Randomness in Computing

LECTURE 27

Last time
• Stationary distributions
• Random walks on graphs
• Algorithm for $s$-$t$-PATH

Today
• Sublinear algorithms
• Differential privacy
A Sublinear-Time Algorithm

randomized algorithm

approximate answer

Quality of approximation

Resources
- number of queries
- running time
Goal: Fundamental Understanding of Sublinear Computation

• What computational tasks?
• How to measure quality of approximation?
• What type of access to the input?
• Can we make our computations robust (e.g., to noise or erased data)?
Fundamental Computational Tasks

• **Property testing**
  • need to answer YES or NO
  ➢ intuition: only require correct answers on two sets of instances that are very different from each other

• **Learning**
  • need an approximate representation of an object
  ➢ input is from a given class (or is close to it)

• **Classical approximation**
  • need to compute a value
  ➢ output should be close to the desired value
Property Testing: Definition

[Rubinfeld Sudan, Goldreich Goldwasser Ron]

Randomized Algorithm

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

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

Property Tester

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

Far from YES

Don’t care

Reject with probability $\geq \frac{2}{3}$

$\varepsilon$-far = differs in many places ($\geq \varepsilon$ fraction of places)
Example: Lipschitz Testing [Jha R]

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

- A list of numbers is **Lipschitz** if $|x_{i+1} - x_i| \leq 1$ for all $i$.
- **Question:** Is the list Lipschitz?
  
  Requires reading entire list: $\Omega(n)$ time

- **Approximate version:** Is the list Lipschitz or $\varepsilon$-far from Lipschitz? (An $\varepsilon$ fraction of $x_i$’s have to be changed to make it Lipschitz.)

**Our result:** $O((\log n)/\varepsilon)$ time

![Graph of a list of numbers](image)
Lipschitz Testing: Attempts

1. **Test**: Pick a random $i$ and reject if $|x_{i+1} - x_i| > 1$

   Fails on:  
   
   $\begin{bmatrix} 0 & 1 & 2 & 3 & 5 & 6 & 7 & 8 \end{bmatrix}$  
   
   $\leftarrow$ 1/2-far from Lipschitz

2. **Test**: Pick random $i < j$ and reject if $|x_j - x_i| > j - i$

   Fails on:  
   
   $\begin{bmatrix} 0 & 2 & 1 & 3 & 2 & 4 & 3 & 5 & 4 & 6 \end{bmatrix}$  
   
   $\leftarrow$ 1/2-far from Lipschitz
Is a list Lipschitz or $\varepsilon$-far from Lipschitz?

Idea: Associate positions in the list with vertices of the directed line.

Construct a graph (2-spanner) 

- by adding a few “shortcut” edges $(i, j)$ for $i < j$
- where each pair of vertices is connected by a path of length at most 2

$\leq n \log n$ edges

[Bhattacharyya Grigorescu Jung R Woodruff]
Is a list Lipschitz or $\varepsilon$-far from Lipschitz?

### Test

Pick a random edge $(i, j)$ from the 2-spanner and reject if $|x_j - x_i| > j - i$.

### Analysis:

- Call a pair $(i, j)$ **violated** if $|x_j - x_i| > j - i$, and **satisfied** otherwise.
- If $i$ is an endpoint of a **violated** edge, call $x_i$ **bad**. Otherwise, call it **good**.

### Claim 1. All pairs of **good** numbers are satisfied.

**Proof:** Consider any two good numbers, $x_i$ and $x_j$.

They are connected by a path of (at most) two **satisfied** edges $(i, k), (k, j)$

1. $|x_k - x_i| \leq k - i$ and $|x_j - x_k| \leq j - k$
2. $|x_j - x_i| \leq |x_j - x_k| + |x_k - x_i| \leq (j - k) + (k - i) = j - i$
Is a list Lipschitz or \( \epsilon \)-far from Lipschitz?

**Analysis:**

- Call a pair \((i, j)\) **violated** if \(|x_j - x_i| > j - i\), and **satisfied** otherwise.
- If \(i\) is an endpoint of a **violated** edge, call \(x_i\) **bad**. Otherwise, call it **good**.

**Claim 1.** All pairs of **good** numbers are satisfied.

**Claim 2.** An \( \epsilon \)-far list **violates** \( \geq \frac{\epsilon}{2 \log n} \) fraction of edges in 2-spanner.

**Proof:** If a list is \( \epsilon \)-far from Lipschitz, it has \( \geq \epsilon n \) **bad** numbers. (Claim 1)

- Each **violated** edge contributes 2 **bad** numbers.
- 2-spanner has \( \geq \frac{\epsilon n}{2} \) **violated** edges out of \( n \log n \).

**Test**

Pick a random edge \((i, j)\) from the 2-spanner and **reject** if \(|x_j - x_i| > j - i\).
Is a list Lipschitz or $\varepsilon$-far from Lipschitz?

**Test**

<table>
<thead>
<tr>
<th>3</th>
<th>$x_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>$x_k$</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>$x_j$</td>
</tr>
</tbody>
</table>

Pick a random edge $(i, j)$ from the 2-spanner and **reject** if $|x_j - x_i| > j - i$.

**Analysis:**

- Call a pair $(i, j)$ **violated** if $|x_j - x_i| > j - i$, and **satisfied** otherwise.

**Claim 2.** An $\varepsilon$-far list **violates** $\geq \varepsilon/(2 \log n)$ fraction of edges in 2-spanner.

**Algorithm**

Sample $\frac{4 \log n}{\varepsilon}$ edges $(x_i, x_j)$ from the 2-spanner and **reject** if $|x_j - x_i| > j - i$.

**Guarantee:** All Lipschitz lists are accepted.

All lists that are $\varepsilon$-far from Lipschitz are rejected with probability $\geq 2/3$.

**Time:** $O((\log n)^2)$
We can determine if a list of $n$ numbers is Lipschitz or $\varepsilon$-far from Lipschitz in $O\left(\frac{\log n}{\varepsilon}\right)$ time.

This cannot be improved.
In polylogarithmic time, we can test a large class of properties of functions $f: \{1, \ldots, n\}^d \to \mathbb{R}$, including:

- Lipschitz property [Jha R]
- Monotonicity [Goldreich Goldwasser Lehman Ron, Dodis Goldreich Lehman R Ron Samorodnitsky]
- Bounded-derivative properties [Chakrabarty Dixit Jha Seshadhri]
- Unateness [Baleshzar Chakrabarty Pallavoor R Seshadhri]
Sublinear Algorithms: Summary

- Many problems admit sublinear-time algorithms
- Algorithms are often simple
- Analysis requires creation of interesting combinatorial, geometric and algebraic tools
- Unexpected connections to other areas
- Many open questions
Typical examples: census, medical studies, what big companies want to publish about our data…

Two conflicting goals

- Protect privacy of individuals
- **Differential privacy** [Dwork McSherry Nissim Smith 06]
- Give accurate answers
Two datasets $x, x'$ are neighbors if they differ in one person’s data.
Privacy Definition

An algorithm $A$ is $\epsilon$-differentially private if for all pairs of neighbors $x, x'$ and all sets of answers $S$:

$$\Pr[A(x) \in S] \leq e^\epsilon \Pr[A(x') \in S]$$
Properties of Differential Privacy

- **Composition:**
  If algorithms $A_1$ and $A_2$ are $\epsilon$-differentially private then algorithm that outputs $(A_1(x), A_2(x))$ is $2\epsilon$-differentially private

- Meaningful in the presence of arbitrary external information
Output Perturbation

*Frameworks for designing differentially private algorithms*
Output Perturbation

Individuals

\[ x = x_1, x_2, x_3, \ldots, x_{d-1}, x_d \]

Curator

Evaluate \( f(x) \)

\[ A(x) = f(x) + \text{noise} \]

Data Analysts
Global sensitivity of a function $f$ is

$$GS_f = \max_{\text{neighbors } x, x'} |f(x) - f(x')|.$$ 

Example: $x_1, \ldots, x_n \in [0,1]$, $\text{ave}(x) = \frac{x_1 + \cdots + x_n}{n}$

- $GS_{\text{ave}} = ?$
Global sensitivity of a function $f$ is

$$GS_f = \max_{\text{neighbors } x, x'} |f(x) - f(x')|.$$ 

**Example:** $x_1, ..., x_n \in [0,1]$, $\text{ave}(x) = \frac{x_1 + \cdots + x_n}{n}$

- $GS_{\text{ave}} = 1/n$

**Theorem** [Dwork McSherry Nissim Smith]

If $A(x) = f(x) + \text{Lap} \left( \frac{GS_f}{\epsilon} \right)$ then $A$ is $\epsilon$-differentially private.
Laplace Mechanism Theorem [Dwork McSherry Nissim Smith]

If \( A(x) = f(x) + \text{Lap} \left( \frac{GS_f}{\epsilon} \right) \) then \( A \) is \( \epsilon \)-differentially private.

Laplace distribution \( \text{Lap}(\lambda) \) has density \( h(y) = \frac{1}{2\lambda} \cdot e^{-\frac{|y|}{\lambda}} \)

(mean 0, standard deviation \( \sqrt{2} \cdot \lambda \))

Sliding Property of \( \text{Lap} \left( \frac{GS_f}{\epsilon} \right) \)

for all \( y, \delta: \frac{h(y)}{h(y+\delta)} \leq e^{\epsilon \cdot \frac{|\delta|}{GS_f}} \)
When is Laplace Mechanism Useful?

- Laplace mechanism is always private.
- When is it accurate?

Example: $x_1, \ldots, x_n \in [0,1]$, $\text{ave}(x) = \frac{x_1 + \cdots + x_n}{n}$

- $G_{\text{ave}} = 1/n$  \quad \text{Noise} = \text{Lap}(\frac{1}{\epsilon n})$

Accurate when GS is low

(and $n$, the size of the database, is sufficiently large)
Can Global Sensitivity Be Too High?

Example: \(x_1, \ldots, x_n \in [0,1]\), median\((x)\) is median of \(x_1, \ldots, x_n\).

- \(GS_{\text{median}} = ?\)

\[
\begin{align*}
    x &= \underbrace{0 \ldots 0}_\frac{n-1}{2} \underbrace{0 1 \ldots 1}_\frac{n-1}{2} \\
    x' &= \underbrace{0 \ldots 0}_\frac{n-1}{2} \underbrace{1 1 \ldots 1}_\frac{n-1}{2}
\end{align*}
\]

\[\text{median}(x) = 0 \quad \text{median}(x') = 1\]

- Noise: \(\text{Lap}\left(\frac{1}{\epsilon}\right)\) \(Too much noise!\)

- But for most neighboring datasets \(x\) and \(x'\),
  \[\text{median}(x) - \text{median}(x')\] is small

- Can we add less noise on "good" datasets?
Smooth Sensitivity Framework

[Nissim Raskhodnikova Smith]
Local Sensitivity

Local sensitivity of a function $f$ at point $x$ is

$$LS_f(x) = \max_{x': \text{neighbor of } x} |f(x) - f(x')|.$$  

Relationship to GS: $GS_f = \max_{\text{datasets } x} LS_f(x)$

Example: median of $0 \leq x_1 \leq \cdots \leq x_n \leq 1$ for odd $n$.

- $LS_{\text{median}}(x) = ?$

- New median when $x' = 0$
- New median when $x' = 1$

0 $x_1$ ... $x_{m-1}$ $x_m$ $x_{m+1}$ ... $x_n$ 1
**Local Sensitivity**

**Local Sensitivity** of a function \( f \) at point \( x \) is

\[
LS_f(x) = \max_{x': \text{neighbor of } x} |f(x) - f(x')|.
\]

**Relationship to GS:**

\[
GS_f = \max_{\text{datasets } x} LS_f(x)
\]

**Example:** median of \( 0 \leq x_1 \leq \cdots \leq x_n \leq 1 \) for odd \( n \).

- \( LS_{\text{median}} (x) = \max(x_{m+1} - x_m, x_m - x_{m-1}) \)

**Goal:** Release \( f(x) \) with less noise when \( LS_f(x) \) is lower.
First Attempt: Local Sensitivity

Noise with magnitude proportional to $LS_f(x)$ instead of $GS_f$?

Problem: noise magnitude might reveal information.

Example: median

$x = \underbrace{0 \ldots 0}_{\frac{n-3}{2}} \underbrace{000}_{\frac{n-3}{2}} \underbrace{1 \ldots 1}_{\frac{n-3}{2}}$

$median(x) = 0$

$LS_{median}(x) = 0$

$Pr[A(x) = 0] = 1$

$x' = \underbrace{0 \ldots 0}_{\frac{n-3}{2}} \underbrace{001}_{\frac{n-3}{2}} \underbrace{1 \ldots 1}_{\frac{n-3}{2}}$

$median(x') = 0$

$LS_{median}(x') = 1$

$Pr[A(x') = 0] = 0$

$A$ is not differentially private

• Idea: make noise magnitude an "insensitive" function
Smooth Bounds on Local Sensitivity

Design sensitivity function $S(x)$

- $S(x)$ is an $\epsilon$-smooth upper bound on $LS_f(x)$ if
  - for all $x$: $S(x) \geq LS_f(x)$
  - for all neighbors $x, x'$: $S(x) \leq e^\epsilon S(x')$

**Theorem**

If $A(x) = f(x) + noise \left(\frac{S(x)}{\epsilon}\right)$ then $A$ is $(\epsilon', \delta)$-diff. private.

**Example:** $GS_f$ is a smooth bound on $LS_f(x)$.
**Smooth Sensitivity**

- For two datasets $x$ and $y$, let $\text{dist}(x, y) = |\{i : x_i \neq y_i\}|$
- **Smooth sensitivity** $S_f^*(x) = \max_{\text{datasets } y} LS_f(y) \cdot e^{-\epsilon \cdot \text{dist}(x, y)}$.
- **Intuition**: $S_f^*(x)$ is low when $x$ is far from sensitive datasets

**Lemma**

1. Smooth sensitivity is an $\epsilon$-smooth upper bound on $LS_f$.
2. For every $\epsilon$-smooth upper bound $S$ on $LS_f$:
   $$S_f^*(x) \leq S(x) \text{ for all } x.$$
Computing Smooth Sensitivity

Recall: Smooth sensitivity
\[ S_f^*(x) = \max_y LS_f(y) \cdot e^{-\varepsilon \cdot \text{dist}(x,y)}. \]

**Observation**

\[ S_f^*(x) = \max_{k=0,1,\ldots,n} LS_f^k(x) \cdot e^{-\varepsilon \cdot k}, \]

where \( LS_f^k(x) = \max_{y: \text{dist}(x,y) \leq k} LS_f(y). \)

**Example:** median

\[ LS^k_{\text{median}}(x) = \max_{t=0,1,\ldots,k+1} (x_{m+t+k+1} - x_{m+t}) \]

This gives \( O(n^2) \) time algorithm for computing \( S^*_{\text{median}}(x). \)

(It can be computed in time \( O(n \log n). \) )
Conclusion: Calibrating Noise

• Adding noise proportional to local sensitivity is not safe.
• Smooth sensitivity framework allows one to calibrate noise to the input dataset.
  – Requires understanding combinatorial structure of the problem.
• There are other frameworks based on local sensitivity:
  – Propose-Test-Release [Dwork Lei, Karwa R Smith Yaroslavtsev]
  – Sample-and-Aggregate [Nissim R Smith]
Conclusion: Differential Privacy

- a rigorous and widely applicable notion of privacy
- is defined in terms of algorithm
- requires the algorithm to be randomized
- puts a restriction on the algorithm, requiring that output distributions on neighboring datasets be close
- is used in 2020 Census, by Apple and Google