CS 229r Information Theory in Computer Science Apr 18, 2019

Lecture 22

Instructor: Madhu Sudan *Scribe: Dor Verbin*

1 Administrative Notes

1. Send an email or upload a text entry to Canvas, containing a few paragraphs for a project check-in.
2. When preparing the talk:
   (a) Talk is ≤ 15 minutes, including time for transitions between talks, questions and answers, technical difficulties, etc.
   (b) Think about what really needs to go into the talk (skippable proofs).
   (c) Figure out division within team.
   (d) Try to link your talk to the course material.

2 Distributions in Theoretical CS

Historically, randomness has entered theoretical CS in randomized algorithms, and specifically in derandomization or randomness extraction. An example of an early use of randomness is von Neumann's solution for generating fair coin tosses using a biased coin [1]. Suppose that a biased coin has probability p to land on H and probability $1-p$ to land on T, where $0 < p < 1$ is unknown. We would like to use this coin to output H or T with probability $1/2$ each. This can be done by flipping the coin twice until we get HT or TH (ignoring any HH or TT results), and taking the output of the first flip.

Later, Blum [3] replaced the biased coin with a (finite sized) Markovian source and presented an algorithm that generates an independent and unbiased sequence of Hs and Ts. Santha and Vazirani [4] then assumed a more general *slightly-random source*. Later, Nisan and Zuckerman [7] introduced *min-entropy sources*, making no assumption about independence but only assuming that the source is an ε-source, i.e. that $H_\infty(x) \geq \varepsilon n$, where $H_\infty(x) = \min_{\omega \in \Omega} \log \frac{1}{\mathbb{P}[X=\omega]}$. Comparing this to $H(x) = \mathbb{E}_{\omega \in \Omega}\left[\log \frac{1}{\mathbb{P}[X=\omega]}\right]$ we see that $H_\infty(x) \leq H(x)$.

Randomness is also heavily used in cryptography, motivating the problem discussed in this lecture. Additional uses of randomness are in distributed algorithms (for example in leader election), and economics and game theory (for example mixed equilibria are inherently probabilistic).

3 Sampleable Distributions

Some distributions are easy to sample from. For example, generating samples from $\text{Bern}(p_1) \times ... \times \text{Bern}(p_n)$ is easy by sampling each $\text{Bern}(p_i)$ variable independently. The most general class of random variables that we can sample is randomness that can be produced by an algorithm.

Assume C is a circuit which takes m input bits and produces n output bits, i.e. $C : \{0,1\}^m \to \{0,1\}^n$ where $m, |C| = \text{poly}(n)$. Any circuit C defines a distribution $C(Y)$ where $Y \sim \text{Bern}(1/2)^m$. Because of the relation between the circuit and its associated distribution we can think of C as the distribution itself. Such distributions which can be described by a circuit C are the only distributions which can be generated efficiently, and we call them *sampleable distributions*.

4 The Statistical Difference Problem

Recall that given two distributions P and Q supported on Ω, we define the statistical difference metric between the two distributions by:

$$\delta(P,Q) = \frac{1}{2}\sum_{\omega\in\Omega}|P(\omega)-Q(\omega)|$$

We can equivalently write:

$$\delta(P,Q) = \max_{T:\Omega\to\{0,1\}}\{\mathbb{E}_{x\in P}[T(x)] - \mathbb{E}_{y\in Q}[T(y)]\}$$

The Statistical Difference problem, $\mathrm{SD}^{c,f}$, has two parameters, $0 \le c < f \le 1$ (c for "close", f for "far"). We define two sets of circuit (distribution) pairs:

1. $\mathrm{CLOSE}^c = \{(C_1,C_2)|\ \delta(C_1,C_2) \le c\}$
2. $\mathrm{FAR}^f = \{(C_1,C_2)|\ \delta(C_1,C_2) > f\}$

The Statistical Difference problem is then: given a circuit pair $(C_1,C_2) \in \mathrm{CLOSE}^c \cup \mathrm{FAR}^f$, decide if $(C_1,C_2) \in \mathrm{CLOSE}^c$ (a "YES" instance) or $(C_1,C_2) \in \mathrm{FAR}^f$ (a "NO" instance).

We don't know how to solve $\mathrm{SD}^{\frac{1}{3},\frac{2}{3}}$. However, the graph isomorphism problem is $\mathrm{GI} \le \mathrm{SD}^{\frac{1}{3},\frac{2}{3}}$. The suspicion is that that are no efficient algorithms for $\mathrm{SD}^{\frac{1}{3},\frac{2}{3}}$.

In 1997, Sahai and Vadhan [8] showed:

Theorem 1. *$SD^{\frac{1}{3},\frac{2}{3}} \equiv SD^{2^{-\sqrt{n}},1-2^{-\sqrt{n}}}$, i.e. there exists a polynomial time algorithm, which ,when given a circuit pair (C_1,C_2) produces a circuit pair (D_1,D_2) such that if $(C_1,C_2) \in CLOSE^{\frac{1}{3}}$ then $(D_1,D_2) \in CLOSE^{2^{-\sqrt{n}}}$ and if $(C_1,C_2) \in FAR^{\frac{2}{3}}$ then $(D_1,D_2) \in FAR^{1-2^{-\sqrt{n}}}$.*

In general for the result of Theorem 1 we require $f^2 > c$ (proof using Pinsker's lemma). The question of whether or not this still holds when $f^2 \le c$ is still open. For example we don't know whether or not $\mathrm{SD}^{\frac{1}{3},\frac{2}{3}} \equiv \mathrm{SD}^{0.49,0.5}$. From now on we consider the first case, i.e. we assume that $f^2 > c$, for example by choosing $c = \frac{1}{3}$ and $f = \frac{2}{3}$. For this case we denote $\mathrm{SD} \triangleq \mathrm{SD}^{\frac{1}{3},\frac{2}{3}}$.

The complement of $\mathrm{SD}^{c,f}$, $\overline{\mathrm{SD}}^{c,f}$, is the same problem, but with flipped outputs, i.e. given a circuit (or distribution) pair (C_1,C_2) we output "NO" if $(C_1,C_2) \in \mathrm{CLOSE}^c$ and "YES" if $(C_1,C_2) \in \mathrm{FAR}^f$.

Theorem 2. *$SD \equiv \overline{SD}$, i.e. there exists a polynomial time algorithm, which ,when given a circuit pair (C_1,C_2) produces a circuit pair (D_1,D_2) such that if $(C_1,C_2) \in CLOSE^{\frac{1}{3}}$ then $(D_1,D_2) \in FAR^{\frac{2}{3}}$ and if $(C_1,C_2) \in FAR^{\frac{2}{3}}$ then $(D_1,D_2) \in CLOSE^{\frac{1}{3}}$.*

We will prove Theorem 2 in the next lecture.

5 Historical Overview of the Statistical Difference Problem

The importance of the Statistical Difference problem originated in cryptography by Goldwasser and Micali [2]. One of the central ingredients in this paper is computational distance, which is a slight variation on statistical difference. The zero-knowledge paper [5] followed, and a later paper [6] showed that the Graph Nonisomorphism (GNI) problem is in SZK (statistical zero-knowledge).

5.1 Graph Isomorphism and Graph Nonisomorphism

The Graph Nonisomorphism problem is related to the simpler Graph Isomorphism (GI) problem. In the GI problem we are given two graphs $G = ([n], E)$ and $H = ([n], F)$, and we want to know if they are isomorphic (denoted $G \cong H$), i.e. if there exists a bijection $\pi : [n] \to [n]$, such that $(i, j) \in E$ iff $(\pi(i), \pi(j)) \in F$. If $G \cong H$ then given a mapping π it is easy to verify if it is an isomorphism. Therefore GI $\in$ NP.

Proving that two graphs are not isomorphic (GNI) is more challenging. There exists an interactive proof protocol for showing that G is not isomorphic to H: let there be two players, *verifier* and *prover*, both getting access to the two graphs $G_0 \triangleq G$, $G_1 \triangleq H$. The verifier chooses a random index $b \in \{0, 1\}$, and selects G_b. It then picks a random permutation $\pi : [n] \to [n]$, and uses it to relabel the vertices and obtain the graph $K = \pi(G_b)$. The verifier then sends K to the prover, which needs to guess if it came from G_0 or G_1, i.e. the prover tries to estimate b and send back to the verifier $\hat{b} = 0$ if it thinks that $b = 0$ (i.e. if $K \cong G_0$) and $\hat{b} = 1$ if it thinks that $b = 1$ (i.e. if $K \cong G_1$). The verifier then accepts if $\hat{b} = b$.

This protocol satisfies the zero-knowledge properties:

1. Completeness: If $G_0 \ncong G_1$ then an unbounded time prover can find $\hat{b} = b$ and send it to the verifier, which then accepts with probability 1.
2. Soundness: If $G_0 \cong G_1$, $I(K; b|G_0, G_1) = 0$, or equivalently $p(K|b = 0, G_0, G_1) = p(K|b = 1, G_0, G_1)$. Therefore the verifier accepts with probability 1/2.
3. Zero-knowledge: The verifier learned nothing other than $G_0 \ncong G_1$ (if that is indeed the case).

5.2 Back to Statistical Difference

Sahai and Vadhan [8] found that $\overline{\text{SD}}^{\frac{1}{3},\frac{2}{3}} \in$ SZK and that GNI $\leq \overline{SD}^{0,1}$, by using the two distributions $p(K|b = 0, G_0, G_1)$ and $p(K|b = 1, G_0, G_1)$. Note that while we cannot explicitly write these distributions, we can still sample from them.

Exercise 3. $\overline{SD}^{2^{-\sqrt{n}},1-2^{-\sqrt{n}}}$ *has a statistical "zero" knowledge proof (exponentially small knowledge rather than exactly zero).*

An additional theorem from [8] (will not be proven):

Theorem 4. $SD^{\frac{1}{3},\frac{2}{3}}$ *is SZK-complete.*

6 Additional Reading

For more information see survey of the Statistical Difference problem by Goldreich and Vadhan [9], available online at `http://www.wisdom.weizmann.ac.il/~oded/COL/entropy.pdf`.

References

[1] von Neumann, John. "Various techniques used in connection with random digits." John von Neumann, Collected Works 5 (1963): 768-770.

[2] Goldwasser, Shafi, and Silvio Micali. "Probabilistic encryption." Journal of computer and system sciences 28.2 (1984): 270-299.

[3] Blum, Manuel. "Independent unbiased coin flips from a correlated biased source—a finite state Markov chain." Combinatorica 6.2 (1986): 97-108.

[4] Santha, Miklos, and Umesh V. Vazirani. "Generating quasi-random sequences from semi-random sources." Journal of computer and system sciences 33.1 (1986): 75-87.

[5] Goldwasser, Shafi, Silvio Micali, and Charles Rackoff. "The knowledge complexity of interactive proof systems." SIAM Journal on computing 18.1 (1989): 186-208.

[6] Goldreich, Oded, Silvio Micali, and Avi Wigderson. "Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems." Journal of the ACM (JACM) 38.3 (1991): 690-728.

[7] Nisan, Noam, and David Zuckerman. "Randomness is linear in space." Journal of Computer and System Sciences 52.1 (1996): 43-52.

[8] Sahai, Amit, and Salil P. Vadhan. "A complete promise problem for statistical zero-knowledge." Proceedings 38th Annual Symposium on Foundations of Computer Science. IEEE, 1997.

[9] Goldreich, Oded, and Salil P. Vadhan. "On the complexity of computational problems regarding distributions (a survey)." Electronic Colloquium on Computational Complexity (ECCC). Vol. 18. 2011.