Finding Monotone Patterns in Sublinear Time

Erik Waingarten (Columbia University)
Clément Canonne (Stanford University)
Omri Ben-Eliezer (Tel-Aviv University)
Shoham Letzter (ETH Zurich)
Given query access to an unknown $f : [n] \rightarrow \mathbb{R}$ and a parameter $\varepsilon > 0$:

- If $f$ monotone, accept w.p. $> 2/3$;
- If $f$ is $\varepsilon$-far from monotone, reject w.p. $> 2/3$;

Minimum query complexity in terms of $n$ and $\varepsilon$?
Testing Monotonicity of an Array: *Sortedness*

Given query access to an unknown $f : [n] \rightarrow \mathbb{R}$ and a parameter $\varepsilon > 0$:

- If $f$ monotone, accept w.p. $> 2/3$;
- If $f$ is $\varepsilon$-far from monotone, reject w.p. $> 2/3$;

Minimum query complexity in terms of $n$ and $\varepsilon$?

- If $f$ is $\varepsilon$-far from monotone, find evidence:
  - $i, j \in [n]$ where $i < j$ and $f(i) > f(j)$. 

```
1 3 2 3 6 7 1 9 4 2 5 6 7
```
Testing Monotonicity of an Array: Sortedness

Given query access to an unknown $f : [n] \rightarrow \mathbb{R}$ and a parameter $\varepsilon > 0$:

- If $f$ monotone, accept w.p. $> 2/3$;
- If $f$ is $\varepsilon$-far from monotone, reject w.p. $> 2/3$;

Minimum query complexity in terms of $n$ and $\varepsilon$?

- If $f$ is $\varepsilon$-far from monotone, find evidence:
  - $i, j \in [n]$ where $i < j$ and $f(i) > f(j)$. one-sided error.
ε-Far-From-Monotone Sequences

- $f$ is $\varepsilon$-far from monotone: for any monotone $g: [n] \rightarrow \mathbb{R}$,

\[
\frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{f(i) \neq g(i)\} \geq \varepsilon.
\]
ε-Far-From-Monotone Sequences

- $f$ is $\varepsilon$-far from monotone: for any monotone $g : [n] \to \mathbb{R}$,

$$\frac{1}{n} \sum_{i=1}^{n} 1\{f(i) \neq g(i)\} \geq \varepsilon.$$

Lemma

Let $f : [n] \to \mathbb{R}$ be $\varepsilon$-far from monotone. There exists set of disjoint pairs,

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) > f(j)\}$$

of size $|T| \geq \varepsilon n/2$. 
Monotonicity Testing

Theorem (Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99)

There exists a non-adaptive, one-sided algorithm for testing monotonicity of \( f : [n] \rightarrow \mathbb{R} \) making \( O((\log n)/\varepsilon) \) queries.
Monotonicity Testing

Theorem (Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99)

There exists a non-adaptive, one-sided algorithm for testing monotonicity of $f : [n] \rightarrow \mathbb{R}$ making $O((\log n)/\varepsilon)$ queries.

- $\Omega((\log n)/\varepsilon)$ queries needed for non-adaptive algorithms.
Monotonicity Testing

**Theorem (Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99)**

There exists a non-adaptive, one-sided algorithm for testing monotonicity of $f : [n] \rightarrow \mathbb{R}$ making $O((\log n)/\varepsilon)$ queries.

- $\Omega((\log n)/\varepsilon)$ queries needed for non-adaptive algorithms.
- $\Omega((\log n)/\varepsilon)$ queries needed for adaptive algorithms too! [Fischer 04].
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]

Definition

Let \( k \in \mathbb{N} \) and \( \pi = (\pi_1, \ldots, \pi_k) \) be a permutation of \([k]\). Given \( f : [n] \to \mathbb{R} \), the \( k \)-tuple \((i_1, \ldots, i_k)\) has order pattern \( \pi \) if:

\[
f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.
\]
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]

Definition

Let $k \in \mathbb{N}$ and $\pi = (\pi_1, \ldots, \pi_k)$ be a permutation of $[k]$. Given $f : [n] \rightarrow \mathbb{R}$, the $k$-tuple $(i_1, \ldots, i_k)$ has order pattern $\pi$ if:

$$f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.$$

If $f : [n] \rightarrow \mathbb{R}$ is $\varepsilon$-far from $\pi$-free, then there exists $T \subset [n]^k$ of disjoint violating $k$-tuples $(i_1, \ldots, i_k)$ with order pattern $\pi$ of size at least $\varepsilon n/k$. 

For fixed $k$ and $\pi$, query complexity of testing $\pi$-freeness?

Some sublinear in $n$ upper bounds for general $\pi$.

$\pi = (132)$ requires $\Omega(\sqrt{n})$ queries for non-adaptive, one-sided algorithms.

[Ben-Eliezer, Canonne 18] Many $\pi$ have complexity $a n^{1-1/(k-\Theta(1))}$. 

5 / 14
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]

Definition

Let $k \in \mathbb{N}$ and $\pi = (\pi_1, \ldots, \pi_k)$ be a permutation of $[k]$. Given $f : [n] \rightarrow \mathbb{R}$, the $k$-tuple $(i_1, \ldots, i_k)$ has order pattern $\pi$ if:

$$f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.$$ 

If $f : [n] \rightarrow \mathbb{R}$ is $\varepsilon$-far from $\pi$-free, then there exists $T \subset [n]^k$ of disjoint violating $k$-tuples $(i_1, \ldots, i_k)$ with order pattern $\pi$ of size at least $\varepsilon n/k$.

For fixed $k$ and $\pi$, query complexity of testing $\pi$-freeness?
### Testing Forbidden Order Patterns

[Newman, Rabinovich, Rajendra prasad, Sohler 17]

#### Definition

Let $k \in \mathbb{N}$ and $\pi = (\pi_1, \ldots, \pi_k)$ be a permutation of $[k]$. Given $f : [n] \to \mathbb{R}$, the $k$-tuple $(i_1, \ldots, i_k)$ has order pattern $\pi$ if:

$$f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.$$

If $f : [n] \to \mathbb{R}$ is $\varepsilon$-far from $\pi$-free, then there exists $T \subset [n]^k$ of disjoint violating $k$-tuples $(i_1, \ldots, i_k)$ with order pattern $\pi$ of size at least $\varepsilon n / k$.

---

For fixed $k$ and $\pi$, query complexity of testing $\pi$-freeness?

- Some sublinear in $n$ upper bounds for general $\pi$. 

---
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]

Definition
Let $k \in \mathbb{N}$ and $\pi = (\pi_1, \ldots, \pi_k)$ be a permutation of $[k]$. Given $f : [n] \rightarrow \mathbb{R}$, the $k$-tuple $(i_1, \ldots, i_k)$ has order pattern $\pi$ if:

$$f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.$$

If $f : [n] \rightarrow \mathbb{R}$ is $\varepsilon$-far from $\pi$-free, then there exists $T \subset [n]^k$ of disjoint violating $k$-tuples $(i_1, \ldots, i_k)$ with order pattern $\pi$ of size at least $\varepsilon n/k$.

For fixed $k$ and $\pi$, query complexity of testing $\pi$-freeness?

- Some sublinear in $n$ upper bounds for general $\pi$.
- $\pi = (132)$ requires $\Omega(\sqrt{n})$ queries for non-adaptive, one-sided algorithms.
Testing Forbidden Order Patterns
[Newman, Rabinovich, Rajendraprasad, Sohler 17]

Definition

Let $k \in \mathbb{N}$ and $\pi = (\pi_1, \ldots, \pi_k)$ be a permutation of $[k]$. Given $f : [n] \to \mathbb{R}$, the $k$-tuple $(i_1, \ldots, i_k)$ has order pattern $\pi$ if:

$$f(i_\ell) < f(i_m) \text{ whenever } \pi_\ell < \pi_m.$$ 

If $f : [n] \to \mathbb{R}$ is $\varepsilon$-far from $\pi$-free, then there exists $T \subset [n]^k$ of disjoint violating $k$-tuples $(i_1, \ldots, i_k)$ with order pattern $\pi$ of size at least $\varepsilon n/k$.

For fixed $k$ and $\pi$, query complexity of testing $\pi$-freeness?

- Some sublinear in $n$ upper bounds for general $\pi$.
- $\pi = (132)$ requires $\Omega(\sqrt{n})$ queries for non-adaptive, one-sided algorithms.
- [Ben-Eliezer, Canonne 18] Many $\pi$ have complexity a $n^{1-1/(k-\Theta(1))}$. 

Finding Monotone Patterns: $\pi = (12\ldots k)$
Finding Monotone Patterns: $\pi = (12\ldots k)$

Given query access to $f : [n] \rightarrow \mathbb{R}$ and a parameter $\varepsilon > 0$ where:

Theorem (NRRS17)

There is a non-adaptive algorithm with query complexity $\frac{\log n}{\varepsilon} \cdot \Theta(k^2)$. 
Finding Monotone Patterns: \( \pi = (12 \ldots k) \)

Given query access to \( f : [n] \rightarrow \mathbb{R} \) and a parameter \( \varepsilon > 0 \) where:

- There exists \( T \subset [n]^k \) of disjoint violating \( k \)-tuples

\[
T = \{(i_1, \ldots, i_k) : i_1 < \cdots < i_k \text{ and } f(i_1) < \cdots < f(i_k)\}
\]

of size \( |T| \geq \varepsilon n/k \).
Finding Monotone Patterns: $\pi = (12 \ldots k)$

Given query access to $f : [n] \rightarrow \mathbb{R}$ and a parameter $\varepsilon > 0$ where:

- There exists $T \subset [n]^k$ of disjoint violating $k$-tuples
  
  $$T = \{ (i_1, \ldots, i_k) : i_1 < \cdots < i_k \text{ and } f(i_1) < \cdots < f(i_k) \}$$

  of size $|T| \geq \varepsilon n/k$.

- Find $i_1 < \cdots < i_k$ where $f(i_1) < \cdots < f(i_k)$. 

Theorem (NRRS17)

There is a non-adaptive algorithm with query complexity $O((\log n)/\varepsilon^{1/2}k^2)$. 
Finding Monotone Patterns: \( \pi = (12 \ldots k) \)

Given query access to \( f : [n] \rightarrow \mathbb{R} \) and a parameter \( \varepsilon > 0 \) where:

- There exists \( T \subset [n]^k \) of disjoint violating \( k \)-tuples
  \[
  T = \{(i_1, \ldots, i_k) : i_1 < \cdots < i_k \text{ and } f(i_1) < \cdots < f(i_k)\}
  \]
  of size \( |T| \geq \varepsilon n/k \).
- Find \( i_1 < \cdots < i_k \) where \( f(i_1) < \cdots < f(i_k) \).

**Theorem (NRRS17)**

*There is a non-adaptive algorithm with query complexity \( (\log n / \varepsilon)^{O(k^2)} \).*
Finding Monotone Patterns: $\pi = (12 \ldots k)$
Finding Monotone Patterns: \( \pi = (12 \ldots k) \)

**Theorem (Upper bound)**

For \( k \in \mathbb{N} \), there exists a non-adaptive algorithm with query complexity

\[
(\log n)^{\lfloor \log_2 k \rfloor} \cdot \text{poly}(1/\varepsilon).
\]
Finding Monotone Patterns: $\pi = (12 \ldots k)$

**Theorem (Upper bound)**

For $k \in \mathbb{N}$, there exists a non-adaptive algorithm with query complexity

$$(\log n)^{\lfloor \log_2 k \rfloor} \cdot \text{poly}(1/\varepsilon).$$

**Theorem (Lower bound)**

Any non-adaptive algorithm needs to make $\Omega \left( (\log n)^{\lfloor \log_2 k \rfloor} \right)$ queries.
The Case of $k = 2$
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$  

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$ 

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2s]$ and query $f(j)$.
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n / 2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$. 

The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$. 

The Case of $k = 2$

Let $f : [n] \rightarrow \mathbb{R}$ and disjoint subset of pairs $T$ of size $\epsilon n/2$ with

$$T = \{ (i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j) \}.$$ 

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$. 

$$f = \begin{array}{c}
\end{array}$$
The Case of $k = 2$

Let $f : [n] \rightarrow \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$ 

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$. 

\[
\begin{array}{c}
\ell \\
f = \hline
\end{array}
\]
The Case of $k = 2$

Let $f : [n] \rightarrow \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$.
The Case of $k = 2$

Let $f : [n] \rightarrow \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$.
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$ 

[Er"gun, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$.
The Case of $k = 2$

Let $f : [n] \to \mathbb{R}$ and disjoint subset of pairs $T$ of size $\varepsilon n/2$ with

$$T = \{(i, j) \in [n]^2 : i < j \text{ and } f(i) < f(j)\}.$$ 

[Ergün, Kannan, Kumar, Rubinfeld, Viswanathan 99] Find increasing pair:

- Sample $\ell \sim [n]$ uniformly and repeat the following for $t$ iterations:
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $i \sim [\ell - 2^s, \ell]$ and query $f(i)$.
  - Sample $s \sim \{0, \ldots, \log_2 n\}$ and $j \sim [\ell, \ell + 2^s]$ and query $f(j)$.
Analysis

Let $T$ be set of monotone pairs. 

$((i, j) \in T)$ scale $s$ when $j - i \approx 2s$.

$Pr \ell \sim [n] [i \leq \ell \leq j] \approx 2s n$. 

$f = \mathbb{R}$
Let $T$ be set of monotone pairs.
Let $T$ be set of monotone pairs.

$(i, j) \in T$. 
Let $T$ be set of monotone pairs.

$(i, j) \in T$. $(i, j)$ scale $s$ when $j - i \approx 2^s$.

$$\Pr_{\ell \sim [n]} [i \leq \ell \leq j] \approx \frac{2^s}{n}.$$
\[ f = \mathbb{R} \]

\[
\mathbb{E}_{\ell \sim [n]} \left[ \sum_{s=1}^{\log n} \frac{\#(i,j) \text{ scale } s \text{ cut}}{2^s + 1} \right] \gtrsim \varepsilon.
\]
Sample $\ell$ such that:

$$\sum_{s=1}^{\log n} \frac{\# (i,j) \text{ scale } s \text{ cut}}{2^{s+1}} = \sum_{s=1}^{\log n} \left( \text{density of } T \text{ at } 2^{s+1} \right) \gtrsim \varepsilon.$$
Sample $\ell$ such that:

$$\frac{\log n}{\sum_{s=1}^{\log n} \frac{\#(i,j) \text{ scale } s \text{ cut}}{2^{s+1}}} = \sum_{s=1}^{\log n} (\text{density of } T \text{ at } 2^{s+1}) \geq \varepsilon.$$
\[ f = \ell \]

Sample \( \ell \) such that:

\[
\sum_{s=1}^{\log n} \frac{\# \{(i,j) \text{ scale } s \text{ cut } 2^s + 1}}{2^{s+1}} = \sum_{s=1}^{\log n} (\text{density of } T \text{ at } 2^{s+1}) \geq \varepsilon.
\]
Sample $\ell$ such that:

$$\sum_{s=1}^{\log n} \frac{\# (i, j) \text{ scale } s \text{ cut}}{2^{s+1}} = \sum_{s=1}^{\log n} \left( \text{density of } T \text{ at } 2^{s+1} \right) \gtrsim \varepsilon.$$
Sample $\ell$ such that:

$$\sum_{s=1}^{\log n} \frac{\# (i,j) \text{ scale } s \text{ cut}}{2^{s+1}} = \sum_{s=1}^{\log n} (\text{density of } T \text{ at } 2^{s+1}) \gtrsim \varepsilon.$$
Density $\delta_s^-$ and $\delta_s^+$: $\sum_{s=1}^{\log n} \delta_s^- \geq \varepsilon$ and $\sum_{s=1}^{\log n} \delta_s^+ \geq \varepsilon$.

$$\Pr_{i_1, \ldots, i_t, j_1, \ldots, j_t} [\text{avoid pair}] \leq \left( \frac{1}{\log n} \sum_{s=1}^{\log n} (1 - \delta_s^-) \right)^t + \left( \frac{1}{\log n} \sum_{s=1}^{\log n} (1 - \delta_s^+) \right)^t \leq \frac{1}{3}.$$
Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

Round 1: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.

Round $r$, $2 \leq r \leq \lfloor \log_2 k \rfloor + 1$: For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i-2s, i+2s]$ and include them in $A$.

Query $f(i)$ for all $i \in A$.

**Query Complexity:** $O(1/\varepsilon) \cdot (O(\log n/\varepsilon)) \lfloor \log_2 k \rfloor$.

**Theorem** Suppose $f: [n] \rightarrow \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p. $\geq 2/3$. 
Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- Round 1: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$. 

Query $f(i)$ for all $i \in A$.

Query Complexity: $O(1/\varepsilon) \cdot (O(\log n/\varepsilon))^\lceil \log_2 k \rceil$.

Theorem: Suppose $f : [n] \to \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p. $\geq 2/3$. 


Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- **Round 1**: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.
- **Round $r$, $2 \leq r \leq \lceil \log_2 k \rceil + 1** : For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i - 2^s, i + 2^s]$ and include them in $A$. 

Query $f(i)$ for all $i \in A$.

**Query Complexity**: $O(1/\varepsilon) \cdot (O(\log n/\varepsilon) \cdot \lceil \log_2 k \rceil)$.

**Theorem**: Suppose $f : [n] \rightarrow R$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p $\geq \frac{2}{3}$. 

Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- **Round 1**: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.
- **Round $r$, $2 \leq r \leq \lfloor \log_2 k \rfloor + 1****: For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i - 2^s, i + 2^s]$ and include them in $A$.
- **Query** $f(i)$ for all $i \in A$.

*Query Complexity:* $O(1/\varepsilon) \cdot (O(\log n/\varepsilon))^{\lfloor \log_2 k \rfloor}$.

*Theorem* Suppose $f: [n] \rightarrow \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p $\geq 2/3$. 

Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- Round 1: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.
- Round $r$, $2 \leq r \leq \lceil \log_2 k \rceil + 1$: For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i - 2^s, i + 2^s]$ and include them in $A$.
- Query $f(i)$ for all $i \in A$.

![Diagram of a function $f$ with marked intervals](image)

**Query Complexity:** $O(1/\varepsilon) \cdot \left(\frac{O(\log n/\varepsilon)}{\varepsilon}\right) \cdot \lceil \log_2 k \rceil$.

**Theorem**

Suppose $f : [n] \rightarrow \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p $\geq 2/3$. 

13 / 14
Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- **Round 1:** Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.
- **Round $r$, $2 \leq r \leq \lfloor \log_2 k \rfloor + 1:** For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i - 2^s, i + 2^s]$ and include them in $A$.
- **Query $f(i)$ for all $i \in A$.**

**Query Complexity:** $O(1/\varepsilon) \cdot (O(\log n/\varepsilon))^{\lfloor \log_2 k \rfloor}$.

Theorem: Suppose $f : [n] \to \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p $\geq 2/3$. 

**Diagram:**

![Diagram of a function $f$ with a section labeled $\ell$.]
Monotone patterns for $k > 2$?

New algorithm for finding $(12 \ldots k)$ patterns:

- Round 1: Sample $O(1/\varepsilon)$ indices from $[n]$, include them in a set $A$.
- Round $r$, $2 \leq r \leq \lfloor \log_2 k \rfloor + 1$: For each $i \in A$ and $s \in \{1, \ldots, \log n\}$, sample $O(1/\varepsilon)$ indices from $[i - 2^s, i + 2^s]$ and include them in $A$.
- Query $f(i)$ for all $i \in A$.

Query Complexity: $O(1/\varepsilon) \cdot (O(\log n/\varepsilon))^{\lfloor \log_2 k \rfloor}$.

**Theorem**

Suppose $f : [n] \rightarrow \mathbb{R}$ contains $\varepsilon n/k$ disjoint $(12 \ldots k)$-patterns, algorithm finds one w.p $\geq 2/3$. 
Open Problems
Open Problems

- More efficient tests for *two-sided* error testing for \((12 \ldots k)\)-freeness.
Open Problems

- More efficient tests for *two-sided* error testing for \((12 \ldots k)\)-freeness.
- Adaptive algorithms for finding \(\pi\)-patterns?
Open Problems

- More efficient tests for \textit{two-sided} error testing for \((12 \ldots k)\)-freeness.
- Adaptive algorithms for finding \(\pi\)-patterns?
  - Possibly \(\text{poly}(\log n)\) adaptive queries for any pattern \(\pi\)?
Open Problems

- More efficient tests for *two-sided* error testing for \((12 \ldots k)\)-freeness.
- Adaptive algorithms for finding \(\pi\)-patterns?
  - Possibly \(\text{poly}(\log n)\) adaptive queries for any pattern \(\pi\)?

Thanks!