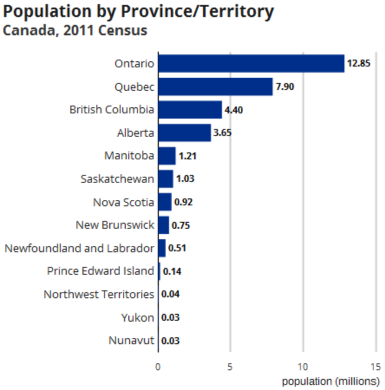
Communication–Efficient Distributed Learning of Discrete Probability Distributions

Krzysztof Onak
IBM T.J. Watson Research Center

Joint work with **Ilias Diakonikolas**, **Elena Grigorescu**, **Jerry Li**, **Abhiram Natarajan**, and **Ludwig Schmidt**.

Discrete Distributions

- Widespread in practice

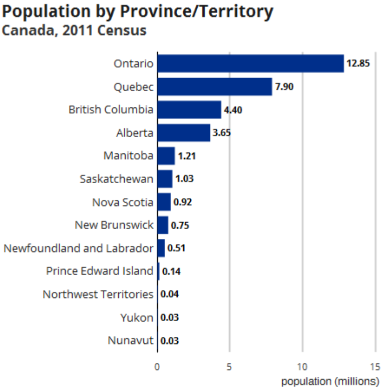


(chart by Srm038, CC BY-SA 4.0)

Discrete Distributions

- Widespread in practice

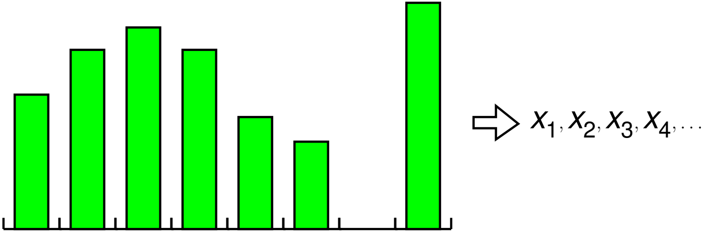
- Sample tasks:
  - Learn the distribution
  - Test a property
  - Estimate a parameter



(chart by Srm038, CC BY-SA 4.0)

Learning Discrete Distributions

$\mathcal{D}$ = probability distribution on $\{1, \ldots, n\}$
Input: Independent samples from $\mathcal{D}$

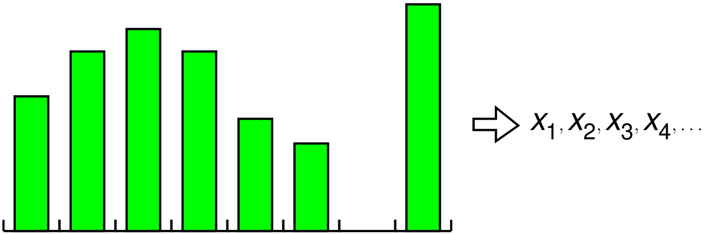


Goal:

Output a distribution $\mathcal{D}'$ such that $\|\mathcal{D} - \mathcal{D}'\|_1 < \epsilon$

Learning Discrete Distributions

$\mathcal{D}$ = probability distribution on $\{1, \ldots, n\}$
Input: Independent samples from $\mathcal{D}$



Goal:

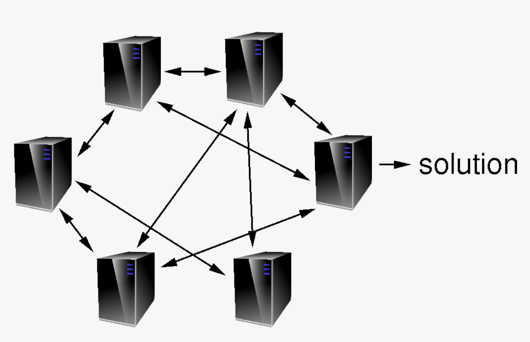
Output a distribution $\mathcal{D}'$ such that $\|\mathcal{D} - \mathcal{D}'\|_1 < \epsilon$

Sample complexity: $\Theta(n/\epsilon^2)$

Communication Complexity

Distributed data: samples held by different players

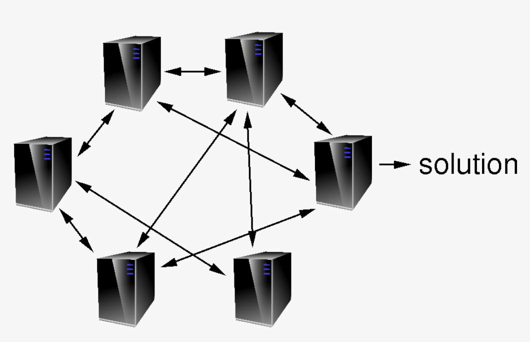
Example: Samples in different data centers



Communication Complexity

Distributed data: samples held by different players

Example: Samples in different data centers



How much do players have to communicate to solve the problem?

Is sublinear communication possible?

Sample Results

Unstructured distributions under ℓ_1-error ϵ:

- Upper bounds:
  - $\log n$ bits to communicate samples
    $\Rightarrow O((n/\epsilon^2)\log n)$ bits suffice

Sample Results

Unstructured distributions under ℓ_1-error ϵ:

- Upper bounds:
  - $\log n$ bits to communicate samples
    $\Rightarrow O((n/\epsilon^2) \log n)$ bits suffice
  - better upper bounds by compressing data
  - more samples per player $\Rightarrow$ less communication

Sample Results

Unstructured distributions under ℓ_1-error ϵ:

- Upper bounds:
  - $\log n$ bits to communicate samples
    $\Rightarrow O((n/\epsilon^2)\log n)$ bits suffice
  - better upper bounds by compressing data
  - more samples per player $\Rightarrow$ less communication

- Lower bounds:
  - $\Omega(n \cdot \log(1/\epsilon))$ always needed

Sample Results

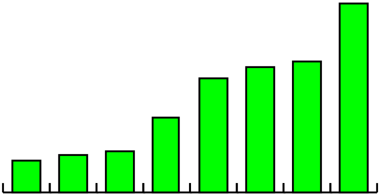
Unstructured distributions under ℓ_1-error ϵ:

- Upper bounds:
  - $\log n$ bits to communicate samples
    $\Rightarrow O((n/\epsilon^2) \log n)$ bits suffice
  - better upper bounds by compressing data
  - more samples per player $\Rightarrow$ less communication

- Lower bounds:
  - $\Omega(n \cdot \log(1/\epsilon))$ always needed
  - One sample per player: $\Omega((n/\epsilon^2) \cdot \log n)$
    (Later in the talk: sketch of less general result)

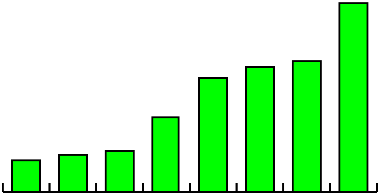
Structured distributions

- Monotone

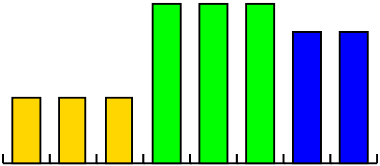


Structured distributions

- Monotone



- k-histograms



Results for Structured Distributions

Monotone distributions:

- Some unstructured upper and lower bounds translate to this setting
- How: use ideas of Birge (1987)
- distribution can be approximated with $O(\epsilon^{-1} \log n)$ uniform buckets

Results for Structured Distributions

Monotone distributions:

- Some unstructured upper and lower bounds translate to this setting
- How: use ideas of Birge (1987)
- distribution can be approximated with $O(\epsilon^{-1} \log n)$ uniform buckets

Upper bounds for k-histograms:

- Main challenge: unknown break points

Results for Structured Distributions

Monotone distributions:

- Some unstructured upper and lower bounds translate to this setting
- How: use ideas of Birge (1987)
- distribution can be approximated with $O(\epsilon^{-1} \log n)$ uniform buckets

Upper bounds for k-histograms:

- Main challenge: unknown break points
- For ℓ_1-error, reuse ideas of Acharya, Diakonikolas, Li, and Schmidt (2017)
- For ℓ_2-error, top-down strategy of partitioning the range

Results for Structured Distributions

Monotone distributions:

- Some unstructured upper and lower bounds translate to this setting
- How: use ideas of Birge (1987)
- distribution can be approximated with $O(\epsilon^{-1} \log n)$ uniform buckets

Upper bounds for k-histograms:

- Main challenge: unknown break points
- For ℓ_1-error, reuse ideas of Acharya, Diakonikolas, Li, and Schmidt (2017)
- For ℓ_2-error, top-down strategy of partitioning the range
- The algorithms are agnostic: good approximation even if input distribution not exactly a k-histogram

Related Work

A lot of recent interest in communication-efficient learning:

DAW12, ZDW13, ZX15, GMN14, KVW14, LBKW14, SSZ14, DJWZ14, LSLT15, BGMNW15

- Both upper and lower bounds.
- Usually more continuous problems.
- Sample problem: estimating the mean of a Gaussian distribution.

Outline

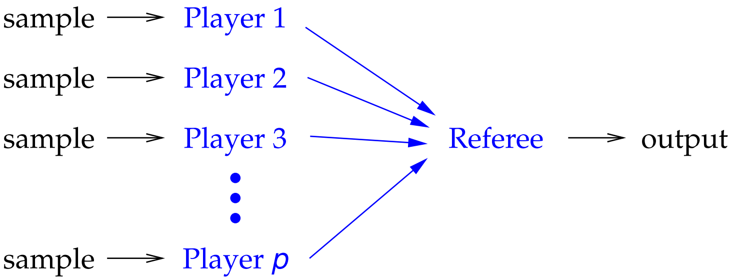
1. Toy Example Presented Today
2. Warm-Up: Single Coin
3. $O(n/\epsilon^2)$ Sample Complexity Review
4. Communication Complexity Lower Bound

Outline

1. Toy Example Presented Today
2. Warm-Up: Single Coin
3. $O(n/\epsilon^2)$ Sample Complexity Review
4. Communication Complexity Lower Bound

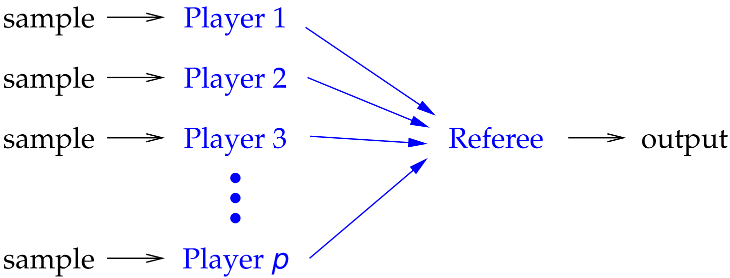
Simultaneous Communication Complexity

- Each player has one sample and sends a single message to a referee
- The referee outputs solution



Simultaneous Communication Complexity

- Each player has one sample
  and sends a single message to a referee
- The referee outputs solution



- Each sample is $\Theta(\log n)$ bits
- Can average communication be made $o(\log n)$?

Outline

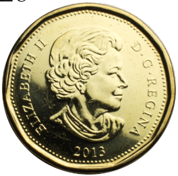
1. Toy Example Presented Today
2. Warm-Up: Single Coin
3. $O(n/\epsilon^2)$ Sample Complexity Review
4. Communication Complexity Lower Bound

Outline

Bias of a Single Coin

Input: Independent coin tosses

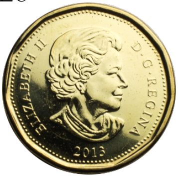
Goal: Estimate the probability of heads up to $\pm\epsilon$
using as few coin tosses as possible



Bias of a Single Coin

Input: Independent coin tosses

Goal: Estimate the probability of heads up to $\pm\epsilon$
using as few coin tosses as possible



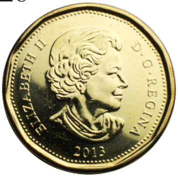
Caveat:

- Can't ever be completely sure
- Happy to answer correctly with probability 90%

Bias of a Single Coin

Input: Independent coin tosses

Goal: Estimate the probability of heads up to $\pm\epsilon$
using as few coin tosses as possible



Caveat:

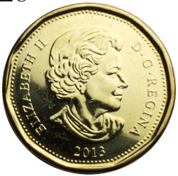
- Can't ever be completely sure
- Happy to answer correctly with probability 90%

Upper bound: $O(1/\epsilon^2)$ via Hoeffding's inequality

Bias of a Single Coin

Input: Independent coin tosses

Goal: Estimate the probability of heads up to $\pm\epsilon$
using as few coin tosses as possible



Caveat:

- Can't ever be completely sure
- Happy to answer correctly with probability 90%

Upper bound: $O(1/\epsilon^2)$ via Hoeffding's inequality

Is this bound optimal?

Hard Instance

Difficult to distinguish:

heads: $\frac{1}{2} - 2\epsilon$ tails: $\frac{1}{2} + 2\epsilon$

vs.

heads: $\frac{1}{2} + 2\epsilon$ tails: $\frac{1}{2} - 2\epsilon$

Hard Instance

Difficult to distinguish:

heads: $\frac{1}{2} - 2\epsilon$ tails: $\frac{1}{2} + 2\epsilon$

vs.

heads: $\frac{1}{2} + 2\epsilon$ tails: $\frac{1}{2} - 2\epsilon$

More formally:

$$\text{probability of heads} = \frac{1}{2} + \delta \cdot 2\epsilon$$

where δ selected uniformly at random from $\{-1, +1\}$

Information Approach

Single coin toss: $X \in \{\text{heads}, \text{tails}\}$

Mutual information: $I(X; \delta) = H(X) - H(X|\delta) = O(\epsilon^2)$

Information Approach

Single coin toss: $X \in \{\text{heads}, \text{tails}\}$

Mutual information: $I(X; \delta) = H(X) - H(X|\delta) = O(\epsilon^2)$

k coin tosses: $X_1, X_2, \ldots, X_k$

$$\sum I(X_i; \delta) = O(\epsilon^2 k)$$

Information Approach

Single coin toss: $X \in \{\text{heads}, \text{tails}\}$

Mutual information: $I(X; \delta) = H(X) - H(X|\delta) = O(\epsilon^2)$

k coin tosses: $X_1, X_2, \ldots, X_k$

$$\sum I(X_i; \delta) = O(\epsilon^2 k)$$

- Is it true that $I(X_1 \ldots X_k; \delta) \le \sum I(X_i; \delta)$?

Information Approach

Single coin toss: $X \in \{\text{heads}, \text{tails}\}$

Mutual information: $I(X;\delta) = H(X) - H(X|\delta) = O(\epsilon^2)$

k coin tosses: $X_1, X_2, \ldots, X_k$

$$\sum I(X_i;\delta) = O(\epsilon^2 k)$$

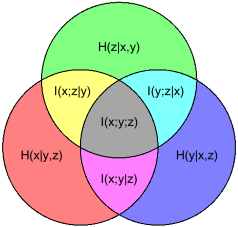
- Is it true that $I(X_1 \ldots X_k;\delta) \leq \sum I(X_i;\delta)$?
- If so and $k = o(1/\epsilon^2)$:
  - $H(\delta|X_1 \ldots X_k) = H(\delta) - I(X_1 \ldots X_k;\delta) = 1 - o(1)$
  - Value of δ distributed almost uniformly on $\{-1, +1\}$
  - Can predict δ given $X_1 \ldots X_k$ with probability only $\frac{1}{2} + o(1)$

Multivariate Mutual Information

(Focus on $k = 2$, larger k by induction)

$$I(X_1 X_2; \delta) = I(X_1; \delta) + I(X_2; \delta) - I(X_1; X_2; \delta)$$

$$\text{where } I(X_1; X_2; \delta) = I(X_1; X_2) - I(X_1; X_2|\delta)$$

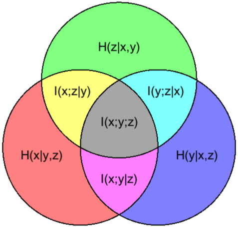


Multivariate Mutual Information

(Focus on $k = 2$, larger k by induction)

$$I(X_1 X_2; \delta) = I(X_1; \delta) + I(X_2; \delta) - I(X_1; X_2; \delta)$$

$$\text{where } I(X_1; X_2; \delta) = I(X_1; X_2) - I(X_1; X_2 | \delta)$$



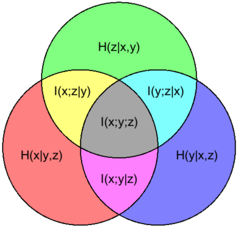
(In general, $I(x; y; z)$ can be negative. Example: $x \oplus y = z$.)

Multivariate Mutual Information

(Focus on $k = 2$, larger k by induction)

$$I(X_1 X_2; \delta) = I(X_1; \delta) + I(X_2; \delta) - I(X_1; X_2; \delta)$$

$$\text{where } I(X_1; X_2; \delta) = I(X_1; X_2) - I(X_1; X_2 | \delta)$$



(In general, $I(x; y; z)$ can be negative. Example: $x \oplus y = z$.)

- $I(X_1; X_2 | \delta) = 0$
- Hence, $I(X_1; X_2; \delta) \geq 0$.
- This proves that $I(X_1 X_2; \delta) \leq I(X_1; \delta) + I(X_2; \delta)$.

Outline

1. Toy Example Presented Today
2. Warm-Up: Single Coin
3. $O(n/\epsilon^2)$ Sample Complexity Review
4. Communication Complexity Lower Bound

Outline

Upper Bound Review

Solution: $\mathcal{D}' =$ empirical distribution of $O(n/\epsilon^2)$ samples

Upper Bound Review

Solution: $\mathcal{D}' =$ empirical distribution of $O(n/\epsilon^2)$ samples

Why this works:

- For every subset of $\{1, \ldots, n\}$ the probabilities under $\mathcal{D}$ and $\mathcal{D}'$ within $\epsilon/2$ with probability $1 - 2^{-2n}$ (via Hoeffding's inequality)

Upper Bound Review

Solution: $\mathcal{D}' =$ empirical distribution of $O(n/\epsilon^2)$ samples

Why this works:

- For every subset of $\{1, \ldots, n\}$ the probabilities under $\mathcal{D}$ and $\mathcal{D}'$ within $\epsilon/2$ with probability $1 - 2^{-2n}$ (via Hoeffding's inequality)
- Union bound: $\leq \epsilon/2$ difference for all subsets with probability $1 - o(1)$

Upper Bound Review

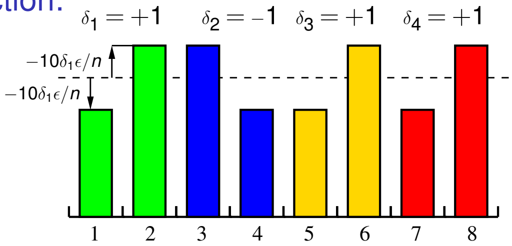
Solution: $\mathcal{D}' =$ empirical distribution of $O(n/\epsilon^2)$ samples

Why this works:

- For every subset of $\{1, \ldots, n\}$ the probabilities under $\mathcal{D}$ and $\mathcal{D}'$ within $\epsilon/2$ with probability $1 - 2^{-2n}$ (via Hoeffding's inequality)
- Union bound: $\leq \epsilon/2$ difference for all subsets with probability $1 - o(1)$
- Equivalent to $\|\mathcal{D} - \mathcal{D}'\|_1 \leq \epsilon$ with probability $1 - o(1)$)

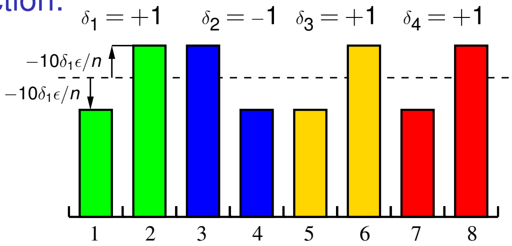
Lower Bound Review

Construction:



Lower Bound Review

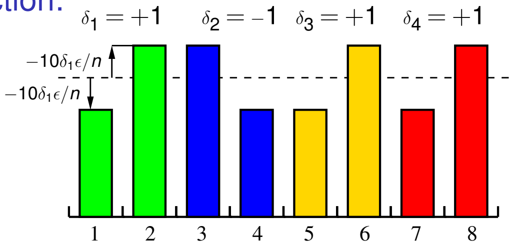
Construction:



- Each pair randomly biased by 10ϵ

Lower Bound Review

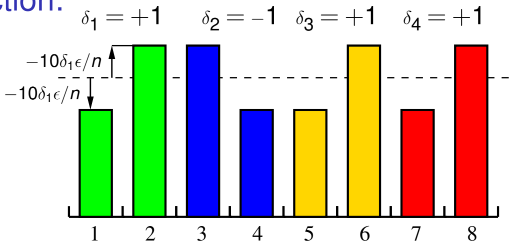
Construction:



- Each pair randomly biased by 10ϵ
- Need to predict bias of more than $\frac{9}{10}$ pairs
  (via averaging/Markov's bound)

Lower Bound Review

Construction:



- Each pair randomly biased by 10ϵ
- Need to predict bias of more than $\frac{9}{10}$ pairs
  (via averaging/Markov's bound)
- This requires $\Omega(n/\epsilon^2)$ samples

Outline

1. Toy Example Presented Today
2. Warm-Up: Single Coin
3. $O(n/\epsilon^2)$ Sample Complexity Review
4. Communication Complexity Lower Bound

Our Claim

No protocol with $o\left(\frac{n}{\epsilon^2}\log n\right)$
communication on average
that succeeds learning the distribution
with probability $99/100$.

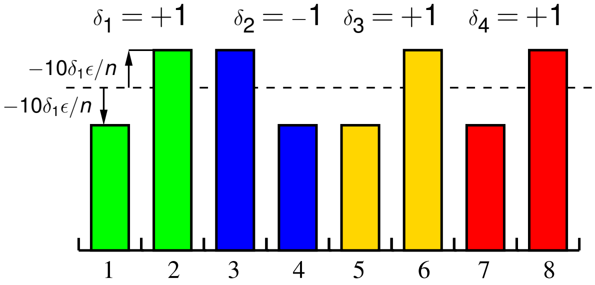
Our Claim

No protocol with $o\left(\frac{n}{\epsilon^2}\log n\right)$
communication on average
that succeeds learning the distribution
with probability $99/100$.

(Can assume at most $O\left(n/\epsilon^2 \log n\right)$ players in the proof)

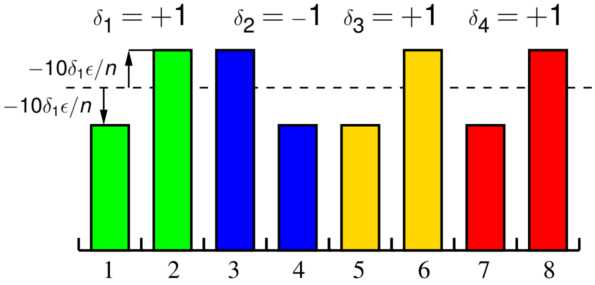
Hard Distribution

Reuse the hard distribution for sampling:



Hard Distribution

Reuse the hard distribution for sampling:



Can assume the protocol is deterministic:

- Slight loss in the probability of success
- Expected communication goes up by constant factor

The Proof Plan

- Assume $o(n\epsilon^{-2} \log n)$ communication protocol

The Proof Plan

- Assume $o(n\epsilon^{-2} \log n)$ communication protocol
- For random i, show that:
  - Messages reveal very little about δ_i
    (even if the referee knows all other δ_i's)
  - The referee can predict δ_i with probability $\frac{1}{2} + o(1)$

The Proof Plan

- Assume $o(n\epsilon^{-2} \log n)$ communication protocol
- For random i, show that:
    - Messages reveal very little about δ_i (even if the referee knows all other δ_i's)
    - The referee can predict δ_i with probability $\frac{1}{2} + o(1)$
- The original protocol correct only on $\frac{1}{2} + o(1)$ fraction of δ_i's most of the time

The Proof Plan

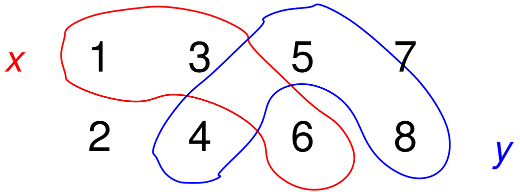
- Assume $o(n\epsilon^{-2} \log n)$ communication protocol
- For random i, show that:
  - Messages reveal very little about δ_i (even if the referee knows all other δ_i's)
  - The referee can predict δ_i with probability $\frac{1}{2} + o(1)$
- The original protocol correct only on $\frac{1}{2} + o(1)$ fraction of δ_i's most of the time

CONTRADICTION!!!

Messages of Single Player

Modify protocol for each pair $2j-1$ and $2j$:

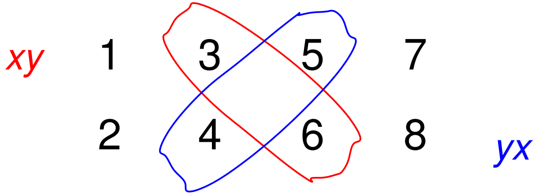
- Before: x sent for $2j-1$ and y sent for $2j$
- After: send xy for $2j-1$ and yx for $2j$



Messages of Single Player

Modify protocol for each pair $2j-1$ and $2j$:

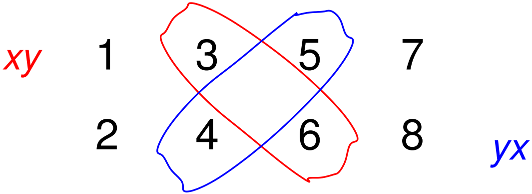
- Before: x sent for $2j-1$ and y sent for $2j$
- After: send xy for $2j-1$ and yx for $2j$



Messages of Single Player

Modify protocol for each pair $2j-1$ and $2j$:

- Before: x sent for $2j-1$ and y sent for $2j$
- After: send xy for $2j-1$ and yx for $2j$



Result:

- Communication complexity only doubles.
- This partitions pairs. Each message reveals bias on a specific subset of pairs.

Messages of Single Player

Three cases for a pair $2i - 1$ and $2i$
and corresponding messages xy and yx:

Messages of Single Player

Three cases for a pair $2i - 1$ and $2i$
and corresponding messages xy and yx:

1. $|xy| > \frac{\log n}{100}$
   - Happens for $o(n/\epsilon^2)$ fraction of players
   - Can assume the message reveals the sample
   - $I(\text{message}; \delta_i) \leq I(\text{sample}; \delta_i) = O(\epsilon^2/n)$

Messages of Single Player

Three cases for a pair $2i - 1$ and $2i$
and corresponding messages xy and yx:

1. $|xy| > \frac{\log n}{100}$
2. $|xy| \leq \frac{\log n}{100}$ & $\leq \sqrt{n}$ pairs with these messages
   - Random i: happens with probability $\frac{n^{0.01} \cdot \sqrt{n}}{n}$
   - Can assume the message reveals the sample
   - $I(\text{message}; \delta_i) \leq I(\text{sample}; \delta_i) = O(\epsilon^2/n)$

Messages of Single Player

Three cases for a pair $2i-1$ and $2i$
and corresponding messages xy and yx:

1. $|xy| > \frac{\log n}{100}$
2. $|xy| \leq \frac{\log n}{100}$ & $\leq \sqrt{n}$ pairs with these messages
3. $|xy| \leq \frac{\log n}{100}$ & $> \sqrt{n}$ pairs with these messages
    - Can happen always
    - δ_i has little impact on probabilities of xy and yx
    - $I(\text{sample}; \delta_i) = O(\epsilon^2/(n \cdot \#\text{pairs})) = O(\epsilon^2/n^{1.5})$

Total Information about δ_i

M_j = message of the j-th player $\quad M = (M_1, M_2, \ldots, M_p)$

Total Information about δ_i

$M_j =$ message of the j-th player $\qquad M = (M_1, M_2, \ldots, M_p)$

For all but $o(1)$ fraction of i's:

$$\sum_j I(\delta_i; M_j) = o\left(\frac{n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n^{0.52}}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n \log n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n^{1.5}}\right) = o(1)$$

Total Information about δ_i

$M_j =$ message of the j-th player $\qquad M = (M_1, M_2, \ldots, M_p)$

For all but $o(1)$ fraction of i's:

$$\sum_j I(\delta_i; M_j) = o\left(\frac{n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n^{0.52}}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n \log n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n^{1.5}}\right) = o(1)$$

Then $I(\delta_i; M) = o(1)$:

- Messages M_j independent once δ_i is fixed
- This implies that $I(\delta_i; M) \leq \sum_j I(\delta_i, M_j)$

Total Information about δ_i

M_j = message of the j-th player $\qquad M = (M_1, M_2, \ldots, M_p)$

For all but $o(1)$ fraction of i's:

$$\sum_j I(\delta_i; M_j) = o\left(\frac{n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n^{0.52}}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n \log n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n^{1.5}}\right) = o(1)$$

Then $I(\delta_i; M) = o(1)$:

- Messages M_j independent once δ_i is fixed
- This implies that $I(\delta_i; M) \leq \sum_j I(\delta_i, M_j)$

And $H(\delta_i | M) = H(\delta_i) - I(\delta_i; M) = 1 - o(1)$

Total Information about δ_i

M_j = message of the j-th player $\quad M = (M_1, M_2, \ldots, M_p)$

For all but $o(1)$ fraction of i's:

$$\sum_j I(\delta_i; M_j) = o\left(\frac{n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n^{0.52}}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n}\right) + O\left(\frac{n \log n}{\epsilon^2}\right) \cdot O\left(\frac{\epsilon^2}{n^{1.5}}\right) = o(1)$$

Then $I(\delta_i; M) = o(1)$:

- Messages M_j independent once δ_i is fixed
- This implies that $I(\delta_i; M) \leq \sum_j I(\delta_i, M_j)$

And $H(\delta_i | M) = H(\delta_i) - I(\delta_i; M) = 1 - o(1)$

Algorithm correct with probability $\frac{1}{2} + o(1)$

Long term goals:

- Reinterpret known distribution testing and learning results in this framework
- Design non-trivial protocols with sublinear amount of communication

Long term goals:

- Reinterpret known distribution testing and learning results in this framework
- Design non-trivial protocols with sublinear amount of communication

Questions?