6.842 Randomness and Computation September 25, 2017

Homework 3

Lecturer: Ronitt Rubinfeld *Due Date: October 2, 2017*

Homework guidelines: You may work with other students, as long as (1) they have not yet solved the problem, (2) you write down the names of all other students with which you discussed the problem, and (3) you write up the solution on your own. No points will be deducted, no matter how many people you talk to, as long as you are honest. If you already knew the answer to one of the problems (call these "famous" problems), then let me know that in your solution writeup – it will not affect your score, but will help me in the future. It's ok to look up famous sums and inequalities that help you to solve the problem, but don't look up an entire solution.

The following problems are for your understanding – do not turn in.

1. Give an "approximate lower bound interactive proof protocol" by fixing the one given in Lecture 5.

2. (optional) Let $R(x, y) : \{0,1\}^n \times \{0,1\}^m \to \{0,1\}$ be a relation that is computable in time polynomial in n, m. Let L be a language $L = \{x \mid \exists y \text{ such that } R(x, y) = 1\}$. Let $\ell_x \equiv |\{y|R(x, y) = 1\}|$. Let A be a polynomial time algorithm (in n, m) that given x uniformly generates y for which $R(x, y) = 1$. Show that there is an "approximate upper bound" interactive proof protocol which allows a prover to convince a polynomial time (in $n, m, 1/\epsilon$) verifier that ℓ_x is not too much more than k: That is, given ϵ and x, if ℓ_x is at most k, the verifier should accept with probability at least $2/3$, and if ℓ_x is greater than $(1 + \epsilon)k$, the verifier should accept with probability at most $1/3$.

The following problems are to be turned in. Please write your solution for each problem on separate sheets of paper.

1. Given a probability distribution p over domain A such that $|A| = n$. Define the *collision probability* of p,

$$c(p) \equiv Pr_{i,j \in_p A}[i = j] = \sum_{i \in A} p_i^2$$

(where the notation $i \in_p A$ denotes that i is chosen according to distribution p from the domain A). Define the L_1 distance between p, q as

$$|p - q|_1 = \sum_{i \in A} |p_i - q_i|$$

Define the L_2 distance between p, q as

$$||p - q||_2 = (\sum_{i \in A} (p_i - q_i)^2)^{1/2}$$

(Note that L_1, L_2 distances are defined for any pairs of real-valued vectors, not just probability distribution vectors). Recall the Cauchy-Schwartz inequality that $|v|_1 \leq \sqrt{d}||v||_2$ (where d is the dimension of the vector). Let U_X denote the uniform distribution over set X.

Let H be a family of pairwise independent hash functions mapping A to T such that $|A| = n$ and $|T| = t$. Assume that the functions in H are indexed by elements in the set B (i.e., each element in B corresponds to exactly one hash function).

Let W be any subset of A. Let distribution q over $B \times T$ be defined as follows: Choose h uniformly from H and x chosen uniformly from W, then output the pair $< h, h(x) >$.

(a) (warmup!) For h chosen uniformly from H, say that x, y are a colliding pair if $h(x) = h(y)$. Show that the expected number of colliding pairs is $\binom{n}{2} \cdot \frac{1}{t}$

(b) For any distribution p over the set A, show that if $c(p) \leq (1+\epsilon^2)/|A|$ then $|p - U_A|_1 \leq \epsilon$.

(c) For q defined as above, show that $c(q) \leq \frac{1+|T|/|W|}{|B \times T|}$

(d) Using the previous two items, show that $|q - U_{B \times T}|_1 \leq \sqrt{|T|/|W|}$.

2. The NP-complete problem CIRCUIT-SAT takes as input a description of a boolean circuit C (assume that the gates are two-input "and", "or" gates or "not" gates) and asks if there is any set of inputs $x = x_1, \ldots, x_r$ such that $C(x) = 1$. So, $L_{\text{CIRCUIT-SAT}} = \{C \mid C \text{ describes a circuit with a satisfying assignment } x \text{ such that } C(x) = 1\}$. Suppose C is a description of a circuit which is *guaranteed* to either have only one solution or to have no solutions at all. Our goal in this problem is to show that determining whether $C \in L_{\text{CIRCUIT-SAT}}$ cannot be much easier than the general CIRCUIT-SAT problem.

Let us first define the problem Π as follows: Given circuit C which takes an r-bit input, a polynomial time computable function h mapping $\{0,1\}^r$ to a set of values T, and a value $\alpha \in T$, is there an input y to C such that $C(y) = 1$ and $h(y) = \alpha$?

We say that randomized one-sided error algorithm A *unique solves* Π, if for all inputs (C, h, α) with no satistying assignments y such that $C(y) = 1$, A outputs "no" (with probability 1), and for all inputs (C, h, α) with exactly one satisfying assignment y such that $C(y) = 1$, A outputs "yes" with probability at least 1/2. Note that for any (C, h, α) which has two or more satisfying assignments, "no" or "yes" is a perfectly legal answer.

Prove that if there is a randomized one-sided error polynomial time algorithm A which unique solves Π, then $RP = NP$. To do this, design an algorithm $B \in RP$ that decides membership in CIRCUIT-SAT, using oracle calls to A.