Distribution Testing with a Confused Collector

Renato Ferreira Pinto Jr.
University of Waterloo, Canada

Nathaniel Harms
EPFL, Lausanne, Switzerland

Abstract

We are interested in testing properties of distributions with systematically mislabeled samples. Our goal is to make decisions about unknown probability distributions, using a sample that has been collected by a *confused collector*, such as a machine-learning classifier that has not learned to distinguish all elements of the domain. The confused collector holds an unknown clustering of the domain and an input distribution μ, and provides two oracles: a *sample oracle* which produces a sample from μ that has been labeled according to the clustering; and a *label-query oracle* which returns the label of a query point x according to the clustering.

Our first set of results shows that identity, uniformity, and equivalence of distributions can be tested efficiently, under the earth-mover distance, with remarkably weak conditions on the confused collector, even when the unknown clustering is *adversarial*. This requires defining a variant of the distribution testing task (inspired by the recent *testable learning* framework of Rubinfeld & Vasilyan), where the algorithm should test a joint property of the distribution and its clustering. As an example, we get efficient testers when the distribution tester is allowed to reject if it detects that the confused collector clustering is "far" from being a decision tree.

The second set of results shows that we can sometimes do significantly better when the clustering is *random* instead of adversarial. For certain one-dimensional random clusterings, we show that uniformity can be tested under the TV distance using $\widetilde{O}\left(\frac{\sqrt{n}}{\rho^{3/2}\epsilon^2}\right)$ samples and *zero* queries, where $\rho \in (0, 1]$ controls the "resolution" of the clustering. We improve this to $O\left(\frac{\sqrt{n}}{\rho\epsilon^2}\right)$ when queries are allowed.

2012 ACM Subject Classification Theory of computation → Streaming, sublinear and near linear time algorithms; Theory of computation → Randomness, geometry and discrete structures; Theory of computation → Computational geometry; Theory of computation → Machine learning theory; Theory of computation → Probabilistic computation; Mathematics of computing → Probability and statistics

Keywords and phrases Distribution testing, property testing, uniformity testing, identity testing, earth-mover distance, sublinear algorithms

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.47

Related Version *Full Version*: `https://arxiv.org/abs/2311.14247`

Funding *Renato Ferreira Pinto Jr.*: Partly funded by an NSERC Canada Graduate Scholarship Doctoral Award.
Nathaniel Harms: Some of this work was done while the author was at the University of Waterloo. Partly funded by an NSERC Postdoctoral Fellowship, and Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number MB22.00026.

Acknowledgements We thank Eric Blais for helpful discussions and comments on the presentation of this article, and Maryam Aliakbarpour for references on testing with imperfect information. We thank the anonymous reviewers for their comments and references to related work.

© Renato Ferreira Pinto Jr. and Nathaniel Harms;
licensed under Creative Commons License CC-BY 4.0

1 Introduction

We are interested in the problem of making decisions about an unknown probability distribution, using only samples from that distribution which have been collected and labeled by an entity who may not be capable of accurate distinctions between elements of the domain. Consider some informal but illustrative examples:

1. We wish to make a decision about the distribution of woodland flora based on a sample that was tabulated by a research assistant who cannot differentiate between black spruce and white spruce, or between red maple and sugar maple, and has counted the spruces together and the maples together by mistake[1].
2. Our sample of a distribution of images has been labeled by an algorithm, such as a machine learning classifier, that fails to distinguish between, say, red squirrels and grey squirrels. For instance, the algorithm might be represented by a decision tree, and we are not certain if the decision tree has sufficient granularity for our task.
3. Sample points have been truncated due to rounding, or hashed in an effort to save space, possibly causing collisions.
4. Random environmental conditions prevent the collector from making perfect distinctions. Say we wish to make decisions about the distribution of fossils over time, but we are unable to distinguish between fossils from year x and year y unless those years are separated by some random geological event that leaves traces in the rock. Note that this is different from having a sample corrupted by random noise, since the mislabelling is *systematic*, applying to all sample points in the same way.

These types of constraints are essentially unavoidable in practice, and they also occur in theoretical analysis of distribution-free property testing algorithms (which we explain briefly in Section 1.3). When faced with a situation like in these examples, we call the collector of the sample a *confused collector*, and our goal is to design *distribution testing* algorithms that work even when faced with a confused collector, which we will define formally below.

Distribution testing is a fundamental type of statistical task, where the goal is to determine whether an unknown probability distribution μ belongs to a property $\mathcal{P}$, or is ϵ-*far* from the property $\mathcal{P}$, meaning that its distance to any distribution $\nu \in \mathcal{P}$ is at least ϵ; the distance metric depends on the problem but is usually assumed to be the total-variation (TV) distance. The tester should make this decision using a random sample from μ that is as small as possible while allowing it to succeed with high probability (usually probability 2/3). See [12] for a recent survey.

To our knowledge, the most closely related work on distribution testing does not capture the phenomenon we are interested in; they assume that the tester either: can gain perfect knowledge of the sample using queries to the sample points [29]; sees random noise applied independently to each sample point [6]; sees samples labeled by a permutation of the correct labels [16]; sees samples through a privacy mechanism [26, 36, 3, 1, 4, 2]; sees "truncated" samples restricted to a subset of the domain [18, 19]; or solves the related but nearly opposite task of testing if the input *can* be clustered to match some known target [15]. Recent work in learning theory [25] states that, while there is extensive applied learning literature on the type of mislabeled samples we describe, little is known theoretically; they study statistical learning problems (e. g. Gaussian mean estimation) in a model similar in spirit to what we will define, but fundamentally different in the details (see Section 1.3 for a comparison).

[1] We thank ecologist Prof. Julie Messier for these examples of trees frequently mistaken by students.

In this paper we will define a general model for the confused collector and show that distribution testing tasks can be performed efficiently under remarkably weak restrictions on the confused collector, even when the collector is *adversarial*. Then, we will show how to significantly improve some of these results when the distinctions made by the collector are *random*.

Modelling the Confused Collector. The idea is that the confused collector holds a *clustering* of the domain, and labels each sample point x with a representative of its cluster. For a fixed domain $\mathcal{X}$, a *clustering* of the domain is a pair (Γ, rep) consisting of a partition $\Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_k\}$ of $\mathcal{X}$ into some number k of *cells* Γ_i, together with a choice of *representatives* of each cell, $\mathsf{rep} : [k] \to \mathcal{X}$ such that $\mathsf{rep}(i) \in \Gamma_i$ for each $i \in [k]$. We define $\gamma : \mathcal{X} \to [k]$ as the function that assigns to each point x the index of its cell, so that $x \in \Gamma_{\gamma(x)}$. The input to the distribution testing algorithm consists of a clustering (Γ, rep) and one or more distributions[2] $\mu_1, \mu_2, \ldots$. The inputs are held by the confused collector, who provides the algorithm with access via the following oracles:

1. *Clustered-Sample Oracle.* For each distribution μ_i in the input, the confused collector provides access to μ_i via a *clustered-sample oracle* $\mathsf{SAMP}(\Gamma, \mathsf{rep}, \mu_i)$. On request, this oracle produces an independent sample point of the form $\mathsf{rep}(\gamma(x))$ where $x \sim \mu_i$; i.e. , the oracle provides the algorithm with the label of x, defined as the representative of the cluster that contains x.
2. *Label Oracle.* Thinking of the confused collector as an entity (e.g. a machine learning classifier) that has labeled the sample, it is sensible to allow the tester to ask the collector about its clustering. The confused collector provides access to a *label oracle* $\mathsf{LABEL}(\Gamma, \mathsf{rep})$ which, on query $x \in \mathcal{X}$, answers with $\mathsf{rep}(\gamma(x))$, the representative of the cluster containing x. Unlike typical property testing models, we think of queries as being cheap, relative to samples. For example, we may have black-box access to the algorithm that provided labels for a sample, without having the ability to request additional samples, or we can ask our research assistant about their clustering without sending them back to the forest.

Not much can be done if we allow the confused collector to hold *any* clustering, while making the same demands on the tester as in the standard model. If a sample of woodland flora were to be labeled by the authors of this paper, not many interesting properties could be tested under the resulting partition into only 2 or 3 cells, even if the tester exactly learns the clustering. It is important to observe that two distributions μ and ν are indistinguishable to the tester if they can be transformed into each other by transporting mass within individual cells of the clustering, and therefore an adversary can force two distributions with TV distance 1 to be indistinguishable to the tester, making standard distribution testing impossible.

Results and Organization. This paper is clustered into two parts. Part I shows how to sidestep the impossibility we just described, even when the clustering is *adversarial*, by defining a natural relaxation of the distribution testing task, where the algorithm tests a joint property of the distribution and its clustering. We present general baseline upper bounds for the *identity* and *equivalence testing* tasks, under adversarial clusterings. One example application, motivated by the example where the confused collector is a machine learning classifier, is that the relaxed testing tasks can be done efficiently when the tester expects the clustering to be realized by a *decision tree* and can test this assumption.

[2] In Section 1.3 we also briefly discuss a setting where each input distribution is held by a different collector.

These results provide a foundation for Part II, where we present more involved technical results showing how to significantly improve upon the above baseline results in certain special cases where we assume *randomized* clusterings. Motivated by the "environmental randomness" example and applications to property testing, we show that certain random clusters of the domain $[n]$ allow the standard (non-relaxed) *uniformity testing* task to be accomplished efficiently with *zero* label queries, and even more efficiently with queries.

Due to space constraints, we include here only a summary of our results. See the full version of this paper ([24]) for the complete set of results and proofs.

1.1 Part I: Adversarial Clustering

We begin our exploration by establishing that natural testing tasks can be performed efficiently even with an *adversarial* confused collector. It is not obvious *a priori* that non-trivial results are even possible under adversarial clusterings, because the adversary can provide a clustering that makes distribution testing impossible. The simple message of Part I is that the tester can detect if this has occurred: the algorithm tests both the input distribution *and* the clustering. Inspired by recent work on *testable learning* [35], we allow the tester to reject if it deems the clustering unsuitable, while requiring the tester to both (1) accept "good" clusterings and (2) function properly as a distribution tester whenever it accepts the clustering, regardless of whether it was truly "good".

We give general upper bounds for the basic *uniformity*, *identity*, and *equivalence testing* problems. In *uniformity testing*, the tester checks if the input distribution is uniform. In *identity testing*, the tester knows a target distribution ν and tests whether the input μ is equal to ν or ϵ-far from ν. In *equivalence testing* (sometimes called *closeness testing*), the tester is given two input distributions μ and ν and decides if $\mu = \nu$, or if they are ϵ-far from each other. Before formally defining the testing task, we give one informal application of our results, motivated by the example where the tester should work properly when the sample is labeled by a "good" machine learning classifier:

▶ **Theorem 1** (Informal). *Suppose the collector is promised to provide a clustering of the constant-dimensional cube* $[0,1]^d$ *into* convex *cells of radius at least* δ*. Then uniformity can be tested using* $\mathrm{poly}(\frac{1}{\epsilon} \cdot \log(1/\delta))$ *samples and queries, under the earth-mover distance, if the tester must accept clusterings realized by decision trees (with nodes of the form "*$x_i < t$*?") that put most of the input distribution in low-diameter cells, and may otherwise reject the clustering.*

1.1.1 The Testing Task

Let us explain how to arrive at a suitable definition of distribution testing when faced with an adversarially-chosen clustering. Consider the simple problem of distinguishing distributions μ supported on a single element, from the distributions ν which have probability mass $1/2$ on each of two elements. This is trivial in the standard distribution testing model, but impossible if the confused collector holds any clustering with a cell of size at least 2, because the two supports of μ may lie in the same cell. One may consider three ways to fix this problem:

Change the distance metric. We should not use TV distance to define our testing task, because the adversary can choose the clustering to "hide" arbitrarily large TV distances. Assuming the domain $\mathcal{X}$ is equipped with an ambient metric denoted by $\mathsf{dist}(\cdot,\cdot)$, we instead

use the *earth-mover* (or *Wasserstein*) distance on distributions, defined as

$$\mathsf{EMD}_{\mathsf{dist}}(\mu,\nu) := \inf_{\pi} \sum_{x,y\in\mathcal{X}} \pi(x,y)\cdot \mathsf{dist}(x,y)\,,$$

where the infimum is taken over all couplings π of μ and ν. This means that transporting mass within a cell of the clustering cannot transform a distribution μ into another one, ν, that is *far* in $\mathsf{EMD}_{\mathsf{dist}}$, as long as the cell is small with respect to the ambient metric dist.

Reject bad clusterings. By itself, using EMD does not solve our problems, since there is no guarantee that the cells are small. One solution is to demand that the confused collector use a low-diameter clustering, but this is too strong: we may not trust that the collector has acquiesced to our demands; low-diameter clusterings of the entire (large) domain may be costly for the collector to compute; and low-diameter clusterings do not depend on the input probability distribution – the collector may itself have learned the clustering from the input distribution, and we may be satisfied with a collector who makes poor distinctions on low-probability elements. Instead, we allow the tester to reject clusterings not belonging to a defined set of "good" clusterings, while requiring it to succeed on any clustering that passes the test.

Restrict the clusterings. Some types of clusterings make the task of rejecting bad clusterings infeasible. Using label queries, the tester can learn the entire clustering by querying every point in the domain, but we want sublinear query complexity. If, for example, the clusters are not even required to be *connected*, then it might become infeasible to efficiently detect bad clusterings. But, by making reasonable assumptions on the clusterings (including, say, that the cells of the clustering are connected or convex), it becomes feasible to check whether the given clustering is suitable.

This leads to our definition of distribution testing. For convenience, we assume that the metric $\mathsf{dist}(\cdot,\cdot)$ is normalized, meaning it has diameter 1.

▶ **Definition 2** (Testing with an Adversarial Confused Collector)**.** *Fix a domain $\mathcal{X}$, let $\mathcal{P}$ be a property of distributions over $\mathcal{X}$, and let $d(\cdot,\cdot)$ be a metric on probability distributions over $\mathcal{X}$. Let $\mathcal{U}$ be a class (*universe*) of clusterings, and let $\mathcal{D}$ be a set of "good" clustering-distribution pairs $((\Gamma,\mathsf{rep}),\mu)$ where $(\Gamma,\mathsf{rep})\in\mathcal{U}$ and μ is an arbitrary distribution. Let $\mathcal{A}$ be an algorithm with clustered-sample and label oracle access to the input $((\Gamma,\mathsf{rep}),\mu)$, whose possible outputs are ACCEPT, REJECT, and CLUSTER-REJECT. Then we say $\mathcal{A}$ is a $(\mathcal{U},\mathcal{D},\epsilon,\delta)$-*distribution tester for $\mathcal{P}$ under metric d *if it satisfies the following on every input $((\Gamma,\mathsf{rep}),\mu)$ where $(\Gamma,\mathsf{rep})\in\mathcal{U}$:*

1. *If $((\Gamma,\mathsf{rep}),\mu)\in\mathcal{D}$ (i.e. (Γ,rep) is a "good" clustering for μ) then the output of $\mathcal{A}$ is CLUSTER-REJECT with probability at most δ;*
2. *If $\mu\in\mathcal{P}$ then the output is in $\{$ACCEPT, CLUSTER-REJECT$\}$ with probability at least $1-\delta$;*
3. *If $d(\mu,\mathcal{P})>\epsilon$ then the output is in $\{$REJECT, CLUSTER-REJECT$\}$ with probability at least $1-\delta$,*

where the probabilities are over the randomness of $\mathcal{A}$ and the responses to the oracle calls. Note that this definition permits standard boosting: an algorithm satisfying the above conditions, with (say) $\delta\le 1/6$, may be boosted to any δ' by taking a majority vote of $\Theta(\log(1/\delta'))$ runs.

Different problems may lead to different notions of "good" clustering. In this paper the "good" clusters will be those belonging to a chosen subclass $\mathcal{G}\subseteq\mathcal{U}$ which also satisfy a "high-probability of low-diameter" (HPLD) condition, which gives the following instance of the testing task:

▶ **Definition 3** (Diameter-Guarded Testing). *For a finite unit-diameter metric space* $(\mathcal{X}, \mathsf{dist})$*, property* $\mathcal{P}$ *of distributions over* $\mathcal{X}$*, universe* $\mathcal{U}$ *of clusterings, and subset* $\mathcal{G} \subseteq \mathcal{U}$*, we say that an algorithm* $\mathcal{A}$ *is a* $(\mathcal{U}, \mathcal{G}, \Delta)$-diameter-guarded (ϵ, δ)-tester *for* $\mathcal{P}$ *if it is a* $(\mathcal{U}, \mathcal{G}_{\Delta,\epsilon}, \epsilon, \delta)$*-distribution tester under* $\mathsf{EMD}_{\mathsf{dist}}$*, where*

$$\mathcal{G}_{\Delta,\epsilon} := \left\{ ((\Gamma, \mathsf{rep}), \mu) \mid (\Gamma, \mathsf{rep}) \in \mathcal{G},\ \mathop{\mathbb{P}}_{\boldsymbol{x}\sim\mu} \left[\mathsf{diam}_{\mathsf{dist}}(\Gamma_{\gamma(\boldsymbol{x})}) > \Delta \right] \leq c \cdot \epsilon \right\}$$

with $c := \frac{1}{384 \ln(24)}$ *(defined this way for convenience in the analysis). If no* δ *is specified, it is assumed to be* $\delta = 1/6$*.*

The above definition does not capture the equivalence testing task, where there are two input distributions, but the adaptation is straightforward (see full version). We have chosen "good" to mean HPLD ($\mathop{\mathbb{P}}_{x\sim\mu} [\mathsf{diam}_{\mathsf{dist}}(\Gamma_\gamma \boldsymbol{x})) > \Delta] \leq c \cdot \epsilon$) over two possible alternatives:

The first alternative is the less permissive condition that Γ is a low-diameter clustering, i.e. $\mathsf{diam}_{\mathsf{dist}}(\Gamma_i) < \Delta$ for all clusters Γ_i. This makes the algorithm's job easier but allows it to CLUSTER-REJECT in many cases where we wish for it to work: if the input μ is concentrated on a small fraction of the domain, it should be acceptable for the clustering to be coarse elsewhere.

The second alternative is the more permissive condition that Γ has low *average* diameter, i.e. $\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mu} \left[\mathsf{diam}_{\mathsf{dist}}(\Gamma_{\gamma(\boldsymbol{x})}) \right] \leq \Delta$. Some of our applications can be strengthened in this way (essentially when the diameter of any cluster in $\mathcal{U}$ can be efficiently estimated), but we chose HPLD as an option that is both reasonable under our motivation and feasible for many applications.

1.1.2 Results

We prove a general lemma that reduces the complexity of identity and equivalence testing to the query complexity of two subroutines:

Cell discovery. Given a representative point r, output an approximation of its cell $\Gamma_{\gamma(r)}$.

Cell rejection. Given a representative point r and two parameters $t_1 < t_2$, distinguish between the case where the cell $\Gamma_{\gamma(r)}$ has diameter at most t_1, or at least t_2.

The query complexities of these subroutines depend on the geometry of the underlying metric space as well as the "universe" $\mathcal{U}$ of clusterings that the confused collector is promised to provide, and the "good" clustering geometry $\mathcal{G}$ that the tester is required to accept. Instead of stating the general result, we summarize the main applications to the following classes of clusterings:

- $\mathcal{C}$ is the class of *connected* clusterings of the hypergrid $[n]^d$. A clustering is connected if each of its cells is a connected subset of the standard hypergrid graph on vertices $[n]^d$, where $x, y \in [n]^d$ have an edge when $\|x - y\|_1 = 1$.
- $\mathcal{CC}$ is the class of *connected convex* clusterings of the hypergrid $[n]^d$, where each cell is both *connected* and *convex*. A subset $S \subseteq [n]^d$ is convex if it is equal to its convex hull.
- $\mathcal{B}$ is the class of *axis-aligned box* clusterings of the hypergrid $[n]^d$, where each cell is an axis-aligned box. Note that $\mathcal{B} \subseteq \mathcal{CC}$, and that decision-tree clusterings are a subclass of $\mathcal{B}$.
- $\mathcal{CV}_\delta$ is the class of δ*-convex* clusterings of the continuous cube $[0, 1]^d$, where each cell is a convex set that is guaranteed to contain the ℓ_2-ball of radius δ around its representative point.
- $\mathcal{B}_\delta$ is the class of clusterings of the continuous cube $[0, 1]^d$, where each cell is an axis-aligned box, and is guaranteed to contain the ℓ_2-ball of radius δ around its representative point.

Table 1 Summary of applications in Part I. The bounds are stated for $d = O(1)$, and the normalized ℓ_p metric with $p \geq 1$. The sample complexity is given by $m(\epsilon)$ at the top of each column, and $q(\epsilon)$ is the query complexity. The main difference between each setting is the promised cell geometry $\mathcal{U}$, the "good" cell geometry $\mathcal{G}$, the "good" diameter Δ (with higher values requiring the algorithm to accept the clustering more often), and the query complexity, which is sublinear.

$(\mathcal{U}, \mathcal{G}, \Delta)$	**Identity**	**Equivalence**
Domain $[n]^d$	$m(\epsilon) = \widetilde{O}(\epsilon^{-\max\{2, \frac{d}{2}\}})$	$m(\epsilon) = \widetilde{O}(\epsilon^{-\max\{2, \frac{2d}{3}\}})$
$(\mathcal{C}, \mathcal{C}, \epsilon/8d^{1/p})$	–	$q(\epsilon) = O(\epsilon^{d-2} n^{d-1})$
$(\mathcal{CC}, \mathcal{CC}, \epsilon/8d^{1/p})$	$q(\epsilon) = O(m(\epsilon) \cdot n^{d-1})$	$q(\epsilon) = O(\epsilon^{d-2} n^{d-1})$
$(\mathcal{B}, \mathcal{B}, \epsilon/8)$	$q(\epsilon) = O(m(\epsilon) \cdot \log n)$	$q(\epsilon) = O(\frac{1}{\epsilon} \cdot \log n)$
$(\mathcal{CC}, \mathcal{B}, \epsilon/8)$ (domain $[n]^2$)	–	$q(\epsilon) = O(\frac{1}{\epsilon} \cdot \log n)$
Domain $[0,1]^d$	**Uniformity**	**Equivalence**
$(\mathcal{CV}_\delta, \mathcal{B}_\delta, \epsilon/16)$	$q(\epsilon) = O(m(\epsilon) \cdot \text{poly} \log \frac{1}{\delta})$	$q(\epsilon) = \widetilde{O}(\frac{1}{\epsilon} \cdot \text{poly} \log \frac{1}{\delta})$
$(\mathcal{CV}_\delta, \mathcal{CV}_\delta, \epsilon/16d^{1/p})$	–	$q(\epsilon) = \widetilde{O}(\frac{1}{\epsilon} \cdot \text{poly} \log \frac{1}{\delta})$

The most difficult instances of $(\mathcal{U}, \mathcal{G}, \Delta)$-diameter-guarded testing are when $\mathcal{U}$ is as inclusive as possible (the universe of clusterings is large) and $\mathcal{G} \subseteq \mathcal{U}$ is as inclusive as possible (the algorithm is required to function as a distribution tester on a wider class of inputs without cluster-rejecting). See Table 1 for a summary of the quantitative bounds.

Connected clusters. The most difficult case for testers on domain $[n]^d$ is when the clusters are only promised to be connected, and it must accept any connected HPLD clustering. Plugging in a simple cell rejection subroutine, we show that $m(\epsilon) = 2^{O(d)} \cdot \widetilde{O}(\epsilon^{-\max\{2, \frac{2d}{3}\}})$ samples and $q(\epsilon) = O(d\epsilon^{d-2} n^{d-1})$ queries suffice for equivalence testing, which is sublinear in the domain size n^d. Since we are testing under EMD, it may be most natural to consider small $\epsilon = O(n^{-c})$ for constants $c > 0$. It may seem odd that the number of queries *decreases* as $\epsilon \to 0$ (when $d > 2$), but this is simply because the query complexity is a balance of the *number* of cells that must be checked (which increases as $\epsilon \to 0$) and the *size* of the cells that must be accepted (which decreases).

The connectivity promise seems too weak to get sublinear query complexity for cell discovery, so the best sublinear-query result that we get is by reducing identity testing to equivalence testing. One may always reduce identity testing to equivalence testing in the confused collector model, by incurring an additive $m(\epsilon)$ label query cost to simulate clustered-sample requests to the known target distribution ν.

Convex clusters. A reasonable, yet still very weak, condition to place on the confused collector is that its clustering has convex cells (and on the hypergrid $[n]^d$, we also keep the condition that it is connected). In the hypergrid $[n]^d$, we are now able to get sublinear query complexity for cell discovery, giving a better bound on identity testing than for merely connected cells. In the continuous domain, we use subroutines for convex optimization using membership oracles [31] to implement the cell rejection procedure to give results for testing equivalence.

Axis-aligned box clusters. One of our main motivations was for clusterings computed by decision trees with nodes of the form "$x_i < t$?", which are a special case of axis-aligned box clusterings; our most interesting results in Part I are for this class. The easiest result is when

the cells are *promised* to be axis-aligned boxes in $[n]^d$: using binary search, the cells can be learned exactly. More interesting is when the clusterings are *not* promised to be boxes, but we only demand that the tester pass the box clusters, so that it detects if the clustering is too far away from a box clustering. In the continuous domain $[0,1]^d$, we show that, as long as the clustering is promised to be *convex*, the algorithm can either learn a good enough approximation of the cells to test uniformity, or it can reject the clustering. We get a similar result for the discrete domain $[n]^2$. Crucially, the algorithm is not required to exactly learn the cells, or even verify that they are exactly boxes, which would be expensive.

Threshold metrics. A simple but helpful example to understand the model, is that we can apply our general result using *threshold ℓ_p metrics*, where $\mathsf{dist}(x,y) = \max\{\ell_p(x,y), t\}$ for some threshold t. This allows to interpolate between the EMD_{ℓ_p} metric and TV distance (see full version of the paper).

Techniques. We consider the problem itself to be the main contribution of Part I since it was not clear to us in advance that adversarial clusterings allow for any interesting algorithmic tasks, and the problem requires careful definitions. First, since we include results for both identity and uniformity testing, let us note that the standard reduction from identity testing to uniformity testing [27] does *not* hold in the confused collector model[3].

Our algorithms will require a variety of techniques. The first step of the algorithm is to verify the HPLD condition, which is done by repeatedly sampling a point $x \sim \mu$ from the clustered-sample oracle, and using label queries to check if the diameter of its (unknown) cell is large. This depends on the geometry of the cells: we use either ad-hoc algorithms or, for convex cells in $[0,1]^d$, an application of convex optimization [31].

The second step of the identity and uniformity testing algorithms is to sample points $x \sim \mu$ from the clustered-sample oracle and discover an approximation of its cluster $\Gamma_{\gamma(x)}$. We may then simulate a sample from an auxiliary $\mu^\bullet$ distribution, by resampling from $\Gamma_{\gamma(x)}$ according to the *target* (known) distribution ν. We then compare the target ν to the auxiliary distribution $\mu^\bullet$ using an EMD tester similar to that of [22], which reduces to standard identity and equivalence testing bounds for TV distance [17, 37]. Cell discovery depends on the geometry of both the *promised* clusterings, and of the *good* clusterings; the most illustrative examples are when the universe $\mathcal{U}$ is promised to be either connected and convex in the grid $[n]^2$, or convex in $[0,1]^d$, and the "good" clusterings $\mathcal{G}$ are the decision trees (axis-aligned boxes). In each case, we can provide sample access to the auxiliary distribution $\mu^\bullet$ *without* exactly learning the cells, by making use of the algorithm's ability to output CLUSTER-REJECT if the cell is too far away from being a box. These examples show, in particular, that the cluster-rejection ability of the algorithm can be more powerful and interesting than simply testing the HPLD condition.

1.2 Part II: Random Clustering

If the confused collector has a *random* clustering instead of an adversarial clustering, one hopes to improve upon the results in Part I, and we show that this is true for certain random clusterings. We now wish for the tester to be correct with high probability, over

[3] A "reduction" in this model would require transforming one (unknown) clustering of the domain into a clustering of another domain (with simulated label queries), and samples from the clustered distribution into samples from a different clustered distribution. It is possible to transform the domain in a way that follows the reduction of [27] but, due to the clustering, it does not actually change the observed samples at all.

both the samples *and* the clustering. Specifically, for a property $\mathcal{P}$, input μ, parameter δ, and distribution $\mathcal{D}$ over the class $\mathcal{U}$ of clusterings, we require that the testing algorithm A satisfies:

- If $\mu \in \mathcal{P}$, then $\mathbb{P}_{(\mathbf{\Gamma},\mathbf{rep})\sim\mathcal{D};\ A}[A \text{ accepts}] \geq 1-\delta$; and
- If μ is ϵ-far from $\mathcal{P}$, then $\mathbb{P}_{(\mathbf{\Gamma},\mathbf{rep})\sim\mathcal{D};\ A}[A \text{ rejects}] \geq 1-\delta$.

It may sometimes be natural to allow the algorithm to CLUSTER-REJECT, but this will not be necessary at present. When the algorithm is not allowed to CLUSTER-REJECT, standard probability boosting techniques *do not work*: the algorithm has no control over the clustering, which is fixed, and therefore the error probability depends on the distribution over clusterings.

We will focus on random clusterings defined as follows. The domain is $[n]$, which we think of as vertices of a path (or cycle) $G = ([n], E)$. For parameter $\rho \in (0, 1]$, which we call the *resolution*, the distribution $\mathcal{U}_\rho$ over clusterings is defined by taking a random subgraph $\boldsymbol{H}$ of G where each edge is deleted independently with probability ρ, and the vertices are clustered by their connected components in $\boldsymbol{H}$. In other words, each consecutive pair of the domain is "separated" into different cells with probability ρ, so that the resolution ρ controls the granularity of the clustering.

We have two motivations for this choice of random clustering. First, it captures a type of "environmental randomness" in the collection of samples, like the example where fossils from different years can be distinguished only if a random geological event occurred between those years. We think of "environmental randomness" as being uncorrelated with the input probability distribution μ, whereas other natural random clusterings (e.g. a machine-learning classifer) may depend on μ. The one-dimensional clusterings we study are a simple and natural starting point for understanding how environmental randomness affects distribution testing tasks.

Second, these random clusterings occur in the study of certain distribution-free property testing problems. These problems are outside the scope of the current paper and will be explained formally in future work[4], but we give a simplified description in Section 1.3.

1.2.1 Results

We give results for testing *uniformity* of distributions under these random clusters, where the algorithm should accept the uniform distribution over $[n]$ and reject any distribution over $[n]$ that is ϵ-far from uniform in TV distance. From Part I, we obtain a tester using $\widetilde{O}(\frac{1}{\epsilon^2})$ samples and $\widetilde{O}(\frac{1}{\epsilon^2})$ queries under the EMD distance, but we now hope to use the TV distance (which corresponds roughly to testing under EMD with parameter $\approx \epsilon/n$). The optimal sample complexity for testing uniformity in the standard distribution testing model is $\Theta(\sqrt{n}/\epsilon^2)$ [34, 37].

Naïve benchmark algorithm. A reasonable algorithm that one might first propose is as follows. Suppose that the tester uses queries to exactly learn the clustering (Γ, rep) of the domain $[n]$. It may then define ν^* as the distribution over the representatives of the clusters $r_i := \mathsf{rep}(\Gamma_i)$ obtained by sampling x according to the uniform distribution over $[n]$ and then taking the representative of its cluster, $\mathsf{rep}(\Gamma_{\gamma(x)})$. Define μ^* as the distribution received from the confused collector, so that μ^* is the distribution over representatives $\mathsf{rep}(\Gamma_{\gamma(x)})$ when $x \sim \mu$ is sampled from the unknown input distribution. Then the algorithm runs an identity test on μ^* and ν^*.

[4] A partial treatment of this occurrence is given in a preprint [23] which contains some results from Part II of this paper, and will be elaborated upon in future work.

A back-of-the-envelope calculation of the complexity of this algorithm is as follows. The expected number of clusters is $\Theta(\rho n)$, so we require at most $O(\rho n \log n)$ queries to learn them with binary search, and the identity test will be performed on a domain of size $O(\rho n)$. When μ goes from the original distribution to the clustered distribution μ^*, it begins with TV distance ϵ from uniform, and might shrink to TV distance $\rho\epsilon$ from the target ν^*, forcing us to set the distance parameter for the identity test no larger than $\rho\epsilon$. (An example of this shrinkage is obtained by adding ϵ/n mass to each odd element of the uniform distribution, and subtracting ϵ/n from each even element, so that the $\pm\epsilon/n$ perturbations in each cluster balance out, except for an expected surplus of $\approx \pm\epsilon/2n$ for each of the ρn clusters.) Plugging in the identity testing bound, we get sample complexity $O\left(\frac{\sqrt{\rho n}}{\rho^2\epsilon^2}\right) = O\left(\frac{\sqrt{n}}{\rho^{3/2}\epsilon^2}\right)$ and query complexity $O(\rho n \log n)$.

Our first result, which is the main technical challenge of this paper, shows that we can in fact get the same bound on the sample complexity while using *zero* queries. This is important because, for some of the "environmental randomness" and property testing motivations we are interested in, it may not be possible to pose queries to the confused collector.

▶ **Theorem 4** (Main theorem; see full version for specific requirements on ρ, ϵ.)**.** *Let G be a path or cycle on n vertices and let $\rho, \epsilon \in (0, 1]$ satisfy $\rho \geq \widetilde{\Omega}(n^{-1/5}\epsilon^{-4/5})$. Then ϵ-testing uniformity under TV distance with the confused collector can be done using $\widetilde{O}\left(\frac{\sqrt{n}}{\rho^{3/2}\epsilon^2}\right)$ samples and zero queries.*

Since the algorithm has no control over the clustering, improvements to the error probability can be achieved by improving the resolution parameter. The proof is summarized in Section 1.2.2 below. It does not use a reduction to identity testing, and is instead a direct analysis of a generalization of the standard uniformity tester, which requires a new technical lemma on the concentration of random quadratic forms. The cycle has a cleaner analysis, but the path and cycle cases do not appear to directly reduce to one another, which is why both are included in the theorem.

With the sample complexity of the natural benchmark algorithm matched by a zero-query algorithm, one may wonder if queries can still be helpful, but it is not clear how to improve the algorithm or analysis of the benchmark: using the instance-optimal tester of [37] does not immediately improve the analysis. However, we show it is indeed possible to improve the sample complexity when queries are allowed. The algorithm is simple but, to us, much less natural than the benchmark. It is the same as the benchmark algorithm except that it tests identity *only on the singleton clusters* (i.e. clusters of size 1), crucially using the instance-optimal algorithm of [37].

▶ **Theorem 5.** *Let G be a path or cycle on n vertices, and let ρ, ϵ satisfy $\rho \geq \Omega((\epsilon n)^{-1/4})$. Then testing uniformity under TV distance with the confused collector requires $O\left(\frac{\sqrt{n}}{\rho\epsilon^2}\right)$ samples and $O(\rho n \log n)$ queries.*

The natural parameter regime is where $\rho = o(1)$; for the allowed constants δ in the theorem, setting $\rho = n^{-\delta}$ produces a sublinear $O(n^{1-\delta}\log n)$ query complexity.

1.2.2 Proof Overview

We now give an overview of the proof of our main technical Theorem 4, which shows that there is a zero-query tester that matches the sample complexity of the benchmark algorithm. Let us review the standard uniformity tester [28, 20] (see also [10]). Let μ be the input distribution over $[n]$. For a sample S of size m, let X_i be the multiplicity of element i in S. The tester

counts the number of "collisions" in the sample: it computes $Y := \frac{1}{m(m-1)} \sum_{i=1}^{n} X_i(X_i - 1)$, and rejects if this is too large. This works because $\mathbb{E}[\boldsymbol{Y}] = \mu^\top \mu = \|\mu\|_2^2$, which is large when μ is far from uniform. Now we describe the zero-query tester for the confused collector. For input distribution μ on domain $\mathbb{Z}_n$ (which are the vertices of the path or cycle), we use the standard Poissonization technique, so that element j occurs in the sample with multiplicity $\boldsymbol{T}_j \sim \mathsf{Poi}(m\mu_j)$ independently of the other elements. Recall that $\boldsymbol{H}$ is the random subgraph of the path or cycle that determines the clusters of the domain, and redefine $\boldsymbol{X}_i$ as the number of sample points contained in the i^{th} connected component of $\boldsymbol{H}$, which the tester cannot distinguish: the $\boldsymbol{X}_i$ variables remain Poisson, but they are not independent. The tester computes a "collision count", as in the standard algorithm:

$$\boldsymbol{Y} := \frac{1}{m^2} \sum_i \boldsymbol{X}_i(\boldsymbol{X}_i - 1) = \frac{1}{m^2} \left(\boldsymbol{T}^\top \boldsymbol{\Phi} \boldsymbol{T} - \|\boldsymbol{T}\|_1 \right) ,$$

where $\boldsymbol{\Phi}$ is the random Boolean matrix with $\boldsymbol{\Phi}_{i,j} = 1$ iff i, j belong to the same connected component of $\boldsymbol{H}$. The expected value is $\mathbb{E}[\boldsymbol{Y}] = \mu^\top \phi \mu$ where $\phi = \mathbb{E}[\boldsymbol{\Phi}]$, and we show that this is again large when μ is far from uniform, using spectral analysis of the matrix ϕ which is either Toeplitz (for paths) or circulant (for cycles).

To complete the analysis, we require concentration of measure for the random quadratic form $\boldsymbol{T}^\top \boldsymbol{\Phi} \boldsymbol{T} - \|\boldsymbol{T}\|_1$. This is similar to Hanson-Wright inequalities, except that Hanson-Wright inequalities apply to fixed matrices Φ whereas our matrix $\boldsymbol{\Phi}$ is random. We prove the following concentration inequality in terms of a concentration measure $\chi(\mu)$, which roughly satisfies $\chi(\mu) \lesssim \|\mu\|_\infty / \rho$ (but is less stringent in general) and which the algorithm can separately test:

▶ **Lemma 6** (Informal; see full version). *Let $\delta \in (0, 1)$ be a constant. Let μ be a probability distribution over $\mathbb{Z}_n$, and suppose $\rho \geq \Omega(n^{-\delta})$ and $m \leq \mathrm{poly}(n)$. Then for all $\tau > 0$,*

$$\mathbb{P}\left[|\boldsymbol{Y} - \mathbb{E}[\boldsymbol{Y}]| \geq \tau \right] \leq \frac{\|\mu\|_2^2}{\rho \tau^2} \cdot \max\{\chi(\mu), 1/m\}^2 \cdot O(\log^2 n) .$$

1.3 Discussion and Open Problems

Related work. [25] studied statistical learning (e.g. Gaussian mean estimation) in a model with similar motivation to ours. The difference is that [25] chooses a random clustering for each sample point independently and provides an explicit representation of the cell containing each point, whereas in our model the sample points are all labeled by the *same* clustering, the algorithm is not given the cell explicitly, and the algorithm can make label queries. If we suppose that the input distribution μ is held by *multiple* confused collectors, and each sample point is labeled by a *random* collector, then we nearly recover the model of [25], except that we allow label queries and do not receive explicit representations of the cells. In this interpretation, [25] require that the collectors jointly hold an "information-preserving" clustering (which approximately preserves TV distance between any pair of distributions), which is unnecessary in our model.

Our model also shares some conceptual similarities with the "huge object" model [29], where the algorithm must test properties of a distribution over $\{0, 1\}^n$ by taking samples and, for each sampled element $x \in \{0, 1\}^n$, querying a subset of its bits. In both models, the algorithm has incomplete information about the sample, and the goal is to test with respect to the earth-mover distance. The difference is that, in the huge object model, the algorithm has incomplete information about the sample because the objects are too large to observe entirely, but perfect knowledge could be obtained if enough queries are used. In the confused collector model, the incomplete information is due to imprecise classification of the sample and may not be possible to acquire.

Questions. We would like to know more about how to manage *multiple* confused collectors. One may consider a setting where the input μ is held by multiple collectors (as above), or where the collectors hold separate inputs $\mu_1, \ldots, \mu_t$ (as studied in [32, 33, 7] for the standard model). Each sample point could be drawn from a random collector, or a chosen collector; or one might pay cost k to get a single sample point labeled by k of the collectors. How efficiently can one detect and select collectors with complementary expertise, and combine their expertise?

In Part I, we reduce identity or equivalence testing in the clustered domain, to the identity or equivalence testing in the standard model, but this is not as easy for properties like monotonicity (e.g. [8, 5]) or k-histograms (e.g. [30, 11, 13, 14]), which are not preserved by clustering; it would be interesting to study these.

An open problem of [22] is to find tight bounds on the sample complexity for estimating EMD between distributions on $[0,1]^d$. This appears to be open still, and it would be helpful in the confused collector model to have optimal bounds on tolerant testing under EMD.

The random clustering model in Part II is tailored to the path and cycle, and may not be sensible for more general graphs. It would be interesting to know which natural random clusterings of graphs allow for efficient zero-query algorithms, as in Theorem 4. We also wonder what properties of distributions, beyond uniformity, admit testers (with and without queries) under TV distance; e.g. is identity testing possible for some non-trivial class of target distributions ν?

Finally, it would be interesting to investigate instance-optimal identity testing [37] (see also [21, 9]) in the confused collector model, since cell discovery etc. can be tailored to the known distribution.

Distribution-free property testing. One of our original motivations for studying the confused collector model is its relation to some distribution-free property testing problems. This will be the subject of future work and is outside the scope of this paper, but the connection boils down to this:

In distribution-free testing of functions $f : \mathcal{X} \to \{0,1\}$, it becomes necessary to test a joint property of the input distribution and the set $f^{-1}(1)$, which we can think of as a union of connected components; if $f : \mathbb{R} \to \{0,1\}$, then $f^{-1}(1)$ is a union of intervals. But, given two samples $x, y \in f^{-1}(1)$, it is not possible to know whether they came from the same or different intervals, unless another sample z occurs in $f^{-1}(0)$ between x, y. Testing a joint property of the input distribution with $f^{-1}(1)$ therefore must be done with the tester seeing only a "coarsening" of $f^{-1}(1)$. The confused collector model allows us to study this phenomenon in a simpler setting from first principles, and there is in fact a formal connection between the problem just described and the random clusterings in Part II of this paper, which will be elaborated in future work.

References

1 Jayadev Acharya, Clément Canonne, Cody Freitag, and Himanshu Tyagi. Test without trust: Optimal locally private distribution testing. In *Proceedings, International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2067–2076. PMLR, 2019.

2 Jayadev Acharya, Clément L Canonne, Cody Freitag, Ziteng Sun, and Himanshu Tyagi. Inference under information constraints iii: Local privacy constraints. *IEEE Journal on Selected Areas in Information Theory*, 2(1):253–267, 2021.

3 Jayadev Acharya, Clément L Canonne, and Himanshu Tyagi. Inference under information constraints: Lower bounds from chi-square contraction. In *Proceedings, Conference on Learning Theory (COLT)*, pages 3–17. PMLR, 2019.

4 Jayadev Acharya, Clément L Canonne, and Himanshu Tyagi. Inference under information constraints ii: Communication constraints and shared randomness. *IEEE Transactions on Information Theory*, 66(12):7856–7877, 2020.

5 Maryam Aliakbarpour, Themis Gouleakis, John Peebles, Ronitt Rubinfeld, and Anak Yodpinyanee. Towards testing monotonicity of distributions over general posets. In *Proceedings, Conference on Learning Theory (COLT)*, pages 34–82. PMLR, 2019.

6 Maryam Aliakbarpour, Ravi Kumar, and Ronitt Rubinfeld. Testing mixtures of discrete distributions. In *Proceedings, Conference on Learning Theory (COLT)*, pages 83–114. PMLR, 2019.

7 Maryam Aliakbarpour and Sandeep Silwal. Testing properties of multiple distributions with few samples. In *Proceedings, Innovations in Theoretical Computer Science (ITCS)*, 2020.

8 Tugkan Batu, Ravi Kumar, and Ronitt Rubinfeld. Sublinear algorithms for testing monotone and unimodal distributions. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 381–390, 2004.

9 Eric Blais, Clément L Canonne, and Tom Gur. Distribution testing lower bounds via reductions from communication complexity. *ACM Transactions on Computation Theory*, 11(2):1–37, 2019.

10 Clément Canonne. Topics and techniques in distribution testing: A biased but representative sample. *Foundations and Trends in Communications and Information Theory*, 19(6):1032–1198, 2022.

11 Clément L Canonne. Are few bins enough: Testing histogram distributions. In *Proceedings, ACM Symposium on Principles of Database Systems (PODS)*, pages 455–463, 2016.

12 Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? *Theory of Computing*, pages 1–100, 2020.

13 Clément L Canonne, Ilias Diakonikolas, Themis Gouleakis, and Ronitt Rubinfeld. Testing shape restrictions of discrete distributions. *Theory of Computing Systems*, 62(1):4–62, 2018.

14 Clément L Canonne, Ilias Diakonikolas, Daniel Kane, and Sihan Liu. Nearly-tight bounds for testing histogram distributions. *Proceedings, Advances in Neural Information Processing Systems (NeurIPS)*, 35:31599–31611, 2022.

15 Clément L Canonne and Karl Wimmer. Testing data binnings. In *Proceedings of APPROX/RANDOM*. Schloss Dagstuhl-Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing, 2020.

16 Clément L Canonne and Karl Wimmer. Identity testing under label mismatch. In *Proceedings, International Symposium on Algorithms and Computation (ISAAC)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

17 Siu-On Chan, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1193–1203. SIAM, 2014.

18 Constantinos Daskalakis, Themis Gouleakis, Chistos Tzamos, and Manolis Zampetakis. Efficient statistics, in high dimensions, from truncated samples. In *Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 639–649. IEEE, 2018.

19 Anindya De, Shivam Nadimpalli, and Rocco A Servedio. Testing convex truncation. In *Proceedings, ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4050–4082. SIAM, 2023.

20 Ilias Diakonikolas, Themis Gouleakis, John Peebles, and Eric Price. Collision-based testers are optimal for uniformity and closeness. *Chicago Journal of Theoretical Computer Science*, 1:1–21, 2019.

21 Ilias Diakonikolas and Daniel M Kane. A new approach for testing properties of discrete distributions. In *Proceedings, IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 685–694. IEEE, 2016.

22 Khanh Do Ba, Huy L Nguyen, Huy N Nguyen, and Ronitt Rubinfeld. Sublinear time algorithms for earth mover's distance. *Theory of Computing Systems*, 48:428–442, 2011.

23 Renato Ferreira Pinto Jr. and Nathaniel Harms. Distribution testing under the parity trace, 2023. arXiv:2304.01374. `arXiv:arXiv:2304.01374`.

24 Renato Ferreira Pinto Jr. and Nathaniel Harms. Distribution testing with a confused collector. *arXiv*, 2023. `arXiv:2311.1424`.

25 Dimitris Fotakis, Alkis Kalavasis, Vasilis Kontonis, and Christos Tzamos. Efficient algorithms for learning from coarse labels. In *Proceedings, Conference on Learning Theory (COLT)*, pages 2060–2079. PMLR, 2021.

26 Marco Gaboardi and Ryan Rogers. Local private hypothesis testing: Chi-square tests. In *Proceedings, International Conference on Machine Learning (ICML)*, pages 1626–1635. PMLR, 2018.

27 Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. In *Computational Complexity and Property Testing: On the Interplay Between Randomness and Computation*. Springer, 2020. ECCC TR16-015. `doi:10.1007/978-3-030-43662-9_10`.

28 Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 68–75. Springer, 2011. `doi:10.1007/978-3-642-22670-0_9`.

29 Oded Goldreich and Dana Ron. Testing distributions of huge objects. In *Proceedings, Innovations in Theoretical Computer Science (ITCS)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

30 Piotr Indyk, Reut Levi, and Ronitt Rubinfeld. Approximating and testing k-histogram distributions in sub-linear time. In *Proceedings, ACM Symposium on Principles of Database Systems (PODS)*, pages 15–22, 2012.

31 Yin Tat Lee, Aaron Sidford, and Santosh S Vempala. Efficient convex optimization with oracles. In *Building Bridges II: Mathematics of László Lovász*, pages 317–335. Springer, 2020.

32 Reut Levi, Dana Ron, and Ronitt Rubinfeld. Testing properties of collections of distributions. *Theory of Computing*, 9(1):295–347, 2013.

33 Reut Levi, Dana Ron, and Ronitt Rubinfeld. Testing similar means. *SIAM Journal on Discrete Mathematics*, 28(4):1699–1724, 2014.

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

35 Ronitt Rubinfeld and Arsen Vasilyan. Testing distributional assumptions of learning algorithms. In *Proceedings, ACM Symposium on Theory of Computing (STOC)*. ACM, 2023.

36 Or Sheffet. Locally private hypothesis testing. In *Proceedings, International Conference on Machine Learning (ICML)*, pages 4605–4614. PMLR, 2018.

37 Gregory Valiant and Paul Valiant. An automatic inequality prover and instance optimal identity testing. *SIAM Journal on Computing*, 46(1):429–455, 2017.