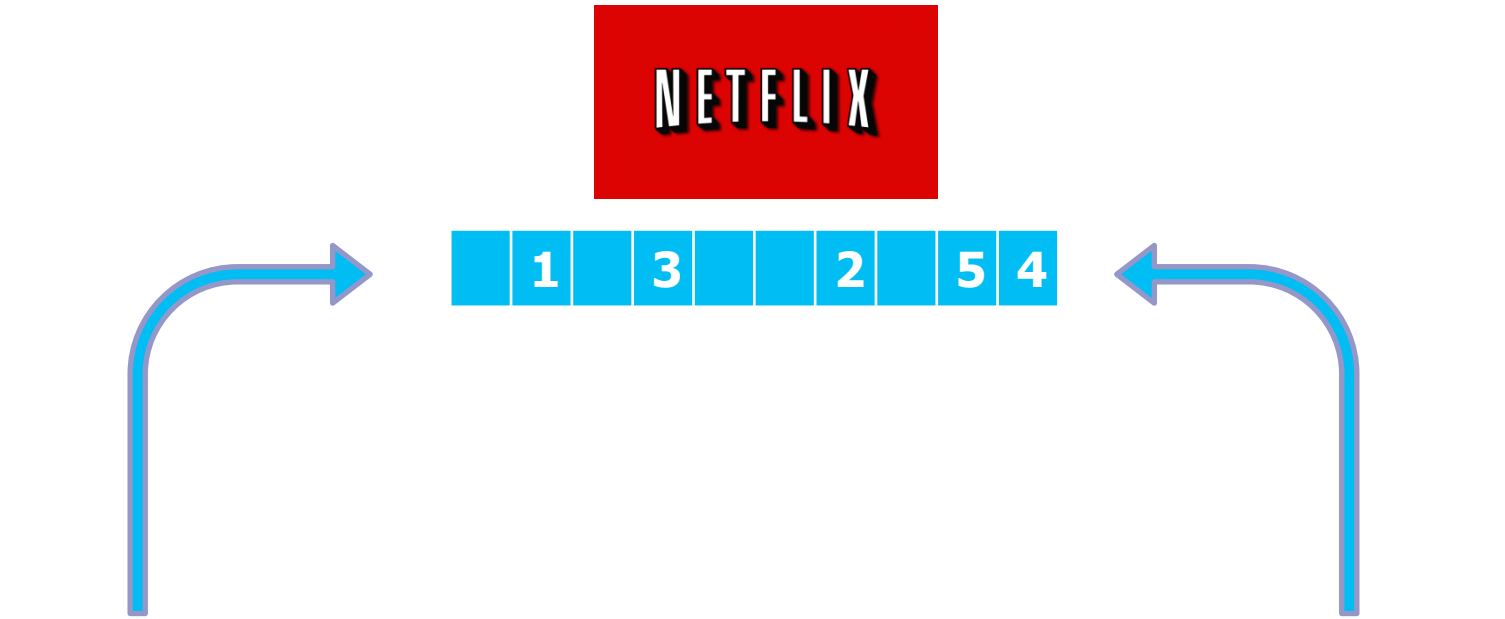


De-anonymizing the Netflix Dataset

- Average subscriber has 214 dated ratings
- Two is enough to reduce to 8 candidate records
- Four is enough to identify uniquely (on average)
- Works even better with relatively rare ratings
  - "The Astro-Zombies" rather than "Star Wars"

- Fat Tail effect helps here:
- most people watch obscure movies (really!)

More linking attacks



- Profile 1 in IMDb
- Profile 2 in AIDS survivors online

Anonymity vs. Privacy

At a literal level, anonymous means "without a name". What does it mean:

1. interacting without using your real name?

or

2. Interacting without using any name at all?

Bitcoin addresses are hashes of public keys

Hence 2 applies; we say in Bitcoin you use pseudo-identity so we have pseudonymty

Anonymity vs. Privacy

- Anonymity is insufficient for privacy
- Anonymity is necessary for privacy
- Anonymity is unachievable in practice

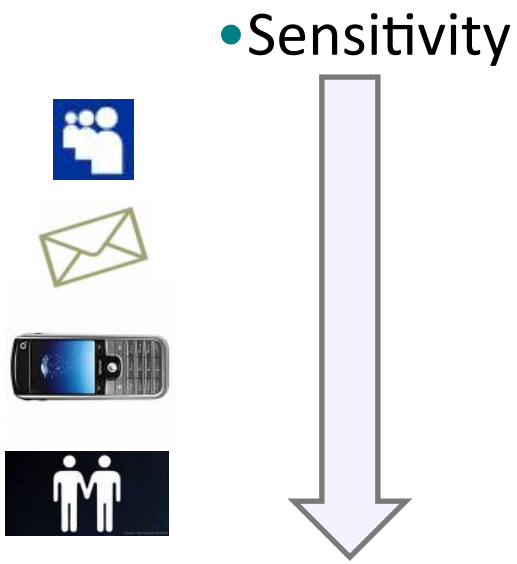
Re-identification attack → anonymity breach → privacy breach

- Just ask Justice Scalia (US Supreme Court)

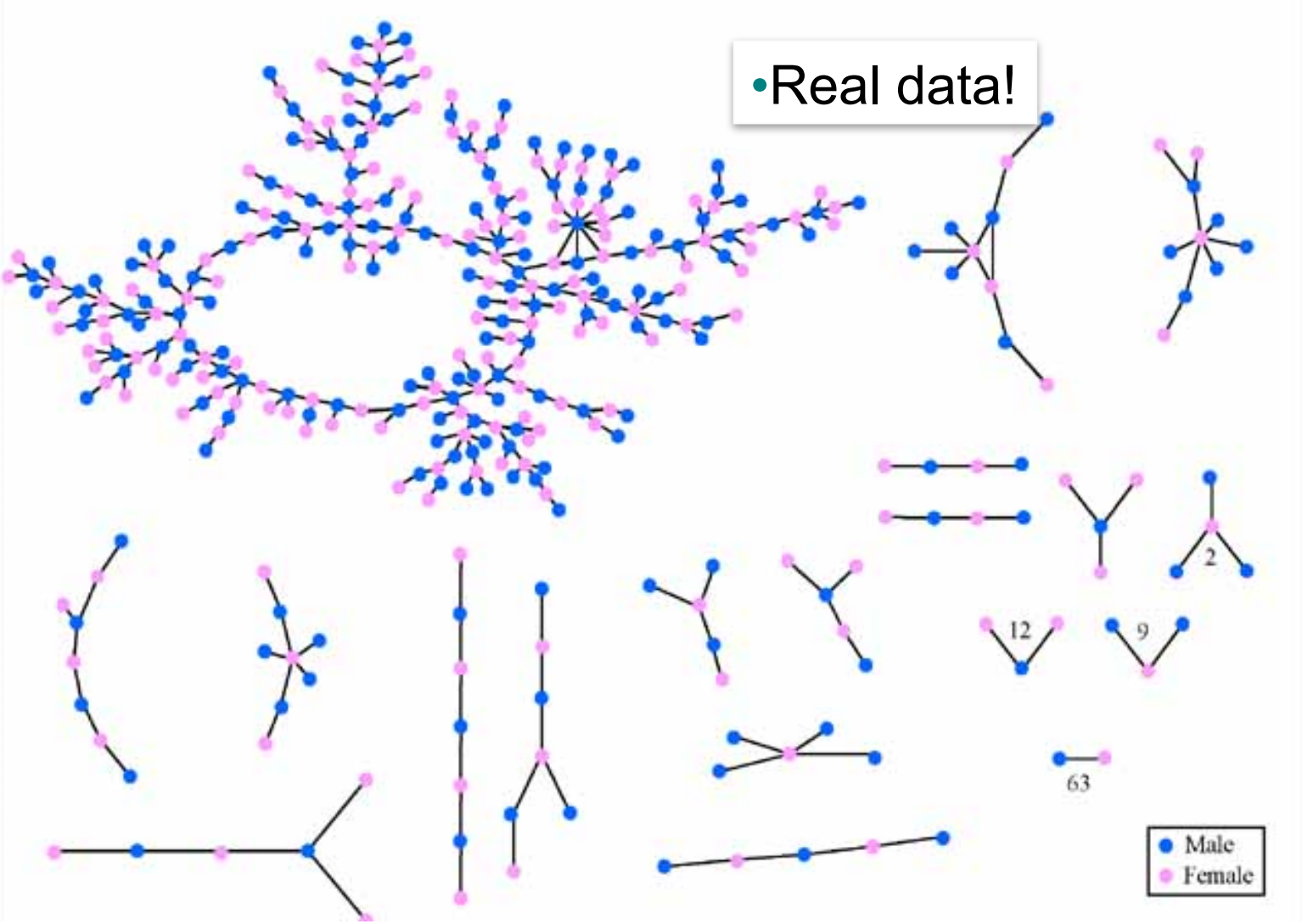
"It is silly to think that every single datum about my life is private"

Social Networks

- Online social network services
- Email, instant messenger
- Phone call graphs
- Plain old real-life relationships



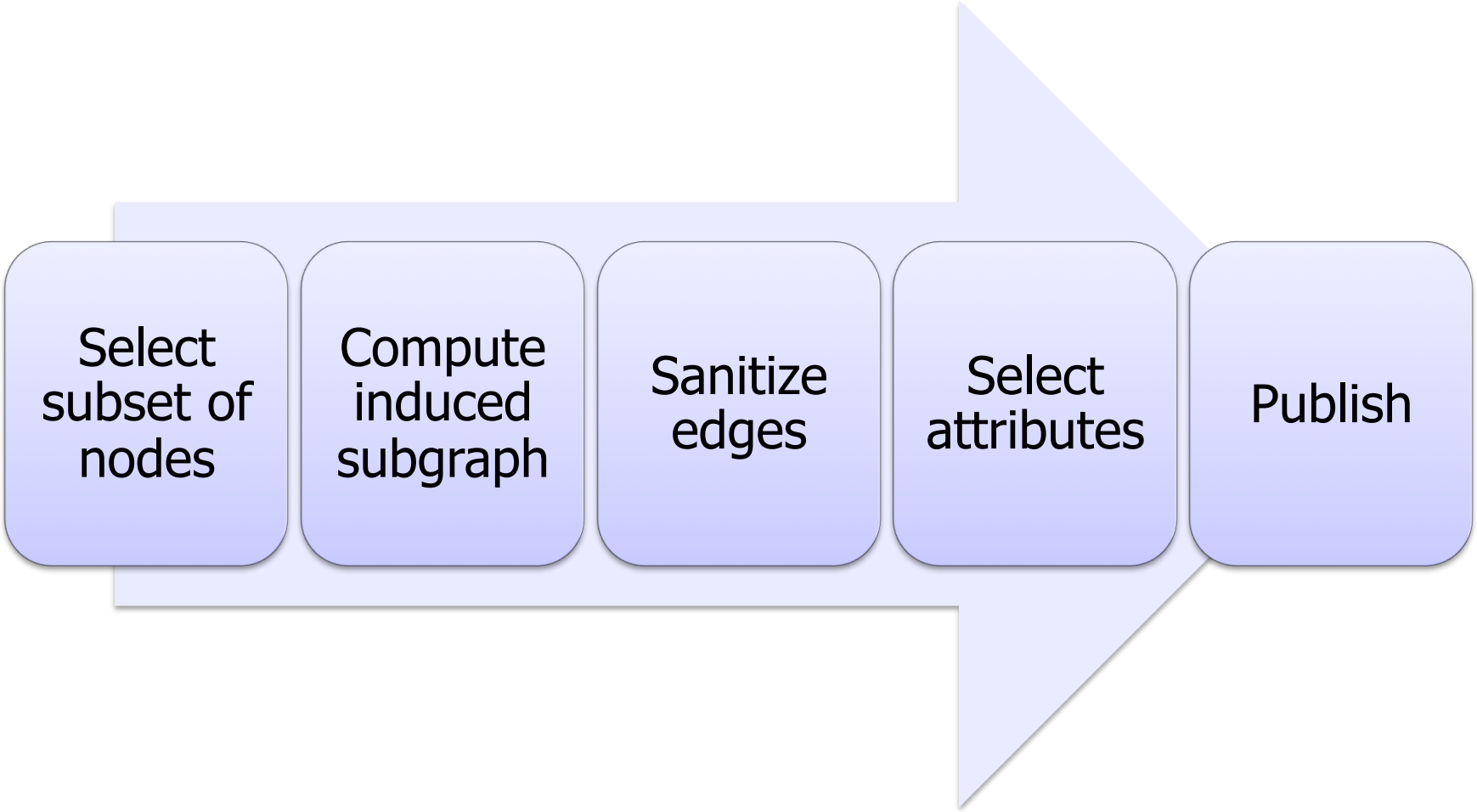
"Jefferson High": Romantic and Sexual Network



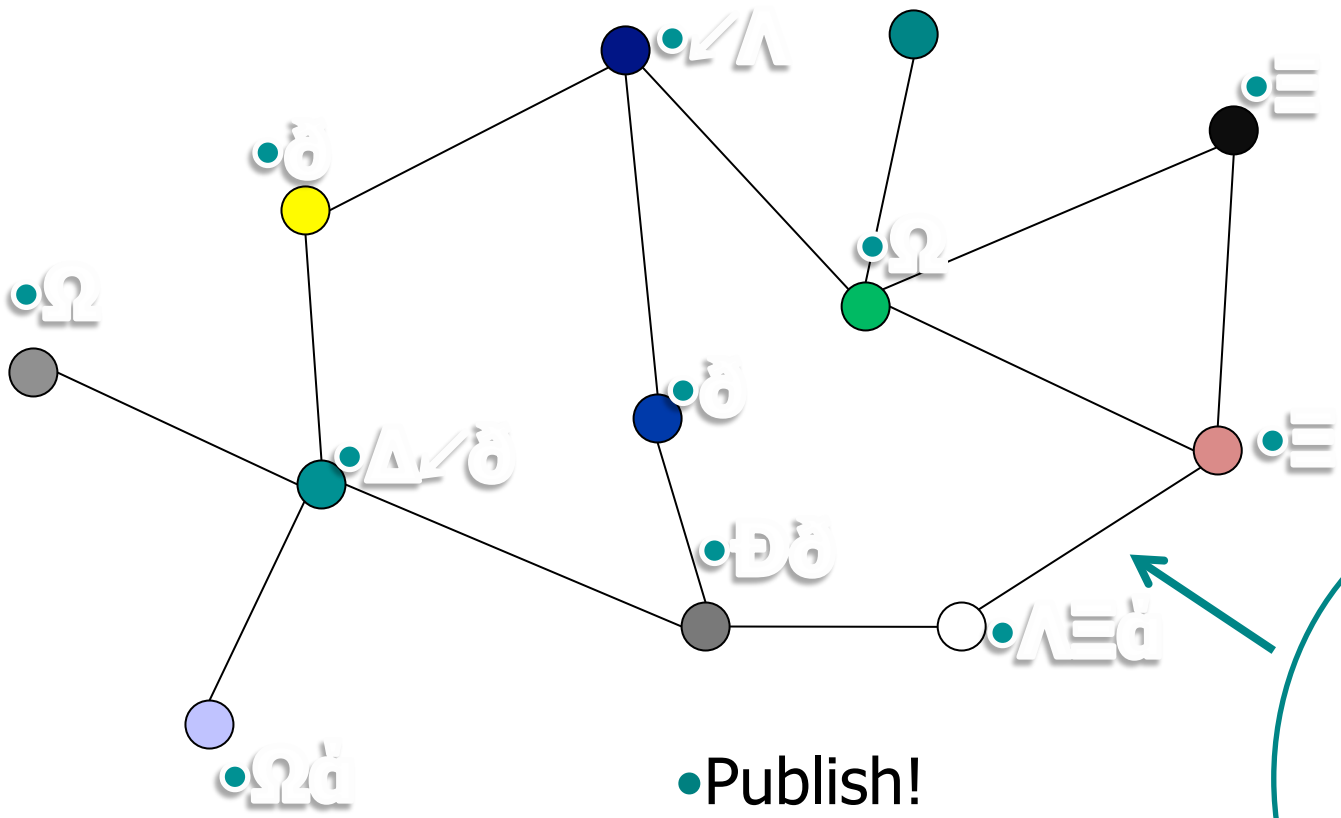
"Jefferson High" romantic dataset

- James Moody at Ohio State
- 1,000 students over 18 months in 1995
  - 537 were sexually active (those were graphed)
- Network is like rural phone lines
  - Main trunk line to individual houses
  - Many adult sexual networks are hub & spoke
  - Easier to control disease without hubs
- One component links 288 students (52%)
  - But 37 degrees of separation maximum
- 63 simple pairs, few cycles
  - No "sloppy seconds"

Social Networks: Data Release



Attack Model



- Publish!

- Large-scale
- Background
- Knowledge

Motivating Scenario: Overlapping Networks

- Social networks A and B have overlapping memberships
- Owner of A releases anonymized, sanitized graph A'
  - say, to enable targeted advertising
- Can owner of B learn sensitive information from released graph A'?

Re-identification: Two-stage Paradigm

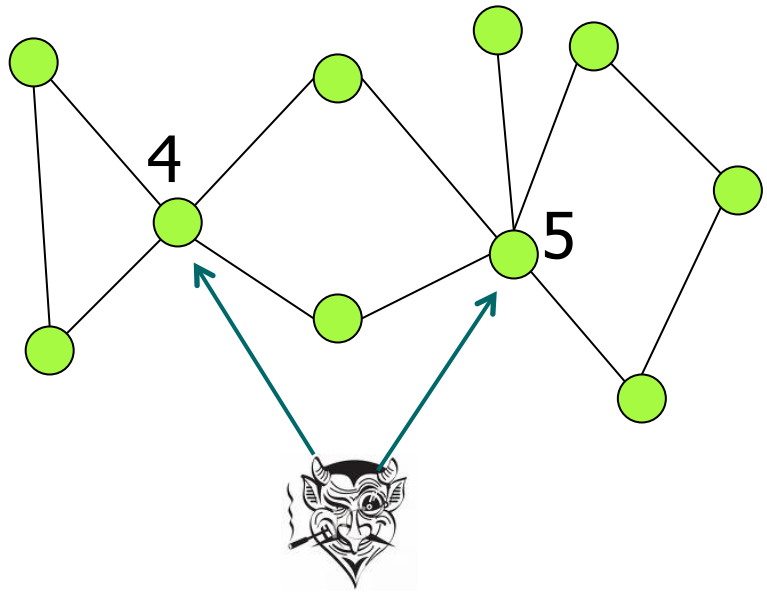
Re-identifying target graph =
Mapping between Aux and target nodes

- Seed identification:
  - Detailed knowledge about small number of nodes
  - Link neighborhood constant
  - In my top 5 call and email list…..my wife
- Propagation: similar to infection model
  - Successively build mappings
  - Use other auxiliary information
    - I'm on facebook and flickr from 8pm-10pm
- Intuition: no two random graphs are the same
  - Assuming enough nodes, of course

Seed Identification: Background Knowledge

How:

- Creating sybil nodes
- Bribing
- Phishing
- Hacked machines
- Stolen cellphones



What: List of neighbors

- Degree
- Number of common neighbors of two nodes

Degrees: (4,5)

Common nbrs: (2)

Data Release Model

- Select subset of nodes $V_{san} \subseteq V$ and subsets

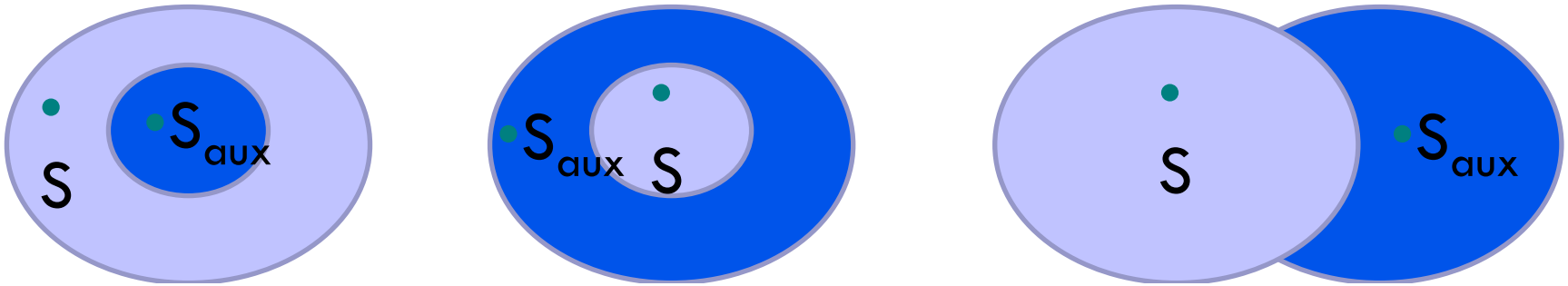
$$X_{san} \subseteq X, Y_{san} \subseteq Y$$

of node and edge attributes to be released

- Compute the induced subgraph on V_{san}
- Remove some edges and add fake edges (perturbation)
- *Summary*: a sanitized subset of nodes and edges with the corresponding attributes.

The Attacker Model

- Many types of attackers (government, marketing, spammers etc)
- Usually have access to different network S_{aux} whose membership partially overlaps with S
  - Might be extracted from S (automatic crawling, malicious third-party application)
  - Aggregation projects
  - Collude with an operator of a different network



Auxiliary Information Def.

- S_{aux}: a graph $G_{aux}=\{V_{aux}, E_{aux}\}$ and a set of probability distributions Aux_X and Aux_Y , one for each attribute of every node in V_{aux} and each attribute of every edge in E_{aux}.

  For example, P[X = “friendship”] = 0.8

  P[X = “contact”] = 0.2

- Seeds: attacker possesses *detailed* information about a *negligible* number of members of S.
  - How? (Active adversary, member of S,..)

Success of an Attack

- Reidentification : Fraction of nodes re-identified.
- Privacy breach: knowledge about edges

- Quality of success depends proportional on the importance in the network

  E.g degree centrality, where each node is weighted in proportion to its degree.

Re-identification Algorithm Def.

- **Reidentification:** identify a node in V_{san}

  Reidentification is not deterministic, but probabilistic

- **Re-identification algorithm** - a probabilistic mapping

$\mu: V_{san} \times V_{aux} \rightarrow [0, 1]$ where $\mu(v_{aux}, v_{san})$ is the probability that v_{aux} is mapped to v_{san}.

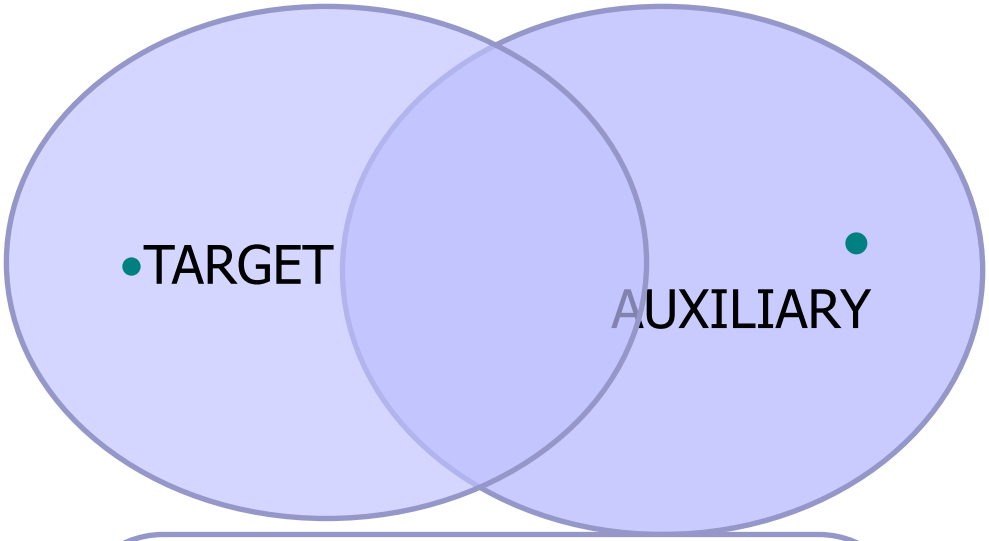
Re-identification Algorithm Def.

- **Privacy breach:** to know something about edges in the network
- **Privacy breach** – for node v_{aux} in V_{aux} let $\mu_{REAL}(v_{aux})=v_{san}$. We say that the privacy of v_{san} is breached w.r.t adversary *Adv* and privacy parameter δ, if for some attribute *X*, that is private:

$$Adv[X,v_{aux},X[v_{aux}]] - Aux[X, v_{aux}, X[v_{aux}]] > \delta$$

DE-ANONYMIZATION ALGORITHM

Seed Identification



Input:

1. A clique of k nodes known to be common to both graphs.
2. Degree of each of these nodes (AUXILIARY)
3. Number of common neighbors for each pair of nodes (AUXILIARY)

Algorithm:

Search for a unique k-clique (on TARGET) that has:

1. matching degrees (with some error factor)
2. common neighbor counts

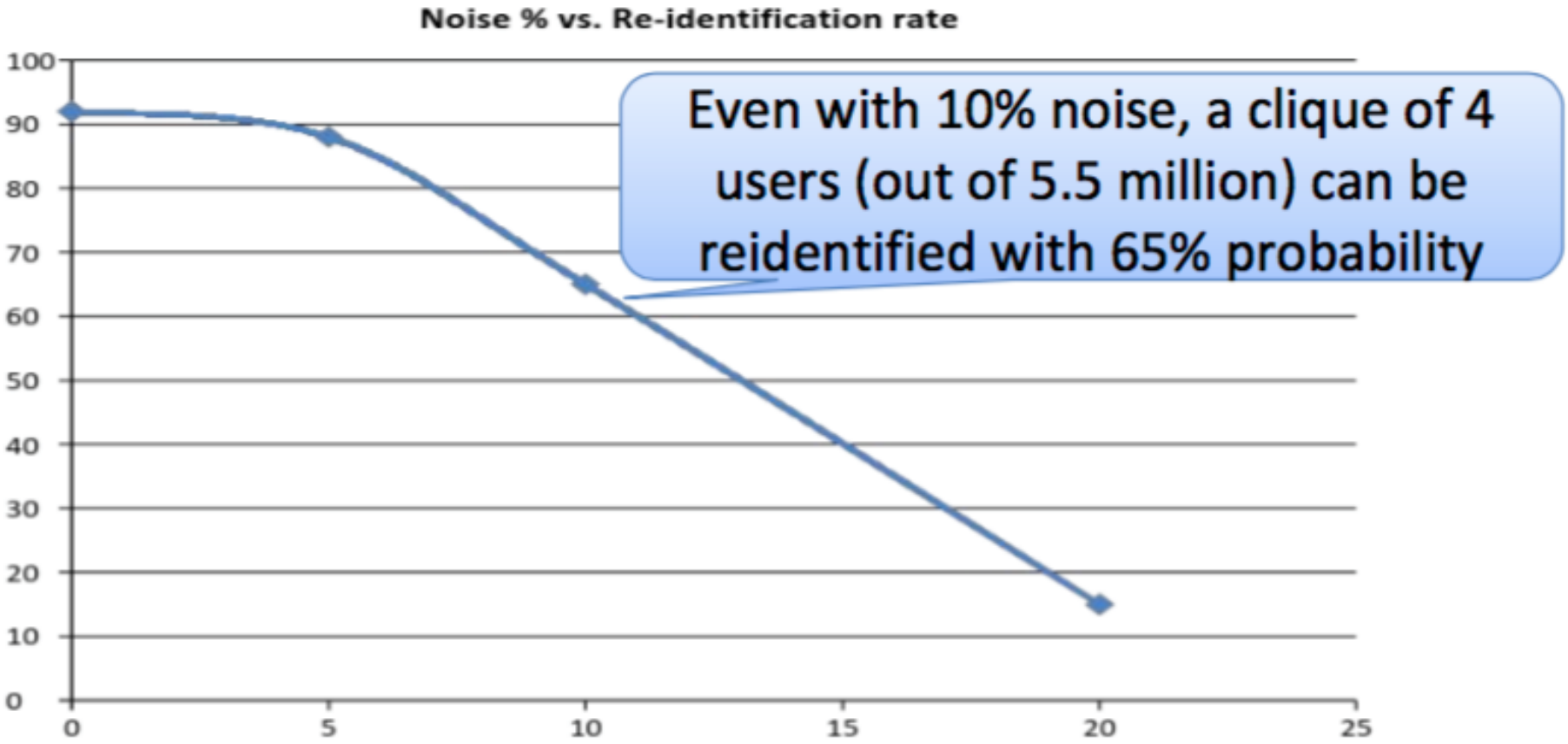
- Outputs μ_s, a partial mapping.

Complexity:Exponential in k.

BUT:

1. If the degree is bounded by *d*, then the complexity is $O(nd^{(k-1)})$.
2. Heavily input dependant. (High running time => Large number of matches)

Seed Identification: Background Knowledge



Reidentifying a clique of 4 usrs in a social network (5 Millions of users)

Propagation Algorithm

- **Input**: two graphs and a partial seed mapping
- **Output**: deterministic 1-1 mapping

Each iteration starts with the accumulated mapping.

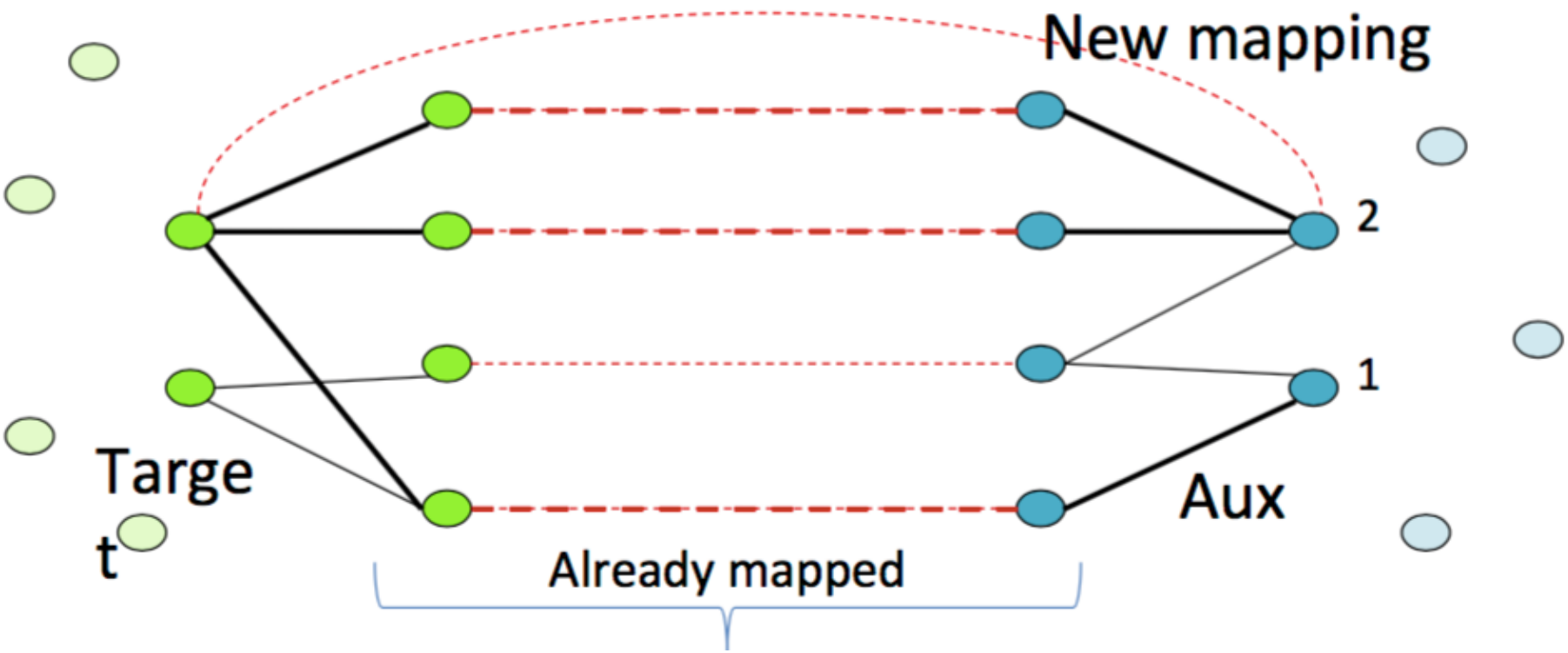
Picks an unmapped node u in V_1 and computes a score for each unmapped node v in V_2 score

Important: The algorithm finds new mappings using the topological structure of the network and the feedback from previous mappings.

Propagation Algorithm Cont.

- **Eccentricity** measures how much an item in a set X "stands out" from the rest. Defined by: $\frac{\max(X) - \max_2(X)}{\sigma(X)}$
  - **standard deviation**
- **Edge Directionality** – mapping scores for nodes u and v are computed separately for incoming and outgoing edges (and then summed).
- **Node Degrees** – the above works in favor of high-degree nodes => divide by square root of degree.
  - **cosine similarity**
- **Revisiting Nodes** – as the algorithm progresses, #mapped nodes increases & errors decrease.
- **Reverse Match** – every match is matched in both directions

Propagation



More complicated on real networks

Propagation

Choose a node v in a network and search most "similar" vertex in the other network

For each node w assign a similarity score ofw and v

Example

Eccentricity measures how much an item in a set X "stands out" from the rest

(m(X) max2(X)) / stddev (X)

Where *max* and *max2* denote the highest and second highest values, respectively

Complexity

- Without revisiting nodes and reverse matches

$$O(|E_1|d_2)$$

- Revisiting: assuming that a node *v* is revisited only if then number of already-mapped neighbors of v has increased:

$$O(|E_1|d_1d_2)$$

- Reverse mapping:

$$O((|E_1| + |E_2|)d_1d_2)$$

EXPERIMENTS

Experiments

Type	Network	Relation.	Nodes	Edges	Av. Deg	Crawled
Target	Twitter	Follow	224K	8.5M	27.7	2007
Auxiliary	Flickr	Contact	3.3M	53M	32.2	2007/8

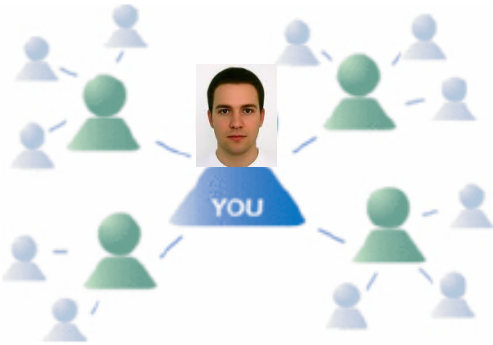
- In both API exposes:
  - Mandatory *username*
  - Optional *name*
  - Optional *location*

Experiment Results

- 30.8% of the mappings were re-identified correctly.
- 57% were not identified
- 12.1% were identified incorrectly
  - 41% of them were mapped to distance 1 nodes from the true mapping
  - 55% of them were mapped to nodes with the same geographic location
  - 27% are completely erroneous

Conclusion

- In reality anonymized graphs are released with some attributes => de-anonymization is even easier!
- Social networks grow
  - overlap between social networks grow
  - Auxiliary info is much richer



Anonymity is not sufficient for privacy when dealing with social networks!

Conclusion

- Anonymization doesn't work on social graphs – Regardless of style of privacy definition
  - Adding noise doesn't help
- Solutions
  - PII should disappear from privacy laws
  - Opt-in rather than opt-out
  - Reputable advertising as a business model

How do I view the web?

- Everything you put on the web is
  - Permanent
  - Public
- Check out my embarassing question on C in 2012
- Check the photo of me posted by my friend Alice on Facebook when I was drunk