Sublinear-time Computation in the Presence of Online Erasures

Iden Kalemaj    Sofya Raskhodnikova    Nithin Varma

22 November 2021

To appear at ITCS '22
Goal of this work: study basic computational tasks in extremely adversarial environments

- property testing tasks
- algorithm has query access to a very large dataset via an oracle
- answer yes/no questions about global properties of the dataset

- an adversary/oracle makes changes to the dataset
- we focus on erasures
- the changes happen "online", as the dataset is being queried
- adversary can adapt to actions of algorithm
Does $f$ have a property, or is it $\varepsilon$-far from having the property?

$\varepsilon$-far: $f$ must be modified in at least $\varepsilon$ fraction of its domain to have the property.

- Accept if $f$ has property
- Reject w.h.p. if $f$ is $\varepsilon$-far from property

- Interested in query complexity of tester
- #queries should be sublinear in size of domain of $f$
We want to make tester robust to:

- data is missing/corrupted
- data is erased/corrupted adversarially
- privacy concerns

Standard property testing model

We want to make tester robust to:

- Accept if $f$ has property
- Reject w.h.p. if $f$ is $\epsilon$-far from property

Tolerant property testing

Erasure-resilient property testing
Offline Erasures Model

• Property testing with erasures was first studied by Dixit Raskhodnikova Thakurta Varma '18

• Oracle erases at most $\alpha$ fraction of the input values, before algorithm makes any queries.

• What if erasures happen during the querying process?

$\alpha$  $1 - \alpha$

Function $f$  $\varepsilon$-tester
Oracle can erase $t$ entries after answering each query of the tester

$t = 1$

<table>
<thead>
<tr>
<th>$x$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x)$</td>
<td>$f(1)$</td>
<td>$f(2)$</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>$f(8)$</td>
</tr>
</tbody>
</table>

Does $f$ have a property, or is it far from having the property?
Does $f$ have a property, or is it far from having the property?

Oracle can erase $t$ entries after answering each query of the tester

$t = 1$

<table>
<thead>
<tr>
<th>$x$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x)$</td>
<td>$f(1)$</td>
<td>$f(2)$</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>$f(8)$</td>
</tr>
</tbody>
</table>

$2 \uparrow \quad f(2) \downarrow$

Does $f$ have a property, or is it far from having the property?
Online Erasures Model

Oracle can erase \( t \) entries after answering each query of the tester

\( t = 1 \)

\[
\begin{array}{cccccccc}
   x    & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
   f(x) &  &  & f(2) & \cdot & \cdot & \cdot & \cdot & f(8) \\
\end{array}
\]

Does \( f \) have a property, or is it far from having the property?
Online Erasures Model

Oracle can erase $t$ entries after answering each query of the tester

$t = 1$

<table>
<thead>
<tr>
<th>$x$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x)$</td>
<td>$f(1)$</td>
<td>$f(2)$</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>$f(8)$</td>
</tr>
</tbody>
</table>

Does $f$ have a **property**, or is it far from having the **property**?
Online-Erasure-Resilient Tester

Oracle can erase $t$ entries after answering each query of the tester

<table>
<thead>
<tr>
<th>$x$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(x)$</td>
<td>$f(1)$</td>
<td>$f(2)$</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>$f(8)$</td>
</tr>
</tbody>
</table>

- Accept if $f$ has property.
- Reject whp if $f$ is $\varepsilon$-far from property.
Assumptions:

• Oracle knows the description of the algorithm

• Oracle does not have access to random coins of algorithm

Example:

"Query location 1 with probability ½ and query location 2 with remaining probability"
Does $f$ have a property, or is it far from having the property?

Oracle can erase $t$ entries after answering each query of the tester.
Motivating Scenarios

- Individuals request that their data be removed from a dataset
  - They are prompted to restrict access to their data after noticing an inquiry into their or other's data (online)
  - Adversarial assumption allows us to study worst-case
- In a criminal investigation / fraud detection setting, adversary reacts by erasing data after some of their records are pulled by authorities
- In legal setting, adversary is served a subpoena; after answering the query, they can destroy related evidence not involved in the subpoena
  - In our model, adversary can make erasures only after answering the query of the algorithm
Results

• Some properties can be tested with the same query complexity as in the standard model:
  • linearity and quadraticity (for constant erasure budget $t$)
• For linearity, we show matching upper and lower bounds in terms of $t$
• Some properties are impossible to test, even for $t = 1$: sortedness of arrays
  • The structure of violations to the property plays a role in determining testability
Plan

- Show the tester for linearity (with a light proof)
- Show the lower bound for linearity
- Show idea behind tester for quadraticity
- Show the impossibility of testing sortedness
Function $f: \{0,1\}^d \to \{0,1\}$ is **linear** if can be expressed as sum of $x[i], \ i \in [d]$

Equivalently, if $f(x) + f(y) = f(x + y)$ for all $x, y$ in domain.

<table>
<thead>
<tr>
<th>Standard Model</th>
<th>Online-Erasures Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Blum Luby Rubinfeld '93] [Bellare Coppersmith Hastad Kiwi Sudan '96]</td>
<td>This work</td>
</tr>
<tr>
<td>$O\left(\frac{1}{\epsilon}\right)$ queries</td>
<td>$\tilde{O}\left(\frac{\log t}{\epsilon}\right)$ queries</td>
</tr>
</tbody>
</table>

**BLR Tester:**
- Sample $x, y \sim \{0,1\}^d$.
- Query $f(x), f(y), f(x + y)$.
- Reject if $f(x) + f(y) \neq f(x + y)$.

**Issue with standard linearity tester:**
- Query $x$. Receive $f(x)$.
- Query $y$. Receive $f(y)$.
- Oracle erases $x + y$.

If $f: \{0,1\}^d \to \{0,1\}$ is $\epsilon$-far from linear then an $\epsilon$-fraction of pairs $(x, y)$ violate linearity.
**Linearity**

BLR Tester:

- Sample $x, y \sim \{0,1\}^d$.
- Query $f(x), f(y), f(x + y)$.
- Reject if $f(x) + f(y) \neq f(x + y)$.

2-player game:

1) Player 1 draws a vertex or edge connecting two vertices in **blue**
2) Player 2 draws an edge between existing vertices in **red**

Can you come up with winning strategy for player 1?
**Linearity**

Function $f: \{0,1\}^d \rightarrow \{0,1\}$ is **linear** if can be expressed as sum of $x[i], \ i \in [d]$

Equivalently, if $f(x) + f(y) = f(x + y)$ for all $x, y$ in domain.

<table>
<thead>
<tr>
<th>Standard Model</th>
<th>Online-Erasures Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Blum Luby Rubinfeld '93] [Bellare Coppersmith Hastad Kiwi Sudan '96]</td>
<td>This work</td>
</tr>
<tr>
<td>$O\left(\frac{1}{\varepsilon}\right)$ queries</td>
<td>$\tilde{O}\left(\frac{\log t}{\varepsilon}\right)$ queries</td>
</tr>
</tbody>
</table>

**BLR Tester:**
- Sample $x, y \sim \{0,1\}^d$.
- Query $f(x), f(y), f(x + y)$.
- Reject if $f(x) + f(y) \neq f(x + y)$.

- **If $f: \{0,1\}^d \rightarrow \{0,1\}$ is $\varepsilon$-far from linear then an $\varepsilon$-fraction of pairs $(x, y)$ violate linearity.**

**Issue with standard linearity tester:**
- Query $x$. Receive $f(x)$.
- Query $y$. Receive $f(y)$.
- Oracle erases $x + y$.

**Thm.** If $f: \{0,1\}^d \rightarrow \{0,1\}$ is $\varepsilon$-far from linear then, for all even $k$, an $\varepsilon$-fraction of $k$-tuples $(x_1, x_2, \ldots, x_k)$ violate linearity.

$f(x_1) + \cdots + f(x_k) \neq f(x_1 + \cdots + x_k)$

**Proof via Fourier analysis**
Linearity

**Algorithm. Online-erasure-resilient linearity tester**

(1) Query $q = 2\log(t/\varepsilon)$ points $x_i \sim \{0,1\}^d$

(2) Repeat $1/\varepsilon$ times:
   - Sample nonempty even-sized subset $I$ of $[q]$
   - Query $f$ at $\sum_{i \in I} x_i$
   - **Reject** if $\sum_{i \in I} f(x_i) \neq f(\sum_{i \in I} x_i)$ (and all points are non-erased)

(3) **Accept**

**Proof.** Algorithm always accepts if $f$ is linear. Suppose $f$ is $\varepsilon$-far from linear.

- **Goal:** obtain, nonerased, all values of some $k$-tuple that violates linearity.
- **Step (1):** All $x_i$ are sampled iid, so they are nonerased with high probability.
- **Step (2):**
  - Number of even-sized subsets of $[q]: 2^{q-1} = t^2/\varepsilon^2$
  - Expected number of violating sets (by structural Theorem): $\varepsilon \cdot 2^{q-1} = t^2/\varepsilon$
  - Number of even-sized sets spoiled by adversary: $t \left( q + \frac{1}{\varepsilon} \right) = 2t \log \frac{t}{\varepsilon} + \frac{t}{\varepsilon} \leq \frac{3t \log t}{\varepsilon}$
  - Expected fraction of nonerased violating even-sized sets $\geq \varepsilon/2$
  - After $O(1/\varepsilon)$ iterations, tester will sample nonerased violating sum
Q. Why not just query sums of pairs, i.e., why do we need the structural theorem?
A. To obtain optimal dependence on $t$ in the query complexity of the tester.

Algorithm. **Online-erasure-resilient linearity tester**

1. Query $q = O(t^2)$ points $x_i \sim \{0,1\}^d$

2. Repeat $1/\varepsilon$ times:
   - Sample nonempty subset $I$ of $[q]$ of size 2.
   - Query $f$ at $\sum_{i \in I} x_i$
   - Reject if $\sum_{i \in I} f(x_i) \neq f(\sum_{i \in I} x_i)$ (and all points are non-erased)

3. Accept
Plan

✓ Show the tester for linearity (with a light proof)
  • Show the lower bound for linearity
  • Show idea behind tester for quadraticity
  • Show the impossibility of testing sortedness
**Thm.** Every online-erasure-resilient linearity tester must make at least \( \log t \) queries.

**Proof.** Via Yao's minimax principle.

To show a lower bound \( q \) on randomized algorithms for testing a property it suffices to show:

- two distributions \( D^+ \) and \( D^- \) over functions \( f \)
- functions from \( D^+ \) have the property
- functions from \( D^- \) are far from the property (w.h.p.)
- a deterministic tester is given query access to \( f \) generated from \( D^+ \) or \( D^- \)
- if the tester makes \( < q \) queries, it cannot decide between \( D^+ \) and \( D^- \) with low prob. of error
Thm. Every online-erasure-resilient linearity tester must make at least $\log t$ queries.

Proof. Via Yao's minimax principle.
To show a lower bound $q$ on randomized algorithms for testing a property it suffices to show:

• two distributions $D^+$ and $D^-$ over functions $f$
• functions from $D^+$ have the property
• functions from $D^-$ are far from the property (w.h.p.)
• an erasure strategy for $t$-online-erasure oracle $\mathcal{O}$
• a deterministic tester is given query access via $\mathcal{O}$ to $f$ generated from $D^+$ or $D^-$
• if the tester makes $< q$ queries, it cannot decide between $D^+$ and $D^-$ with low prob. of error
Linearity Lower Bound

Thm. Every online-erasure-resilient linearity tester must make at least $\log t$ queries.

Proof. Via Yao's minimax principle.

- $D^+$: Uniform distribution over linear functions on $\{0,1\}^d$
- $D^-$: Uniform distribution over all Boolean functions on $\{0,1\}^d$ ($\frac{1}{4}$-far from linear w.h.p)
- Oracle $\mathcal{O}$: erase $t$ sums of previous queries of the tester (in some specific order)
- If tester makes $q < \log t$ queries, with $t$ erasures the oracle can erase $t > 2^q$ points
- i.e., oracle erases all sums of queried elements
- Tester only sees linearly independent vectors from $\{0,1\}^d$
- For a uniformly random linear function, the distribution of values over a set of linearly independent vectors is uniform
- A linear function is fully specified by its values on the basis vectors for $\{0,1\}^d$
- If tester makes $< \log t$ queries, it cannot distinguish $D^+$ from $D^-$
Plan

✓ Show the tester for linearity (with a light proof)
✓ Show the lower bound for linearity
  • Show idea behind tester for quadraticity
  • Show the impossibility of testing sortedness
## Quadraticity

Function $f: \{0,1\}^d \rightarrow \{0,1\}$ is **quadratic** if can be expressed as a polynomial of degree at most 2 e.g., $f(x) = x[1]x[2] + x[3]$

<table>
<thead>
<tr>
<th>Standard Model</th>
<th>Online Erasures Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$O\left(\frac{1}{\epsilon}\right)$ queries</td>
<td>$O\left(\frac{1}{\epsilon}\right)$ queries for constant $t$</td>
</tr>
<tr>
<td>${\text{Alon Kaufman Krivelevich Litsyn Ron '05}}$</td>
<td>Doubly exponential in $t$</td>
</tr>
<tr>
<td>${\text{Bhattacharyya Kopparty Schoenebeck Sudan Zuckerman '10}}$</td>
<td>This work</td>
</tr>
</tbody>
</table>

**Tester:**
- Sample $x_1, x_2, x_3 \sim \{0,1\}^d$
- For all nonempty $S \subseteq [3]$, query $\sum_{i \in S} x_i$
- Reject if the sum of $f$ on 7 queries is 1.

Raise of hands: Can one modify this tester to work with erasures?

Recall 2 player game.
Quadraticity

2-player game:

• Player 1 draws a vertex or edge connecting two vertices or colors a triangle in blue
• Player 2 draws an edge between existing vertices or colors a triangle in red

Raise of hands: Can one modify this tester to work with erasures?

Recall 2 player game.
Quadraticity
Quadraticity

\[ y_{1,1} \quad y_1 \quad y_{1,2} \]

\[ x_3 \quad x_1 \quad x_4 \quad x_2 \]

\[ y_{1,2} \quad y_1 \quad y_{1,2} \]

\[ x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12} \]
Quadraticity

From the game to the algorithm:
• Probability that the queries made by the tester are nonerased when queried?
• Probability that the "triangle" completed violates quadraticity?

Generalize to $t$: A strategy for Player 1 with $t^{O(t)}$ moves.
Plan

✓ Show the tester for linearity (with a light proof)
✓ Show the lower bound for linearity
✓ Show idea behind tester for quadraticity
• Show the impossibility of testing sortedness
Array $f: [n] \rightarrow \mathbb{N}$ is sorted if $f(x) \leq f(y)$ for all $x \leq y$

### Standard Model

- [Ergun Kannan Kumar Rubinfeld Viswanathan ’00]
- [Fischer Lehman Newman, Raskhodnikova Rubinfeld Alex Samorodnitsky ’04]
- [Fischer ’06]
- [Bhattacharyya Grigorescu Jung Raskhodnikova Woodruff ’12]
- [Chakrabarty Seshadhri ’18]
- [Belovs ’18]

<table>
<thead>
<tr>
<th>$\Theta(\log \epsilon n / \epsilon)$ queries</th>
<th>$O(\sqrt{n/\epsilon})$ uniform iid queries</th>
</tr>
</thead>
</table>

### Offline-Erasures Model

- [Dixit Raskhodnikova Thakurta Varma ’18]
- $O(\log n / \epsilon)$ queries

### Online-Erasures Model

- Impossible to test

---

- array is $\frac{1}{2}$-far from sorted
- all violations are disjoint
- in linearity and quadraticity, violations overlap with each other
Plan

✓ Show the tester for linearity (with a light proof)
✓ Show the lower bound for linearity
✓ Show idea behind tester for quadraticity
✓ Show the impossibility of testing sortedness
Conclusions & Open Questions

• Designed efficient testers for several important properties (linearity and quadraticity)
• Showed tight bounds for testing linearity in terms of erasure budget $t$
• Showed that some basic properties cannot be tested in our model, even for $t = 1$

• Sortedness can be tested in the offline erasures model, but not in the online erasures model.
  • Is there a property that has smaller query complexity in online model vs offline model?
• Is there a tester for testing that a function is polynomial of degree at most $k$ for $k \geq 3$?
  • In standard model this is possible with $O(2^k / \varepsilon)$ queries
• What is the query complexity for testing quadraticity in terms of $t$?
  • Current tester has doubly exponential dependence on $t$