Sublinear Algorithms December 1, 2015

Lecture 26

Lecturer: Sofya Raskhodnikova *Scribe(s): Hui-Ju Hung*

1 Introduction

Today we will introduce a new distance metric for comparing functions/properties, called L_p-metric. We will first introduce the basic definition of the L_p-metric and then two different testing models using L_p-metric, L_p-testing and tolerant L_p-testing. Then, we discuss the relation between those testing models. In the last, we take a look at the monotonicity property using L_p-metrics and raise some open questions. Most of the following materials are taken from [BRY] and the lecture slides with slight modifications.

2 L_p-Testing

Let f be a real-valued function over a finitedomain D. For $p \geq 1$, the L_p-norm of f is $\|f\|_p = \left(\sum_{x \in D} |f(x)|^p\right)^{1/p}$. For $p = 0$, let $\|f\|_0 = \sum_{x \in D} |f(x)|^0$ be the number of non-zero values of f. Let $\mathbb{1}$ denote the function that evaluates to 1 on all $x \in D$. A *property* $\mathcal{P}$ is a set of functions over D. For real-valued functions $f : D \to [0, 1]$ and a property $\mathcal{P}$, we define relative L_p distance as follows:

$$d_p(f, \mathcal{P}) = \inf_{g \in \mathcal{P}} \frac{\|f - g\|_p}{\|\mathbb{1}\|_p} = \inf_{g \in \mathcal{P}} \left(\mathbb{E}[|f - g|^p]\right)^{1/p},$$

where the first equality holds for $p \geq 0$ and the second for $p > 0$. The normalization by a factor $\|\mathbb{1}\|_p$ ensures that $d_p(f, \mathcal{P}) \in [0, 1]$. For $p \geq 0$, a function f is ϵ-far from a property $\mathcal{P}$ w.r.t. the L_p distance if $d_p(f, \mathcal{P}) \geq \epsilon$. Otherwise, f is ϵ-close to $\mathcal{P}$. Otherwise, f is ϵ-close to $\mathcal{P}$.

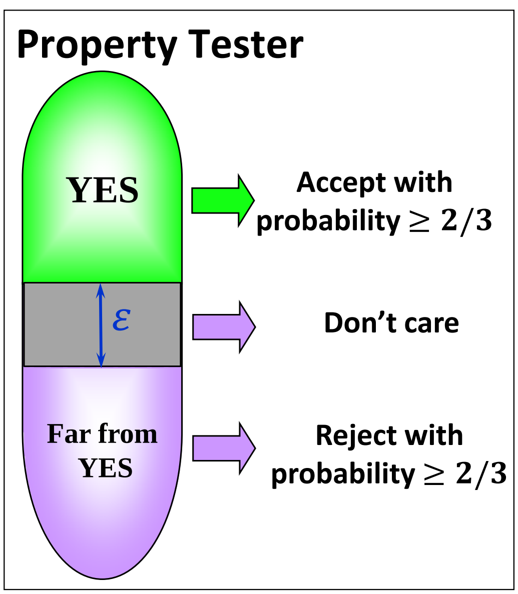

Figure 1: Property testing illustration

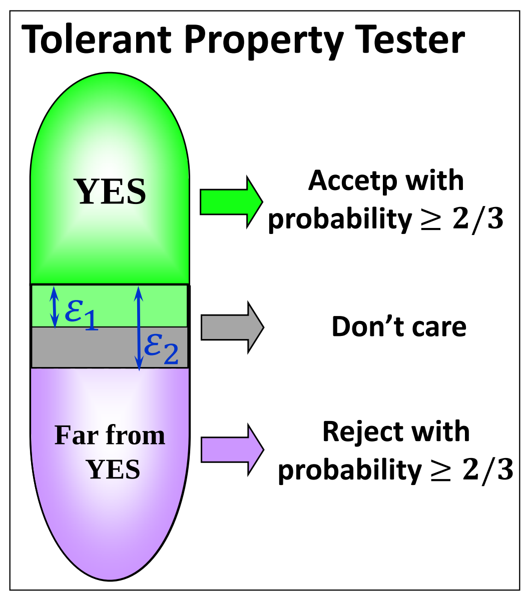

Figure 2: Tolerant property testing illustration

Definition 1 (L_p-tester). *An* L_p-tester *for a property* $\mathcal{P}$ *is a randomized algorithm that, given a proximity parameter* $\epsilon \in (0,1)$ *and oracle access to a function* $f: \mathcal{D} \to [0,1]$*,*

1. *accepts with probability at least 2/3 if* $f \in \mathcal{P}$*;*
2. *rejects with probability at least 2/3 if* $d_p(f, \mathcal{P}) \geq \epsilon$*.*

An illustration is displayed in Figure 1. The corresponding algorithmic problem is called *L_p-testing*. Standard property testing corresponds to L_0-testing, which we also call *Hamming testing*.

3 Tolerant L_p-testing

An important motivation for measuring distances to properties of real-valued functions w.r.t. L_p metrics is noise-tolerance. In order to be able to withstand noise of bounded Hamming weight (small number of outliers) in the property testing framework, [PRR06] introduced *tolerant* property testing. One justification for L_p-testing is that in applications involving real-valued data, noise added to the function often has large Hamming weight, but bounded L_p-norm for some $p > 0$ (e.g., Brownian motion, white Gaussian noise, etc.). This leads us to the following definition, which generalizes tolerant testing [PRR06].

Definition 2 (Tolerant L_p-tester). *An* (ϵ_1, ϵ_2)*-tolerant* L_p*-tester for a property* $\mathcal{P}$ *is a randomized algorithm which, given* $\epsilon_1, \epsilon_2 \in (0,1)$*, where* $\epsilon_1 < \epsilon_2$*, and oracle access to a function* $f: \mathcal{D} \to [0,1]$*,*

1. *accepts with probability at least 2/3 if* $d_p(f, \mathcal{P}) \leq \epsilon_1$*.*
2. *rejects with probability at least 2/3 if* $d_p(f, \mathcal{P}) \geq \epsilon_2$*.*

If the tester works for arbitrary $\epsilon_1 < \epsilon_2$*, it is called* fully tolerant.

An illustration is displayed in Figure 2.

4 Relationships between L_p-Testing Models

Denote $C_p(\mathcal{P}, \epsilon)$ as complexity of L_p-testing property $\mathcal{P}$ with distance parameter ϵ. Note that 1) the complexity may be time complexity or query complexity, etc., and 2) the tests may be general or restricted (e.g., nonadaptive) tests. Then, for any properties $\mathcal{P}$:

1. L_1-testing is no harder than Hamming testing: $C_1(\mathcal{P}, \epsilon) \leq C_0(\mathcal{P}, \epsilon)$.
2. L_p-testing for $p \geq 1$ is close in complexity to L_1-testing: $C_1(\mathcal{P}, \epsilon) \leq C_p(\mathcal{P}, \epsilon) \leq C_1(\mathcal{P}, \epsilon^p)$.

For properties $\mathcal{P}$ of boolean functions $f: D \to \{0,1\}$, we have:

1. L_1-testing is is equivalent to Hamming testing: $C_1(\mathcal{P}, \epsilon) = C_0(\mathcal{P}, \epsilon)$.
2. L_p-testing for $p \geq 1$ is equivalent to L_1-testing with appropriate distance parameter: $C_p(\mathcal{P}, \epsilon) = C_1(\mathcal{P}, \epsilon^p)$.

5 Monotonicity Property in L_p-testing Models

In a domain $D = [n]^d$ (i.e., vertices of a d-dimension hypercube), a function $f: D \to \mathbb{R}$ is *monotone* if increasing a coordinate of a node x does not decrease $f(x)$. The formal definition is in Definition 3. In a special case with $d = 1$, the problem become to decide whether $f(1), ..., f(n)$ is sorted or not.

Definition 3 (Monotone function). *Let D be a (finite) domain equipped with a partial order* $\preceq$*. A function* $f: D \to \mathbb{R}$ *is* monotone *if* $f(x) \leq f(y)$ *for all* $x, y \in D$ *satisfying* $x \preceq y$*. Note that* $x = (x_1, ...x_d)$ *and* $y = (y_1...y_d)$ *satisfy* $x \preceq y$ *if* $x_1 \leq y_1$*, ...,* $x_d \leq y_d$*.*

The running time complexities of monotonicity testers are listed in Table 1 .

Monotonicity		
Domain	Hamming Testing	L_p-Testing for $p \geq 1$
$[n]$	$O\left(\frac{\log n}{\epsilon}\right)$ n.a. 1-s. [EKK+00]	$O\left(\frac{1}{\epsilon^p}\right)$ n.a. 1-s. [BRY]
$[n]^d$	$O\left(\frac{d \log n}{\epsilon}\right)$ n.a. 1-s. [CS13]	$O\left(\frac{d}{\epsilon^p} \log \frac{d}{\epsilon^p}\right)$ n.a. 1-s. [BRY]

Table 1: Query complexity of L_p-testing monotonicity of a function $f : \mathcal{D} \to [0,1]$ (a./n.a. = adaptive/nonadaptive, 1-s./2-s. = 1-sided error/2-sided error).

5.1 Characterization Theorem

First, we define the threshold function in Definition 4.

Definition 4. *For a function* $f\colon D \to [0,1]$ *and* $t \in [0,1]$*, the* threshold function $f_{(t)}\colon D \to \{0,1\}$ *is:*

$$f_{(t)}(x) = \begin{cases} 1 & \text{if } f(x) \geq t; \\ 0 & \text{if } f(x) < t. \end{cases}$$

Then, we have:

Lemma 5 (Characterization of L_1 distance to monotone)**.** *For every function* $f\colon D \to [0,1]$,

$$L_1(f, \mathcal{M}) = \int_0^1 L_1(f_{(t)}, \mathcal{M})$$

, where $\mathcal{M}$ *is the class of monotone functions.*

Proof. First, we prove that $L_1(f, \mathcal{M}) \leq \int_0^1 L_1(f_{(t)}, \mathcal{M})\, dt$. For all $t \in [0,1]$, let g_t be the closest monotone (Boolean) function to $f_{(t)}$. Define $g = \int_0^1 g_t\, dt$. Since g_t is monotone for all $t \in [0,1]$, function g is also monotone. Then

$$\begin{aligned} L_1(f, \mathcal{M}) &\leq \|f - g\|_1 = \left\| \int_0^1 f_{(t)}\, dt - \int_0^1 g_t\, dt \right\|_1 \\ &= \left\| \int_0^1 (f_{(t)} - g_t)\, dt \right\|_1 \\ &\leq \int_0^1 \|f_{(t)} - g_t\|_1\, dt = \int_0^1 L_1(f_{(t)}, \mathcal{M})\, dt. \end{aligned}$$

Next, we prove that $L_1(f, \mathcal{M}) \geq \int_0^1 L_1(f_{(t)}, \mathcal{M})\, dt$. Let g denote the closest monotone function to f in L_1 distance. Then $g_{(t)}$ is monotone for all $t \in [0,1]$. We obtain:

$$
\begin{aligned}
L_1(f,\mathcal{M}) &= \|f-g\|_1 = \left\| \int_0^1 (f_{(t)} - g_{(t)})\, dt \right\|_1 \\
&= \sum_{x: f(x)\geq g(x)} \int_0^1 (f_{(t)}(x) - g_{(t)}(x))\, dt \\
&\quad + \sum_{x: f(x) < g(x)} \int_0^1 (g_{(t)}(x) - f_{(t)}(x))\, dt \\
&= \int_0^1 \Big(\sum_{x: f(x)\geq g(x)} (f_{(t)}(x) - g_{(t)}(x)) \\
&\quad + \sum_{x: f(x) < g(x)} (g_{(t)}(x) - f_{(t)}(x)) \Big)\, dt \\
&= \int_0^1 \|f_{(t)} - g_{(t)}\|_1\, dt \geq \int_0^1 L_1(f_{(t)},\mathcal{M})\, dt.
\end{aligned}
$$

The theorem follows. □

5.2 L_1-Testers for Boolean Ranges

Lemma 6. *If T is a nonadaptive 1-sided error ϵ-test for monotonicity of functions $f : D \to \{0,1\}$ then it is also a nonadaptive 1-sided error ϵ-test w.r.t. the L_1 distance for monotonicity of functions $f : D \to [0,1]$.*

Proof. Observe that a 1-sided error nonadaptive test consists of querying f on a random (not necessarily uniformly random) set of points $Q \subseteq D$ and accepting if and only if f is monotone on Q. Such a test always accepts monotone functions. It remains to prove that if f is ϵ-far from monotone w.r.t. the L_1 distance, then T will reject with probability at least 2/3.

Assume that $d_{\mathcal{M}}(f) \geq \epsilon$. Then, it follows from Lemma 5 that there exists a threshold $t^* \in [0,1]$ such that $d_{\mathcal{M}}(f_{(t^*)}) \geq \epsilon$. Recall that for Boolean functions, Hamming distance is the same as the L_1 distance. Since T is an ϵ-test for monotonicity of Boolean functions, for the random set Q selected by T, the restriction $f_{(t^*)}|_Q$ is not monotone with probability at least 2/3. It is well known (see [FLN$^+$02]) that for every function h, the restriction $h|_Q$ is not monotone iff there is a pair of points in Q *violated* by h, i.e., $x, y \in Q$ such that $x \prec y$ and $h(x) > h(y)$. That is, if $f_{(t^*)}|_Q$ is not monotone then $f_{(t^*)}(x) = 1$ and $f_{(t^*)}(y) = 0$ for some $x \prec y$, where $x, y \in Q$. But then this pair (x, y) is also violated by f, since $f(x) \geq t^* > f(y)$, and the restriction $f|_Q$ is not monotone. Thus, $f|_Q$ is not monotone with probability 2/3, and the test T satisfies the requirements in the lemma. □

6 Some Related Open Problems

1. Our L_1-tester for monotonicity is nonadaptive, but we show that adaptivity helps for Boolean range. Is there a better adaptive tester?

2. All our algorithms for L_p-testing for $p \geq 1$ were obtained directly from L_1-testers. Can one design better algorithms by working directly with L_p-distances?

3. We designed tolerant tester only for monotonicity ($d = 1, 2$). Is there tolerant testers for higher dimensions? Is there other properties that can use tolerant testers?

References

[BRY] Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev. L_p-testing. *Unpublished Draft.*

[CS13] Deeparnab Chakrabarty and C. Seshadhri. Optimal bounds for monotonicity and lipschitz testing over hypercubes and hypergrids. In *STOC*, pages 419–428, 2013.

[EKK+00] F. Ergun, S. Kannan, S. R. Kumar, R. Rubinfeld, and M. Viswanathan. Spot-checkers. *JCSS*, 60(3):717–751, 2000. Preliminary version in 30th STOC, 1998.

[FLN+02] Eldar Fischer, Eric Lehman, Ilan Newman, Sofya Raskhodnikova, Ronitt Rubinfeld, and Alex Samorodnitsky. Monotonicity testing over general poset domains. In *STOC*, pages 474–483, 2002.

[PRR06] Michal Parnas, Dana Ron, and Ronitt Rubinfeld. Tolerant property testing and distance approximation. *J. Comput. Syst. Sci.*, 72(6):1012–1042, 2006.