Privately Solving Linear Programs

Justin Hsu[1] Aaron Roth[1]
Tim Roughgarden[2] Jonathan Ullman[3]

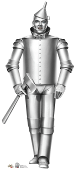
[1]University of Pennsylvania

[2]Stanford University

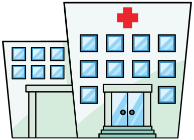
[3]Harvard University

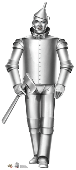
July 8th, 2014

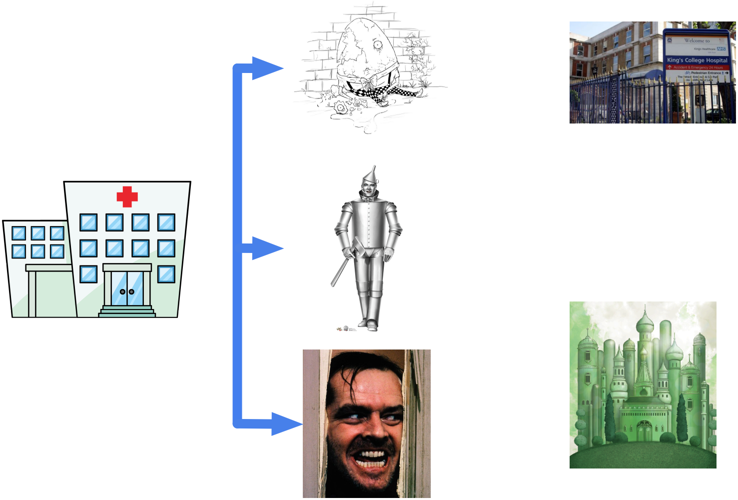
A motivating example



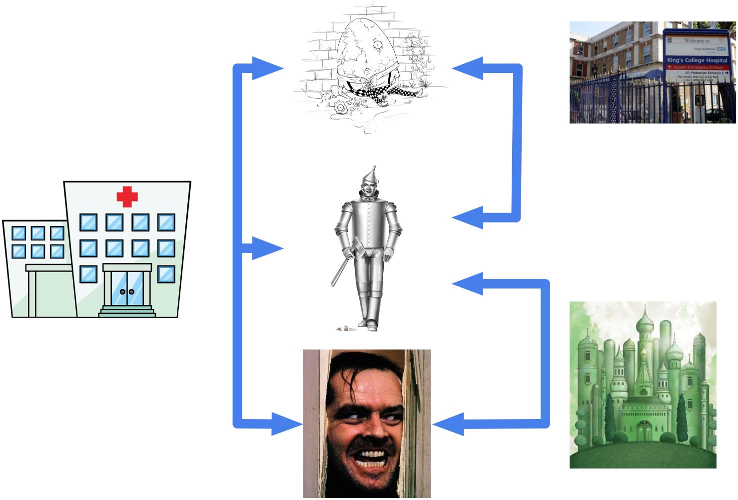
A motivating example

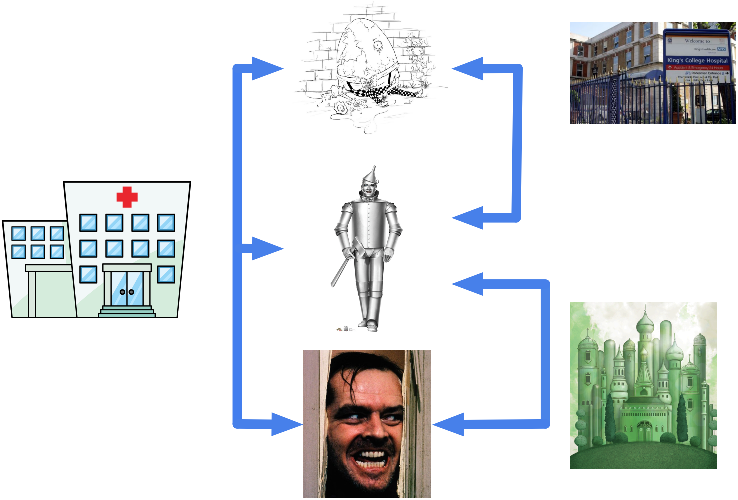





Welcome to
King's College Hospital
Pedestrian Entrance

A motivating example





How to pick hospitals, privately?

Set cover

- Approximate solution by solving a linear program (LP):

$$\begin{aligned}
&\text{minimize } \sum_{S} x_S \\
&\text{such that } \sum_{S \ni i} x_S \geq 1 \quad \text{for every person } i \\
&0 \leq x_S \leq 1 \quad \text{for every set } S
\end{aligned}$$

Set cover

- Approximate solution by solving a linear program (LP):

$$\text{minimize } \sum_{S} x_S$$

$$\text{such that } \sum_{S \ni i} x_S \geq 1 \quad \text{for every person } i$$

$$0 \leq x_S \leq 1 \quad \text{for every set } S$$

One person,
one constraint

Set cover (Private?)

- Approximate solution by solving a linear program (LP):

$$\text{minimize } \sum_{S} x_S$$

$$\text{such that } \sum_{S \ni i} x_S \geq 1 \quad \text{for every person } i$$

$$0 \leq x_S \leq 1 \quad \text{for every set } S$$

One person,
one constraint

Set cover (Private?)

- Approximate solution by solving a linear program (LP):

$$\text{minimize } \sum_{S} x_S$$

$$\text{such that } \sum_{S \ni i} x_S \geq 1 \quad \text{for every person } i$$

$$0 \leq x_S \leq 1 \quad \text{for every set } S$$

One person,
one constraint

More generally...

- Solving LPs is a very common tool
- Can we solve LPs privately?

The plan

- LPs and privacy
- “Neighboring” LPs
- A private LP solver
- The state of private LPs

General form

$$
\begin{aligned}
&\text{maximize } c^\top x \\
&\text{such that } \begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{md} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \leq \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}
\end{aligned}
$$

General form

$$
\begin{aligned}
&\text{maximize } c^\top x \\
&\text{such that } \begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{md} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \leq \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}
\end{aligned}
$$

We'll assume

- Optimum objective value known
- Just want to find feasible solution

General form

find x

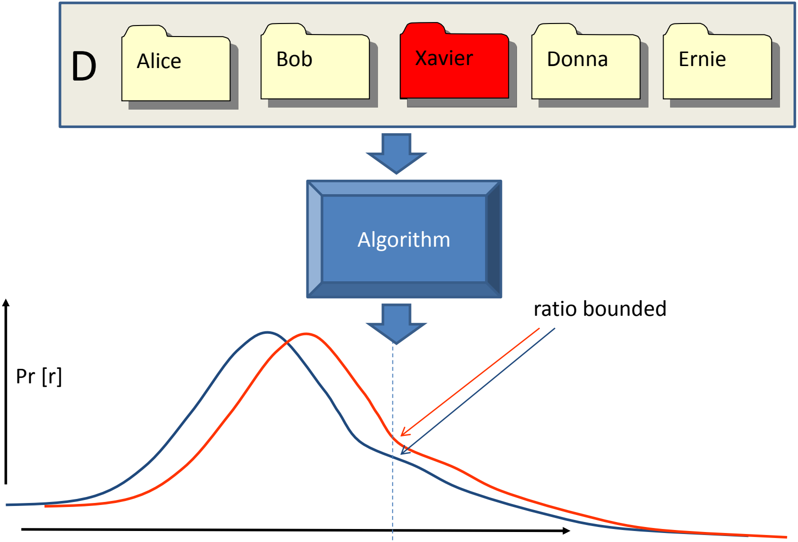
$$\text{maximize } c^\top x$$

$$\text{such that } \begin{pmatrix} a_{11} & \cdots & a_{1d} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{md} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} \leq \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$$

We'll assume

- Optimum objective value known
- Just want to find feasible solution

Differential privacy [DMNS]



Definition (DMNS)

Let M be a randomized mechanism from databases to range $\mathcal{R}$, and let D, D' be databases differing in one record. M is (ε, δ)-differentially private if for every $r \in \mathcal{R}$,

$$\Pr[M(D) = r] \leq e^{\varepsilon} \cdot \Pr[M(D') = r] + \delta.$$

Definition (DMNS)

Let M be a randomized mechanism from databases to range $\mathcal{R}$, and let D, D' be databases differing in one record. M is <u>(ε, δ)-differentially private</u> if for every $r \in \mathcal{R}$,

$$\Pr[M(D) = r] \leq e^{\varepsilon} \cdot \Pr[M(D') = r] + \delta.$$

For us

- database $\Longrightarrow$ linear program

In words...

Definition (DMNS)

Let M be a randomized mechanism from databases to range $\mathcal{R}$, and let D, D' be databases differing in one record. M is <u>(ε, δ)-differentially private</u> if for every $r \in \mathcal{R}$,

$$\Pr[M(D) = r] \leq e^{\varepsilon} \cdot \Pr[M(D') = r] + \delta.$$

For us

- database $\Longrightarrow$ linear program
- differing in one record $\Longrightarrow$??

Definition (DMNS)

Let M be a randomized mechanism from databases to range $\mathcal{R}$, and let D, D' be databases differing in one record. M is <u>(ε, δ)-differentially private</u> if for every $r \in \mathcal{R}$,

$$\Pr[M(D) = r] \leq e^{\varepsilon} \cdot \Pr[M(D') = r] + \delta.$$

For us

- database $\Longrightarrow$ linear program
- differing in one record $\Longrightarrow$??

$$\boxed{\text{What are “neighboring” LPs?}}$$

Define what data can change on "neighboring" LPs

- One row of constraint matrix
- One column of constraint matrix
- The objective
- The scalars

Define what data can change on “neighboring” LPs

- One row of constraint matrix
- One column of constraint matrix
- The objective
- The scalars

Qualitatively different results (and algorithms)

Prior work

- Known iterative solvers for LPs (multiplicative weights [PST])
- Private version of this technique used for query release [HR]
- Also used for analyst private query release [HRU]

Detour: Some context

Prior work

- Known iterative solvers for LPs (multiplicative weights [PST])
- Private version of this technique used for query release [HR]
- Also used for analyst private query release [HRU]

Our contribution

- Observe the private query release problem is equivalent to solving a LP under "scalar privacy"
- Extend known techniques to additional classes of private LPs

Define what data can change on "neighboring" LPs

- One row of constraint matrix
- One column of constraint matrix
- The objective
- The scalars

$\boxed{\text{Qualitatively different results (and algorithms)}}$

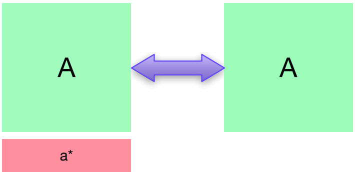
Define what data can change on "neighboring" LPs

- One row of constraint matrix
- One column of constraint matrix
- The objective
- The scalars

$\boxed{\text{Qualitatively different results (and algorithms)}}$

"Constraint privacy"

- Neighboring databases have constraint matrices:



- All other data unchanged
- Hide presence or absence of a single constraint
- Example: private set cover LP

Iterative LP solver [PST]

- Maintain distribution over constraints
- In a loop:
  - Find point satisfying (a single) “weighted” constraint
  - Reweight to emphasize unsatisfied constraints
- Repeat

Iterative LP solver [PST]

- Maintain distribution over constraints
- In a loop:
  - Find point satisfying (a single) "weighted" constraint
  - Reweight to emphasize unsatisfied constraints

MW
update rule

- Repeat

Iterative LP solver [PST]

- Maintain distribution over constraints
- In a loop:
  - Find point satisfying (a single) “weighted” constraint
  - Reweight to emphasize unsatisfied constraints

MW
update rule

- Repeat
- Average of points is approximately feasible solution

Constraint privacy?

Recall: hide presence or absence of a single constraint

- Select point satisfying weighted constraint privately
- Adapt known algorithms from privacy literature

Recall: hide presence or absence of a single constraint

- Select point satisfying weighted constraint privately
- Adapt known algorithms from privacy literature

One more key idea

- Cap weight on any single constraint by projecting distribution
- Limit influence of a single constraint on chosen point
- Pay in the accuracy...

Two ways of being inaccurate

- Solution satisfies most constraint to within additive α
- The other constraints can be arbitrarily infeasible
- Precise theorem depends on how points satisfying the weighted constraints are chosen, specific LP, etc...

How good is the solution?

Two ways of being inaccurate

- Solution satisfies most constraint to within additive α
- The other constraints can be arbitrarily infeasible
- Precise theorem depends on how points satisfying the weighted constraints are chosen, specific LP, etc...

Theorem

Let OPT *be the size of the optimal cover. There is an* (ε, δ)*-constraint private algorithm that with high probability produces a fractional collection of sets covering all but* s *people to at least* $1 - \alpha$*, where*

$$s = \tilde{O}\left(\frac{\mathrm{OPT}^2 \log^{1/2}(1/\delta)}{\alpha^2 \cdot \varepsilon}\right).$$

Why not all satisfy all constraints?

- Not hard to see: can't hope to hide presence of a constraint if all constraints must be approximately satisfied

Why not all satisfy all constraints?

- Not hard to see: can't hope to hide presence of a constraint if all constraints must be approximately satisfied

Even more discouraging results...

Why not all satisfy all constraints?

- Not hard to see: can't hope to hide presence of a constraint if all constraints must be approximately satisfied

Even more discouraging results...

- Objective private LPs? Impossible.

Why not all satisfy all constraints?

- Not hard to see: can't hope to hide presence of a constraint if all constraints must be approximately satisfied

Even more discouraging results...

- Objective private LPs? Impossible.
- Column private LPs? Impossible.

Lower bounds

Why not all satisfy all constraints?

- Not hard to see: can't hope to hide presence of a constraint if all constraints must be approximately satisfied

Even more discouraging results...

- Objective private LPs? Impossible.
- Column private LPs? Impossible.
- Scalar private LPs? Impossible.

What is there to do?

Needed: finer distinctions

- LPs encode an extremely broad range of problems
- Little hope to solve all LPs privately, for any notion of privacy
- Lower bounds are all for very simple, "unnatural" LPs
- Focus on smaller classes of LPs/neighboring LPs

A simple distinction: sensitivity

Bounding the degree of change

- In privacy for databases, number of records n
- As n increases, accuracy often improves
- Adapt same idea to private LPs

A simple distinction: sensitivity

Bounding the degree of change

- In privacy for databases, number of records n
- As n increases, accuracy often improves
- Adapt same idea to private LPs

Distinguishing two kinds of privacy guarantees

- High sensitivity: degree of change constant in n
- Low sensitivity: degree of change decreasing in n
- Example: LP data derived from averages over a population

Future directions: Other possible classifications?

Joint Differential Privacy [KPRU]

- Variables and data partitioned among different agents
- No need to publish the entire solution

Future directions: Other possible classifications?

Joint Differential Privacy [KPRU]

- Variables and data partitioned among different agents
- No need to publish the entire solution

Other classifications?

- So far: modify privacy guarantee, definition of neighboring...
- Structural properties of LPs to aid private solvability?

The state of private LPs

Location of change	High sensitivity	Low sensitivity
Objective	No	Yes
Scalars	No	Yes
Row of constraints	Yes	Yes
Column of constraints	No	Yes

Table : Efficient, accurate, private solvability

More directions

- Huge literature on techniques for non-privately solving LPs (primal-dual, interior point methods, etc.)
- Can any of these techniques be made private?

Privately Solving Linear Programs

Justin Hsu[1] Aaron Roth[1]
Tim Roughgarden[2] Jonathan Ullman[3]

[1]University of Pennsylvania

[2]Stanford University

[3]Harvard University

July 8th, 2014