Computational Complexity & Differential Privacy

Salil Vadhan
Harvard University

Joint works with Cynthia Dwork, Kunal Talwar, Andrew McGregor, Ilya Mironov, Moni Naor, Omkant Pandey, Toni Pitassi, Omer Reingold, Guy Rothblum, Jon Ullman

Computational Complexity

When do computational resource constraints change what is possible?

Examples:

- Computational Learning Theory [Valiant `84]:
  small VC dimension $\not\Rightarrow$ learnable with efficient algorithms
  (bad news)

- Cryptography [Diffie & Hellman `76]: don't need long shared secrets against a computationally bounded adversary
  (good news)

Today: Computational Complexity in Differential Privacy

I. Computationally bounded curator

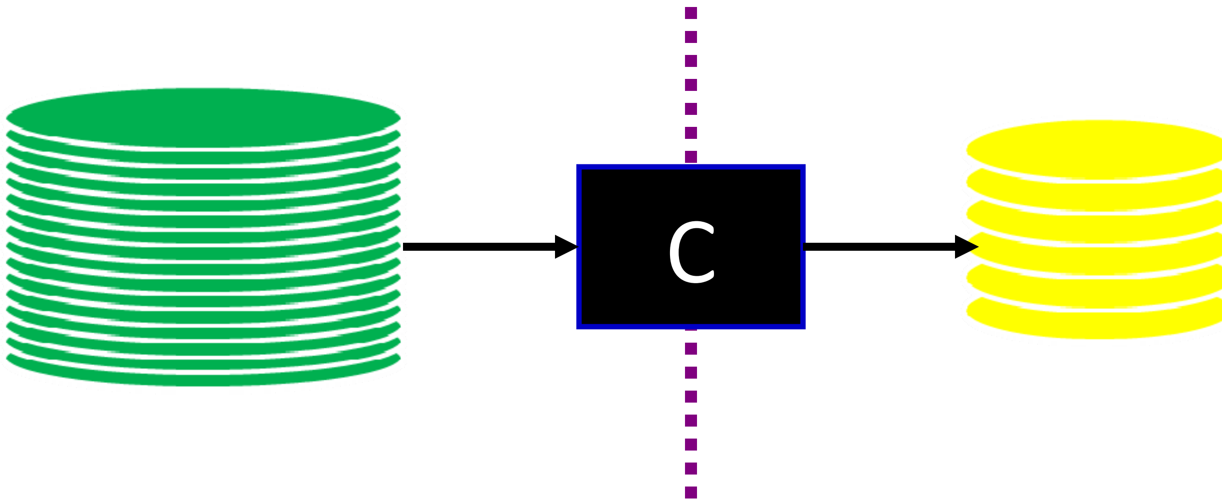
- Makes differential privacy harder
- Differentially private & accurate synthetic data infeasible to construct
- Open: release other types of summaries/models?

II. Computationally bounded adversary

- Makes differential privacy easier
- Provable gain in accuracy for 2-party protocols
  (e.g. for estimating Hamming distance)

PART I: COMPUTATIONALLY BOUNDED CURATORS

Cynthia's Dream: Noninteractive Data Release



Original Database D

Sanitization C(D)

Noninteractive Data Release: Desidarata

- (ε,δ)-differential privacy:
  for every D_1, D_2 that differ in one row and every set T,
  $$\Pr[C(D_1) \in T] \le \exp(\varepsilon) \cdot \Pr[C(D_2) \in T] + \delta,$$
  with δ negligible

- Utility: C(D) allows answering many questions about D

- Computational efficiency: C is polynomial-time computable.

Utility: Counting Queries

- $D = (x_1,\dots,x_n) \in X^n$
- $P = \{ \pi : X \rightarrow \{0,1\}\}$
- For any $\pi \in P$, want to estimate (from C(D)) counting query

$$\pi(D) := (\sum_i \pi(x_i))/n$$

  within accuracy error $\pm \alpha$

- Example:
  $X = \{0,1\}^d$
  P = {conjunctions on $\leq$ k variables}
  Counting query = k-way marginal

  e.g. What fraction of people in D smoke and have cancer?

>35	Smoker?	Cancer?
0	1	1
1	1	0
1	0	1
1	1	1
0	1	0
1	1	1

Form of Output

- Ideal: C(D) is a synthetic dataset
  - $\forall \pi \in P \ |\pi(C(D))-\pi(D)| \leq \alpha$
  - Values consistent
  - Use existing software

>35	Smoker?	Cancer?
1	0	0
0	1	1
0	1	0
0	1	1
0	1	0

- Alternatives?
  - Explicit list of |P| answers (e.g. contingency table)
  - Median of several synthetic datasets [RR10]
  - Program M s.t. $\forall \pi \in P \ |M(\pi)-\pi(D)| \leq \alpha$

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N		

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \rightarrow \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N	poly(n,|P|)	poly(n,d^k)

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \rightarrow \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N	poly(n,|P|)	$\text{poly}(n,d^k)$
[BDCKMT07]		$O(d^k/\alpha\varepsilon)$	Y		$\text{poly}(n,2^d)$

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \rightarrow \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N	poly(n,|P|)	$poly(n,d^k)$
[BDCKMT07]		$\tilde{O}((2d)^k/\alpha\varepsilon)$	Y		$poly(n,2^d)$
[BLR08]	$O(d\cdot\log\|P\|/\alpha^3\varepsilon)$	$\tilde{O}(dk/\alpha^3\varepsilon)$	Y		

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \rightarrow \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N	poly(n,|P|)	poly(n,d^k)
[BDCKMT07]		$\tilde{O}((2d)^k/\alpha\varepsilon)$	Y		poly(n,2^d)
[BLR08]	$O(d\cdot\log\|P\|/\alpha^3\varepsilon)$	$\tilde{O}(dk/\alpha^3\varepsilon)$	Y	qpoly(n,|P|,2^d)	qpoly(n,2^d)

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \to \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Positive Results

	minimum database size			computational complexity	
reference	general P	k-way marginals	synthetic	general P	k-way marginals
[DN03,DN04, BDMN05]	$O(\|P\|^{1/2}/\alpha\varepsilon)$	$O(d^{k/2}/\alpha\varepsilon)$	N	poly(n,|P|)	poly(n,d^k)
[BDCKMT07]		$\tilde{O}((2d)^k/\alpha\varepsilon)$	Y		poly(n,2^d)
[BLR08]	$O(d \cdot \log\|P\|/\alpha^3\varepsilon)$	$\tilde{O}(dk/\alpha^3\varepsilon)$	Y	qpoly(n,|P|,2^d)	qpoly(n,2^d)
[DNRRV09, DRV10]	$O(d \cdot \log^2\|P\|/\alpha^2\varepsilon)$	$\tilde{O}(dk^2/\alpha^2\varepsilon)$	Y	poly(n,|P|,2^d)	poly(n,|P|,2^d)

Summary: Can construct synthetic databases accurate on huge families of counting queries, but complexity may be exponential in dimensions of data and query set P.

- $D = (x_1,\ldots,x_n) \in (\{0,1\}^d)^n$
- $P = \{ \pi : \{0,1\}^d \rightarrow \{0,1\}\}$
- $\pi(D) := (1/n) \sum_i \pi(x_i)$
- α = accuracy error
- ε = privacy

Question: is this inherent?

Negative Results for Synthetic Data

Summary:

- Producing accurate & differentially private synthetic data is as hard as breaking cryptography (e.g. factoring large integers).
- Inherently exponential in dimensionality of data (and in dimensionality of queries).

Negative Results for Synthetic Data

- Thm [DNRRV09]: Under standard crypto assumptions (OWF), there is no n=poly(d) and curator that:
  - Produces synthetic databases.
  - Is differentially private.
  - Runs in time poly(n,d).
  - Achieves accuracy error α=.99 for P = {circuits of size d^2} (so $|P| \sim 2^{d^2}$)
- Thm [UV10]: Under standard crypto assumptions (OWF), there is no n=poly(d) and curator that:
  - Produces synthetic databases.
  - Is differentially private.
  - Runs in time poly(n,d).
  - Achieves accuracy error α=.01 for 2-way marginals.

Tool 1: Digital Signature Schemes

A digital signature scheme consists of algorithms (Gen,Sign,Ver):

- On security parameter d, Gen(d) = (SK,PK) $\in \{0,1\}^d \times \{0,1\}^d$
- On $m \in \{0,1\}^d$, can compute $\sigma = \mathrm{Sign}_{SK}(m) \in \{0,1\}^d$ s.t. $\mathrm{Ver}_{PK}(m,\sigma)=1$
- Given many (m,σ) pairs, infeasible to generate new (m',σ') satisfying Ver_{PK}
- Gen, Sign, Ver all computable by circuits of size d^2.

Hard-to-Sanitize Databases

- Generate random $(PK,SK) \leftarrow Gen(d)$, $m_1, m_2, \ldots, m_n \leftarrow \{0,1\}^d$

D

m_1	$Sign_{SK}(m_1)$
m_2	$Sign_{SK}(m_2)$
m_3	$Sign_{SK}(m_3)$
m_n	$Sign_{SK}(m_n)$

curator $\Rightarrow$

C(D)

m'_1	σ_1
m'_2	σ_2
m'_k	σ_k

- $Ver_{PK} \in$ {circuits of size d^2}=P
- $Ver_{PK}(D) = 1$

$\Longrightarrow$

- $Ver_{PK}(C(D)) \geq 1-\alpha > 0$
- $\exists$ i $Ver_{PK}(m'_i,\sigma_i)=1$

Case 1: $m'_i \notin D \Rightarrow$ Forgery!

Case 2: $m'_i \in D \Rightarrow$ Reidentification!

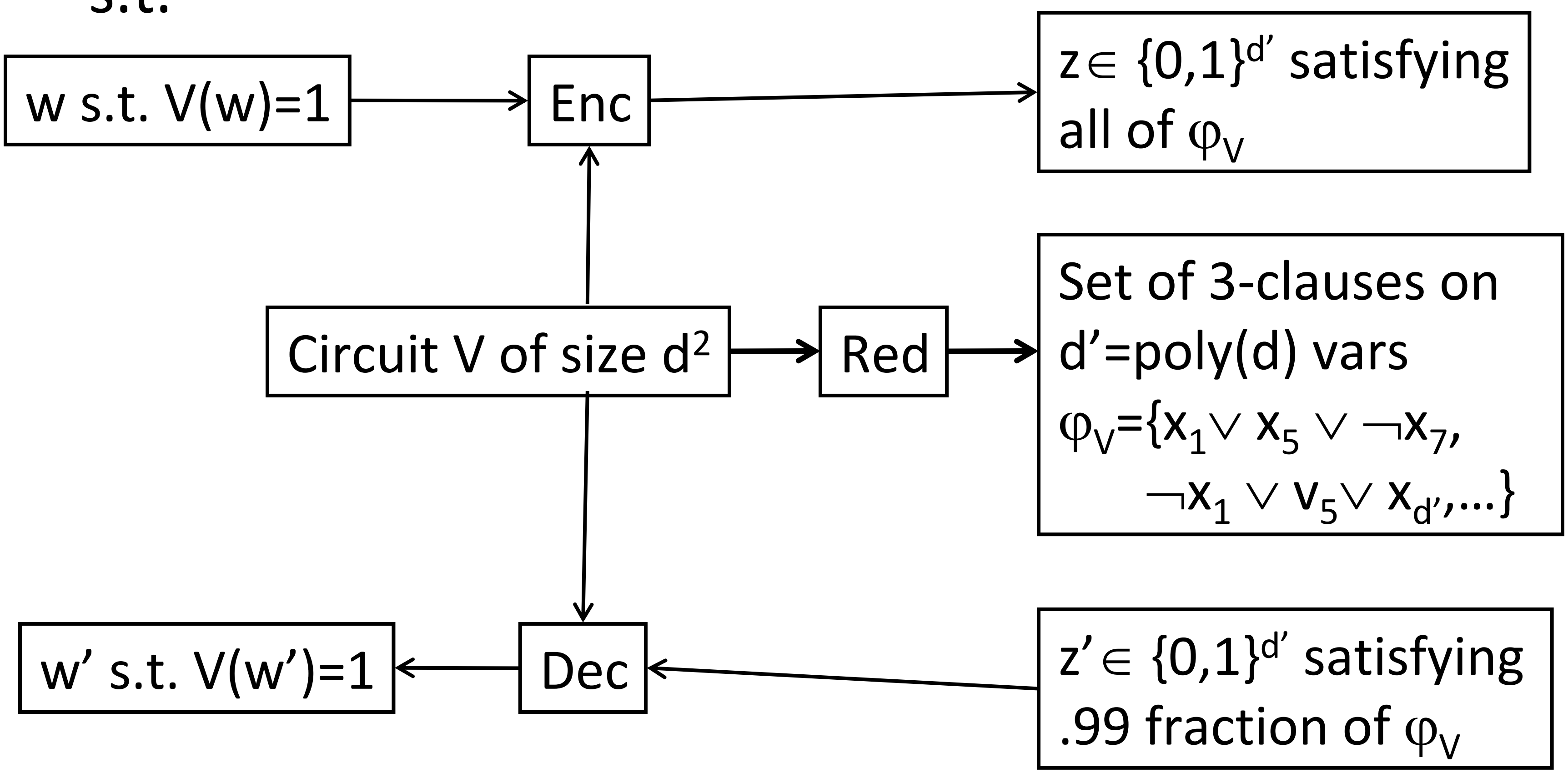
Negative Results for Synthetic Data

- Thm [DNRRV09]: Under standard crypto assumptions (OWF), there is no n=poly(d) and curator that:
  - Produces synthetic databases.
  - Is differentially private.
  - Runs in time poly(n,d).
  - Achieves accuracy error α=.99 for P = {circuits of size d^2} (so $|P| \sim 2^{d^2}$)

- Thm [UV10]: Under standard crypto assumptions (OWF), there is no n=poly(d) and curator that:
  - Produces synthetic databases.
  - Is differentially private.
  - Runs in time poly(n,d).
  - Achieves accuracy error α=.01 for 3-way marginals.

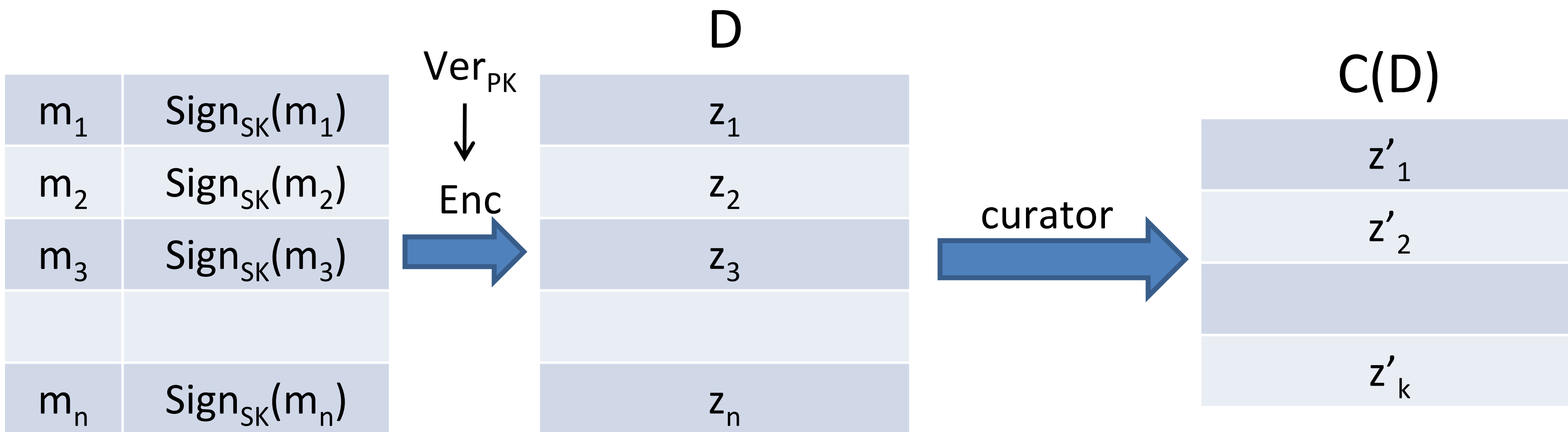
Tool 2: Probabilistically Checkable Proofs

The PCP Theorem: $\exists$ efficient algorithms (Red,Enc,Dec) s.t.



Hard-to-Sanitize Databases

- Generate random $(PK,SK) \leftarrow Gen(d)$, $m_1, m_2, \ldots, m_n \leftarrow \{0,1\}^d$



- Let $\varphi_{PK} = Red(Ver_{PK})$
- Each clause in φ_{PK} is satisfied by all z_i

- Each clause in φ_{PK} is satisfied by $\geq 1-\alpha$ of the z'_i
- $\exists$ i s.t. z'_i satisfies $\geq 1-\alpha$ of the clauses
- $Dec(z'_i)$ = valid (m'_i, σ_i)

Case 1: $m'_i \notin D \Rightarrow$ Forgery!

Case 2: $m'_i \in D \Rightarrow$ Reidentification!

Part I Conclusions

- Producing private, synthetic databases that preserve simple statistics requires computation exponential in the dimension of the data.

How to bypass?

- Average-case accuracy: Heuristics that don't give good accuracy on all databases, only those from some class of models.
- Non-synthetic data:
  - Thm [DNRRV09]: For general P (e.g. P={circuits of size d^2}), $\exists$ efficient curators "iff" $\neg\exists$ efficient "traitor-tracing" schemes
  - But for structured P (e.g. P={all marginals}), wide open!

PART II: COMPUTATIONALLY BOUNDED ADVERSARIES

Motivation

- Differential privacy protects even against adversaries with unlimited computational power.
- Can we gain by restricting to adversaries with bounded (but still huge) computational power?
  - Better accuracy/utility?
  - Enormous success in cryptography from considering computationally bounded adversaries.

Definitions [MPRV09]

- (ε,neg(k))-differential privacy: for all D_1, D_2 differing in one row, every set T, and security parameter k,

$$\Pr[C_k(D_1) \in T] \leq \exp(\varepsilon) \cdot \Pr[C_k(D_2) \in T] + \mathrm{neg}(k),$$

- Computational ε-differential privacy v1: for all D_1, D_2 differing in one row, every probabilistic poly(k)-time algorithm T, and security parameter k,

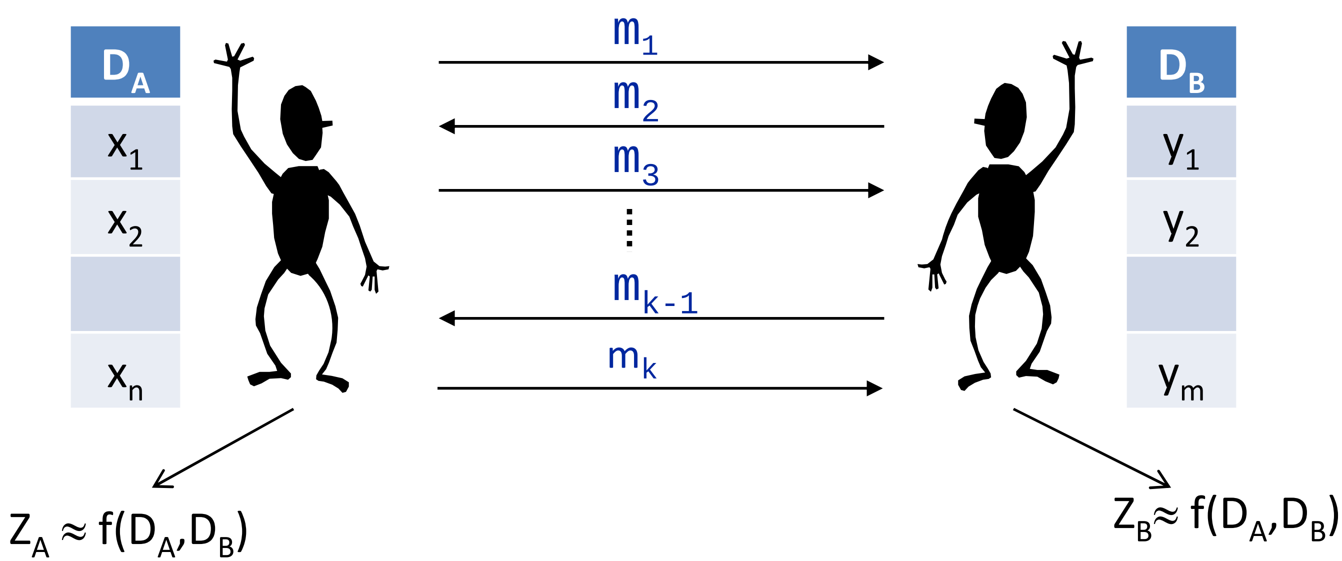
$$\Pr[T(C_k(D_1))=1] \leq \exp(\varepsilon) \cdot \Pr[T(C_k(D_2))=1] + \mathrm{neg}(k)$$

immediate ⇑ ⇓ open: requires generalization of Dense Model Thm [GT04,RTTV08]

- Computational ε-differential privacy v2: $\exists$ (ε,neg(k))-differentially private C'_k such that for all D, $C_k(D)$ and $C'_k(D)$ are computationally indistinguishable.

2-Party Privacy

- 2-party (& multiparty) privacy: each party has a sensitive dataset, want to do a joint computation $f(D_A,D_B)$



- A's view should be a (computational) differentially private function of D_B (even if A deviates from protocol), and vice-versa

Benefit of Computational Differential Privacy

Thm: Under standard cryptographic assumptions (OT),
$\exists$ 2-party computational ε-differentially private protocol for estimating Hamming distance of bitvectors, with error $O(1/\varepsilon)$.

Proof: generic paradigm

- Centralized Solution: Trusted third party could compute diff. private approx. to Hamming distance w/error $O(1/\varepsilon)$
- Distribute via Secure Function Evaluation [Yao86,GMW86]: Centralized solution $\rightarrow$ distributed protocol s.t. no computationally bounded party can learn anything other than its output.

Remark: More efficient or improved protocols by direct constructions [DKMMN06,BKO08,MPRV09]

Benefit of Computational Differential Privacy

Thm: Under standard cryptographic assumptions (OT), $\exists$ 2-party computational ε-differentially private protocol for estimating Hamming distance of bitvectors, with error $O(1/\varepsilon)$.

Thm [MPRV09,MMPRTV10]: The best 2-party differentially private protocol (vs. unbounded adversaries) for estimating Hamming distance has error $\tilde{\Theta}(n^{1/2})$.

Computational privacy $\Rightarrow$ significant gain in accuracy!

And efficiency gains too [BNO06].

Conclusions

- Computational complexity is relevant to differential privacy.
- Bad news: producing synthetic data is intractable
- Good news: better protocols against bounded adversaries

Interaction with differential privacy likely to benefit complexity theory too.