PETs for Data Anonymization

Asst. Prof. Sinem Sav

Topics:

1. PETs for data anonymization
    - 1.1 K-anonymity, L-diversity, t-closeness
    - 1.2 Differential privacy

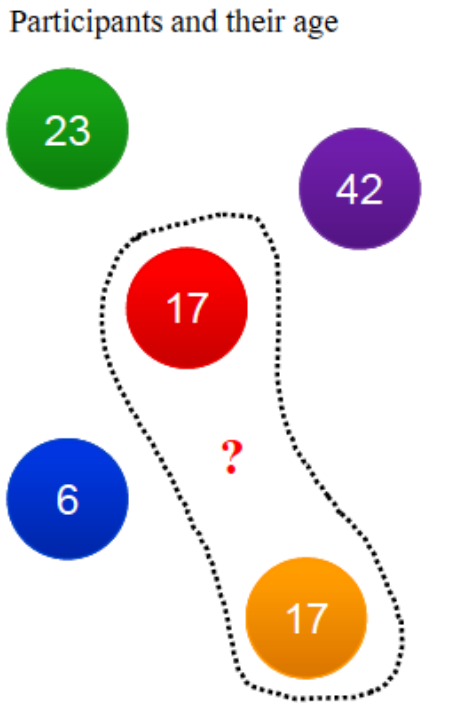
Some slides/ideas adapted from: Jean-Pierre Hubaux, Erman Ayday, Carmela Troncoso, Bryan Ford, Vitaly Shmatikov, Wouter Lueks, David Froelicher, Christian Mouchet, Sylvain Chatel and Florian Tramer

Databases

- Many databases contain sensitive (personal) data
  - Hospital records, internet search information, the set of friends on different social sites, etc.
- It is a common scenario that the release of a function/statistic on such data is socially beneficial
  - Used for apportioning resources, evaluating medical therapies, understanding the spread of disease, improving economic utility, and informing us about ourselves as a species
  - E.g., the usage of hospital records greatly helps medical research
- Hard to publish databases in a privacy-preserving way
- Crucial to ensure that the release of a function on a database does not leak too much information about the individuals
  - Differential privacy is a quite recent notion that tries to formalize this requirement

What is Anonymous?

- One is anonymous, who can not be identified within a set of subjects
  - Anonymity set!
  - Identifying attributes are the same
  - Point of view can be local or global
  - Determined by the attacker model

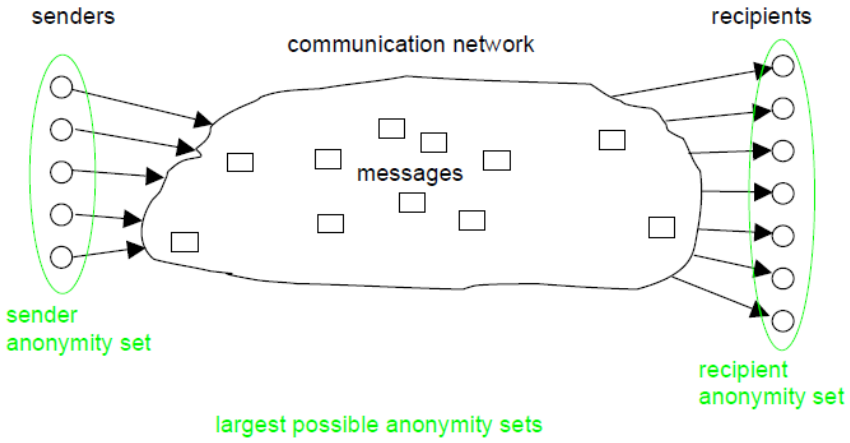


The A_1 anonymity set:
Bob is the one who
is 17 year old. Which one?

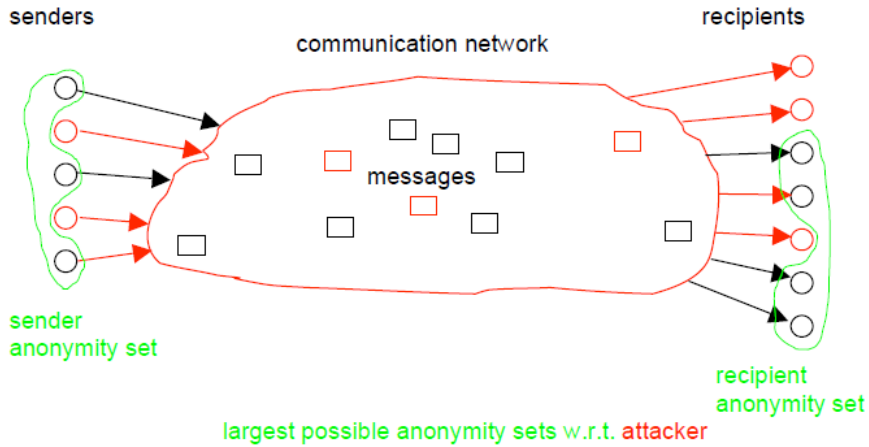
Figure: Gabor Gorgy Gulyas

Reminder - Anonymity

- Anonymity: state of being not identifiable within a set of subjects (the anonymity set)
- All other things being equal, anonymity is the stronger if
  - the respective anonymity set is larger
  - the sending or receiving of the subjects within that set is more evenly distributed



largest possible anonymity sets



largest possible anonymity sets w.r.t. attacker

1. PETs for Data anonymization

Scenario:

You have a set of data that contains personal data and you would like to anonymize it to:

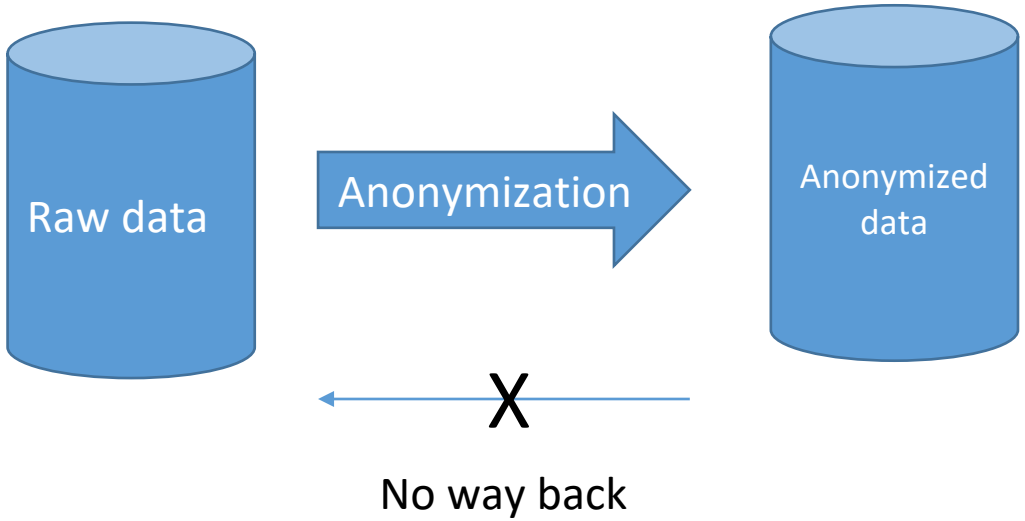
- not be subject to data protection while processing
- make it public for profit
- make it public for researchers

Goal:

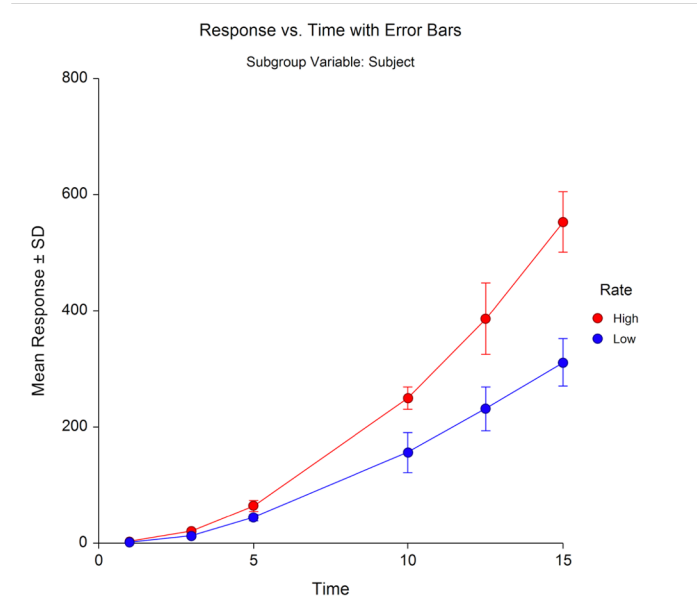
Produce a dataset that **preserves the utility** of the original dataset **without leaking information** about individuals. *This process is known as "*database sanitization*"*

REMEMBER: ANONYMITY IS ABOUT DECOUPLING IDENTITY AND ACTION!

The Quest for Data Anonymization



Analyst



Statistical insights

Privacy Mechanisms for Databases

- Non-interactive mechanisms
  - Database publishes a sanitized dataset
  - Researcher asks arbitrary queries on the sanitized dataset

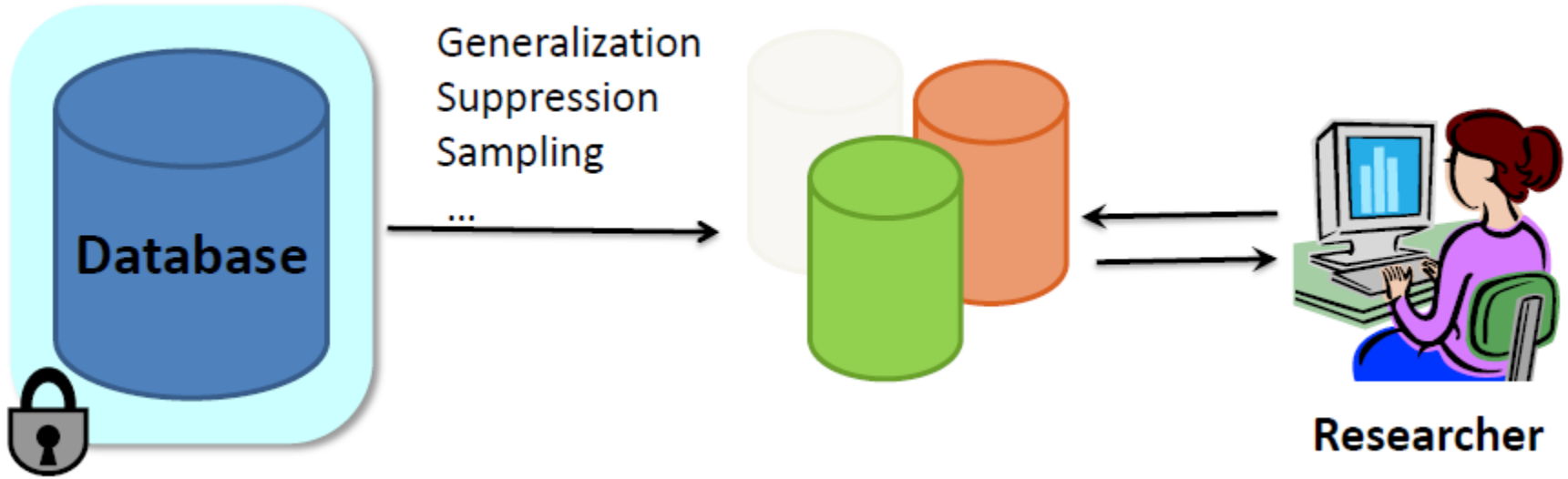


Figure: Ashwin Machanavajjhala

Privacy Mechanisms for Databases

- Interactive mechanisms
  - Researcher directly asks queries to the database
  - Database can choose to answer truthfully or answer with noise
  - Auditor may keep track of all the queries pose to the database and deny queries
- Next …

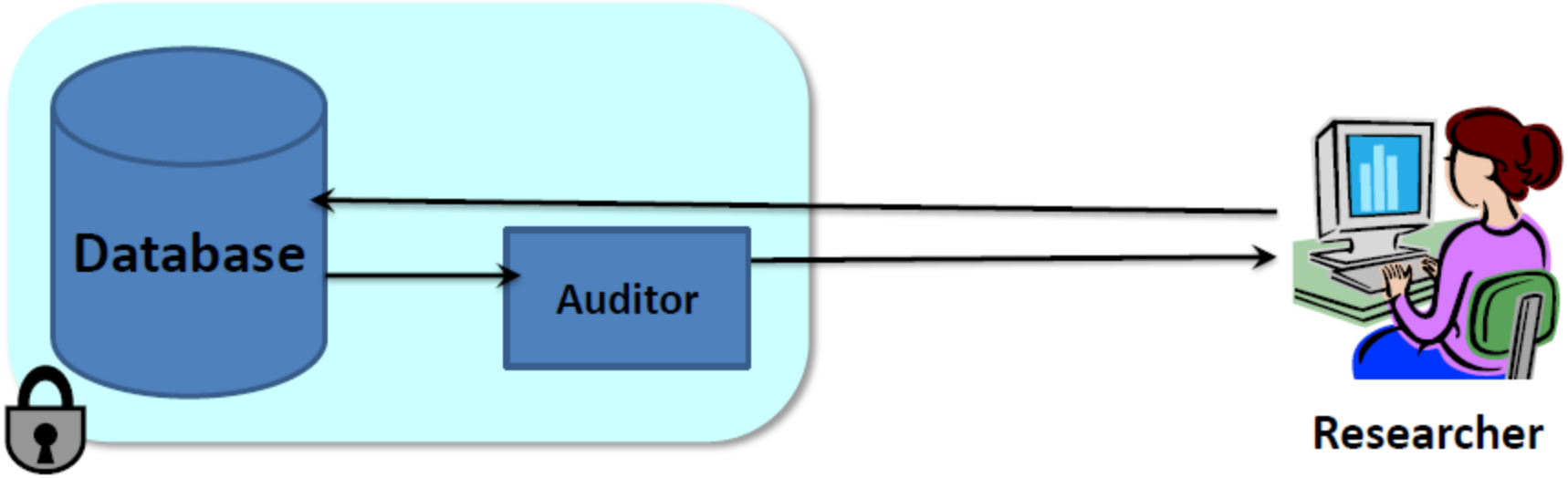


Figure: Ashwin Machanavajjhala

Latanya Sweeney's Attack (1997)

Massachusetts hospital discharge dataset

Medical Data Released as Anonymous

SSN	Name	Ethnicity	Date Of Birth	Sex	ZIP	Marital Status	Problem
		asian	09/27/64	female	02139	divorced	hypertension
		asian	09/30/64	female	02139	divorced	obesity
		asian	04/18/64	male	02139	married	chest pain
		asian	04/15/64	male	02139	married	obesity
		black	03/13/63	male	02138	married	hypertension
		black	03/18/63	male	02138	married	shortness of breath
		black	09/13/64	female	02141	married	shortness of breath
		black	09/07/64	female	02141	married	obesity
		white	05/14/61	male	02138	single	chest pain
		white	05/08/61	male	02138	single	obesity
		white	09/15/61	female	02142	widow	shortness of breath

Voter List

Name	Address	City	ZIP	DOB	Sex	Party	.....
.....	.....	.....	.....	.....	.....	.....	
.....	.....	.....	.....	.....	.....	.....	
Sue J. Carlson	1459 Main St.	Cambridge	*02142*	*9/15/61*	*female*	democrat	.....
.....	.....	.....	.....	.....	.....	.....	

Figure 1: re-identifying anonymous data by linking to external data

Public voter dataset

How Identifiable Are We?

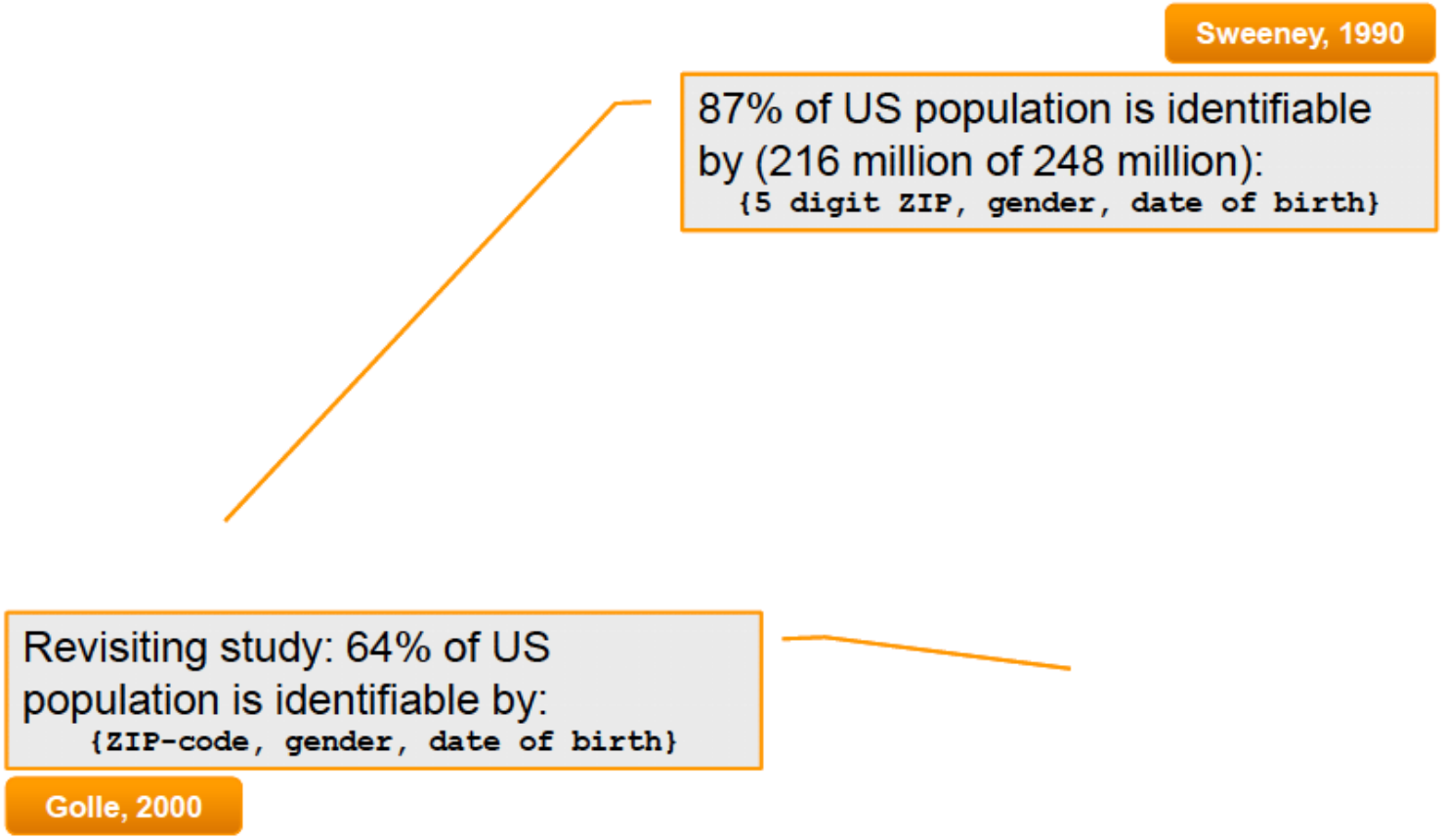


Figure: Gabor Gorgy Gulyas

Attributes

Key Attribute / Identifier	Quasi-identifier		Sensitive attribute
name	**gender**	**zipcode**	**problem**
John	Male	1012	Cancer
Zoey	Female	1013	Flu
Nathan	Male	1016	Heart Disease
Lucas	Male	1015	Heart Disease
Sam	Female	1003	Flu
Max	Male	1012	Flu
Mathias	Male	1014	HIV+
Sarah	Female	1012	Herpes
Julia	Female	1012	Flu

Definition: Quasi-Identifiers

- Key attributes
  - Name, address, phone number - uniquely identifying!
  - Always removed before release
- Quasi-identifiers
  - (5-digit ZIP code, birth date, gender) uniquely identify 87% of the population in the U.S.
  - Can be used for linking anonymized dataset with other datasets

Definition: Sensitive attributes

- Private info
- Medical records, salaries, etc.
- These attributes is what the researchers need, so they are always released directly

Key Attribute	Quasi-identifier			Sensitive attribute
Name	DOB	Gender	Zipcode	Disease
Andre	1/21/76	Male	53715	Heart Disease
Beth	4/13/86	Female	53715	Hepatitis
Carol	2/28/76	Male	53703	Brochitis
Dan	1/21/76	Male	53703	Broken Arm
Ellen	4/13/86	Female	53706	Flu
Eric	2/28/76	Female	53706	Hang Nail

To achieve anonymity we must **decouple** user identities from user attributes

✗ Let's make users pseudonymous

Medical Data

ID	QID			SA
Name	Zipcode	Age	Sex	Disease
13241	47677	29	F	Ovarian Cancer
542562	47602	22	M	Ovarian Cancer
5377	47678	27	F	Prostate Cancer
73563	47905	43	F	Flu
994356	47909	52	M	Heart Disease
24562	47906	47	F	Heart Disease

To achieve anonymity we must **decouple** user identities from user attributes

✗ Possible existence of other databases…

Medical Data

ID	QID			SA
Name	Zipcode	Age	Sex	Disease
13241	47677	29	F	Ovarian Cancer
542562	47602	22	M	Ovarian Cancer
5377	47678	27	F	Prostate Cancer
73563	47905	43	F	Flu
994356	47909	52	M	Heart Disease
24562	47906	47	F	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** user identities from user attributes

✗ Possible existence of other databases…

Medical Data

ID	QID			SA
Name	Zipcode	Age	Sex	Disease
13241	47677	29	F	Ovarian Cancer
542562	47602	22	M	Ovarian Cancer
5377	47678	27	F	Prostate Cancer
73563	47905	43	F	Flu
994356	47909	52	M	Heart Disease
24562	47906	47	F	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** user identities from user attributes

- ✗ Let's make users pseudonymous
- ✗ Let's remove identities

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	F	Ovarian Cancer
47602	22	M	Ovarian Cancer
47678	27	F	Prostate Cancer
47905	43	F	Flu
47909	52	M	Heart Disease
47906	47	F	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** user identities from user attributes

✘ Let's make users pseudonymous

✘ Let's remove identities; that's of little help because some attributes are quasi-identifiers

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	F	Ovarian Cancer
47602	22	M	Ovarian Cancer
47678	27	F	Prostate Cancer
47905	43	F	Flu
47909	52	M	Heart Disease
47906	47	F	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** user identities from user attributes

- ✗ Let's make users pseudonymous
- ✗ Let's remove identities; that's of little help because some attributes are quasi-identifiers
- ✗ Let's remove some attributes

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	*	Ovarian Cancer
47602	22	*	Ovarian Cancer
47678	27	*	Prostate Cancer
47905	43	*	Flu
47909	52	*	Heart Disease
47906	47	*	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

To achieve anonymity we must **decouple** user identities from user attributes

✘ Let's make users pseudonymous

✘ Let's remove identities; that's of little help because some attributes are quasi-identifiers

✘ Let's remove some attributes

Impossible to know what will be a QID

Medical Data

QID			SA
Zipcode	Age	Sex	Disease
47677	29	*	Ovarian Cancer
47602	22	*	Ovarian Cancer
47678	27	*	Prostate Cancer
47905	43	*	Flu
47909	52	*	Heart Disease
47906	47	*	Heart Disease

Voter registration data

Name	Zipcode	Age	Sex
Alice	47677	29	F
Bob	47983	65	M
Carol	47677	22	F
Dan	47532	23	M
Ellen	46789	43	F

Caucasian	HIV+	Flu
Asian	HIV-	Flu
Asian	HIV+	Herpes
Caucasian	HIV-	Acne
Caucasian	HIV-	Herpes
Caucasian	HIV-	Acne

k-Anonymity - Overview

- Each person contained in the database cannot be distinguished from at least k-1 other individuals whose information also appears in the released database.

- Attributes can be: explicit id, quasi id, sensitive

Employee database

Name	Birth date	City
John	1980-01-31	New York
Emily	1976-06-25	Flint
Bob	1985-09-05	New York
Dave	1973-02-07	South Bend
...		

Healthcare database

Birth date	City	Diagnosis
1985-09-05	New York	Stroke
1973-02-07	South Bend	-
1980-01-31	New York	Flu
1976-06-25	Flint	HIV
...		

k-Anonymity Example

Employee database

Name	Birth date	City
John	1980-01-31	New York
Emily	1976-06-25	Flint
Bob	1985-09-05	New York
Dave	1973-02-07	South Bend

Healthcare database

Birth date	City	Diagnosis
198*	New York	Stroke
197*	South Bend	-
198*	New York	Flu
197*	Flint	HIV

?

Better: P(„John has flu")=1 → P(„John has flu")= ½

Employee database

Name	Birth date	City
John	1980-01-31	New York
Emily	1976-06-25	Flint
Bob	1985-09-05	New York
Dave	1973-02-07	South Bend

Healthcare database

Birth date	City	Diagnosis
198*	New York	Stroke
197*	[small city]	-
198*	New York	Flu
197*	[small city]	HIV

Even better: probs are now ½ for all! (2-anonymity)

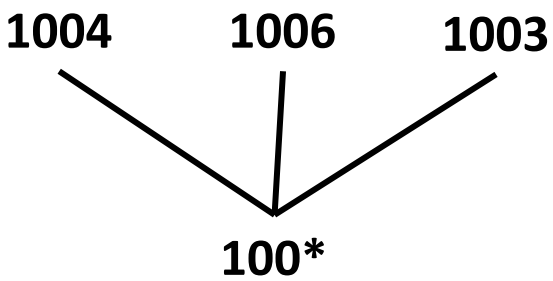
Figure: Gabor Gorgy Gulyas

Anonymization: k-anonymity

- Each person contained in the database cannot be distinguished from at least k-1 other individuals whose information also appears in the released database.

***Generalization**: replace attributes with less specific, but semantically consistent values*

EXAMPLE
zipcode



QID

name	gender	zipcode	problem	
	Male	1012	Cancer	
	Female	100*	Flu	
	Male	100*	Heart Disease	
	Male	100*	Heart Disease	
	Female	100*	Flu	
	Male	1012	Flu	
	Male	100*	HIV+	
	Female	1012	Herpes	
	Female	1012	Flu	

k=2

k-anonymity: example

Assume all people in a given hospital department are asked for their favorite flavor (desert choice!);
Assume the database containing this information is unprotected and gets leaked.

QID

name	gender	zipcode	Favorite flavor
John	Male	1012	Vanilla
Zoey	Female	1013	Chocolate
Nathan	Male	1016	Pistachio
Lucas	Male	1015	Chocolate
Sam	Female	1003	Blueberry
Max	Male	1012	Vanilla
Mathias	Male	1014	Vanilla
Sarah	Female	1012	Pistachio
Julia	Female	1012	Chocolate

QID

name	gender	zipcode	problem	
	Male	1012	Cancer	
	Female	100**	Flu	
	Male	100**	Heart Disease	
	Male	100**	Heart Disease	
	Female	100**	Flu	
	Male	1012	Flu	
	Male	100**	HIV+	
	Female	1012	Herpes	
	Female	1012	Flu	

k=2

k-anonymity: example

QID

name	gender	zipcode	Favorite flavor
John	Male	1012	Vanilla
Zoey	Female	1003	Chocolate
Nathan	Male	1006	Pistachio
Lucas	Male	1005	Chocolate
Sam	Female	1003	Blueberry
Max	Male	1012	Vanilla
Mathias	Male	1004	Vanilla
Sarah	Female	1012	Pistachio
Julia	Female	1012	Chocolate

Who has cancer, John or Max?

QID

name	gender	zipcode	problem	
	Male	1012	Cancer	
	Female	100*	Flu	
	Male	100*	Heart Disease	
	Male	100*	Heart Disease	
	Female	100*	HIV+	
	Male	1012	Flu	
	Male	100*	HIV+	
	Female	1012	Herpes	
	Female	1012	Flu	

k=2

k-anonymity: example

QID

name	gender	zipcode	Favorite flavor
John	Male	1012	Vanilla
Zoey	Female	1013	Chocolate
Nathan	Male	1016	Pistachio
Lucas	Male	1015	Chocolate
Sam	Female	1003	Blueberry
Max	Male	1012	Vanilla
Mathias	Male	1014	Vanilla
Sarah	Female	1012	Pistachio
Julia	Female	1012	Chocolate

Does John have cancer or flu?

QID

name	gender	zipcode	problem
	Male	1012	Cancer
	Female	10**	Flu
	Male	10**	Heart Disease
	Male	10**	Heart Disease
	Female	10**	HIV+
	Male	1012	Flu
	Male	10**	HIV+
	Female	1012	Herpes
	Female	1012	Flu

k=2

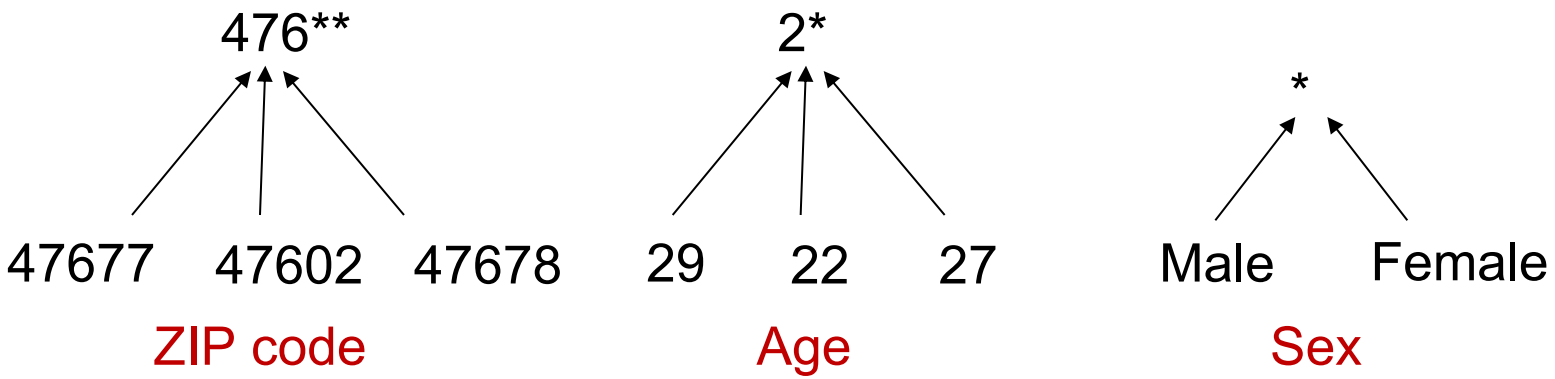
Achieving k-Anonymity

- Generalization
  - Replace specific quasi-identifiers with less specific values until get k identical values
  - Partition ordered-value domains into intervals
- Suppression
  - “Not releasing any value at all”
  - When generalization causes too much information loss
    - This is common with “outliers”
- Lots of algorithms in the literature
  - Aim to produce “useful” anonymizations

  … usually without any clear notion of utility

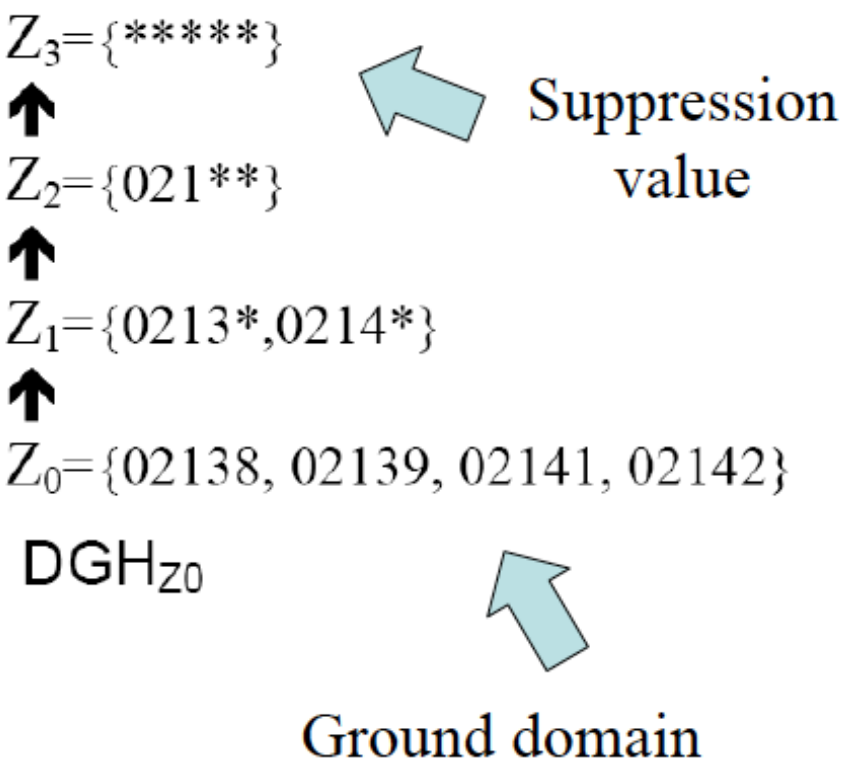
Generalization

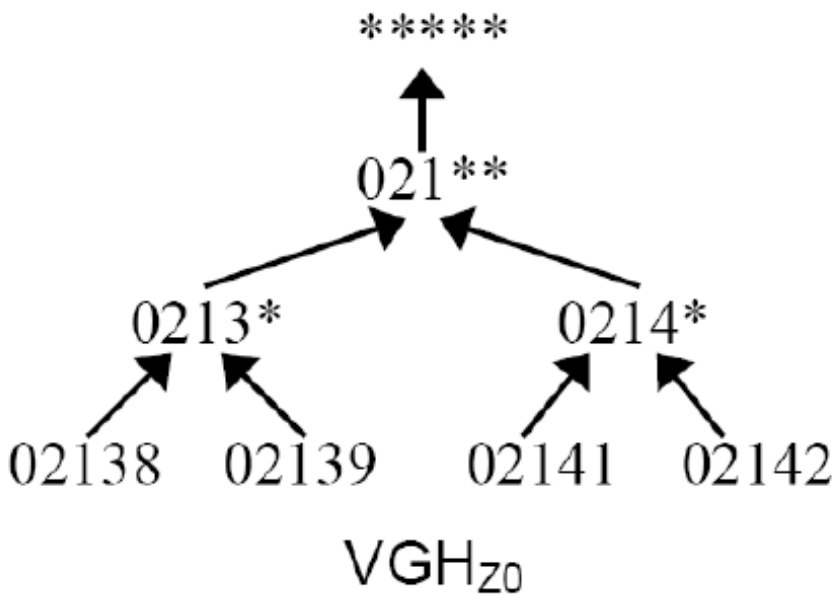
- Goal of k-Anonymity
  - Each record is indistinguishable from at least k-1 other records
  - These k records form an equivalence class
- Generalization: replace quasi-identifiers with less specific, but semantically consistent values



Domain and Value Generalization Hierarchies

- ZIP attribute





Different Generalizations

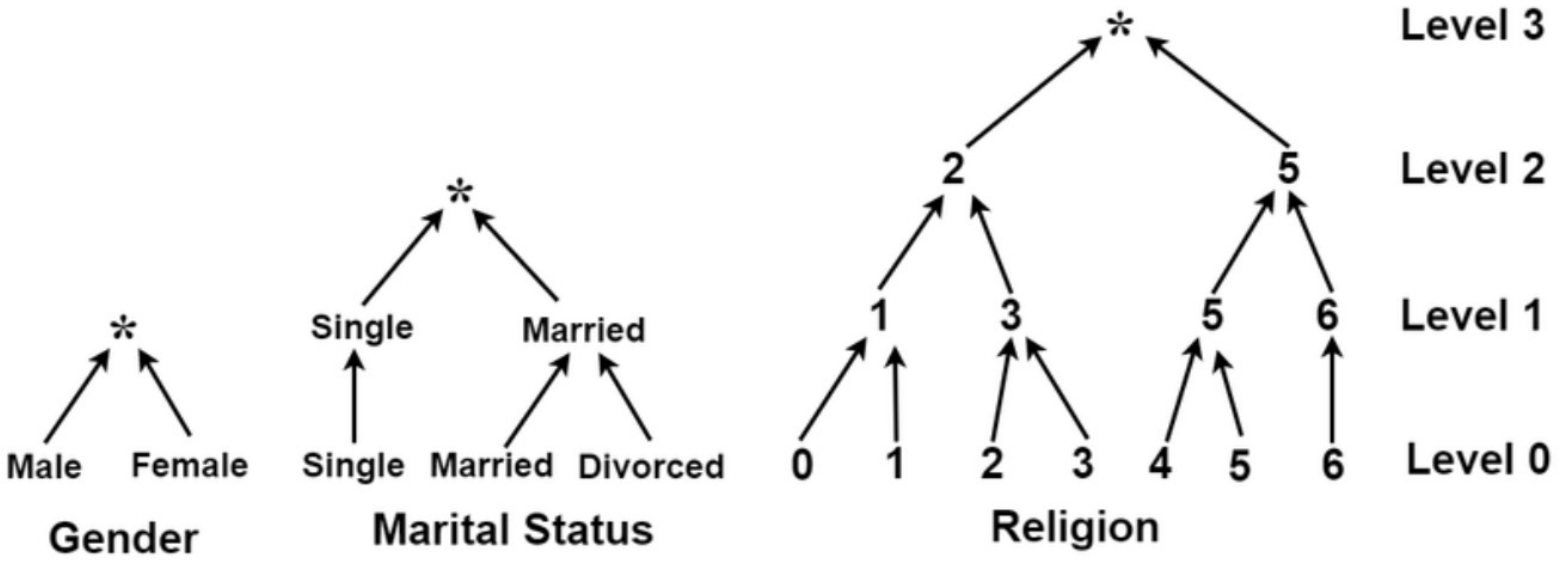
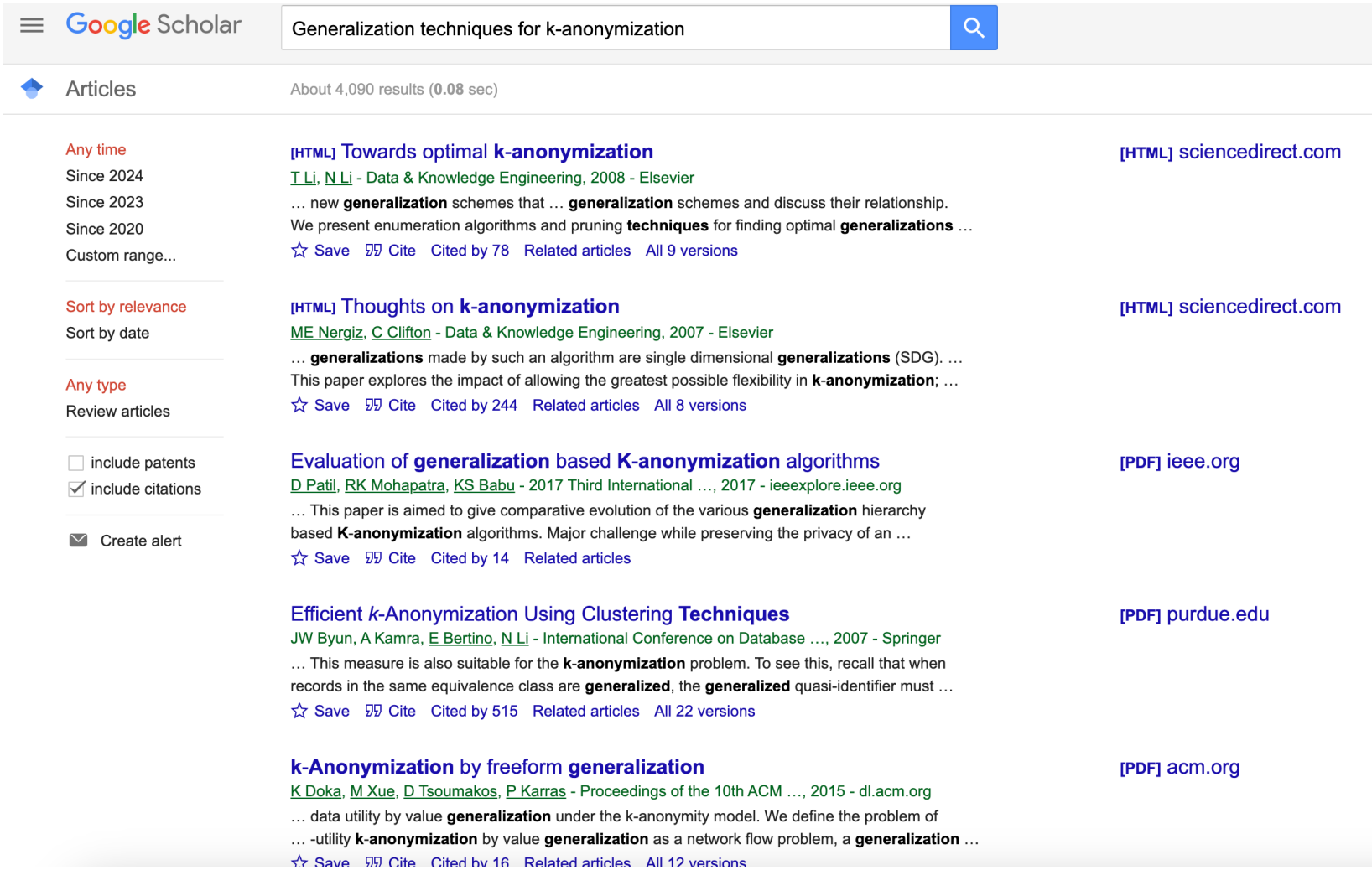


Image taken from: Olatunji, Iyiola E., et al. "A review of anonymization for healthcare data." *Big data* (2022).


Google Scholar
Generalization techniques for k-anonymization
Articles
About 4,090 results (0.08 sec)
Any time
Since 2024
Since 2023
Since 2020
Custom range...
Sort by relevance
Sort by date
Any type
Review articles
include patents
include citations
Create alert
[HTML] Towards optimal k-anonymization
T Li, N Li - Data & Knowledge Engineering, 2008 - Elsevier
… new generalization schemes that … generalization schemes and discuss their relationship. We present enumeration algorithms and pruning techniques for finding optimal generalizations …
Save Cite Cited by 78 Related articles All 9 versions
[HTML] sciencedirect.com
[HTML] Thoughts on k-anonymization
ME Nergiz, C Clifton - Data & Knowledge Engineering, 2007 - Elsevier
… generalizations made by such an algorithm are single dimensional generalizations (SDG). … This paper explores the impact of allowing the greatest possible flexibility in k-anonymization; …
Save Cite Cited by 244 Related articles All 8 versions
[HTML] sciencedirect.com
Evaluation of generalization based K-anonymization algorithms
D Patil, RK Mohapatra, KS Babu - 2017 Third International …, 2017 - ieeexplore.ieee.org
… This paper is aimed to give comparative evolution of the various generalization hierarchy based K-anonymization algorithms. Major challenge while preserving the privacy of an …
Save Cite Cited by 14 Related articles
[PDF] ieee.org
Efficient k-Anonymization Using Clustering Techniques
JW Byun, A Kamra, E Bertino, N Li - International Conference on Database …, 2007 - Springer
… This measure is also suitable for the k-anonymization problem. To see this, recall that when records in the same equivalence class are generalized, the generalized quasi-identifier must …
Save Cite Cited by 515 Related articles All 22 versions
[PDF] purdue.edu
k-Anonymization by freeform generalization
K Doka, M Xue, D Tsoumakos, P Karras - Proceedings of the 10th ACM …, 2015 - dl.acm.org
… data utility by value generalization under the k-anonymity model. We define the problem of … -utility k-anonymization by value generalization as a network flow problem, a generalization …
Save Cite Cited by 16 Related articles All 12 versions
[PDF] acm.org

k-Anonymity via Generalization

- QI = {Race, ZIP}
- k = 2

- k-anonymous relation should have at least 2 tuples with the same values on

$$\text{Dom}(\text{Race}_i) \times \text{Dom}(\text{ZIP}_j)$$

where Race_i and ZIP_j are chosen from corresponding DGHs

k-Anonymity via Generalization

Race E_0	ZIP Z_0
Black	02138
Black	02139
Black	02141
Black	02142
White	02138
White	02139
White	02141
White	02142

PT

$Z_3=\{*****\}$

↑

$Z_2=\{021**\}$

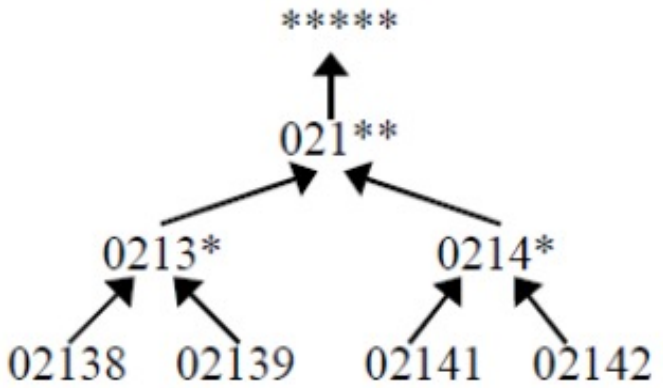
↑

$Z_1=\{0213*,0214*\}$

↑

$Z_0=\{02138, 02139, 02141, 02142\}$

DGH_{Z0}



VGH_{Z0}

$Z_2=\{******\}$

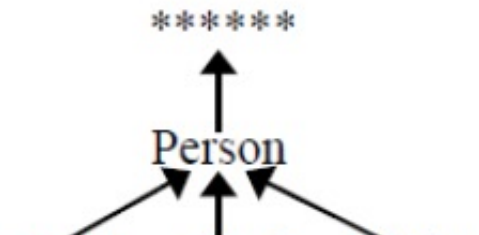
↑

$Z_1=\{Person\}$

↑

$Z_0=\{Asian,Black,White\}$

DGH_{E0}



VGH_{E0}

k-Anonymity via Generalization

Race E_0	ZIP Z_0
Black	02138
Black	02139
Black	02141
Black	02142
White	02138
White	02139
White	02141
White	02142

PT

Race E_1	ZIP Z_0
Person	02138
Person	02139
Person	02141
Person	02142
Person	02138
Person	02139
Person	02141
Person	02142

$GT_{[1,0]}$

Race E_1	ZIP Z_1
Person	0213*
Person	0213*
Person	0214*
Person	0214*
Person	0213*
Person	0213*
Person	0214*
Person	0214*

$GT_{[1,1]}$

Race E_0	ZIP Z_2
Black	021**
Black	021**
Black	021**
Black	021**
White	021**
White	021**
White	021**
White	021**

$GT_{[0,2]}$

Race E_0	ZIP Z_1
Black	0213*
Black	0213*
Black	0214*
Black	0214*
White	0213*
White	0213*
White	0214*
White	0214*

$GT_{[0,1]}$

- The number of generalizations, enforced at the attribute level, for table T is:

$$\prod_{i=1}^{n}(|DGH_i| + 1)$$

- Total number of generalizations for PT is:

(DGH_Race+1).(DGH_ZIP + 1) = 12

Do they achieve k=2 anonymity? Which generalization to use?

k-Minimal Generalization

- Given $|R| \geq k$, there is always a trivial solution
  - Generalize all attributes to VGH root
  - Not very useful if there exists another k-anonymization with higher granularity (more specific) values

- k-minimal generalization
  - Satisfies k-anonymity
  - None of its specializations satisfies k-anonymity
  - E.g., [0,2] is not minimal, since [0,1] is k-anonymous
  - E.g., [1,0] is minimal, since [0,0] is not k-anonymous

- A table T, generalization of PT, is k-minimal if it satisfies k-anonymity and there does not exist a generalization of PT satisfying k-anonymity of which T is a generalization.

Precision Metric, Prec(.)

- Multiple k-minimal generalizations may exist
  - E.g., [1,0] and [0,1] from the example
- Precision metric indicates the generalization with minimal information loss and maximal usefulness
- Problem: how to define usefulness

Precision Metric, Prec(.)

- Precision: average height of generalized values, normalized by VGH depth per attribute per record

$$Prec(T') = 1 - \frac{\sum_{i=1}^{N_A} \sum_{j=1}^{N'} \frac{h}{|DGH_{A_i}|}}{N \times N_A}$$

- N_A : number of attributes (quasi-identifiers)
- N: data set size (number of rows in the original table)
- N': number of rows in the generalized table T'
- h: generalization level of the attribute
- |DGH(A_i)| : depth of the VGH for attribute A_i

$$Prec(T') = 1 - \frac{\sum_{i=1}^{N_A} \sum_{j=1}^{N'} \frac{h}{|DGH_{A_i}|}}{N \times N_A}$$

- N = N’ if no rows of the original table are deleted/suppressed
- When T = T’, each value is in the ground domain
  - Each h = 0, and hence Prec(T’) = 1
- When each value in T’ is the maximal element of its hierarchy
  - Each h = |DGH(A_i)|, and hence Prec(T’) = 0
- GT[1,0] and GT[0,1] each generalize values up one level
  - Since |DGH_Race| = 2 and |DGH_ZIP| = 3, Prec(GT[0,1]) > Prec(GT[1,0]).

k-Minimal Distortion

- Most precise release that adheres to k-anonymity
- Precision measured by *Prec(.)*
- Any k-minimal distortion is a k-minimal generalization

- In the example, only [0,1] is a k-minimal distortion
  - [0,0] is not k-anonymous
  - [1,0] and others are less precise

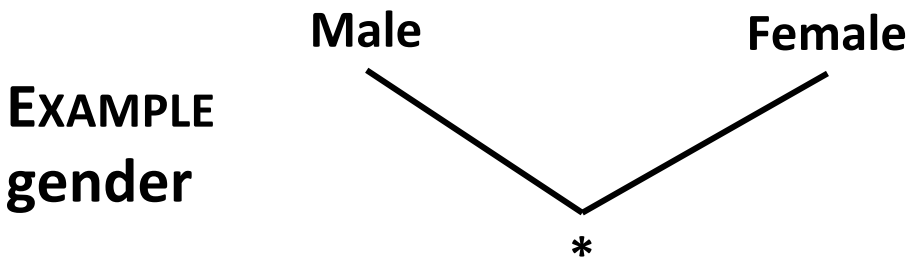
Complexity

- Given some data set *R* and a QI *Q*, does *R* satisfy k- anonymity over *Q*?
  - Easy to tell in polynomial time
- Finding an *optimal* anonymization is not easy
  - NP-hard
- Heuristic solutions exist
  - DataFly, Incognito, Mondrian, etc.

Suppression (removal of attributes)

- To improve anonymity, identifying attributes can be ***suppressed***

(note that suppression is the ultimate generalization!)



QID

name	gender	zipcode	problem
	*	1012	Cancer
	*	100*	Flu
	*	100*	Heart Disease
	*	100*	Heart Disease
	*	100*	Flu
	*	1012	Cancer
	*	100*	HIV+
	*	1012	Herpes
	*	1012	Flu

k=4

Example of Generalization (1)

Released table

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

External data Source

Name	Birth	Gender	ZIP	Race
Andre	1964	m	02135	White
Beth	1964	f	55410	Black
Carol	1964	f	90210	White
Dan	1967	m	02174	White
Ellen	1968	f	02237	White

By linking these 2 tables, you still don't learn Andre's problem

Example of Generalization (2)

Microdata

QID			SA
Zipcode	Age	Sex	Disease
47677	29	F	Ovarian Cancer
47602	22	F	Ovarian Cancer
47678	27	M	Prostate Cancer
47905	43	M	Flu
47909	52	F	Heart Disease
47906	47	M	Heart Disease

Generalized table

QID			SA
Zipcode	Age	Sex	Disease
476**	2*	*	Ovarian Cancer
476**	2*	*	Ovarian Cancer
476**	2*	*	Prostate Cancer
4790*	[43,52]	*	Flu
4790*	[43,52]	*	Heart Disease
4790*	[43,52]	*	Heart Disease

!!

- Released table is 3-anonymous
- If the adversary knows Alice's quasi-identifier (47677, 29, F), he still does not know which of the first 3 records corresponds to Alice's record

Things to be Careful About

- Homogeneity attack
- Background knowledge attack
- Unsorted matching attack

Limitations of k-anonymity

3-anonymous patient table – are there privacy issues?

QID

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	≥40	Flu
4790*	≥40	Heart Disease
4790*	≥40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

Limitations of k-anonymity

A 3-anonymous patient table

Homogeneity attack

Bob	
Zipcode	***Age***
47678	27

QID

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	≥40	Flu
4790*	≥40	Heart Disease
4790*	≥40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

Does not provide privacy when sensitive values lack **diversity**

It does not matter who John is, **anyone with zipcode 476** in their 20s has a heart disease**

Limitations of k-anonymity

Homogeneity attack

Bob	
Zipcode	***Age***
47678	27

Background knowledge attack

Carl	
Zipcode	***Age***
47673	36

A 3-anonymous patient table

Zipcode	Age	Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
476**	2*	Heart Disease
4790*	≥40	Flu
4790*	≥40	Heart Disease
4790*	≥40	Cancer
476**	3*	Heart Disease
476**	3*	Cancer
476**	3*	Cancer

If the adversary has background knowledge, e.g., "Carl is Japanese and the tendency of heart diseases in Japan is lower", then **it is likely that Carl has cancer**

What type of attack was this?

Microdata

QID			SA
Zipcode	Age	Sex	Disease
47677	29	F	Ovarian Cancer
47602	22	F	Ovarian Cancer
47678	27	M	Prostate Cancer
47905	43	M	Flu
47909	52	F	Heart Disease
47906	47	M	Heart Disease

Generalized table

QID			SA
Zipcode	Age	Sex	Disease
476**	2*	*	Ovarian Cancer
476**	2*	*	Ovarian Cancer
476**	2*	*	Prostate Cancer
4790*	[43,52]	*	Flu
4790*	[43,52]	*	Heart Disease
4790*	[43,52]	*	Heart Disease

!!

- Released table is 3-anonymous
- If the adversary knows Alice's quasi-identifier (47677, 29, F), he still does not know which of the first 3 records corresponds to Alice's record

Unsorted Matching Attack

- Problem: records appear in the same order in the released table as in the original table
- Solution: randomize order before releasing

Race	ZIP
Asian	02138
Asian	02139
Asian	02141
Asian	02142
Black	02138
Black	02139
Black	02141
Black	02142
White	02138
White	02139
White	02141
White	02142

PT

Race	ZIP
Person	02138
Person	02139
Person	02141
Person	02142
Person	02138
Person	02139
Person	02141
Person	02142
Person	02138
Person	02139
Person	02141
Person	02142

GT1

Race	ZIP
Asian	02130
Asian	02130
Asian	02140
Asian	02140
Black	02130
Black	02130
Black	02140
Black	02140
White	02130
White	02130
White	02140
White	02140

GT2

k-Anonymity – Discussion

- Generalization fundamentally relies on spatial locality
  - Each record must have k close neighbors
- Real-world datasets are very sparse
  - Many attributes (dimensions)
    - Netflix Prize dataset: 17,000 dimensions
    - Amazon customer records: several million dimensions
  - “Nearest neighbor” is very far
- Projection to low dimensions loses all info ⇒ k-anonymized datasets are useless

k-Anonymity Discussion

- These attacks show that in addition to k-anonymity, the sanitized table should also ensure diversity
- All tuples that share the same values of their quasi-identifiers should have diverse values for their sensitive attributes

l-diversity and t-closeness

l-Diversity

- An equivalence class is said to have l-diversity if there are at least l well-represented values for the sensitive attribute
- A table is said to have l-diversity if every equivalence class of the table has l-diversity

	ZIP Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

A 3-diverse hospital records dataset

[1] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data, vol. 1, no. 1, Mar. 2007

53

Example case about ℓ-Diversity (why k-anonymity was not enough)

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Table with raw data

”Anonymization”

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

4-anonymous table

A. Machanavajjhala et al., ”l-diversity: privacy beyond k-anonymity, ACM Trans. on KDD, 2007

Problem: if the adversary knows that Bob is in his 30s, he can immediately infer that Bob has cancer.

Example about ℓ-Diversity

- An equivalence class has ℓ-diversity if there are at least ℓ well-represented values for the sensitive attribute.
- A database has ℓ-diversity if every equivalence class has ℓ–diversity.

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	*	Heart Disease
4	1305*	≤ 40	*	Viral Infection
9	1305*	≤ 40	*	Cancer
10	1305*	≤ 40	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

What is l in this table?

Example about ℓ-Diversity

- An equivalence class has ℓ-diversity if there are at least ℓ well-represented values for the sensitive attribute.
- A database has ℓ-diversity if every equivalence class has ℓ–diversity.

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	*	Heart Disease
4	1305*	≤ 40	*	Viral Infection
9	1305*	≤ 40	*	Cancer
10	1305*	≤ 40	*	Cancer
5	1485*	> 40	*	Cancer
6	1485*	> 40	*	Heart Disease
7	1485*	> 40	*	Viral Infection
8	1485*	> 40	*	Viral Infection
2	1306*	≤ 40	*	Heart Disease
3	1306*	≤ 40	*	Viral Infection
11	1306*	≤ 40	*	Cancer
12	1306*	≤ 40	*	Cancer

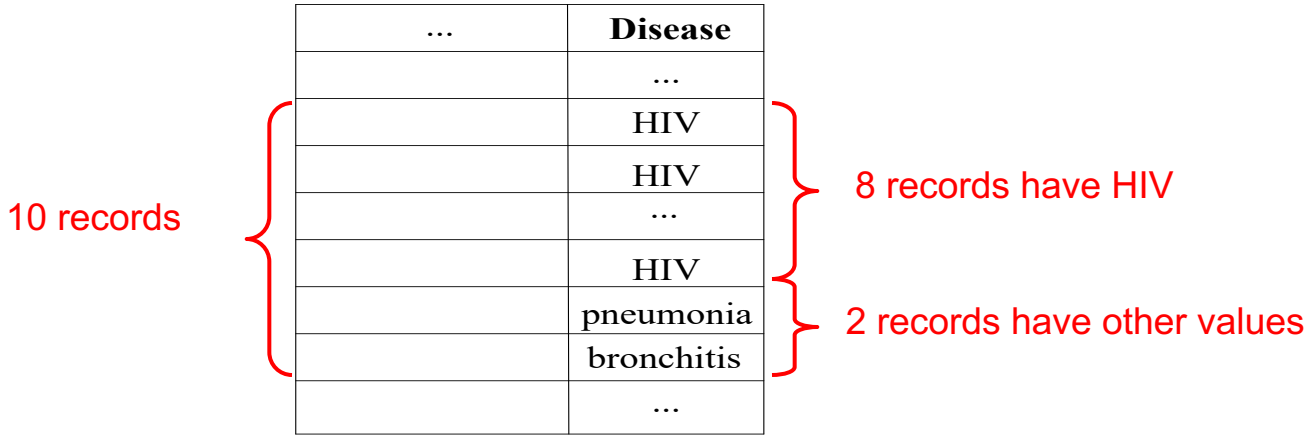
3-diverse table: there are (at least) 3 different sensitive conditions in each equivalence class

l-Diversity Variations

- Distinct l-Diversity
- Entropy l-Diversity
- Recursive (c,l)-Diversity

Distinct l-Diversity

- Each equivalence class has at least l well-represented sensitive values
- Doesn't prevent probabilistic inference attacks



Entropy l-Diversity

- In each equivalence class, different sensitive values must be distributed evenly
- The entropy of the distribution of sensitive values in each equivalence class is at least log(l)
- Entropy of an equivalence class:

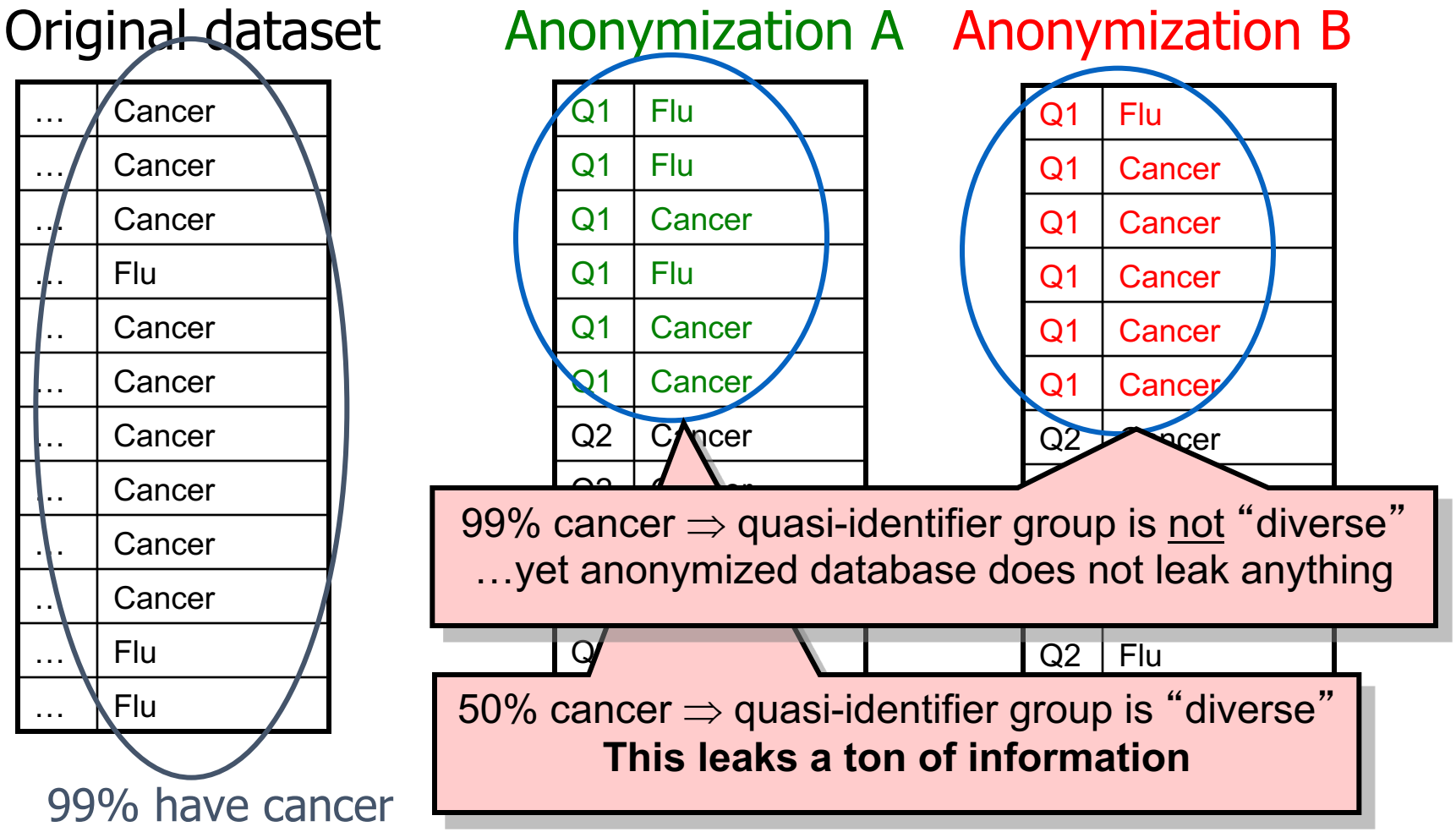
$$Entropy(E) = -\sum_{s \in S} p(E,s) \log p(E,s)$$

  - p(E,s): fraction of records in E that have sensitive value s.
- May be too restrictive
  - The entropy of the entire table may be low if a few values are very common

Recursive (c,l)-Diversity

- $r_1 < c(r_l + r_{l+1} + \ldots + r_m)$
  - r_i is the frequency of the i^{th} most frequent value
  - m: number of distinct sensitive attributes in an equivalence class
  - Should hold for all equivalence classes
- Intuition: the most frequent value does not appear too frequently
  - And the less frequent values do not appear too rarely.

l-Diversity Limitations



l-Diversity Limitations

	ZIP Code	Age	Salary	Disease
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Original salary/disease table

	ZIP Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

(*)

3-diverse version of the table

(*) This subset is vulnerable: if the attacker knows that the targeted individual has a low income or is in her 20s, he will know that she has a stomach-related disease.

l-Diversity Limitations

- Example: sensitive attribute is HIV+ (1%) or HIV- (99%)
  - Very different degrees of sensitivity!
- l-diversity is unnecessary
  - 2-diversity is unnecessary for an equivalence class that contains only HIV- records
- l-diversity is difficult to achieve in scenarios like above
  - Suppose there are 10000 records in total in Example database
  - To have distinct 2-diversity, there can be at most 10000*1%=100 equivalence classes

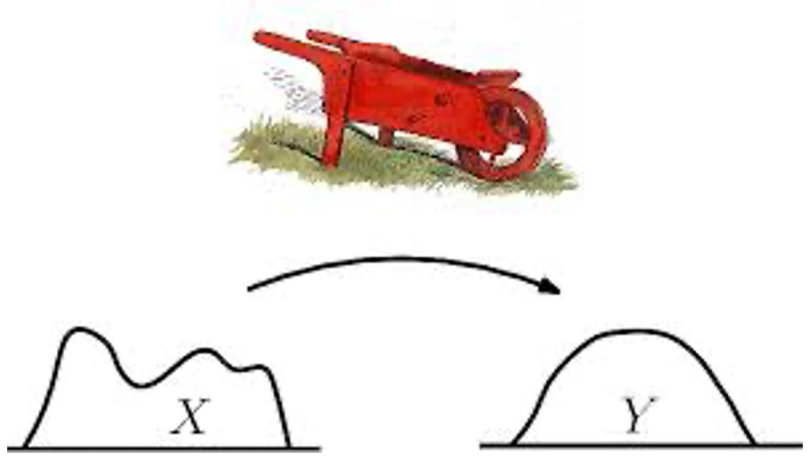
Skewness Attack

- Example: sensitive attribute is HIV+ (1%) or HIV- (99%)
- Consider an equivalence class that contains an equal number of HIV+ and HIV- records
- Diverse, but potentially violates privacy!
- l-diversity does not differentiate:
  - Equivalence class 1: 49 HIV+ and 1 HIV-
  - Equivalence class 2: 1 HIV+ and 49 HIV-

l-diversity does not consider overall distribution of sensitive values!

t-closeness

- An equivalence class has *t-closeness* if the **distance** between the distribution of a sensitive attribute in this class and the distribution of the attribute in the whole table is no more than a *threshold* t.
- A table has *t-closeness* if all equivalence classes have t-*closeness.*
- The distance is usually measured as earth mover's distance (EMD), also known as Wasserstein metric : Minimal amount of work needed to transform one distribution to another by moving distribution mass between each other



N. Li, T. Li and S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity", 2007 doi: 10.1109/ICDE.2007.367856.

Limitations of t-closeness

Caucasian	787XX	HIV+	Flu
Asian	787XX	HIV-	Flu
Asian	787XX	HIV+	Herpes
Caucasian	787XX	HIV-	Acne
Caucasian	787XX	HIV-	Herpes
Caucasian	787XX	HIV-	Acne

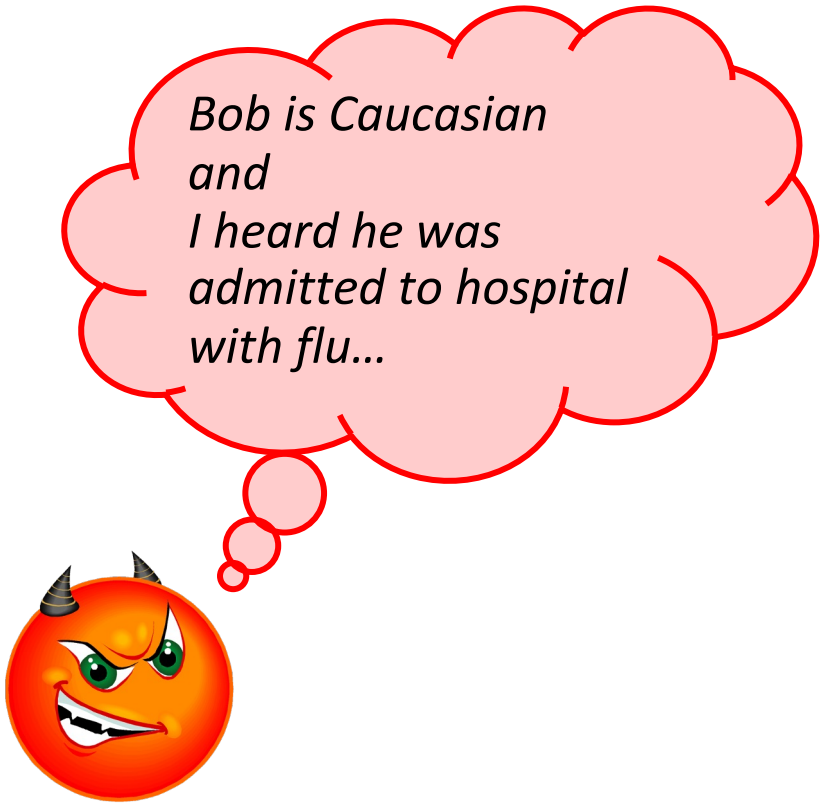
t-Closeness

Caucasian	787XX	HIV+	Flu
Asian	787XX	HIV-	Flu
Asian	787XX	HIV+	Herpes
Caucasian	787XX	HIV-	Acne
Caucasian	787XX	HIV-	Herpes
Caucasian	787XX	HIV-	Acne

This is k-anonymous, l-diverse and t-close…

…so secure, right?

What Does the Attacker Know?



Caucasian	787XX	HIV+	Flu
Asian	787XX	HIV-	Flu
Asian	787XX	HIV+	Herpes
Caucasian	787XX	HIV-	Acne
Caucasian	787XX	HIV-	Herpes
Caucasian	787XX	HIV-	Acne

Takeaways

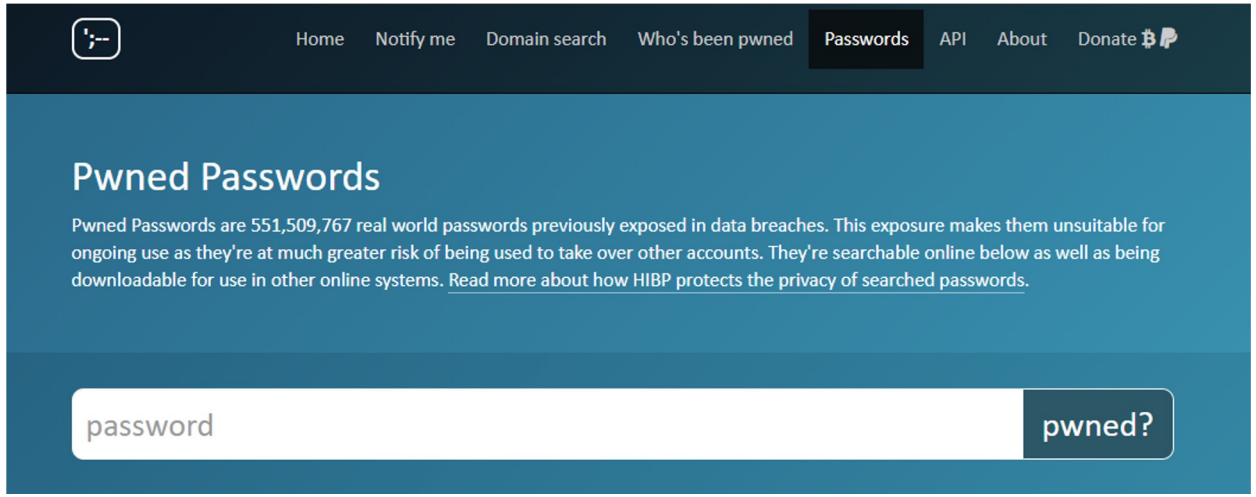
Anonymizing a dataset via generalization and suppression is extremely hard

The k-anonymity idea focuses on transforming the dataset not its semantics

Achieving k-anonymity, l-diversity, t-closeness is hard, and still does not guarantee privacy

The adversary's background **can be anything (auxiliary information)**

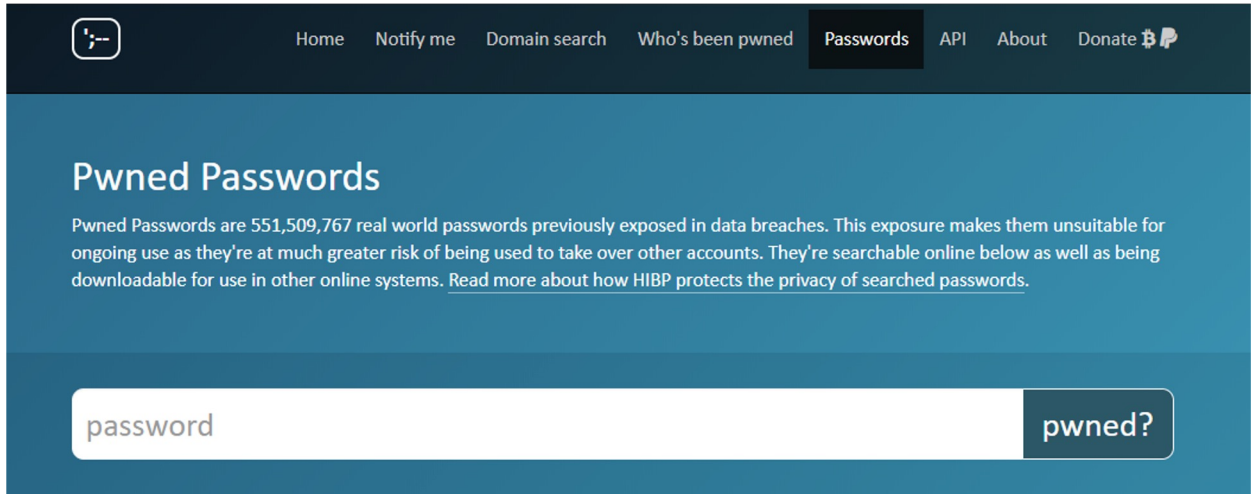
Let’s exercise your privacy brain



Would you send your password in the clear?

https://haveibeenpwned.com/

Let’s exercise your privacy brain

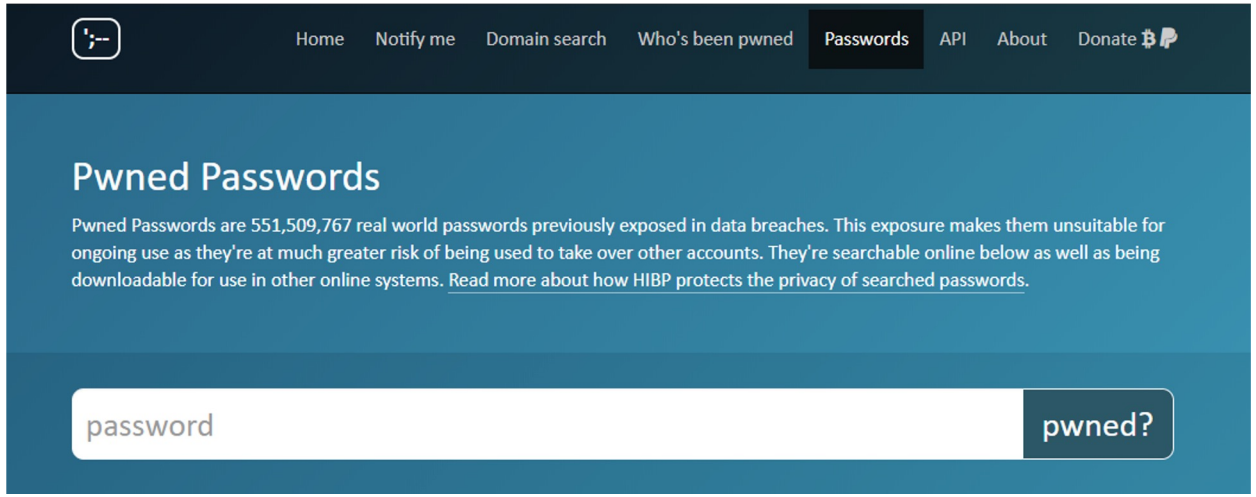


Would you send your password in the clear?

Would you send a hash?

https://haveibeenpwned.com/

Let’s exercise your privacy brain



Would you send your password in the clear?

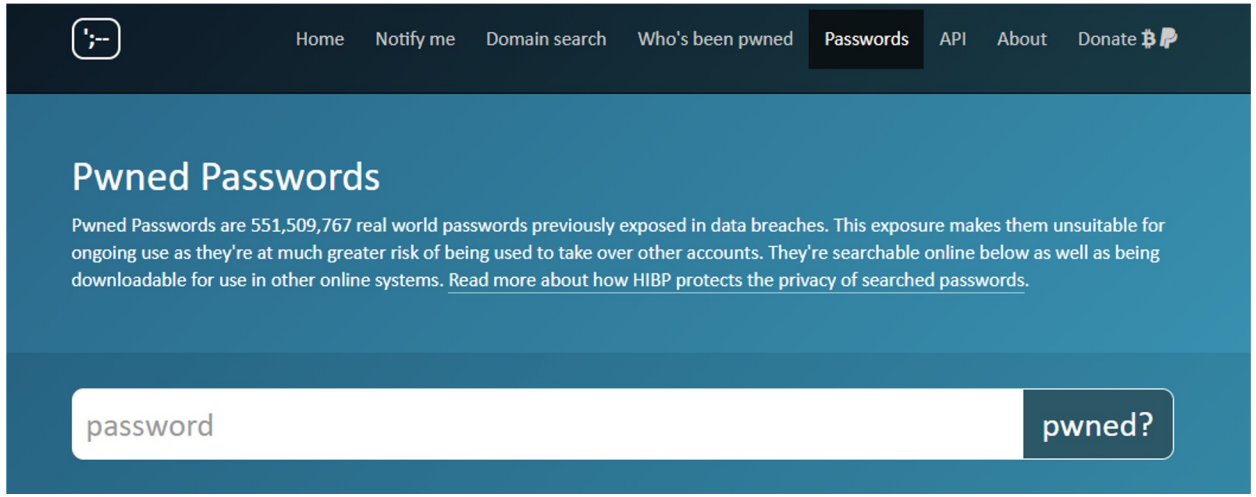
Would you send a hash?

What they do: send the first 5 bytes of the hash of the password and receive a list of 475 suffixes to check offline

1. (21BD1) **0018A45C4D1DEF81644B54AB7F969B88D65:1** (password "lauragpe")
2. (21BD1) **00D4F6E8FA6EECAD2A3AA415EEC418D38EC:2** (password "alexguo029")
3. (21BD1) **011053FD0102E94D6AE2F8B83D76FAF94F6:1** (password "BDnd9102")
4. (21BD1) **012A7CA357541F0AC487871FEEC1891C49C:2** (password "melobie")
5. (21BD1) **0136E006E24E7D15213981 5FB0FC6A50B15:2** (password "quvekyny")
6. …

https://haveibeenpwned.com/
https://blog.cloudflare.com/validating-leaked-passwords-with-k-anonymity/

Let’s exercise your privacy brain



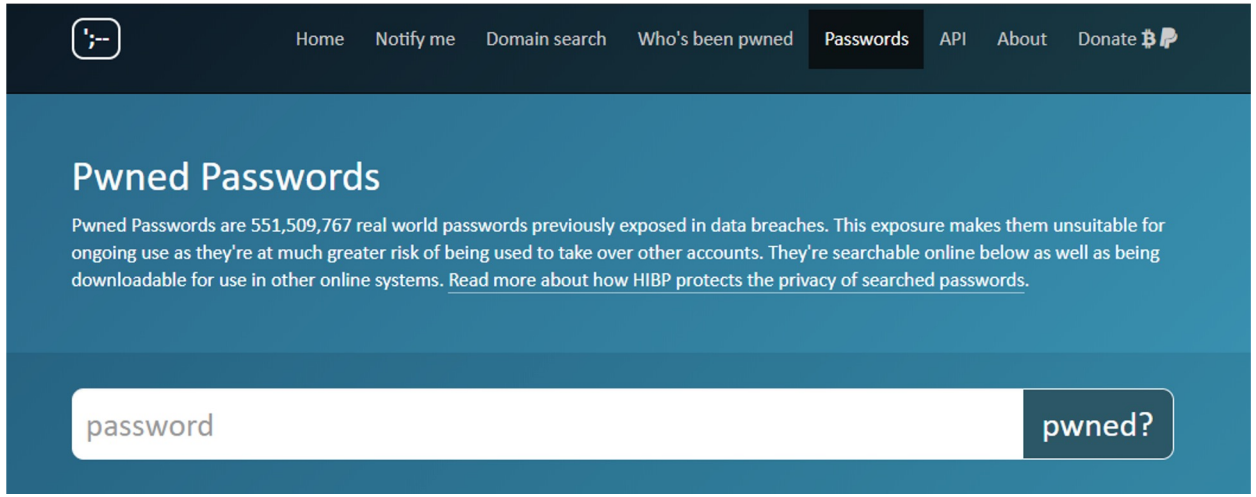
Would you send your password in the clear?

Would you send a hash?

What they do: send the first 5 bytes of the hash of the password and receive a list of 475 suffixes to check offline

https://haveibeenpwned.com/
https://blog.cloudflare.com/validating-leaked-passwords-with-k-anonymity/

Let’s exercise your privacy brain



From the point of view of the server (that receives the 5-bytes suffix

What is the privacy of the password?

Would you send your password in the clear?

Would you send a hash?

What they do: send the first 5 bytes of the hash of the password and receive a list of 475 suffixes to check offline

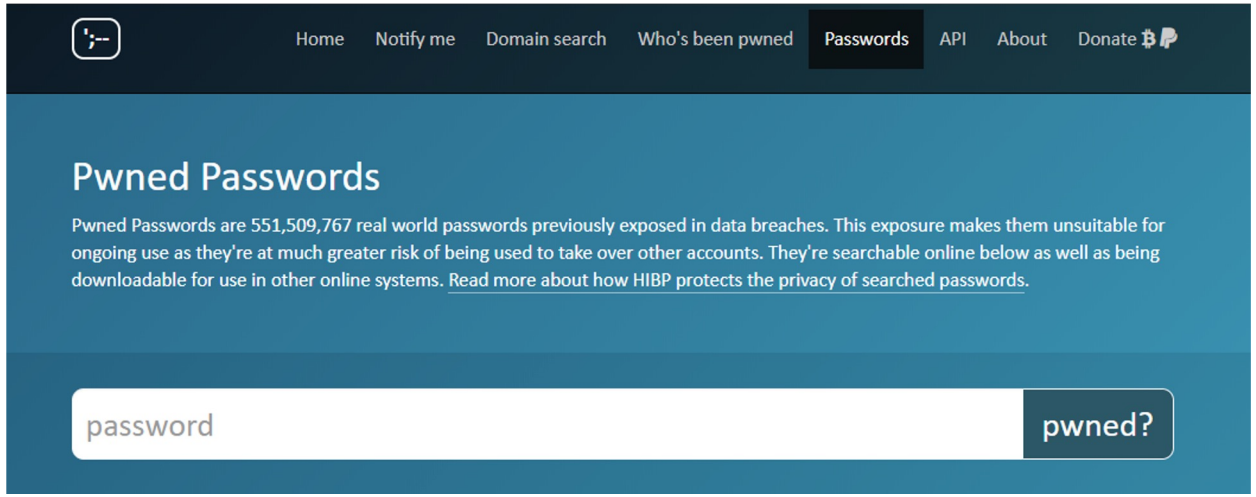
Send

Receive

1. (21BD1) 0018A45C4D1DEF81644B54AB7F969B88D65:1 (password "lauragpe")
2. (21BD1) 00D4F6E8FA6EECAD2A3AA415EEC418D38EC:2 (password "alexguo029")
3. (21BD1) 011053FD0102E94D6AE2F8B83D76FAF94F6:1 (password "BDnd9102")
4. (21BD1) 012A7CA357541F0AC487871FEEC1891C49C:2 (password "melobie")
5. (21BD1) 0136E006E24E7D15213981 5FB0FC6A50B15:2 (password "quvekyny")
6. …

https://haveibeenpwned.com/
https://blog.cloudflare.com/validating-leaked-passwords-with-k-anonymity/

Let’s exercise your privacy brain



Would you send your password in the clear?

Would you send a hash?

What they do: send the first 5 bytes of the hash of the password and receive a list of 475 suffixes to check offline

From the point of view of the server (that receives the 5-bytes suffix)

What is the privacy of the password?
The password is 475-anonymous!

Send **Receive**

1. (21BD1) 0018A45C4D1DEF81644B54AB7F969B88D65:1 (password "lauragpe")
2. (21BD1) 00D4F6E8FA6EECAD2A3AA415EEC418D38EC:2 (password "alexguo029")
3. (21BD1) 011053FD0102E94D6AE2F8B83D76FAF94F6:1 (password "BDnd9102")
4. (21BD1) 012A7CA357541F0AC487871FEEC1891C49C:2 (password "melobie")
5. (21BD1) 0136E006E24E7D15213981 5FB0FC6A50B15:2 (password "quvekyny")
6. …

https://haveibeenpwned.com/
https://blog.cloudflare.com/validating-leaked-passwords-with-k-anonymity/

Privacy in social networks

As an example for sanitization…

Structural De-anonymization in Social Networks

- Privacy Properties
  - Social network = nodes, edges (relationships between nodes), and information associated with each node and each edge
  - Information about nodes obviously wants to satisfy a level of privacy
  - Most social networks make relationships between nodes public by default (few users change)

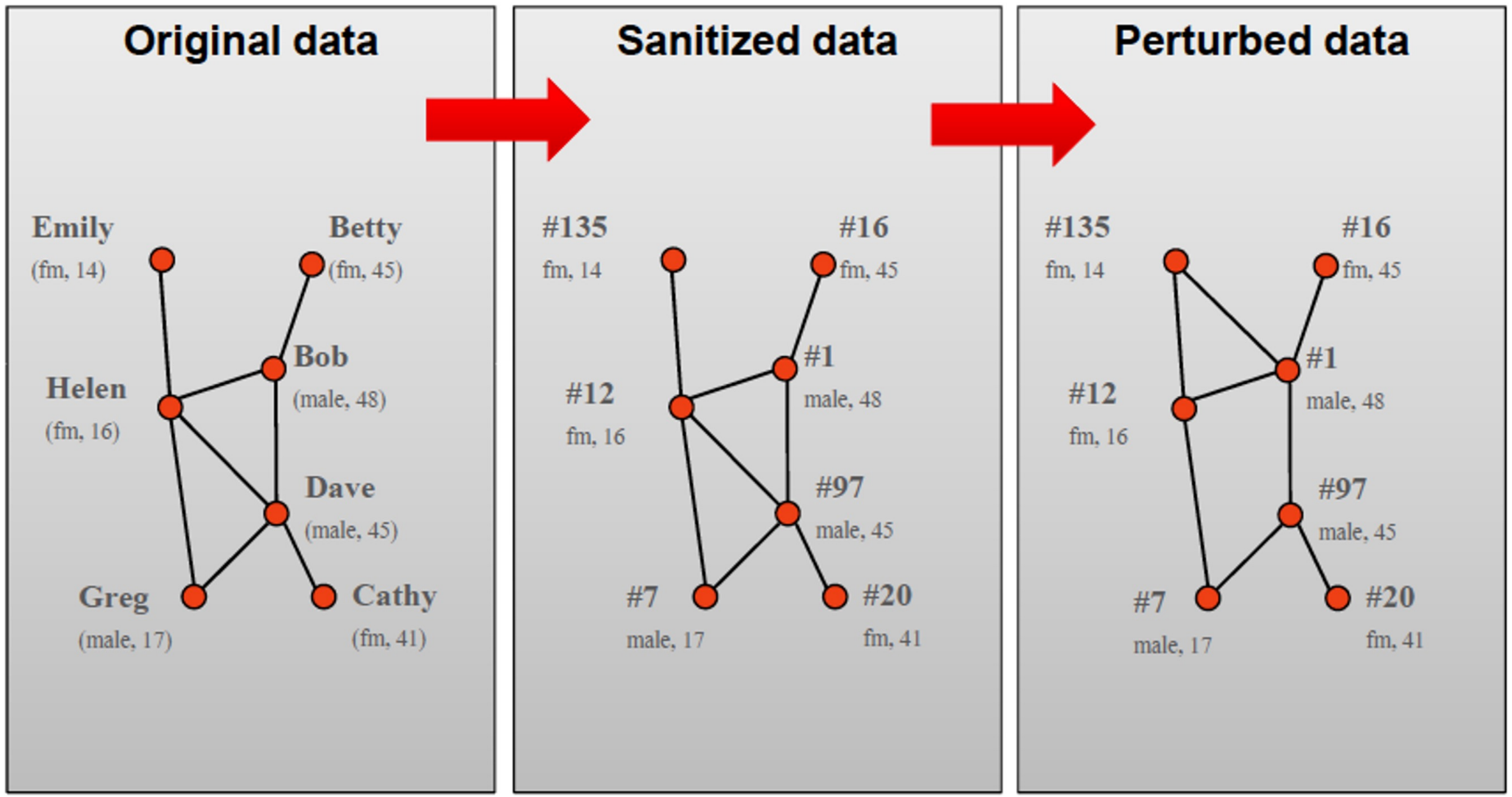
Model – Social Network

- Let us define a social network *S* consists of
  1. A directed graph G = (V,E)
  2. A set of attributes X for each node in V and a set of attributes Y for each edge in E

Attributes for nodes: (i.e. name, telephone #)

Attributes for edges: (i.e. type of relationship)

Graph Sanitization and Perturbation



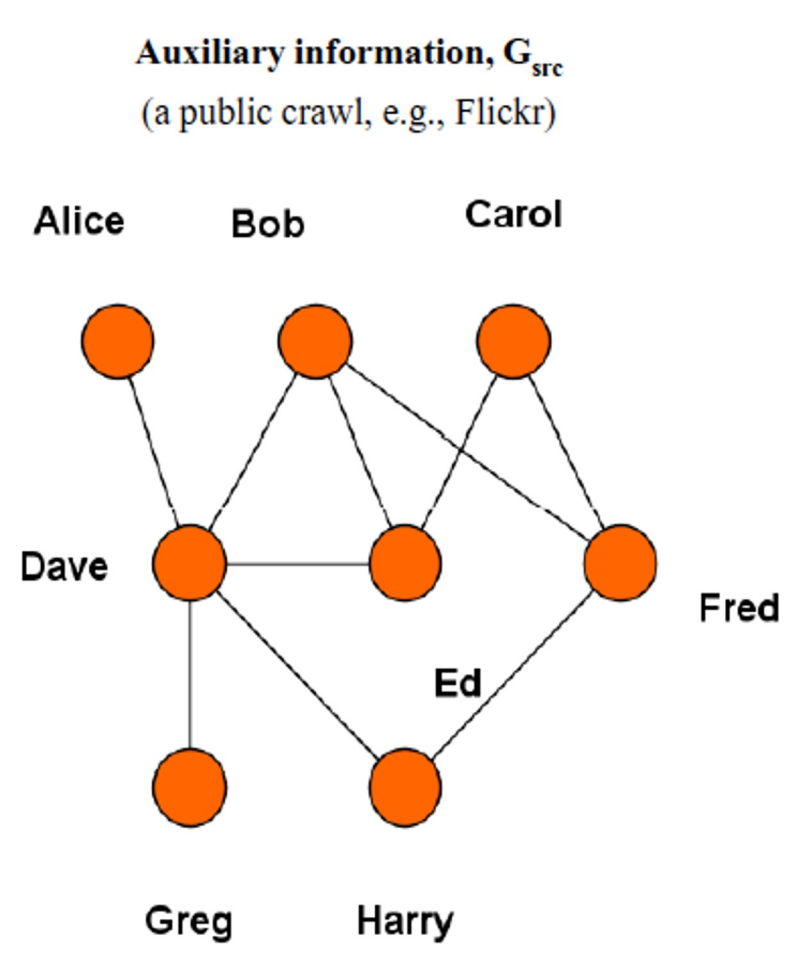
Attacker Model

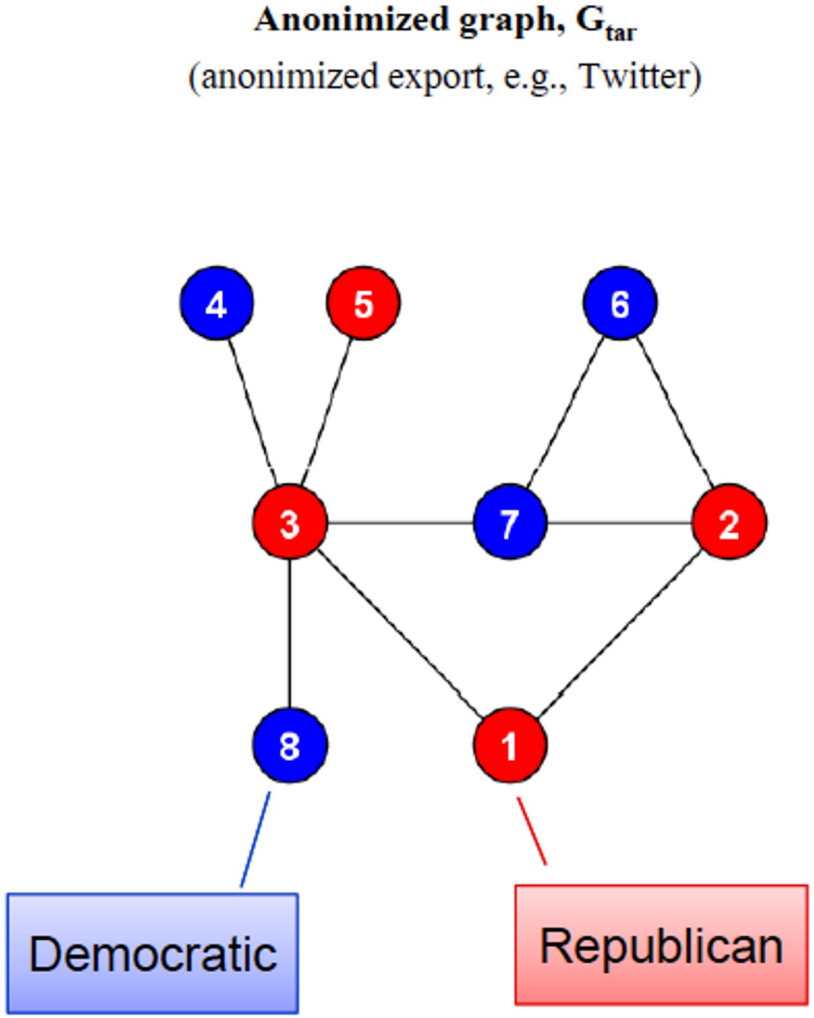
- Assume an attacker has access to an anonymized, sanitized, target network S_{SAN} and also access to a different network S_{AUX} whose members partially overlap with S_{SAN}
- This is a very real and plausible assumption
- Facebook -> Myspace or Twitter -> Flickr
- Even with an extensive auxiliary network S_{AUX} , de-anonymizing the target network S_{SAN} is difficult

Auxiliary Information

- Auxiliary information is global in nature
  - Many social networking sites overlap one another
  - Facebook, Myspace, Twitter, etc. (correlate)
- Can be used for large-scale re-identification
- Feedback based attack
  - Re-identification of some nodes provides the attacker with even more auxiliary information

Example


Auxiliary information, G_{src}
(a public crawl, e.g., Flickr)
Alice
Bob
Carol
Dave
Fred
Ed
Greg
Harry


Anonimized graph, G_{tar}
(anonimized export, e.g., Twitter)
4
5
6
3
7
2
8
1
Democratic
Republican

De-anonymization

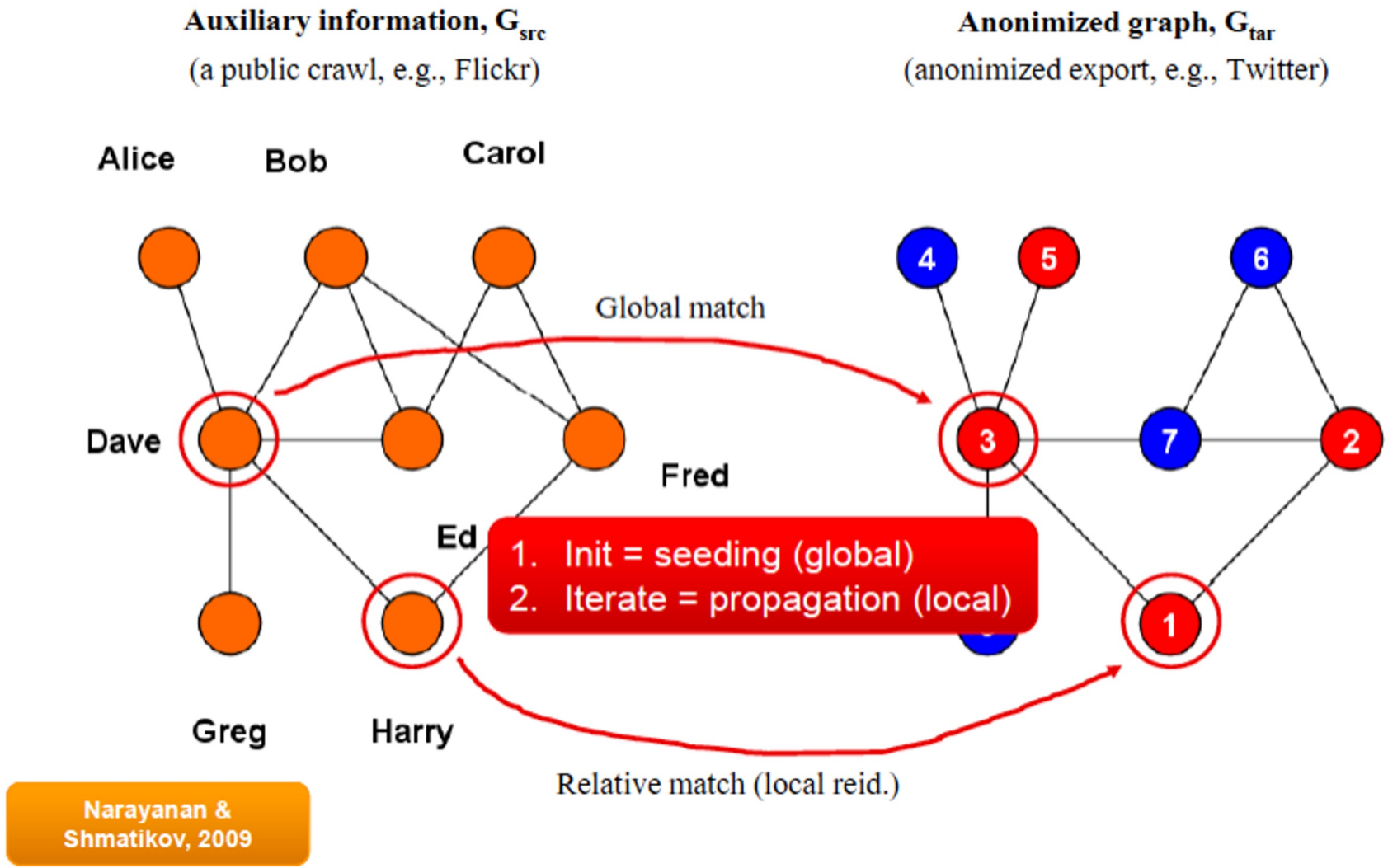
- Two Stages

1. Seed Identification
   - attacker identifies a small group of “seed” nodes which are present in both the anonymous target graph and the attacker’s auxiliary graph, and maps them to each other

2. Propogation
   - a self-reinforcing process in which the seed mapping is extended to new nodes using only the topology of the network, and the new mapping is fed back to the algorithm.

- Result is a huge mapping between subgraphs of the auxiliary and target networks which re-identifies (de-anonymizes) those mapped nodes.

De-anonymization



Limitations of Structural De-anonymization

Some platforms do not have a graphical structure at all

- forums

Graphical structure of some OSNs does not resemble the real-life connections

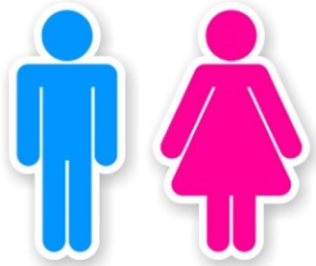
- patientslikeme.com -> people share sensitive information about their health conditions, diseases, diagnosis, and the drugs they use, users usually don't follow their friends but follow similar conditioned people

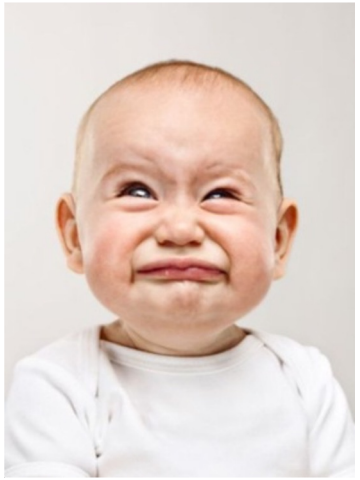
most users do not share their real identity

Unstructured OSNs

What is the background information for de-anonymization?

- Location
- Profile photo
- Gender
- Free text
- Account activity patterns
- User name
- Interdependent information
- Domain specific information





Need tools to quantify and show the risk of the profile matching attack in unstructured OSNs

Need efficient techniques to process this information

How about countermeasures?

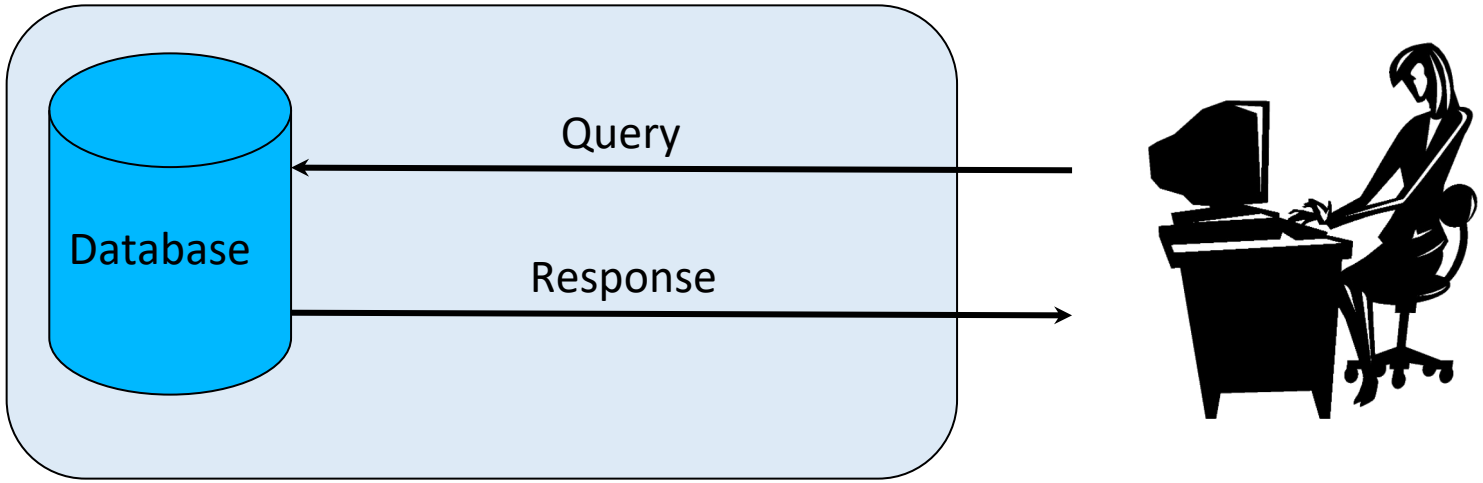
Privacy Mechanisms for Databases: The interactive scenario

The interactive scenario

Many times we do not want the data, we want statistics!

Redefined Goal for the interactive case:

Produce an **answer** that **preserves the utility** of the **statistics** **without leaking information** about individuals.



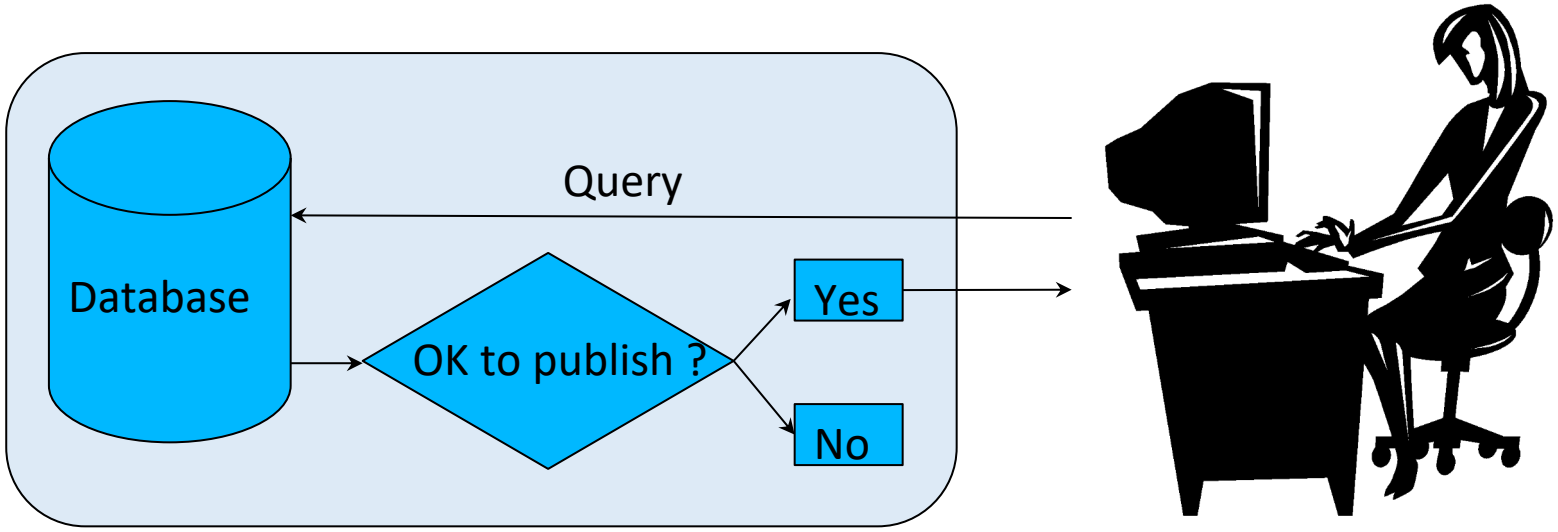
Methods to Release Statistics

- Large query sets
  - Disallows queries about a specific individual or small set of individuals
  - But, how about the below queries?
    - “How many people in the database have the sickle cell trait?”
    - “How many people, not named X, in the database have the sickle cell trait?”

Name	Sickle cell trait
A	Yes
B	Yes
C	No
D	No
X	No
Y	Yes
Z	No

The interactive scenario

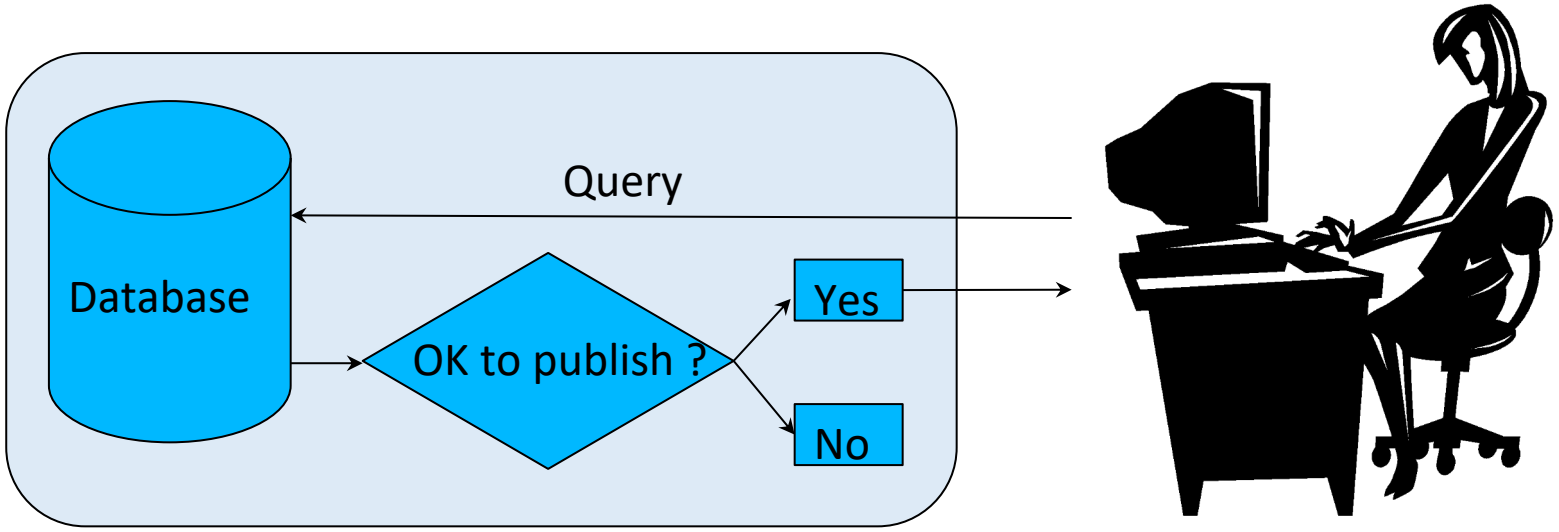
Let’s audit the queries, if the query will leak, deny!
Either answer truthfully or state that there will be no answer



Database assumed to contain *numeric* values.

The interactive scenario

Let’s audit the queries, if the query will leak, deny!
Either answer truthfully or state that there will be no answer

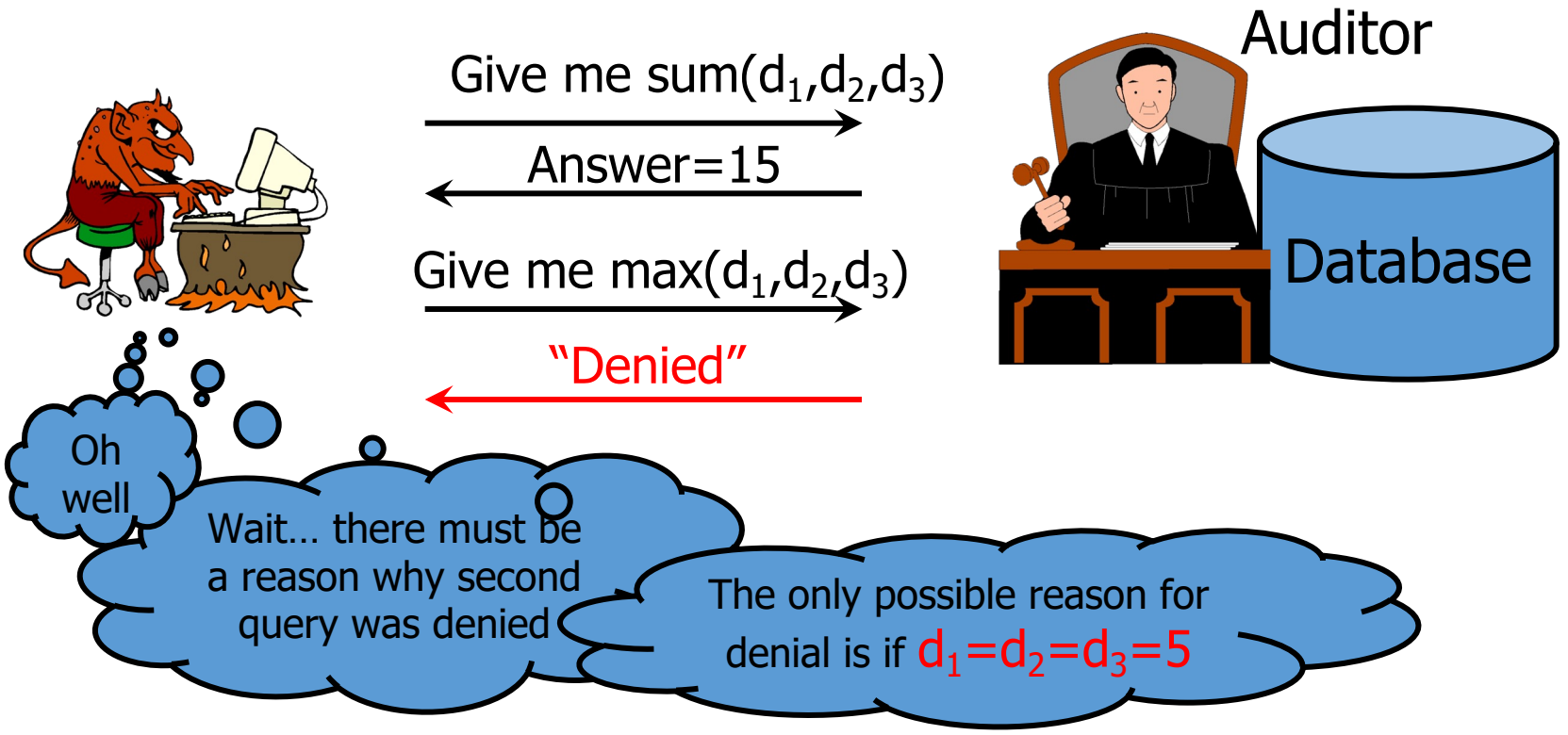


Database assumed to contain *numeric* values.

! Not answering already reveals some information !

When denying fails: learning exact values

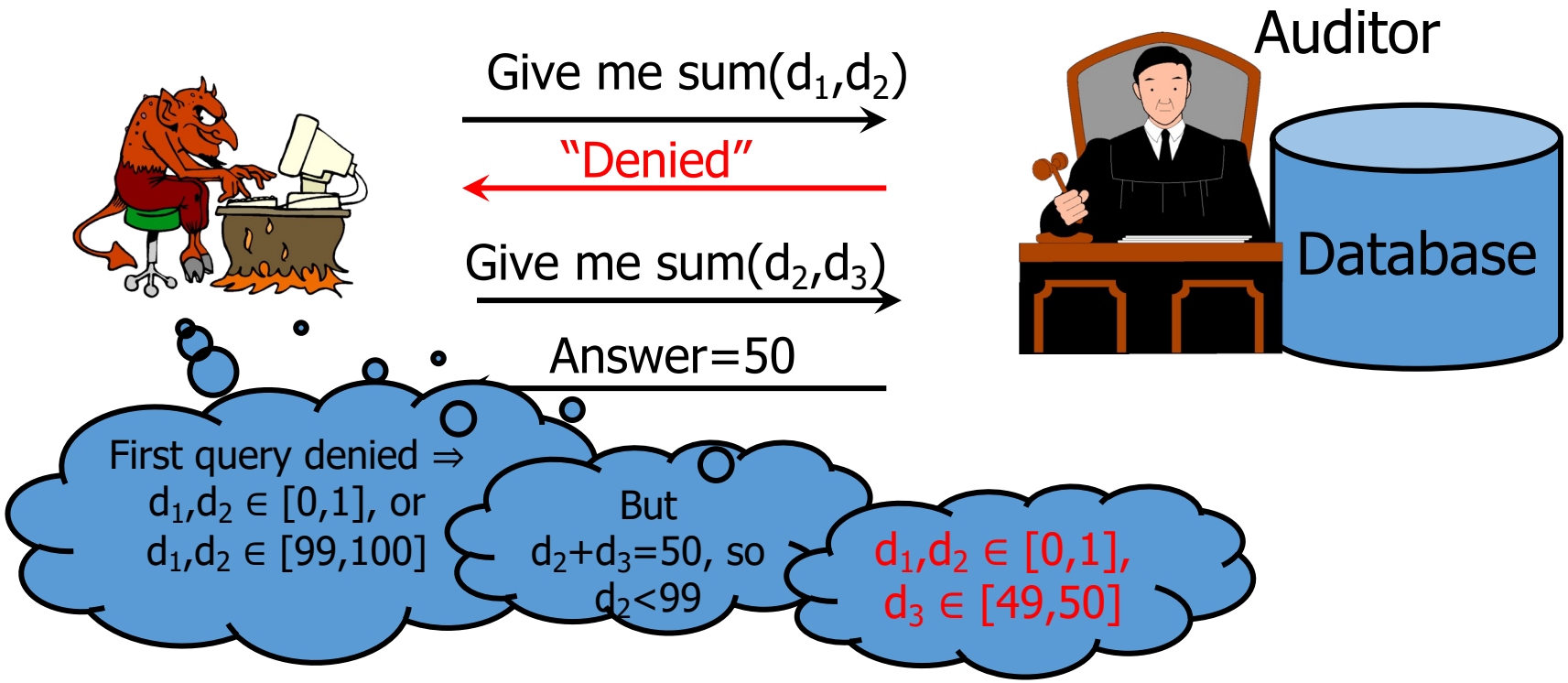
- Variables d_i are real, privacy breached if adversary learns some d_i



When denying fails: learning intervals

- Assume privacy as

$d_i \in [0,100]$, privacy breached if adversary learns some $d_i \pm 1$



Can I make sure that the next query does not leak?

Query auditing

- Keeps the query history to determine if a response would be disclosive
- Computationally infeasible
- Refusal to respond to a query may itself be disclosive

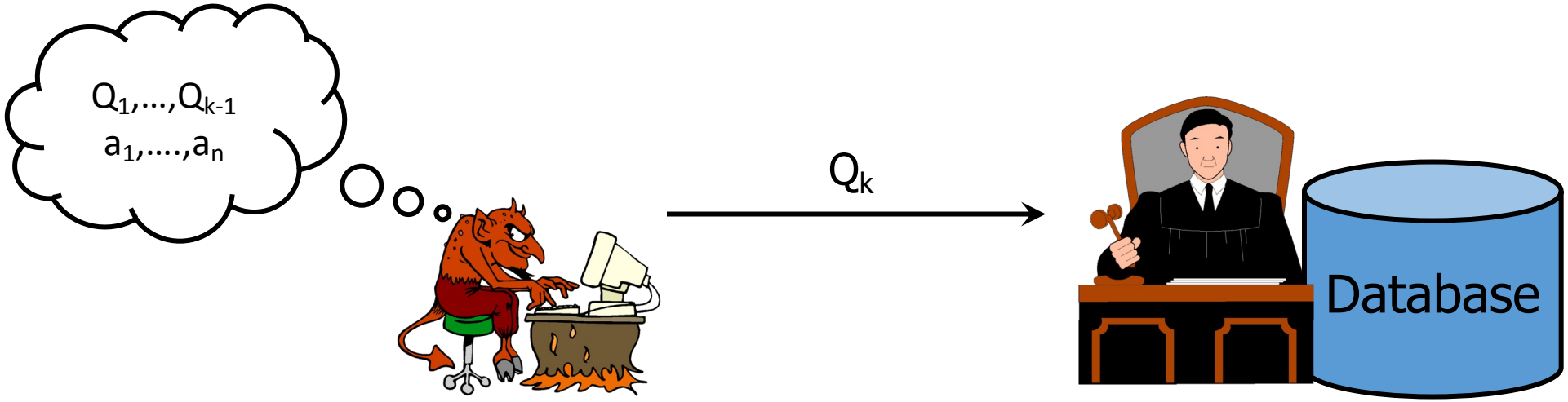
- Example:
  - Max sensitive value of males?
    => 2
  - Max sensitive value of 1st year PhD students?
    =>3
  - Xi: only female 1st year PhD student
    - Sensitive value of Xi: 3

Name	1st year PhD	Gender	Sensitive value
Ben	Y	M	1
Bha	N	M	1
Ios	Y	M	1
Jan	N	M	2
Jian	Y	M	2
Jie	N	M	1
Joe	N	M	2
Moh	N	M	1
Son	N	F	1
Xi	Y	F	3
Yao	N	M	2

Can I make sure that the next query does not leak?

Simulatable auditing

One cannot learn anything from the denial **if the decision to deny or give an answer is independent of the actual data set and the real answer**.

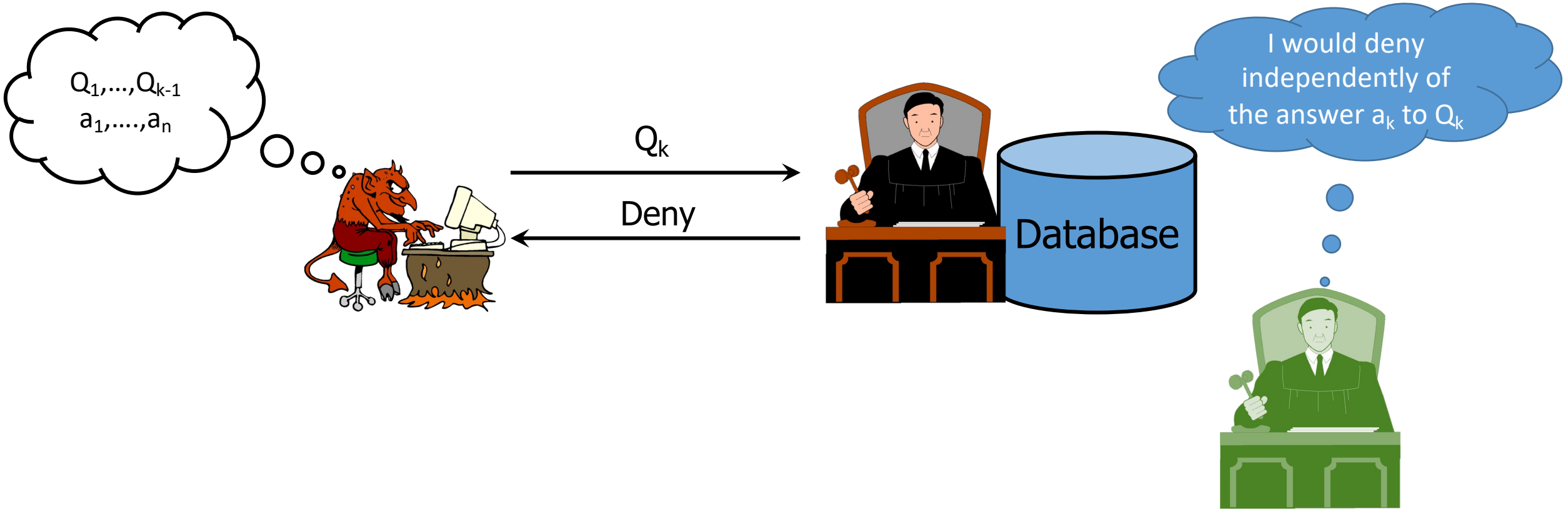


https://theory.stanford.edu/~nmishra/SimulatableAuditing.pdf

Can I make sure that the next query does not leak?

Simulatable auditing

One cannot learn anything from the denial **if the decision to deny or give an answer is independent of the actual data set and the real answer**.

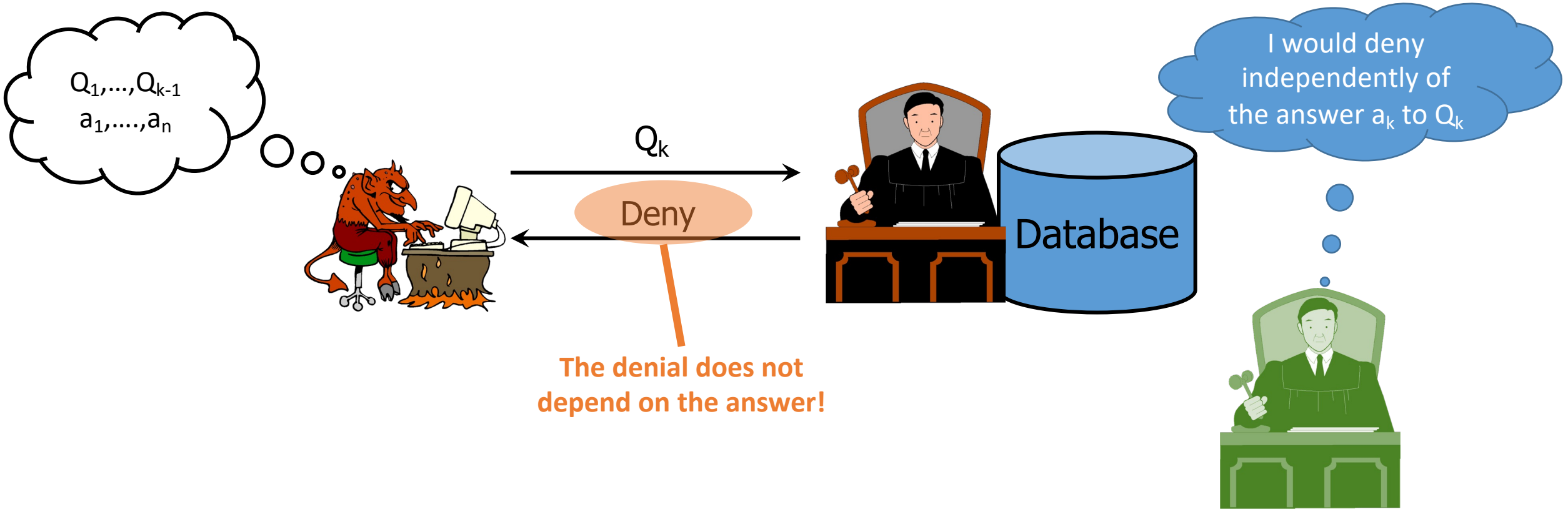


https://theory.stanford.edu/~nmishra/SimulatableAuditing.pdf

Can I make sure that the next query does not leak?

Simulatable auditing

One cannot learn anything from the denial **if the decision to deny or give an answer is independent of the actual data set and the real answer**.



https://theory.stanford.edu/~nmishra/SimulatableAuditing.pdf

Auditing has problems

- Privacy definition? Privacy of Values? Groups? Exact?
- Algorithmic limitations
  - Secure deniability implies using algorithms computationally prohibitive
  - Feasible focus mostly on simple queries
- Collusion? Either high cost or no security
- Utility?
  - Percentage of denials may not be the best measure

What else can we do? Modifying inputs

- **Subsampling**
  - A subset of the rows is chosen at random and released and **statistics are computed on the subsample**
  - Uneven privacy for users, being in a subsample may have unfortunate consequences

    Not being may too!
- **Input perturbation**
  - **Data or queries are modified before** a response is generated
  - How can we quantify the leakage?
  - How to balance for utility?

What else can we do? Modifying outputs

- **Adding random noise to the output**

  Would this work?

What else can we do? Modifying outputs

- **Adding random noise to the output**
  - **Naively**, this approach will fail
    - E.g., if the same query is asked repeatedly, then the responses can be averaged, and the true answer will eventually emerge.
  - This cannot be fixed by recording each query and providing the same response each time a query is re-issued.
    - **Syntactically different queries may be semantically equivalent**, and, if the query language is sufficiently rich, then the equivalence problem itself is undecidable.

What else can we do? Modifying outputs

- **Adding random noise to the output**
  - **Naively**, this approach will fail
    - E.g., if the same query is asked repeatedly, then the responses can be averaged, and the true answer will eventually emerge.
  - This cannot be fixed by recording each query and providing the same response each time a query is re-issued.
    - **Syntactically different queries may be semantically equivalent**, and, if the query language is sufficiently rich, then the equivalence problem itself is undecidable.
- **Randomized response**
  - Respondents to a query **flip a coin and, based on the outcome, they either honestly respond or respond randomly**
  - Privacy comes from the uncertainty of how to interpret a reported individual value.
  - Yet, data can be useful because **randomness can be averaged out**
  - **Not usable for every case, or combined with other techniques**

Differential privacy

Remember the Goal for the interactive case:

Produce an **answer** that preserves the utility of the statistics without leaking information about individuals.

To have any utility we must allow the leakage of some information, but we can set a bound on the extent of leakage!

Differential Privacy:

Output is similar whether any single individual's record is included in the database or not.

Guarantees minimal similarity

Cynthia Dwork. Differential privacy: A survey of results.
International conference on theory and applications of models of computation, April 2008.

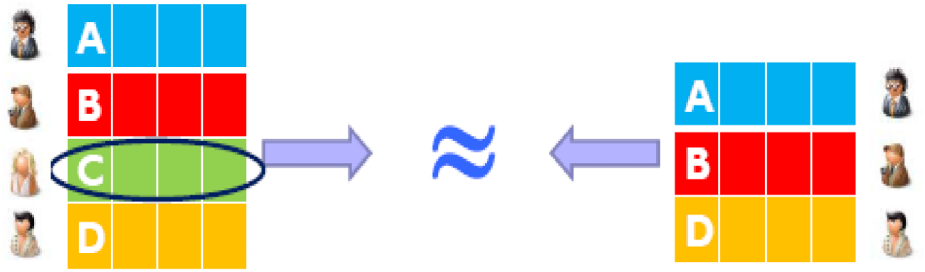
More on Differential Privacy

- **Basic philosophy:** instead of the real answer to a query, output a random answer, such that by a small change in the database (someone joins or leaves), the distribution of the answer does not change much.
- **A new privacy goal:** minimize the increased risk incurred by an individual when joining (or leaving) a given database.
- **Motivation:** A privacy guarantee that limits risk incurred by joining, therefore encourages participation in the dataset, increasing social utility.

- Differential Privacy is a privacy notion **NOT** a mechanism
  We use mechanisms to *achieve* differential privacy

Differential Privacy - Informal Definition

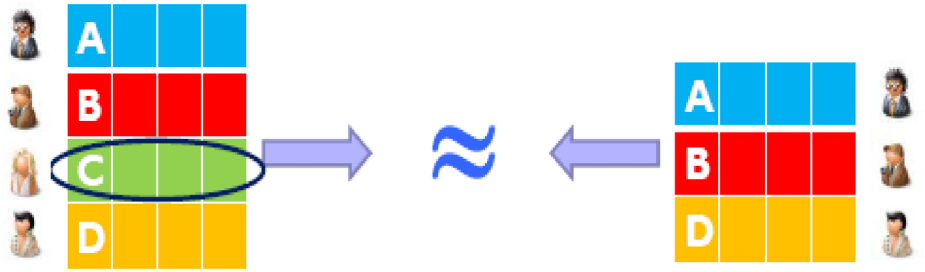
Output is similar whether any single individual's record is included in the database or not.



C's inclusion of her record in the computation does not make her *significantly* worse off.

Differential Privacy - Informal Definition

Output is similar whether any single individual's record is included in the database or not.



C's inclusion of her record in the computation does not make her *significantly* worse off.

If there is already some risk of revealing a secret of C by combining auxiliary information and something learned from DB, then **that risk is still there** but not *significantly* increased by C's participation in the database.

ϵ-Differential Privacy – Formal Definition

- $\mathcal{D}$: The set of input databases
- R: Output space of the query
- F: Query function

  $F: \mathcal{D} \rightarrow R$
- d: Distance function on the set of databases
- *Neighboring databases*: Pairs of databases ($\mathcal{D}$, $\mathcal{D}_{-r}$) differing only in one row *r* (e.g., individual)

  $d(\mathcal{D}\text{-}\mathcal{D}_{-r}) = 1$

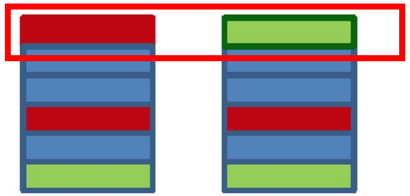
ϵ-Differential Privacy – Formal Definition

- Principle
  - The **removal/addition** of a **single record** in the **database should/does *not* substantially affect the values** of the computed function/statistics.
- Formalization
  - Let A be the **randomized function** (namely a **mechanism**) to be computed on a set of records.
    - A is the actual function to be computed f + **noise**.
  - Let S be a subset of the possible values taken by A.
  - A provides ϵ-differential privacy if for all r, S:

$$\mathrm{P}[A(D) \in S] \leq e^{\epsilon} \times \mathrm{P}[A(D_{-r}) \in S]$$

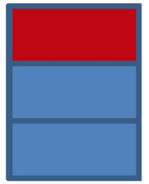
Differential Privacy - Intuition

For every pair of inputs that differ in one value



D_1 D_2

For every output …

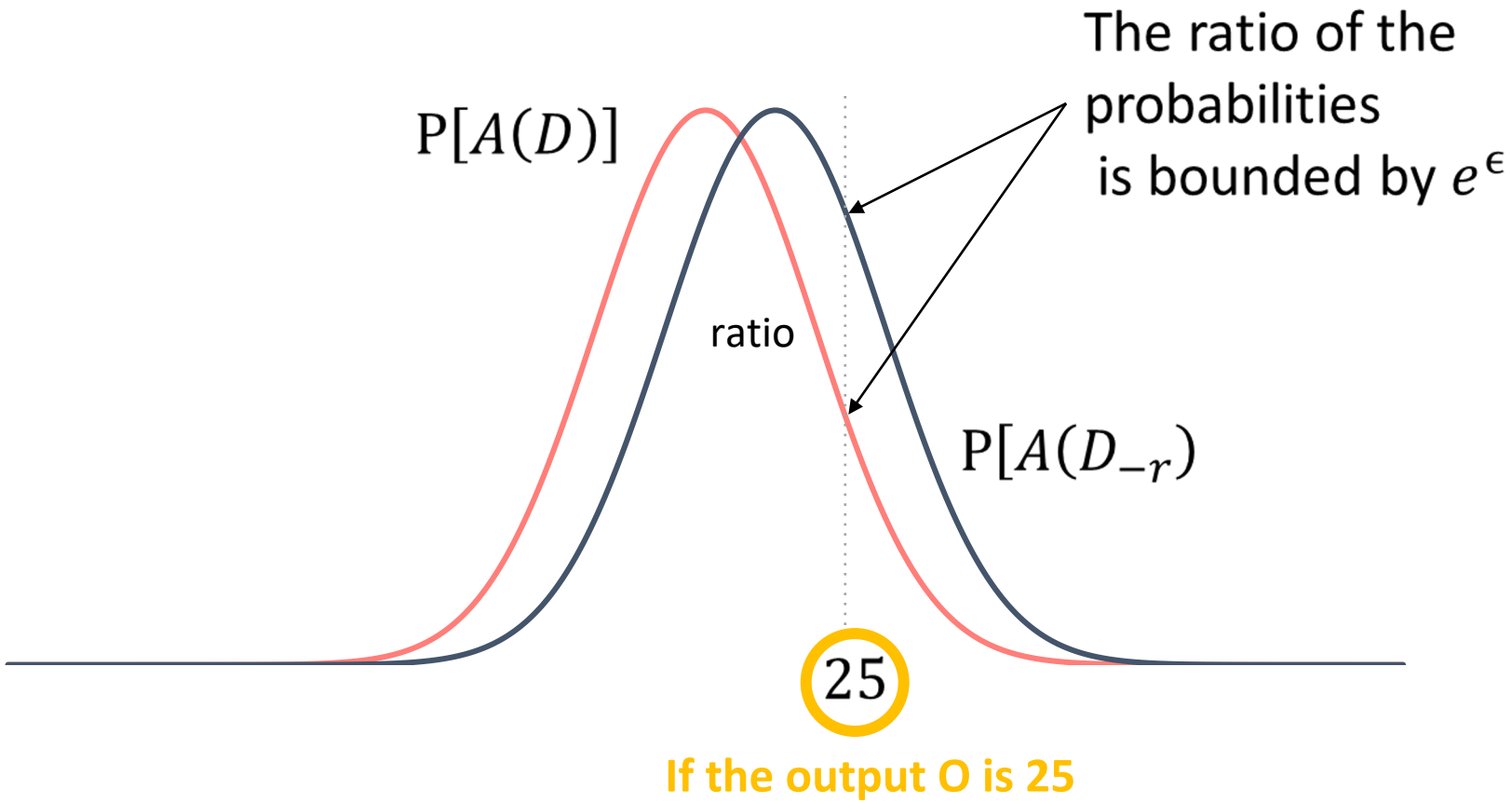


O

Adversary should not be able to distinguish between any D_1 and D_2 based on any O

$$\log\left|\frac{\Pr[A(D_1) = O]}{\Pr[A(D_2) = O]}\right| < \varepsilon \qquad (\varepsilon>0)$$

ϵ-Differential Privacy



How to achieve ϵ-Differential Privacy (simple case)

- How to achieve ϵ-differential privacy (simple case)?

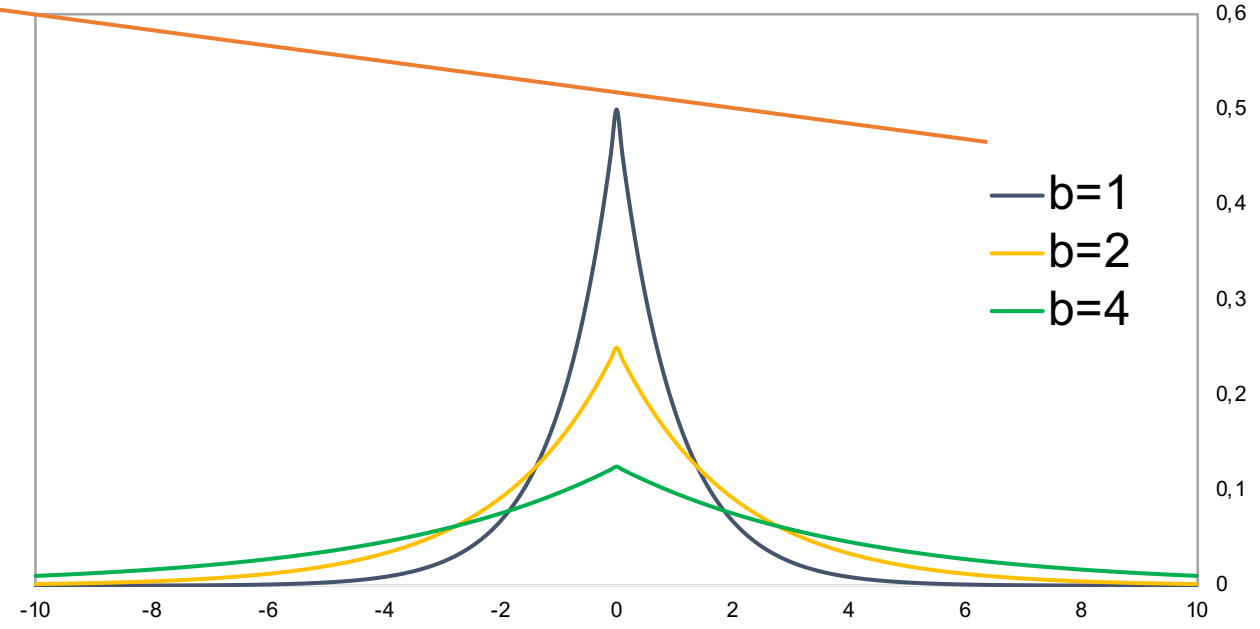
  Assume f is a scalar function, i.e., $f: \mathcal{D} \rightarrow \mathbb{R}$ (e.g., "number of records with cancer"?)

  Return $\boldsymbol{A(D) = f(D) + \mathbf{Lap}\left(\frac{\Delta f}{\epsilon}\right)}$

  $\mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right)$ is **noise** drawn from a **Laplacian** distribution of parameter $\frac{\Delta f}{\epsilon}$

  Δf is the **sensitivity** of function f:

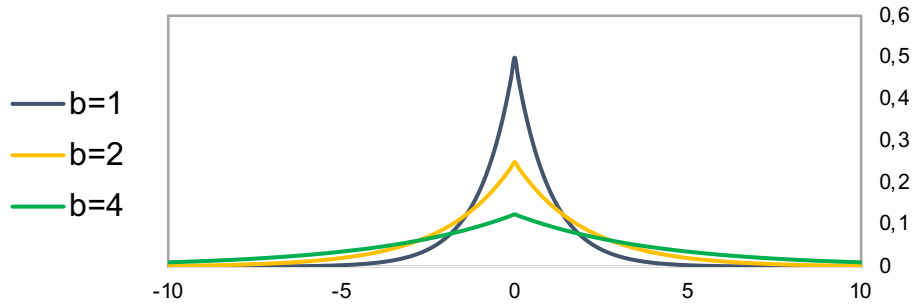
  $$\Delta f = \max_r |f(D) - f(D_{-r})|$$



Why Laplacian Distribution?

- The Laplacian distribution is: $\mathrm{Lap}\left(\frac{\Delta f}{\epsilon}\right) = \frac{\epsilon}{2\Delta f} \exp(-\frac{x\epsilon}{\Delta f})$.
- The distortion of the result depends on both the sensitivity and privacy guarantee:
  - The higher the sensitivity, the higher the distortion
  - The higher the privacy guarantee (the lower ϵ), the higher the distortion

- This distribution has highest density at 0 (good for accuracy).
- This distribution is symmetric about 0 and has a heavy tail.

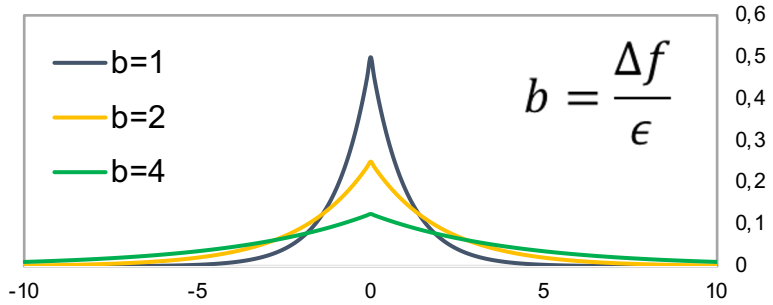


How to choose the parameters ?

Selecting ε

- The parameter ε is public (remember: no security by obscurity
- Selection of ε by Cynthia Dwork:
  - *"We tend to think of ε as 0.01, 0.1, or in some cases, ln2 or ln3"*
  - Smaller ε means better privacy
  - But, what about the utility ?

Sensitivity of a Query $\Delta f = \max_r |f(D) - f(D_{-r})|$



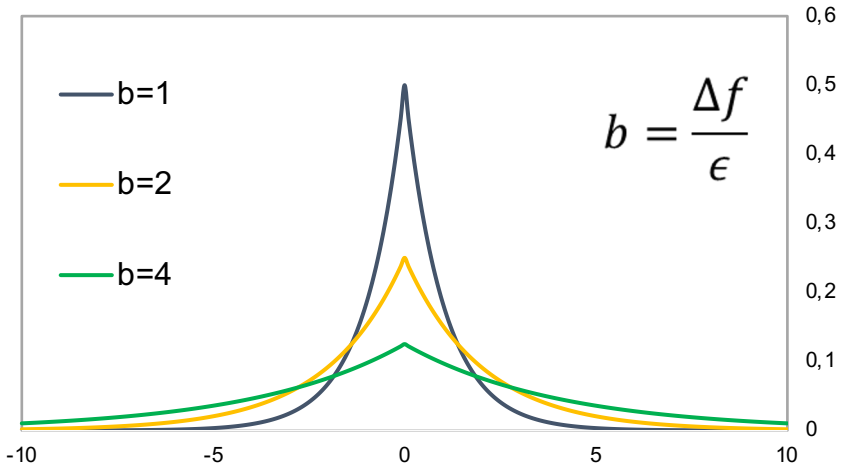
How to choose the parameters ?

Selecting ε

- The parameter ε is public (remember: no security by obscurity
- Selection of ε by Cynthia Dwork:
  - *"We tend to think of ε as 0.01, 0.1, or in some cases, ln2 or ln3"*
  - Smaller ε means better privacy
  - But, what about the utility ?

It depends on the sensitivity!

$$\Delta f = \max_{r} |f(D) - f(D_{-r})|$$



What is the sensitivity of… ?

For any two neighboring databases (D, D_{-r}):

$$\Delta f = \max_{D, D_{\pm r}} || F(D) - F(D_{-r}) ||$$

Sensitivity of counting queries:

- The number of elements in the database with a given property *P.*

Sensitivity of histogram queries:

- Suppose each entry in d takes values in $\{c_1, c_2, \ldots, c_n\}$.
- *Histogram*$(d) = \{m_1, m_2, \ldots, m_n\}$, m_i = (#entries in d with value c_i)

What is the sensitivity of… ?

For any two neighboring databases (D, D_{-r}):

$$\Delta f = \max_{D, D_{\pm r}} || F(D) - F(D_{-r}) ||$$

Sensitivity of counting queries:

- The number of elements in the database with a given property *P.*
- By adding or deleting one element of the database, *F* can change by at most 1.
- **Δf(counting) = 1**

Sensitivity of histogram queries:

- Suppose each entry in d takes values in $\{c_1, c_2, \ldots, c_n\}$.
- *Histogram*(d) = $\{m_1, m_2, \ldots, m_n\}$, m_i = (#entries in d with value c_i)
- **If one element moves from one entry to another: Δf(histogram) = 2**

Composability of Differential Privacy

Theorem: If algorithms $F_1, F_2, \ldots, F_k$ use independent randomness and each F_i satisfies ε_i-differential privacy, respectively. Then outputting all the answers together satisfies differential privacy with

$$\varepsilon = \varepsilon_1 + \varepsilon_2 + \ldots + \varepsilon_k$$

Does privacy increase or decrease?

How to ensure differential privacy ?

- **Input perturbation**
  - Add noise directly to the database (≠ perturbed dataset can be published)
    + independent of the algorithm & easy to reproduce
    - determining the amount of required noise is difficult

- **Output perturbation**
  - Add noise to the function (statistic) output
    + easier to control privacy & better guarantees than input perturbation
    - results cannot be reproduced

- **Algorithm Perturbation**
  - Inherently add noise to the algo
    + algorithm can be optimized with the noise addition
    - difficult to generalize & depends on the inputs

Why is DP possible (while anonymization was impossible):

The final result depends on multiple personal records

However it does not depend much on any particular one (sensitivity)

Therefore adding a little bit of noise to the result, suffices to hide any record contribution

For full anonymization.... one would need to add a lot of noise to all the entries

Interactive!

But… the architecture is different: **one Trusted-Third-Party holds the data (Interactive)**

Also… after some uses utility drops

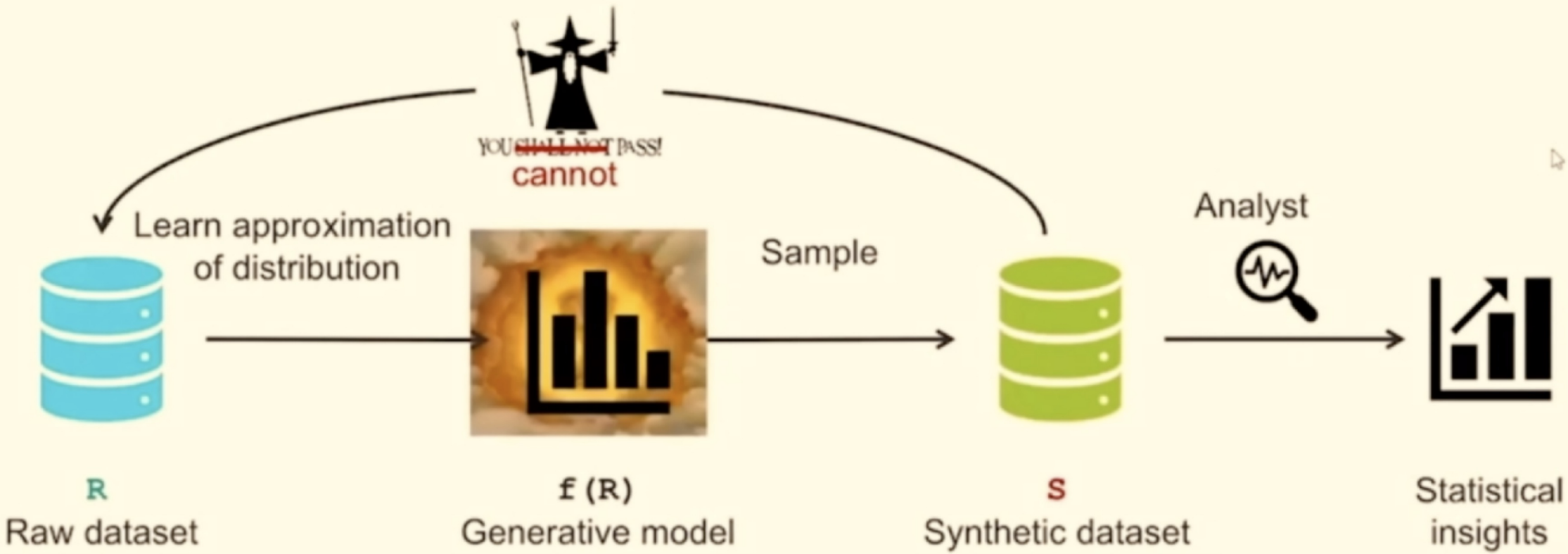
best use: one-time **data collection**

Google RAPPOR -> Collect data from phones
Apple -> Collect data from phones
Federated learning -> Share models
Smart energy -> Collect measurements

LIVE 46

EPFL

The promise of synthetic data



Takeaway: Privacy Definitions

- There are many privacy definitions
  - k-anonymity, l-diversity, t-closeness, Differential privacy

- Why can't we have a single definition for privacy?
- No Free Lunch Theorem [1]:
  - For every algorithm that outputs a D with even a sliver of utility, there is some adversary with a prior such that privacy is not guaranteed

[1] "No Free Lunch in Data Privacy", Kifer and Machanavajjhala, 2011