THEORY OF COMPUTING, Volume 10 (20), 2014, pp. 535–570
www.theoryofcomputing.org

Learning k-Modal Distributions via Testing

Constantinos Daskalakis* Ilias Diakonikolas† Rocco A. Servedio‡

Received April 11, 2013; Revised November 9, 2014; Published December 31, 2014

Abstract: A k-modal probability distribution over the discrete domain $\{1,\dots,n\}$ is one whose histogram has at most k "peaks" and "valleys." Such distributions are natural generalizations of monotone ($k=0$) and unimodal ($k=1$) probability distributions, which have been intensively studied in probability theory and statistics.

In this paper we consider the problem of *learning* (i. e., performing density estimation of) an unknown k-modal distribution with respect to the L_1 distance. The learning algorithm is given access to independent samples drawn from an unknown k-modal distribution p, and it must output a hypothesis distribution $\widehat{p}$ such that with high probability the total variation distance between p and $\widehat{p}$ is at most ε. Our main goal is to obtain *computationally efficient* algorithms for this problem that use (close to) an information-theoretically optimal number of samples.

We give an efficient algorithm for this problem that runs in time $\mathrm{poly}(k, \log(n), 1/\varepsilon)$. For $k \le \tilde{O}(\log n)$, the number of samples used by our algorithm is very close (within an

A preliminary version of this work appeared in the Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2012) [8].

*costis@csail.mit.edu. Research supported by NSF CAREER award CCF-0953960 and by a Sloan Foundation Fellowship.

†ilias.d@ed.ac.uk. Most of this research was done while the author was at UC Berkeley supported by a Simons Postdoctoral Fellowship. Some of this work was done at Columbia University, supported by NSF grant CCF-0728736, and by an Alexander S. Onassis Foundation Fellowship.

‡rocco@cs.columbia.edu. Supported by NSF grants CNS-0716245, CCF-0915929, and CCF-1115703.

ACM Classification: F.2.2, G.3

AMS Classification: 68W20, 68Q25, 68Q32

Key words and phrases: computational learning theory, learning distributions, k-modal distributions

 DOI: 10.4086/toc.2014.v010a020

$\tilde{O}(\log(1/\varepsilon))$ factor) to being information-theoretically optimal. Prior to this work computationally efficient algorithms were known only for the cases $k = 0, 1$ (Birgé 1987, 1997).

A novel feature of our approach is that our learning algorithm crucially uses a new algorithm for *property testing of probability distributions* as a key subroutine. The learning algorithm uses the property tester to efficiently decompose the k-modal distribution into k (near-)monotone distributions, which are easier to learn.

1 Introduction

This paper considers a natural unsupervised learning problem involving *k-modal* distributions over the discrete domain $[n] = \{1, \dots, n\}$. A distribution is k-modal if the plot of its probability density function (pdf) has at most k "peaks" and "valleys" (see Section 2.1 for a precise definition). Such distributions arise both in theoretical (see, e. g., [7, 19, 6]) and applied (see, e. g., [20, 1, 13]) research; they naturally generalize the simpler classes of monotone ($k = 0$) and unimodal ($k = 1$) distributions that have been intensively studied in probability theory and statistics (see the discussion of related work below).

Our main aim in this paper is to give an efficient algorithm for *learning* an unknown k-modal distribution p to total variation distance ε, given access only to independent samples drawn from p. As described below there is an information-theoretic lower bound of $\Omega(k\log(n/k)/\varepsilon^3)$ samples for this learning problem, so an important goal for us is to obtain an algorithm whose sample complexity is as close as possible to this lower bound. An equally important goal is for our algorithm to be computationally efficient, i. e., to run in time polynomial in the size of its input sample. Our main contribution in this paper is a computationally efficient algorithm that has nearly optimal sample complexity for small (but super-constant) values of k.

1.1 Background and relation to previous work

There is a rich body of work in the statistics and probability literatures on estimating distributions under various kinds of "shape" or "order" restrictions. In particular, many researchers have studied the risk of different estimators for monotone ($k = 0$) and unimodal ($k = 1$) distributions; see for example the works of [23, 26, 17, 3, 4, 5], among many others. These and related papers from the probability/statistics literature mostly deal with information-theoretic upper and lower bounds on the sample complexity of learning monotone and unimodal distributions. In contrast, a central goal of the current work is to obtain *computationally efficient* learning algorithms for larger values of k.

It should be noted that some of the works cited above do give efficient algorithms for the cases $k = 0$ and $k = 1$; in particular we mention the results of Birgé [4, 5], which give computationally efficient $O(\log(n)/\varepsilon^3)$-sample algorithms for learning unknown monotone or unimodal distributions over $[n]$ respectively. (Birgé [3] also showed that this sample complexity is asymptotically optimal, as we discuss below; we describe the algorithm of [4] in more detail in Section 2.2, and indeed use it as an ingredient of our approach throughout this paper.) However, for these relatively simple $k = 0, 1$ classes of distributions the main challenge is in developing sample-efficient estimators, and the algorithmic aspects are typically rather straightforward (as is the case in [4]). In contrast, much more challenging and interesting algorithmic issues arise for the general values of k which we consider here.

1.2 Our results

Our main result is a highly efficient algorithm for learning an unknown k-modal distribution over $[n]$:

Theorem 1.1. *Let p be any unknown k-modal distribution over $[n]$. There is an algorithm that uses*[1]

$$\left(\frac{k\log(n/k)}{\varepsilon^3}+\frac{k^2}{\varepsilon^3}\cdot\log\frac{k}{\varepsilon}\cdot\log\log\frac{k}{\varepsilon}\right)\cdot\tilde{O}(\log(1/\delta))$$

samples from p, runs for $\mathrm{poly}(k,\log n,1/\varepsilon,\log(1/\delta))$ *bit operations, and with probability* $1-\delta$ *outputs a (succinct description of a) hypothesis distribution h over $[n]$ such that the total variation distance between p and h is at most ε.*

As alluded to earlier, Birgé [3] gave a sample complexity lower bound for learning monotone distributions. The lower bound in [3] is stated for continuous distributions but the arguments are easily adapted to the discrete case; [3] shows that (for $\varepsilon \geq 1/n^{\Omega(1)}$)[2] any algorithm for learning an unknown monotone distribution over $[n]$ to total variation distance ε must use $\Omega(\log(n)/\varepsilon^3)$ samples. By a simple construction which concatenates k copies of the monotone lower bound construction over intervals of length n/k, using the monotone lower bound it is possible to show:

Proposition 1.2. *Any algorithm for learning an unknown k-modal distribution over $[n]$ to variation distance ε (for $\varepsilon \geq 1/n^{\Omega(1)}$) must use $\Omega(k\log(n/k)/\varepsilon^3)$ samples.*

Thus our learning algorithm is nearly optimal in its sample complexity; more precisely, for $k \leq \tilde{O}(\log n)$ (and ε as bounded above), our sample complexity in Theorem 1.1 is asymptotically optimal up to a factor of $\tilde{O}(\log(1/\varepsilon))$. Since each draw from a distribution over $[n]$ is a $\log(n)$-bit string, Proposition 1.2 implies that the running time of our algorithm is optimal up to polynomial factors. As far as we are aware, prior to this work no learning algorithm for k-modal distributions was known that simultaneously had $\mathrm{poly}(k,\log n)$ sample complexity and even running time $q(n)$ for a fixed polynomial $q(n)$ (where the exponent does not depend on k).

1.3 Our approach

As mentioned in Section 1.1 Birgé, gave a highly efficient algorithm for learning a *monotone* distribution in [4]. Since a k-modal distribution is simply a concatenation of $k+1$ monotone distributions (first non-increasing, then non-decreasing, then non-increasing, etc.), it is natural to try to use Birgé's algorithm as a component of an algorithm for learning k-modal distributions, and indeed this is what we do.

The most naive way to use Birgé's algorithm would be to guess all possible $\binom{n}{k}$ locations of the k "modes" of p. While such an approach can be shown to have good sample complexity, the resulting $\Omega(n^k)$ running time is grossly inefficient. A "moderately naive" approach, which we analyze in Section 3.1, is to partition $[n]$ into roughly k/ε intervals each of weight roughly ε/k, and run Birgé's algorithm

[1] We write $\tilde{O}(\cdot)$ to hide factors which are poly-logarithmic in the argument to $\tilde{O}(\cdot)$; thus for example $\tilde{O}(a\log b)$ denotes a quantity which is $O((a\log b)\cdot(\log(a\log b))^c)$ for some absolute constant c.

[2] For ε sufficiently small the generic upper bound of Fact A.1, which says that any distribution over $[n]$ can be learned to variation distance ε using $O(n/\varepsilon^2)$ samples, provides a better bound.

separately on each such interval. Since the target distribution is k-modal, at most k of the intervals can be non-monotone; Birgé's algorithm can be used to obtain an ε-accurate hypothesis on each monotone interval, and even if it fails badly on the (at most) k non-monotone intervals, the resulting total contribution towards the overall error from those failures is at most $O(\varepsilon)$. This approach is much more efficient than the totally naive approach, giving running time polynomial in k, $\log n$, and $1/\varepsilon$, but its sample complexity turns out to be polynomially worse than the $O(k\log(n)/\varepsilon^3)$ that we are shooting for. (Roughly speaking, this is because the approach involves running Birgé's $O(\log(n)/\varepsilon^3)$-sample algorithm $\Omega(k/\varepsilon)$ times, so it uses at least $k\log(n)/\varepsilon^4$ samples.)

Our main learning result is achieved by augmenting the "moderately naive" algorithm sketched above with a new *property testing* algorithm. Unlike a learning algorithm, a property testing algorithm for probability distributions need not output a high-accuracy hypothesis; instead, it has the more modest goal of successfully (with high probability) distinguishing between probability distributions that have a given property of interest, versus distributions that are far (in total variation distance) from every distribution that has the property. See [16, 25, 14] for broad overviews of property testing.

We give a property testing algorithm for the following problem: given samples from a distribution p over $[n]$ which is promised to be k-modal, output "yes" (with high probability) if p is *monotone* and "no" (with high probability) if p is ε-far in total variation distance from every monotone distribution. Crucially, our testing algorithm uses $O(k/\varepsilon^2)$ samples *independent of* n for this problem. Roughly speaking, by using this algorithm $O(k/\varepsilon)$ times we are able to identify $k+1$ intervals that (i) collectively contain almost all of p's mass, and (ii) are each (close to) monotone and thus can be handled using Birgé's algorithm. Thus the overall sample complexity of our approach is (roughly) $O(k^2/\varepsilon^3)$ (for the $O(k/\varepsilon)$ runs of the tester) plus $O(k\log(n)/\varepsilon^3)$ (for the k runs of Birgé's algorithm), which gives Theorem 1.1 and is very close to optimal for k not too large.

1.4 Discussion

Our learning algorithm highlights a novel way that property testing algorithms can be useful for learning. Much research has been done on understanding the relation between property testing algorithms and learning algorithms, see, e. g., [16, 18] and the lengthy survey [24]. As Goldreich has noted [15], an often-invoked motivation for property testing is that (inexpensive) testing algorithms can be used as a "preliminary diagnostic" to determine whether it is appropriate to run a (more expensive) learning algorithm. In contrast, in this work we are using property testing rather differently, as an inexpensive way of decomposing a "complex" object (a k-modal distribution) which we do not *a priori* know how to learn, into a collection of "simpler" objects (monotone or near-monotone distributions) which can be learned using existing techniques. We are not aware of prior learning algorithms that successfully use property testers in this way; we believe that this high-level approach to designing learning algorithms, by using property testers to decompose "complex" objects into simpler objects that can be efficiently learned, may find future applications elsewhere.

2 Preliminaries

2.1 Notation and problem statement

For $n \in \mathbb{Z}_+$, denote by $[n]$ the set $\{1,\dots,n\}$; for $i,j \in \mathbb{Z}_+$, $i \le j$, denote by $[i,j]$ the set $\{i,i+1,\dots,j\}$. We write $v(i)$ to denote the i-th element of vector $v \in \mathbb{R}^n$. For $v = (v(1),\dots,v(n)) \in \mathbb{R}^n$ denote by

$$\|v\|_1 = \sum_{i=1}^{n} |v(i)|$$

its L_1-norm.

We consider discrete probability distributions over $[n]$, which are functions $p : [n] \to [0,1]$ such that $\sum_{i=1}^{n} p(i) = 1$. For $S \subseteq [n]$ we write $p(S)$ to denote $\sum_{i\in S} p(i)$. For $S \subseteq [n]$, we write p_S to denote the *conditional distribution* over S that is induced by p. We use the notation P for the *cumulative distribution function (cdf)* corresponding to p, i. e., $P : [n] \to [0,1]$ is defined by $P(j) = \sum_{i=1}^{j} p(i)$.

A distribution p over $[n]$ is non-increasing (resp. non-decreasing) if $p(i+1) \le p(i)$ (resp. $p(i+1) \ge p(i)$), for all $i \in [n-1]$; p is *monotone* if it is either non-increasing or non-decreasing. We call a nonempty interval $I = [a,b] \subseteq [2,n-1]$ a *max-interval of* p if $p(i) = c$ for all $i \in I$ and $\max\{p(a-1), p(b+1)\} < c$; in this case, we say that the point a is a *left max point* of p. Analogously, a *min-interval of* p is an interval $I = [a,b] \subseteq [2,n-1]$ with $p(i) = c$ for all $i \in I$ and $\min\{p(a-1), p(b+1)\} > c$; the point a is called a *left min point* of p. If $I = [a,b]$ is either a max-interval or a min-interval (it cannot be both) we say that I is an *extreme-interval* of p, and a is called a *left extreme point* of p. Note that any distribution uniquely defines a collection of extreme-intervals (hence, left extreme points). We say that p is *k-modal* if it has at most k extreme-intervals. We write $\mathcal{D}_n$ (resp. $\mathcal{M}_n^k$) to denote the set of all distributions (resp. k-modal distributions) over $[n]$.

Let p,q be distributions over $[n]$ with corresponding cdfs P,Q. The *total variation distance* between p and q is

$$d_{\mathrm{TV}}(p,q) := \max_{S\subseteq[n]} |p(S) - q(S)| = (1/2)\cdot\|p-q\|_1 .$$

The *Kolmogorov distance* between p and q is defined as

$$d_{\mathrm{K}}(p,q) := \max_{j\in[n]} |P(j) - Q(j)| .$$

Note that $d_{\mathrm{K}}(p,q) \le d_{\mathrm{TV}}(p,q)$.

We will also need a more general distance measure that captures the above two metrics as special cases. Fix a family of subsets $\mathcal{A}$ over $[n]$. We define the *$\mathcal{A}$–distance* between p and q by

$$\|p-q\|_{\mathcal{A}} := \max_{A\in\mathcal{A}} |p(A) - q(A)| .$$

(Note that if $\mathcal{A} = 2^{[n]}$, the powerset of $[n]$, then the $\mathcal{A}$–distance is identified with the total variation distance, while when $\mathcal{A} = \{[1,j], j \in [n]\}$ it is identified with the Kolmogorov distance.) Also recall that the *VC–dimension* of $\mathcal{A}$ is the maximum size of a subset $X \subseteq [n]$ that is shattered by $\mathcal{A}$ (a set X is shattered by $\mathcal{A}$ if for every $Y \subseteq X$ some $A \in \mathcal{A}$ satisfies $A \cap X = Y$).

Learning k-modal Distributions. Given independent samples from an unknown k-modal distribution $p \in \mathcal{M}_n^k$ and $\varepsilon > 0$, the goal is to output a hypothesis distribution h such that with probability $1-\delta$ we have $d_{\mathrm{TV}}(p,h) \le \varepsilon$. We say that such an algorithm $\mathcal{A}$ *learns p to accuracy ε and confidence δ*. The parameters of interest are the number of samples and the running time required by the algorithm.

2.2 Basic tools

We recall some useful tools from probability theory.

The VC inequality. Given m independent samples $s_1, \dots, s_m$, drawn from $p : [n] \to [0,1]$, the *empirical distribution* $\widehat{p}_m : [n] \to [0,1]$ is defined as follows: for all $i \in [n]$,

$$\widehat{p}_m(i) = \frac{|\{j \in [m] \mid s_j = i\}|}{m}.$$

Fix a family of subsets $\mathcal{A}$ over $[n]$ of VC–dimension d. The *VC inequality* states that for $m = \Omega(d/\varepsilon^2)$, with probability $9/10$ the empirical distribution $\widehat{p}_m$ will be ε-close to p in $\mathcal{A}$-distance. This sample bound is asymptotically optimal.

Theorem 2.1 (VC inequality, [12, p.31])**.** *Let $\widehat{p}_m$ be an empirical distribution of m samples from p. Let $\mathcal{A}$ be a family of subsets of VC–dimension d. Then*

$$\mathbf{E}\left[\|p - \widehat{p}_m\|_{\mathcal{A}}\right] \le O(\sqrt{d/m}).$$

Uniform convergence. We will also use the following uniform convergence bound:

Theorem 2.2 ([12, p.17])**.** *Let $\mathcal{A}$ be a family of subsets over $[n]$, and $\widehat{p}_m$ be an empirical distribution of m samples from p. Let X be the random variable $\|p - \widehat{p}_m\|_{\mathcal{A}}$. Then we have*

$$\Pr\left[X - \mathbf{E}[X] > \eta\right] \le e^{-2m\eta^2}.$$

Our second tool, due to Birgé [4], provides a sample-optimal and computationally efficient algorithm to learn monotone distributions to ε-accuracy in total variation distance. Before we state the relevant theorem, we need a definition. We say that a distribution p is δ-close to being non-increasing (resp. non-decreasing) if there exists a non-increasing (resp. non-decreasing) distribution q such that $d_{\mathrm{TV}}(p,q) \le \delta$. We are now ready to state Birgé's result:

Theorem 2.3 ([4], Theorem 1)**.** *(semi-agnostic learner) There is an algorithm $\mathrm{L}^{\downarrow}$ with the following performance guarantee: Given m independent samples from a distribution p over $[n]$ which is opt-close to being non-increasing, $\mathrm{L}^{\downarrow}$ performs*

$$\tilde{O}\big(m \cdot \log n + m^{1/3} \cdot (\log n)^{5/3}\big)$$

bit operations and outputs a (succinct description of a) hypothesis distribution $\widetilde{p}$ over $[n]$ that satisfies

$$\mathbf{E}[d_{\mathrm{TV}}(\widetilde{p}, p)] \le 2 \cdot \mathsf{opt} + O\Big(\big(\log n/(m+1)\big)^{1/3}\Big).$$

The aforementioned algorithm partitions the domain $[n]$ in $O(m^{1/3} \cdot (\log n)^{2/3})$ intervals and outputs a hypothesis distribution that is uniform within each of these intervals.

By taking $m = \Omega(\log n/\varepsilon^3)$, one obtains a hypothesis such that $\mathbf{E}[d_{\mathrm{TV}}(\widetilde{p}, p)] \leq 2 \cdot \mathsf{opt} + \varepsilon$. We stress that Birgé's algorithm for learning non-increasing distributions [4] is in fact "semi-agnostic," in the sense that it also learns distributions that are close to being non-increasing; this robustness will be crucial for us later (since in our final algorithm we will use Birgé's algorithm on distributions identified by our tester, that are close to monotone but not necessarily perfectly monotone). This semi-agnostic property is not explicitly stated in [4] but it can be shown to follow easily from his results. We show how the semi-agnostic property follows from Birgé's results in Appendix A. Let $\mathrm{L}^{\uparrow}$ denote the corresponding semi-agnostic algorithm for learning non-decreasing distributions.

Our final tool is a routine to do *hypothesis testing*, i. e., to select a high-accuracy hypothesis distribution from a collection of hypothesis distributions one of which has high accuracy. The need for such a routine arises in several places; in some cases we know that a distribution is monotone, but do not know whether it is non-increasing or non-decreasing. In this case, we can run both algorithms $\mathrm{L}^{\uparrow}$ and $\mathrm{L}^{\downarrow}$ and then choose a good hypothesis using hypothesis testing. Another need for hypothesis testing is to "boost confidence" that a learning algorithm generates a high-accuracy hypothesis. Our initial version of the algorithm for Theorem 1.1 generates an ε-accurate hypothesis with probability at least $9/10$; by running it $O(\log(1/\delta))$ times using a hypothesis testing routine, it is possible to identify an $O(\varepsilon)$-accurate hypothesis with probability $1-\delta$. Routines of the sort that we require have been given in, e. g., [12] and [9]; we use the following theorem from [9]:

Theorem 2.4. *There is an algorithm* `Choose-Hypothesis`$^p(h_1, h_2, \varepsilon', \delta')$ *which is given sample access to p, two hypothesis distributions h_1, h_2 for p, an accuracy parameter ε', and a confidence parameter δ'. It makes $m = O(\log(1/\delta')/\varepsilon'^2)$ draws from p and returns a hypothesis $h \in \{h_1, h_2\}$. If one of h_1, h_2 has $d_{\mathrm{TV}}(h_i, p) \leq \varepsilon'$ then with probability $1-\delta'$ the hypothesis h that* `Choose-Hypothesis` *returns has $d_{\mathrm{TV}}(h, p) \leq 6\varepsilon'$.*

For the sake of completeness, we describe and analyze the `Choose-Hypothesis` algorithm in Appendix B.

3 Learning k-modal distributions

In this section, we present our main result: a nearly sample-optimal and computationally efficient algorithm to learn an unknown k-modal distribution. In Section 3.1 we present a simple learning algorithm with a suboptimal sample complexity. In Section 3.2 we present our main result which involves a property testing algorithm as a subroutine.

3.1 Warm-up: A simple learning algorithm

In this subsection, we give an algorithm that runs in time $\mathrm{poly}(k, \log n, 1/\varepsilon, \log(1/\delta))$ and learns an unknown k-modal distribution to accuracy ε and confidence δ. The sample complexity of the algorithm is essentially optimal as a function of k (up to a logarithmic factor), but suboptimal as a function of ε, by a polynomial factor.

In the following pseudocode we give a detailed description of the algorithm `Learn-kmodal-simple` (a precise description appears as Algorithm 1, below); the algorithm outputs an ε-accurate hypothesis with confidence $9/10$ (see Theorem 3.3). We explain how to boost the confidence to $1-\delta$ after the proof of the theorem.

Algorithm `Learn-kmodal-simple` works as follows: We start by partitioning the domain $[n]$ into consecutive intervals of mass "approximately ε/k." To do this, we draw $\Theta(k/\varepsilon^3)$ samples from p and greedily partition the domain into disjoint intervals of empirical mass roughly ε/k. (Some care is needed in this step, since there may be "heavy" points in the support of the distribution; however, we gloss over this technical issue for the sake of this intuitive explanation.) Note that we do *not* have a guarantee that *each* such interval will have true probability mass $\Theta(\varepsilon/k)$. In fact, it may well be the case that the additive error δ between the true probability mass of an interval and its empirical mass (roughly ε/k) is $\delta=\omega(\varepsilon/k)$. The error guarantee of the partitioning is more "global" in that the *sum* of these errors across all such intervals is at most ε. In particular, as a simple corollary of the VC inequality, we can deduce the following statement that will be used several times throughout the paper:

Fact 3.1. *Let p be any distribution over $[n]$ and $\widehat{p}_m$ be the empirical distribution of m samples from p. For*

$$m=\Omega\left((d/\varepsilon^2)\log(1/\delta)\right),$$

with probability at least $1-\delta$, for any *collection $\mathcal{J}$ of (at most) d disjoint intervals in $[n]$, we have that*

$$\sum_{J\in\mathcal{J}}|p(J)-\widehat{p}_m(J)|\le\varepsilon\,.$$

Proof. Note that

$$\sum_{J\in\mathcal{J}}|p(J)-\widehat{p}_m(J)|=2|p(A)-\widehat{p}_m(A)|\,, \tag{3.1}$$

where

$$A=\{J\in\mathcal{J}:p(J)>\widehat{p}_m(J)\}\,.$$

Since $\mathcal{J}$ is a collection of at most d intervals, it is clear that A is a union of at most d intervals. If $\mathcal{A}_d$ is the family of all unions of at most d intervals, then the right hand side of (3.1) is at most $2\|p-\widehat{p}_m\|_{\mathcal{A}_d}$. Since the VC–dimension of $\mathcal{A}_d$ is $2d$, Theorem 2.1 implies that the quantity (3.1) has expected value at most $\varepsilon/2$. The claim now follows by an application of Theorem 2.2 with $\eta=\varepsilon/2$. □

If this step is successful, we have partitioned the domain into a set of $O(k/\varepsilon)$ consecutive intervals of probability mass "roughly ε/k." The next step is to apply Birgé's monotone learning algorithm to each interval.

A caveat comes from the fact that not all such intervals are guaranteed to be monotone (or even close to being monotone). However, since our input distribution is assumed to be k-modal, all but (at most) k of these intervals are monotone. Call a non-monotone interval "bad." Since all intervals have empirical probability mass at most ε/k and there are at most k bad intervals, it follows from Fact 3.1 that these intervals contribute at most $O(\varepsilon)$ to the total mass. So even though Birgé's algorithm gives no guarantees for bad intervals, these intervals do not affect the error by more than $O(\varepsilon)$.

Let us now focus on the monotone intervals. For each such interval, we do not know if it is monotone increasing or monotone decreasing. To overcome this difficulty, we run both monotone algorithms $\mathrm{L}^{\downarrow}$ and $\mathrm{L}^{\uparrow}$ for each interval and then use hypothesis testing to choose the correct candidate distribution.

Also, note that since we have $O(k/\varepsilon)$ intervals, we need to run each instance of both the monotone learning algorithms and the hypothesis testing algorithm with confidence $1-O(\varepsilon/k)$, so that we can guarantee that the overall algorithm has confidence $9/10$. Note that Theorem 2.3 and Markov's inequality imply that if we draw $\Omega(\log n/\varepsilon^3)$ samples from a non-increasing distribution p, the hypothesis $\widetilde{p}$ output by $\mathrm{L}^{\downarrow}$ satisfies $d_{\mathrm{TV}}(\widetilde{p},p)\le\varepsilon$ with probability $9/10$. We can boost the confidence to $1-\delta$ with an overhead of

$$O(\log(1/\delta)\log\log(1/\delta))$$

in the sample complexity:

Fact 3.2. *Let p be a non-increasing distribution over $[n]$. There is an algorithm $\mathrm{L}^{\downarrow}{}_{\delta}$ with the following performance guarantee: Given*

$$(\log n/\varepsilon^3)\cdot\tilde{O}(\log(1/\delta)))$$

samples from p, $\mathrm{L}^{\downarrow}{}_{\delta}$ performs

$$\tilde{O}\left((\log^2 n/\varepsilon^3)\cdot\log^2(1/\delta)\right)$$

bit operations and outputs a (succinct description of a) hypothesis distribution $\widetilde{p}$ over $[n]$ that satisfies $d_{\mathrm{TV}}(\widetilde{p},p)\le\varepsilon$ with probability at least $1-\delta$.

Algorithm $\mathrm{L}^{\downarrow}{}_{\delta}$ runs $\mathrm{L}^{\downarrow}$ $O(\log(1/\delta))$ times and performs a tournament among the candidate hypotheses using `Choose-Hypothesis`. Let $\mathrm{L}^{\uparrow}{}_{\delta}$ denote the corresponding algorithm for learning non-decreasing distributions with confidence δ. We postpone further details on these algorithms to Appendix C.

Theorem 3.3. *Algorithm* `Learn-kmodal-simple` *(Algorithm 1) uses*

$$\frac{k\log n}{\varepsilon^4}\cdot\tilde{O}(\log(k/\varepsilon))$$

samples, performs $\operatorname{poly}(k,\log n,1/\varepsilon)$ *bit operations, and learns a k-modal distribution to accuracy* $O(\varepsilon)$ *with probability* $9/10$.

Proof. First, it is easy to see that the algorithm has the claimed sample complexity. Indeed, the algorithm draws a total of $r+m+m'$ samples in Steps 1, 4 and 5. The running time is also easy to analyze, as it is easy to see that every step can be performed in polynomial time (in fact, nearly linear time) in the sample size.

We need to show that with probability $9/10$ (over its random samples), algorithm `Learn-kmodal-simple` outputs a hypothesis h such that $d_{\mathrm{TV}}(h,p)\le O(\varepsilon)$.

Since $r=\Theta(d/\varepsilon^2)$ samples are drawn in Step 1, Fact 3.1 implies that with probability of failure at most $1/100$, for each family $\mathcal{J}$ of at most d disjoint intervals from $[n]$, we have

$$\sum_{J\in\mathcal{J}}|p(J)-\widehat{p}_m(J)|\le\varepsilon\,. \tag{3.2}$$

For the rest of the analysis of `Learn-kmodal-simple` we condition on this "good" event.

Since every atomic interval $I\in\mathcal{I}$ has $\widehat{p}(I)\ge\varepsilon/(10k)$ (except potentially the rightmost one), it follows that the number ℓ of atomic intervals constructed in Step 2 satisfies $\ell\le 10\cdot(k/\varepsilon)$. By the construction

Algorithm 1 `Learn-kmodal-simple`

Inputs: $\varepsilon > 0$; sample access to k-modal distribution p over $[n]$

1. Fix $d := \lceil 20k/\varepsilon \rceil$. Draw $r = \Theta(d/\varepsilon^2)$ samples from p and let $\widehat{p}$ denote the resulting empirical distribution.

2. Greedily partition the domain $[n]$ into ℓ *atomic intervals* $\mathcal{I} := \{I_i\}_{i=1}^{\ell}$ as follows:

   (a) $I_1 := [1, j_1]$, where $j_1 := \min\{j \in [n] \mid \widehat{p}([1, j]) \geq \varepsilon/(10k)\}$.

   (b) For $i \geq 1$, if $\bigcup_{j=1}^{i} I_j = [1, j_i]$, then $I_{i+1} := [j_i + 1, j_{i+1}]$, where j_{i+1} is defined as follows:
      - If $\widehat{p}([j_i + 1, n]) \geq \varepsilon/(10k)$, then $j_{i+1} := \min\{j \in [n] \mid \widehat{p}([j_i + 1, j]) \geq \varepsilon/(10k)\}$.
      - Otherwise, $j_{i+1} := n$.

3. Construct a set of ℓ *light intervals* $\mathcal{I}' := \{I'_i\}_{i=1}^{\ell}$ and a set $\{b_i\}_{i=1}^{t}$ of $t \leq \ell$ *heavy points* as follows:

   (a) For each interval $I_i = [a, b] \in \mathcal{I}$, if $\widehat{p}(I_i) \geq \varepsilon/(5k)$ define $I'_i := [a, b-1]$ and make b a heavy point. (Note that it is possible to have $I'_i = \emptyset$.)

   (b) Otherwise, define $I'_i := I_i$.

   Fix $\delta' := \varepsilon/(500k)$.

4. Draw $m = (k/\varepsilon^4) \cdot \log(n) \cdot \tilde{\Theta}(\log(1/\delta'))$ samples $\mathbf{s} = \{s_i\}_{i=1}^{m}$ from p. For each light interval I'_i, $i \in [\ell]$, run both $\mathrm{L}^{\downarrow}{}_{\delta'}$ and $\mathrm{L}^{\uparrow}{}_{\delta'}$ on the conditional distribution $p_{I'_i}$ using the samples in $\mathbf{s} \cap I'_i$. Let $\widetilde{p}^{\downarrow}_{I'_i}$, $\widetilde{p}^{\uparrow}_{I'_i}$ be the corresponding conditional hypothesis distributions.

5. Draw $m' = \Theta((k/\varepsilon^4) \cdot \log(1/\delta'))$ samples $\mathbf{s}' = \{s'_i\}_{i=1}^{m'}$ from p. For each light interval I'_i, $i \in [\ell]$, run $\texttt{Choose-Hypothesis}^p(\widetilde{p}^{\uparrow}_{I'_i}, \widetilde{p}^{\downarrow}_{I'_i}, \varepsilon, \delta')$ using the samples in $\mathbf{s}' \cap I'_i$. Denote by $\widetilde{p}_{I'_i}$ the returned conditional distribution on I'_i.

6. Output the hypothesis $h = \sum_{j=1}^{\ell} \widehat{p}(I'_j) \cdot \widetilde{p}_{I'_j} + \sum_{j=1}^{t} \widehat{p}(b_j) \cdot \mathbf{1}_{b_j}$.

in Steps 2 and 3, every light interval $I' \in \mathcal{I}'$ has $\widehat{p}(I') \leq \varepsilon/(5k)$. Note also that every heavy point b has $\widehat{p}(b) \geq \varepsilon/(10k)$ and the number of heavy points t is at most ℓ.

Since the light intervals and heavy points form a partition of $[n]$, we can write

$$p = \sum_{j=1}^{\ell} p(I'_j) \cdot p_{I'_j} + \sum_{j=1}^{t} p(b_j) \cdot \mathbf{1}_{b_j}.$$

Therefore, we can bound the variation distance as follows:

$$d_{\mathrm{TV}}(h,p) \leq \sum_{j=1}^{\ell} |\widehat{p}(I'_j) - p(I'_j)| + \sum_{j=1}^{t} |\widehat{p}(b_j) - p(b_j)| + \sum_{j=1}^{\ell} p(I'_j) \cdot d_{\mathrm{TV}}(\widetilde{p}_{I'_j}, p_{I'_j}). \tag{3.3}$$

Since $\ell + t \leq d$, by Fact 3.1 and our conditioning, the contribution of the first two terms to the sum is upper bounded by ε.

We proceed to bound the contribution of the third term. Since p is k-modal, at most k of the light intervals I'_j are not monotone for p. Call these intervals "bad" and denote by $\mathcal{B}$ as the set of bad intervals. Even though we have not identified the bad intervals, we know that all such intervals are light. Therefore, their total empirical probability mass (under $\widehat{p}_m$) is at most $k \cdot \varepsilon/(5k) = \varepsilon/5$, i. e., $\sum_{I \in \mathcal{B}} \widehat{p}(I) \leq \varepsilon/5$. By our conditioning (see equation (3.2)) and the triangle inequality it follows that

$$\left| \sum_{I \in \mathcal{B}} p(I) - \sum_{I \in \mathcal{B}} \widehat{p}(I) \right| \leq \sum_{I \in \mathcal{B}} |p(I) - \widehat{p}(I)| \leq \varepsilon$$

which implies that the true probability mass of the bad intervals is at most $\varepsilon/5 + \varepsilon = 6\varepsilon/5$. Hence, the contribution of bad intervals to the third term of the right hand side of (3.3) is at most $O(\varepsilon)$. (Note that this statement holds true independent of the samples $\mathbf{s}$ we draw in Step 4.)

It remains to bound the contribution of monotone intervals to the third term. Let $\ell' \leq \ell$ be the number of monotone light intervals and assume after renaming the indices that they are $\widetilde{\mathcal{I}} := \{I'_j\}_{j=1}^{\ell'}$. To bound from above the right hand side of (3.3), it suffices to show that with probability at least $19/20$ (over the samples drawn in Steps 4-5) it holds

$$\sum_{j=1}^{\ell'} p(I'_j) \cdot d_{\mathrm{TV}}(\widetilde{p}_{I'_j}, p_{I'_j}) = O(\varepsilon). \tag{3.4}$$

To prove (3.4) we partition the set $\widetilde{\mathcal{I}}$ into three subsets based on their probability mass under p. Note that we do not have a lower bound on the probability mass of intervals in $\widetilde{\mathcal{I}}$. Moreover, by our conditioning (see equation (3.2)) and the fact that each interval in $\widetilde{\mathcal{I}}$ is light, it follows that any $I \in \widetilde{\mathcal{I}}$ has $p(I) \leq \widehat{p}(I) + \varepsilon \leq 2\varepsilon$. We define the partition of $\widetilde{\mathcal{I}}$ into the following three sets:

$$\begin{aligned}
\widetilde{\mathcal{I}}_1 &= \{I \in \widetilde{\mathcal{I}} : p(I) \leq \varepsilon^2/(20k)\}, \\
\widetilde{\mathcal{I}}_2 &= \{I \in \widetilde{\mathcal{I}} : \varepsilon^2/(20k) < p(I) \leq \varepsilon/k\}, \quad \text{and} \\
\widetilde{\mathcal{I}}_3 &= \{I \in \widetilde{\mathcal{I}} : \varepsilon/k < p(I) \leq 2\varepsilon\}.
\end{aligned}$$

We bound the contribution of each subset in turn. It is clear that the contribution of $\widetilde{\mathcal{I}_1}$ to (3.4) is at most

$$\sum_{I\in\widetilde{\mathcal{I}_1}} p(I) \le |\widetilde{\mathcal{I}_1}|\cdot\varepsilon^2/(20k) \le \ell'\cdot\varepsilon^2/(20k) \le \ell\cdot\varepsilon^2/(20k) \le \varepsilon/2\,.$$

To bound from above the contribution of $\widetilde{\mathcal{I}_2}$ to (3.4), we partition $\widetilde{\mathcal{I}_2}$ into

$$g_2 = \lceil\log_2(20/\varepsilon)\rceil = \Theta(\log(1/\varepsilon))$$

groups. For $i\in[g_2]$, the set $(\widetilde{\mathcal{I}_2})^i$ consists of those intervals in $\widetilde{\mathcal{I}_2}$ that have mass under p in the range $\left(2^{-i}\cdot(\varepsilon/k), 2^{-i+1}\cdot(\varepsilon/k)\right]$. The following statement establishes the variation distance closeness between the conditional hypothesis for an interval in the i-th group $(\widetilde{\mathcal{I}_2})^i$ and the corresponding conditional distribution.

Claim 3.4. *With probability at least* $19/20$ *(over the sample* $\mathbf{s},\mathbf{s}'$*), for each* $i\in[g_2]$ *and each monotone light interval* $I'_j\in(\widetilde{\mathcal{I}_2})^i$ *we have* $d_{TV}(\widetilde{p_{I'_j}}, p_{I'_j}) = O(2^{i/3}\cdot\varepsilon)$.

Proof. Since in Step 4 we draw m samples, and each interval $I'_j\in(\widetilde{\mathcal{I}_2})^i$ has

$$p(I'_j)\in\left[2^{-i}\cdot(\varepsilon/k), 2^{-i+1}\cdot(\varepsilon/k)\right],$$

a standard coupon collector argument [22] tells us that with probability $99/100$, for *each* (i,j) pair, the interval I'_j will get at least $2^{-i}\cdot(\log(n)/\varepsilon^3)\cdot\tilde{\Omega}(\log(1/\delta'))$ many samples. Let's rewrite this as

$$\left(\log(n)/(2^{i/3}\cdot\varepsilon)^3\right)\cdot\tilde{\Omega}(\log(1/\delta'))$$

samples. We condition on this event.

Fix an interval $I'_j\in(\widetilde{\mathcal{I}_2})^i$. We first show that with failure probability at most $\varepsilon/(500k)$ after Step 4, either $\widetilde{p}^{\downarrow}_{I'_j}$ or $\widetilde{p}^{\uparrow}_{I'_j}$ will be $(2^{i/3}\cdot\varepsilon)$-accurate. Indeed, by Fact 3.2 and taking into account the number of samples that landed in I'_j, with probability $1-\varepsilon/(500k)$ over $\mathbf{s}$,

$$d_{\mathrm{TV}}(\widetilde{p}^{\alpha_i}_{I'_j}, p_{I'_j}) \le 2^{i/3}\varepsilon\,,$$

where $\alpha_i=\downarrow$ if $p_{I'_j}$ is non-increasing and $\alpha_i=\uparrow$ otherwise. By a union bound over all (at most ℓ many) (i,j) pairs, it follows that with probability at least $49/50$, for each interval $I'_j\in(\widetilde{\mathcal{I}_2})^i$ one of the two candidate hypothesis distributions is $(2^{i/3}\varepsilon)$-accurate. We condition on this event.

Now consider Step 5. Since this step draws m' samples, and each interval $I'_j\in(\widetilde{\mathcal{I}_2})^i$ has

$$p(I'_j)\in\left(2^{-i}\cdot(\varepsilon/k), 2^{-i+1}\cdot(\varepsilon/k)\right],$$

as before a standard coupon collector argument [22] tells us that with probability $99/100$, for *each* (i,j) pair, the interval I'_j will get at least

$$\left(1/(2^{i/3}\cdot\varepsilon)^3\right)\cdot\tilde{\Omega}(\log(1/\delta'))$$

many samples in this step; we henceforth assume that this is indeed the case for each I'_j. Thus, Theorem 2.4 applied to each fixed interval I'_j implies that the algorithm `Choose-Hypothesis` will output a hypothesis that is $6 \cdot (2^{i/3}\varepsilon)$-close to $p_{I'_j}$ with probability $1-\varepsilon/(500k)$. By a union bound, it follows that with probability at least $49/50$, the above condition holds for all monotone light intervals under consideration. Therefore, except with failure probability $19/20$, the statement of the claim holds. □

Given the claim, we exploit the fact that for intervals I'_j such that $p(I'_j)$ is small we can afford larger error on the total variation distance. More precisely, let $c_i = |(\widetilde{\mathcal{I}_2})^i|$, the number of intervals in $(\widetilde{\mathcal{I}_2})^i$, and note that $\sum_{i=1}^{g_2} c_i \leq \ell$. Hence, we can bound the contribution of $\widetilde{\mathcal{I}_2}$ to (3.4) by

$$\sum_{i=1}^{g_2} c_i \cdot (\varepsilon/k) \cdot 2^{-i+1} \cdot O(2^{i/3} \cdot \varepsilon) \leq O(1) \cdot (2\varepsilon^2/k) \cdot \sum_{i=1}^{g_2} c_i \cdot 2^{-2i/3}\,.$$

Since $\sum_{i=1}^{g_2} c_i = |\widetilde{\mathcal{I}_2}| \leq \ell$, the above expression is maximized for $c_1 = |\widetilde{\mathcal{I}_2}| \leq \ell$ and $c_i = 0$, $i > 1$, and the maximum value is at most

$$O(1) \cdot (\varepsilon^2/k) \cdot \ell = O(\varepsilon)\,.$$

Bounding the contribution of $\widetilde{\mathcal{I}_3}$ to (3.4) is very similar. We partition $\widetilde{\mathcal{I}_3}$ into

$$g_3 = \lceil \log_2 k \rceil + 1 = \Theta(\log(k))$$

groups. For $i \in [g_3]$, the set $(\widetilde{\mathcal{I}_3})^i$ consists of those intervals in $\widetilde{\mathcal{I}_3}$ that have mass under p in the range $\left(2^{-i+1} \cdot \varepsilon, 2^{-i+2} \cdot \varepsilon\right]$. The following statement is identical to Claim 3.4 albeit with different parameters:

Claim 3.5. *With probability at least $19/20$ (over the sample $\mathbf{s}, \mathbf{s}'$), for each $i \in [g_3]$ and each monotone light interval $I'_j \in (\widetilde{\mathcal{I}_3})^i$, we have $d_{TV}(\widetilde{p_{I'_j}}, p_{I'_j}) = O(2^{i/3} \cdot \varepsilon \cdot k^{-1/3})$.*

Let $f_i = |(\widetilde{\mathcal{I}_3})^i|$, the number of intervals in $(\widetilde{\mathcal{I}_3})^i$. Each interval $I \in (\widetilde{\mathcal{I}_3})^i$ has $p(I) \in (d_i, 2d_i]$, where $d_i := 2^{-i+1} \cdot \varepsilon$. We therefore have

$$\sum_{i=1}^{g_3} d_i f_i \leq p(\widetilde{\mathcal{I}_3}) \leq 1\,. \tag{3.5}$$

We can now bound from above the contribution of $\widetilde{\mathcal{I}_3}$ to (3.4) by

$$\sum_{i=1}^{g_3} 2 d_i f_i \cdot O\big(2^{i/3} \cdot \varepsilon \cdot k^{-1/3}\big) \leq O(1) \cdot \big(\varepsilon/k^{1/3}\big) \cdot \sum_{i=1}^{g_3} d_i f_i \cdot 2^{i/3}\,.$$

By (3.5) it follows that the above expression is maximized for $d_{g_3} f_{g_3} = 1$ and $d_i f_i = 0$, $i < g_3$. The maximum value is at most

$$O(1) \cdot (\varepsilon/k^{1/3}) \cdot 2^{g_3/3} = O(\varepsilon)$$

where the final equality uses the fact that $2^{g_3} \leq 4k$ as follows by our definition of g_3. This proves (3.4) and completes the proof of Theorem 3.3. □

In order to obtain an $O(\varepsilon)$-accurate hypothesis with probability $1-\delta$, we can simply run `Learn-kmodal-simple` $O(\log(1/\delta))$ times and then perform a tournament using Theorem 2.4. This increases the sample complexity by a $\tilde{O}(\log(1/\delta))$ factor. The running time increases by a factor of $O(\log^2(1/\delta))$. We postpone the details for Appendix C.

3.2 Main result: Learning k-modal distributions using testing

Here is some intuition to motivate our k-modal distribution learning algorithm and give a high-level idea of why the dominant term in its sample complexity is $O(k\log(n/k)/\varepsilon^3)$.

Let p denote the target k-modal distribution to be learned. As discussed above, optimal (in terms of time and sample complexity) algorithms are known for learning a monotone distribution over $[n]$, so if the locations of the k modes of p were known then it would be straightforward to learn p very efficiently by running the monotone distribution learner over $k+1$ separate intervals. But it is clear that in general we cannot hope to efficiently identify the modes of p exactly (for instance it could be the case that $p(a) = p(a+2) = 1/n$ while $p(a+1) = 1/n+1/2^n$). Still, it is natural to try to decompose the k-modal distribution into a collection of (nearly) monotone distributions and learn those. At a high level that is what our algorithm does, using a novel *property testing* algorithm.

More precisely, we give a distribution testing algorithm with the following performance guarantee: Let q be a k-modal distribution over $[n]$. Given an accuracy parameter τ, our tester takes $\mathrm{poly}(k/\tau)$ samples from q and outputs "yes" with high probability if q is monotone and "no" with high probability if q is τ-far from every monotone distribution. (We stress that the assumption that q is k-modal is essential here, since an easy argument given in [2] shows that $\Omega(n^{1/2})$ samples are required to test whether a general distribution over $[n]$ is monotone versus $\Theta(1)$-far from monotone.)

With some care, by running the above-described tester $O(k/\varepsilon)$ times with accuracy parameter τ, we can decompose the domain $[n]$ into

- at most $k+1$ "superintervals," which have the property that the conditional distribution of p over each superinterval is almost monotone (τ-close to monotone);
- at most $k+1$ "negligible intervals," which have the property that each one has probability mass at most $O(\varepsilon/k)$ under p (so ignoring all of them incurs at most $O(\varepsilon)$ total error); and
- at most $k+1$ "heavy" points, each of which has mass at least $\Omega(\varepsilon/k)$ under p.

We can ignore the negligible intervals, and the heavy points are easy to handle; however some care must be taken to learn the "almost monotone" restrictions of p over each superinterval. A naive approach, using a generic $\log(n)/\varepsilon^3$-sample monotone distribution learner that has no performance guarantees if the target distribution is not monotone, leads to an inefficient overall algorithm. Such an approach would require that τ (the closeness parameter used by the tester) be at most 1/(the sample complexity of the monotone distribution learner), i. e., $\tau < \varepsilon^3/\log(n)$. Since the sample complexity of the tester is $\mathrm{poly}(k/\tau)$ and the tester is run $\Omega(k/\varepsilon)$ times, this approach would lead to an overall sample complexity that is unacceptably high.

Fortunately, instead of using a generic monotone distribution learner, we can use the semi-agnostic monotone distribution learner of Birgé (Theorem 2.3) that can handle deviations from monotonicity far more efficiently than the above naive approach. Recall that given draws from a distribution q over $[n]$ that is τ-close to monotone, this algorithm uses $O(\log(n)/\varepsilon^3)$ samples and outputs a hypothesis distribution that is $(2\tau+\varepsilon)$-close to monotone. By using this algorithm we can take the accuracy parameter τ for our tester to be $\Theta(\varepsilon)$ and learn the conditional distribution of p over a given superinterval to accuracy

$O(\varepsilon)$ using $O(\log(n)/\varepsilon^3)$ samples from that superinterval. Since there are $k+1$ superintervals overall, a careful analysis shows that $O(k\log(n)/\varepsilon^3)$ samples suffice to handle all the superintervals.

We note that the algorithm also requires an additional additive $\mathrm{poly}(k/\varepsilon)$ samples (independent of n) besides this dominant term (for example, to run the tester and to estimate accurate weights with which to combine the various sub-hypotheses). The overall sample complexity we achieve is stated in Theorem 3.6 below.

Theorem 3.6 (Main)**.** *Algorithm* `Learn-kmodal` *(Algorithm 2) uses*

$$O\left(k\log(n/k)/\varepsilon^3 + (k^2/\varepsilon^3)\cdot\log(k/\varepsilon)\cdot\log\log(k/\varepsilon)\right)$$

samples, performs $\mathrm{poly}(k,\log n,1/\varepsilon)$ *bit operations, and learns any k-modal distribution to accuracy ε and confidence* $9/10$*.*

Theorem 1.1 follows from Theorem 3.6 by running `Learn-kmodal` $O(\log(1/\delta))$ times and using hypothesis testing to boost the confidence to $1-\delta$. We give details in Appendix C.

Algorithm `Learn-kmodal` makes essential use of an algorithm $\mathrm{T}^{\uparrow}$ for testing whether a k-modal distribution over $[n]$ is non-decreasing. Algorithm $\mathrm{T}^{\uparrow}(\varepsilon,\delta)$ uses $O(\log(1/\delta))\cdot(k/\varepsilon^2)$ samples from a k-modal distribution p over $[n]$, and behaves as follows:

- (Completeness) If p is non-decreasing, then $\mathrm{T}^{\uparrow}$ outputs "yes" with probability at least $1-\delta$;
- (Soundness) If p is ε-far from non-decreasing, then $\mathrm{T}^{\uparrow}$ outputs "yes" with probability at most δ.

Let $\mathrm{T}^{\downarrow}$ denote the analogous algorithm for testing whether a k-modal distribution over $[n]$ is non-increasing (we will need both algorithms). The description and proof of correctness for $\mathrm{T}^{\uparrow}$ is postponed to the following subsection (Section 3.4).

3.3 Algorithm `Learn-kmodal` and its analysis

Algorithm `Learn-kmodal` (Algorithm 2) is described in detail below. The analysis of the algorithm is the topic of Theorem 3.6, proved next.

Proof of Theorem 3.6. Before entering into the proof we record two observations; we state them explicitly here for the sake of the exposition.

Fact 3.7. *Let $R\subseteq[n]$. If p_R is neither non-increasing nor non-decreasing, then R contains at least one left extreme point.*

Fact 3.8. *Suppose that $R\subseteq[n]$ does not contain a left extreme point. For any ε,τ, if $\mathrm{T}^{\uparrow}(\varepsilon,\tau)$ and $\mathrm{T}^{\downarrow}(\varepsilon,\tau)$ are both run on p_R, then the probability that both calls return "no" is at most τ.*

Proof of Fact 3.8. By Fact 3.7 p_R is either non-decreasing or non-increasing. If p_R is non-decreasing then $\mathrm{T}^{\uparrow}$ will output "no" with probability at most τ, and similarly, if p_R is non-increasing then $\mathrm{T}^{\downarrow}$ will output "no" with probability at most τ. □

Algorithm 2 `Learn-kmodal`

Inputs: $\varepsilon > 0$; sample access to k-modal distribution p over $[n]$

1. Fix $\tau := \varepsilon/(100k)$. Draw $r = \Theta(1/\tau^2)$ samples from p and let $\widehat{p}$ denote the empirical distribution.
2. Greedily partition the domain $[n]$ into ℓ *atomic intervals* $\mathcal{I} := \{I_i\}_{i=1}^{\ell}$ as follows:
   (a) $I_1 := [1, j_1]$, where $j_1 := \min\{j \in [n] \mid \widehat{p}([1, j]) \geq \varepsilon/(10k)\}$.
   (b) For $i \geq 1$, if $\bigcup_{j=1}^{i} I_j = [1, j_i]$, then $I_{i+1} := [j_i + 1, j_{i+1}]$, where j_{i+1} is defined as follows:
      - If $\widehat{p}([j_i + 1, n]) \geq \varepsilon/(10k)$, then $j_{i+1} := \min\{j \in [n] \mid \widehat{p}([j_i + 1, j]) \geq \varepsilon/(10k)\}$.
      - Otherwise, $j_{i+1} := n$.
3. Set $\tau' := \varepsilon/(2000k)$. Draw $r' = \Theta((k^2/\varepsilon^3) \cdot \log(1/\tau') \log\log(1/\tau'))$ samples $\mathbf{s}$ from p to use in Steps 4-5.
4. Run both $\mathrm{T}^{\uparrow}(\varepsilon, \tau')$ and $\mathrm{T}^{\downarrow}(\varepsilon, \tau')$ over $p_{\bigcup_{i=1}^{j} I_i}$ for $j = 1, 2, \ldots$, to find the leftmost atomic interval I_{j_1} such that both $\mathrm{T}^{\uparrow}$ and $\mathrm{T}^{\downarrow}$ return "no" over $p_{\bigcup_{i=1}^{j_1} I_i}$.

   Let $I_{j_1} = [a_{j_1}, b_{j_1}]$. We consider two cases:

   Case 1: If $\widehat{p}[a_{j_1}, b_{j_1}] \geq 2\varepsilon/(10k)$, define $I'_{j_1} := [a_{j_1}, b_{j_1} - 1]$ and b_{j_1} is a *heavy* point.

   Case 2: If $\widehat{p}[a_{j_1}, b_{j_1}] < 2\varepsilon/(10k)$ then define $I'_{j_1} := I_{j_1}$.

   Call I'_{j_1} a *negligible* interval. If $j_1 > 1$ then define the first *superinterval* S_1 to be $\bigcup_{i=1}^{j_1-1} I_i$, and set $a_1 \in \{\uparrow, \downarrow\}$ to be $a_1 = \uparrow$ if $\mathrm{T}^{\uparrow}$ returned "yes" on $p_{\bigcup_{i=1}^{j_1-1} I_i}$ and to be $a_1 = \downarrow$ if $\mathrm{T}^{\downarrow}$ returned "yes" on $p_{\bigcup_{i=1}^{j_1-1} I_i}$.
5. Repeat Step 3 starting with the next interval I_{j_1+1}, i. e., find the leftmost atomic interval I_{j_2} such that both $\mathrm{T}^{\uparrow}$ and $\mathrm{T}^{\downarrow}$ return "no" over $p_{\bigcup_{i=j_1+1}^{j_2} I_i}$. Continue doing this until all intervals through I_ℓ have been used.

   Let $S_1, \ldots, S_t$ be the superintervals obtained through the above process and

   $$(a_1, \ldots, a_t) \in \{\uparrow, \downarrow\}^t$$

   be the corresponding string of bits.
6. Draw $m = \Theta(k \cdot \log(n/k)/\varepsilon^3)$ samples $\mathbf{s}'$ from p. For each superinterval S_i, $i \in [t]$, run A^{a_i} on the conditional distribution p_{S_i} of p using the samples in $\mathbf{s}' \cap S_i$. Let $\widetilde{p}_{S_i}$ be the hypothesis thus obtained.
7. Output the hypothesis $h = \sum_{i=1}^{t} \widehat{p}(S_i) \cdot \widetilde{p}_{S_i} + \sum_j \widehat{p}(\{b_j\}) \cdot \mathbf{1}_{b_j}$.

Since $r = \Theta(1/\tau^2)$ samples are drawn in the first step, Fact 3.1 (applied for $d = 1$) implies that with probability of failure at most $1/100$ each interval $I \subseteq [n]$ has $|\widehat{p}(I) - p(I)| \leq 2\tau$. For the rest of the proof we condition on this good event.

Since every atomic interval $I \in \mathcal{I}$ has $\widehat{p}(I) \geq \varepsilon/(10k)$ (except potentially the rightmost one), it follows that the number ℓ of atomic intervals constructed in Step 2 satisfies $\ell \leq 10 \cdot (k/\varepsilon)$. Moreover, by our conditioning, each atomic interval I_i has $p(I_i) \geq 8\varepsilon/(100k)$.

Note that in Case (1) of Step 4, if $\widehat{p}[a_{j_1}, b_{j_1}] \geq 2\varepsilon/(10k)$ then it must be the case that $\widehat{p}(b_{j_1}) \geq \varepsilon/(10k)$ (and thus $p(b_{j_1}) \geq 8\varepsilon/(100k)$). In this case, by definition of how the interval I_{j_1} was formed, we must have that $I'_{j_1} = [a_{j_1}, b_{j_1} - 1]$ satisfies $\widehat{p}(I'_{j_1}) < \varepsilon/(10k)$. So both in Case 1 and Case 2, we now have that $\widehat{p}(I'_{j_1}) \leq 2\varepsilon/(10k)$, and thus $p(I'_{j_1}) \leq 22\varepsilon/(100k)$. Entirely similar reasoning shows that every negligible interval constructed in Steps 4 and 5 has mass at most $22\varepsilon/(100k)$ under p.

In Steps 4–5 we invoke the testers $\mathrm{T}^{\downarrow}$ and $\mathrm{T}^{\uparrow}$ on the conditional distributions of (unions of contiguous) atomic intervals. Note that we need enough samples in every atomic interval, since otherwise the testers provide no guarantees. We claim that with probability at least $99/100$ over the sample $\mathbf{s}$ of Step 3, *each* atomic interval gets $b = \Omega\left((k/\varepsilon^2) \cdot \log(1/\tau')\right)$ samples. This follows by a standard coupon collector's argument, which we now provide. As argued above, each atomic interval has probability mass $\Omega(\varepsilon/k)$ under p. So, we have $\ell = O(k/\varepsilon)$ bins (atomic intervals), and we want each bin to contain b balls (samples). It is well-known [22] that after taking $\Theta(\ell \cdot \log \ell + \ell \cdot b \cdot \log\log \ell)$ samples from p, with probability $99/100$ each bin will contain the desired number of balls. The claim now follows by our choice of parameters. Conditioning on this event, any execution of the testers $\mathrm{T}^{\uparrow}(\varepsilon, \tau')$ and $\mathrm{T}^{\downarrow}(\varepsilon, \tau')$ in Steps 4 and 5 will have the guaranteed completeness and soundness properties.

In the execution of Steps 4 and 5, there are a total of at most ℓ occasions when $\mathrm{T}^{\uparrow}(\varepsilon, \tau')$ and $\mathrm{T}^{\downarrow}(\varepsilon, \tau')$ are both run over some union of contiguous atomic intervals. By Fact 3.8 and a union bound, the probability that (in any of these instances the interval does not contain a left extreme point and yet both calls return "no") is at most $(10k/\varepsilon)\tau' \leq 1/200$. So with failure probability at most $1/200$ for this step, each time Step 4 identifies a group of consecutive intervals $I_j, \ldots, I_{j+r}$ such that both $\mathrm{T}^{\uparrow}$ and $\mathrm{T}^{\downarrow}$ output "no," there is a left extreme point in $\bigcup_{i=j}^{j+r} I_i$. Since p is k-modal, it follows that with failure probability at most $1/200$ there are at most $k+1$ total repetitions of Step 4, and hence the number t of superintervals obtained is at most $k+1$.

We moreover claim that with very high probability each of the t superintervals S_i is very close to non-increasing or non-decreasing (with its correct orientation given by a_i):

Claim 3.9. *With failure probability at most* $1/100$*, each* $i \in [t]$ *satisfies the following: if* $a_i = \uparrow$ *then* p_{S_i} *is* ε*-close to a non-decreasing distribution and if* $a_i = \downarrow$ *then* p_{S_i} *is* ε*-close to a non-increasing distribution.*

Proof. There are at most $2\ell \leq 20k/\varepsilon$ instances when either $\mathrm{T}^{\downarrow}$ or $\mathrm{T}^{\uparrow}$ is run on a union of contiguous intervals. For any fixed execution of $\mathrm{T}^{\downarrow}$ over an interval I, the probability that $\mathrm{T}^{\downarrow}$ outputs "yes" while p_I is ε-far from every non-increasing distribution over I is at most τ', and similarly for $\mathrm{T}^{\uparrow}$. A union bound and the choice of τ' conclude the proof of the claim. □

Thus we have established that with overall failure probability at most $5/100$, after Step 5 the interval $[n]$ has been partitioned into:

1. A set $\{S_i\}_{i=1}^{t}$ of $t \le k+1$ superintervals, with $p(S_i) \ge 8\varepsilon/(100k)$ and p_{S_i} being ε-close to either non-increasing or non-decreasing according to the value of bit a_i.

2. A set $\{I'_i\}_{i=1}^{t'}$ of $t' \le k+1$ negligible intervals, such that $p(I'_i) \le 22\varepsilon/(100k)$.

3. A set $\{b_i\}_{i=1}^{t''}$ of $t'' \le k+1$ heavy points, each with $p(b_i) \ge 8\varepsilon/(100k)$.

We condition on the above good events, and bound from above the expected total variation distance (over the sample $\mathbf{s}'$). In particular, we have the following lemma:

Lemma 3.10. *Conditioned on the above good events 1–3, we have that* $\mathbf{E}_{\mathbf{s}'}\left[d_{\mathrm{TV}}(h,p)\right] = O(\varepsilon)$.

Proof of Lemma 3.10. By the discussion preceding the lemma statement, the domain $[n]$ has been partitioned into a set of superintervals, a set of negligible intervals and a set of heavy points. As a consequence, we can write

$$p = \sum_{j=1}^{t} p(S_j) \cdot p_{S_j} + \sum_{j=1}^{t''} p(\{b_j\}) \cdot \mathbf{1}_{b_j} + \sum_{j=1}^{t'} p(I'_j) \cdot p_{I'_j}\,.$$

Therefore, we can bound the total variation distance as follows:

$$d_{\mathrm{TV}}(h,p) \le \sum_{j=1}^{t} |\widehat{p}(S_j) - p(S_j)| + \sum_{j=1}^{t''} |\widehat{p}(b_j) - p(b_j)| + \sum_{j=1}^{t'} p(I'_j) + \sum_{j=1}^{t} p(S_j) \cdot d_{\mathrm{TV}}(\widetilde{p}_{S_j}, p_{S_j})\,.$$

Recall that each term in the first two sums is bounded from above by 2τ. Hence, the contribution of these terms to the RHS is at most $2\tau \cdot (2k+2) \le \varepsilon/10$. Since each negligible interval I'_j has $p(I'_j) \le 22\varepsilon/(100k)$, the contribution of the third sum is at most $t' \cdot 22\varepsilon/(100k) \le \varepsilon/4$. It thus remains to bound the contribution of the last sum.

We will show that

$$\mathop{\mathbf{E}}_{\mathbf{s}'}\left[\sum_{j=1}^{t} p(S_j) \cdot d_{\mathrm{TV}}(\widetilde{p}_{S_j}, p_{S_j})\right] = O(\varepsilon)\,.$$

Denote $n_i = |S_i|$. Clearly, $\sum_{i=1}^{t} n_i \le n$. Since we are conditioning on the good events (1)-(3), each superinterval is ε-close to monotone with a known orientation (non-increasing or non-decreasing) given by a_i. Hence we may apply Theorem 2.3 for each superinterval.

Recall that in Step 5 we draw a total of m samples. Let m_i, $i \in [t]$ be the number of samples that land in S_i; observe that m_i is a binomially distributed random variable with $m_i \sim \mathrm{Bin}(m, p(S_i))$. We apply Theorem 2.3 for each ε-monotone interval, conditioning on the value of m_i, and get

$$d_{\mathrm{TV}}(\widetilde{p}_{S_i}, p_{S_i}) \le 2\varepsilon + O\left((\log n_i/(m_i+1))^{1/3}\right)\,.$$

Hence, we can bound from above the desired expectation as follows

$$\sum_{j=1}^{t} p(S_j)\cdot \mathop{\mathbf{E}}_{\mathbf{s}'}\left[d_{\mathrm{TV}}(\widetilde{p}_{S_j}, p_{S_j})\right] \leq \left(\sum_{j=1}^{t} 2\varepsilon\cdot p(S_j)\right) + O\left(\sum_{j=1}^{t} p(S_j)\cdot(\log n_j)^{1/3}\cdot \mathop{\mathbf{E}}_{\mathbf{s}'}[(m_j+1)^{-1/3}]\right).$$

Since $\sum_j p(S_j) \leq 1$, to prove the lemma, it suffices to show that the second term is bounded, i. e., that

$$\sum_{j=1}^{t} p(S_j)\cdot(\log n_j)^{1/3}\cdot \mathop{\mathbf{E}}_{\mathbf{s}'}[(m_j+1)^{-1/3}] = O(\varepsilon).$$

To do this, we will first need the following claim:

Claim 3.11. *For a binomial random variable $X \sim \mathrm{Bin}(m,q)$ it holds $\mathbf{E}[(X+1)^{-1/3}] < (mq)^{-1/3}$.*

Proof. Jensen's inequality implies that

$$\mathbf{E}[(X+1)^{-1/3}] \leq (\mathbf{E}[1/(X+1)])^{1/3}.$$

We claim that $\mathbf{E}[1/(X+1)] < 1/\mathbf{E}[X]$. This can be shown as follows: We first recall that $\mathbf{E}[X] = m\cdot q$. For the expectation of the inverse, we can write:

$$\begin{aligned}\mathbf{E}\left[1/(X+1)\right] &= \sum_{j=0}^{m}\frac{1}{j+1}\binom{m}{j}q^j(1-q)^{m-j}\\ &= \frac{1}{m+1}\cdot\sum_{j=0}^{m}\binom{m+1}{j+1}q^j(1-q)^{m-j}\\ &= \frac{1}{q\cdot(m+1)}\cdot\sum_{i=1}^{m+1}\binom{m+1}{i}q^i(1-q)^{m+1-i}\\ &= \frac{1-(1-q)^{m+1}}{q\cdot(m+1)} < \frac{1}{m\cdot q}.\end{aligned}$$

The claim now follows by the monotonicity of the mapping $x \mapsto x^{1/3}$. □

By Claim 3.11, applied to $m_i \sim \mathrm{Bin}(m, p(S_i))$, we have that

$$\mathop{\mathbf{E}}_{\mathbf{s}'}[(m_i+1)^{-1/3}] < m^{-1/3}\cdot(p(S_i))^{-1/3}.$$

Therefore, our desired quantity can be bounded from above by

$$\sum_{j=1}^{t}\frac{p(S_j)\cdot(\log n_j)^{1/3}}{m^{1/3}\cdot(p(S_j))^{1/3}} = O(\varepsilon)\cdot\sum_{j=1}^{t}(p(S_j))^{2/3}\cdot\left(\frac{\log n_j}{k\cdot\log(n/k)}\right)^{1/3}.$$

We now claim that the second term in the RHS above is upper bounded by 2. Indeed, this follows by an application of Hölder's inequality for the vectors

$$\left(p(S_j)^{2/3}\right)_{j=1}^{t} \qquad \text{and} \qquad \left(\left(\frac{\log n_j}{k\cdot\log(n/k)}\right)^{1/3}\right)_{j=1}^{t},$$

with Hölder conjugates $3/2$ and 3. That is,

$$\sum_{j=1}^{t}(p(S_j))^{2/3}\cdot\left(\frac{\log n_j}{k\cdot\log(n/k)}\right)^{1/3} \le \left(\sum_{j=1}^{t}p(S_j)\right)^{2/3}\cdot\left(\sum_{j=1}^{t}\frac{\log n_j}{k\cdot\log(n/k)}\right)^{1/3} \le 2\,.$$

The first inequality is Hölder and the second uses the fact that

$$\sum_{j=1}^{t}p(S_j)\le 1 \qquad \text{and} \qquad \sum_{j=1}^{t}\log(n_j)\le t\cdot\log(n/t)\le(k+1)\cdot\log(n/k)\,.$$

This last inequality is a consequence of the concavity of the logarithm and the fact that $\sum_j n_j \le n$. This completes the proof of the lemma. □

By applying Markov's inequality and a union bound, we get that with probability $9/10$ the algorithm `Learn-kmodal` outputs a hypothesis h that has $d_{\mathrm{TV}}(h,p){=}O(\varepsilon)$ as required.

It is clear that the algorithm has the claimed sample complexity. The running time is also easy to analyze, as it is easy to see that every step can be performed in polynomial time in the sample size. This completes the proof of Theorem 3.6. □

3.4 Testing whether a k-modal distribution is monotone

In this section we describe and analyze the testing algorithm $\mathrm{T}^{\uparrow}$ (Algorithm 3). Given sample access to a k-modal distribution q over $[n]$ and $\tau>0$, our tester $\mathrm{T}^{\uparrow}$ uses $O(k/\tau^2)$ many samples from q and has the following properties:

- If q is non-decreasing, $\mathrm{T}^{\uparrow}$ outputs "yes" with probability at least $2/3$.
- If q is τ-far from non-decreasing, $\mathrm{T}^{\uparrow}$ outputs "no" with probability at least $2/3$.

(Algorithm $\mathrm{T}^{\uparrow}(\tau,\delta)$ is obtained by repeating $\mathrm{T}^{\uparrow}$ $O(\log(1/\delta))$ times and taking the majority vote.)

Before we describe the algorithm we need some notation. Let q be a distribution over $[n]$. For $a\le b<c\in[n]$ define

$$E(q,a,b,c) := \frac{q([a,b])}{(b-a+1)} - \frac{q([b+1,c])}{(c-b)}\,.$$

We also denote

$$T(q,a,b,c) := \frac{E(q,a,b,c)}{\frac{1}{(b-a+1)}+\frac{1}{(c-b)}}\,.$$

Intuitively, the quantity $E(q,a,b,c)$ captures the difference between the average value of q over $[a,b]$ versus over $[b+1,c]$; it is negative iff the average value of q is higher over $[b+1,c]$ than it is over $[a,b]$. The quantity $T(q,a,b,c)$ is a scaled version of $E(q,a,b,c)$.

The idea behind tester $\mathrm{T}^{\uparrow}$ is simple. It is based on the observation that if q is a non-decreasing distribution, then for any two consecutive intervals $[a,b]$ and $[b+1,c]$ the average of q over $[b+1,c]$ must be at least as large as the average of q over $[a,b]$. Thus any non-decreasing distribution will pass a test that checks "all" pairs of consecutive intervals looking for a violation. Our tester $\mathrm{T}^{\uparrow}$ checks "all" sums of (at most) k consecutive intervals looking for a violation. Our analysis shows that in fact such a test is complete as well as sound if the distribution q is guaranteed to be k-modal. The key ingredient is a structural result (Lemma 3.13 below), which is proved using a procedure reminiscent of "Myerson ironing" [21] to convert a k-modal distribution to a non-decreasing distribution.

Algorithm 3 `Tester` $\mathrm{T}^{\uparrow}(\tau)$

Inputs: $\tau > 0$; sample access to k-modal distribution q over $[n]$

1. Draw $r = \Theta(k/\tau^2)$ samples $\mathbf{s}$ from q and let $\widehat{q}$ be the resulting empirical distribution.
2. If there exists $\ell \in [k]$ and $\{a_i, b_i, c_i\}_{i=1}^{\ell} \in \mathbf{s} \cup \{n\}$ with $a_i \le b_i < c_i < a_{i+1}, \quad i \in [\ell-1]$, such that

$$\sum_{i=1}^{\ell} T(\widehat{q}, a_i, b_i, c_i - 1) \ge \tau/4 \tag{3.6}$$

then output "no," otherwise output "yes."

The following theorem establishes correctness of the tester.

Theorem 3.12. *Algorithm* $\mathrm{T}^{\uparrow}$ *(Algorithm 3) uses* $O(k/\tau^2)$ *samples from q, performs* $\mathrm{poly}(k/\tau)\cdot\log n$ *bit operations and satisfies the desired completeness and soundness properties.*

Proof. We start by showing that the algorithm has the claimed completeness and soundness properties. Let us say that the sample $\mathbf{s}$ is *good* if for every collection $\mathcal{I}$ of (at most) $3k$ intervals in $[n]$ it holds

$$\sum_{I \in \mathcal{I}} |q(I) - \widehat{q}(I)| \le \tau/20\,.$$

By Fact 3.1 with probability at least $2/3$ the sample $\mathbf{s}$ is good. We henceforth condition on this event.

For $a \le b < c \in [n]$ let us denote $\gamma = |q([a,b]) - \widehat{q}([a,b])|$ and $\gamma' = |q([b+1,c]) - \widehat{q}([b+1,c])|$. Then we can write

$$|E(q,a,b,c) - E(\widehat{q},a,b,c)| \le \frac{\gamma}{b-a+1} + \frac{\gamma'}{c-b} \le (\gamma+\gamma')\cdot\left(\frac{1}{b-a+1} + \frac{1}{c-b}\right)$$

which implies that

$$|T(q,a,b,c) - T(\widehat{q},a,b,c)| \le \gamma + \gamma'\,. \tag{3.7}$$

Now consider any $\{a_i,b_i,c_i\}_{i=1}^{\ell} \in [n]$, for some $\ell \le k$, with $a_i \le b_i < c_i < a_{i+1}$, $i \in [\ell-1]$. Similarly denote $\gamma_i = |q([a_i,b_i]) - \widehat{q}([a_i,b_i])|$ and $\gamma_i' = |q([b_i+1,c_i]) - \widehat{q}([b_i+1,c_i])|$. With this notation we have

$$\left|\sum_{i=1}^{\ell} T(q,a_i,b_i,c_i) - \sum_{i=1}^{\ell} T(\widehat{q},a_i,b_i,c_i)\right| \le \sum_{i=1}^{\ell} |T(q,a_i,b_i,c_i) - T(\widehat{q},a_i,b_i,c_i)| \le \sum_{i=1}^{\ell}(\gamma_i+\gamma_i')$$

where we used the triangle inequality and (3.7). Note that the rightmost term is the sum of the "additive errors" for the collection $\{[a_i,b_i],[b_i+1,c_i]\}_{i=1}^{\ell}$ of 2ℓ intervals. Hence, it follows from our conditioning that the last term is bounded from above by $\tau/20$, i. e.,

$$\left|\sum_{i=1}^{\ell} T(q,a_i,b_i,c_i) - \sum_{i=1}^{\ell} T(\widehat{q},a_i,b_i,c_i)\right| \le \tau/20\,. \tag{3.8}$$

We first establish completeness. Suppose that q is non-decreasing. Then the average probability value in any interval $[a,b]$ is a non-decreasing function of a. That is, *for all $a \le b < c \in [n]$ it holds* $E(q,a,b,c) \le 0$, hence $T(q,a,b,c) \le 0$. This implies that for any choice of $\{a_i,b_i,c_i\}_{i=1}^{\ell} \in [n]$ with $a_i \le b_i < c_i < a_{i+1}$, we will have $\sum_{i=1}^{\ell} T(q,a_i,b_i,c_i) \le 0$. By (3.8) we now get that

$$\sum_{i=1}^{\ell} T(\widehat{q},a_i,b_i,c_i) \le \tau/20\,,$$

i. e., the tester says "yes" with probability at least $2/3$.

To prove soundness, we will crucially need the following structural lemma:

Lemma 3.13. *Let q be a k-modal distribution over $[n]$ that is τ-far from being non-decreasing. Then* there exists *$\ell \in [k]$ and $\{a_i,b_i,c_i\}_{i=1}^{\ell} \subseteq [n]^{3\ell}$ with $a_i \le b_i < c_i < a_{i+1}$, $i \in [\ell-1]$, such that*

$$\sum_{i=1}^{\ell} T(q,a_i,b_i,c_i) \ge \tau/2\,. \tag{3.9}$$

We first show how the soundness follows from the lemma. Let q be a k-modal distribution over $[n]$ that is τ-far from non-decreasing. Denote $\mathbf{s}' := \mathbf{s} \cup \{n\} = \{s_1,s_2,\ldots,s_{r'}\}$ with $r' \le r+1$ and $s_j < s_{j+1}$. We want to show that there exist points in $\mathbf{s}'$ that satisfy (3.6). Namely, that there exists $\ell \in [k]$ and $\{s_{a_i},s_{b_i},s_{c_i}\}_{i=1}^{\ell} \in \mathbf{s}'$ with $s_{a_i} \le s_{b_i} < s_{c_i} < s_{a_{i+1}}$, $i \in [\ell-1]$, such that

$$\sum_{i=1}^{\ell} T(\widehat{q},s_{a_i},s_{b_i},s_{c_i}-1) \ge \tau/4\,. \tag{3.10}$$

By Lemma 3.13, there exists $\ell \in [k]$ and $\{a_i,b_i,c_i\}_{i=1}^{\ell} \in [n]$ with $a_i \le b_i < c_i < a_{i+1}$, $i \in [\ell-1]$, such that $\sum_{i=1}^{\ell} T(q,a_i,b_i,c_i) \ge \tau/2$. Combined with (3.8) the latter inequality implies that

$$\sum_{i=1}^{\ell} T(\widehat{q},a_i,b_i,c_i) \ge \tau/2 - \tau/20 > \tau/4\,. \tag{3.11}$$

First note that it is no loss of generality to assume that $\widehat{q}([a_i,b_i]) > 0$ for all $i \in [\ell]$. (If there is some $j \in [\ell]$ with $\widehat{q}([a_j,b_j]) = 0$, then by definition we have $T(\widehat{q},a_j,b_j,c_j) \le 0$; hence, we can remove this term from the above sum and the RHS does not decrease.)

Given the domain points $\{a_i,b_i,c_i\}_{i=1}^{\ell}$ we define the sample points s_{a_i},s_{b_i},s_{c_i} such that:

(i) $[s_{a_i}, s_{b_i}] \subseteq [a_i, b_i]$,

(ii) $[s_{b_i}+1, s_{c_i}-1] \supseteq [b_i+1, c_i]$,

(iii) $\widehat{q}([s_{a_i}, s_{b_i}]) = \widehat{q}([a_i, b_i])$ and

(iv) $\widehat{q}([s_{b_i}+1, s_{c_i}-1]) = \widehat{q}([b_i+1, c_i])$.

To achieve these properties we select:

- s_{a_i} to be the leftmost point of the sample in $[a_i, b_i]$; s_{b_i} to be the rightmost point of the sample in $[a_i, b_i]$ (note that by our assumption that $\widehat{q}([a_i, b_i]) > 0$ at least one sample falls in $[a_i, b_i]$);
- s_{c_i} to be the leftmost point of the sample in $[c_i+1, n]$; or the point n if $[c_i+1, n]$ has no samples or is empty.

We can rewrite (3.11) as follows:

$$\sum_{i=1}^{\ell} \frac{\widehat{q}([a_i, b_i])}{1+\frac{b_i-a_i+1}{c_i-b_i}} \geq \tau/4 + \sum_{i=1}^{\ell} \frac{\widehat{q}([b_i+1, c_i])}{1+\frac{c_i-b_i}{b_i-a_i+1}}\,. \tag{3.12}$$

Now note that by properties (i) and (ii) above it follows that

$$b_i - a_i + 1 \geq s_{b_i} - s_{a_i} + 1 \qquad \text{and} \qquad c_i - b_i \leq s_{c_i} - s_{b_i} - 1\,.$$

Combining with properties (iii) and (iv) we get

$$\frac{\widehat{q}([a_i, b_i])}{1+\frac{b_i-a_i+1}{c_i-b_i}} = \frac{\widehat{q}([s_{a_i}, s_{b_i}])}{1+\frac{b_i-a_i+1}{c_i-b_i}} \leq \frac{\widehat{q}([s_{a_i}, s_{b_i}])}{1+\frac{s_{b_i}-s_{a_i}+1}{s_{c_i}-s_{b_i}-1}} \tag{3.13}$$

and similarly

$$\frac{\widehat{q}([b_i+1, c_i])}{1+\frac{c_i-b_i}{b_i-a_i+1}} = \frac{\widehat{q}([s_{b_i}+1, s_{c_i}-1])}{1+\frac{c_i-b_i}{b_i-a_i+1}} \geq \frac{\widehat{q}([s_{b_i}+1, s_{c_i}-1])}{1+\frac{s_{c_i}-s_{b_i}-1}{s_{b_i}-s_{a_i}+1}}\,. \tag{3.14}$$

A combination of (3.12), (3.13), (3.14) yields the desired result (3.10).

It thus remains to prove Lemma 3.13.

Proof of Lemma 3.13. We will prove the contrapositive. Let q be a k-modal distribution over $[n]$ such that for any $\ell \leq k$ and $\{a_i, b_i, c_i\}_{i=1}^{\ell} \subseteq [n]^{3\ell}$ such that $a_i \leq b_i < c_i < a_{i+1}$, $i \in [\ell-1]$, we have

$$\sum_{i=1}^{\ell} T(q, a_i, b_i, c_i) \leq \tau/2\,. \tag{3.15}$$

We will construct a non-decreasing distribution $\widetilde{q}$ that is τ-close to q.

The high level idea of the argument is as follows: the construction of $\widetilde{q}$ proceeds in (at most) k stages where in each stage, we reduce the number of modes by at least one and incur small error in the total variation distance. In particular, we iteratively construct a sequence of distributions $\{q^{(i)}\}_{i=0}^{\ell}$, $q^{(0)} = q$ and $q^{(\ell)} = \widetilde{q}$, for some $\ell \le k$, such that for all $i \in [\ell]$ we have that $q^{(i)}$ is $(k-i)$-modal and $d_{\mathrm{TV}}(q^{(i-1)}, q^{(i)}) \le 2\tau_i$, where the quantities τ_i will be defined in the course of the analysis below. By appropriately using (3.15), we will show that

$$\sum_{i=1}^{\ell} \tau_i \le \tau/2. \tag{3.16}$$

Assuming this, it follows from the triangle inequality that

$$d_{\mathrm{TV}}(\widetilde{q}, q) \le \sum_{i=1}^{\ell} d_{\mathrm{TV}}(q^{(i)}, q^{(i-1)}) \le 2 \cdot \sum_{i=1}^{\ell} \tau_i \le \tau$$

as desired, where the last inequality uses (3.16).

Consider the graph (histogram) of the discrete density q. The x-axis represents the n points of the domain and the y-axis the corresponding probabilities. We first informally describe how to obtain $q^{(1)}$ from q. The construction of $q^{(i)}$ from $q^{(i-1)}$, $i \in [\ell]$, is essentially identical. Let j_1 be the *leftmost* (i. e., having minimum x-coordinate) left-extreme point (mode) of q, and assume that it is a local maximum with height (probability mass) $q(j_1)$. (A symmetric argument works for the case that it is a local minimum.) The idea of the proof is based on the following simple process (reminiscent of Myerson's ironing process [21]): We start with the horizontal line $y = q(j_1)$ and move it downwards until we reach a height $h_1 < q(j_1)$ so that the total mass "cut-off" equals the mass "missing" to the right; then we make the distribution "flat" in the corresponding interval (hence, reducing the number of modes by at least one).

We now proceed with the formal argument, assuming as above that the leftmost left-extreme point j_1 of q is a local maximum. We say that the line $y = h$ *intersects* a point $i \in [n]$ in the domain of q if $q(i) \ge h$. The line $y = h$, $h \in [0, q(j_1)]$, intersects the graph of q at a unique interval $I(h) \subseteq [n]$ that contains j_1. Suppose $I(h) = [a(h), b(h)]$, where $a(h), b(h) \in [n]$ depend on h. By definition this means that $q(a(h)) \ge h$ and $q(a(h)-1) < h$ (since q is supported on $[n]$, we adopt the convention that $q(0) = 0$). Recall that the distribution q is non-decreasing in the interval $[1, j_1]$ and that $j_1 \ge a(h)$. The term "the mass cut-off by the line $y = h$" means the quantity

$$A(h) = q\left(I(h)\right) - h \cdot \left(b(h) - a(h) + 1\right),$$

i. e., the "mass of the interval $I(h)$ above the line."

The height h of the line $y = h$ defines the points $a(h), b(h) \in [n]$ as described above. We consider values of h such that q is unimodal (increasing then decreasing) over $I(h)$. In particular, let j_1' be the leftmost mode of q to the right of j_1, i. e., $j_1' > j_1$ and j_1' is a local minimum. We consider values of $h \in (q(j_1'), q(j_1))$. For such values, the interval $I(h)$ is indeed unimodal (as $b(h) < j_1'$). For $h \in (q(j_1'), q(j_1))$ we define the point $c(h) \ge j_1'$ as follows: It is the rightmost point of the largest interval containing j_1' whose probability mass does not exceed h. That is, all points in $[j_1', c(h)]$ have probability mass at most h and $q(c(h)+1) > h$ (or $c(h) = n$).

Consider the interval $J(h) = [b(h)+1, c(h)]$. This interval is non-empty, since $b(h) < j_1' \le c(h)$. (Note that $J(h)$ is not necessarily a unimodal interval; it contains at least one mode j_1' of q, but it may also contain more modes.) The term "the mass missing to the right of the line $y = h$" means the quantity

$$B(h) = h \cdot (c(h) - b(h)) - q(J(h)) .$$

Consider the function $C(h) = A(h) - B(h)$ over $[q(j_1'), q(j_1)]$. This function is continuous in its domain; moreover, we have that

$$C(q(j_1)) = A(q(j_1)) - B(q(j_1)) < 0 ,$$

as $A(q(j_1)) = 0$, and

$$C\left(q(j_1')\right) = A\left(q(j_1')\right) - B\left(q(j_1')\right) > 0 ,$$

as $B(q(j_1')) = 0$. Therefore, by the intermediate value theorem, there exists a value $h_1 \in (q(j_1'), q(j_1))$ such that

$$A(h_1) = B(h_1) .$$

The distribution $q^{(1)}$ is constructed as follows: We move the mass $\tau_1 = A(h_1)$ from $I(h_1)$ to $J(h_1)$. Note that the distribution $q^{(1)}$ is identical to q outside the interval $[a(h_1), c(h_1)]$, hence the leftmost mode of $q^{(1)}$ is in $(c(h_1), n]$. It is also clear that

$$d_{\mathrm{TV}}(q^{(1)}, q) \leq 2\tau_1 .$$

Let us denote $a_1 = a(h_1)$, $b_1 = b(h_1)$ and $c_1 = c(h_1)$. We claim that $q^{(1)}$ has at least one mode less than q. Indeed, $q^{(1)}$ is non-decreasing in $[1, a_1 - 1]$ and constant in $[a_1, c_1]$. (By our "flattening" process, all the points in the latter interval have probability mass exactly h_1.) Recalling that

$$q^{(1)}(a_1) = h_1 \geq q^{(1)}(a_1 - 1) = q(a_1 - 1) ,$$

we deduce that $q^{(1)}$ is non-decreasing in $[1, c_1]$.

We will now argue that

$$\tau_1 = T(q, a_1, b_1, c_1) . \tag{3.17}$$

Recall that we have $A(h_1) = B(h_1) = \tau_1$, which can be written as

$$q([a_1, b_1]) - h_1 \cdot (b_1 - a_1 + 1) = h_1 \cdot (c_1 - b_1) - q([b_1 + 1, c_1]) = \tau_1 .$$

From this, we get

$$\frac{q([a_1, b_1])}{(b_1 - a_1 + 1)} - \frac{q([b_1 + 1, c_1])}{(c_1 - b_1)} = \frac{\tau_1}{(b_1 - a_1 + 1)} + \frac{\tau_1}{(c_1 - b_1)}$$

or equivalently

$$E(q, a_1, b_1, c_1) = \frac{\tau_1}{(b_1 - a_1 + 1)} + \frac{\tau_1}{(c_1 - b_1)}$$

which gives (3.17).

We construct $q^{(2)}$ from $q^{(1)}$ using the same procedure. Recalling that the leftmost mode of $q^{(1)}$ lies in the interval $(c_1, n]$ an identical argument as above implies that

$$d_{\mathrm{TV}}(q^{(2)}, q^{(1)}) \le 2\tau_2$$

where

$$\tau_2 = T(q^{(1)}, a_2, b_2, c_2)$$

for some $a_2, b_2, c_2 \in [n]$ satisfying $c_1 < a_2 \le b_2 < c_2$. Since $q^{(1)}$ is identical to q in $(c_1, n]$, it follows that

$$\tau_2 = T(q, a_2, b_2, c_2)\,.$$

We continue this process iteratively for $\ell \le k$ stages until we obtain a non-decreasing distribution $q^{(\ell)}$. (Note that we remove at least one mode in each iteration, hence it may be the case that $\ell < k$.) It follows inductively that for all $i \in [\ell]$, we have that $d_{\mathrm{TV}}(q^{(i)}, q^{(i-1)}) \le 2\tau_i$ where $\tau_i = T(q, a_i, b_i, c_i)$, for $c_{i-1} < a_i \le b_i < c_i$.

We therefore conclude that

$$\sum_{i=1}^{\ell} \tau_i = \sum_{i=1}^{\ell} T(q, a_i, b_i, c_i)$$

which is bounded from above by $\tau/2$ by (3.15). This establishes (3.16) completing the proof of Lemma 3.13. □

The upper bound on the sample complexity of the algorithm is straightforward, since only Step 1 uses samples.

It remains to analyze the running time. The only non-trivial computation is in Step 2 where we need to decide whether there exist $\ell \le k$ "ordered triples" $\{a_i, b_i, c_i\}_{i=1}^{\ell} \in \mathbf{s}'$ with $a_i \le b_i < c_i < a_{i+1}$, $i \in [\ell-1]$, such that $\sum_{i=1}^{\ell} T(\widehat{q}, a_i, b_i, c_i - 1) \ge \tau/4$. Even though a naive brute-force implementation would need time $\Omega(r^k) \cdot \log n$, there is a simple dynamic programming algorithm that runs in $\mathrm{poly}(r,k) \cdot \log n$ time.

We now provide the details. Consider the objective function

$$\mathcal{T}(\ell) = \max\left\{ \sum_{i=1}^{\ell} T(\widehat{q}, a_i, b_i, c_i - 1) \;\middle|\; \{a_i, b_i, c_i\}_{i=1}^{\ell} \in \mathbf{s}' \text{ with } a_i \le b_i < c_i < a_{i+1}, i \in [\ell-1] \right\},$$

for $\ell \in [k]$. We want to decide whether $\max_{\ell \le k} \mathcal{T}(\ell) \ge \tau/4$. For $\ell \in [k]$ and $j \in [r']$, we use dynamic programming to compute the quantities

$$\mathcal{T}(\ell, j) = \max\left\{ \sum_{i=1}^{\ell} T(\widehat{q}, a_i, b_i, c_i - 1) \;\middle|\; \begin{array}{l} \{a_i, b_i, c_i\}_{i=1}^{\ell} \in \mathbf{s}' \text{ with } a_i \le b_i < c_i < a_{i+1}, \\ i \in [\ell-1] \text{ and } c_\ell = s_j \end{array} \right\}.$$

(This clearly suffices as $\mathcal{T}(\ell) = \max_{j \in [r']} \mathcal{T}(\ell, j)$.) The dynamic program is based on the recursive identity

$$\mathcal{T}(\ell+1, j) = \max_{j' \in [r'], j' < j} \mathcal{T}(\ell, j') + \mathcal{T}'(j'+1, j)\,,$$

where we define

$$\mathcal{T}'(\alpha, \beta) = \max\{T(\widehat{q}, a, b, \beta) \mid a, b \in \mathbf{s}', \alpha \le a \le b < \beta\}\,.$$

Note that all the values $\mathcal{T}'(j'+1, j)$ (where $j', j \in [r']$ and $j' < j$) can be computed in $O(r^3)$ time. Fix $\ell \in [k]$. Suppose we have computed all the values $\mathcal{T}(\ell, j')$, $j' \in [r']$. Then, for fixed $j \in [r']$, we can compute the value $\mathcal{T}(\ell+1, j)$ in time $O(r)$ using the above recursion. Hence, the total running time of the algorithm is $O(kr^2 + r^3)$. This completes the run time analysis and the proof of Theorem 3.12. □

4 Conclusions and future work

At the level of techniques, this work illustrates the viability of a new general strategy for developing efficient learning algorithms, namely by using "inexpensive" property testers to decompose a complex object (for us these objects are k-modal distributions) into simpler objects (for us these are monotone distributions) that can be more easily learned. It would be interesting to apply this paradigm in other contexts such as learning Boolean functions.

At the level of the specific problem we consider—learning k-modal distributions—our results show that k-modality is a useful type of structure which can be strongly exploited by sample-efficient and computationally efficient learning algorithms. Our results motivate the study of computationally efficient learning algorithms for distributions that satisfy other kinds of "shape restrictions." Possible directions here include multivariate k-modal distributions, log-concave distributions, monotone hazard rate distributions and more.

At a technical level, any improvement in the sample complexity of our property testing algorithm of Section 3.4 would directly improve the "extraneous" additive $\tilde{O}(k^2/\varepsilon^3)$ term in the sample complexity of our algorithm. We suspect that it may be possible to improve our testing algorithm (although we note that it is easy to give an $\Omega(\sqrt{k}/\varepsilon^2)$ lower bound using standard constructions).

Our learning algorithm is not proper, i. e., it outputs a hypothesis that is not necessarily k-modal. Obtaining an efficient proper learning algorithm is an interesting question. Finally, it should be noted that our approach for learning k-modal distributions requires a priori knowledge of the parameter k. We leave the case of unknown k as an intriguing open problem.

Acknowledgement

We thank the anonymous reviewers for their helpful comments.

A Birgé's algorithm as a semi-agnostic learner

In this section we briefly explain why Birgé's algorithm [4] also works in the semi-agnostic setting, thus justifying the claims about its performance made in the statement of Theorem 2.3. To do this, we need to explain his approach. For this, we will need the following fact (which follows as a special case of the VC inequality, Theorem 2.1), which gives a tight bound on the number of samples required to learn an arbitrary distribution with respect to *total variation distance*.

Fact A.1. *Let p be any distribution over $[n]$. We have:* $\mathbf{E}[d_{\mathrm{TV}}(p, \widehat{p}_m)] = O(\sqrt{n/m})$.

Let p be a non-increasing distribution over $[n]$. (The analysis for the non-decreasing case is identical.) Conceptually, we view algorithm $\mathrm{L}^{\downarrow}$ as working in three steps:

- In the first step, it partitions the set $[n]$ into a carefully chosen set $I_1, \dots, I_\ell$ of consecutive intervals, with $\ell = O(m^{1/3} \cdot (\log n)^{2/3})$. Consider the *flattened* distribution p_f over $[n]$ obtained from p by averaging the weight that p assigns to each interval over the entire interval. That is, for $j \in [\ell]$ and $i \in I_j$, $p_f(i) = \sum_{t \in I_j} p(t)/|I_j|$. Then a simple argument given in [4] gives that
$$d_{\mathrm{TV}}(p_f, p) = O\left((\log n/(m+1))^{1/3}\right).$$
- Let p_r be the *reduced* distribution corresponding to p and the partition $I_1, \dots, I_\ell$. That is, p_r is a distribution over $[\ell]$ with $p_r(i) = p(I_i)$ for $i \in [\ell]$. In the second step, the algorithm uses the m samples to learn p_r. (Note that p_r is not necessarily monotone.) After m samples, one obtains a hypothesis $\widehat{p_r}$ such that
$$\mathbf{E}[d_{\mathrm{TV}}(p_r, \widehat{p_r})] = O\left(\sqrt{\ell/m}\right) = O\left((\log n/(m+1))^{1/3}\right).$$
The first equality follows from Fact A.1 (since p_r is distribution over ℓ elements) and the second inequality follows from the choice of ℓ.
- Finally, the algorithm outputs the flattened hypothesis $(\widehat{p_r})_f$ over $[n]$ corresponding to $\widehat{p_r}$, i. e., obtained by $\widehat{p_r}$ by subdividing the mass of each interval uniformly within the interval. It follows from the above two steps that
$$\mathbf{E}[d_{\mathrm{TV}}((\widehat{p_r})_f, p_f)] = O\left((\log n/(m+1))^{1/3}\right).$$
- The combination of the first and third steps yields that
$$\mathbf{E}[d_{\mathrm{TV}}((\widehat{p_r})_f, p)] = O\left((\log n/(m+1))^{1/3}\right).$$

The above arguments are entirely due to Birgé [4]. We now explain how his analysis can be extended to show that his algorithm is in fact a semi-agnostic learner as claimed in Theorem 2.3. To avoid clutter in the expressions below let us fix
$$\delta := O\left((\log n/(m+1))^{1/3}\right).$$

The second and third steps in the algorithm description above are used to learn the distribution p_f to variation distance δ. Note that these steps do not use the assumption that p is non-increasing. The following claim, which generalizes Step 1 above, says that if p is τ-close to non-increasing, the flattened distribution p_f (defined as above) is $(2\tau+\delta)$-close to p. Therefore, it follows that, for such a distribution p, algorithm $\mathrm{L}^{\downarrow}$ succeeds with expected (total variation distance) error $(2\tau+\delta)+\delta$.
We have:

Claim A.2. *Let p be a distribution over $[n]$ that is τ-close to non-increasing. Then, the flattened distribution p_f (obtained from p by averaging its weight on every interval I_j) satisfies*

$$d_{\mathrm{TV}}(p_f, p) \le (2\tau + \delta).$$

Proof. Let $p^{\downarrow}$ be the non-increasing distribution that is τ-close to p. Let τ_j denote the L_1-distance between p and $p^{\downarrow}$ in the interval I_j. Then, we have that

$$\sum_{j=1}^{\ell} \tau_j \le \tau. \tag{A.1}$$

By Birgé's arguments, it follows that the flattened distribution $(p^{\downarrow})_f$ corresponding to $p^{\downarrow}$ is δ-close to $p^{\downarrow}$, hence $(\tau+\delta)$-close to p. That is,

$$d_{\mathrm{TV}}\left((p^{\downarrow})_f, p\right) \le \tau + \delta. \tag{A.2}$$

We want to show that

$$d_{\mathrm{TV}}\left((p^{\downarrow})_f, p_f\right) \le \tau. \tag{A.3}$$

Assuming (A.3) holds, we can conclude by the triangle inequality that

$$d_{\mathrm{TV}}(p, p_f) \le 2\tau + \delta$$

as desired.

Observe that, by assumption, p and $p^{\downarrow}$ have L_1-distance at most τ_j in each I_j interval. In particular, this implies that, for all $j \in [\ell]$, it holds

$$\left| p(I_j) - p^{\downarrow}(I_j) \right| \le \tau_j.$$

Now note that, within each interval I_j, p_f and $(p^{\downarrow})_f$ are both uniform. Hence, the contribution of I_j to the variation distance between p_f and $(p^{\downarrow})_f$ is at most $|p(I_j) - p^{\downarrow}(I_j)|$.

Therefore, by (A.1) we deduce

$$d_{\mathrm{TV}}(p_f, (p^{\downarrow})_f) \le \tau$$

which completes the proof of the claim. □

B Hypothesis testing

Our hypothesis testing routine `Choose-Hypothesis`^{p} runs a simple "competition" to choose a winner between two candidate hypothesis distributions h_1 and h_2 over $[n]$ that it is given in the input either explicitly, or in some succinct way. We show that if at least one of the two candidate hypotheses is close to the target distribution p, then with high probability over the samples drawn from p the routine selects as winner a candidate that is close to p. This basic approach of running a competition between candidate hypotheses is quite similar to the "Scheffé estimate" proposed by Devroye and Lugosi (see [10, 11] and Chapter 6 of [12]), which in turn built closely on the work of [27], but there are some small differences between our approach and theirs; the [12] approach uses a notion of the "competition" between two hypotheses which is not symmetric under swapping the two competing hypotheses, whereas our competition is symmetric.

We now prove Theorem 2.4.

Proof of Theorem 2.4. Let $\mathcal{W}$ be the support of p. To set up the competition between h_1 and h_2, we define the following subset of $\mathcal{W}$:

$$\mathcal{W}_1 = \mathcal{W}_1(h_1, h_2) := \{w \in \mathcal{W} \mid h_1(w) > h_2(w)\} . \tag{B.1}$$

Let then $p_1 = h_1(\mathcal{W}_1)$ and $q_1 = h_2(\mathcal{W}_1)$. Clearly, $p_1 > q_1$ and $d_{\mathrm{TV}}(h_1, h_2) = p_1 - q_1$.

The competition between h_1 and h_2 is carried out as follows:

1. If $p_1 - q_1 \leq 5\varepsilon'$, declare a draw and return either h_i. Otherwise:
2. Draw
$$m = O\left(\frac{\log(1/\delta')}{\varepsilon'^2}\right)$$
samples $s_1, \ldots, s_m$ from p, and let
$$\tau = \frac{1}{m}\left|\{i \mid s_i \in \mathcal{W}_1\}\right|$$
be the fraction of samples that fall inside $\mathcal{W}_1$.
3. If $\tau > p_1 - \frac{3}{2}\varepsilon'$, declare h_1 as winner and return h_1; otherwise,
4. if $\tau < q_1 + \frac{3}{2}\varepsilon'$, declare h_2 as winner and return h_2; otherwise,
5. declare a draw and return either h_i.

It is not hard to check that the outcome of the competition does not depend on the ordering of the pair of distributions provided in the input; that is, on inputs (h_1, h_2) and (h_2, h_1) the competition outputs the same result for a fixed sequence of samples $s_1, \ldots, s_m$ drawn from p.

The correctness of `Choose-Hypothesis` is an immediate consequence of the following lemma.

Lemma B.1. *Suppose that $d_{\mathrm{TV}}(p, h_1) \leq \varepsilon'$. Then:*

(i) If $d_{\mathrm{TV}}(p, h_2) > 6\varepsilon'$, then the probability that the competition between h_1 and h_2 does not declare h_1 as the winner is at most $e^{-m\varepsilon'^2/2}$. (Intuitively, if h_2 is very bad then it is very likely that h_1 will be declared winner.)

(ii) If $d_{\mathrm{TV}}(p, h_2) > 4\varepsilon'$, the probability that the competition between h_1 and h_2 declares h_2 as the winner is at most $e^{-m\varepsilon'^2/2}$. (Intuitively, if h_2 is only moderately bad then a draw is possible but it is very unlikely that h_2 will be declared winner.)

Proof. Let $r = p(\mathcal{W}_1)$. The definition of the total variation distance implies that $|r - p_1| \leq \varepsilon'$. Let us define the $0/1$ (indicator) random variables $\{Z_j\}_{j=1}^m$ as $Z_j = 1$ iff $s_j \in \mathcal{W}_1$. Clearly,

$$\tau = \frac{1}{m}\sum_{j=1}^{m} Z_j \qquad \text{and} \qquad \mathbb{E}[\tau] = \mathbb{E}[Z_j] = r\,.$$

Since the Z_j's are mutually independent, it follows from the Chernoff bound that

$$\Pr[\tau \leq r - \varepsilon'/2] \leq e^{-m\varepsilon'^2/2}\,.$$

Using $|r - p_1| \leq \varepsilon'$ we get that

$$\Pr[\tau \leq p_1 - 3\varepsilon'/2] \leq e^{-m\varepsilon'^2/2}\,.$$

- For part (i): If $d_{\mathrm{TV}}(p, h_2) > 6\varepsilon'$, from the triangle inequality we get that $p_1 - q_1 = d_{\mathrm{TV}}(h_1, h_2) > 5\varepsilon'$. Hence, the algorithm will go beyond Step 1, and with probability at least $1 - e^{-m\varepsilon'^2/2}$, it will stop at Step 3, declaring h_1 as the winner of the competition between h_1 and h_2.

- For part (ii): If $p_1 - q_1 \leq 5\varepsilon'$ then the competition declares a draw, hence h_2 is not the winner. Otherwise we have $p_1 - q_1 > 5\varepsilon'$ and the above arguments imply that the competition between h_1 and h_2 will declare h_2 as the winner with probability at most $e^{-m\varepsilon'^2/2}$.

This concludes the proof of Lemma B.1. □

The proof of the theorem is now complete. □

C Using the hypothesis tester

In this section, we explain in detail how we use the hypothesis testing algorithm `Choose-Hypothesis` throughout this paper. In particular, the algorithm `Choose-Hypothesis` is used in the following places:

- In Step 4 of algorithm `Learn-kmodal-simple` we need an algorithm $\mathrm{L}^{\downarrow}{}_{\delta'}$ (resp. $\mathrm{L}^{\uparrow}{}_{\delta'}$) that learns a non-increasing (resp. non-increasing) distribution within total variation distance ε and confidence δ'. Note that the corresponding algorithms $\mathrm{L}^{\downarrow}$ and $\mathrm{L}^{\uparrow}$ provided by Theorem 2.3 have confidence $9/10$. To boost the confidence of $\mathrm{L}^{\downarrow}$ (resp. $\mathrm{L}^{\uparrow}$) we run the algorithm $O(\log(1/\delta'))$ times and use `Choose-Hypothesis` in an appropriate tournament procedure to select among the candidate hypothesis distributions.

- In Step 5 of algorithm `Learn-kmodal-simple` we need to select among two candidate hypothesis distributions (with the promise that at least one of them is close to the true conditional distribution). In this case, we run `Choose-Hypothesis` once to select between the two candidates.

- Also note that both algorithms `Learn-kmodal-simple` and `Learn-kmodal` generate an ε-accurate hypothesis with probability $9/10$. We would like to boost the probability of success to $1-\delta$. To achieve this we again run the corresponding algorithm $O(\log(1/\delta))$ times and use `Choose-Hypothesis` in an appropriate tournament to select among the candidate hypothesis distributions.

We now formally describe the "tournament" algorithm to boost the confidence to $1-\delta$.

Lemma C.1. *Let p be any distribution over a finite set $\mathcal{W}$. Suppose that $\mathcal{D}_\varepsilon$ is a collection of N distributions over $\mathcal{W}$ such that there exists $q \in \mathcal{D}_\varepsilon$ with $d_{\mathrm{TV}}(p,q) \le \varepsilon$. Then there is an algorithm that uses $O(\varepsilon^{-2} \log N \log(1/\delta))$ samples from p and with probability $1-\delta$ outputs a distribution $p' \in \mathcal{D}_\varepsilon$ that satisfies $d_{\mathrm{TV}}(p,p') \le 6\varepsilon$.*

Devroye and Lugosi (Chapter 7 of [12]) prove a similar result by having all pairs of distributions in the cover compete against each other using their notion of a competition, but again there are some small differences: their approach chooses a distribution in the cover which wins the maximum number of competitions, whereas our algorithm chooses a distribution that is never defeated (i. e., won or achieved a draw against all other distributions in the cover). Instead we follow the approach from [9].

Proof. The algorithm performs a tournament by running the competition $\texttt{Choose-Hypothesis}^p(h_i, h_j, \varepsilon, \delta/(2N))$ for every pair of distinct distributions h_i, h_j in the collection $\mathcal{D}_\varepsilon$. It outputs a distribution $q^\star \in \mathcal{D}_\varepsilon$ that was never a loser (i. e., won or achieved a draw in all its competitions). If no such distribution exists in $\mathcal{D}_\varepsilon$ then the algorithm outputs "failure."

By definition, there exists some $q \in \mathcal{D}_\varepsilon$ such that $d_{\mathrm{TV}}(p,q) \le \varepsilon$. We first argue that with high probability this distribution q never loses a competition against any other $q' \in \mathcal{D}_\varepsilon$ (so the algorithm does not output "failure"). Consider any $q' \in \mathcal{D}_\varepsilon$. If $d_{\mathrm{TV}}(p,q') > 4\varepsilon$, by Lemma B.1(ii) the probability that q loses to q' is at most $2e^{-m\varepsilon^2/2} = O(1/N)$. On the other hand, if $d_{\mathrm{TV}}(p,q') \le 4\delta$, the triangle inequality gives that $d_{\mathrm{TV}}(q,q') \le 5\varepsilon$ and thus q draws against q'. A union bound over all N distributions in $\mathcal{D}_\varepsilon$ shows that with probability $1-\delta/2$, the distribution q never loses a competition.

We next argue that with probability at least $1-\delta/2$, every distribution $q' \in \mathcal{D}_\varepsilon$ that never loses has small variation distance from p. Fix a distribution q' such that $d_{\mathrm{TV}}(q',p) > 6\varepsilon$; Lemma B.1(i) implies that q' loses to q with probability $1 - 2e^{-m\varepsilon^2/2} \ge 1 - \delta/(2N)$. A union bound gives that with probability $1-\delta/2$, every distribution q' that has $d_{\mathrm{TV}}(q',p) > 6\varepsilon$ loses some competition.

Thus, with overall probability at least $1-\delta$, the tournament does not output "failure" and outputs some distribution $q^\star$ such that $d_{\mathrm{TV}}(p,q^\star)$ is at most 6ε. This proves the lemma. □

We now explain how the above lemma is used in our context: Suppose we perform $O(\log(1/\delta))$ runs of a learning algorithm that constructs an ε-accurate hypothesis with probability at least $9/10$. Then, with failure probability at most $\delta/2$, at least one of the hypotheses generated is ε-close to the true distribution in variation distance. Conditioning on this good event, we have a collection of distributions with cardinality $O(\log(1/\delta))$ that satisfies the assumption of the lemma. Hence, using $O\left((1/\varepsilon^2) \cdot \log\log(1/\delta) \cdot \log(1/\delta)\right)$ samples we can learn to accuracy 6ε and confidence $1-\delta/2$. The overall sample complexity is $O(\log(1/\delta))$ times the sample complexity of the learning algorithm run with confidence $9/10$, plus this additional $O\left((1/\varepsilon^2) \cdot \log\log(1/\delta) \cdot \log(1/\delta)\right)$ term.

In terms of running time,we make the following easily verifiable remarks: When the hypothesis testing algorithm `Choose-Hypothesis` is run on a pair of distributions that are produced by Birgé's algorithm, its running time is polynomial in the succinct description of these distributions, i. e., in $\log^2(n)/\varepsilon$. Similarly, when `Choose-Hypothesis` is run on a pair of outputs of `Learn-kmodal-simple` or `Learn-kmodal`, its running time is polynomial in the succinct description of these distributions. More specifically, in the former case, the succinct description has bit complexity $O\left(k\cdot\log^2(n)/\varepsilon^2\right)$ (since the output consists of $O(k/\varepsilon)$ monotone intervals, and the conditional distribution on each interval is the output of Birgé's algorithm for that interval). In the latter case, the succinct description has bit complexity $O\left(k\cdot\log^2(n)/\varepsilon\right)$, since the algorithm `Learn-kmodal` constructs only k monotone intervals. Hence, in both cases, each execution of the testing algorithm performs $\mathrm{poly}(k,\log n,1/\varepsilon)$ bit operations. Since the tournament invokes the algorithm `Choose-Hypothesis` $O(\log^2(1/\delta))$ times (for every pair of distributions in our pool of $O(\log(1/\delta))$ candidates) the upper bound on the running time follows.

References

[1] GUILLERMO ALVAREZ DE TOLEDO AND JULIO M. FERNANDEZ: Patch-clamp measurements reveal multimodal distribution of granule sizes in rat mast cells. *J. Cell Biology*, 110(4):1033–1039, 1990. [doi:10.1083/jcb.110.4.1033] 536

[2] TUĞKAN BATU, RAVI KUMAR, AND RONITT RUBINFELD: Sublinear algorithms for testing monotone and unimodal distributions. In *Proc. 36th STOC*, pp. 381–390. ACM Press, 2004. [doi:10.1145/1007352.1007414] 548

[3] LUCIEN BIRGÉ: Estimating a density under order restrictions: Nonasymptotic minimax risk. *Annals of Statistics*, 15(3):995–1012, 1987. Available at JSTOR. 536, 537

[4] LUCIEN BIRGÉ: On the risk of histograms for estimating decreasing densities. *Annals of Statistics*, 15(3):1013–1022, 1987. Available at JSTOR. 536, 537, 540, 541, 561, 562

[5] LUCIEN BIRGÉ: Estimation of unimodal densities without smoothness assumptions. *Annals of Statistics*, 25(3):970–981, 1997. Available at JSTOR. 536

[6] KUNG-SIK CHAN AND HOWELL TONG: Testing for multimodality with dependent data. *Biometrika*, 91(1):113–123, 2004. [doi:10.1093/biomet/91.1.113] 536

[7] LOREN COBB, PETER KOPPSTEIN, AND NENG HSIN CHEN: Estimation and moment recursion relations for multimodal distributions of the exponential family. *J. American Statistical Association*, 78(381):124–130, 1983. [doi:10.1080/01621459.1983.10477940] 536

[8] CONSTANTINOS DASKALAKIS, ILIAS DIAKONIKOLAS, AND ROCCO A. SERVEDIO: Learning k-modal distributions via testing. In *Proc. 23rd Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'12)*, pp. 1371–1385. ACM Press, 2012. Available at ACM DL. 535

[9] CONSTANTINOS DASKALAKIS, ILIAS DIAKONIKOLAS, AND ROCCO A. SERVEDIO: Learning Poisson binomial distributions. In *Proc. 44th STOC*, pp. 709–728. ACM Press, 2012. [doi:10.1145/2213977.2214042, arXiv:1107.2702] 541, 566

[10] LUC DEVROYE AND GÁBOR LUGOSI: A universally acceptable smoothing factor for kernel density estimates. *Annals of Statistics*, 24(6):2499–2512, 1996. Available at JSTOR. 564

[11] LUC DEVROYE AND GÁBOR LUGOSI: Nonasymptotic universal smoothing factors, kernel complexity and Yatracos classes. *Annals of Statistics*, 25(6):2626–2637, 1997. Available at JSTOR. 564

[12] LUC DEVROYE AND GÁBOR LUGOSI: *Combinatorial Methods in Density Estimation*. Springer, 2001. 540, 541, 564, 566

[13] FRANCESCO R. FERRARO, BARBARA PALTRINIERI, FLAVIO FUSI PECCI, ROBERT T. ROOD, AND BEN DORMAN: Multimodal distributions along the horizontal branch. *The Astrophysical Journal*, 500(1):311–319, 1998. [doi:10.1086/305712] 536

[14] ODED GOLDREICH, editor. *Property Testing: Current Research and Surveys*. Springer, 2010. LNCS 6390. 538

[15] ODED GOLDREICH: Highlights of the Bertinoro workshop on Sublinear Algorithms (unpublished comments). Available at author's website, 2011. 538

[16] ODED GOLDREICH, SHAFI GOLDWASSER, AND DANA RON: Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998. Preliminary version at FOCS'96. [doi:10.1145/285055.285060] 538

[17] PIET GROENEBOOM: Estimating a monotone density. In *Proc. of the Berkeley Conference in Honor of Jerzy Neyman and Jack Kiefer*, pp. 539–555, 1985. Available at CWI. 536

[18] MICHAEL J. KEARNS AND DANA RON: Testing problems with sublearning sample complexity. *J. Comput. System Sci.*, 61(3):428–456, 2000. Preliminary version in COLT'98. [doi:10.1006/jcss.1999.1656] 538

[19] JOHANNES H. B. KEMPERMAN: Mixtures with a limited number of modal intervals. *Annals of Statistics*, 19(4):2120–2144, 1991. Available at JSTOR. 536

[20] EDMOND A. MURPHY: One cause? many causes?: The argument from the bimodal distribution. *J. Chronic Diseases*, 17(4):301–324, 1964. [doi:10.1016/0021-9681(64)90073-6] 536

[21] ROGER B. MYERSON: Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981. [doi:10.1287/moor.6.1.58] 555, 558

[22] DONALD J. NEWMAN AND LAWRENCE SHEPP: The double dixie cup problem. *American Mathematical Monthly*, 67(1):58–61, 1960. [doi:10.2307/2308930] 546, 551

[23] B. L. S. PRAKASA RAO: Estimation of a unimodal density. *Sankhya Ser. A*, 31(1):23–36, 1969. Available at JSTOR. 536

[24] DANA RON: Property Testing: A Learning Theory Perspective. *Foundations and Trends in Machine Learning*, 1(3):307–402, 2008. [doi:10.1561/2200000004] 538

[25] DANA RON: Algorithmic and analysis techniques in property testing. *Foundations and Trends in Theoretical Computer Science*, 5:73–205, 2010. [doi:10.1561/0400000029] 538

[26] EDWARD J. WEGMAN: Maximum likelihood estimation of a unimodal density. I. and II. *Ann. Math. Statist.*, 41:457–471, 2169–2174, 1970. Available at JSTOR. 536

[27] YANNIS G. YATRACOS: Rates of convergence of minimum distance estimators and Kolmogorov's entropy. *Annals of Statistics*, 13(2):768–774, 1985. Available at JSTOR. 564

AUTHORS

Constantinos Daskalakis
Associate professor
Department of Electrical Engineering and Computer Science
Massachussetts Institute of Technology
costis@csail.mit.edu
http://people.csail.mit.edu/costis/

Ilias Diakonikolas
Assistant professor
School of Informatics
University of Edinburgh
ilias.d@ed.ac.uk
http://www.iliasdiakonikolas.org

Rocco A. Servedio
Associate professor
Department of Computer Science Columbia University
rocco@cs.columbia.edu
http://www.cs.columbia.edu/~rocco

ABOUT THE AUTHORS

CONSTANTINOS DASKALAKIS (addressed as "Costis" by friends and colleagues) is an associate professor in the Department of Electrical Engineering and Computer Science at MIT. He holds a diploma in Electrical and Computer Engineering from the National Technical University of Athens, and a Ph. D. in Electrical Engineering and Computer Sciences from UC Berkeley, where his Ph. D. was supervised by Christos Papadimitriou. His research interests lie in the interface of computation with economics and probability, in particular the complexity of equilibria and fixed points, mechanism design, and machine learning.

ILIAS DIAKONIKOLAS is an assistant professor in the School of Informatics at the University of Edinburgh. He obtained a diploma in Electrical and Computer Engineering from the National Technical University of Athens, and a Ph. D. in Computer Science from Columbia University advised by Mihalis Yannakakis. His research interests lie in algorithms, learning, statistics, and game theory. The current focus of his work is on algorithms for machine learning.

ROCCO SERVEDIO is an associate professor in the Department of Computer Science at Columbia University. He graduated from Harvard University, where his Ph. D. was supervised by Les Valiant. He is interested in computational learning theory, computational complexity, and property testing, with the study of Boolean functions as an underlying theme tying these topics together. He enjoys spending time with his family and hopes to share a bottle of Cutty Sark with H. M. in the afterlife.