CS 744: Pseudorandomness Generators

Lecture 11: Computational Indistinguishability

January 25, 2015

In probability theory, we have various theorems which hold "with probability 1", and capture some properties of random sequences - for example, the strong law of large numbers, which has the consequence that almost every binary sequence is *normal*, and the law of iterated logarithm, which *upper bounds* the speed of convergence of a random binary sequence to its expected behaviour, implying that if a binary sequence has approximately $n/2$ zeroes from very small n, then it is not random.

However, in computational complexity, we abstract away from this notion and define the notion of an (algorithmic) statistical test, which allows the existence of *pseudorandom* distributions.

A *statistical test* on $\{0,1\}^N$ is any algorithm A which outputs 0 or 1. An N-*source* is a probability distribution on $\{0,1\}^N$. We denote the uniform distribution on $\{0,1\}^N$ by U_N. For a statistical test A and an N-source S, we define $P(A;S)$ by

$$P(A;S) \;=\; \text{Probability}(B \sim S \mid A(B) = 1) \;=\; S(B \mid A(B) = 1).$$

Definition 1. *We say that an N-source S passes the statistical test A with tolerance $\epsilon > 0$ if*

$$|P(A;S) - P(A;U_N)| \leq \epsilon.$$

Instead of arbitrary statistical test, we could reformulate the notion of being computationally indistinguishable from the uniform distribution using the notion of *predictors*.

Definition 2. *For $k \in \{0,\ldots,N-1\}$, a* predictor $A_k : \{0,1\}^N \to \{0,1\}$ *is an algorithm that depends only on the first k bits of the input.* i.e. *for any $B \in \{0,1\}^N$ and $C, D \in \{0,1\}^{N-k}$,*

$$A(B_1 \ldots B_k \cdot C) = A(B_1 \ldots B_k \cdot D).$$

A prediction test A_k^P *is defined as*

$$A_k^P \;=\; A_k(B) \oplus B_{k+1} \oplus 1 \;=\; \begin{cases} 1 & \text{if } A_k(B) = B_{k+1} \\ 0 & \text{otherwise.} \end{cases}$$

These two notions lead to almost equivalent notions of computational indistinguishability. Clearly, every a prediction test which distinguishes S from U_N with tolerance ϵ is a statistical test which distinguishes it from U_N with the same tolerance. We have the following by way of the converse.

Lemma 3. *Let S be an N-source which fails test A with tolerance $\epsilon > 0$. Then there is a $k \in [0, N-1]$ and a prediction test A_{k+1}^P on which S fails with tolerance ϵ/N.*

The proof proceeds by a widely applicable technique called the "hybrid argument", so called because it constructs "hybrids" of S and U_N to establish the result.

Proof. By complementing the output of A if necessary, we can assume that

$$P(A; S) - P(A; U_N) \geq \epsilon.$$

Consider the algorithms $F_1, \ldots, F_N$ where F_k works as follows. Given $B \in \{0,1\}^N$, it flips $N-k$ coins to produce a bit string $B'_{k+1} \ldots B'_N$. Then it outputs $A(B_1 \ldots B_k\ B'_{k+1} \ldots B'_N)$. Clearly, $P(F_0; S) = P(A; U_N)$ and $P(F_N; S) = P(A; S)$. Then, by assumption, $P(F_N; S) - P(F_0; S) \geq \epsilon$.

Now, writing the above as a telescoping sum, we have

$$\sum_{k=0}^{N-1} P(F_{k+1}; S) - P(F_k; S) \geq \epsilon,$$

whence it follows that there is a $k \in [0, N-1]$ where

$$P(F_{k+1}; S) - P(F_k; S) \geq \frac{\epsilon}{N}.$$

Now, we need to find a prediction test A_k^P with

$$P(A_k^P; S) - P(A_k^P; U_N) \geq \frac{\epsilon}{N}.$$

Define the predictor $A_k(B) = F_k(B)$. The corresponding predictor test $A1_k^P(B) = F_k(B) \oplus B'_{k+1} \oplus 1$ can be written equivalently as

$$A2_k^P(B) = F_k(B) \oplus B'_{k+1} \oplus B_{k+1}?$$

It is certainly true that

$$P(A1_k^P(B); S) = P(A2_k^P(B); S).$$