Non-Stochastic CDF Estimation Using Threshold Queries

Princewill Okoroafor[1], Vaishnavi Gupta[1], Robert Kleinberg[1], and Eleanor Goh[1]

[1]Cornell University, Ithaca, NY

pco9@cornell.edu, vg222@cornell.edu, rdk@cs.cornell.edu, eg552@cornell.edu

Abstract

Estimating the empirical distribution of a scalar-valued data set is a basic and fundamental task. In this paper, we tackle the problem of estimating an empirical distribution in a setting with two challenging features. First, the algorithm does not directly observe the data; instead, it only asks a limited number of threshold queries about each sample. Second, the data are not assumed to be independent and identically distributed; instead, we allow for an arbitrary process generating the samples, including an adaptive adversary. These considerations are relevant, for example, when modeling a seller experimenting with posted prices to estimate the distribution of consumers' willingness to pay for a product: offering a price and observing a consumer's purchase decision is equivalent to asking a single threshold query about their value, and the distribution of consumers' values may be non-stationary over time, as early adopters may differ markedly from late adopters.

Our main result quantifies, to within a constant factor, the sample complexity of estimating the empirical CDF of a sequence of elements of $[n]$, up to ε additive error, using one threshold query per sample. The complexity depends only logarithmically on n, and our result can be interpreted as extending the existing logarithmic-complexity results for noisy binary search to the more challenging setting where noise is non-stochastic. Along the way to designing our algorithm, we consider a more general model in which the algorithm is allowed to make a limited number of simultaneous threshold queries on each sample. We solve this problem using Blackwell's Approachability Theorem and the exponential weights method. As a side result of independent interest, we characterize the minimum number of simultaneous threshold queries required by deterministic CDF estimation algorithms.

1 Introduction

Estimating the empirical distribution of a scalar-valued data set is a basic and fundamental task. For example, estimating quantiles of a data stream is one of the oldest and most well-studied problems in streaming algorithms (Greenwald and Khanna, 2001; Karnin et al., 2016; Manku et al., 1998; Munro and Paterson, 1980), with applications to databases (Greenwald and Khanna, 2001), network health monitoring (Cormode et al., 2004), and wireless sensor networks (Shrivastava et al., 2004), among others. Ideally, a data analyst would like to be able to assume that the data values are independent and identically distributed, and that they are directly observable. However, these assumptions might be violated in applications of interest.

1. In many settings, samples can only be evaluated indirectly by comparing them to specified thresholds and learning whether or not each sample is less than or equal to its corresponding threshold. This is the case, for instance, when a seller experiments with varying posted prices in order to estimate the distribution of consumers' willingness to pay for a product or service. Other examples arise when eliciting information about the distribution of individuals' abilities using pass-fail tests with a variable level of difficulty (e.g. swimming tests) or when evaluating the quality of a new product by asking consumers to compare it against products of known quality. (There is ample evidence in the behavioral sciences that human subjects' quality judgments can be elicited more reliably with ordinal comparisons than with subjective numerical ratings (Ali and Ronaldson, 2012; Chapelle et al., 2012; Larichev et al., 1995; Moshkovich et al., 2002).)

2. The assumption that samples are independent and identically distributed may also be violated. Returning to the posted-pricing application, early adopters of a product might differ markedly from late adopters in their willingness to pay for the product, and the late adopters' willingness to pay may even depend on the rate of adoption by earlier consumers, which in turn depends on the posted prices they were offered. A similar application arises in an auction setting with repeated bidding - an internet advertiser estimates the distribution of winning bids by varying their own bid. To account for complex influences on the behavior of other bidders, assuming a worst-case input sequence rather than i.i.d. is very useful (Weed et al., 2016).

In this work we tackle the problem of estimating the empirical distribution of a sequence of numbers using threshold queries, in a non-stochastic setting that makes no assumptions about the process by which the sequence is generated. Our model even allows the sequence to be constructed by an adaptive adversary. We assume the algorithm asks one threshold query about each element of the sequence, and the query and its answer are revealed to both parties before the next element of the sequence is generated by the adversary. The key question we aim to resolve is: *what is the sample complexity of estimating the empirical CDF of a distribution on* $[n] = \{1, 2, \ldots, n\}$ *to within* ε*?* In more detail, what is the smallest T such that there exists a randomized algorithm that succeeds, with probability at least $3/4$, in learning an estimate of the empirical CDF of an arbitrary sequence $x_1, \ldots, x_T$ that differs from the true empirical CDF (in L_∞ norm) by at most ε? In this paper, we resolve the question to within a constant factor, by proving asymptotically matching upper and lower bounds. In fact, our lower bound is valid even in a stochastic setting where the elements $x_1, \ldots, x_T$ are i.i.d. samples from a distribution on $[n]$. Hence our results reveal, perhaps surprisingly, that *up to a constant factor, there is no difference in the sample complexity of CDF estimation in the stochastic and non-stochastic settings.*

1.1 Relation to noisy binary search and median estimation

Let us say that $m \in [n]$ is an *ε-approximate median* of the sequence $x_1, x_2, \ldots, x_T$ if at least $(\frac{1}{2} - \varepsilon)T$ elements of the sequence are less than or equal to m and at least $(\frac{1}{2} - \varepsilon)T$ of them are greater than or equal to m. Approximate median estimation reduces to approximate CDF estimation: if $\hat{F}$ is an ε-accurate estimate of the empirical CDF of $x_1, \ldots, x_T$ then an index m that satisfies $\hat{F}(m-1) < \frac{1}{2} \leq \hat{F}(m)$ is an approximate median.

In the special case when $x_1, x_2, \ldots, x_T$ is restricted to be a constant sequence, CDF estimation and median estimation both become equivalent to binary search: the empirical CDF is a $\{0, 1\}$-valued step function with a step at some $x \in [n]$ and x is the unique approximate median, so both tasks become equivalent to identifying the value of x using queries of the form $x \stackrel{?}{\leq} q_t$. The problem our work addresses can thus be interpreted as a generalization of binary search in which the answers to comparison queries are perturbed by non-stochastic noise.

One easy consequence of this connection to binary search is a lower bound of $\lfloor \log_2(n) \rfloor$ on the sample complexity of approximate CDF estimation and approximate median estimation. In the important special case when ε is a small constant (e.g., $\varepsilon = 0.01$), the algorithms we present in this paper match this trivial lower bound to within a constant factor, exponentially improving the best previously known bounds for CDF estimation and median estimation in the non-stochastic setting.

1.2 Techniques

Given that CDF estimation generalizes binary search and that the sample complexity bound we are aiming for — $O(\log n)$ in the case of constant ε — matches the query complexity of binary search, a natural idea is to try designing CDF estimation algorithms with a recursive structure resembling that of binary search. Indeed, in the stochastic setting, Karp and Kleinberg (2007) presented a median estimation algorithm, based on binary search with backtracking, whose sample complexity is $O(\log n)$ when ε is constant. Using this algorithm as a subroutine, Meister and Nietert (2021) showed how to solve CDF estimation in the stochastic setting at the cost of an additional $1/\varepsilon$ factor in sample complexity. In the non-stochastic setting, one can similarly attempt to base CDF estimation or median estimation on divide-and-conquer strategies that zero in on intervals where the density of samples is high. However, there is an obvious difficulty: the past samples need not have any relation to those in the present and future. Thus, focusing on intervals that contained many past samples could draw the algorithm's attention away from the intervals containing most of the present samples, making it impossible to maintain an accurate CDF estimate. We are not aware of any way to overcome this difficulty and base a non-stochastic CDF estimation algorithm on the principle of divide-and-conquer with backtracking.

Instead, to design our algorithm we take a detour through a more general model in which the CDF estimation algorithm is allowed to make k simultaneous threshold queries for each sample. When $k = 1$ this matches our original model, but when k exceeds $\frac{1}{\varepsilon}$ the problem undergoes an interesting qualitative change: it becomes solvable by deterministic algorithms. To solve it, we show that the problem of using threshold queries to compute a CDF estimate that is accurate with probability 1 is equivalent to a question about the approachability of a convex set in a two-player game with vector payoffs. Blackwell's Approachability Theorem gives us a criterion for determining the number of simultaneous queries necessary to solve CDF estimation using a Las Vegas randomized algorithm that almost surely terminates and outputs an ε-accurate answer. Using the exponential weight approachability algorithm of Perchet (2015), we show that this objective can in fact

be achieved by a deterministic algorithm in only $O(\log(n)/\varepsilon)$ rounds, with $O(1/\varepsilon)$ simultaneous queries per round. We believe the design and analysis of this deterministic, simultaneous-query algorithm for CDF estimation may be of independent interest. It is also a vital step in designing a randomized algorithm that solves CDF estimation in the original non-stochastic setting with only one threshold query per sample. Our algorithm for that problem can be interpreted as a randomized simulation of the deterministic simultaneous-query algorithm: it randomly samples one of the $O(1/\varepsilon)$ simultaneous queries recommended by the deterministic algorithm, then uses importance weighting to produce an unbiased estimate of the payoff vector that would have resulted from making all of the recommended queries simultaneously.

1.3 Related work

As noted earlier, our problem generalizes noisy binary search to a setting with non-stochastic noise. The first paper to study this generalization is by Meister and Nietert (2021), who proved a sample complexity upper bound $O(n\log(n)/\varepsilon^2)$ using a naïve algorithm that queries a uniformly random threshold $q_t \in [n]$ at each time $t \in [T]$ and estimates $\hat{F}(i)$ by simply averaging the values observed in the time steps t when $q_t = i$. In other words, the naïve algorithm breaks down the problem of estimating a CDF over $[n]$ into n independent point-estimation problems, one for each $i \in [n]$, which are each solved by directly querying $F(i)$ often enough that the average of the sampled queries approximates the population average. This ignores the fact that the empirical CDF must be a monotone function, and that shape constraints such as monotonicity typically improve the sample complexity of estimation (Barlow et al., 1972). Perhaps surprisingly, Meister and Nietert showed that when $\varepsilon = O(1/n)$, there is a lower bound for ε-accurate CDF estimation that matches the naïve algorithm's sample complexity up to a constant factor. This still left an exponential gap between the upper and lower bounds for the case of general $\varepsilon > 0$. Our work closes this gap, proving a tight bound (up to constant factors) for all n and $\varepsilon > 0$ which exponentially improves the Meister-Nietert upper bound in the case $\varepsilon = \Omega(1)$. Prior to our work, it was not known whether CDF estimation algorithms could obtain any asymptotic improvement at all over the naïve algorithm.

The earliest works on noisy binary search assumed a stochastic noise model that correctly answers each comparison query with probability $\frac{1}{2} + \varepsilon$ and otherwise flips the answer. In this model, an algorithm with sample complexity $O(\log(n)/\varepsilon^2)$ was presented and analyzed by Burnashev and Zigangirov (1974), who actually showed that their algorithm's complexity is optimal up to a $1 + o(1)$ factor. An even more precise sample complexity bound for the same algorithm was later provided by Ben-Or and Hassidim (2008). In the same model of stochastic noise, Feige et al. (1994) provided a different noisy binary search algorithm with $O(\log(n)/\varepsilon^2)$ complexity; they also supplied algorithms for a number of other fundamental problems such as sorting in the same noisy comparison model. Karp and Kleinberg (2007) generalized the stochastic noisy comparison model to a setting in which the probability of a correct answer to a query depends on the identities of the elements being compared, but is always greater than $\frac{1}{2}$, and they showed that the $O(\log(n)/\varepsilon^2)$ sample complexity bound for noisy binary search continues to hold in this setting. Meister and Nietert (2021) showed how to use the Karp-Kleinberg noisy binary search algorithm as a subroutine in a CDF estimation algorithm that achieves sample complexity $O(\log(n)/\varepsilon^3)$ when the adversary is stochastic. (In the notation introduced earlier, this means the sequence $x_1, x_2, \ldots, x_T$ is created by drawing i.i.d. samples from a fixed but unknown distribution.)

Many works in the literature have studied distribution learning under specific constraints. Han et al. (2018) and Barnes et al. (2020) proved various minimax lower bounds for the problem of learning structured distributions in distributed networks. In their setting, every node in the network observes one independent sample

drawn from the underlying distribution, and there is a central processor to which every node in the network communicates k bits. Acharya et al. (2020) significantly generalized Barnes et al.'s result, presenting unified lower bounds for distributed parametric estimation under a wide variety of local information constraints including communication, privacy, and data access constraints. They modeled their setting by considering a set of channels through which samples are passed to obtain observations. Using this general model, Acharya et al. were able to recover the bounds presented by Barnes et al.. A similar setting where n nodes can observe m samples and communicate information using l bits was presented and analyzed in Acharya et al. (2021). Like previous works, we model our problem as distribution learning under a set of constraints. Rather than focusing on communication channels in distributed models, our constraints are defined on the sequence of threshold queries the algorithm can make.

The distribution estimation problem is also studied within the context of online dynamic pricing and auctions. A seller repeatedly interacts with a buyer by setting prices for an item and observing whether the buyer purchases or not. An auctioneer learns a winning distribution by adaptively choosing reserve prices. Although both of these settings are limited to binary feedback: observing whether the item is bought, the literature (Kleinberg and Leighton, 2003; Leme et al., 2021a; Blum et al., 2015; Leme et al., 2021b) in these contexts assume the distribution is fixed or slowly-changing and often while trying to minimize a notion of regret with respect to the best fixed price in hindsight. We don't make any assumptions about the distribution.

2 Threshold Query Model

We start by describing the CDF estimation problem as defined by Meister and Nietert (2021). At each time step t, the adversary produces a sample $x_t \in [n+1]$, and the algorithm $\mathcal{A}$ generates a query $q_t \in [n]$, each ignorant of the other's choice. Then, the algorithm receives feedback $\mathbf{1}(x_t \le q_t)$ and produces a CDF estimate $\hat{F}_t$ of $x_1, \ldots, x_t$ while q_t is revealed to the adversary. The adversary is allowed to be adaptive, i.e may select x_t based on the history of the prior $t-1$ time steps. Let F_t, defined by $F_t(i) = \frac{1}{t}\sum_{\tau=1}^{t} \mathbf{1}(x_\tau \le i)$ be the empirical CDF of the sequence $x_1, \ldots, x_t$, where $F_t(0) = 0$. They define the estimation error of the algorithm at time t as the Kolmogorov-Smirnov distance between the empirical CDF, F_t, and the algorithm's estimate $\hat{F}_t : [n] \to [0,1]$. In other words, the estimation error is $\|\hat{F}_t - F_t\|_\infty := \sup_{i\in[n]} |\hat{F}_t(i) - F_t(i)|$.

[•] We generalize this formulation in two ways. First, we allow the adversary to pick any monotone function $v_t : [n] \to [0,1]$ at each time step instead of a sample $x_t \in [n]$. This generalizes the original setting since $v_t(i) = \mathbf{1}(x_t \le i)$ is a monotone step function. Thus, picking $x_t \in [n]$ is equivalent to choosing a monotone step function. Then, F_t can easily be redefined to be the empirical average of the monotone functions instead, i.e. $F_t(i) = \frac{1}{t}\sum_{\tau=1}^{t} v_\tau(i)$. Second, rather than insisting that the algorithm must make only one query per sample, we [•] allow the algorithm a specified number of queries per sample, where this query budget could be anywhere between 1 and n. We now proceed to formalize this Threshold Query Model (TQM). Consider an online estimation environment defined by parameters n, k, where the timing of each round t is as follows:

[vg: i added a newline here]

pco deleted here

1. Adversary chooses a monotone function $v_t : [n] \to [0,1]$.
2. Algorithm chooses a set of (up to) k query points: $q_{1,t} \le \ldots \le q_{k,t} \in [n]$; adversary observes these query points.
3. Algorithm receives feedback: $y_{1,t} \le \ldots \le y_{k,t} \in [0,1]$, where $y_{i,t} = v_t(q_{i,t})$.

We will refer to an environment with this interaction structure as the k-TQM, and we will refer to the parameter k as the *query budget*. Since much of our focus is on the case when the query budget equals 1, we will refer to the 1-TQM simply as the TQM.

At the end of the T rounds, the algorithm returns a function $G_T : [n] \to [0,1]$. We say this algorithm has *accuracy* ε and *sample complexity* T if it satisfies the guarantee that for all $\tau \geq T$, the probability that $\|G_\tau - F_\tau\| \leq \varepsilon$ is at least $\frac{3}{4}$, against any (potentially adaptive) adversary. For brevity, we will sometimes refer to an algorithm with accuracy ε and sample complexity T as an (ε, T)-algorithm. We are interested in the following questions:

1. For a fixed accuracy ε and query budget k, what is the minimum sample complexity? In other words, what is the smallest T for which there exists an (ε, T)-algorithm for the k-TQM?
2. For a fixed accuracy ε and sample complexity T, how large must the query budget be? In other words, what is the smallest k for which there exists a (ε, T)-algorithm for the k-TQM?

For $k = 1$, our work shows that the answer to the first question is $O\left(\frac{\log n}{\varepsilon^3}\right)$, which is tight up to a constant factor. We also resolve the second question precisely when T is sufficiently large as a function of n and ε, showing that query budget $k = \frac{1}{2\varepsilon} - 1$ is necessary and sufficient for deterministic algorithms and that query budget $k = 1$ is necessary and sufficient for randomized algorithms.

For simplicity, we shall say a pair (k, T) is **achievable** if there is an (ε, T)-algorithm for the k-TQM. Algorithms for the k-TQM may be deterministic or randomized. Observe that an ε-accurate deterministic algorithm for the k-TQM must achieve the guarantee that $\|G_\tau - F_\tau\|_\infty \leq \varepsilon$ *with probability 1* for all $\tau \geq T$ and all adversaries.

2.1 Deterministic Algorithms for the Threshold Query Model

The Meister and Nietert (2021) result can be interpreted as an $\left(\varepsilon, O\left(\frac{n \log n}{\varepsilon^2}\right)\right)$-algorithm for the TQM. In contrast, there does not exist a deterministic algorithm for the TQM. In fact, by using a pigeonhole argument[1] one can show that deterministic algorithms with accuracy ε and finite sample complexity must have query budget $k \geq \frac{1}{2\varepsilon} - 1$. This shows a qualitative distinction between deterministic and randomized algorithms for the k-TQM, when k is small.

Perhaps the most important reason we study deterministic algorithms with query budget $k > 1$ is that it informs our design of a randomized algorithm with query budget 1. In fact, our algorithms for the TQM (i.e., with query budget 1) work by simulating a deterministic algorithm with a larger query budget. The algorithm in Meister and Nietert (2021) can be thought of as simulating a deterministic algorithm with query budget n that simply queries every point at each timestep. This observation suggests a strategy for improving the sample complexity of the algorithm of Meister and Nietert (2021) by first designing more query-efficient deterministic k-TQM algorithms and then simulating them using randomized algorithms. Our goal is two-fold: making the query budget and sample complexity simultaneously as small as possible. Naturally, the sample complexity of any simulation algorithm for TQM would depend on these two parameters. However, the smaller the query budget, the greater the sample complexity required for ε-accurate CDF estimation. The trivial deterministic n-TQM algorithm guarantees an ε-accurate CDF estimate for every $T \geq 1$, i.e, its

[1] The argument is presented in Lemma C.1

sample complexity is 1. In Appendix C, we show elementary deterministic algorithms, one with query budget $O(\sqrt{n}/\varepsilon)$ and sample complexity $O(1/\varepsilon)$, the other with query budget $O\left(\frac{\log n}{\varepsilon}\right)$ and sample complexity $O\left(\frac{\log n}{\varepsilon}\right)$. However, neither of these elementary algorithms attains the lowest possible query budget for deterministic algorithms. Using the Blackwell's Approachability Theorem, in Section 3, we show the existence of a deterministic algorithm with query budget $O(1/\varepsilon)$, accuracy ε, and finite sample complexity. Then, in Section 4, we show using the multiplicative weights method that this algorithm can be designed to have sample complexity $O\left(\frac{\log n}{\varepsilon}\right)$. In Section 5, we adapt this algorithm for the TQM using importance weighting and show that its sample complexity is bounded by $O\left(\frac{\log n}{\varepsilon^3}\right)$. In Section 6, we prove a lower bound on TQM that matches our upper bound up to a constant factor and in Section 7, we consider some generalizations and future work.

3 Using Approachability to solve TQM

3.1 Review of Blackwell Approachability

Blackwell approachability (Blackwell, 1956) generalizes the problem of playing a repeated two-player zero-sum game to games whose payoffs are vectors instead of scalars. In a Blackwell approachability game, at all times t, two players interact in this order: first, Player 1 selects an action $a_t \in A$; then, Player 2 selects an action $b_t \in B$; finally, Player 1 incurs the vector-valued payoff $h(a_t, b_t) \in \mathbb{R}^d$. The sets A, B of player actions are assumed to be compact convex subsets of finite-dimensional vector spaces, and h is assumed to be a biaffine function on $A \times B$. Player 1's objective is to guarantee that the average payoff converges to some desired closed convex target set $S \subseteq \mathbb{R}^d$. Formally, given target set $S \subseteq \mathbb{R}^d$, Player 1's goal is to pick actions $a_1, a_2, \ldots \in A$ such that no matter the actions $b_1, b_2, \ldots \in B$ played by Player 2,

$$\texttt{dist}\left(\frac{1}{T}\sum_{t=1}^{T} h(x_t, y_t), S\right) \to 0 \quad \text{as} \quad T \to \infty \tag{1}$$

The action a_t is allowed to depend on the realized payoff vectors $h_s(a_s, b_s)$ for $s = 1, 2, \ldots, t-1$. We say the set S is approachable if Player 1 has a strategy that attains the goal (1) no matter how Player 2 plays. Blackwell's Approachability Theorem asserts that a convex set $S \subset \mathbb{R}^d$ is approachable if and only if every closed halfspace containing S is approachable. This is a convenient criterion for approachability, because one can test whether a halfspace is approachable by computing the value of an associated zero-sum game, or equivalently by solving a linear program.

3.2 Approachability Reduction

A deterministic algorithm for the k-TQM chooses queries $q_{1,t} \le q_{2,t} \le \cdots \le q_{k,t}$ at time t, and receives feedback $y_{1,t} \le y_{2,t} \le \cdots \le y_{k,t}$, where $y_{i,t} = v_t(q_{i,t})$. For notational convenience, we will interpret $q_{0,t} = 0, q_{k+1,t} = n+1, y_{0,t} = 0, y_{k+1,t} = 1$.

[•] For index $i \in [n]$, define $\ell_t(i)$ and $u_t(i)$ by

[vg: i just added a newline here]

$$\ell_t(i) = \max\{y_{j,t} \mid 0 \le j \le k+1,\ q_{j,t} \le i\}, \qquad u_t(i) = \min\{y_{j,t} \mid 0 \le j \le k+1,\ q_{j,t} \ge i\}.$$

These are the tightest lower and upper bounds on $F_t(i)$ that can be deduced from the values the algorithm queried. Let $d_t(i) = u_t(i) - \ell_t(i)$. At time T, the best lower and upper bounds on $\frac{1}{T}\sum_{t=1}^{T} v_t(i)$ that can be

deduced from the values queried are $\frac{1}{T}\sum_{t=1}^{T}\ell_t(i)$ and $\frac{1}{T}\sum_{t=1}^{T}u_t(i)$. Hence, we know that $\frac{1}{T}\sum_{t=1}^{T}v_t(i)$ belongs to an interval of width $\frac{1}{T}\sum_{t=1}^{T}d_t(i)$. When this interval width is less than or equal to 2ε for every $i \in [n]$, it is safe to stop and output an estimate $G_T : [n] \to [0,1]$ defined by setting $G_T(i)$ to be the midpoint of the interval $\left[\frac{1}{T}\sum_{t=1}^{T}\ell_t(i), \frac{1}{T}\sum_{t=1}^{T}u_t(i)\right]$.

This suggests the following formulation of the deterministic query model as a game with vector payoff. In each round:

1. Adversary chooses monotone non-decreasing $v_t : [n] \to [0,1]$.
2. Algorithm simultaneously chooses $q_{1,t} \le \cdots \le q_{k,t}$. This corresponds to Player 1's action.
3. Feedback $y_{i,t} = v_t(q_{i,t})$ is revealed for $i = 1, 2, \ldots, k$. This corresponds to Player 2's action.
4. The n-dimensional loss vector is $\mathbf{d}_t = (d_t(1), d_t(2), \cdots, d_t(n))$. In our reduction, this corresponds to the vector-valued payoff at time t. [•]

[vg: I can also give the blackwell approachability correspondences in this list]

The questions we seek to understand are: for which values of k is there an algorithm that guarantees to approach the set $(-\infty, 2\varepsilon]^n$? If the set is approachable, how large must T be to ensure that the algorithm's L_∞ distance from that set is $O(\varepsilon)$?

Proposition 3.1. In the vector payoff game corresponding to query budget k, the set $(-\infty, 2\varepsilon]^n$ is approachable whenever $k + 1 \ge \frac{1}{2\varepsilon}$.

Proof. To show that $S = (-\infty, 2\varepsilon]^n$ is an approachable set, we need to show that every halfspace containing S is approachable. A halfspace containing S is a set H of the form

$$H = \left\{ \mathbf{x} \,\middle|\, \sum_{i=1}^{n} a_i x_i \le b \right\}$$

where $a_1, a_2, \ldots, a_n$ are non-negative, at least one of them is strictly positive, and $b \ge \sum_{i=1}^{n} a_i(2\varepsilon)$. Without loss of generality[2], b is equal to $2\varepsilon\sum_{i=1}^{n} a_i$. Also, without changing the halfspace H, we can normalize $a_1, \ldots, a_n, b$ so that $\sum_{i=1}^{n} a_i = 1$ and $b = 2\varepsilon$. Assume henceforth that we have adopted such a normalization.

For $j = 1, \ldots, k$ let

$$q_j = \min\left\{ q \,\middle|\, \sum_{i=1}^{q} a_i \ge \frac{j}{k+1} \right\}. \tag{2}$$

We aim to show that for any choice of v_t by the adversary, the loss vector $\mathbf{d}_t$ belongs to H when the algorithm chooses $q_1, \ldots, q_k$ as defined in Equation (2).[3] For notational convenience, let $q_0 = 0$, $q_{k+1} = n + 1$ and let $v_t(0) = 0$, $v_t(n+1) = 1$. Observe, by the definition of q_j, that

$$\sum_{i=q_j+1}^{q_{j+1}-1} a_i = \sum_{i=1}^{q_{j+1}-1} a_i - \sum_{i=1}^{q_j} a_i < \frac{j+1}{k+1} - \frac{j}{k+1} = \frac{1}{k+1}. \tag{3}$$

[2] Otherwise, H is a proper superset of another halfspace H' that also contains S, and to show H is approachable it suffices to show H' is approachable.

[3] For brevity, we are using the notation q_j while referring to the query $q_{j,t}$

Also observe that for $i = q_j$ we have $u_t(i) = \ell_t(i) = v_t(q_j)$, $d_t(i) = 0$, while for $q_j < i < q_{j+1}$ we have $u_t(i) = v_t(q_{j+1})$, $\ell_t(i) = v_t(q_j)$, $d_t(i) = v_t(q_{j+1}) - v_t(q_j)$. Hence,

$$\sum_{i=1}^{n} a_i d_t(i) = \sum_{j=0}^{k} \sum_{i=q_j+1}^{q_{j+1}-1} a_i (v_t(q_{j+1}) - v_t(q_j))$$
$$\leq \sum_{j=0}^{k} \frac{v_t(q_{j+1}) - v_t(q_j)}{k+1} = \frac{v_t(q_{k+1}) - v_t(q_0)}{k+1} = \frac{1}{k+1}.$$

The right side is less than or equal to $b = 2\varepsilon$ whenever $k + 1 \geq \frac{1}{2\varepsilon}$. This confirms that every halfspace containing S is approachable, hence S is approachable. ∎

4 Multiplicative Weights Algorithm for CDF estimation with parallel queries

In this section, we transform the proof of approachability (Proposition 3.1) into a deterministic algorithm with query budget $k = \lfloor 1/\varepsilon \rfloor$, accuracy ε, and sample complexity $\frac{9 \ln n}{\varepsilon}$. The key to designing the algorithm will be to select coefficients $a_{i,t}$ in each round t using the multiplicative weights algorithm, and then respond to these coefficients by choosing query points $q_{1,t}, \ldots, q_{k,t}$ using Equation (2) as in the proof of approachability.

ALGORITHM 1: Multiplicative weights algorithm for k-TQM

Initialize: $\eta = 1/3$;
$w_{i,0} = \frac{1}{n}$ for $i \in [n]$;
for $t = 1, 2, \ldots, T$ **do**
  $W = \sum_{i=1}^{n} w_{i,t-1}$;
  For $i \in [n]$ let $a_{i,t} = w_{i,t-1}/W$;
  For $j \in [k]$ let $q_{j,t} = \min \left\{ q \;\middle|\; \sum_{i=1}^{q} a_{i,t} \geq \frac{j}{k+1} \right\}$;
  Query points $q_{1,t}, \ldots, q_{k,t}$ and receive answers $y_{1,t}, \ldots, y_{k,t}$;
  Let $q_{0,t} = 0$ and $q_{k+1,t} = n$;
  for $i \in [n]$ **do**
    $\ell_t(i) = \max\{y_{j,t} \mid 0 \leq j \leq k+1,\ q_{j,t} \leq i\}$;
    $u_t(i) = \min\{y_{j,t} \mid 0 \leq j \leq k+1,\ q_{j,t} \geq i\}$;
    $d_t(i) = u_t(i) - \ell_t(i)$;
    $w_{i,t} = w_{i,t-1} \cdot (1+\eta)^{d_t(i)}$;
  end
end
Output: $G_T[i] = \frac{1}{2T} \sum_{t=1}^{T} (\ell_t(i) + u_t(i))$ for all $i \in [n]$.

Theorem 4.1. When $k + 1 \geq \frac{1}{\varepsilon}$, Algorithm 1 solves the k-TQM with accuracy ε and sample complexity $\frac{9 \ln n}{\varepsilon}$.

Proof. The weights $w_{i,t}$ and coefficients $a_{i,t}$ in Algorithm 1 evolve according to the update equations of the standard Hedge algorithm with parameter $\eta = 1/3$, and according to the analysis of that algorithm in Arora et al. (2012) [•], we have the inequality

[rdk: Citation needed!]

$$\sum_{t=1}^{T}\sum_{i=1}^{n} a_i d_t(i) \geq (1-\eta)\max_{i\in[n]}\{\sum_{t=1}^{T} d_t(i)\} - \frac{\ln n}{\eta}. \tag{4}$$

From the proof of Proposition 3.1 we know that for all t, $\sum_{i=1}^{n} a_i d_t(i) \leq \frac{1}{k+1}$. Substituting this bound into Inequality (4) we obtain

$$\frac{T}{k+1} \geq (1-\eta)\max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} - \frac{\ln n}{\eta} = \frac{2}{3}\max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} - 3\ln n \geq \frac{2}{3}\max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} - \frac{\varepsilon}{3}T, \tag{5}$$

where the second inequality follows from the fact that $T \geq \frac{9\ln n}{\varepsilon}$. Recalling that $\eta = \frac{1}{3}, k+1 \geq \frac{1}{\varepsilon}$, we find that

$$\begin{aligned}
\varepsilon T &\geq \frac{2}{3}\max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} - \frac{\varepsilon}{3}T \\
\frac{4}{3}\varepsilon T &\geq \frac{2}{3}\max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} \\
2\varepsilon &\geq \max_{i\in[n]}\left\{\frac{1}{T}\sum_{t=1}^{T} d_t(i)\right\}.
\end{aligned}$$

The right side of the last inequality is equal to the width of the interval $\left[\frac{1}{T}\sum_{t=1}^{T}\ell_t(i),\ \frac{1}{T}\sum_{t=1}^{T}u_t(i)\right]$. That interval is guaranteed to contain $\frac{1}{T}\sum_{t=1}^{T} v_t(i)$, and its midpoint is $G_T(i)$, so we are assured that $|G_T(i) - \frac{1}{T}\sum_{t=1}^{T} v_t(i)| \leq \frac{1}{2}(2\varepsilon) = \varepsilon$, as desired. ∎

5 Randomized algorithm using importance weighting

In Section 4, we presented a deterministic algorithm with accuracy ε, query budget $\lfloor\frac{1}{\varepsilon}\rfloor$, and sample complexity $\frac{9\ln n}{\varepsilon}$. Earlier we noted that no deterministic algorithm can obtain accuracy ε with a query budget less than $\frac{1}{2\varepsilon} - 1$. [•] In this section, we turn our attention to *randomized* algorithms with query budget 1.

[rdk: Off by a factor of 2?]

One natural approach would be to simulate the deterministic algorithm from the previous section; that is, run one step of Algorithm 1 to obtain a set of k query points, and over the course of the next $O\left(\frac{k\log n}{\varepsilon^2}\right)$ time-steps, randomly sample one of the k points to query to get an ε-accurate estimate of the CDF at each of the k query points. This approach can be carried out successfully, although we omit the details from this paper. However, the resulting sample complexity bound exceeds the optimal bound by a factor of $\Omega\left(\frac{\ln\ln n}{\varepsilon}\right)$. One of the main reasons for this is that the algorithm commits to sampling from a *fixed* set of k points for $O\left(\frac{k\log n}{\varepsilon^2}\right)$ time steps even though the algorithm's CDF estimate is changing and Algorithm 1 might suggest a different set of k query points.

To circumvent this issue of committing to a fixed set of query points, we use an approach from the bandit literature known as importance weighting. We can't query all the points $q_{1,t},\ldots,q_{k,t}$ to receive feedback $y_{1,t},\ldots,y_{k,t}$, so we instead choose one point uniformly at random $q_{m,t}$, to receive feedback $y_{m,t}$. We set the values $\hat{y}_{j,t}$ to $k\cdot y_{m,t}$ if $j = m$ and 0 otherwise. Then we proceed with the rest of the algorithm with values $\hat{y}_{1,t},\ldots,\hat{y}_{k,t}$ instead of $y_{1,t},\ldots,y_{k,t}$.

ALGORITHM 2: Randomized MW algorithm using importance weighting

Initialize: $k = \frac{2}{\varepsilon}, \eta = \varepsilon^2/16$;
$w_{i,0} = \frac{1}{n}$ for $i \in [n]$;
for $t = 1, 2, \ldots, T$ **do**
  $W = \sum_{i=1}^{n} w_{i,t-1}$;
  For $i \in [n]$ let $a_{i,t} = w_{i,t-1}/W$;
  For $j \in [k]$ let $q_{j,t} = \min\left\{q \;\middle|\; \sum_{i=1}^{q} a_{i,t} \geq \frac{j}{k+1}\right\}$;
  Let $q_{0,t} = 0$ and $q_{k+1,t} = n$;
  For $j \in [k]$ let $\hat{y}_{j,t} = 0$;
  Sample m uniformly from $[k]$;
  Query $q_{m,t}$ and receive $y_{m,t}$;
  Set $\hat{y}_{m,t} = k \cdot y_{m,t}$;
  for $i \in [n]$ **do**
    $\hat{\ell}_t(i) = \hat{y}_{r,t}$ where $r = \max\{j \mid 0 \leq j \leq k+1,\ q_{j,t} \leq i\}$;
    $\hat{u}_t(i) = \hat{y}_{r,t}$ where $r = \min\{j \mid 0 \leq j \leq k+1,\ q_{j,t} \geq i\}$;
    $\hat{d}_t(i) = \hat{u}_t(i) - \hat{\ell}_t(i)$;
    $w_{i,t} = w_{i,t-1} \cdot (1+\eta)^{\hat{d}_t(i)}$;
  end
end
Output: $\hat{G}_T[i] = \frac{1}{2T}\sum_{t=1}^{T}(\hat{\ell}_t(i) + \hat{u}_t(i))$ for all $i \in [n]$.

[rdk: Should the output be denoted by G_T rather than $\hat{G}_T$?]

Theorem 5.1. Algorithm 2 solves the TQM with accuracy ε and sample complexity $\frac{64 \log n}{\varepsilon^3}$.

The proof of the theorem is presented in Appendix B.

6 Lower Bound

In this section we sketch a proof that the sample complexity of Line 2 is information-theoretically optimal, up to a constant factor. The full proof appears in Appendix A.

Theorem 6.1. For any $n > 1, \varepsilon > 0$, every algorithm that solves the CDF estimation problem with accuracy ε, using one threshold query per sample, requires at least $T_0 = \Omega(\min\{n, \frac{1}{\varepsilon} \cdot \log(n)/\varepsilon^2)$ samples.

When $n \leq \frac{1}{\varepsilon}$ this is Theorem 6 of Meister and Nietert (2021), so for the remainder of this section we discuss the proof when $n > \frac{1}{\varepsilon}$. Assume for convenience[4] that the numbers $k = \frac{1}{6\varepsilon}$ and $m = n/k = 6\varepsilon n$ are positive integers. We will then define a family of probability distributions parameterized by $\theta \in [m]^k$. Their cumulative distribution functions, $\{F_\theta \mid \theta \in [m]^k\}$, are designed to have three properties.

1. For any function $\hat{F}$ there is at most one $\theta \in [m]^k$ such that $\|\hat{F} - F_\theta\|_\infty < \frac{3\varepsilon}{2}$. Hence, given a $\frac{3\varepsilon}{2}$-accurate estimate of F_θ we can deduce the value of θ. (Lemma A.2)

[4]These assumptions are without loss of generality. First increase ε by a factor of at most 6, to ensure that $\frac{1}{6\varepsilon}$ is an integer less than or equal to n, then decrease n by a factor of at most 2 to ensure that n is divisible by $\frac{1}{6\varepsilon}$. These changes to ε and n only affect the implicit constant in the big-Ω expression for the lower bound.

2. Informally, any sequence of $1/\varepsilon^2$ threshold queries reveals at most $O(1)$ bits of information about θ. In the proof, this statement is formalized information-theoretically in terms of the expected KL-divergence between the learner's prior and posterior distributions over θ. (Lemma A.13)

3. Starting from a uniform prior over $\theta \in [m]^k$, the posterior distribution after any sequence of threshold queries is a product distribution. In other words, writing θ as a k-tuple $(\theta_1, \ldots, \theta_k)$, the k coordinates of the tuple are mutually independent under the posterior distribution. (Lemma A.3)

Now suppose the adversary generates a sequence $x_1, \ldots, x_T$ by sampling $\theta \in [m]^k$ uniformly at random and then drawing T independent samples from the distribution with CDF F_θ. Using the Dvoretzky-Kiefer-Wolfowitz Inequality, we will argue that with probability at least $\frac{7}{8}$, the empirical CDF of the samples differs from F_θ by less than $\frac{\varepsilon}{2}$ in L_∞ norm. Hence, an ε-accurate estimate of the empirical distribution is a $\frac{3\varepsilon}{2}$-accurate estimate of F_θ, and consequently it uniquely determines the value of θ. This implies that an algorithm which succeeds, with probability at least $\frac{3}{4}$, in outputting an ε-accurate estimate of the empirical distribution of the samples, must also succeed with probability at least $\frac{5}{8}$ in learning the exact value of θ, a random variable of entropy $k\log(m)$. Since it takes $\Omega(1/\varepsilon^2)$ queries to learn a single bit of information about θ, it takes $\Omega(k\log(m)/\varepsilon^2)$ queries to learn the exact value of θ with constant probability. Recalling the definitions of k and m, we see that this bound is $\Omega(\log(\varepsilon n)/\varepsilon^3)$. Theorem 6.1 asserts the stronger lower bound $\Omega(\log(n)/\varepsilon^3)$, which is asymptotically greater when $1/\varepsilon = n^{1-o(1)}$. To strengthen the lower bound in this case, we use the third property of the construction — that the posterior distribution over θ is a product distribution — to prove a stronger lower bound on the expected KL divergence between the prior and posterior distributions at the time when the algorithm outputs its estimate.

7 Discussion and Open Problems

In addressing the online CDF estimation problem, we took a detour to a more general setting - the Threshold Query Model. Although we completely characterize the sample complexity of online CDF estimation using threshold queries, this only resolves the sample complexity question for the k-TQM for $k = 1$. One direction for future work is to characterize the optimal sample complexity for every value of the query budget k. Another direction is to extend the Threshold Query Model to other distance metrics besides the Kolmogorov-Smirov distance. It is important to note that the presented algorithms are fully adaptive. In some practical settings, however, there might issues of latency and delays. This raises a question of whether the query budget and sample complexity is higher for non-adaptive algorithms, and if so, by how much? Some special cases of this are addressed in Appendix C.

A natural extension to consider is the continuous-support setting, where the queries and samples can be any real value in the interval $[0, 1]$. Without any additional assumptions, CDF estimation in this setting becomes intractable. This is because there are infinitely many values and the algorithm cannot cover all of them with a finite number of queries. However, Meister and Nietert (2021) point out that if we specify some resolution r of interest, then by setting $n = O(1/r)$, this reduces to the discrete-support case. We wonder what assumptions on the adversary's probability density function would make the CDF estimation problem tractible and if our techniques would be applicable. Another direction of interest is to consider higher-dimensional generalizations that estimate multivariate distributions - the samples are now vector-valued, and the algorithm queries the data using linear threshold functions (i.e., halfspaces).

In Section 1, we pointed out some related works in relevant application areas like online dynamic pricing and auctions. We wonder if our techniques can be extended to these settings as well.

References

Acharya, J., Canonne, C., Liu, Y., Sun, Z., and Tyagi, H. (2021). Distributed estimation with multiple samples per user: Sharp rates and phase transition. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*, volume 34, pages 18920–18931. Curran Associates, Inc. 4

Acharya, J., Canonne, C. L., Sun, Z., and Tyagi, H. (2020). Unified lower bounds for interactive high-dimensional estimation under information constraints. *CoRR*, abs/2010.06562. 4

Ali, S. and Ronaldson, S. (2012). Ordinal preference elicitation methods in health economics and health services research: using discrete choice experiments and ranking methods. *British medical bulletin*, 103(1):21–44. 1

Arora, S., Hazan, E., and Kale, S. (2012). The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(6):121–164. 8

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. (2002). The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77. 23

Barlow, R. E., Bartholomew, D. J., Bremner, J. M., and Brunk, H. D. (1972). *Statistical inference under order restrictions: The theory and application of isotonic regression*. Wiley. 3

Barnes, L. P., Han, Y., and Ozgur, A. (2020). Lower bounds for learning distributions under communication constraints via fisher information. *Journal of Machine Learning Research*, 21(236):1–30. 3, 4

Ben-Or, M. and Hassidim, A. (2008). The bayesian learner is optimal for noisy binary search (and pretty good for quantum as well). In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 221–230. IEEE. 3

Blackwell, D. (1956). An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6(1):1 – 8. 6

Blum, A., Mansour, Y., and Morgenstern, J. (2015). Learning valuation distributions from partial observations. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 798–804. AAAI Press. 4

Burnashev, M. V. and Zigangirov, K. S. (1974). An interval estimation problem for controlled observations. *Problems of Information Transmission*, 10:223–231. (published in Russian). 3

Chapelle, O., Joachims, T., Radlinski, F., and Yue, Y. (2012). Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems (TOIS)*, 30(1):1–41. 1

Cormode, G., Johnson, T., Korn, F., Muthukrishnan, S., Spatscheck, O., and Srivastava, D. (2004). Holistic udafs at streaming speeds. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 35–46. 1

Dvoretzky, A., Kiefer, J., and Wolfowitz, J. (1956). Asymptotic Minimax Character of the Sample Distribution Function and of the Classical Multinomial Estimator. *The Annals of Mathematical Statistics*, 27(3):642 – 669. 22

Feige, U., Raghavan, P., Peleg, D., and Upfal, E. (1994). Computing with noisy information. *SIAM Journal on Computing*, 23(5):1001–1018. 3

Freedman, D. A. (1975). On Tail Probabilities for Martingales. *The Annals of Probability*, 3(1):100 – 118. 24

Greenwald, M. and Khanna, S. (2001). Space-efficient online computation of quantile summaries. *ACM SIGMOD Record*, 30(2):58–66. 1

Han, Y., Özgür, A., and Weissman, T. (2018). Geometric lower bounds for distributed parameter estimation under communication constraints. In *Proceedings of the 31st Conference on Learning Theory, COLT 2018*, volume 75 of *Proceedings of Machine Learning Research*, pages 3163–3188. PMLR. The arXiv (v3) version from 2020 corrects some issues and includes more results. 3

Karnin, Z., Lang, K., and Liberty, E. (2016). Optimal quantile approximation in streams. In *2016 ieee 57th annual symposium on foundations of computer science (focs)*, pages 71–78. IEEE. 1

Karp, R. M. and Kleinberg, R. (2007). Noisy binary search and its applications. In *Proc. 18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 881–890. 2, 3

Kleinberg, R. and Leighton, T. (2003). The value of knowing a demand curve: bounds on regret for online posted-price auctions. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 594–605. 4

Larichev, O., Olson, D., Moshkovich, H., and Mechitov, A. (1995). Numerical vs cardinal measurements in multiattribute decision making: How exact is enough? *Organizational Behavior and Human Decision Processes*, 64(1):9–21. 1

Leme, R. P., Sivan, B., Teng, Y., and Worah, P. (2021a). Learning to price against a moving target. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6223–6232. PMLR. 4

Leme, R. P., Sivan, B., Teng, Y., and Worah, P. (2021b). Pricing query complexity of revenue maximization. *CoRR*, abs/2111.03158. 4

Manku, G. S., Rajagopalan, S., and Lindsay, B. G. (1998). Approximate medians and other quantiles in one pass and with limited memory. *ACM SIGMOD Record*, 27(2):426–435. 1

Meister, M. and Nietert, S. (2021). Learning with comparison feedback: Online estimation of sample statistics. In *Algorithmic Learning Theory*, pages 983–1001. PMLR. 2, 3, 4, 5, 10, 11, 15, 21

Moshkovich, H. M., Mechitov, A. I., and Olson, D. L. (2002). Ordinal judgments in multiattribute decision analysis. *European Journal of Operational Research*, 137(3):625–641. 1

Munro, J. I. and Paterson, M. S. (1980). Selection and sorting with limited storage. *Theoretical computer science*, 12(3):315–323. 1

Perchet, V. (2015). Exponential weight approachability, applications to calibration and regret minimization. *Dynamic Games and Applications*, 5:136–153. 2

Shrivastava, N., Buragohain, C., Agrawal, D., and Suri, S. (2004). Medians and beyond: new aggregation techniques for sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 239–249. 1

Weed, J., Perchet, V., and Rigollet, P. (2016). Online learning in repeated auctions. In Feldman, V., Rakhlin, A., and Shamir, O., editors, *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1562–1583, Columbia University, New York, New York, USA. PMLR. 1

A Proof of Lower Bound

Throughout this section we assume $n > \frac{1}{\varepsilon}$, since the case $n \leq \frac{1}{\varepsilon}$ of Theorem 6.1 was already proven by Meister and Nietert (2021). Additionally, as explained in Section 6, we assume without loss of generality that $n = km$, where $k = \frac{1}{6\varepsilon}$.

We begin in Appendix A.1 by describing the construction of a family of distributions over $[n]$ parameterized by $\theta \in [m]^k$. Then, in Appendix A.2 we review some basic facts about KL divergence, the main information theoretic tool in the proof. Finally, Appendix A.3 completes the proof.

A.1 Family of distributions

Since $n = km$, each element of $[n]$ can be uniquely expressed in the form $(a-1)m + b$ where $a \in [k]$ and $b \in [m]$. For $\theta = (\theta_1, \theta_2, \ldots, \theta_k) \in [m]^k$ let I_θ denote the set

$$I_\theta = \{(a-1)m + \theta_a \mid a \in [k]\}$$

and let D_θ denote the probability distribution on $[n]$ representing the output of the following sampling rule.

1. With probability $\frac{1}{2}$ output a uniformly random element of I_θ.
2. With probability $\frac{1}{4}$ output 1.
3. With probability $\frac{1}{4}$ output n.

The cumulative distribution function of D_θ is the function F_θ whose value at $i = (a-1)m + b$, when $a \in [k]$, $b \in [m]$, is given by the formula

$$F_\theta((a-1)m + b) = \begin{cases} \frac{1}{4} + 3(a-1)\varepsilon & \text{if } b < \theta_a \\ \frac{1}{4} + 3a\varepsilon & \text{if } b \geq \theta_a,\ (a-1)m + b < n \\ 1 & \text{if } (a-1)m + b = n. \end{cases} \tag{6}$$

Lemma A.1. If θ, θ' are any two distinct elements of $[m]^k$ then $\|F_\theta - F_{\theta'}\|_\infty = 3\varepsilon$.

Proof. From the definition of F_θ and $F_{\theta'}$ it is apparent that $F_\theta(n) = F_{\theta'}(n) = 1$ and that for $i = (a-1)m + b < n$, $F_\theta(i)$ and $F_{\theta'}(i)$ both belong to the set $\{\frac{1}{4} + 3(a-1)\varepsilon,\ \frac{1}{4} + 3a\varepsilon\}$, so $|F_\theta(i) - F_{\theta'}(i)|$ cannot exceed 3ε. Thus $\|F_\theta - F_{\theta'}\|_\infty \leq 3\varepsilon$.

Since we are assuming $\theta \neq \theta'$, there exists some $a \in [k]$ such that $\theta_a \neq \theta'_a$. Assume without loss of generality that $\theta_a < \theta'_a$. Then, $i = (a-1)m + \theta_a$, we have $F_\theta(i) - F_{\theta'}(i) = 3\varepsilon$. Thus, $\|F_\theta - F_{\theta'}\|_\infty \geq 3\varepsilon$. ∎

Lemma A.2. For any function $\hat{F} : [n] \to [0, 1]$ there is at most one $\theta \in [m]^k$ satisfying $\|\hat{F} - F_\theta\|_\infty < \frac{3\varepsilon}{2}$.

Proof. The lemma follows immediately from Lemma A.1 and fact that the L_∞ norm satisfies the triangle inequality. ∎

In the proof to follow, we will be considering executing a fixed but arbitrary CDF estimation algorithm on an input sequence $x_1, x_2, \ldots, x_T$ generated as follows: first sample $\theta \in [m]^k$ uniformly at random, then let

$x_1, x_2, \ldots, x_T$ be T independent samples from D_θ. Let $(q_1, y_1), (q_2, y_2), \ldots, (q_T, y_T)$ be the random $[n]\times\{0,1\}$-valued sequence representing the algorithm's queries and the responses to those queries. For $0 \le t \le T$ let $p_t(\theta)$ denote the posterior distribution of θ given $(q_1, y_1), \ldots, (q_t, y_t)$. (When $t = 0$ this is simply the prior distribution of θ, i.e. the uniform distribution on $[m]^k$.)

Lemma A.3. For $0 \le t \le T$, for all sequences of queries and responses $(q_1, y_1), \ldots, (q_t, y_t)$, the posterior distribution p_t is a product distribution. In other words, if $\theta = (\theta_1, \ldots, \theta_k) \in [m]^k$ is distributed according to p_t then the random variables $\theta_1, \ldots, \theta_k$ are mutually independent.

Proof. The proof is by induction in t. In the base case, p_0 is the uniform distribution on $[m]^k$, which is a product distribution. For the induction step, write $q_t \in [n]$ as $q_t = (a_t - 1)m + b_t$ where $a_t \in [k], b_t \in [m]$. The distribution of y_t given q_t depends only on the parameter θ_{a_t}. Therefore, if p_{t-1} is a product distribution, a Bayesian update conditioning on (q_t, y_t) will alter the marginal distribution of θ_{a_t} while leaving it independent of θ_j for all $j \ne a_t$. ∎

A.2 Review of KL divergence

For two probability distributions p, q on a finite set Ω, their Kullback-Leibler divergence, henceforth called KL divergence, is defined as

$$D_{KL}(p \,\|\, q) = \sum_{\omega\in\Omega} p(\omega) \ln\left(\frac{p(\omega)}{q(\omega)}\right). \tag{7}$$

When $p(\omega) = 0$ the summand on the right side of (7) is interpreted as zero.

In this section and the following one, if p is a distribution over pairs (X, Y) then $p(X)$, $p(Y)$ denote the marginal distribution of X and Y, respectively, and $p(X|Y)$, $p(Y|X)$ denote the conditional distribution of X given Y and the conditional distribution of Y given X, respectively.

Lemma A.4. If p, q are any two probability distributions on the same set, then $D_{KL}(p \,\|\, q) \ge 0$.

Proof. The function $\phi(x) = -\ln(x)$ is convex, so by Jensen's inequality,

$$\begin{aligned}
\sum_{\omega\in\Omega} p(\omega) \ln\left(\frac{p(\omega)}{q(\omega)}\right) &= \sum_{\omega\in\Omega} p(\omega)\phi\left(\frac{q(\omega)}{p(\omega)}\right) \\
&\ge \phi\left(\sum_{\omega\in\Omega} p(\omega)\cdot\frac{q(\omega)}{p(\omega)}\right) \\
&= \phi\left(\sum_{\omega\in\Omega} q(\omega)\right) = \phi(1) = 0.
\end{aligned}$$

∎

A.2.1 The chain rule and its corollaries

Lemma A.5 (Chain rule for KL divergence)**.** If p, q are probability distributions on pairs (X, Y) then

$$D_{KL}(p \,\|\, q) = D_{KL}(p(Y) \,\|\, q(Y)) + \mathbb{E}_{Y\sim p} D_{KL}(p(X|Y) \,\|\, q(X|Y)). \tag{8}$$

Proof. For any y, Bayes' Law implies

$$\ln p(X = x|Y = y) = \ln p(X = x,\ Y = y) - \ln p(Y = y)$$
$$\ln q(X = x|Y = y) = \ln q(X = x,\ Y = y) - \ln q(Y = y)$$
$$\ln\left(\frac{p(X = x|Y = y)}{q(X = x|Y = y)}\right) = \ln\left(\frac{p(X = x,\ Y = y)}{q(X = x,\ Y = y)}\right) - \ln\left(\frac{p(Y = y)}{q(Y = y)})\right)$$

hence

$$\begin{aligned}
\mathbb{E}_{Y\sim p} D_{KL}\left(p(X|Y)\,\|\,q(X|Y)\right) &= \sum_y p(Y = y) \sum_x p(X = x|Y = y) \ln\left(\frac{p(X = x|Y = y)}{q(X = x|Y = y)}\right) \\
&= \sum_{x,y} p(X = x, Y = y) \ln\left(\frac{p(X = x|Y = y)}{q(X = x|Y = y)}\right) \\
&= \sum_{x,y} p(X = x, Y = y) \ln\left(\frac{p(X = x,\ Y = y)}{q(X = x,\ Y = y)}\right) - \sum_{x,y} p(X = x, Y = y) \ln\left(\frac{p(Y = y)}{q(Y = y)})\right) \\
&= \sum_{x,y} p(X = x, Y = y) \ln\left(\frac{p(X = x,\ Y = y)}{q(X = x,\ Y = y)}\right) - \sum_{y} p(Y = y) \ln\left(\frac{p(X = x,\ Y = y)}{q(X = x,\ Y = y)}\right) \\
&= D_{KL}\left(p\,\|\,q\right) - D_{KL}\left(p(Y)\,\|\,q(Y)\right).
\end{aligned}$$

■

The chain rule has several corollaries which will be of use to us.

Lemma A.6. If p, q are probability distributions on t-tuples $X_1, \ldots, X_t$, then

$$D_{KL}\left(p\,\|\,q\right) = \sum_{s=1}^{t} \mathbb{E}_{X_1,\ldots,X_{s-1}\sim p} D_{KL}\left(p(X_s|X_1, \ldots, X_{s-1})\,\|\,q(X_s|X_1, \ldots, X_{s-1})\right). \tag{9}$$

Proof. The lemma follows by induction on t, which the base case $t = 1$ being trivial and the induction step being a direct application of Lemma A.5. ■

Lemma A.7. If p, q are probability distributions on t-tuples $X_1, \ldots, X_t$, and both p and q are product distributions, then

$$D_{KL}\left(p\,\|\,q\right) = \sum_{s=1}^{t} D_{KL}\left(p(X_s)\,\|\,q(X_s)\right). \tag{10}$$

Proof. Since p is a product distribution, $p(X_s|X_1, \ldots, X_{s-1}) = p(X_s)$, and similarly for q. The lemma now follows directly from Lemma A.6. ■

Lemma A.8. If p, q are probability distributions on pairs (X, Y) that have the same marginals — i.e., $p(X) = q(X)$ and $p(Y) = q(Y)$ — then

$$\mathbb{E}_{X\sim p} D_{KL}\left(p(Y|X)\,\|\,q(Y|X)\right) = D_{KL}\left(p\,\|\,q\right) = \mathbb{E}_{Y\sim p} D_{KL}\left(p(X|Y)\,\|\,q(X|Y)\right). \tag{11}$$

Proof. The chain rule for KL divergence yields the equations

$$\begin{aligned} D_{KL}(p \,\|\, q) &= D_{KL}(p(X) \,\|\, q(X)) + \mathbb{E}_{X\sim p} D_{KL}(p(Y|X) \,\|\, q(Y|X)) \\ &= D_{KL}(p(Y) \,\|\, q(Y)) + \mathbb{E}_{Y\sim p} D_{KL}(p(X|Y) \,\|\, q(X|Y)). \end{aligned}$$

The lemma now follows from the fact that the KL divergence of two identical distributions is zero. ∎

Lemma A.9. Suppose Ω, Σ are finite sets and $f : \Omega \to \Sigma$ is a function. For two probability distributions p, q on Ω, let p^f and q^f denote the distributions of $f(\omega)$ when ω is sampled from p or from q, respectively. Then

$$D_{KL}(p \,\|\, q) \geq D_{KL}\left(p^f \,\|\, q^f\right). \tag{12}$$

Proof. Let $\Gamma \subset \Omega \times \Sigma$ denote the graph of f, i.e. $\Gamma = \{(\omega, f(\omega)) \mid \omega \in \Omega\}$. Let $\tilde{p}$, $\tilde{q}$ denote the distributions of $(\omega, f(\omega))$ when ω is sampled from p or from q, respectively. The function $\omega \mapsto (\omega, f(\omega))$ is a probability-preserving bijection between the probability spaces (Ω, p) and $(\Gamma, \tilde{p})$ and also between the probability spaces (Ω, q) and $(\Gamma, \tilde{q})$. Consequently, $D_{KL}(p \,\|\, q) = D_{KL}(\tilde{p} \,\|\, \tilde{q})$. Now, using the chain rule and the non-negativity of KL divergence (Lemmas A.4 and A.5),

$$D_{KL}(\tilde{p} \,\|\, \tilde{q}) = D_{KL}\left(p^f \,\|\, q^f\right) + \mathbb{E}_{f(\omega)\sim p^f} D_{KL}(p(\omega|f(\omega)) \,\|\, q(\omega|f(\omega))) \geq D_{KL}\left(p^f \,\|\, q^f\right).$$

∎

Finally, we shall make use of two lemmas bounding the KL divergence of Bernoulli distributions. The first is an upper bound on $D_{KL}(p \,\|\, q)$ when neither $q(0)$ nor $q(1)$ is close to zero. The second is a lower bound on $D_{KL}(p \,\|\, q)$ when $q(0)$ is close to zero and $p(0)$ is far from zero.

Lemma A.10. If p, q are two distributions on $\{0, 1\}$ such that $q(0), q(1) > 0$, then

$$D_{KL}(p \,\|\, q) \leq (p(0) - q(0))^2 \left(\frac{1}{q(0)} + \frac{1}{q(1)}\right). \tag{13}$$

Proof. Using the inequality $\ln(1 + x) \leq x$ and the fact that $p(0) - q(0) = -(p(1) - q(1))$, we find that

$$\begin{aligned} D_{KL}(p \,\|\, q) &= p(0) \ln\left(\frac{p(0)}{q(0)}\right) + p(1) \ln\left(\frac{p(1)}{q(1)}\right) \\ &= p(0) \ln\left(1 + \frac{p(0) - q(0)}{q(0)}\right) + p(1) \ln\left(1 + \frac{p(1) - q(1)}{q(1)}\right) \\ &\leq p(0) \cdot \frac{p(0) - q(0)}{q(0)} + p(1) \cdot \frac{p(1) - q(1)}{q(1)} \\ &= p(0) \cdot \frac{p(0) - q(0)}{q(0)} - p(1) \cdot \frac{p(0) - q(0)}{q(1)} \\ &= (p(0) - q(0)) \cdot \left(\frac{p(0)}{q(0)} - \frac{p(1)}{q(1)}\right) \\ &= (p(0) - q(0)) \cdot \left(1 + \frac{p(0) - q(0)}{q(0)} - 1 - \frac{p(1) - q(1)}{q(1)}\right) \\ &= (p(0) - q(0)) \cdot \left(\frac{p(0) - q(0)}{q(0)} + \frac{p(0) - q(0)}{q(1)}\right) \\ &= (p(0) - q(0))^2 \cdot \left(\frac{1}{q(0)} + \frac{1}{q(1)}\right). \end{aligned}$$

∎

Lemma A.11. If p, q are distributions on $\{0, 1\}$ such that $q(1) \leq \frac{1}{2} \leq p(1)$ then

$$D_{KL}(p \,\|\, q) \geq \frac{1}{2} \ln\left(\frac{1}{4q(1)}\right). \tag{14}$$

Proof. Let $x = p(0)$. We begin by calculating the partial derivative of $D_{KL}(p \,\|\, q)$ with respect to x.

$$\begin{aligned}
\frac{\partial}{\partial x} D_{KL}(p \,\|\, q) &= \frac{\partial}{\partial x}\left[x \ln(x) - x \ln(q(0)) + (1-x)\ln(1-x) - (1-x)\ln(q(1))\right] \\
&= ln(x) + 1 - \ln(q(0)) - \ln(1-x) - 1 + \ln(q(1)) \\
&= \ln\left(\frac{xq(1)}{(1-x)q(0)}\right) \\
&= \ln\left(\frac{x - xq(0)}{q(0) - xq(0)}\right).
\end{aligned}$$

The partial derivative is negative when $x < q(0)$ and positive when $x > q(0)$. Since we are assuming $p(0) \leq \frac{1}{2} \leq q(0)$, as $p(0)$ varies over the range $[0, \frac{1}{2}]$ the minimum of $D_{KL}(p \,\|\, q)$ occurs at $p(0) = \frac{1}{2}$, when

$$D_{KL}(p \,\|\, q) = \frac{1}{2}\ln\left(\frac{1}{2q(0)}\right) + \frac{1}{2}\ln\left(\frac{1}{2q(1)}\right) = \frac{1}{2}\ln\left(\frac{1}{4q(0)q(1)}\right) \geq \frac{1}{2}\ln\left(\frac{1}{4q(1)}\right).$$

■

Lemma A.12. Suppose p and q are product distributions on k-tuples $X = (X_1, \ldots, X_k)$ and that there exists a k-tuple $x^* = (x_1^*, x_2^*, \ldots, x_k^*)$ and a number $m \geq 2$ such that $p(X = x^*) \geq \frac{1}{2}$ while $q(X_i = x_i^*) \leq \frac{1}{m}$ for all i. Then

$$D_{KL}(p \,\|\, q) \geq \frac{k}{2}\ln\left(\frac{m}{4}\right) \quad \text{and} \quad D_{KL}(q \,\|\, p) \geq \frac{k}{2}\ln\left(\frac{k}{3}\right). \tag{15}$$

Proof. From Lemma A.7 we know that $D_{KL}(p \,\|\, q) = \sum_{i=1}^k D_{KL}(p(X_i) \,\|\, q(X_i))$ and $D_{KL}(q \,\|\, p) = \sum_{i=1}^k D_{KL}(q(X_i) \,\|\, p(X_i))$. For convenience let p_i denote the distribution $p(X_i)$ and let q_i denote the distribution $q(X_i)$. If we define $f(x_i)$ to be 1 if $x_i = x_i^*$ and 0 if $x_i \neq x_i^*$, then $p_i^f(1) = p(X_i = x_i^*) \geq p(X = x^*) \geq \frac{1}{2}$ and $q_i^f(1) = q(X_i = x_i^*) \leq \frac{1}{m} \leq \frac{1}{2}$. By Lemmas A.9 and A.11,

$$D_{KL}(p_i \,\|\, q_i) \geq D_{KL}\left(p_i^f \,\|\, q_i^f\right) \geq \frac{1}{2}\ln\left(\frac{1}{4q_i^f(1)}\right) \geq \frac{1}{2}\ln\left(\frac{m}{4}\right).$$

Summing over i we obtain the bound $D_{KL}(p \,\|\, q) \geq \frac{k}{2}\ln\left(\frac{m}{4}\right)$.

To prove the lower bound on $D_{KL}(q \,\|\, p)$, first define $z_i = p(X_i \neq x_i^*)$. Since $p(X = x^*) \geq \frac{1}{2}$ we know that $\prod_{i=1}^k (1 - z_i) \geq \frac{1}{2}$. Using the AM-GM inequality, this implies

$$\begin{aligned}
\frac{1}{k}\sum_{i=1}^k (1 - z_i) \geq (1/2)^{1/k} = e^{-\ln(2)/k} > 1 - \frac{\ln 2}{k} \\
\frac{1}{k}\sum_{i=1}^k z_i < \frac{\ln 2}{k}.
\end{aligned}$$

Another application of the AM-GM inequality leads to

$$\prod_{i=1}^{k} z_i < \left(\frac{\ln 2}{k}\right)^k$$
$$\sum_{i=1}^{k} \ln(z_i) < k \ln\left(\frac{\ln(2)}{k}\right) < k \ln\left(\frac{3}{4k}\right)$$
$$\sum_{i=1}^{k} \ln(4z_i) < k \ln\left(\frac{3}{k}\right).$$

Now, we proceed similarly to the previous paragraph. Let $g(X_i) = 1$ if $X_i \neq x_i^*$ and $g(X_i) = 0$ if $X_i = x_i^*$. We have $p_i^g(1) = p_i(X_i \neq x_i^*) = z_i$, which is less than or equal to $\frac{1}{2}$ since the inequality $\prod_{i=1}^{k}(1 - z_i) \geq \frac{1}{2}$ implies $1 - z_i \geq \frac{1}{2}$ for each i. Meanwhile, $q_i^g(1) = q_i(X_i \neq x_i^*) = 1 - q_i(X_i = x_i^*) \geq 1 - \frac{1}{m} \geq \frac{1}{2}$. By Lemmas A.9 and A.11,

$$D_{KL}\left(q_i \,\|\, p_i\right) \geq D_{KL}\left(q_i^f \,\|\, p_i^f\right) \geq \frac{1}{2} \ln\left(\frac{1}{4p_i^f(1)}\right) = -\frac{1}{2}\ln\left(4z_i\right).$$

Summing over i, we obtain

$$D_{KL}\left(q \,\|\, p\right) \geq -\frac{1}{2}\sum_{i=1}^{k} \ln(4z_i) > \frac{k}{2}\ln\left(\frac{k}{3}\right).$$

■

A.3 Completing the proof

In this section we complete the proof of Theorem 6.1. To this end, suppose that $n > 1/\varepsilon$ and that we are given an algorithm that uses T threshold queries to produce a CDF estimate that is ε-accurate with probability at least $\frac{3}{4}$. We aim to prove that $T \geq \Omega(k \log(n)/\varepsilon^2) = \Omega(\log(n)/\varepsilon^3)$.

Let $H = ((q_1, y_1), \ldots, (q_T, y_T))$ denote the random history of queries and responses obtained when running the algorithm on a (potentially random) sequence $x_1, \ldots, x_T$. We will consider two joint distributions $p, \mathring{p}$ over pairs (θ, H) consisting of a parameter vector θ and history H. Distribution p is the distribution obtained by sampling parameter vector $\theta \in [m]^k$ uniformly at random, then drawing T independent samples $x_1, \ldots, x_T$ from D_θ, and finally running the algorithm on input sequence $x_1, \ldots, x_T$ to generate a history, H. Distribution $\mathring{p}$ is the product distribution $p(\theta) \times p(H)$. In other words, a random sample from $\mathring{p}$ is defined by sampling a random history H from the marginal distribution $p(H)$, and independently drawing a uniformly random parameter vector $\theta \in [m]^k$ that bears no relation to H.

Since p and $\mathring{p}$ have identical marginals, Lemma A.8 tells us that

$$\mathbb{E}_\theta D_{KL}\left(p(H|\theta) \,\|\, \mathring{p}(H|\theta)\right) = D_{KL}\left(p \,\|\, \mathring{p}\right) = \mathbb{E}_H D_{KL}\left(p(\theta|H) \,\|\, \mathring{p}(\theta|H)\right) \tag{16}$$
$$\mathbb{E}_\theta D_{KL}\left(\mathring{p}(H|\theta) \,\|\, p(H|\theta)\right) = D_{KL}\left(p \,\|\, \mathring{p}\right) = \mathbb{E}_H D_{KL}\left(\mathring{p}(\theta|H) \,\|\, p(\theta|H)\right) \tag{17}$$

The next lemma furnishes an upper bound on the quantities appearing on the left sides of (16) and (17).

Lemma A.13. For any $\theta \in [m]^k$,

$$D_{KL}\left(p(H|\theta) \,\|\, \mathring{p}(H|\theta)\right) \leq 48\varepsilon^2 T \quad \text{and} \quad D_{KL}\left(\mathring{p}(H|\theta) \,\|\, p(H|\theta)\right) \leq 48\varepsilon^2 T \tag{18}$$

Proof. We use the chain rule for KL divergence, Lemma A.5. For $s = 1, 2, \ldots, T$ let H_s denote the random variable $((q_1, y_1), \ldots, (q_s, y_s))$ consisting of the first s queries and responses.

$$D_{KL}\left(p(H|\theta) \,\|\, \mathring{p}(H|\theta)\right) = \sum_{s=1}^{T} D_{KL}\left(p((q_s, y_s)|H_{s-1}, \theta) \,\|\, \mathring{p}((q_s, y_s)|H_{s-1}, \theta)\right) \tag{19}$$

A second application of the chain rule for KL divergence allows us to break down each term of the sum even further.

$$\begin{aligned} D_{KL}\left(p((q_s, y_s)|H_{s-1}, \theta) \,\|\, \mathring{p}((q_s, y_s)|H_{s-1}, \theta)\right) &= D_{KL}\left(p(q_s|H_{s-1}, \theta) \,\|\, \mathring{p}(q_s|H_{s-1}, \theta)\right) \\ &\quad + D_{KL}\left(p(y_s|q_s, H_{s-1}, \theta) \,\|\, \mathring{p}(y_s|q_s, H_{s-1}, \theta)\right) \end{aligned} \tag{20}$$

Since the algorithm selects q_s based on H_{s-1} only, the conditional distributions $p(q_s|H_{s-1}, \theta)$ and $\mathring{p}(q_s|H_{s-1}, \theta)$ are identical, hence the first term on the right side of (20) is zero:

$$D_{KL}\left(p(q_s|H_{s-1}, \theta) \,\|\, \mathring{p}(q_s|H_{s-1}, \theta)\right) = 0 \tag{21}$$

As for the second term, to simplify notation we will denote $p(y_s|q_s, H_{s-1}, \theta)$ and $\mathring{p}(y_s|q_s, H_{s-1}, \theta)$ by p_s and $\mathring{p}_s$, respectively. Then $p_s(0) = 1 - F_\theta(q_s)$, whereas $\mathring{p}_s(0) = \mathbb{E}[1 - F_{\theta'}(q_s)]$ with the expectation on the right side being computed by sampling θ' from the conditional distribution $p(\theta|H_{s-1})$. Lemma A.1 tells us that $|F_\theta(q_s) - F_{\theta'}(q_s)| \le 3\varepsilon$ for every θ', so

$$|p_s(0) - \mathring{p}_s(0)| \le 3\varepsilon. \tag{22}$$

If $q_s = n$ then y_s is deterministically equal to 1 under both distributions, p and $\mathring{p}$, so $D_{KL}\left(p_s \,\|\, \mathring{p}_s\right) = 0$ when $q_s = n$. Otherwise, $\mathring{p}_s(0)$ belongs to the interval $\left[\frac{1}{4}, \frac{3}{4}\right]$ and $\mathring{p}_s(1) = 1 - \mathring{p}_s(0)$, so

$$\frac{1}{\mathring{p}_s(0)} + \frac{1}{\mathring{p}_s(1)} \le \frac{1}{1/4} + \frac{1}{3/4} = \frac{16}{3}. \tag{23}$$

Now, applying Lemma A.10,

$$D_{KL}\left(p_s \,\|\, \mathring{p}_s\right) \le (p_s(0) - \mathring{p}_s(0))^2 \cdot \frac{16}{3} \le \frac{16}{3}(3\varepsilon)^2 = 48\varepsilon^2. \tag{24}$$

The inequality $D_{KL}\left(p(H|\theta) \,\|\, \mathring{p}(H|\theta)\right) \le 48\varepsilon^2 T$ follows by combining lines (19),(20),(21),(24). The second inequality in the statement of the lemma follows by an identical argument with the roles of p and $\mathring{p}$ reversed. ∎

The proof of Theorem 6.1 concludes as follows.

Proof of Theorem 6.1. Theorem 6 of Meister and Nietert (2021) already proves there is a universal constant $c > 0$ such that $T \ge cn\log(n)/\varepsilon^2$ when $\varepsilon \le 1/(n+1)$. This has two consequences for our proof. First, to complete the proof we only need to consider the case that $\varepsilon > 1/(n+1)$, in which case the theorem asserts a lower bound of the form $T = \Omega(\log(n)/\varepsilon^3)$. Second, if we define $n' = \left\lfloor \frac{1}{\varepsilon} \right\rfloor - 1$ then $\varepsilon \le 1/(n'+1)$, so Theorem 6 of Meister and Nietert (2021) proves that the CDF estimation problem for distributions on $[n']$ requires at least

$$cn'\log(n')/\varepsilon^2 \ge c'\ln(1/\varepsilon)/\varepsilon^3 \tag{25}$$

samples. (Here, $c' > 0$ is another univeral constant.) Since CDF estimation on $[n]$ generalizes CDF estimation on $[n']$, the sample complexity of CDF estimation on $[n]$ is also bounded below by the right side of (25):

$$T \ge c'\ln(1/\varepsilon)/\varepsilon^3. \tag{26}$$

Recall from Section 1 that CDF estimation generalizes binary search, so $T \geq \lfloor \log(n) \rfloor \geq \frac{1}{2}\log(n)$. If $\varepsilon \geq \min\{c', 1/256\}$ then $\frac{1}{2}\log(n) \geq \frac{1}{2} \cdot (\min\{c', 1/256\})^{-3} \cdot \log(n)/\varepsilon^3$, so $T = \Omega(\log(n)/\varepsilon^3)$ as claimed. Thus, for the remainder of the proof we may assume

$$\varepsilon < \min\left\{c', \frac{1}{256}\right\}. \tag{27}$$

Let F_{emp} denote the empirical distribution of the T samples $x_1, \ldots, x_T$. Under distribution p, these samples are i.i.d. draws from D_θ, so by the Dvoketzky-Kiefer-Wolfowitz Inequality (Dvoretzky et al., 1956)

$$p\left(\|F_{\text{emp}} - F_\theta\|_\infty \geq \frac{\varepsilon}{2}\right) \leq 2e^{-T\varepsilon^2/2}. \tag{28}$$

Substituting the lower bound for T in (26) and the upper bound for ε in (27) we find that

$$2e^{-T\varepsilon^2/2} \leq 2e^{-c' \ln(1/\varepsilon)/(2\varepsilon)} \leq 2e^{-\ln(1/\varepsilon)/2} \leq 2e^{-\ln(256)/2} = \frac{1}{8}. \tag{29}$$

If $\hat{F}$ denotes the CDF estimate produced by our algorithm, then the algorithm's probabilistic approximate correctness guarantee asserts that

$$p\left(\|\hat{F} - F_{\text{emp}}\|_\infty > \varepsilon\right) \leq \frac{1}{4}. \tag{30}$$

Combining inequalities (27),(28),(29) and using the union bound and triangle inequality, we have

$$\begin{aligned} p\left(\|F_{\text{emp}} - F_\theta\|_\infty \geq \frac{\varepsilon}{2} \text{ or } \|\hat{F} - F_{\text{emp}}\|_\infty > \varepsilon\right) &\leq \frac{3}{8} \\ p\left(\|F_{\text{emp}} - F_\theta\|_\infty < \frac{\varepsilon}{2} \text{ and } \|\hat{F} - F_{\text{emp}}\|_\infty \leq \varepsilon\right) &\geq \frac{5}{8} \\ p\left(\|\hat{F} - F_\theta\|_\infty < \frac{3\varepsilon}{2}\right) &\geq \frac{5}{8}. \end{aligned}$$

Now let us define two random variables $\hat{\theta}$, θ_{MLE} as follows: $\hat{\theta}$ is the value of θ whose associated CDF $F_{\hat{\theta}}$ is nearest to F_{emp} in L_∞, while θ_{MLE} is the value of θ whose conditional probability $p(\theta = \theta_{\text{MLE}} \mid H)$ is greatest. From Lemma A.1 we know that $\theta = \hat{\theta}$ whenever $\|\hat{F} - F_\theta\|_\infty < \frac{3\varepsilon}{2}$, so $p(\theta = \hat{\theta}) \geq \frac{5}{8}$. By the definition of θ_{MLE}, we have

$$p(\theta = \theta_{\text{MLE}}|H) \geq p(\theta = \hat{\theta}|H) \tag{31}$$

for all H. Take the expectation of both sides of (31) with respect to H and use the law of iterated expectation to deduce

$$p(\theta = \theta_{\text{MLE}}) \geq p(\theta = \hat{\theta}) = \frac{5}{8}. \tag{32}$$

Let us define a *good history* to be a history H such that $p(\theta = \theta_{\text{MLE}}|H) \geq \frac{1}{2}$. We have

$$\frac{5}{8} \leq p(\theta = \theta_{\text{MLE}}) \leq p(H \text{ is good}) \cdot 1 + p(H \text{ is not good}) \cdot \frac{1}{2} = \frac{1}{2} + \frac{1}{2}p(H \text{ is good}), \tag{33}$$

so $p(H \text{ is good}) \geq \frac{1}{4}$.

When the history H is good, it means that $p(\theta = \theta_{\text{MLE}}|H) \geq \frac{1}{2}$ whereas $q(\theta|H)$ is the uniform distribution over $[m]^k$ so $q(\theta_i = (\theta_{\text{MLE}})_i|H) = \frac{1}{m}$ for all $i \in [k]$. The conditions for Lemma A.12 are satisfied, so we may

conclude that the inequalities $D_{KL}(p(\theta|H)\,\|\,q(\theta|H)) \geq \frac{k}{2}\ln\left(\frac{m}{4}\right)$ and $D_{KL}(q(\theta|H)\,\|\,p(\theta|H)) \geq \frac{k}{2}\ln\left(\frac{k}{3}\right)$ hold when H is good. Since the probability that H is good is at least $\frac{1}{4}$, we have

$$\mathbb{E}_H D_{KL}(p(\theta|H)\,\|\,q(\theta|H)) \geq \frac{k}{8}\ln\left(\frac{m}{4}\right) \tag{34}$$

$$\mathbb{E}_H D_{KL}(q(\theta|H)\,\|\,p(\theta|H)) \geq \frac{k}{8}\ln\left(\frac{k}{3}\right). \tag{35}$$

Using Lemma A.13 together with Equations (16) and (17), we find that

$$48T\varepsilon^2 \geq \mathbb{E}_\theta D_{KL}(p(H|\theta)\,\|\,q(H|\theta)) = \mathbb{E}_H D_{KL}(p(\theta|H)\,\|\,q(\theta|H)) \geq \frac{k}{8}\ln\left(\frac{m}{4}\right) \tag{36}$$

$$48T\varepsilon^2 \geq \mathbb{E}_\theta D_{KL}(q(H|\theta)\,\|\,p(H|\theta)) = \mathbb{E}_H D_{KL}(q(\theta|H)\,\|\,p(\theta|H)) \geq \frac{k}{8}\ln\left(\frac{k}{3}\right). \tag{37}$$

Summing inequalities (36) and (37), we obtain

$$96T\varepsilon^2 \geq \frac{k}{8}\ln\left(\frac{km}{12}\right). \tag{38}$$

Recalling that $k = \frac{1}{6\varepsilon}$ and that $km = n$, we find that $T > \frac{1}{4800\varepsilon^3}\ln\left(\frac{n}{12}\right)$, which concludes the proof that $T = \Omega(\log(n)/\varepsilon^3)$. ∎

B Proof of Theorem 5.1

Proof. In the proof, we will use $\mathbb{E}_t[\cdots]$ as a notation for the conditional expectation of random variables, conditioning on the history of the first $t-1$ rounds and on the adversary's choice of v_t. We make a similar notation change for $\mathrm{Var}[\cdots]$. We note that $\mathbb{E}_t[\hat{u}_t] = u_t$ and $\mathbb{E}_t[\hat{\ell}_t] = \ell_t$. This follows from the fact that $u_t(i) = k \cdot u_t(i)$ with probability $\frac{1}{k}$ and 0 otherwise. The argument follows similarly for ℓ_t. Thus, $\mathbb{E}_t[\hat{d}_t] = d_t$. Intuitively, Algorithm 2 works because even though we don't observe d_t, we can still obtain an unbiased estimate of d_t.

The weights $w_{i,t}$ and coefficients $a_{i,t}$ in Algorithm 1 evolve according to the update equations of the standard EXP3 algorithm of Auer et al. (2002). By the analysis of that algorithm, [rdk: Would be nice to reference an actual theorem from that paper here, so that readers could easily find the bound we're using from the Exp3 paper.]

$$\sum_{t=1}^{T}\sum_{i=1}^{n} a_i \hat{d}_t(i) \geq \max_{i\in[n]}\left\{\sum_{t=1}^{T}\hat{d}_t(i)\right\} - \frac{\ln n}{\eta} - \eta\sum_{t=1}^{T}\sum_{j=1}^{n} a_i \hat{d}_t^2(i). \tag{39}$$

[rdk: In the foregoing displayed equation, and almost every other one on this page, I revised the size of curly braces from $\{\sum_{t=1}^{T}\cdots\}$ to $\left\{\sum_{t=1}^{t}\cdots\right\}$.] Taking expectations, we get that

$$\sum_{t=1}^{T}\sum_{i=1}^{n} a_i d_t(i) \geq \max_{i\in[n]}\left\{\sum_{t=1}^{T} d_t(i)\right\} - \frac{\ln n}{\eta} - \eta\sum_{t=1}^{T}\sum_{j=1}^{n} a_i \mathbb{E}[\hat{d}_t^2(i)]. \tag{40}$$

To bound the last term on the right side, we make use of the law of iterated expectation: $\mathbb{E}[\hat{d}_t^2(i)] = \mathbb{E}[\mathbb{E}_t[\hat{d}_t^2(i)]]$. Now, if $i = q_{j,t}$ for some j then $\hat{u}_t(i) = \hat{\ell}_t(i)$ so $\hat{d}_t(i) = 0$. On the other hand, if $q_{j,t} < i < q_{j+1,t}$

then $\hat{d}_t(i) = k \cdot u_t(i)$ with probability $1/k$, $\hat{d}_t(i) = -k \cdot \ell_t(i)$ with probability $1/k$, and otherwise $\hat{d}_t(i) = 0$. Thus,

$$\mathbb{E}_t[\hat{d}_t^2(i)] = \frac{1}{k}\left(k^2 u_t^2(i) + k^2 \ell_t^2(i)\right) \le \frac{1}{k}\left(k^2 + k^2\right) = 2k. \tag{41}$$

Applying the law of iterated expectation, we have $\mathbb{E}[\hat{d}_t^2(i)] \le 2k$. This, together with the fact that $\sum_{j=1}^n a_i \le 1$, allows us to simplify the equation to

$$\sum_{t=1}^T \sum_{i=1}^n a_i d_t(i) \ge \max_{i\in[n]} \left\{ \sum_{t=1}^T d_t(i) \right\} - \frac{\ln n}{\eta} - 2\eta T k \tag{42}$$

From the proof of Proposition 3.1 we know that for all t, $\sum_{i=1}^n a_i d_t(i) \le \frac{1}{k+1}$. Substituting this bound

$$\frac{T}{k+1} \ge \max_{i\in[n]} \left\{ \sum_{t=1}^T d_t(i) \right\} - \frac{\ln n}{\eta} - 2\eta k T \tag{43}$$

$$\frac{T}{k+1} + \frac{\ln n}{\eta} + 2\eta k T \ge \max_{i\in[n]} \left\{ \sum_{t=1}^T d_t(i) \right\} \tag{44}$$

Recalling that $k = \frac{2}{\varepsilon}$, $\eta = \frac{\varepsilon^2}{16} = \frac{\varepsilon}{8k}$, $T \ge \frac{64\ln(n)}{\varepsilon^3}$, we find that $\frac{T}{k+1} \le \frac{\varepsilon T}{2}$, $\frac{\ln n}{\eta} \le \frac{\varepsilon T}{4}$, $2\eta k T \le \frac{\varepsilon T}{4}$. Hence

$$\frac{\varepsilon}{2}T + \frac{\varepsilon}{4}T + \frac{\varepsilon}{4}T \ge \max_{i\in[n]} \left\{ \sum_{t=1}^T d_t(i) \right\}$$

$$\varepsilon \ge \max_{i\in[n]} \left\{ \frac{1}{T} \sum_{t=1}^T d_t(i) \right\}.$$

The right side of the last inequality is equal to the width of the interval $\left[\frac{1}{T}\sum_{t=1}^T \ell_t(i),\ \frac{1}{T}\sum_{t=1}^T u_t(i)\right]$. That interval is guaranteed to contain $\frac{1}{T}\sum_{t=1}^T v_t(i)$, and its midpoint is $G_T(i)$, so we are assured that $|G_T(i) - \frac{1}{T}\sum_{t=1}^T v_t(i)| \le \frac{\varepsilon}{2}$. However, because the algorithm only observes estimates $\hat{u}_t$ and $\hat{\ell}_t$, it doesn't know the actual values of $\frac{1}{T}\sum_{t=1}^T \ell_t(i)$ and $\frac{1}{T}\sum_{t=1}^T u_t(i)$. We will now rely on concentration results to show that after $T \ge \frac{64\ln n}{\varepsilon^3}$, with high probability, the estimates $\frac{1}{T}\sum_{t=1}^T \hat{\ell}_t(i)$ and $\frac{1}{T}\sum_{t=1}^T \hat{u}_t(i)$ will be close to the true values $\frac{1}{T}\sum_{t=1}^T \ell_t(i)$ and $\frac{1}{T}\sum_{t=1}^T u_t(i)$. Since the estimates, $\hat{\ell}_t$ and $\hat{u}_t$, are correct in expectation and $\mathbb{E}[\hat{\ell}_t^2(i)] = k\ell_t^2(i)$ (also $\mathbb{E}[\hat{u}_t^2(i)] = k u_t^2(i)$), we have

$$\mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^T \hat{\ell}_t(i)\right) = \frac{1}{T^2}\sum_{t=1}^T \mathrm{Var}\left(\hat{\ell}_t(i)\right) = \frac{1}{T^2}\sum_{t=1}^T (k-1)\ell_t^2(i) \le \frac{k}{T} \tag{45}$$

for each $i \in [n]$. Using the martingale variant of Bernstein's inequality proved in Freedman (1975), we have

$$\mathrm{P}\left[\left|\frac{1}{T}\sum_{t=1}^T \hat{\ell}_t(i) - \frac{1}{T}\sum_{t=1}^T \ell_t(i)\right| \ge \frac{\varepsilon}{2}\right] \le 2\exp\left(-\frac{T^2(\varepsilon/2)^2}{2(k-1)(\frac{1}{T}\sum_{t=1}^T \ell_t(i)) + \frac{2}{3}kT(\varepsilon/2)}\right) \le 2\exp\left(-\frac{T(\varepsilon/2)^2}{3k}\right) \tag{46}$$

which is less than $\frac{1}{4n}$ since $T \ge \frac{64\ln n}{\varepsilon^3}$.

Following similar analysis for $u_t(i)$, we get that $\mathrm{P}\left[\left|\frac{1}{T}\sum_{t=1}^T \hat{u}_t(i) - \frac{1}{T}\sum_{t=1}^T u_t(i)\right| \ge \frac{\varepsilon}{2}\right]$ is less than $\frac{1}{4n}$. As a result, with high probability, the interval $\left[\frac{1}{T}\sum_{t=1}^T \hat{\ell}_t(i), \frac{1}{T}\sum_{t=1}^T \hat{u}_t(i)\right]$ contains $\frac{1}{T}\sum_{t=1}^T v_t(i)$ and is $\frac{\varepsilon}{2}$ close to the midpoint of the true interval $\left[\frac{1}{T}\sum_{t=1}^T \ell_t(i), \frac{1}{T}\sum_{t=1}^T u_t(i)\right]$. Thus, with high probability,

$$\left|\hat{G}_T(i) - \frac{1}{T}\sum_{t=1}^T v_t(i)\right| \le \left|\hat{G}_T(i) - G_T(i)\right| + \left|G_T(i) - \frac{1}{T}\sum_{t=1}^T v_t(i)\right| \le \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon \text{ as desired.} \tag{47}$$

■

C Elementary Algorithms for Threshold Query Model

Lemma C.1. No deterministic algorithm can obtain accuracy ε with a query budget less than $\frac{1}{2\varepsilon} - 1$

Proof. Suppose for sake of contradiction that an algorithm $\mathcal{A}$ obtains accuracy ε with a query budget of $k = \frac{1}{2\varepsilon} - 2$. Suppose $n \geq k+1$. After T timesteps, there exists a point, p, that has not been queried for at least $\frac{T}{k+1}$. If this was not the case, then all the points would have been queried for $n \cdot \frac{kT}{k+1} \geq (k+1) \cdot \frac{kT}{k+1} \geq kT$ (a contradiction). Let $G_T(p)$ be the algorithm's estimate at that point at time T and let $F_T(p)$ be the average during timesteps when p was queried. Since the algorithm doesn't know whether the function's value at p was 0 or 1 during the $\frac{T}{k+1}$ timesteps, it follows that

$$|G(p) - F_T(p)| > \frac{1}{2}\left|\left(F_T(p) + \frac{1}{k+1}\right) - \left(F_T(p) + \frac{0}{k+1}\right)\right| > \frac{1}{2}\left(\frac{1}{k+1}\right) > \varepsilon \tag{48}$$

■

Lemma C.2. The pair $\left(k = (\frac{1}{\epsilon} + 1)\sqrt{n}, T_0 = \frac{1}{\epsilon}\right)$ is achievable for monotone step functions. The algorithm behaves as follows: the algorithm always queries points $\{0, \lfloor\sqrt{n}\rfloor, 2\lfloor\sqrt{n}\rfloor, \ldots, n\}$ every timestep. The algorithm maintains the average F of the step functions at these points. For any interval, $[m\sqrt{n}, (m+1)\sqrt{n}]$ such that $F[(m+1)\sqrt{n}] - F[m\sqrt{n}] \geq \epsilon$, the algorithm queries every point in the interval.

Proof. This claim follows by observing that from timesteps 1 till $\frac{1}{\epsilon}$, for every interval $(m\sqrt{n}, (m+1)\sqrt{n})$, there is at most 1 timestep where we didn't know the value of every point in the interval. ■

Lemma C.3. The pair $\left(k = (\frac{4}{\epsilon} + 1)\sqrt{n}, T_0 = \frac{4}{\epsilon}\right)$ is robustly-achievable [5] for any monotone function. The algorithm behaves as follows: the algorithm always queries points $\{0, \lfloor\sqrt{n}\rfloor, 2\lfloor\sqrt{n}\rfloor, \ldots, n\}$ on every timestep. The algorithm maintains the average $\hat{F}$ of the step functions at these points. For any interval, $[m\sqrt{n}, (m+1)\sqrt{n}]$ such that $\hat{F}[(m+1)\sqrt{n}] - \hat{F}[m\sqrt{n}] \geq \epsilon/4$, the algorithm queries every point in the interval.

Proof. We prove this by induction on T. We will show that after each timestep t, $|\hat{F} - \frac{1}{t}\sum v_t|_\infty \leq \frac{1}{t} + \frac{3\epsilon}{4}$. Assume the statement is true after the t-th timestep. At timestep $t+1$, every interval $(m\sqrt{n}, (m+1)\sqrt{n})$ such that $\hat{F}[(m+1)\sqrt{n}] - \hat{F}[m\sqrt{n}] \geq \epsilon/4$ gets queried by the algorithm. Note that by IH, no interval has a gap of more than $\frac{1}{t+1} + \frac{3\epsilon}{4}$. Thus, after the $t+1$ timestep, for any point p in a tracked interval (i.e one such that $\hat{F}[(m+1)\sqrt{n}] - \hat{F}[m\sqrt{n}] \geq \epsilon/4$),

$$\begin{aligned}\left|\hat{F}^{t+1}(p) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p)\right| &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) + y_p\right) - \frac{1}{t+1}\left(\sum_{s=0}^{t} v_s(p) + v_{t+1}(p)\right)\right| \\ &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) - \sum_{s=0}^{t} v_s(p)\right) + \frac{1}{t+1}\left(y_p - v_{t+1}(p)\right)\right|\end{aligned}$$

by inductive hypothesis

$$< \left|\frac{t}{t+1}\left(\frac{1}{t} + \frac{3\epsilon}{4}\right)\right| + \left|\frac{1}{t+1}\left(y_p - v_{t+1}(p)\right)\right|$$

[5] In our original model, we assumed the algorithm observed y_p such that $|y_p - F_{t+1}(p)| \leq \epsilon/4$ instead of directly receiving $F_{t+1}(p)$. This was done to make simulation easier but turns out to be unnecessary for the final algorithm.

by guarantee $|y_p - v_{t+1}(p)| \le \epsilon/4$ [vg: just wanted to double check that it is correct to change this to lowercase f]

$$\begin{aligned} &< \frac{1}{t+1} + \frac{t(3\epsilon/4)}{t+1} + \frac{\epsilon/4}{t+1} \\ &< \frac{1}{t+1} + \frac{3\epsilon}{4} \end{aligned}$$

For an untracked interval $[p_1, p_2]$, that is, one such that $\hat{F}[p_2] - \hat{F}[p_1] < \epsilon/4$, we have that after the $t+1$ timestep,

$$\begin{aligned} \left|\hat{F}^{t+1}(p_2) - \hat{F}^{t+1}(p_1)\right| &\le \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p_2) + y_{p_2}\right) - \frac{1}{t+1}\left(t \cdot \hat{F}^t(p_1) + y_{p_1}\right)\right| \\ &\le \left|\frac{t}{t+1}\left(\hat{F}^t(p_2) - \hat{F}^t(p_1)\right) + \frac{1}{t+1}\left(y_{p_2} - y_{p_1}\right)\right| \end{aligned}$$

since the interval is untracked

$$\begin{aligned} &< \left|\frac{t}{t+1}(\epsilon/4) + \frac{1}{t+1}\right| \\ &< \frac{1}{t+1} + \frac{\epsilon}{4} \end{aligned}$$

Since points p_1 and p_2 have been tracked from the start, it follows that $\left|\hat{F}^{t+1}(p_1) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_1)\right| \le \epsilon/4$ and $\left|\hat{F}^{t+1}(p_2) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_2)\right| \le \epsilon/4$. The $\epsilon/4$ comes from the noise assumption i.e algorithm observes y such that $|y - v_t| \le \epsilon/2$ but does not observe v_t exactly. By monotonicity, we know that $\frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_1) \le \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p) \le \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_2)$. Thus, we have that for a point p in an untracked interval $[p_1, p_2]$

$$\begin{aligned} \left|\hat{F}^{t+1}(p) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p)\right| &\le \left|\hat{F}^{t+1}(p_1) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_1)\right| + \left|\hat{F}^{t+1}(p_2) - \hat{F}^{t+1}(p_1)\right| \\ &\quad + \left|\hat{F}^{t+1}(p_2) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p_2)\right| \\ &< \epsilon/4 + \epsilon/4 + \frac{1}{t+1} + \epsilon/4 \\ &< \frac{1}{t+1} + \frac{3\epsilon}{4} \end{aligned}$$

Thus, after $T \ge \frac{4}{\epsilon}$, it follows that $|\hat{F} - \frac{1}{T}\sum v_T|_\infty \le \frac{1}{T} + \frac{3\epsilon}{4} \le \epsilon$. ∎

Lemma C.4. The pair $\left(k = \frac{\log n}{\epsilon^2}, T_0 = \frac{\log n}{\epsilon}\right)$ is achievable for monotone step functions. The algorithm behaves as follows: insert a new query point at the midpoint of the interval with the uncertainty of the current timestep. That is, suppose your query points were $q_1, q_2, \ldots, q_k$. There's an interval (q_i, q_{i+1}) where $v_t[q_i] = 0$ and $v_t[q_{i+1}] = 1$. Insert the new query point at $q = \lfloor (q_i + q_{i+1})/2 \rfloor$

Proof. We prove this by induction on T. We will show that after every timestep t, for any point $p \in [p_1, p_2]$ where p_1, p_2 are adjacent active points, then $|\hat{F}[p] - \frac{1}{t}\sum v_t[p]| \le \frac{\log n - \log(p_2 - p_1)}{2t}$. Assume the statement is true after the t-th timestep. At timestep $t+1$, the adversary chooses a point r to make the step for v_{t+1}. Let

$[r_1, r_2]$ be the adjacent active points that r falls in. Since the algorithm will observe that $v_{t+1}[r_1] = 0$ and $v_{t+1}[r_2] = 1$, for any point p outside this interval, it follows that

$$\begin{aligned}\left|\hat{F}^{t+1}(p) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p)\right| &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) + v_{t+1}(p)\right) - \frac{1}{t+1}\left(\sum_{s=0}^{t} v_s(p) + v_{t+1}(p)\right)\right| \\ &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) - \sum_{s=0}^{t} v_s(p)\right)\right|\end{aligned}$$

by inductive hypothesis

$$\leq \frac{\log n - \log(p_2 - p_1)}{2(t+1)}$$

Recall that the algorithm inserts a new point r_3 at the midpoint of r_1, r_2. For a point p in the interval $[r_1, r_2]$,

$$\begin{aligned}\left|\hat{F}^{t+1}(p) - \frac{1}{t+1}\sum_{s=0}^{t+1} v_s(p)\right| &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) + \frac{1}{2}\right) - \frac{1}{t+1}\left(\sum_{s=0}^{t} v_s(p) + v_{t+1}(p)\right)\right| \\ &\leq \left|\frac{1}{t+1}\left(t \cdot \hat{F}^t(p) - \sum_{s=0}^{t} v_s(p)\right) + \frac{1}{t+1}\left(\frac{1}{2} - v_{t+1}(p)\right)\right|\end{aligned}$$

by inductive hypothesis

$$\begin{aligned}&= \frac{\log n - \log(r_2 - r_1)}{2(t+1)} + \frac{1}{2(t+1)} \\ &= \frac{\log n - \log(r_2 - r_1) - \log 2}{2(t+1)} \\ &= \frac{\log n - \log(r_2 - r_3)}{2(t+1)}\end{aligned}$$

■