

# Ensuring (Statistical) Privacy

Daniel Alabi

Ph.D. student in the Theory of Computation group

@ Harvard SEAS (School of Engineering and Applied Sciences)

Advised by Salil Vadhan

Code for this talk: [https://github.com/alabid/pre\\_college\\_2019](https://github.com/alabid/pre_college_2019)

See full course materials here: <http://people.seas.harvard.edu/~salil/cs208/>

My email: [alabid@g.harvard.edu](mailto:alabid@g.harvard.edu)



# Table of Contents

- Motivations
    - Reidentification via Linkage Attacks
    - Reconstruction and Inference Attacks
  - Definitions
    - K-anonymity
    - Differential Privacy
  - Mechanisms
    - Laplace Mechanism
  - Code Demonstrations
    - Python Code



# The Problem

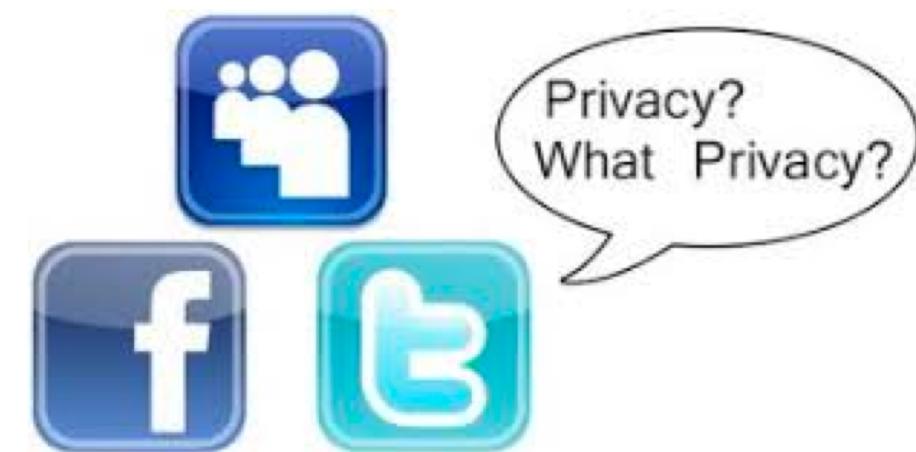
We have a dataset with sensitive information, such as:

1. Health records (e.g. reveals which disease a patient has)
2. Census data (e.g. reveals income range)
3. Social network activity (e.g. which pages you like)



How can we:

1. Allow the use of the data?
2. Protect the privacy of the data subjects?
3. Achieve both (1) and (2)?



# Some Approaches to Solve the Problem

Encrypt the Data:

Name	Sex	Blood	...	HIV?
James	M	B	...	N
Peter	M	O	...	Y
...	...	...	...	...
Paul	M	A	...	N
Eve	F	B	...	Y

# Some Approaches to Solve the Problem

Encrypt the Data:

Name	Sex	Blood	...	HIV?
James	M	B	...	N
Peter	M	O	...	Y
...	...	...	...	...
Paul	M	A	...	N
Eve	F	B	...	Y

Name	Sex	Blood	...	HIV?
10101	01010	01000	...	00001
11010	01101	10111	...	10111
...	...	...	...	...
10100	10000	11101	...	01111
11000	10001	11110	...	10001

# Some Approaches to Solve the Problem

Encrypt the Data: Are we happy with this solution? Why or why not?

Name	Sex	Blood	...	HIV?
James	M	B	...	N
Peter	M	O	...	Y
...	...	...	...	...
Paul	M	A	...	N
Eve	F	B	...	Y



Name	Sex	Blood	...	HIV?
10101	01010	01000	...	00001
11010	01101	10111	...	10111
...	...	...	...	...
10100	10000	11101	...	01111
11000	10001	11110	...	10001

# Some Approaches to Solve the Problem

“Anonymize the Data”: Are we happy with this solution? Why or why not?

Name	Sex	Blood	...	HIV?
James	M	B	...	N
Peter	M	O	...	Y
...	...	...	...	...
Paul	M	A	...	N
Eve	F	B	...	Y



Name	Sex	Blood	...	HIV?
XXXXX	M	B	...	N
XXXXX	M	O	...	Y
...	...	...	...	...
XXXXX	M	A	...	N
XXXXX	F	B	...	Y

# Some Approaches to Solve the Problem

“Anonymize the Data”: Not sufficient because of linkage attacks!

87% of US population have unique date of birth, gender, and postal code!

[Golle and Partridge ‘09]

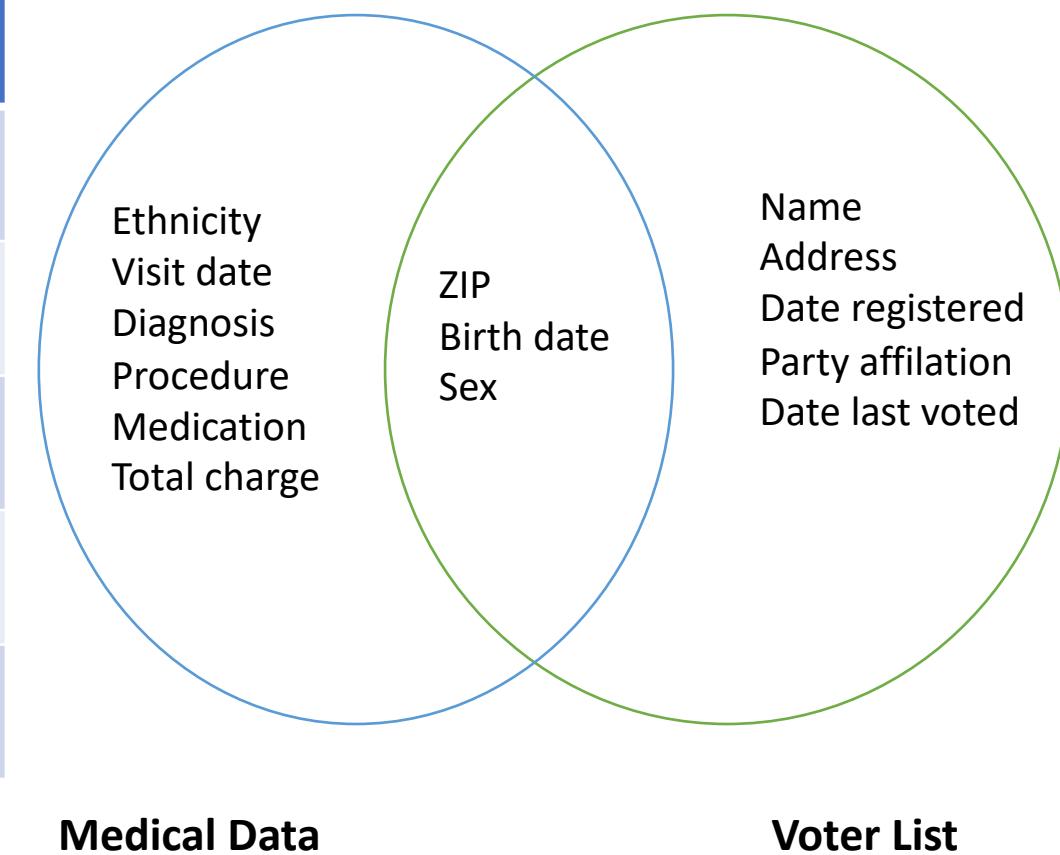


# Some Approaches to Solve the Problem

“Anonymize the Data”: Reidentification via Linkage

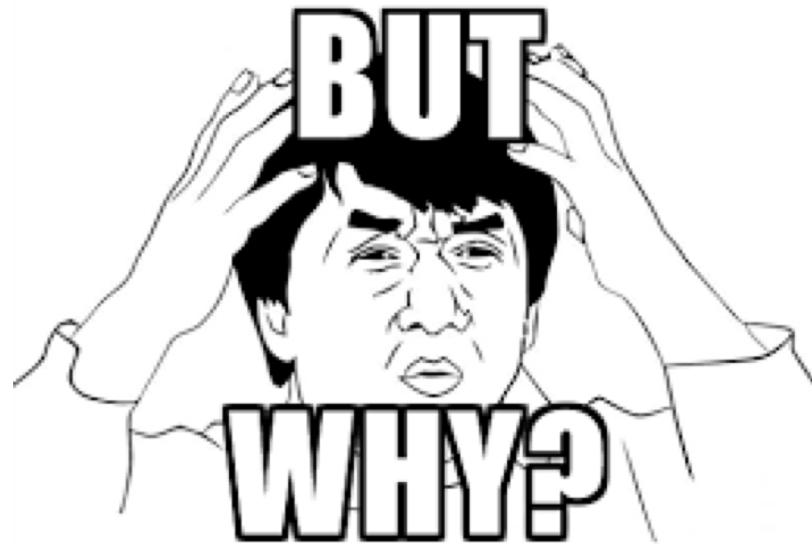
Can uniquely identify > 60% of the U.S. population [Sweeney '00, Golle '06, Sweeney '97]

Name	Sex	Blood	...	HIV?
XXXXX	M	B	...	N
XXXXX	M	O	...	Y
...	...	...	...	...
XXXXX	M	A	...	N
XXXXX	F	B	...	Y



# Message

Releasing too many statistics with too much accuracy can lead to a reconstruction of the entire dataset or inference attacks



# The story so far

- Motivations
  - Reidentification via Linkage Attacks
  - Reconstruction and Inference Attacks

Reconstruction attack: If we have dataset  $x \in \{0, 1\}^n$  and person  $i$  has sensitive bit  $x_i$  and attacker/adversary gets  $q_S(x) = \sum_{i \in S} x_i$  for any  $S \subseteq [n]$ .

# The story so far

- Motivations
  - Reidentification via Linkage Attacks
  - Reconstruction and Inference Attacks

Reconstruction attack: If we have dataset  $x \in \{0, 1\}^n$  and person  $i$  has sensitive bit  $x_i$  and attacker/adversary gets  $q_S(x) = \sum_{i \in S} x_i$  for any  $S \subseteq [n]$ .

[Dinur-Nissim '03]: With high probability, adversary can reconstruct 0.99 fraction of the dataset  $x \in \{0, 1\}^n$  if noise added to each query is less than  $o(\sqrt{n})$  and #queries is  $n$ .

# The story so far

- Motivations
  - Reidentification via Linkage Attacks
  - Reconstruction and Inference Attacks

Reconstruction attack: If we have dataset  $x \in \{0, 1\}^n$  and person  $i$  has sensitive bit  $x_i$  and attacker/adversary gets  $q_S(x) = \sum_{i \in S} x_i$  for any  $S \subseteq [n]$ .

[Dinur-Nissim '03]: With high probability, adversary can reconstruct 0.99 fraction of the dataset  $x \in \{0, 1\}^n$  if noise added to each query is less than  $o(\sqrt{n})$  and #queries is  $n$ .

Inference attack: Attacker gets  $n^2$  answers and needs to know if someone is in dataset or not.

The story so far

## Message

Releasing too many statistics with too much accuracy can lead to a reconstruction of the entire dataset or inference attacks

# Some Approaches to Solve the Problem

So now what?

- Encryption doesn't work
- Anonymization doesn't work
- Even adding insufficient noise to an attacker's query is not good enough

# Some Approaches to Solve the Problem

So now what?

- Encryption doesn't work
- Anonymization doesn't work
- Even adding insufficient noise to an attacker's query is not good enough

Possible Responses:

- Privacy is an illusion!

# Some Approaches to Solve the Problem

So now what?

- Encryption doesn't work
- Anonymization doesn't work
- Even adding insufficient noise to an attacker's query is not good enough

Possible Responses:

- Privacy is an illusion!
- In the long run, it's better to use data for research!  
Ignore privacy!



# Some Approaches to Solve the Problem

So now what?

- Encryption doesn't work
- Anonymization doesn't work
- Even adding insufficient noise to an attacker's query is not good enough

Possible Responses:

- Privacy is an illusion!
- In the long run, it's better to use data for research!  
Ignore privacy!
- Never release statistics about any dataset!



# Some Approaches to Solve the Problem

So now what?

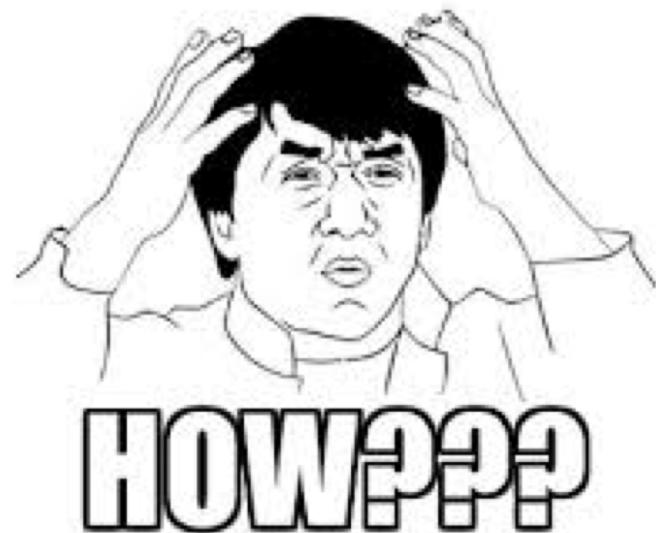
- Encryption doesn't work
- Anonymization doesn't work
- Even adding insufficient noise to an attacker's query is not good enough

Possible Responses:

- Privacy is an illusion!
- In the long run, it's better to use data for research! Ignore privacy!
- Never release statistics about any dataset!
- Is there a way to add enough noise to queries and still allow for usefulness?

# Main Message of this Talk

Yes, there is a way to add enough noise to queries and  
still allow for usefulness!



# An approach: K-anonymity

[Sweeney '02]: A mechanism satisfies  $k$ -anonymity if for every dataset, the output of the mechanism has the property that every distinct row occurs at least  $k$  times.

# An approach: K-anonymity

[Sweeney '02]: A mechanism satisfies  $k$ -anonymity if for every dataset, the output of the mechanism has the property that every distinct row occurs at least  $k$  times.

*Intuition:* Privacy ensured if I can't isolate you!

# An approach: K-anonymity

[Sweeney '02]: A mechanism satisfies  $k$ -anonymity if for every dataset, the output of the mechanism has the property that every distinct row occurs at least  $k$  times.

3-anonymous dataset:

Zip code	Age	Nationality
021**	< 30	*
021**	< 30	*
021**	< 30	*
021**	> 40	*
021**	> 40	*
021**	> 40	*
021**	3*	*
021**	3*	*
021**	3*	*

# An approach: K-anonymity

It's a nice approach but:

- Doesn't cwell (e.g. if you have two  $k$ -anonymous datasets)
- Utility not as quantifiable as other approaches

Zip code	Age	Nationality
021**	< 30	*
021**	< 30	*
021**	< 30	*
021**	> 40	*
021**	> 40	*
021**	> 40	*
021**	3*	*
021**	3*	*
021**	3*	*

# Table of Contents

- Motivations
  - Reidentification via Linkage Attacks
  - Reconstruction and Inference Attacks
- Definitions
  - K-anonymity
  - Differential Privacy
- Mechanisms
  - Laplace Mechanism
- Code Demonstrations
  - Python Code

# Differential Privacy

- Utility
- Privacy
- Definition

# Differential Privacy

- Utility: enable “statistical analysis” on datasets
  - Can release (noisy) statistics such as means, sums, medians, etc.
  - Predictions from trained machine learning models

# Differential Privacy

- Utility: enable “statistical analysis” on datasets
  - Can release (noisy) statistics such as means, sums, medians, etc.
  - Predictions from trained machine learning models
- Privacy: protect each individual in dataset against all possible attack strategies
  - Now and in the future!
  - Even with use of auxiliary information or datasets!
  - Group privacy also allowed!

# Differential Privacy

- Utility: enable “statistical analysis” on datasets
  - Can release (noisy) statistics such as means, sums, medians, etc.
  - Predictions from trained machine learning models
- Privacy: protect each individual in dataset against all possible attack strategies
  - Now and in the future!
  - Even with use of auxiliary information or datasets!
  - Group privacy also allowed!
- Definition: pure and approximate

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

Other references:

Motivated from and based off of work in

[Dinur-Nissim '03, Dwork-Nissim '04, Blum-Dwork-McSherry-Nissim '05]

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

*Intuition*: for a statistic, the effect of each individual (whether in the dataset or not) should be close to nothing.

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

*Intuition*: for a statistic, the effect of each individual (whether in the dataset or not) should be close to nothing.

*Worst-case notion*: protects against all possible adversaries and any kind of individual.

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

For any algorithm  $\mathcal{A}$ , it satisfies  $\epsilon$  differential privacy if

For all datasets  $D, D'$  differing in exactly one row all queries  $q$

Distribution of  $\mathcal{A}(D, q)$  is at most  $\epsilon$  away from  $\mathcal{A}(D', q)$

The smaller  $\epsilon$  is, the more privacy is ensured!

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

For any algorithm  $\mathcal{A}$ , it satisfies  $\epsilon$  differential privacy if

For all datasets  $D, D'$  differing in exactly one row all queries  $q$

Distribution of  $\mathcal{A}(D, q)$  is at most  $\epsilon$  away from  $\mathcal{A}(D', q)$

For all sets  $T$ ,

$$\Pr[\mathcal{A}(D, q) \in T] \leq (1 + \epsilon) \Pr[\mathcal{A}(D', q) \in T]$$

# Differential Privacy

Definition: pure and approximate

[Dwork-McSherry-Nissim-Smith '06]

For any algorithm  $\mathcal{A}$ , it satisfies  $\epsilon$  differential privacy if

For all datasets  $D, D'$  differing in exactly one row all queries  $q$

Distribution of  $\mathcal{A}(D, q)$  is at most  $\epsilon$  away from  $\mathcal{A}(D', q)$

The probability is only over the randomness of the algorithm  $\mathcal{A}$

# Table of Contents

- Motivations
  - Reidentification via Linkage Attacks
  - Reconstruction and Inference Attacks
- Definitions
  - K-anonymity
  - Differential Privacy
- Mechanisms
  - Laplace Mechanism
- Code Demonstrations
  - Python Code

# Mechanisms for Differential Privacy

Examples:

1. Laplace Mechanism [we'll discuss and implement this one!]
2. Gaussian Mechanism
3. Exponential Mechanism
4. Geometric Mechanism

....

# Laplace Mechanism

$\text{Lap}(s)$  is the Laplace Distribution with scale  $s$ .

Some properties:

- Has mean 0
- Has standard deviation  $\sqrt{2} \cdot s$
- It's a “double-exponential” distribution

# Laplace Mechanism for Sum and Average

$$1. \mathcal{A}(x) = \sum_{i=1} x_i + \text{Lap}\left(\frac{1}{\epsilon}\right)$$

where  $x_i \in [0, 1]$  for all  $i \in [n]$ .

# Laplace Mechanism for Sum and Average

$$1. \mathcal{A}(x) = \sum_{i=1} x_i + \text{Lap}\left(\frac{1}{\epsilon}\right)$$

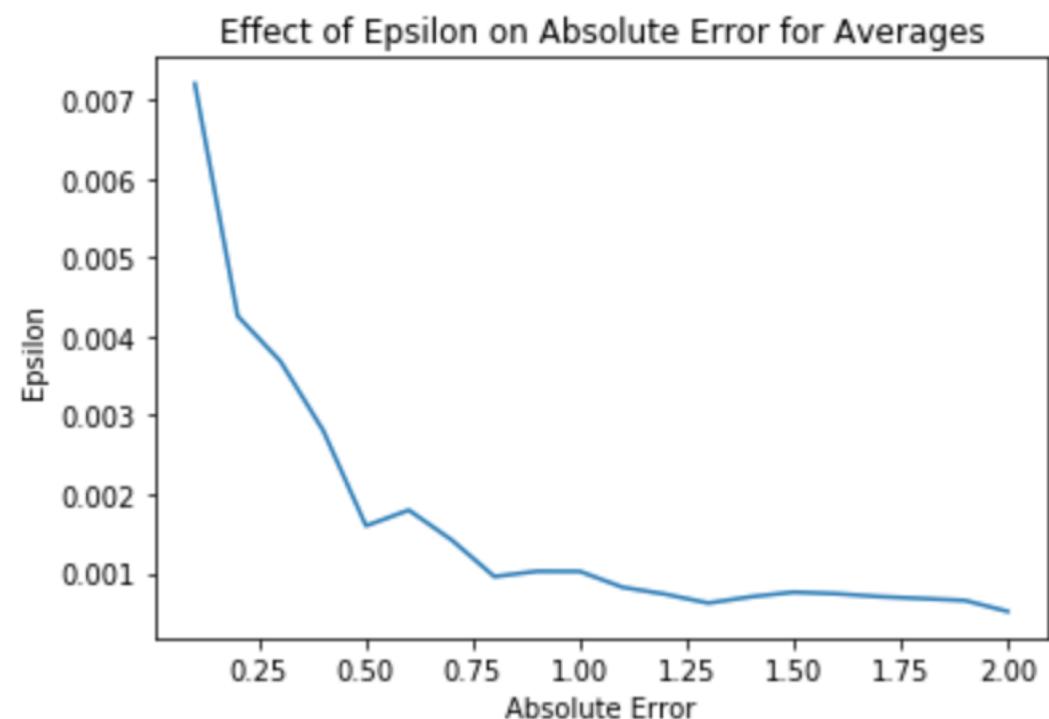
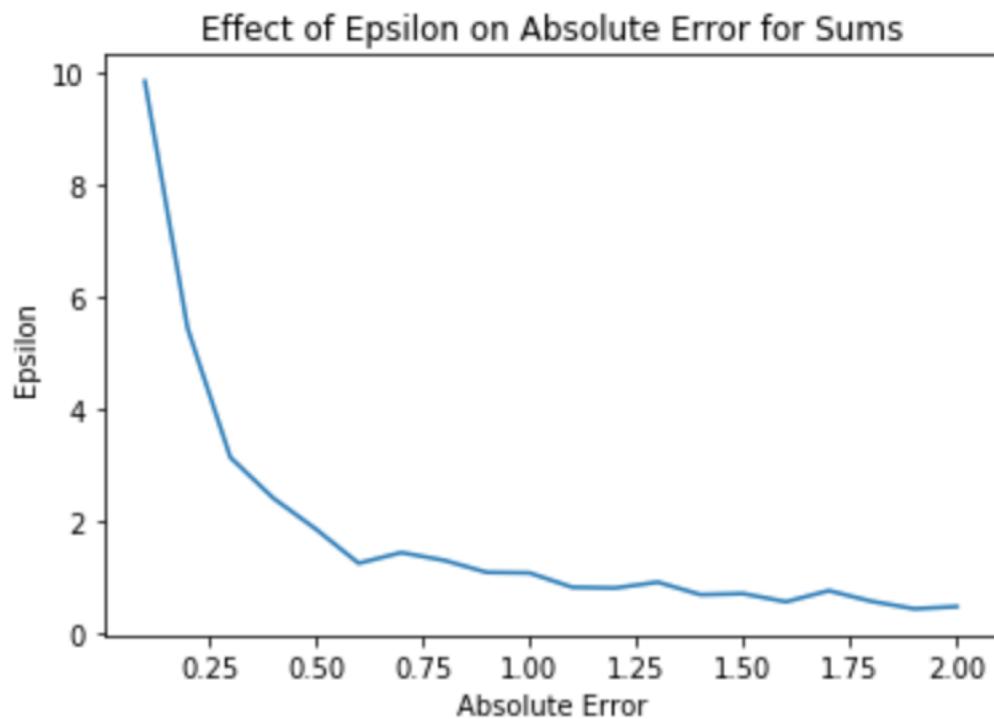
where  $x_i \in [0, 1]$  for all  $i \in [n]$ .

$$2. \mathcal{A}(x) = \frac{1}{n} \cdot \sum_{i=1} x_i + \text{Lap}\left(\frac{1}{n \cdot \epsilon}\right)$$

where  $x_i \in [0, 1]$  for all  $i \in [n]$ .

# Code Demonstration

- <https://github.com/alabid/pre college 2019>



# Conclusion

- Differential Privacy is a mathematically rigorous definition of individual data privacy.
- YOU can code it up. See GitHub page – clone it and play with the code there! [https://github.com/alabid/pre\\_college\\_2019](https://github.com/alabid/pre_college_2019)
- It's being used by the U.S. Census Bureau (for the 2020 Decennial Census), Google, Apple.