CS 2429 - Foundations of Communication Complexity

Lecture #11: Novemeber 28th, 2012

Lecturer: Wesley George and Yu Wu

In the last lecture, we have seen the connection between PAR (Privacy Approximation Ratio) and Communication Complexity. Today we are going to show some results from the paper 'The Limits of Two-Party Differential Privacy' by Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar and Salil Vadhan [7], which shows the connectoin between Two-Party Differential Privacy and Communication Complexity. We will also discuss the separation between the Client-Server Differential Privacy and the Two-Party Differential Privacy, as well as some compressibility result.

1 Introduction to Differential Privacy and Definitions

Differential Privacy is a very powerful general-purpose notion, introduced by [4, 5]. There is an excellent survey on the topic of differential privacy [3]. Informally, a randomized function of a database is *differentially private* if its output distribution is insensitive to the presence or absence of any particular record in the database. Therefore, if the analyses allowed on a database are promised to preserve differential privacy, there is little incentive for an individual to conceal his or her information from the database.

In this lecture, we will use the definition of differential privacy for mechanisms defined over strings from a finite alphabet Σ and generalize it to interactive protocols.

Definition (Differential Privacy) A mechanism M on Σ^n is a family of probability distributions $\{\mu_x : x \in \Sigma^n\}$ on $\mathcal{R}$. The mechanism is ϵ- differentially private if for every x and x' such that $|x - x'|_H = 1$ and every measurable subset $S \subset \mathcal{R}$ we have

$$\mu_x(S) \leq \exp(\epsilon)\mu_{x'}(S).$$

where $|x - x'|_H$ denotes the Hamming distance between x and x'.

To distinguish it from the later generalization, we call it as the Client-Server Differential Privacy. For a mechanism that satisfy ϵ-differential privacy, it has the following properties:

(1). ϵ is usually a very small number, and depends on specific problems. The term $\exp(\epsilon)$ is roughly $(1 + \epsilon)$ for small ϵ. The exponential form is easy to work with: suppose one wants to estimate the ratio $\frac{\mu_x(S)}{\mu_{x''}(S)}$ for $|x - x''|_H = 2$, then it can be bounded as:

$$\exp(-2\epsilon) \leq \frac{\mu_x(S)}{\mu_{x''}(S)} \leq \exp(2\epsilon).$$

(2). Usually we want the output of the mechanism M (denoted as $M(x)$) is close to a function $f(x)$ on x. It is common to use the expectation of difference in $M(x)$ and $f(x)$ as a measure of error.

A common relaxation of ϵ-differential privacy is the following definition of δ-approximate ϵ-differential privacy, abbreviated as (ϵ, δ)-differential privacy.

Definition (Approximate differential privacy) The mechanism M satisfies δ-approximate ϵ-differential privacy if for every x and x' such that $|x - x'|_H = 1$ and every measurable subset $S \subset \mathcal{R}$ we have

$$\mu_x(S) \leq \exp(\epsilon)\mu_{x'}(S) + \delta.$$

The definition of differential privacy naturally extends to interactive protocols, by requiring that the *views* of all parties be differentially private in respect to other parties' inputs. The following definition assumes semi-honest parties, i.e., parties that are guaranteed to follow the protocol (Note that for models with weaker restriction on adversarial parities, the lower bounds on accuracy of differentially private protocols still apply).

Let $VIEW_P^A(x, y)$ be the joint probability distribution over x, the transcript of the protocol P, private randomness of the party A, where the probability space is private randomness of both parties. For each x, $VIEW_P^A(x, y)$ is a mechanism over the y's. Let $VIEW_P^B(x, y)$ be similarly defined view of B whose input is y.

Definition (Differential Privacy for two-party protocols) We say that a protocol P has ϵ-differential privacy if the mechanism $VIEW_P^A(x, y)$ is ϵ-differentially private for all values of x and same holds for $VIEW_P^B(x, y)$ and all values of y.

The notion of *sensitivity* of a function is frequently used in designing differentially private mechanisms. We present its definition here:

Definition (Sensitivity) For a real-valued function $f : \Sigma^n \to R$ define its *sensitivity* as the maximal difference in value on adjacent inputs, i.e., $\max_{|x-y|_H=1} |f(x) - f(y)|$.

2 Differential Privacy and Communication Complexity

In this section, we are going to explore the connections between two-party differential privacy and communication complexity. We show that they are all connected to the information cost of the protocols. In section 2.1, we show that every differentially private protocol has (relatively) small information cost, along with some compressibility results. In section 2.2 we show that a deterministic protocol that computes or approximates a function of sensitivity 1 with low communication complexity can be converted to a differentially private protocol with some loss of accuracy.

2.1 Differential Privacy and Information Cost

Recall the definition of information cost is:

Definition (Information Cost). Given a distribution μ over inputs X and Y to the two parties of protocol P, the information cost of P for distribution μ is defined as

$$ICost_\mu(P) = I(XY; \Pi(X,Y)),$$

where $\Pi(X,Y)$ is the random transcript of the protocol on input (X,Y).

Intuitively, the setting of two-party differential privacy can be seen as adding a strong restriction on two-party communication protocols, in the sense that the information cost of a protocol that is differentially private would have relatively small information cost.

Proposition 1 *If $P(x,y)$ has ϵ-differential privacy, where $x,y \in \Sigma^n$ for a finite alphabet Σ, then for every distribution μ on $\Sigma^n \times \Sigma^n$, the information cost of P is bounded as follows:*

$$ICost_\mu(P) \leq 3\epsilon n.$$

If $\Sigma = \{0,1\}$ and μ is the uniform distribution, then the bound can be improved to $ICost_\mu(P) \leq 1.5\epsilon^2 n$.

Proof Let (X,Y) be the random variables of the input, $\Pi(X,Y)$ to be the random variable of the transcript. For every (x,y), (x',y'), differential privacy implies that

$$\exp(-2\epsilon n) \leq \frac{Pr[\Pi(x,y)=\pi]}{Pr[\Pi(x',y')=\pi]} \leq \exp(2\epsilon n).$$

so that

$$\exp(-2\epsilon n) \leq \frac{Pr[\Pi(x,y)=\pi]}{Pr[\Pi(X,Y)=\pi]} \leq \exp(2\epsilon n).$$

where (X,Y) is an independent sample from μ.

$$\begin{aligned} ICost_\mu(P) &= I(X,Y;\Pi(X,Y)) \\ &= H(\Pi(X,Y)) - H(\Pi(X,Y)|X,Y) \\ &= \mathbb{E}_{(x,y,\pi)} \log \frac{Pr[\Pi(X,Y)=\pi | X=x, Y=y]}{Pr[\Pi(X,Y)=\pi]} \\ &\leq 2(\log_2 e)\epsilon n \end{aligned}$$

For the special case where μ is the uniform distribution and $\Sigma = \{0,1\}$, we can in fact prove that $ICost_\mu(P) \leq 1.5\epsilon^2 n$, by using additivity of mutual information and the fact that each bit of the input is totally independent of the other input bits. Details can be found in [7].

In Lecture 7 we saw that communication protocols with low information cost can be compressed into protocols with low communication complexity. Since Proposition 1 shows that differentially private protocols have low information cost, it immediately yields that differentially private protocols can be compressed into protocols that have low communication complexity (not necessarily differentially private). Formally, we have the following theorem.

Theorem 2 *Let P be an ϵ-differentially private protocol P with output $out(P)$ where the input (X,Y) is distributed according to an arbitrary product distribution μ. Then for every $\gamma > 0$, there exists functions f_A, f_B, and a protocol Q such that $\|f_A(X,Q(X,Y)) - out(P)\|_{SD} < \gamma$, $Pr(f_A(X,Q(X,Y)) \neq f_B(Y,Q(X,Y))) < \gamma$ and $CC(Q) \leq 3\epsilon\gamma^{-1} n \cdot polylog(CC(P)/\gamma))$.*

For differentially private protocols with constant rounds, compression can be done while maintaining differential privacy.

Theorem 3 *Let P be an ϵ-differentially private protocol with r rounds. Then for every $\delta > 0$, there exists an $O(r\epsilon)$-differentially private protocol $P*$ that has communication complexity $O(r(\epsilon n + \log\log\frac{1}{\epsilon\delta}))$ and except with probability $r\delta$, simulates P perfectly. In other words, there exists functions π_x, π_y such that $Pr[\pi_x(VIEW^A_{P*}(x,y)) = VEIW^A_P(x,y)] \geq 1 - r\delta$, and similarly for B.*

The proof of Theorem 2 and Theorem 3 can be found in [7].

2.2 From Low Communication to Differential Privacy

In section 2.1, we saw that differentially private protocols have relatively small information cost and can be compressed to protocols with low communication complexity. In this section, we present some result from the other direction: protocols for certain functions with (deterministic) communication complexity can be converted to differentially private protocols. Namely, we have the following theorem:

Theorem 4 *Let P be a deterministic protocol with communication complexity $CC(P)$ approximating a sensitivity-1 function $f : \Sigma^n \times \Sigma^n \to \mathbf{Z}$ with error bounded by Δ. Then there exists an ϵ-differentially private protocol with the same communication complexity and the number of rounds which computes f with expected additive error $\Delta + O(CC(P)r/\epsilon)$.*

Note that the above theorem is still a bit far from optimal: first of all, it only considerate deterministic protocols; secondly, the loss of accuracy in the differentially private protocol depends on the sensitivity of the function, as well as the number of rounds of the original communication protocol. Any improvement of Theorem 4 would be very interesting.

Now let's see how to prove Theorem 4. First, we introduce the exponential mechanism due to McSherry and Talwar [9] as it is a key component in proving Theorem 4.

Definition (Exponential Mechanism) A real-valued score function $q(x, r)$ is defined over the space of all possible inputs x and outputs r. For given x and privacy parameter ϵ the exponential mechanism denoted as $\xi^\epsilon_q(x)$ outputs r with probability proportional to $\exp(-\epsilon q(x,r)/2)$.

McSherry and Talwar [9] proved that for a sensitivity-1 score function, the exponential mechanism satisfies ϵ-differential privacy. Moreover, on expectation,

$$\mathbb{E}[q(x, \xi^\epsilon_q(x))] < \min_r q(x,r) + 4\log|\mathcal{R}|/\epsilon.$$

The high-level idea of proving Theorem 4 is the following: given a deterministic protocol for computing the sensitivity-1 function $f(x,y)$ we construct an ϵ-differentially private protocol by sampling messages of the new protocol using the exponential mechanism. The score function $q(x,m)$, which specifies the exponential mechanism, is defined as the smallest number of bits one has to flip in the input x to make the protocol output m. Now let's see how to use the exponential mechanism to prove Theorem 4:

Proof Let π_i be the transcript up to and including the i-th round of the protocol P, and let the protocol be specified as r functions $m_i(\cdot,\cdot)$, so that the first message of the protocol is $m_1(x,\pi_0)$, where π_0 is empty, the second message is $m_2(y,\pi_1)$, etc.

We define a new differentially protocol P^* by applying the exponential mechanism at each round to sample from the set of messages consistent with the transcript of the protocol so far. Assume wlog that i is odd, and let $X_i \subset \Sigma^n$ be Alice's set of inputs that are consistent with the transcript π^*_{i-1} under the original protocol P. In other words, if the j-th message in π^*_{i-1} is μ_j, it holds that $\mu_j = m_j(x,\pi^*_{j-1}), \quad \forall x \in X_i$ and odd $j < i$. If the length of the i-th message of P is k_i bits, let $M_i \in \{0,1\}^{k_i}$ be the set of all messages that the protocol P may output for the given transcript, i.e.,

$$M_i = \{\mu \in \{0,1\}^{k_i} : \exists x' \in X_i, s.t., m_i(x', \pi^*_{i-1}) = \mu\}.$$

Define the score function $q : \Sigma^n \times M_i \to R$ as

$$q_i(x,\mu) = \min_{m_i(x',\pi^*_{i-1})=\mu, x' \in X_i} \|x - x'\|_1,$$

Let the i-th message of the new randomized protocol $P^*(x,y)$ be the output of the exponential mechanism $\xi_{q_i}^{\epsilon/[r/2]}(x)$. To compute the function, if the party receiving the last message of the protocol is Alice, she finds the closest $x' \in X_r$ to her input x and outputs $f_A(x', \pi^*_r)$, and similarly to Bob.

To prove the theorem we demonstrate the following three properties of the protocol P^*:

(1). P^* is well-defined. Since the i-th round of P^* is the output of the exponential mechanism, the only possibility for the protocol's not completing is for M_i to be empty for some i. However, this cannot happen because for every feasible output μ, there is an input x' which is consistent with μ. As the sets X_i and M_i never become empty, the protocol never aborts.

(2). P^* is ϵ-differentially private. P^* is the combination of a sequence of $\epsilon^* = \epsilon/[r/2]$-differentially private applications of the exponential mechanism. The total privacy budge consumed by each party is thus at most ϵ.

(3). Its additive error is bounded as $\Delta + O(CC(P)r/\epsilon)$. Let $\epsilon^* = \epsilon/[r/2]$ and $K_i = \Sigma_{j=1}^{i-1} k_j$ where k_j is the length of π^*_{i-1}. We claim that for the closest to x element $x' \in X_i$, the distance between x and x' is dominated as a random variable by $K_i/\epsilon^* + \Gamma(i, 1/\epsilon^*)$. The proof is by induction on the round number i. For the first round, X_1 includes all possible inputs, and the distance $||x - x'||_1 = 0$. For subsequent rounds, if x' is the closest to x element of X_i, the optimal value of the score function $q_i(x,\mu)$ is $||x - x'||$. By the property of exponential mechanism, (..) Finally, when Alice approximates the value of the function by computing $f_A(x', \pi^*_r)$ for $x' \in X_r$ closest to her input x, the expected error

$$\begin{aligned}
\mathbb{E}(|f(x,y) - f_A(x', \pi^*_r)| &\leq |f(x,y) - f(x',y')| + |f(x',y') - f_A(x', \pi^*_r)| \\
&\leq \mathbb{E}(|x - x'| + |y - y'|) + \Delta \\
&\leq \Delta + 2K_r/\epsilon^* + r/\epsilon^*.
\end{aligned}$$

where y' is similarly defined value closest to Bob's input y and consistent with the protocol's transcript. Since $K_r > r$ and $\epsilon < 3r\epsilon^*$, the expected error of P^* is $\Delta + O(CC(P)r/\epsilon)$ as claimed.

3 Lowerbounds for 2-party differential-privacy

We now turn to the separation of the (standard) client-server setting, and the two-party setting (defined in this note). We will be interested in functions

1. well approximated by protocols that are 1-party differentially private, but that
2. are poorly approximated by any protocol that is 2-party differentially private.

3.1 Preliminaries

When one is interested in a differentially private mechanism, one has in mind a particular function f and is interested in differentially private mechanism $\mathcal{K}$ that is *correlated* with f (so that $\mathcal{K}(x)$ can be used as a proxy for $f(x)$). What does it mean for a function and a mechanism to be correlated? For any x, $\mathcal{K}(x) - f(x)$ is distribution over errors. Certainly if $\mathcal{K}(x)$ is a stand-in for $f(x)$, this distribution of errors can't be "too big". Specifying a norm on the space of distributions is a quantification of size.

Our norm, which we call (γ, k)-boundedness, is the following.

Definition [(γ, k) boundedness, correlation] A distribution D over $\mathbb{R}$ is (γ, k)-bounded if

$$\Pr_{X \sim D}[|X| > k] < \gamma \tag{1}$$

Definition We will say that mechanism $\mathcal{K}$ and function f are (γ, k)-correlated if $\mathcal{E}(x) = \mathcal{K}(x) - f(x)$ is γ, k-bounded for every x.

Looking ahead, a mechanism that is (γ, k) correlated with f gives rise to a protocol computing f within an additive factor k with error probability at most γ.

Note that (γ, k)-correlation is a weak notion of correlation; for example being $(\gamma = O(1), k = O(1))$-bounded does not imply that $\| \cdot \|_1 < \infty$. This choice of error-norm is appropriate as we are aiming at lower-bounds for differential privacy. I.e. we want to give examples of functions that cannot be approximated meaningfully by differentially private mechanisms; by choosing a weak notion of correlation we are pointing out that even if one is satisfied with a very weak notion of correlation, one cannot. (In computation-theory parlance: this weak notion of correlation gives strong lower-bounds).

Collecting the above discussion, our central object of study is the following.

Definition Mechanism $\mathcal{K}$ is an (ϵ, γ, k)-realization of the function f if

1. $\mathcal{K}$ is ϵ-differentially private
2. $\forall x$ $\mathcal{E}(x) = M(x) - f(x)$, the error-distribution for input x, is (γ, k) bounded.

Given a partition of f's input into two pieces and Π is a protocol who's input/output syntax matches f, we say that Π is an (ϵ, γ, k) 2-party realization of f if

1. $\forall y$, $\mathcal{K}(x) = \text{View}^A_\Pi(x, y)$ is ϵ-differentially private
2. $\forall x$, $\mathcal{K}(y) = \text{View}^B_\Pi(x, y)$ is ϵ-differentially private

3. $\forall x, y$, $\mathcal{E}(x, y) = \Pi(x, y) - f(x, y)$, the error-distribution for input (x, y), is (γ, k) bounded.

Note that we view a mechanism as taking ϵ as an input[1]. As ϵ gets smaller, distributions for neighbouring databases are required to be closer, so this is a stronger constraint. The suggestion of this work is that such constraints can only be realized by adding more noise to the output, increasing the error. Thus the quality of a mechanism is quantified by expressing γ,k as a function of ϵ. A mechanism is more or less usable only if γ and k depend only on ϵ (and not, say, $|x|$, the size of the "database").

3.2 Theorem statements

We have now developed the language and motivation to state our two main theorems. Each theorem each give a function that is realizable in the 1-party setting, but not realizable in the 2-party setting for any reasonable choices of correlation parameters γ, k. The first theorem concerns a natural function, the hamming distance.

Theorem 5 *Let $f_H(x, y) = d_H(x, y)$.*

1. *There exists an $(\epsilon, \gamma = O_\epsilon(1), k = O_\epsilon(1))$ 1-party realization of f_H.*
2. *Let Π be an (ϵ, γ, k) 2-party realization of f_H.*

$$\exists \epsilon_0 \forall \epsilon \leq \epsilon_0 \quad \gamma = O_\epsilon(1) \implies k = \Omega(\sqrt{n})$$

Note that the error of $k = \Omega(\sqrt{n})$ should be compared to the range of $f(x, y)$, which is $[n]$ (the suggestion being that a number that is more than $\sqrt{n}$ from the hamming distance of two strings is unlikely to be of much use for an application that needs hamming distance)

The second proposition is of identical form, but gives a function with even larger error relative to the range of the function. The function is slightly less natural function in that doesn't have a name, but has a short description that uses IP_n, the inner-product of $GF(2)^n$, and a "good" error-correcting code.

Theorem 6 *Let $C : \{0,1\}^n \to \{0,1\}^m$ be a code with distance $d = \Omega(n)$ and where $m = \Theta(n)$. Let $f_{IP} : \{0,1\}^m \times \{0,1\}^m \to \mathbb{R}$ such that*

$$f_{IP}(x, y) = IP_n(Dec(x), Dec(y))(d - d(x, C) - d(y, C))$$

1. *There exists an $(\epsilon, \gamma = O_\epsilon(1), k = O_\epsilon(1))$ 1-party realization of f_{IP}.*
2. *Let Π be an (ϵ, γ, k) 2-party realization of f_{IP}.*

$$\exists \epsilon_0 \forall \epsilon \leq \epsilon_0 \quad \gamma = O_\epsilon(1) \implies k = \Omega(n)$$

Since the functions in both Theorems 5 and 6 have low-sensitivity, the laplacian mechanism (i.e. adding noise distributed as the Laplacian distribution with mean $1/\epsilon$. This development is given in Section 3.3. Both lower-bounds use machinery from information complexity; these proofs are given in Section 3.4. Section 3.5 discusses of the computational relaxation of differential privacy, which offers some consolation for the lower-bounds.

[1]Deploying a mechanism on some database requires a choice of ϵ appropriate for the domain of the data. This is best chosen by someone with a strong sense of the semantics of data. Making ϵ a parameter is an attempt to formulate a clean problem of mechanism design without thinking too much about the semantics of particular data.

3.3 Realizing insensitive functions in the client-server setting

In this section we discuss the "upper-bounds" of for Theorems 5 and 6: i.e. that there exists differentially-private mechanisms in the 1-party setting that are well correlated with f_H and f_{IP}. Our "client-server" setting is the standard setting for differential privacy, so mechanisms for this setting are well-known. We will use the simplest such mechanism, the Laplacian mechanism, obtained by adding noise distributed as $Lap(1/\epsilon)$ to the value of the function; the laplacian distribution is tightly concentrated around its mean and so can easily be shown to be $(\gamma = O(1), k = O(1))$ bounded for $\epsilon = \Theta(1)$. This mechanism works for functions with low-sensitivity, which happens to be the case for both f_H and f_{IP}.

We now give some details.

Definition Let f be real-valued function. The sensitivity of f, which we denote Δf is the quantity

$$\max_{x,x':d_H(x,x')=1} |f(x) - f(x')|$$

(we only say that f is real-valued so we know what metric to apply on the range of f; the definition makes sense given any metric on the range of f). Both f_H and f_{IP} are functions with sensitivity 1: that f_H has sensitivity one is trivial; as for f_{IP}, $h(x) = d(x, C)$ has sensitivity at most 1, thus so too does f_{IP}.

Let $\mathcal{K}$ be the mechanism that on input x, ϵ outputs $f(x)+E$ where E is distributed as $\mathrm{Lap}(\delta f/\epsilon)$. (Recall $X \sim \mathrm{Lap}(\lambda)$ if $\Pr[X = x] \propto \exp(-|x|/\lambda)$).

Proposition 7 *$\mathcal{K}$ is an $(\epsilon, \gamma = O(\Delta f/\epsilon), k = O(\Delta f/\epsilon))$ realization of f.*

Proof Note that a random variable with distribution $Lap(\lambda)$ has mean 0 and variance $2\lambda^2$. Applying Chebyshev's inequality

$$\Pr[|X - \mu| \geq \ell\sigma] \leq 1/\ell^2$$

for any value of $\ell > 1$ gives that the error distribution of $\mathcal{K}(x)$ is $(\gamma = 1 - 1/\ell^2, k = \ell\epsilon/\Delta f)$ bounded.

Let D_1,D_2 be two inputs to f that differ in one position. Unpacking the definition of $\mathcal{K}$ and Lap, we have that for all x

$$\Pr[\mathcal{K}(D_1) = x] \propto \exp(-|f(D_1) - x|\epsilon/\Delta f)$$
$$\Pr[\mathcal{K}(D_2) = x] \propto \exp(-|f(D_2) - x|\epsilon/\Delta f)$$

Taking the ratio of these quantities we get

$$\frac{\Pr[\mathcal{K}(D_1) = x]}{\Pr[\mathcal{K}(D_2) = x]} = \exp(-(|f(D_1) - x| - |f(D_2) - x|)\epsilon/\Delta f) \tag{2}$$

Observing that

$$|f(D_1) - x| - |f(D_2) - x| \leq |f(D_1) - f(D_2)| \leq \Delta f$$

the exponents in the RHS of (2) collapse and the LHS is lowerbounded by $\exp(-\epsilon)$. By symmetry of D_1,D_2, we get that for all x

$$\Pr[\mathcal{K}(D_1) = x] \leq \exp(\epsilon) \Pr[\mathcal{K}(D_2) = x] \tag{3}$$

The condition of ϵ-differential privacy requires that (3) holds for arbitrary sets S and not just individual points, but we can write each set as a union of points and use the law of total probability to derive the condition of ϵ-differential privacy.

3.4 Lower-bounds for the 2-party setting

We now turn to establishing the lower-bounds of Theorems 5 and 6.

Given Proposition 1, i.e. that

$$\Pi \text{ is } \epsilon\text{-differentially private } \implies IC(\Pi) \leq 3\epsilon n$$

to lowerbound ϵ, it suffices to get a lowerbound on the information complexity of the protocol in question, which we do with a reduction from a standard problem.

3.4.1 Hamming-distance

Proposition 8 *Let Π_H be a (ϵ, γ, k) 2-party realization of d_H. There exists a constant ϵ_0 such that for all $\epsilon \leq \epsilon_0$*

$$\gamma = O_\epsilon(1) \implies k(\epsilon) = \Omega(\sqrt{n})$$

.

Proof Let Π_H be a protocol that $(\gamma = O(1), k = o(\sqrt{n})$ realizes d_H. We will use Π_H to compute the following function

$$f_{GapH}(x,y) = \begin{cases} 0 & d_H(x,y) < n/2 - \sqrt{n} \\ \perp & |d_H(x,y) - n/2| \leq \sqrt{n} \\ 1 & d_H(x,y) > n/2 + \sqrt{n} \end{cases}$$

Let Π_{GapH} be the following (obvious) protocol: each player simply runs Π_H on their input to obtain α, an approximation of $d_H(x,y)$. If $|\alpha - n/2| < \sqrt{n}$ output $\perp$, otherwise output 0 or 1 based on what side of $n/2$ is α.

Clearly Π_{GapH} computes f_{GapH} with probability at least γ.

Given this reduction, we have that $IC(d_H) \geq IC(f_{GapHam})$. We will show shortly that

$$IC(f_{GapH}) = \Omega(n) \tag{4}$$

Combining these facts with Proposition 1 we have that

$$\Omega(n) \leq IC(d_H) \leq 3\epsilon n$$

So there is some constant lowerbound below which we cannot obtain ϵ differential privacy for d_H if we care about error $o(\sqrt{n})$.

As for (4), Chakrabarti and Regev [1] showed that f_{GapH} has a smooth rectangle bound of $2^{\Omega(n)}$, implying a partition bound of the same value. Using the technology of [6], we have a $\Theta(n)$ lowerbound on IC(f_{GapHam}).

Note that a different proof appears in [8] as at its time of publication, the machinery of [6] did not exist and another proof of (4) was not known.

3.4.2 f_{IP}

Proposition 9 *Let $C : \{0,1\}^n \to \{0,1\}^m$ be a code with distance $d = \Omega(n)$ and where $m = \Theta(n)$. Let $f_{IP} : \{0,1\}^m \times \{0,1\}^m \to \mathbb{R}$ such that*

$$f_{IP}(x,y) = IP_n(Dec(x), Dec(y))(d - d(x,C) - d(y,C))$$

Let Π be an (ϵ, γ, k) 2-party realization of f_{IP}.

$$\exists \epsilon_0 \forall \epsilon \leq \epsilon_0 \quad \gamma = O_\epsilon(1) \implies k = \Omega(n)$$

Proof As before, let Π_f is a (ϵ, γ, k) realization of f. We show that if k is $o(n)$, then ϵ has a non-zero constant lowerbound.

We can compute $g = IP$ using Π_f in the trivial way: on input x, run Π_f on $C(x)$ to obtain output β. If $\beta > \alpha m/2$, output 1, otherwise output 0. As before we have

$$\Omega(n) = IC(IP) \leq IC(f_{IP}) \leq 3\epsilon n$$

giving us our non-zero constant lower bound for ϵ.

3.5 Consolation in computational differential-privacy

One might ask: is it possible to relax our definition of 2-party differential privacy without compromising totally on security, but while allowing us to circumvent the earlier lower-bounds?

Instead of considering the maximum ratio of probabilities for neighbouring distributions, one might relax the distribution by requiring only that the maximum ratio of the distinguishing advantage of polytime adversaries is similarly bounded. That is, for a particular mechanism $\mathcal{K}$, rather than require that for all neighbouring x,x',

$$\forall S \Pr[\mathcal{K}(x) \in S] / \Pr[\mathcal{K}(x') \in S] \leq \exp(\epsilon)$$

we ask instead that

$$\forall A \ \Pr[A(\mathcal{K}(x)) = 1] / \Pr[A(\mathcal{K}(x')) = 1] \leq \exp(\epsilon) + \delta$$

where A is constrained to run in polynomial time. (Introducing δ is necessary for technical reasons concerning the power of non-uniform polytime). The above notion is one of the possible formalizations of *computational differential privacy*. This definition and others is considered in [10], where the problem of 2-party differential privacy was first formulated.

This relaxed notion is interesting as we can use machinery of cryptography to obtain a generic transformations protocols that are differentially private in the 1-party setting, to protocols giving identical privacy in the 2-party setting. The machinery in question is that which solves the (very general) problem of "secure" multi-party computation. A more precise, though not concise, name might be "private-input public-output function evaluation"; i.e. there is a single function f, each of the two parties have an input proper to them (x_0, x_1), they want to collaborate to compute $f(x_0, x_1)$ such that the value of $f(x_0, x_1)$ is computed and disclosed to both parties, but nothing else about their respective inputs is revealed to either party.

The transformation is simple: Let $f(x_1, x_2)$ be a function with an ϵ-differentially private mechanism $\mathcal{K}$ (that correlates with f by some factor δ). To build an ϵ-differentially private *protocol*

for computing f (wherein the views of both parties are ϵ-differentially private mechanisms of the other's input), it suffices run a computationally secure MPC protocol for evaluating $\mathcal{K}$. The claim is that the resulting protocol will be differentially private.

Proposition 10 *If f has an ϵ-differentially private implementation in the client-server setting, then f also has an* ϵ-computationally *approximately differentially private implementation (with the same error distribution) in the two-party setting.*

3.5.1 Malicious players and computational differential privacy

In this section we outline a second sense in which the notion of computational differential privacy is more permissive (we already saw examples of fucntions that could not be implemented in a manner that is statistically differentially private in the two party setting, but could be implemented in the 1-party setting).

The 2-party differential privacy discussed so far is that the view of a protocol remains differentially private assuming that both players follow the protocol. In the cryptographic literature, this notion of security would be referred to as "security against honest but curious adversaries". There are protocols that are secure against honest-but-curious adversaries, but if an adversary were allowed to deviate from the protocol, all security would be lost. The first protocol for secure multi-party computation, Yao's garbled circuit evaluation [11] is one such protocol. To obtain security against arbitrary adversarial behaviours, one may insist that all players prove at each step of the protocol that their messages were honestly generated; the proof system must have a zero-knowledge property so that the proof does not disclose anything inappropriate about the other parties input.

We thus can strengthen the earlier Proposition 10 to the following:

Proposition 11 *If f has an ϵ-differentially private implementation in the client-server setting, then f also has an* ϵ-computationally *approximately differentially private implementation (with the same error distribution) in the two-party,* even against malicious adversaries.

If one is interested in statistical differential privacy, such a general theorem is not possible for the space of all functions. There are known protocols for giving information-theoretically secure multi-party computation of arbitrary functions *provided there are a minimum number of players following the protocol*: a strict majority of players need be honest. In the two-party setting, this means that both players need be honest. [10] cites Benny Chor and Eyal Kushilevitz "A Zero-One Law for Boolean Privacy", [2], as witnessing functions that cannot be implemented by a protocol that retains information-theoretic privacy of a party's input in the face of arbitrary behaviour of the other player.

References

[1] A. Chakrabarti and O. Regev, *An optimal lower bound on the communication complexity of gap-hamming-distance*, Electronic Colloquium on Computational Complexity (ECCC), 17 (2010), p. 140.

[2] B. Chor and E. Kushilevitz, *A zero-one law for boolean privacy*, SIAM J. Discrete Math., 4 (1991), pp. 36–47.

[3] C. Dwork, *Differential privacy: A survey of results*, Theory and Applications of Models of Computation, (2008), pp. 1–19.

[4] C. Dwork, F. McSherry, K. Nissim, and A. Smith, *Calibrating noise to sensitivity in private data analysis*, Theory of Cryptography, (2006), pp. 265–284.

[5] C. Dwork and K. Nissim, *Privacy-preserving datamining on vertically partitioned databases*, in Advances in Cryptology–CRYPTO 2004, Springer, 2004, pp. 134–138.

[6] I. Kerenidis, S. Laplante, V. Lerays, J. Roland, and D. Xiao, *Lower bounds on information complexity via zero-communication protocols and applications*, Electronic Colloquium on Computational Complexity (ECCC), 19 (2012), p. 38.

[7] A. McGregor, I. Mironov, T. Pitassi, O. Reingold, K. Talwar, and S. Vadhan, *The limits of two-party differential privacy*, in 51st Annual Symposium on Foundations of Computer Science (FOCS), 2010, pp. 81–90.

[8] A. McGregor, I. Mironov, T. Pitassi, O. Reingold, K. Talwar, and S. P. Vadhan, *The limits of two-party differential privacy*, in FOCS, IEEE Computer Society, 2010, pp. 81–90.

[9] F. McSherry and K. Talwar, *Mechanism design via differential privacy*, in Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on, IEEE, 2007, pp. 94–103.

[10] I. Mironov, O. Pandey, O. Reingold, and S. P. Vadhan, *Computational differential privacy*, in CRYPTO, S. Halevi, ed., vol. 5677 of Lecture Notes in Computer Science, Springer, 2009, pp. 126–142.

[11] A. C.-C. Yao, *Protocols for secure computations (extended abstract)*, in FOCS, IEEE Computer Society, 1982, pp. 160–164.