CSCI B609: “Foundations of Data Science”

Lecture 23: L_p-testing and isotonic regression

Slides at http://grigory.us/data-science-class.html

Grigory Yaroslavtsev

http://grigory.us

Testing Big Data

- **Q**: How to make sense of big data?
- **Q**: How to understand properties 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?

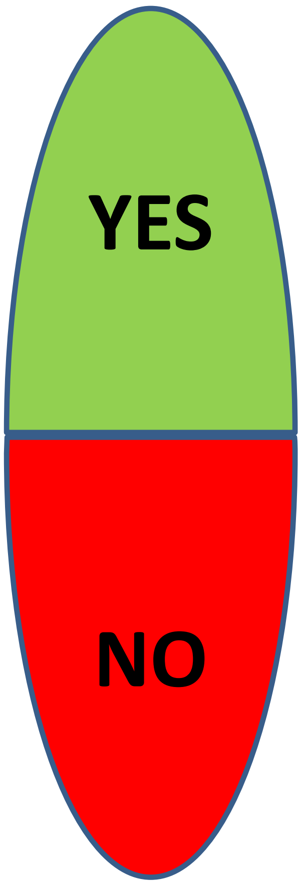
Which stocks were growing steadily?

Data from http://finance.google.com

Property Testing

[Goldreich, Goldwasser, Ron; Rubinfeld, Sudan]

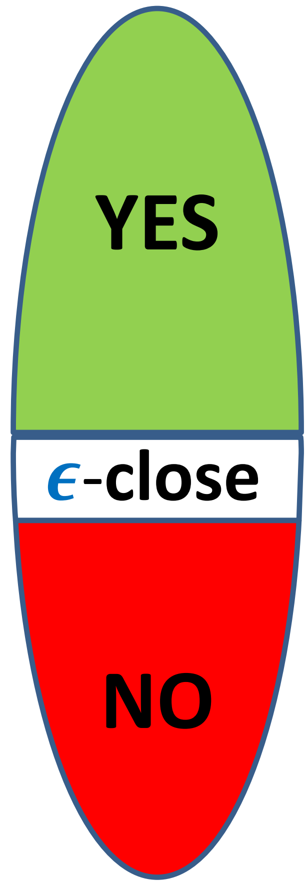
Randomized Algorithm



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

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

Property Tester



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

ϵ-close $\Rightarrow$ **Don't care**

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

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

Which stocks were growing steadily?

Microsoft

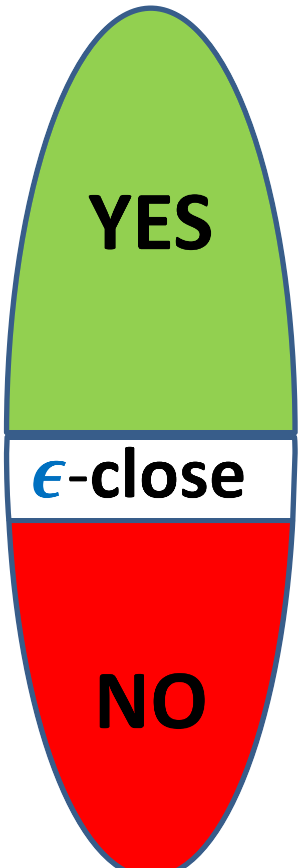
IBM

Data from http://finance.google.com

Tolerant Property Testing

[Parnas, Ron, Rubinfeld]

Property Tester

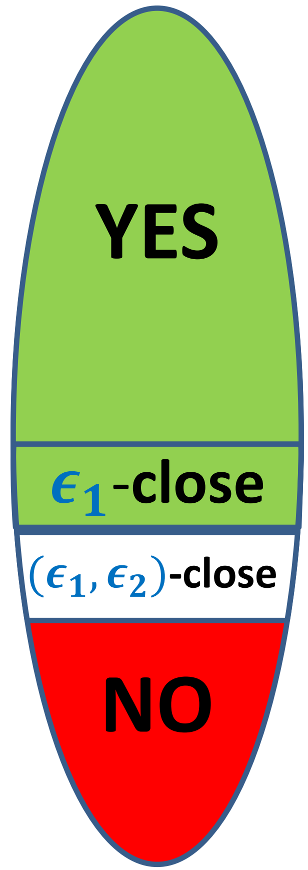


$\Rightarrow$ **Accept with probability $\geq \frac{2}{3}$**

$\Rightarrow$ **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 $\geq \frac{2}{3}$**

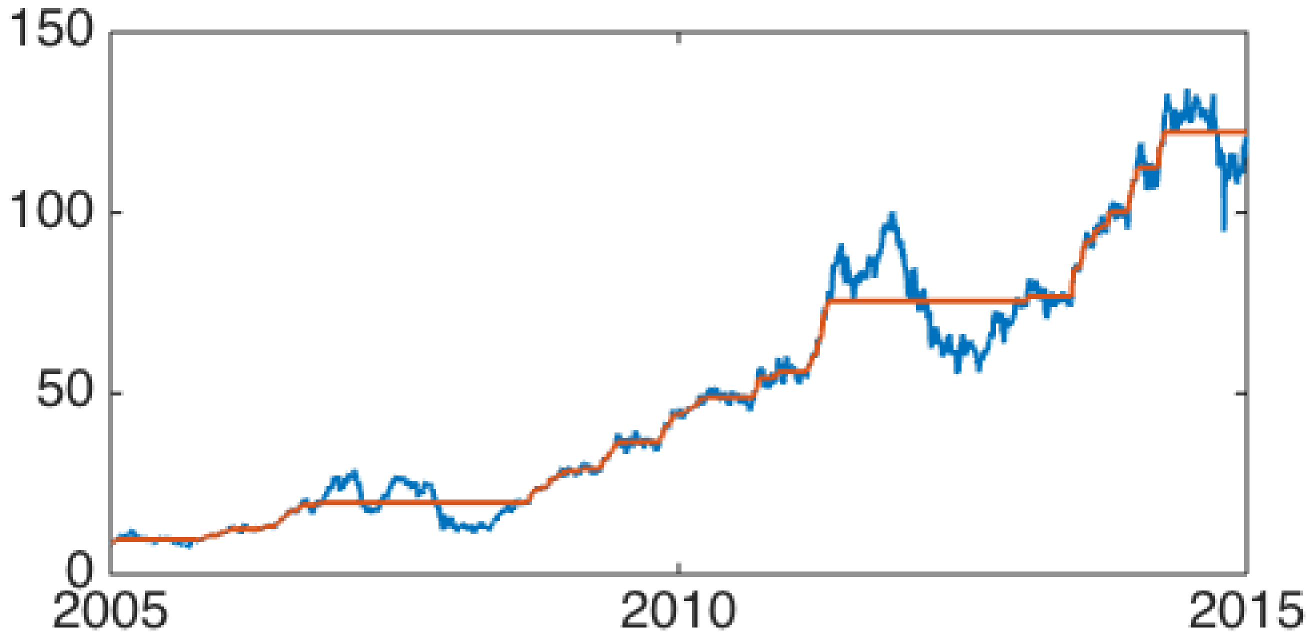
ϵ-**close :** $\leq \epsilon$ fraction has to be changed to become **YES**

Which stocks were growing steadily?

Data from http://finance.google.com

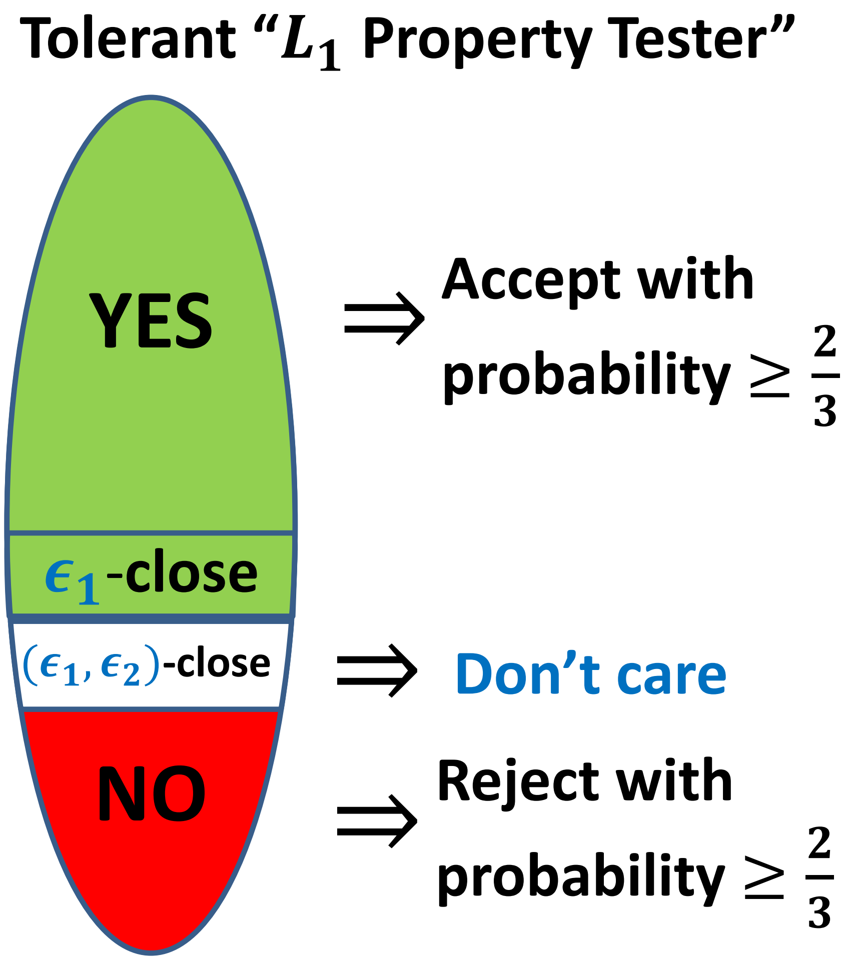
L_1-Isotonic Regression

- Running time $O(n \log n)$ [Ahuja, Orlin]



Tolerant "L_1 Property Testing"

- $\boldsymbol{f}: \{1, \ldots, n\} \to [0,1]$
- $\boldsymbol{P}$ = class of monotone functions
- $dist_1(\boldsymbol{f}, \boldsymbol{P}) = \dfrac{\min_{\boldsymbol{g} \in \boldsymbol{P}} |\boldsymbol{f} - \boldsymbol{g}|_1}{n}$
- $\boldsymbol{\epsilon}$-close: $dist_1(\boldsymbol{f}, \boldsymbol{P}) \le \boldsymbol{\epsilon}$



New L_p-Testing Model for Real-Valued Data

- **Generalizes** standard Hamming testing

- For $p > 0$ still has a **probabilistic interpretation**:
$$d_p(f,g) = (\mathbf{E}[|\boldsymbol{f} - \boldsymbol{g}|^{\boldsymbol{p}}])^{1/p}$$

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

- For Boolean functions, $d_0(f,g) = d_p(f,g)^p$.

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

- $\boldsymbol{f}: D \rightarrow [0,1]$ (D = finite domain/poset)
- $||\boldsymbol{f}||_{\boldsymbol{p}} = (\sum_{x \in D} |\boldsymbol{f}(x)|^{\boldsymbol{p}})^{1/\boldsymbol{p}}$, for $\boldsymbol{p} \geq 1$
- $||\boldsymbol{f}||_{\boldsymbol{0}}$ = Hamming weight (# of non-zero values)
- Property $\boldsymbol{P}$ = class of functions (monotone, convex, linear, Lipschitz, …)
- $dist_{\boldsymbol{p}}(\boldsymbol{f}, \boldsymbol{P}) = \dfrac{\min_{g \in \boldsymbol{P}} ||\boldsymbol{f} - g||_{\boldsymbol{p}}}{||1||_{\boldsymbol{p}}}$

Relationships: L_p-Testing

$Q_{\boldsymbol{p}}(\boldsymbol{P},\boldsymbol{\epsilon})$ = query complexity of $L_{\boldsymbol{p}}$-testing property $\boldsymbol{P}$ at distance $\boldsymbol{\epsilon}$

- $Q_{\mathbf{1}}(\boldsymbol{P},\boldsymbol{\epsilon}) \leq Q_{\mathbf{0}}(\boldsymbol{P},\boldsymbol{\epsilon})$
- $Q_{\mathbf{1}}(\boldsymbol{P},\boldsymbol{\epsilon}) \leq Q_{\mathbf{2}}(\boldsymbol{P},\boldsymbol{\epsilon})$ (Cauchy-Shwarz)
- $Q_{\mathbf{1}}(\boldsymbol{P},\boldsymbol{\epsilon}) \geq Q_{\mathbf{2}}(\boldsymbol{P},\sqrt{\boldsymbol{\epsilon}})$

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

$Q_{\mathbf{0}}(\boldsymbol{P},\boldsymbol{\epsilon}) = Q_{\mathbf{1}}(\boldsymbol{P},\boldsymbol{\epsilon}) = Q_{\mathbf{2}}(\boldsymbol{P},\sqrt{\boldsymbol{\epsilon}})$

Relationships: Tolerant L_p-Testing

$Q_p(P, \epsilon_1, \epsilon_2)$ = query complexity of tolerant L_p-testing property P with distance parameters ϵ_1, ϵ_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

$$Q_1(P, \varepsilon_1^p, \varepsilon_2) \leq Q_p(P, \varepsilon_1, \varepsilon_2) \leq Q_1(P, \varepsilon_1, \varepsilon_2^p)$$

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

$Q_0(P, \varepsilon_1, \varepsilon_2) = Q_1(P, \varepsilon_1, \varepsilon_2) = Q_p(P, \epsilon_1^{1/p}, \varepsilon_2^{1/p})$

Testing Monotonicity

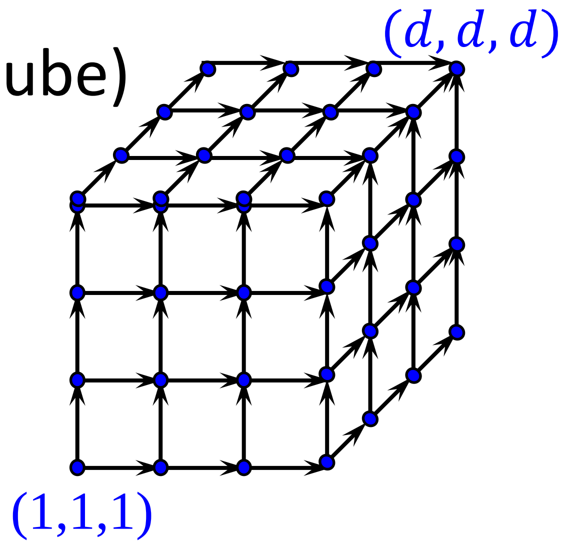
- Line ($D = [\boldsymbol{n}]$)

	L_0	L_1
Upper bound	$O\left(\log \boldsymbol{n}/\epsilon\right)$ [Ergun, Kannan, Kumar, Rubinfeld, Viswanathan'00]	$O(1/\epsilon)$
Lower bound	$\Omega(\log \boldsymbol{n}/\epsilon)$ [Fischer'04]	$\Omega(1/\epsilon)$

Monotonicity

- Domain D=$[n]^d$ (vertices of d-dim hypercube)
- A function $f: D \to \mathbb{R}$ is monotone

if increasing a coordinate of x does not decrease $f(x)$.



- Special case $d = 1$

$f: [n] \to \mathbb{R}$ is monotone $\Leftrightarrow f(1), \ldots f(n)$ is sorted.

One of the most studied properties in property testing [Ergün Kannan Kumar Rubinfeld Viswanathan , Goldreich Goldwasser Lehman Ron, Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky, Batu Rubinfeld White, Fischer Lehman Newman Raskhodnikova Rubinfeld Samorodnitsky, Fischer, Halevy Kushilevitz, Bhattacharyya Grigorescu Jung Raskhodnikova Woodruff, ..., Chakrabarty Seshadhri, Blais, Raskhodnikova Yaroslavtsev, Chakrabarty Dixit Jha Seshadhri]

Monotonicity: Key Lemma

- M = class of monotone functions
- Boolean slicing operator $\boldsymbol{f}_{\boldsymbol{y}}: D \rightarrow \{0,1\}$

$$\boldsymbol{f}_{\boldsymbol{y}}(x) = 1, \text{ if } \boldsymbol{f}(x) \geq \boldsymbol{y},$$

$$\boldsymbol{f}_{\boldsymbol{y}}(x) = 0, \text{ otherwise.}$$

- **Theorem:**

$$dist_1(\boldsymbol{f}, M) = \int_0^1 dist_0(\boldsymbol{f}_{\boldsymbol{y}}, M)\, d\boldsymbol{y}$$

Proof sketch: slice and conquer

1) Closest monotone function with **minimal $\boldsymbol{L_1}$-norm** is **unique** (can be denoted as an operator $M_{\boldsymbol{f}}^1$).
2) $\|f - g\|_1 = \int_0^1 \left\| f_{\boldsymbol{y}} - g_{\boldsymbol{y}} \right\| d\boldsymbol{y}$
3) $M_{\boldsymbol{f}}^1$ and $\boldsymbol{f}_{\boldsymbol{y}}$ commute: $\left(M_{\boldsymbol{f}}^1\right)_{\boldsymbol{y}} = M^1{}_{(\boldsymbol{f}_{\boldsymbol{y}})}$

$$dist_1(f, M) \overset{1)}{=} \frac{\left\| f - M_f^1 \right\|_1}{|D|} \overset{2)}{=} \frac{\int_0^1 \left\| f_{\boldsymbol{y}} - (M_f^1)_{\boldsymbol{y}} \right\|_1 d\boldsymbol{y}}{|D|} \overset{3)}{=}$$

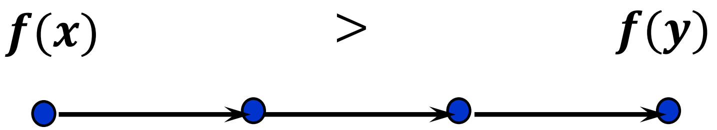
$$= \frac{\int_0^1 \left\| f_{\boldsymbol{y}} - M^1_{(f_{\boldsymbol{y}})} \right\|_1 d\boldsymbol{y}}{|D|} = \int_0^1 dist_0\left(f_{\boldsymbol{y}}, M\right) d\boldsymbol{y}$$

L_1-Testers from Boolean Testers

Thm: A nonadaptive, 1-sided error L_0-test for monotonicity of $f: D \to \{0,1\}$ is also an L_1-test for monotonicity of $f: D \to [0,1]$.

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 $f: D \to [0,1]$ is monotone, Q will not contain a violation.
- If $d_1(f, M) \geq \varepsilon$ then $\exists \boldsymbol{t}^*: d_0\left(\boldsymbol{f}_{(\boldsymbol{t}^*)}, M\right) \geq \boldsymbol{\varepsilon}$
- W.p. $\geq 2/3$, set Q contains a violation (x, y) for $\boldsymbol{f}_{(\boldsymbol{t}^*)}$

$$\boldsymbol{f}_{(\boldsymbol{t}^*)}(x) = 1, \boldsymbol{f}_{(\boldsymbol{t}^*)}(y) = 0$$

$$\Downarrow$$

$$\boldsymbol{f}(x) > \boldsymbol{f}(y)$$

Our Results: Testing Monotonicity

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

	L_0	L_1
Upper bound	$O\left(\frac{d \log n}{\epsilon}\right)$ [Dodis et al. '99,…, Chakrabarti, Seshadhri '13]	$O\left(\frac{d}{\epsilon} \log \frac{d}{\epsilon}\right)$
Lower bound	$\Omega\left(\frac{d \log n}{\epsilon}\right)$ [Dodis et al.'99…, Chakrabarti, Seshadhri '13]	$\Omega\left(\frac{1}{\epsilon} \log \frac{1}{\epsilon}\right)$ Non-adaptive 1-sided error

- $2^{O(d)}/\epsilon$ **adaptive** tester for Boolean functions

Testing Monotonicity of $[\boldsymbol{n}]^{\boldsymbol{d}} \to \{0,1\}$

- $e^{\boldsymbol{i}} = (0 \dots 1 \dots 0)$ = $\boldsymbol{i}$-th unit vector.
- For $\boldsymbol{i} \in [\boldsymbol{d}], \alpha \in [\boldsymbol{n}]^{\boldsymbol{d}}$ where $\alpha_{\boldsymbol{i}} = 0$ an axis-parallel line along dimension $\boldsymbol{i}$: $\{\alpha + x_{\boldsymbol{i}} e^{\boldsymbol{i}} | x_{\boldsymbol{i}} \in [\boldsymbol{n}]\}$
- $L_{\boldsymbol{n},\boldsymbol{d}}$= set of all $\boldsymbol{d}\boldsymbol{n}^{\boldsymbol{d}-1}$ axis-parallel lines
- Dimension reduction for $\boldsymbol{f}$: $[\boldsymbol{n}]^{\boldsymbol{d}} \to \{0,1\}$[Dodis et al.'99]:

$$E_{\ell \sim L_{\boldsymbol{n},\boldsymbol{d}}} \left[dist \left(\boldsymbol{f} \Big|_{\ell}, M \right) \right] \geq \frac{dist(\boldsymbol{f}, M)}{2\boldsymbol{d}}$$

- If $dist(\boldsymbol{f}|_{\ell}, M) \geq \boldsymbol{\delta}$ => $O\left(\frac{1}{\boldsymbol{\delta}}\right)$-sample detects a violation

Testing Monotonicity on $[\boldsymbol{n}]^{\boldsymbol{d}}$

- Dimension reduction for $\boldsymbol{f}: [\boldsymbol{n}]^{\boldsymbol{d}} \to \{0,1\}$[Dodis et al.'99]:

$$E_{\ell \sim L_{\boldsymbol{n},\boldsymbol{d}}} \left[dist\left(\boldsymbol{f}\Big|_{\ell}, M \right) \right] \geq \frac{dist(\boldsymbol{f}, M)}{2\boldsymbol{d}}$$

- If $dist(\boldsymbol{f}|_{\ell}, M) \geq \boldsymbol{\delta}$ => $O\left(\frac{1}{\boldsymbol{\delta}}\right)$-sample can detect a violation
- "Inverse Markov": For $\mathrm{r.v.}\, \boldsymbol{X} \in [0,1]$ with $\mathrm{E}[\mathbf{X}] = \boldsymbol{\mu}$ and $c < 1$

$$\Pr[\boldsymbol{X} \leq c\boldsymbol{\mu}] \leq \frac{1 - \boldsymbol{\mu}}{1 - c\,\boldsymbol{\mu}} \Rightarrow \Pr\left[\boldsymbol{X} \leq \frac{\boldsymbol{\mu}}{2}\right] \leq 1 - \frac{\boldsymbol{\mu}}{2 - \boldsymbol{\mu}} \leq 1 - \frac{\boldsymbol{\mu}}{2}$$

- $\Pr\left[dist(\boldsymbol{f}|_{\ell}, M) \geq \frac{dist(\boldsymbol{f}, M)}{4\boldsymbol{d}} \right] \geq \frac{dist(\boldsymbol{f}, M)}{4\boldsymbol{d}} \Rightarrow O\left(\frac{\boldsymbol{d}^2}{\boldsymbol{\epsilon}^2}\right)$-test
- [Dodis et al.] $O\left(\frac{\boldsymbol{d}}{\boldsymbol{\epsilon}} \log^2 \frac{\boldsymbol{d}}{\boldsymbol{\epsilon}}\right)$ via "Levin's economical work investment strategy" (used in other papers for testing connectedness of a graph, properties of images, etc.)

Testing Monotonicity on $[\boldsymbol{n}]^{\boldsymbol{d}}$

- "Discretized Inverse Markov"

For $\mathrm{r.v.}\,\boldsymbol{X} \in [0,1]$ with $\mathrm{E}[\mathbf{X}] = \boldsymbol{\mu} \leq \frac{1}{2}$ and $\boldsymbol{t} = 3\log 1/\boldsymbol{\mu}$

$$\exists\, \boldsymbol{j} \in [\boldsymbol{t}]:\ \Pr\left[\boldsymbol{X} \geq 2^{-j}\right] \geq \frac{2^{j}\boldsymbol{\mu}}{4}$$

- For each $\boldsymbol{i} \in [\boldsymbol{t}]$ pick $O\left(\frac{1}{\mu\, 2^{i}}\right)$ samples of size $O(2^{i})$ => complexity $O\left(\frac{1}{\mu}\log\frac{1}{\mu}\right)$
- For the good bucket $\mathbf{j}$ the test rejects with constant probability
- $\boldsymbol{\mu} = E_{\ell\sim L_{\boldsymbol{n},\boldsymbol{d}}}[dist(\boldsymbol{f}|_{\ell}, M)] \geq \frac{dist(\boldsymbol{f},M)}{2\boldsymbol{d}}$ => $O\left(\frac{\boldsymbol{d}}{\epsilon}\log\frac{\boldsymbol{d}}{\epsilon}\right)$-test

Distance Approximation and Tolerant Testing

Approximating L_1-distance to monotonicity $\pm\delta\ w.p. \geq 2/3$		
f	L_0	L_1
$[n] \to [0,1]$	$\text{polylog}\, n \cdot \left(\frac{1}{\delta}\right)^{O(1/\delta)}$ [Saks Seshadhri 10]	$\Theta\left(\frac{1}{\delta^2}\right)$

- Sublinear algorithm for isotonic regression
- 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 appoximation for $\epsilon_2 \ll 1$
  - Improves $\tilde{O}\left(\frac{1}{\delta^2}\right)$ adaptive distance approximation of [Fattal,Ron'10] for Boolean functions

Distance Approximation $f: [n] \rightarrow [0,1]$

Theorem: with constant probability over the choice of a random sample **S** of size $O\left(\frac{1}{\delta^2}\right)$:

$$|dist_1(f|_S, M) - dist_1(f, M)| < \delta$$

- Implies an $O\left(\frac{1}{(\epsilon_2 - \epsilon_1)^2}\right)$ tolerant tester by setting $\delta = \frac{(\epsilon_2 - \epsilon_1)}{3}$
- $dist_1(\boldsymbol{f}, M) = \int_0^1 dist_0(\boldsymbol{f}_{\boldsymbol{y}}, M) d\boldsymbol{y}$
- Suffices: $\forall \boldsymbol{y}$: $\left|dist_0\left(\boldsymbol{f}_{\boldsymbol{y}}|_S, M\right) - dist_0\left(\boldsymbol{f}_{\boldsymbol{y}}, M\right)\right| < \delta$
- Improves previous $\tilde{O}(1/\delta^2)$ algorithm [Fattal, Ron'10]

Distance Approximation

For $\boldsymbol{f}: [n] \rightarrow \{0,1\}$ violation graph $\boldsymbol{G_f}([n], E)$:

$$\text{edge } (x_1, x_2) \text{ if } x_1 \leq x_2, f(x_1) = 1, f(x_2) = 0$$

$\mathbf{MM}(\mathrm{G})$ = maximum matching

$\mathbf{VC}(\mathrm{G})$ = minimum vertex cover

- $dist_0(\boldsymbol{f}, M) = \frac{|\boldsymbol{MM}(G_f)|}{|D|} = \frac{|\boldsymbol{VC}(G_f)|}{|D|}$ [Fischer et al.'02]
- $dist_0(\boldsymbol{f}|_S, M) = \frac{\left|\boldsymbol{MM}\left(G_{f|_S}\right)\right|}{|S|} = \frac{\left|\boldsymbol{VC}\left(G_{f|_S}\right)\right|}{|S|}$

$$dist_0(\boldsymbol{f}|_{\boldsymbol{S}}, M) - dist_0(\boldsymbol{f}, M) < O\left(\frac{1}{\sqrt{|\boldsymbol{S}|}}\right)$$

Define: $\boldsymbol{Y}(\boldsymbol{S}) = \frac{|\boldsymbol{VC}_{f} \cap \boldsymbol{S}|}{|\boldsymbol{S}|}$

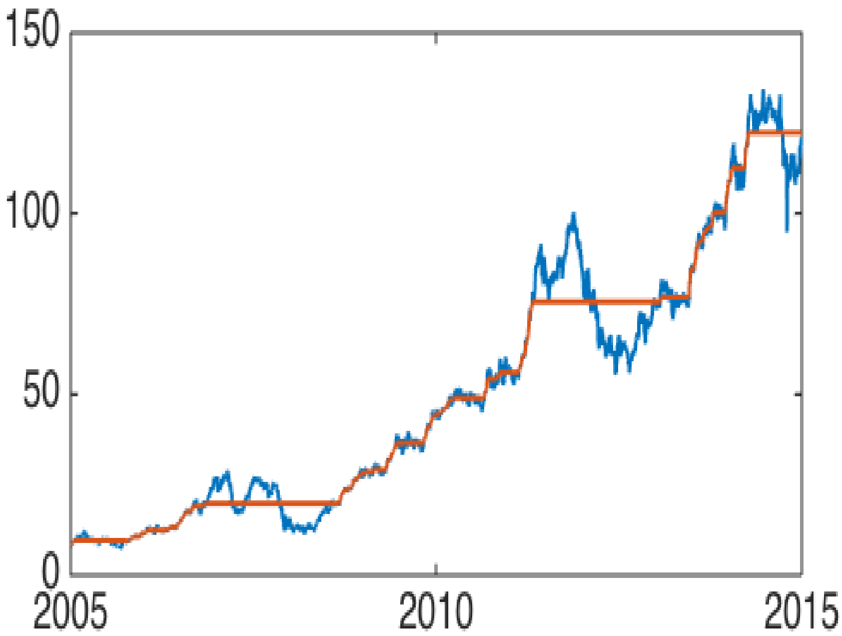
- $dist_0(\boldsymbol{f}|_{\boldsymbol{S}}, M) = \frac{|\boldsymbol{VC}_{f|_{\boldsymbol{S}}}|}{|\boldsymbol{S}|} \leq \frac{|\boldsymbol{VC}_{f} \cap \boldsymbol{S}|}{|\boldsymbol{S}|} = \boldsymbol{Y}(\boldsymbol{S})$

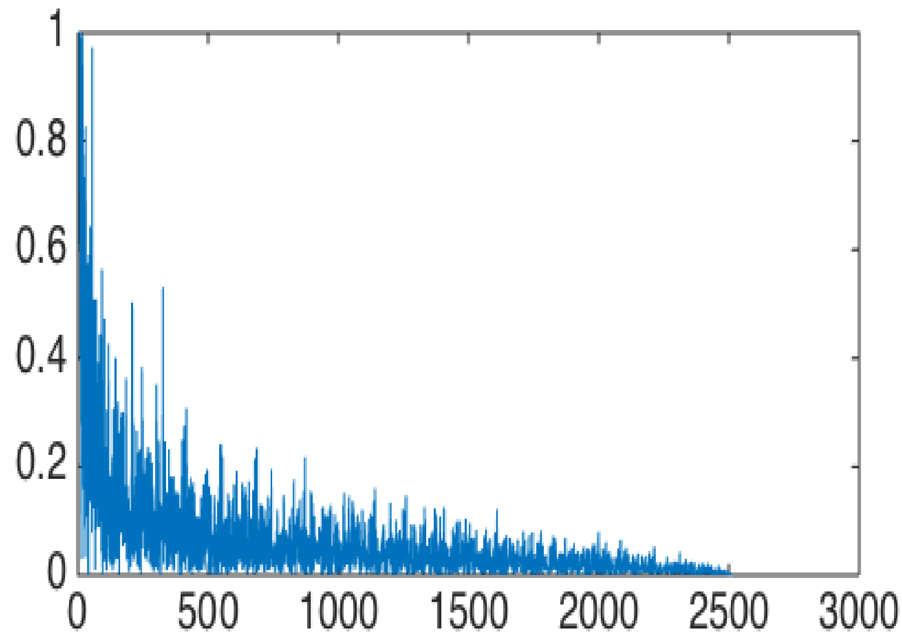
$\boldsymbol{Y}(\boldsymbol{S})$ *has hypergeometric distribution:*

- $E[\boldsymbol{Y}(\boldsymbol{S})] = \frac{|\boldsymbol{VC}_{f}|}{|D|} = dist_0(\boldsymbol{f}, M)$
- $Var[Y(\boldsymbol{S})] \leq \frac{|\boldsymbol{S}||VC_f|}{|D||\boldsymbol{S}|^2} = \frac{dist_0(f,M)}{|\boldsymbol{S}|} \leq \frac{1}{|\boldsymbol{S}|}$

Experiments

- Data: Apple stock price data (2005-2015) from Google Finance
- Left: L_1-isotonic regression
- Right: error vs. sample size





L_1-Testers for Other Properties

Via combinatorial characterization of L_1-distance to the property

- Lipschitz property $\boldsymbol{f}: [\boldsymbol{n}]^{\boldsymbol{d}} \to [0,1]$:

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

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

- Convexity $\boldsymbol{f}: [\boldsymbol{n}]^{\boldsymbol{d}} \to [0,1]$**:**

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

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

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

Open Problems

- All our algorithms for 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?