Sublinear Algorithms

LECTURE 4

Last time

- Testing if a graph is connected.
- Estimating the number of connected components.
- Estimating the weight of a MST

Today

- Limitations of sublinear-time algorithms
- Yao's Minimax Principle

HW2 is out tonight, due next Thursday at 11am

Sofya Raskhodnikova;Boston University

Query Complexity

- Query complexity of an algorithm is the maximum number of queries the algorithm makes.
  - Usually expressed as a function of input length (and other parameters)
  - Example: the test for sortedness (from Lecture 2) had query complexity $O(\log n)$ for constant ε, more precisely $O\left(\frac{\log n}{\varepsilon}\right)$
  - running time $\geq$ query complexity
- Query complexity of a problem P, denoted $q(P)$, is the query complexity of the best algorithm for the problem.
  - What is $q(\text{testing sortedness})$? How do we know that there is no better algorithm?

Today: Techniques for proving lower bounds on $q(P)$.

Yao's Principle

A Method for Proving Lower Bounds

Yao's Minimax Principle

Consider a computational problem on a finite domain.

- The following statements are equivalent.

Statement 1

For every **probabilistic** algorithm $\mathcal{A}$ of complexity q there exists an input x s.t.

$$\Pr_{coin\ tosses\ of\ \mathcal{A}}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

Statement 2

There is a distribution D on the inputs,

s.t. for every **deterministic** algorithm $\mathcal{A}$ of complexity q,

$$\Pr_{x \leftarrow D}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

- The direction needed for lower bounds:

Yao's Minimax Principle (easy direction): Statement 2 $\Rightarrow$ Statement 1.

Proof of Easy Direction of Yao's Principle

- Consider a finite set of inputs X (e.g., all inputs of length n).
- Consider a randomized algorithm that takes an input $x \in X$, makes $\leq q$ queries to x and outputs accept or reject.
- Every randomized algorithm can be viewed as a distribution μ on deterministic algorithms (which are decision trees).

- Let Y be the set of all q-query deterministic algorithms that run on inputs in X.

Proof of Easy Direction of Yao's Principle

- Consider a matrix M with
  - rows indexed by inputs x from X,
  - columns indexed by algorithms y from Y,
  - entry $M(x, y) = \begin{cases} 1 & \text{if algorithm } y \text{ is correct on input } x \\ 0 & \text{if algorithm } y \text{ is wrong on input } x \end{cases}$

	y_1	y_2	…	
x_1	**1**	**0**		
x_2	**1**	**1**		
…			⋱	

- Then an algorithm $\mathcal{A}$ is a distribution μ over columns Y with probabilities satisfying $\sum_{y \in Y} \mu(y) = 1$.

Rephrasing Statements 1 and 2 in Terms of M

Statement 1

For every **probabilistic** algorithm $\mathcal{A}$ of complexity q there exists an input x s.t.

$$\Pr_{coin\ tosses\ of\ \mathcal{A}}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

- For all distributions μ over columns Y, there exists a row x s.t.

$$\Pr_{y \leftarrow \mu}[M(x, y) = 0] > 1/3.$$

Statement 2

There is a distribution D on the inputs,

s.t. for every **deterministic** algorithm $\mathcal{A}$ of complexity q,

$$\Pr_{x \leftarrow D}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

- There is a distribution D over rows X, s.t. for all columns y,

$$\Pr_{x \leftarrow D}[M(x, y) = 0] > 1/3.$$

Statement 2 ⇒ Statement 1

- Suppose there is a distribution D over X, s.t. for all columns y,

$$\Pr_{x \leftarrow D}[M(x, y) = 0] > 1/3.$$

- Then for all distributions μ over Y,

$$\Pr_{\substack{x \leftarrow D \\ y \leftarrow \mu}}[M(x, y) = 0] > 1/3.$$

- Then for all distributions μ over Y, there exists a row x,

$$\Pr_{y \leftarrow \mu}[M(x, y) = 0] > 1/3.$$

	y_1	y_2	…	
x_1	1	0		
x_2	1	1		
…			⋱	

Yao's Principle (Easy Direction)

Statement 1

For every **probabilistic** algorithm $\mathcal{A}$ of complexity q there exists an input x s.t.

$$\Pr_{coin\ tosses\ of\ \mathcal{A}}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

Statement 2

There is a distribution D on the inputs,

s.t. for every **deterministic** algorithm of complexity q,

$$\Pr_{x \leftarrow D}[\mathcal{A}(x) \text{ is wrong}] > 1/3.$$

- The direction needed for lower bounds:

Yao's Minimax Principle (easy direction): Statement 2 $\Rightarrow$ Statement 1.

NOTE: Also applies to restricted algorithms

- 1-sided error tests
- nonadaptive tests

Yao's Minimax Principle as a game

Players: Evil algorithms designer Al and poor lower bound prover Lola.

Game1

Move 1. Al selects a q-query **randomized** algorithm $\mathcal{A}$ for the problem.

Move 2. Lola selects an input on which $\mathcal{A}$ errs with largest probability.

Game2

Move 1. Lola selects a distribution on inputs.

Move 2. Al selects a q-query **deterministic** algorithm with as large probability of success on Lola's distribution as possible.

*Toy Example: a Lower Bound for Testing 0**

Input: string of n bits

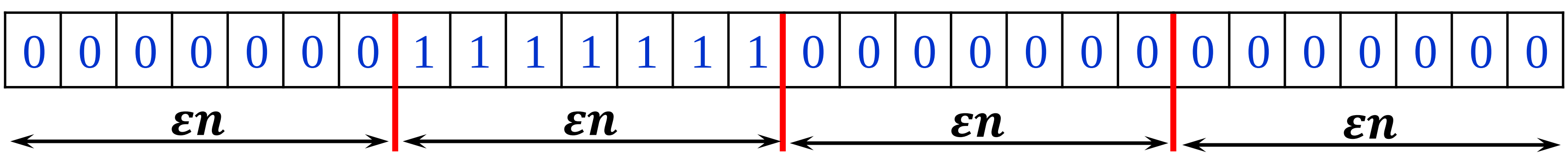
Question: Does the string contain only 0's or is it ε-far form the all-0 string?

Claim. Any algorithm needs $\Omega(1/\varepsilon)$ queries to answer this question w.p. $\geq \mathbf{2/3}$.

Proof: By Yao's Minimax Principle, enough to prove Statement 2.

Distribution D on n-bit strings

- Divide the input string into $1/\varepsilon$ blocks of size εn.
- Let y_i be the string where the ith block is 1s and remaining bits are 0.
- Distribution D gives the all-0 string w.p. 1/2 and y_i with w.p. 1/2, where i is chosen uniformly at random from 1, …, $1/\varepsilon$.



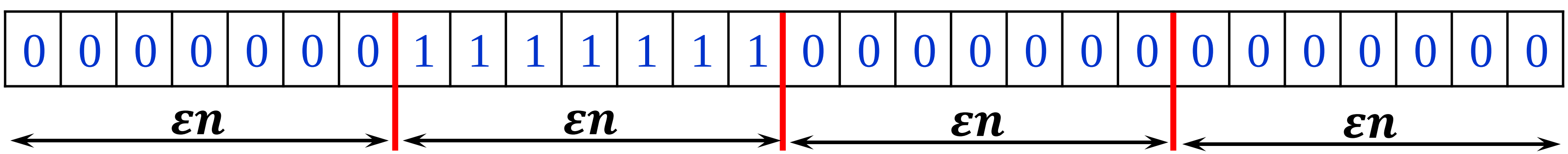
A Lower Bound for Testing 0*

Claim. Any ε-test for 0* needs $\Omega(1/\varepsilon)$ queries.

Proof (continued): Now fix a deterministic tester $\mathcal{A}$ making q < $\frac{1}{3\varepsilon}$ queries.

1. $\mathcal{A}$ must accept if all answers are 0. Otherwise, it would be wrong on all-0 string, that is, with probability 1/2 with respect to D.
2. Let $i_1, \ldots, i_q$ be the positions $\mathcal{A}$ queries when it sees only 0s. The test can choose its queries based on previous answers. However, since all these answers are 0 and since $\mathcal{A}$ is deterministic, the query positions are fixed.

- At least $\frac{1}{\varepsilon} - q > \frac{2}{3\varepsilon}$ of the blocks do not hold any queried indices.
- Therefore, $\mathcal{A}$ accepts > 2/3 of the inputs y_i. Thus, it is wrong with probability > $\frac{2}{3\varepsilon} \cdot \frac{\varepsilon}{2} = \frac{1}{3}$



Context: [Alon Krivelevich Newman Szegedy 99]

Every regular language can be tested in $O(1/\varepsilon \text{ polylog } 1/\varepsilon)$ time

A Lower Bound for Testing Sortedness

Input: a list of n numbers $x_1, x_2, \ldots, x_n$

Question: Is the list sorted or ε-far from sorted?

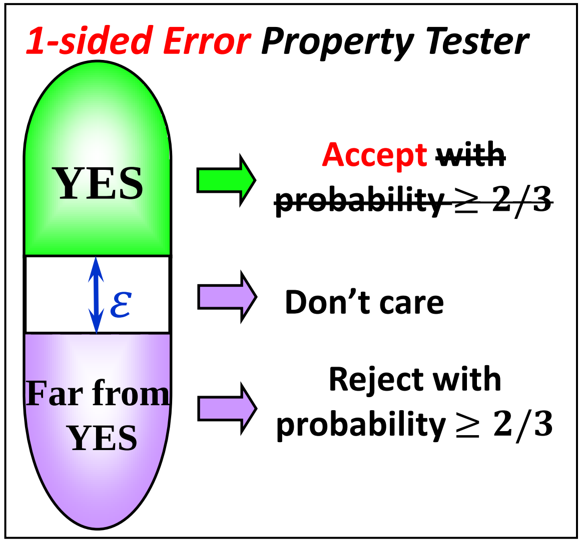
Already saw: an $O((\log n)/\varepsilon)$ time tester.

Known [Ergün Kannan Kumar Rubinfeld Viswanathan 98, Fischer 01]:

$\Omega(\log n)$ queries are required for all constant $\varepsilon \leq 1/2$

Today: $\Omega(\log n)$ queries are required for all constant $\varepsilon \leq 1/2$ for every 1-sided error nonadaptive test.

- A test has 1-sided error if it always accepts all YES instances.
- A test is nonadaptive if its queries do not depend on answers to previous queries.



1-Sided Error Tests Must Catch "Mistakes"

- A pair (i, j) is **violated** if $i < j$ but $x_i > x_j$

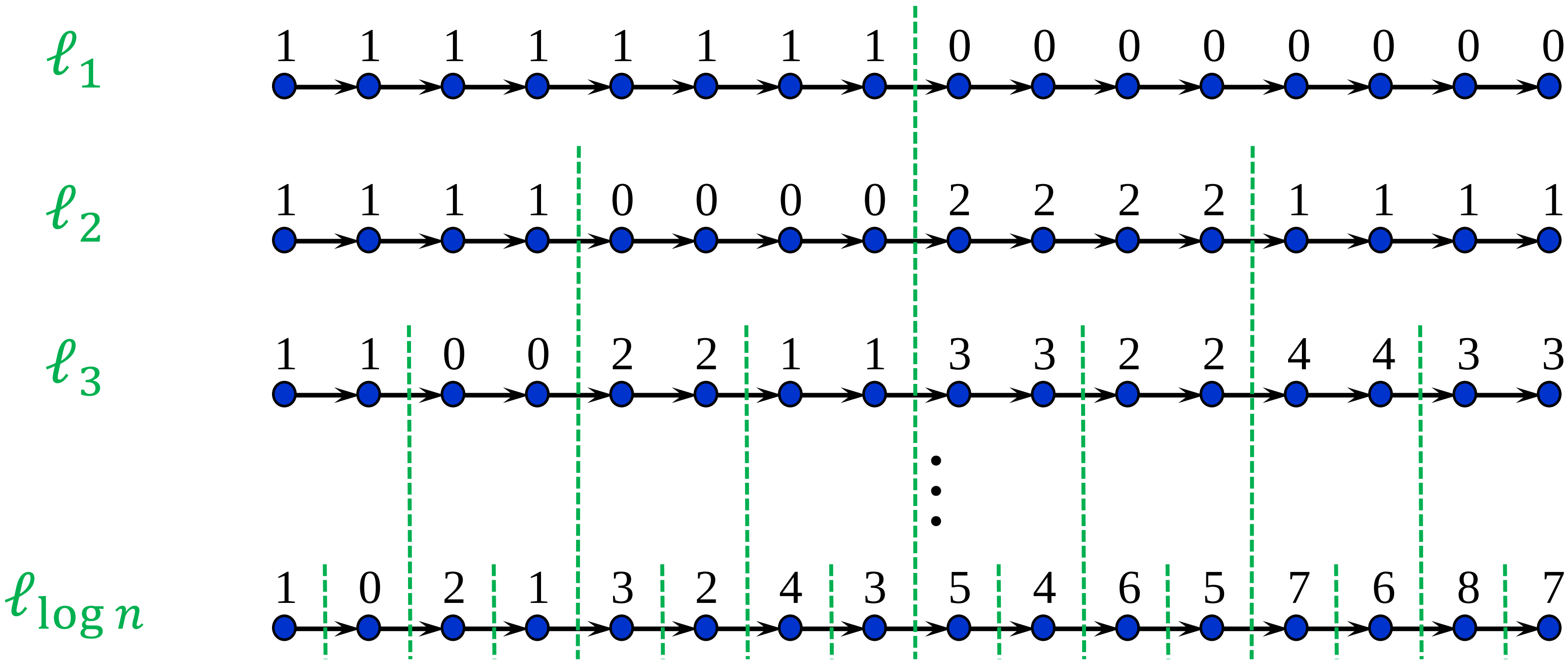
Claim. A 1-sided error test can reject only if it finds a violated pair.

Proof: Every sorted partial list can be extended to a sorted list.

1	?	?	4	…	7	?	?	9

Yao's Principle Game [Jha]

Lola's distribution is uniform over the following $\log n$ lists:

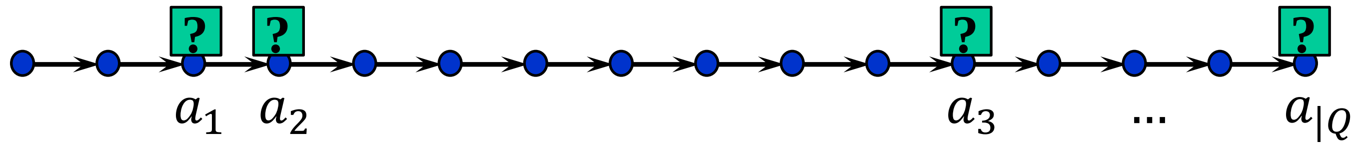


Claim 1. All lists above are 1/2-far from sorted.

Claim 2. Every pair (i, j) is violated in exactly one list above.

Yao's Principle Game: Al's Move

Al picks a set $Q = \{a_1, a_2, \ldots, a_{|Q|}\}$ of positions to query.



- His test must be correct, i.e., must find a violated pair with probability $\geq 2/3$ when input is picked according to Lola's distribution.
- Q contains a violated pair $\Leftrightarrow$ (a_i, a_{i+1}) is violated for some i

$$\Pr_{\ell \leftarrow \text{Lola's distribution}}[(a_i, a_{i+1}) \text{ for some } i \text{ is vilolated in list } \ell] \leq \frac{|Q|-1}{\log n}$$

By the Union Bound

- If $|Q| \leq \frac{2}{3} \log n$ then this probability is $< \frac{2}{3}$
- So, $|Q| = \Omega(\log n)$
- By Yao's Minimax Principle, every randomized 1-sided error nonadaptive test for sortedness must make $\Omega(\log n)$ queries.

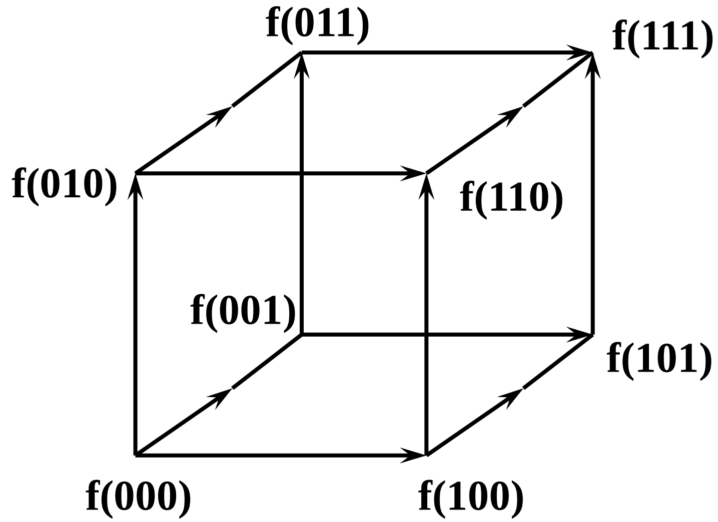
Testing Monotonicity of functions on Hypercube

Non-adaptive 1-sided error Lower Bound

Boolean Functions $f : \{0,1\}^n \to \{0,1\}$

Graph representation:

n-dimensional hypercube



- vertices: bit strings of length n
- edges: (x, y) is an edge if y can be obtained from x by increasing one bit from 0 to 1

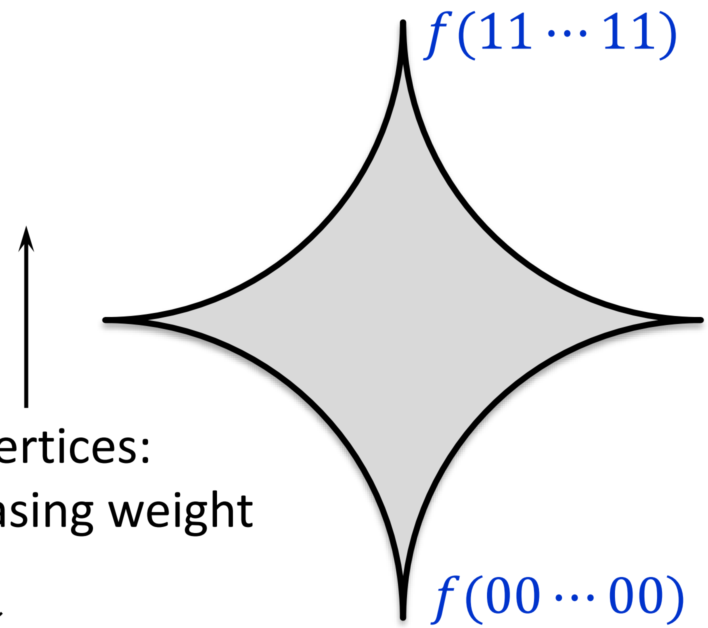
x	001001
y	011001

- each vertex x is labeled with $f(x)$

Boolean Functions $f : \{0,1\}^n \rightarrow \{0,1\}$

Graph representation:

n-dimensional hypercube



- 2^n vertices: bit strings of length n
- $2^{n-1}n$ edges: (x, y) is an edge if y can be obtained from x by increasing one bit from 0 to 1

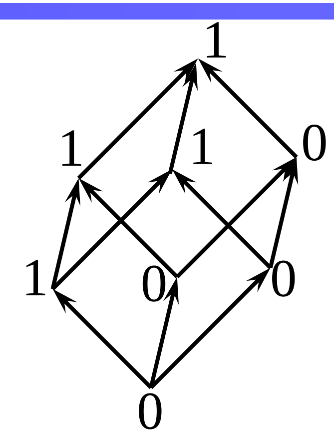
x	001001
y	011001

- each vertex x is labeled with $f(x)$

Monotonicity of Functions

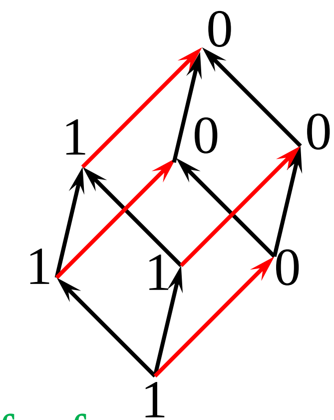
[Goldreich Goldwasser Lehman Ron Samorodnitsky,
Dodis Goldreich Lehman Raskhodnikova Ron Samorodnitsky
Fischer Lehman Newman Raskhodnikova Rubinfeld Samorodnitsky]

- A function $f : \{0,1\}^n \to \{0,1\}$ is monotone if increasing a bit of x does not decrease $f(x)$.



monotone

- Is f monotone or ε-far from monotone (f has to change on many points to become monontone)?
  – Edge $x \to y$ is violated by f if $f(x) > f(y)$.



$\frac{1}{2}$-far from monotone

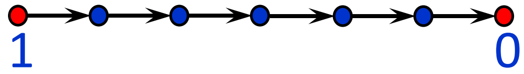
Time:
- – $O(n/\varepsilon)$, logarithmic in the size of the input, 2^n
- – $\Omega(\sqrt{n}/\varepsilon)$ for 1-sided error, nonadaptive tests
- – Advanced techniques: $\Theta(\sqrt{n}/\varepsilon^2)$ for nonadaptive tests, $\Omega(\sqrt[3]{n})$
  [Khot Minzer Safra 15, Chen De Servidio Tang 15, Chen Waingarten Xie 17]

Hypercube 1-sided Error Lower Bound

Lemma [Fischer Lehman Newman Raskhodnikova Rubinfeld Samorodnitsky]

Every 1-sided error nonadaptive test for monotonicity of functions $f : \{0,1\}^n \to \{0,1\}$ requires $\Omega(\sqrt{n})$ queries.

- 1-sided error test must accept if no violated pair is uncovered.

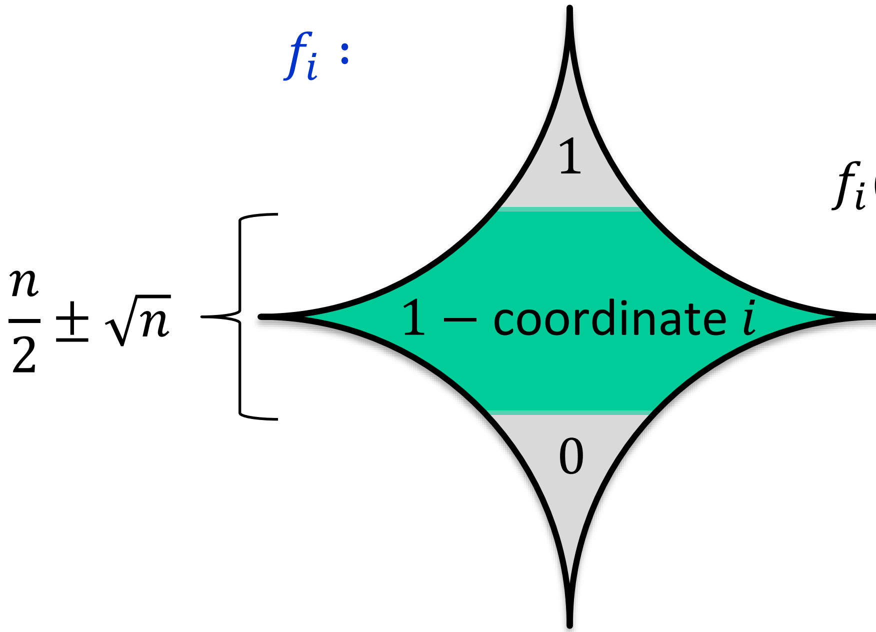
Violated pair: 

  - A distribution on far from monotone functions suffices.

Hypercube 1-sided Error Lower Bound

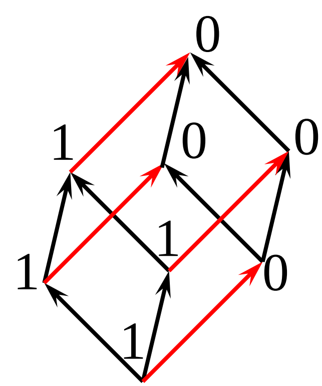
- Hard distribution: pick coordinate i at random and output f_i.

f_i :



$$f_i(x) = \begin{cases} 1 & \text{if } |x| > \frac{n}{2} + \sqrt{n} \\ 1 - x_i & \text{if } |x| = \frac{n}{2} \pm \sqrt{n} \\ 0 & \text{if } |x| < \frac{n}{2} - \sqrt{n} \end{cases}$$

- A ``truncation'' of an antidictator



antidictator

The Fraction of Nodes in Middle Layers

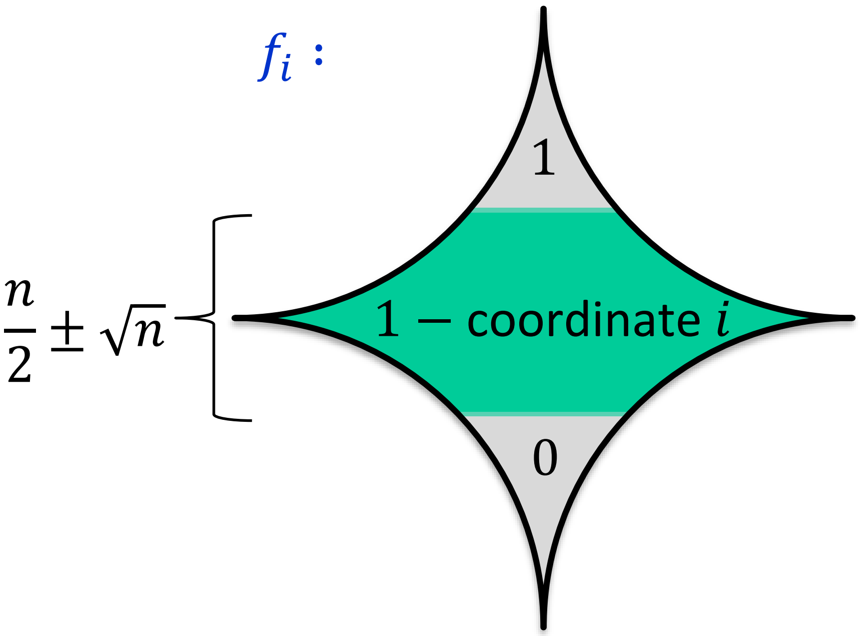
Hoeffding Bound

Let $Y_1, \dots, Y_s$ be independently distributed random variables in [0,1].

Let $Y = \frac{1}{s} \cdot \sum_{i=1}^{s} Y_i$ (called *sample mean*). Then $\Pr[|Y - \mathbb{E}[Y]| \geq \varepsilon] \leq 2e^{-2s\varepsilon^2}$.

$\mathbb{E}[Y]=$

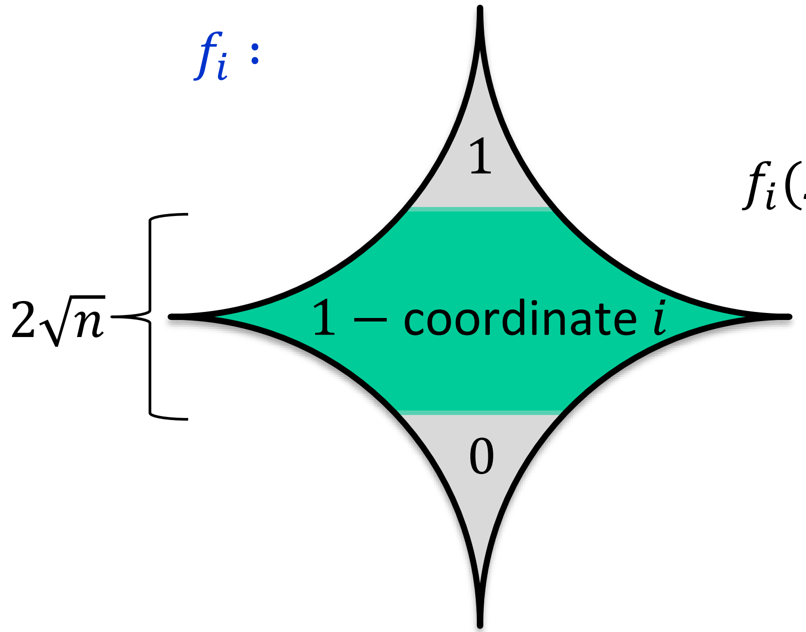
$\varepsilon =$



Hard Functions are Far

- Hard distribution: pick coordinate i at random and output f_i.

f_i :



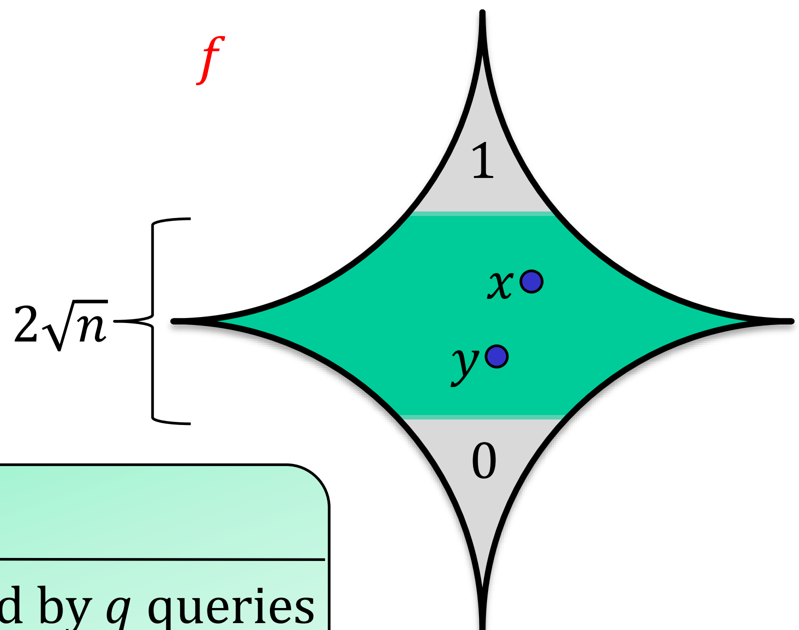
$$f_i(x) = \begin{cases} 1 & \text{if } |x| > \frac{n}{2} + \sqrt{n} \\ 1 - x_i & \text{if } |x| = \frac{n}{2} \pm \sqrt{n} \\ 0 & \text{if } |x| < \frac{n}{2} - \sqrt{n} \end{cases}$$

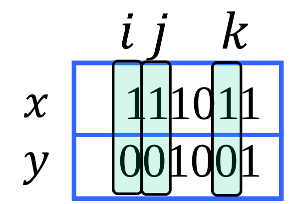
Analysis

- The middle contains a constant fraction of vertices.
- Edges from $(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$ to $(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$ are violated if both endpoints are in the middle.
- All n functions are ε-far from monotone for some constant ε.

Hypercube 1-sided Error Lower Bound

- How many functions does a set of q queries expose?





Pair (x, y)
can expose only
functions f_i, f_j and f_k

Naive Analysis

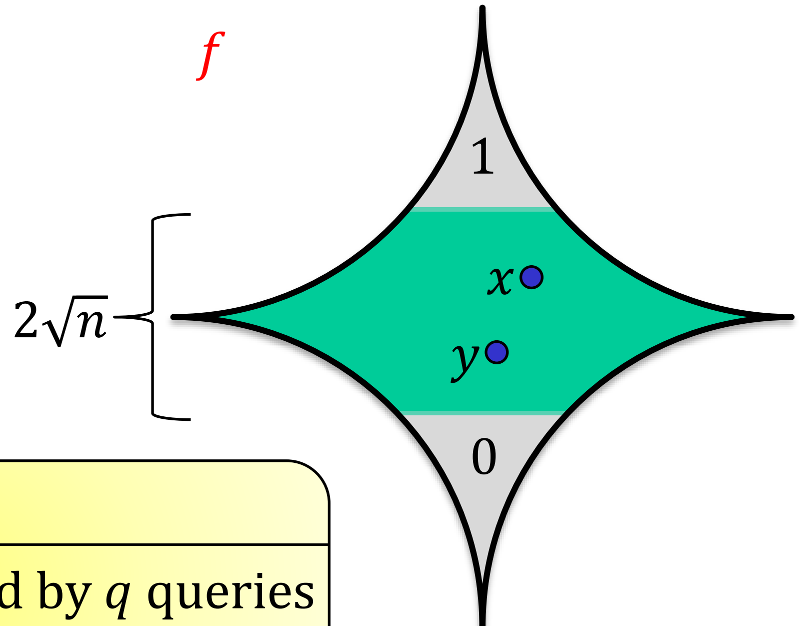
\# functions exposed by q queries
$\leq q^2 \cdot 2\sqrt{n}$

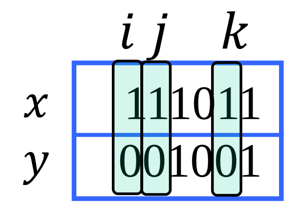
\# functions that a query pair (x, y) exposes
$\leq$ # coordinates on which x and y differ
$\leq 2\sqrt{n}$

Only pairs of queries in the Green Band can be violated $\Rightarrow$ disagreements $\leq 2\sqrt{n}$

Hypercube 1-sided Error Lower Bound

- How many functions does a set of q queries expose?





Pair (x, y)
can expose only
functions f_i, f_j and f_k

Claim

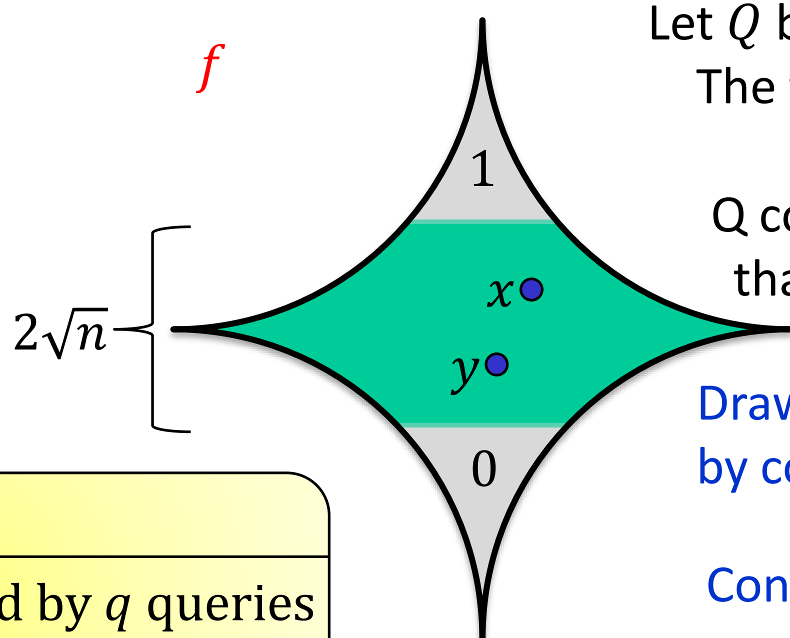
\# functions exposed by q queries
$\leq (q-1) \cdot 2\sqrt{n}$

\# functions that a query pair (x, y) exposes
$\leq$ # coordinates on which x and y differ
$\leq 2\sqrt{n}$

Only pairs of queries in the Green Band can be violated $\Rightarrow$ disagreements $\leq 2\sqrt{n}$

Hypercube 1-sided Error Lower Bound

- How many functions does a set of q queries expose?



Let Q be the set of queries made.
The tester catches a violation
⇕
Q contains comparable x, y
that differ in coordinate i

Draw an undirected graph (Q, E)
by connected comparable queries

Consider its spanning forest.

x, y exist
⇕
there are adjacent vertices on the path
from x to y that differ in coordinate i

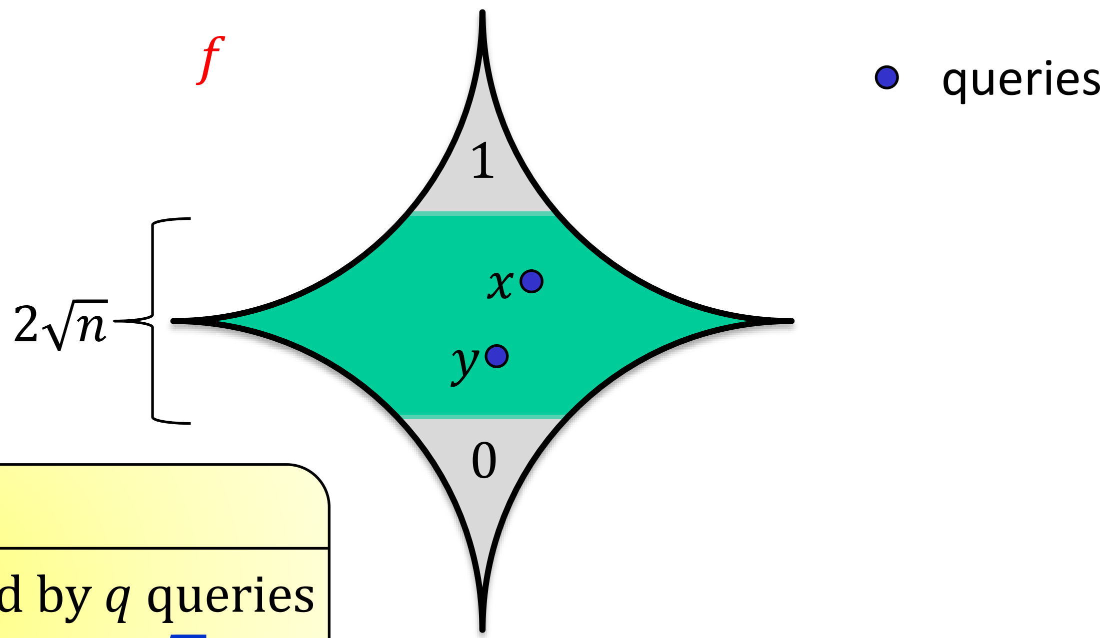
Claim

\# functions exposed by q queries
$\leq (q-1) \cdot 2\sqrt{n}$

sufficient to consider adjacent vertices in a minimum spanning forest on the query set

Hypercube 1-sided Error Lower Bound

- How many functions does a set of q queries expose?



Claim

\# functions exposed by q queries
$\leq (q-1) \cdot 2\sqrt{n}$

$\Downarrow$

Claim

Every deterministic test that makes a set Q of q queries (in the middle) succeeds with probability $O\left(\frac{q}{\sqrt{n}}\right)$ on our distribution.

Hypercube 1-sided Error Lower Bound

Lemma [Fischer Lehman Newman Raskhodnikova Rubinfeld Samorodnitsky]

Every 1-sided error nonadaptive test for monotonicity of functions $f : \{0,1\}^n \to \{0,1\}$ requires $\Omega(\sqrt{n})$ queries.