WHEN FOURIER SIIRVS: FOURIER-BASED TESTING FOR FAMILIES OF DISTRIBUTIONS

Clément Canonne[1], Ilias Diakonikolas,[2] and Alistair Stewart[2]

March 19, 2018

Stanford University[1] and University of Southern California[2]

BACKGROUND, CONTEXT, AND MOTIVATION

Sublinear-time,

Sublinear-time, approximate,

Sublinear-time, approximate, randomized

PROPERTY TESTING

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

- Big Dataset: too big

PROPERTY TESTING

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

- Big Dataset: too big
- Expensive access: pricey data

PROPERTY TESTING

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

- Big Dataset: too big
- Expensive access: pricey data
- “Model selection”: many options

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

- Big Dataset: too big
- Expensive access: pricey data
- “Model selection”: many options
- Good Enough: a priori knowledge

PROPERTY TESTING

Sublinear-time, approximate, randomized decision algorithms that make local queries to their input.

- Big Dataset: too big
- Expensive access: pricey data
- “Model selection”: many options
- Good Enough: a priori knowledge

Need to infer information – one bit – from the data: quickly, or with very few lookups.

Figure: Property Testing: Inside the yolk, or outside the egg.

Introduced by [RS96, GGR98] – has been a very active area since.

- Known space (e.g., $\{0,1\}^N$)
- Property $\mathcal{P} \subseteq \{0,1\}^N$
- Oracle access to unknown $x \in \{0,1\}^N$
- Proximity parameter $\varepsilon \in (0,1]$

Must decide

$$x \in \mathcal{P} \quad \text{vs.} \quad \mathrm{dist}(x,\mathcal{P}) > \varepsilon$$

(has the property, or is ε-far from it)

Many variants, subareas, with a plethora of results (see e.g. [Ron08, Ron10, Gol10, Gol17, BY17]).

Now, our "big object" is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

Now, our "big object" is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

- instead of queries: samples

DISTRIBUTION TESTING

Now, our "big object" is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

- instead of queries: samples
- instead of Hamming distance: total variation

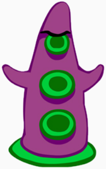
DISTRIBUTION TESTING

Now, our "big object" is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

- instead of queries: samples
- instead of Hamming distance: total variation
- instead of functions/graphs/strings: distributions

Now, our “big object” is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

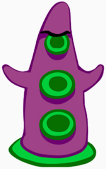
- instead of queries: samples
- instead of Hamming distance: total variation
- instead of functions/graphs/strings: distributions



DISTRIBUTION TESTING

Now, our “big object” is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

- instead of queries: samples
- instead of Hamming distance: total variation
- instead of functions/graphs/strings: distributions

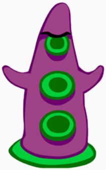


Focus on the sample complexity, with efficiency as ancillary (yet important) goal.

DISTRIBUTION TESTING

Now, our "big object" is a probability distribution over a (discrete) domain Ω (here,* $|\Omega| = n$).

- instead of queries: samples
- instead of Hamming distance: total variation
- instead of functions/graphs/strings: distributions



Focus on the sample complexity, with efficiency as ancillary (yet important) goal.

BACKGROUND

Over the past 15 years, many results on many properties:

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR+00, Pan08, DGPP16]
- Identity [BFF+01, VV17, BCG17]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR+00, Pan08, DGPP16]
- Identity [BFF+01, VV17, BCG17]
- Equivalence [BFR+00, Val11, CDVV14]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]
- Identity [BFF$^+$01, VV17, BCG17]
- Equivalence [BFR$^+$00, Val11, CDVV14]
- Independence [BFF$^+$01, LRR13, DK16, ADK15]

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]
- Identity [BFF$^+$01, VV17, BCG17]
- Equivalence [BFR$^+$00, Val11, CDVV14]
- Independence [BFF$^+$01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]
- Identity [BFF$^+$01, VV17, BCG17]
- Equivalence [BFR$^+$00, Val11, CDVV14]
- Independence [BFF$^+$01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]
- Poisson Binomial Distributions [AD15, CDGR16]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]
- Identity [BFF$^+$01, VV17, BCG17]
- Equivalence [BFR$^+$00, Val11, CDVV14]
- Independence [BFF$^+$01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]
- Poisson Binomial Distributions [AD15, CDGR16]
- Log-concavity [CDGR16, ADK15]

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR+00, Pan08, DGPP16]
- Identity [BFF+01, VV17, BCG17]
- Equivalence [BFR+00, Val11, CDVV14]
- Independence [BFF+01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]
- Poisson Binomial Distributions [AD15, CDGR16]
- Log-concavity [CDGR16, ADK15]
- and more... [Rub12, Can15]

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR$^+$00, Pan08, DGPP16]
- Identity [BFF$^+$01, VV17, BCG17]
- Equivalence [BFR$^+$00, Val11, CDVV14]
- Independence [BFF$^+$01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]
- Poisson Binomial Distributions [AD15, CDGR16]
- Log-concavity [CDGR16, ADK15]
- and more... [Rub12, Can15]

Much has been done;

BACKGROUND

Over the past 15 years, many results on many properties:

- Uniformity [GR00, BFR+00, Pan08, DGPP16]
- Identity [BFF+01, VV17, BCG17]
- Equivalence [BFR+00, Val11, CDVV14]
- Independence [BFF+01, LRR13, DK16, ADK15]
- Monotonicity [BKR04, BFRV11, CDGR16, ADK15]
- Poisson Binomial Distributions [AD15, CDGR16]
- Log-concavity [CDGR16, ADK15]
- and more... [Rub12, Can15]

Much has been done; and yet...

Techniques

Most algorithms and results are somewhat ad hoc, and property-specific.

Techniques

Most algorithms and results are somewhat ad hoc, and property-specific.

Can we...

design general algorithms and approaches that apply to many testing problems at once?

General Trend

In learning: [CDSS13, CDSS14, CDSX14, ADLS17]

General Trend

In learning: [CDSS13, CDSS14, CDSX14, ADLS17]

and recently...

In testing: [Val11, VV11, CDGR16, ADK15, DK16, BCG17]

OUTLINE OF THE TALK

OUTLINE OF THE TALK

- Notation, Preliminaries
- Overall Goal, Restated
    - The shape restrictions approach [CDGR16]
    - The Fourier approach [CDS17]

SOME NOTATION

GLOSSARY

- Probability distributions over $[n] := \{1, \ldots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \ : \ \sum_{i=1}^{n} p(i) = 1 \right\}$$

GLOSSARY

- Probability distributions over $[n] := \{1, \ldots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Probability distributions over $[n] := \{1, \dots, n\}$

$$\Delta([n]) = \left\{ p \colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Total variation distance (statistical distance, ℓ_1 distance):

$$d_{\mathrm{TV}}(p,q) = \sup_{S \subseteq \Omega}(p(S) - q(S)) = \frac{1}{2}\sum_{x \in \Omega} |p(x) - q(x)| \in [0,1]$$

GLOSSARY

- Probability distributions over $[n] := \{1, \dots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Total variation distance (statistical distance, ℓ_1 distance):

$$d_{\mathrm{TV}}(p, q) = \sup_{S \subseteq \Omega} (p(S) - q(S)) = \frac{1}{2} \sum_{x \in \Omega} |p(x) - q(x)| \in [0, 1]$$

Domain size $n \in \mathbb{N}$ is big ("goes to ∞").

- Probability distributions over $[n] := \{1, \dots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Total variation distance (statistical distance, ℓ_1 distance):

$$d_{\mathrm{TV}}(p,q) = \sup_{S \subseteq \Omega} (p(S) - q(S)) = \frac{1}{2} \sum_{x \in \Omega} |p(x) - q(x)| \in [0,1]$$

Domain size $n \in \mathbb{N}$ is big ("goes to ∞"). Proximity parameter $\varepsilon \in (0,1]$ is small.

GLOSSARY

- Probability distributions over $[n] := \{1, \ldots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Total variation distance (statistical distance, ℓ_1 distance):

$$d_{\mathrm{TV}}(p, q) = \sup_{S \subseteq \Omega}(p(S) - q(S)) = \frac{1}{2}\sum_{x \in \Omega} |p(x) - q(x)| \in [0,1]$$

Domain size $n \in \mathbb{N}$ is big ("goes to ∞"). Proximity parameter $\varepsilon \in (0,1]$ is small. Lowercase Greek letters are in $(0,1]$.

GLOSSARY

- Probability distributions over $[n] := \{1, \dots, n\}$

$$\Delta([n]) = \left\{ p\colon [n] \to [0,1] \;:\; \sum_{i=1}^{n} p(i) = 1 \right\}$$

- Property (or class) of distributions over $[n]$:

$$\mathcal{P} \subseteq \Delta([n])$$

- Total variation distance (statistical distance, ℓ_1 distance):

$$d_{\mathrm{TV}}(p,q) = \sup_{S \subseteq \Omega} (p(S) - q(S)) = \frac{1}{2} \sum_{x \in \Omega} |p(x) - q(x)| \in [0,1]$$

Domain size $n \in \mathbb{N}$ is big ("goes to ∞"). Proximity parameter $\varepsilon \in (0,1]$ is small. Lowercase Greek letters are in $(0,1]$. Asymptotics $\tilde{O}$, $\tilde{\Omega}$, $\tilde{\Theta}$ hide logarithmic factors.*

ROUND UP THE USUAL SUSPECTS

- Poisson Binomial Distribution (PBD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{0, 1\} \text{ independent.}$$

ROUND UP THE USUAL SUSPECTS

- Poisson Binomial Distribution (PBD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{0, 1\} \text{ independent.}$$

- k-Sum of Independent Integer Random Variables (k-SIIRV)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{0, 1, \ldots, k-1\} \text{ independent.}$$

- Poisson Binomial Distribution (PBD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{0,1\} \text{ independent.}$$

- k-Sum of Independent Integer Random Variables (k-SIIRV)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{0,1,\ldots,k-1\} \text{ independent.}$$

- Poisson Multinomial Distribution (PMD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \ldots, X_n \in \{e_1,\ldots,e_k\} \text{ independent.}$$

ROUND UP THE USUAL SUSPECTS

- Poisson Binomial Distribution (PBD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \dots, X_n \in \{0, 1\} \text{ independent.}$$

- k-Sum of Independent Integer Random Variables (k-SIIRV)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \dots, X_n \in \{0, 1, \dots, k-1\} \text{ independent.}$$

- Poisson Multinomial Distribution (PMD)

$$X = \sum_{j=1}^{n} X_j, \text{ with } X_1 \dots, X_n \in \{e_1, \dots, e_k\} \text{ independent.}$$

- (Discrete) Log-Concave

$$p(k)^2 \geq p(k-1)p(k+1) \text{ and supported on an interval}$$

BUT… WILL WE EVER LEARN?

Trivial baseline in property testing: "you can learn, so you can test."

TESTING BY LEARNING

Trivial baseline in property testing: "you can learn, so you can test."

(i) Learn p without assumptions using a learner for $\Delta([n])$

Trivial baseline in property testing: “you can learn, so you can test.”

(i) Learn p without assumptions using a learner for $\Delta([n])$

(ii) Check if $d_{\mathrm{TV}}(\hat{p}, \mathcal{P}) \leq \frac{\varepsilon}{3}$ (Computational)

Trivial baseline in property testing: "you can learn, so you can test."

(i) Learn p without assumptions using a learner for $\Delta([n])$

(ii) Check if $d_{\mathrm{TV}}(\hat{p}, \mathcal{P}) \leq \frac{\varepsilon}{3}$ (Computational)

Yes, but...

(i) has sample complexity $\Theta(n/\varepsilon^2)$.

“Folklore” baseline in property testing: “if you can learn, you can test.”

"Folklore" baseline in property testing: "if you can learn, you can test."

(i) Learn p as if $p \in \mathcal{P}$ using a learner for $\mathcal{P}$

"Folklore" baseline in property testing: "if you can learn, you can test."

(i) Learn p as if $\mathsf{p} \in \mathcal{P}$ using a learner for $\mathcal{P}$

(ii) Test $\mathsf{d}_{\mathrm{TV}}(\hat{\mathsf{p}}, \mathsf{p}) \leq \frac{\varepsilon}{3}$ vs. $\mathsf{d}_{\mathrm{TV}}(\hat{\mathsf{p}}, \mathsf{p}) \geq \frac{2\varepsilon}{3}$

"Folklore" baseline in property testing: "if you can learn, you can test."

(i) Learn p as if $p \in \mathcal{P}$ using a learner for $\mathcal{P}$

(ii) Test $d_{\mathrm{TV}}(\hat{p}, p) \leq \frac{\varepsilon}{3}$ vs. $d_{\mathrm{TV}}(\hat{p}, p) \geq \frac{2\varepsilon}{3}$

(iii) Check if $d_{\mathrm{TV}}(\hat{p}, \mathcal{P}) \leq \frac{\varepsilon}{3}$ (Computational)

"Folklore" baseline in property testing: "if you can learn, you can test."

(i) Learn p as if $\mathsf{p} \in \mathcal{P}$ using a learner for $\mathcal{P}$

(ii) Test $\mathsf{d}_{\mathrm{TV}}(\hat{\mathsf{p}}, \mathsf{p}) \leq \frac{\varepsilon}{3}$ vs. $\mathsf{d}_{\mathrm{TV}}(\hat{\mathsf{p}}, \mathsf{p}) \geq \frac{2\varepsilon}{3}$

(iii) Check if $\mathsf{d}_{\mathrm{TV}}(\hat{\mathsf{p}}, \mathcal{P}) \leq \frac{\varepsilon}{3}$ (Computational)

The triangle inequality does the rest.

"Folklore" baseline in property testing: "if you can learn, you can test."

(i) Learn p as if $p \in \mathcal{P}$ using a learner for $\mathcal{P}$

(ii) Test if $d_{TV}(\hat{p}, p) \leq \frac{\varepsilon}{3}$ vs. $d_{TV}(\hat{p}, p) \geq \frac{2\varepsilon}{3}$

(iii) Check if $d_{TV}(\hat{p}, \mathcal{P}) \leq \frac{\varepsilon}{3}$ (Computational)

Not quite.

(ii) fine for functions. But for distributions? Requires $\Omega(\frac{n}{\log n})$ samples [VV11, JYW17]

UNIFIED APPROACHES: LEVERAGING STRUCTURE

What we want

General algorithms applying to all (or many) distribution testing problems.

What we want

General algorithms applying to all (or many) distribution testing problems.

Theorem (Wishful)

Let $\mathcal{P}$ be a class of distributions that all exhibit some "nice structure." If $\mathcal{P}$ can be tested with q queries, algorithm $\mathfrak{T}$ can too, with "roughly" q queries as well.

What we want

General algorithms applying to all (or many) distribution testing problems.

Theorem (Wishful)

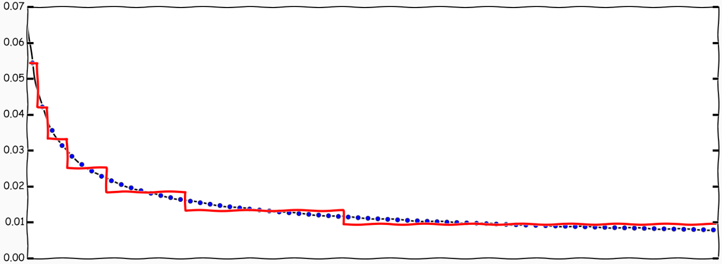
Let $\mathcal{P}$ be a class of distributions that all exhibit some "nice structure." If $\mathcal{P}$ can be tested with q queries, algorithm $\mathfrak{T}$ can too, with "roughly" q queries as well.

More formally, we want:

Goal

Design general-purpose testing algorithms that, when applied to a property $\mathcal{P}$, have (tight, or at least reasonable) sample complexity $q(\varepsilon, \tau)$ as long as $\mathcal{P}$ satisfies some structural assumption $\mathcal{S}_\tau$ parameterized by τ.

Structural assumption $\mathcal{S}_\tau$: every distribution in $\mathcal{P}$ is well-approximated (in a specific ℓ_2-type sense) by a piecewise-constant distribution with $L_{\mathcal{P}}(\tau)$ pieces.



Structural assumption $\mathcal{S}_\tau$: every distribution in $\mathcal{P}$ is well-approximated (in a specific ℓ_2-type sense) by a piecewise-constant distribution with $L_{\mathcal{P}}(\tau)$ pieces.

Theorem ([CDGR16])

There exists an algorithm which, given sampling access to an unknown distribution p over [n] and parameter $\varepsilon \in (0, 1]$, can distinguish with probability 2/3 between **(a)** $p \in \mathcal{P}$ versus (b) $d_{\mathrm{TV}}(p, \mathcal{P}) > \varepsilon$, with $\tilde{O}(\sqrt{nL_{\mathcal{P}}(\varepsilon)}/\varepsilon^3 + L_{\mathcal{P}}(\varepsilon)/\varepsilon^2)$ samples.

Outline: Abstracting ideas from [BKR04] (for monotonicity):

1. **decomposition step:** recursively build a partition Π of $[n]$ in $O(L_{\mathcal{P}}(\varepsilon))$ intervals s.t. p is roughly uniform on each piece. If successful, then p will be close to its "flattening" q on Π; if not, we have proof that $p \notin \mathcal{P}$ and we can reject.
2. **approximation step:** learn q. Can be done with few samples since Π has few intervals.
3. **projection step:** (computational) verify that $d_{\mathrm{TV}}(q, \mathcal{P}) < O(\varepsilon)$.

Outline: Abstracting ideas from [BKR04] (for monotonicity):

1. **decomposition step:** recursively build a partition Π of $[n]$ in $O(L_{\mathcal{P}}(\varepsilon))$ intervals s.t. p is roughly uniform on each piece. If successful, then p will be close to its “flattening” q on Π; if not, we have proof that $p \notin \mathcal{P}$ and we can reject.
2. **approximation step:** learn q. Can be done with few samples since Π has few intervals.
3. **projection step:** (computational) verify that $d_{\mathrm{TV}}(q, \mathcal{P}) < O(\varepsilon)$.

Applications

- monotonicity
- unimodality
- k-modality
- k-histograms
- log-concavity
- Poisson Binomial
- Monotone Hazard Rate

...

THAT'S GREAT! BUT...

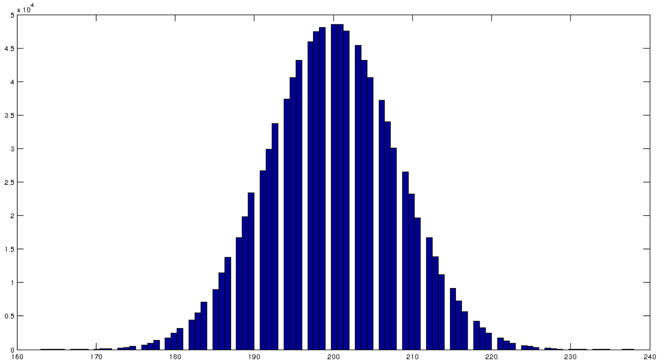


Figure: A 3-SIIRV (for $n = 100$). Like all of us, it has ups and downs.

Structural assumption $\mathcal{S}_\tau$: every distribution in $\mathcal{P}$ has sparse Fourier and effective support: $\exists M_{\mathcal{P}}(\tau), S_{\mathcal{P}}(\tau)$ s.t. $\forall p \in \mathcal{P}$, $\exists I_p \subseteq [n]$ with $|I_p| \leq M_{\mathcal{P}}(\tau)$

$$\|\hat{p}\mathbb{1}_{\overline{S_{\mathcal{P}}(\varepsilon)}}\|_2 \leq O(\varepsilon), \quad \|p\mathbb{1}_{\overline{I_p}}\|_1 \leq O(\varepsilon)$$

Theorem ([CDS17])

There exists an algorithm which, given sampling access to an unknown distribution p over [n] and parameter $\varepsilon \in (0, 1]$, can distinguish with probability 2/3 between (a) $p \in \mathcal{P}$ versus (b) $d_{\mathrm{TV}}(p, \mathcal{P}) > \varepsilon$, with $\tilde{O}(\sqrt{|S_{\mathcal{P}}(\varepsilon)| M_{\mathcal{P}}(\varepsilon)}/\varepsilon^2 + |S_{\mathcal{P}}(\varepsilon)| /\varepsilon^2)$ samples.

Outline:

1. **effective support test:** take samples to identify a candidate I_p, and check $|I_p| \leq M(\varepsilon)$
2. **Fourier effective support test:** invoke a Fourier sparsity subroutine to check that $\|\hat{p}\mathbf{1}_{\overline{S_{\mathcal{P}}(\varepsilon)}}\|_2 \leq O(\varepsilon)$ (if so learn q, inverse Fourier transform of $\hat{p}\mathbf{1}_{\overline{S_{\mathcal{P}}(\varepsilon)}}$)
3. **projection step:** (computational) verify that $d_{\mathrm{TV}}(q, \mathcal{P}) < O(\varepsilon)$.

Outline:

1. **effective support test:** take samples to identify a candidate I_p, and check $|I_p| \leq M(\varepsilon)$
2. **Fourier effective support test:** invoke a Fourier sparsity subroutine to check that $\|\hat{p}\mathbf{1}_{\overline{S_{\mathcal{P}}(\varepsilon)}}\|_2 \leq O(\varepsilon)$ (if so learn q, inverse Fourier transform of $\hat{p}\mathbf{1}_{\overline{S_{\mathcal{P}}(\varepsilon)}}$)
3. **projection step:** (computational) verify that $d_{\mathrm{TV}}(q, \mathcal{P}) < O(\varepsilon)$.

Applications

- k-SIIRVS
- Poisson Binomial
- Poisson Multinomial
- log-concavity

IN MORE DETAIL

Theorem (Testing SIIRVs)

There exists an algorithm that, given $k, n \in \mathbb{N}$, $\varepsilon \in (0,1]$, and sample access to $p \in \Delta(\mathbb{N})$, tests the class of k-SIIRVs with

$$O\left(\frac{kn^{1/4}}{\varepsilon^2}\log^{1/4}\frac{1}{\varepsilon} + \frac{k^2}{\varepsilon^2}\log^2\frac{k}{\varepsilon}\right)$$

samples from p, and runs in time $n(k/\varepsilon)^{O(k\log(k/\varepsilon))}$.

Theorem (Testing SIIRVs)

There exists an algorithm that, given $k, n \in \mathbb{N}$, $\varepsilon \in (0, 1]$, and sample access to $p \in \Delta(\mathbb{N})$, tests the class of k-SIIRVs with

$$O\left(\frac{kn^{1/4}}{\varepsilon^2} \log^{1/4} \frac{1}{\varepsilon} + \frac{k^2}{\varepsilon^2} \log^2 \frac{k}{\varepsilon}\right)$$

samples from p, and runs in time $n(k/\varepsilon)^{O(k \log(k/\varepsilon))}$.

First non-trivial tester for SIIRVs.

Theorem (Testing SIIRVs)

There exists an algorithm that, given $k, n \in \mathbb{N}$, $\varepsilon \in (0,1]$, and sample access to $p \in \Delta(\mathbb{N})$, tests the class of k-SIIRVs with

$$O\left(\frac{kn^{1/4}}{\varepsilon^2}\log^{1/4}\frac{1}{\varepsilon} + \frac{k^2}{\varepsilon^2}\log^2\frac{k}{\varepsilon}\right)$$

samples from p, and runs in time $n(k/\varepsilon)^{O(k\log(k/\varepsilon))}$.

First non-trivial tester for SIIRVs. Near-optimal for constant k: lower bound of $\Omega\left(\frac{k^{1/2}n^{1/4}}{\varepsilon^2}\right)$ [CDGR16].

Theorem (Testing SIIRVs)

There exists an algorithm that, given $k, n \in \mathbb{N}$, $\varepsilon \in (0,1]$, and sample access to $p \in \Delta(\mathbb{N})$, tests the class of k-SIIRVs with

$$O\left(\frac{kn^{1/4}}{\varepsilon^2}\log^{1/4}\frac{1}{\varepsilon}+\frac{k^2}{\varepsilon^2}\log^2\frac{k}{\varepsilon}\right)$$

samples from p, and runs in time $n(k/\varepsilon)^{O(k\log(k/\varepsilon))}$.

First non-trivial tester for SIIRVs. Near-optimal for constant k: lower bound of $\Omega\left(\frac{k^{1/2}n^{1/4}}{\varepsilon^2}\right)$ [CDGR16].

k-SIIRVs...

- are very badly approximated by histograms

k-SIIRVs...

- are very badly approximated by histograms
- have sparse effective support

k-SIIRVs...

- are very badly approximated by histograms
- have sparse effective support
- have nicely bounded ℓ_2 norm

k-SIIRVs...

- are very badly approximated by histograms
- have sparse effective support
- have nicely bounded ℓ_2 norm
- have very nice Fourier spectrum

Theorem (General Testing Statement)

Let $\mathcal{P} \subseteq \Delta(\mathbb{N})$ be a property satisfying the following. $\exists\, S\colon (0,1] \to 2^{\mathbb{N}}$, $M\colon (0,1] \to \mathbb{N}$, and $q_I\colon (0,1] \to \mathbb{N}$ s.t. for all $\varepsilon \in (0,1]$,

1. *Fourier sparsity:* $\forall p \in \mathcal{P}$, the Fourier transform (modulo $M(\varepsilon)$) of p is concentrated on $S(\varepsilon)$: namely, $\|\widehat{p}\mathbf{1}_{\overline{S(\varepsilon)}}\|_2^2 \leq O(\varepsilon^2)$.
2. *Support sparsity:* $\forall p \in \mathcal{P}$, $\exists$ interval $I \subseteq \mathbb{N}$ with $|I| \leq M(\varepsilon)$ such that (i) p is concentrated on I: $p(I) \geq 1 - O(\varepsilon)$ and (ii) I can be identified w.h.p. with $q_I(\varepsilon)$ samples.
3. *Projection:* there is a procedure $\textsc{Project}_{\mathcal{P}}$ which, on input ε and the explicit description of $h \in \Delta(\mathbb{N})$, runs in time $T(\varepsilon)$ and distinguishes between $d_{\rm TV}(h, \mathcal{P}) \leq \frac{2\varepsilon}{5}$, and $d_{\rm TV}(h, \mathcal{P}) > \frac{\varepsilon}{2}$.
4. *(Optional) L_2-norm bound:* $\exists b \in (0,1]$ s.t. $\|p\|_2^2 \leq b\ \forall p \in \mathcal{P}$.

Then, $\exists$ a tester for $\mathcal{P}$ with sample complexity m equal to

$$O\left(\frac{\sqrt{|S(\varepsilon)|\, M(\varepsilon)}}{\varepsilon^2} + \frac{|S(\varepsilon)|}{\varepsilon^2} + q_I(\varepsilon)\right)$$

(if (iv) holds, can replace by $O\left(\frac{\sqrt{b}M(\varepsilon)}{\varepsilon^2} + \frac{|S(\varepsilon)|}{\varepsilon^2} + q_I(\varepsilon)\right)$); and runs in time $O(m\,|S| + T(\varepsilon))$.

Further, when the algorithm accepts, it also learns p: i.e., outputs hypothesis h s.t. $d_{\rm TV}(p, h) \leq \varepsilon$.

Require: sample access to a distribution $p \in \Delta(\mathbb{N})$, parameter $\varepsilon \in (0,1]$, $b \in (0,1]$, functions $S\colon (0,1] \to 2^{\mathbb{N}}$, $M\colon (0,1] \to \mathbb{N}$, $q_I\colon (0,1] \to \mathbb{N}$, and procedure $\textsc{Project}_{\mathcal{P}}$

```
Effective Support
    Take q_I(ε) samples to identify a "candidate set" I.            ▷ Works s.h.p if p ∈ 𝒫.
    Take O(1/ε) samples to distinguish b/w p(I) ≥ 1 − ε/5 and p(I) < 1 − ε/4.    ▷ Correct w.h.p.
    if |I| > M(ε) or we detected that p(I) > ε/4 then
        return reject
    end if

Fourier Effective Support
    Simulating sample access to p′ = p mod M(ε), call TestFourierSupport on p′ with
    parameters M(ε), ε/(5√M(ε)), b, and S(ε).
    if TestFourierSupport returned reject then
        return reject
    end if
    Let ĥ = (ĥ(ξ))_{ξ∈S(ε)} be the Fourier coefficients it outputs, and h their inverse Fourier
    transform (modulo M(ε))                                          ▷ Do not actually compute h here.

Projection Step
    Call Project_𝒫 on parameters ε and h, and return accept if it does, reject otherwise.
```

With this in hand...

The testing result for k-SIIRVs immediately follows.

With this in hand...

The testing result for k-SIIRVs immediately follows.(Modulo one little lie.)

With this in hand...

The testing result for k-SIIRVs immediately follows.(Modulo one little lie.)

Other results...

For PBD ($k = 2$) and PMDs (multidimensional) as well, the second w/ the suitable generalization of discrete Fourier transform.

Theorem (Testing Fourier Sparsity)

Given parameters $M \geq 1$, $\varepsilon, b \in (0, 1]$, subset $S \subseteq [M]$ and sample access to $q \in \Delta([M])$, TestFourierSupport either rejects or outputs Fourier coefficients $\widehat{h'} = (\widehat{h'}(\xi))_{\xi \in S}$ s.t., w.h.p., all the following holds.

1. if $\|q\|_2^2 > 2b$, then it rejects;
2. if $\|q\|_2^2 \leq 2b$ and $\forall q^* : [M] \to \mathbb{R}$ with $\widehat{q^*}$ supported entirely on S, $\|q - q^*\|_2 > \varepsilon$, then it rejects;
3. if $\|q\|_2^2 \leq b$ and $\exists q^* : [M] \to \mathbb{R}$ with $\widehat{q^*}$ supported entirely on S s.t. $\|q - q^*\|_2 \leq \frac{\varepsilon}{2}$, then it does not reject;
4. if it does not reject, then $\|\widehat{q}\mathbf{1}_S - \widehat{h'}\|_2 \leq O(\varepsilon\sqrt{M})$ and the inverse Fourier transform (modulo M) h' of the Fourier coefficients it outputs satisfies $\|q - h'\|_2 \leq O(\varepsilon)$.

Moreover, it takes $m = O\left(\frac{\sqrt{b}}{\varepsilon^2} + \frac{|S|}{M\varepsilon^2} + \sqrt{M}\right)$ samples from q, and runs in time $O(m\,|S|)$.

Idea

Consider the Fourier coefficients of the empirical distribution (from few samples).

Idea

Consider the Fourier coefficients of the empirical distribution (from few samples).

Second idea

Do not consider directly these coefficients (timewise, expensive). Instead, rely on (the analysis of) an ℓ_2 identity tester [CDVV14]+Plancherel to get guarantees on the FC.

Idea

Consider the Fourier coefficients of the empirical distribution (from few samples).

Second idea

Do not consider directly these coefficients (timewise, expensive). Instead, rely on (the analysis of) an ℓ_2 identity tester [CDVV14]+Plancherel to get guarantees on the FC.

OPEN QUESTIONS, AND QUESTIONS.

OPEN QUESTIONS

- More applications: what is your favorite property?

OPEN QUESTIONS

- More applications: what is your favorite property?
- Uncertainty Principle: what about this $\sqrt{|S(\varepsilon)|\, M(\varepsilon)}$ term?

OPEN QUESTIONS

- More applications: what is your favorite property?
- Uncertainty Principle: what about this $\sqrt{|S(\varepsilon)|\, M(\varepsilon)}$ term?
- Fourier works: what about other bases?

Thank You.

Jayadev Acharya and Constantinos Daskalakis.
Testing Poisson Binomial Distributions.
In Proceedings of SODA, pages 1829–1840, 2015.

Jayadev Acharya, Constantinos Daskalakis, and Gautam C. Kamath.
Optimal Testing for Properties of Distributions.
In C. Cortes, N.D. Lawrence, D.D. Lee, M. Sugiyama, R. Garnett, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, pages 3577–3598. Curran Associates, Inc., 2015.

Jayadev Acharya, Ilias Diakonikolas, Jerry Zheng Li, and Ludwig Schmidt.
Sample-optimal density estimation in nearly-linear time.
In Proceedings of SODA, pages 1278–1289. SIAM, 2017.

Eric Blais, Clément L. Canonne, and Tom Gur.
Distribution testing lower bounds via reductions from communication complexity.
In Computational Complexity Conference, volume 79 of LIPIcs, pages 28:1–28:40. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

Tuğkan Batu, Eldar Fischer, Lance Fortnow, Ravi Kumar, Ronitt Rubinfeld, and Patrick White.
Testing random variables for independence and identity.
In Proceedings of FOCS, pages 442–451, 2001.

Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White.
Testing that distributions are close.
In Proceedings of FOCS, pages 189–197, 2000.

Arnab Bhattacharyya, Eldar Fischer, Ronitt Rubinfeld, and Paul Valiant.
Testing monotonicity of distributions over general partial orders.
In Proceedings of ITCS, pages 239–252, 2011.

Tuğkan Batu, Ravi Kumar, and Ronitt Rubinfeld.
Sublinear algorithms for testing monotone and unimodal distributions.
In Proceedings of STOC, pages 381–390, New York, NY, USA, 2004. ACM.

Arnab Bhattacharyya and Yuichi Yoshida.
Property Testing.
Forthcoming, 2017.

Clément L. Canonne.
A Survey on Distribution Testing: your data is Big. But is it Blue?
Electronic Colloquium on Computational Complexity (ECCC), 22:63, April 2015.

Clément L. Canonne, Ilias Diakonikolas, Themis Gouleakis, and Ronitt Rubinfeld.
Testing Shape Restrictions of Discrete Distributions.
In Proceedings of STACS, 2016.

See also [CDGR17] (full version).

Clément L. Canonne, Ilias Diakonikolas, Themis Gouleakis, and Ronitt Rubinfeld.
Testing shape restrictions of discrete distributions.
Theory of Computing Systems, pages 1–59, 2017.

Yu Cheng, Ilias Diakonikolas, and Alistair Stewart.
Playing anonymous games using simple strategies.
In Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, Proceedings of SODA, pages 616–631, Philadelphia, PA, USA, 2017. Society for Industrial and Applied Mathematics.

Siu-on Chan, Ilias Diakonikolas, Rocco A. Servedio, and Xiaorui Sun.
Learning mixtures of structured distributions over discrete domains.
In Proceedings of SODA, pages 1380–1394, 2013.

Siu-on Chan, Ilias Diakonikolas, Rocco A. Servedio, and Xiaorui Sun.
Efficient density estimation via piecewise polynomial approximation.
In Proceedings of STOC, pages 604–613. ACM, 2014.

Siu-on Chan, Ilias Diakonikolas, Rocco A. Servedio, and Sun. Xiaorui.
Near-optimal density estimation in near-linear time using variable-width histograms.

In Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pages 1844–1852, 2014.

Siu-on Chan, Ilias Diakonikolas, Gregory Valiant, and Paul Valiant.
Optimal algorithms for testing closeness of discrete distributions.
In Proceedings of SODA, pages 1193–1203, 2014.

Ilias Diakonikolas, Themis Gouleakis, John Peebles, and Eric Price.
Collision-based testers are optimal for uniformity and closeness.
Electronic Colloquium on Computational Complexity (ECCC), 23:178, 2016.

Ilias Diakonikolas and Daniel M. Kane.
A new approach for testing properties of discrete distributions.
In Proceedings of FOCS. IEEE Computer Society, 2016.

Oded Goldreich, Shafi Goldwasser, and Dana Ron.
Property testing and its connection to learning and approximation.
Journal of the ACM, 45(4):653–750, July 1998.

Oded Goldreich, editor.
Property Testing: Current Research and Surveys.
Springer, 2010.
LNCS 6390.

Oded Goldreich.
Introduction to Property Testing.
Forthcoming, 2017.

Oded Goldreich and Dana Ron.
On testing expansion in bounded-degree graphs.
Technical Report TR00-020, Electronic Colloquium on Computational Complexity (ECCC), 2000.

Jiantao Jiao, Han Yanjun, and Tsachy Weissman.
Minimax Estimation of the L_1 Distance.
ArXiv e-prints, May 2017.

Reut Levi, Dana Ron, and Ronitt Rubinfeld.
Testing properties of collections of distributions.
Theory of Computing, 9:295–347, 2013.

Liam Paninski.
A coincidence-based test for uniformity given very sparsely sampled discrete data.
IEEE Transactions on Information Theory, 54(10):4750–4755, 2008.

Dana Ron.
Property Testing: A Learning Theory Perspective.

Foundations and Trends in Machine Learning, 1(3):307–402, 2008.

Dana Ron.
Algorithmic and analysis techniques in property testing.
Foundations and Trends in Theoretical Computer Science, 5:73–205, 2010.

Ronitt Rubinfeld and Madhu Sudan.
Robust characterization of polynomials with applications to program testing.
SIAM Journal on Computing, 25(2):252–271, 1996.

Ronitt Rubinfeld.
Taming big probability distributions.
XRDS: Crossroads, The ACM Magazine for Students, 19(1):24, sep 2012.

Paul Valiant.
Testing symmetric properties of distributions.
SIAM Journal on Computing, 40(6):1927–1968, 2011.

Gregory Valiant and Paul Valiant.
Estimating the unseen: An $n/\log n$-sample estimator for entropy and support size, shown optimal via new clts.
In Proceedings of STOC, pages 685–694, 2011.

Gregory Valiant and Paul Valiant.
An automatic inequality prover and instance optimal identity testing.

In Proceedings of FOCS, 2014.

Gregory Valiant and Paul Valiant.
An automatic inequality prover and instance optimal identity testing.
SIAM Journal on Computing, 46(1):429–455, 2017.
Journal version of [VV14].