Private Disclosure of Information in Health Tele-monitoring

Daniel Aranki, Ruzena Bajcsy

University of California, Berkeley
{daranki,bajcsy}@eecs.berkeley.edu

May 7, 2015

Motivation


Intervention
Patient
Activity
Surveys
Reporting
Transmission
Analysis

Example

1. Patient Bob wants to update his physician Alice about his Body Mass Index (BMI) and weight (x).

Example

1. Patient Bob wants to update his physician Alice about his Body Mass Index (BMI) and weight (x).
2. Alice already knows the BMI category of Bob (c).

Example

1. Patient Bob wants to update his physician Alice about his Body Mass Index (BMI) and weight (x).
2. Alice already knows the BMI category of Bob (c).
3. Alice and Bob want to keep the BMI category c private from Eve, a passive eavesdropper, after observing the communication.

Setting and Threat Model

Setting

Disclosed Identity

The identity of the sender (s) is attached to each disclosed piece of information.

Setting and Threat Model

Setting

Disclosed Identity

The identity of the sender (s) is attached to each disclosed piece of information.

Intended Recipient's Knowledge

The sender belongs to a class (c) that is known to the intended recipient.

Setting and Threat Model

Setting

Disclosed Identity

The identity of the sender (s) is attached to each disclosed piece of information.

Intended Recipient's Knowledge

The sender belongs to a class (c) that is known to the intended recipient.

Threat Model

Adversary is a passive man in the middle interested in inferring the class c of the sender s based on the disclosed information.

Idea

The sender discloses an encoded version z of x, where the encoding depends on her class c.

Objectives

Decoding Condition

The intended recipient can make full use of the sent information z, i.e. obtain the original message x from the transmitted message z.

Objectives

Decoding Condition

The intended recipient can make full use of the sent information z, i.e. obtain the original message x from the transmitted message z.

Hiding Class Condition

The adversary's ability to make inference about c given s, based on the sent information z is minimized.

Some Definitions

- $\mathcal{S}$ is the set of senders' identities

Some Definitions

- $\mathcal{S}$ is the set of senders' identities
- Σ is the set of senders' classes

Some Definitions

- $\mathcal{S}$ is the set of senders' identities
- Σ is the set of senders' classes
- $\mathcal{I}$ is the set of pieces of information

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function)

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function) (Equivalent to $R : \Sigma \times \mathcal{I} \to \mathcal{I}$ being injective in the second argument)

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function) (Equivalent to $R : \Sigma \times \mathcal{I} \to \mathcal{I}$ being injective in the second argument)

Sending Information

- Sender $s \in \mathcal{S}$ (from class $c \in \Sigma$) wants to send information $x \in \mathcal{I}$.

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function) (Equivalent to $R : \Sigma \times \mathcal{I} \to \mathcal{I}$ being injective in the second argument)

Sending Information

- Sender $s \in \mathcal{S}$ (from class $c \in \Sigma$) wants to send information $x \in \mathcal{I}$.
- Let the sender encode $z = [R(c)](x)$, and send z.

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function) (Equivalent to $R : \Sigma \times \mathcal{I} \to \mathcal{I}$ being injective in the second argument)

Sending Information

- Sender $s \in \mathcal{S}$ (from class $c \in \Sigma$) wants to send information $x \in \mathcal{I}$.
- Let the sender encode $z = [R(c)](x)$, and send z.

Receiving Information

- The intended recipient knows the identity of s and her class c.

The Process

The Disclosure Process

Let $R : \Sigma \to \mathcal{I}^{\mathcal{I}}$ (Privacy Mapping Function) (Equivalent to $R : \Sigma \times \mathcal{I} \to \mathcal{I}$ being injective in the second argument)

Sending Information

- Sender $s \in \mathcal{S}$ (from class $c \in \Sigma$) wants to send information $x \in \mathcal{I}$.
- Let the sender encode $z = [R(c)](x)$, and send z.

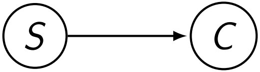
Receiving Information

- The intended recipient knows the identity of s and her class c.
- The intended recipient then can decode $x \leftarrow [R(c)]^{I}(z)$.

Statistical Graphical Model

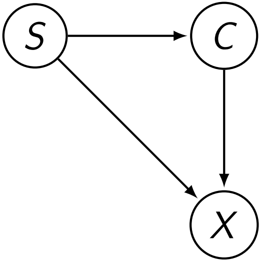
$$P(S)$$

Statistical Graphical Model



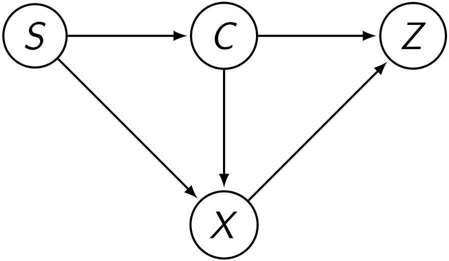
$$P(C|S)$$

Statistical Graphical Model



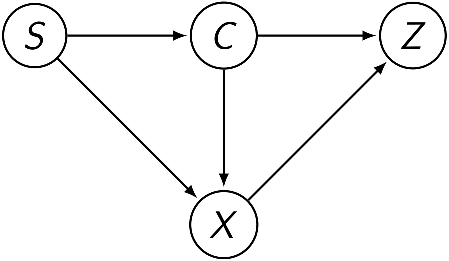
$$P(X|C,S)$$

Statistical Graphical Model



$$p(Z = z | X = x, C = c) \triangleq \delta\left(z - [R(c)]\,(x)\right)$$

Statistical Graphical Model



$$P(S) \quad P(C|S) \quad P(X|C,S)$$

Formulation of Problem

$$\text{minimize } I(C, Z|S; R)$$
$$\text{w.r.t } R \in \left(\Sigma \to \mathcal{I}^{\mathcal{I}}\right)$$

Formulation of Problem

$$\text{minimize } I(C, Z|S; R)$$
$$\text{w.r.t } R \in \left(\Sigma \to \mathcal{I}^{\mathcal{I}}\right)$$

1. Properties?

Formulation of Problem

$$\text{minimize } I(C, Z|S; R)$$
$$\text{w.r.t } R \in \left(\Sigma \to \mathcal{I}^{\mathcal{I}}\right)$$

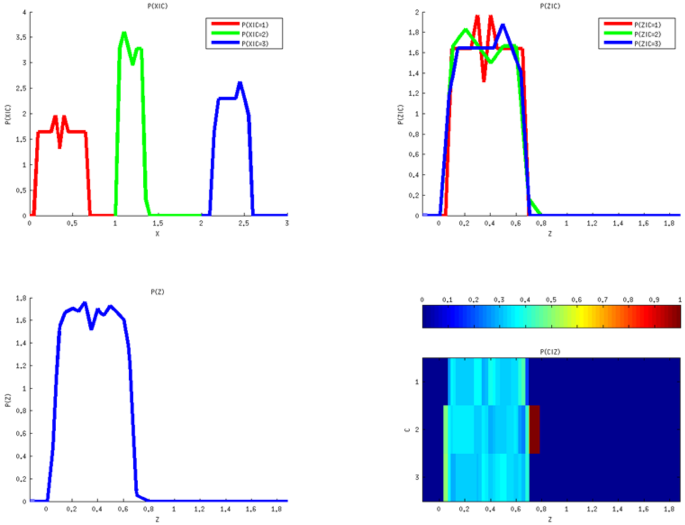
1. Properties?
2. How do we learn such a privacy mapping function, R?

Theorem 1

If there exists a privacy mapping function R such that $p(Z = z|C = c, S = s; R) = f(z, s)$ for all $c \in \Sigma$ then:

1. $I(C, Z|S; R) = 0$ *(global optimum)*
2. $p(C = c|Z = z, S = s; R) = p(C = c|S = s)$ *(Bayesian updates prevented)*

Intuition



Gaussian Information

Theorem 2

If $X|C = c, S = s \sim N(\mu_c, \Sigma_c)$ *(Normal distribution) for every* $c \in \Sigma$ *and* $s \in \mathcal{S}$*, then* $[R(c)](x) = \Sigma_c^{-\frac{1}{2}} \cdot (x - \mu_c)$ *yields* $I(C, Z|S; R) = 0$ *and "prevents Bayesian updates".*

Exponentially Distributed Information

Theorem 3

If $X|C = c, S = s \sim Exp(\lambda_c)$ (Exponential distribution) for every $c \in \Sigma$ and $s \in \mathcal{S}$, then $[R(c)](x) = \lambda_c x$ yields $I(C, Z|S; R) = 0$ and "prevents Bayesian updates".

Gamma Distributed Information

Theorem 4

If $X|C = c, S = s \sim Gamma(k, \theta_c)$ (Gamma distribution with shape and scale parameters) for every $c \in \Sigma$ and $s \in \mathcal{S}$, then $[R(c)](x) = \frac{x}{\theta_c}$ yields $I(C, Z|S; R) = 0$ and "prevents Bayesian updates".

Uniform Information

Theorem 5

If $X|C = c, S = s \sim U(a_c, b_c)$ *(Uniform distribution) for every* $c \in \Sigma$ *and* $S \in \mathcal{S}$*, then* $[R(c)](x) = \frac{x-a_c}{b_c-a_c}$ *yields* $I(C, Z|S; R) = 0$ *and "prevents Bayesian updates".*

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \rightarrow R(\cdot; \theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot; \theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot; \theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"
    - $p(S)$ is uniform.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot;\theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"
   - $p(S)$ is uniform.
   - $p(C|S = s)$ is invariant in s.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot;\theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"
   - $p(S)$ is uniform.
   - $p(C|S=s)$ is invariant in s.
   - $p(X|C=c, S=s)$ is invariant in s.

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot;\theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"
   - $p(S)$ is uniform.
   - $p(C|S = s)$ is invariant in s.
   - $p(X|C = c, S = s)$ is invariant in s.
3. minimize $I(C, Z; R(\cdot;\theta))$ w.r.t. $\theta \in \Theta$

The Learning Problem

Hard problem:

1. $I(C, Z|S; R)$ is non-convex in R.
2. Search space is hard to compute over.

MATLAB Implementation as a toolbox:

1. Parametrize $R(\cdot) \to R(\cdot;\theta)$ where $\theta \in \Theta$ a (vector) of parameter(s) from a parameter space.
2. Treat all subjects as "equal"
   - $p(S)$ is uniform.
   - $p(C|S = s)$ is invariant in s.
   - $p(X|C = c, S = s)$ is invariant in s.
3. minimize $I(C, Z; R(\cdot;\theta))$ w.r.t. $\theta \in \Theta$
4. Non-parametric modeling of $p(X|C)$ and $p(C)$

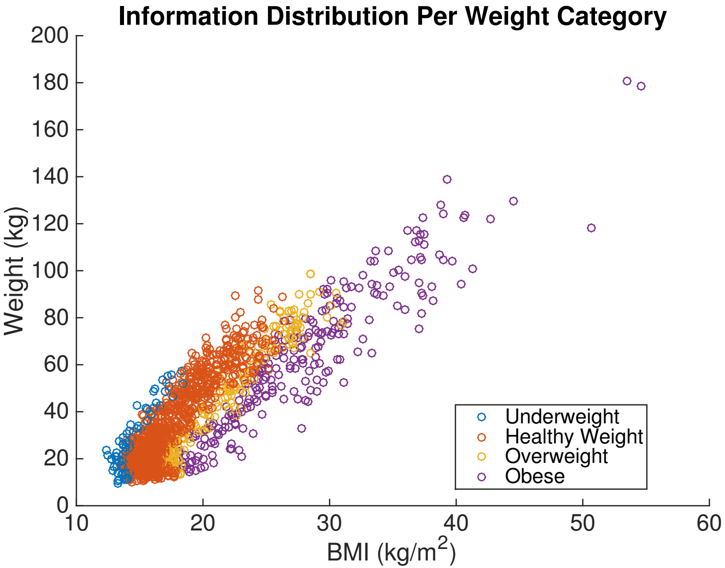
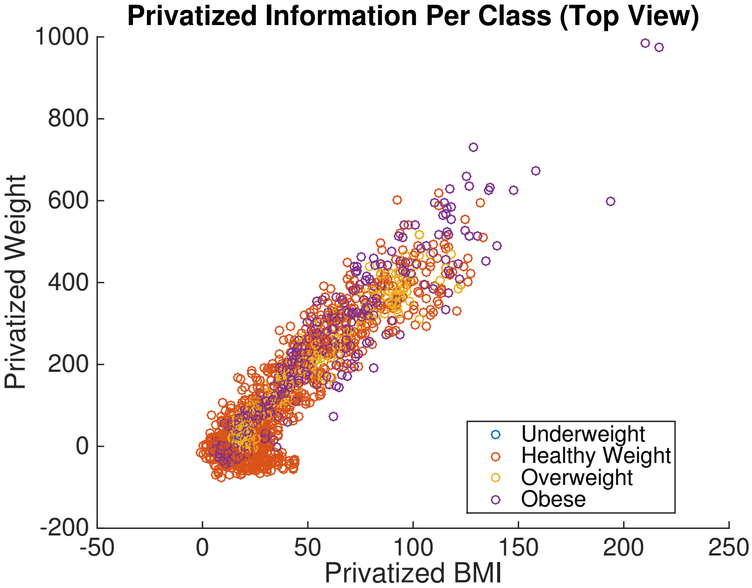

Information Distribution Per Weight Category
Weight (kg)
BMI (kg/m^2)
0
20
40
60
80
100
120
140
160
180
200
10
20
30
40
50
60
Underweight
Healthy Weight
Overweight
Obese

Table: Confusion Matrix. UW = Underweight, HW = Healthy Weight, OW = Overweight, OB = Obese

		Ground Truth Category			
		UW	**HW**	**OW**	**OB**
Predicted Category	**UW**	**47**	20	0	0
	HW	14	**1203**	66	1
	OW	0	45	**194**	47
	OB	0	2	37	**308**

$$trace(\text{Confusion Matrix})/sum(\text{Confusion Matrix}) = 88.31\%$$

```
pdi_begin
   % data/information space
   pdi_dimension BMI 0:2:60;
   pdi_dimension weight 0:5:180;
   % define classes
   pdi_class underweight healthy_weight overweight obese
   % provide data
   pdi_datapoints underweight fv_uw
   pdi_datapoints healthy_weight fv_hw
   pdi_datapoints overweight fv_ow
   pdi_datapoints obese fv_ob
   % parameter space
   pdi_var shift(pdi_nrdimensions, pdi_nrclasses);
   pdi_var scale(pdi_nrdimensions, pdi_nrclasses);
   % z = scale.*(x-shift)
   pdi_reference @(x, cn) bsxfun(@times, bsxfun(@minus,
       x, shift(:,cn)), scale(:,cn));
   % such that
   scale(:,1) == 1; % entry-wise
   shift(:,1) == 0; % entry-wise
   scale>=.1; % entry-wise
   shift>=0; % entry-wise
pdi_end
```


Privatized Information Per Class (Top View)
Privatized Weight
Privatized BMI
1000
800
600
400
200
0
-200
-50
0
50
100
150
200
250
Underweight
Healthy Weight
Overweight
Obese

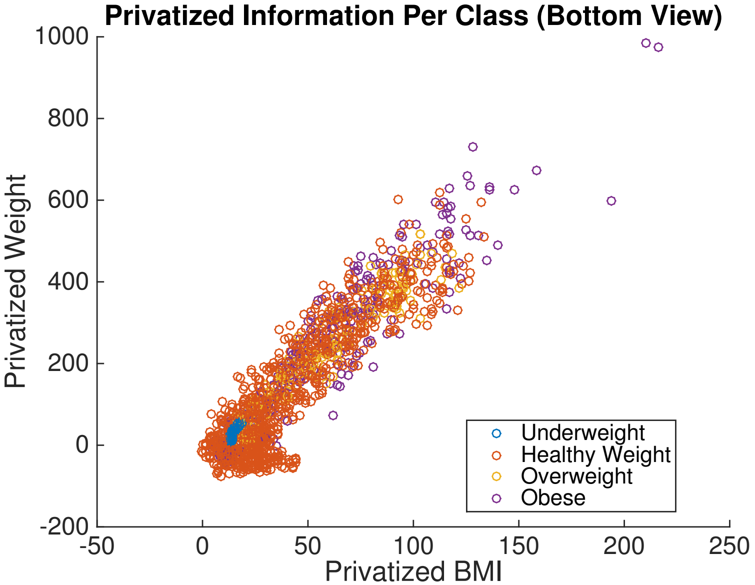

Privatized Information Per Class (Bottom View)
Privatized Weight
Privatized BMI
1000
800
600
400
200
0
-200
-50
0
50
100
150
200
250
Underweight
Healthy Weight
Overweight
Obese

Table: Confusion Matrix After Privatizing. UW = Underweight, HW = Healthy Weight, OW = Overweight, OB = Obese

		Ground Truth Category			
		UW	HW	OW	OB
Predicted Category	UW	**48**	14	8	5
	HW	13	**1217**	**276**	**290**
	OW	0	25	13	29
	OB	0	14	0	32

$$trace(\text{Confusion Matrix})/sum(\text{Confusion Matrix}) = 66.03\%$$

Table: Confusion Matrix After Privatizing. UW = Underweight, HW = Healthy Weight, OW = Overweight, OB = Obese

Predicted Category		Ground Truth Category: UW	HW	OW	OB
	UW	**48**	14	8	5
	HW	13	**1217**	**276**	**290**
	OW	0	25	13	29
	OB	0	14	0	32

$$trace(\text{Confusion Matrix})/sum(\text{Confusion Matrix}) = 66.03\%$$

from 88.31%

Table: Confusion Matrix After Privatizing. UW = Underweight, HW = Healthy Weight, OW = Overweight, OB = Obese

Predicted Category		Ground Truth Category UW	HW	OW	OB
	UW	**48**	14	8	5
	HW	13	**1217**	**276**	**290**
	OW	0	25	13	29
	OB	0	14	0	32

$$trace(\text{Confusion Matrix})/sum(\text{Confusion Matrix}) = 66.03\%$$

from 88.31%
lower bound: $\#\text{HW}/sum(\text{Confusion Matrix}) = 64.01\%$

Future Directions

- Bounds on privacy.

Future Directions

- Bounds on privacy.
- Sensitivity analysis.

Future Directions

- Bounds on privacy.
- Sensitivity analysis.
- Relaxing the assumption of perfect classification knowledge for the intended recipient.

Future Directions

- Bounds on privacy.
- Sensitivity analysis.
- Relaxing the assumption of perfect classification knowledge for the intended recipient.
- Markov-type relaxation.

Future Directions

- Bounds on privacy.
- Sensitivity analysis.
- Relaxing the assumption of perfect classification knowledge for the intended recipient.
- Markov-type relaxation.
- Study the relationships between $I(C, Z|S)$ and $I(X, Z|S)$.

Future Directions

- Bounds on privacy.
- Sensitivity analysis.
- Relaxing the assumption of perfect classification knowledge for the intended recipient.
- Markov-type relaxation.
- Study the relationships between $I(C, Z|S)$ and $I(X, Z|S)$.
- Parametric modeling of $p(X|C)$ for learning.

References I

- Daniel Aranki and Ruzena Bajcsy, Differential disclosure of information, Tech. Report UCB/EECS-2014-47, EECS Department, University of California, Berkeley, May 2014.
- ______, Private disclosure of information in health tele-monitoring, arXiv preprint arXiv:1504.07313 (2015).

Acknowledgments

- Gregorij Kurillo
- Yusuf Erol
- Arash Nourian
- This work was supported in part by TRUST, Team for Research in Ubiquitous Secure Technology, which receives funding support for the National Science Foundation (NSF award number CCF-0424422).