

# Privacy Compliant Data Release

## Differential Privacy

Jorge Cuellar


WS 18-19

# What to learn

## Main focus of the course (WS 18-19)

- ▶ Cryptography methods (Slide sets Crypto-PETs-1, -2, -3)
- ▶ Anonymizing databases, and then De-anonymizing them
  - ▶ and Differential Privacy

# Contents (Differential-privacy)

- ▶ How "anonymized" databases can be linked together and "deanonymized"
  - ▶ AOL Search debacle
  - ▶ Netflix
  - ▶ Massachusetts GIC medical DB
- ▶ Types of fields in privacy-sensitive DBs
  - ▶ identifiers
  - ▶ quasi-identifiers and
  - ▶ sensitive data
- ▶ k-anonymity and variants
  - ▶  $l$ -diversity
  - ▶   $t$ -closenes

# Contents (Differential-privacy), contd

- ▶ Query-response model for DiffPriv
- ▶ Differential Privacy
  - ▶ Definition
  - ▶ The Laplace-Noise Method
  - ▶ Properties and problems

# Digitalization and Privacy

## Data protection and privacy authorities

- ▶ ... are closely monitoring the developments in Digitalization, Big Data, IoT
- ▶ Such data, high in quantity and quality
  - ▶ allows the inference of personal information and
    - ▶ identifiability becomes possible

## "IoT data should be regarded and treated as personal data"

- ▶ Conf of DP & Privacy Commissioners

# Semantic Security for Crypto

Anything that can be learned from a ciphertext

- ▶ can be learned without the ciphertext
  - ▶ (Learned = deduced by a polynomial algorithm)

# Does a ciphertext contain information about the cleartext?

- ▶ For a One-Time-Pad: No, it does not contain info
  - ▶ This is the "perfect security" of OTP
- ▶ In general, yes: the ciphertext contains info about the clear text
  - ▶ but (if the encryption is non-deterministic, etc)
    - ▶ this information can almost never be used by a polynomial algorithm

# Recall: Indistinguishability

Indistinguishability is a very basic concept in **security**

- ▶ Differential Privacy is something similar
  - ▶ but for Privacy

## Recall Indistinguishability:

An attacker does not know

- ▶ which one of two possibilities is right





# "Semantic Security"

## Game Semantics for Non-deterministic encryption

- ▶ Assume, for simplicity, a *defender* shows a ciphertext and
  - ▶ offers the *attacker* the choice
    - ▶ "Is this the encryption of  $m_0$  or  $m_1$ ?"
    - ▶ the attacker has to guess correctly
- ▶ Following Definition is **too strong**: A system is *secure*
  - ▶ if the attacker can never win the game any better than
    - ▶ an attacker that does not see the ciphertext (and guesses randomly)
- ▶ Correct Definition (**semantic security**): A system is *secure*
  - ▶ if the attacker has only a negligible advantage
    - ▶ over an attacker without seeing the ciphertext

⇒ alternatives  $m_0, m_1$  are *undistinguishable* for the attacker

# Semantic Security

- ▶ There are several variants
  - ▶ depending on assumptions on capabilities of the attacker
- ▶ This is normally presented as a game:

## A cryptosystem is secure

- ▶ if no attacker can "win the game"
  - ▶ with significantly greater probability
  - ▶ than an attacker who must guess randomly

# Semantic Security

## A cryptosystem is **secure**

- ▶ If the attacker who sees the encrypted data
  - ▶ has only a negligible advantage over
  - ▶ an attacker that sees nothing,
    - ▶ that is, one that is randomly guessing
- ▶ He wins the following game with probability  $< 0.5 + \varepsilon(\ell)$ 
  - ▶ where  $\varepsilon$  is a negligible function of  $\ell$

# Semantic Security

1. The defender generates a key pair  $PK, SK$  of key size  $\ell$ 
  - ▶ publishes  $PK$
2. The attacker performs a number of encryptions
3. polynomially bounded
4. The attacker chooses 2 plaintexts  $m_0, m_1$
5. The defender selects one of them at random
  - ▶ and presents the ciphertext  $c = \mathcal{E}(PK, m_i)$  to the attacker
6. The attacker wins if he is able to guess  $m_0$  or  $m_1$



# Information as a change in probability

## "Event $F$ has no information about event $E$ "

- ▶ means: if I know whether  $F$  happens or not
  - ▶ this tells me **nothing** about
    - ▶ whether  $E$  happens or not
- ▶ More precisely,
  - ▶ the probability that event  $E$  happens
  - ▶ does not change, adding the information  $F$ :

$$\text{Prob}[ E ] = \text{Prob}[ E | F ]$$

- ▶ Note that " $F$  has no information about  $E$ "
  - ▶ is the same as  $F$  and  $E$  are independent
    - ▶  $\Rightarrow E$  has no information about  $F$



# Information as a change in probability

## Be careful: Look at the context

- ▶ Even if " $F$  has no information about  $E$ "
  - ▶ There still will probably be some "a-priori" information  $I$
  - ▶ Or – in other words – some context or situation
    - ▶ (and in this context we gain some "a-priori knowledge")
- ▶ And  $F$  **has information** about  $E$  **under the information  $I$**

$$\text{Prob}[ E \mid I ] \neq \text{Prob}[ E \mid I \wedge F ]$$



# Information as a change in probability

## Publishing global statistics

- ▶ Assume you want to publish the result:
  - ▶  $F$  = "smoking produces cancer"
    - ▶ (... and assume that nobody knew that)
    - ▶ (Or, assume your research shows "eating green bananas produces cancer")
- ▶ Does this information tell anything about
  - ▶ the chances that  $E$  = "Peter Pan has cancer"?
- ▶ No, if you do not know whether Peter Pan smokes

## But in a context

- ▶ where you know that he smokes
  - ▶  $F$  has information about  $E$



# There is a tension between . . .

## Utility

- ▶ Accurate, usable statistical info is released

## Privacy

- ▶ Each individual's sensitive info remains hidden
- ▶ Is there a method for obfuscating a DB
  - ▶ or responses to DB queries, s.t.
    - ▶ responses are useful
    - ▶ responses do not release private information?
- ▶ Can you use Big Data?
  - ▶ But **make sure that no conclusions can be drawn**
    - ▶ **for any particular individual?**



## This is what Differential Privacy attempts to solve

- ▶ ... But, with the above formulation
  - ▶ This is **impossible**
- ▶ ... making privacy very difficult
  - ▶ If you disclose some very innocuous information  $F$ 
    - ▶ that you think is not privacy-relevant
  - ▶ Still, under some unexpected context (I)
    - ▶ the information disclosed will release information
    - ▶ about some fact  $E$  which is clearly personal

# Disclosure Prevention

## Def (Dalenius, 1977):

Anything that can be learned about a respondent

- ▶ from the statistical database
- ▶ can be learned without access to the database

## Impossibility Result

It is impossible to design any (non-trivial) mechanism

- ▶ that satisfies such strong notion of privacy
  - ▶ (A trivial mechanism is to disclose already known information)

# Auxiliary Information

Common theme in privacy violations:

- ▶ Existence of side information
  - ▶ Netflix challenge: IMDB
  - ▶ Massachusetts GIC medical DB: Voter Registration List
  - ▶ AOL Search: (lots of info)

# Disclosure Minimization

See slides "intro.pdf", Privacy Principles

- ▶ One of the principles (not discussed in those slides) is

## Disclosure Minimization

For any primary or secondary purpose

- ▶ if it is legitimate, say for research

the disclosure of personal data

- ▶ to third parties
- ▶ or to the public
  - ▶ must be minimal
    - ▶ as far as existing technical PETs permit

# Disclosure Minimization

In other words

- ▶ if for a certain purpose two or more alternatives exist
- ▶ and those alternatives yield comparable results
  - ▶ in terms of the utility and necessity for the service provided

The solution which discloses the

- ▶ least amount of personal information should be preferred

How do I know that a solution

- ▶ discloses only "a small amount of information"
  - ▶ about an individual?



# Big Data Analytics: Utility

- ▶ Service to find a route or parking lot
- ▶ Traffic Congestion Management
- ▶ Diagnosis in Water Supply
- ▶ City Planning
- ▶ Research links between Illnesses
- ▶ Monitor and Diagnose of Equipment
- ▶ Smart Power Grid management

## Example (Big Data DB)



# DiffPriv Context and Goal

## Context: "Private Data Release"

Data was collected gathering information from

- ▶ a sample of users  $\{U_1, U_2, \dots, U_i, \dots\}$  from a population  $Pop$

The released information may let an attacker learn something about  $U_i$

- ▶ the question is:
  - ▶ **could he have learned it also if  $U_i$  had not been in the sample?**

## Differential Privacy

Technology for introducing

- ▶ **just enough noise** to ensure:
  - ▶ If an attacker learns something about  $U_i$
  - ▶ he could have learned it also if  $U_i$  was not in the sample

# Data Release Scenario: Two Models

## Sanitized Database

Non-Interactive:

- ▶ Data is sanitized and released

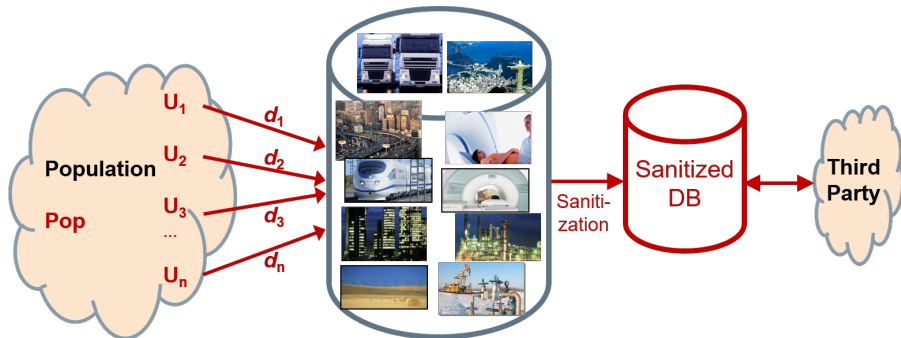
## Query-Response

Interactive sanitization:

- ▶ Respond sequentially to Queries
- ▶ An attacker may want to
  - ▶ Adaptively choose the queries
  - ▶ to gain the most information
- ▶ But the responses may also adapt
  - ▶ to reduce leakage



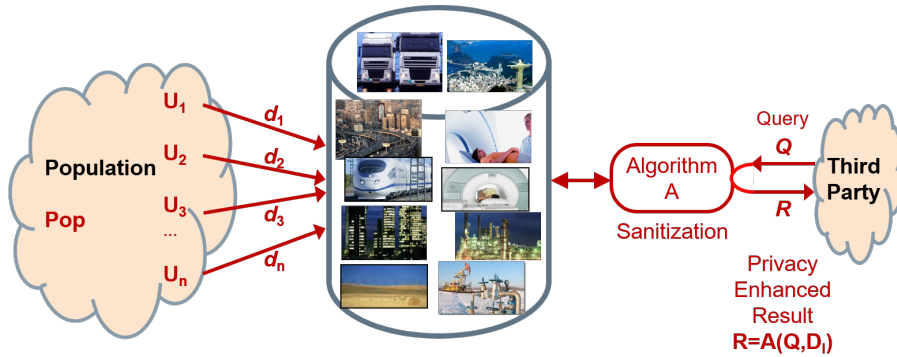
# Sanitized Database Scenario



Data Subjects =  
Sample =  $\{U_i \mid i \in I\}$

$DB = D_I = \{d_i \mid i \in I\}$ , where  
 $d_i$ : data contributed by  $U_i$

# Query-Response Scenario via Interactive Sanitization



Data Subjects =  
Sample =  $\{U_i \mid i \in I\}$

$DB = D_I = \{d_i \mid i \in I\}$ , where  
 $d_i$ : data contributed by  $U_i$

Figure: Query-Response Scenario

 Scenario

## Data Subject Set

- ▶ Users in the sample =  $\{U_i | i \in I\}$

## Sample = DB

- ▶  $D_I = \{d_i | i \in I, \text{ where } d_i \text{ data contributed by } U_i\}$

## R = disclosed results

## Sanitized DB

- ▶  $R = \text{SanitizedDB} = A(D_I)$

Answering a Query  $Q$ 

- ▶  $R = A(Q, D_I)$



# Examples of Sanitization Algorithms

- ▶ Input perturbation
- ▶ Add random noise to DB, release
- ▶ Summary statistics
  - ▶ 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



# Can we tell $U_i$ "we will publish some results, but . . . ?"

## The results will not depend on your data?

- ▶ Nonsense!
  - ▶ If it does not depend on  $U_1, U_2, U_3$ , etc
  - ▶ then it does not depend on the data at all



# Can we tell $U_i$ "we will publish some results, but . . . ?"

## Anything that can be learned about $U_i$ from $R$

- ▶ can be learned without access to  $R$
- ▶ Nonsense!
  - ▶ If a study reveals that 80% of the mathematicians have a poor memory,
    - ▶ . . . we have learned something about me
    - ▶ As we saw, information is a matter of probabilities
  - ▶ And this information may have consequences:
    - ▶ The health insurance for *John Doe* may go up
      - ▶ Because a report proved that smoking was unhealthy

# Can we tell $U_i$ "we will publish some results, but . . . ?"

## The results are anonymous . . . ?"

- ▶ Unfortunately not!
  - ▶ Reason: there is side information to correlate with
- ▶ Some examples:
  - ▶ 1. The AOL Search debacle
  - ▶ 2. Korolova 2011's Facebook microtargeting attack
  - ▶ 3. Netflix Prize
  - ▶ 4. Massachusetts Group Insurance Commission (GIC) medical encounter DB
  - ▶ 5. Metadata and Mobility DBs



# AOL Search Debacle (2006)

## A Face Is Exposed for AOL Searcher No. 4417749

By MICHAEL BARBARO and TOM ZELLER Jr.

Published: August 9, 2006

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

**The New York  
Technol**

Name: Thelma A  
Age: 62  
Widow  
Residence: Lilbu



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

And search by search, click by click, the identity of AOL user No. 4417749 became easier to discern. There are

Figure: AOL Debacle



# Korolova 2011's Facebook microtargeting attack

## Facebook does not sell information to advertisers

- ▶ But has an advertising systems that enable
  - ▶ personalized social microtargeted advertising
- ▶ It has an intermediate layer between individual user data and advertisers
  - ▶ the system collects
    - ▶ ads advertisers want to display
    - ▶ and targeting criteria
  - ▶ and delivers the ads to people who fit those criteria
- ▶ But this does not ensure
  - ▶ "ads delivery reveals no personal information to the advertiser"
- ▶ She communicated her findings to Facebook Jul 2010,
  - ▶ FB immediately changed their advertising system
    - ▶ to make these attacks difficult to implement



# Massachusetts GIC medical DB

- ▶ (GIC = Group Insurance Commission)

## L Sweeney (CMU) linked the anonymized GIC DB

- ▶ with the Voter Registration List for Cambridge, MA
- ▶ The published GIC DB included zip code, date of birth, and gender
  - ▶ sufficient to uniquely identify a significant fraction of the population
  - ▶ Medical visits for many individuals can be easily identified
    - ▶ including for the governor of Massachusetts (W Weld)
- ▶ Note: Birthdate, gender, zip code of many people is public information
  - ▶ (say, via FB)
    - ▶ thus the linking with voret registration DB was really not necessary
- ▶ The GIC re-identification attack directly motivated  $k$ -anonymity



# Types of Fields in Data

Assume a DB (or table) in the form of a matrix:

- ▶ the rows (or entries) correspond to the different individuals (data owners)
- ▶ The columns are the fields of the data

We assume 3 types of fields:

- ▶ IDs
- ▶ QId
- ▶ SAs



# Types of Fields in Data

Identifiers: Attributes that usually identify individuals

- ▶ Name, Address, Phone No, Id Number

Quasi-identifiers (QIs): Attributes like

- ▶ Zip-code, Birth-date, and Gender
- ▶ QIs can be linked with external data to
  - ▶ uniquely identify individuals in the population

Sensitive Attributes (SA): Personal information that should not be publicly linked to a person/user/identifier

- ▶ Disease, Salary
- ▶ the adversary is assumed to know the QIs of some subjects
  - ▶ but not the SAs (and wants to learn the SAs)

Problem: Distinction btw. Quasi-identifiers vs Sensitive Attributes

- ▶ Not always clear-cut

$k$ -Anonymity requires the division of attributes into

- ▶ quasi-identifiers (QIs) and
- ▶ sensitive attributes (SA)



# "Sanitizing / Anonymizing a DB"

- ▶ Identifiers must be eliminated
- ▶ QIs (and also SAs) can be
  - ▶ **generalized**
    - ▶ by replacing the data value with a less precise value that is semantically consistent
- ▶ Whole entries (rows) can be **suppressed**:
  - ▶ removing whole tuples that stand out



# *k*-Anonymity

## Assume

- ▶ a DB containing only QIs and SAs is disclosed
- ▶ an attacker knows the QIs of his victims
  - ▶ perhaps: he knows the QIs of all persons in the DB:
  - ▶ he knows: Peter Smith has QIs xyz, Maria Baum QIs abc, etc

But assume this attacker who only knows the QIs (not any SAs)

- ▶ two individuals (= records) are indistinguishable for the attacker
  - ▶ if they have the same QIs

## *k*-anonymity

Make every record in the table indistinguishable

- ▶ from at least  $k - 1$  other records
  - ▶ given only the quasi-identifiers



# $k$ -Anonymity, more formally

## A sanitized DB satisfies $k$ -anonymity

Consider two entries/tuples in the table/DB:

- ▶ They are QI-equivalent  $\Leftrightarrow$ 
  - ▶ those tuples agree on the QIs

Every combination of QIs that appears in the table

- ▶ must appear at least  $k$  times

In other words:

- ▶ the QI-equivalence classes have at least  $k$  elements
- ▶  $k$ -anonymity provides is simple and easy to understand



# k-Anonymity

	ZIP Code	Age	Disease		ZIP Code	Age	Disease
1	47677	29	Heart Disease	1	476**	2*	Heart Disease
2	47602	22	Heart Disease	2	476**	2*	Heart Disease
3	47678	27	Heart Disease	3	476**	2*	Heart Disease
4	47905	43	Flu	4	4790*	$\geq 40$	Flu
5	47909	52	Heart Disease	5	4790*	$\geq 40$	Heart Disease
6	47906	47	Cancer	6	4790*	$\geq 40$	Cancer
7	47605	30	Heart Disease	7	476**	3*	Heart Disease
8	47673	36	Cancer	8	476**	3*	Cancer
9	47607	32	Cancer	9	476**	3*	Cancer

Original Table

A 3-Anonymous Version



# $k$ -Anonymity

- ▶ The separation between QIs and sensitive attributes
  - ▶ is hard to get in real-life
- ▶ Some attributes in the GIC data were considered as QIs
  - ▶ but it is arbitrary to say they are the only QIs
  - ▶ Other attributes include visit date, diagnosis, etc
- ▶ There may exist adversaries who know this information about someone
  - ▶ and if then the record can be re-identified
    - ▶ this it is still a serious privacy breach

# $k$ -Anonymity

- ▶ The same happens for any kind of DB
  - ▶ When publishing anonymized microdata
  - ▶ one should defend against all kinds of adversaries
    - ▶ some know one set of attributes
    - ▶ others know different sets
- ▶ An attribute about one individual may be
  - ▶ known by some adversaries, and
  - ▶ unknown for others
    - ▶ and should be considered sensitive

# $k$ -Anonymity

- ▶ Any separation between QIs and SAs
  - ▶ is essentially making assumptions
    - ▶ about the adversary's background knowledge
- ▶ But the assumption may be wrong
  - ▶ rendering the privacy protection invalid

# $k$ -Anonymity

- ▶ With  $k$ -anonymity the adversary may not
  - ▶ identify the record of the target, but he could **infer**
    - ▶ the  $SA$  value from the published data
- ▶ Maybe all other users in the  $k$ -anonymity group
  - ▶ Share some sensitive data

# k-Anonymity

## Skewness Attack

- ▶ In this example (with 4-anonymity)
  - ▶ the probability that an artist has *HIV* is 75%
  - ▶ which is not the same as in the probability in the population
    - ▶ If you know that the artist visited the hospital
    - ▶ you may guess with  $p = .75$  that she has Aids:

Job	Gender	Age	Disease
Professional	Male	[35-40)	Hepatitis
Professional	Male	[35-40)	Hepatitis
Professional	Male	[35-40)	Hepatitis
Professional	Male	[35-40)	HIV
Artist	Female	[30-35)	Flu
Artist	Female	[30-35)	HIV
Artist	Female	[30-35)	HIV
Artist	Female	[30-35)	HIV



# $\ell$ -diversity

## $\ell$ -diversity addresses this limitation of $k$ -anonymity

- ▶ Requires that every  $QI$  group should contain at least  $\ell$  "well-represented"  $SA$  values
  - ▶ Say, at least  $\ell$  distinct  $SA$  values in each  $QI$  group
- ▶ But if the distribution of  $SA$  values is skewed in the population, but not in the table
  - ▶ the sensitive value of individuals may still be revealed

# $k$ -Anonymity and $\ell$ -diversity: Composition Attack

First Database (4-anonymous, 3-diverse) form one Hospital

	Non-Sensitive		Sensitive
	Zip code	Age	Condition
1	130**	<30	AIDS
2	130**	<30	Heart Disease
3	130**	<30	Viral Infection
4	130**	<30	Viral Infection
5	130**	$\geq 40$	Cancer
6	130**	$\geq 40$	Heart Disease
7	130**	$\geq 40$	Viral Infection
8	130**	$\geq 40$	Viral Infection
9	130**	3*	Cancer
10	130**	3*	Diabetes
11	130**	3*	Cancer
12	130**	3*	Tuberculosis

# $k$ -Anonymity and $\ell$ -diversity: Composition Attack

Second Database (6-anonymous, 4-diverse) form another Hospital

	Non-Sensitive		Sensitive
	Zip code	Age	Condition
1	130**	<35	AIDS
2	130**	<35	Tuberculosis
3	130**	<35	Flu
4	130**	<35	Flu
5	130**	<35	Cancer
6	130**	<35	Cancer
7	130**	$\geq 35$	Cancer
8	130**	$\geq 35$	Heart Disease
9	130**	$\geq 35$	Viral Infection
10	130**	$\geq 35$	Tuberculosis
11	130**	$\geq 35$	Flu
12	130**	$\geq 35$	Viral Infection



# $k$ -Anonymity and $l$ -diversity: Composition Attack

- ▶ Example of a composition attack
  - ▶ If you know Alice is 28, lives in zip code 13012 and visits both hospitals
    - ▶ you learn she has AIDS



# $t$ -closeness

- ▶ To avoid skewness attacks, which also happen for DBs with  $\ell$ -diversity
  - ▶  $t$ -closeness was invented, which requires
  - ▶ the distribution of  $SA$  values in any  $QI$  group,  $P$
  - ▶ must be close to the distribution of  $SA$  values in the whole data set,  $Q$ 
    - ▶ within a maximum distance  $t$



# Netflix Prize

## Netflix wanted to get better "suggestions"

- ▶ ... a collaborative filtering algorithm to predict user ratings for films
- ▶ And released a training dataset for the competing developers to train their systems
  - ▶ "All personal information has been removed", etc
  - ▶ V Shmatikov (Austin) linked this DB
    - ▶ with the IMDB DB, compromising the identity of users

# Metadata and Mobility DBs

12 points are needed to uniquely identify a fingerprint

- ▶ 1930: Edmond Locard showed that

4 spatio-temporal points are enough to uniquely identify 95% of the people

- ▶ Even if resolution is low
  - ▶ De Montjoye (MIT)
- ▶ Thus coarse or blurred mobility datasets provide little anonymity

# You only need 33 bits

- ▶ Birth date, postcode, gender
  - ▶ Unique for 87% of  $U_S$  population (Sweeney 1997)
- ▶ Preference in movies
  - ▶ 99% of 500K with 8 rating (Narayanan 2007)
- ▶ Web browser
  - ▶ 94% of 500K users (Eckersley)
- ▶ Writing style
  - ▶ 20% accurate out of 100K users (Narayanan 2012)
- ▶ In an anonymized credit card data-set
  - ▶ 4 *randomly selected credit card transactions*
    - ▶ are sufficient to uniquely identify most people
  - ▶ This implies *every transaction* in the enormous data-set
    - ▶ is a quasi-identifier



# Can we tell $U_i$ we will publish some results, but

... an attacker will not be able to distinguish ...

- ▶ Regardless of external knowledge, an adversary
  - ▶ with access to the sanitized database
    - ▶ draws almost the same conclusions
  - ▶ whether or not my data is included in the original data
- ▶ This is Differential Privacy



# Can we tell $U_i$ "we will publish some results, ..."

- ▶ But the chance that the sanitized result will be
  - ▶ nearly the same
  - ▶ whether you submit your information or not

$$\frac{\text{Prob}(A(D_I) = R)}{\text{Prob}(A(D_I \pm i) = R)} < e^\epsilon \approx 1 + \epsilon$$

- ▶ The two databases  $D_I, D_I \pm i$  are called "neighbors"
  - ▶ since they differ only on one "row" (that is. on one data subject)

# Properties of DiffPriv

## Calibration

It is possible to adapt the sanitization to offer

- ▶ more usability, less privacy
- ▶ or viceversa

## Composability

Applying the sanitization several times

- ▶ yields a graceful degradation

## Robustness to side information

No matter what the adversary knows

- ▶ the adversary wants to know the SAs of an individual
  - ▶ and he has lots of information about him

The information he obtains is almost the same

- ▶ if the individual participated in the survey or not





# Neighbor DBs

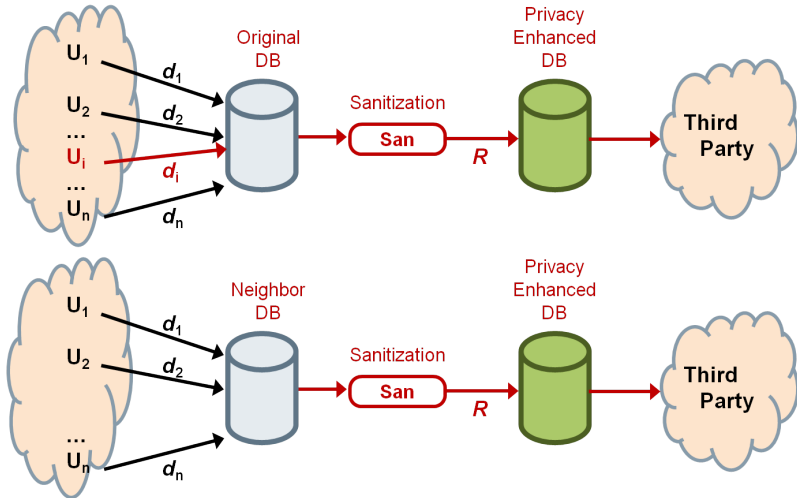


Figure: The two samples differ only on one entry



# Diff Priv Condition

$$\frac{\text{Prob}(A(D_I) = R)}{\text{Prob}(A(D_I \pm i) = R)} < e^\epsilon \approx 1 + \epsilon$$

- ▶ The two databases  $D_I, D_I \pm i$  are called "neighbor" DBs
  - ▶ Note that the condition above is equivalent to
    - ▶  $|\text{Prob}(A(D_I) = R) - \text{Prob}(A(D_I \pm i) = R)| < \epsilon'$ 
      - ▶ for an  $\epsilon'$  very close to the original  $\epsilon$
- ▶  $A$  is the query-sanitization algorithm
  - ▶  $\epsilon > 0$  small chosen by the designer  $e^\epsilon \approx 1$
- ▶ If  $e^\epsilon \gg 1$ , very little privacy is offered
- ▶ If  $e^\epsilon = 1$ , individuals have no effect and there is zero utility

# Basic Algorithm

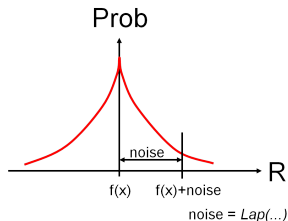
User- $\Psi$	Age	y1	y2
xyz	20	1	0
abc	55	3	1
rin	18	5	1
vhp	36	4	0
zuv	42	2	1
ier	47	8	1
mqw	63	4	0
...	...	...	...
Range	1-100	1-10	0-1

Assume the DB manager gets a query

- ▶ say: How many users with age  $> 30$  have  $y2 = 1$  ?
- ▶ For that, the DB manager must first answer:
  - ▶ How much could the answer change, when
    - ▶ Adding or removing a user?
- ▶ Here: by  $\pm 1$
- ▶ Call this value  $GS_f$

# Basic Algorithm

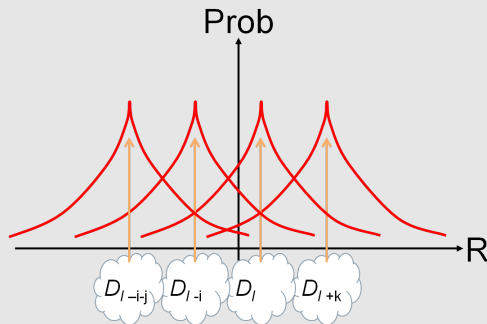
- ▶ Thus  $GS_f$ :
  - ▶  $GS_f := \max_{l,i} \|f(D_l \pm i) - f(D_l)\|$
- ▶ Sanitize  $f(x)$  using:
 
$$A(x) = f(x) + \text{Lap}\left(\frac{GS_f}{\epsilon}\right)$$
- ▶ Q (query): How many users with age > 30 have  $y_2 = 1$  ?
- ▶ R:  $f(x) + \text{Lap}\left(\frac{1}{\epsilon}\right)$





# "Almost indistinguishability"

In which world am I?



**Figure:** Given a value of the response  $R$ , it is difficult to infer in which "world"  $D_I \pm \dots$  we are

# Theorem

Theorem:

$$A(x) = f(x) + \text{Lap}\left(\frac{GS_f}{\epsilon}\right) \text{ is } \epsilon\text{-DP}$$

Figure: Theorem: One algorithm for Differential Privacy

# It looks good !

## Not so fast. . .

- ▶ Quite a bit of problems:
- ▶ In some cases, too restrictive
- ▶ Ok if  $GS_f = 1$ 
  - ▶ But in the case of age  $GS_f = 100$
  - ▶ . . . in the case of mean  $GS_f = \infty$
  - ▶ . . . in the case of correlation  $GS_f = \infty$
- ▶ Assumes a much to strong attacker
- ▶ That knows basically anything he could know about the population
- ▶ Anonymizing dynamically changing DBs is not trivial

# Apparent Properties of DP

---

## Simple Properties

---

- Post-processing
  - All Non-trivial differentially private mechanisms must be random
  - If  $A_1, A_2$  are  $\epsilon_1, \epsilon_2$ - DP respectively,  
then  $(A_1, A_2)$  is  $\epsilon_1 + \epsilon_2$ - DP
-



# Non-properties

- ▶ "If an attacker can't tell whether or not you submitted a survey, they can't learn anything about you from the results"
  - ▶ With the right background information
    - ▶ an attacker can learn about Peter Pan just from general information about the population, even if he didn't submit a survey!
- ▶ "An attacker can't possibly guess with high probability whether you took the survey"
  - ▶ Differential privacy hides the differences between data sets that differ by one individual, not whole groups
    - ▶ If Peter Pan is part of a group, the "lost boys" the whole group may have a detectable impact on the results
    - ▶ and an attacker might correctly guess that if the group was involved, Peter Pan also was

# Non-properties

- ▶ Differential privacy ensures that the released result  $R$  gives minimal evidence about whether or not any given individual contributed to the data set
  - ▶ If individuals only provide information about themselves
    - ▶ this protects Personally Identifiable Information to the strictest possible degree
  - ▶ But you may *indirectly* provide information about others:
    - ▶ Say, if Goofy likes Hospitals where many people go
    - ▶ if you learn from a response  $R$  that a hospital has many patients
    - ▶ then you may deduce that he is part of the DB
    - ▶ and use this to perhaps learn something about Goofy

# Non-properties

- ▶ Differential privacy ensures that the released result  $R$  gives minimal evidence about whether or not any given individual contributed to the data set
- ▶ It protects all personal information in the data set
- ▶ It does not prevent attackers from drawing conclusions about individuals from the aggregate results over the population: Researchers still need to be careful that their studies are ethical
- ▶ Differential privacy ensures that the released result  $R$  gives minimal evidence about whether or not any given individual contributed to the data set
- ▶ It protects all personal information in the data set
- ▶ It does not prevent attackers from using aggregate results
- ▶ It does not prevent attackers from learning information about known cohesive groups in the data set. The distribution of the population and the invasiveness of the query should be

# "Safe" $k$ -Anonymity plus random sampling $\Rightarrow$ DiffPriv

- ▶ Almost all  $k$ -anonymization methods
  - ▶ proposed in the literature are vulnerable
    - ▶ because the generation scheme to be applied
    - ▶ is overly dependent on tuples that contain extreme values
    - ▶ leaking information about these tuples
- ▶ One way to avoid that
  - ▶ is to use a generalization scheme that is independent of the input dataset:
    - ▶ the algorithm applies a fixed generation scheme to the input tuples and
    - ▶ then suppresses any tuple that appears less than  $k$  times

# "Safe" $k$ -Anonymity plus random sampling $\Rightarrow$ DiffPriv

- ▶ This is called a **safe**  $k$ -anonymization algorithm
  - ▶ It provides intuitively some level of privacy protection
    - ▶ as each tuple is indeed "hiding in a crowd of at least  $k$ "
- ▶ But the algorithm still does not satisfy differential privacy
  - ▶ simply because the algorithm is deterministic
- ▶ A safe  $k$ -anonymization
  - ▶ preceded with a random sampling step
    - ▶ satisfies  $\epsilon$ -differential privacy with
    - ▶ reasonable parameter  $\epsilon$

# Info Utility

- ▶ Given a privacy-infusing query processor  $\text{San}$ ,
  - ▶ If  $\text{San}$  can be used to answer some query with reasonable accuracy
    - ▶ then we say the  $\text{San}$  has some Utility (or: is useful)
- ▶ We can measure the Utility like this:
  - ▶ we say that **San has Utility** if for any  $\varepsilon > 0$ 
    - ▶ there exist 2 possible database instances  $D_1, D_2$  and
    - ▶ disjoint sets  $S_1, S_2$  such that
    - ▶  $P(\text{San}(D_i) \in S_i) \geq 1 - \varepsilon$  for  $i = 1, 2$  (the randomness only depends on  $\text{San}$ )



## Utility Example

- ▶ Suppose we ask the query
  - ▶ how many cancer patients are in the data?
- ▶ Choose  $\epsilon = 0.05$ , to illustrate
- ▶ Suppose San works as follows:
  - ▶ if there are 0 cancer patients,
    - ▶ it outputs some number in the range  $[0, 1000]$  with probability  $1 - \epsilon = 0.95$
  - ▶ if there are 10, 000 cancer patients,
    - ▶ it outputs some number in the range  $[9000, 11000]$  with probability  $1 - \epsilon = 0.95$
  - ▶ Let  $D1$  to be any database with 0 cancer patients
  - ▶ and  $D2$  to be any database with 10, 000 cancer patients
  - ▶  $S1 = [0, 1000]$ ,  $S2 = [9000, 11000]$



# No-Free-Lunch Theorem

- ▶ Let  $q$  be a sensitive query with 2 possible outcomes
  - ▶ ..., say an assertion about Peter Pan is true or not
- ▶ Let  $A$  be a privacy-infusing query processor  $\text{San}$  with Utility
- ▶ Then for any  $\varepsilon > 0$ 
  - ▶ there exists a probability distribution  $P$  over database instances  $D$ 
    - ▶ such that  $q(D) = 0.5$  for all  $D$ ,
    - ▶ but the attacker wins with probability at least  $1 - \varepsilon$ 
      - ▶ when given  $A(D)$





# Covert-Channel Attacks

- ▶ In a DiffPriv system
  - ▶ the adversary can
  - ▶ learn with perfect certainty whether Peter Pan has a girlfriend different from Wendy
    - ▶ a blatant violation of differential privacy
- ▶ Differential Privacy under Fire, Andreas Haeberlen, Benjamin C. Pierce, Arjun Narayan

# Quiz

## If I randomly sample one record from a large DB

- ▶ consisting of many records, and
  - ▶ publish that record,
  - ▶ would this be differentially private?
- ▶ Prove or disprove that

## If I have a very large DB

- ▶ containing ages of all people living in Bavaria
  - ▶ and I publish the average age of all people in the DB
    - ▶ Intuitively, do you think this preserves users' privacy?
  - ▶ Is this differentially private? Prove or disprove that

## Pros and cons of differential privacy?

# Literature

- ▶ No Free Lunch in Data Privacy, Kifer, Machanavajjhala, 2011
- ▶ Differential Privacy Under Fire, Haeberlen, Pierce, Narayan, 2011
- ▶ Answering  $n^{\{2+o(1)\}}$  counting queries with DiffPriv is hard\* J Ullman, 2013
- ▶ Cynthia Dwork's video tutorial on DP
- ▶ Differential Privacy (Invited talk at ICALP 2006)
- ▶ Privacy Integrated Queries
- ▶ GUPT: Privacy Preserving Data Analysis Made Easy
- ▶ The Differential Privacy Frontier

# Questions?