Yale University
Department of Computer Science

Computational Measures of Similarity for Probabilistic Functions

Michael J. Fischer and Sophia A. Paleologou

YALEU/DCS/TR-1111
June 17, 1996

Computational Measures of Similarity for Probabilistic Functions

Michael J. Fischer
Department of Computer Science
Yale University
fischer-michael@cs.yale.edu

Sophia A. Paleologou
KenCast, Inc.
Yale University
paleologou-sophia@cs.yale.edu

1 Introduction

Randomized computations play a central role in theoretical computer science today, arising naturally in the study of efficient algorithms, pseudorandom number generators, zero knowledge interactive proofs, cryptographic protocols, and probabilistically checkable proofs.

A randomized algorithm computes a probabilistic function f, where for each x, $f(x)$ is a probability distribution. Most applications of randomized algorithms, however, do not fully specify the function f to be computed; rather, they only require that f approximates some target function g. The output distribution of a probabilistic primality test, for example, should approximate the true primality predicate is_prime(x). Similarly, the output distribution of a pseudorandom number generator should approximate $u(x)$, the uniform distribution on binary strings of length $|x|$.

Both of these examples share the notion of a randomized algorithm computing an approximation to a desired function. However, the particular notion of approximation differs in the two cases. For the primality test, the probability that $f(x) \neq$ is_prime(x) is required to be exponentially small in $|x|$. For pseudorandom number generators, it is required that $f(x)$ "looks like" $u(x)$ to polynomial-time Turing machines, a much weaker condition. Disparate notions of approximation that have appeared in the literature are *perfect* (equality), *statistical*, and *computational* indistinguishability.

In this paper, we generalize and extend computational indistinguishability to allow for arbitrary classes of judges (distinguishers), arbitrary numbers of samples and arbitrary families of tolerance functions defining the distinguishing threshold. We also generalize statistical indistinguishability to arbitrary families of tolerance functions. We compare our various notions of indistinguishability with each other and investigate how restrictions on the indistinguishability parameters affect the corresponding indistinguishability relation.

Of particular interest to our general indistinguishability framework is the number of samples given to a judge. Under the usual definition of computational indistinguishability, the judge is given one sample from either $f(x)$ or $g(x)$. The judge is said to distinguish between f and g if its output behavior varies significantly depending on which distribution the sample was drawn from.

A natural question to ask is, "What happens to the distinguishing power of a family of judges when given two samples from the (same) unknown distribution?" In the case of families of polynomial-size circuits, if a judge distinguishes f from g given two samples, then a simple modification of that judge distinguishes f from g given only one sample. In light of this observation,

one might be tempted to conjecture that changes in the number of samples would affect the distinguishing power of arbitrary judges only in small quantitative ways. It is particularly surprising that the conjecture does not hold for Turing machine judges. To the contrary, there exists a simple and fast Turing machine judge that, given two samples, can distinguish pairs of probabilistic functions that are otherwise indistinguishable by *any* polynomial time Turing machine given one sample.

The additional complexity of two-sample indistinguishabilities is reflected in the techniques needed to construct pairs of statistically different but judge indistinguishable probabilistic functions. One-sample judges give rise to linear equations over unknown probabilities. In the case of two or more samples, the corresponding equations are non-linear, and the techniques from linear algebra that solve the one-sample case [PS82] do not apply. Instead, deep results from algebraic topology [Mas89] are needed.

At the heart of many of our indistinguishability results lies a standard diagonalization, a rather basic tool that we borrow from the realm of recursive function theory [Rog67]. Tools from probability theory [DeG86] and statistics [Pol84] are naturally also of great use when dealing with probabilistic functions.

We informally summarize our indistinguishability results below.

1. For any recursively presentable class $\mathcal{J}$ of Turing machine judges, we can "shrink" a probabilistic function f in such a way that the new function has arbitrarily small (yet growing) support at every x, but it is indistinguishable from the original f.

2. By sufficiently shrinking u as in 1, we construct a probabilistic function g that is statistically different from the uniform function u, yet indistinguishable from u for the judges in $\mathcal{J}$ given one sample.

3. We present a simple judge $J_{2\alpha}$ that distinguishes u from the function g of 2 given two samples.

4. For any recursively presentable family $\mathcal{J}$ of Turing machine judges and any arbitrary number of samples, we construct a pair of computable probabilistic functions f, g that are indistinguishable for the judges in $\mathcal{J}$ from the given number of samples.

5. For any pair f, g of statistically different probabilistic functions, we present a judge that distinguishes them given sufficiently many samples.

2 Probabilistic Functions

A probabilistic function is a probabilistic analog to an ordinary deterministic function.

Definition 1 A *probabilistic function (or probabilistic ensemble) f with domain A and range B*, denoted $f : A \to B$, is a function mapping every element $x \in A$ to a probability distribution $f(x)$ over B.

In the literature of pseudorandom number generators and zero knowledge interactive proofs, a probabilistic function $f : A \to B$ is often denoted $\{f(x)\}_{x \in A}$. This notation suggests that a probabilistic function can also be thought of as a collection of probability distributions over the set B, one distribution for every $x \in A$.

Our formal model of probabilistic computation is the probabilistic Turing machine [Gil77]. It is obtained from the usual Turing machine when given access to an unbiased coin or, equivalently, to a random tape containing an infinite sequence of bits independently and uniformly distributed in $\{0,1\}$. The probability associated with any given finite computation is $1/2^k$, where $k \in \mathcal{N}$ is the number of random bits read during the computation. Then, the probability that the machine does not halt on a certain input is equal to one minus the probability of all finite computations. We use the symbol $\perp$ to denote the output of an infinite computation. Without loss of generality, we also set $\Sigma = \{0,1\}$.

A probabilistic Turing machine M computes a probabilistic function. On input $x \in \Sigma^*$, M computes a probability distribution over all possible output values $y \in \Sigma^* \cup \{\perp\}$. Computable probabilistic functions, however, are only a special case of probabilistic functions. Just as there exist non-computable deterministic functions, there also exist non-computable probabilistic functions.

Definition 2 Let $f : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be a probabilistic function. f is *computable* iff there exists a probabilistic Turing machine M_f such that, for every x, $M_f(x) = f(x)$. f is *total* iff, for every x, $\Pr[f(x) = \perp] = 0$.

Definition 3 A family $\mathcal{F}$ of probabilistic (resp. deterministic) functions is *recursively presentable* iff there exists an effective enumeration of probabilistic (resp. deterministic) Turing machines $M_1, M_2, \ldots$ computing exactly the functions in $\mathcal{F}$.

Of special importance to our indistinguishability theory is the *uniform function* $u : \Sigma^* \to \Sigma^* \cup \{\perp\}$. For every $x \in \Sigma^*$, $u(x)$ assigns equal probability to all strings $y \in \Sigma^*$ that have the same length as x and probability 0 to all other strings. Obviously, u is total and computable.

3 Generalized Indistinguishabilities

In this section, we generalize the four standard notions of indistinguishability studied in the literature of pseudo-random number generators [Yao82, BM84], probabilistic encryption [GM84], and zero-knowledge interactive proofs [GMR89, GMW91].

3.1 Statistical Indistinguishabilities

We generalize standard statistical indistinguishability by parametrizing the distinguishing threshold and allowing for arbitrary thresholds other than inverse polynomials.

Definition 4 A function $\varepsilon : \Sigma^* \to (0,1)$ is called a *tolerance function.*

Definition 5 Let $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be two probabilistic functions. The *statistical difference of* f *and* g is a function $\mathit{Sdiff} : \Sigma^* \to [0,1]$ such that, for every $x \in \Sigma^*$,

$$\mathit{Sdiff}(x) \quad = \quad \sum_{\alpha \in \Sigma^* \cup \{\perp\}} \Big| \Pr[f(x) = \alpha] - \Pr[g(x) = \alpha] \Big| \tag{1}$$

Definition 6 Let $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ be a family of tolerance functions. Two probabilistic functions $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ are *statistically indistinguishable at tolerance* $\mathcal{E}$ iff, for all $\varepsilon \in \mathcal{E}$ and all sufficiently long strings $x \in \Sigma^*$,

$$\mathit{Sdiff}(x) < \varepsilon(x) \tag{2}$$

Two probabilistic functions that are not statistically indistinguishable at tolerance $\mathcal{E}$ are said to be *(weakly) statistically distinguishable at tolerance* $\mathcal{E}$*.*

As many of our results involve a stronger notion of statistical distinguishability than simply the negation of statistical indistinguishability, we also introduce *strong* statistical distinguishability.

Definition 7 Let $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ be a family of tolerance functions. Two probabilistic functions $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ are *strongly statistically distinguishable at tolerance* $\mathcal{E}$ iff there exists an $\varepsilon \in \mathcal{E}$ such that, for all sufficiently long strings $x \in \Sigma^*$,

$$\mathit{Sdiff}(x) \geq \varepsilon(x) \tag{3}$$

3.2 Judge Indistinguishabilities

The literature considers two kinds of computational indistinguishabilities. One is based on polynomial-time probabilistic Turing machines and the other on polynomial-size circuits. In both cases, the computational devices, which we call *judges*, are used as statistical tests. If the statistical behavior of a judge varies significantly depending on the distribution from which its input is drawn, then we say that the judge distinguishes the corresponding functions. Two functions are computationally indistinguishable if no judge in the family can distinguish them.

We generalize computational indistinguishability by considering arbitrary families of judges. We also parameterize the distinguishing threshold, as we did for statistical indistinguishability, and present the judges with a set of samples of arbitrary size instead of just a single sample.

A judge J is himself a probabilistic function which we use to distinguish between two probabilistic functions f and g. J takes two inputs, an index x and a set of samples y. The individual samples in y are independently drawn either all from $f(x)$ or all from $g(x)$ and are separated with #'s. The judge's distinguishing power is measured on the absolute difference between the probability of his outputting 0 when he sees a sample set from f and the probability of his outputting 0 when he sees a sample set from g. The line between distinguishing and not distinguishing is set by a family of tolerance functions $\mathcal{E}$. Judges are not necessarily computable, total or polynomially bounded.

Definition 8 A probabilistic function $J : \Sigma^* \times A \to \Sigma^* \cup \{\perp\}$ with

$$A \;=\; \Big\{\alpha_1\#\alpha_2\#\cdots\#\alpha_k \mid k \in \mathcal{N} \text{ and } \alpha_i \in \Sigma^* \cup \{\perp\}, 1 \leq i \leq k\Big\} \tag{4}$$

is called a *judge*.

Definition 9 Let $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be two probabilistic functions, $a : \Sigma^* \to \mathcal{N}$ a function and J a judge. The *distinguishing power of* J *between* f *and* g *from* $a(x)$ *samples* is a function $\mathit{Dpower}_{J,a} : \Sigma^* \to [0,1]$ such that, for every $x \in \Sigma^*$,

$$\begin{aligned}&\mathit{Dpower}_{J,a}(x) = \\ &\quad\Big|\Pr[J(x, Y_1\#Y_2\#\ldots\#Y_{a(x)}) = 0] - \Pr[J(x, Z_1\#Z_2\#\ldots\#Z_{a(x)}) = 0]\Big|\end{aligned} \tag{5}$$

where Y_i, Z_i are random variables independently distributed according to $f(x), g(x)$ respectively, for all $i = 1, 2 \ldots, a(x)$.

Definition 10 Let $\mathcal{J}$ be a class of judges, $a : \Sigma^* \to \mathcal{N}$ a function and $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ a family of tolerance functions. Two probabilistic functions $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ are *$\mathcal{J}$-indistinguishable from $a(x)$ samples at tolerance $\mathcal{E}$* iff, for all judges $J \in \mathcal{J}$, functions $\varepsilon \in \mathcal{E}$ and sufficiently long strings $x \in \Sigma^*$,

$$Dpower_{J,a}(x) < \varepsilon(x) \tag{6}$$

Two probabilistic functions that are not $\mathcal{J}$-indistinguishable from $a(x)$ samples at tolerance $\mathcal{E}$ are said to be *(weakly) $\mathcal{J}$-distinguishable from $a(x)$ samples at tolerance $\mathcal{E}$.*

Definition 11 Let $\mathcal{J}$ be a class of judges, $a : \Sigma^* \to \mathcal{N}$ a function and $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ a family of tolerance functions. Two probabilistic functions $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ are *strongly $\mathcal{J}$-distinguishable from $a(x)$ samples at tolerance $\mathcal{E}$* iff there exists a judge $J \in \mathcal{J}$ and a function $\varepsilon \in \mathcal{E}$ such that, for all sufficiently long strings $x \in \Sigma^*$,

$$Dpower_{J,a}(x) \geq \varepsilon(x) \tag{7}$$

3.3 Downward Closure of Tolerance Functions

All standard indistinguishabilities—perfect, statistical, computational—are equivalence relations over the set of computable probabilistic functions. Furthermore, perfect indistinguishability is a refinement of statistical, statistical is a refinement of computational by families of polynomial-size circuits, and computational by families of polynomial-size circuits is a refinement of computational by polynomial-time Turing machines.

Generalized statistical and judge indistinguishabilities defined on arbitrary families of tolerance functions are not necessarily equivalence relations, as they do not necessarily satisfy transitivity. A special property of tolerance families, *downward closure*, guarantees that our indistinguishabilities are equivalence relations.

Definition 12 Let $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ be a family of tolerance functions. $\mathcal{E}$ is *downward closed* iff, for all $\varepsilon \in \mathcal{E}$, there exists an $\hat{\varepsilon} \in \mathcal{E}$ such that, for all sufficiently long strings x,

$$\hat{\varepsilon}(x) < \frac{1}{2} \cdot \varepsilon(x) \tag{8}$$

Theorem 1 *Let $\mathcal{F} \subseteq \{f : \Sigma^* \to \Sigma^* \cup \{\perp\}\}$ be a family of probabilistic functions. Let $\mathcal{J}$ be a class of judges, $a : \Sigma^* \to \mathcal{N}$ a function and $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ a family of tolerance functions. If $\mathcal{E}$ is downward closed, then*

(a) statistical indistinguishability at tolerance $\mathcal{E}$ and

(b) $\mathcal{J}$-indistinguishability from $a(x)$ samples at tolerance $\mathcal{E}$

are equivalence relations over $\mathcal{F}$.

4 One-Sample Judge Indistinguishabilities

4.1 Statistical Refines One-Sample Judge Indistinguishabilities

In the same way that standard statistical indistinguishability refines computational indistinguishabilities [GMW91], our generalized statistical indistinguishability refines any one-sample judge indistinguishability of the same tolerance.

Theorem 2 *Let $\varepsilon : \Sigma^* \to (0,1)$ be a tolerance function. For every pair $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ of probabilistic functions, if a judge J weakly (resp. strongly) distinguishes between f and g from one sample at tolerance $\{\varepsilon\}$, then f and g are weakly (resp. strongly) statistically distinguishable at tolerance $\{\varepsilon\}$.*

Corollary 3 *Let $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ be a downward closed family of tolerance functions. For every class of judges $\mathcal{J}$, statistical indistinguishability at tolerance $\mathcal{E}$ is a refinement of $\mathcal{J}$-indistinguishability from one sample at tolerance $\mathcal{E}$.*

Proof: Immediate from theorems 1 and 2. ■

4.2 Shrinkable Probabilistic Functions

Intuitively, a probabilistic function f is $m(x)$-shrinkable, when we can compute another probabilistic function g that is indistinguishable from the original f, although the support of $g(x)$ has size at most $m(x)$ for every x.[1] In theorem 4, we show that, under some basic computability restrictions, every computable probabilistic function is $m(x)$-shrinkable given an m that grows without bound.

Definition 13 Let $\mathcal{J}$ be a class of judges, $\mathcal{E}$ a family of tolerance functions and $m : \Sigma^* \to \mathcal{N}$ a function. A probabilistic function $f : \Sigma^* \to \Sigma^* \cup \{\perp\}$ is *$m(x)$-shrinkable for $\mathcal{J}$ at tolerance $\mathcal{E}$* iff there exists a probabilistic function $g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ such that:

1. for every $x \in \Sigma^*$, $Support_{g(x)} \subseteq Support_{f(x)}$ and $|Support_{g(x)}| \leq m(x)$;
2. f and g are $\mathcal{J}$-indistinguishable from one sample at tolerance $\mathcal{E}$.

If g is also computable, then f is *recursively $m(x)$-shrinkable for $\mathcal{J}$ at tolerance $\mathcal{E}$*.

Theorem 4 *Let $f : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be a total computable probabilistic function such that, for every $x \in \Sigma^*$, $f(x)$ has finite support, and the family $\mathcal{S} = \{Support_{f(x)}\}_{x \in \Sigma^*}$ is computable. Let $\mathcal{J}$ be a recursively presentable class of total judges, $\mathcal{E}_J$ a recursively presentable class of tolerance functions and $m : \Sigma^* \to \mathcal{N}$ a computable function with $\liminf m(x) = \infty$. Then, f is recursively $m(x)$-shrinkable for $\mathcal{J}$ at tolerance $\mathcal{E}_J$.*

Proof sketch: Diagonalization gives rise to a feasible and highly underconstrained system of linear equations. There exists a basic feasible solution with at most $m(x)$ non-zero coordinates [PS82]. We can easily compute it and use it to define g. ■

Corollary 5 *For every recursively presentable class of total judges $\mathcal{J}$, recursively presentable family of tolerance functions $\mathcal{E}_J$, and computable function $m : \Sigma^* \to \mathcal{N}$ with $\liminf m(x) = \infty$, the uniform function u is recursively $m(x)$-shrinkable for $\mathcal{J}$ at tolerance $\mathcal{E}_J$.*

[1] Given a probability distribution P over a set B, the *support of P*, denoted $Support_P$, is the subset of B containing all elements assigned positive probability by P.

4.3 Support Size and Statistical Difference

In this section, we show that the uniform function is statistically different from any other probabilistic function whose support is sufficiently smaller than the support of the uniform at every x. Having already proven that u is $m(x)$-shrinkable, we combine the two to compute probabilistic functions of arbitrary (growing) supports that are indistinguishable from the uniform for recursively presentable judges, one sample, and arbitrary tolerance.

Theorem 6 *Let $\varepsilon_s : \Sigma^* \to (0,1)$ be a tolerance function. Let $u : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be the uniform function, and let $g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be a probabilistic function such that, for infinitely (resp. all but finitely) many $x \in \Sigma^*$,*

$$| \, Support_{g(x)} \, | \quad \leq \quad \frac{1}{\varepsilon_s(x) + \frac{1}{2^{|x|}}} \tag{9}$$

Then, u and g are weakly (resp. strongly) statististically distinguishable at tolerance $\{\varepsilon_s\}$.

Corollary 7 *Let $\mathcal{J}$ be a recursively presentable class of total judges, $\mathcal{E}_J$ a recursively presentable family of tolerance functions and $u : \Sigma^* \to \Sigma^* \cup \{\perp\}$ the uniform function. For every computable function $\varepsilon_s : \Sigma^* \to (0,1)$ with $\limsup \varepsilon_s(x) = 0$, there exists a total computable probabilistic function $g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ such that:*

1. *u and g are $\mathcal{J}$-indistinguishable from one sample at tolerance $\mathcal{E}_J$;*
2. *u and g are strongly statistically distinguishable at tolerance $\{\varepsilon_s\}$.*

Proof: Consider the function $m : \Sigma^* \to \mathcal{N}$ such that, for every $x \in \Sigma^*$,

$$m(x) = \left\lfloor \frac{1}{\varepsilon_s(x) + \frac{1}{2^{|x|}}} \right\rfloor \tag{10}$$

Since m is computable and $\liminf m(x) = \infty$, we can apply corollary 5 to m. The result follows from theorem 6. ■

Corollary 8 *Let $\mathcal{J}$ be a recursively presentable class of total judges, and let $\mathcal{E} \subseteq \{\varepsilon : \Sigma^* \to (0,1)\}$ be a recursively presentable and downward closed family of tolerance functions. Restricted to total computable probabilistic functions, statistical indistinguishability at tolerance $\mathcal{E}$ is a strict refinement of $\mathcal{J}$-indistinguishability from one sample at tolerance $\mathcal{E}$.*

Proof: Immediate from corollaries 3 and 7. ■

5 Many-Sample Judge Indistinguishabilities

5.1 Two Samples

It is well-known that circuit judges are insensitive to changes in the size of the sample set. The distinguishing power of families of polynomial-size circuits, for example, remains the same as the

number of samples increases from one to polynomially many [GMR89]. In sharp contrast, we show that the distinguishing power of Turing-machine judges is greatly affected by the slightest change in the size of the sample set.

Consider the judge $J_{2\alpha}$ defined by:

$$J_{2\alpha}(x, \alpha_1 \# \alpha_2) = \begin{cases} 0 & \text{if } \alpha_1 = \alpha_2 \\ 1 & \text{otherwise} \end{cases} \tag{11}$$

for all $x, \alpha_1, \alpha_2 \in \Sigma^*$. Judge $J_{2\alpha}$ is total and computable.

Theorem 9 *Let $\varepsilon_J : \Sigma^* \to (0,1)$ be a tolerance function. Let $u : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be the uniform function, and let $g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ be a probabilistic function such that, for infinitely (resp. all but finitely) many $x \in \Sigma^*$,*

$$| \, Support_{g(x)} \, | \quad \leq \quad \frac{1}{\sqrt{\varepsilon_J(x) + \frac{1}{2^{2|x|}}}} \tag{12}$$

Then, judge $J_{2\alpha}$ weakly (resp. strongly) distinguishes between u and g from two samples at tolerance $\{\varepsilon_J\}$.

Corollary 10 *Let $\mathcal{J}$ be a recursively presentable class of total judges with $J_{2\alpha} \in \mathcal{J}$, $\mathcal{E}_J$ a recursively presentable family of tolerance functions, and $u : \Sigma^* \to \Sigma^* \cup \{\perp\}$ the uniform function. For every computable tolerance function $\varepsilon_2 : \Sigma^* \to (0,1)$ with $\limsup \varepsilon_2(x) = 0$, there exists a total computable probabilistic function $g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ such that:*

1. *u and g are $\mathcal{J}$-indistinguishable from one sample at tolerance $\mathcal{E}_J$;*
2. *u and g are strongly $\mathcal{J}$-distinguishable from two samples at tolerance $\{\varepsilon_2\}$.*

Proof: Consider the function $m : \Sigma^* \to \mathcal{N}$ such that, for every $x \in \Sigma^*$,

$$m(x) = \left\lfloor \frac{1}{\sqrt{\varepsilon_2(x) + \frac{1}{2^{2|x|}}}} \right\rfloor \tag{13}$$

Since m is computable and $\liminf m(x) = \infty$, we can apply corollary 5 to m. The result follows from theorem 9. ■

Corollary 11 *Let $\mathcal{J}$ be a recursively presentable class of total judges with $J_{2\alpha} \in \mathcal{J}$, and let $\mathcal{E} \subseteq \{\varepsilon_2 : \Sigma^* \to (0,1)\}$ be a recursively presentable and downward closed family of tolerance functions. Restricted to total computable probabilistic functions, $\mathcal{J}$-indistinguishability from two samples at tolerance $\mathcal{E}$ is a strict refinement of $\mathcal{J}$-indistinguishability from one sample at tolerance $\mathcal{E}$.*

Proof: Immediate from corollaries 3 and 10. ■

5.2 Arbitrarily Many Samples

Theorem 12 *Let $\mathcal{J}$ be a recursively presentable class of total judges, $a : \mathcal{N} \to \mathcal{N}$ a computable function and $\mathcal{E}_J$ a recursively presentable family of tolerance functions. Let $\mathcal{S} = \{S_x\}_{x\in\Sigma^*}$ be a computable family of finite subsets of Σ^*, and let $\varepsilon_s : \Sigma^* \to (0,1)$ be a computable tolerance function such that, for infinitely (resp. all but finitely) many $x \in \Sigma^*$, $\varepsilon_s(x) < 2/|S_x|$. Then, there exist two total computable probabilistic functions $f, g : \Sigma^* \to \Sigma^* \cup \{\perp\}$ such that:*

1. *for all $x \in \Sigma^*$, $Support_{f(x)} \subseteq S_x$ and $Support_{g(x)} \subseteq S_x$;*
2. *f and g are $\mathcal{J}$-indistinguishable from $a(x)$ samples at tolerance $\mathcal{E}_J$;*
3. *f and g are weakly (resp. strongly) statistically distinguishable at tolerance $\{\varepsilon_s\}$.*

Proof sketch: Diagonalization gives rise to a continuous, multi-dimensional function Φ. By applying the Borsuk-Ulam theorem to Φ (see appendix), we get a pair $\mathbf{p}, \mathbf{q}$ of antipodal points on a simplex that are mapped to the same point through Φ. We can closely approximate those points and use them to define f and g. ■

5.3 Many Samples Allow Arbitrarily Fine Discrimination

In section 4, we showed that statistical indistinguishabilities refine one-sample judge indistinguishabilities at the same tolerance $\mathcal{E}$. Here, we show that the distinguishing power of judges can be made arbitrarily large by increasing the number of samples sufficiently. In particular, if f and g are statistically distinguishable at *some* tolerance $\{\varepsilon_s\}$, then they are judge distinguishable at *any* arbitrarily small tolerance $\{\varepsilon_J\}$, provided the judge is presented with enough samples from $f(x)$ or $g(x)$. The number of samples required depends on ε_s and ε_J as well as on the size of the support of $f(x)$ and $g(x)$.

The distinguishing judge J_d depends on f and a threshold function d and is defined in figure 1. J_d is presented with a set of $t(x)$ samples, either all drawn from $f(x)$ or all drawn from $g(x)$. It generates a second set of $t(x)$ samples from $f(x)$ and computes a frequency table for the input sample set and for the generated sample set. If the tables are similar enough, as defined by the threshold $d(x)$, J_d outputs 0. Otherwise it outputs 1. We show that, given a sufficient number of samples, J_d can distinguish f from any other statistically different function g.

Theorem 13 *Let $\varepsilon_s, \varepsilon_J : \Sigma^* \to (0,1)$ be tolerance functions. Let $f, g : \Sigma^* \to \Sigma^* \cup \{0,1\}$ be two total probabilistic functions that are weakly (resp. strongly) statistically distinguishable at tolerance $\{\varepsilon_s\}$ and such that, for every $x \in \Sigma^*$, $|Support_{f(x)} \cup Support_{g(x)}| \leq k(x)$. Then, there exist functions $d : \Sigma^* \to (0,1)$ and $t : \Sigma^* \to \mathcal{N}$ such that judge J_d weakly (resp. strongly) distinguishes between f and g from $t(x)$ samples at tolerance $\{\varepsilon_J\}$. If f and ε_s are computable, then judge J_d is also computable.*

Proof: We define d and t as follows. For all $x \in \Sigma^*$,

$$d(x) = \left\lfloor \frac{\varepsilon_s(x)}{2} \right\rfloor \text{ and } t(x) = \left\lceil \frac{2 \cdot k(x)^2 \cdot \left[\log(2\sqrt{2}) - \log(\sqrt{2} - \sqrt{1 + \varepsilon_J(x)})\right]}{\log e \cdot d(x)^2} \right\rceil \tag{14}$$

The result follows from Höffding's inequality (see appendix). ■

Input:
x, $\alpha_1\#\alpha_2\#\cdots\#\alpha_t$ with $x, \alpha_i \in \Sigma^*$;
Description:
draw k samples $\beta_1, \beta_2, \cdots, \beta_t$ from $f(x)$;
for every string α in the sample sets
set $\hat{f}_\alpha \leftarrow$ number of α's in $\alpha_1\#\alpha_2\#\cdots\#\alpha_t$;
set $\hat{h}_\alpha \leftarrow$ number of α's in $\beta_1\#\beta_2\#\cdots\#\beta_t$;
if $\sum_\alpha |\hat{f}_\alpha - \hat{h}_\alpha|/t < d(x)$ then output 0;
else output 1.

Figure 1: Judge J_d.

6 Related Work

In related work, Goldreich and Krawczyk prove the existence of sparse pseudorandom probabilistic functions that are not statistically close to any distribution induced by probabilistic polynomial-time algorithms [GK92]. By applying their results to zero-knowledge interactive proofs, they show that there exist protocols that are zero-knowledge in the original sense of the term, but auxiliary-input zero-knowledge.

Ostrovsky and Wigderson compare the notions of statistical and polynomial indistinguishability under the assumption that one-way functions do not exist [OW93]. They conclude that zero-knowledge is trivial under this assumption, as only languages in BPP can have zero-knowledge proofs.

Meyer separates the four standard indistinguishabilities from the zero-knowledge literature by constructing appropriate pairs of computable probabilistic functions [Mey94]. His separation proofs make use of a method for deterministically simulating in space S^2 an S space-bounded probabilistic Turing machine with running time bounded by 2^S.

7 Future Directions

One interesting problem that arises in our general indistinguishability framework is whether the distinguishing power of Turing machine judges always (or infinitely often) improves as the number of samples given to them increases from k to $(k+1)$, $k \in \mathcal{N}$. Most importantly, however, it would be interesting to apply our results to the domains where indistinguishabilities were originally defined; that is, randomized algorithms, pseudorandom number generators, zero-knowledge interactive proofs, etc. Results of this flavor were presented in [KR88, Bac87], where randomized algorithms were used to judge the quality of certain pseudorandom number generators.

References

[Bac87] E. Bach. Realistic Analysis of Some Randomized Algorithms. In *Proc. of 19th ACM Symp. on Theory of Computing*, pages 453–461, 1987.

[BM84] M. Blum and S. Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *SIAM Journal of Computing*, 13(4):850–864, November 1984.

[DeG86] M. H. DeGroot. *Probability and Statistics*. Addison-Wesley, second edition, 1986.

[Gil77] J. Gill. Computational Complexity of Probabilistic Turing Machines. *SIAM Journal of Computing*, 6(4):675–695, 1977.

[GK92] O. Goldreich and H. Krawczyk. Sparse Pseudorandom Distributions. *Random Structures and Algorithms*, 3(2):163–174, 1992.

[GM84] S. Goldwasser and S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.

[GMR89] S. Goldwasser, S. Micali, and C. Rackoff. The Knowledge Complexity of Interactive Proof Systems. *SIAM Journal of Computing*, 18(1):186–208, February 1989.

[GMW91] O. Goldreich, S. Micali, and A. Wigderson. Proofs that Yield Nothing But Their Validity or All Languages in NP Have Zero-Knowledge Proof Systems. *Journal of the Association of Computing Machinery*, 38(1):691–729, July 1991.

[Hoe63] W. Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[KR88] H.J. Karloff and Prabhakar Raghavan. Randomized Algorithms and Pseudorandom Numbers. In *Proc. of 20th ACM Symp. on Theory of Computing*, pages 310–321, 1988.

[Mas89] W. S. Massey. *Algebraic Topology: An Introduction*. Springer-Verlag, 1989.

[Mey94] B. Meyer. Constructive separation of classes of indistinguishable ensembles. In *Proc. of Structures*, 1994.

[OW93] R. Ostrovsky and A. Wigderson. One-Way Functions are Essential for Non-Trivial Zero-Knowledge. In *Proc. of 34th IEEE Symp. on Foundations of Computer Science*, pages 3–17, 1993.

[Pol84] D. Pollard. *Convergence of Stochastic Processes*. Springer Series in Statistics. Springer-Verlag, 1984.

[PS82] C. H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization, Algorithms and Complexity*. Prentice Hall, 1982.

[Rog67] H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill Book Company, 1967.

[Yao82] A. C. Yao. Theory and Applications of Trapdoor Functions. In *Proc. of 23th IEEE Symp. on Foundations of Computer Science*, pages 80–91, 1982.

A Appendix

A.1 The Borsuk-Ulam Theorem

Consider the two objects S^m and $\hat{S}^m$, $m \in \mathcal{N}$:

$$S^m = \left\{ \mathbf{x} \in \mathcal{R}^{m+1} \mid \sum_{i=1}^{m+1} x_i^2 = 1 \right\} \quad (15)$$

$$\hat{S}^m = \left\{ \mathbf{x} \in \mathcal{R}^{m+2} \mid \sum_{i=1}^{m+2} x_i = 1 \text{ and } \min_i x_i = 0 \right\} \quad (16)$$

Notice that both the sphere S^m and the simplex $\hat{S}^m$ are m-dimensional objects embedded in the spaces $\mathcal{R}^{m+1}$ and $\mathcal{R}^{m+2}$ respectively. The Borsuk-Ulam theorem from algebraic topology [Mas89] shows that any continuous mapping of the m-dimensional sphere to $\mathcal{R}^m$ has two antipodal[2] points mapped to the same value. The formal statement of the theorem follows.

Theorem 14 (Borsuk-Ulam [Mas89]) *For every continuous function $F : S^m \to \mathcal{R}^m$, there exists a pair of antipodal points $\mathbf{p}, \mathbf{q} \in S^m$ such that $F(\mathbf{p}) = F(\mathbf{q})$.*

As the sphere can be mapped onto the simplex through a continuous mapping that preserves antipodality, a similar theorem is true for the simplex.

Theorem 15 *For every continuous function $F : \hat{S}^m \to \mathcal{R}^m$, there exists a pair of antipodal points $\hat{\mathbf{p}}, \hat{\mathbf{q}} \in \hat{S}^m$ such that $F(\hat{\mathbf{p}}) = F(\hat{\mathbf{q}})$.*

Antipodal points on the sphere and the simplex are of special interest because of their large distance from each other. We use them to define probability distributions of large statistical difference. The following corollary of theorem 15, formalizes this idea for the case of the simplex.

Corollary 16 *For every continuous function $F : \hat{S}^m \to \mathcal{R}^m$, there exists a pair of points $\hat{\mathbf{p}}, \hat{\mathbf{q}} \in \hat{S}^m$ with $\sum_i |\hat{p}_i - \hat{q}_i| \geq 2/(m+2)$ such that $F(\hat{\mathbf{p}}) = F(\hat{\mathbf{q}})$.*

A.2 Höffding Inequality

Theorem 17 (Höffding's Inequality [Hoe63]) *Let Y_1, Y_2, ..., Y_t be independent random variables with zero means and bounded ranges: $a_i \leq Y_i \leq b_i$, $i = 1, 2, \ldots, t$. For every $\delta > 0$,*

$$\Pr\left[Y_1 + Y_2 + \ldots + Y_t \geq \delta\right] \leq \exp\left(\frac{-2\delta^2}{\sum_{i=1}^{t}(b_i - a_i)^2}\right) \quad (17)$$

[2] In general, two points $\mathbf{p}$, $\mathbf{q}$ on an object S are antipodal iff the center $\mathbf{c}$ of S lies on the line segment connecting $\mathbf{p}$ and $\mathbf{q}$. In the special case of S^m, $\mathbf{p}, \mathbf{q}$ are antipodal iff $\mathbf{p} = -\mathbf{q}$.