$L_p$-Testing

Grigory Yaroslavtsev

Joint work with Piotr Berman and Sofya Raskhodnikova
Testing Big Data

• **Q**: How to understand properties of large data looking only at a small sample?
• **Q**: How to ignore noise and outliers?
• **Q**: How to minimize assumptions about the sample generation process?
• **Q**: How to optimize running time?
Which stocks were growing steadily?

Data from http://finance.google.com
Property Testing

[Goldreich, Goldwasser, Ron; Rubinfeld, Sudan]

Randomized Algorithm

- **YES**
  - Accept with probability $\geq \frac{2}{3}$
  - Reject with probability $\geq \frac{2}{3}$

- **NO**
  - Accept with probability $\geq \frac{2}{3}$
  - Reject with probability $\geq \frac{2}{3}$

Property Tester

- **YES**
  - Accept with probability $\geq \frac{2}{3}$
  - Don’t care

- **NO**
  - Reject with probability $\geq \frac{2}{3}$

$\epsilon$-close: $\leq \epsilon$ fraction has to be changed to become **YES**
Tolerant Property Testing

[Parnas, Ron, Rubinfeld]

$\epsilon$-close: $\leq \epsilon$ fraction has to be changed to become YES

Property Tester

- $\Rightarrow$ Accept with probability $\geq \frac{2}{3}$
- $\Rightarrow$ Don’t care
- $\Rightarrow$ Reject with probability $\geq \frac{2}{3}$

Tolerant Property Tester

- $\Rightarrow$ Accept with probability $\geq \frac{2}{3}$
- $\Rightarrow$ Don’t care
- $\Rightarrow$ Reject with probability $\geq \frac{2}{3}$
Which stocks were growing steadily?

Data from http://finance.google.com
Tolerant “$L_1$ Property Testing”

- $f: \{1, ..., n\} \to [0,1]$
- $P =$ class of monotone functions
- $\text{dist}_1(f, P) = \frac{\min_{g \in P} |f - g|_1}{n}$
- $\epsilon$-close: $\text{dist}_1(f, P) \leq \epsilon$

Tolerant “$L_1$ Property Tester”

⇒ Accept with probability $\geq \frac{2}{3}$
⇒ Don’t care
⇒ Reject with probability $\geq \frac{2}{3}$
New $L_\rho$-Testing Model for Real-Valued Data

• **Generalizes** standard Hamming testing

• For $\rho > 0$ still have a **probabilistic interpretation**:
  \[ d_\rho(f, g) = \left( \mathbf{E} [|f - g|^\rho] \right)^{1/\rho} \]

• **Compatible with existing PAC-style learning models**
  (preprocessing for model selection)

• For Boolean functions, $d_0(f, g) = d_\rho(f, g)^\rho$. 
Our Contributions

1. Relationships between $L_p$-testing models
2. Algorithms
   - $L_p$-testers for $p \geq 1$
     - monotonicity, Lipschitz, convexity
   - Tolerant $L_p$-tester for $p \geq 1$
     - monotonicity in 1D (sublinear algorithm for isotonic regression)

- Our $L_p$-testers beat lower bounds for Hamming testers
- Simple algorithms backed up by involved analysis
- Uniformly sampled (or easy to sample) data suffices

3. Nearly tight lower bounds
Implications for Hamming Testing

Some techniques/results carry over to Hamming testing

- Improvement on Levin’s work investment strategy
  - Connectivity of bounded-degree graphs [Goldreich, Ron ‘02]
  - Properties of images [Raskhodnikova ‘03]
  - Multiple-input problems [Goldreich ‘13]

- First example of monotonicity testing problem where adaptivity helps
- Improvements to Hamming testers for Boolean functions
Definitions

• $f: D \to [0,1]$ ($D$ = finite domain/poset)

• $\|f\|_p = (\sum_{x \in D} |f(x)|^p)^{1/p}$, for $p \geq 1$

• $\|f\|_0 = \text{Hamming weight} ($\# \text{ of non-zero values}$)

• Property $P$ = class of functions (monotone, convex, linear, Lipschitz, ...)

• $\text{dist}_p(f, P) = \min_{g \in P} \|f - g\|_p / \|1\|_p$
Relationships: $L_p$-Testing

$Q_p(P, \epsilon) =$ query complexity of $L_p$-testing property $P$ at distance $\epsilon$

- $Q_1(P, \epsilon) \leq Q_0(P, \epsilon)$
- $Q_1(P, \epsilon) \leq Q_2(P, \epsilon)$ (Cauchy-Shwarz)
- $Q_1(P, \epsilon) \geq Q_2(P, \sqrt{\epsilon})$

Boolean functions $f : D \rightarrow \{0, 1\}$

$Q_0(P, \epsilon) = Q_1(P, \epsilon) = Q_2(P, \sqrt{\epsilon})$
Relationships: Tolerant $L_p$-Testing

$$Q_p(P, \epsilon_1, \epsilon_2) = \text{query complexity of tolerant } L_p \text{-testing property } P \text{ with distance parameters } \epsilon_1, \epsilon_2$$

- No general relationship between tolerant $L_1$-testing and tolerant Hamming testing
- $L_p$-testing for $p > 1$ is close in complexity to $L_1$-testing
  $$Q_1(P, \epsilon_1^p, \epsilon_2) \leq Q_p(P, \epsilon_1, \epsilon_2) \leq Q_1(P, \epsilon_1, \epsilon_2^p)$$

For Boolean functions $f: D \rightarrow \{0,1\}$

$$Q_0(P, \epsilon_1, \epsilon_2) = Q_1(P, \epsilon_1, \epsilon_2) = Q_p(P, \epsilon_1^{1/p}, \epsilon_2^{1/p})$$
Our Results: Testing Monotonicity

- Hypergrid ($D = [n]^d$)

<table>
<thead>
<tr>
<th></th>
<th>$L_0$</th>
<th>$L_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Upper bound</strong></td>
<td>$O\left(\frac{d \log n}{\epsilon}\right)$</td>
<td>$O\left(\frac{d}{\epsilon} \log \frac{d}{\epsilon}\right)$</td>
</tr>
<tr>
<td></td>
<td>[Dodis et al.'99,..., Chakrabarti, Seshadhri '13]</td>
<td></td>
</tr>
<tr>
<td><strong>Lower bound</strong></td>
<td>$\Omega\left(\frac{d \log n}{\epsilon}\right)$</td>
<td>$\Omega\left(\frac{1}{\epsilon} \log \frac{1}{\epsilon}\right)$</td>
</tr>
<tr>
<td></td>
<td>[Dodis et al.'99,..., Chakrabarti, Seshadhri '13]</td>
<td>Non-adaptive 1-sided error</td>
</tr>
</tbody>
</table>

- $2^{O(d)}/\epsilon$ adaptive tester for Boolean functions
Monotonicity: Key Lemma

- M = class of monotone functions
- Boolean slicing operator \( f_y : D \rightarrow \{0,1\} \)
  \[
  f_y(x) = 1, \text{ if } f(x) \geq y, \\
  f_y(x) = 0, \text{ otherwise.}
  \]
- Theorem:

\[
\text{dist}_1(f, M) = \int_0^1 \text{dist}_0(f_y, M)dy
\]
Proof sketch: slice and conquer

1) Closest monotone function with **minimal** $L_1$-**norm** is **unique** (can be denoted as an operator $M^1_f$).

2) $\|f - g\|_1 = \int_0^1 \|f_y - g_y\| dy$

3) $M^1_f$ and $f_y$ commute: $(M^1_f)_y = M^1(f_y)$

$$
\text{dist}_1(f, M) = \frac{\|f - M^1_f\|_1}{|D|} = \frac{\int_0^1 \|f_y - (M^1_f)_y\|_1 dy}{|D|}
$$
**Thm:** A nonadaptive, 1-sided error $L_0$-test for monotonicity of $f : D \rightarrow \{0,1\}$ is also an $L_1$-test for monotonicity of $f : D \rightarrow [0,1]$.

**Proof:**

- A violation $(x, y)$:

- A nonadaptive, 1-sided error test queries a random set $Q \subseteq D$ and rejects iff $Q$ contains a violation.
- If $f : D \rightarrow [0,1]$ is monotone, $Q$ will not contain a violation.
- If $d_1(f, M) \geq \varepsilon$ then $\exists t^*: d_0(f(t^*), M) \geq \varepsilon$
- W.p. $\geq 2/3$, set $Q$ contains a violation $(x, y)$ for $f(t^*)$

\[
\begin{align*}
    f(t^*)(x) &= 1, & f(t^*)(y) &= 0 \\
    f(x) &> f(y)
\end{align*}
\]
Distance Approximation and Tolerant Testing

Approximating $L_1$-distance to monotonicity $\pm \delta$ w. $p. \geq \frac{2}{3}$

<table>
<thead>
<tr>
<th>$f$</th>
<th>$L_0$</th>
<th>$L_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[n] \to [0,1]$</td>
<td>$\text{polylog } n \cdot \left(\frac{1}{\delta}\right)^{o(1/\delta)}$</td>
<td>$\Theta\left(\frac{1}{\delta^2}\right)$</td>
</tr>
</tbody>
</table>

[Saks Seshadhri 10]

- Time complexity of tolerant $L_1$-testing for monotonicity is $O\left(\frac{\varepsilon_2}{(\varepsilon_2 - \varepsilon_1)^2}\right)$
  - Better dependence than what follows from distance approximation for $\varepsilon_2 \ll 1$
  - Improves $\tilde{O}\left(\frac{1}{\delta^2}\right)$ adaptive distance approximation of [Fattal,Ron’10] for Boolean functions
\section*{$L_1$-Testers for Other Properties}

Via combinatorial characterization of $L_1$-distance to the property

- Lipschitz property $f: [n]^d \rightarrow [0,1]$:

  $$\Theta\left(\frac{d}{\epsilon}\right)$$

  Via (implicit) \textbf{proper learning}: approximate in $L_1$ up to error $\epsilon$, test approximation on a random $O(1/\epsilon)$-sample

- Convexity $f: [n]^d \rightarrow [0,1]$:

  $$0\left(\epsilon^{-\frac{d}{2}} + \frac{1}{\epsilon}\right) \text{ (tight for } d \leq 2)$$

- Submodularity $f: \{0,1\}^d \rightarrow [0,1]$:

  $$2\tilde{O}\left(\frac{1}{\epsilon}\right) + \text{poly}\left(\frac{1}{\epsilon}\right)\log d \text{ [Feldman, Vondrak 13]}$$
Open Problems

• All our algorithms for \( p > 1 \) were obtained directly from \( L_1 \)-testers.

Can one design better algorithms by working directly with \( L_p \)-distances?

• Our complexity for \( L_p \)-testing convexity grows exponentially with \( d \).

Is there an \( L_p \)-testing algorithm for convexity with subexponential dependence on the dimension?

• Our \( L_1 \)-tester for monotonicity is nonadaptive, but we show that adaptivity helps for Boolean range.

Is there a better adaptive tester?

• We designed tolerant tester only for monotonicity (\( d=1,2 \)).

Tolerant testers for higher dimensions?

Other properties?