Consistent Closures

A Novel Methodology for Analyzing Privacy Definitions

Daniel Kifer

Department of Computer Science & Engineering
Penn State University

(with Bing-Rong Lin)

Outline

What is a Privacy Definition?

- Goal: apply algorithm $\mathcal{M}$ to sensitive data D to produce sanitized output S
- A privacy definition is a contract
  - Restricts behavior of a sanitization algorithm.
  - Provides guarantees about leakage of sensitive information.
- How do we analyze contracts?

What is a Privacy Definition?

- Goal: apply algorithm $\mathcal{M}$ to sensitive data D to produce sanitized output S
- A privacy definition is a contract
  - Restricts behavior of a sanitization algorithm.
  - Provides guarantees about leakage of sensitive information.
- How do we analyze contracts?
  1. Hire lawyers at €£\$ $\times 10^5$ per hour.
  2. Wait many hours.
  3. Hope they get it right.
- Profitable model for privacy research?

What is a Privacy Definition?

- Goal: apply algorithm $\mathcal{M}$ to sensitive data D to produce sanitized output S
- A privacy definition is a contract
  - Restricts behavior of a sanitization algorithm.
  - Provides guarantees about leakage of sensitive information.
- How do we analyze contracts?
  - Spend much time crafting attacks for specific algorithms.
    - Disclosure Risk Evaluation [Rei05] (and many more!)
    - Minimality attack [WFWP07]
    - de Finetti attack [Kif09]
    - Active attacks [BDK07]
    - Homer's attack [HSR$^+$08]
  - Use software
    - Record linkage
- Brittleness/Incompleteness
  - What if our attack does not work?
  - What if software does not find a disclosure?
  - Easy to evade specific attack code.
  - What else is protected?

Methodology of Consistent Closures

- Analytic approach to evaluating privacy definitions.
  - Can identify what is not protected.
  - Can identify what is protected.
    - E.g., Randomized response = protecting parity.
- Evaluates privacy definition rather than specific algorithm and specific input data.
  - Some algorithms provide more protections than others.
  - Interested in base guarantees provided by all algorithms satisfying a privacy definition.
- Helpful to think of privacy definition as a set of algorithms.
  - Often expressed as constraints on algorithm.
  - Eliminates vagueries.
- Overview
  1. Rephrase privacy definition in a normal form.
  2. Extract linear constraints on algorithm's behavior.
  3. Provide Bayesian interpretation of protections.

Intended Scenario

- Attacker knows there exists a sensitive dataset D.
  - Schema of D is known.
- Attacker will know sanitization algorithm $\mathcal{M}$
  - Avoids security by obscurity
  - Allows researchers to judge significance of their results (utility).
- Attacker sees an output $S = \mathcal{M}(D)$
- Attacker's inference considers all possible input datasets $D_1, \ldots, D_n$
  - Inference based on $P(\mathcal{M}(D_1) = S), \ldots, P(\mathcal{M}(D_n) = S)$.
  - Attacker is computationally unbounded (information-theoretic).
  - Attacker may be Bayesian.
- Goal: make statements about how attacker's beliefs will change.

Outline

1. Introduction
2. Consistent Closures Methodology
   - Representation of Algorithms and Privacy Definitions
   - A Normal Form
   - Example: *k*-Anonymity
   - Algorithmic Constraints via Convex Analysis
3. Examples
   - Differential Privacy
   - Randomized Response
   - FRAPP

Representation of $\mathcal{M}$

$$\begin{array}{cc}
 & \begin{array}{cccc} \quad D_1 \quad & \quad D_2 \quad & \dots & \quad D_n \quad \end{array} \\
\begin{array}{c} S_1 \\ S_2 \\ S_3 \\ \vdots \end{array} &
\begin{pmatrix}
P(\mathcal{M}(D_1)=S_1) & P(\mathcal{M}(D_2)=S_1) & \dots & P(\mathcal{M}(D_n)=S_1) \\
P(\mathcal{M}(D_1)=S_2) & P(\mathcal{M}(D_2)=S_2) & \dots & P(\mathcal{M}(D_n)=S_2) \\
P(\mathcal{M}(D_1)=S_3) & P(\mathcal{M}(D_2)=S_3) & \dots & P(\mathcal{M}(D_n)=S_3) \\
\vdots & \vdots & \vdots & \vdots
\end{pmatrix}
\end{array}$$

- Any algorithm $\mathcal{M}$ is a matrix
  - Yes, even deterministic algorithms.
- Rows indexed by outputs S_i
- Columns indexed by datasets D_j
  - Columns correspond to datasets, not individual records!!

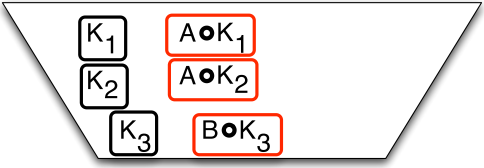
Representation of $\mathfrak{Priv}$

- Privacy definitions expressed as various constraints on algorithms:
  - *k*-Anonymity.
  - Differential Privacy.
  - Randomized Response.
- $\therefore$ A privacy definition $\mathfrak{Priv}$ is just a set of algorithms.

- Not all sets capture intuitive properties of "privacy"
  - Need to normalize sets.

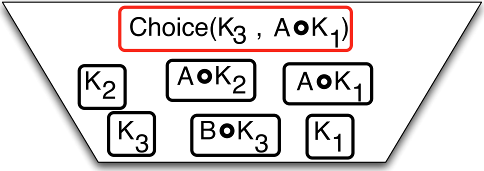
Normalization and Post-processing

- Assumption 1: postprocessing sanitized data does not decrease privacy [KL].
  - (as long as we do not bring in external information)
  - Sanitized data is to be released (postprocessed).
- If $\mathcal{M}$ satisfies privacy and $\mathcal{A}$ is a postprocessing algorithm:
  - $\mathcal{A}(\mathcal{M}(D))$ satisfies privacy..
  - In matrix notation, the new algorithm is $\mathcal{A}\mathcal{M}$.
- Add all possible $\mathcal{A} \circ \mathcal{M}$ to our set.

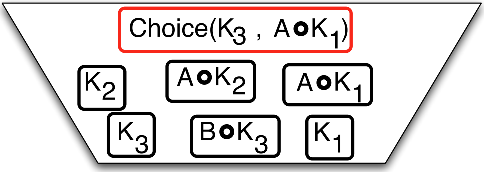


Normalization and Post-processing

- Assumption 2: Convexity [KL].
  - If $\mathcal{M}_1$ satisfies privacy.
  - And $\mathcal{M}_2$ satisfies privacy.
  - Flip coin $P(\text{HEADS}) = p$.
  - $\text{Choice}_p(\mathcal{M}_1, \mathcal{M}_2)$: run $\mathcal{M}_1$ if heads, $\mathcal{M}_2$ if tails.
  - $\text{Choice}_p(\mathcal{M}_1, \mathcal{M}_2)$ satisfies privacy.
  - Why? Increases uncertainty.
- Add all possible $\text{Choice}_p(\mathcal{M}_1, \mathcal{M}_2)$ to our set.



Consistent Closure



- Now our set $\mathfrak{Priv}$ is consistent with basic intuitions on privacy.
- This is called consistent closure.
- Turns implicit assumptions into explicit assumptions
  - Same privacy properties as before.
- Privacy properties easier to see.
  - Can extract linear constraints on the probabilities $P(\mathcal{M}(D_j) = S_i)$.
  - Coefficients of linear constraints $\approx$ prior probabilities.

Outline

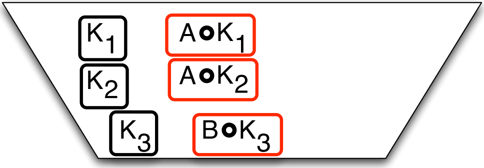
1. Introduction
2. Consistent Closures Methodology
   - Representation of Algorithms and Privacy Definitions
   - A Normal Form
   - Example: k-Anonymity
   - Algorithmic Constraints via Convex Analysis
3. Examples

Example: k-Anonymity

- Start with all algorithms satisfying k-anonymity.

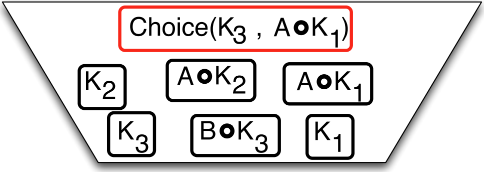
Example: k-Anonymity

- Add all algorithms that produce k-anonymous table then build decision tree
- Add all algorithms that produce k-anonymous table then return linear regression coefficients.
- Add all algorithms that produce k-anonymous table then $\cdots$



Example: k-Anonymity

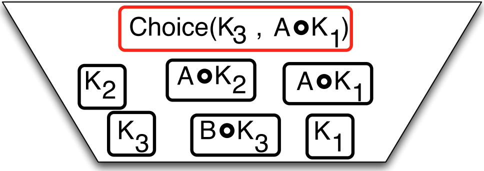
- Add all random choices of algorithms based on coin flips.



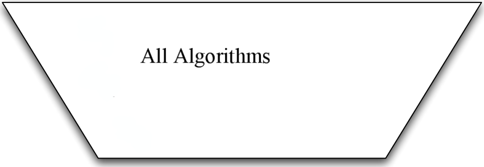
- What do we get?

Example: k-Anonymity

- Add all random choices of algorithms based on coin flips.



- What do we get?



- No guarantees
  - (similar results for many syntactic methods)

Why? Side Channels

- If input table is
  - Then output
- In general:
  - Type of coarsening is unrestricted
  - Can encode entire input as side channel
  - Can efficiently decode it from output.

Zip Code	Age	Nationality	Disease
13053	25	Indian	Cold
13068	39	Russian	Stroke
13053	27	American	Flu
14850	43	American	Cancer
14850	57	Russian	Cancer
14853	40	Indian	Cancer

Zip Code	Age	Nationality	Disease
130**	< 40	*	Cold
130**	< 40	*	Stroke
130**	< 40	*	Flu
1485*	≥ 40	*	Cancer
1485*	≥ 40	*	Cancer
1485*	≥ 40	*	Cancer

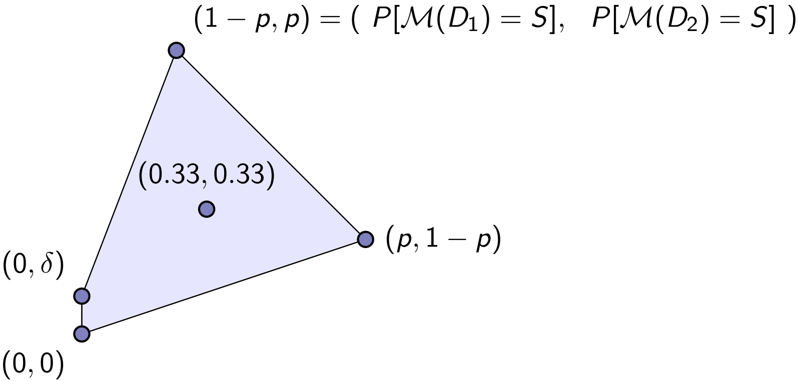
Algorithmic Constraints

$$\begin{array}{cc} & \begin{array}{cccc} \quad D_1 \quad & \quad D_2 \quad & \dots & \quad D_n \quad \end{array} \\ \begin{array}{c} S_1 \\ S_2 \\ S_2 \\ \vdots \end{array} & \begin{pmatrix} P(\mathcal{M}(D_1)=S_1) & P(\mathcal{M}(D_2)=S_1) & \dots & P(\mathcal{M}(D_n)=S_1) \\ P(\mathcal{M}(D_1)=S_2) & P(\mathcal{M}(D_2)=S_2) & \dots & P(\mathcal{M}(D_n)=S_2) \\ P(\mathcal{M}(D_1)=S_3) & P(\mathcal{M}(D_2)=S_3) & \dots & P(\mathcal{M}(D_n)=S_3) \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} \end{array}$$

- Recall matrix view of algorithms.
  - Postprocessing by $\mathcal{A}$ = matrix multiplication $\mathcal{A}\mathcal{M}$.
  - Choice = convex combination of matrices.
- Resulting basic operations on rows.
  - Multiply row by constant
  - Add two rows
- Set of possible rows in consistent closure belongs to a convex set.
  - Convex sets are intersections of half-spaces.
  - Convex sets are solutions to systems of linear inequalities.
  - Linear inequalities can be interpreted as statements about posterior distributions.

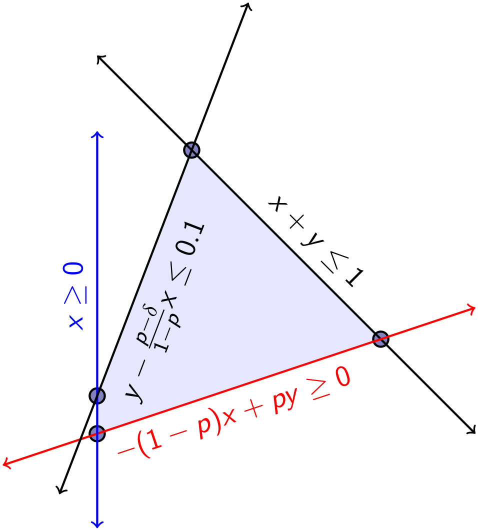
Convex Analysis

- Convex polytope of allowable rows.



Convex Analysis

- Defining linear constraints.



Outline

Outline

1. Introduction
2. Consistent Closures Methodology
3. Examples
   - Differential Privacy
   - Randomized Response
   - FRAPP

Warmup – Differential Privacy

- Differential Privacy is its own convex closure.
- Linear constraints: $P(\mathcal{M}(D_1) = S) \leq e^\epsilon P(\mathcal{M}(D_2) = S)$ for all pairs of neighboring databases.
- Interpretation:

$$\boxed{\frac{P(\text{input} = D_1 \mid \text{output} = S)}{P(\text{input} = D_2 \mid \text{output} = S)}} = \frac{P(D_1)P(\mathcal{M}(D_1) = S)}{P(D_2)P(\mathcal{M}(D_2) = S)}$$

$$\leq \boxed{e^\epsilon \frac{P(input = D_1)}{P(input = D_2)}}$$

- Bounds on increase/decrease of odds ratios of neighboring tables.

Outline

Randomized Response

- In simplest setting:
  - Database is a bit string
  - Each individual corresponds to a bit
  - Value of bit is binary attribute of individual

Definition (Randomized Response)

Flip each bit independently keep it with probability $p > 1/2$ or flip with probability $1 - p$.

$$\text{Outputs:} \quad \begin{array}{c c} & \text{Inputs:} \\ & \begin{array}{cccc} 11 & 10 & 01 & 00 \end{array} \\ \begin{array}{c} 11 \\ 10 \\ 01 \\ 00 \end{array} & \begin{pmatrix} p^2 & p(1-p) & p(1-p) & (1-p)^2 \\ p(1-p) & p^2 & (1-p)^2 & p(1-p) \\ p(1-p) & (1-p)^2 & p^2 & p(1-p) \\ (1-p)^2 & p(1-p) & p(1-p) & (1-p)^2 \end{pmatrix} \end{array}$$

Consistent Closure of Randomized Response

- 2^n linear inequality constraints (n=number of tuples in database)
  - Completely characterize the consistent closure
  - $\mathcal{M}$ is in the consistent closure $\Leftrightarrow$ every row of $\mathcal{M}$ satisfies all constraints.
- Example $n = 2$
  - Notation: $x^s_{11} = P(\mathcal{M}(11) = S)$
  - Constraints on rows are:

$$\begin{aligned}
p^2 x^s_{11} + (1-p)^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01}\\
(1-p)^2 x^s_{11} + p^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01}\\
p^2 x^s_{10} + (1-p)^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00}\\
(1-p)^2 x^s_{10} + p^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00}
\end{aligned}$$

Consistent Closure of Randomized Response

- 2^n linear inequality constraints (n=number of tuples in database)
  - Completely characterize the consistent closure
  - $\mathcal{M}$ is in the consistent closure $\Leftrightarrow$ every row of $\mathcal{M}$ satisfies all constraints.
- Example $n = 2$
  - Notation: $x^s_{11} = P(\mathcal{M}(11) = S)$
  - Constraints on rows are:

$$\begin{aligned}
p^2 x^s_{11} + (1-p)^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01} \\
(1-p)^2 x^s_{11} + p^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01} \\
p^2 x^s_{10} + (1-p)^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00} \\
(1-p)^2 x^s_{10} + p^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00}
\end{aligned}$$

- All inputs with same parity are grouped together!

Randomized Response

- Constraints have interpretation in terms of protecting parity.
- If attacker believes each bit b_i has $P(b_i = 1) \geq p$ or $P(b_i = 0) \geq p$
    - Then attacker has some (tiny amount of) certainty about parity of each subset of dataset
- After seeing output, none of the relative beliefs about parity will change.
    - For any subset of the data, If $P(\text{parity=even}) > P(\text{parity=odd})$ then
    - $P(\text{parity=even} \mid \text{output}) \geq P(\text{parity=odd} \mid \text{output})$
    - and vice versa.
- Utility: it looks like we are protecting too much.
    - But what can we do?
    - Relax privacy definition
    - Tool: Fourier-Motzkin elimination
        - Analogue of Guass-Jordan elimination for linear inequalities.

Fourier-Motzkin Elimination

$$
\begin{aligned}
p^2 x^s_{11} + (1-p)^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01} \\
(1-p)^2 x^s_{11} + p^2 x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01} \\
p^2 x^s_{10} + (1-p)^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00} \\
(1-p)^2 x^s_{10} + p^2 x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00}
\end{aligned}
$$

Fourier-Motzkin Elimination

$$\begin{aligned} p^2 x_{11}^s + (1-p)^2 x_{00}^s &\geq p(1-p)x_{10}^s + p(1-p)x_{01}^s \\ p^2 x_{10}^s + (1-p)^2 x_{01}^s &\geq p(1-p)x_{11}^s + p(1-p)x_{00}^s \end{aligned}$$

Fourier-Motzkin Elimination

$$\begin{aligned} p^2 x^s_{11} + (1-p)^2 x^s_{00} &\geq p(1-p) x^s_{10} + p(1-p) x^s_{01} \\ p^2 x^s_{10} + (1-p)^2 x^s_{01} &\geq p(1-p) x^s_{11} + p(1-p) x^s_{00} \end{aligned}$$

$$\frac{p}{1-p} x^s_{11} + \frac{1-p}{p} x^s_{00} - x^s_{01} \geq x^s_{10}$$

$$x^s_{10} \geq \frac{1-p}{p} x^s_{11} + \frac{1-p}{p} x^s_{00} - \frac{(1-p)^2}{p^2} x^s_{01}$$

Fourier-Motzkin Elimination

$$\begin{aligned} p^2x_{11}^s + (1-p)^2x_{00}^s &\geq p(1-p)x_{10}^s + p(1-p)x_{01}^s \\ p^2x_{10}^s + (1-p)^2x_{01}^s &\geq p(1-p)x_{11}^s + p(1-p)x_{00}^s \end{aligned}$$

$$\frac{p}{1-p}x_{11}^s + \frac{1-p}{p}x_{00}^s - x_{01}^s \geq x_{10}^s$$

$$x_{10}^s \geq \frac{1-p}{p}x_{11}^s + \frac{1-p}{p}x_{00}^s - \frac{(1-p)^2}{p^2}x_{01}^s$$

$$P(\mathcal{M}(01)=s) = \boxed{x_{01}^s \leq \frac{p}{1-p}x_{11}^s} = \frac{p}{1-p}P(\mathcal{M}(11)=s)$$

Fourier-Motzkin Elimination

$$\begin{aligned} p^2x^s_{11} + (1-p)^2x^s_{00} &\geq p(1-p)x^s_{10} + p(1-p)x^s_{01} \\ p^2x^s_{10} + (1-p)^2x^s_{01} &\geq p(1-p)x^s_{11} + p(1-p)x^s_{00} \end{aligned}$$

$$\frac{p}{1-p}x^s_{11} + \frac{1-p}{p}x^s_{00} - x^s_{01} \geq x^s_{10}$$

$$x^s_{10} \geq \frac{1-p}{p}x^s_{11} + \frac{1-p}{p}x^s_{00} - \frac{(1-p)^2}{p^2}x^s_{01}$$

$$P(\mathcal{M}(01)=s) = \boxed{x^s_{01} \leq \frac{p}{1-p}x^s_{11}} = \frac{p}{1-p}P(\mathcal{M}(11)=s)$$

- One of the ϵ-differential privacy constraints ($\epsilon = \frac{p}{1-p}$)
- Can get all of them using FM-elimination

Outline

Analysis of FRAPP

- Similar to PRAM
- Like randomized response but data is not binary.
- (simplified) idea:
  - Each tuple is perturbed using a matrix $\mathcal{P}$.
  - p_{ij} = probability value i gets perturbed to value j.
  - (simplification) $\mathcal{P}$ is a symmetric matrix
  - (simplification) each $p_{ij} \geq c$ (a privacy parameter)
- Protects a general notion of privacy
  - For each person, choose one tuple value to be the 1 other tuple values are 0
  - If attacker believes each person has a tuple value with prior probability $\geq p^*$ then relative belief in parity will not change.

Questions?

Lars Backstrom, Cynthia Dwork, and Jon Kleinberg.
Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography.
In *WWW*, 2007.

Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V. Pearson, Dietrich A. Stephan, Stanley F. Nelson, and David W. Craig.
Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays.
PLoS Genet, 4(8), 08 2008.

Daniel Kifer.
Attacks on privacy and de finetti's theorem.
In *SIGMOD*, 2009.

Daniel Kifer and Bing-Rong Lin.
An axiomatic view of statistical privacy and utility.
To appear in Journal of Privacy and Confidentiality.

J.P. Reiter.
Estimating risks of identification disclosure for microdata.
Journal of the American Statistical Association, 100:1103 – 1113, 2005.

Raymond Wong, Ada Fu, Ke Wang, and Jian Pei.
Minimality attack in privacy preserving data publishing.
In *VLDB*, 2007.